

**DIGITALISASI *MONITORING* INFORMASI DATA PRODUKSI  
BIBIT TANAMAN KARET BERBASIS *ANDROID***

**SKRIPSI**



**DIMAS KURNIA RAMADHAN**

**2022230026**

**PROGRAM STUDI INFORMATIKA  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS PRABUMULIH  
TAHUN 2026**

**DIGITALISASI *MONITORING* INFORMASI DATA PRODUKSI  
BIBIT TANAMAN KARET BERBASIS *ANDROID***

Untuk Memenuhi Salah Satu Syarat Menyusun Skripsi  
Program Studi Informatika Universitas Prabumulih



**DIMAS KURNIA RAMADHAN**

**2022230026**

**FAKULTAS ILMU KOMPUTER  
PROGRAM STUDI INFORMATIKA  
UNIVERSITAS PRABUMULIH  
2026**

## HALAMAN PENGESAHAN

# DIGITALISASI *MONITORING* INFORMASI DATA PRODUKSI BIBIT TANAMAN KARET BERBASIS *ANDROID*

## SKRIPSI

Diajukan Sebagai Salah Satu Syarat Untuk Mendapat Gelar Sarjana  
Program Studi Informatika Universitas Prabumulih

Oleh :

Dimas Kurnia Ramadhan  
2022230026

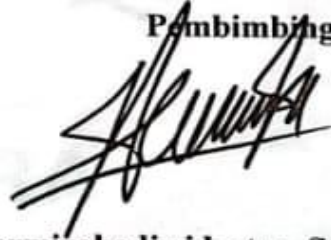
Prabumulih, 2 Juni 2026

Pembimbing I



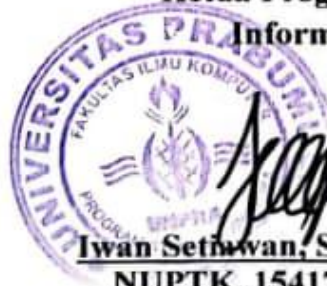
Iwan Setiawan, S.Kom., M.Kom.  
NUPTK. 1541769670130303

Pembimbing II



Hermizahadiwidastra, S.Kom., M.Kom.  
NUPTK. 2863762663130192

Ketua Program Studi  
Informatika



Iwan Setiawan, S.Kom., M.Kom.  
NUPTK. 1541769670130303

## HALAMAN PERSETUJUAN

Skripsi ini dengan judul “Digitalisasi *Monitoring* Informasi Data Produksi Bibit Tanaman Karet Berbasis *Android*” telah dipertahankan dan disetujui dihadapan Tim Penguji Skripsi Program Studi Informatika Fakultas Ilmu Komputer Universitas Prabumulih pada tanggal 2 Juni 2026.

**Prabumulih, 2 Juni 2026**

**Tim Penguji Tugas Akhir**

**Ketua:**

Iwan Setiawan, S.Kom., M.Kom  
NUPTK. 1541769670130303

(  )

**Anggota:**

1. Hermizahadiwadstra, S.Kom., M.Kom.  
NUPTK. 2863762663130192

(  )

2. Suhartini, S.Kom., M.Kom.  
NUPTK. 253475765823105

(  )

**Mengetahui,  
Dekan Fakultas Ilmu Komputer**

**Ketua Program Studi Informatika**

  
Khana Wijaya, S.Kom., M.Kom.  
NUPTK. 7146768669130363

  
Iwan Setiawan, S.Kom., M.Kom.  
NUPTK. 1541769670130303

## PENYATAAN INTEGRITAS

Saya yang bertanda tangan di bawah ini

Nama : Dimas Kurnia Ramadhan

NIM : 2022230026

Program Studi : Informatika

Menyatakan dengan sebenarnya bahwa, Skripsi ini benar-benar merupakan hasil karya saya dan tidak terdapat karya orang lain, apabila di kemudian hari terbukti atau dapat dibuktikan bahwa Skripsi ini hasil *plagiasi*, maka saya bersedia menerima sanksi sesuai dengan ketentuan yang berlaku atas perbuatan tersebut.



Prabumulih, 2 Juni 2026

Yang membuat pernyataan,



Dimas Kurnia Ramadhan

NIM. 2022230026

# **Digitalisasi *Monitoring* Informasi Data Produksi Bibit Tanaman Karet Berbasis *Android***

**Dimas Kurnia Ramadhan, 2022230026, 2026**

Digitalisasi *monitoring* informasi data produksi bibit tanaman karet merupakan kegiatan penting dalam mendukung efektivitas proses budidaya dan pengelolaan perkebunan. Namun, proses pencatatan yang masih dilakukan secara manual di PT. Kendi Arindo sering menimbulkan permasalahan seperti keterlambatan pelaporan, ketidaktepatan data, serta kesulitan dalam pengarsipan dan pengecekan ulang informasi produksi. Penelitian ini bertujuan untuk mengembangkan aplikasi digitalisasi *monitoring* informasi data produksi berbasis *android* sebagai solusi untuk meningkatkan efisiensi, akurasi, dan keteraturan proses pencatatan di lapangan. Aplikasi yang dirancang memungkinkan petugas untuk mencatat, mengelola, serta memantau informasi produksi bibit tanaman karet secara lebih cepat dan terstruktur melalui perangkat *android*. Hasil penelitian menunjukkan bahwa sistem yang dibangun mampu mempercepat alur kerja, meminimalkan kesalahan input, serta mendukung proses monitoring secara lebih efektif. Dengan demikian, aplikasi ini berpotensi menjadi sarana pendukung penting dalam transformasi digital di PT. Kendi Arindo.

**Kata Kunci:** Digitalisasi, *Monitoring* Produksi, Bibit Tanaman Karet, *Android*, Informasi Produksi.

## ***ABSTRACT***

### ***Android-Based Digitalization of Rubber Plant Seedling Production Data Information Monitoring***

**Dimas Kurnia Ramadhan, 2022230026, 2026**

*Digitalization monitoring production information for rubber plant seedlings is an essential activity in supporting the effectiveness of cultivation and plantation management. However, the manual recording process used at PT. Kendi Arindo often leads to issues such as delayed reporting, inaccurate data, and difficulties in archiving and reviewing production information. This study aims to develop an Android-based digital monitoring application as a solution to improve efficiency, accuracy, and organization in field data recording. The application enables field staff to record, manage, and monitor production information more quickly and systematically using an Android device. The results of this study indicate that the developed system can accelerate workflow, minimize input errors, and support more effective monitoring processes. Therefore, the application has the potential to become an important tool in supporting the digital transformation at PT. Kendi Arindo.*

**Keywords:** *Digitalization, Production Monitoring, Rubber Seedlings, Android, Production Information.*

## KATA PENGANTAR

Puji dan syukur penulis ucapkan kehadirat Allah SWT atas limpahan rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan Skripsi yang berjudul “Digitalisasi *Monitoring* Informasi Data Produksi Bibit Tanaman Karet Berbasis *Android*”, Skripsi ini merupakan salah satu syarat untuk mendapatkan gelar Sarjana di Program Studi S1 Informatika Fakultas Ilmu Komputer Universitas Prabumulih. Penulis banyak menerima bantuan, arahan, dan bimbingan dari berbagai pihak. Penulis mengucapkan terima kasih kepada:

1. Ibu Dr. Yuniar Pratiwi, S.Si., M.Si selaku Rektor Universitas Prabumulih
2. Bapak Andi Christian. S.Kom., M.Kom selaku Wakil Rektor Universitas Prabumulih
3. Bapak Khana Wijaya, S.Kom., M.Kom selaku Dekan Fakultas Universitas Prabumulih.
4. Ibu Suhartini, S.Kom., M.Kom selaku Wakil Dekan Bidang Keuangan Fakultas Ilmu Komputer Universitas Prabumulih
5. Ibu Nur Aini Hutagalung, S.Kom., M.Si., M.Kom Selaku Wakil Dekan Bidang Kemahasiswaan Fakultas Ilmu Komputer Universitas Prabumulih
6. Bapak Iwan Setiawan, S.Kom., M.Kom selaku Ketua Program Studi Informatika sekaligus sebagai Pembimbing I.
7. Bapak Hermizahadiwidastra, S.Kom., M.Kom sebagai pembimbing II.
8. Bapak dan Ibu Dosen Program Studi Informatika. Fakultas Ilmu Komputer Universitas Prabumulih.
9. Bapak/Ibu staf administrasi Fakultas Ilmu Komputer Universitas Prabumulih.
10. Kepala Perusahaan, Pimpinan, dan seluruh Staf Perusahaan PT. Kendi Arindo yang telah memberikan izin dan membantu untuk melakukan penelitian.

Penulisan Skripsi ini masih terdapat kekurangan sehingga membutuhkan kritik dan saran yang bersifat membangun untuk kesempurnaan Skripsi.

Prabumulih, 2 Juni 2026



Dimas Kurnia Ramadhan  
2022230026

## DAFTAR ISI

|                                                    | Halaman  |
|----------------------------------------------------|----------|
| HALAMAN COVER DALAM .....                          | i        |
| HALAMAN PENGESAHAN .....                           | ii       |
| HALAMAN PERSETUJUAN .....                          | iii      |
| PENYATAAN INTEGRITAS .....                         | iv       |
| ABSTRAK .....                                      | v        |
| ABSTRACT .....                                     | vi       |
| KATA PENGANTAR .....                               | vii      |
| DAFTAR ISI .....                                   | viii     |
| DAFTAR TABEL .....                                 | xii      |
| DAFTAR GAMBAR .....                                | xiii     |
| DAFTAR LAMPIRAN .....                              | xvii     |
| DAFTAR SINGKATAN .....                             | xviii    |
| <b>BAB I PENDAHULUAN .....</b>                     | <b>1</b> |
| 1.1 Latar Belakang .....                           | 1        |
| 1.2 Rumusan Masalah .....                          | 3        |
| 1.3 Batasan Masalah .....                          | 4        |
| 1.4 Tujuan Penelitian .....                        | 4        |
| 1.5 Manfaat Penelitian .....                       | 5        |
| 1.6 Sistematika Penulisan .....                    | 6        |
| <b>BAB II TINJAUAN PUSTAKA .....</b>               | <b>8</b> |
| 2.1 Pengertian Digitalisasi .....                  | 8        |
| 2.2 Pengertian Monitoring .....                    | 8        |
| 2.3 Pengertian Manajemen Informasi .....           | 9        |
| 2.4 Pengertian <i>Android</i> .....                | 10       |
| 2.5 Pengertian Produksi .....                      | 11       |
| 2.6 Pengertian Tanaman .....                       | 12       |
| 2.7 Pengertian Virtual Studio Code (VS Code) ..... | 12       |
| 2.8 Pengertian Flutter .....                       | 13       |
| 2.9 Pengertian Dart Programing Language .....      | 14       |
| 2.10 Pengertian <i>Firebase</i> .....              | 15       |

|                                                         |            |
|---------------------------------------------------------|------------|
| 2.11 Penelitian Terdahulu.....                          | 16         |
| 2.12 Kerangka Berpikir Penelitian.....                  | 17         |
| <b>BAB III METODE PENELITIAN .....</b>                  | <b>20</b>  |
| 3.1 Subjek dan Objek Penelitian .....                   | 20         |
| 3.2 Metode Penelitian.....                              | 26         |
| 3.3 Sumber Data.....                                    | 27         |
| 3.4 Teknik Pengumpulan Data .....                       | 28         |
| 3.5 Metode Pengembangan Sistem .....                    | 31         |
| 3.6 Alat Bantu Perancangan.....                         | 35         |
| 3.7 Pengujian Sistem.....                               | 42         |
| <b>BAB IV ANALISA DAN PERANCANGAN.....</b>              | <b>45</b>  |
| 4.1. Analisa Sistem.....                                | 45         |
| 4.2 Perancangan Sistem (Desain) yang Diusulkan .....    | 53         |
| <b>BAB V HASIL DAN PEMBAHASAN .....</b>                 | <b>112</b> |
| 5.1 Spesifikasi Perangkat.....                          | 112        |
| 5.1.1. Perangkat Keras Yang Digunakan.....              | 112        |
| 5.1.2. Perangkat Lunak yang Digunakan .....             | 112        |
| 5.2 Implementasi <i>Database</i> .....                  | 113        |
| 5.2.1. Implementasi Tabel Karyawan .....                | 114        |
| 5.2.2. Implementasi Tabel Kepala Bagian .....           | 115        |
| 5.2.3. Implementasi Tabel Pimpinan .....                | 116        |
| 5.2.4. Implementasi Tabel Produksi Bibit .....          | 117        |
| 5.2.5 Implementasi Tabel Produksi Tanaman .....         | 118        |
| 5.3 Tampilan Antar Muka .....                           | 120        |
| 5.3.1. <i>Splash Screen</i> .....                       | 120        |
| 5.3.2. Halaman <i>Login</i> .....                       | 121        |
| 5.3.3. Halaman <i>Dashboard</i> Karyawan Lapangan ..... | 122        |
| 5.3.4. Halaman Data Produksi Bibit.....                 | 123        |
| 5.3.5. Halaman Detail Produksi Bibit .....              | 124        |
| 5.3.6. Halaman <i>Input</i> Data Produksi Tanaman.....  | 125        |
| 5.3.7. Halaman Detail Produksi Tanaman.....             | 126        |
| 5.3.8. Halaman <i>Input</i> Data Produksi Bibit .....   | 127        |
| 5.3.9. Halaman <i>Input</i> Data Produksi Tanaman.....  | 128        |

|                                                          |            |
|----------------------------------------------------------|------------|
| 5.3.10 Halaman Laporan .....                             | 129        |
| 5.3.11 Halaman Notifikasi .....                          | 130        |
| 5.3.12 Halaman Profil Karyawan Lapangan .....            | 131        |
| 5.3.13 Halaman <i>Dashboard</i> Kepala Bagian .....      | 132        |
| 5.3.14 Halaman Data Produksi Bibit .....                 | 133        |
| 5.3.15 Halaman Detail Produksi Bibit .....               | 134        |
| 5.3.16 Halaman Data Produksi Tanaman .....               | 135        |
| 5.3.17 Halaman Detail Produksi Tanaman .....             | 136        |
| 5.3.18 Halaman Data Validasi .....                       | 137        |
| 5.3.19 Halaman <i>Monitoring</i> Produksi .....          | 138        |
| 5.3.20 Halaman Laporan .....                             | 139        |
| 5.3.21. Halaman Manajemen Pengguna .....                 | 140        |
| 5.3.22. Halaman Profil Kepala Bagian .....               | 142        |
| 5.3.23. Halaman <i>Dashboard</i> Pimpinan .....          | 143        |
| 5.3.24. Halaman <i>Monitoring</i> Produksi .....         | 144        |
| 5.3.25. Halaman Laporan .....                            | 145        |
| 5.3.26. Halaman Notifikasi .....                         | 146        |
| 5.3.27. Halaman Profil Pimpinan .....                    | 147        |
| 5.4 Hasil Pengujian Sistem .....                         | 148        |
| 5.4.1. Pengujian Fungsi <i>Login</i> .....               | 148        |
| 5.4.2. Pengujian Fungsi <i>Logout</i> .....              | 158        |
| 5.4.3. Pengujian Fungsi Simpan Data Bibit .....          | 162        |
| 5.4.4. Pengujian Fungsi Simpan Data Tanaman .....        | 171        |
| 5.4.5. Pengujian Fungsi Validasi Data .....              | 181        |
| 5.4.6. Pengujian Fungsi Realisasi Tahapan .....          | 190        |
| 5.4.7. Pengujian Fungsi <i>Monitoring</i> Produksi ..... | 203        |
| 5.4.8. Pengujian Fungsi Hapus Data .....                 | 213        |
| <b>BAB VI KESIMPULAN DAN SARAN.....</b>                  | <b>220</b> |
| 6.1 Kesimpulan .....                                     | 220        |
| 6.2 Saran .....                                          | 221        |
| <b>DAFTAR PUSTAKA.....</b>                               | <b>222</b> |
| <b>LAMPIRAN.....</b>                                     | <b>228</b> |



## DAFTAR TABEL

|                                                                   | <b>Halaman</b> |
|-------------------------------------------------------------------|----------------|
| Tabel 2.1 Penelitian Terdahulu.....                               | 14             |
| Tabel 3.1 Jadwal Penelitian .....                                 | 33             |
| Tabel 3.2 <i>Use Case</i> .....                                   | 33             |
| Tabel 3.3 <i>Flowchart</i> .....                                  | 34             |
| Tabel 4.1 Tabel Karyawan .....                                    | 81             |
| Tabel 4.2 Tabel Kepala Bagian.....                                | 81             |
| Tabel 4.3 Tabel Pimpinan.....                                     | 82             |
| Tabel 4.4 Tabel Data Produksi Bibit.....                          | 82             |
| Tabel 4.5 Tabel Data Produksi Tanaman.....                        | 83             |
| Tabel 5.1 Identifikasi Statement Fungsi <i>Login</i> .....        | 132            |
| Tabel 5.2 Test Case Fungsi Login.....                             | 139            |
| Tabel 5.3 Identifikasi Statement Fungsi Logout.....               | 141            |
| Tabel 5.4 Test Case Fungsi Logout.....                            | 144            |
| Tabel 5.5 Identifikasi Statement Fungsi Simpan Data Bibit.....    | 148            |
| Tabel 5.6 Test Case Fungsi Simpan Data Bibit.....                 | 153            |
| Tabel 5.7 Identifikasi Statement Fungsi Simpan Data Tanaman.....  | 158            |
| Tabel 5.8 Test Case Fungsi Simpan Data Tanaman.....               | 164            |
| Tabel 5.9 Identifikasi Statement Fungsi Validasi Data.....        | 167            |
| Tabel 5.10 Test Case Fungsi Validasi Data.....                    | 172            |
| Tabel 5.11 Identifikasi Statement Fungsi Realisasi Tahapan.....   | 180            |
| Tabel 5.12 Test Case Fungsi Realisasi Tahapan.....                | 185            |
| Tabel 5.13 Identifikasi Statement Fungsi Monitoring Produksi..... | 189            |
| Tabel 5.14 Test Case Fungsi Monitoring Produksi.....              | 196            |
| Tabel 5.15 Identifikasi Statement Fungsi Hapus Data.....          | 200            |
| Tabel 5.16 Test Case Fungsi Hapus Data.....                       | 203            |

## DAFTAR GAMBAR

|                                                                                                              | Halaman |
|--------------------------------------------------------------------------------------------------------------|---------|
| Gambar 2.1 Kerangka Berpikir Penelitian.....                                                                 | 18      |
| Gambar 3.1 Struktur Organisasi.....                                                                          | 25      |
| Gambar 3.2 Metode Pengembangan Sistem <i>Design Thinking</i> .....                                           | 35      |
| Gambar 3.3 <i>White Box Testing Process</i> .....                                                            | 45      |
| Gambar 4.1 <i>Usecase</i> Keseluruhan.....                                                                   | 55      |
| Gambar 4.2 <i>Usecase</i> Diagram Karyawan Lapangan .....                                                    | 57      |
| Gambar 4.3 <i>Usecase</i> Diagram Kepala Bagian.....                                                         | 59      |
| Gambar 4.4 <i>Usecase</i> Diagram Pimpinan.....                                                              | 61      |
| Gmabar 4.5 <i>Activity Diagram</i> Login Karyawan .....                                                      | 63      |
| Gambar 4.6 <i>Activity Diagram</i> Karyawan Lapangan Melihat Menu Utama.....                                 | 64      |
| Gambar 4.7 <i>Activity Diagram</i> Karyawan Lapangan Melihat Data Produksi .....                             | 65      |
| Gambar 4.8 <i>Activity Diagram</i> Karyawan Lapangan Input Data Produksi.....                                | 66      |
| Gambar 4.9 <i>Activity Diagram</i> Karyawan Lapangan <i>Update</i> Data Produksi .....                       | 67      |
| Gambar 4.10 <i>Activity Diagram</i> Karyawan Lapangan Hapus Data Produksi.....                               | 68      |
| Gambar 4.11 <i>Activity Diagram</i> Karyawan Lapangan Input Laporan Produksi.....                            | 69      |
| Gambar 4.12 <i>Activity Diagram</i> Notifikasi Karyawan Lapangan.....                                        | 70      |
| Gambar 4.13 <i>Activity Diagram</i> Logout Karyawan Lapangan.....                                            | 71      |
| Gambar 4.14 <i>Activity Diagram</i> Login Kepala Bagian .....                                                | 72      |
| Gambar 4.15 <i>Activity Diagram</i> Kepala Bagian Melihat ( <i>Dashboard</i> ).....                          | 73      |
| Gambar 4.16 <i>Activity Diagram</i> Kepala Bagian Melihat Data Produksi dan Memvalidasi Data Produksi.....   | 74      |
| Gambar 4.17 <i>Activity Diagram</i> Kepala Bagian <i>Monitoring</i> Laporan Produksi.....                    | 75      |
| Gambar 4.18 <i>Activity Diagram</i> Logout Kepala.....                                                       | 76      |
| Gambar 4.19 <i>Activity Diagram</i> Login Pimpinan.....                                                      | 77      |
| Gambar 4.20 <i>Activity Diagram</i> Pimpinan Melihat Menu Utama Dan <i>Monitoring</i> Laporan Produksi ..... | 78      |
| Gambar 4.21 <i>Activity Diagram</i> Pimpinan.....                                                            | 79      |
| Gambar 4.22 <i>Class Diagram</i> .....                                                                       | 80      |
| Gambar 4.23 Halaman Awal.....                                                                                | 84      |

|                                                              |     |
|--------------------------------------------------------------|-----|
| Gambar 4.24 Halaman <i>Login</i> .....                       | 85  |
| Gambar 4.25 Halaman <i>Dashboard</i> Karyawan Lapangan ..... | 86  |
| Gambar 4.26 Halaman Data Produksi Bibit.....                 | 87  |
| Gambar 4.27 Halaman Detail Produksi Bibit .....              | 88  |
| Gambar 4.28 Halaman Data Produksi Tanaman .....              | 89  |
| Gambar 4.29 Halaman Detail Produksi Tanaman.....             | 90  |
| Gambar 4.31 Halaman <i>Input</i> Data Produksi Bibit .....   | 91  |
| Gambar 4.31 Halaman <i>Input</i> Data Produksi Tanaman.....  | 92  |
| Gambar 4.32 Halaman Laporan .....                            | 93  |
| Gambar 4.33 Halaman Notifikasi.....                          | 94  |
| Gambar 4.34 Halaman Profil Karyawan Lapangan.....            | 96  |
| Gambar 4.35 Halaman <i>Dashboard</i> Kepala Bagian.....      | 97  |
| Gambar 4.36 Halaman Data Produksi Bibit.....                 | 98  |
| Gambar 4.37 Halaman Detail Produksi Bibit.....               | 99  |
| Gambar 4.38 Halaman Data Produksi Tanaman.....               | 100 |
| Gambar 4.39 Halaman Detail Produksi Tanaman.....             | 101 |
| Gambar 4.40 Halaman Validasi Produksi.....                   | 102 |
| Gambar 4.41 Halaman <i>Monitoring</i> Produksi.....          | 103 |
| Gambar 4.42 Halaman Laporan.....                             | 104 |
| Gambar 4.43 Halaman Manajemen Pengguna.....                  | 105 |
| Gambar 4.44 Halaman Profil Kepala Bagian.....                | 107 |
| Gambar 4.45 Halaman <i>Dashboard</i> Pimpinan.....           | 108 |
| Gambar 4.46 Halaman <i>Monitoring</i> Produksi.....          | 109 |
| Gambar 4.47 Halaman Laporan.....                             | 110 |
| Gambar 4.48 Halaman Notifikasi.....                          | 111 |
| Gambar 4.49 Halaman Profil Pimpinan .....                    | 112 |
| Gambar 5.1 Tabel Karyawan.....                               | 116 |
| Gambar 5.2 Tabel Kepala Bagian.....                          | 117 |
| Gambar 5.3 Tabel Pimpinan.....                               | 118 |
| Gambar 5.4 Tabel Produksi Bibit.....                         | 119 |

|                                                             |     |
|-------------------------------------------------------------|-----|
| Gambar 5.5 Tabel Produksi Tanaman.....                      | 120 |
| Gambar 5.6 <i>Splash Screen</i> .....                       | 122 |
| Gambar 5.7 Halaman <i>Login</i> .....                       | 123 |
| Gambar 5.8 Halaman <i>Dashboard</i> Karyawan Lapangan ..... | 124 |
| Gambar 5.9 Halaman Data Produksi Bibit.....                 | 125 |
| Gambar 5.10 Halaman Detail Produksi Bibit .....             | 126 |
| Gambar 5.11 Halaman Data Produksi Tanaman .....             | 127 |
| Gambar 5.12 Halaman Detail Produksi Tanaman.....            | 128 |
| Gambar 5.13 Halaman <i>Input</i> Data Produksi Bibit .....  | 129 |
| Gambar 5.14 Halaman <i>Input</i> Data Produksi Tanaman..... | 130 |
| Gambar 5.15 Halaman Laporan .....                           | 131 |
| Gambar 5.16 Halaman Notifikasi.....                         | 132 |
| Gambar 5.17 Halaman Profil Karyawan Lapangan.....           | 133 |
| Gambar 5.18 Halaman <i>Dashboard</i> Kepala Bagian.....     | 134 |
| Gambar 5.19 Halaman Data Produksi Bibit.....                | 135 |
| Gambar 5.20 Halaman Detail Produksi Bibit.....              | 136 |
| Gambar 5.21 Halaman Data Produksi Tanaman.....              | 137 |
| Gambar 5.22 Halaman Detail Produksi Tanaman.....            | 138 |
| Gambar 5.23 Halaman Validasi Produksi.....                  | 139 |
| Gambar 5.24 Halaman <i>Monitoring</i> Produksi.....         | 140 |
| Gambar 5.25 Halaman Laporan.....                            | 141 |
| Gambar 5.26 Halaman Manajemen Pengguna.....                 | 142 |
| Gambar 5.27 Halaman Profil Kepala Bagian.....               | 144 |
| Gambar 5.28 Halaman <i>Dashboard</i> Pimpinan.....          | 145 |
| Gambar 5.29 Halaman <i>Monitoring</i> Produksi.....         | 146 |
| Gambar 5.30 Halaman Laporan.....                            | 147 |
| Gambar 5.31 Halaman Notifikasi.....                         | 148 |
| Gambar 5.32 Halaman Profil Pimpinan.....                    | 149 |
| Gambar 5.33 Flowchart Fungsi <i>Login</i> .....             | 153 |

|                                                       |     |
|-------------------------------------------------------|-----|
| Gambar 5.34 Flowgraph Fungsi <i>Login</i> .....       | 155 |
| Gambar 5.35 Flowchart Fungsi <i>Logout</i> .....      | 161 |
| Gambar 5.36 Flowgraph Fungsi <i>Logout</i> .....      | 162 |
| Gambar 5.37 Flowchart Fungsi Simpan Data Bibit.....   | 168 |
| Gambar 5.38 Flowgraph Fungsi Simpan Data Bibit.....   | 170 |
| Gambar 5.39 Flowchart Fungsi Simpan Data Tanaman..... | 178 |
| Gambar 5.40 Flowgraph Fungsi Simpan Data Tanaman..... | 180 |
| Gambar 5.41 Flowchart Fungsi Validasi Data.....       | 187 |
| Gambar 5.42 Flowgraph Fungsi Validasi Data.....       | 189 |
| Gambar 5.43 Flowchart Fungsi Realisasi Tahapan.....   | 200 |
| Gambar 5.44 Flowgraph Fungsi Realisasi Tahapan.....   | 202 |
| Gambar 5.45 Flowchart Fungsi Monitoring Produksi..... | 209 |
| Gambar 5.46 Flowgraph Fungsi Monitoring Produksi..... | 211 |
| Gambar 5.47 Flowchart Fungsi Hapus Data.....          | 218 |
| Gambar 5.48 Flowgraph Fungsi Hapus Data.....          | 219 |



## DAFTAR LAMPIRAN

|                                                          | Halaman |
|----------------------------------------------------------|---------|
| Lampiran 1. Surat Permohonan Penelitian.....             | 230     |
| Lampiran 2. Surat Balasan Institusi Lokasi Kegiatan..... | 231     |
| Lampiran 3. Dokumentasi Wawancara & Observasi.....       | 232     |
| Lampiran 4. Daftar Pertanyaan Wawancara.....             | 234     |



## DAFTAR SINGKATAN

| Singkatan      | Kepanjangan                                 |
|----------------|---------------------------------------------|
| PT             | : Perseroan Terbatas                        |
| PT.KA          | : PT Kendi Arindo                           |
| PP             | : Peraturan Pemerintah                      |
| KBLI           | : Klasifikasi Baku Lapangan Usaha Indonesia |
| APL            | : Area Peruntukan Lain                      |
| CLI            | : Command Line Interface                    |
| <i>IoT</i>     | : <i>Internet of Things</i>                 |
| JSON           | : JavaScript Object Notation                |
| SDK            | : Software Development Kit                  |
| <i>VS Code</i> | : <i>Visual Studio Code</i>                 |
| AMDAL          | : Analisis Mengenai Dampak Lingkungan Hidup |

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Pengelolaan data produksi bibit tanaman karet di lapangan masih menghadapi berbagai tantangan, terutama karena sebagian besar proses pencatatan dan *monitoring* masih dilakukan secara manual. Kondisi ini merupakan fenomena yang umum terjadi pada sektor perkebunan, khususnya pada perusahaan dengan area operasional yang luas. Akibatnya, informasi produksi tidak tersampaikan secara cepat, rentan terhadap kesalahan pencatatan, serta sulit diakses kembali saat diperlukan. Di tengah kebutuhan industri perkebunan yang semakin menuntut kecepatan dan ketepatan informasi, peningkatan kualitas sistem *monitoring* menjadi hal yang sangat penting untuk diterapkan.

Teknologi digital hadir sebagai solusi untuk mengatasi permasalahan tersebut. Teknologi digital merupakan teknologi yang pengoperasiannya tidak lagi membutuhkan banyak tenaga manusia dan bertujuan untuk menggunakan sistem otomatis dengan sistem komputer (Wibowo, 2023). Penerapan teknologi digital dalam sektor perkebunan semakin mempermudah proses pengawasan, pencatatan, dan pengambilan keputusan secara cepat serta akurat. Melalui sistem digital, berbagai aktivitas seperti pemantauan pertumbuhan tanaman, manajemen bibit, distribusi pupuk, hingga analisis hasil produksi dapat dilakukan secara otomatis dan *real-time*.

Pemanfaatan teknologi ini tidak hanya meningkatkan produktivitas, tetapi juga membantu optimalisasi penggunaan sumber daya, meminimalkan kesalahan manusia, serta mendukung penerapan pertanian modern yang berkelanjutan. Dengan demikian, pemanfaatan teknologi digital berperan penting dalam meningkatkan efisiensi operasional secara signifikan (Alqamari, 2025).

PT. Kendi Arindo merupakan perusahaan yang bergerak di bidang perkebunan karet dengan area operasional yang cukup luas. Sebagai salah satu

pelaku industri perkebunan, perusahaan ini menjalankan berbagai proses pengelolaan bibit dan tanaman karet yang menjadi faktor utama dalam menentukan produktivitas serta kualitas hasil panen. Namun, sistem pencatatan dan *monitoring* informasi data produksi yang masih dilakukan secara manual menimbulkan beberapa permasalahan. Permasalahan tersebut antara lain keterlambatan informasi karena laporan baru diterima setelah proses rekapitulasi manual selesai, risiko kehilangan data akibat penyimpanan berbasis dokumen fisik, serta kurangnya efisiensi waktu dalam pengolahan dan pelaporan data.

Penerapan sistem informasi di lingkungan perusahaan berperan penting dalam meningkatkan efisiensi operasional melalui otomatisasi dan pengelolaan data digital yang terstruktur. Sistem informasi yang dirancang dengan baik mampu mempercepat proses pengolahan data, mengurangi potensi kesalahan manusia, serta mempermudah proses pengambilan keputusan secara *real-time*. Selain itu, penerapan teknologi berbasis digital juga mendukung transparansi dan integrasi data antarbagian dalam perusahaan, sehingga seluruh proses bisnis dapat berjalan lebih efektif, akurat, dan terukur (Arujisaputra, 2025).

Meskipun demikian, berdasarkan hasil kajian terhadap penelitian-penelitian sebelumnya, masih terbatas penelitian yang secara khusus membahas pengembangan sistem digital *monitoring* informasi data produksi bibit tanaman karet berbasis *android* yang diterapkan langsung pada lingkungan operasional perusahaan perkebunan. Sebagian penelitian cenderung berfokus pada sistem pengelolaan data pertanian secara umum dan belum mengintegrasikan monitoring produksi bibit secara *real-time* melalui perangkat *mobile*. Hal ini menunjukkan adanya celah penelitian yang perlu dikaji lebih lanjut.

Selain itu, perkembangan perangkat *mobile* berbasis *android* yang semakin luas digunakan di berbagai sektor industri membuka peluang besar dalam implementasi sistem *monitoring* lapangan yang lebih fleksibel. Perangkat *mobile* memungkinkan petugas lapangan untuk melakukan input data secara langsung di lokasi tanpa harus menunggu proses rekapitulasi manual di kantor. Hal ini tentu memberikan keunggulan dalam hal kecepatan pelaporan, akurasi data, serta efisiensi waktu kerja. Pemanfaatan aplikasi berbasis *android* juga dinilai lebih

praktis karena dapat dioperasikan dengan perangkat yang relatif terjangkau dan mudah digunakan oleh pengguna.

Dalam konteks pengelolaan bibit tanaman karet, *monitoring* yang dilakukan secara berkala dan terdokumentasi dengan baik sangat berpengaruh terhadap kualitas dan keberhasilan pertumbuhan tanaman. Setiap tahapan produksi, mulai dari pembibitan, perawatan, hingga distribusi, memerlukan pencatatan yang sistematis agar dapat dianalisis secara berkelanjutan. Tanpa sistem yang terintegrasi, data historis produksi akan sulit ditelusuri kembali, sehingga menyulitkan evaluasi kinerja dan perencanaan produksi pada periode berikutnya.

Oleh karena itu, kebaruan (*novelty*) dari penelitian ini terletak pada pengembangan sistem digitalisasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android* yang dirancang untuk menggantikan proses pencatatan manual dengan sistem yang terintegrasi, *real-time*, dan mudah diakses melalui perangkat *mobile*. Sistem ini diimplementasikan secara langsung pada PT. Kendi Arindo sebagai studi kasus, sehingga mampu memberikan gambaran penerapan sistem secara nyata di lingkungan industri perkebunan karet.

Urgensi penelitian ini didasarkan pada kebutuhan PT. Kendi Arindo untuk segera beradaptasi dengan perkembangan teknologi digital guna meningkatkan efektivitas *monitoring* dan efisiensi pengelolaan data produksi bibit tanaman karet. Dengan adanya sistem digital ini, diharapkan penyampaian informasi dapat dilakukan secara lebih cepat dan akurat, risiko kehilangan data dapat diminimalkan, serta pengambilan keputusan dapat dilakukan secara tepat waktu. Dengan demikian, penelitian ini diharapkan dapat mendukung transformasi digital perusahaan dan meningkatkan produktivitas di lingkungan PT. Kendi Arindo.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan, maka rumusan masalah dalam penelitian ini adalah sebagai berikut:

1. Bagaimana kondisi sistem *monitoring* dan pencatatan data produksi bibit tanaman karet yang saat ini masih dilakukan secara manual di PT. Kendi Arindo serta kendala yang dihadapi dalam proses pengelolaannya?

2. Bagaimana perancangan sistem digitalisasi *monitoring* dan informasi data produksi bibit tanaman karet berbasis *android* untuk mengatasi permasalahan keterlambatan pencatatan, kesalahan data, dan keterbatasan akses informasi di lapangan di PT. Kendi Arindo?
3. Bagaimana kebutuhan sistem aplikasi berbasis *android* di PT. Kendi Arindo yang mampu mempermudah proses pencatatan, pemantauan, dan pelaporan data produksi bibit tanaman karet sesuai dengan kebutuhan operasional?

### 1.3 Batasan Masalah

Agar penelitian ini lebih terarah dan tidak menyimpang dari tujuan yang diharapkan, maka ditetapkan beberapa batasan sebagai berikut:

1. Hasil akhir penelitian berupa aplikasi berbasis *android* yang menampilkan sistem *monitoring* dan pengelolaan informasi data produksi bibit tanaman karet di PT. Kendi Arindo.
2. Fitur utama aplikasi meliputi digitalisasi proses *monitoring* dan pengelolaan informasi data produksi bibit tanaman karet, termasuk pencatatan, penyimpanan, serta pelaporan data secara digital.
3. Tampilan aplikasi dibatasi pada *form input* data produksi, halaman *monitoring* data produksi, dan tampilan laporan informasi data produksi.
4. Aplikasi hanya menampilkan hasil pengolahan data produksi bibit tanaman karet, tanpa menampilkan fitur atau informasi lain di luar kebutuhan tersebut.

### 1.4 Tujuan Penelitian

Adapun tujuan dari penelitian ini yaitu:

1. Untuk mengetahui kondisi sistem *monitoring* dan pencatatan data produksi bibit tanaman karet yang saat ini diterapkan di PT. Kendi Arindo serta kendala yang dihadapi dalam proses pengelolaannya.
2. Untuk merancang sistem digitalisasi *monitoring* dan informasi data produksi bibit tanaman karet berbasis *android* yang sesuai dengan kebutuhan operasional PT. Kendi Arindo.

3. Untuk menganalisis kebutuhan sistem aplikasi berbasis *android* yang mampu mempermudah proses pencatatan, pemantauan, dan pelaporan data produksi bibit tanaman karet sesuai dengan kebutuhan operasional PT. Kendi Arindo.

## 1.5 Manfaat Penelitian

Penelitian ini diharapkan memberikan manfaat sebagai berikut:

### 1. Manfaat Teoritis

- a. Memberikan kontribusi terhadap pengembangan ilmu pengetahuan di bidang rekayasa perangkat lunak dan sistem informasi, khususnya dalam penerapan aplikasi berbasis *Android* pada sektor perkebunan.
- b. Mendukung kebijakan digitalisasi dan transformasi teknologi dalam pengelolaan data produksi di sektor perkebunan.

### 2. Manfaat Praktis

#### 2.1 Bagi Perusahaan

- a. Mempermudah proses pencatatan dan *monitoring* informasi data produksi bibit tanaman karet secara digital dan terintegrasi.
- b. Meningkatkan efisiensi, akurasi, dan kecepatan pengelolaan data produksi di lingkungan operasional perusahaan.
- c. Menyediakan laporan yang lebih cepat untuk mendukung manajemen perusahaan.

#### 2.2 Bagi Akademisi/Peneliti Selanjutnya

Dapat menjadi referensi dalam penelitian sejenis mengenai pengembangan aplikasi berbasis *android* untuk sektor perkebunan atau industri lainnya.

## 1.6 Sistematika Penulisan

Agar memudahkan dalam memberikan gambaran secara utuh terhadap penulisan skripsi ini adalah sebagai berikut

### 1. BAB I PENDAHULUAN

Bab ini menjelaskan latar belakang dilakukannya penelitian, rumusan masalah yang didapat, batasan masalah yang ditetapkan, tujuan penelitian, manfaat penelitian, serta sistematika penulisan yang diterapkan.

### 2. BAB II TINJAUAN PUSTAKA

Tinjauan pustaka pada penelitian ini membahas teori-teori yang mendukung pengembangan sistem digitalisasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android*. Sistem operasi *android* dipilih karena bersifat *open source*, fleksibel, mudah dikembangkan, serta banyak digunakan di berbagai perangkat *mobile*, sehingga lebih praktis digunakan dalam proses pencatatan dan pemantauan data produksi di lapangan.

### 3. BAB III METODE PENELITIAN

Bab ini menjelaskan metode yang digunakan dalam penelitian untuk menyelesaikan permasalahan yang telah dirumuskan. Penelitian ini menggunakan model pengembangan sistem *design thinking* yang meliputi tahapan analisis kebutuhan, perancangan sistem, implementasi, serta pengujian aplikasi. Pembahasan dalam bab ini mencakup *use case diagram* dan desain antarmuka aplikasi digitalisasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android*.

### 4. BAB IV ANALISIS DAN PERANCANGAN

Bab ini membahas proses analisis dan perancangan aplikasi digitalisasi *monitoring* informasi data produksi bibit tanaman karet berbasis *Android*. Tahapan analisis dilakukan untuk mengidentifikasi permasalahan serta kebutuhan sistem yang berkaitan dengan proses *monitoring* dan pengelolaan data produksi bibit tanaman karet. Hasil dari analisis tersebut digunakan sebagai dasar dalam perancangan sistem, meliputi

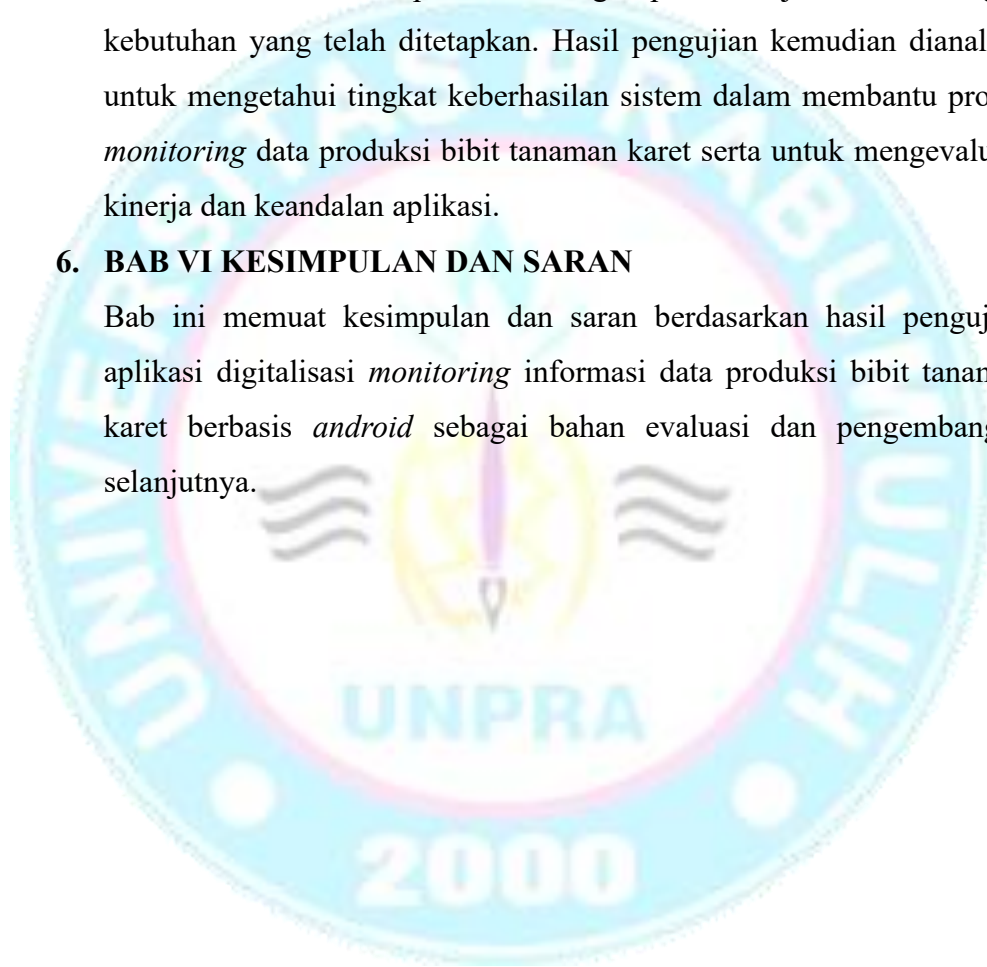
perancangan alur proses, basis data, serta antarmuka aplikasi yang akan dikembangkan agar dapat mendukung kegiatan *monitoring* data produksi secara efektif dan efisien.

## **5. BAB V HASIL DAN PEMBAHASAN**

Bab ini menyajikan hasil implementasi dan pengujian dari aplikasi digitalisasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android* yang telah dirancang. Pengujian dilakukan untuk memastikan bahwa setiap fitur dan fungsi aplikasi berjalan sesuai dengan kebutuhan yang telah ditetapkan. Hasil pengujian kemudian dianalisis untuk mengetahui tingkat keberhasilan sistem dalam membantu proses *monitoring* data produksi bibit tanaman karet serta untuk mengevaluasi kinerja dan keandalan aplikasi.

## **6. BAB VI KESIMPULAN DAN SARAN**

Bab ini memuat kesimpulan dan saran berdasarkan hasil pengujian aplikasi digitalisasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android* sebagai bahan evaluasi dan pengembangan selanjutnya.



## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Pengertian Digitalisasi**

Digitalisasi adalah proses mengubah data analog menjadi digital dengan menggunakan teknologi dan mengoperasikannya secara otomatis melalui sistem komputerisasi (Bainar, 2023). Digitalisasi adalah proses pergeseran dari teknologi analog ke digital, atau penggunaan data digital dan teknologi untuk meningkatkan kinerja, efisiensi, atau model bisnis (Aprilia Lusiana, 2024).

Digitalisasi dilakukan untuk membuat arsip dokumen berbentuk digital, digitalisasi memerlukan peralatan seperti komputer, scanner, operator media sumber dan software pendukung. Digitalisasi merupakan proses alih media dari cetak atau analog ke dalam media digital atau elektronik melalui proses *scanning*, digital *photograph* atau teknik lainnya (Hidayatullah, 2023).

Digitalisasi adalah proses dimana sebelumnya pekerjaan dilakukan secara manual kemudian pekerjaan tersebut dilakukan secara otomatisasi dengan menggunakan teknologi. Karena memang pada saat ini teknologi sedang berkembang pesat di seluruh belahan negara. Digitalisasi merupakan terobosan terbaru di tengah tengah arus globalisasi yang sedang berkembang pesat pada saat ini. Di Semua sektor kehidupan kini telah memanfaatkan digitalisasi (Iqbal Hasyim et al., 2023).

Dapat disimpulkan bahwa digitalisasi adalah proses mengubah informasi, sistem, atau aktivitas yang sebelumnya bersifat manual atau analog menjadi bentuk digital menggunakan teknologi informasi dan komunikasi.

#### **2.2 Pengertian Monitoring**

*Monitoring* adalah langkah mengkaji, mengidentifikasi kegiatan yang dilaksanakan sesuai dengan rencana untuk mencapai tujuan dan lebih berarti bagi penerima (Khalid et al., 2022). *Monitoring* adalah suatu langkah pengumpulan dan analisis informasi berdasarkan keterangan yang ditetapkan secara sistematis

tentang suatu kegiatan atau program sehingga dapat dilakukan tindakan koreksi guna penyempurnaan kegiatan atau program itu selanjutnya (Khairat et al., 2022).

*Monitoring* merupakan kegiatan untuk mengetahui apakah program yang dibuat itu berjalan dengan baik sebagaimana mestinya sesuai dengan yang direncanakan, adakah hambatan yang terjadi dan bagaimana para pelaksana kebijakan itu mengatasi hambatan tersebut (Nasihi et al., 2022). *Monitoring* merupakan bentuk pemantauan dari proses pengumpulan dan analisis informasi berdasarkan suatu kegiatan dengan tujuan untuk mengetahui dan mengevaluasi serta mengambil tindakan atas kesalahan yang dihasilkan (Megawaty & Putra, 2020).

*Monitoring* adalah suatu sistem yang dirancang untuk mengamati, mencatat, dan melaporkan kondisi atau perubahan dari suatu objek atau proses secara terus-menerus dalam jangka waktu tertentu. Tujuan utama dari sistem ini adalah untuk memberikan informasi secara *real-time* atau periodik guna mendukung pengambilan keputusan yang cepat dan tepat (Siregar & Handoko, 2021).

Berdasarkan beberapa pendapat tersebut, *monitoring* dapat disimpulkan sebagai aktivitas pengawasan dan pengendalian terhadap proses atau kegiatan tertentu secara terus-menerus untuk memastikan ketercapaian tujuan sesuai dengan rencana.

### 2.3 Pengertian Manajemen Informasi

Manajemen adalah strategi pemanfaatan tenaga dan pikiran orang lain untuk melaksanakan suatu aktivitas yang diarahkan pada pencapaian tujuan yang telah ditentukan sebelumnya (Nurmadiyah, 2022). Manajemen informasi adalah manajemen dari proses dan sistem yaitu membuat, mengambil, mengatur, menyimpan, mendistribusikan, dan menggunakan informasi (Dhae, 2021).

Manajemen informasi merupakan segala kegiatan yang berkaitan dengan perolehan informasi, dalam penggunaan informasi seefektif mungkin dan juga pembuangan terhadap data yang tidak berguna pada waktu yang tepat (Nurul et al., 2022). Sistem informasi adalah seperangkat sistem yang saling berhubungan yang mengumpulkan (mengambil), memproses, menyimpan dan mendistribusikan

informasi untuk mendukung pengambilan keputusan dan pengendalian dalam suatu organisasi (Saputra et al., 2023).

Informasi adalah suatu pertambahan dalam ilmu pengetahuan yang menyumbangkan kepada konsep kerangka kerja yang umum dan fakta-fakta yang diketahui. Informasi adalah data yang sudah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat dalam pengambilan keputusan saat ini saat mendatang. Informasi adalah data yang diolah menjadi bentuk yang lebih berguna masukan dan lebih berarti bagi yang menerimanya (Tuasamu Zinab et al., 2023).

Dapat disimpulkan bahwa manajemen informasi merupakan proses terstruktur dalam mengumpulkan, mengelola, menyimpan, dan mendistribusikan informasi agar dapat digunakan secara efektif untuk mendukung pengambilan keputusan, meningkatkan efisiensi organisasi, serta memastikan ketersediaan informasi yang akurat dan tepat waktu.

## 2.4 Pengertian *Android*

*Android* merupakan sistem operasi untuk telepon seluler yang berbasis *linux*. *Android* menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti bergerak (Budiyanto, 2023). *Android* adalah sebuah sistem operasi untuk perangkat *mobile* berbasis *linux* yang mencakup sistem operasi, *middleware* dan aplikasi (Yusuf & Nopitasari, 2021).

*Android* adalah sebuah sistem operasi untuk perangkat *mobile* berbasis *linux* yang mencakup sistem operasi, *middleware*, dan aplikasi. *android* adalah sistem operasi untuk telepon seluler yang berbasis *linux*. *Android* menyediakan platform terbuka bagi para pengembang untuk membuat aplikasi. Pada awalnya dikembangkan oleh *android inc*, sebuah perusahaan pendatang baru yang membuat perangkat lunak untuk ponsel yang kemudian dibeli oleh google inc (Sulistyorinia et al., 2022).

*Android* itu sendiri adalah sebuah sistem operasi untuk perangkat *mobile* berbasis *linux* yang mencakup sistem operasi, *middleware* dan aplikasi. *Android* menyediakan platfrom terbuka bagi para pengembang untuk menciptakan aplikasi

mereka. *Android* adalah sebuah toolkit software yang baru untuk perangkat bergerak yang dibuat oleh google dan open handset alliance. Dalam beberapa tahun, *android* diharapkan dapat ditemukan dalam jutaan handphone dan berbagai perangkat bergerak, membuat *android* menjadi platform utama untuk pengembang aplikasi. Sudah ada banyak platform *mobile* di pasar saat ini, termasuk *Symbian*, *iPhone*, *Windows Mobile*, *BlackBerry*, *Java Mobile Edition*, *Linux Mobile* (LiMo) (Fitriani & Apridiansyah, 2021).

Dapat disimpulkan bahwa *android* adalah sistem operasi *mobile* berbasis *linux* yang menyediakan *platform* terbuka bagi pengembang untuk membuat dan menjalankan aplikasi di berbagai perangkat bergerak

## 2.5 Pengertian Produksi

Produksi adalah proses pengubahan bahan baku menjadi barang jadi atau juga sebagai menambah nilai pada suatu produk (barang dan jasa) agar bisa memenuhi kebutuhan masyarakat (Arwini, 2021). Teori istilah produksi diartikan sebagai penggunaan atau pemanfaatan sumber daya yang mengubah suatu komoditi menjadi komoditi lainnya yang sama sekali berbeda, baik dalam pengertian apa, dan dimana atau kapan komoditi-komoditi itu dilokasikan, maupun dalam pengertian apa yang dapat dikerjakan oleh konsumen terhadap komoditi itu (Sandi et al., 2021).

Produksi merupakan dapurnya perusahaan tersebut, apabila pada kegiatan produksi terdapat berbagai macam hambatan dalam proses produksi, maka akan menimbulkan berbagai kerugian baik material maupun immaterial. Sehingga diperlukan pengendalian kualitas dalam suatu perusahaan untuk meminimalisasi biaya operasional dalam produk, serta dapat memenuhi standar kualitas dari waktu yang telah ditentukan sebelumnya (Wahyu Nusantara, 2022).

Berdasarkan beberapa pendapat tersebut, dapat disimpulkan bahwa produksi adalah proses terintegrasi untuk menghasilkan produk bernilai tambah dengan memanfaatkan sumber daya yang tersedia secara efisien.

## 2.6 Pengertian Tanaman

Pengertian bibit adalah tanaman dalam ukuran kecil yang diperoleh dengan cara pengembangan *vegetatife*, *generative*, kultur jaringan dan yang lainnya bisa juga dari kombinasi keduanya dalam hal pengembangannya (Darnita et al., 2022). Tanaman karet merupakan salah satu komoditas perkebunan yang mempunyai arti penting dalam aspek kehidupan sosial ekonomi masyarakat Indonesia, yaitu sebagai salah satu komoditi penghasil devisa negara (Versi Putra Jaya Hulu, 2022).

Pembibitan merupakan pertumbuhan awal suatu tanaman sebagai penentu pertumbuhan selanjutnya maka pemeliharaan dalam pembibitan harus lebih intensif dan diperhatikan. Tanaman sebagai penentu pertumbuhan selanjutnya maka pemeliharaan dalam pembibitan harus lebih intensif dan diperhatikan. Selain pemupukan, pertumbuhan bibit juga dipengaruhi jenis tanah yang digunakan sebagai media (Ovender et al., 2021).

Dapat disimpulkan bahwa bibit merupakan cikal bakal tanaman yang diperoleh melalui berbagai metode pengembangan seperti vegetatif, generatif, maupun kultur jaringan, sedangkan tanaman merupakan hasil dari proses pertumbuhan bibit yang memiliki nilai ekonomi dan sosial penting. Dengan demikian, bibit dan tanaman dapat dipahami sebagai dua komponen utama dalam siklus produksi pertanian yang saling berhubungan dan menjadi dasar dalam pengelolaan sistem perkebunan secara berkelanjutan.

## 2.7 Pengertian Virtual Studio Code (VS Code)

*Visual studio code (VS Code)* adalah editor kode sumber terbuka dan gratis yang dikembangkan oleh Microsoft. *Vs code* mendukung berbagai bahasa pemrograman dan memiliki fitur canggih seperti *debugging*, *IntelliSense* (saran kode otomatis), integrasi Git, dan ekstensi untuk meningkatkan fungsionalitas editor. *Vs code* sangat populer di kalangan pengembang karena kecepatan, fleksibilitas, dan kemampuan untuk dikustomisasi sesuai kebutuhan proyek atau preferensi pengguna (Budiarti & Juleha, 2025).

*Visual studio code* adalah sebuah teks editor ringan dan handal yang dibuat oleh Microsoft untuk sistem operasi multiplatform, artinya tersedia juga untuk versi *Linux*, *Mac*, dan *Windows*. Teks editor ini secara langsung mendukung bahasa

pemrograman *Javascript*, *Typescript*, dan *Node.js*, serta bahasa pemrograman lainnya dengan bantuan plugin yang dapat dipasang via marketplace *Visual Studio Code* seperti : *C++*, *C#*, *Python*, *Go*, *Java*, *PHP* (Ningsih et al., 2022).

*Visual studio code* bisa menjadi *code editor* lebih dari 10 bahasa pemrograman, termasuk Bahasa pemrograman PHP yang dipakai dalam penelitian ini. *Visual Studio Code* adalah salah satu *Integrated Development Environment (IDE)* yang sangat efektif dan populer di kalangan pengembang. *IDE* ini memiliki fitur-fitur yang sangat lengkap dan memudahkan pengembang dalam membuat aplikasi *mobile* (Gunawan, 2024).

Dapat disimpulkan bahwa *visual studio code (VS Code)* merupakan editor kode sumber terbuka yang ringan, cepat, dan fleksibel, dikembangkan oleh Microsoft dan tersedia untuk berbagai sistem operasi seperti Windows, Linux, dan Mac. Editor ini mendukung banyak bahasa pemrograman secara bawaan maupun melalui ekstensi, termasuk PHP yang digunakan dalam penelitian ini. *Vs code* dilengkapi fitur-fitur penting seperti *debugging*, *IntelliSense*, integrasi Git/GitHub, serta beragam *plugin* yang membuatnya sangat populer dan mudah disesuaikan dengan kebutuhan pengembang modern.

## 2.8 Pengertian Flutter

Flutter adalah sebuah *framework open-source* yang dikembangkan oleh Google untuk membangun antar muka (*user interface*) aplikasi *android* dan *ios*. Jika kita berbicara bagaimana cara membuat aplikasi *android* maupun *ios*, biasanya akan dihadapi dengan banyak pilihan kenapa aplikasi tersebut dapat dibangun. Banyak perusahaan besar di dunia telah mengadopsi flutter untuk mengembangkan aplikasi. Sebut saja Google, Alibaba Group, dan Grab (Setiawan et al., 2022).

Flutter adalah kerangka pengembangan aplikasi *mobile open-source* yang dikembangkan oleh Google. Dengan menggunakan flutter, pengembang dapat membangun aplikasi yang memiliki antarmuka pengguna menarik, *responsive*, dan dapat berjalan di berbagai *platform* (Putra & Kurniawan, 2023). Flutter adalah SDK untuk pengembangan aplikasi *mobile* yang dikembangkan oleh Google untuk membangun antarmuka (*user interface*) aplikasi *android* dan *ios*. Rilis pada juni 2018 sama seperti react native, *framework* ini dapat digunakan untuk membuat atau

mengembangkan aplikasi *mobile* yang dapat berjalan pada *device ios* dan *android* (Sudradjat, 2021).

Dapat disimpulkan bahwa flutter merupakan *framework* atau *SDK open-source* yang dikembangkan oleh Google untuk membangun antarmuka aplikasi *mobile* yang menarik, responsif, dan dapat berjalan di berbagai *platform*, terutama *android* dan *ios*. *Framework* ini memungkinkan pengembang membuat aplikasi *cross-platform* dengan satu basis kode, sehingga lebih efisien dan mudah dikembangkan. Sejak dirilis pada tahun 2018, flutter telah banyak diadopsi oleh perusahaan besar seperti Google, Alibaba, dan Grab karena kemampuannya dalam menghasilkan aplikasi berkualitas tinggi dengan performa cepat dan tampilan modern.

## 2.9 Pengertian Dart Programming Language

*Dart* adalah merupakan bahasa pemrograman yang dikembangkan oleh *front-end* untuk kebutuhan dalam membuat aplikasi *android* atau *mobile*, *front-end*, *web*, *IoT*, *back-end* (CLI), dan *Game* (Suryana, 2021). *Dart* adalah bahasa pemrograman terstruktur *open source* untuk membuat aplikasi *web* berbasis *browser* yang kompleks. *Dart* merupakan bahasa pemrograman yang dibuat oleh *Front-end* untuk menggantikan *JavaScript* (Kamil & Lestari, 2023).

*Dart* adalah bahasa pemrograman terstruktur *open source* untuk membuat aplikasi *web* berbasis *browser* yang kompleks. Pengguna dapat menjalankan aplikasi yang dibuat di *Dart* baik dengan menggunakan *browser* yang secara langsung mendukung kode *Dart* atau dengan mengkompilasi kode *Dart* pengguna ke *JavaScript*. Salah satu tujuan utama *Dart* adalah agar bahasa itu tampak akrab (Nelly Sofi & Riza Dharmawan, 2022).

Dapat disimpulkan bahwa *Dart* merupakan bahasa pemrograman terstruktur dan *open source* yang dikembangkan oleh *Front-end*, digunakan untuk membangun berbagai jenis aplikasi seperti *mobile*, *web*, *IoT*, *back-end*, dan *game*, serta dirancang sebagai bahasa modern pengganti *JavaScript* dalam pengembangan aplikasi berbasis *browser* maupun *android*.

## 2.10 Pengertian *Firebase*

*Firebase* adalah *cloud service provider* dan *backend as a service* yang dimiliki Google. *Firebase* merupakan solusi yang ditawarkan oleh Google untuk mempermudah dalam pengembangan aplikasi mobile maupun web dan bersifat *realtime database*. *Firebase database* merupakan penyimpanan basis data *nonSQL* yang memungkinkan untuk menyimpan beberapa tipe data. Tipe data itu antara lain *String*, *Long*, dan *Boolean*. Data pada *firebase database* disimpan sebagai objek *JSON tree*, tidak seperti basis data *SQL*, tidak ada tabel dan baris pada basis data *non-SQL* (Andrianto & Munandar, 2022).

*Firebase* adalah platform terpadu Google yang menyatukan berbagai fitur atau layanan canggih untuk aplikasi, termasuk backend seluler, analisis, dan fitur pertumbuhan. *Firebase* dapat digunakan untuk *realtime database* yang di-host di cloud dan datanya disimpan sebagai *JSON* sehingga mudah disinkronkan secara *real-time* dengan setiap klien yang terhubung, keunggulannya adalah ini gratis dari google untuk menyederhanakan pekerjaan developer dalam membuat aplikasi bergerak (Prasetyawan et al., 2021).

*Firebase* adalah platform basis data milik Google Inc yang terhubung dengan *cloud* dan disimpan sebagai *JSON* dan disinkronkan secara langsung kepada klien yang terhubung dengan internet. Dalam *firebase* memiliki fitur *database realtime* dimana setiap klien bisa berbagi informasi secara langsung apabila terhubung dengan internet. *Realtime database* adalah basis data yang tidak menggunakan *SQL*, sehingga memiliki optimalisasi dan fungsionalitas yang berbeda dengan basis data *SQL*. *API Realtime Database* dibuat agar operasi yang dilakukan berjalan dengan cepat. *Firebase* adalah platform untuk aplikasi *realtime*, ketika data berubah maka aplikasi yang terhubung akan secara langsung melakukan perubahan secara langsung terhadap setiap perangkat yang dipakai, baik *website* maupun *mobile* (Yoga Pratama & Somya, 2021).

Dapat disimpulkan bahwa *firebase* adalah *platform* layanan *cloud* milik Google yang menyediakan berbagai fitur untuk mempermudah pengembangan aplikasi *mobile* maupun *web*, terutama melalui layanan *realtime database*. Sebagai basis data *non-SQL*, *Firebase* menyimpan data dalam format *JSON* dan memungkinkan sinkronisasi data secara langsung ke seluruh klien yang terhubung.

Dengan kemampuannya yang *real-time*, mudah digunakan, serta mendukung berbagai kebutuhan *backend* seperti analisis dan pengelolaan aplikasi, *firebase* menjadi solusi praktis untuk *developer* dalam membangun aplikasi *modern* berbasis internet.

## 2.11 Penelitian Terdahulu

Sebagai bahan acuan bagi peneliti dalam melakukan penelitian, penulisan merujuk pada penelitian-penelitian yang pernah ada dalam bentuk jurnal-jurnal yang akan dijelaskan pada tabel 2.1 sebagai berikut:

Tabel 2.1. Penelitian Terdahulu

| No | Judul Penelitian                                                                                                                               | Nama Peneliti & Tahun               | Pembahasan                                                                                                                                                                                                                                                                        |
|----|------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1  | Sistem Monitoring Pertumbuhan Spike Pada Bibit Tanaman Anggrek Berbasis Pengolahan Citra                                                       | (Apriadi, 2022)                     | Penelitian merancang sistem monitoring pertumbuhan spike anggrek menggunakan metode pengolahan citra digital. Membahas UML, use-case, activity diagram, class diagram, dan proses identifikasi pertumbuhan spike.                                                                 |
| 2  | Smart Farming: Teknologi Monitoring Produksi dan Pemasaran Kebun Organik                                                                       | (Shara & Adzkia Putri Marsya, 2022) | Membahas perancangan aplikasi smart farming untuk monitoring produksi dan pemasaran kebun organik. Aplikasi membantu petani dalam pencatatan pembibitan, pemupukan, lokasi tanam, panen, dan pemasaran digital.                                                                   |
| 3  | Persepsi Petani Terhadap Digitalisasi Pertanian untuk Mendukung Kemandirian Petani                                                             | (Johan Daniel et al., 2022)         | Meneliti bagaimana persepsi, karakteristik, dan peran pendamping memengaruhi digitalisasi pertanian di Kabupaten Sambas. Menggunakan metode SEM dan FGD, hasilnya menunjukkan digitalisasi sangat dipengaruhi oleh faktor-faktor tersebut meski persepsi petani bernilai negatif. |
| 4  | Narative Review Pemanfaatan Internet-of-Things untuk Aplikasi Seed Monitoring and Management System pada Media Tanaman Hidroponik di Indonesia | (Fuada et al., 2023)                | Penelitian ini merancang tentang penggunaan IoT pada pembibitan hidroponik menggunakan sensor suhu, kelembaban, pH, TDS, dan mikrokontroler.                                                                                                                                      |
| 5  | Pengembangan Aplikasi <i>Android</i> Untuk Peningkatan Efisiensi Operasi Pendataan Pengamatan Bunga Tumbuhan                                   | (Aziz & Ernawati, 2023)             | Penelitian ini Mengembangkan aplikasi <i>Android</i> untuk digitalisasi pengamatan bunga. Menggunakan data mining (K-Means) untuk pengelompokan data dan meningkatkan efisiensi operasional                                                                                       |
| 6  | Penerapan Sistem Informasi Monitoring Hasil Panen dan Produksi di                                                                              | (Daulay et al., 2024)               | Penelitian ini Mengembangkan sistem informasi monitoring hasil panen–produksi                                                                                                                                                                                                     |

|   |                                                                                                              |                         |                                                                                                                                                                                       |
|---|--------------------------------------------------------------------------------------------------------------|-------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|   | PTPN III Kebun Sei Baruhur                                                                                   |                         | sawit berbasis web dengan metode RAD untuk meminimalkan kesalahan pencatatan.                                                                                                         |
| 7 | Penerapan Total Quality Management (TQM) dalam Menjaga Kualitas Bibit Tanaman dan Buah di Persada Farm Bogor | (Ananda et al., 2025)   | Implementasi TQM untuk menjaga kualitas bibit dan buah melalui SOP, kontrol mutu, pelatihan, dan evaluasi berkelanjutan.                                                              |
| 8 | Peningkatan Akurasi Sistem Pemantauan Suhu dan Kelembapan Pada Laboratorium Pengujian Benih Tanaman          | (Susantok et al., 2025) | Fokus pada peningkatan akurasi sensor untuk pemantauan kondisi laboratorium benih. Menggunakan sensor suhu dan kelembapan dan melakukan pengujian untuk meningkatkan stabilitas data. |

*Sumber : Penulis, 2025*

## 2.12 Kerangka Berpikir Penelitian

Kerangka berpikir atau kerangka pemikiran adalah dasar pemikiran dari penelitian yang disintesis dari fakta-fakta, observasi dan kajian kepustakaan. Oleh karena itu, kerangka berpikir memuat teori, dalil atau konsep-konsep yang akan dijadikan dasar dalam penelitian. Di dalam kerangka pemikiran variabel-variabel penelitian dijelaskan secara mendalam dan relevan dengan permasalahan yang teliti, sehingga dapat dijadikan dasar untuk menjawab permasalahan penelitian (Riduwan, 2023).

Berikut adalah kerangka berpikir pada penelitian ini :



Sumber : data diolah oleh Penulis, 2025

Gambar 2.1. Kerangka Berpikir Penelitian

Dari gambar 2.1 diatas dapat dijelaskan dan disimpulkan bahwa :

- Identifikasi Masalah: Proses pencatatan data produksi di PT. Kendi Arindo masih manual, menyebabkan keterlambatan informasi dan potensi kesalahan data.
- Analisis Kebutuhan Sistem: Diperlukan sistem digital berbasis *android* agar pencatatan dan pelaporan data produksi dapat dilakukan secara cepat, efisien, dan *real-time*.

- c. Kajian Teori dan Metode Pengembangan: Penelitian menggunakan metode *design thinking* yang sistematis, dimulai dari analisis, perancangan, implementasi, hingga pengujian aplikasi.
- d. Perancangan Sistem: Dirancang aplikasi dengan antarmuka sederhana yang memudahkan pengguna mencatat data bibit, tanaman, dan hasil produksi.



## **BAB III**

### **METODE PENELITIAN**

#### **3.1 Subjek dan Objek Penelitian**

##### **3.1.1. Subjek Penelitian**

Subjek dalam penelitian ini adalah pihak-pihak yang terlibat secara langsung dalam proses pengelolaan dan monitoring data produksi bibit tanaman karet di PT Kendi Arindo. Subjek penelitian meliputi kepala bagian produksi yang bertanggung jawab dalam mengawasi seluruh proses produksi, memastikan target tercapai, serta melakukan evaluasi terhadap hasil produksi yang diperoleh. Selain itu, terdapat kepala bagian perencanaan yang berperan dalam menyusun rencana produksi, mengatur jadwal kerja, serta menganalisis data produksi sebagai dasar pengambilan keputusan dan perencanaan periode berikutnya.

Adapun karyawan atau petugas lapangan juga menjadi subjek penelitian karena mereka terlibat langsung dalam proses pencatatan dan pelaporan data produksi di lapangan serta menjadi pengguna utama aplikasi berbasis *android* yang dikembangkan. Keterlibatan ketiga pihak tersebut sangat penting karena mereka memiliki peran utama dalam alur pengelolaan data produksi, sehingga dapat memberikan informasi yang relevan dalam proses analisis kebutuhan, perancangan, hingga evaluasi sistem yang dibangun.

##### **3.1.2. Objek Penelitian**

Objek penelitian ini adalah PT Kendi Arindo, sebuah perusahaan yang bergerak di bidang perkebunan karet, yang berlokasi di Jalan Lingkar Talang Gunung, Kelurahan Jaya Loka, Kecamatan Tebing Tinggi, Kabupaten Empat Lawang, Provinsi Sumatera Selatan. Penelitian ini berfokus pada digitalisasi sistem *monitoring* dan informasi data produksi bibit tanaman karet berbasis aplikasi *android*. Fokus penelitian diarahkan pada perancangan dan pengembangan aplikasi yang dapat mempermudah proses pencatatan, pelaporan, serta pemantauan data produksi secara *real-time*.

Pemilihan objek penelitian didasarkan pada kondisi di lapangan yang masih menggunakan metode manual dalam pencatatan dan pelaporan data, sehingga

sering menimbulkan keterlambatan informasi dan potensi kesalahan dalam penyampaian data. Dengan adanya digitalisasi melalui aplikasi berbasis *android*, diharapkan efisiensi kerja, akurasi data, serta efektivitas pengawasan dan pengelolaan produksi bibit tanaman karet di PT. Kendi Arindo dapat meningkat secara signifikan.

### 3.1.3. Sejarah

PT. Kendi Arindo (PT.KA) awalnya merupakan perusahaan yang bergerak dibidang produksi dan ekspor arang kayu dimana hingga saat ini arang diproduksi berasal dari bahan baku limbah kayu gergajian sawmill berupa kayu babiran atau pemanfaatan limbah kayu yang terdapat pada tapak lahan perkebunan. Dengan memperkirakan sumber bahan baku akan semakin sedikit maka perusahaan berinisiatif untuk membangun kebun bahan baku berupa perkebunan bahan baku kayu. Semenjak terbit dan diimplementasikannya PP No.5 Tahun 2021 tersebut beberapa KBLI diiadakan termasuk KBLI 02117 (Pengelolaan Tanaman Akasia) dan KBLI 02118 (Pengelolaan Tanaman Eukaliptus) yang merupakan sandaran hukum berdirinya perkebunan PT.Kendi Arindo (PT.KA) sebagai perkebunan penyedia bahan baku arang harus berubah.

Setelah melakukan koordinasi, konsultasi dengan beberapa pihak yang terkait hingga kementerian Lingkungan Hidup dan Kehutanan, karena status lahan PT.KA adalah Area Peruntukan Lain / APL (bukan lahan Kawasan hutan), maka bentuk izin usaha PT.KA harus berupa Perkebunan dan harus mengelola tanaman perkebunan bukan jenis tanaman kehutanan (termasuk diantaranya adalah Akasia dan Eukaliptus). memohon perizinan baru dengan format sebagai usaha bidang Perkebunan tanaman Karet. Sehingga pada tanggal 24 Mei 2022 melalui surat Direktur PT.KA mengajukan permohonan AMDAL perkebunan karet dan industri arang PT.KA perkebunan.

Tanaman karet (*Hevea brasiliensis*) berasal dari kawasan hutan hujan lembah Amazon, Brasil, pada posisi lintang 5° sekitar ekuator dan pada ketinggian tempat di bawah 200 m dari permukaan laut. Karakteristik lainnya adalah memiliki temperatur udara rata- rata bulanan 25-28°C, banyak hujan dengan tidak ada periode kering yang jelas, dan angin yang ringan sepanjang tahun. Wilayah yang cocok

untuk mengusahakan tanaman karet sebaiknya mirip dengan keadaan tersebut di atas. Namun saat ini pengusahaan karet semakin meluas ke wilayah di luar ciri-ciri tersebut dengan tingkat kesesuaian lahan yang rendah. Tanaman karet mampu beradaptasi pada berbagai tipe lahan kecuali pada lahan tergenang (Rusdi, 2023).

### **3.1.4. Visi dan Misi**

Sebagai salah satu perusahaan yang bergerak di bidang perkebunan karet, PT Kendi Arindo memiliki komitmen kuat untuk berperan aktif dalam mendukung pembangunan nasional melalui pengelolaan sumber daya alam yang berkelanjutan. Perusahaan ini berfokus pada pengembangan usaha perkebunan yang tidak hanya berorientasi pada produktivitas, tetapi juga memperhatikan aspek lingkungan dan kesejahteraan masyarakat sekitar. Dalam menjalankan kegiatan operasionalnya, PT Kendi Arindo menetapkan arah dan pedoman kerja yang tertuang dalam visi dan misi, perusahaan sebagai landasan dalam mencapai keberhasilan dan keberlanjutan usaha.

#### **1. Visi**

Menjadi perusahaan perkebunan karet terkemuka bersama masyarakat sekitar melalui peningkatan pertumbuhan dan hasil getah karet serta hasil hutan lainnya secara berkelanjutan dan meningkat dari waktu ke waktu.

#### **2. Misi**

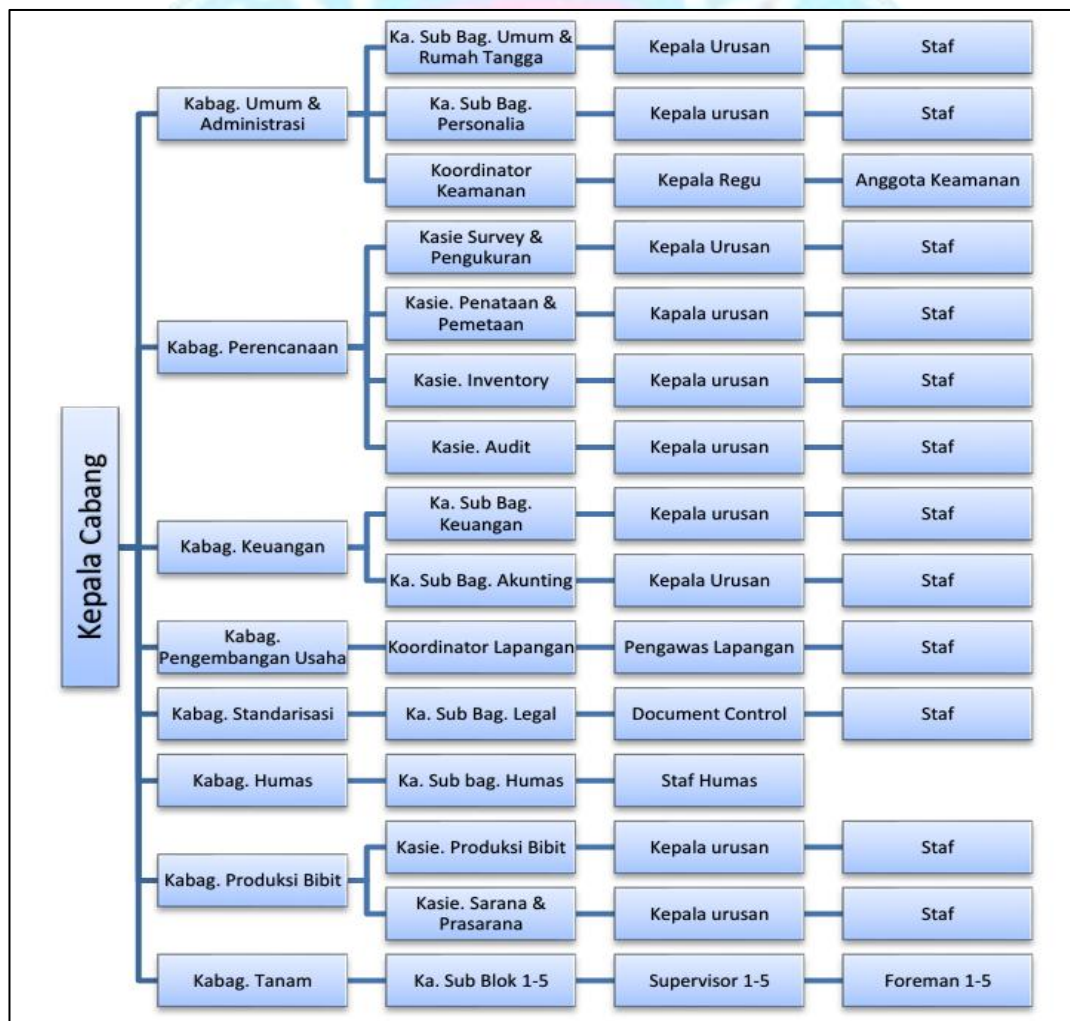
Misi utama PT. KA adalah menunjang pembangunan nasional di bidang perkebunan khususnya penyediaan bahan baku getah karet untuk berbagai keperluan industri, peningkatan produktivitas lahan, peningkatan kualitas lingkungan hidup dan menjalankan usaha dalam bidang pengusahaan dan pengelolaan lahan, yang diimplementasikan dalam kegiatan operasional, meliputi:

1. Kegiatan pengelolaan perkebunan secara lestari dan berwawasan lingkungan.
2. Meningkatkan kuantitas dan kualitas tegakan pokok karet untuk memenuhi kebutuhan bahan baku industry karet.
3. Meningkatkan kualitas lingkungan hidup melalui rehabilitasi lahan yang tidak produktif.
4. Memperluas kesempatan kerja dan berusaha.

5. Meningkatkan pendapatan dan kesejahteraan karyawan dan masyarakat sekitar.

### 3.1.5. Struktur Organisasi

Struktur organisasi merupakan hal penting dalam suatu perusahaan karena menggambarkan pembagian tugas, wewenang, dan tanggung jawab setiap bagian di dalamnya. Adapun struktur organisasi PT Kendi Arindo adalah sebagai berikut:



Sumber : PT.Kendi Arindo ,2024

Gambar 3.1 Struktur Organisasi

Berdasarkan gambar 3.1 struktur organisasi PT Kendi Arindo, dapat dijelaskan bahwa struktur ini menggambarkan susunan dan hubungan antar unsur yang berperan dalam penyelenggaraan kegiatan kerja di PT Kendi Arindo.

- a. Kepala Cabang, bertanggung jawab penuh terhadap seluruh kegiatan operasional di tingkat cabang. Kepala cabang mengoordinasikan seluruh bagian agar kegiatan perusahaan berjalan sesuai dengan kebijakan dan target yang telah ditetapkan oleh manajemen pusat.
- b. Kabag. Umum & Administrasi, mengelola kegiatan administrasi umum, personalia, dan keamanan perusahaan. Bagian ini memastikan seluruh kebutuhan operasional, kepegawaian, dan tata usaha berjalan dengan baik.
  - a. Ka.SubBag. Umum & Rumah Tangga: Mengatur kebutuhan fasilitas kantor, perawatan aset, dan layanan rumah tangga perusahaan.
  - b. Ka.SubBag. Personalia: Mengelola data karyawan, absensi, penggajian, serta urusan kepegawaian lainnya.
  - c. Koordinator Keamanan: Bertanggung jawab atas keamanan lingkungan kerja dan pengawasan anggota keamanan.
- c. Kabag. Perencanaan, mengatur dan menyusun rencana kegiatan operasional, termasuk survei, pemetaan, dan audit internal.
- d. Kabag. Keuangan, bertanggung jawab dalam pengelolaan dan pelaporan keuangan perusahaan cabang.
  - a. Ka.SubBag. Keuangan: Mengatur arus kas, pembayaran, dan administrasi keuangan.
  - b. Ka.SubBag. Akunting: Menyusun laporan keuangan, pembukuan, dan pengawasan transaksi keuangan.
- e. Kabag. Pengembangan Usaha, bertugas mencari peluang pengembangan baru, memperluas jaringan usaha, dan memastikan perusahaan tetap kompetitif di bidang perkebunan karet.
- f. Kabag. Standarisasi, mengawasi penerapan standar mutu, efisiensi kerja, serta kepatuhan terhadap prosedur perusahaan agar seluruh kegiatan operasional memenuhi standar yang ditetapkan.
- g. Kabag. Humas, menjalin komunikasi dengan pihak eksternal, termasuk masyarakat dan pemerintah daerah.
  - a. Ka.Sub Bag. Legal: Mengurus dokumen hukum, perizinan, dan kontrak kerja sama.

- b. Ka.Sub Bag. Humas: Mengelola hubungan publik dan penyampaian informasi dari perusahaan kepada pihak luar.
- h. Kabag. Produksi Bibit, bertanggung jawab terhadap seluruh kegiatan produksi bibit tanaman karet mulai dari pembibitan hingga distribusi ke lapangan.
  - a. Kasie Produksi Bibit: Mengawasi kegiatan produksi bibit agar sesuai standar kualitas.
  - b. Kasie Sarana & Prasarana: Menjamin ketersediaan dan pemeliharaan alat serta fasilitas produksi.
- i. Kabag. Tanam, mengatur kegiatan penanaman, pemeliharaan, dan pengawasan blok tanaman di lapangan.
  - a. Ka.Sub Blok 1–5: Mengelola masing-masing area atau blok perkebunan.
  - b. Supervisor 1–5: Mengawasi pelaksanaan kegiatan lapangan agar sesuai dengan rencana kerja.
  - c. Foreman 1–5: Melaksanakan pengawasan langsung terhadap pekerja di lapangan dan memastikan kegiatan berjalan efektif.

Struktur organisasi ini menunjukkan bahwa PT Kendi Arindo memiliki sistem manajemen yang terstruktur dengan pembagian tanggung jawab yang jelas di setiap bagian, sehingga seluruh proses dari administrasi, keuangan, hingga produksi bibit dan penanaman dapat berjalan secara efisien dan terkoordinasi.

### 3.1.6. Jadwal Penelitian

Pelaksanaan penelitian ini direncanakan berlangsung selama kurang lebih enam bulan, dimulai pada tanggal 21 Oktober 2025 hingga 21 April 2026. Selama periode tersebut, kegiatan penelitian dilaksanakan secara bertahap dan sistematis agar setiap proses dapat berjalan dengan efektif dan terarah. Tahapan penelitian mencakup penyusunan rencana penelitian, pengumpulan dan pengolahan data, analisis kebutuhan sistem, perancangan dan pengembangan aplikasi, hingga tahap evaluasi serta penyusunan laporan akhir. Adapun rincian jadwal pelaksanaan penelitian dapat dilihat pada tabel 3.1 berikut.

Tabel 3.1 Jadwal Penelitian

| No | Kegiatan                                | Okt<br>2025 | Nov<br>2025 | Des<br>2025 | Jan<br>2026 | Feb<br>2026 | Mar<br>2026 | Apr<br>2026 |
|----|-----------------------------------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| 1  | Penyusunan Rencana Penelitian           | ✓           | ✓           |             |             |             |             |             |
| 2  | Pengumpulan & Pengolahan Data           |             | ✓           | ✓           |             |             |             |             |
| 3  | Analisis Kebutuhan Sistem               |             |             | ✓           | ✓           |             |             |             |
| 4  | Perancangan & Pengembangan Sistem       |             |             |             |             | ✓           |             |             |
| 5  | Evaluasi & Pengujian Sistem             |             |             |             |             |             | ✓           |             |
| 6  | Penyusunan Laporan Akhir & Revisi Akhir |             |             |             |             |             |             | ✓           |

Sumber: data diolah oleh Penulis, 2026

### 3.2 Metode Penelitian

Metode penelitian adalah cara atau langkah-langkah sistematis yang digunakan oleh peneliti untuk memperoleh data dan informasi yang dapat digunakan dalam menjawab pertanyaan penelitian atau memecahkan suatu masalah. Metode ini berfungsi sebagai pedoman dalam pelaksanaan penelitian agar prosesnya berjalan secara ilmiah, terarah, dan dapat dipertanggung jawabkan. Melalui metode penelitian, data dikumpulkan, diolah, dan dianalisis dengan tujuan menghasilkan kesimpulan yang valid serta relevan dengan tujuan penelitian.

Metode penelitian merupakan serangkaian kegiatan dalam mencari kebenaran suatu studi penelitian, yang diawali dengan suatu pemikiran yang membentuk rumusan masalah sehingga menimbulkan hipotesis awal, dengan dibantu dan persepsi penelitian terdahulu, sehingga penelitian bisa diolah dan dianalisis yang akhirnya membentuk suatu kesimpulan (Sahir, 2022).

Pada penelitian ini, penulis memilih metode penelitian kualitatif. Metode penelitian kualitatif adalah metode penelitian yang digunakan untuk memahami fenomena secara mendalam melalui pengumpulan data deskriptif berupa kata-kata, gambar, atau perilaku, bukan angka. Penelitian kualitatif berfokus pada makna, pemahaman, serta proses yang terjadi dalam suatu situasi atau objek penelitian.

Tujuannya bukan untuk mengukur, tetapi untuk menggali dan menjelaskan bagaimana serta mengapa suatu fenomena terjadi.

### 3.3 Sumber Data

Sumber data utama dalam penelitian kualitatif ialah kata-kata, dan tindakan selebihnya adalah data tambahan seperti dokumen dan lain-lain (Dua & Rumerung, 2022). Dalam penelitian ini, sumber data merupakan komponen penting yang digunakan untuk memperoleh informasi yang relevan dengan permasalahan penelitian. Sumber data diperoleh dari dua jenis, yaitu data primer dan data sekunder, yang keduanya saling melengkapi dalam mendukung hasil penelitian.

#### 1. Data Primer

Data primer adalah data yang dikumpulkan langsung oleh peneliti untuk tujuan penelitian tertentu (Daruhadi & Sopiati, 2024). Pada penelitian ini, data primer diperoleh melalui wawancara langsung dan observasi lapangan terkait kegiatan operasional di PT Kendi Arindo.

Sumber data primer meliputi:

- a. Kepala Bagian Umum dan Administrasi, yang memberikan informasi mengenai alur pencatatan administrasi, dokumen produksi, serta permasalahan administrasi yang terjadi pada sistem manual.
- b. Kepala Bagian Perencanaan, yang memberikan data mengenai proses perencanaan kebutuhan bibit, penyusunan jadwal produksi, dan evaluasi monitoring produksi.
- c. Kepala Bagian Produksi Bibit, yang memberikan penjelasan mendalam tentang alur kerja produksi bibit di lapangan, aktivitas harian, serta kendala dalam proses pencatatan dan pelaporan.

Data primer tersebut digunakan untuk memahami kondisi nyata sistem pencatatan yang masih manual, kebutuhan pengguna internal, serta hambatan yang terjadi sebelum diterapkannya sistem digital berbasis *android*.

## 2. Data Sekunder

Data sekunder adalah data yang sudah ada, yang dikumpulkan untuk tujuan lain tetapi digunakan oleh peneliti untuk analisis lanjutan (Daruhadi & Sopiati, 2024). Dalam penelitian ini, sumber data sekunder meliputi:

a. Arsip dan dokumen internal Perusahaan.

Arsip ini mencakup laporan produksi, dokumen perencanaan bibit, serta formulir pencatatan manual yang digunakan dalam proses monitoring di PT Kendi Arindo. Dokumen tersebut memberikan informasi mengenai alur pencatatan dan pelaporan yang masih bersifat konvensional, sehingga membantu peneliti memahami kondisi sistem sebelum diterapkannya digitalisasi.

b. Literatur ilmiah dan referensi terkait.

Literatur yang digunakan meliputi buku, jurnal, artikel ilmiah, serta penelitian terdahulu yang membahas digitalisasi sistem, monitoring produksi, dan pemanfaatan aplikasi berbasis *Android*. Literatur ini menjadi landasan teoritis untuk memperkuat konsep penelitian dan mendukung analisis terhadap data primer.

Berdasarkan pemanfaatan berbagai sumber data sekunder, dapat disimpulkan bahwa data sekunder berperan penting sebagai dasar pendukung dalam penelitian ini. Melalui arsip dan dokumen internal perusahaan, peneliti memperoleh gambaran mengenai sistem pencatatan dan pelaporan produksi yang masih manual. Sementara itu, literatur ilmiah dan referensi terkait memberikan landasan teori yang kuat mengenai konsep digitalisasi dan penerapan teknologi berbasis *android*. Dengan demikian, data sekunder tidak hanya melengkapi data primer, tetapi juga memperkaya analisis serta membantu peneliti dalam memahami konteks penelitian secara lebih komprehensif.

### 3.4 Teknik Pengumpulan Data

Tahapan pengumpulan data merupakan proses untuk mengumpulkan informasi atau fakta yang relevan guna menjawab pertanyaan penelitian atau memecahkan masalah yang dihadapi (Rifa'i, 2023). Untuk memperoleh data yang

akurat dan relevan, penelitian ini menggunakan beberapa metode pengumpulan data, yaitu observasi, wawancara, dokumentasi, dan studi literatur.

Tahapan pengumpulan data ini dilakukan secara sistematis untuk memastikan bahwa informasi yang diperoleh benar-benar mencerminkan kondisi nyata di lapangan. Setiap metode yang digunakan memiliki peran masing-masing dalam melengkapi data penelitian, mulai dari pengamatan langsung terhadap proses produksi, wawancara dengan pihak terkait, pengumpulan dokumen pendukung, hingga penelusuran referensi ilmiah. Dengan kombinasi berbagai metode tersebut, penelitian dapat menghasilkan data yang lebih komprehensif, objektif, dan dapat dipertanggungjawabkan sebagai dasar dalam proses analisis dan perumusan solusi yang sesuai dengan permasalahan yang terjadi.

#### **3.4.1. Observasi**

Observasi adalah merupakan teknik pengumpulan data yang dilakukan melalui sesuatu pengamatan, dengan disertai pencatatan-pencatatan terhadap keadaan atau perilaku objek sasaran. Observasi adalah pengamatan dan pencatatan yang sistematis terhadap gejala-gejala yang diteliti (Hasibuan & Azmi, 2023).

Observasi dilakukan dengan mengamati langsung proses *monitoring* data produksi bibit dan tanaman karet di PT Kendi Arindo. Tujuannya adalah untuk mengetahui kondisi nyata kegiatan *monitoring*, meliputi metode pencatatan, pelaporan, serta efisiensi kerja di lapangan. Dari hasil observasi ditemukan bahwa proses pencatatan dan pelaporan masih dilakukan secara manual, sehingga kurang efisien, memerlukan waktu lama, dan berpotensi menimbulkan kesalahan data. Temuan ini menjadi dasar dalam pengembangan sistem digitalisasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android*, agar proses pengumpulan dan pelaporan data dapat dilakukan secara lebih cepat, akurat, dan terintegrasi.

#### **3.4.2. Wawancara**

Wawancara adalah proses tanya jawab lisan antara dua orang atau lebih untuk mengetahui tanggapan dan pendapat seseorang terhadap suatu objek. Wawancara

merupakan cara yang paling tepat untuk mengungkapkan keadaan pribadi dari orang yang diwawancarai (Irawati, 2021).

Wawancara dilakukan dengan karyawan di PT Kendi Arindo sebagai informan utama. Tujuannya adalah untuk memperoleh informasi mendalam mengenai proses *monitoring* data produksi bibit dan tanaman karet, kendala yang dihadapi dalam pencatatan dan pelaporan manual, serta kebutuhan akan sistem yang lebih efisien dan terintegrasi. Hasil wawancara ini menjadi dasar dalam perancangan fitur dan fungsi pada sistem digitalisasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android*, agar sesuai dengan kebutuhan dan kondisi nyata pengguna di lapangan.

### 3.4.3. Dokumentasi

Dokumentasi adalah metode mencari data mengenai hal-hal yang berupa catatan, buku, transkrip, surat kabar, prasasti, majalah, notulen rapat, agenda serta foto-foto kegiatan (Saádi Ahmad, 2025). Dokumentasi dilakukan dengan mengumpulkan berbagai dokumen dan data pendukung yang berkaitan dengan proses *monitoring* data produksi bibit dan tanaman karet di PT Kendi Arindo. Dokumen yang dikumpulkan meliputi data bibit dan tanaman, laporan hasil produksi, serta arsip kegiatan *monitoring* yang relevan.

Data tersebut kemudian dianalisis untuk memahami alur kerja, kebutuhan pencatatan, serta kendala yang sering muncul dalam pengelolaan data di lapangan. Teknik dokumentasi ini membantu peneliti dalam menyesuaikan rancangan sistem digitalisasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android* agar lebih efektif, akurat, dan sesuai dengan kebutuhan operasional perusahaan.

### 3.4.4. Studi Literatur

Studi literatur atau studi kepustakaan adalah metode pengumpulan data yang melibatkan analisis dan penelaahan terhadap buku-buku, literatur-literatur, catatan-catatan, dan laporan-laporan yang relevan dengan masalah yang sedang di investigasikan (Habsy et al., 2023). Studi literatur dilakukan dengan menelaah berbagai sumber seperti jurnal, buku, artikel ilmiah, dan penelitian terdahulu yang

berkaitan dengan digitalisasi sistem *monitoring*, manajemen data produksi, serta penerapan teknologi berbasis *android*.

Tujuannya adalah untuk memperoleh landasan teori yang kuat dalam pengembangan digitalisasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android*, serta memahami metode dan teknologi yang tepat untuk meningkatkan efisiensi, akurasi, dan integrasi proses *monitoring* produksi di PT Kendi Arindo.

### 3.5 Metode Pengembangan Sistem

Metode pengembangan sistem secara umum diartikan sebagai urutan langkah-langkah yang terstruktur untuk mengembangkan sebuah sistem informasi berbasis computer (Faithullah Akbar, 2023). Dalam penelitian ini digunakan metode pengembangan sistem *design thinking*, yaitu metode pengembangan yang berfokus pada pemahaman kebutuhan pengguna serta menghasilkan solusi yang kreatif dan inovatif melalui beberapa tahap yang saling berhubungan. Metode ini dipilih karena dinilai paling sesuai untuk pengembangan sistem digitalisasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android*, yang menuntut solusi inovatif dan adaptif terhadap kondisi lapangan di PT. Kendi Arindo.

Melalui tahapan yang iteratif, *design thinking* memungkinkan proses perancangan, pengujian, dan penyempurnaan sistem dilakukan secara lebih fleksibel berdasarkan masukan langsung dari pengguna. *Design thinking* adalah pendekatan yang melibatkan kolaborasi dengan pengguna untuk menemukan solusi bagi suatu masalah. Tujuan utamanya adalah menciptakan layanan yang inovatif, sesuai dengan kebutuhan pengguna, dan dapat mengatasi masalah yang ada (Ansori & Hendradi, 2023).

Adapun alasan peneliti menggunakan metode *design thinking* adalah sebagai berikut:

1. Berpusat pada pengguna (*User-Centered*).

Metode ini memungkinkan peneliti untuk memahami kebutuhan dan kendala pengguna secara langsung, khususnya petugas lapangan di PT. Kendi Arindo, sehingga solusi yang dikembangkan benar-benar sesuai dengan kondisi di lapangan.

2. Fleksibel dan Iteratif.

Setiap tahap dalam *Design Thinking* dapat diulang berdasarkan hasil uji coba dan umpan balik pengguna, sehingga sistem dapat terus disempurnakan hingga mencapai hasil yang optimal.

3. Mendorong kreativitas dan inovasi.

Tahapan *Ideate* memberikan ruang bagi peneliti untuk mengeksplorasi berbagai ide dan alternatif solusi digital dalam merancang sistem digitalisasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android*.

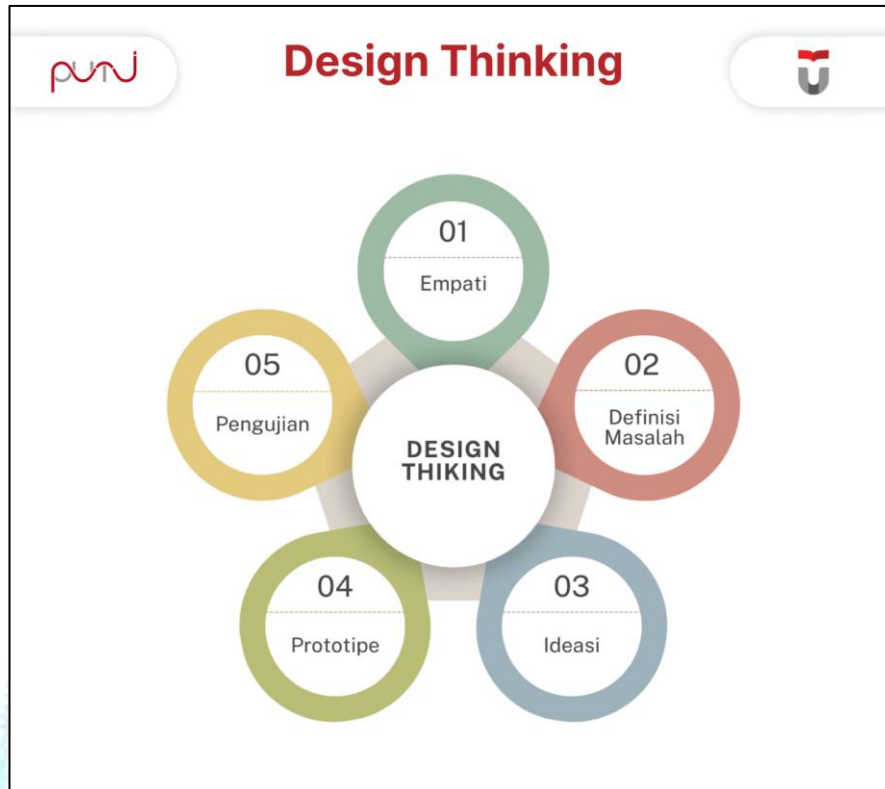
4. Efisien dalam identifikasi masalah.

Melalui tahap *Empathize* dan *Define*, peneliti dapat menemukan akar permasalahan secara lebih akurat, sehingga pengembangan sistem menjadi lebih fokus dan terarah.

5. Meningkatkan keterlibatan pengguna.

Metode ini melibatkan pengguna dalam proses pengembangan melalui uji coba dan umpan balik pada tahap *Prototype* dan *Test*, sehingga sistem yang dihasilkan lebih mudah diterima dan digunakan dalam aktivitas operasional perusahaan.

Berikut adalah gambar 3.2 yang menunjukkan tahapan dalam metode *Design Thinking* yang digunakan dalam penelitian ini. Metode ini terdiri dari lima tahap utama, yaitu *Empathize*, *Define*, *Ideate*, *Prototype*, dan *Test*. Setiap tahap saling berhubungan dan membentuk proses pengembangan yang berfokus pada kebutuhan pengguna. Dengan mengikuti tahapan ini, sistem digitalisasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android* diharapkan dapat dikembangkan secara lebih efektif, inovatif, dan sesuai dengan permasalahan yang dihadapi di lapangan.



Sumber: Meilinaeka, 2024

Gambar 3.2 Metode Pengembangan Sistem *Design Thinking*

Dari gambar 3.2 tentang tahapan metode pengembangan *design thinking* dapat dijelaskan dan disimpulkan sebagai berikut :

1. *Empathize* (Empati)

*Empathize* (Empati) adalah tahap untuk memahami kebutuhan, kendala, dan kondisi pengguna secara langsung melalui observasi, wawancara, dan studi lapangan (Ar Rizqi et al., 2023). Pada tahap ini dilakukan pengumpulan informasi di PT. Kendi Arindo untuk mengetahui secara mendalam bagaimana proses *monitoring* informasi data produksi bibit tanaman karet dilakukan. Hasil dari tahap empati menunjukkan adanya beberapa permasalahan, seperti pencatatan manual yang kurang efisien, sering terjadi kesalahan input data, serta keterlambatan pelaporan yang berdampak pada proses *monitoring* di lapangan.

2. *Define* (Definisi Masalah)

*Define* adalah tahap untuk merumuskan inti permasalahan berdasarkan informasi yang telah diperoleh pada tahap *empathize* (Ar Rizqi et al., 2023).

Pada tahap ini, peneliti mengidentifikasi permasalahan utama yang dihadapi pengguna, yaitu belum adanya sistem digital yang mampu mengintegrasikan data produksi bibit dan tanaman karet secara *real-time*. Kondisi ini menyebabkan proses *monitoring* menjadi kurang efektif dan sering menimbulkan kesenjangan informasi. Hasil perumusan masalah ini menjadi dasar dalam menentukan arah dan fokus solusi yang akan dikembangkan.

3. *Ideate* (Ideasi)

*Ideate* adalah tahap menghasilkan berbagai ide dan solusi kreatif untuk mengatasi permasalahan yang telah didefinisikan (Ar Rizqi et al., 2023). Pada tahap ini, peneliti mengembangkan konsep sistem digitalisasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android*. Ide utama yang dihasilkan meliputi fitur input data bibit dan tanaman secara digital, proses rekap otomatis, serta penyajian laporan produksi yang dapat dengan mudah diakses oleh petugas lapangan maupun admin.

4. *Prototype* (Prototipe)

*Prototype* adalah tahap pembuatan rancangan awal atau model sederhana dari aplikasi yang berfungsi memvisualisasikan ide solusi sebelum dikembangkan secara penuh (Ar Rizqi et al., 2023). Pada tahap ini dibuat prototipe yang menggambarkan struktur tampilan, alur kerja, serta fungsi utama aplikasi. Prototipe disusun menggunakan desain antarmuka sederhana agar pengguna dapat memberikan masukan secara langsung sebelum sistem diimplementasikan pada tahap pengembangan selanjutnya.

5. *Test* (Pengujian)

*Test* adalah tahap untuk menguji prototipe guna memastikan bahwa solusi yang dibangun benar-benar sesuai dengan kebutuhan pengguna dan mudah digunakan (Ar Rizqi et al., 2023). Pengujian dilakukan oleh pengguna di lapangan untuk menilai apakah alur pencatatan, tampilan, dan fitur aplikasi sudah memenuhi kebutuhan kerja mereka. Berdasarkan hasil pengujian tersebut, dilakukan revisi dan penyempurnaan agar aplikasi dapat berfungsi secara optimal dalam mendukung proses *monitoring* informasi data produksi bibit tanaman karet di PT. Kendi Arindo.

### 3.6 Alat Bantu Perancangan

Perancangan sistem adalah strategi untuk memecahkan masalah dan mengembangkan solusi terbaik bagi permasalahan itu (Wayahdi & Ruziq, 2023). Alat bantu perancangan sistem pada penelitian ini berperan penting dalam membantu peneliti merancang, menggambarkan, serta mengimplementasikan sistem. Dalam penelitian ini, peneliti menggunakan beberapa alat bantu, baik untuk perancangan sistem, desain antarmuka, maupun proses *debugging* dan pengembangan. Berikut merupakan uraian dari alat bantu yang digunakan:

#### 3.6.1. Alat Bantu Utama

*Visual studio code* adalah editor kode sumber terbuka dan gratis yang dikembangkan oleh Microsoft. *Visual studio code* mendukung berbagai bahasa pemrograman dan memiliki fitur canggih seperti *debugging*, *IntelliSense* (saran kode otomatis), integrasi Git, dan ekstensi untuk meningkatkan fungsionalitas editor. *Visual studio code* sangat populer di kalangan pengembang karena kecepatan, fleksibilitas, dan kemampuan untuk dikustomisasi sesuai kebutuhan proyek atau preferensi pengguna (Budiarti & Juleha, 2025).

*Visual studio code* didasarkan pada keunggulannya yang ringan, fleksibel, serta mendukung berbagai bahasa pemrograman dan *framework modern*, sehingga sangat sesuai digunakan dalam pengembangan aplikasi *monitoring* berbasis *android*.

Flutter digunakan sebagai *framework* utama dalam pembuatan aplikasi ini. *Framework* ini berperan dalam membangun tampilan antarmuka pengguna (*User Interface*) sekaligus mengatur interaksi antara pengguna dan sistem. Dengan pendekatan *single codebase*, flutter memungkinkan pengembang membuat aplikasi yang responsif, modern, dan memiliki performa tinggi pada perangkat *android*. Selain itu, fitur *hot reload* yang dimilikinya membantu mempercepat proses pengujian dan perbaikan aplikasi, sehingga pengembangan sistem digitalisasi *monitoring* informasi data produksi bibit tanaman karet dapat dilakukan secara lebih efisien, interaktif, dan mudah digunakan oleh staf lapangan maupun pihak manajemen perusahaan.

### 3.6.2. Alat Bantu Pemodelan Visual

Pemodelan adalah penggambaran dari realita yang simple dan dituangkan dalam bentuk pemetaan dengan aturan tertentu. Pemodelan juga merupakan juga merupakan suatu model yang digunakan untuk menguraikan sistem menjadi bagian-bagian yang dapat diatur dan mengkomunikasikan ciri konseptual dan fungsional kepada pengamat (Farhany, 2021).

Bahasa pemodelan visual digunakan untuk membantu menggambarkan struktur dan alur kerja aplikasi secara konseptual sebelum tahap implementasi dilakukan. Melalui pemodelan visual, pengembang dapat memahami hubungan antar komponen sistem, aliran proses *monitoring* dan pengelolaan data produksi bibit tanaman karet, serta interaksi antara pengguna (seperti staf lapangan, bagian produksi, dan manajemen) dengan sistem secara lebih jelas dan terstruktur.

Selain itu, diagram visual juga berperan sebagai sarana komunikasi antara pengembang, pembimbing, dan pihak perusahaan agar rancangan aplikasi dapat dipahami dengan baik sebelum proses pengkodean dimulai. Dengan adanya pemodelan visual ini, pengembangan aplikasi menjadi lebih terarah, efisien, serta meminimalkan risiko kesalahan pada tahap implementasi.

Adapun alat bantu pemodelan visual yang digunakan dalam penelitian ini meliputi:

1. *Unified Modeling Language (UML)*

*UML* adalah bahasa standar berdasarkan diagram atau gambar yang bertujuan untuk memvisualisasikan, mendefinisikan dan membangun model. *UML* salah alat yang sangat handal dalam Pengembangan sistem berorientasi objek karena *UML* menyediakan bahasa pemodelan visual memungkinkan pengembang sistem untuk membuat cetak biru untuk visi mereka dalam bentuk standar. *Unified Modeling Language(UML)* adalah bahasa spesifikasi standar yang digunakan untuk mendokumentasikan, menentukan, dan membuat perangkat lunak (Haris & Sitanggang, 2023).

2. *Use Case Diagram*

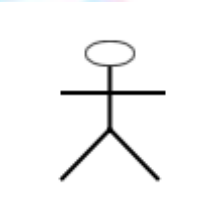
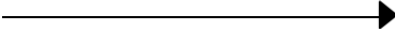
*Use case diagram* merupakan salah satu alat bantu (*tools*) yang digunakan dalam proses pengembangan sistem untuk menggambarkan hubungan antara aktor

dengan sistem secara visual. Diagram ini berfungsi untuk menunjukkan bagaimana pengguna berinteraksi dengan sistem serta fitur-fitur utama yang dapat diakses oleh masing-masing aktor.

*Use case diagram* merupakan *tools* untuk mengembangkan sistem dengan menjelaskan hubungan antara *actor* yang terlibat dalam sistem (Wayahdi & Ruziq, 2023).

Berikut adalah simbol-simbol *use case diagram* dapat dilihat pada tabel 3.1 berikut :

Tabel 3.2 *Use Case*

| Simbol                                                                                                                      | Deskripsi                                                                                                                                                  |
|-----------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <br><i>UseCase</i>                         | Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit dan aktor.                                                   |
| <br><b>Aktor/Actor</b>                   | Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi.                                        |
| <b>Asosiasi/association</b><br>          | Komunikasi antar aktor dan <i>Use Case</i> yang berpartisipasi.                                                                                            |
| <b>Ekstensi/extend</b><br><<extend>><br> | Relasi <i>Use Case</i> tambahan ke sebuah <i>Use Case</i> dimana <i>Use Case</i> yang ditambah dapat berdiri sendiri walau tanpa <i>Use Case</i> tambahan. |
| <b>Generalisasi/generalization</b><br>   | Hubungan generalisasi dan spesialisasi antara dua buah <i>Use Case</i>                                                                                     |

|                                                           |                                                                                                                                                               |
|-----------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                                                           | yang mana fungsi yang satu lebih umum dari yang lainnya.                                                                                                      |
| <b>Include/Use Case</b><br><b>&lt;&lt;include&gt;&gt;</b> | Relasi <i>Use Case</i> tambahan ke sebuah <i>Use Case</i> dimana <i>Use Case</i> yang ditambahkan memerlukan <i>Use Case</i> ini untuk menjalankan fungsinya. |
| <b>Uses</b>                                               |                                                                                                                                                               |

Sumber : Wijaya et al., 2024

### 3. Class Diagram

*Class diagram* adalah diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. *Class* adalah kumpulan objek-objek dengan dan yang mempunyai struktur umum, *behavior* umum, relasi umum, dan *semantic*/kata yang umum (Malius, 2021).

Berikut adalah simbol-simbol *class diagram* dapat dilihat pada tabel 3.2 berikut:

Tabel 3.3. *Class Diagram*

| Nama              | Keterangan                                                                                                                                                                                                                                                    | Simbol                                                                                                                                                                                            |
|-------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>Dependency</i> | Penggunaan <i>dependency</i> digunakan untuk menunjukkan operasi pada suatu <i>class</i> yang menggunakan <i>class</i> yang lain.                                                                                                                             |                                                                                                              |
| <i>Class</i>      | <i>Class</i> adalah blok-blok pembangun pada pemrograman berorientasi objek. <i>class</i> digambarkan sebagai sebuah kotak yang terbagi atas 3 bagian. Bagian atas adalah bagian nama dari <i>class</i> . Bagian tengah mendefinisikan atribut <i>class</i> . | <div style="border: 1px solid black; padding: 10px; border-radius: 10px;"> <p><b>Nama <i>Class</i></b></p> <p>+ atribut</p> <p>+ atribut</p> <p>- <i>method</i></p> <p>- <i>method</i></p> </div> |

|                    |                                                                                                                                                                                                                                                                                                                       |                                                                                       |
|--------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------|
|                    | Bagian akhir mendefinisikan <i>method-method</i>                                                                                                                                                                                                                                                                      |                                                                                       |
| <i>Association</i> | Sebuah asosiasi merupakan sebuah <i>relationship</i> paling umum antara 2 <i>class</i> dan dilambangkan oleh sebuah garis yang menghubungkan antara 2 <i>class</i> . Garis ini bisa melambangkan tipe-tipe <i>relationship</i> dan juga dapat menampilkan hukum - hukum multiplisitas pada sebuah <i>relationship</i> | L.n Owned by 1                                                                        |
| <i>Composition</i> | Jika sebuah <i>class</i> tidak bisa berdiri sendiri dan harus merupakan bagian dari <i>class</i> yang lain, maka <i>class</i> tersebut memiliki relasi <i>Composition</i> terhadap class tempat dia bergantung tersebut.                                                                                              |  |

Sumber : Malius, 2021

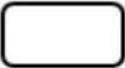
#### 4. Activity Diagram

*Activity diagram* merupakan salah satu yang paling banyak digunakan diantara semua diagram UML. Alur *activity diagram* dapat digunakan untuk mengontrol urutan tindakan yang dieksekusi atau untuk merepresentasikan komunikasi data. Biasanya diagram ini digunakan untuk desain tingkat tinggi seperti memodelkan proses bisnis, dan desain tingkat rendah seperti untuk mendeskripsikan algoritma yang akan diimplementasikan. (Wayahdi & Ruziq, 2023)

*Activity diagram* menunjukkan adanya alur kerja atau aktivitas yang berjalan dari suatu sistem. Diagram ini berfokus pada aliran kontrol (*control flow*) dari satu aktivitas ke aktivitas lainnya, berupa sebuah proses yang didorong oleh *internal*

*processing*. Diagram aktivitas bersifat hierarkis, maksudnya adalah suatu aktivitas yang dibuat dari tindakan atau grafik sub-aktivitas dan aliran objek yang terkait.(Wayahdi & Ruziq, 2023)

Berikut adalah simbol-simbol activity diagram dapat dilihat pada gambar 3.3 berikut:

| Simbol                                                                              | Nama                   | Keterangan                                                                                   |
|-------------------------------------------------------------------------------------|------------------------|----------------------------------------------------------------------------------------------|
|    | Status awal            | Sebuah diagram aktivitas memiliki sebuah status awal.                                        |
|    | Aktivitas              | Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.               |
|    | Percabangan / Decision | Percabangan dimana ada pilihan aktivitas yang lebih dari satu.                               |
|  | Penggabungan / Join    | Penggabungan dimana yang mana lebih dari satu aktivitas lalu digabungkan jadi satu.          |
|  | Status Akhir           | Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir    |
|  | Swimlane               | Swimlane memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi |

Sumber : Wayahdi & Ruziq, 2023

Gambar 3.3 Activity Diagram

### 3.6.3. Alat Bantu Desain

Dalam proses pengembangan aplikasi digitalisasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android*, tahap perancangan memiliki peranan penting dalam mengubah kebutuhan pengguna menjadi rancangan visual dan fungsional dari sistem yang akan dibangun. Tahap ini mencakup pembuatan tata

letak tampilan, penentuan elemen antarmuka, pemilihan warna dan ikon, serta penyusunan alur interaksi antara pengguna dengan aplikasi. Proses perancangan dilakukan sebelum tahap implementasi agar pengembang memiliki panduan yang jelas dalam membangun tampilan dan fitur aplikasi sesuai kebutuhan pengguna di lapangan.

Figma digunakan sebagai alat bantu perancangan antarmuka pengguna (*User Interface Design Tool*) pada tahap ini. Melalui figma, peneliti dapat membuat rancangan visual berupa tata letak halaman, posisi elemen, ikon, warna, dan alur navigasi aplikasi sebelum diimplementasikan. Penggunaan figma bertujuan untuk mempermudah proses perancangan tampilan agar hasil desain sesuai dengan kebutuhan pengguna serta mengikuti prinsip *user experience* (UX) yang baik. Selain itu, figma mendukung kolaborasi secara real-time dan penyimpanan berbasis *cloud*, sehingga proses revisi desain dapat dilakukan secara lebih cepat dan efisien sebelum dikembangkan menjadi aplikasi berbasis *android* pada tahap implementasi.

#### 3.6.4. Alat Bantu *Debugging* dan Pengembangan

Dalam proses pengembangan aplikasi digitalisasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android*, tahap *debugging* memiliki peranan penting untuk memastikan bahwa sistem berjalan sesuai dengan rancangan dan bebas dari kesalahan (*error*). Kegiatan ini bertujuan untuk menemukan, menganalisis, serta memperbaiki kesalahan pada kode program yang dapat memengaruhi tampilan, alur kerja, maupun kinerja aplikasi secara keseluruhan.

Pada penelitian ini, proses *debugging* dilakukan menggunakan *visual studio code* dengan memanfaatkan fitur *debugger* yang terintegrasi dengan *flutter*. Pengujian dilakukan baik melalui emulator *android* maupun secara langsung pada perangkat *smartphone* guna memastikan kestabilan, kecepatan, dan fungsionalitas aplikasi dalam kondisi penggunaan nyata.

Melalui proses ini, pengembang dapat memantau jalannya program secara *real-time*, mengidentifikasi bagian kode yang mengalami kendala, serta melakukan perbaikan secara cepat dengan memanfaatkan fitur *hot reload* yang disediakan oleh *flutter*. Dengan demikian, proses pengujian dan penyempurnaan aplikasi dapat

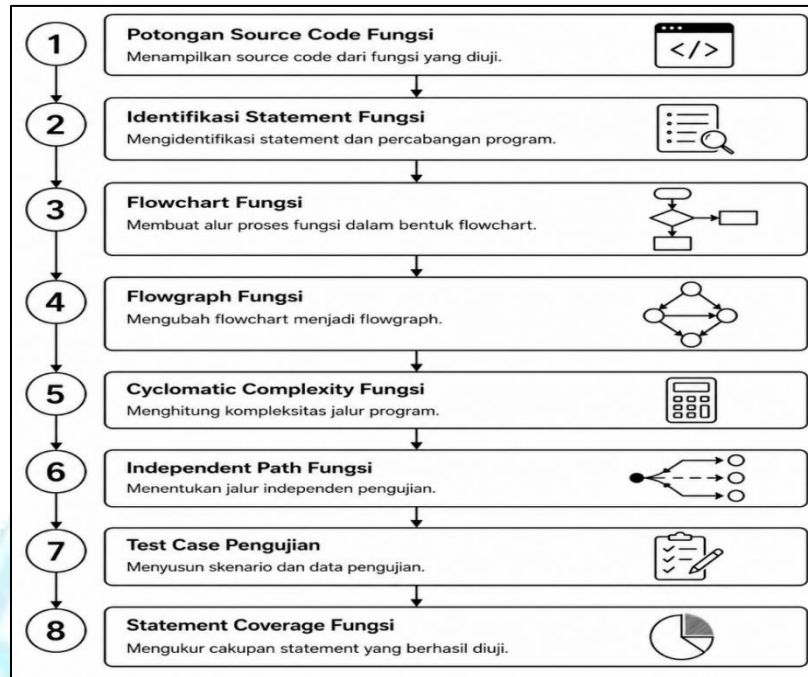
dilakukan secara lebih efisien, terarah, serta menghasilkan sistem yang sesuai dengan kebutuhan digitalisasi *monitoring* dan informasi data produksi bibit tanaman karet di PT. Kendi Arindo.

### 3.7 Pengujian Sistem

Pengujian sistem merupakan tahap penting dalam proses pengembangan aplikasi untuk memastikan bahwa seluruh fungsi dan logika program berjalan sesuai dengan kebutuhan serta bebas dari kesalahan (*error*). Tahap ini dilakukan untuk mengevaluasi keandalan, efisiensi, dan stabilitas sistem sebelum diterapkan secara penuh di lingkungan operasional.

Pada penelitian ini, pengujian dilakukan menggunakan metode *white-box testing*, yaitu metode pengujian yang berfokus pada pemeriksaan struktur internal dan alur logika program. Pengujian ini bertujuan untuk memastikan bahwa setiap fungsi dan perintah dalam kode program bekerja sesuai dengan rancangan serta menghasilkan keluaran yang benar. *White-box testing* menguji lebih dalam dari *black-box testing* jadi mengapa peneliti memilih metode ini dan menggunakan uji unit. Karena ini menguji lebih dalam dan hasilnya dapat memeriksa lebih kompleks dengan pemrograman dan proses yang berjalan (Wintana et al., 2022).

Secara umum, proses pengujian *white-box testing* dilakukan melalui beberapa tahapan yang terstruktur, dimulai dari pengumpulan kebutuhan dan analisis *source code*, penyusunan rencana pengujian, pembuatan test case dan test data, hingga pelaksanaan pengujian dan perbaikan kesalahan yang ditemukan. Alur umum pengujian *white-box testing* tersebut dapat dilihat pada Gambar 3.4 dibawah ini.



Sumber : Solisa et al, 2025

Gambar 3.3 *White box Testing Process*

Berdasarkan Gambar 3.3 di atas, alur pengujian *white-box testing* dilakukan melalui beberapa tahapan yang dapat dijelaskan sebagai berikut:

1. Potongan *Source Code* Fungsi

Pada tahap ini, penguji menampilkan potongan *source code* dari fungsi yang akan diuji. Tujuan dari tahap ini adalah untuk mengetahui struktur kode program, proses yang dijalankan, serta logika yang digunakan pada fungsi tersebut sehingga pengujian dapat dilakukan secara lebih terarah.

2. Identifikasi *Statement* Fungsi

Tahap selanjutnya adalah mengidentifikasi *statement* yang terdapat pada fungsi program, seperti percabangan, kondisi, dan proses yang dijalankan sistem. Identifikasi ini bertujuan untuk mengetahui bagian-bagian kode yang perlu diuji pada proses *white-box testing*.

3. *Flowchart* Fungsi

Pada tahap ini, *source code* fungsi diubah ke dalam bentuk *flowchart*. *Flowchart* digunakan untuk menggambarkan alur proses program secara visual sehingga urutan proses, percabangan, dan keputusan pada fungsi dapat dipahami dengan lebih mudah.

#### 4. *Flowgraph* Fungsi

Setelah *flowchart* dibuat, tahap berikutnya adalah mengubah *flowchart* menjadi *flowgraph*. *Flowgraph* digunakan untuk merepresentasikan hubungan antar *node* dan *edge* pada alur logika program sehingga proses analisis jalur program dapat dilakukan dengan lebih jelas.

#### 5. *Cyclomatic Complexity* Fungsi

Tahap selanjutnya adalah menghitung nilai *cyclomatic complexity* berdasarkan *flowgraph* yang telah dibuat. Nilai *cyclomatic complexity* digunakan untuk mengetahui tingkat kompleksitas logika program serta menentukan jumlah jalur independen yang harus diuji pada fungsi tersebut.

#### 6. *Independent Path* Fungsi

Pada tahap ini, penguji menentukan jalur independen (*independent path*) berdasarkan hasil *flowgraph* dan *cyclomatic complexity*. Jalur independen merupakan jalur logika unik yang harus dilalui dalam proses pengujian agar seluruh alur program dapat diuji secara menyeluruh.

#### 7. *Test Case* Pengujian

Setelah jalur independen diperoleh, penguji menyusun *test case* pengujian berdasarkan jalur-jalur tersebut. *Test case* digunakan untuk memastikan bahwa setiap proses, kondisi, dan percabangan pada fungsi dapat berjalan sesuai dengan hasil yang diharapkan.

#### 8. *Statement Coverage* Fungsi

Tahap terakhir adalah melakukan *statement coverage* untuk mengetahui tingkat cakupan *statement* yang berhasil diuji. Pengujian ini bertujuan untuk memastikan bahwa seluruh *statement* pada fungsi program telah dieksekusi dan diuji dengan baik.

Berdasarkan tahapan tersebut, dapat disimpulkan bahwa *white-box testing* merupakan metode pengujian yang dilakukan secara sistematis dan terstruktur dengan menitikberatkan pada analisis *source code* serta alur logika program. Setiap tahapan saling berkaitan, dimulai dari potongan *source code* fungsi hingga *statement coverage*. Melalui proses ini, pengujian dapat memastikan bahwa seluruh jalur logika program telah diuji dengan baik sehingga kesalahan pada program dapat dideteksi lebih awal dan kualitas sistem menjadi lebih optimal.

## BAB IV

### ANALISA DAN PERANCANGAN

#### 4.1. Analisa Sistem

##### 4.1.1. Analisa Dan Evaluasi Sistem Berjalan

###### a. Analisa sistem berjalan

Analisa sistem berjalan dilakukan untuk mengetahui kondisi nyata sistem *monitoring* dan pengelolaan informasi data produksi bibit tanaman karet yang saat ini diterapkan di PT. Kendi Arindo. Berdasarkan hasil observasi serta studi terhadap proses pengelolaan data yang berlangsung di lapangan, diketahui bahwa proses pencatatan data produksi bibit masih dilakukan secara manual menggunakan dokumen tertulis atau buku pencatatan. Data yang telah dicatat kemudian direkap kembali secara manual untuk dijadikan laporan kepada pihak manajemen.

Kondisi tersebut menimbulkan beberapa permasalahan dalam proses *monitoring* produksi bibit tanaman karet. Salah satu permasalahan utama adalah keterlambatan dalam penyampaian informasi data produksi. Hal ini terjadi karena proses pencatatan dan rekapitulasi data membutuhkan waktu yang cukup lama sebelum informasi dapat disampaikan kepada pihak yang membutuhkan. Selain itu, penggunaan dokumen fisik juga berpotensi menyebabkan kesalahan pencatatan serta risiko kehilangan atau kerusakan data.

Dari sisi kelemahan sistem, dapat diketahui bahwa sistem yang berjalan saat ini belum memanfaatkan perkembangan teknologi informasi secara optimal, khususnya teknologi berbasis *mobile* yang dapat mempermudah proses pencatatan dan pemantauan data produksi di lapangan. Proses *monitoring* yang dilakukan masih bersifat konvensional sehingga menyulitkan pihak manajemen dalam memperoleh informasi secara cepat dan akurat. Selain itu, sistem yang ada belum mampu menyediakan akses data secara *real-time* yang dapat digunakan untuk mendukung proses pengambilan keputusan secara efektif.

Berdasarkan permasalahan dan kelemahan sistem tersebut, diperlukan suatu sistem yang mampu memberikan solusi berupa aplikasi digital berbasis *android*

yang dapat digunakan untuk melakukan *monitoring* informasi data produksi bibit tanaman karet secara lebih efektif dan efisien. Sistem yang dikembangkan diharapkan mampu mempermudah proses pencatatan data produksi di lapangan, mempercepat penyampaian informasi kepada pihak manajemen, serta mengurangi potensi kesalahan pencatatan yang sering terjadi pada sistem manual.

Dengan adanya sistem digital berbasis *android* ini, proses *monitoring* produksi bibit tanaman karet dapat dilakukan secara lebih terstruktur dan terintegrasi. Data yang diinput oleh petugas di lapangan dapat langsung tersimpan dalam sistem dan dapat diakses oleh pihak terkait secara *real-time*. Hal ini diharapkan mampu meningkatkan efisiensi pengelolaan data produksi, mempermudah proses pengawasan, serta mendukung peningkatan produktivitas kerja di lingkungan PT. Kendi Arindo.

#### **b. Evaluasi sistem berjalan**

Evaluasi sistem berjalan dilakukan untuk menilai efektivitas sistem *monitoring* dan pengelolaan informasi data produksi bibit tanaman karet yang saat ini diterapkan di PT. Kendi Arindo. Evaluasi ini didasarkan pada hasil analisa sistem berjalan yang menunjukkan bahwa proses pencatatan dan penyampaian informasi data produksi masih dilakukan secara manual dan belum memanfaatkan teknologi informasi secara optimal.

Berdasarkan kondisi yang ada, sistem yang berjalan saat ini belum mampu memberikan informasi secara cepat dan akurat kepada pihak yang membutuhkan. Proses pencatatan yang masih menggunakan dokumen fisik menyebabkan data harus direkap kembali sebelum dijadikan laporan, sehingga membutuhkan waktu yang cukup lama.

Selain itu, penggunaan media pencatatan manual juga memiliki keterbatasan dalam hal penyimpanan dan pengelolaan data, serta berpotensi menimbulkan kesalahan pencatatan maupun kehilangan data.

Kondisi tersebut juga menyebabkan proses *monitoring* produksi bibit tanaman karet menjadi kurang efektif. Pihak manajemen tidak dapat memperoleh informasi produksi secara langsung karena data yang tersedia masih bersifat

tertunda dan belum terintegrasi dalam satu sistem yang dapat diakses secara *real-time*. Hal ini tentu dapat mempengaruhi proses pengawasan dan pengambilan keputusan yang berkaitan dengan pengelolaan produksi bibit tanaman karet.

Berdasarkan hasil evaluasi tersebut, solusi yang ditawarkan dalam penelitian ini adalah pengembangan sistem digitalisasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android*. Sistem ini dirancang untuk mempermudah proses pencatatan data produksi di lapangan, menyediakan akses informasi secara cepat dan *real-time*, serta meminimalkan kesalahan dalam pengolahan data. Dengan adanya sistem ini, diharapkan proses *monitoring* produksi dapat berjalan lebih efektif, efisien, dan mampu mendukung peningkatan kinerja operasional di lingkungan PT. Kendi Arindo.

Analisa Kebutuhan :

### **1. Kebutuhan Pengguna ( *user* )**

Kebutuhan pengguna dalam sistem ini mengacu pada pihak-pihak yang akan berinteraksi langsung dengan aplikasi digitalisasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android*. Berdasarkan hasil analisis kebutuhan, pengguna sistem terdiri dari dua kategori utama, yaitu petugas lapangan sebagai pengguna utama sistem dan administrator (admin) dari pihak perusahaan.

Petugas lapangan merupakan pengguna utama aplikasi yang memanfaatkan sistem untuk melakukan pencatatan serta pelaporan data produksi bibit tanaman karet secara langsung di lokasi kerja. Melalui aplikasi ini, petugas dapat memasukkan data terkait jumlah produksi bibit, kondisi bibit, serta informasi lain yang berkaitan dengan proses pengelolaan bibit tanaman karet.

Data yang telah diinput kemudian akan tersimpan secara otomatis dalam sistem sehingga dapat diakses oleh pihak yang berkepentingan. Pengguna dari kategori ini memiliki latar belakang kemampuan teknologi yang beragam, sehingga sistem harus dirancang dengan tampilan antarmuka yang sederhana, mudah dipahami, serta mudah digunakan tanpa memerlukan keterampilan teknis yang kompleks.

Selain itu, terdapat administrator (admin) dari pihak perusahaan yang bertugas mengelola sistem secara keseluruhan. Admin memiliki peran dalam

mengelola data pengguna, memantau data produksi bibit yang telah diinput oleh petugas lapangan, serta melakukan pengolahan dan pengelompokan data untuk keperluan pelaporan kepada pihak manajemen. Melalui sistem ini, admin juga dapat melakukan pengecekan dan pengawasan terhadap data yang masuk sehingga proses *monitoring* produksi bibit tanaman karet dapat berjalan dengan lebih terstruktur dan terkontrol.

Dengan adanya pembagian peran pengguna tersebut, sistem yang dikembangkan diharapkan mampu mendukung proses pencatatan, pengelolaan, serta *monitoring* data produksi bibit tanaman karet secara lebih efektif dan efisien di lingkungan PT. Kendi Arindo.

## **2. Kebutuhan Data :**

Dalam pengembangan sistem digitalisasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android*, sistem membutuhkan beberapa jenis data yang mendukung operasional aplikasi. Data yang dibutuhkan meliputi:

### **1. Data identitas petugas lapangan.**

Digunakan sebagai identitas pengguna yang melakukan pencatatan data produksi bibit tanaman karet di lapangan. Data ini berfungsi untuk mengetahui siapa petugas yang melakukan input data, memudahkan proses pemantauan aktivitas pengguna, serta sebagai bagian dari sistem keamanan agar hanya petugas yang terdaftar yang dapat mengakses sistem. Selain itu, data identitas ini juga digunakan oleh admin untuk melakukan pengelolaan pengguna dalam sistem.

### **2. Data produksi bibit tanaman karet.**

Digunakan untuk mencatat jumlah bibit tanaman karet yang diproduksi atau dikelola dalam periode tertentu. Data ini berfungsi sebagai informasi utama dalam proses *monitoring* produksi bibit serta sebagai dasar dalam pembuatan laporan produksi yang akan digunakan oleh pihak manajemen dalam melakukan evaluasi dan pengambilan keputusan.

3. Data kondisi atau status bibit tanaman karet.

Digunakan untuk mengetahui kondisi bibit tanaman karet yang ada di lapangan, seperti kondisi baik, rusak, atau tidak layak tanam. Data ini membantu pihak perusahaan dalam melakukan pengawasan terhadap kualitas bibit serta mempermudah proses pengelolaan dan perencanaan produksi bibit tanaman karet.

4. Data waktu pencatatan produksi.

Digunakan untuk mengetahui kapan data produksi bibit dicatat oleh petugas lapangan. Data ini berfungsi sebagai acuan dalam proses *monitoring* produksi secara berkala, serta membantu admin dan manajemen dalam melakukan pelacakan data produksi berdasarkan waktu tertentu.

5. Data *login* pengguna (*username* dan *password*).

Digunakan sebagai sistem keamanan untuk memastikan hanya pengguna yang telah terdaftar yang dapat mengakses sistem. Data ini juga berfungsi untuk membedakan hak akses antara petugas lapangan dan admin, serta untuk melindungi data yang tersimpan di dalam sistem dari akses yang tidak sah.

Sumber data diperoleh dari *input* yang dilakukan oleh petugas lapangan melalui aplikasi saat melakukan pencatatan data produksi bibit tanaman karet. Selain itu, data juga dikelola oleh administrator yang bertugas memantau, memverifikasi, serta mengolah data produksi untuk keperluan pelaporan kepada pihak manajemen perusahaan. Tujuan pengelolaan data ini adalah untuk memastikan sistem mampu menyediakan informasi produksi bibit tanaman karet secara cepat, akurat, dan terintegrasi sehingga dapat mendukung proses *monitoring* serta pengambilan keputusan di lingkungan PT. Kendi Arindo.

### 3. Kebutuhan Fungsional :

Kebutuhan fungsional merupakan fungsi-fungsi utama yang harus tersedia dalam sistem agar dapat berjalan sesuai dengan tujuan pengembangan aplikasi. Pada sistem digitalisasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android*, fungsi-fungsi ini dirancang untuk mempermudah proses

pencatatan, pengelolaan, serta pemantauan data produksi bibit tanaman karet di lingkungan PT. Kendi Arindo.

Adapun fungsi yang harus tersedia dalam sistem ini meliputi:

1. Fitur registrasi dan *login* pengguna.

Fitur ini digunakan oleh pengguna untuk melakukan pendaftaran akun serta masuk ke dalam sistem menggunakan *username* dan *password* yang telah terdaftar. Fungsi ini bertujuan untuk menjaga keamanan sistem serta memastikan hanya pengguna yang memiliki hak akses yang dapat menggunakan aplikasi.

2. Fitur input data produksi bibit tanaman karet.

Fitur ini digunakan oleh petugas lapangan untuk melakukan pencatatan data produksi bibit tanaman karet secara langsung melalui aplikasi. Data yang dimasukkan meliputi informasi jumlah bibit, kondisi bibit, serta data pendukung lainnya yang berkaitan dengan proses produksi.

3. Fitur penyimpanan dan pengelolaan data produksi.

Fitur ini berfungsi untuk menyimpan seluruh data produksi yang telah diinput oleh petugas lapangan ke dalam sistem *database* sehingga data dapat dikelola, diperbarui, serta digunakan untuk keperluan *monitoring* dan pelaporan.

4. Fitur *monitoring* data produksi bibit tanaman karet.

Fitur ini memungkinkan admin atau pihak manajemen untuk melihat dan memantau data produksi bibit tanaman karet yang telah dicatat oleh petugas lapangan secara terstruktur dan mudah dipahami.

5. Fitur notifikasi otomatis (*reminder*).

Fitur ini digunakan untuk mengingatkan pengguna bahwa proses *monitoring* bibit tanaman karet dapat dilanjutkan ke tahap selanjutnya.

6. Fitur *logout* pengguna.

Fitur ini digunakan untuk keluar dari sistem setelah pengguna selesai menggunakan aplikasi. Fungsi ini bertujuan untuk menjaga keamanan akun serta mencegah akses yang tidak sah terhadap sistem.

Fungsi-fungsi tersebut dirancang untuk mendukung kemudahan penggunaan sistem, meningkatkan keakuratan data produksi bibit tanaman karet, serta membantu perusahaan dalam melakukan *monitoring* produksi secara lebih efektif dan efisien.

#### 4. Kebutuhan Non-Fungsional

Kebutuhan non-fungsional berkaitan dengan kualitas sistem yang harus dipenuhi agar sistem digitalisasi monitoring informasi data produksi bibit tanaman karet berbasis *android* dapat berjalan secara optimal. Kebutuhan ini berfokus pada aspek kinerja sistem, keamanan data, kemudahan penggunaan, serta kemampuan sistem dalam mendukung aktivitas pengguna secara efektif. Beberapa kebutuhan non-fungsional dalam sistem ini antara lain:

a. Keamanan (*Security*)

Sistem harus memiliki mekanisme keamanan yang mampu melindungi data pengguna dan data produksi bibit tanaman karet agar tidak mudah diakses oleh pihak yang tidak berwenang. Keamanan ini diterapkan melalui penggunaan sistem *login* dengan *username* dan *password* serta pengaturan hak akses antara petugas lapangan dan administrator.

b. Performa (*Performance*)

Aplikasi harus mampu berjalan dengan cepat dan responsif ketika digunakan oleh pengguna. Sistem diharapkan dapat melakukan proses penyimpanan dan pengambilan data produksi secara efisien sehingga tidak menimbulkan keterlambatan atau gangguan saat aplikasi digunakan di lapangan.

c. Kemudahan Penggunaan (*Usability*)

Antarmuka aplikasi harus dirancang dengan tampilan yang sederhana, jelas, dan mudah dipahami oleh pengguna. Hal ini penting karena pengguna aplikasi berasal dari berbagai latar belakang kemampuan teknologi, sehingga sistem harus mudah dioperasikan tanpa memerlukan pelatihan khusus.

d. Keandalan (*Reliability*)

Sistem harus mampu berjalan secara stabil dan dapat digunakan secara berkelanjutan tanpa mengalami gangguan yang dapat menghambat proses pencatatan maupun *monitoring* data produksi bibit tanaman karet. Selain itu, sistem

harus mampu menyimpan data dengan baik sehingga dapat diakses kembali saat diperlukan.

e. **Kompatibilitas (*Compatibility*)**

Aplikasi harus dapat berjalan pada berbagai perangkat *smartphone* berbasis *android* yang digunakan oleh petugas lapangan maupun admin. Dengan demikian, sistem dapat diakses secara fleksibel tanpa bergantung pada jenis perangkat tertentu.

Dengan terpenuhinya kebutuhan non-fungsional tersebut, sistem digitalisasi *monitoring* informasi data produksi bibit tanaman karet diharapkan mampu memberikan kinerja yang optimal, aman, serta mudah digunakan oleh seluruh pengguna di lingkungan PT. Kendi Arindo.

## 5. **Analisa Perangkat Keras & Lunak (*Hardware/Software*):**

Agar aplikasi dapat berjalan dengan baik, diperlukan spesifikasi perangkat keras dan perangkat lunak minimum sebagai berikut:

### 1. **Perangkat Keras (*Hardware*)**

#### **Untuk Pengguna (*User*):**

- a. *Smartphone* berbasis *Android 12*
- b. *RAM* minimal 4 GB
- c. Penyimpanan internal minimal 32 GB
- d. Koneksi internet stabil

#### **Untuk Pengembang (*Developer*):**

- a. *Laptop/PC* dengan prosesor minimal *Intel Pentium* atau yang setara
- b. *RAM* minimal 8 GB
- c. Penyimpanan minimal 256 GB
- d. Koneksi internet

### 2. **Perangkat Lunak (*Software*)**

- a. Sistem Operasi *Windows 10* atau lebih tinggi (untuk pengembangan)
- b. *Visual Studio Code* sebagai *tools* pengembangan aplikasi
- c. Bahasa Pemrograman Dart
- d. *Framework* Flutter Web
- e. *Database* menggunakan *Firebase*
- f. *Android OS* minimal versi 8.0 (Oreo) untuk pengguna

Dengan analisa kebutuhan perangkat keras dan perangkat lunak ini, diharapkan sistem dapat dikembangkan dan diimplementasikan secara optimal sesuai dengan tujuan penelitian.

#### **4.2 Perancangan Sistem (Desain) yang Diusulkan**

Perancangan sistem yang diusulkan merupakan tahapan lanjutan setelah dilakukan analisis terhadap sistem yang berjalan. Pada tahap ini, peneliti merancang solusi sistem yang mampu mengatasi permasalahan yang telah diidentifikasi sebelumnya, yaitu belum optimalnya penyampaian informasi serta masih dilakukannya pencatatan data produksi bibit tanaman karet secara manual yang menyebabkan keterlambatan informasi, kesalahan pencatatan, dan kesulitan dalam proses *monitoring*.

Sistem yang diusulkan berupa aplikasi berbasis digital (*android*) yang dirancang untuk memudahkan proses pencatatan, pengelolaan, dan monitoring data produksi bibit tanaman karet secara terstruktur dan *real-time*. Dengan adanya sistem ini, diharapkan informasi produksi dapat diakses dengan cepat, akurat, dan efisien oleh pihak yang berkepentingan.

Perancangan sistem ini dilakukan dengan pendekatan *User Centered Design* (UCD), sehingga kebutuhan dan kenyamanan pengguna menjadi fokus utama dalam proses desain, baik bagi admin maupun petugas lapangan sebagai pengguna sistem. Perancangan sistem dibagi menjadi dua bagian utama, yaitu perancangan sistem secara logis dan perancangan sistem secara fisik. Perancangan logis menggambarkan bagaimana sistem bekerja secara konseptual, sedangkan perancangan fisik menjelaskan implementasi teknis sistem.

##### **4.2.1 Perancangan Sistem (Logis)**

Perancangan sistem secara logis bertujuan untuk menggambarkan hubungan antara pengguna dengan sistem serta fungsi-fungsi yang tersedia tanpa membahas aspek teknis secara detail. Pada tahap ini digunakan model pemodelan *Unified Modeling Language* (UML) untuk merepresentasikan rancangan sistem yang diusulkan. Secara logis, sistem yang dirancang melibatkan dua aktor utama, yaitu admin dan petugas lapangan. Admin bertugas mengelola dan memonitor data

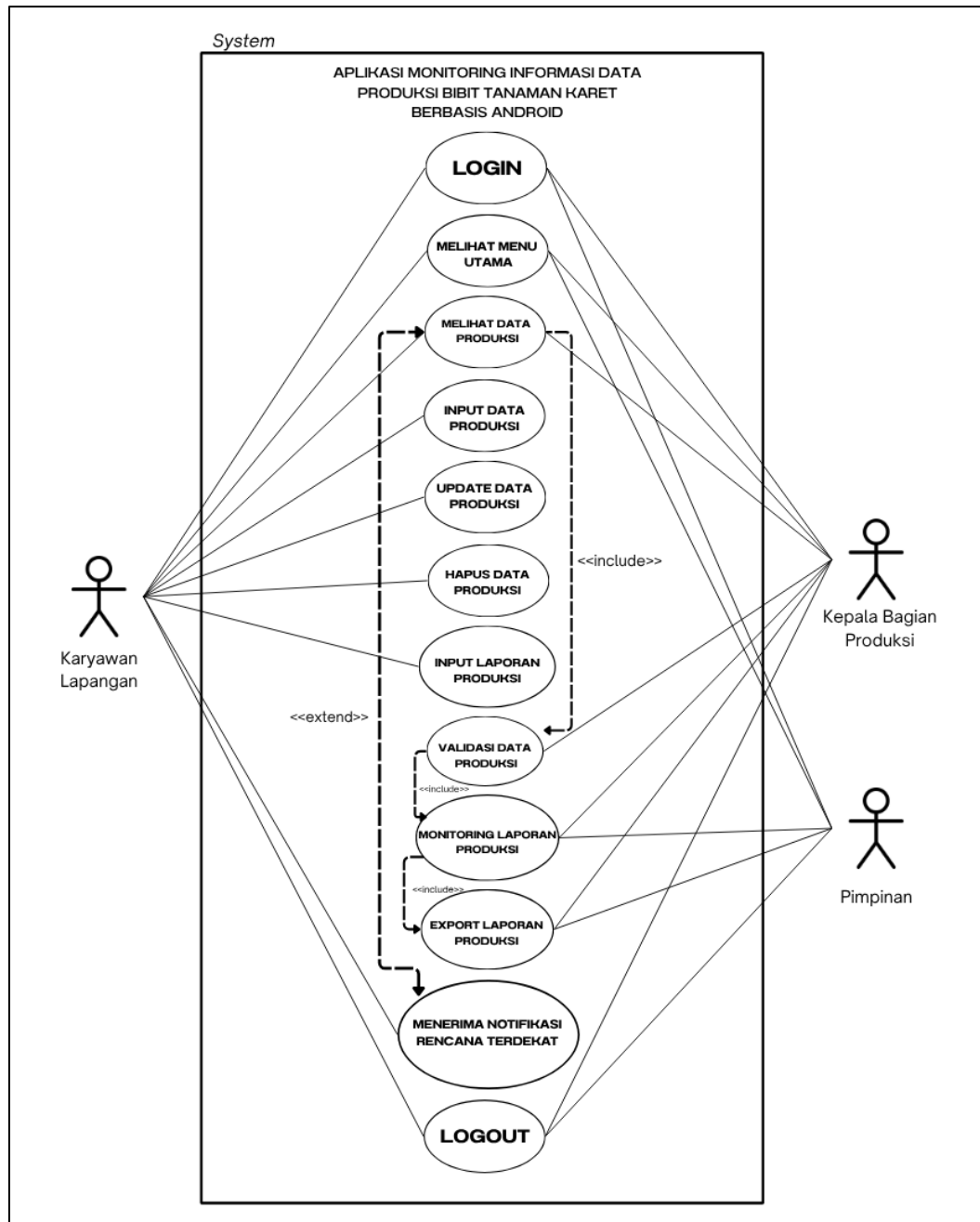
produksi, sedangkan petugas lapangan bertugas melakukan input data produksi bibit tanaman karet secara langsung di lapangan.

### 1. *Use case diagram*

Sebelum dilakukan perancangan sistem secara lebih rinci, diperlukan suatu gambaran umum mengenai interaksi antara pengguna dengan sistem yang akan dibangun. Gambaran tersebut dapat direpresentasikan melalui *Use Case Diagram*. *Use Case Diagram* digunakan untuk mengidentifikasi aktor yang terlibat serta fungsi-fungsi utama yang tersedia dalam sistem digitalisasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android*..Berikut adalah gambar *Usecase Diagram* pada aplikasi *Monitoring Informasi Data Produksi Bibit Tanaman Karet* berbasis *android* :



a. *Usecase Diagram Keseluruhan*



Sumber : data diolah oleh Penulis, 2026

Gambar 4.1 *Use Case Diagram Keseluruhan*

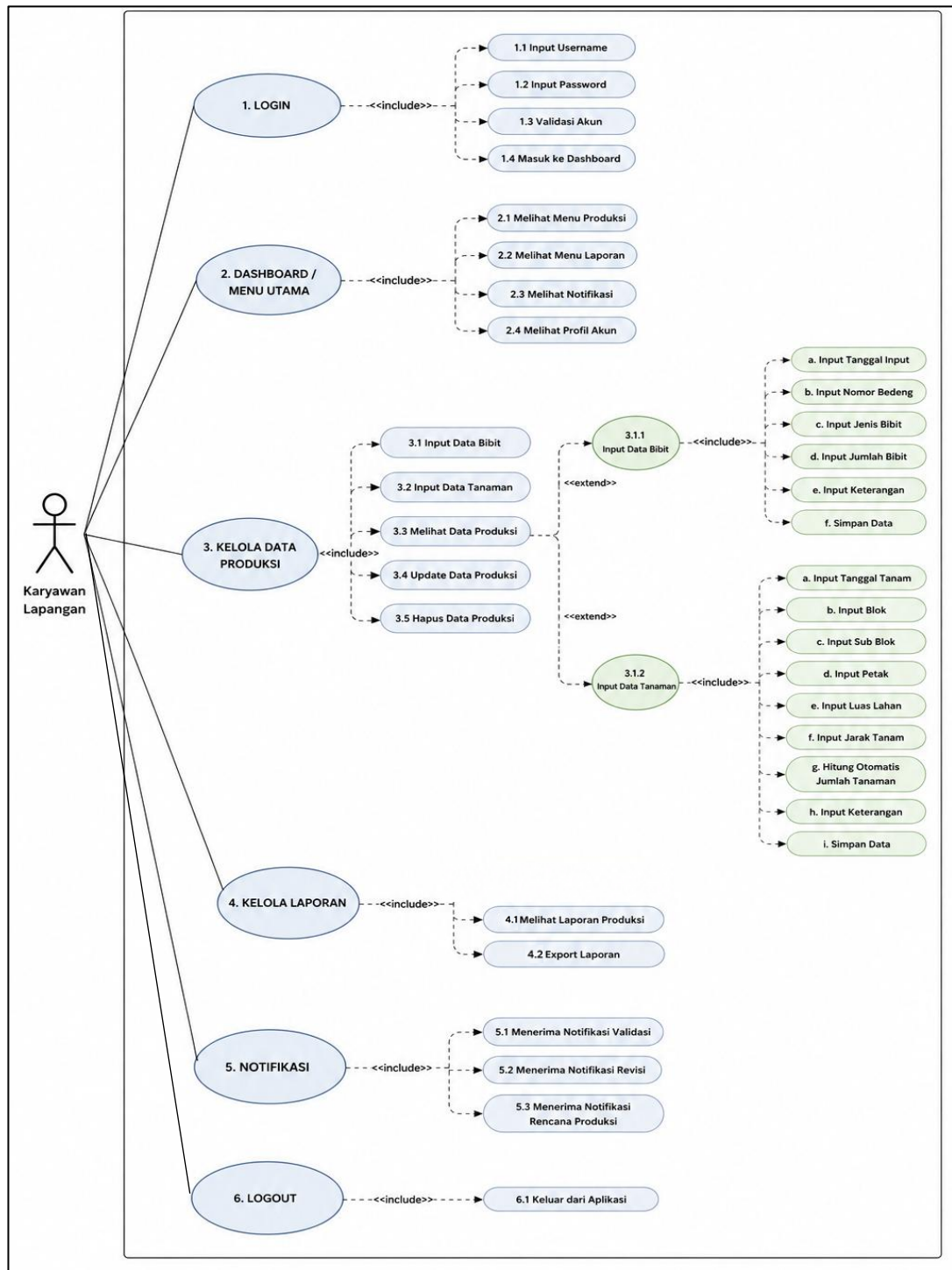
*Diagram usecase* tersebut menggambarkan sistem aplikasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android* yang melibatkan tiga aktor utama yaitu Karyawan Lapangan, Kepala Bagian Produksi, dan Pimpinan. Ketiga aktor harus melakukan proses *login* terlebih dahulu untuk dapat mengakses sistem. Setelah berhasil masuk, pengguna akan diarahkan ke menu utama yang

berfungsi sebagai pusat navigasi dalam sistem. Melalui menu tersebut, pengguna dapat melihat data produksi yang tersedia dalam sistem secara menyeluruh.

Karyawan lapangan berperan dalam melakukan input data produksi untuk mencatat hasil kegiatan di lapangan. Selain itu, karyawan juga dapat melakukan update dan penghapusan data produksi jika diperlukan. Hal ini bertujuan agar data yang tersimpan selalu sesuai dengan kondisi aktual di lapangan. Dengan demikian, data produksi dapat terkelola dengan baik dan terstruktur. Selain itu, sistem juga menyediakan fitur *input* laporan produksi yang digunakan untuk menyusun laporan berdasarkan data yang telah diinput sebelumnya. Laporan tersebut kemudian akan melalui proses validasi data produksi yang dilakukan oleh kepala bagian produksi untuk memastikan kebenaran dan keakuratan informasi.

Kepala bagian juga memiliki peran dalam melakukan *monitoring* laporan produksi untuk melihat perkembangan data secara berkala. Selain itu, sistem menyediakan fitur export laporan produksi agar data dapat diunduh dalam format tertentu seperti *pdf* atau *excel*. Fitur ini memudahkan dalam penyusunan laporan resmi dan dokumentasi. Pimpinan memiliki hak akses yang lebih terbatas yaitu hanya melakukan *monitoring* laporan produksi dan *export* laporan. Hal ini bertujuan agar pimpinan dapat fokus pada pengambilan keputusan berdasarkan data yang tersedia. Dengan adanya fitur tersebut, sistem mampu mendukung proses evaluasi dan pengambilan keputusan secara efektif.

Sistem juga dilengkapi dengan fitur notifikasi rencana kegiatan yang berfungsi sebagai pengingat bagi karyawan lapangan terhadap aktivitas produksi yang harus dilakukan. Notifikasi ini muncul berdasarkan waktu tertentu dari data produksi yang telah diinput sebelumnya. Fitur ini membantu karyawan dalam menjalankan kegiatan produksi secara tepat waktu.

b. *Usecase Diagram Karyawan Lapangan*

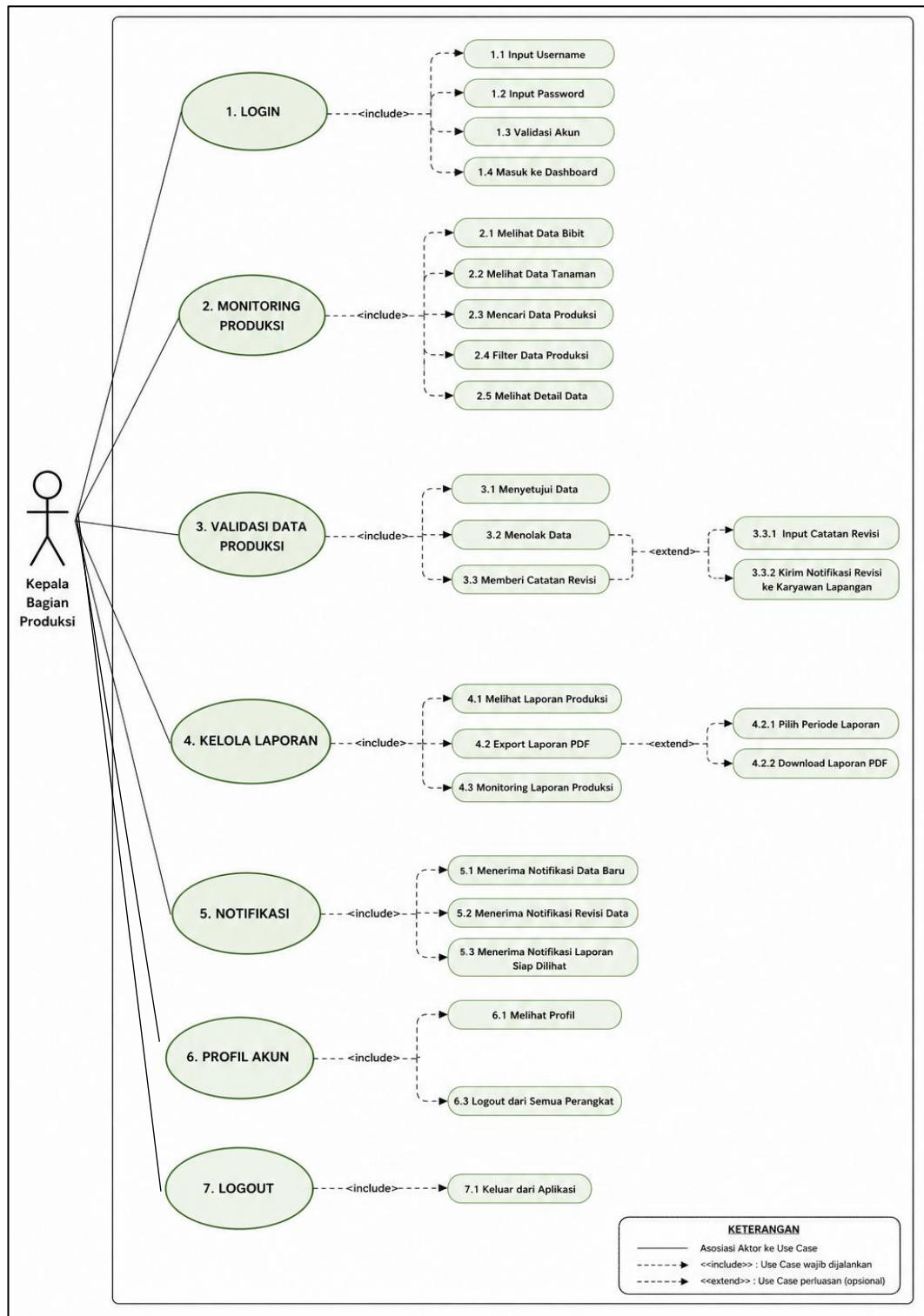
Sumber : data diolah oleh Penulis, 2026

Gambar 4.2 *Use Case Diagram Karyawan Lapangan*

*Use case diagram* karyawan lapangan menggambarkan interaksi pengguna terhadap sistem aplikasi *monitoring* produksi bibit tanaman karet berbasis *android*.

Karyawan lapangan memiliki akses untuk melakukan login ke dalam sistem menggunakan *email* dan *password* yang telah terdaftar. Setelah berhasil *login*, pengguna dapat mengakses *dashboard* utama yang berisi menu pengelolaan data produksi dan laporan. Pengguna dapat melakukan input data bibit dan data tanaman, melihat data produksi, memperbarui data, serta menghapus data yang diperlukan. Selain itu, pengguna juga dapat membuat laporan produksi dan melihat informasi notifikasi dari sistem. Sistem membantu pengguna dalam menghitung jumlah tanaman secara otomatis berdasarkan luas lahan yang diinput. Setelah seluruh aktivitas selesai dilakukan, pengguna dapat keluar dari aplikasi melalui fitur *logout*.



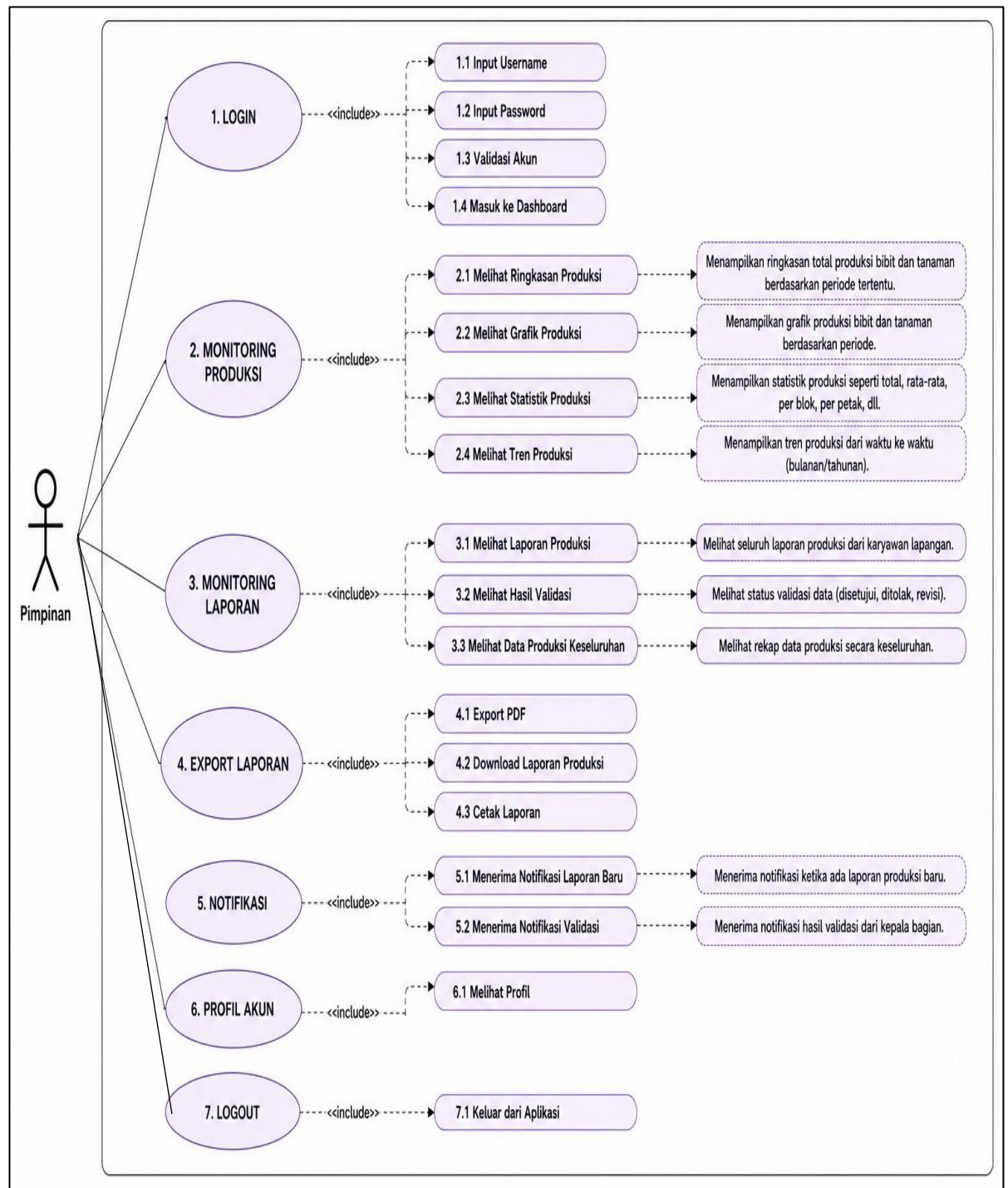
c. *Usecase Diagram Kepala Bagian*

Sumber : data diolah oleh Penulis, 2026

Gambar 4.3 Use Case Diagram Kepala Bagian

*Use case diagram* kepala bagian produksi menggambarkan aktivitas kepala bagian dalam melakukan *monitoring* dan validasi data produksi tanaman karet. Kepala bagian harus melakukan *login* terlebih dahulu sebelum mengakses sistem. Setelah berhasil masuk, pengguna dapat melihat *dashboard* yang berisi data produksi dan laporan dari karyawan lapangan. Kepala bagian memiliki hak akses untuk melakukan *monitoring* data bibit dan tanaman, mencari data produksi, serta memfilter data berdasarkan kebutuhan. Selain itu, kepala bagian dapat melakukan validasi data produksi dengan menyetujui, menolak, atau memberikan catatan revisi terhadap data yang dikirim oleh karyawan lapangan. Pengguna juga dapat melihat laporan produksi dan melakukan *export* laporan dalam bentuk *pdf*. Sistem memberikan notifikasi terkait data baru maupun revisi laporan. Setelah selesai menggunakan aplikasi, kepala bagian dapat keluar dari sistem melalui fitur *logout*.



d. *Usecase Diagram Pimpinan*

Sumber : data diolah oleh Penulis, 2026

Gambar 4.4 *Use Case Diagram* Pimpinan

*Use case diagram* pimpinan menggambarkan interaksi pimpinan terhadap sistem aplikasi *monitoring* produksi bibit tanaman karet berbasis *android*. Pimpinan memiliki hak akses untuk melakukan *login* ke dalam sistem menggunakan *email*

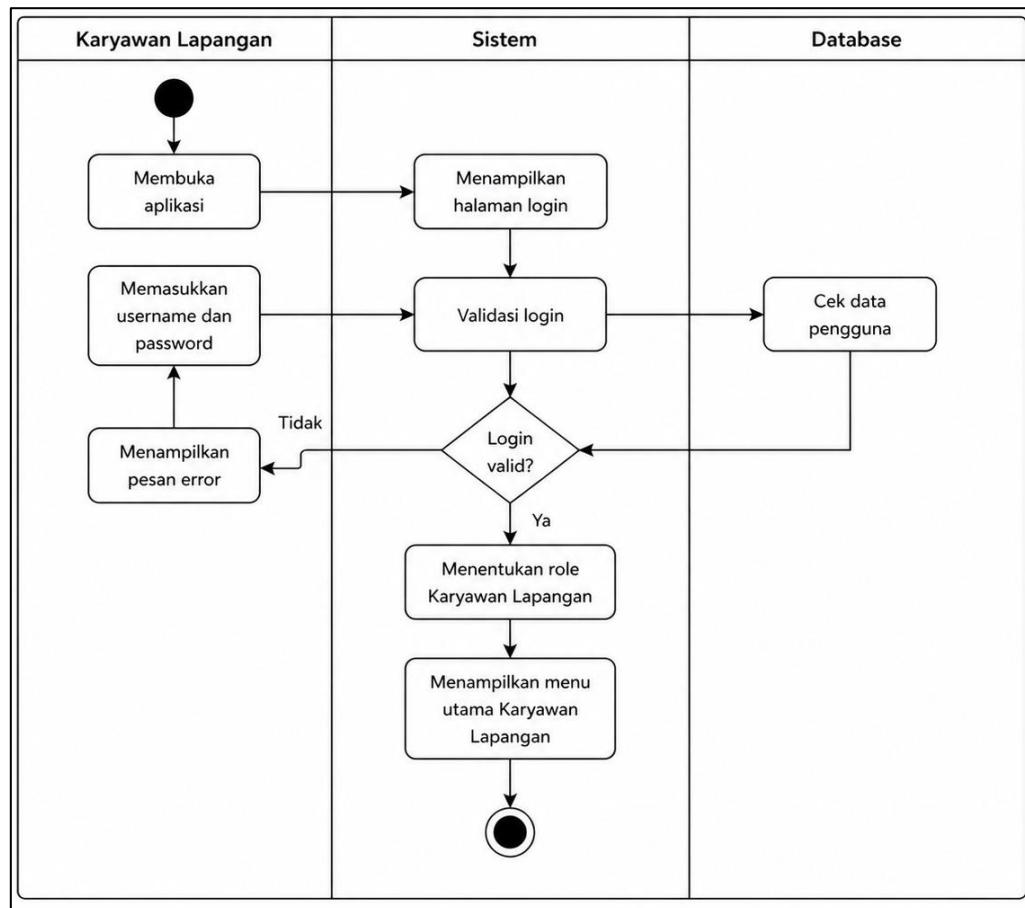
dan *password* yang telah terdaftar. Setelah berhasil *login*, pimpinan dapat melihat *dashboard* yang berisi ringkasan produksi, grafik produksi, statistik produksi, serta tren produksi berdasarkan periode tertentu. Selain itu, pimpinan juga dapat melakukan *monitoring* laporan produksi dan melihat hasil validasi data dari kepala bagian produksi. Sistem menyediakan fitur *export* laporan dalam bentuk *pdf* maupun cetak laporan untuk mempermudah penyampaian informasi produksi. Pimpinan juga dapat menerima notifikasi terkait laporan baru dan hasil validasi data produksi. Selain itu tersedia fitur profil akun untuk melihat informasi akun. Setelah seluruh aktivitas selesai dilakukan, pimpinan dapat keluar dari aplikasi melalui fitur *logout*.



## 2. Activity Diagram

*Activity diagram* menyediakan gambaran untuk memodelkan proses atau alur kerja model dalam suatu sistem informasi. Berikut adalah gambar *activity diagram* pada aplikasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android* :

1. *Activity diagram login* karyawan lapangan yang diusulkan.



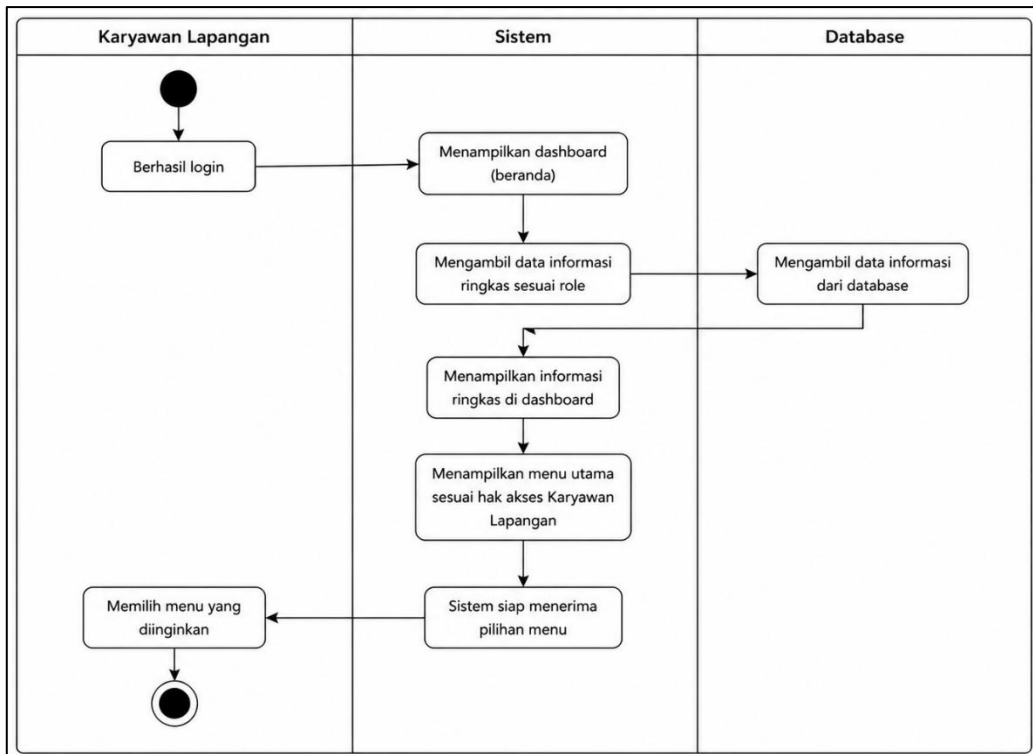
Sumber: data diolah oleh Penulis, 2026

Gambar 4.5 Activity Diagram Login Karyawan Lapangan yang diusulkan

Berdasarkan gambar diatas, *activity diagram* ini menggambarkan proses *login* yang dilakukan oleh karyawan lapangan untuk masuk ke dalam sistem. Proses dimulai ketika pengguna membuka aplikasi dan memasukkan *email* serta *password*. Selanjutnya, sistem melakukan validasi data dengan mencocokkannya ke *database*. Jika data tidak valid, sistem akan menampilkan pesan *error* dan pengguna diminta untuk menginput ulang. Namun, jika data valid, sistem akan mengenali pengguna

sebagai karyawan lapangan dan menampilkan menu utama sesuai dengan hak aksesnya.

2. *Activity diagram* karyawan lapangan melihat menu utama yang diusulkan

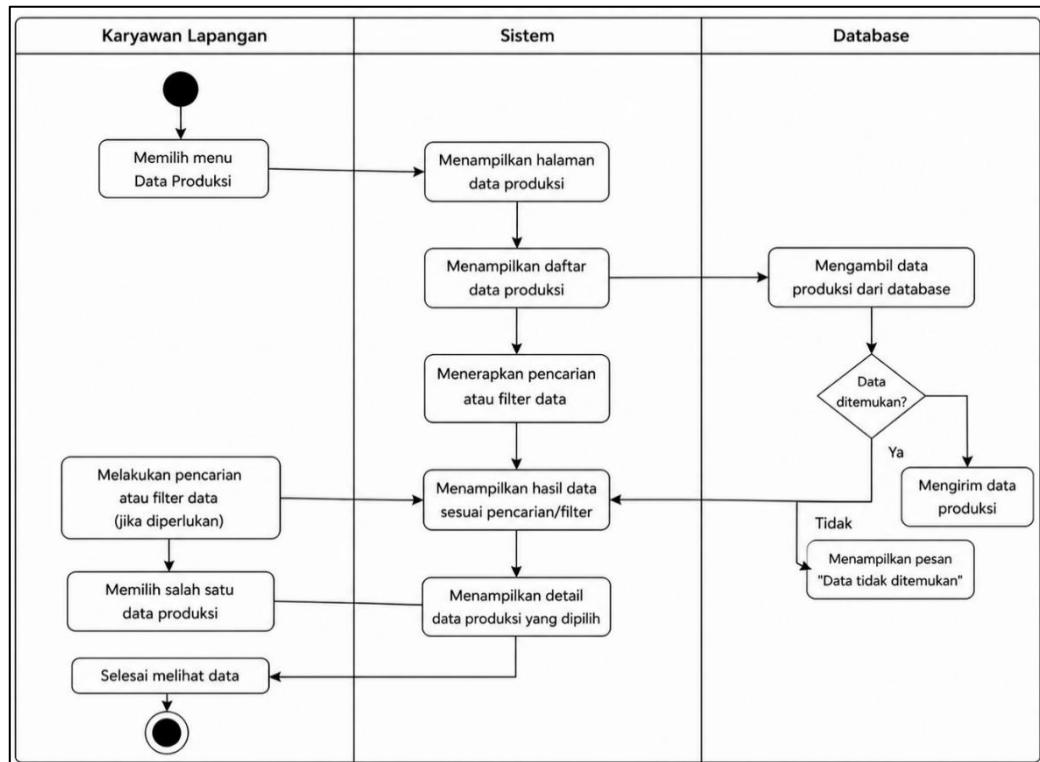


Sumber: data diolah oleh Penulis, 2026

Gambar 4.6 *Activity Diagram* Karyawan Lapangan Melihat Menu Utama yang diusulkan

Berdasarkan gambar diatas, *activity diagram* ini menggambarkan proses setelah karyawan lapangan berhasil *login* ke dalam sistem. Sistem akan menampilkan *dashboard* sebagai halaman utama. Selanjutnya, sistem mengambil data informasi singkat dari *database* dan menampilkannya pada *dashboard*. Setelah itu, sistem menampilkan menu utama yang sesuai dengan hak akses karyawan lapangan. Pengguna kemudian dapat memilih menu yang diinginkan untuk melanjutkan ke proses berikutnya.

3. *Activity diagram* karyawan melihat data produksi yang diusulkan

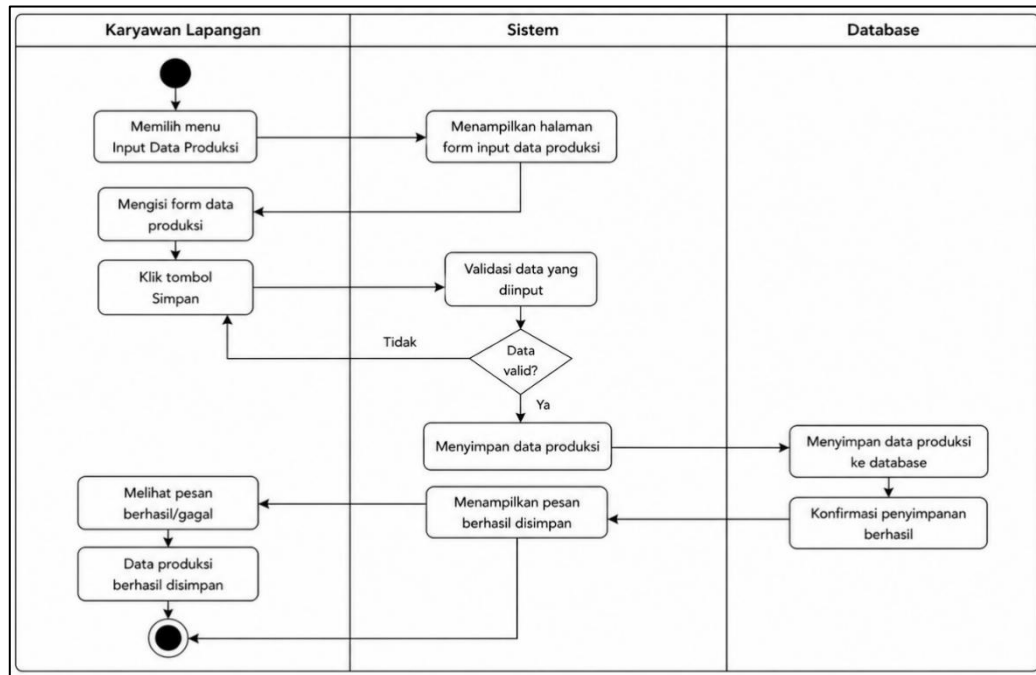


Sumber: data diolah oleh Penulis, 2026

Gambar 4.7 Activity Diagram Karyawan Lapangan Melihat Data Produksi yang diusulkan

Berdasarkan gambar diatas, *activity diagram* ini menggambarkan proses karyawan lapangan dalam melihat data produksi yang tersedia di dalam sistem. Proses dimulai ketika pengguna memilih menu data produksi. Sistem kemudian mengambil data dari *database* dan menampilkan daftar data produksi. Pengguna dapat melakukan pencarian atau filter untuk menemukan data yang dibutuhkan. Jika data ditemukan, sistem akan menampilkan hasil sesuai pencarian. Selanjutnya, pengguna dapat memilih salah satu data untuk melihat detailnya. Proses berakhir setelah pengguna selesai melihat data produksi.

4. *Activity diagram* karyawan lapangan input data produksi yang diusulkan.

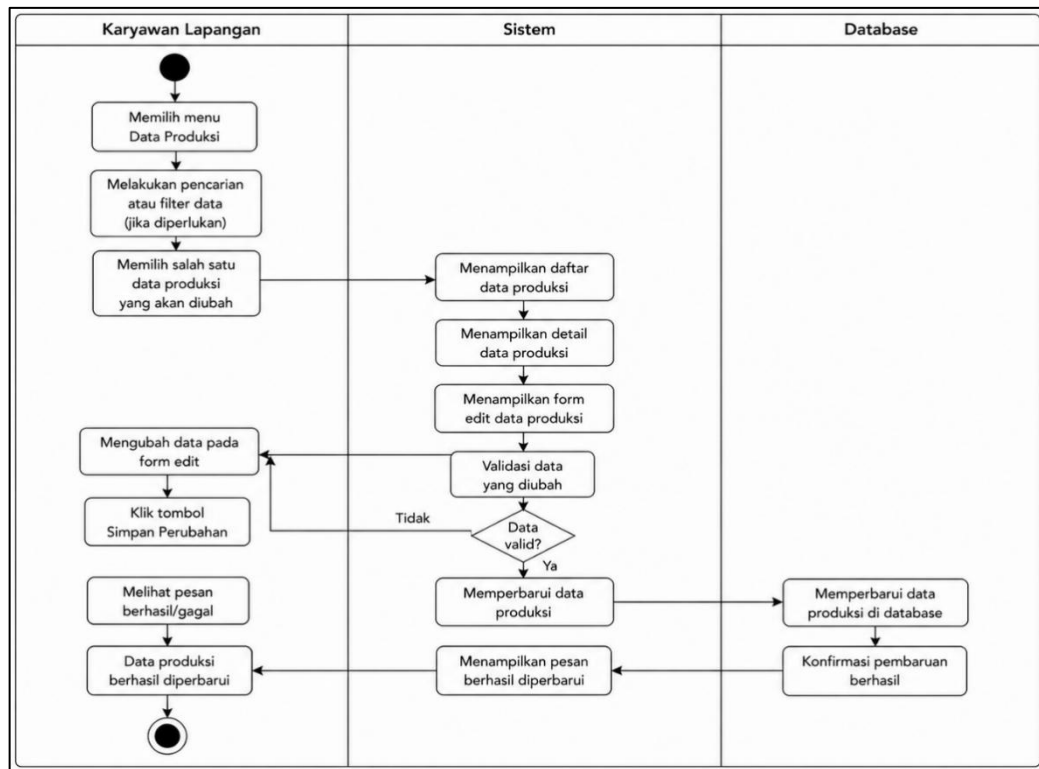


Sumber: data diolah oleh Penulis, 2026

Gambar 4.8 *Activity Diagram* Karyawan Lapangan Input Data Produksi yang diusulkan

Berdasarkan gambar diatas, *activity diagram* ini menggambarkan proses karyawan lapangan dalam memasukkan data produksi ke dalam sistem. Proses dimulai ketika pengguna memilih menu *input* data produksi, kemudian sistem menampilkan *form input*. Pengguna mengisi data produksi dan menekan tombol simpan. Selanjutnya, sistem melakukan validasi terhadap data yang diinput. Jika data tidak valid, sistem akan menampilkan pesan *error* dan pengguna diminta untuk memperbaiki data. Namun, jika data valid, sistem akan menyimpan data ke dalam *database* dan menampilkan notifikasi bahwa data berhasil disimpan.

5. *Activity diagram* karyawan lapangan *update* data produksi yang diusulkan.

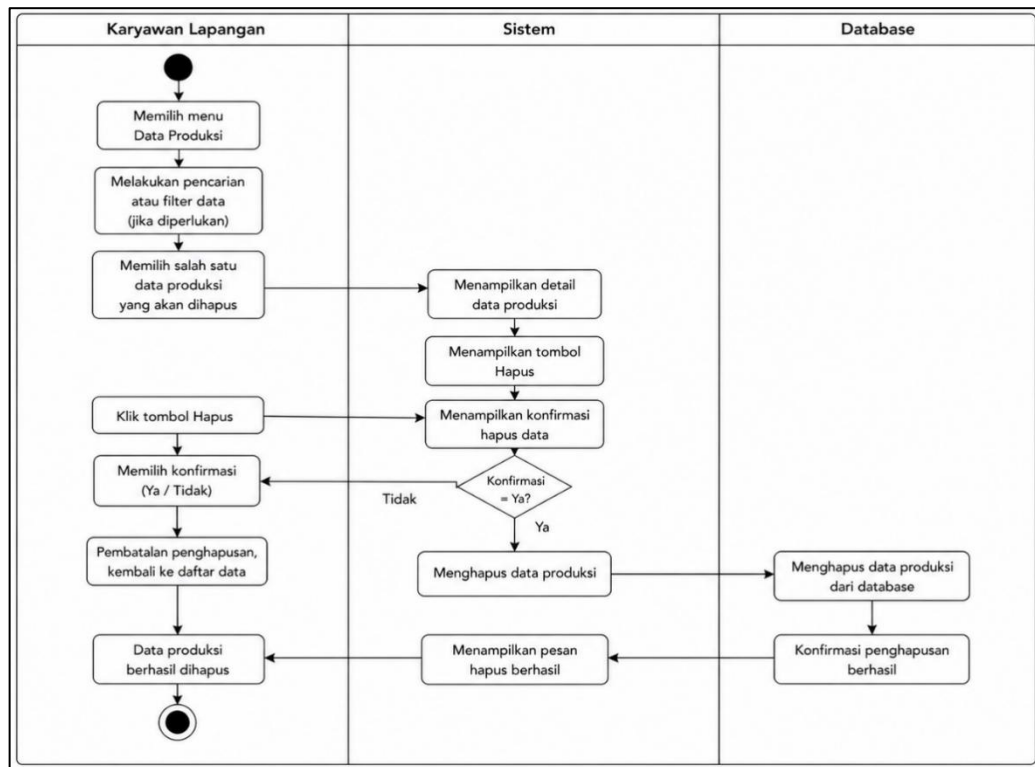


Sumber: data diolah oleh Penulis, 2026

Gambar 4.9 *Activity Diagram* Karyawan Lapangan *Update* Data Produksi yang diusulkan.

Berdasarkan gambar diatas, *activity diagram* ini menggambarkan proses karyawan lapangan dalam memperbarui data produksi yang sudah tersimpan di dalam sistem. Proses dimulai ketika pengguna memilih data produksi yang ingin diubah. Sistem kemudian menampilkan detail data, dan pengguna memilih opsi edit. Setelah itu, sistem menampilkan *form* edit yang berisi data sebelumnya. Pengguna melakukan perubahan data sesuai kebutuhan dan menekan tombol simpan. Selanjutnya, sistem melakukan validasi terhadap data yang diubah. Jika data tidak valid, sistem akan menampilkan pesan *error*. Namun, jika data valid, sistem akan memperbarui data di *database* dan menampilkan notifikasi bahwa data berhasil di-*update*.

6. *Activity diagram* karyawan lapangan hapus data produksi yang diusulkan.

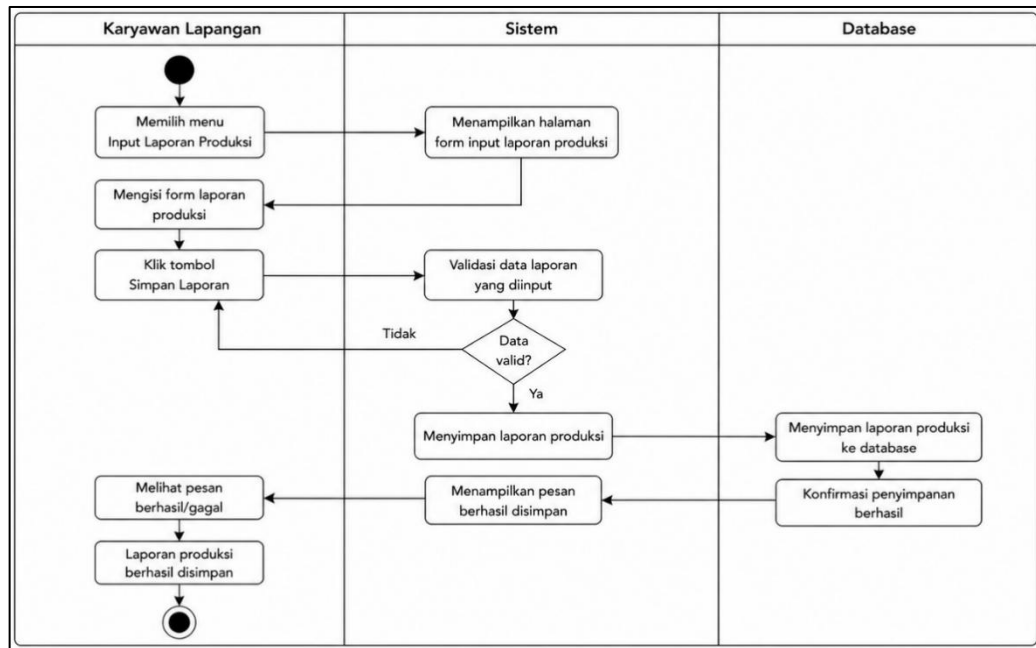


Sumber: data diolah oleh Penulis, 2026

Gambar 4.10 *Activity Diagram* Karyawan Lapangan Hapus Data Produksi yang diusulkan.

Berdasarkan gambar diatas, *activity diagram* ini menggambarkan proses karyawan lapangan dalam menghapus data produksi dari sistem. Proses dimulai ketika pengguna memilih data produksi yang akan dihapus. Sistem menampilkan detail data dan menyediakan tombol hapus. Setelah pengguna menekan tombol hapus, sistem akan menampilkan konfirmasi untuk memastikan tindakan tersebut. Jika pengguna memilih tidak, maka proses dibatalkan dan kembali ke daftar data. Namun, jika pengguna memilih ya, sistem akan menghapus data dari *database* dan menampilkan notifikasi bahwa data berhasil dihapus.

7. *Activity diagram* karyawan lapangan input laporan produksi yang diusulkan.

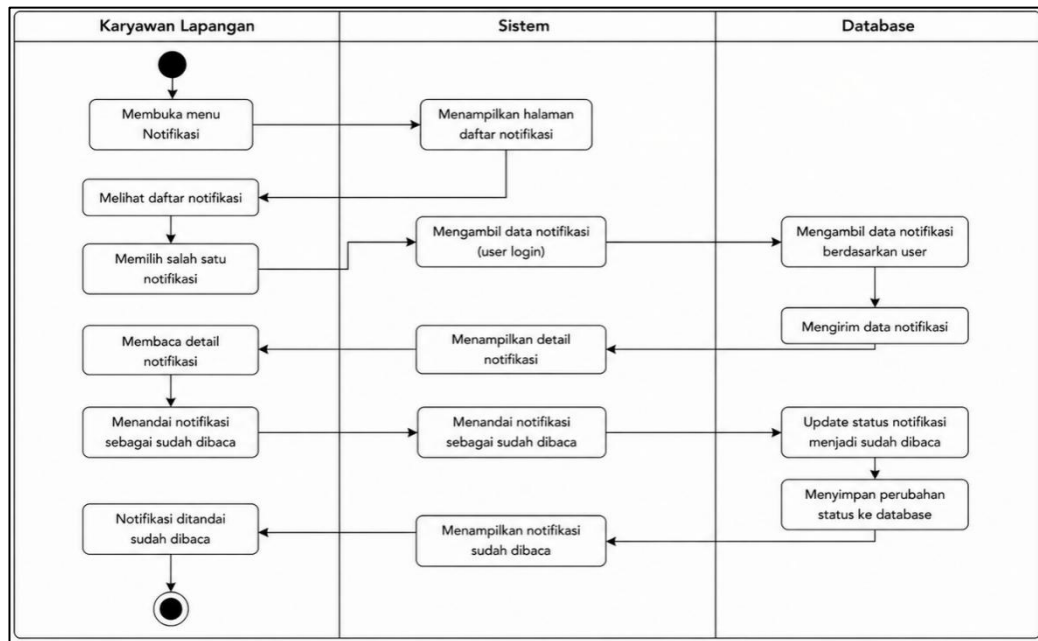


Sumber: data diolah oleh Penulis, 2026

Gambar 4.11 *Activity Diagram* Karyawan Lapangan Input Laporan Produksi yang diusulkan.

Berdasarkan gambar diatas, *activity diagram* ini menggambarkan proses karyawan lapangan dalam memasukkan laporan produksi ke dalam sistem. Proses dimulai ketika pengguna memilih menu *input* laporan produksi, kemudian sistem menampilkan *form input* laporan. Pengguna mengisi data laporan dan menekan tombol simpan. Selanjutnya, sistem melakukan validasi terhadap data yang diinput. Jika data tidak valid, sistem akan menampilkan pesan *error* dan pengguna diminta untuk memperbaiki data. Namun, jika data valid, sistem akan menyimpan laporan ke dalam *database* dan menampilkan notifikasi bahwa laporan berhasil disimpan.

8. *Activity diagram* notifikasi karyawan lapangan yang diusulkan.

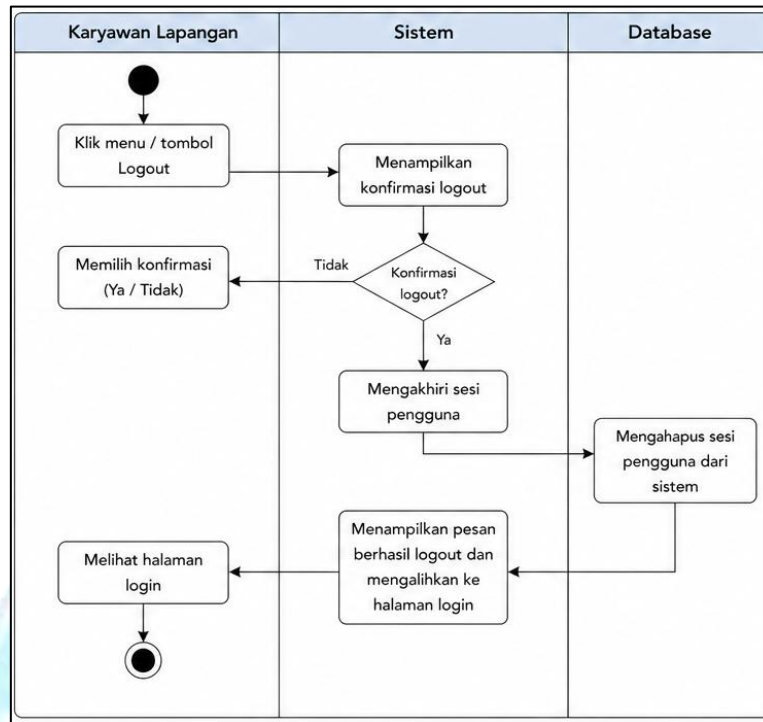


Sumber: data diolah oleh Penulis, 2026

Gambar 4.12 *Activity Diagram* Notifikasi Karyawan Lapangan yang diusulkan.

Berdasarkan gambar diatas, *activity diagram* ini menggambarkan proses karyawan lapangan dalam menerima dan melihat notifikasi dari sistem. Proses dimulai ketika pengguna membuka menu notifikasi. Sistem kemudian mengambil data notifikasi dari *database* sesuai dengan pengguna yang sedang *login* dan menampilkan daftar notifikasi. Pengguna dapat memilih salah satu notifikasi untuk melihat detailnya. Setelah membaca, pengguna dapat menandai notifikasi sebagai sudah dibaca. Sistem kemudian memperbarui status notifikasi di *database* dan menampilkan bahwa notifikasi telah dibaca.

9. *Activity diagram logout* karyawan lapangan yang diusulkan.

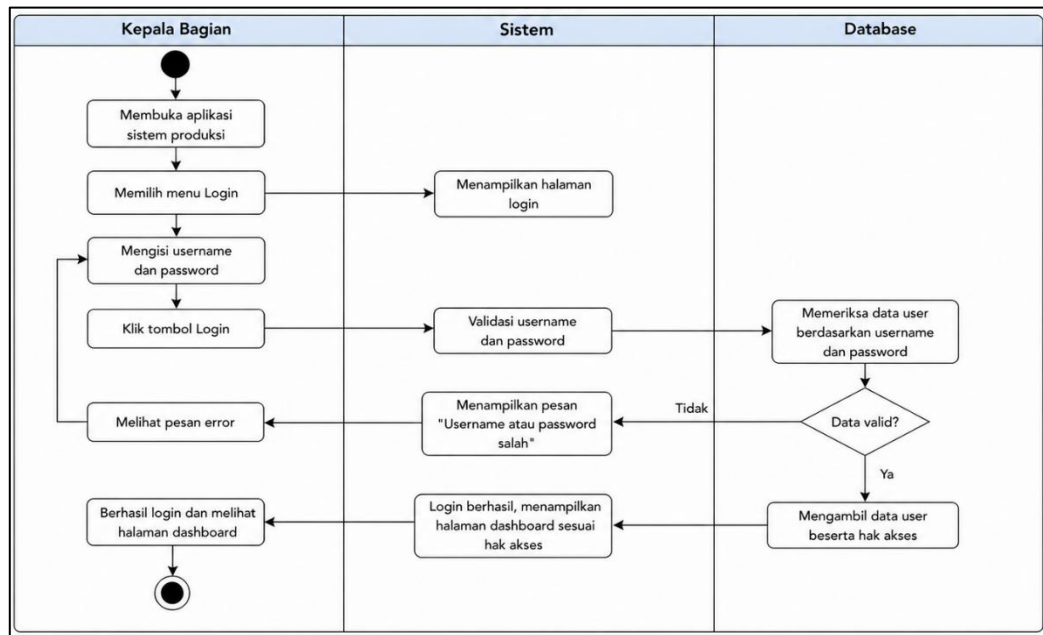


Sumber: data diolah oleh Penulis, 2026

Gambar 4.13 *Activity Diagram Logout* Karyawan Lapangan yang diusulkan.

Berdasarkan gambar diatas, *activity diagram* ini menggambarkan proses karyawan lapangan keluar dari sistem. Proses dimulai ketika pengguna memilih menu atau tombol *logout*. Sistem kemudian menampilkan konfirmasi untuk memastikan tindakan *logout*. Jika pengguna memilih tidak, maka proses dibatalkan dan kembali ke halaman sebelumnya. Namun, jika pengguna memilih ya, sistem akan mengakhiri sesi pengguna dan menghapus data sesi dari sistem. Setelah itu, sistem menampilkan pesan *logout* berhasil dan mengarahkan pengguna ke halaman *login*.

10. *Activity diagram login kepala bagian yang diusulkan.*

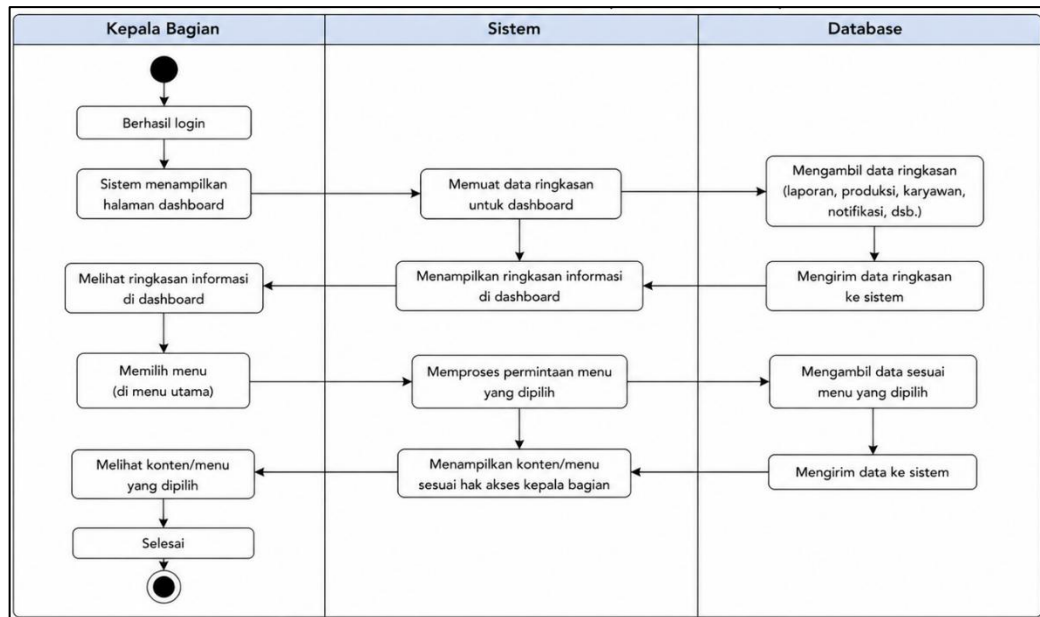


Sumber: data diolah oleh Penulis, 2026

Gambar 4.14 *Activity Diagram Login Kepala Bagian yang diusulkan*

Berdasarkan gambar diatas, *activity diagram* ini menggambarkan proses *login* yang dilakukan oleh kepala bagian untuk mengakses sistem. Proses dimulai ketika pengguna membuka aplikasi dan memilih menu *login*. Pengguna kemudian memasukkan *email* dan *password*, lalu sistem melakukan validasi dengan mencocokkan data ke *database*. Jika data tidak valid, sistem akan menampilkan pesan *error* dan pengguna diminta untuk menginput ulang. Namun, jika data valid, sistem akan mengenali pengguna sebagai kepala bagian dan menampilkan halaman *dashboard* atau menu utama sesuai dengan hak aksesnya.

11. *Activity diagram* kepala bagian melihat menu utama (*Dashboard*) yang diusulkan.

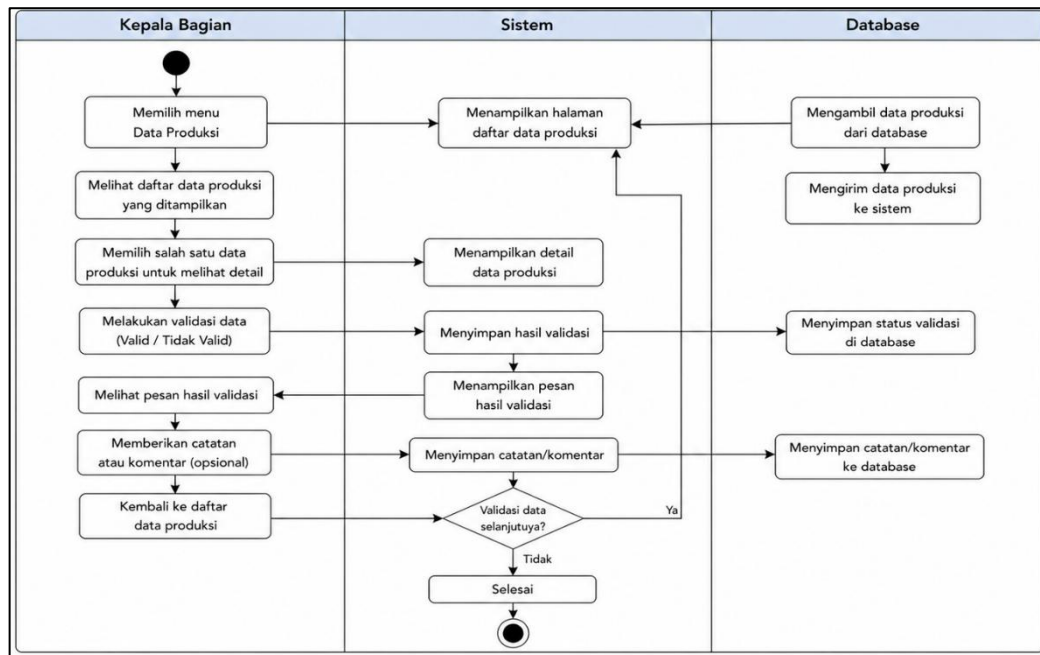


Sumber: data diolah oleh Penulis, 2026

Gambar 4.15 *Activity Diagram* Kepala Bagian Melihat Menu Utama (*Dashboard*) yang diusulkan.

Berdasarkan gambar diatas, *activity diagram* ini menggambarkan proses kepala bagian setelah berhasil *login* ke dalam sistem. Sistem akan menampilkan halaman *dashboard* yang berisi ringkasan informasi. Selanjutnya, sistem mengambil data dari *database* seperti data produksi, laporan, dan informasi lainnya, kemudian menampilkannya dalam bentuk ringkasan di *dashboard*. Kepala bagian dapat melihat informasi tersebut dan memilih menu yang tersedia sesuai kebutuhan. Sistem kemudian memproses permintaan dan menampilkan halaman atau fitur sesuai dengan hak akses kepala bagian.

12. *Activity diagram* kepala bagian melihat data produksi dan memvalidasi data produksi yang diusulkan.

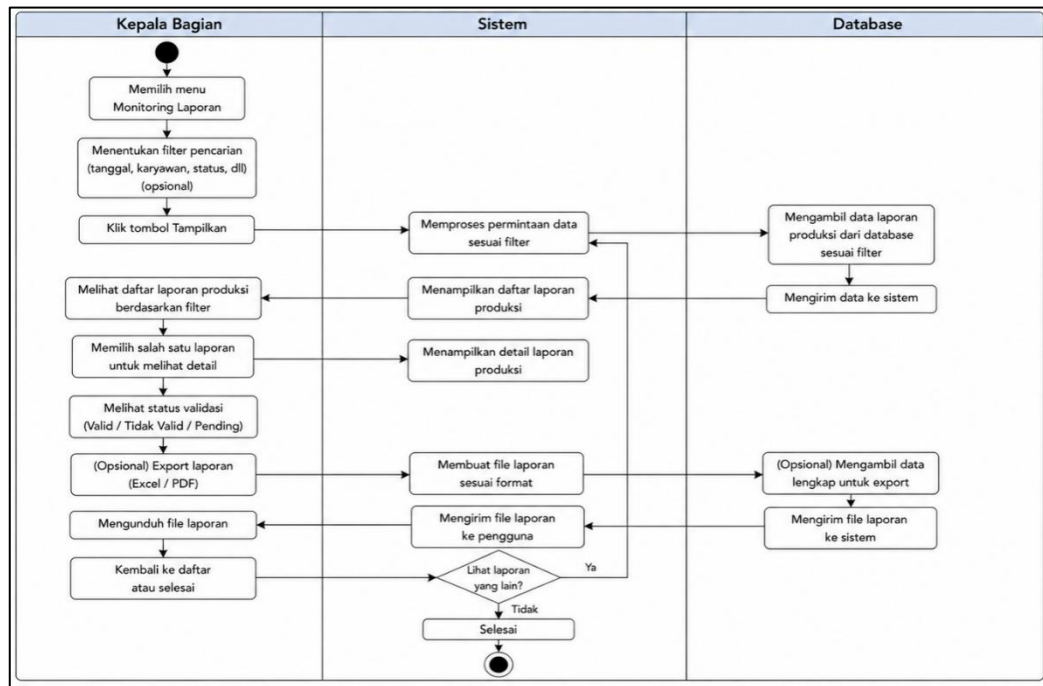


Sumber: data diolah oleh Penulis, 2026

Gambar 4.16 *Activity Diagram* Kepala Bagian Melihat Data Produksi dan Memvalidasi Data Produksi yang diusulkan

Berdasarkan gambar diatas, *activity diagram* ini menggambarkan proses kepala bagian dalam melihat dan melakukan validasi terhadap data produksi yang diinput oleh karyawan lapangan. Proses dimulai ketika pengguna memilih menu data produksi. Sistem kemudian mengambil dan menampilkan daftar data produksi dari *database*. Kepala bagian dapat memilih salah satu data untuk melihat detailnya. Setelah itu, kepala bagian melakukan validasi data dengan menentukan apakah data tersebut valid atau tidak. Hasil validasi akan disimpan oleh sistem ke dalam *database* dan sistem menampilkan pesan hasil validasi. Kepala bagian juga dapat memberikan catatan atau komentar tambahan (opsional). Setelah selesai, pengguna dapat kembali ke daftar data untuk melakukan validasi data lainnya atau mengakhiri proses.

13. *Activity diagram* kepala bagian *monitoring* laporan produksi yang diusulkan.

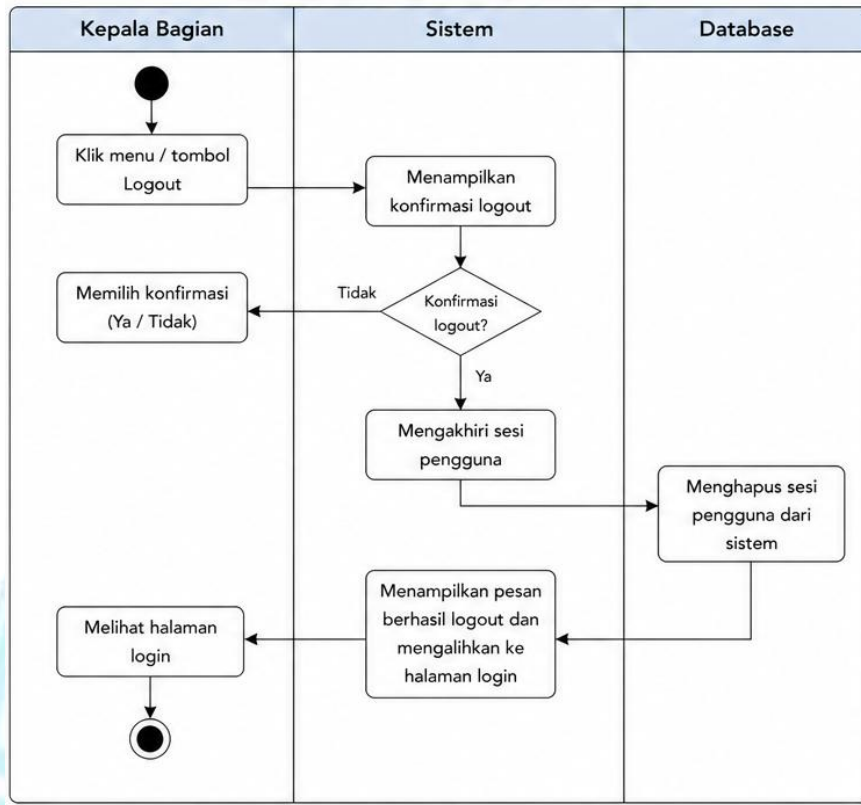


Sumber: data diolah oleh Penulis, 2026

Gambar 4.17 *Activity Diagram* Kepala Bagian *Monitoring* Laporan Produksi yang diusulkan

Berdasarkan gambar diatas, *activity diagram* ini menggambarkan proses kepala bagian dalam melakukan *monitoring* terhadap laporan produksi yang telah diinput oleh karyawan lapangan. Proses dimulai ketika pengguna memilih menu *monitoring* laporan. Sistem kemudian mengambil data laporan dari *database* sesuai dengan filter yang dipilih (jika ada), dan menampilkannya dalam bentuk daftar. Kepala bagian dapat melihat daftar laporan, memilih salah satu laporan untuk melihat detail, serta mengetahui status validasi laporan tersebut. Selain itu, pengguna juga dapat melakukan *export* laporan ke dalam format tertentu seperti *pdf* atau *excel*. Setelah selesai, pengguna dapat kembali ke daftar laporan untuk melihat data lainnya atau mengakhiri proses.

14. *Activity diagram logout kepala bagian yang diusulkan.*

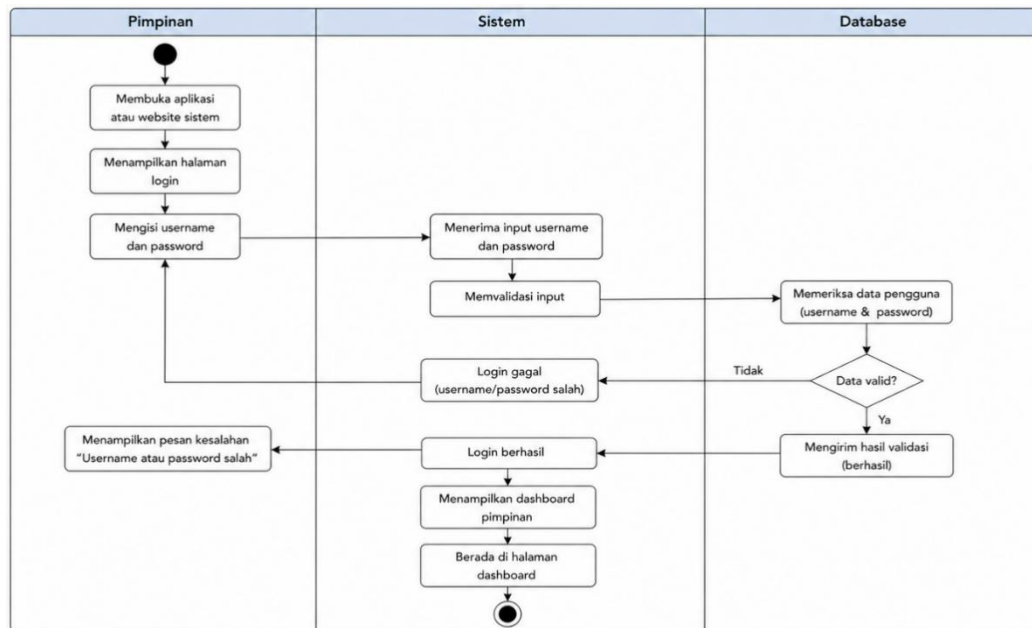


Sumber: data diolah oleh Penulis, 2026

Gambar 4.18 *Activity Diagram Logout Kepala Bagian yang diusulkan*

Berdasarkan gambar diatas, *activity diagram* ini menggambarkan proses kepala bagian keluar dari sistem. Proses dimulai saat pengguna memilih menu *logout*. Sistem akan menampilkan konfirmasi *logout*. Jika pengguna membatalkan, maka sistem akan kembali ke halaman sebelumnya. Jika pengguna melanjutkan, sistem akan mengakhiri sesi dan menghapus data *login* dari sistem. Selanjutnya, sistem menampilkan notifikasi *logout* berhasil dan mengarahkan pengguna ke halaman *login*.

15. *Activity diagram login pimpinan yang diusulkan.*

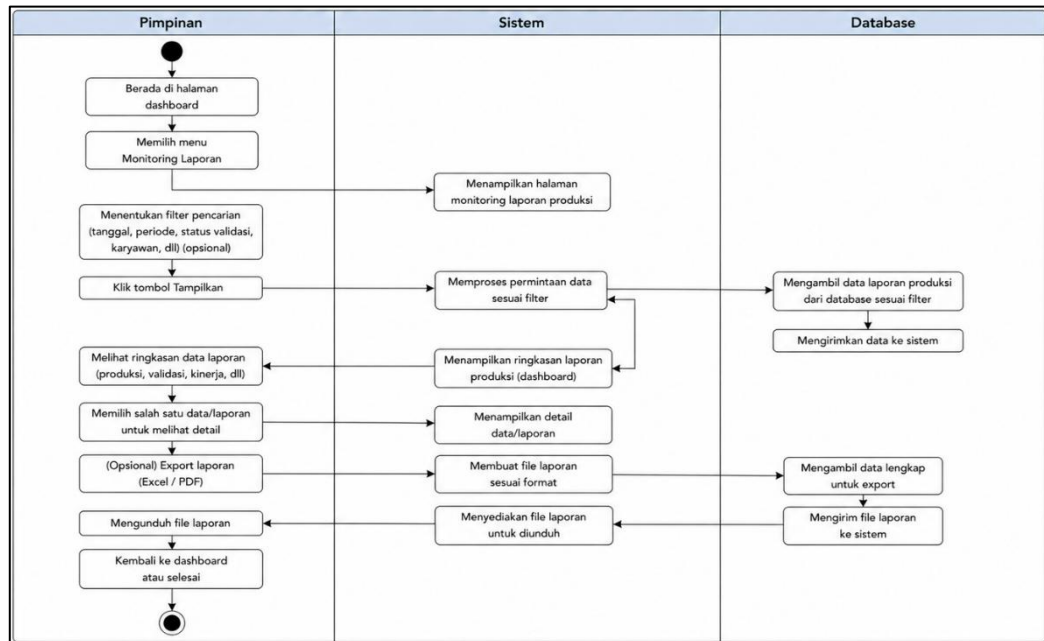


Sumber: data diolah oleh Penulis, 2026

Gambar 4.19 *Activity Diagram Login Pimpinan yang diusulkan*

Berdasarkan gambar diatas, *activity diagram* ini menggambarkan proses *login* yang dilakukan oleh pimpinan mengakses sistem. Proses dimulai ketika pengguna membuka aplikasi, kemudian sistem menampilkan halaman *login*. Pimpinan memasukkan *email* dan *password*, lalu sistem menerima dan melakukan validasi dengan mencocokkan data ke *database*. Jika data tidak valid, sistem akan menampilkan pesan kesalahan dan pengguna diminta untuk menginput ulang. Namun, jika data valid, sistem akan mengarahkan pimpinan ke halaman *dashboard* yang berisi informasi *monitoring* laporan produksi sesuai dengan hak aksesnya.

16. *Activity diagram* pimpinan melihat menu utama dan *monitoring* laporan produksi yang diusulkan.

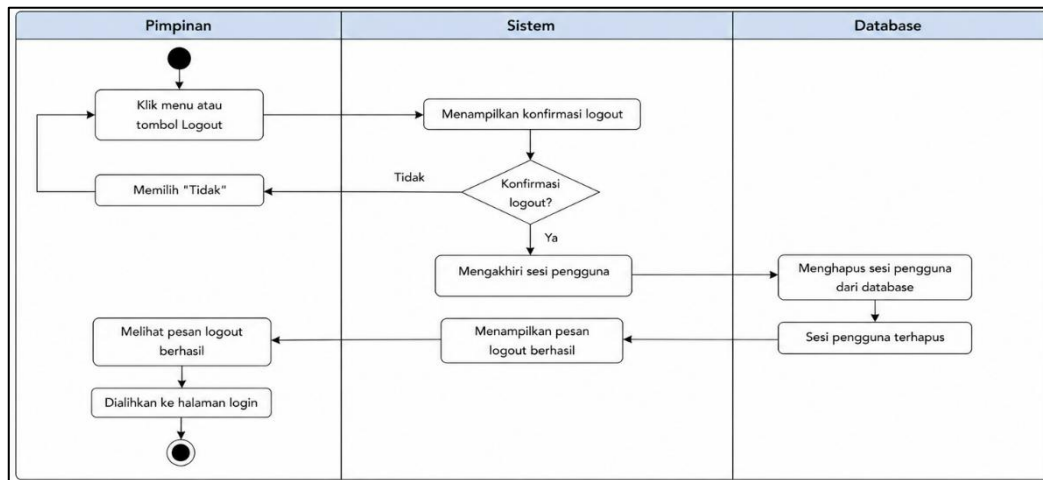


Sumber: data diolah oleh Penulis, 2026

Gambar 4.20 *Activity Diagram* Pimpinan Melihat Menu Utama Dan *Monitoring* Laporan Produksi yang diusulkan

Berdasarkan gambar diatas, *activity diagram* ini menggambarkan proses pimpinan dalam melakukan *monitoring* laporan produksi melalui sistem. Proses dimulai ketika pimpinan berada di halaman *dashboard* dan memilih menu *monitoring* laporan. Selanjutnya, pimpinan dapat menentukan filter pencarian seperti tanggal, periode, atau status untuk menampilkan data yang diinginkan. Sistem kemudian memproses permintaan dan mengambil data dari *database* sesuai filter, lalu menampilkannya dalam bentuk ringkasan pada *dashboard*. Pimpinan dapat melihat ringkasan data, memilih salah satu laporan untuk melihat detail, serta memahami kondisi produksi secara keseluruhan. Selain itu, pimpinan juga dapat melakukan *export* laporan ke format tertentu seperti *pdf* atau *excel*. Setelah selesai, pimpinan dapat kembali ke *dashboard* atau mengakhiri proses.

17. *Activity diagram logout pimpinan yang diusulkan*



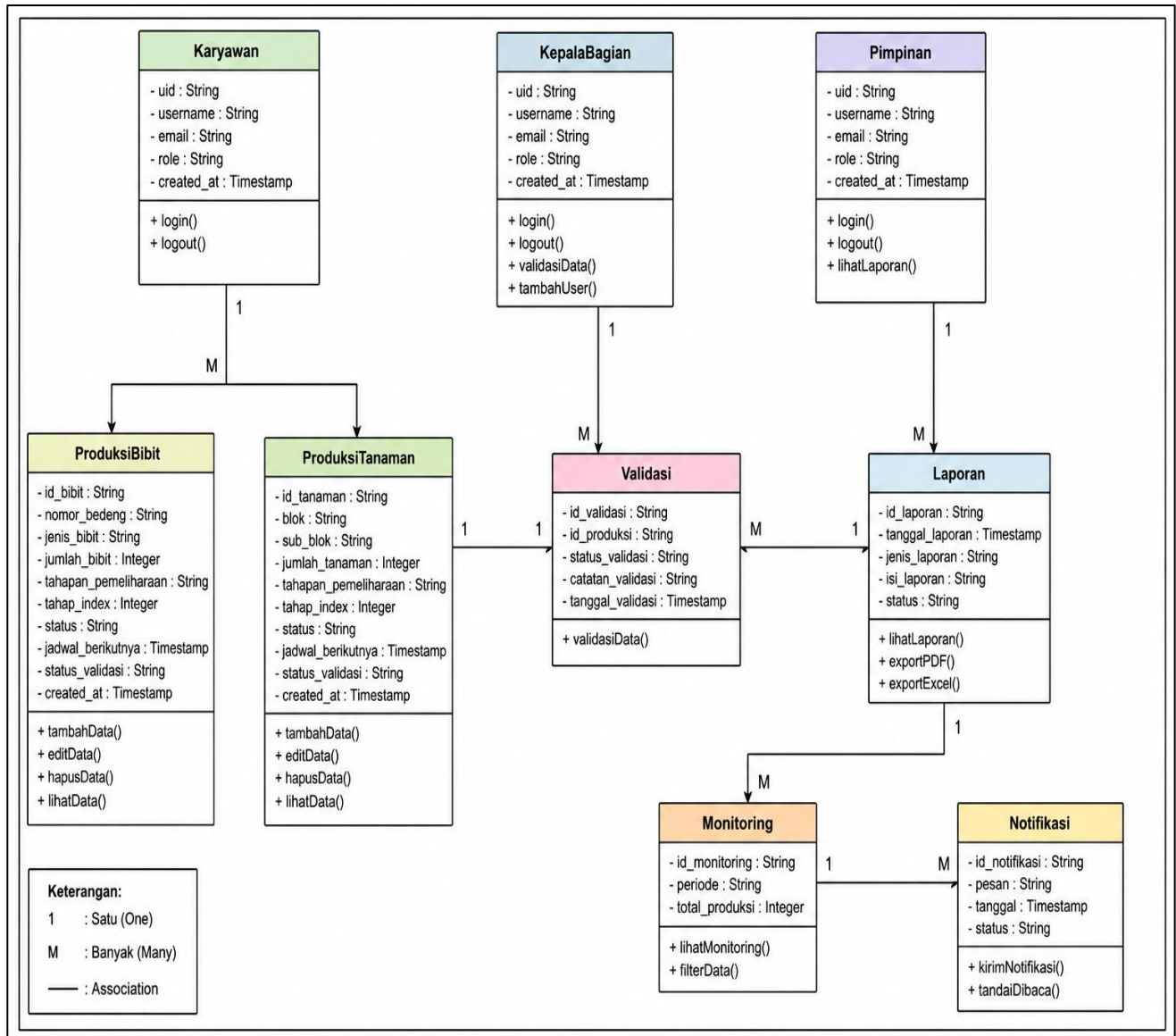
Sumber: data diolah oleh Penulis, 2026

Gambar 4.21 *Activity Diagram Logout Pimpinan yang diusulkan*

Berdasarkan gambar diatas, *activity diagram* ini menggambarkan proses pimpinan keluar dari sistem. Proses dimulai ketika pengguna memilih menu atau tombol *logout*. Sistem kemudian menampilkan konfirmasi untuk memastikan tindakan *logout*. Jika pengguna memilih tidak, maka proses dibatalkan dan pengguna kembali ke halaman sebelumnya. Namun, jika pengguna memilih ya, sistem akan mengakhiri sesi pengguna dan menghapus data sesi dari *database*. Setelah itu, sistem menampilkan pesan bahwa *logout* berhasil dan mengarahkan pengguna ke halaman *login*.

### 3. Class Diagram

Berdasarkan dari *usecase* dan *activity diagram* yang diusulkan, maka *class diagram* dari perancangan aplikasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android* yang diusulkan adalah sebagai berikut :



Sumber: data diolah oleh Penulis, 2026

Gambar 4.22 Class Diagram yang diusulkan

#### 4.2.2 Perancangan *Database* (Fisik)

Adapun beberap perancangan *database* yang akan digunakan pada pengembangan aplikasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android* yang diusulkan adalah sebagai berikut:

##### 1. Tabel Karyawan

Nama: karyawan

Fungsi: Digunakan untuk menyimpan data akun pengguna dengan *role* karyawan lapangan yang bertugas melakukan input dan *monitoring* data produksi bibit maupun tanaman karet..

Tabel 4.1 Tabel Karyawan

| Field Name | Data Type | Keterangan                     |
|------------|-----------|--------------------------------|
| username   | String    | Nama pengguna karyawan         |
| email      | String    | Email akun karyawan            |
| role       | String    | Role pengguna sebagai karyawan |
| created_at | Timestamp | Waktu pembuatan akun karyawan  |

##### 2. Tabel Kepala Bagian

Nama: kepala\_bagian

Fungsi: Digunakan untuk menyimpan data akun pengguna dengan *role* kepala bagian yang bertugas melakukan validasi data produksi dan manajemen pengguna.

Tabel 4.2 Tabel Kepala Bagian

| Field Name | Data Type | Keterangan                          |
|------------|-----------|-------------------------------------|
| username   | String    | Nama pengguna kepala bagian         |
| email      | String    | Email akun kepala bagian            |
| role       | String    | Role pengguna sebagai kepala bagian |
| created_at | Timestamp | Waktu pembuatan akun kepala bagian  |

### 3. Tabel Pimpinan

Nama: pimpinan

Fungsi: Digunakan untuk menyimpan data akun pengguna dengan *role* pimpinan yang bertugas memantau laporan dan hasil *monitoring* produksi tanaman karet.

Tabel 4.3 Tabel Pimpinan

| Field Name | Data Type | Keterangan                     |
|------------|-----------|--------------------------------|
| username   | String    | Nama pengguna pimpinan         |
| email      | String    | Email akun pimpinan            |
| role       | String    | Role pengguna sebagai pimpinan |
| created_at | Timestamp | Waktu pembuatan akun pimpinan  |

### 4. Tabel Data Produksi Bibit

Nama: produksi\_bibit

Fungsi: Digunakan untuk menyimpan data produksi bibit tanaman karet yang di *input* oleh karyawan lapangan.

Tabel 4.4 Tabel Produksi Bibit

| Field Name           | Data Type | Keterangan                  |
|----------------------|-----------|-----------------------------|
| nomor_bedeng         | String    | Nomor bedeng bibit          |
| jenis_bibit          | String    | Jenis bibit tanaman         |
| jumlah_bibit         | Integer   | Jumlah bibit                |
| tahapan_pemeliharaan | String    | Tahapan produksi saat ini   |
| tahap_index          | Integer   | Urutan tahapan produksi     |
| status               | String    | Status produksi             |
| jadwal_berikutnya    | Timestamp | Jadwal kegiatan berikutnya  |
| riwayat_tahapan      | Array     | Riwayat tahapan produksi    |
| realisasi            | Boolean   | Status realisasi kegiatan   |
| status_realisasi     | String    | Status pelaksanaan kegiatan |

| Field Name        | Data Type | Keterangan                    |
|-------------------|-----------|-------------------------------|
| tanggal_realisasi | Timestamp | Tanggal realisasi kegiatan    |
| catatan_realisasi | String    | Catatan realisasi kegiatan    |
| status_validasi   | String    | Status validasi kepala bagian |
| catatan_validasi  | String    | Catatan validasi              |
| created_at        | Timestamp | Waktu input data              |

##### 5. Tabel Data Produksi Tanaman

Nama: produksi\_tanaman

Fungsi: Untuk menyimpan data *monitoring* produksi tanaman karet yang telah ditanam di lapangan.

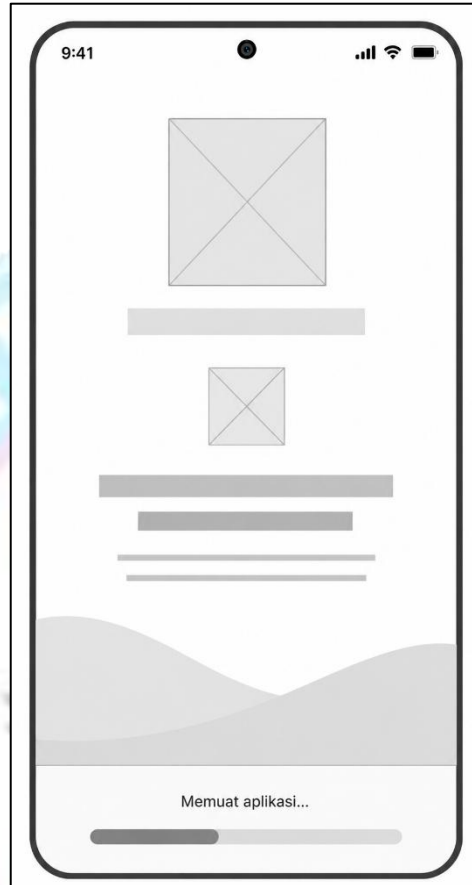
Tabel 4.5 Tabel Produksi Tanaman

| Field Name                  | Data Type | Keterangan                    |
|-----------------------------|-----------|-------------------------------|
| <b>blok</b>                 | String    | Blok tanaman                  |
| <b>sub_blok</b>             | String    | Sub blok tanaman              |
| <b>jumlah_tanaman</b>       | Integer   | Jumlah tanaman                |
| <b>tahapan_pemeliharaan</b> | String    | Tahapan produksi saat ini     |
| <b>tahap_index</b>          | Integer   | Urutan tahapan produksi       |
| <b>status</b>               | String    | Status produksi               |
| <b>jadwal_berikutnya</b>    | Timestamp | Jadwal kegiatan berikutnya    |
| <b>riwayat_tahapan</b>      | Array     | Riwayat tahapan produksi      |
| <b>realisasi</b>            | Boolean   | Status realisasi kegiatan     |
| <b>status_realisasi</b>     | String    | Status pelaksanaan kegiatan   |
| <b>tanggal_realisasi</b>    | Timestamp | Tanggal realisasi kegiatan    |
| <b>catatan_realisasi</b>    | String    | Catatan realisasi kegiatan    |
| <b>status_validasi</b>      | String    | Status validasi kepala bagian |
| <b>catatan_validasi</b>     | String    | Catatan validasi              |
| <b>created_at</b>           | Timestamp | Waktu input data              |

#### 4.2.3 Perancangan Antarmuka (UI/UX)

Berikut merupakan perancangan antarmuka (UI/UX) untuk pengembangan aplikasi *monitoring* informasi data produksi bibit tanaman karet.

##### 1. Halaman Awal



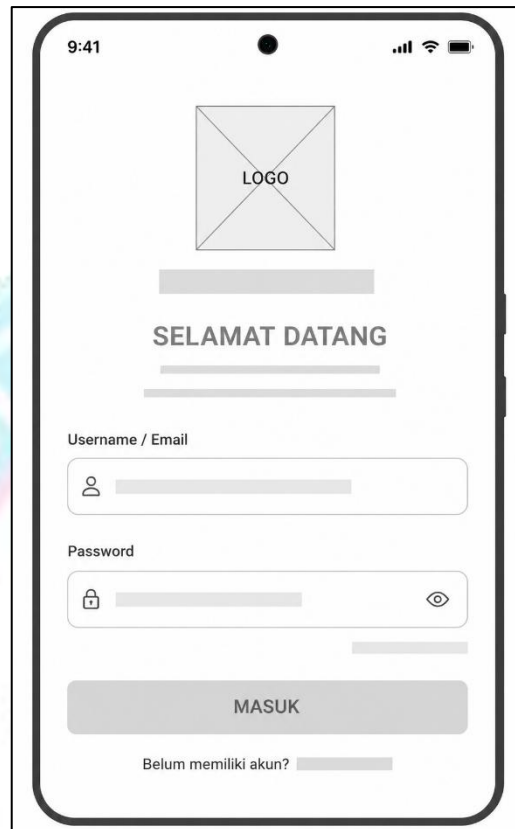
Sumber : data diolah oleh penulis, 2026

Gambar 4.23 Halaman Awal

Halaman awal atau *splash screen* merupakan tampilan pertama yang muncul saat aplikasi dijalankan. Pada halaman ini ditampilkan logo perusahaan PT Kendi Arindo sebagai identitas aplikasi, serta judul aplikasi yaitu *monitoring* produksi bibit tanaman karet. Selain itu, terdapat deskripsi singkat yang menjelaskan fungsi aplikasi sebagai sistem *monitoring* informasi data produksi. Di bagian bawah, ditampilkan indikator proses *loading* yang menunjukkan bahwa aplikasi sedang memuat sistem. Halaman ini dirancang dengan tampilan sederhana dan menggunakan tema warna hijau yang konsisten untuk memberikan kesan profesional serta sesuai dengan bidang pertanian. Setelah proses *loading* selesai,

sistem akan secara otomatis mengarahkan pengguna ke halaman *login* tanpa memerlukan interaksi tambahan.

## 2. Halaman *Login*



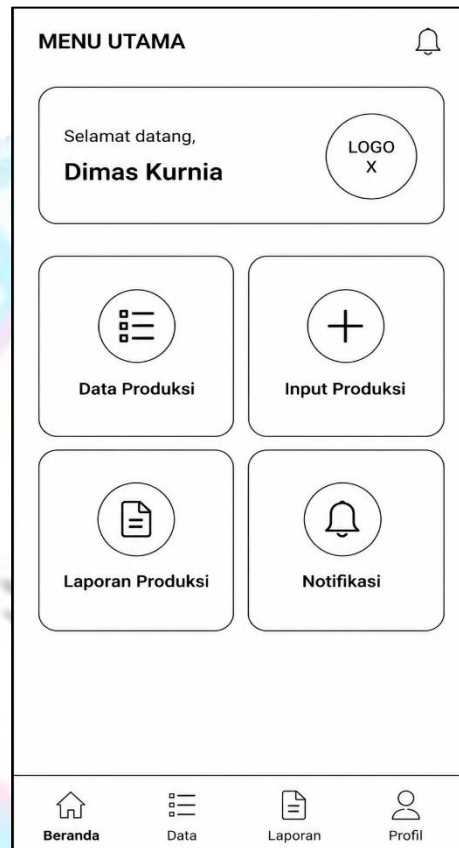
Sumber : data diolah oleh penulis, 2026

Gambar 4.24 Halaman *Login*

Halaman *login* merupakan halaman yang digunakan sebagai proses autentikasi pengguna sebelum dapat mengakses sistem aplikasi. Pada halaman ini ditampilkan identitas aplikasi berupa logo PT Kendi Arindo serta judul “Selamat Datang” sebagai tampilan pembuka. Pengguna diminta untuk memasukkan *email* atau email serta *password* pada kolom yang telah disediakan. Terdapat juga fitur “Lupa Password” yang dapat digunakan apabila pengguna mengalami kendala saat masuk ke dalam sistem. Setelah data diinput, pengguna dapat menekan tombol masuk untuk melanjutkan ke halaman utama aplikasi. Jika data yang dimasukkan benar, maka sistem akan mengarahkan pengguna ke halaman menu utama. Sebaliknya, jika terjadi kesalahan, sistem akan memberikan notifikasi kepada

pengguna. Selain itu, terdapat informasi tambahan bagi pengguna yang belum memiliki akun untuk menghubungi admin. Halaman ini dirancang dengan tampilan sederhana, menggunakan tema warna hijau yang konsisten, sehingga memudahkan pengguna dalam melakukan proses login dengan cepat dan efisien.

### 3. Halaman *Dashboard* Karyawan Lapangan



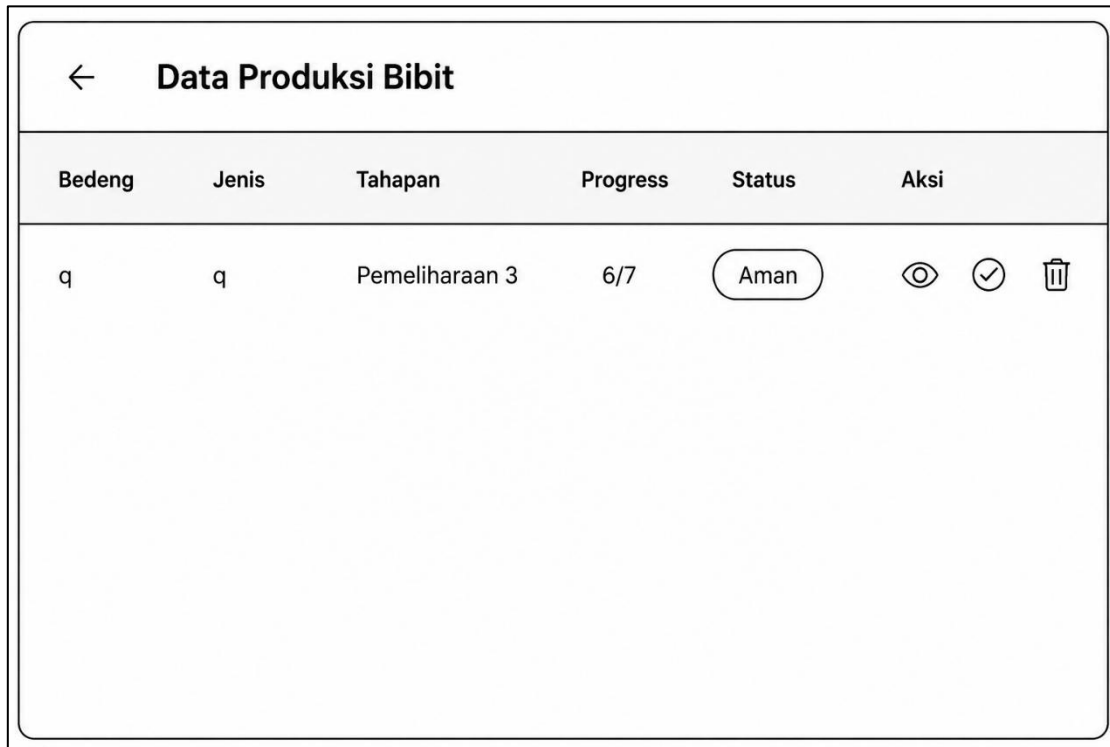
Sumber : data diolah oleh penulis, 2026

Gambar 4.25 Halaman *Dashboard* Karyawan Lapangan

Halaman *dashboard* karyawan dibuat sebagai rancangan awal tampilan antarmuka aplikasi sebelum proses implementasi dilakukan. Pada halaman ini terdapat beberapa menu utama seperti data produksi, input produksi, laporan produksi, dan notifikasi yang digunakan untuk membantu proses *monitoring* produksi bibit tanaman karet. Rancangan *wireframe* dibuat dengan tampilan sederhana berwarna hitam putih agar lebih fokus pada tata letak menu dan navigasi aplikasi. Selain itu, *wireframe* digunakan untuk mempermudah proses

pengembangan tampilan aplikasi sebelum diimplementasikan menggunakan Flutter.

#### 4. Halaman Data Produksi Bibit



| Bedeng | Jenis | Tahapan        | Progress | Status | Aksi                                                                                                                                                                                                                                                        |
|--------|-------|----------------|----------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| q      | q     | Pemeliharaan 3 | 6/7      | Aman   |    |

*Sumber : data diolah oleh penulis, 2026*

Gambar 4.26 Halaman Data Produksi Bibit

Halaman data produksi bibit digunakan untuk menampilkan seluruh data *monitoring* produksi bibit karet yang telah diinput oleh karyawan lapangan. Data ditampilkan dalam bentuk tabel agar lebih rapi dan memudahkan proses pencarian data ketika jumlah data produksi semakin banyak.

## 5. Halaman Detail Produksi Bibit

Detail Bibit

Bedeng q

q

Aman

Jumlah Bibit

522 Bibit

Tahapan Saat Ini

Pemeliharaan 3

Jadwal Berikutnya

20 Oct 2026

Progress

6/7

Status Realisasi

Belum

Catatan

-

Status Validasi

-

Catatan Validasi

-

Riwayat Tahapan

- Persiapan Bedeng
- Penyemaian
- Penyiraman
- Pemupukan
- Penyiangan
- Pengendalian Hama

*Sumber : data diolah oleh penulis, 2026*

Gambar 4.27 Halaman Detail Produksi Bibit

*Wireframe* halaman detail bibit merupakan rancangan awal tampilan halaman yang digunakan untuk menampilkan informasi detail data produksi bibit. Pada halaman ini ditampilkan informasi seperti jumlah bibit, tahapan saat ini, jadwal berikutnya, *progress* produksi, status realisasi, status validasi, serta riwayat tahapan produksi bibit.

## 6. Halaman Data Produksi Tanaman

| ← Data Produksi Tanaman |     |              |          |        |      |   |
|-------------------------|-----|--------------|----------|--------|------|---|
| Blok                    | Sub | Tahapan      | Progress | Status | Aksi |   |
| 1                       | 1   | Penyiangan 1 | 0/26     | Aman   | 👁️   | ✅ |
| 9                       | 9   | Semprot 1    | 1/26     | Aman   | 👁️   | ✅ |
|                         |     |              |          |        |      |   |

Sumber : data diolah oleh penulis, 2026

Gambar 4.28 Halaman Data Produksi Tanaman

Halaman data produksi tanaman digunakan untuk menampilkan seluruh data *monitoring* produksi tanaman karet yang telah ditanam di lapangan. Data ditampilkan dalam bentuk tabel agar memudahkan proses *monitoring* dan pengelolaan data produksi tanaman. Informasi yang ditampilkan meliputi blok tanaman, sub blok, tahapan pemeliharaan, progress produksi, status produksi, dan aksi detail data tanaman.

## 7. Halaman Detail Produksi Tanaman

**Detail Tanaman**

**Blok 1** Tertunda

Sub Blok : 1

Jumlah Tanaman **1050 Tanaman**

Tahapan Saat Ini **Semprot 1**

Jadwal Berikutnya **24 Nov 2026**

Progress **1/26**

Status Realisasi **Tertunda**

Catatan **Semprot 1 ditunda**

**Riwayat Tahapan**

- ☒ Penyiangan 1
- ☐ Semprot 1
- ☐ Semprot 2
- ☐ Pemupukan 1
- ☐ Semprot 3
- ☐ Pemupukan 2
- ☐ Pembokoran
- ☐ Pewiwilan
- ☐ Pemberantasan Hama & Penyakit
- ☐ Semprot 4
- ☐ Pemupukan 3
- ☐ Semprot 5
- ☐ Pemupukan 4
- ☐ Semprot 6
- ☐ Pemupukan 5
- ☐ Pemberantasan Hama & Penyakit
- ☐ Semprot 7
- ☐ Pemupukan 6
- ☐ Semprot 8
- ☐ Pemupukan 7
- ☐ Semprot 9
- ☐ Pemupukan 8
- ☐ Semprot 10
- ☐ Pemupukan 9
- ☐ Semprot 11

Sumber : data diolah oleh penulis, 2026

Gambar 4.29 Halaman Detail Produksi Tanaman

Halaman detail produksi tanaman digunakan untuk menampilkan informasi detail mengenai data produksi tanaman yang dipilih pengguna. Pada halaman ini ditampilkan riwayat tahapan produksi, jadwal kegiatan berikutnya, jumlah tanaman, serta status realisasi kegiatan produksi tanaman.

## 8. Halaman *Input* Data Produksi Bibit

← **Input Produksi**

**Data Bibit** **Data Tanaman**

**Tanggal Input**

25 May 2026

**Nomor Bedeng**

Masukkan nomor bedeng

**Jenis Bibit**

Masukkan jenis bibit

**Jumlah Bibit**

Masukkan jumlah bibit

**SIMPAN DATA**

Sumber : data diolah oleh penulis, 2026

Gambar 4.30 Halaman *Input* Data Produksi Bibit

*Wireframe* halaman input produksi merupakan rancangan awal tampilan halaman yang digunakan untuk menambahkan data produksi bibit ke dalam sistem. Pada halaman ini terdapat beberapa form input seperti tanggal *input*, nomor bedeng, jenis bibit, dan jumlah bibit. Tampilan *wireframe* dibuat sederhana dengan konsep hitam putih agar lebih fokus pada tata letak *form*, tombol, dan navigasi sebelum diimplementasikan ke dalam aplikasi menggunakan Flutter.

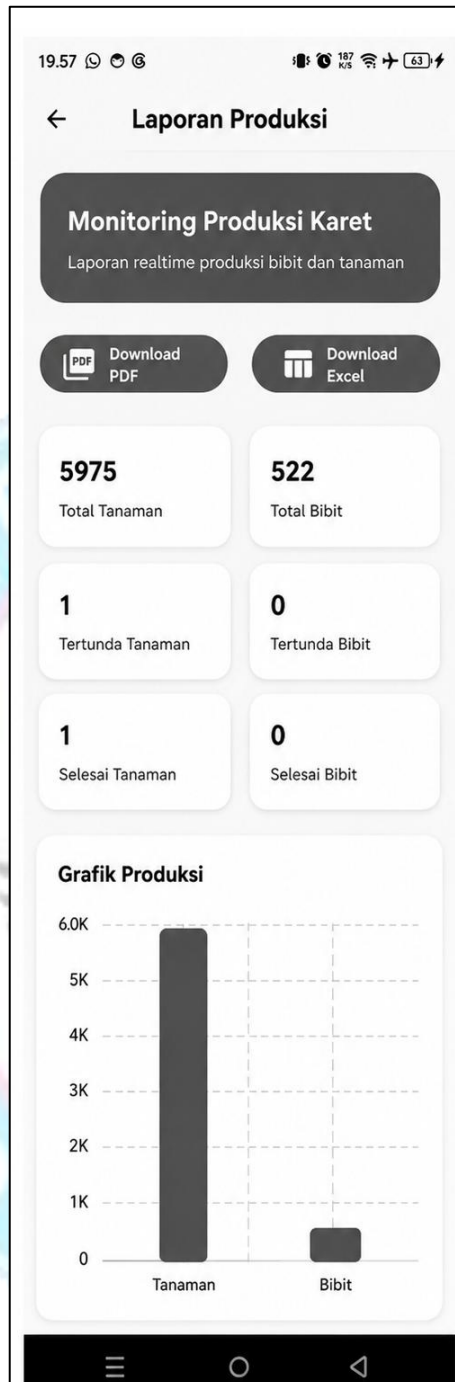
## 9. Halaman *Input* Data Produksi Tanaman

Sumber : data diolah oleh penulis, 2026

Gambar 4.31 Halaman *Input* Data Produksi Tanaman

*Wireframe* halaman input produksi tanaman merupakan rancangan awal tampilan halaman yang digunakan untuk menambahkan data produksi tanaman ke dalam sistem. Pada halaman ini terdapat beberapa *form input* seperti tanggal tanam, blok, sub blok, petak, luas lahan, dan jarak tanam. Tampilan *wireframe* dibuat sederhana dengan konsep hitam putih agar lebih fokus pada tata letak *form* dan navigasi sebelum diimplementasikan ke dalam aplikasi menggunakan Flutter.

## 10. Halaman Laporan



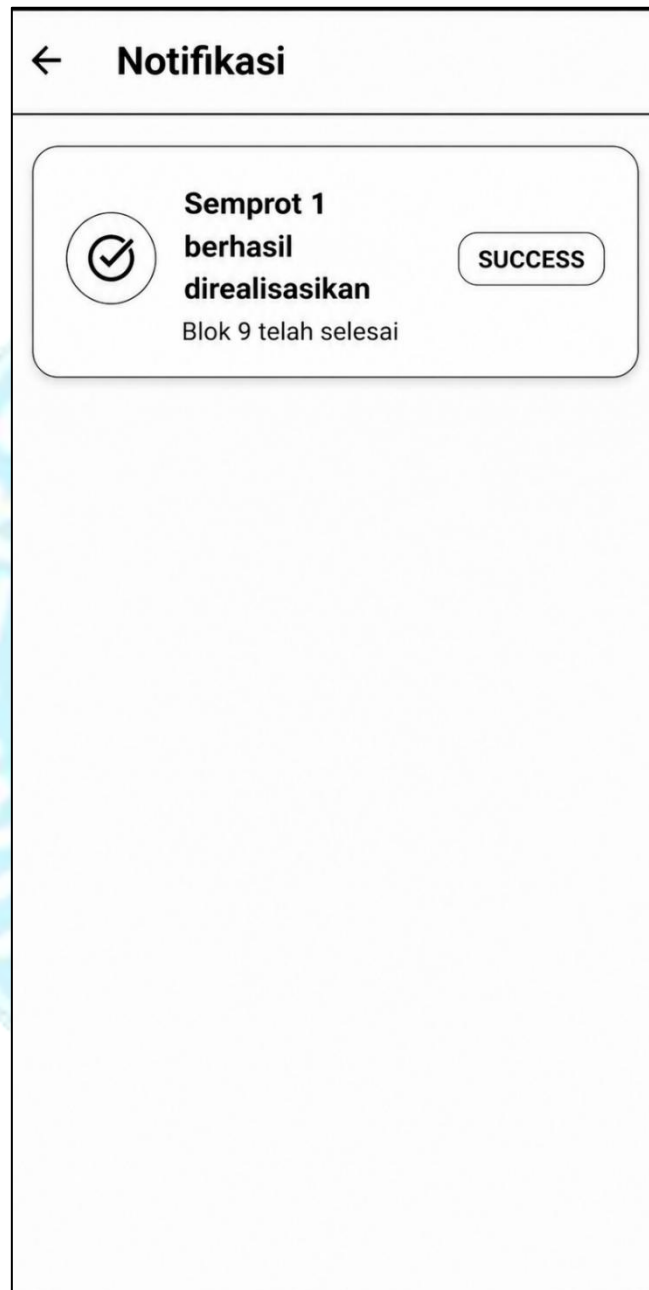
Sumber : data diolah oleh penulis, 2026

Gambar 4.32 Halaman Laporan

Halaman laporan digunakan untuk menampilkan hasil rekapitulasi data produksi bibit dan tanaman karet yang telah tersimpan pada sistem. Pada halaman ini data ditampilkan dalam bentuk tabel dan grafik untuk mempermudah pengguna dalam membaca informasi hasil produksi. Laporan produksi berisi informasi

jumlah produksi bibit, jumlah produksi tanaman, data produksi yang telah divalidasi, serta statistik hasil *monitoring* produksi secara keseluruhan.

#### 11. Halaman Notifikasi



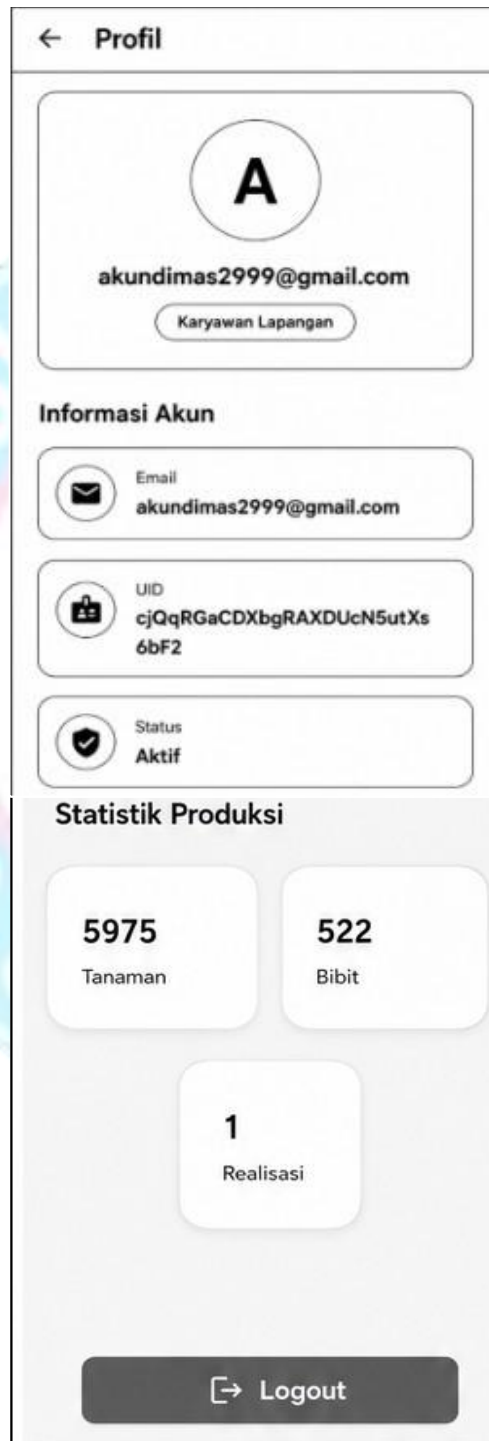
Sumber : data diolah oleh penulis, 2026

Gambar 4.33 Halaman Notifikasi

Halaman notifikasi digunakan untuk menampilkan informasi pengingat jadwal kegiatan produksi yang akan datang. Sistem akan memberikan notifikasi

berdasarkan jadwal tahapan produksi bibit maupun tanaman yang mendekati waktu pelaksanaan.

## 12. Halaman Profil Karyawan

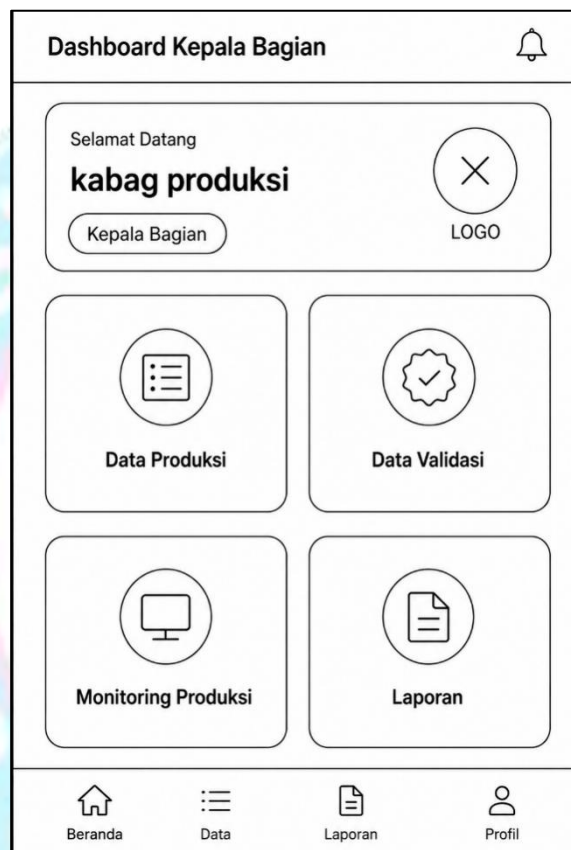


Sumber : data diolah oleh penulis, 2026

Gambar 4.34 Halaman Profil Karyawan

Halaman profil digunakan untuk menampilkan informasi akun pengguna yang sedang *login* ke dalam aplikasi. Pada halaman ini pengguna dapat melihat informasi *username*, *email*, dan *role* pengguna. Selain itu, halaman profil juga menyediakan fitur *logout* yang digunakan untuk keluar dari aplikasi dan kembali menuju halaman *login*.

### 13. Halaman *Dashboard* Kepala Bagian



Sumber : data diolah oleh penulis, 2026

Gambar 4.35 Halaman *Dashboard* Kepala Bagian

Halaman *dashboard* kepala bagian merupakan rancangan awal tampilan halaman utama setelah pengguna berhasil *login* sebagai kepala bagian. Pada halaman ini terdapat beberapa menu utama seperti data produksi, data validasi, *monitoring* produksi, dan laporan. Tampilan *wireframe* dibuat sederhana dengan konsep hitam putih agar lebih fokus pada tata letak menu dan navigasi sebelum diimplementasikan ke dalam aplikasi menggunakan Flutter.

## 14. Halaman Data Produksi Bibit

| <div> <span>←</span> <b>Data Produksi Bibit</b> </div> |       |                |          |        |                                                                                                                                                                                                                                                             |
|--------------------------------------------------------|-------|----------------|----------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Bedeng                                                 | Jenis | Tahapan        | Progress | Status | Aksi                                                                                                                                                                                                                                                        |
| q                                                      | q     | Pemeliharaan 3 | 6/7      | Aman   |    |

*Sumber : data diolah oleh penulis, 2026*

Gambar 4.36 Halaman Data Produksi Bibit

Halaman data produksi bibit digunakan untuk menampilkan seluruh data *monitoring* produksi bibit karet yang telah diinput oleh karyawan lapangan. Data ditampilkan dalam bentuk tabel agar lebih rapi dan memudahkan proses pencarian data ketika jumlah data produksi semakin banyak.

## 15. Halaman Detail Produksi Bibit

←

Detail Bibit



Bedeng q

q

✓

Aman

🌿

Jumlah Bibit

522 Bibit

🕒

Tahapan Saat Ini

Pemeliharaan 3

📅

Jadwal Berikutnya

20 Oct 2026

📊

Progress

6/7

📋

Status Realisasi

Belum

📄

Catatan

-

📄

Status Validasi

-

📄

Catatan Validasi

-

Riwayat Tahapan

○

Persiapan Bedeng

●

Penyemaian

○

Penyiraman

○

Pemupukan

○

Penyiangan

○

Pengendalian Hama

Sumber : data diolah oleh penulis, 2026

Gambar 4.37 Halaman Detail Produksi Bibit

*Wireframe* halaman detail bibit merupakan rancangan awal tampilan halaman yang digunakan untuk menampilkan informasi detail data produksi bibit. Pada halaman ini ditampilkan informasi seperti jumlah bibit, tahapan saat ini, jadwal berikutnya, *progress* produksi, status realisasi, status validasi, serta riwayat tahapan produksi bibit.

## 16. Halaman Data Produksi Tanaman

| ← Data Produksi Tanaman |     |              |          |        |      |   |
|-------------------------|-----|--------------|----------|--------|------|---|
| Blok                    | Sub | Tahapan      | Progress | Status | Aksi |   |
| 1                       | 1   | Penyiangan 1 | 0/26     | Aman   | 👁️   | ✅ |
| 9                       | 9   | Semprot 1    | 1/26     | Aman   | 👁️   | ✅ |
|                         |     |              |          |        |      |   |

Sumber : data diolah oleh penulis, 2026

Gambar 4.38 Halaman Data Produksi Tanaman

Halaman data produksi tanaman digunakan untuk menampilkan seluruh data *monitoring* produksi tanaman karet yang telah ditanam di lapangan. Data ditampilkan dalam bentuk tabel agar memudahkan proses *monitoring* dan pengelolaan data produksi tanaman. Informasi yang ditampilkan meliputi blok tanaman, sub blok, tahapan pemeliharaan, progress produksi, status produksi, dan aksi detail data tanaman.

## 17. Halaman Detail Produksi Tanaman

**Detail Tanaman**

**Blok 1** Tertunda

Sub Blok : 1

Jumlah Tanaman **1050 Tanaman**

Tahapan Saat Ini **Semprot 1**

Jadwal Berikutnya **24 Nov 2026**

Progress **1/26**

Status Realisasi **Tertunda**

Catatan **Semprot 1 ditunda**

**Riwayat Tahapan**

- ☒ Penyiangan 1
- ☐ Semprot 1
- ☐ Semprot 2
- ☐ Pemupukan 1
- ☐ Semprot 3
- ☐ Pemupukan 2
- ☐ Pembokoran
- ☐ Pewiwilan
- ☐ Pemberantasan Hama & Penyakit
- ☐ Semprot 4
- ☐ Pemupukan 3
- ☐ Semprot 5
- ☐ Pemupukan 4
- ☐ Semprot 6
- ☐ Pemupukan 5
- ☐ Pemberantasan Hama & Penyakit
- ☐ Semprot 7
- ☐ Pemupukan 6
- ☐ Semprot 8
- ☐ Pemupukan 7
- ☐ Semprot 9
- ☐ Pemupukan 8
- ☐ Semprot 10
- ☐ Pemupukan 9
- ☐ Semprot 11

Sumber : data diolah oleh penulis, 2026

Gambar 4.39 Halaman Detail Produksi Tanaman

Halaman detail produksi tanaman digunakan untuk menampilkan informasi detail mengenai data produksi tanaman yang dipilih pengguna. Pada halaman ini ditampilkan riwayat tahapan produksi, jadwal kegiatan berikutnya, jumlah tanaman, serta status realisasi kegiatan produksi tanaman.

## 18. Halaman Validasi Produksi

← **Data Validasi**

**Validasi Tanaman**

 Belum ada validasi tanaman

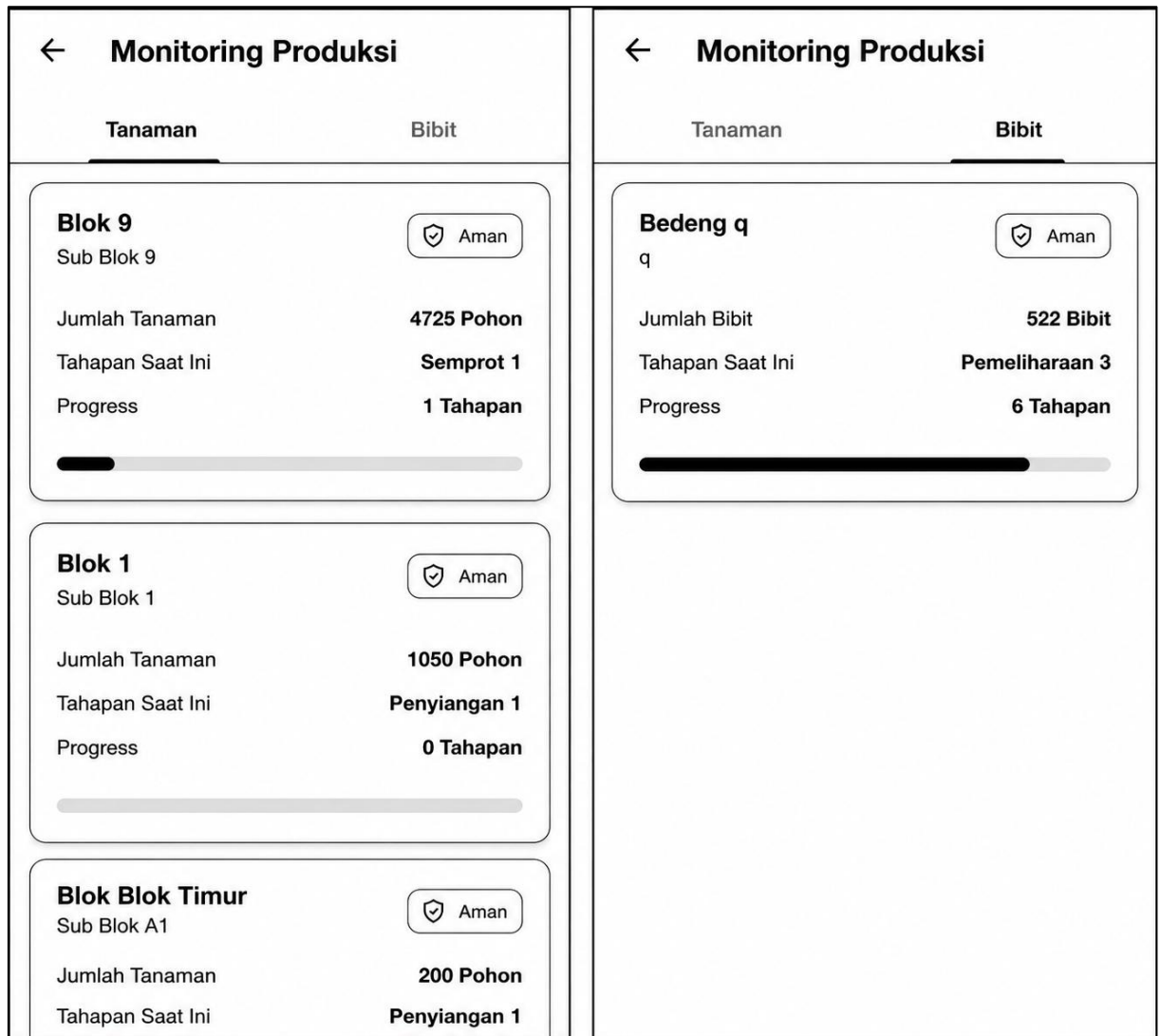
**Validasi Bibit**

 Belum ada validasi bibit

*Sumber : data diolah oleh penulis, 2026*

Gambar 4.40 Halaman Validasi Produksi

Halaman validasi produksi digunakan oleh kepala bagian untuk melakukan pengecekan dan validasi terhadap data produksi yang telah diselesaikan oleh karyawan lapangan. Proses validasi dilakukan untuk memastikan bahwa seluruh tahapan produksi telah dilaksanakan sesuai dengan prosedur perusahaan.

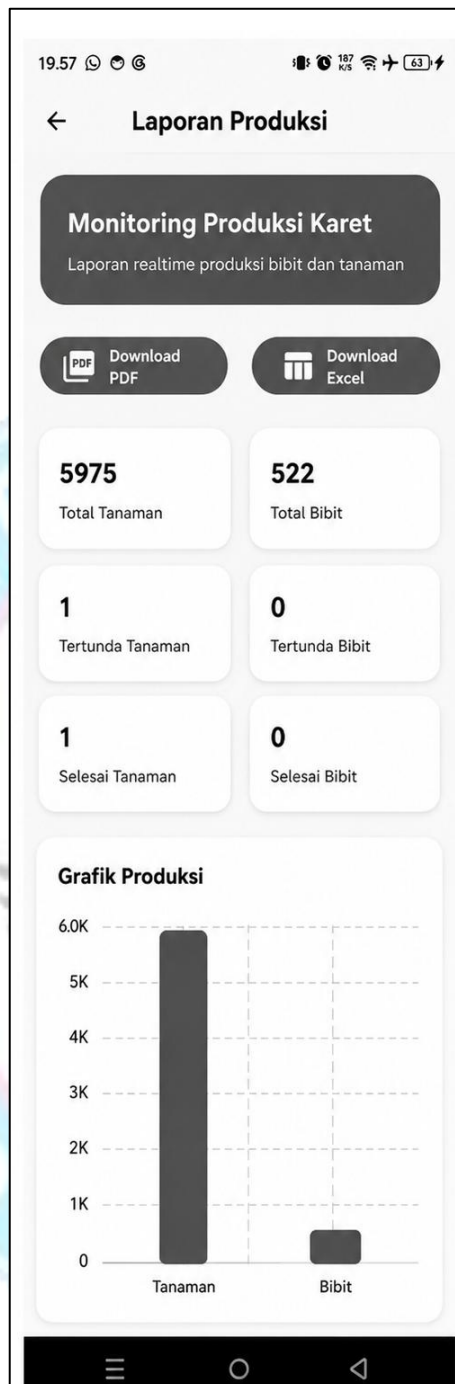
19. Halaman *Monitoring* Produksi

Sumber : data diolah oleh penulis, 2026

Gambar 4.41 Halaman *Monitoring* Produksi

Halaman *monitoring* produksi digunakan untuk memantau seluruh proses produksi bibit dan tanaman karet yang sedang berjalan secara *realtime*. Pada halaman ini pengguna dapat melihat perkembangan tahapan produksi, status realisasi kegiatan, jadwal kegiatan berikutnya, serta *progress* pekerjaan yang telah dilakukan oleh karyawan lapangan.

## 20. Halaman Laporan



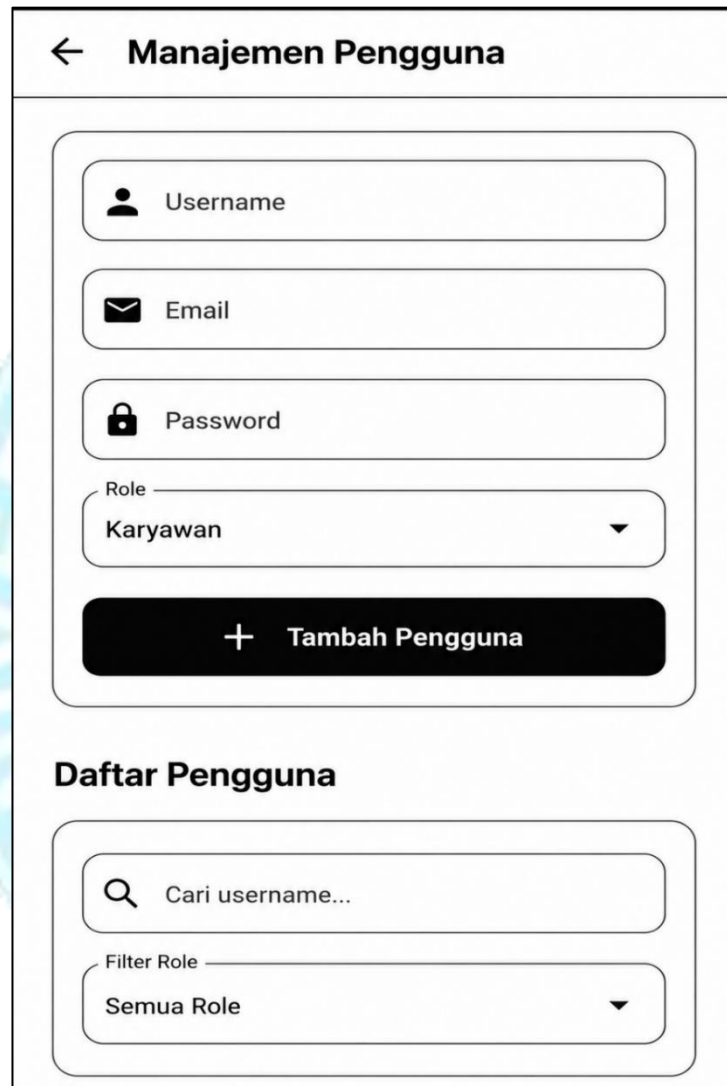
Sumber : data diolah oleh penulis, 2026

Gambar 4.42 Halaman Laporan

Halaman laporan digunakan untuk menampilkan hasil rekapitulasi data produksi bibit dan tanaman karet yang telah tersimpan pada sistem. Pada halaman ini data ditampilkan dalam bentuk tabel dan grafik untuk mempermudah pengguna

dalam membaca informasi hasil produksi. Laporan produksi berisi informasi jumlah produksi bibit, jumlah produksi tanaman, data produksi yang telah divalidasi, serta statistik hasil *monitoring* produksi secara keseluruhan.

## 21. Halaman Manajemen Pengguna



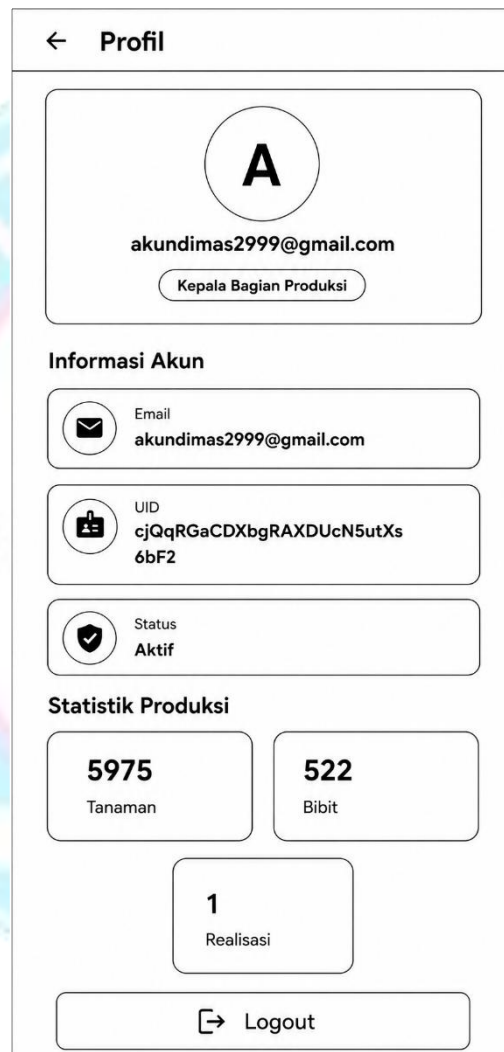
Sumber : data diolah oleh penulis, 2026

Gambar 4.43 Halaman Manajemen Pengguna

Halaman manajemen pengguna merupakan halaman yang digunakan oleh kepala bagian untuk mengelola seluruh akun pengguna yang dapat mengakses aplikasi digitalisasi *monitoring* Informasi data produksi bibit tanaman karet berbasis *android*. Halaman ini dirancang untuk mempermudah proses pengelolaan akun pengguna tanpa perlu melakukan penginputan data secara langsung melalui

*firebase Firestore*. Pada halaman ini kepala bagian dapat menambahkan akun pengguna baru dengan mengisi data seperti *username*, *email*, *password*, dan *role* pengguna. Sistem akan secara otomatis menyimpan data akun ke *firebase authentication* dan *firebase firestore* sehingga akun dapat langsung digunakan untuk *login* ke dalam aplikasi.

## 22. Halaman Profil Kepala Bagian



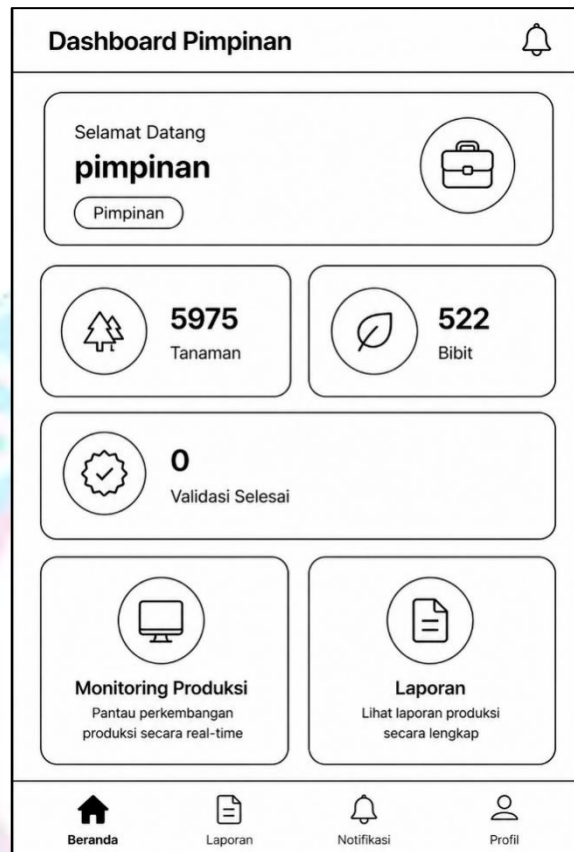
Sumber : data diolah oleh penulis, 2026

Gambar 4.44 Halaman Profil Kepala Bagian

Halaman profil digunakan untuk menampilkan informasi akun pengguna yang sedang *login* ke dalam aplikasi. Pada halaman ini pengguna dapat melihat informasi *username*, *email*, dan *role* pengguna. Selain itu, halaman profil juga

menyediakan fitur *logout* yang digunakan untuk keluar dari aplikasi dan kembali menuju halaman *login*.

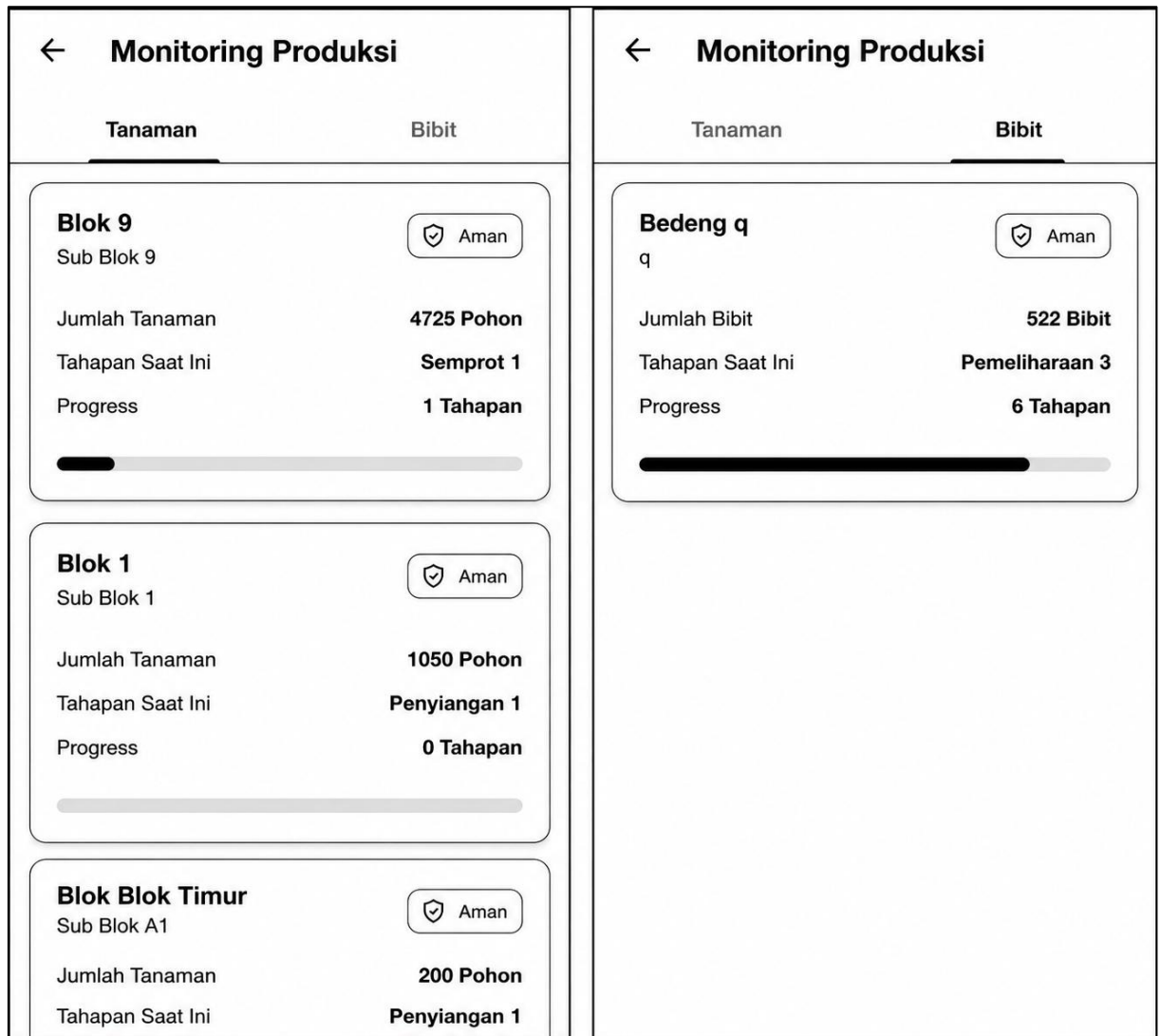
### 23. Halaman *Dashboard* Pimpinan



Sumber : data diolah oleh penulis, 2026

Gambar 4.45 Halaman *Dashboard* Pimpinan

Halaman *dashboard* pimpinan merupakan rancangan awal tampilan halaman utama yang digunakan oleh pimpinan untuk melakukan *monitoring* hasil produksi. Pada halaman ini ditampilkan informasi ringkasan produksi seperti jumlah tanaman, jumlah bibit, validasi selesai, *monitoring* produksi, dan laporan. Tampilan *wireframe* dibuat sederhana dengan konsep hitam putih agar lebih fokus pada struktur tata letak menu dan navigasi sebelum diimplementasikan ke dalam aplikasi menggunakan Flutter.

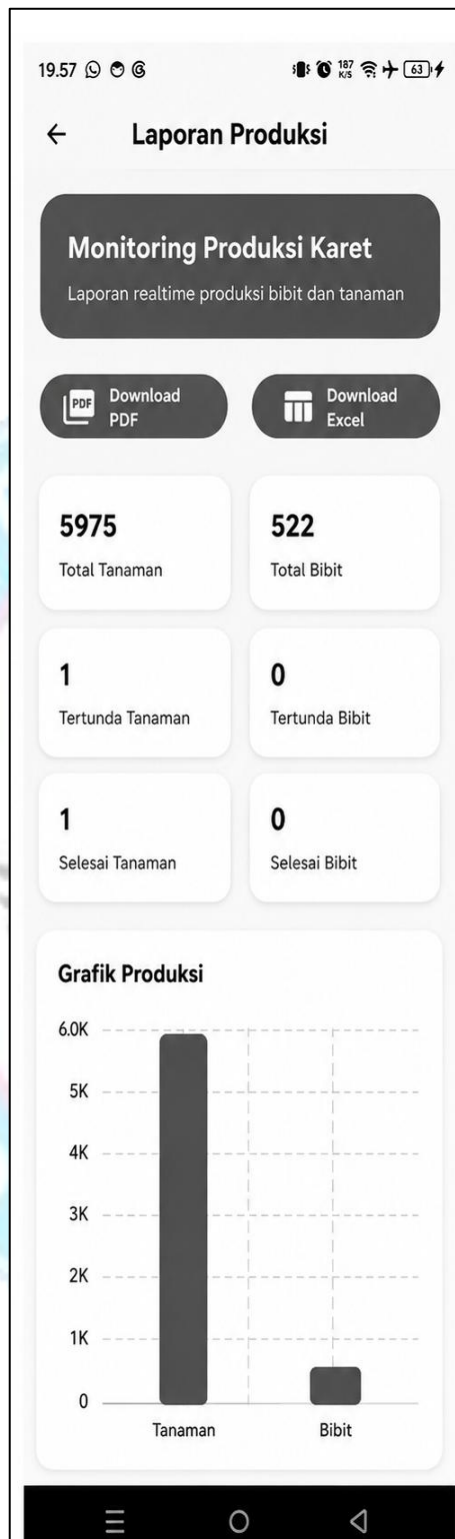
24. Halaman *Monitoring* Produksi

Sumber : data diolah oleh penulis, 2026

Gambar 4.46 Halaman *Monitoring* Produksi

Halaman *monitoring* produksi digunakan untuk memantau seluruh proses produksi bibit dan tanaman karet yang sedang berjalan secara *realtime*. Pada halaman ini pengguna dapat melihat perkembangan tahapan produksi, status realisasi kegiatan, jadwal kegiatan berikutnya, serta *progress* pekerjaan yang telah dilakukan oleh karyawan lapangan.

## 25. Halaman Laporan

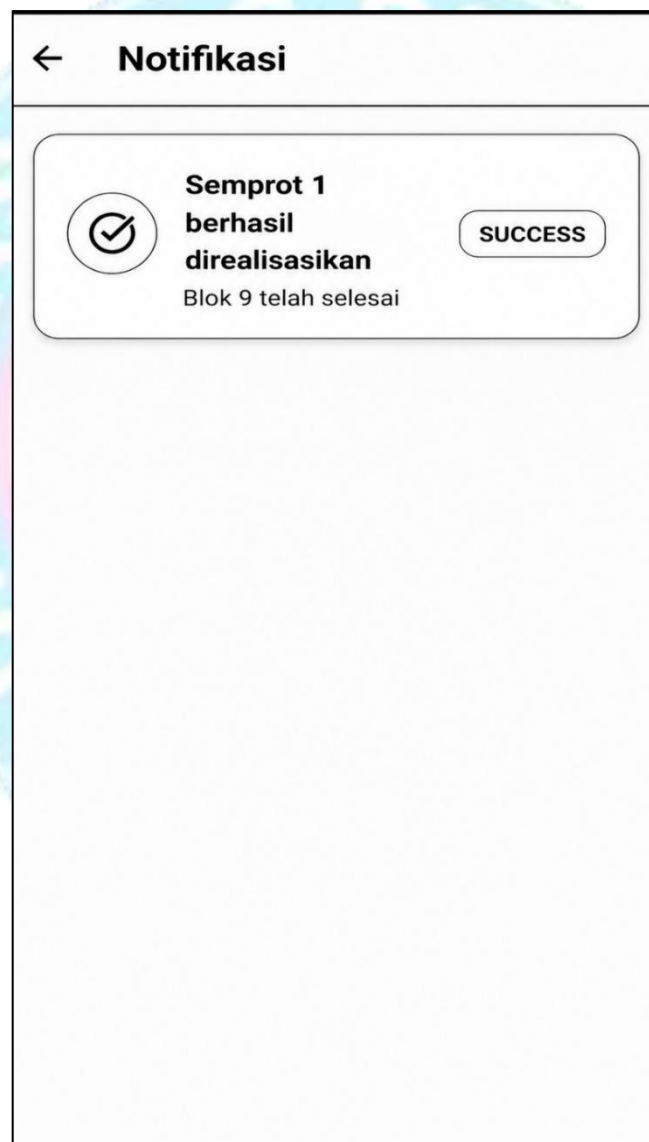


Sumber : data diolah oleh penulis, 2026

Gambar 4.47 Halaman Laporan

Halaman laporan digunakan untuk menampilkan hasil rekapitulasi data produksi bibit dan tanaman karet yang telah tersimpan pada sistem. Pada halaman ini data ditampilkan dalam bentuk tabel dan grafik untuk mempermudah pengguna dalam membaca informasi hasil produksi. Laporan produksi berisi informasi jumlah produksi bibit, jumlah produksi tanaman, data produksi yang telah divalidasi, serta statistik hasil *monitoring* produksi secara keseluruhan.

#### 26. Halaman Notifikasi

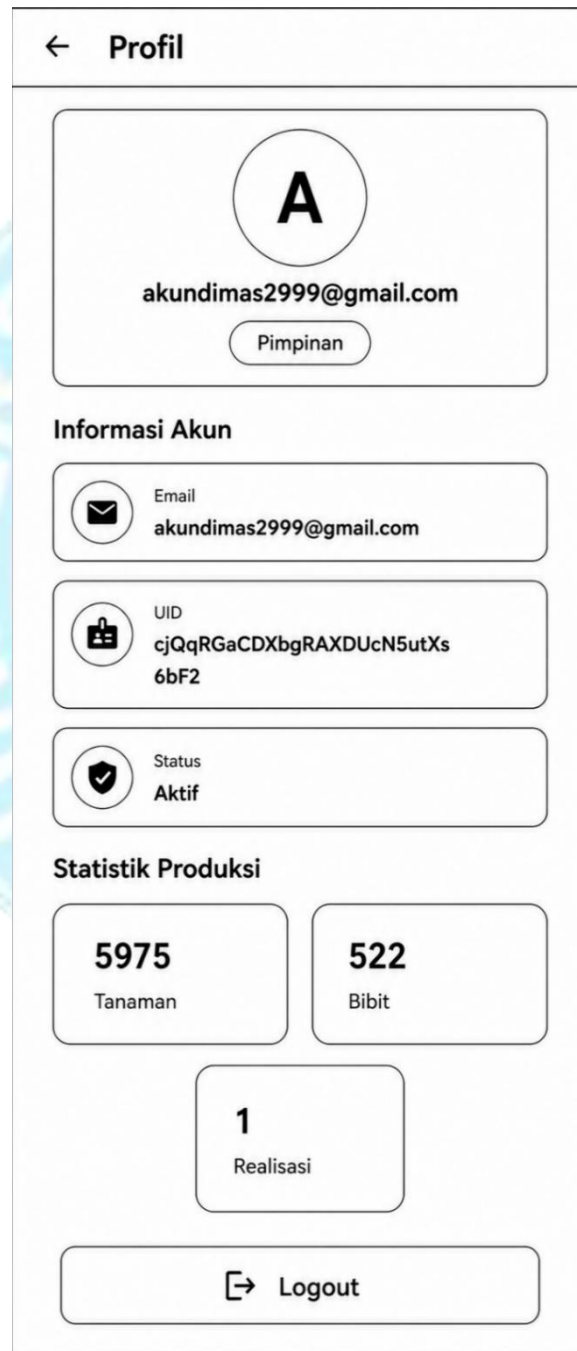


Sumber : data diolah oleh penulis, 2026

Gambar 4.48 Halaman Notifikasi

Halaman notifikasi digunakan untuk menampilkan informasi pengingat jadwal kegiatan produksi yang akan datang. Sistem akan memberikan notifikasi berdasarkan jadwal tahapan produksi bibit maupun tanaman yang mendekati waktu pelaksanaan.

## 27. Halaman Profil Pimpinan



← **Profil**

A

akundimas2999@gmail.com

Pimpinan

**Informasi Akun**

Email

akundimas2999@gmail.com

UID

cjQqRGaCDXbgRAXDUcN5utXs6bF2

Status

Aktif

**Statistik Produksi**

5975

Tanaman

522

Bibit

1

Realisasi

Logout

Sumber : data diolah oleh penulis, 2026

Gambar 4.49 Halaman Profil Pimpinan

Halaman profil digunakan untuk menampilkan informasi akun pengguna yang sedang *login* ke dalam aplikasi. Pada halaman ini pengguna dapat melihat informasi *username*, *email*, dan *role* pengguna. Selain itu, halaman profil juga menyediakan fitur *logout* yang digunakan untuk keluar dari aplikasi dan kembali menuju halaman *login*.



## **BAB V**

### **HASIL DAN PEMBAHASAN**

#### **5.1 Spesifikasi Perangkat**

Pada tahap ini dilakukan implementasi sistem digitalisasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android* sesuai dengan hasil analisis dan perancangan sistem pada bab sebelumnya. Implementasi dilakukan menggunakan *framework* flutter dengan bahasa pemrograman dart serta *firebase* sebagai media penyimpanan *database* secara *realtime*. Sistem ini dirancang untuk membantu proses *monitoring* dan pengelolaan data produksi bibit tanaman karet pada PT. Kendi Arindo agar lebih efektif dan efisien.

##### **5.1.1. Perangkat Keras Yang Digunakan**

Perangkat keras yang digunakan dalam proses pengembangan dan implementasi aplikasi digitalisasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android* dapat dilihat pada tabel berikut.:

1. Laptop Acer Aspire 514-55
2. *Processor* : Intel Core i5-1235U Gen 12
3. *RAM* : 8.00 GB
4. *Storage* : 512 GB
5. *Handphone Android OS* 14
6. *Keyboard*
7. Kabel *Usb type c*
8. *Mouse*

##### **5.1.2. Perangkat Lunak yang Digunakan**

Untuk mendukung sistem yang diusulkan berjalan dengan optimal, dibutuhkan *software* pengolah data, adapun perangkat lunak yang digunakan untuk mendukung pembuatan program aplikasi ini sebagai berikut :

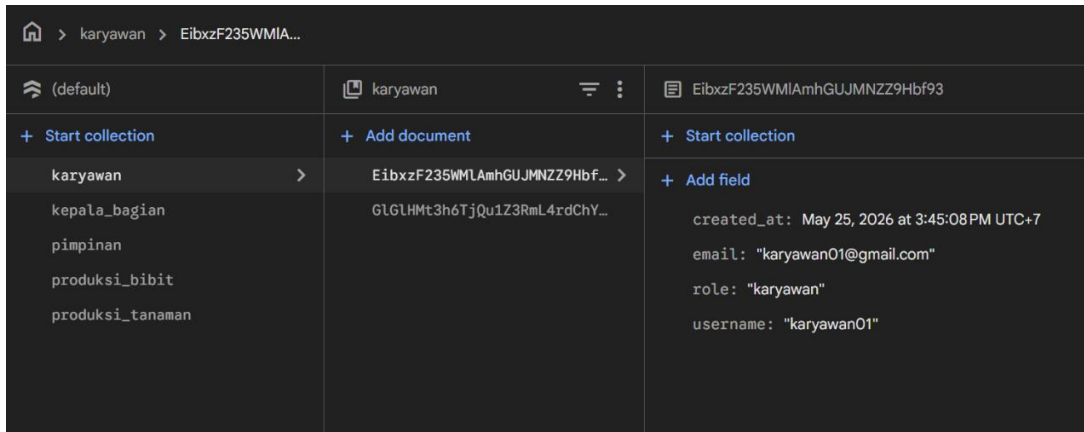
1. Sistem Operasi : *Windows 11 Home Single*
2. Bahasa Pemrograman: *Dart SDK version: 3.134.0*
3. *Framework : Flutter 3.134.0*
4. *Firebase\_core : 4.7.0*
5. *Cloud\_firestore : 6.3.0*
6. *Firebase\_auth : 6.4.0*
7. *Code Editor : Visual Studio Code 1.121.0*

## 5.2 Implementasi Database

Implementasi *database* pada aplikasi digitalisasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android* menggunakan *firebase firestore* sebagai media penyimpanan data secara *realtime*. *Database* digunakan untuk menyimpan seluruh informasi yang berkaitan dengan data pengguna, data produksi bibit, data produksi tanaman, *monitoring* tahapan produksi, laporan, serta validasi data produksi. *Firebase firestore* dipilih karena memiliki kemampuan penyimpanan data berbasis *cloud* yang dapat diakses secara *realtime*, sehingga memudahkan proses *monitoring* data produksi secara cepat dan efisien.

Selain itu, *firestore* juga mendukung integrasi dengan *framework* flutter yang digunakan dalam pengembangan aplikasi ini. Pada implementasinya, *database* dibagi menjadi beberapa *collection* utama yang memiliki fungsi berbeda sesuai kebutuhan sistem. *Collection* tersebut antara lain *collection users* untuk menyimpan data pengguna, *collection produksi\_bibit* untuk menyimpan data produksi bibit, dan *collection produksi\_tanaman* untuk menyimpan data produksi tanaman.

### 5.2.1. Implementasi Tabel Karyawan



| Collection         | Document                        | Fields                                       |
|--------------------|---------------------------------|----------------------------------------------|
| (default)          | karyawan                        | EibxzF235WMLAmhGUJMNZZ9Hbf93                 |
| + Start collection | + Add document                  | + Start collection                           |
| karyawan >         | EibxzF235WMLAmhGUJMNZZ9Hbf... > | + Add field                                  |
| kepala_bagian      | GLGLHMT3h6TjQu1Z3RmL4rdChY...   | created_at: May 25, 2026 at 3:45:08 PM UTC+7 |
| pimpinan           |                                 | email: "karyawan01@gmail.com"                |
| produksi_bibit     |                                 | role: "karyawan"                             |
| produksi_tanaman   |                                 | username: "karyawan01"                       |

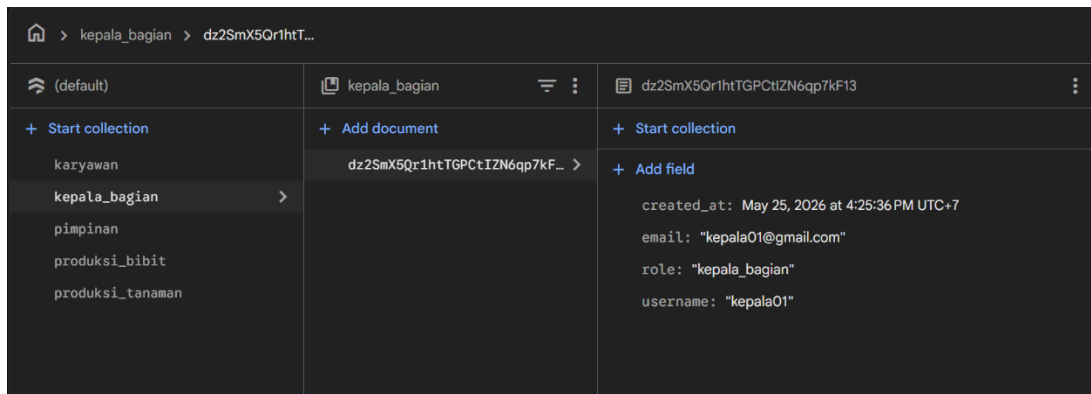
Sumber : data diolah oleh penulis, 2026

Gambar 5.1 Tabel Karyawan

*Collection* karyawan digunakan untuk menyimpan data akun pengguna dengan role sebagai karyawan lapangan pada aplikasi monitoring produksi bibit dan tanaman karet berbasis *android*. Setiap data pengguna disimpan dalam bentuk dokumen dengan uid yang diperoleh dari *firebase authentication* sebagai identitas unik pengguna. Pada *collection* ini terdapat beberapa *field* seperti *username*, *email*, *role*, dan *created\_at*.

*Field username* digunakan sebagai identitas nama pengguna pada aplikasi, *email* digunakan untuk proses *login* autentikasi, *role* digunakan untuk menentukan hak akses pengguna sebagai karyawan lapangan, sedangkan *created\_at* digunakan untuk menyimpan waktu pembuatan akun pengguna. Data pada *collection* karyawan digunakan untuk proses *login* aplikasi serta mendukung aktivitas *input* data produksi bibit dan produksi tanaman yang dilakukan oleh karyawan lapangan.

### 5.2.2. Implementasi Tabel Kepala Bagian



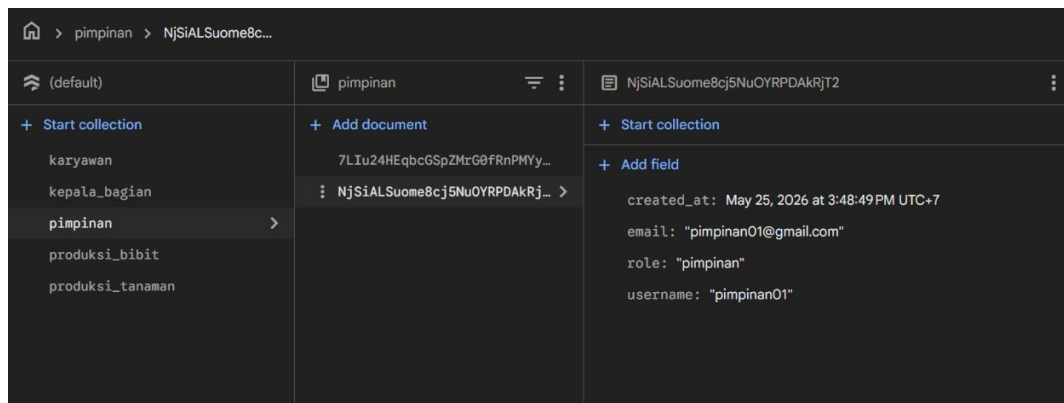
Sumber : data diolah oleh penulis, 2026

Gambar 5.2 Tabel Kepala Bagian

*Collection* kepalabagian digunakan untuk menyimpan data akun pengguna dengan *role* sebagai kepala bagian pada aplikasi *monitoring* produksi bibit dan tanaman karet berbasis *android*. Setiap data pengguna disimpan dalam bentuk dokumen dengan uid yang diperoleh dari *firebase authentication* sebagai identitas unik pengguna. Pada *collection* ini terdapat beberapa *field* seperti *username*, *email*, *role*, dan *created\_at*.

*Field username* digunakan sebagai identitas nama pengguna pada aplikasi, *email* digunakan untuk proses *login* autentikasi, *role* digunakan untuk menentukan hak akses pengguna sebagai kepala bagian, sedangkan *created\_at* digunakan untuk menyimpan waktu pembuatan akun pengguna. Data pada *collection* kepalabagian digunakan untuk proses *login* aplikasi serta mendukung aktivitas validasi data produksi, *monitoring* data produksi, dan manajemen pengguna yang dilakukan oleh kepala bagian.

### 5.2.3. Implementasi Tabel Pimpinan



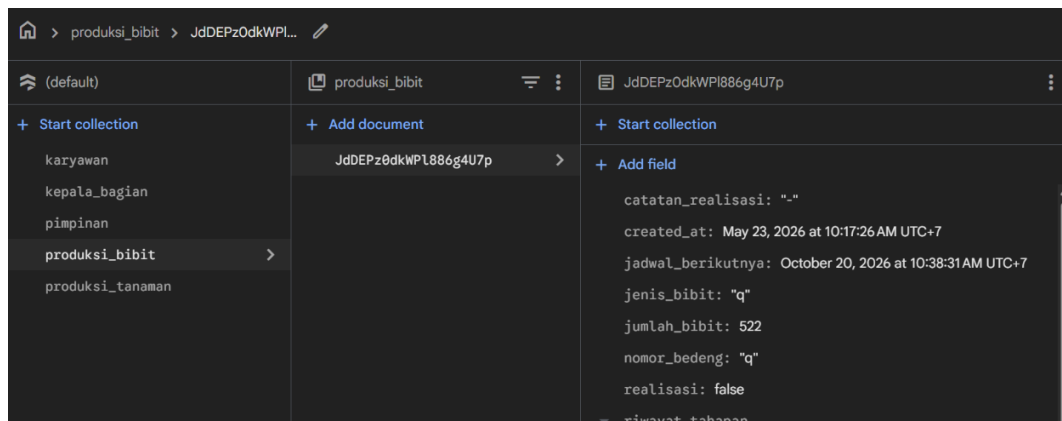
Sumber : data diolah oleh penulis, 2026

Gambar 5.3 Tabel Pimpinan

*Collection* pimpinan digunakan untuk menyimpan data akun pengguna dengan *role* sebagai pimpinan pada aplikasi *monitoring* produksi bibit dan tanaman karet berbasis *android*. Setiap data pengguna disimpan dalam bentuk dokumen dengan uid yang diperoleh dari *firebase authentication* sebagai identitas unik pengguna. Pada *collection* ini terdapat beberapa *field* seperti *username*, *email*, *role*, dan *created\_at*.

*Field username* digunakan sebagai identitas nama pengguna pada aplikasi, *email* digunakan untuk proses *login* autentikasi, *role* digunakan untuk menentukan hak akses pengguna sebagai pimpinan, sedangkan *created\_at* digunakan untuk menyimpan waktu pembuatan akun pengguna. Data pada *collection* pimpinan digunakan untuk proses *login* aplikasi serta mendukung aktivitas *monitoring* laporan produksi, melihat hasil validasi data produksi, dan memantau perkembangan produksi bibit maupun tanaman karet secara keseluruhan.

#### 5.2.4. Implementasi Tabel Produksi Bibit



Sumber : data diolah oleh penulis, 2026

Gambar 5.4 Tabel Produksi Bibit

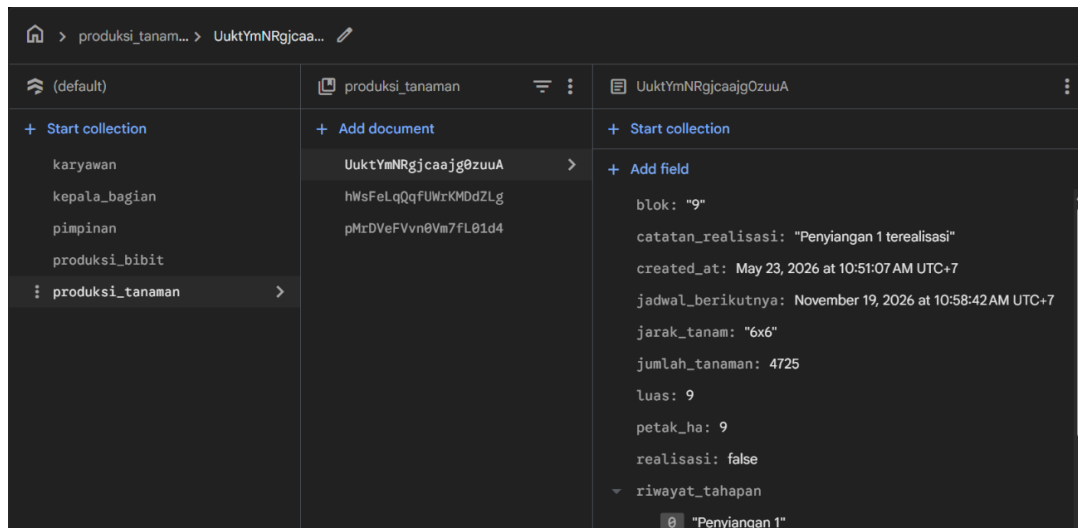
Tabel `produksi_bibit` digunakan untuk menyimpan seluruh data *monitoring* produksi bibit tanaman karet yang dilakukan oleh karyawan lapangan. *Collection* ini memiliki fungsi utama sebagai media penyimpanan data tahapan produksi bibit mulai dari proses awal hingga proses selesai produksi. Pada *collection* ini terdapat beberapa data penting yang disimpan, seperti nomor bedeng, jenis bibit, jumlah bibit, tahapan pemeliharaan, jadwal kegiatan berikutnya, status produksi, serta riwayat tahapan produksi bibit. Data tersebut digunakan untuk mempermudah proses *monitoring* perkembangan produksi bibit secara realtime melalui aplikasi *android*.

Selain itu, *collection* `produksi_bibit` juga digunakan untuk mencatat proses realisasi setiap tahapan produksi yang dilakukan oleh karyawan lapangan. Setiap tahapan yang telah selesai dikerjakan akan tercatat secara otomatis pada riwayat tahapan produksi. Apabila suatu tahapan belum dapat dilaksanakan, maka sistem akan memberikan status tertunda sehingga proses *monitoring* dapat dilakukan dengan lebih terstruktur. Data pada *collection* `produksi_bibit` juga terhubung dengan fitur validasi produksi yang dilakukan oleh kepala bagian. Setelah seluruh tahapan produksi selesai dilaksanakan, data produksi akan masuk ke proses validasi untuk memastikan bahwa seluruh kegiatan produksi telah dilakukan sesuai prosedur yang berlaku.

Implementasi *collection* `produksi_bibit` pada *firebase firestore* memungkinkan data produksi dapat tersimpan secara realtime sehingga informasi *monitoring* dapat langsung diperbarui dan ditampilkan pada aplikasi tanpa perlu

melakukan *refresh* secara manual. Dengan adanya sistem ini, proses pengelolaan dan *monitoring* produksi bibit tanaman karet menjadi lebih efektif, terstruktur, dan mudah dipantau oleh seluruh pengguna aplikasi sesuai dengan hak akses masing-masing.

### 5.2.5 Implementasi Tabel Produksi Tanaman



Sumber : data diolah oleh penulis, 2026

Gambar 5.5 Tabel Produksi Tanaman

Tabel `produksi_tanaman` digunakan untuk menyimpan seluruh data *monitoring* produksi tanaman karet yang telah memasuki tahap penanaman di lapangan. *Collection* ini berfungsi sebagai media penyimpanan data kegiatan pemeliharaan tanaman karet agar proses *monitoring* produksi dapat dilakukan secara lebih terstruktur dan realtime melalui aplikasi *android*. Data yang tersimpan pada *collection* ini meliputi blok tanaman, sub blok, jumlah tanaman, tahapan pemeliharaan, jadwal kegiatan berikutnya, status produksi, dan riwayat tahapan produksi tanaman. Informasi tersebut digunakan untuk memantau perkembangan setiap tanaman berdasarkan lokasi dan tahapan pekerjaan yang sedang berlangsung.

Selain digunakan untuk *monitoring*, *collection* `produksi_tanaman` juga digunakan untuk mencatat realisasi kegiatan pemeliharaan tanaman seperti penyiangan, pemupukan, penyemprotan, dan pengendalian hama penyakit. Setiap tahapan yang telah selesai dikerjakan akan tercatat pada riwayat tahapan produksi sehingga memudahkan proses pengecekan kegiatan yang telah dilakukan

sebelumnya. *Collection* ini juga mendukung fitur validasi produksi yang dilakukan oleh kepala bagian. Setelah seluruh tahapan pemeliharaan selesai dilaksanakan, data produksi tanaman akan masuk ke tahap validasi untuk memastikan bahwa kegiatan produksi telah dilakukan sesuai prosedur perusahaan.

Dengan menggunakan *firebase firestore*, seluruh data pada *collection* produksi\_tanaman dapat tersimpan dan diperbarui secara *realtime*. Hal tersebut memudahkan pengguna aplikasi dalam melakukan monitoring produksi tanaman secara cepat, efektif, dan efisien tanpa perlu melakukan pencatatan manual seperti sebelumnya.



## 5.3 Tampilan Antar Muka

### 5.3.1. *Splash Screen*

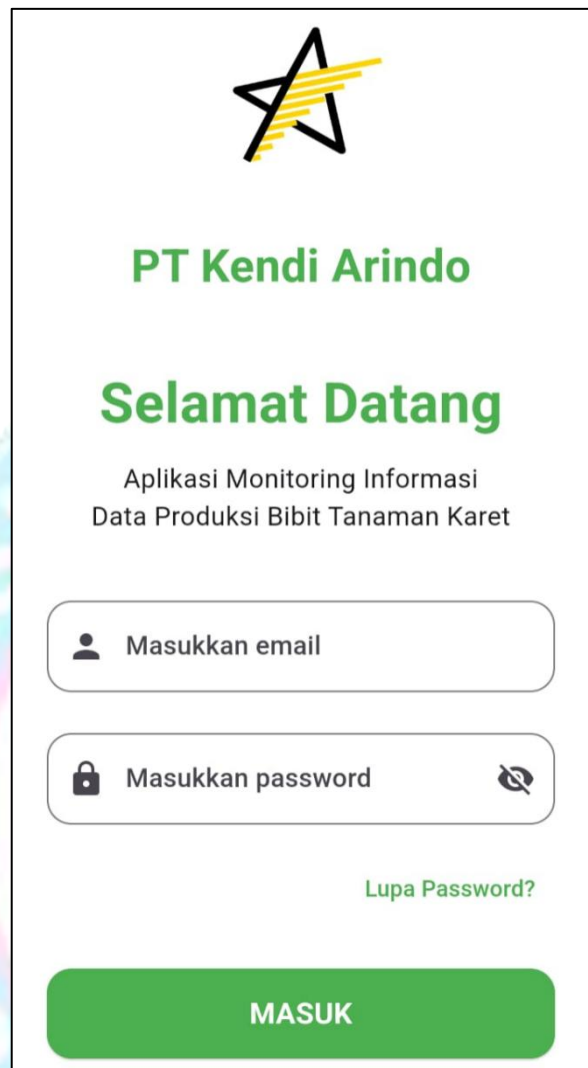


*Sumber : data diolah oleh penulis, 2026*

*Gambar 5.6 Splash Screen*

Halaman *splash screen* merupakan halaman awal yang tampil ketika aplikasi pertama kali dijalankan. Halaman ini berfungsi sebagai tampilan pembuka aplikasi sekaligus proses loading sebelum pengguna masuk ke halaman *login*. Pada halaman *splash screen* ditampilkan logo dan identitas aplikasi digitalisasi *monitoring informasi data produksi bibit tanaman karet berbasis android*. Selain itu, halaman ini juga digunakan sistem untuk melakukan proses pengecekan awal aplikasi sebelum pengguna mengakses fitur-fitur yang tersedia di dalam sistem.

### 5.3.2. Halaman *Login*





**PT Kendi Arindo**

**Selamat Datang**

Aplikasi Monitoring Informasi  
Data Produksi Bibit Tanaman Karet

 Masukkan email

 Masukkan password 

[Lupa Password?](#)

**MASUK**

Sumber : data diolah oleh penulis, 2026

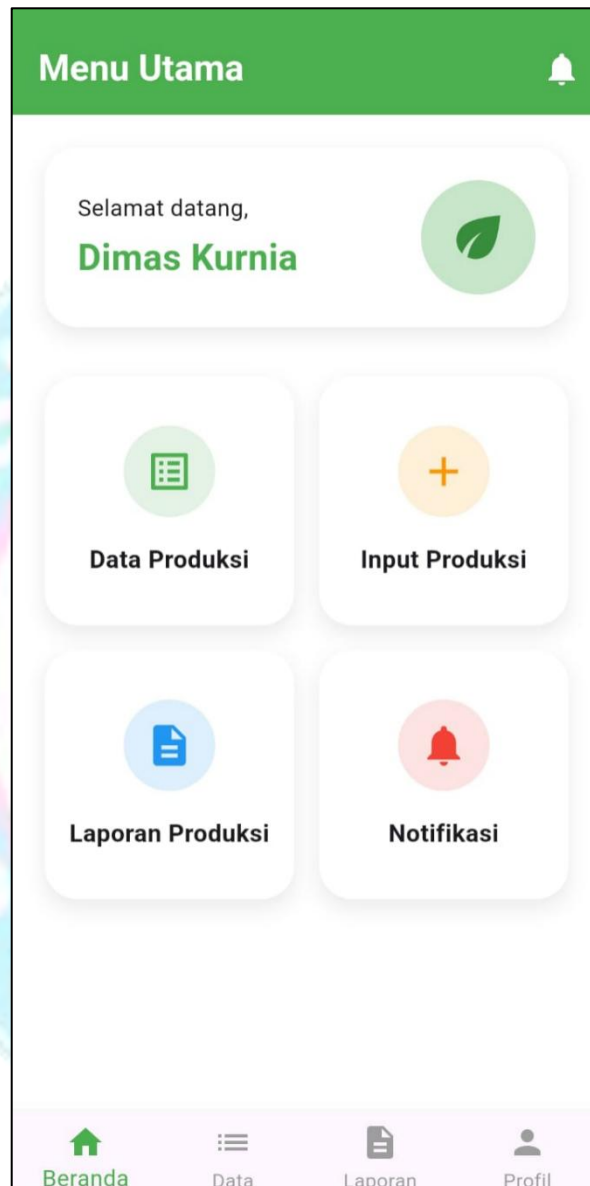
Gambar 5.7 Halaman *Login*

Halaman *login* merupakan halaman yang digunakan pengguna untuk masuk ke dalam sistem aplikasi menggunakan akun yang telah terdaftar. Pada halaman ini pengguna diminta untuk memasukkan *email* dan *password* sesuai dengan akun masing-masing. Sistem *login* pada aplikasi menggunakan *firebase authentication* sehingga proses autentikasi pengguna dapat dilakukan dengan aman dan *realtime*.

Selain itu, halaman *login* juga dilengkapi dengan fitur lupa *password* yang memungkinkan pengguna melakukan *reset password* melalui *email* yang terhubung dengan akun pengguna. Setelah proses *login* berhasil dilakukan, sistem akan secara otomatis mengarahkan pengguna menuju halaman *dashboard* sesuai dengan *role*

masing-masing, seperti *dashboard* karyawan, *dashboard* kepala bagian, atau *dashboard* pimpinan.

### 5.3.3. Halaman *Dashboard* Karyawan Lapangan



Sumber : data diolah oleh penulis, 2026

Gambar 5.8 Halaman *Dashboard* Karyawan Lapangan

Halaman *dashboard* karyawan merupakan halaman utama yang digunakan oleh karyawan lapangan setelah berhasil *login* ke dalam aplikasi. Pada halaman ini ditampilkan beberapa menu utama yang digunakan untuk mengelola data produksi bibit dan tanaman karet. Menu yang tersedia pada *dashboard* karyawan antara lain

menu data produksi bibit, data produksi tanaman, laporan produksi, notifikasi, dan profil pengguna. Selain itu, *dashboard* juga menampilkan informasi singkat mengenai jumlah data produksi yang sedang berjalan sehingga memudahkan pengguna dalam melakukan *monitoring* kegiatan produksi.

#### 5.3.4. Halaman Data Produksi Bibit



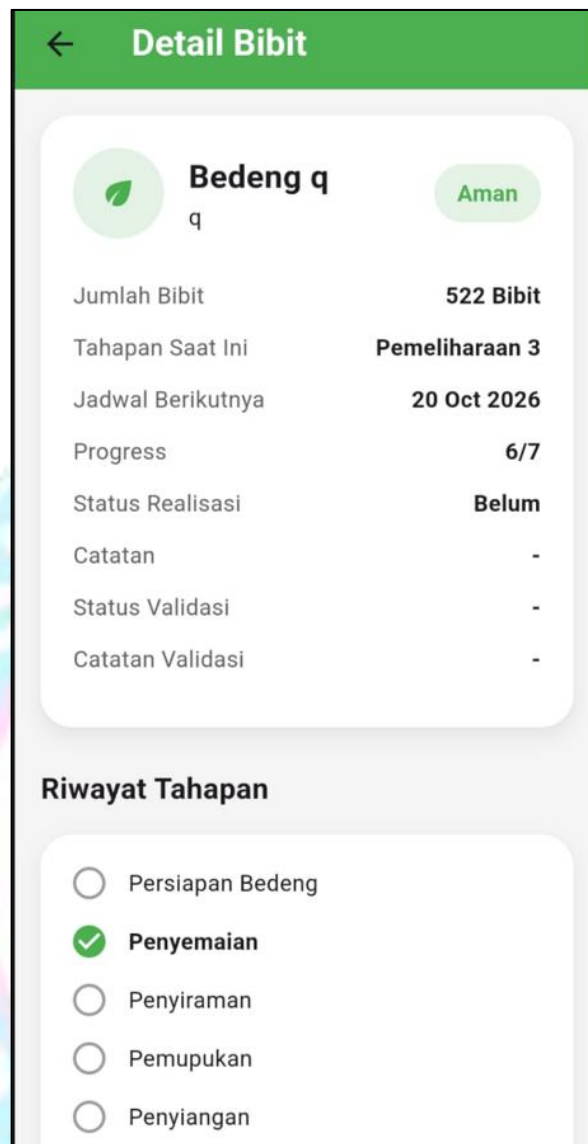
| Bedeng | Jenis | Tahapan        | Progress | Status | Aksi                                                                                                                                                                                                                                                        |
|--------|-------|----------------|----------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| q      | q     | Pemeliharaan 3 | 6/7      | Aman   |    |

Sumber : data diolah oleh penulis, 2026

Gambar 5.9 Halaman Data Produksi Bibit

Halaman data produksi bibit digunakan untuk menampilkan seluruh data *monitoring* produksi bibit tanaman karet yang telah diinput oleh pengguna. Data ditampilkan dalam bentuk tabel agar lebih rapi dan memudahkan proses pencarian data ketika jumlah data produksi semakin banyak. Pada halaman ini pengguna dapat melihat informasi seperti nomor bedeng, jenis bibit, tahapan produksi, progress produksi, status produksi, serta aksi detail data. Selain itu, pengguna juga dapat melakukan proses realisasi tahapan produksi dan penghapusan data produksi apabila diperlukan.

### 5.3.5. Halaman Detail Produksi Bibit



← **Detail Bibit**


**Bedeng q**
Aman

|                   |                |
|-------------------|----------------|
| Jumlah Bibit      | 522 Bibit      |
| Tahapan Saat Ini  | Pemeliharaan 3 |
| Jadwal Berikutnya | 20 Oct 2026    |
| Progress          | 6/7            |
| Status Realisasi  | Belum          |
| Catatan           | -              |
| Status Validasi   | -              |
| Catatan Validasi  | -              |

**Riwayat Tahapan**

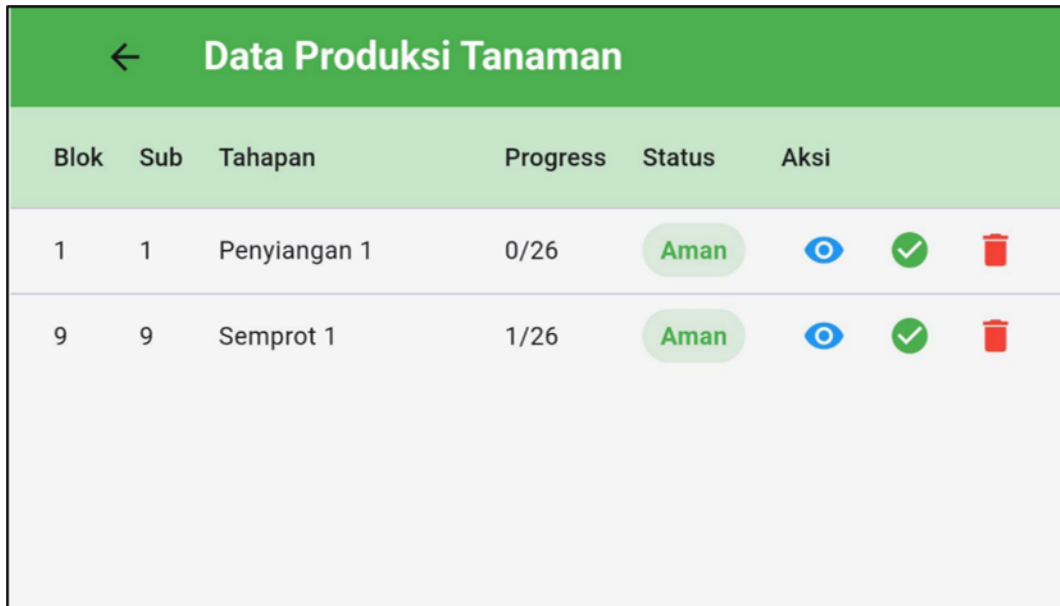
- ☐ Persiapan Bedeng
- ☒ **Penyemaian**
- ☐ Penyiraman
- ☐ Pemupukan
- ☐ Penyiangan

Sumber : data diolah oleh penulis, 2026

Gambar 5.10 Halaman Detail Produksi Bibit

Halaman detail produksi bibit digunakan untuk menampilkan informasi detail dari data produksi bibit yang dipilih pengguna. Pada halaman ini ditampilkan informasi lengkap mengenai tahapan produksi, jumlah bibit, jadwal kegiatan berikutnya, status produksi, dan riwayat tahapan produksi bibit. Halaman detail ini membantu pengguna dalam melakukan *monitoring* perkembangan produksi bibit secara lebih rinci dan terstruktur.

### 5.3.6. Halaman *Input* Data Produksi Tanaman



| Blok | Sub | Tahapan      | Progress | Status | Aksi                                                                                                                                                                                                                                                        |
|------|-----|--------------|----------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1    | 1   | Penyiangan 1 | 0/26     | Aman   |    |
| 9    | 9   | Semprot 1    | 1/26     | Aman   |    |

*Sumber : data diolah oleh penulis, 2026*

Gambar 5.11 Halaman Data Produksi Tanaman

Halaman data produksi tanaman digunakan untuk menampilkan seluruh data *monitoring* produksi tanaman karet yang telah ditanam di lapangan. Data ditampilkan dalam bentuk tabel agar memudahkan proses *monitoring* dan pengelolaan data produksi tanaman. Informasi yang ditampilkan meliputi blok tanaman, sub blok, tahapan pemeliharaan, progress produksi, status produksi, dan aksi detail data tanaman. Selain itu, pengguna juga dapat melakukan realisasi tahapan produksi tanaman secara langsung melalui aplikasi.

Gambar 5.12 Halaman Detail Produksi Tanaman

Universitas Prabumulih

### 5.3.8. Halaman *Input* Data Produksi Bibit



Sumber : data diolah oleh penulis, 2026

Gambar 5.13 Halaman *Input* Data Produksi Bibit

Halaman *input* data produksi bibit digunakan oleh karyawan lapangan untuk menambahkan data produksi bibit tanaman karet ke dalam sistem aplikasi. Pada halaman ini pengguna dapat memasukkan berbagai informasi terkait produksi bibit seperti nomor bedeng, jenis bibit, jumlah bibit, tahapan produksi, jadwal kegiatan berikutnya, serta keterangan tambahan lainnya. Data yang telah diinput akan langsung tersimpan ke dalam *firebase firestore* secara *realtime* sehingga dapat langsung dimonitor oleh kepala bagian maupun pimpinan perusahaan melalui aplikasi *android*.

Halaman ini dirancang sederhana dan mudah digunakan agar proses *input* data dapat dilakukan dengan cepat dan efisien di lapangan. Selain itu, halaman *input* data produksi bibit juga membantu mengurangi kesalahan pencatatan data yang sebelumnya masih dilakukan secara manual menggunakan buku atau kertas.

### 5.3.9. Halaman *Input* Data Produksi Tanaman

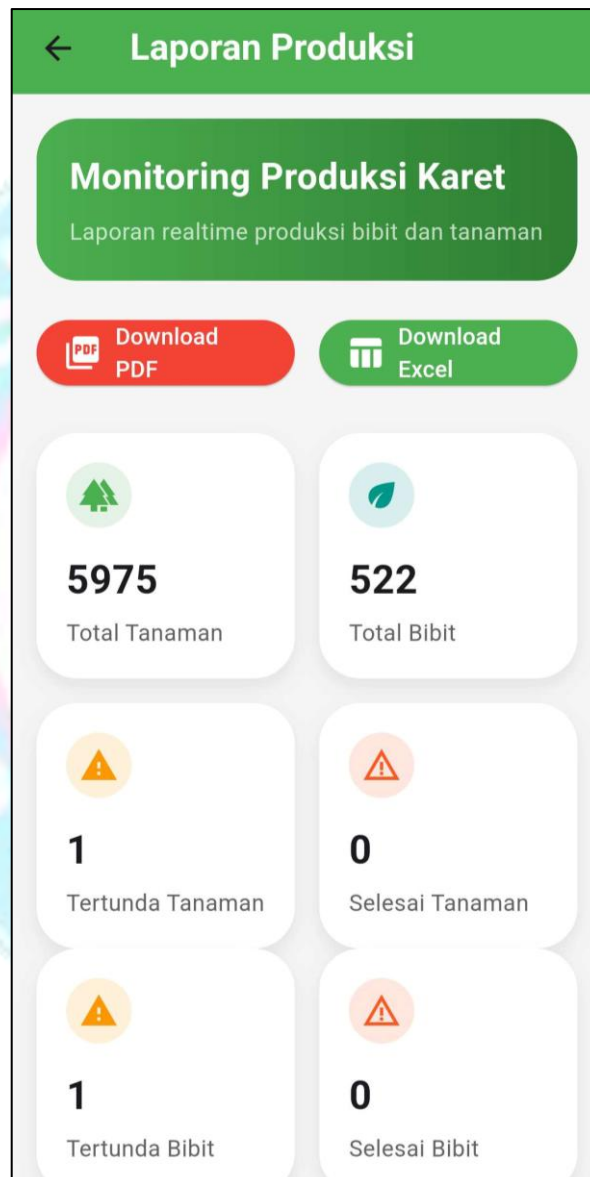
Sumber : data diolah oleh penulis, 2026

Gambar 5.14 Halaman *Input* Data Produksi Tanaman

Halaman *input* data produksi tanaman digunakan untuk menambahkan data produksi tanaman karet yang telah ditanam di lapangan. Pada halaman ini pengguna dapat memasukkan data seperti blok tanaman, sub blok, jumlah tanaman, tahapan

pemeliharaan, jadwal kegiatan berikutnya, dan informasi pendukung lainnya. Data yang diinput akan langsung tersimpan ke *database firebase firestore* sehingga proses *monitoring* produksi dapat dilakukan secara realtime oleh seluruh pengguna sesuai hak akses masing-masing.

### 5.3.10. Halaman Laporan



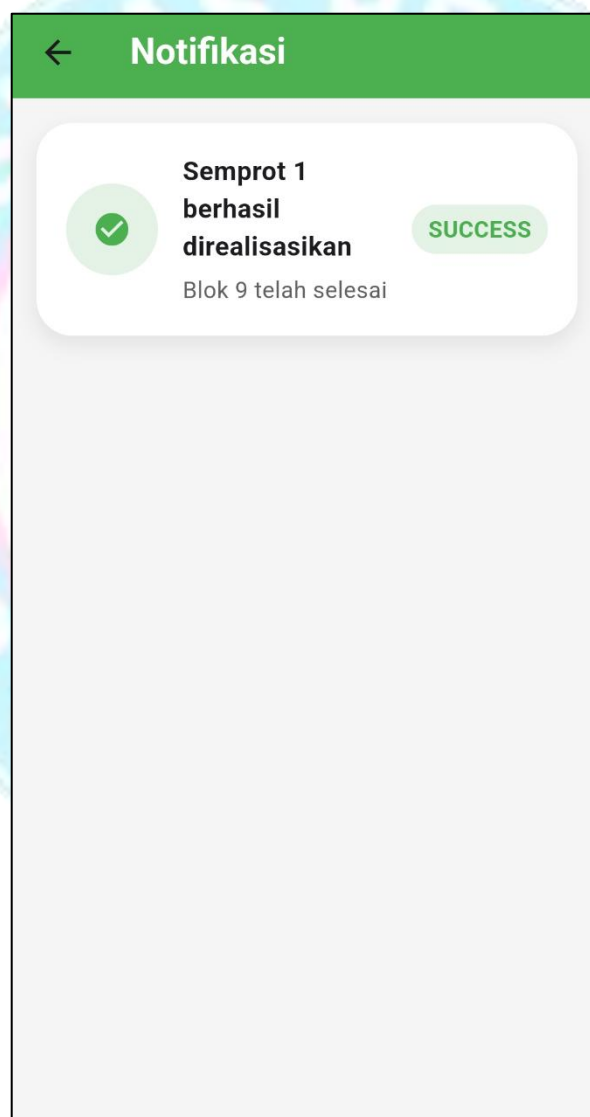
Sumber : data diolah oleh penulis, 2026

Gambar 5.15 Halaman Laporan

Halaman laporan digunakan untuk menampilkan hasil rekapitulasi data produksi bibit dan tanaman karet yang telah tersimpan pada sistem. Pada halaman ini data ditampilkan dalam bentuk tabel dan grafik untuk mempermudah pengguna

dalam membaca informasi hasil produksi. Laporan produksi berisi informasi jumlah produksi bibit, jumlah produksi tanaman, data produksi yang telah divalidasi, serta statistik hasil *monitoring* produksi secara keseluruhan. Selain ditampilkan pada aplikasi, laporan juga dapat dicetak atau diunduh dalam format *pdf* maupun *excel* sebagai dokumentasi perusahaan. Halaman laporan membantu pimpinan dan kepala bagian dalam melakukan evaluasi hasil produksi secara lebih cepat, terstruktur, dan efisien.

#### 5.3.11. Halaman Notifikasi

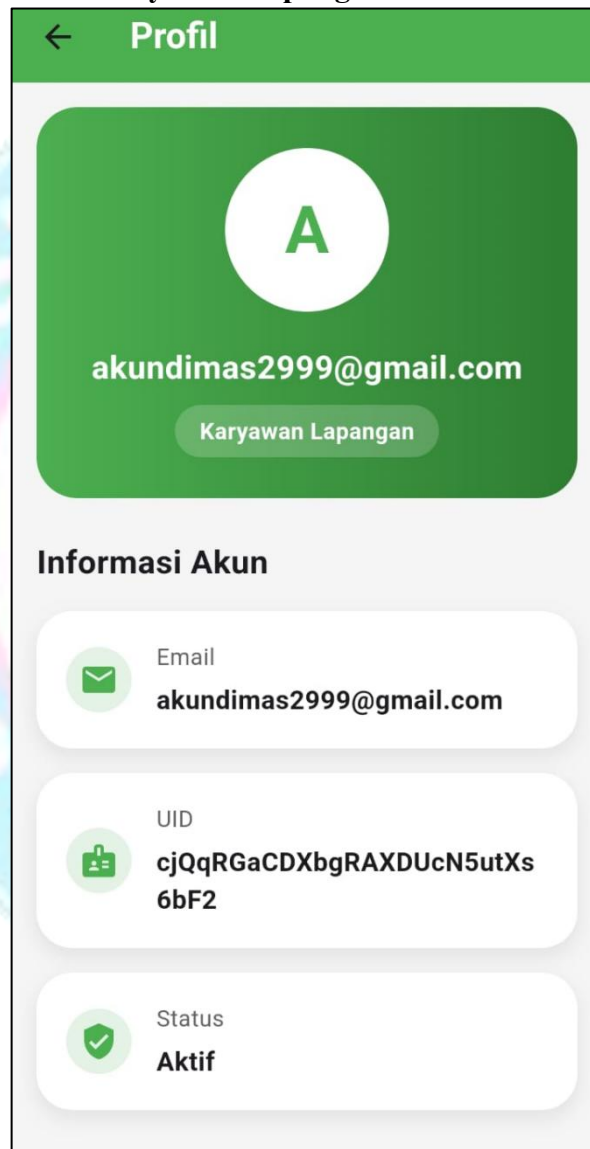


Sumber : data diolah oleh penulis, 2026

Gambar 5.16 Halaman Notifikasi

Halaman notifikasi digunakan untuk menampilkan informasi pengingat jadwal kegiatan produksi yang akan datang. Sistem akan memberikan notifikasi berdasarkan jadwal tahapan produksi bibit maupun tanaman yang mendekati waktu pelaksanaan. Halaman ini membantu pengguna agar tidak terlambat dalam melaksanakan kegiatan produksi di lapangan.

### 5.3.12. Halaman Profil Karyawan Lapangan



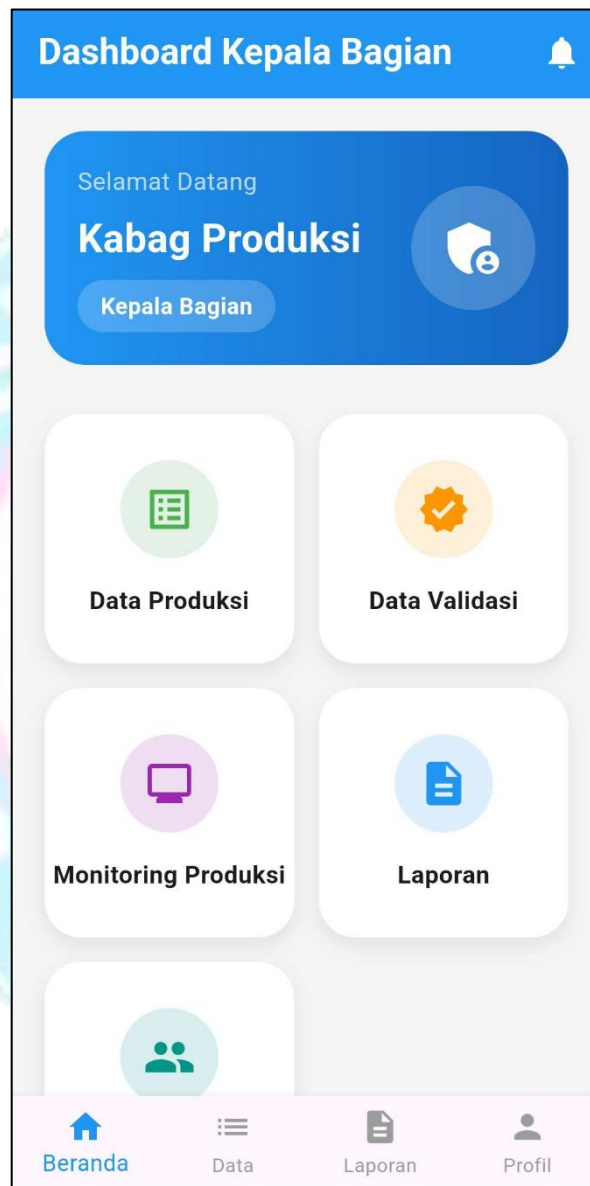
Sumber : data diolah oleh penulis, 2026

Gambar 5.17 Halaman Profil Karyawan Lapangan

Halaman profil digunakan untuk menampilkan informasi akun pengguna yang sedang *login* ke dalam aplikasi. Pada halaman ini pengguna dapat melihat

informasi *username*, *email*, dan *role* pengguna. Selain itu, halaman profil juga menyediakan fitur *logout* yang digunakan untuk keluar dari aplikasi dan kembali menuju halaman *login*.

### 5.3.13. Halaman *Dashboard* Kepala Bagian



Sumber : data diolah oleh penulis, 2026

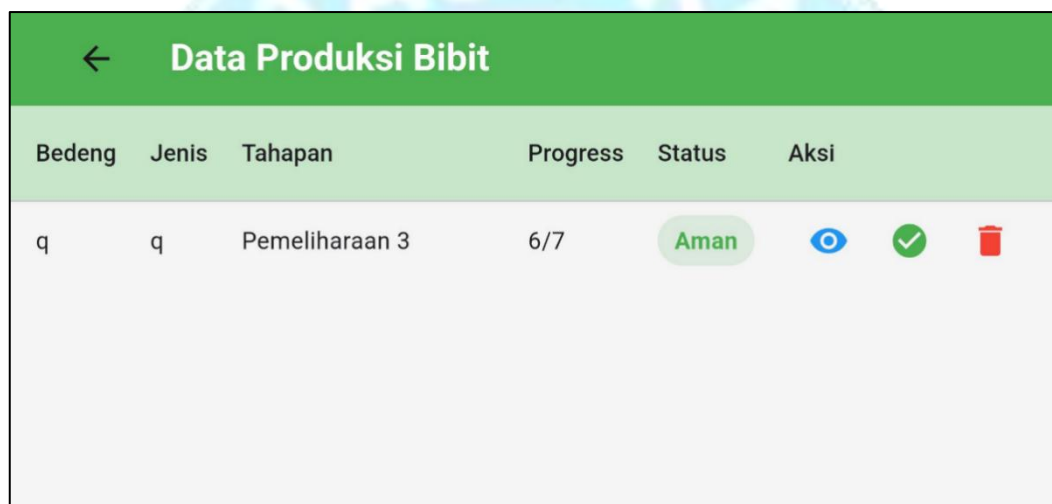
Gambar 5.18 Halaman *Dashboard* Kepala Bagian

Halaman *dashboard* kepala bagian merupakan halaman utama yang digunakan oleh kepala bagian untuk melakukan *monitoring* dan pengelolaan data produksi. Pada halaman ini kepala bagian memiliki hak akses lebih luas

dibandingkan karyawan lapangan. *Dashboard* kepala bagian menyediakan beberapa menu seperti *monitoring* produksi, validasi produksi, laporan, notifikasi, profil pengguna, dan manajemen pengguna.

Pada menu manajemen pengguna, kepala bagian dapat menambahkan akun pengguna baru serta melihat daftar seluruh pengguna aplikasi berdasarkan *role* masing-masing. Selain itu, kepala bagian juga dapat melakukan validasi terhadap data produksi yang telah diselesaikan oleh karyawan lapangan sebelum data tersebut diteruskan menjadi laporan *monitoring* produksi.

#### 5.3.14. Halaman Data Produksi Bibit



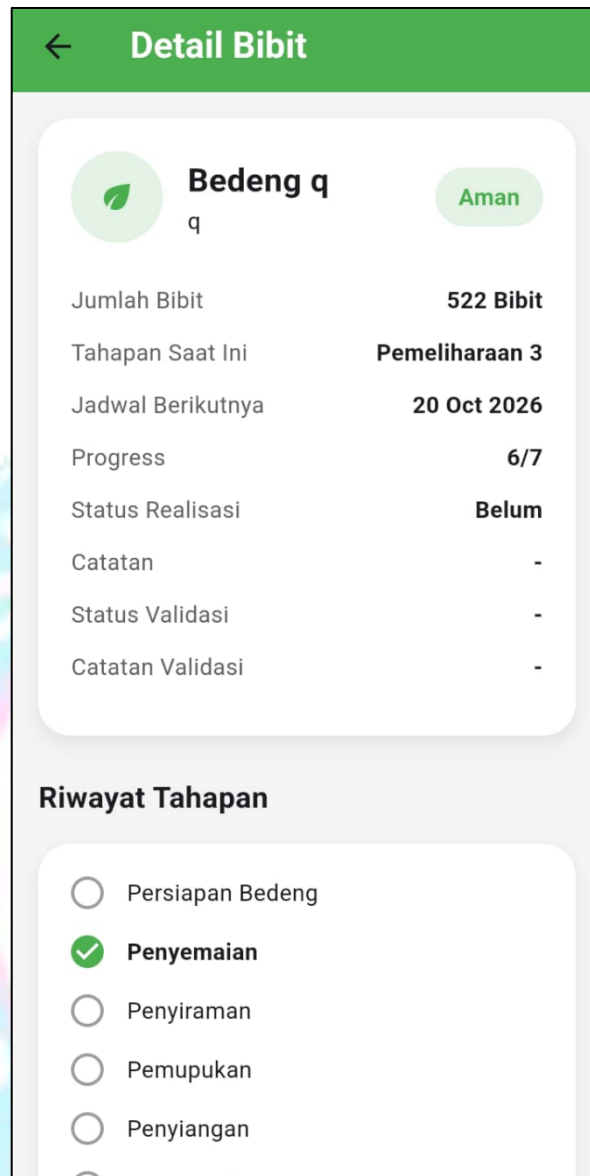
| Bedeng | Jenis | Tahapan        | Progress | Status | Aksi                                                                                                                                                                                                                                                           |
|--------|-------|----------------|----------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| q      | q     | Pemeliharaan 3 | 6/7      | Aman   |    |

Sumber : data diolah oleh penulis, 2026

Gambar 5.19 Halaman Data Produksi Bibit

Halaman data produksi bibit digunakan untuk menampilkan seluruh data *monitoring* produksi bibit tanaman karet yang telah diinput oleh pengguna. Data ditampilkan dalam bentuk tabel agar lebih rapi dan memudahkan proses pencarian data ketika jumlah data produksi semakin banyak. Pada halaman ini pengguna dapat melihat informasi seperti nomor bedeng, jenis bibit, tahapan produksi, progress produksi, status produksi, serta aksi detail data. Selain itu, pengguna juga dapat melakukan proses realisasi tahapan produksi dan penghapusan data produksi apabila diperlukan.

### 5.3.15. Halaman Detail Produksi Bibit



← **Detail Bibit**


**Bedeng q**
Aman

Jumlah Bibit **522 Bibit**

Tahapan Saat Ini **Pemeliharaan 3**

Jadwal Berikutnya **20 Oct 2026**

Progress **6/7**

Status Realisasi **Belum**

Catatan -

Status Validasi -

Catatan Validasi -

**Riwayat Tahapan**

☐ Persiapan Bedeng

☒ **Penyemaian**

☐ Penyiraman

☐ Pemupukan

☐ Penyiangan

Sumber : data diolah oleh penulis, 2026

Gambar 5.20 Halaman Detail Produksi Bibit

Halaman detail produksi bibit digunakan untuk menampilkan informasi detail dari data produksi bibit yang dipilih pengguna. Pada halaman ini ditampilkan informasi lengkap mengenai tahapan produksi, jumlah bibit, jadwal kegiatan berikutnya, status produksi, dan riwayat tahapan produksi bibit. Halaman detail ini membantu pengguna dalam melakukan *monitoring* perkembangan produksi bibit secara lebih rinci dan terstruktur.

### 5.3.16. Halaman Data Produksi Tanaman

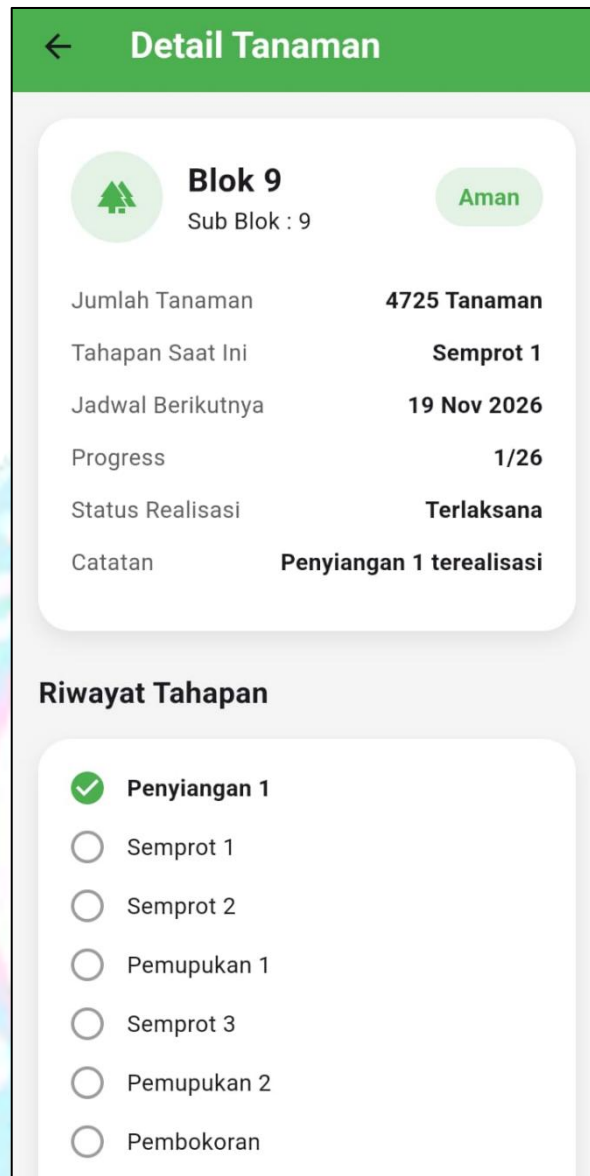
| <div> <span>←</span> Data Produksi Tanaman </div> |     |              |          |        |                                                                                     |                                                                                     |
|---------------------------------------------------|-----|--------------|----------|--------|-------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|
| Blok                                              | Sub | Tahapan      | Progress | Status | Aksi                                                                                |                                                                                     |
| 1                                                 | 1   | Penyiangan 1 | 0/26     | Aman   |  |  |
| 9                                                 | 9   | Semprot 1    | 1/26     | Aman   |  |  |

Sumber : data diolah oleh penulis, 2026

Gambar 5.21 Halaman Data Produksi Tanaman

Halaman data produksi tanaman digunakan untuk menampilkan seluruh data *monitoring* produksi tanaman karet yang telah ditanam di lapangan. Data ditampilkan dalam bentuk tabel agar memudahkan proses *monitoring* dan pengelolaan data produksi tanaman. Informasi yang ditampilkan meliputi blok tanaman, sub blok, tahapan pemeliharaan, *progress* produksi, status produksi, dan aksi detail data tanaman. Selain itu, pengguna juga dapat melakukan realisasi tahapan produksi tanaman secara langsung melalui aplikasi.

### 5.3.17. Halaman Detail Produksi Tanaman





Blok 9

Sub Blok : 9

Aman

|                   |                          |
|-------------------|--------------------------|
| Jumlah Tanaman    | 4725 Tanaman             |
| Tahapan Saat Ini  | Semprot 1                |
| Jadwal Berikutnya | 19 Nov 2026              |
| Progress          | 1/26                     |
| Status Realisasi  | Terlaksana               |
| Catatan           | Penyiangan 1 terealisasi |

Riwayat Tahapan

☒

Penyiangan 1

☐

Semprot 1

☐

Semprot 2

☐

Pemupukan 1

☐

Semprot 3

☐

Pemupukan 2

☐

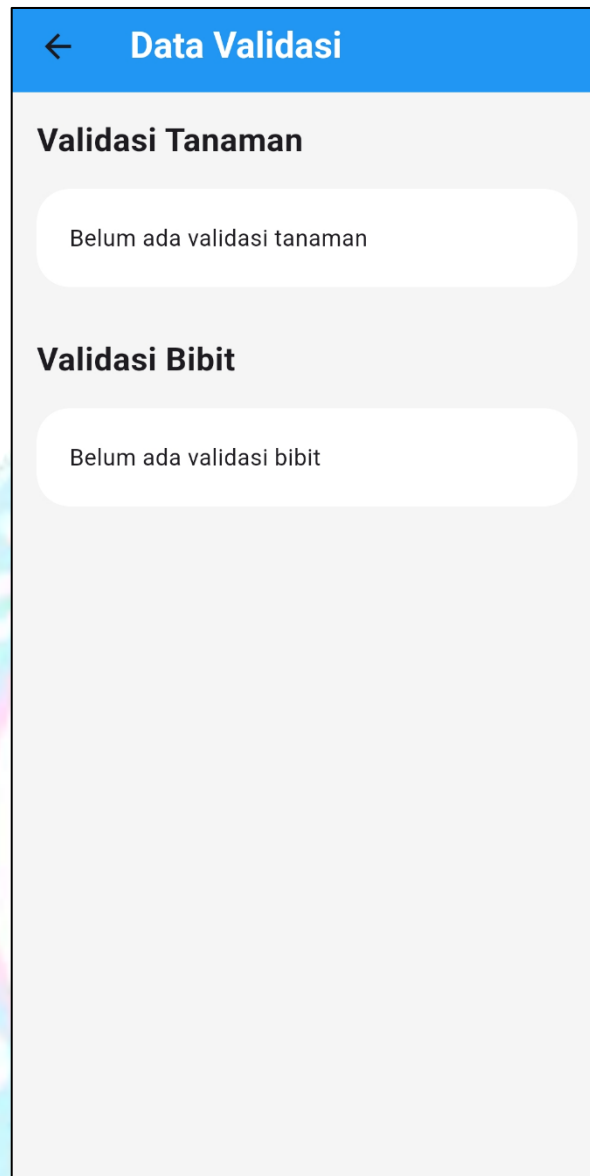
Pembokoran

Sumber : data diolah oleh penulis, 2026

Gambar 5.22 Halaman Detail Produksi Tanaman

Halaman detail produksi tanaman digunakan untuk menampilkan informasi detail mengenai data produksi tanaman yang dipilih pengguna. Pada halaman ini ditampilkan riwayat tahapan produksi, jadwal kegiatan berikutnya, jumlah tanaman, serta status realisasi kegiatan produksi tanaman. Halaman ini membantu pengguna dalam melakukan pengecekan tahapan produksi tanaman secara lebih detail dan *realtime*.

### 5.3.18. Halaman Data Validasi

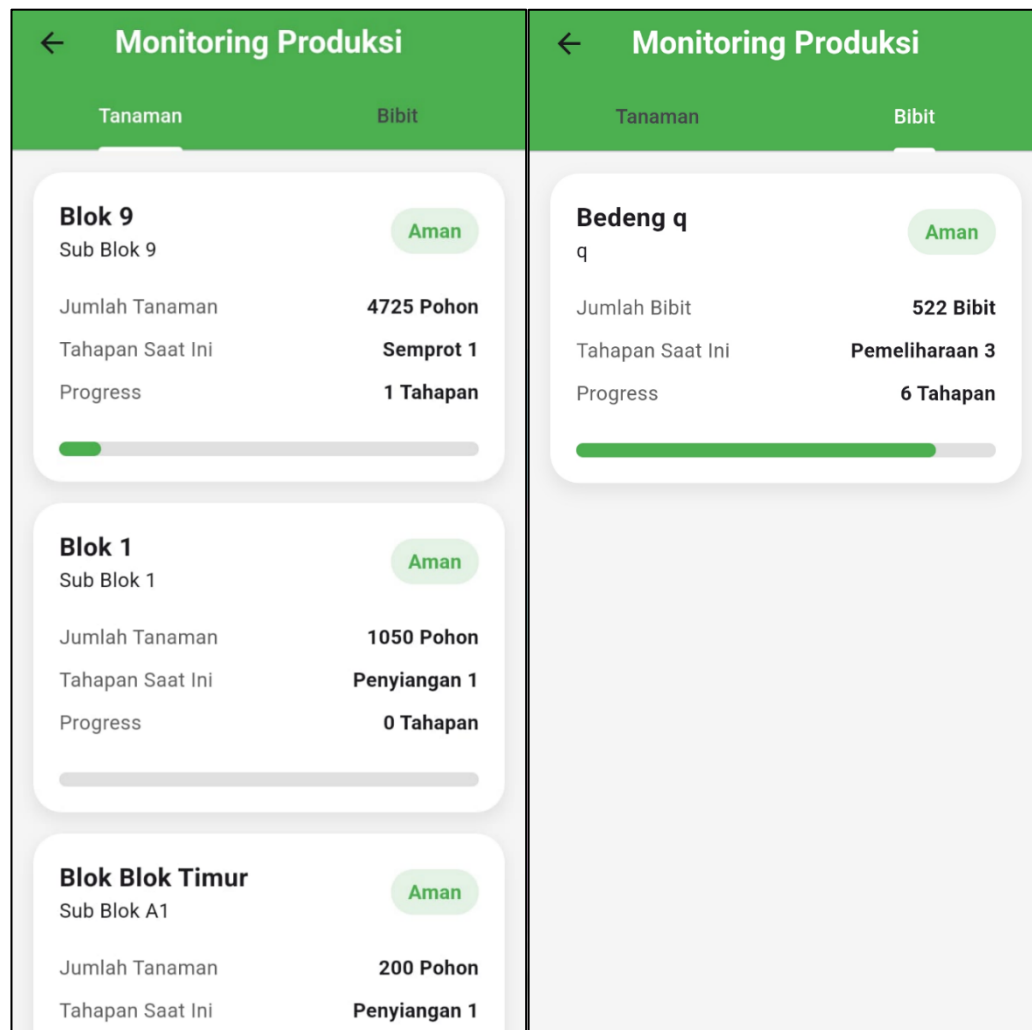


*Sumber : data diolah oleh penulis, 2026*

Gambar 5.23 Halaman Validasi Produksi

Halaman validasi produksi digunakan oleh kepala bagian untuk melakukan pengecekan dan validasi terhadap data produksi yang telah diselesaikan oleh karyawan lapangan. Pada halaman ini kepala bagian dapat melihat status realisasi kegiatan produksi sebelum memberikan validasi akhir terhadap data tersebut. Proses validasi dilakukan untuk memastikan bahwa seluruh tahapan produksi telah dilaksanakan sesuai dengan prosedur perusahaan.

### 5.3.19. Halaman *Monitoring Produksi*



Sumber : data diolah oleh penulis, 2026

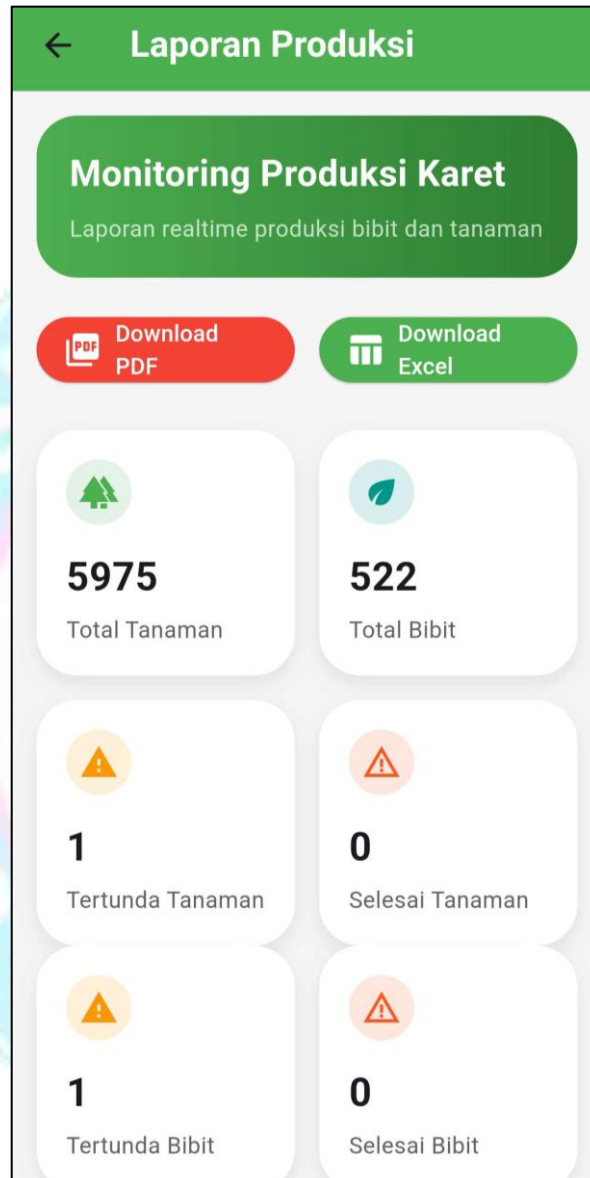
Gambar 5.24 Halaman *Monitoring Produksi*

Halaman *monitoring* produksi digunakan untuk memantau seluruh proses produksi bibit dan tanaman karet yang sedang berjalan secara *realtime*. Pada halaman ini pengguna dapat melihat perkembangan tahapan produksi, status realisasi kegiatan, jadwal kegiatan berikutnya, serta *progress* pekerjaan yang telah dilakukan oleh karyawan lapangan.

Halaman *monitoring* produksi membantu kepala bagian dan pimpinan dalam melakukan pengawasan terhadap aktivitas produksi tanpa harus melakukan pengecekan langsung ke lapangan. Informasi yang ditampilkan akan diperbarui secara *realtime* berdasarkan data yang di *input* oleh pengguna melalui aplikasi *android*. Selain itu, halaman *monitoring* produksi juga mempermudah proses

evaluasi kegiatan produksi karena seluruh riwayat tahapan produksi dapat dilihat secara langsung pada sistem.

### 5.3.20. Halaman Laporan



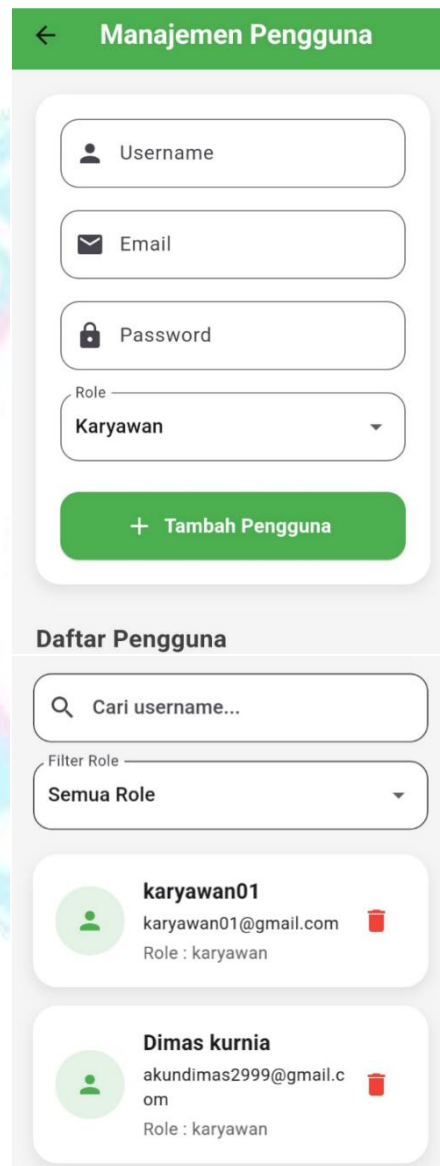
Sumber : data diolah oleh penulis, 2026

Gambar 5.25 Halaman Laporan

Halaman laporan digunakan untuk menampilkan hasil rekapitulasi data produksi bibit dan tanaman karet yang telah tersimpan pada sistem. Pada halaman ini data ditampilkan dalam bentuk tabel dan grafik untuk mempermudah pengguna dalam membaca informasi hasil produksi. Laporan produksi berisi informasi jumlah produksi bibit, jumlah produksi tanaman, data produksi yang telah

divalidasi, serta statistik hasil *monitoring* produksi secara keseluruhan. Selain ditampilkan pada aplikasi, laporan juga dapat dicetak atau diunduh dalam format *pdf* maupun *excel* sebagai dokumentasi perusahaan. Halaman laporan membantu pimpinan dan kepala bagian dalam melakukan evaluasi hasil produksi secara lebih cepat, terstruktur, dan efisien.

### 5.3.21. Halaman Manajemen Pengguna



**Manajemen Pengguna**

Username

Email

Password

Role  
Karyawan

+ Tambah Pengguna

**Daftar Pengguna**

Cari username...

Filter Role  
Semua Role

|                                                                                     |                                                                   |                                                                                     |
|-------------------------------------------------------------------------------------|-------------------------------------------------------------------|-------------------------------------------------------------------------------------|
|  | <b>karyawan01</b><br>karyawan01@gmail.com<br>Role : karyawan      |  |
|  | <b>Dimas kurnia</b><br>akundimas2999@gmail.com<br>Role : karyawan |  |

Sumber : data diolah oleh penulis, 2026

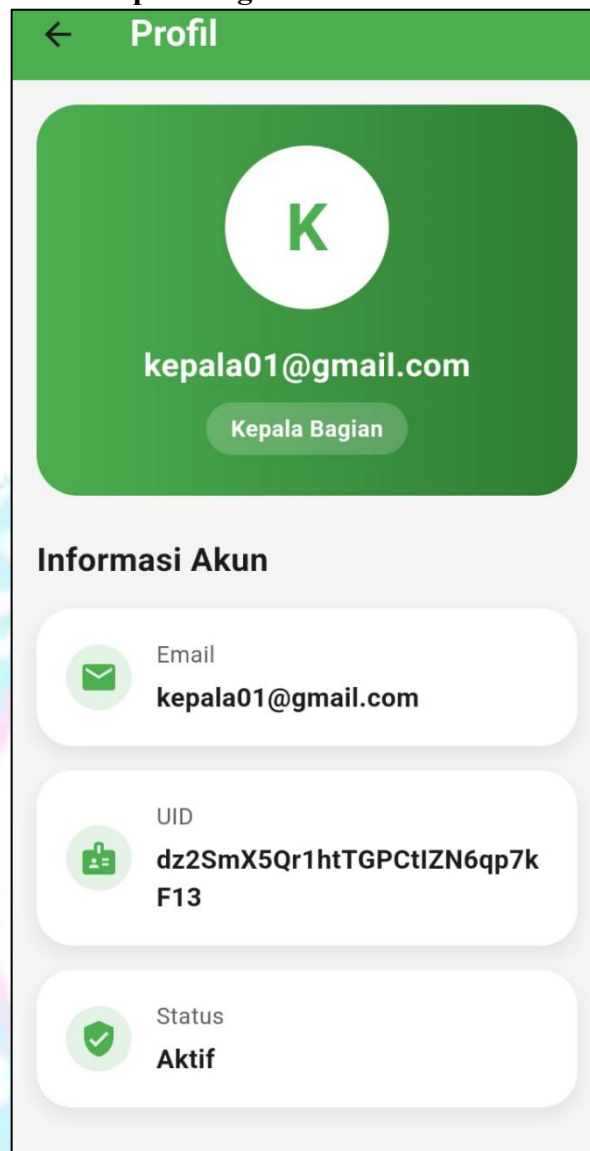
Gambar 5.26 Halaman Manajemen Pengguna

Halaman manajemen pengguna merupakan halaman yang digunakan oleh kepala bagian untuk mengelola seluruh akun pengguna yang dapat mengakses aplikasi digitalisasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android*. Halaman ini dirancang untuk mempermudah proses pengelolaan akun pengguna tanpa perlu melakukan penginputan data secara langsung melalui *firebase firestore*. Pada halaman ini kepala bagian dapat menambahkan akun pengguna baru dengan mengisi data seperti *username*, *email*, *password*, dan *role* pengguna. Sistem akan secara otomatis menyimpan data akun ke *firebase authentication* dan *firebase firestore* sehingga akun dapat langsung digunakan untuk *login* ke dalam aplikasi.

Selain digunakan untuk menambahkan pengguna, halaman manajemen pengguna juga menampilkan daftar seluruh akun yang telah terdaftar pada sistem secara *realtime*. Informasi yang ditampilkan meliputi *username*, *email*, dan *role* pengguna seperti karyawan, kepala bagian, dan pimpinan. Halaman ini juga dilengkapi dengan fitur pencarian pengguna berdasarkan *username* serta fitur filter *role* pengguna untuk mempermudah proses pencarian data apabila jumlah pengguna dalam sistem sudah banyak.

Dengan adanya fitur tersebut, proses pengelolaan akun pengguna menjadi lebih cepat, terstruktur, dan efisien. Implementasi halaman manajemen pengguna membantu kepala bagian dalam mengontrol akses pengguna terhadap aplikasi sesuai dengan hak akses masing-masing pengguna. Selain itu, halaman ini juga mendukung proses digitalisasi pengelolaan akun pengguna agar lebih modern dan *realtime* dibandingkan pengelolaan manual sebelumnya.

### 5.3.22. Halaman Profil Kepala Bagian

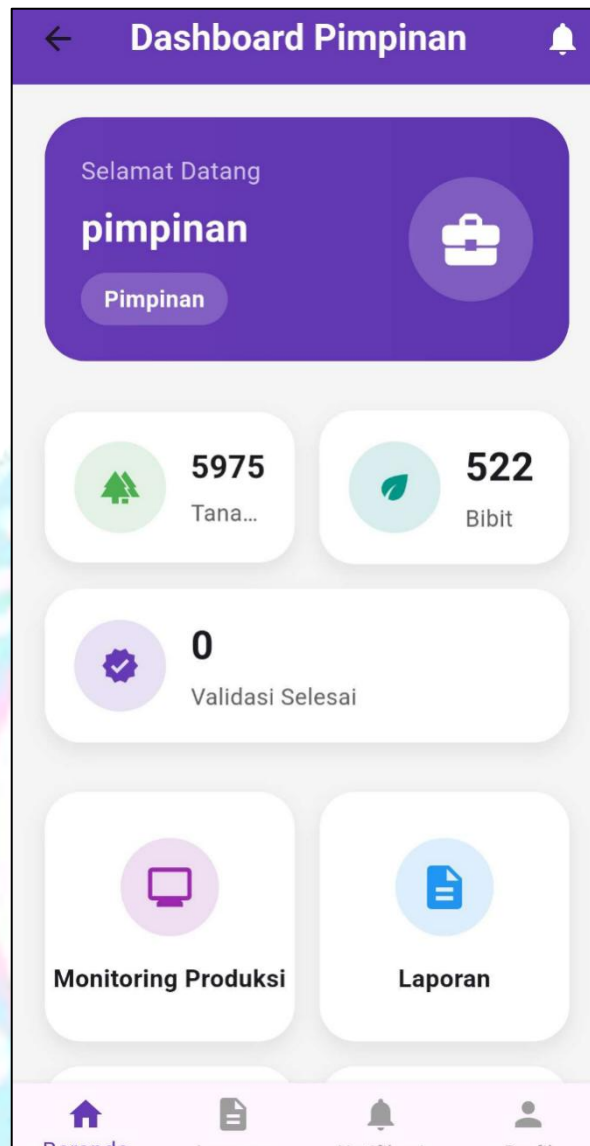


Sumber : data diolah oleh penulis, 2026

Gambar 5.27 Halaman Profil Kepala Bagian

Halaman profil digunakan untuk menampilkan informasi akun pengguna yang sedang *login* ke dalam aplikasi. Pada halaman ini pengguna dapat melihat informasi *username*, *email*, dan *role* pengguna. Selain itu, halaman profil juga menyediakan fitur *logout* yang digunakan untuk keluar dari aplikasi dan kembali menuju halaman *login*.

### 5.3.23. Halaman *Dashboard Pimpinan*

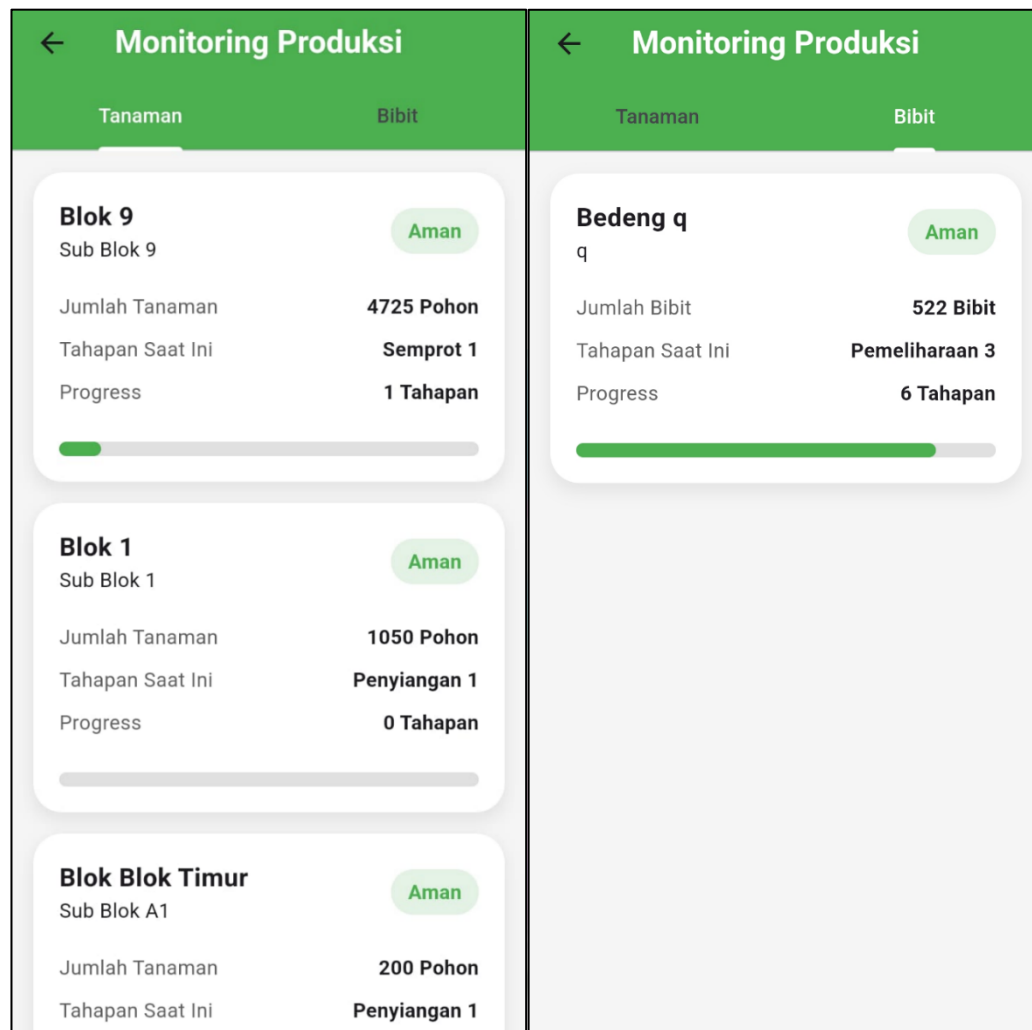


Sumber : data diolah oleh penulis, 2026

Gambar 5.28 Halaman *Dashboard Pimpinan*

Halaman *dashboard* pimpinan digunakan oleh pimpinan perusahaan untuk melakukan *monitoring* hasil produksi secara keseluruhan. Pada halaman ini ditampilkan informasi ringkasan data produksi bibit dan tanaman dalam bentuk tabel maupun grafik *monitoring* produksi. Pimpinan dapat melihat perkembangan data produksi, jumlah produksi yang telah selesai divalidasi, serta laporan hasil *monitoring* produksi secara *realtime* melalui aplikasi *android*. Tampilan *dashboard* pimpinan dibuat lebih sederhana dan fokus pada informasi laporan serta *monitoring* hasil produksi perusahaan.

### 5.3.24. Halaman *Monitoring Produksi*



Sumber : data diolah oleh penulis, 2026

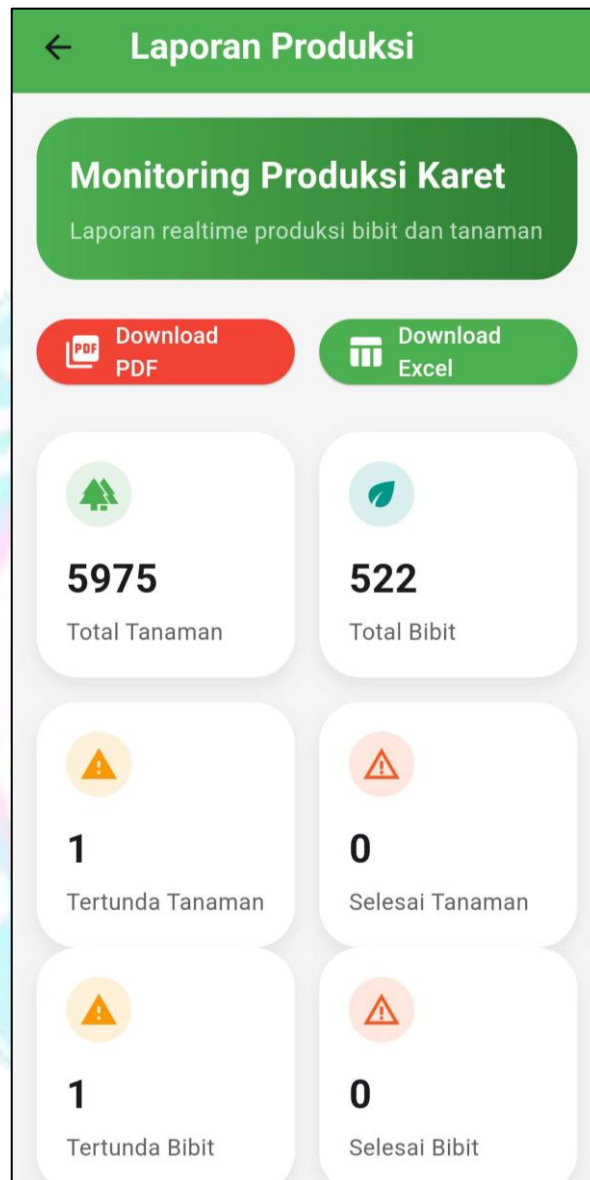
Gambar 5.29 Halaman *Monitoring Produksi*

Halaman *monitoring* produksi digunakan untuk memantau seluruh proses produksi bibit dan tanaman karet yang sedang berjalan secara *realtime*. Pada halaman ini pengguna dapat melihat perkembangan tahapan produksi, status realisasi kegiatan, jadwal kegiatan berikutnya, serta *progress* pekerjaan yang telah dilakukan oleh karyawan lapangan.

Halaman *monitoring* produksi membantu kepala bagian dan pimpinan dalam melakukan pengawasan terhadap aktivitas produksi tanpa harus melakukan pengecekan langsung ke lapangan. Informasi yang ditampilkan akan diperbarui secara *realtime* berdasarkan data yang di *input* oleh pengguna melalui aplikasi *android*. Selain itu, halaman *monitoring* produksi juga mempermudah proses

evaluasi kegiatan produksi karena seluruh riwayat tahapan produksi dapat dilihat secara langsung pada sistem.

### 5.3.25. Halaman Laporan



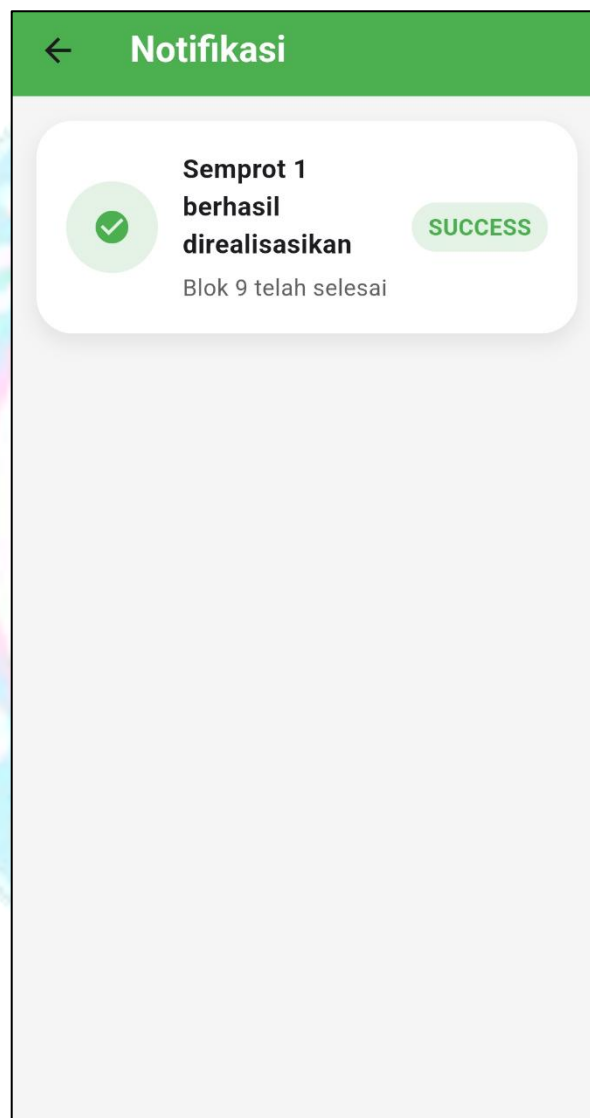
Sumber : data diolah oleh penulis, 2026

Gambar 5.30 Halaman Laporan

Halaman laporan digunakan untuk menampilkan hasil rekapitulasi data produksi bibit dan tanaman karet yang telah tersimpan pada sistem. Pada halaman ini data ditampilkan dalam bentuk tabel dan grafik untuk mempermudah pengguna dalam membaca informasi hasil produksi. Laporan produksi berisi informasi jumlah produksi bibit, jumlah produksi tanaman, data produksi yang telah

divalidasi, serta statistik hasil *monitoring* produksi secara keseluruhan. Selain ditampilkan pada aplikasi, laporan juga dapat dicetak atau diunduh dalam format *pdf* maupun *excel* sebagai dokumentasi perusahaan. Halaman laporan membantu pimpinan dan kepala bagian dalam melakukan evaluasi hasil produksi secara lebih cepat, terstruktur, dan efisien.

### 5.3.26. Halaman Notifikasi



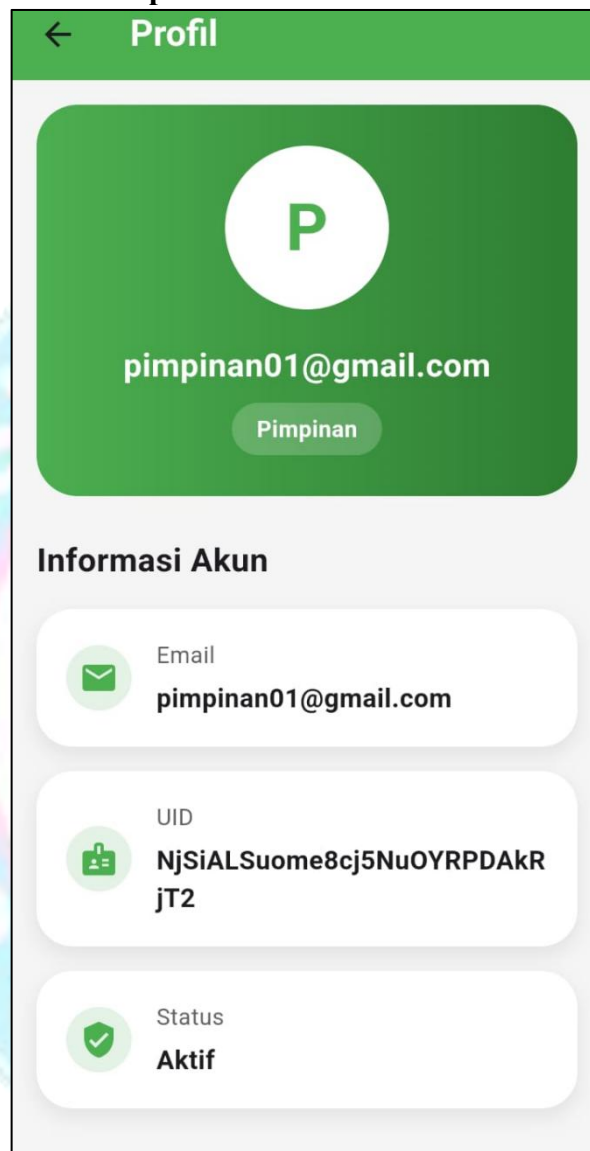
Sumber : data diolah oleh penulis, 2026

Gambar 5.31 Halaman Notifikasi

Halaman notifikasi digunakan untuk menampilkan informasi pengingat jadwal kegiatan produksi yang akan datang. Sistem akan memberikan notifikasi berdasarkan jadwal tahapan produksi bibit maupun tanaman yang mendekati waktu

pelaksanaan. Halaman ini membantu pengguna agar tidak terlambat dalam melaksanakan kegiatan produksi di lapangan.

### 5.3.27. Halaman Profil Pimpinan



Sumber : data diolah oleh penulis, 2026

Gambar 5.32 Halaman Profil Pimpinan

Halaman profil digunakan untuk menampilkan informasi akun pengguna yang sedang *login* ke dalam aplikasi. Pada halaman ini pengguna dapat melihat informasi *username*, *email*, dan *role* pengguna. Selain itu, halaman profil juga menyediakan fitur *logout* yang digunakan untuk keluar dari aplikasi dan kembali menuju halaman *login*.

## 5.4 Hasil Pengujian Sistem

### 5.4.1. Pengujian Fungsi *Login*

#### a. Potongan *Source Code* Fungsi *Login*

Fungsi *login* pada aplikasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android* digunakan untuk melakukan proses autentikasi pengguna sebelum masuk ke dalam sistem. Proses *login* dilakukan menggunakan *firebase authentication* dan *firebase firestore* untuk memeriksa *role* pengguna berdasarkan *collection* yang tersedia, yaitu *karyawan*, *kepala\_bagian*, dan *pimpinan*. Berikut adalah potongan *source code* fungsi *login* yang digunakan dalam pengujian:

```
Future<void> loginUser() async {
  return;
}

DocumentSnapshot kepala =
  await FirebaseFirestore.instance
    .collection("kepala_bagian")
    .doc(uid)
    .get();

if (kepala.exists) {
  Navigator.pushReplacement(
    context,
    MaterialPageRoute(
      builder: (_) => KepalaBagianDashboard(
        username: kepala['username'],
      ),
    ),
  );
  return;
}
```

```

DocumentSnapshot pimpinan =
    await FirebaseFirestore.instance
        .collection("pimpinan")
        .doc(uid)
        .get();

if (pimpinan.exists) {
    Navigator.pushReplacement(
        context,
        MaterialPageRoute(
            builder: (_) => PimpinanDashboard(
                username: pimpinan['username'],
            ),
        ),
    );
    return;
}

on FirebaseAuthException catch (e) {
    ScaffoldMessenger.of(context).showSnackBar(
        SnackBar(
            content: Text("Login gagal"),
        ),
    );
}

setState() {

```

```

isLoading = false;

});

}

```

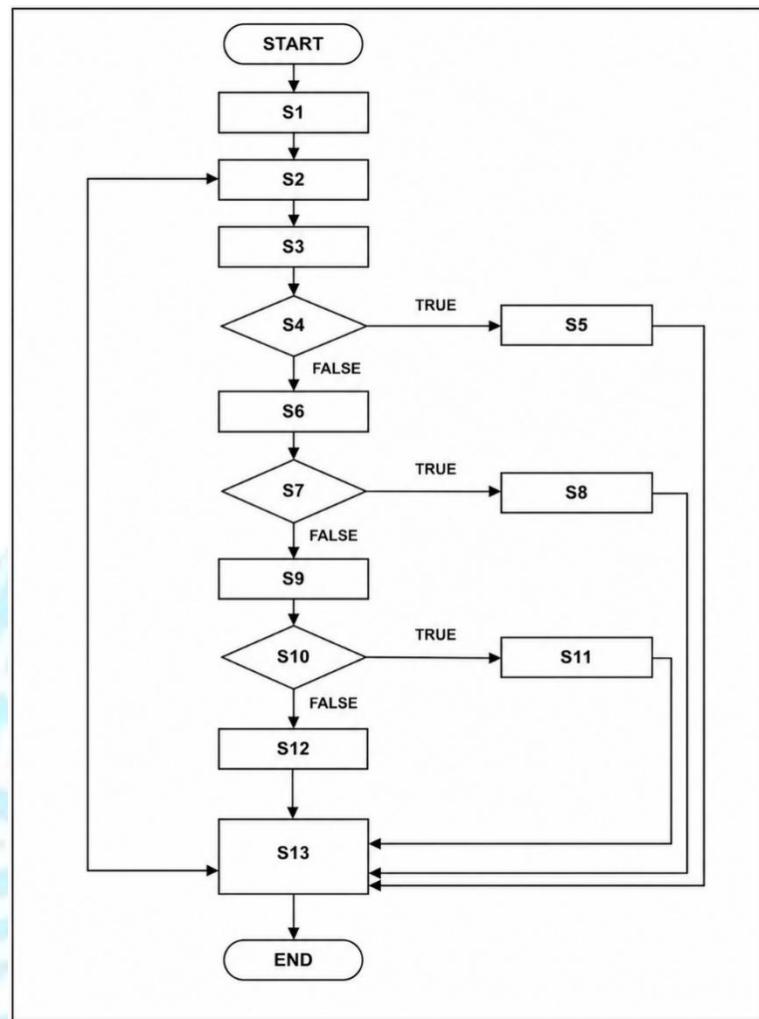
### b. Identifikasi *Statement Fungsi Login*

Tabel 5.2 Identifikasi *Statement Fungsi Login*

| Statement | Source Code                 | Keterangan                    |
|-----------|-----------------------------|-------------------------------|
| S1        | isLoading = true            | Mengaktifkan loading          |
| S2        | signInWithEmailAndPassword  | Login Firebase Authentication |
| S3        | collection("karyawan")      | Mengecek data karyawan        |
| S4        | if(karyawan.exists)         | Validasi role karyawan        |
| S5        | Navigator.pushReplacement   | Masuk dashboard karyawan      |
| S6        | collection("kepala_bagian") | Mengecek data kepala bagian   |
| S7        | if(kepala.exists)           | Validasi role kepala bagian   |
| S8        | Navigator.pushReplacement   | Masuk dashboard kepala bagian |
| S9        | collection("pimpinan")      | Mengecek data pimpinan        |
| S10       | if(pimpinan.exists)         | Validasi role pimpinan        |
| S11       | Navigator.pushReplacement   | Masuk dashboard pimpinan      |
| S12       | FirebaseAuthException       | Menangani error login         |
| S13       | isLoading = false           | Menonaktifkan loading         |

Sumber : data diolah oleh penulis, 2026

c. *Flowchart Fungsi Login*



Sumber : data diolah oleh penulis, 2026

Gambar 5.33 *Flowchart Fungsi Login*

Berdasarkan diagram alur pada gambar *flowchart* fungsi *login* di atas, urutan logika kerja fungsi *login* dapat dijabarkan sebagai berikut:

1. S1 (Mengaktifkan *Loading*)

Sistem memulai proses *login* dengan mengaktifkan status *loading* menggunakan *isLoading = true* agar pengguna mengetahui bahwa proses autentikasi sedang berlangsung.

2. S2 (*Login Firebase Authentication*)

Sistem melakukan proses autentikasi menggunakan *firebase authentication* melalui fungsi *signInWithEmailAndPassword()* berdasarkan *email* dan *password* yang dimasukkan pengguna.

3. S3 dan S4 (Pengecekan Data Karyawan)

Sistem melakukan pengecekan data pengguna pada *collection* karyawan. Jika kondisi pada S4 bernilai *true*, maka sistem menjalankan S5 untuk mengarahkan pengguna menuju *dashboard* karyawan.

4. S5 (*Dashboard* Karyawan)

Pengguna berhasil *login* sebagai karyawan dan sistem mengarahkan pengguna menuju halaman *dashboard* karyawan.

5. S6 dan S7 (Pengecekan Data Kepala Bagian)

Jika pengguna tidak ditemukan pada *collection* karyawan, maka sistem melanjutkan pengecekan pada *collection* kepala\_bagian. Jika kondisi pada S7 bernilai *true*, maka sistem menjalankan S8 untuk mengarahkan pengguna menuju *dashboard* kepala bagian.

6. S8 (*Dashboard* Kepala Bagian)

Pengguna berhasil *login* sebagai kepala bagian dan sistem mengarahkan pengguna menuju halaman *dashboard* kepala bagian.

7. S9 dan S10 (Pengecekan Data Pimpinan)

Jika pengguna tidak ditemukan pada *collection* kepala\_bagian, maka sistem melakukan pengecekan data pada *collection* pimpinan. Jika kondisi pada S10 bernilai *true*, maka sistem menjalankan S11 untuk mengarahkan pengguna menuju *dashboard* pimpinan.

8. S11 (*Dashboard* Pimpinan)

Pengguna berhasil *login* sebagai pimpinan dan sistem mengarahkan pengguna menuju halaman *dashboard* pimpinan.

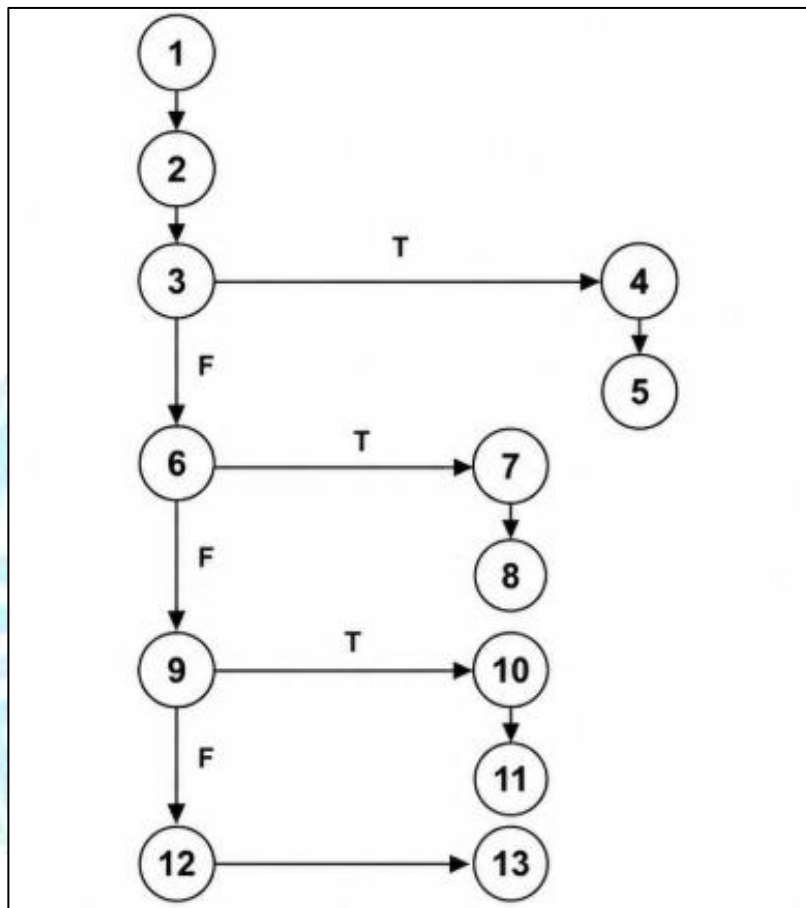
9. S12 (Penanganan *Error Login*)

Jika proses autentikasi gagal, seperti email atau *password* salah, maka sistem menampilkan pesan kesalahan login menggunakan *FirebaseAuthException*.

#### 10. S13 (Menonaktifkan *Loading*)

Setelah seluruh proses *login* selesai dilakukan, sistem menonaktifkan status *loading* menggunakan *isLoading = false* dan proses login berakhir.

#### d. *Flowgraph Fungsi Login*



Sumber : data diolah oleh penulis, 2026

Gambar 5.34 *Flowgraph Fungsi Login*

Berdasarkan gambar *flowgraph fungsi login* di atas, alur proses dimulai dari *node 1* yang menunjukkan proses awal sistem dengan mengaktifkan status *loading* menggunakan *isLoading = true*. Selanjutnya sistem menuju *node 2* untuk melakukan autentikasi pengguna menggunakan *firebase authentication* melalui fungsi *signInWithEmailAndPassword()*. Setelah proses autentikasi berhasil dilakukan, sistem melanjutkan ke *node 3* untuk melakukan pengecekan data pengguna pada *collection* karyawan. Pada *node* tersebut terdapat percabangan kondisi. Jika kondisi bernilai *true* atau data pengguna ditemukan sebagai karyawan, maka sistem menuju *node 4* untuk melakukan validasi data pengguna sebagai

karyawan dan dilanjutkan ke *node* 5 untuk mengarahkan pengguna menuju *dashboard* karyawan.

Namun apabila kondisi pada *node* 4 bernilai *false*, maka sistem melanjutkan proses ke *node* 6 untuk melakukan pengecekan data pengguna pada *collection* kepala\_bagian. Selanjutnya pada *node* 7 dilakukan validasi apakah pengguna terdaftar sebagai kepala bagian. Jika kondisi bernilai *true* maka sistem menuju *node* 8 untuk mengarahkan pengguna menuju *dashboard* kepala bagian. Apabila pengguna tidak ditemukan pada *collection* kepala\_bagian, maka sistem melanjutkan proses ke *node* 9 untuk melakukan pengecekan data pengguna pada *collection* pimpinan. Pada *node* 10 dilakukan validasi data pengguna pimpinan. Jika kondisi bernilai *true* maka sistem menuju *node* 11 untuk mengarahkan pengguna menuju *dashboard* pimpinan.

Sebaliknya, apabila proses autentikasi gagal seperti *email* atau *password* salah, maka sistem akan menuju *node* 12 yang berfungsi untuk menangani kesalahan autentikasi menggunakan *firebaseauthexception*. Setelah proses selesai dilakukan, sistem menuju *node* 13 untuk menonaktifkan status *loading* menggunakan *isLoading = false* dan proses *login* berakhir. Berdasarkan *flowgraph* tersebut dapat diketahui bahwa fungsi *login* menggunakan beberapa percabangan kondisi untuk menentukan *role* pengguna berdasarkan data pada *firebase firestore* sehingga setiap pengguna dapat diarahkan menuju *dashboard* sesuai hak akses masing-masing pada aplikasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android*.

#### **e. Cyclomatic Complexity Fungsi Login**

Berdasarkan *flowgraph* yang telah dibentuk, terdapat 3 *predicate node* (P), yaitu pada *node* 4 yang melakukan validasi data pengguna pada *collection* karyawan, *node* 7 yang melakukan validasi data pengguna pada *collection* kepala\_bagian, dan *node* 10 yang melakukan validasi data pengguna pada *collection* pimpinan. Ketiga *node* tersebut merupakan percabangan kondisi yang menentukan alur proses *login* berdasarkan *role* pengguna pada sistem. Oleh karena itu, nilai *cyclomatic complexity* dapat dihitung menggunakan rumus berikut:

$$V(G) = P + 1$$

$$V(G) = 3 + 1$$

$$V(G) = 4$$

Dari hasil perhitungan tersebut diperoleh nilai *cyclomatic complexity* sebesar 4. Nilai tersebut menunjukkan bahwa terdapat empat jalur independen (*independent path*) dalam proses eksekusi fungsi *login* yang merepresentasikan seluruh kemungkinan alur logika berdasarkan kondisi yang terjadi pada program. Nilai *cyclomatic complexity* tersebut kemudian digunakan sebagai dasar untuk menentukan jumlah *independent path* yang akan diuji pada tahap pengujian berikutnya, sehingga setiap jalur eksekusi pada fungsi *login* dapat diverifikasi untuk memastikan seluruh kondisi pada sistem autentikasi telah berjalan dengan benar sesuai kebutuhan aplikasi.

#### **f. Independent Path Fungsi Login**

Berdasarkan hasil perhitungan *cyclomatic complexity* yang bernilai 4, maka terdapat empat *independent path* yang harus diuji pada fungsi *login*. Jalur independen tersebut digunakan untuk merepresentasikan seluruh kemungkinan alur eksekusi program berdasarkan kondisi yang terjadi pada sistem autentikasi pengguna. Berikut merupakan *independent path* pada fungsi *login*:

1. *Path 1*

$1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 13$

*Path* ini menunjukkan kondisi ketika pengguna berhasil *login* sebagai karyawan sehingga sistem langsung mengarahkan pengguna menuju *dashboard* karyawan.

2. *Path 2*

$1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 13$

*Path* ini menunjukkan kondisi ketika data pengguna tidak ditemukan pada *collection* karyawan, namun ditemukan pada *collection* kepala bagian sehingga sistem mengarahkan pengguna menuju *dashboard* kepala bagian.

3. *Path 3*

$$1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 6 \rightarrow 7 \rightarrow 9 \rightarrow 10 \rightarrow 11 \rightarrow 13$$

*Path* ini menunjukkan kondisi ketika data pengguna tidak ditemukan pada *collection* karyawan dan kepala bagian, namun ditemukan pada *collection* pimpinan sehingga sistem mengarahkan pengguna menuju *dashboard* pimpinan.

4. *Path 4*

$$1 \rightarrow 2 \rightarrow 12 \rightarrow 13$$

*Path* ini menunjukkan kondisi ketika proses autentikasi gagal, seperti email atau *password* salah, sehingga sistem menampilkan pesan *error* login menggunakan *FirebaseAuthException*.

**g. Test Case Pengujian Fungsi Login**

Pengujian *test case* dilakukan untuk memastikan fungsi *login* dapat berjalan dengan baik sesuai dengan kondisi input yang diberikan oleh pengguna. Pengujian dilakukan terhadap beberapa kemungkinan kondisi autentikasi pengguna berdasarkan *role* yang terdapat pada sistem. Berikut merupakan hasil *test case* pada fungsi *login*:

Tabel 5.2 *Test Case* Fungsi Login

| Test Case | Email              | Password                | Jalur Eksekusi                                                                                                        | Output                         |
|-----------|--------------------|-------------------------|-----------------------------------------------------------------------------------------------------------------------|--------------------------------|
| TC1       | Kosong             | Kosong                  | $1 \rightarrow 2 \rightarrow 12$                                                                                      | Pesan error login tampil       |
| TC2       | Email valid        | karyawan Password benar | $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5$                                                           | Dashboard karyawan tampil      |
| TC3       | Email bagian valid | kepala Password benar   | $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 6 \rightarrow 7 \rightarrow 8$                               | Dashboard kepala bagian tampil |
| TC4       | Email valid        | pimpinan Password benar | $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 6 \rightarrow 7 \rightarrow 9 \rightarrow 10 \rightarrow 11$ | Dashboard pimpinan tampil      |
| TC5       | Email valid        | Password salah          | $1 \rightarrow 2 \rightarrow 12$                                                                                      | Pesan error login tampil       |
| TC6       | Email terdaftar    | tidak Password valid    | $1 \rightarrow 2 \rightarrow 12$                                                                                      | Pesan error login tampil       |

Sumber : data diolah oleh penulis, 2026

#### h. *Statement Coverage Fungsi Login*

Berdasarkan hasil pengujian yang telah dilakukan menggunakan beberapa *test case*, seluruh *statement* pada fungsi *login* berhasil dieksekusi sesuai dengan alur logika program. Pengujian dilakukan untuk memastikan setiap *statement* pada proses autentikasi pengguna dapat berjalan dengan baik pada seluruh kondisi yang mungkin terjadi. Berikut merupakan jalur *statement* yang dieksekusi pada masing-masing *test case*:

1. TC1

$S1 \rightarrow S2 \rightarrow S12 \rightarrow S13$

2. TC2

$S1 \rightarrow S2 \rightarrow S3 \rightarrow S4 \rightarrow S5 \rightarrow S13$

3. TC3

$S1 \rightarrow S2 \rightarrow S3 \rightarrow S4 \rightarrow S6 \rightarrow S7 \rightarrow S8 \rightarrow S13$

4. TC4

$S1 \rightarrow S2 \rightarrow S3 \rightarrow S4 \rightarrow S6 \rightarrow S7 \rightarrow S9 \rightarrow S10 \rightarrow S11 \rightarrow S13$

5. TC5

$S1 \rightarrow S2 \rightarrow S12 \rightarrow S13$

6. TC6

$S1 \rightarrow S2 \rightarrow S12 \rightarrow S13$

Berdasarkan hasil pengujian tersebut, seluruh *statement* pada fungsi *login* telah berhasil dijalankan minimal satu kali selama proses pengujian berlangsung. Dengan demikian nilai *statement coverage* pada fungsi *login* diperoleh sebesar:

$$\text{Statement Coverage} = (\text{Jumlah statement yang dieksekusi} / \text{Total statement}) \times 100\%$$

$$\text{Statement Coverage} = (13 / 13) \times 100\%$$

$$\text{Statement Coverage} = 100\%$$

Hasil tersebut menunjukkan bahwa seluruh *statement* pada fungsi *login* telah diuji dan dieksekusi dengan baik sehingga fungsi *login* dinyatakan berjalan sesuai kebutuhan sistem aplikasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android*.

### 5.4.2. Pengujian Fungsi *Logout*

#### a. Potongan Source Code Fungsi *Logout*

Fungsi logout pada aplikasi monitoring informasi data produksi bibit tanaman karet berbasis android digunakan untuk mengakhiri sesi pengguna yang sedang login pada sistem. Proses logout dilakukan menggunakan firebase authentication dengan menghapus sesi autentikasi pengguna, kemudian sistem akan mengarahkan pengguna kembali menuju halaman login. Berikut adalah potongan source code fungsi logout yang digunakan dalam pengujian:

```
Future<void> logout(
    BuildContext context,
) async {

    await FirebaseAuth.instance.signOut();

    Navigator.pushAndRemoveUntil(
        context,
        MaterialPageRoute(
            builder: (_) => const LoginPage(),
        ),
        (route) => false,
    );
}
```

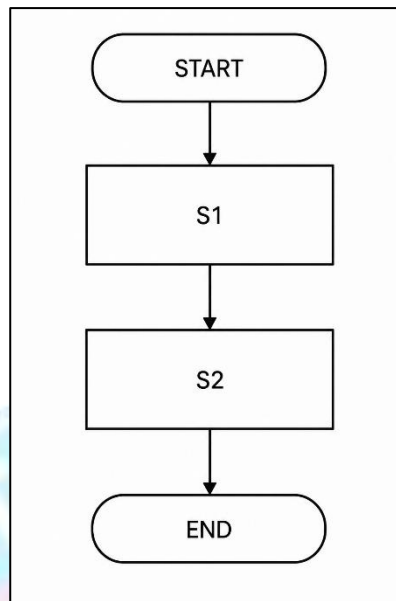
#### b. Identifikasi Statement Fungsi *Logout*

Tabel 5.3 Identifikasi Statement Fungsi *Logout*

| Statement | Source Code                     | Keterangan                            |
|-----------|---------------------------------|---------------------------------------|
| S1        | FirebaseAuth.instance.signOut() | Menghapus sesi login pengguna         |
| S2        | Navigator.pushAndRemoveUntil()  | Mengarahkan pengguna ke halaman login |

Sumber : data diolah oleh penulis, 2026

### c. *Flowchart Fungsi Logout*



Sumber : data diolah oleh penulis, 2026

Gambar 5.35 *Flowchart Fungsi Logout*

Berdasarkan diagram alur pada gambar *flowchart* fungsi *logout* di atas, urutan logika kerja fungsi *logout* dapat dijabarkan sebagai berikut:

1. S1 (Menghapus Sesi *Login* Pengguna)

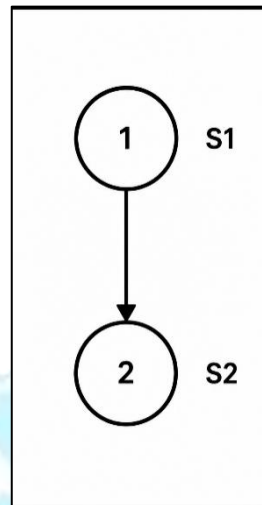
Sistem menjalankan fungsi `FirebaseAuth.instance.signOut()` untuk menghapus sesi autentikasi pengguna yang sedang *login* pada aplikasi.

2. S2 (Mengalihkan ke Halaman *Login*)

Setelah proses *logout* berhasil dilakukan, sistem menjalankan `Navigator.pushAndRemoveUntil()` untuk mengarahkan pengguna kembali menuju halaman *login* dan menghapus seluruh halaman sebelumnya dari navigasi aplikasi.

Berdasarkan *flowchart* tersebut dapat diketahui bahwa fungsi *logout* digunakan untuk mengakhiri sesi pengguna dan mengembalikan pengguna menuju halaman *login* aplikasi

#### d. *Flowgraph Fungsi Logout*



Sumber : data diolah oleh penulis, 2026

Gambar 5.36 *Flowgraph Fungsi Logout*

Berdasarkan gambar *flowgraph* fungsi *logout* di atas, alur proses dimulai dari *node* 1 yang menunjukkan proses penghapusan sesi *login* pengguna menggunakan *FirebaseAuth.instance.signOut()*. Selanjutnya sistem menuju *node* 2 untuk mengarahkan pengguna kembali menuju halaman *login* menggunakan *Navigator.pushAndRemoveUntil()*. Setelah proses selesai dilakukan, program berakhir setelah *node* 2 selesai dijalankan. Berdasarkan *flowgraph* tersebut dapat diketahui bahwa fungsi *logout* memiliki alur proses yang sederhana tanpa percabangan kondisi.

#### e. *Cyclomatic Complexity Fungsi Logout*

Berdasarkan *flowgraph* yang telah dibentuk, fungsi *logout* tidak memiliki *predicate node* (P) karena tidak terdapat percabangan kondisi pada alur program. Oleh karena itu, nilai *cyclomatic complexity* dapat dihitung menggunakan rumus berikut:

$$V(G) = P + 1$$

$$V(G) = 0 + 1$$

$$V(G) = 1$$

Dari hasil perhitungan tersebut diperoleh nilai *cyclomatic complexity* sebesar 1. Nilai tersebut menunjukkan bahwa hanya terdapat satu jalur independen (*independent path*) pada fungsi *logout*.

#### f. Independent Path Fungsi Logout

Berdasarkan hasil perhitungan **cyclomatic complexity** yang bernilai 1, maka terdapat satu **independent path** yang harus diuji pada fungsi *logout*.

- *Path 1*

$1 \rightarrow 2$

*Path* ini menunjukkan kondisi ketika pengguna berhasil *logout* dari aplikasi dan sistem mengarahkan pengguna kembali menuju halaman *login*.

#### g. Test Case Pengujian Fungsi Logout

Pengujian *test case* dilakukan untuk memastikan fungsi *logout* dapat berjalan dengan baik sesuai dengan kebutuhan sistem.

Tabel 5.4 *Test Case* Fungsi Logout

| Test Case | Kondisi        | Jalur Eksekusi              | Output                                                |
|-----------|----------------|-----------------------------|-------------------------------------------------------|
| TC1       | Tombol ditekan | logout<br>$1 \rightarrow 2$ | Pengguna berhasil logout dan kembali ke halaman login |

Sumber : data diolah oleh penulis, 2026

#### h. Statement Coverage Fungsi Logout

Berdasarkan hasil pengujian yang telah dilakukan, seluruh *statement* pada fungsi *logout* berhasil dieksekusi sesuai dengan alur logika program. Berikut merupakan jalur *statement* yang dieksekusi pada fungsi *logout*:

1. TC1

$S1 \rightarrow S2$

Berdasarkan hasil pengujian tersebut, seluruh *statement* pada fungsi *logout* telah berhasil dijalankan minimal satu kali selama proses pengujian berlangsung. Dengan demikian nilai **statement coverage** pada fungsi *logout* diperoleh sebesar:  
 $Statement\ Coverage = (2 / 2) \times 100\%$

Hasil tersebut menunjukkan bahwa seluruh *statement* pada fungsi *logout* telah diuji dan dieksekusi dengan baik sehingga fungsi *logout* dinyatakan berjalan sesuai

kebutuhan sistem aplikasi monitoring informasi data produksi bibit tanaman karet berbasis *android*.

### 5.4.3. Pengujian Fungsi Simpan Data Bibit

#### a. Potongan Source Code Fungsi Simpan Data Bibit

Fungsi simpan data bibit pada aplikasi monitoring informasi data produksi bibit tanaman karet berbasis android digunakan untuk menyimpan data produksi bibit ke database firebase firestore. Proses penyimpanan data dilakukan dengan menentukan tahapan awal produksi, menghitung jadwal kegiatan berikutnya, menentukan status produksi, kemudian menyimpan seluruh data ke collection produksi\_bibit. Berikut adalah potongan source code fungsi simpan data bibit yang digunakan dalam pengujian:

```
Future<void> simpanDataBibit() async {
    DateTime sekarang = DateTime.now();

    int tahapIndex = 0;

    String tahapan = tahapanBibit[0]["nama"];

    int hari = tahapanBibit[0]["hari"];

    DateTime jadwalBerikutnya =
        sekarang.add(
            Duration(days: hari),
        );

    int selisihHari =
        jadwalBerikutnya
            .difference(sekarang)
            .inDays;

    String status = "Aman";
```

```

if (selisihHari <= 7) {
    status = "Mendekati Jadwal";
}
await FirebaseFirestore.instance
    .collection("produksi_bibit")
    .add({

        "tanggal_input":
            Timestamp.fromDate(
                sekarang,
            ),

        "nomor_bedeng":
            nomorBedengController.text,

        "jenis_bibit":
            jenisBibitController.text,

        "jumlah_bibit":
            int.tryParse(
                jumlahBibitController.text,
            ) ??
            0,

        "tahapan_pemeliharaan":
            tahapan,

        "tahap_index":
            tahapIndex,

        "riwayat_tahapan": [],

```

```
"realisasi": false,  
  
"jadwal_berikutnya":  
    Timestamp.fromDate(  
        jadwalBerikutnya,  
    ),  
  
"status": status,  
  
"created_at":  
    Timestamp.now(),  
});  
  
ScaffoldMessenger.of(context)  
    .showSnackBar(  
    const SnackBar(  
        content: Text(  
            "Data bibit berhasil disimpan",  
        ),  
    ),  
);  
  
nomorBedengController.clear();  
  
jenisBibitController.clear();  
jumlahBibitController.clear();  
}
```

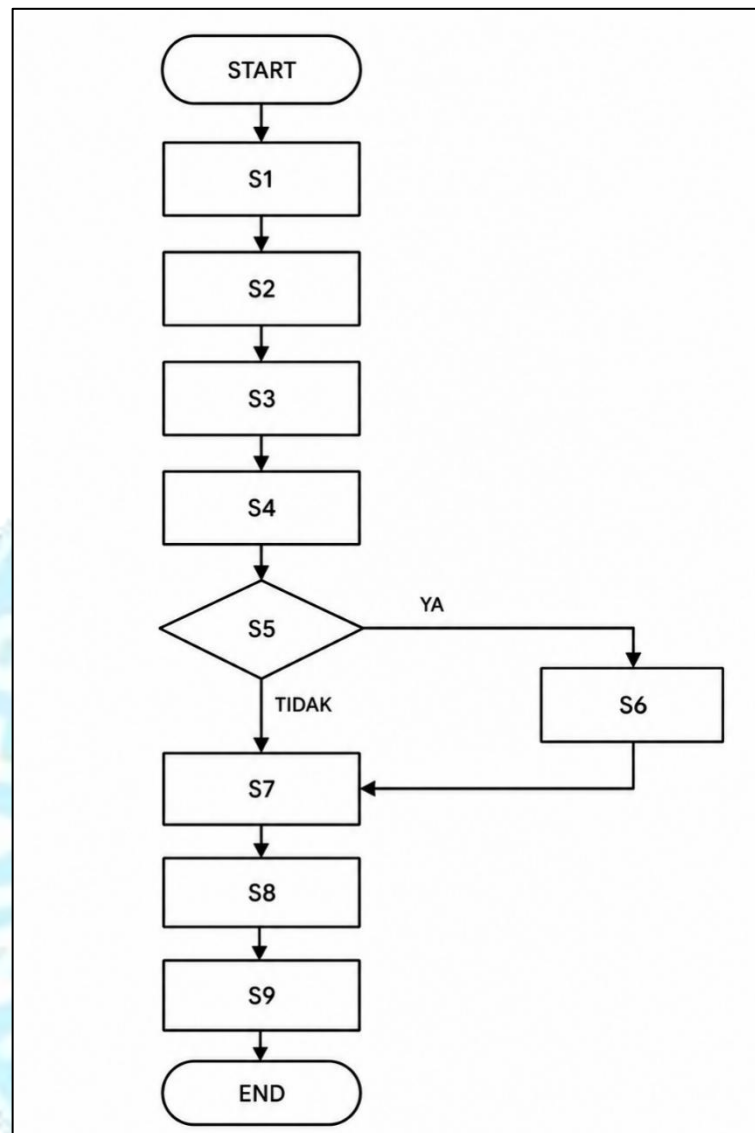
### b. Identifikasi *Statement* Fungsi Simpan Data Bibit

Tabel 5.5 Identifikasi *Statement* Fungsi Simpan Data Bibit

| Statement Source Code |                                               | Keterangan                   |
|-----------------------|-----------------------------------------------|------------------------------|
| S1                    | DateTime.now()                                | Mengambil waktu saat ini     |
| S2                    | tahapIndex = 0                                | Menentukan tahap awal        |
| S3                    | tahapanBibit[0]                               | Mengambil tahapan awal bibit |
| S4                    | sekarang.add()                                | Menghitung jadwal berikutnya |
| S5                    | if (selisihHari <= 7)                         | Validasi status jadwal       |
| S6                    | status = "Mendekati Jadwal"                   | Mengubah status jadwal       |
| S7                    | FirebaseFirestore.instance.collection().add() | Menyimpan data bibit         |
| S8                    | showSnackBar()                                | Menampilkan pesan berhasil   |
| S9                    | clear()                                       | Mengosongkan form input      |

Sumber : data diolah oleh penulis, 2026

c. *Flowchart Fungsi Simpan Data Bibit*



Sumber : data diolah oleh penulis, 2026

Gambar 5.37 *Flowchart Fungsi Simpan Data Bibit*

Berdasarkan diagram alur pada gambar *flowchart* fungsi simpan data bibit di atas, urutan logika kerja fungsi simpan data bibit dapat dijabarkan sebagai berikut:

1. S1 (Mengambil Waktu Saat Ini)

Sistem mengambil waktu saat ini menggunakan *DateTime.now()* sebagai dasar penjadwalan produksi bibit.

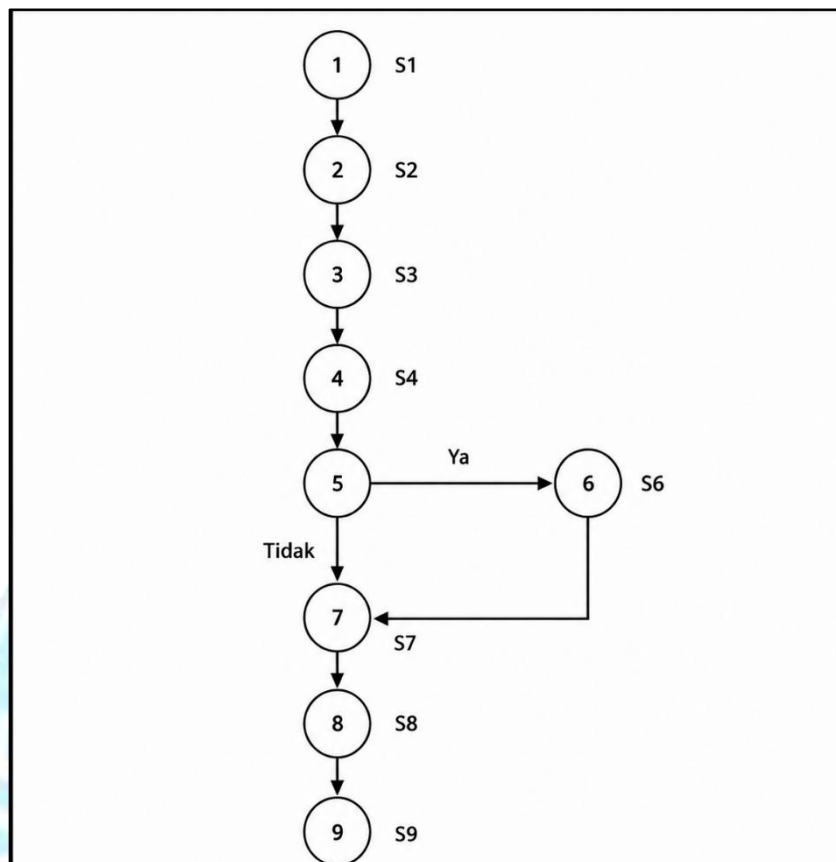
2. S2 (Menentukan Tahap Awal)

Sistem menentukan indeks tahapan awal produksi bibit dengan nilai awal 0.

3. S3 (Mengambil Tahapan Awal Bibit)  
Sistem mengambil data tahapan awal dari list tahapanBibit.
4. S4 (Menghitung Jadwal Berikutnya)  
Sistem menghitung jadwal kegiatan berikutnya berdasarkan jumlah hari pada tahapan bibit.
5. S5 (Validasi Status Jadwal)  
Sistem melakukan pengecekan apakah selisih hari menuju jadwal berikutnya kurang dari atau sama dengan tujuh hari.
6. S6 (Mengubah Status Jadwal)  
Jika kondisi bernilai *true*, maka status produksi diubah menjadi “Mendekati Jadwal”.
7. S7 (Menyimpan Data Bibit)  
Sistem menyimpan seluruh data produksi bibit ke *database firebase firestore collection* produksi\_bibit.
8. S8 (Menampilkan Pesan Berhasil)  
Sistem menampilkan notifikasi bahwa data bibit berhasil disimpan.
9. S9 (Mengosongkan *Form* Input)  
Sistem menghapus seluruh data input pada *form* setelah proses penyimpanan berhasil dilakukan.

Berdasarkan *flowchart* tersebut dapat diketahui bahwa fungsi simpan data bibit digunakan untuk mengelola proses penyimpanan data produksi bibit secara otomatis ke *firebase firestore*.

#### d. *Flowgraph* Fungsi Simpan Data Bibit



Sumber : data diolah oleh penulis, 2026

Gambar 5.38 *Flowgraph* Fungsi Simpan Data Bibit

Berdasarkan gambar *flowgraph* fungsi simpan data bibit di atas, alur proses dimulai dari *node* 1 yang menunjukkan proses pengambilan waktu saat ini menggunakan *DateTime.now()*. Selanjutnya sistem menuju *node* 2 untuk menentukan tahap awal produksi bibit dengan nilai awal 0. Setelah itu sistem melanjutkan proses ke *node* 3 untuk mengambil tahapan awal bibit dari list *tahapanBibit*. Pada *node* 4 sistem menghitung jadwal kegiatan berikutnya berdasarkan jumlah hari pada tahapan produksi bibit.

Selanjutnya pada *node* 5 dilakukan pengecekan kondisi apakah selisih hari menuju jadwal berikutnya kurang dari atau sama dengan tujuh hari. Jika kondisi pada *node* 5 bernilai *true*, maka sistem menuju *node* 6 untuk mengubah status produksi menjadi “Mendekati Jadwal”. Setelah itu proses dilanjutkan menuju *node* 7 untuk menyimpan data produksi bibit ke *database firebase firestore*. Namun apabila kondisi pada *node* 5 bernilai *false*, maka sistem langsung menuju *node* 7

tanpa mengubah status jadwal produksi. Setelah data berhasil disimpan, sistem melanjutkan proses menuju *node* 8 untuk menampilkan pesan bahwa data berhasil disimpan.

Selanjutnya sistem menuju *node* 9 untuk mengosongkan seluruh *form input* yang telah digunakan sebelumnya dan seluruh proses selesai dilakukan, setelah seluruh proses selesai dilakukan, sistem mengakhiri proses pada *node* 9. Berdasarkan *flowgraph* tersebut dapat diketahui bahwa fungsi simpan data bibit memiliki satu percabangan kondisi yang digunakan untuk menentukan status jadwal produksi sebelum data disimpan ke *firebase firestore*.

#### e. *Cyclomatic Complexity* Fungsi Simpan Data Bibit

Berdasarkan *flowgraph* yang telah dibentuk, terdapat 1 *predicate node* (P), yaitu pada *node* 5 yang melakukan pengecekan kondisi status jadwal produksi bibit. Oleh karena itu, nilai *cyclomatic complexity* dapat dihitung menggunakan rumus berikut:

$$V(G) = P + 1$$

$$V(G) = 1 + 1$$

$$V(G) = 2$$

Dari hasil perhitungan tersebut diperoleh nilai *cyclomatic complexity* sebesar 2. Nilai tersebut menunjukkan bahwa terdapat dua jalur independen (*independent path*) pada fungsi simpan data bibit.

#### f. *Independent Path* Fungsi Simpan Data Bibit

Berdasarkan hasil perhitungan **cyclomatic complexity** yang bernilai 2, maka terdapat dua **independent path** yang harus diuji pada fungsi simpan data bibit.

##### a. *Path* 1

$$1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9$$

*Path* ini menunjukkan kondisi ketika status jadwal produksi mendekati jadwal sehingga sistem mengubah status menjadi “Mendekati Jadwal” sebelum data disimpan.

b. *Path 2*

1 → 2 → 3 → 4 → 5 → 7 → 8 → 9

*Path* ini menunjukkan kondisi ketika jadwal produksi masih aman sehingga sistem langsung menyimpan data tanpa mengubah status jadwal.

g. **Test Case Pengujian Fungsi Simpan Data Bibit**Tabel 5.6 *Test Case* Fungsi Simpan Data Bibit

| Test Case | Kondisi               | Jalur Eksekusi                    | Output                                                  |
|-----------|-----------------------|-----------------------------------|---------------------------------------------------------|
| TC1       | Selisih hari $\leq 7$ | 1 → 2 → 3 → 4 → 5 → 6 → 7 → 8 → 9 | Data berhasil disimpan dengan status “Mendekati Jadwal” |
| TC2       | Selisih hari $> 7$    | 1 → 2 → 3 → 4 → 5 → 7 → 8 → 9     | Data berhasil disimpan dengan status “Aman”             |

Sumber : data diolah oleh penulis, 2026

h. **Statement Coverage Fungsi Simpan Data Bibit**

Berdasarkan hasil pengujian yang telah dilakukan, seluruh *statement* pada fungsi simpan data bibit berhasil dieksekusi sesuai dengan alur logika program. Berikut merupakan jalur *statement* yang dieksekusi pada masing-masing **test case**:

## 1. TC1

S1 → S2 → S3 → S4 → S5 → S6 → S7 → S8 → S9

## 2. TC2

S1 → S2 → S3 → S4 → S5 → S7 → S8 → S9

Berdasarkan hasil pengujian tersebut, seluruh *statement* pada fungsi simpan data bibit telah berhasil dijalankan minimal satu kali selama proses pengujian berlangsung. Dengan demikian nilai **statement coverage** pada fungsi simpan data bibit diperoleh sebesar:

$$\text{Statement Coverage} = (9 / 9) \times 100\%$$

Hasil tersebut menunjukkan bahwa seluruh *statement* pada fungsi simpan data bibit telah diuji dan dieksekusi dengan baik sehingga fungsi simpan data bibit

dinyatakan berjalan sesuai kebutuhan sistem aplikasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android*.

#### 5.4.4. Pengujian Fungsi Simpan Data Tanaman

##### a. Potongan *Source Code* Fungsi Simpan Data Tanaman

Fungsi *simpan data tanaman* pada aplikasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android* digunakan untuk menyimpan data produksi tanaman ke database *firebase firestore*. Proses penyimpanan data dilakukan dengan menghitung jumlah tanaman berdasarkan luas lahan, menentukan tahapan awal produksi, menghitung jadwal kegiatan berikutnya, menentukan status produksi, kemudian menyimpan seluruh data ke *collection* produksi\_tanaman. Berikut adalah potongan *source code* fungsi simpan data tanaman yang digunakan dalam pengujian:

```
Future<void> simpanDataTanaman() async {
    DateTime sekarang = DateTime.now();
    double luas = double.tryParse(
        luasController.text,
    ) ??
    0;
    int jumlahTanaman =
        (luas * 525).toInt();

    int tahapIndex = 0;

    String tahapan =
        tahapanTanaman[0]["nama"];

    int bulan =
        tahapanTanaman[0]["bulan"];
```

```
DateTime jadwalBerikutnya =
```

```
    sekarang.add(  
        Duration(  
            days: bulan * 30,  
        ),  
    );
```

```
int selisihHari =
```

```
    jadwalBerikutnya  
        .difference(sekarang)  
        .inDays;
```

```
String status = "Aman";
```

```
if (selisihHari <= 7) {  
    status = "Mendekati Jadwal";  
}
```

```
await FirebaseFirestore.instance
```

```
    .collection(  
        "produksi_tanaman",  
    )
```

```
    .add({
```

```
        "tanggal_tanam":
```

```
            Timestamp.fromDate(  
                sekarang,  
            ),
```

```
"blok":
    blokController.text,

"sub_blok":
    subBlokController.text,

"petak_ha":
    double.tryParse(
        petakController.text,
    ) ??
    0,

"luas": luas,

"jarak_tanam":
    jarakTanamController.text,

"jumlah_tanaman":
    jumlahTanaman,

"umur_tanaman": 0,

"tahapan_pemeliharaan":
    tahapan,

"tahap_index":
    tahapIndex,

"riwayat_tahapan": [],
```

```

"realisasi": false,

"jadwal_berikutnya":
    Timestamp.fromDate(
        jadwalBerikutnya,
    ),

"status": status,

"created_at":
    Timestamp.now(),
});

ScaffoldMessenger.of(context)
    .showSnackBar(
        const SnackBar(
            content: Text(
                "Data tanaman berhasil disimpan",
            ),
        ),
    );

blokController.clear();

subBlokController.clear();

petakController.clear();

luasController.clear();

```

```

    jarakTanamController.clear();

    jumlahTanamanController.clear();
}

```

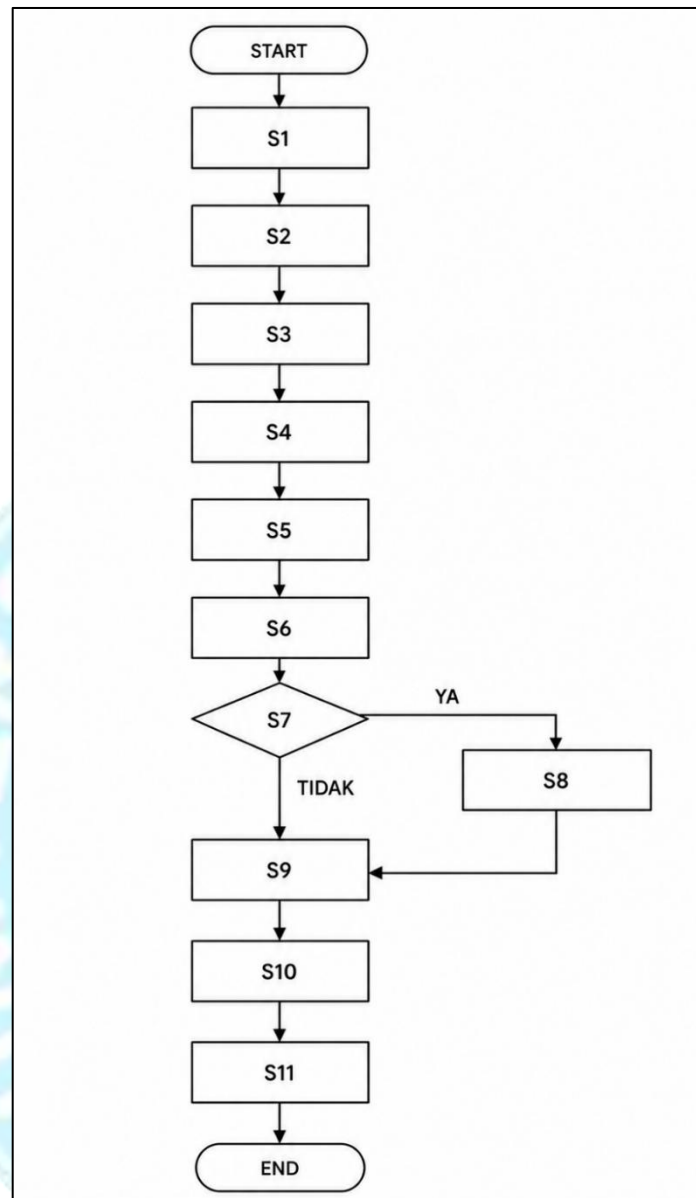
### b. Identifikasi *Statement* Fungsi Simpan Data Tanaman

Tabel 5.7 Identifikasi *Statement* Fungsi Simpan Data Tanaman

| Statement | Source Code                                                | Keterangan                     |
|-----------|------------------------------------------------------------|--------------------------------|
| S1        | <code>DateTime.now()</code>                                | Mengambil waktu saat ini       |
| S2        | <code>double.tryParse()</code>                             | Mengambil nilai luas           |
| S3        | <code>jumlahTanaman = (luas * 525).toInt()</code>          | Menghitung jumlah tanaman      |
| S4        | <code>tahapIndex = 0</code>                                | Menentukan tahap awal          |
| S5        | <code>tahapanTanaman[0]</code>                             | Mengambil tahapan awal tanaman |
| S6        | <code>sekarang.add()</code>                                | Menghitung jadwal berikutnya   |
| S7        | <code>if (selisihHari &lt;= 7)</code>                      | Validasi status jadwal         |
| S8        | <code>status = "Mendekati Jadwal"</code>                   | Mengubah status jadwal         |
| S9        | <code>FirebaseFirestore.instance.collection().add()</code> | Menyimpan data tanaman         |
| S10       | <code>showSnackBar()</code>                                | Menampilkan pesan berhasil     |
| S11       | <code>clear()</code>                                       | Mengosongkan form input        |

Sumber : data diolah oleh penulis, 2026

c. *Flowchart Fungsi Simpan Data Tanaman*



Sumber : data diolah oleh penulis, 2026

Gambar 5.39 *Flowchart Fungsi Simpan Data Tanaman*

Berdasarkan diagram alur pada gambar *flowchart* fungsi simpan data tanaman di atas, urutan logika kerja fungsi simpan data tanaman dapat dijabarkan sebagai berikut:

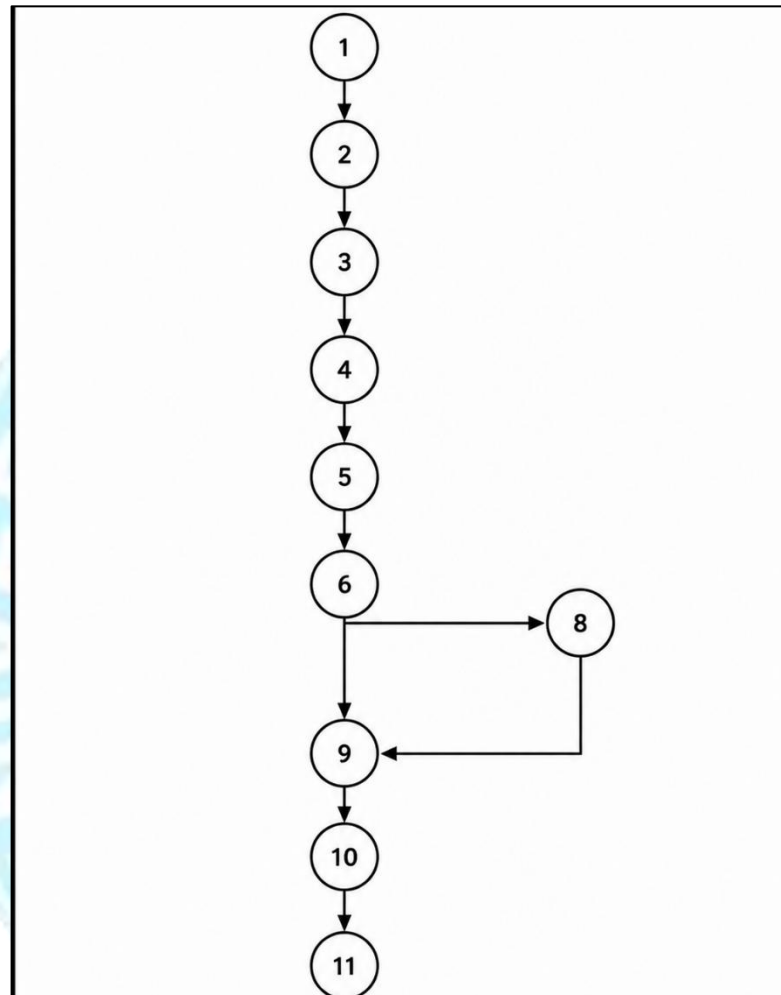
1. S1 (Mengambil Waktu Saat Ini)

Sistem mengambil waktu saat ini menggunakan *DateTime.now()* sebagai dasar penjadwalan produksi tanaman.

2. S2 (Mengambil Nilai Luas)  
Sistem mengambil nilai luas lahan dari *input* pengguna menggunakan *double.TryParse()*.
3. S3 (Menghitung Jumlah Tanaman)  
Sistem menghitung jumlah tanaman secara otomatis berdasarkan luas lahan dikalikan 525.
4. S4 (Menentukan Tahap Awal)  
Sistem menentukan indeks tahapan awal produksi tanaman dengan nilai awal 0.
5. S5 (Mengambil Tahapan Awal Tanaman)  
Sistem mengambil tahapan awal dari list tahapanTanaman.
6. S6 (Menghitung Jadwal Berikutnya)  
Sistem menghitung jadwal kegiatan berikutnya berdasarkan tahapan produksi tanaman.
7. S7 (Validasi Status Jadwal)  
Sistem melakukan pengecekan apakah selisih hari menuju jadwal berikutnya kurang dari atau sama dengan tujuh hari.
8. S8 (Mengubah Status Jadwal)  
Jika kondisi bernilai *true* maka sistem mengubah status menjadi “Mendekati Jadwal”.
9. S9 (Menyimpan Data Tanaman)  
Sistem menyimpan seluruh data produksi tanaman ke *database firebase firestore collection* produksi\_tanaman.
10. S10 (Menampilkan Pesan Berhasil)  
Sistem menampilkan notifikasi bahwa data tanaman berhasil disimpan.
11. S11 (Mengosongkan *Form Input*)  
Sistem menghapus seluruh data input pada *form* setelah proses penyimpanan berhasil dilakukan.

Berdasarkan *flowchart* tersebut dapat diketahui bahwa fungsi simpan data tanaman digunakan untuk mengelola proses penyimpanan data produksi tanaman secara otomatis ke *firebase firestore*.

#### d. *Flowgraph* Fungsi Simpan Data Tanaman



Sumber : data diolah oleh penulis, 2026

Gambar 5.40 *Flowgraph* Fungsi Simpan Data Tanaman

Berdasarkan gambar *flowgraph* fungsi Simpan Data Tanaman di atas, alur proses dimulai dari *node* 1 yang menunjukkan proses pengambilan waktu saat ini menggunakan *DateTime.now()*. Selanjutnya sistem menuju *node* 2 untuk mengambil nilai luas lahan dari input pengguna. Pada *node* 3 sistem menghitung jumlah tanaman secara otomatis berdasarkan luas lahan. Selanjutnya sistem menuju *node* 4 untuk menentukan tahap awal produksi tanaman dan *node* 5 untuk mengambil tahapan awal tanaman dari list *tahapanTanaman*. Pada *node* 6 sistem menghitung jadwal kegiatan berikutnya berdasarkan tahapan produksi tanaman.

Pada *node 7* dilakukan pengecekan kondisi apakah selisih hari menuju jadwal berikutnya kurang dari atau sama dengan tujuh hari. Jika kondisi pada *node 7* bernilai *true* maka sistem menuju *node 8* untuk mengubah status produksi menjadi “Mendekati Jadwal”.

Selanjutnya sistem menuju *node 9* untuk menyimpan data produksi tanaman ke *firebase firestore*. Setelah data berhasil disimpan, sistem melanjutkan proses menuju *node 10* untuk menampilkan pesan bahwa data berhasil disimpan dan *node 11* untuk mengosongkan seluruh *form input*. Setelah seluruh proses selesai dilakukan, setelah seluruh proses selesai dilakukan, sistem mengakhiri proses pada *node 11*. Berdasarkan *flowgraph* tersebut dapat diketahui bahwa fungsi simpan data tanaman memiliki satu percabangan kondisi yang digunakan untuk menentukan status jadwal produksi sebelum data disimpan ke *firebase firestore*.

#### e. *Cyclomatic Complexity* Fungsi Simpan Data Tanaman

Berdasarkan *flowgraph* yang telah dibentuk, terdapat 1 *predicate node* (P), yaitu pada *node 7* yang melakukan pengecekan kondisi status jadwal produksi tanaman. Oleh karena itu, nilai *cyclomatic complexity* dapat dihitung menggunakan rumus berikut:

$$V(G) = P + 1$$

$$V(G) = 1 + 1$$

$$V(G) = 2$$

Dari hasil perhitungan tersebut diperoleh nilai *cyclomatic complexity* sebesar 2. Nilai tersebut menunjukkan bahwa terdapat dua jalur independen (*independent path*) pada fungsi simpan data tanaman.

#### f. *Independent Path* Fungsi Simpan Data Tanaman

Berdasarkan hasil perhitungan *cyclomatic complexity* yang bernilai 2, maka terdapat dua **independent path** yang harus diuji pada fungsi simpan data tanaman.

##### 1. *Path 1*

$$1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 10 \rightarrow 11$$

*Path* ini menunjukkan kondisi ketika status jadwal produksi mendekati jadwal sehingga sistem mengubah status menjadi “Mendekati Jadwal” sebelum data disimpan.

## 2. *Path* 2

1 → 2 → 3 → 4 → 5 → 6 → 7 → 9 → 10 → 11

*Path* ini menunjukkan kondisi ketika jadwal produksi masih aman sehingga sistem langsung menyimpan data tanpa mengubah status jadwal.

## g. Test Case Pengujian Fungsi Simpan Data Tanaman

Tabel 5.8 *Test Case* Fungsi Simpan Data Tanaman

| Test Case | Kondisi          | Jalur Eksekusi                              | Output                                                  |
|-----------|------------------|---------------------------------------------|---------------------------------------------------------|
| TC1       | Selisih hari ≤ 7 | 1 → 2 → 3 → 4 → 5 → 6 → 7 → 8 → 9 → 10 → 11 | Data berhasil disimpan dengan status “Mendekati Jadwal” |
| TC2       | Selisih hari > 7 | 1 → 2 → 3 → 4 → 5 → 6 → 7 → 9 → 10 → 11     | Data berhasil disimpan dengan status “Aman”             |

Sumber : data diolah oleh penulis, 2026

## h. Statement Coverage Fungsi Simpan Data Tanaman

Berdasarkan hasil pengujian yang telah dilakukan, seluruh *statement* pada fungsi simpan data tanaman berhasil dieksekusi sesuai dengan alur logika program. Berikut merupakan jalur *statement* yang dieksekusi pada masing-masing **test case**:

### 1. TC1

S1 → S2 → S3 → S4 → S5 → S6 → S7 → S8 → S9 → S10 → S11

### 2. TC2

S1 → S2 → S3 → S4 → S5 → S6 → S7 → S9 → S10 → S11

Berdasarkan hasil pengujian tersebut, seluruh *statement* pada fungsi simpan data tanaman telah berhasil dijalankan minimal satu kali selama proses pengujian berlangsung. Dengan demikian nilai **statement coverage** pada fungsi simpan data tanaman diperoleh sebesar:

$$\text{Statement Coverage} = (11 / 11) \times 100\%$$

Hasil tersebut menunjukkan bahwa seluruh *statement* pada fungsi simpan data tanaman telah diuji dan dieksekusi dengan baik sehingga fungsi simpan data tanaman dinyatakan berjalan sesuai kebutuhan sistem aplikasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android*.

#### 5.4.5. Pengujian Fungsi Validasi Data

##### a. Potongan *Source Code* Fungsi Validasi Data

Fungsi validasi data pada aplikasi monitoring informasi data produksi bibit tanaman karet berbasis *android* digunakan untuk melakukan proses validasi data produksi oleh kepala bagian. Proses validasi dilakukan dengan memberikan status validasi berupa “Disetujui” atau “Revisi”, kemudian sistem akan menyimpan hasil validasi beserta catatan validasi ke database *firebase firestore*. Berikut adalah potongan *source code* fungsi validasi data yang digunakan dalam pengujian:

```
Future<void> validasiData({
  required BuildContext context,
  required String collection,
  required String docId,
  required String status,
}) async {
```

```
  TextEditingController
  catatanController =
    TextEditingController();
```

```
  showDialog(
    context: context,
    builder: (context) {
```

```
    return AlertDialog(
      title: Text(
        status == "Disetujui"
          ? "Setujui Data"
```



```

    ),

    ElevatedButton(
      onPressed: () async {

        await FirebaseFirestore
          .instance
          .collection(collection)
          .doc(docId)
          .update({

            "status_validasi":
              status,

            "catatan_validasi":
              catatanController
                .text
                .isEmpty
                ? "-"
                : catatanController
                  .text,
          });

        Navigator.pop(
          context,
        );
      },

      child: Text(
        status,
      ),
    ),
  ),

```

```

    ],
    );
  },
);
}

```

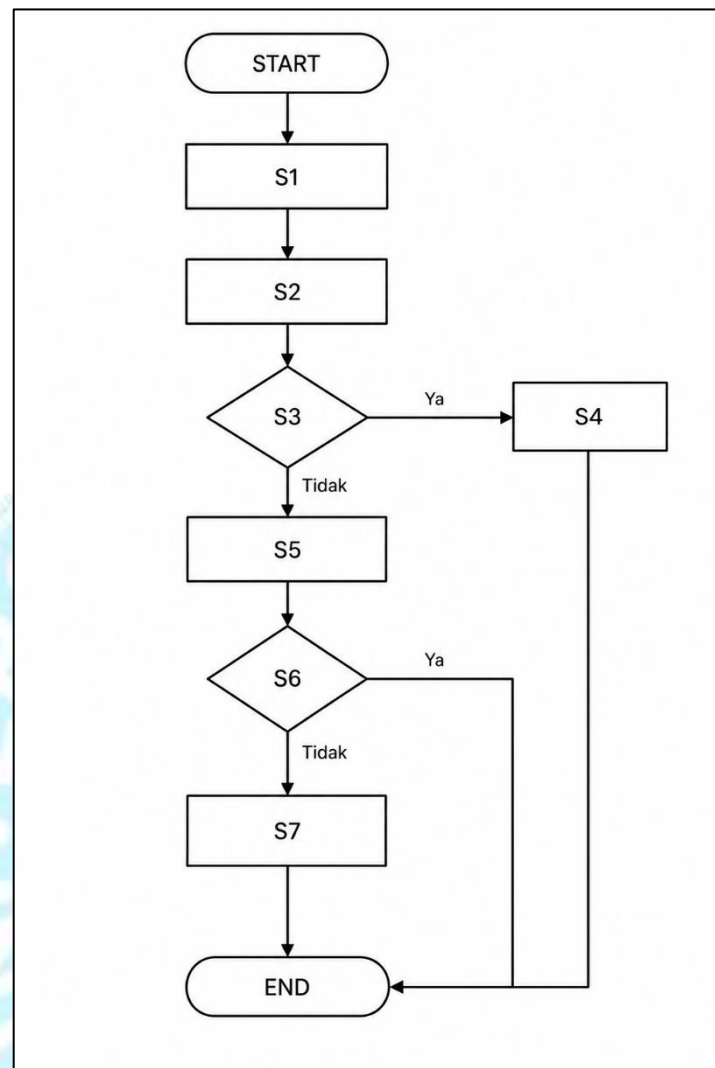
## b. Identifikasi *Statement* Fungsi Validasi Data

Tabel 5.9 Identifikasi *Statement* Fungsi Validasi Data

| Statement Source Code |                                                        | Keterangan                          |
|-----------------------|--------------------------------------------------------|-------------------------------------|
| S1                    | TextEditingController()                                | Membuat controller catatan validasi |
| S2                    | showDialog()                                           | Menampilkan dialog validasi         |
| S3                    | status == "Disetujui"                                  | Validasi status data                |
| S4                    | Navigator.pop()                                        | Menutup dialog batal                |
| S5                    | FirebaseFirestore.instance.collection().doc().update() | Memperbarui data validasi           |
| S6                    | catatanController.text.isEmpty                         | Validasi catatan kosong             |
| S7                    | Navigator.pop()                                        | Menutup dialog setelah validasi     |

Sumber : data diolah oleh penulis, 2026

c. *Flowchart Fungsi Validasi Data*



Sumber : data diolah oleh penulis, 2026

Gambar 5.41 *Flowchart Fungsi Validasi Data*

Berdasarkan diagram alur pada gambar *flowchart* Fungsi Validasi Data di atas, urutan logika kerja fungsi validasi data dapat dijabarkan sebagai berikut:

1. S1 (Membuat *Controller* Catatan Validasi)  
Sistem membuat *controller* untuk menyimpan input catatan validasi dari kepala bagian.
2. S2 (Menampilkan Dialog Validasi)  
Sistem menampilkan dialog validasi yang berisi *form* catatan validasi dan tombol aksi validasi.

3. S3 (Validasi Status Data)

Sistem melakukan pengecekan status validasi apakah data akan disetujui atau direvisi.

4. S4 (Menutup Dialog Batal)

Jika tombol batal ditekan maka sistem akan menutup dialog validasi.

5. S5 (Memperbarui Data Validasi)

Sistem memperbarui data validasi pada *firebase firestore* berdasarkan status validasi yang dipilih.

6. S6 (Validasi Catatan Kosong)

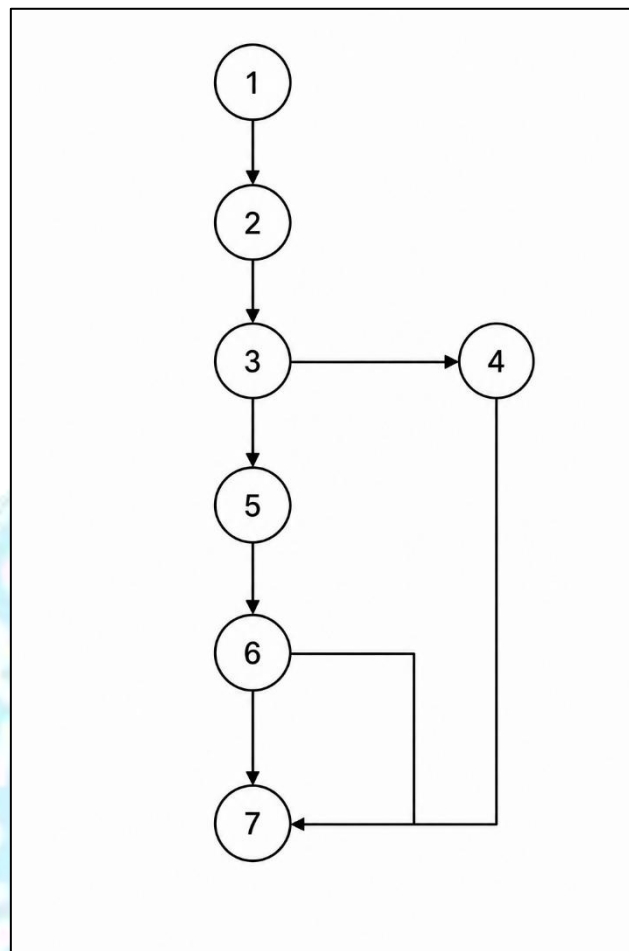
Sistem melakukan pengecekan apakah catatan validasi kosong. Jika kosong maka sistem menyimpan tanda “-”.

7. S7 (Menutup Dialog Validasi)

Setelah proses validasi selesai dilakukan, sistem menutup dialog validasi.

Berdasarkan *flowchart* tersebut dapat diketahui bahwa fungsi validasi data digunakan untuk mengelola proses persetujuan atau revisi data produksi oleh kepala bagian.

#### d. *Flowgraph* Fungsi Validasi Data



Sumber : data diolah oleh penulis, 2026

Gambar 5.42 *Flowgraph* Fungsi Validasi Data

Berdasarkan gambar flowgraph fungsi Validasi Data di atas, alur proses dimulai dari *node* 1 yang menunjukkan proses pembuatan *controller* catatan validasi menggunakan *TextEditingController()*. Selanjutnya sistem menuju *node* 2 untuk menampilkan dialog validasi menggunakan *showDialog()*. Pada *node* 3 sistem melakukan pengecekan kondisi status validasi apakah data akan disetujui atau direvisi. Jika pengguna menekan tombol batal maka sistem menuju *node* 4 untuk menutup dialog validasi.

Namun apabila proses validasi dilanjutkan, maka sistem menuju *node* 5 untuk memperbarui data validasi pada *firebase firestore*. Pada *node* 6 dilakukan pengecekan kondisi apakah catatan validasi kosong atau tidak. Jika catatan kosong maka sistem akan menyimpan simbol “-” sebagai catatan validasi. Selanjutnya sistem menuju *node* 7 untuk menutup dialog validasi dan proses validasi data

selesai dilakukan. Berdasarkan *flowgraph* tersebut dapat diketahui bahwa fungsi validasi data memiliki percabangan kondisi pada proses penentuan status validasi dan pengecekan catatan validasi sebelum data diperbarui pada *firebase firestore*.

#### e. *Cyclomatic complexity* Fungsi Validasi Data

Berdasarkan *flowgraph* yang telah dibentuk, terdapat 2 *predicate node* (P), yaitu pada *node* 3 yang melakukan pengecekan status validasi dan *node* 6 yang melakukan pengecekan catatan validasi kosong. Oleh karena itu, nilai *cyclomatic complexity* dapat dihitung menggunakan rumus berikut:

$$V(G) = P + 1$$

$$V(G) = 2 + 1$$

$$V(G) = 3$$

Dari hasil perhitungan tersebut diperoleh nilai *cyclomatic complexity* sebesar 3. Nilai tersebut menunjukkan bahwa terdapat tiga jalur independen (*independent path*) pada fungsi validasi data.

#### f. *Independent Path* Fungsi Validasi Data

Berdasarkan hasil perhitungan **cyclomatic complexity** yang bernilai 3, maka terdapat tiga **independent path** yang harus diuji pada fungsi validasi data.

##### 1. *Path* 1

$$1 \rightarrow 2 \rightarrow 3 \rightarrow 4$$

*Path* ini menunjukkan kondisi ketika pengguna menekan tombol batal sehingga dialog validasi ditutup.

##### 2. *Path* 2

$$1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 6(T) \rightarrow 7$$

*Path* ini menunjukkan kondisi ketika proses validasi dilakukan dengan catatan validasi kosong sehingga sistem menyimpan simbol “-”.

### 3. Path 3

$1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 6(F) \rightarrow 7$

*Path* ini menunjukkan kondisi ketika proses validasi dilakukan dengan catatan validasi terisi sehingga sistem menyimpan catatan validasi sesuai *input* pengguna.

## g. Test Case Pengujian Fungsi Validasi Data

Tabel 5.10 *Test Case* Fungsi Validasi Data

| Test Case | Kondisi                | Jalur Eksekusi                                                                           | Output                                           |
|-----------|------------------------|------------------------------------------------------------------------------------------|--------------------------------------------------|
| TC1       | Tombol ditekan         | batal<br>$1 \rightarrow 2 \rightarrow 3 \rightarrow 4$                                   | Dialog validasi ditutup                          |
| TC2       | Catatan kosong         | validasi<br>$1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 6(T) \rightarrow 7$ | Data berhasil divalidasi dengan catatan “_”      |
| TC3       | Catatan validasi diisi | $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 6(F) \rightarrow 7$             | Data berhasil divalidasi dengan catatan validasi |

Sumber : data diolah oleh penulis, 2026

## h. Statement Coverage Fungsi Validasi Data

Berdasarkan hasil pengujian yang telah dilakukan, seluruh *statement* pada fungsi validasi data berhasil dieksekusi sesuai dengan alur logika program. Berikut merupakan jalur *statement* yang dieksekusi pada masing-masing **test case**:

### 1. TC1

$S1 \rightarrow S2 \rightarrow S3 \rightarrow S4$

### 2. TC2

$S1 \rightarrow S2 \rightarrow S3 \rightarrow S5 \rightarrow S6(T) \rightarrow S7$

### 3. TC3

$S1 \rightarrow S2 \rightarrow S3 \rightarrow S5 \rightarrow S6(F) \rightarrow S7$

Berdasarkan hasil pengujian tersebut, seluruh *statement* pada fungsi validasi data telah berhasil dijalankan minimal satu kali selama proses pengujian berlangsung. Dengan demikian nilai **statement coverage** pada fungsi validasi data diperoleh sebesar:

$$\text{Statement Coverage} = (7 / 7) \times 100\%$$

Hasil tersebut menunjukkan bahwa seluruh *statement* pada fungsi validasi data telah diuji dan dieksekusi dengan baik sehingga fungsi validasi data dinyatakan berjalan sesuai kebutuhan sistem aplikasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android*.

#### 5.4.6. Pengujian Fungsi Realisasi Tahapan

##### a. Potongan Source Code Fungsi Realisasi Tahapan

Fungsi realisasi tahapan pada aplikasi monitoring informasi data produksi bibit tanaman karet berbasis android digunakan untuk memperbarui status tahapan produksi bibit berdasarkan realisasi kegiatan lapangan. Pada proses ini pengguna dapat memilih status kegiatan berupa “Tertunda” atau “Terlaksana”. Sistem kemudian akan memperbarui tahapan produksi, jadwal berikutnya, riwayat tahapan, serta status produksi pada database firebase firestore. Berikut adalah potongan source code fungsi realisasi tahapan yang digunakan dalam pengujian :

```
Future<void> realisasiTahapan(
    BuildContext context,
    DocumentSnapshot item,
) async {
```

```
    showDialog(
        context: context,
        builder: (context) {
```

```
        return AlertDialog(
            title: const Text(
                "Status Realisasi",
            ),
```

```
            content: const Text(
```

```
"Pilih status realisasi kegiatan",
),
```

```
actions: [
```

```
// ===== TERTUNDA =====
```

```
  TextButton(
    onPressed: () async {

      await FirebaseFirestore.instance
        .collection(
          "produksi_bibit",
        )
        .doc(item.id)
        .update({

          "status_realisasi":
            "Tertunda",

          "catatan_realisasi":
            "${item['tahapan_pemeliharaan']} ditunda",

          "status":
            "Tertunda",
        });

      Navigator.pop(
        context,
```

```

    );
  },

  child: const Text(
    "Tertunda",
  ),
),

// ===== TERLAKSANA =====

ElevatedButton(
  onPressed: () async {

    int tahapIndex =
      item['tahap_index'];

    List<dynamic> riwayat =
      item['riwayat_tahapan'];

    String tahapanSaatIni =
      item['tahapan_pemeliharaan'];

    riwayat.add(
      tahapanSaatIni,
    );

    // ===== NEXT TAHAP =====

    if (tahapIndex <

```

```
tahapanBibit.length - 1) {  
  
    int nextIndex =  
        tahapIndex + 1;  
  
    String nextTahapan =  
        tahapanBibit[nextIndex]["nama"];  
  
    int hari =  
        tahapanBibit[nextIndex]["hari"];  
  
    DateTime sekarang =  
        DateTime.now();  
  
    DateTime nextJadwal =  
        sekarang.add(  
            Duration(  
                days: hari,  
            ),  
        );  
  
    int selisihHari =  
        nextJadwal  
            .difference(  
                sekarang,  
            )  
            .inDays;  
  
    String status = "Aman";
```

```
if (selisihHari <= 7) {  
    status =  
        "Mendekati Jadwal";  
}
```

```
await FirebaseFirestore
```

```
    .instance  
    .collection(  
        "produksi_bibit",  
    )  
    .doc(item.id)  
    .update({  
  
        "tahap_index":  
            nextIndex,  
  
        "tahapan_pemeliharaan":  
            nextTahapan,  
  
        "riwayat_tahapan":  
            riwayat,  
  
        "jadwal_berikutnya":  
            Timestamp.fromDate(  
                nextJadwal,  
            ),  
  
        "status":
```

```

        status,

        "status_realisasi":
            "Terlaksana",

        "tanggal_realisasi":
            Timestamp.now(),

        "catatan_realisasi":
            "$tahapansaatini terealisasi",
    });
}

// ===== SELESAI =====

else {

    await FirebaseFirestore
        .instance
        .collection(
            "produksi_bibit",
        )
        .doc(item.id)
        .update({

            "realisasi":
                true,

            "status":

```

```

        "Selesai",

        "status_validasi":
            "Menunggu",

        "catatan_validasi":
            "-",

        "riwayat_tahapan":
            riwayat,

        "status_realisasi":
            "Terlaksana",

        "tanggal_realisasi":
            Timestamp.now(),

        "catatan_realisasi":
            "$tahapannya selesai",
    });
}

Navigator.pop(
    context,
);

},

child: const Text(
    "Terlaksana",

```

```

    ),
    ),
  ],
);
},
);
}

```

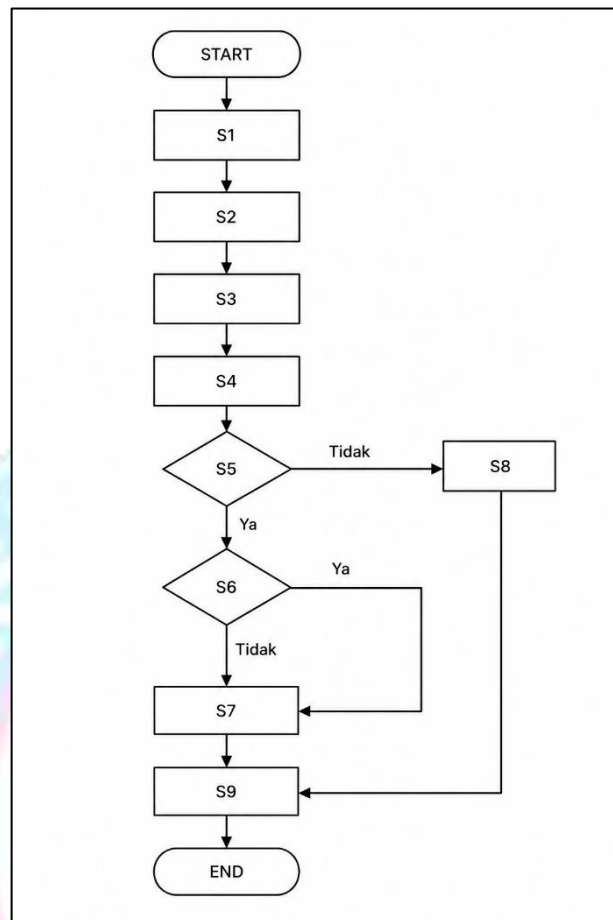
### b. Identifikasi *Statement* Fungsi Realisasi Tahapan

Tabel 5.11 Identifikasi *Statement* Fungsi Realisasi Tahapan

| Statement | Source Code                                      | Keterangan                     |
|-----------|--------------------------------------------------|--------------------------------|
| S1        | showDialog()                                     | Menampilkan dialog realisasi   |
| S2        | FirebaseFirestore.instance.collection().update() | Memperbarui status tertunda    |
| S3        | tahapIndex = item['tahap_index']                 | Mengambil indeks tahapan       |
| S4        | riwayat.add()                                    | Menambahkan riwayat tahapan    |
| S5        | if (tahapIndex < tahapanBibit.length - 1)        | Validasi tahapan berikutnya    |
| S6        | if (selisihHari <= 7)                            | Validasi status jadwal         |
| S7        | FirebaseFirestore.instance.collection().update() | Memperbarui tahapan berikutnya |
| S8        | FirebaseFirestore.instance.collection().update() | Memperbarui status selesai     |
| S9        | Navigator.pop()                                  | Menutup dialog realisasi       |

Sumber : data diolah oleh penulis, 2026

### c. *Flowchart Fungsi Realisasi Tahapan*



Sumber : data diolah oleh penulis, 2026

Gambar 5.43 *Flowchart Fungsi Realisasi Tahapan*

Berdasarkan diagram alur pada gambar *flowchart* Fungsi Realisasi Tahapan di atas, urutan logika kerja fungsi realisasi tahapan dapat dijabarkan sebagai berikut:

1. S1 (Menampilkan Dialog Realisasi)

Sistem menampilkan dialog pilihan status realisasi kegiatan produksi.

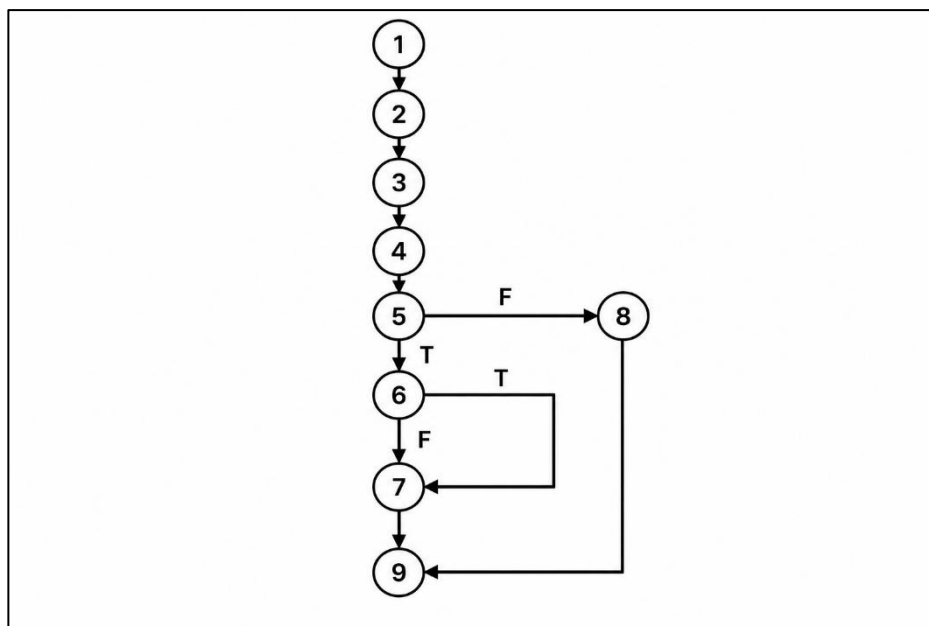
2. S2 (Memperbarui Status Tertunda)

Jika pengguna memilih status “Tertunda”, maka sistem memperbarui status produksi menjadi tertunda pada *firebase firestore*.

3. S3 (Mengambil Indeks Tahapan)  
Sistem mengambil indeks tahapan produksi saat ini dari data produksi bibit.
4. S4 (Menambahkan Riwayat Tahapan)  
Sistem menambahkan tahapan produksi saat ini ke dalam riwayat tahapan produksi.
5. S5 (Validasi Tahapan Berikutnya)  
Sistem melakukan pengecekan apakah masih terdapat tahapan produksi berikutnya.
6. S6 (Validasi Status Jadwal)  
Sistem melakukan pengecekan apakah selisih jadwal kegiatan berikutnya kurang dari atau sama dengan tujuh hari.
7. S7 (Memperbarui Tahapan Berikutnya)  
Jika masih terdapat tahapan berikutnya, maka sistem memperbarui data tahapan produksi berikutnya ke *firebase firestore*.
8. S8 (Memperbarui Status Selesai)  
Jika seluruh tahapan produksi telah selesai dilakukan, maka sistem memperbarui status produksi menjadi selesai.
9. S9 (Menutup Dialog Realisasi)  
Setelah proses realisasi selesai dilakukan, sistem menutup dialog realisasi.

Berdasarkan *flowchart* tersebut dapat diketahui bahwa fungsi realisasi tahapan digunakan untuk memperbarui progres tahapan produksi bibit berdasarkan kegiatan realisasi lapangan.

#### d. *Flowgraph* Fungsi Realisasi Tahapan



Sumber : data diolah oleh penulis, 2026

Gambar 5.44 *Flowgraph* Fungsi Realisasi Tahapan

Berdasarkan gambar *flowgraph* fungsi realisasi tahapan di atas, alur proses dimulai dari *node* 1 yang menunjukkan proses menampilkan dialog realisasi menggunakan *showDialog()*. Jika pengguna memilih status tertunda maka sistem menuju *node* 2 untuk memperbarui status tertunda pada *firebase firestore* dan dilanjutkan menuju *node* 9 untuk menutup dialog. Namun apabila pengguna memilih status terlaksana maka sistem menuju *node* 3 untuk mengambil indeks tahapan produksi saat ini. Selanjutnya sistem menuju *node* 4 untuk menambahkan tahapan produksi ke dalam riwayat tahapan.

Pada *node* 5 sistem melakukan pengecekan apakah masih terdapat tahapan produksi berikutnya. Jika kondisi bernilai *true* maka sistem menuju *node* 6 untuk melakukan pengecekan status jadwal kegiatan berikutnya. Jika selisih hari kurang dari atau sama dengan tujuh hari maka status produksi diubah menjadi “Mendekati Jadwal”. Selanjutnya sistem menuju *node* 7 untuk memperbarui data tahapan produksi berikutnya pada *firebase firestore*. Namun apabila tahapan produksi telah selesai dilakukan maka sistem menuju *node* 8 untuk memperbarui status produksi menjadi selesai.

Setelah seluruh proses selesai dilakukan sistem menuju *node* 9 untuk menutup dialog realisasi. Berdasarkan *flowgraph* tersebut dapat diketahui bahwa

fungsi realisasi tahapan memiliki beberapa percabangan kondisi untuk menentukan status tahapan produksi sebelum data diperbarui pada *firebase firestore*.

#### e. *Cyclomatic complexity* Fungsi Realisasi Tahapan

Berdasarkan *flowgraph* yang telah dibentuk, terdapat 2 *predicate node* (P), yaitu pada *node* 5 yang melakukan pengecekan tahapan berikutnya dan *node* 6 yang melakukan pengecekan status jadwal kegiatan berikutnya. Oleh karena itu, nilai *cyclomatic complexity* dapat dihitung menggunakan rumus berikut:

$$V(G) = P + 1$$

$$V(G) = 2 + 1$$

$$V(G) = 3$$

Dari hasil perhitungan tersebut diperoleh nilai *cyclomatic complexity* sebesar 3. Nilai tersebut menunjukkan bahwa terdapat tiga jalur independen (*independent path*) pada fungsi realisasi tahapan.

#### f. *Independent Path* Fungsi Realisasi Tahapan

Berdasarkan hasil perhitungan **cyclomatic complexity** yang bernilai 3, maka terdapat tiga **independent path** yang harus diuji pada fungsi realisasi tahapan.

##### 1. *Path* 1

$$1 \rightarrow 2 \rightarrow 9$$

*Path* ini menunjukkan kondisi ketika pengguna memilih status tertunda sehingga sistem memperbarui status produksi menjadi tertunda.

##### 2. *Path* 2

$$1 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6(T) \rightarrow 7 \rightarrow 9$$

*Path* ini menunjukkan kondisi ketika tahapan produksi masih berlanjut dan status jadwal berubah menjadi “Mendekati Jadwal”.

##### 3. *Path* 3

$$1 \rightarrow 3 \rightarrow 4 \rightarrow 5(F) \rightarrow 8 \rightarrow 9$$

*Path* ini menunjukkan kondisi ketika seluruh tahapan produksi telah selesai sehingga sistem memperbarui status produksi menjadi selesai.

#### g. Test Case Pengujian Fungsi Realisasi Tahapan

Tabel 5.12 *Test Case* Fungsi Realisasi Tahapan

| Test Case | Kondisi                     | Jalur Eksekusi               | Output                                      |
|-----------|-----------------------------|------------------------------|---------------------------------------------|
| TC1       | Status dipilih tertunda     | 1 → 2 → 9                    | Status produksi berubah menjadi tertunda    |
| TC2       | Tahapan berikutnya tersedia | 1 → 3 → 4 → 5 → 6(T) → 7 → 9 | Data tahapan berikutnya berhasil diperbarui |
| TC3       | Tahapan produksi selesai    | 1 → 3 → 4 → 5(F) → 8 → 9     | Status produksi berubah menjadi selesai     |

Sumber : data diolah oleh penulis, 2026

#### h. Statement Coverage Fungsi Realisasi Tahapan

Berdasarkan hasil pengujian yang telah dilakukan, seluruh *statement* pada fungsi realisasi tahapan berhasil dieksekusi sesuai dengan alur logika program. Berikut merupakan jalur *statement* yang dieksekusi pada masing-masing **test case**:

1. TC1

S1 → S2 → S9

2. TC2

S1 → S3 → S4 → S5 → S6(T) → S7 → S9

3. TC3

S1 → S3 → S4 → S5(F) → S8 → S9

Berdasarkan hasil pengujian tersebut, seluruh *statement* pada fungsi realisasi tahapan telah berhasil dijalankan minimal satu kali selama proses pengujian berlangsung. Dengan demikian nilai **statement coverage** pada fungsi realisasi tahapan diperoleh sebesar:

$$\text{Statement Coverage} = (9 / 9) \times 100\%$$

Hasil tersebut menunjukkan bahwa seluruh *statement* pada fungsi realisasi tahapan telah diuji dan dieksekusi dengan baik sehingga fungsi realisasi tahapan dinyatakan berjalan sesuai kebutuhan sistem aplikasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android*.

#### 5.4.7. Pengujian Fungsi *Monitoring* Produksi

##### a. Potongan *source code* Fungsi *Monitoring* Produksi

Fungsi *monitoring produksi* pada aplikasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android* digunakan untuk menampilkan data produksi tanaman secara realtime dari *firebase firestore*. Sistem akan melakukan pengecekan kondisi koneksi, ketersediaan data, serta menentukan warna status berdasarkan kondisi produksi tanaman. Berikut adalah potongan *source code* fungsi *monitoring* produksi yang digunakan dalam pengujian:

```
class MonitoringTanaman extends StatelessWidget {
  const MonitoringTanaman({
    super.key,
  });

  @override
  Widget build(BuildContext context) {

    return StreamBuilder<QuerySnapshot> (
      stream: FirebaseFirestore.instance
        .collection(
          "produksi_tanaman",
        )
        .snapshots(),

      builder: (context, snapshot) {
```

```
// ===== LOADING =====

if (snapshot.connectionState ==
    ConnectionState.waiting) {

    return const Center(
      child:
        CircularProgressIndicator(),
    );
}

// ===== DATA KOSONG =====

if (!snapshot.hasData ||
    snapshot.data!.docs.isEmpty) {

    return const Center(
      child: Text(
        "Belum ada data tanaman",
      ),
    );
}

var data = snapshot.data!.docs;

return ListView.builder(
  itemCount: data.length,

  itemBuilder: (context, index) {
```

```
var item = data[index];

String status =
    item['status'];

Color warnaStatus =
    Colors.green;

// ===== TERTUNDA =====

if (status == "Tertunda") {
    warnaStatus =
        Colors.orange;
}

// ===== MENDEKATI =====

if (status ==
    "Mendekati Jadwal") {

    warnaStatus =
        Colors.red;
}

// ===== SELESAI =====

if (status == "Selesai") {
    warnaStatus =
```

```

        Colors.blue;
    }

    return Container(
      child: Text(
        status,
      ),
    );
  },
);
},
);
}
}
}

```

### b. Identifikasi *Statement* Fungsi *Monitoring* Produksi

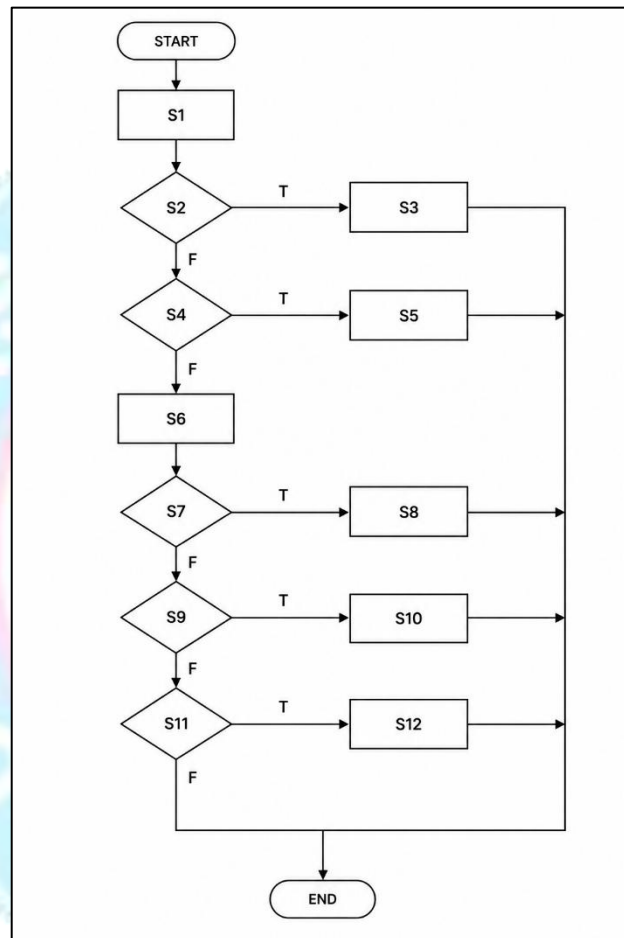
Tabel 5.13 Identifikasi *Statement* Fungsi *Monitoring* Produksi

| Statement | Source Code                                          | Keterangan                       |
|-----------|------------------------------------------------------|----------------------------------|
| S1        | StreamBuilder()                                      | Menampilkan data realtime        |
| S2        | if(snapshot.connectionState == waiting)              | Validasi loading data            |
| S3        | CircularProgressIndicator()                          | Menampilkan loading              |
| S4        | if(!snapshot.hasData<br>snapshot.data!.docs.isEmpty) | Validasi data kosong             |
| S5        | Text("Belum ada data tanaman")                       | Menampilkan pesan data kosong    |
| S6        | ListView.builder()                                   | Menampilkan daftar data          |
| S7        | if(status == "Tertunda")                             | Validasi status tertunda         |
| S8        | warnaStatus = Colors.orange                          | Mengubah warna tertunda          |
| S9        | if(status == "Mendekati Jadwal")                     | Validasi status mendekati jadwal |
| S10       | warnaStatus = Colors.red                             | Mengubah warna mendekati jadwal  |

| Statement | Source Code               | Keterangan              |
|-----------|---------------------------|-------------------------|
| S11       | if(status == "Selesai")   | Validasi status selesai |
| S12       | warnaStatus = Colors.blue | Mengubah warna selesai  |

Sumber : data diolah oleh penulis, 2026

### c. Flowchart Fungsi Monitoring Produksi



Sumber : data diolah oleh penulis, 2026

Gambar 5.45 Flowchart Fungsi Monitoring Produksi

Berdasarkan diagram alur pada gambar *flowchart* fungsi *monitoring* produksi di atas, urutan logika kerja fungsi *monitoring* produksi dapat dijabarkan sebagai berikut:

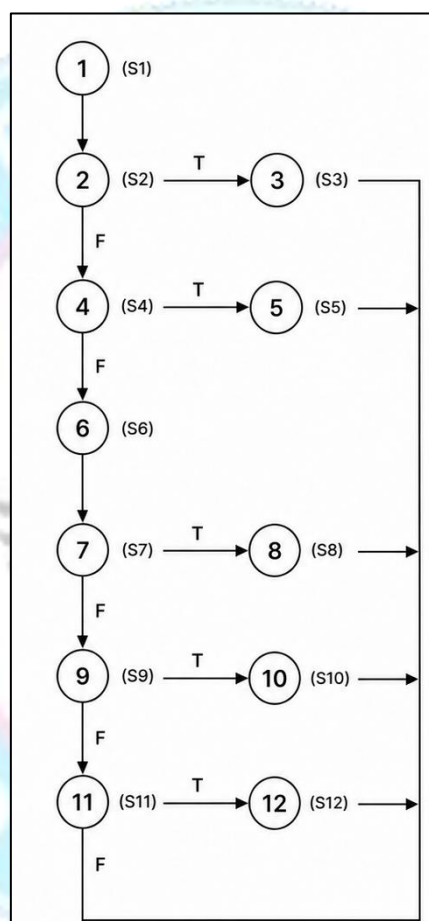
1. S1 (Menampilkan Data *Realtime*)  
Sistem mengambil data produksi tanaman secara *realtime* dari *firebase firestore* menggunakan *StreamBuilder()*.
2. S2 (Validasi *Loading* Data)  
Sistem melakukan pengecekan apakah data masih dalam proses *loading*.
3. S3 (Menampilkan *Loading*)  
Jika data masih *loading* maka sistem menampilkan *CircularProgressIndicator()*.
4. S4 (Validasi Data Kosong)  
Sistem melakukan pengecekan apakah data produksi tersedia atau kosong.
5. S5 (Menampilkan Pesan Data Kosong)  
Jika data kosong maka sistem menampilkan pesan “Belum ada data tanaman”.
6. S6 (Menampilkan Daftar Data)  
Jika data tersedia maka sistem menampilkan daftar data produksi tanaman menggunakan *ListView.builder()*.
7. S7 (Validasi Status Tertunda)  
Sistem melakukan pengecekan apakah status produksi bernilai “Tertunda”.
8. S8 (Mengubah Warna Tertunda)  
Jika status tertunda maka warna status diubah menjadi oranye.
9. S9 (Validasi Status Mendekati Jadwal)  
Sistem melakukan pengecekan apakah status produksi bernilai “Mendekati Jadwal”.
10. S10 (Mengubah Warna Mendekati Jadwal)  
Jika status mendekati jadwal maka warna status diubah menjadi merah.
11. S11 (Validasi Status Selesai)  
Sistem melakukan pengecekan apakah status produksi bernilai “Selesai”.

## 12. S12 (Mengubah Warna Selesai)

Jika status selesai maka warna status diubah menjadi biru.

Berdasarkan *flowchart* tersebut dapat diketahui bahwa fungsi *monitoring* produksi digunakan untuk memantau status produksi tanaman secara *realtime* berdasarkan kondisi data pada *firebase firestore*.

### d. Flowgraph Fungsi Monitoring Produksi



Sumber : data diolah oleh penulis, 2026

Gambar 5.46 Flowgraph Fungsi Monitoring Produksi

Berdasarkan gambar *flowgraph* fungsi *monitoring* produksi di atas, alur proses dimulai dari *node* 1 yang menunjukkan proses pengambilan data *realtime* menggunakan *StreamBuilder()*. Selanjutnya sistem menuju *node* 2 untuk melakukan pengecekan apakah data masih loading. Jika kondisi bernilai *true* maka sistem menuju *node* 3 untuk menampilkan *loading indicator*. Namun apabila data berhasil dimuat maka sistem menuju *node* 4 untuk melakukan pengecekan apakah

data kosong. Jika data kosong maka sistem menuju *node* 5 untuk menampilkan pesan bahwa data tanaman belum tersedia. Namun apabila data tersedia maka sistem menuju *node* 6 untuk menampilkan daftar data produksi tanaman.

Pada *node* 7 sistem melakukan pengecekan apakah status produksi bernilai “Tertunda”. Jika *true* maka sistem menuju *node* 8 untuk mengubah warna status menjadi oranye. Selanjutnya sistem menuju *node* 9 untuk melakukan pengecekan apakah status produksi bernilai “Mendekati Jadwal”. Jika *true* maka sistem menuju *node* 10 untuk mengubah warna status menjadi merah. Kemudian sistem menuju *node* 11 untuk melakukan pengecekan apakah status produksi bernilai “Selesai”. Jika *true* maka sistem menuju *node* 12 untuk mengubah warna status menjadi biru.

Berdasarkan *flowgraph* tersebut dapat diketahui bahwa fungsi *monitoring* produksi memiliki beberapa percabangan kondisi untuk menentukan tampilan status produksi tanaman pada aplikasi *monitoring*.

#### **e. Cyclomatic complexity Fungsi Monitoring Produksi**

Berdasarkan *flowgraph* yang telah dibentuk, terdapat 5 *predicate node* (P), *node* 2 melakukan pengecekan loading data. *Node* 4 melakukan pengecekan data kosong. *Node* 7 melakukan pengecekan status tertunda. *Node* 9 melakukan pengecekan status mendekati jadwal. *Node* 11 melakukan pengecekan status selesai

Oleh karena itu, nilai *cyclomatic complexity* dapat dihitung menggunakan rumus berikut:

$$V(G) = P + 1$$

$$V(G) = 5 + 1$$

$$V(G) = 6$$

Dari hasil perhitungan tersebut diperoleh nilai *cyclomatic complexity* sebesar 6. Nilai tersebut menunjukkan bahwa terdapat enam jalur independen (*independent path*) pada fungsi *monitoring* produksi.

#### f. *Independent Path* Fungsi *Monitoring* Produksi

Berdasarkan hasil perhitungan **cyclomatic complexity** yang bernilai 6, maka terdapat enam **independent path** yang harus diuji pada fungsi *monitoring* produksi.

1. *Path* 1

$1 \rightarrow 2(T) \rightarrow 3$

*Path* ini menunjukkan kondisi ketika data masih dalam proses loading sehingga sistem menampilkan *loading indicator*.

2. *Path* 2

$1 \rightarrow 2(F) \rightarrow 4(T) \rightarrow 5$

*Path* ini menunjukkan kondisi ketika data produksi kosong sehingga sistem menampilkan pesan data kosong.

3. *Path* 3

$1 \rightarrow 2(F) \rightarrow 4(F) \rightarrow 6 \rightarrow 7(T) \rightarrow 8$

*Path* ini menunjukkan kondisi ketika status produksi bernilai “Tertunda” sehingga warna status berubah menjadi oranye.

4. *Path* 4

$1 \rightarrow 2(F) \rightarrow 4(F) \rightarrow 6 \rightarrow 7(F) \rightarrow 9(T) \rightarrow 10$

*Path* ini menunjukkan kondisi ketika status produksi bernilai “Mendekati Jadwal” sehingga warna status berubah menjadi merah.

5. *Path* 5

$1 \rightarrow 2(F) \rightarrow 4(F) \rightarrow 6 \rightarrow 7(F) \rightarrow 9(F) \rightarrow 11(T) \rightarrow 12$

*Path* ini menunjukkan kondisi ketika status produksi bernilai “Selesai” sehingga warna status berubah menjadi biru.

6. *Path* 6

$1 \rightarrow 2(F) \rightarrow 4(F) \rightarrow 6 \rightarrow 7(F) \rightarrow 9(F) \rightarrow 11(F)$

*Path* ini menunjukkan kondisi ketika status produksi normal sehingga warna status tetap hijau.

### g. Test Case Pengujian Fungsi Monitoring Produksi

Tabel 5.14 Test Case Fungsi Monitoring Produksi

| Test Case | Kondisi                 | Jalur Eksekusi                                 | Output                      |
|-----------|-------------------------|------------------------------------------------|-----------------------------|
| TC1       | Data loading            | 1 → 2(T) → 3                                   | Loading indicator tampil    |
| TC2       | Data kosong             | 1 → 2(F) → 4(T) → 5                            | Pesan data kosong tampil    |
| TC3       | Status tertunda         | 1 → 2(F) → 4(F) → 6 → 7(T) → 8                 | Warna status menjadi oranye |
| TC4       | Status mendekati jadwal | 1 → 2(F) → 4(F) → 6 → 7(F) → 9(T) → 10         | Warna status menjadi merah  |
| TC5       | Status selesai          | 1 → 2(F) → 4(F) → 6 → 7(F) → 9(F) → 11(T) → 12 | Warna status menjadi biru   |
| TC6       | Status aman             | 1 → 2(F) → 4(F) → 6 → 7(F) → 9(F) → 11(F)      | Warna status tetap hijau    |

Sumber : data diolah oleh penulis, 2026

### h. Statement Coverage Fungsi Monitoring Produksi

Berdasarkan hasil pengujian yang telah dilakukan, seluruh *statement* pada fungsi *monitoring* produksi berhasil dieksekusi sesuai dengan alur logika program.

1. TC1

S1 → S2(T) → S3

2. TC2

S1 → S2(F) → S4(T) → S5

3. TC3

S1 → S2(F) → S4(F) → S6 → S7(T) → S8

4. TC4

S1 → S2(F) → S4(F) → S6 → S7(F) → S9(T) → S10

## 5. TC5

S1 → S2(F) → S4(F) → S6 → S7(F) → S9(F) → S11(T) → S12

## 6. TC6

S1 → S2(F) → S4(F) → S6 → S7(F) → S9(F) → S11(F)

Berdasarkan hasil pengujian tersebut, seluruh *statement* pada fungsi *monitoring* produksi telah berhasil dijalankan minimal satu kali selama proses pengujian berlangsung. Dengan demikian nilai **statement coverage** pada fungsi *monitoring* produksi diperoleh sebesar:

$$\text{Statement Coverage} = (12 / 12) \times 100\%$$

Hasil tersebut menunjukkan bahwa seluruh *statement* pada fungsi *monitoring* produksi telah diuji dan dieksekusi dengan baik sehingga fungsi *monitoring* produksi dinyatakan berjalan sesuai kebutuhan sistem aplikasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android*.

#### 5.4.8. Pengujian Fungsi Hapus Data

##### a. Potongan *Source Code* Fungsi Hapus Data

Fungsi hapus data pada aplikasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android* digunakan untuk menghapus data produksi yang tersimpan pada *firebase firestore*. Sebelum proses penghapusan dilakukan, sistem akan menampilkan dialog konfirmasi untuk memastikan pengguna benar-benar ingin menghapus data tersebut. Berikut adalah potongan *source code* fungsi hapus data yang digunakan dalam pengujian:

```
Future<void> hapusData(
    BuildContext context,
    String collection,
    String docId,
) async {

    showDialog(
```

```

context: context,
builder: (context) {

return AlertDialog(
  title: const Text(
    "Hapus Data",
  ),

  content: const Text(
    "Apakah yakin ingin menghapus data?",
  ),
  actions: [
    TextButton(
      onPressed: () {
        Navigator.pop(
          context,
        );
      },
      child: const Text(
        "Batal",
      ),
    ),
    ElevatedButton(
      onPressed: () async {
        await FirebaseFirestore
          .instance
          .collection(collection)
          .doc(docId)

```

```

        .delete();
    Navigator.pop(
        context,
    );
},
child: const Text(
    "Hapus",
),
),
],
);
},
);
}

```

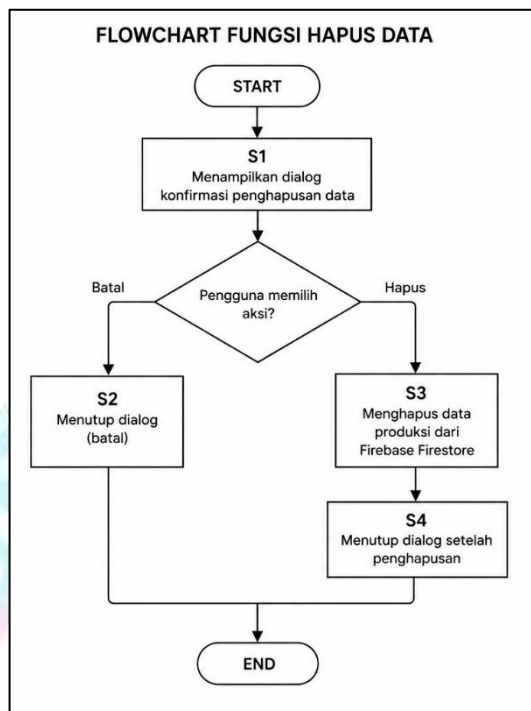
#### b. Identifikasi *Statement* Fungsi Hapus Data

Tabel 5.15 Identifikasi *Statement* Fungsi Hapus Data

| Statement | Source Code                                    | Keterangan                         |
|-----------|------------------------------------------------|------------------------------------|
| S1        | showDialog()                                   | Menampilkan dialog konfirmasi      |
| S2        | Navigator.pop()                                | Menutup dialog batal               |
| S3        | Firestore.instance.collection().doc().delete() | Menghapus data produksi            |
| S4        | Navigator.pop()                                | Menutup dialog setelah penghapusan |

Sumber : data diolah oleh penulis, 2026

### c. Flowchart Fungsi Hapus Data



Sumber : data diolah oleh penulis, 2026

Gambar 5.47 Flowchart Fungsi Hapus Data

Berdasarkan diagram alur pada gambar *flowchart* fungsi hapus data di atas, urutan logika kerja fungsi hapus data dapat dijabarkan sebagai berikut:

1. S1 (Menampilkan Dialog Konfirmasi)
 

Sistem menampilkan dialog konfirmasi penghapusan data.
2. S2 (Menutup Dialog Batal)
 

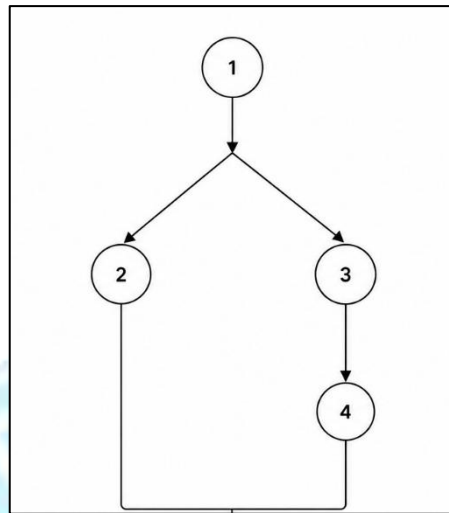
Jika pengguna memilih tombol batal maka sistem menutup dialog konfirmasi.
3. S3 (Menghapus Data Produksi)
 

Jika pengguna memilih tombol hapus maka sistem menghapus data produksi dari *firebase firestore*.
4. S4 (Menutup Dialog Penghapusan)
 

Setelah proses penghapusan selesai dilakukan, sistem menutup dialog konfirmasi.

Berdasarkan *flowchart* tersebut dapat diketahui bahwa fungsi hapus data digunakan untuk menghapus data produksi berdasarkan konfirmasi pengguna.

#### d. *Flowgraph* Fungsi Hapus Data



Sumber : data diolah oleh penulis, 2026

Gambar 5.48 *Flowgraph* Fungsi Hapus Data

Berdasarkan gambar *flowgraph* fungsi hapus data di atas, alur proses dimulai dari *node* 1 yang menunjukkan proses menampilkan dialog konfirmasi penghapusan data menggunakan *showDialog()*. Selanjutnya sistem melakukan percabangan kondisi berdasarkan pilihan pengguna. Jika pengguna memilih tombol batal maka sistem menuju *node* 2 untuk menutup dialog konfirmasi.

Namun apabila pengguna memilih tombol hapus maka sistem menuju *node* 3 untuk menghapus data produksi dari *firebase firestore*. Setelah proses penghapusan selesai dilakukan, sistem menuju *node* 4 untuk menutup dialog konfirmasi. Berdasarkan *flowgraph* tersebut dapat diketahui bahwa fungsi hapus data memiliki percabangan kondisi berdasarkan aksi pengguna pada dialog konfirmasi penghapusan data.

#### e. *Cyclomatic Complexity* Fungsi Hapus Data

Berdasarkan *flowgraph* yang telah dibentuk, terdapat 1 *predicate node* (P), yaitu pada *node* percabangan pilihan pengguna antara tombol batal dan tombol hapus. Oleh karena itu, nilai *cyclomatic complexity* dapat dihitung menggunakan rumus berikut:

$$V(G) = P + 1$$

$$V(G) = 1 + 1$$

$$V(G) = 2$$

Dari hasil perhitungan tersebut diperoleh nilai *cyclomatic complexity* sebesar 2. Nilai tersebut menunjukkan bahwa terdapat dua *independent path* pada fungsi hapus data.

#### f. Independent Path Fungsi Hapus Data

Berdasarkan hasil perhitungan **cyclomatic complexity** yang bernilai 2, maka terdapat dua **independent path** yang harus diuji pada fungsi hapus data.

1. *Path 1*

$$1 \rightarrow 2$$

*Path* ini menunjukkan kondisi ketika pengguna memilih tombol batal sehingga dialog konfirmasi ditutup.

2. *Path 2*

$$1 \rightarrow 3 \rightarrow 4$$

*Path* ini menunjukkan kondisi ketika pengguna memilih tombol hapus sehingga data produksi berhasil dihapus dari *firebase firestore*.

#### g. Test Case Pengujian Fungsi Hapus Data

Tabel 5.16 *Test Case* Fungsi Hapus Data

| Test Case | Kondisi              | Jalur Eksekusi | Output                    |
|-----------|----------------------|----------------|---------------------------|
| TC1       | Tombol batal ditekan | 1 → 2          | Dialog konfirmasi ditutup |
| TC2       | Tombol hapus ditekan | 1 → 3 → 4      | Data berhasil dihapus     |

Sumber : data diolah oleh penulis, 2026

#### h. Statement Coverage Fungsi Hapus Data

Berdasarkan hasil pengujian yang telah dilakukan, seluruh *statement* pada fungsi hapus data berhasil dieksekusi sesuai dengan alur logika program.

1. TC1

$$S1 \rightarrow S2$$

## 2. TC2

S1 → S3 → S4

Berdasarkan hasil pengujian tersebut, seluruh *statement* pada fungsi hapus data telah berhasil dijalankan minimal satu kali selama proses pengujian berlangsung. Dengan demikian nilai **statement coverage** pada fungsi hapus data diperoleh sebesar:

$$\text{Statement Coverage} = (4 / 4) \times 100\%$$

Hasil tersebut menunjukkan bahwa seluruh *statement* pada fungsi hapus data telah diuji dan dieksekusi dengan baik sehingga fungsi hapus data dinyatakan berjalan sesuai kebutuhan sistem aplikasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android*.



## BAB VI

### KESIMPULAN DAN SARAN

#### 6.1 Kesimpulan

Berdasarkan hasil penelitian mengenai digitalisasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android* pada PT. Kendi Arindo, maka dapat disimpulkan sebagai berikut:

1. Kondisi Sistem *Monitoring* dan Pencatatan Data Produksi

Berdasarkan hasil observasi dan wawancara, sistem *monitoring* dan pencatatan data produksi bibit tanaman karet di PT. Kendi Arindo sebelumnya masih dilakukan secara manual. Kondisi tersebut menimbulkan beberapa kendala, seperti keterlambatan penyampaian informasi, kesalahan pencatatan data, kesulitan dalam pencarian data, serta kurang efektifnya proses *monitoring* dan pelaporan data produksi.

2. Perancangan Sistem Digitalisasi *Monitoring* Berbasis *Android*

Penelitian ini berhasil merancang dan membangun aplikasi digitalisasi *monitoring* informasi data produksi bibit tanaman karet berbasis *android* menggunakan flutter dan *firebase*. Aplikasi yang dikembangkan memiliki fitur *login*, *input* data produksi, validasi data, *monitoring* produksi, realisasi tahapan produksi, dan hapus data. Sistem yang dibangun mampu mendukung proses pengelolaan data produksi secara lebih terstruktur dan terintegrasi.

3. Analisis Kebutuhan Sistem Aplikasi Berbasis *Android*

Berdasarkan hasil analisis kebutuhan dan pengujian menggunakan metode *white box testing*, aplikasi yang dikembangkan telah mampu memenuhi kebutuhan pengguna dalam proses pencatatan, pemantauan, dan pelaporan data produksi bibit tanaman karet. Seluruh fungsi utama aplikasi dapat berjalan dengan baik sesuai dengan kebutuhan operasional PT. Kendi Arindo sehingga dapat membantu meningkatkan efektivitas dan efisiensi pengelolaan informasi produksi.

## 6.2 Saran

Berdasarkan hasil penelitian yang telah dilakukan, maka saran yang dapat diberikan adalah sebagai berikut:

1. PT. Kendi Arindo disarankan untuk menerapkan aplikasi ini secara berkelanjutan guna mendukung proses *monitoring* dan pengelolaan data produksi bibit tanaman karet secara lebih efektif dan efisien.
2. Perlu dilakukan pemeliharaan dan pengembangan sistem secara berkala agar aplikasi tetap berjalan dengan baik serta dapat menyesuaikan kebutuhan operasional perusahaan.
3. Penelitian selanjutnya dapat menambahkan fitur laporan otomatis, pencarian data, dan penyaringan data untuk meningkatkan kemudahan dalam pengelolaan informasi produksi.



## DAFTAR PUSTAKA

- Alqamari, M. (2025). Peran Perkebunan Kelapa Sawit Dalam Menghadapi Sustainable Development Goals ( SDGS). *Jurnal Sosial Humaniora Komunikasi*, 6(1), 191–200. <https://doi.org/10.53695/js.v6i1.1352>
- Ananda, M. R., Decency, H. A., Azhari, A. N., Dzikra, M., Chandra, A. H. S., Apriliani, F., & Vokasi, S. (2025). Penerapan Total Quality Management ( TQM ) dalam Menjaga Kualitas Bibit Tanaman dan Buah di Persada Farm Bogor. *Jurnal Pengabdian Masyarakat: Pemberdayaan, Inovasi Dan Perubahan*, 5(4), 1–6. <https://doi.org/10.59818/jpm.v5i4.1707>
- Andrianto, R., & Munandar, M. H. (2022). Aplikasi E-Commerce Penjualan Pakaian Berbasis. *Journal Computer Science and Information Technology(JCoInT)*, 3(1), 20–29. <https://jurnal.ulb.ac.id/index.php/JCoInT/article/view/2478>
- Ansori, S., & Hendradi, P. (2023). Penerapan Metode Design Thinking dalam Perancangan UI / UX Aplikasi Mobile sipropmawa. *Journal of Information System Research*, 4(4), 3–10. <https://doi.org/10.47065/josh.v4i4.3648>
- Apriadi. (2022). Sistem Monitoring Pertumbuhan Spike Pada Bibit Tanaman Anggrek Berbasis Pengolahan Citra. *Journal Computer Science and Information Technology(JCoInT)*, 2(2), 8–22. <https://elibrary.unikom.ac.id/id/eprint/6336/>
- Aprilia Lusiana, P. (2024). Transformasi Akuntansi Di Era 5.0 : Analisis Pengaruh Teknologi, Keterlibatan Artificial Intelligent, Dan Digitalisasi Terhadap Laporan Keuangan. *Determinasi: Jurnal Penelitian Ekonomi Manajemen Dan Akuntansi*, 2(2), 9–18. <https://doi.org/10.23917/determinasi.v2i2.216>
- Ar Rizqi, M. F., Prihandani, K., & Voutama, A. (2023). Implementasi Design Thinking Dalam Perancangan Ui/Ux Aplikasi Kesehatan Berbasis Mobile (Studi Kasus: Pt. Sintasi). *Jurnal Ilmiah Wahana Pendidikan*, 9(16), 555. <https://doi.org/10.5281/zenodo.8251429>
- Arujisaputra, E. T. (2025). Penerapan Sistem Informasi untuk Meningkatkan Efisiensi Operasional dan Pengambilan Keputusan di Perusahaan. *Journal Scientific of Mandalika (Jsm) e-ISSN*, 6(3), 2809–0543. <https://doi.org/10.36312/10.36312/vol6iss3pp700-709>
- Arwini, N. P. D. (2021). Roti, Pemilihan Bahan Dan Proses Pembuatan. *Jurnal Ilmiah Vastuwidya*, 4(1), 33–40. <https://doi.org/10.47532/jiv.v4i1.249>
- Aziz, A., & Ernawati, I. (2023). Pengembangan Aplikasi *Android* Untuk Peningkatkan Efisiensi Operasi Pendataan Pengamatan Bunga Tumbuhan. *Jurnal informatik*, 1(April), 28–36. <https://doi.org/10.52958/iftk.v19i1.4714>
- Bainar. (2023). Peluang dan Tantangan Artificial Intelligence Bagi Guru Pendidikan Agama Islam. *Jurnal Ilmiah Keislaman*, 2(2), 78. <https://doi.org/10.47435/al->

- Budiarti, R., & Juleha, M. (2025). Sistem Informasi Pengelolaan Data Operasional Kegiatan Media Berita Pada Lembaga Radio Republik Indonesia (Rri) Jambi Berbasis Web. *FORTECH (Journal of Information Technology)*, 9(1), 9–13. <https://doi.org/10.53564/fortech.v9i1.1516>
- Budiyanto, A. (2023). Perancangan Aplikasi Pembukuan Keuangan Warung Sembako Jakarta Timur Berbasis Manajemen Keuangan dengan *Android*. *Jurnal Esensi Infokom : Jurnal Esensi Sistem Informasi Dan Sistem Komputer*, 7(1), 90–94. <https://doi.org/10.55886/infokom.v7i1.650>
- Darnita, Y., Toyib, R., & Rumbiyani, R. (2022). Penentuan Penerima Bibit Pertanian Di Kabupaten Seluma Dengan Metode Weighthed Product. *Jurnal Pseudocode*, 9(2), 61–72. <https://doi.org/10.33369/pseudocode.9.2.61-72>
- Daruhadi, G., & Sopiati, P. (2024). Metode Pengumpulan Data Penelitian. *Jurnal Cendekia Ilmiah*, 3(5), 5423–5443. <https://doi.org/10.56799/jceki.v3i5.5181>
- Daulay, Z. K., Santoso, H., Islam, U., & Sumatera, N. (2024). Penerapan sistem informasi monitoring hasil panen dan. *Journal of Science and Social Research*, 4307(3), 980–986. <https://doi.org/10.54314/jssr.v7i3.2096>
- Dhae, Y. K. I. D. D. (2021). *Dhae/ Journal of management (SME's), Vol.14, No.1, 2021,p87-103.14(1),87–103.* <https://ejurnal.undana.ac.id/index.php/JEM/article/view/3815>
- Dr.Arie Gunawan, S.Kom., M. M. S. I. (2024). *Mobile Programming Menggunakan Flutter dan Visual Studio Code Untuk Pemula.* [www.penerbitlitnus.co.id](http://www.penerbitlitnus.co.id)
- Dua, I. L., & Rumerung, J. J. (2022). Karyawan bidang administrasi pada pt . Manado. *JurnalMABP*, 4(1), 118–132. <http://download.garuda.kemdikbud.go.id>
- Faithullah Akbar, M. (2023). Penerapan Metode Waterfall pada Sistem Informasi Penjualan Dan Persediaan Pada Warung Makan Hejo Karawang. *Indonesian JournalComputerScience*, 2(1), 29–34. <https://doi.org/10.31294/ijcs.v2i1.1902>
- Fitriani, F., & Apridiansyah, Y. (2021). Aplikasi Antrian Pembayaran Uang Kuliah Berbasis *Android* Menggunakan Algoritma Fifo Di Universitas Muhammadiyah Bengkulu. *JUSIBI (Jurnal Sistem Informasi Dan E-Bisnis)*, 3(2), 91–103. <https://doi.org/10.54650/jusibi.v3i2.384>
- Fuada, S., Setyowati, E., Riani, D. W., & Aulia, G. I. (2023). Narative review pemanfaatan internet-of-things untuk aplikasi seed monitoring and management system pada media tanaman. *Infotech Journal*, 9(1), 40–45. <https://www.academia.edu/download/110359214/2461.pdf>
- Ganesh Lindung Nusantara, Rian Andrian, & Nuur Wachid Abdulmajid. (2025). Implementation of a Web-Based Student and Teacher Attendance System With QR Code Integration using the RAD. *Jurnal INOVTEK Polbeng - Seri Informatika*, 10(1), 99–110. <https://doi.org/10.35314/f2qvfs64>

- Habsy, B. A., Mufidha, N., Shelomita, C., Rahayu, I., & Muckorobin, M. I. (2023). Filsafat Dasar dalam Konseling Psikoanalisis : Studi Literatur. *Indonesian Journal of Educational Counseling*, 7(2), 189–199.  
<https://doi.org/10.30653/001.202372.266>
- Haris, A. R., & Sitanggang, D. P. (2023). *Sistem penunjang keputusan (spk) pemilihan supplier terbaik dengan metode moora*. 79–84.
- Hasibuan, M. P., & Azmi, R. (2023). Analisis Pengukuran Temperatur Udara Dengan Metode Observasi. *GABDIMAS: Jurnal Garuda Pengabdian Kepada Masyarakat*, 1(1), 2–8.  
<https://journal.aira.or.id/index.php/gabdimas/article/view/582>
- Hidayatullah, H. (2023). Strategi Pembelajaran Pendidikan Agama Islam Era Digitalisasi Di Smp Sultan Agung Seyegan Sleman Yogyakarta. *ADDABANA: Jurnal Pendidikan Agama Islam*, 6(2), 119–133.  
<https://doi.org/10.47732/adb.v6i2.249>
- Iqbal Hasyim, Ziadatu Zzulfa, & Ardila Riski Lukmana. (2023). Digitalisasi Ekonomi Umkm Berbasis Kemasyarakatan. *Seminar Nasional Pariwisata DanKewirausahaan(SNPK)*, 2, 217–224.  
<https://doi.org/10.36441/snpk.vol2.2023.12>
- Irawati, S. Y. H. (2021). Implementasi Metode Penugasan untuk Meningkatkan Kemampuan Melakukan Wawancara untuk Mahasiswa STAB Maitreyawira pada Mata Kuliah Bahasa Indonesia. *Jurnal Maitreyawira*, 2(2), 167–186.  
<https://doi.org/10.69607/jm.v2i2.40>
- Johan Daniel, Syamsul Maarif .M, & Zulbainarni Nimmi. (2022). Persepsi petani terhadap digitalisasi pertanian untuk. *Jurnal Aplikasi Bisnis Dan Manajemen (JABM)*, 8(1), 203–216.  
<https://journal.ipb.ac.id/index.php/jabm/article/view/39901>
- Kamil, M., & Lestari, P. (2023). Implementasi Framework Flutter Pada Rancang Bangun Aplikasi Konsultasi Dokter Hewan. *Jurnal Buletin Sistem Informasi DanTeknologiIslam(BUSITI)*, 4(4), 296–305.  
<https://jurnal.fikom.umi.ac.id/index.php/BUSITI/article/view/1674>
- Khairat, U., Basri, B., & Fakhurrozi, W. A. (2022). Monitoring Suhu Ruang Budidaya Jamur Tiram Menggunakan *Android* Berbasis Arduino. *Technomedia Journal*, 7(1), 1–10. <https://doi.org/10.33050/tmj.v7i1.1762>
- Khalid, M., Akram, R., & Muttaqin, K. (2022). Sistem Monitoring Pasang Surut Air Laut Berbasis Web Menggunakan Fuzzy Logic Pada Kuala Langsa. *Journal of Information and Technology*, 2(2), 65–69.  
<https://doi.org/10.32938/jitu.v2i2.3254>
- Malius, H. (2021). *Sistem informasi sekolah berbasis web pada sekolah dasar negeri ( sdn ) 109 seriti*. 1(3), 156–168.
- Megawaty, D. A., & Putra, M. E. (2020). Aplikasi Monitoring Aktivitas Akademik Mahasiswa Program Studi Informatika Universitas Xyz Berbasis *Android*. *Jurnal Informatika Dan Rekayasa Perangkat Lunak*, 1(1), 65–74.

<https://doi.org/10.33365/jatika.v1i1.177>

- Nasihi, A., Asihati Ratna Hapsari, T., & Kota Jakarta Selatan, K. (2022). Indonesian Journal of Teaching and Learning. *Journal of Teaching and Learning*, 1(1), 77–88. <http://journals.eduped.org/index.php/intel>
- Nelly Sofi, & Riza Dharmawan. (2022). Perancangan Aplikasi Bengkel Csm Berbasis *Android* Menggunakan Framework Flutter (Bahasa Dart). *Jurnal Teknik Dan Science*, 1(2), 53–64. <https://doi.org/10.56127/jts.v1i2.125>
- Ningsih, K. S., Aruan, N. J., & Siahaan, A. T. A. A. (2022). Aplikasi Buku Tamu Menggunakan Fitur Kamera Dan Ajax Berbasis Website Pada Kantor Dispora Kota Medan. *SITek: Jurnal Sains, Informatika, Dan Teknologi*, 1(3), 94–99. <https://jurnal.insanciptamedan.or.id/index.php/sitek/article/view/75>
- Nuerita Maharani, R. B., Nasution, M. I. P., & Triase, T. (2021). Sistem Informasi Payroll Pegawai dengan Absensi QR Code. *Jurnal Informatika Dan Teknologi Pendidikan*, 1(1), 23–35. <https://doi.org/10.25008/jitp.v1i1.9>
- Nurmadiyah, N. (2022). Manajemen Sarana Dan Prasarana. *Jurnal Al-Afkar : Manajemen Pendidikan Islam*, 6(1), 30–50. <https://doi.org/10.32520/afkar.v6i1.190>
- Nurul, S., Shynta Anggrainy, & Siska Aprelyani. (2022). Faktor-Faktor Yang Mempengaruhi Keamanan Sistem Informasi: Keamanan Informasi, Teknologi Informasi Dan Network (Literature Review Sim). *Jurnal Ekonomi Manajemen Sistem Informasi*, 3(5), 564–573. <https://doi.org/10.31933/jemsi.v3i5.992>
- Ovender, F., Hartawan, R., & Marwan, E. (2021). Respon Pertumbuhan Bibit Kakao (*Theobroma cacao* L.) Terhadap Pemberian Kompos Limbah Kelapa Sawit. *Jurnal Media Pertanian*, 6(2), 57. <https://doi.org/10.33087/jagro.v6i2.118>
- Prasetyawan, P., Samsugi, S., & Prabowo, R. (2021). Internet of Thing Menggunakan Firebase dan Nodemcu untuk Helm Pintar. *Jurnal ELTIKOM*, 5(1), 32–39. <https://doi.org/10.31961/eltikom.v5i1.239>
- Putra, M. Y., & Kurniawan, D. E. (2023). Implementasi Sistem Reminder Jadwal pada eLearning Moodle Berbasis API Menggunakan Framework Flutter. *Journal of Applied Computer Science and Technology*, 4(1), 7–11. <https://doi.org/10.52158/jacost.v4i1.490>
- Riduwan. (2023). Kerangka berfikir penelitian. *Tarbiyah: Jurnal Ilmu Pendidikan Dan Pengajaran*, 2(1), 160–166. <https://jurnal.diklinko.id/index.php/tarbiyah/article/view/25>
- Rifa'i, Y. (2023). Analisis Metodologi Penelitian Kualitatif dalam Pengumpulan Data di Penelitian Ilmiah pada Penyusunan Mini Riset. *Cendekia Inovatif Dan Berbudaya: Jurnal Ilmu Sosial Dan Humaniora*, 1(1), 31–37. <https://doi.org/10.59996/cendib.v1i1.155>
- Rusdi, E. (2023). Karet manajemen dan pengelolaan kebun. *Buku Manajemen Dan Pengelolaan Kebun Karet* 10–26.

<https://www.scribd.com/document/626004813/Pohon-karet>

- S Farhany. (2021). Unikom\_Sidqia Farhany\_Bab 2. *Jurnal Cendekia Ilmiah*, 3(139),9–33. <https://elibrary.unikom.ac.id/id/>
- Saádi Ahmad. (2025). Pengumpulan Data Yang Efisien pada Penelitian Tindakan Kelas: Teknik, Alat, dan Tantangan. *Al-Amin: Jurnal Ilmu Pendidikan Dan Sosial Humaniora*, 2, 90–108. <https://doi.org/10.53398/alaman.v2i2.377>
- Sahir, S. H. (2022). *Buku ini di tulis oleh Dosen Universitas Medan Area Hak Cipta di Lindungi oleh Undang-Undang Telah di Deposit ke Repository UMA pada tanggal 27 Januari 2022*. <https://books.google.co.id/books>
- Sandi, A., Mandyara, D. R. M., & Burhanuddin, B. (2021). Pengaruh Faktor Produksi Tahu Terhadap Pendapatan Pada Usaha Mikro Kecil Dan Menengah (Ukm) Gunung Kijang Di Lingkungan Sarata Kelurahan Paruga Kota Bima. *Jurnal PenKomi: Kajian Pendidikan Dan Ekonomi*, 4(1), 18–28. <https://doi.org/10.33627/pk.v4i1.438>
- Saputra, M. G., Munaa, N., Anggraini, Y., Ummah, F., Rahmawati, N. V., Kusdiyana, A., & Nuryati, N. (2023). Evaluasi Implementasi Sistem Informasi Manajemen Rumah Sakit dengan Metode HOT-Fit di RSUD Muhammadiyah Babat. *J-REMI: Jurnal Rekam Medik Dan Informasi Kesehatan*, 4(4), 248–256. <https://doi.org/10.25047/j-remi.v4i4.4047>
- Setiawan, P. R., Ramadhan, R. A., & Labellapansa, A. (2022). Jurnal Pengabdian Masyarakat dan Penerapan Ilmu Pengetahuan Pelatihan Pemrograman Flutter. *Jurnal Pengabdian Masyarakat Dan Penerapan Ilmu Pengetahuan*, 3(1), 22–27. <https://journal.uir.ac.id/index.php/jpmpip/article/view/10699/4628>
- Shara, D. H., & Adzkia Putri Marsya. (2022). Smart farming teknologi monitoring produksi dan. *Jusikom: Jurnal Sistem Komputer Musirawas*, 7(1), 20–31. <https://d1wqtxts1xzle7.cloudfront.net>
- Siregar, J. A. S., & Handoko, K. (2021). Perancangan sistem monitoring kelembapan tanah pada tanaman cabai berbasis internet of things. *Jurnal ComasieJurnalComasie*, 6(2), 40–51. <http://ejournal.upbatam.ac.id/index.php>
- Sudradjat, B. (2021). Penggunaan Teknologi Flutter dalam Aplikasi Mobile untuk Pengembangan Kedai Kopi. *Jurnal Remik: Riset Dan E-Jurnal Manajemen Informatika Komputer*, 6(1), 1–8. <https://doi.org/10.33395/remik.v6i1.11123>
- Sulistyorinia, T., Sofib, N., & Sova, E. (2022). Pemanfaatan Nodemcu Esp8266 Berbasis Android ( Blynk ) Sebagai Alat Alat. *Jurnal Ilmiah Teknik*, 1(3), 40–53. <https://doi.org/10.56127/juit.v1i3.334>
- Suryana, T. (2021). Belajar Bahasa Pemrograman Dart. *Teknik Informatika Unikom*, 1, 4.
- Susantok, M., Urip, A., Wibowo, A., Akbar, M., Riau, P. C., & Korespondensi, P. (2025). Peningkatan akurasi sistem pemantauan suhu dan kelembapan pada laboratorium pengujian benih tanaman menggunakan. *Jurnal Teknologi Informasi Dan Ilmu Komputer (JTIK)*, 12(1), 153–164.

<https://doi.org/10.25126/jtiik.2025129083>

- Syam, M. L., & Erdisna. (2022). Sistem Informasi Stok Barang Menggunakan QR-Code Berbasis *Android*. *Jurnal Informatika Ekonomi Bisnis*, 4. <https://doi.org/10.37034/infeb.v4i1.108>
- Tuasamu Zinab, Lewwary Nur Afni Intan M, Idris Muhamad Rivaldi, Syafaat Abdillah Bill Nazari, Fadlan Mariam, Nadiva Putri, & Efendi Rahmi. (2023). Analisis Sistem Informasi Akuntansi Siklus Pendapatan Menggunakan DFD Dan Flowchart Pada Bisnis Porobico. *Jurnal Bisnis Dan Manajemen (JURBISMAN)*, 1(2), 495–510. <https://doi.org/10.61930/jurbisman.v1i2.181>
- Versi Putra Jaya Hulu. (2022). Pengaruh Pemberian Inokulan Fungi Mikoriza Arbuskula Dan Pemupukan Fosfor Terhadap Pertumbuhan Bibit Karet (Hevea Brasiliensis Muell. Arg. *Jurnal Sapta Agrica*, 1(1), 1–11. <https://doi.org/10.57094/agrotek.v1i1.372>
- Wahyu Nusantara, A. (2022). Pengendalian Kualitas Proses Produksi Konveksi PT Kaosta Sukses Mulia. *Jurnal Kewirausahaan*, 9(1), 66–82. <http://stieamsir.ac.id/journal/index.php/man/article/view/146>
- Wayahdi, M. R., & Ruziq, F. (2023). Pemodelan Sistem Penerimaan Anggota Baru dengan Unified Modeling Language (UML) (Studi Kasus: Programmer Association of Battuta). *Jurnal Minfo Polgan*, 12(1), 1514–1521. <https://doi.org/10.33395/jmp.v12i1.12870>
- Wibowo, S. H. (2023). *Teknologi Digital Di Era Modern*.
- Wintana, D., Pribadi, D., & Nurhadi, M. Y. (2022). Analisis Perbandingan Efektifitas White-Box Testing dan Black-Box Testing. *Jurnal larik ladang artikel ilmu komputer ladang artikel ilmu komputer*, 2(1), 8–16. <http://jurnal.bsi.ac.id/index.php/larik>
- Yoga Pratama, R., & Somya, R. (2021). Perancangan Aplikasi Kasir Berbasis *Android* (Studi kasus Warkup Vape Salatiga). *Jurnal Teknik Informatika Dan Sistem Informasi*, 8(4), 2407–4322. <http://jurnal.mdp.ac.id>
- Yusuf, A. M., & Nopitasari, L. (2021). Rancang Bangun Sistem Presensi Mahasiswa Dengan Menggunakan QR Code Berbasis *Android*. *Elkom : Jurnal Elektronika Dan Komputer*, 14(1), 47–58. <http://journal.stekom.ac.id/index.php/elkom>

## LAMPIRAN

### Lampiran 1 Surat Permohonan Penelitian

  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS PRABUMULIH**

Prabumulih, 09 Oktober 2025

Nomor : 017 /FASILKOM/UNPRA/Inf-S1/X/2025  
Lampiran : -  
Perihal : Permohonan Pengambilan Data

Kepada Yth,  
Pimpinan PT. KENDI ARINDO  
Di  
Tempat

Dalam rangka pelaksanaan skripsi mahasiswa Program Studi Informatika Fakultas Ilmu Komputer Universitas Prabumulih, maka kami harapkan bantuan dari bapak/ ibu kiranya mengizinkan mahasiswa:

| No. | Nama                 | NIM        | Program Studi |
|-----|----------------------|------------|---------------|
| 1.  | Dimas Kurnia         | 2022230026 | Informatika   |
| 2.  | Rahmat Akbar Hidayat | 2022230034 |               |
| 3.  | Aji Muksen           | 2022230033 |               |

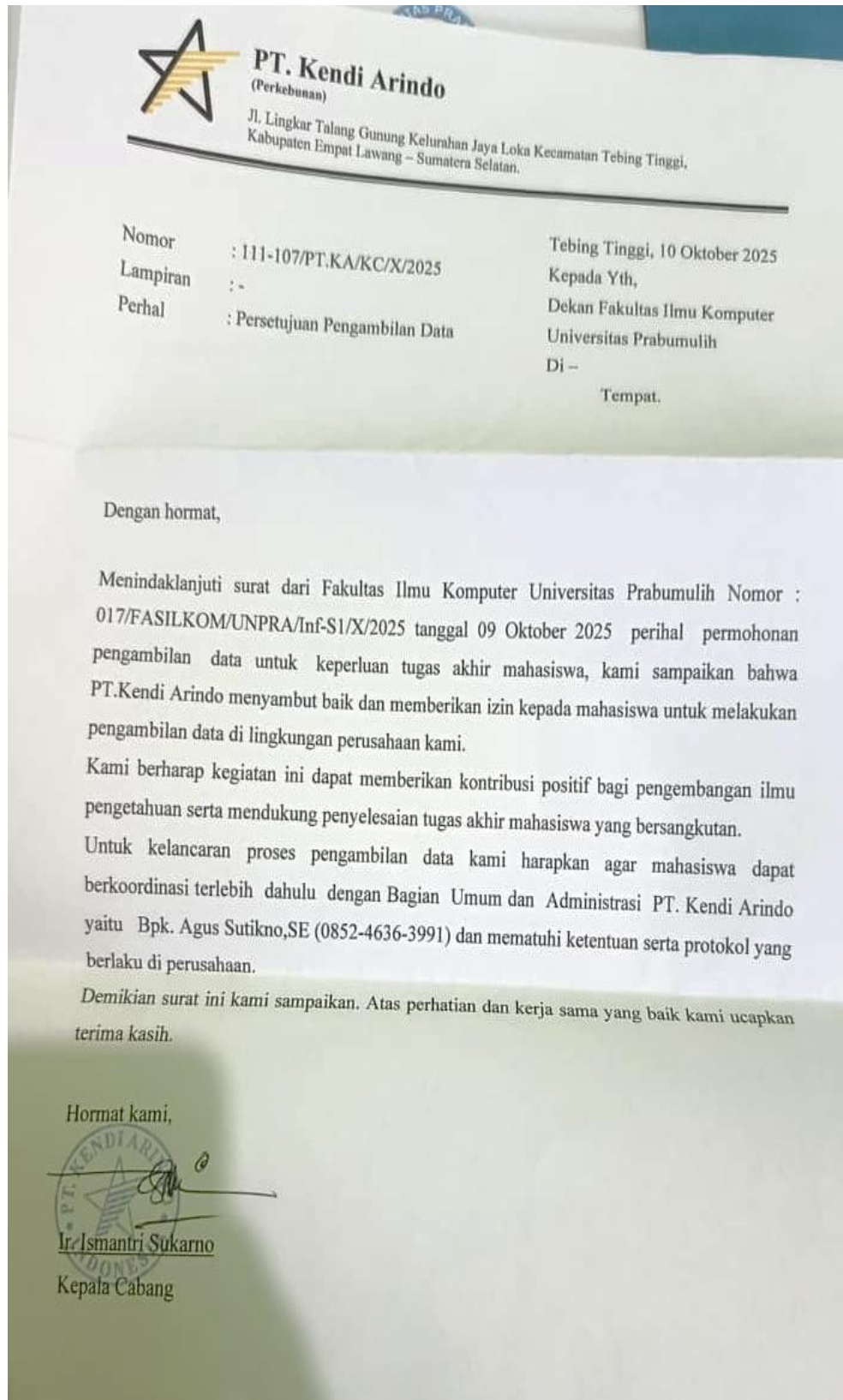
Untuk melaksanakan pengambilan data ditempat Bapak/Ibu Pimpin guna menyelesaikan skripsi. Pengambilan data tersebut akan dilaksanakan pada tanggal 22 Oktober 2025 s/d 28 Oktober 2025. Atas bantuan dan kerjasamanya kami ucapkan terima kasih.

Ketua Program Studi,  
Informatika,  
  
**Iwan Setiawan, S.Kom., M.Kom.**  
NIDN.0209129101

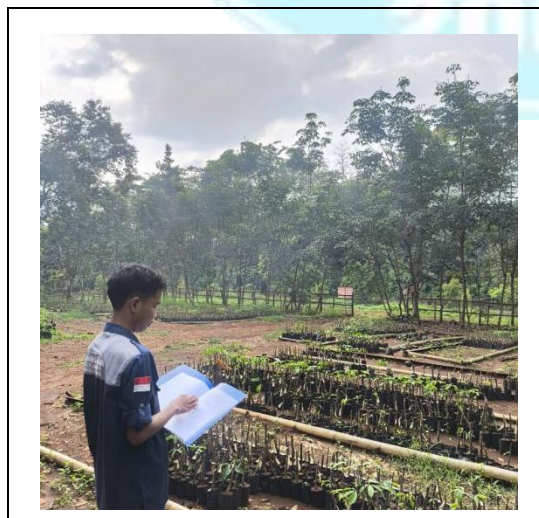
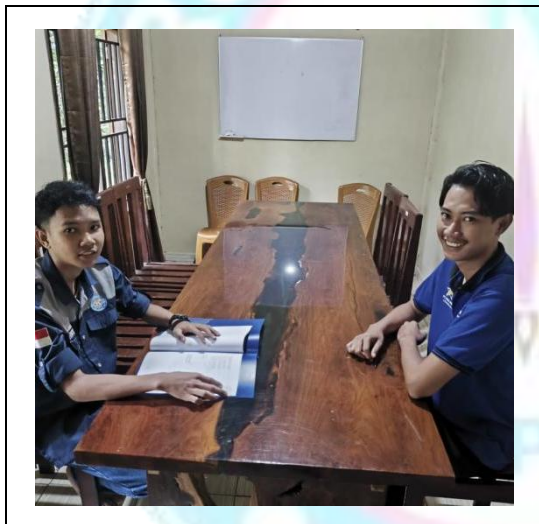


Selatan  
322418

## Lampiran 2 Surat Balasan Permohonan Penelitian



### Lampiran 3 Dokumentasi Wawancara dan Observasi





## Lampiran 4 Daftar Pertanyaan Wawancara

### DRAFT WAWANCARA

Nama Narasumber : Arif Junaidi  
Jabatan : Kepala Bagian Perencanaan  
Tempat Wawancara : Kantor PT Kendi Arindo  
Tanggal Wawancara : 22 Oktober 2025  
Pewawancara : Dimas Kurnia Ramadhan

| No | Pertanyaan                                                                                                       | Jawaban                                                                                                                                                                                                                                         |
|----|------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1  | Dapatkah bapak menjelaskan secara singkat mengenai profil PT Kendi Arindo dan kegiatan operasional utamanya?     | PT Kendi Arindo merupakan perusahaan yang bergerak di bidang perkebunan karet. Kegiatan utamanya meliputi pengelolaan lahan, pemeliharaan tanaman, penjadwalan, hingga pengolahan dan penyaluran hasil produksi karet.                          |
| 2  | Bagaimana struktur organisasi secara umum, khususnya yang terkait dengan pengelolaan bibit tanaman karet?        | Struktur organisasi kami terdiri dari pimpinan perusahaan, kepala bagian, dan beberapa subbagian, termasuk bagian produksi bibit yang bertanggung jawab terhadap pencatatan jumlah bibit, kondisi bibit, dan proses distribusi ke lokasi tanam. |
| 3  | Apa saja tugas dan tanggung jawab bapak di bagian pembibitan atau produksi bibit tanaman karet?                  | Tugas kami meliputi pendataan jumlah bibit masuk, bibit siap tanam, kondisi bibit, <i>monitoring</i> perawatan, dan membuat laporan perkembangan produksi bibit setiap periode.                                                                 |
| 4  | Sistem apa yang saat ini digunakan untuk mencatat data produksi bibit tanaman karet?                             | Saat ini pencatatan masih dilakukan secara manual menggunakan buku tulis dan laporan Excel sederhana, kemudian direkap menjadi laporan bulanan.                                                                                                 |
| 5  | Menurut bapak, apa saja kendala atau kelemahan dari sistem pencatatan manual tersebut?                           | Kendalanya antara lain risiko kesalahan pencatatan, proses rekap yang memakan waktu, laporan tidak real-time, serta sulit menelusuri riwayat data jika jumlahnya sudah banyak.                                                                  |
| 6  | Apakah pernah terjadi keterlambatan laporan atau ketidaksesuaian data pada proses pencatatan produksi bibit?     | Ya, terkadang terjadi perbedaan angka antara catatan lapangan dan rekap akhir, terutama saat volume bibit tinggi dan pencatatan dilakukan oleh lebih dari satu petugas.                                                                         |
| 7  | Data apa saja yang biasanya dibutuhkan pimpinan terkait produksi bibit untuk evaluasi dan pengambilan keputusan? | Pimpinan biasanya membutuhkan data jumlah bibit harian, bibit siap tanam, bibit rusak/mati, progres produksi setiap periode, serta grafik perkembangan pertumbuhan bibit.                                                                       |

- |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>8 Apakah pimpinan dapat mengakses laporan produksi bibit secara cepat dan akurat dengan sistem yang ada saat ini?</p> <p>9 Menurut bapak, Seberapa besar kebutuhan perusahaan terhadap sistem digitalisasi informasi produksi bibit tanaman karet?</p> <p>10 Bagaimana pandangan bapak mengenai rencana pembuatan sistem digitalisasi monitoring informasi data produksi bibit tanaman karet menggunakan aplikasi <i>android</i>?</p> <p>11 Menurut bapak, fitur apa saja yang dianggap penting untuk ada dalam aplikasi monitoring produksi bibit tanaman karet tersebut?</p> <p>12 Apakah aplikasi <i>android</i> ini dapat mengurangi kesalahan pencatatan dibandingkan metode manual?</p> <p>13 Menurut bapak, apakah penggunaan aplikasi ini akan membantu petugas lapangan dalam pencatatan harian produksi bibit?</p> <p>14 Manfaat apa yang diharapkan perusahaan setelah penerapan sistem digital ini?</p> <p>15 Apakah bapak bersedia jika dilakukan uji coba aplikasi yang sedang dikembangkan oleh peneliti</p> | <p>Belum, laporan masih harus disiapkan secara manual sehingga pimpinan tidak bisa mengakses data secara langsung atau real-time.</p> <p>Menurut kami kebutuhannya cukup besar, karena produksi bibit merupakan kegiatan vital dan membutuhkan data yang akurat, cepat diakses, dan mudah dipantau oleh pimpinan.</p> <p>Kami sangat mendukung, sistem digital dapat mempermudah pencatatan data di lapangan dan mempercepat akses informasi bagi pimpinan maupun bagian operasional.</p> <p>Fitur input data bibit, dan tanaman <i>monitoring</i> pertumbuhan, foto kondisi bibit dan tanaman, laporan otomatis, grafik perkembangan, dan penyimpanan riwayat data.</p> <p>Iya, karena data langsung tersimpan dalam sistem dan mengurangi risiko kesalahan tulis atau hilangnya catatan.</p> <p>Sangat membantu, karena pencatatan bisa dilakukan cepat di lokasi, tanpa perlu buku manual dan tanpa rekap ulang.</p> <p>Data lebih akurat, pengolahan laporan lebih cepat, monitoring produksi lebih terstruktur, serta mempermudah pimpinan melakukan evaluasi dan perencanaan.</p> <p>Insya Allah bersedia, selama uji coba berjalan dengan jelas, tidak mengganggu pekerjaan harian, dan mampu memberikan manfaat Perusahaan.</p> |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

Menyetujui,

**Narasumber**



**Arif Junaidi**

**Kepala Bagian Perencanaan**

Empat Lawang, 22 Oktober 2025

**Pewawancara**

**Dimas Kurnia Ramadhan**

## DRAFT WAWANCARA

Nama Narasumber : Imam Wahyu Romadhon

Jabatan : Kepala Bagian Produksi

Tempat Wawancara : Kantor PT Kendi Arindo

Tanggal Wawancara : 24 Januari 2026

Pewawancara : Dimas Kurnia Ramadhan

| No | Pertanyaan                                                                                  | Jawaban                                                                                                                                                                                                                     |
|----|---------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1  | Dapatkah bapak menjelaskan secara singkat peran bagian produksi di PT Kendi Arindo?         | Bagian produksi bertanggung jawab terhadap seluruh proses operasional di lapangan, mulai dari pengelolaan pembibitan, pemeliharaan tanaman, hingga memastikan produksi bibit berjalan sesuai target dan standar perusahaan. |
| 2  | Apa saja aktivitas utama yang dilakukan di bagian produksi bibit tanaman karet?             | Aktivitas utama meliputi penerimaan bibit, perawatan dan pemantauan pertumbuhan bibit, pencatatan kondisi bibit, serta pendistribusian bibit ke area tanam.                                                                 |
| 3  | Bagaimana proses pencatatan data produksi bibit yang dilakukan saat ini di bagian produksi? | Saat ini pencatatan dilakukan secara manual oleh petugas lapangan menggunakan buku catatan, kemudian data tersebut direkap kembali ke dalam laporan Excel.                                                                  |
| 4  | Data apa saja yang dicatat oleh bagian produksi terkait bibit tanaman karet?                | Data yang dicatat antara lain jumlah bibit masuk, bibit hidup, bibit mati atau rusak, bibit siap tanam, serta kondisi pertumbuhan bibit secara berkala.                                                                     |
| 5  | Apa kendala yang sering dihadapi dalam proses pencatatan data produksi bibit secara manual? | Kendala yang sering terjadi adalah kesalahan pencatatan, data tidak sinkron antar petugas, pencatatan memakan waktu lama, dan kesulitan saat melakukan rekap data dalam jumlah besar                                        |
| 6  | Apakah pernah terjadi perbedaan data antara catatan lapangan dan laporan akhir produksi?    | Ya, perbedaan data terkadang terjadi, terutama saat aktivitas produksi padat dan pencatatan dilakukan oleh lebih dari satu petugas lapangan.                                                                                |
| 7  | Bagaimana proses pelaporan data produksi bibit kepada pimpinan atau manajemen?              | Data produksi direkap terlebih dahulu oleh bagian produksi, kemudian disampaikan kepada pimpinan dalam bentuk laporan tertulis atau file Excel secara berkala.                                                              |
| 8  | Menurut bapak, apakah sistem pencatatan saat ini sudah efektif dan efisien?                 | Menurut kami belum sepenuhnya efektif, karena masih bergantung pada pencatatan manual yang berisiko menimbulkan kesalahan dan keterlambatan laporan.                                                                        |

- |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>9 Seberapa penting ketersediaan data produksi bibit yang cepat dan akurat bagi bagian produksi?</p> <p>10 Bagaimana pendapat bapak mengenai rencana penerapan sistem digitalisasi monitoring data produksi bibit tanaman karet?</p> <p>11 Fitur apa saja yang menurut bapak perlu tersedia dalam aplikasi monitoring produksi bibit tersebut?</p> <p>12 Apakah aplikasi berbasis <i>android</i> dapat membantu pekerjaan petugas produksi di lapangan?</p> <p>13 Apakah penggunaan aplikasi ini berpotensi mengurangi kesalahan pencatatan data produksi?</p> <p>14 Manfaat apa yang paling dirasakan bagian produksi jika sistem digital ini diterapkan?</p> <p>15 Apakah bapak bersedia mendukung dan mengikuti uji coba aplikasi monitoring produksi bibit yang dikembangkan?</p> | <p>Sangat penting, karena data tersebut menjadi dasar pengambilan keputusan operasional dan perencanaan produksi selanjutnya.</p> <p>Kami menilai rencana tersebut sangat baik dan dibutuhkan, karena dapat membantu mempercepat pencatatan dan meminimalkan kesalahan data.</p> <p>Fitur yang diperlukan antara lain input data produksi, monitoring pertumbuhan bibit, laporan otomatis, dan pemantauan perkembangan produksi.</p> <p>Ya, sangat membantu karena petugas bisa langsung mencatat data di lokasi tanpa harus menulis manual dan melakukan rekap ulang.</p> <p>Iya, karena data langsung tersimpan dalam sistem dan mengurangi risiko kesalahan tulis atau kehilangan catatan.</p> <p>Manfaatnya antara lain pencatatan lebih rapi, data mudah dicari, laporan lebih cepat dibuat, serta monitoring produksi menjadi lebih terstruktur.</p> <p>Insha Allah bersedia, selama aplikasi mudah digunakan, tidak mengganggu aktivitas produksi, dan memberikan manfaat nyata bagi pekerjaan di lapangan.</p> |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

Menyetujui,

**Narasumber**



**Imam Wahyu Romadhon**  
**Kepala Bagian Produksi**

Empat Lawang, 24 Januari 2026

**Pewawancara**



**Dimas Kurnia Ramadhan**



**FAKULTAS ILMU KOMPUTER  
UNIVERSITAS PRABUMULIH**

**KESEDIAAN SEBAGAI PEMBIMBING SKRIPSI  
PROGRAM STUDI INFORMATIKA**

Yang bertanda tangan di bawah ini:

Nama : Iwan Setiawan, S.Kom., M.Kom.  
NUPTK : 1541769670130303  
Program Studi : Informatika  
Fakultas : Ilmu Komputer

Bersedia menjadi pembimbing I Skripsi, dan

Nama : Hermizahadiwidastra, S.Kom., M.Kom.  
NUPTK : 2863762663130192  
Program Studi : Informatika  
Fakultas : Ilmu Komputer

Bersedia menjadi pembimbing II Skripsi

dari mahasiswa Program Studi Informatika Fakultas Ilmu Komputer Universitas Prabumulih  
di bawah ini:

Nama : Dimas Kurnia Ramadhan  
NIM : 2022230026  
Judul Skripsi : Digitalisasi *Monitoring* Informasi Data Produksi Bibit  
Tanaman Karet Berbasis *Android*

Demikian surat kesediaan membimbing ini kami buat, atas perhatiannya kami ucapkan terima kasih.

Dosen Pembimbing I

Prabumulih,  
Dosen Pembimbing II

2026

(Iwan Setiawan, S.Kom., M.Kom.)  
NUPTK. 1541769670130303

(Hermizahadiwidastra, S.Kom., M.Kom.)  
NUPTK. 2863762663130192



FAKULTAS ILMU KOMPUTER  
UNIVERSITAS PRABUMULIH

KARTU BIMBINGAN SKRIPSI

NAMA : DIMAS KURNIA RAMADHAN  
NIM : 2022230026  
PROGRAM STUDI : INFORMATIKA  
NAMA PEMBIMBING I : IWAN SETIAWAN, S.Kom., M.Kom  
JUDUL : DIGITALISASI *MONITORING* INFORMASI DATA  
PRODUKSI BIBIT TANAMAN KARET BERBASIS *ANDROID*

| NO | TGL. KONSULTASI | TOPIK POKOK YANG DIBICARAKAN                                                                | TANDA TANGAN PEMB. | TGL MENGHADAP KEMBALI |
|----|-----------------|---------------------------------------------------------------------------------------------|--------------------|-----------------------|
| 1  | 11-2-2026       | BAB I Acc                                                                                   |                    |                       |
|    |                 | BAB II Acc                                                                                  |                    |                       |
|    |                 | BAB III<br>a. Diagon                                                                        |                    |                       |
| 2. | 15-2-2026       | BAB III Acc                                                                                 |                    |                       |
| 3. | 14-4-2026       | BAB IV<br>a. kebutuhan data<br>b. Bahaya asing<br>c. P. Sistem<br>d. Patching<br>e. UI / UX |                    |                       |

Mulai Bimbingan : 11-2-2026  
Batas Akhir Bimbingan : .....

Mengetahui Kaprodi Informatika

IWAN SETIAWAN, S.Kom.M.Kom.

PERHATIAN !  
TIDAK BOLEH HILANG  
SETIAP BIMBINGAN HARUS DIBAWA





FAKULTAS ILMU KOMPUTER  
UNIVERSITAS PRABUMULIH

KARTU BIMBINGAN SKRIPSI

MA : DIMAS KURNIA RAMADHAN  
M : 2022230026  
OGRAM STUDI : INFORMATIKA  
MA PEMBIMBING I : IWAN SETIAWAN, S.Kom., M.Kom  
DUL : DIGITALISASI *MONITORING* INFORMASI DATA  
PRODUKSI BIBIT TANAMAN KARET BERBASIS *ANDROID*

| NO | TGL KONSULTASI | TOPIK POKOK YANG DIBICARAKAN         | TANDA TANGAN PEMB. I | TGL MENGHADAP KEMBALI |
|----|----------------|--------------------------------------|----------------------|-----------------------|
| 2. | 29/5/2026      | BAB V ACC                            | <i>[Signature]</i>   |                       |
| 3. | 29/5/2026      | BAB VI<br>a. KeSimputan<br>b halaman | <i>[Signature]</i>   |                       |
| 0  | 29/5/2026      | BAB VI ACC                           | <i>[Signature]</i>   |                       |
| 1. | 3/6/2026       | Kelengkapan SIAP UJI                 | <i>[Signature]</i>   |                       |
|    |                |                                      |                      |                       |
|    |                |                                      |                      |                       |
|    |                |                                      |                      |                       |
|    |                |                                      |                      |                       |
|    |                |                                      |                      |                       |
|    |                |                                      |                      |                       |

ulai Bimbingan : .....  
tas Akhir Bimbingan : .....

Mengetahui Kaprodi Informatika

*[Signature]*

IWAN SETIAWAN, S.Kom.M.Kom.

PERHATIAN !  
TIDAK BOLEH HILANG  
SETIAP BIMBINGAN HARUS DIBAWA



FAKULTAS ILMU KOMPUTER  
UNIVERSITAS PRABUMULIH

KARTU BIMBINGAN SKRIPSI

NAMA : DIMAS KURNIA RAMADHAN  
NIM : 2022230026  
PROGRAM STUDI : INFORMATIKA  
NAMA PEMBIMBING II : HERMIZAHADIWIDASTRA, S.Kom., M.Kom  
JUDUL : DIGITALISASI MONITORING INFORMASI DATA  
PRODUKSI BIBIT TANAMAN KARET BERBASIS ANDROID

| NO | TGL. KONSULTASI | TOPIK POKOK YANG DIBICARAKAN                                                                                                                                             | TANDA TANGAN PEMB. II | TGL MENGHADAP KEMBALI |
|----|-----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|-----------------------|
| 1  | 11/2026<br>02   | BAB I 1.5. teori & praktis.<br>BAB II ACC <del>for</del> lanjut.<br>BAB III Subra & OL 2026<br>- DODANIL PENELITIAN TABEL<br>- TEORI penelitian.<br>- AKTIFITI. Diagram. |                       |                       |
| 2  | 1/03 2026       | ACC BAB I lanjut BAB II<br>BAB II ACC <del>for</del> lanjut BAB II<br>BAB III 3.6.2.<br>Diagram.                                                                         |                       |                       |
| 3  | 7/04 2026       | ACC BAB II                                                                                                                                                               |                       |                       |

Mulai Bimbingan : .....  
Batas Akhir Bimbingan : .....

Mengetahui Kaprodi Informatika

IWAN SETIAWAN, S.Kom.M.Kom.

PERHATIAN !  
TIDAK BOLEH HILANG  
SETIAP BIMBINGAN HARUS DIBAWA



FAKULTAS ILMU KOMPUTER  
UNIVERSITAS PRABUMULIH

KARTU BIMBINGAN SKRIPSI

NAMA : DIMAS KURNIA RAMADHAN  
NIM : 2022230026  
PROGRAM STUDI : INFORMATIKA  
NAMA PEMBIMBING II : HERMIZAHADIWIDASTRA, S.Kom., M.Kom  
JUDUL : DIGITALISASI MONITORING INFORMASI DATA  
PRODUKSI BIBIT TANAMAN KARET BERBASIS ANDROID

| NO | TGL. KONSULTASI | TOPIK POKOK YANG DIBICARAKAN                                                                                                                                                                                                        | TANDA TANGAN PEMB. II | TGL MENGHADAP KEMBALI |
|----|-----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|-----------------------|
| 4  | 9/04 2026       | BAB IV<br>- Bab IV tentang.<br>- 4.2.1. - use case ?<br>- activity ?<br>- class Diagram ?                                                                                                                                           |                       |                       |
| 5  | 14/05 2026      | BAB IV<br>4.2.1 Use case function ✓<br>4.2.3 UI/UX Gambar ✓<br>- format penulisan paragraf ✓<br>- Gambar ✓<br>- Bab IV UI/UX<br>- Laporan produksi ✓<br>+ menu tanda tanaman bibit<br>+ menu seleksi tanaman bibit<br>UI/UX Gambar. |                       |                       |

Mulai Bimbingan : .....

Status Akhir Bimbingan : .....

Mengetahui Kaprodi Informatika

PERHATIAN !  
TIDAK BOLEH HILANG  
SETIAP BIMBINGAN HARUS DIBAWA

IWAN SETIAWAN, S.Kom.M.Kom.



FAKULTAS ILMU KOMPUTER  
UNIVERSITAS PRABUMULIH

KARTU BIMBINGAN SKRIPSI

NAMA : DIMAS KURNIA RAMADHAN  
NIM : 2022230026  
PROGRAM STUDI : INFORMATIKA  
NAMA PEMBIMBING II : HERMIZAHADIWIDASTRA, S.Kom., M.Kom  
JUDUL : DIGITALISASI MONITORING INFORMASI DATA  
PRODUKSI BIBIT TANAMAN KARET BERBASIS ANDROID

| NO | TGL KONSULTASI | TOPIK POKOK YANG DIBICARAKAN                                                                                                                      | TANDA TANGAN PEMB. II | TGL MENGHADAP KEMBALI |
|----|----------------|---------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|-----------------------|
| 6. | 02/2026<br>/06 | Basis IV ACC 242<br>Cungut Basis IV<br>- format penulisan - ✓<br>- bahasa asing.<br>- permasalahan yg berkaitan gambar.<br>- Kapan lagi penulisan | #52<br>#52            |                       |
| 7  | 04/2026<br>/06 | - tata letak menu<br>karyawan, kepelatihan, pmp.<br>- Halaman Manajemen Pengaruh<br>kepada bahan SS kejuruan<br>Desain Data aplikasi.             | #52<br>#52            |                       |
| 8  | 04/2026<br>/06 |                                                                                                                                                   |                       |                       |

Mulai Bimbingan : .....

Batas Akhir Bimbingan : .....

Mengetahui Kaprodi Informatika

PERHATIAN!  
TIDAK BOLEH HILANG  
SETIAP BIMBINGAN HARUS DIBAWA

IWAN SETIAWAN, S.Kom.M.Kom.



**FAKULTAS ILMU KOMPUTER  
UNIVERSITAS PRABUMULIH**

## KARTU BIMBINGAN SKRIPSI

**NAMA : DIMAS KURNIA RAMADHAN**  
**NIM : 2022230026**  
**PROGRAM STUDI : INFORMATIKA**  
**NAMA PEMBIMBING II : HERMIZAHADIWIDASTRA, S.Kom., M.Kom**  
**JUDUL : DIGITALISASI *MONITORING* INFORMASI DATA**  
**PRODUKSI BIBIT TANAMAN KARET BERBASIS *ANDROID***

| NO | TGL KONSULTASI | TOPIK POKOK YANG DIBICARAKAN                                 | TANDA TANGAN PEMB. II | TGL MENGHADAP KEMBALI |
|----|----------------|--------------------------------------------------------------|-----------------------|-----------------------|
| 9. | 09/06/2026     | - ACC BAB V #07<br>- lanjut BAB VI #02<br>- ACC BAB B.VI #02 | #07<br>#02            |                       |
|    |                | Siap Uji                                                     |                       |                       |

Mulai Bimbingan : .....  
Batas Akhir Bimbingan : .....

Mengetahui Kaprodi Informatika

**PERHATIAN !  
TIDAK BOLEH HILANG  
SETIAP BIMBINGAN HARUS DIBAWA**

**IWAN SETIAWAN, S.Kom.M.Kom.**



**FAKULTAS ILMU KOMPUTER  
UNIVERSITAS PRABUMULIH**

**PENILAIAN REVIEW SKRIPSI**

Nama : Dimas Kurnia Ramadhan  
NIM : 2022230026  
Judul Skripsi : Digitalisasi *Monitoring* Informasi Data Produksi Bibit  
Tanaman Karet Berbasis *Android*  
Hari/Tanggal : JUMAT / 19-6-2026  
Saran dan masukan :

IKUTI ARAHAN PENGUJI DENGAN KONSULTASI  
KE PEMBIMBING DAHULU

Desi Siap Jilid

16-7-2026

**Rekomendasi\*:**

- a. Dilanjutkan tanpa perbaikan
- b. Dilanjutkan dengan perbaikan
- c. Ditolak

Prabumulih, 19-6-2026  
Pembimbing I

(Iwan Setiawan, S.Kom., M.Kom.)  
NUPTK. 1541769670130303

\*pilih salah satu



**FAKULTAS ILMU KOMPUTER  
UNIVERSITAS PRABUMULIH**

**PENILAIAN REVIEW SKRIPSI**

Nama : Dimas Kurnia Ramadhan  
NIM : 2022230026  
Judul Skripsi : Digitalisasi *Monitoring* Informasi Data Produksi Bibit  
Tanaman Karet Berbasis *Android*  
Hari/Tanggal : *Umot 19 Juni 2026*  
Saran dan masukan :

*lentin Saya arahkan Dari Dosen penguji*

*Sup Mld 15/06  
2026*  
*Sec*

**Rekomendasi\*:**

- a. Dilanjutkan tanpa perbaikan
- ☒ b. Dilanjutkan dengan perbaikan
- c. Ditolak

Prabumulih, *Umot 19 Juni 2026*  
Pembimbing II

*[Signature]*  
(Hermizahadiwidastra, S.Kom., M.Kom)  
NUPTK. 2863762663130192

*\*pilih salah satu*



**FAKULTAS ILMU KOMPUTER  
UNIVERSITAS PRABUMULIH**

**PENILAIAN REVIEW SKRIPSI**

Nama : Dimas Kurnia Ramadhan  
NIM : 2022230026  
Judul Skripsi : Digitalisasi *Monitoring* Informasi Data Produksi Bibit  
Tanaman Karet Berbasis *Android*  
Hari/Tanggal : 19 Juni 2026  
Saran dan masukan :

1. Keres setiap bedang. y setiap nuclei gambar.
- 2.

*per gild SW*

**Rekomendasi\*:**

- a. Dilanjutkan tanpa perbaikan
- b. Dilanjutkan dengan perbaikan
- c. Ditolak

Prabumulih, 24 - 6 - 2026  
Dosen Penguji

(Suhartini, S.Kom., M.Kom)  
NUPTK. 253475765823105

\*pilih salah satu