

**IMPLEMENTASI METODE *RAPID APPLICATION  
DEVELOPMENT (RAD)* DALAM PERANCANGAN DAN  
PEMBUATAN APLIKASI MOBILE KEBUNQ BPP LAMPUNG**

**TUGAS AKHIR**

Diajukan sebagai syarat menyelesaikan jenjang strata Satu (S-1) di Program Studi  
Teknik Informatika, Jurusan Teknologi, Produksi dan Industri, Institut Teknologi  
Sumatera

Oleh :

**RIZKI JULIANSYAH**

14116151



**PROGRAM STUDI TEKNIK INFORMATIKA  
JURUSAN TEKNOLOGI PRODUKSI DAN INDUSTRI  
INSTITUT TEKNOLOGI SUMATERA  
LAMPUNG SELATAN**

2022

**IMPLEMENTASI METODE *RAPID APPLICATION  
DEVELOPMENT (RAD)* DALAM PERANCANGAN DAN  
PEMBUATAN APLIKASI MOBILE KEBUNQ BPP LAMPUNG**

**TUGAS AKHIR**

Diajukan sebagai syarat menyelesaikan jenjang strata Satu (S-1) di Program Studi  
Teknik Informatika, Jurusan Teknologi, Produksi dan Industri, Institut Teknologi  
Sumatera

Oleh :

**RIZKI JULIANSYAH**

**14116151**



**PROGRAM STUDI TEKNIK INFORMATIKA  
JURUSAN TEKNOLOGI PRODUKSI DAN INDUSTRI  
INSTITUT TEKNOLOGI SUMATERA  
LAMPUNG SELATAN**

**2022**

## LEMBAR PENGESAHAN

Tugas Akhir dengan judul “Implementasi Metode *Rapid Application Development* (RAD) Dalam Perancangan dan Pembuatan Aplikasi Mobile KebunQ BPP Lampung” adalah benar dibuat oleh saya sendiri dan belum pernah dibuat dan diserahkan sebelumnya, baik sebagian ataupun seluruhnya, baik oleh saya ataupun orang lain, baik di Institut Teknologi Sumatera maupun di institusi pendidikan lainnya.

Lampung Selatan, DD-MM-YYYY

Penulis,

Foto

Rizki Juliansyah

NIM. 14116151

Diperiksa dan disetujui oleh,

Pembimbing

1. Nama Pembimbing,

NIP. XXXXX

Tanda Tangan

Disahkan oleh,  
Koordinator Program Studi Teknik Informatika  
Jurusan Teknologi, Produksi dan Industri  
Institut Teknologi Sumatera

Kaprodi, S.Si, M.Si  
NIP. XXXXXXXX

## **HALAMAN PERNYATAAN ORISINALITAS**

Tugas Akhir dengan judul “TULIS JUDUL DISINI” adalah karya saya sendiri, dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan benar.

Yang bertanda tangan di bawah ini:

Nama : Rizki Juliansyah

NIM : 14116151

Program Studi : Teknik Informatika

Fakultas : Teknik

Jenjang Pendidikan : Strata 1  
Menyatakan bahwa skripsi yang saya buat dengan judul:

### **PERANCANGAN SISTEM INFORMASI KOPERASI SIMPAN PINJAM BERBASIS WEB**

1. Merupakan hasil karya tulis ilmiah sendiri, bukan merupakan karya yang pernah diajukan untuk memperoleh gelar akademik oleh pihak lain, dan bukan merupakan hasil plagiat.
2. Saya ijin untuk dikelola oleh Universitas Pamulang sesuai dengan norma dan etika yang berlaku.

Pernyataan ini saya buat dengan penuh tanggung jawab dan saya bersedia menerima konsekuensi apapun sesuai aturan yang berlaku apabila dikemudian hari pernyataan ini tidak benar.

Pamulang, 01 Agustus 2014

(Nama Orang)

## **LEMBAR PERSETUJUAN**

Nama : Imron Rosdiadna  
NIM : 2010140419  
Program Studi : Teknik Informatika  
Fakultas : Teknik  
Jenjang Pendidikan : Strata 1  
Judul Skripsi : PERANCANGAN SISTEM INFORMASI KOPERASI  
SIMPAN PINJAM BERBASIS WEB

Skripsi ini telah diperiksa dan disetujui.

Pamulang, 01 Agustus 2014

Pembimbing 1, S.Kom, M, Kom

Pembimbing

Mengeetahui,

Kaprodi, S.Si, M.Si

KaProdi Teknik Informatika

## **KATA PENGANTAR**

Puji syukur kehadiran Allah SWT atas limpahan rahmat, karunia, serta petunjuk-Nya sehingga penyusunan tugas akhir ini telah terselesaikan dengan baik. Dalam penyusunan tugas akhir ini penulis telah banyak mendapatkan arahan, bantuan, serta dukungan dari berbagai pihak. Oleh karena itu pada kesempatan ini penulis mengucapkan terima kasih kepada:

Pamulang, 01 Agustus 2014

Imron Rosdiana

## ABSTRAK

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante.

Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Keywords: *Sistem Informasi, Testing Project*



## **ABSTRACT**

Halaman ABSTRAK berisi uraian tentang latar belakang, tujuan, metodologi penelitian, hasil / kesimpulan. Ditulis dalam BAHASA INDONESIA tidak lebih dari 250 kata, dengan jarak antar baris satu spasi. Pada akhir abstrak ditulis kata “Kata Kunci” yang dicetak tebal, diikuti tanda titik dua dan kata kunci yang tidak lebih dari 5 kata. Kata kunci terdiri dari kata-kata yang khusus menunjukkan dan berkaitan dengan bahan yang diteliti, metode/instrumen yang digunakan, topik penelitian. Kata kunci diketik pada jarak dua spasi dari baris akhir isi abstrak.

*Keywords: Information System, Testing Project*

## DAFTAR ISI

<b>LEMBAR PENGESAHAN</b>	<b>ii</b>
<b>LEMBAR PERNYATAAN</b>	<b>iv</b>
<b>LEMBAR PERSETUJUAN</b>	<b>v</b>
<b>KATA PENGANTAR</b>	<b>vi</b>
<b>ABSTRAK</b>	<b>vii</b>
<b>ABSTRACT</b>	<b>viii</b>
<b>DAFTAR ISI</b>	<b>x</b>
<b>DAFTAR TABEL</b>	<b>xi</b>
<b>DAFTAR GAMBAR</b>	<b>xii</b>
<b>I PENDAHULUAN</b>	<b>1</b>
1.1 Latar Belakang . . . . .	1
1.2 Rumusan Masalah . . . . .	2
1.3 Tujuan Penelitian . . . . .	2
1.4 Batasan Masalah . . . . .	2
1.5 Manfaat Penelitian . . . . .	2
1.6 Sistematika Penulisan . . . . .	2
<b>II LANDASAN TEORI</b>	<b>4</b>
2.1 Tinjauan Pustaka . . . . .	4
2.2 Dasar Teori . . . . .	4
2.2.1 <i>Monitoring</i> dan Kontrol . . . . .	4
2.2.2 <i>Aplikasi Mobile</i> . . . . .	4
2.2.3 <i>Rapid Application Development</i> (RAD) . . . . .	5
2.2.4 Flutter . . . . .	6
2.2.5 <i>Application Programming Interface</i> (API) . . . . .	7
2.2.6 <i>Database</i> . . . . .	7
2.2.7 <i>Flowchart</i> . . . . .	8
2.2.8 <i>Use Case Diagram</i> . . . . .	9

2.2.9	<i>Sequence Diagram</i>	9
2.2.10	<i>Black Box Testing</i>	10
2.2.11	<i>User Acceptance Testing (UAT)</i>	10
2.2.12	<i>Skala Likert</i>	10
<b>III</b>	<b>METODE PENELITIAN</b>	<b>11</b>
3.1	Alur Penelitian	11
3.2	Penjabaran Langkah Penelitian	11
3.2.1	Studi Literatur	12
3.2.2	Observasi	12
3.2.3	RAD	12
3.2.4	Uji Lapangan	12
3.2.5	Kesimpulan	12
3.3	Alat dan Bahan Tugas Akhir	12
3.3.1	Alat	12
3.3.2	Bahan	13
3.4	Metode Tugas Akhir	13
3.5	Ilustrasi Perhitungan Metode	13
3.6	Rancangan Pengujian	15
<b>IV</b>	<b>HASIL PENELITIAN DAN PEMBAHASAN</b>	<b>16</b>
4.1	Hasil Penelitian	16
4.1.1	Penerapan RAD	16
4.2	Uji Lapangan	21
4.2.1	Data Hasil Observasi	21
4.3	Analisis Hasil Penelitian	21
<b>V</b>	<b>KESIMPULAN DAN SARAN</b>	<b>22</b>
5.1	Kesimpulan	22
5.2	Saran	22
	<b>DAFTAR PUSTAKA</b>	<b>23</b>
	<b>LAMPIRAN</b>	<b>26</b>

## **DAFTAR TABEL**

3.1	Bobot nilai jawaban . . . . .	13
3.2	Nilai Presentase . . . . .	14
3.3	Nilai Kesimpulan . . . . .	15

## DAFTAR GAMBAR

2.1	Gambar Ilustrasi Model RAD . . . . .	5
2.2	Simbol - simbol <i>flowchart</i> . . . . .	8
2.3	Komponen <i>Use case</i> . . . . .	9
3.1	Alur Penelitian . . . . .	11
3.2	Skala Penilaian . . . . .	15
4.1	Skema Tabel <i>Database</i> . . . . .	16
4.2	<i>Use Case Diagram</i> . . . . .	16
4.3	<i>Sequence Diagram Login</i> . . . . .	17
4.4	<i>Sequence Diagram</i> Melihat Menu / Alat . . . . .	17
4.5	<i>Sequence Diagram</i> Melihat Detail Alat . . . . .	17
4.6	<i>Sequence Diagram</i> Melakukan Kontrol . . . . .	18
4.7	<i>Sequence Diagram</i> Melakukan Kontrol Otomatis . . . . .	18
4.8	Rancangan layout UI . . . . .	19
4.9	<i>assets</i> Logo aplikasi . . . . .	19
4.10	<i>assets</i> Ikon Sensor . . . . .	20
4.11	<i>assets</i> Kontrol . . . . .	20
4.12	Gambar Observasi . . . . .	21

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Sektor pertanian merupakan sumber daya alam yang seharusnya dikelola dengan sebaik-baiknya. Pengelolaan sektor pertanian yang baik dipengaruhi oleh penggunaan teknologi yang tepat guna dan keefektifan dalam pengoperasiannya. Dalam referensi [1] teknologi tepat guna sederhana adalah teknologi yang dibuat atas dasar ketersediaan komponen lokal, dan dapat dikembangkan oleh sumber daya manusia lokal. Namun, pengoperasian teknologi pada sektor pertanian beberapa diantaranya masih memakan waktu yang lama dan menggunakan tenaga kerja manual. Urgensi pengoperasian teknologi yang efektif mempengaruhi produktivitas pertanian, yaitu mempermudah pekerjaan petani sehingga memakan waktu yang tidak lama serta tidak dibutuhkannya lagi tenaga kerja manual.

Berdasarkan hasil wawancara terhadap penanggungjawab program smart farming BPP Lampung, menyatakan bahwa pengolahan lahan di Balai Pelatihan Pertanian (BPP) Lampung mengharuskan tenaga kerja datang ke lokasi untuk melakukan *monitoring* kondisi lahan, diantaranya: pengecekan suhu udara, kelembapan udara, intensitas cahaya, suhu air, suhu tanah, ppm air, pH tanah, pH air, kelembapan tanah, dan tekanan udara menggunakan alat pengukur. Selain *monitoring*, dilakukan juga kontrol sistem penyiraman pada lahan. Sistem *monitoring* dan *controlling* tersebut tergolong tidak efektif dikarenakan masih beroperasi menggunakan tenaga kerja manual sehingga memakan waktu yang lama. Maka daripada itu diperlukannya inovasi yang dapat mendukung keefektifan para petani dalam mengoperasikan teknologi.

Teknologi yang berkembang pesat saat ini adalah penggunaan *smartphone*. Dalam referensi [2] jumlah pengguna *smartphone* di Indonesia mencapai 170,4 juta. 19,6% dari pengguna *smartphone* merupakan petani di Indonesia. Jumlah petani di Indonesia akan terus bertambah mengingat perekonomian nasional sangat bergantung pada sektor pertanian sesuai dengan referensi [3] yang menyatakan bahwa sektor pertanian menyumbang 14,9% dari Produk Domestik Bruto (PDB). Berdasarkan data tersebut, ketersediaan *smartphone* di kalangan petani Indonesia dapat memberikan dampak positif yaitu peningkatan produktivitas pertanian melalui penerapan Teknologi Informasi dan Komunikasi (TIK).

Berdasarkan permasalahan di atas maka dilakukan penelitian pengembangan dengan judul **“Implementasi metode *Rapid Application Development* (RAD) dalam perancangan dan pembuatan aplikasi *mobile KebunQ BPP Lampung*”**. Penelitian ini dilakukan sekaligus untuk membantu program *Low Cost Smart Farming* BPP Lampung. Pemilihan metode RAD pada penelitian ini didasarkan atas ketersediaan waktu pengerjaan yang pendek [4] dan jumlah tim pengembangan yang kecil [5].

## **1.2 Rumusan Masalah**

Berdasarkan latar belakang yang telah dipaparkan, maka rumusan masalahnya adalah bagaimana implementasi metode RAD dalam perancangan dan pembuatan aplikasi *mobile* KEBUNQ?

## **1.3 Tujuan Penelitian**

Adapun tujuan dalam penelitian ini adalah merancang dan membuat aplikasi *mobile* KEBUNQ menggunakan metode RAD

## **1.4 Batasan Masalah**

Adapun batasan masalah dalam penelitian ini adalah penelitian perancangan dan pembuatan aplikasi KEBUNQ ini dibatasi pada pengoperasian aplikasi di sistem operasi android.

## **1.5 Manfaat Penelitian**

Penelitian ini diharapkan dapat bermanfaat bagi BPP Lampung dalam *monitoring* dan *controlling* pengelolaan lahan yang lebih efektif serta berguna dalam berjalannya program *low cost smart farming* BPP Lampung.

## **1.6 Sistematika Penulisan**

Pada penelitian ini peneliti menyusun berdasarkan sistematika penulisan sebagai berikut:

- **BAB I PENDAHULUAN**

Bab ini berisi tentang latar belakang, rumusan masalah, tujuan penelitian, batasan masalah, manfaat penelitian, dan sistematika penulisan.

- **BAB II TINJAUAN PUSTAKA**

Bab ini berisi tentang teori-teori yang mendukung atau berhubungan dengan aplikasi ini.

- **BAB III METODE PENELITIAN**

Bab ini berisi tentang teori-teori yang mendukung atau berhubungan dengan aplikasi ini.

- **BAB IV HASIL IMPLEMENTASI DAN PEMBAHASAN**

Bab ini berisi tentang hasil implementasi dari rancangan penelitian beserta pembahasannya.

- **BAB V KESIMPULAN DAN SARAN**

Bab ini berisi kesimpulan dan saran yang didapatkan dari penelitian ini.



## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Tinjauan Pustaka**

Banyak penelitian sebelumnya yang berkaitan dengan pengembangan, perancangan, dan pembuatan sebuah aplikasi menggunakan *Systems Development Life Cycle* (SDLC) *Rapid Application Development* (RAD). Dalam penelitian yang dikerjakan oleh Meidyan Permata Putri dan Hendra Effendi [6] dihasilkan kesimpulan bahwa penerapan metode RAD sudah dapat memberikan hasil maksimal ditunjukkan dengan sistem yang dibuat dapat memenuhi kebutuhan pengguna. Dalam penelitian yang dikerjakan oleh Abdul Rahman [7] dihasilkan kesimpulan bahwa pembuatan aplikasi pembelajaran daring dengan menggunakan metode RAD bisa diterima dengan baik oleh pengguna dengan presentase sebesar 91% dengan pengujian *User Acceptance Testing* (UAT). Dalam penelitian yang dikerjakan oleh Diah Aryani, Malabay, dan Hani Dewi Ariessanti [8] dihasilkan kesimpulan bahwa penggunaan RAD dapat memudahkan pihak terkait dalam melakukan penelusuran dan pengelolaan umpan balik dari kuesioner dan pengujian UAT menghasilkan 2 nilai yaitu 91% dari pihak kemahasiswaan dan 88% dari pihak mahasiswa yang kedua nilai tersebut menunjukkan penerimaan pengembangan sistem tersebut.

#### **2.2 Dasar Teori**

##### **2.2.1 Monitoring dan Kontrol**

Monitoring adalah proses pengumpulan dan analisis informasi berdasarkan indikator yang ditetapkan secara sistematis dan berkelanjutan tentang kegiatan/program sehingga dapat dilakukan tindakan koreksi untuk penyempurnaan program/kegiatan itu selanjutnya [9]. Sistem kontrol atau sistem kendali adalah kumpulan dari beberapa komponen yang terhubung satu sama lainnya, sehingga membentuk suatu tujuan tertentu yaitu mengendalikan atau mengatur suatu sistem [10].

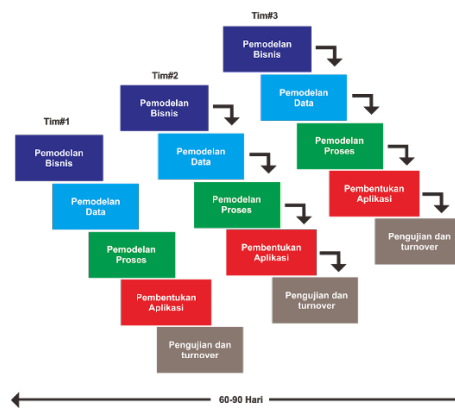
##### **2.2.2 Aplikasi Mobile**

Aplikasi *mobile* adalah program perangkat lunak yang dirancang untuk dijalankan di smartphone, tablet, dan perangkat lain [11]. Urgensi penggunaan Aplikasi yang dikembangkan berbasis mobile adalah semakin meningkatnya

pengguna smartphone yang membutuhkan berbagai macam alat sebagai fasilitator dalam kegiatan sehari-hari. Sebagai fasilitator, aplikasi mobile harus mampu menyediakan informasi agar dapat menjadi sumber data bagi penggunanya dan mampu meningkatkan produktifitas pengguna.

### 2.2.3 *Rapid Application Development (RAD)*

Dalam referensi [4] terdapat lima tahapan dalam model RAD : (1) Pemodelan bisnis, (2) Pemodelan data, (3) Pemodelan proses, (4) Pembentukan aplikasi, (5) Pengujian dan *turnover* .



**Gambar 2.1** Gambar Ilustrasi Model RAD

Berdasarkan Gambar 3, dapat diperhatikan penjabaran sebagai berikut :

a. Pemodelan Bisnis

Pada tahap ini output yang dihasilkan berupa dokumen *Software Requirements Specification* (SRS) yang meliputi informasi ketentuan aplikasi yang akan dibuat. Dokumen tersebut mencakup informasi apa saja

yang harus dibuat, siapa yang harus membuat informasi itu, bagaimana alur informasi itu, proses apa saja yang terkait informasi itu.

b. **Pemodelan Data**

Memodelkan data apa saja yang dibutuhkan berdasarkan pemodelan bisnis dan mendefinisikan atribut-atributnya beserta relasinya dengan data-data yang lain.

c. **Pemodelan Proses**

Mengimplementasikan fungsi bisnis yang sudah didefinisikan terkait dengan pendefinisian data.

d. **Pembuatan Aplikasi**

Mengimplementasikan pemodelan proses dan data menjadi program. Model RAD sangat menganjurkan pemakaian komponen yang sudah ada jika dimungkinkan.

e. **Pengujian dan Pergantian**

Menguji komponen-komponen yang dibuat. Jika sudah teruji maka tim pengembang komponen dapat beranjak untuk mengembangkan komponen berikutnya.

#### **2.2.4 Flutter**

Flutter merupakan sebuah framework aplikasi mobile yang bersifat open source (terbuka) yang diciptakan oleh Google. Flutter dapat digunakan dalam pengembangan aplikasi mobile untuk sistem operasi Android dan iOS, bahkan juga dapat digunakan dalam pengembangan Web ataupun Desktop dari codebase tunggal. Flutter merupakan framework dengan penggunaan Bahasa Dart. Berikut ini beberapa kelebihan dari Flutter [12] :

1. *Package modules* sudah terkoneksi secara otomatis di dalam flutter, sehingga tidak terlalu repot untuk memanggil secara manual melalui terminal
2. Dart menggunakan konsep OOP (*Object Oriented Programming*)
3. *Setup* secara manual jauh lebih mudah, apabila kita memerlukan *library* baru, cukup tambahkan di bagian *pubspec.yaml*
4. Performa cepat dan *smooth*
5. Data management menggunakan state sehingga lebih mudah dalam penggunaannya

6. Adanya fitur *Hot Reload* yang membantu debug lebih cepat
7. Disupport oleh IDE yang sudah familiar dikalangan developer android, seperti Android Studio dan Visual Code

### 2.2.5 *Application Programming Interface (API)*

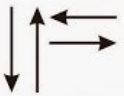






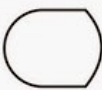
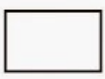
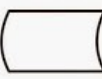



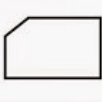
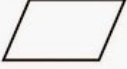

*Application Programming Interface* atau API memungkinkan sebuah perusahaan untuk membuka data dan fungsionalitas aplikasi kepada *developer* pihak ketiga / eksternal, mitra bisnis, dan departemen internal di dalam perusahaan tersebut. Hal ini memungkinkan layanan dan produk untuk berkomunikasi satu sama lain dan memanfaatkan data dan fungsionalitas satu sama lain melalui *interface* yang terdokumentasi. *Developer* tidak perlu tahu bagaimana API diimplementasikan; *developer* hanya menggunakan *interface* untuk berkomunikasi dengan produk dan layanan lain. Penggunaan API telah melonjak selama dekade terakhir, sampai pada tingkat di mana banyak aplikasi web paling populer saat ini tidak akan mungkin tanpa API [13]. Secara tradisional, API merujuk ke *interface* yang terhubung ke aplikasi yang mungkin telah dibuat dengan salah satu bahasa pemrograman tingkat rendah, seperti Javascript. API modern mematuhi prinsip REST dan format JSON dan biasanya dibuat untuk HTTP, menghasilkan *interface* yang ramah kepada *developer* yang mudah diakses dan dipahami secara luas oleh aplikasi yang ditulis dalam Java, Ruby, Python, dan banyak bahasa lainnya.

### 2.2.6 *Database*

*Database* adalah *collection* dari informasi terstruktur yang terorganisir atau data, biasanya disimpan secara elektronik dalam sistem komputer [14]. Sebuah *database* biasanya dikendalikan oleh *Database Management System* (DBMS). Data dan DBMS bersama dengan aplikasi yang terkait dengannya disebut sebagai sistem basis data atau sering disingkat menjadi basis data saja. Data dalam tipe *database* yang paling umum yang beroperasi saat ini biasanya dimodelkan sebagai baris dan kolom dalam serangkaian tabel untuk membuat pemrosesan dan kueri data menjadi efisien. Data kemudian dapat dengan mudah diakses, dikelola, dimodifikasi, diperbarui, dikendalikan, dan diatur. Sebagian besar *database* menggunakan *structured query language* (SQL) untuk menulis dan mengolah data. Salah satu SQL yang sering digunakan adalah MySQL yang merupakan sistem manajemen basis data relasional yang tersedia *open source* sehingga memudahkan pengguna.

### 2.2.7 Flowchart

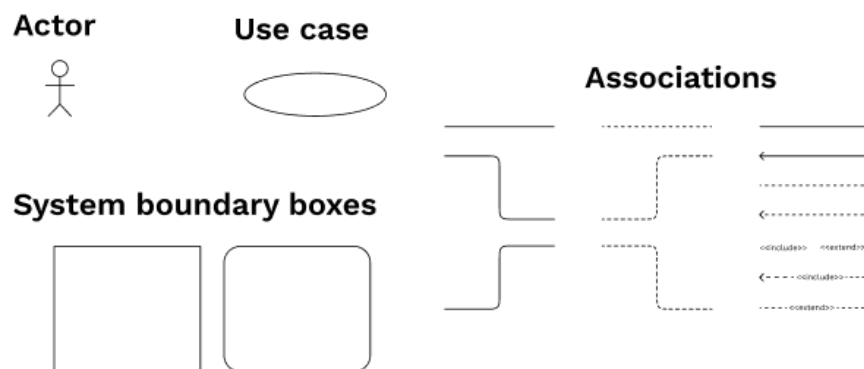
*Flowchart* sering juga disebut dengan *process flowchart*, *process flow diagram* atau diagram alir. *Flowchart* adalah gambaran langkah-langkah terpisah dari suatu proses secara berurutan. *Flowchart* sangat umum digunakan dan dapat diadaptasi untuk berbagai tujuan, dapat digunakan untuk menggambarkan berbagai proses, seperti proses manufaktur, proses administrasi/layanan, atau gambaran rencana proyek [15]. Beberapa kondisi yang disarankan untuk menggunakan *flowchart* yaitu (1) Untuk membangun pemahaman tentang bagaimana suatu proses dilakukan, (2) Untuk mempelajari proses guna melakukan perbaikan, (3) Untuk mengomunikasikan kepada orang lain bagaimana suatu proses dilakukan, (4) Ketika komunikasi yang lebih baik diperlukan antara orang-orang yang terlibat dengan proses yang sama, (5) Untuk mendokumentasikan sebuah proses saat perencanaan proyek. *Flowchart* memiliki simbol-simbol dengan arti tertentu, berdasarkan referensi [16], dapat dilihat simbol-simbol *flowchart* pada gambar di bawah ini.

	<b>Flow Direction symbol</b> Yaitu simbol yang digunakan untuk menghubungkan antara simbol yang satu dengan simbol yang lain. Simbol ini disebut juga connecting line.		<b>Simbol Manual Input</b> Simbol untuk pemasukan data secara manual on-line keyboard
	<b>Terminator Symbol</b> Yaitu simbol untuk permulaan (start) atau akhir (stop) dari suatu kegiatan		<b>Simbol Preparation</b> Simbol untuk mempersiapkan penyimpanan yang akan digunakan sebagai tempat pengolahan di dalam storage.
	<b>Connector Symbol</b> Yaitu simbol untuk keluar - masuk atau penyambungan proses dalam lembar / halaman yang sama.		<b>Simbol Predefine Proses</b> Simbol untuk pelaksanaan suatu bagian (sub-program)/prosedure
	<b>Connector Symbol</b> Yaitu simbol untuk keluar - masuk atau penyambungan proses pada lembar / halaman yang berbeda.		<b>Simbol Display</b> Simbol yang menyatakan peralatan output yang digunakan yaitu layar, plotter, printer dan sebagainya.
	<b>Processing Symbol</b> Simbol yang menunjukkan pengolahan yang dilakukan oleh komputer		<b>Simbol disk and On-line Storage</b> Simbol yang menyatakan input yang berasal dari disk atau disimpan ke disk.
	<b>Simbol Manual Operation</b> Simbol yang menunjukkan pengolahan yang tidak dilakukan oleh computer		<b>Simbol magnetik tape Unit</b> Simbol yang menyatakan input berasal dari pita magnetik atau output disimpan ke pita magnetik.
	<b>Simbol Decision</b> Simbol pemilihan proses berdasarkan kondisi yang ada.		<b>Simbol Punch Card</b> Simbol yang menyatakan bahwa input berasal dari kartu atau output ditulis ke kartu
	<b>Simbol Input-Output</b> Simbol yang menyatakan proses input dan output tanpa tergantung dengan jenis peralatannya		<b>Simbol Dokumen</b> Simbol yang menyatakan input berasal dari dokumen dalam bentuk kertas atau output dicetak ke kertas.

Gambar 2.2 Simbol - simbol *flowchart*

### 2.2.8 Use Case Diagram

*Use case diagram* adalah penggambaran yang dibuat untuk menunjukkan interaksi aktor atau pengguna dengan sistem yang dirancang, bertujuan untuk memudahkan orang lain dalam membaca informasi tersebut [17, 18]. Berdasarkan referensi [17] terdapat dua fungsi utama dari *use case* dalam sebuah sistem yang dibuat, yaitu untuk memperlihatkan urutan aktivitas proses dan untuk menggambarkan *bussines process*. Ada beberapa komponen *use case* [17, 19] yaitu, aktor, *use case*, sistem, dan relasi. (1) Aktor merupakan pengguna komponen sistem untuk melakukan sesuatu dan berasal dari luar sistem. Aktor dapat berupa perangkat, manusia, atau sistem lain yang berperan. (2) *Use case* merupakan gambaran fungsional dari sebuah sistem. (3) Sistem merupakan komponen yang menyatakan batasan dari sistem dan memiliki relasi dengan aktor sebagai penggunanya. dan (4) *Association* merupakan teknik yang digunakan menunjukkan interaksi antara komponen aktor dan *use case* tertentu dan digambarkan dengan garis penghubung.



Gambar 2.3 Komponen *Use case*

### 2.2.9 Sequence Diagram

*Sequence diagram* adalah suatu diagram yang menggambarkan interaksi antar objek dan mengindikasikan komunikasi diantara objek-objek tersebut [20]. Diagram ini sekaligus menunjukkan serangkaian pesan yang diperlukan oleh objek-objek yang aktif. Objek tersebut dibaca dan diurutkan dari kiri ke kanan, aktor yang menginisiasi interaksi ditaruh pada bagian paling kiri dalam diagram. Dimensi vertikal pada diagram ini menunjukkan representasi waktu. Bagian paling atas menjadi titik awal dan waktu berjalan ke bawah. Garis vertikal tersebut disebut *lifeline*, diposisikan pada setiap objek atau aktor. *lifeline* digambarkan menjadi kotak (*activation box*) ketika objek melakukan sebuah operasi sekaligus disebut objek mempunyai *live activation*. Pesan antar objek digambarkan dengan sebuah panah yang di atasnya diberikan label keterangan.

#### **2.2.10 *Black Box Testing***

Pengujian *black box* adalah salah satu metode pengujian *software* yang berfokus pada bagian fungsionalitas, terkait *input* dan *output* yang ada apakah sudah sesuai dengan yang dituju atau belum. Tahan pengujian atau *testing* merupakan hal wajib dalam siklus pengembangan perangkat lunak [21].

#### **2.2.11 *User Acceptance Testing (UAT)***

User Acceptance Testing (UAT) adalah pengujian langsung oleh user pada pengujian akhir, dan saat berlangsungnya pengujian dilakukan pembuatan dokumen guna sebagai bukti penerimaan sistem [22].

#### **2.2.12 *Skala Likert***

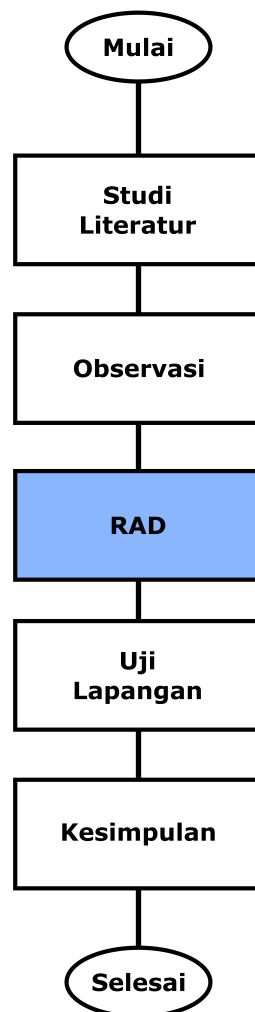
Skala *Likert* adalah skala yang digunakan untuk mengukur sikap, pendapat, dan persepsi seseorang atau sekelompok orang tentang sesuatu yang diamati. Semua pilihan jawaban diberi skor, maka responden harus menggambarkan, mendukung pernyataan (positif) atau tidak mendukung pernyataan (negatif) [23].

## **BAB III**

### **METODE PENELITIAN**

#### **3.1 Alur Penelitian**

Langkah-langkah penelitian yang akan dilakukan sebagai bagian dari penelitian ini dapat dilihat pada Gambar 3.1



**Gambar 3.1** Alur Penelitian

#### **3.2 Penjabaran Langkah Penelitian**

Berikut ini merupakan prosedur penelitian yang dilakukan.



### 3.2.1 Studi Literatur

Mencari dan mengumpulkan referensi yang berkaitan dengan penelitian melalui media buku, jurnal dan e-book.

### 3.2.2 Observasi

Melakukan pengamatan di Balai Pelatihan Pertanian (BPP) Lampung terkait sistem pengolahan lahan cabai dan greenhouse.

### 3.2.3 RAD

Melakukan perancangan dan pembuatan aplikasi KEBUNQ dengan mengikuti langkah proses yang tercantum dalam *Rapid Application Development* (RAD). Pada langkah ini dilakukan lima tahapan yaitu, (1) Pemodelan Bisnis, (2) Pemodelan Data, (3) Pemodelan Proses, (4) Pembuatan Aplikasi, dan (5) Pengujian.

### 3.2.4 Uji Lapangan

Melakukan pengujian aplikasi KEBUNQ dengan alat yang terpasang pada lahan.

### 3.2.5 Kesimpulan

Melakukan analisa dan menulis kesimpulan dari penelitian ini.

## 3.3 Alat dan Bahan Tugas Akhir

### 3.3.1 Alat

Alat yang digunakan dalam penelitian ini.

1. Macbook Pro (13-inch, 2016, *Four Thunderbolt 3 Ports*) dengan OS Monterey Version 12.3.1 (21E258), *processor* 2,9 GHz Dual-Core Intel Core i5, *memory* 8 GB 2133 MHz LPDDR3, *graphics* Intel Iris Graphics 550 1536 MB
2. *Smartphone* dengan spesifikasi minimum OS Android 6.0 (*marshmallow*). Pada penelitian ini digunakan untuk melakukan *testing* dalam proses pembuatan aplikasi
3. Visual Studi Code digunakan sebagai *code editor* dalam pemrograman
4. Postman digunakan sebagai alat bantu dalam melakukan *testing* API
5. Figma dan Inkscape digunakan sebagai alat dalam pembuatan *User Interface Layout* dan *assets*

### 3.3.2 Bahan

1. Dokumen *Software Requirements Specification* sebagai standar dan batasan dalam pengembangan aplikasi KEBUNQ
2. Data Kuesioner yang diisi saat pengujian aplikasi

### 3.4 Metode Tugas Akhir

Metode yang digunakan dalam pengerjaan tugas akhir ini

1. Metode pengembangan yang digunakan adalah *Rapid Application Development* (RAD)
2. Cara pengumpulan data yang digunakan adalah kuesioner dan pengujian

### 3.5 Ilustrasi Perhitungan Metode

Pengujian penerimaan aplikasi oleh pengguna dilakukan dengan menggunakan metode pengujian UAT, skala *likert* digunakan untuk menilai bobot dari hasil pengisian kuesioner. Bobot jawaban kuesioner [24] dapat dilihat pada Tabel 3.1

**Tabel 3.1** Bobot nilai jawaban

Jawaban	Bobot
Sangat Setuju	5
Setuju	4
Netral	3
Tidak Setuju	2
Sangat Tidak Setuju	1

setelah dilakukan pengisian kuesioner dan didapatkan data responden, data kemudian diolah dengan cara mengalikan data jawaban dengan bobot yang tertera pada tabel 3.1. Kemudian hasil tersebut dijumlahkan dan hasilnya dikalikan

dengan bobot jawaban :

- Jumlah skor responden Sangat Setuju (SS) = Total x 5 = nilai SS
- Jumlah skor responden Setuju (S) = Total x 4 = nilai S
- Jumlah skor responden Netral (N) = Total x 3 = nilai N
- Jumlah skor responden Tidak Setuju (TS) = Total x 2 = nilai TS
- Jumlah skor responden Sangat Tidak Setuju (STS) = Total x 1 = nilai STS

Maka akan diketahui jumlah skor total = nilai SS + nilai S + nilai N + nilai TS + nilai STS

Setelah didapatkan jumlah skor total, selanjutnya mencari nilai terendah dan tertinggi. Misalnya ada 5 orang responden, maka dapat dihitung:

- Nilai terendah = 5 x jumlah pertanyaan x 1 = (jika jawaban STS semua)
- Nilai tertinggi = 5 x jumlah pertanyaan x 5 = (jika jawaban SS semua)

Setelah diperoleh skor total, maka penilaian tingkat penerimaan oleh pengguna pada aplikasi dapat dihitung dengan menggunakan rumus berikut [24]:

$$P = \frac{f}{n} \times 100\% \quad (\text{III.1})$$

Keterangan :

P = Presentase

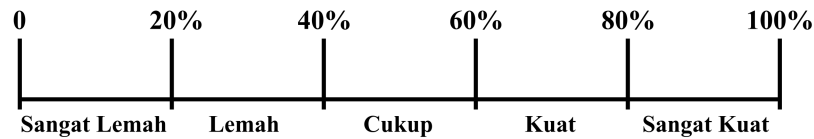
n = Jumlah responden

f = Frekuensi jawaban

**Tabel 3.2** Nilai Presentase

No	Nilai	Pernyataan	kode
1	80% - 100%	Sangat Setuju	SS
2	60% - 89,99%	Setuju	S
3	40% - 59,99%	Netral	N
4	20% - 39,99%	Tidak Setuju	TS
5	0% - 19,99%	Sangat Tidak Setuju	STS

Hasil pengujian UAT merepresentasikan pengujian penerimaan oleh pengguna, berdasarkan pengujian penerimaan tersebut ditarik kesimpulan apakah aplikasi yang telah diuji dapat diterima atau tidaknya oleh pengguna. Presentase yang sudah diolah diklasifikasikan berdasarkan skala penilaian berikut [24] :



**Gambar 3.2** Skala Penilaian

Berikut nilai kesimpulan berdasarkan referensi [24] :

**Tabel 3.3** Nilai Kesimpulan

<b>Nilai</b>	<b>Pernyataan</b>
0 % - 20%	Sangat Lemah
21% - 40%	Lemah
41% - 60%	Cukup
61% - 80%	Kuat
81% - 100%	Sangat Kuat

Jika nilai akhir dari perhitungan ternyata didapatkan hasil desimal, maka dapat dibulatkan ke nilai terdekat.

### 3.6 Rancangan Pengujian

Terdapat dua pengujian yang digunakan dalam penelitian ini. Pertama pengujian fungsionalitas dengan menggunakan metode *black box testing*, dan yang kedua menggunakan metode pengujian *User Acceptance Testing* (UAT) guna mengetahui tingkat penerimaan pengguna. Kedua pengujian tersebut akan dirancang komponen pengujiannya setelah penelitian sudah melewati proses pemodelan proses yakni tahap sebelum tahap pembuatan aplikasi dalam penelitian ini.

## BAB IV

### HASIL PENELITIAN DAN PEMBAHASAN

#### 4.1 Hasil Penelitian

##### 4.1.1 Penerapan RAD

###### 4.1.1.1 Pemodelan Bisnis

Berdasarkan hasil analisa ditentukan bahwa aplikasi KEBUNQ dirancang untuk satu jenis pengguna. Sehingga tidak terdapat level pengguna pada akses aplikasi. Berikut analisa kebutuhan pengguna

1. Pengguna harus melakukan login
2. Pengguna dapat melihat daftar dan status nyala alat
3. Pengguna dapat melihat detail alat yang terdiri dari data sensor dan kontrol yang tersedia
4. Pengguna dapat melakukan kontrol pada alat yang dipilih
5. Pengguna dapat melakukan setting otomatis pada suatu kontrol yang tersedia mode otomatisnya

###### 4.1.1.2 Pemodelan Data



**Gambar 4.1** Skema Tabel *Database*

###### 4.1.1.3 Pemodelan Proses

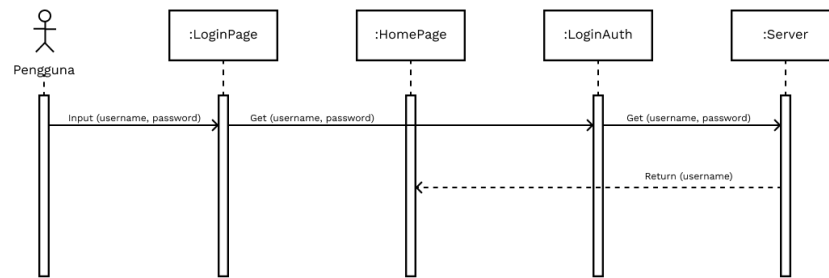
###### a. *Use Case*

*Use case* perancangan aplikasi mobile KEBUNQ BPP Lampung

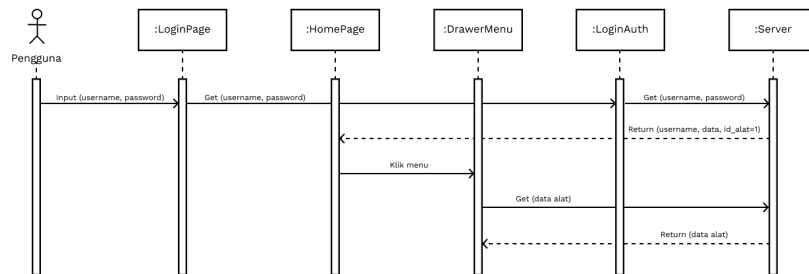


**Gambar 4.2** *Use Case Diagram*

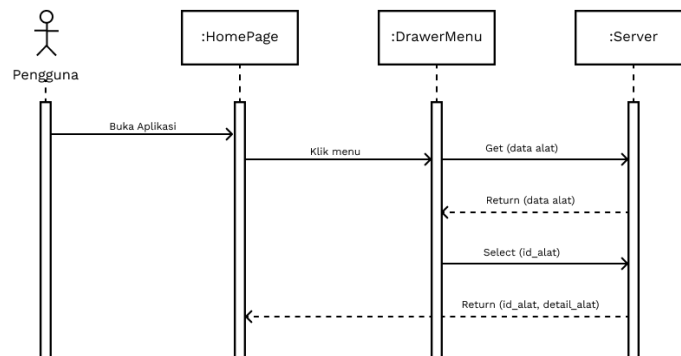
b. *Sequence Diagram*



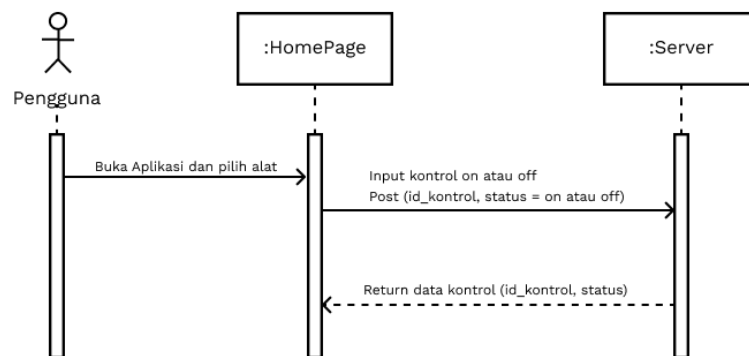
**Gambar 4.3** *Sequence Diagram Login*



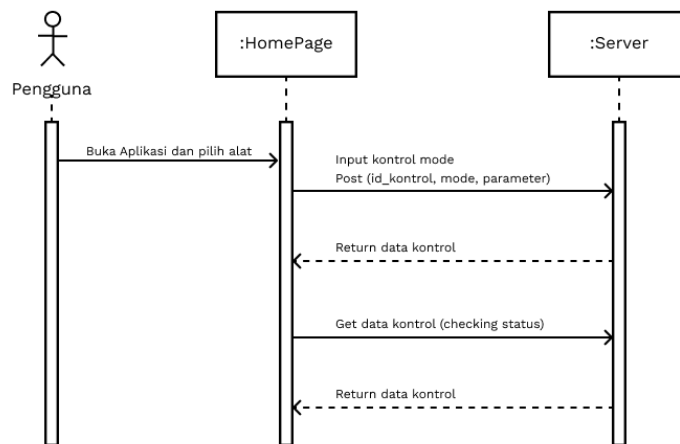
**Gambar 4.4** *Sequence Diagram Melihat Menu / Alat*



**Gambar 4.5** *Sequence Diagram Melihat Detail Alat*



**Gambar 4.6** *Sequence Diagram* Melakukan Kontrol

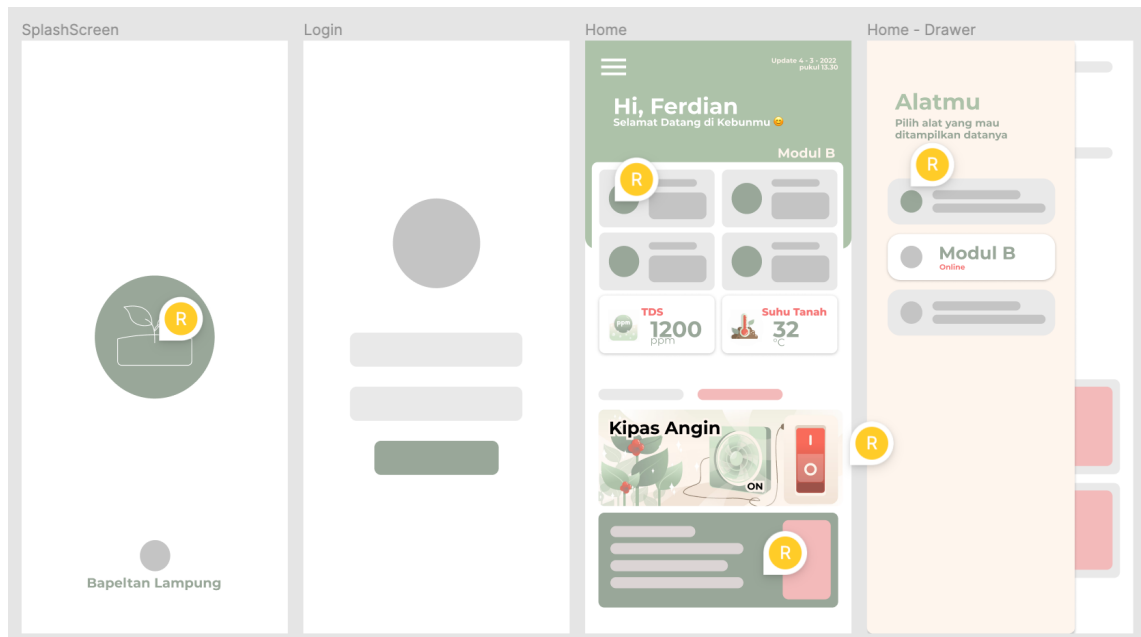


**Gambar 4.7** *Sequence Diagram* Melakukan Kontrol Automatis

#### 4.1.1.4 Pembuatan Aplikasi

Dalam proses pembuatan aplikasi peneliti melakukan 3 pengerjaan (1) Perancangan desain layout *User Interface*, (2) Pembuatan *assets* mencakup logo aplikasi, ikon sensor, dan ilustrasi kontrol, dan (3) Pengerjaan pembuatan aplikasi menggunakan *framework* Flutter dengan bahasa pemrograman Dart

##### 1. Perancangan Desain Layout *User Interface*



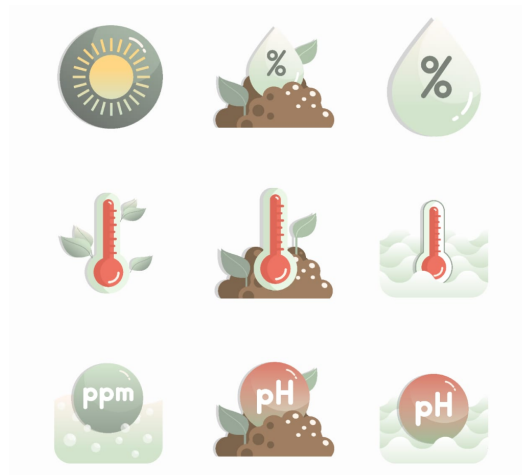
**Gambar 4.8** Rancangan layout UI

## 2. Pembuatan *assets*

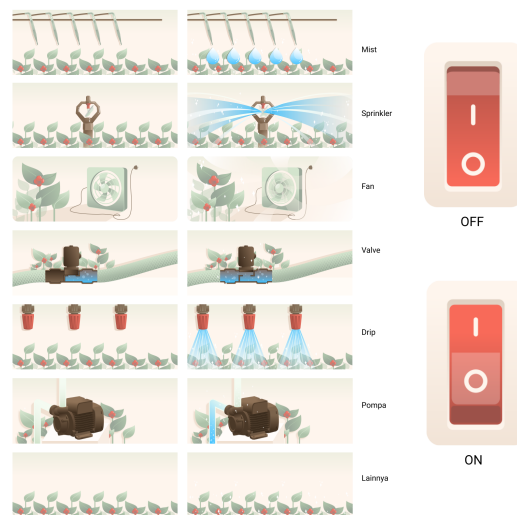


**Gambar 4.9** *assets* Logo aplikasi





**Gambar 4.10** *assets* Ikon Sensor



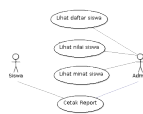
**Gambar 4.11** *assets* Kontrol

### 3. Pembuatan Aplikasi

#### 4.1.1.5 Pengujian dan Pergantian

### 4.2 Uji Lapangan

#### 4.2.1 Data Hasil Observasi



**Gambar 4.12** Gambar Observasi

Peneliti mengamati kebutuhan sensor dan kontrol yang diperlukan petani untuk diaplikasikan pada lahannya.

### 4.3 Analisis Hasil Penelitian

## **BAB V**

### **KESIMPULAN DAN SARAN**

**5.1 Kesimpulan**

**5.2 Saran**

## DAFTAR PUSTAKA

- [1] BPTP Balitbangtan Jambi, "Alat dan Mesin Pertanian Tepat Guna untuk Tanaman Padidalam Mendukung Program Peningkatan Produksi Beras Nasional (P2BN)" 2020, Available: <http://jambi.litbang.pertanian.go.id/ind/images/PDF/Kiki1.pdf>
- [2] Y. Pusparisa, "Daftar Negara Pengguna smartphone Terbanyak, Indonesia urutan berapa?: Databoks," *Databoks Pusat Data Ekonomi dan Bisnis Indonesia*, 01-Jul-2021. [Online]. Available: <https://databoks.katadata.co.id/datapublish/2021/07/01/daftar-negara-pengguna-smartphone-terbanyak-indonesia-urutan-berapa>. [Accessed: 06-Feb-2022].
- [3] Direktorat Pangan dan Pertanian, "Studi Pendahuluan Rencana Pembangunan Jangka Menengah Nasional (RPJMN) Bidang Pangan dan Pertanian 2015 – 2016", Direktorat Pangan dan Pertanian Kementerian Perencanaan Pembangunan Nasional/Badan Perencanaan Pembangunan Nasional, 2013.
- [4] Sukamto, R. A. dan Shalahudin, M.. *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung : Informatika Bandung, 2016.
- [5] P. Beynon-Davies, C. Carne, H. Mackay, and D. Tudhope, "Rapid application development (RAD): An empirical review," *European Journal of Information Systems*, vol. 8, no. 3, pp. 212, 1999.
- [6] M. P. Putri and H. Effendi, "Implementasi metode RAD Pada website service guide 'Tour waterfall south sumatera'," *Jurnal Sisfokom (Sistem Informasi dan Komputer)*, vol. 7, no. 2, pp. 130–136, Sep. 2018.
- [7] A. Rahman, "Rapid Application Development Sistem Pembelajaran Daring Berbasis Android," *Jurnal Intech*, vol. 1, no. 2, pp. 20-25, Nov. 2020.
- [8] D. Aryani, Malabay, H. D. Ariessanti, "Penerapan Rapid Application Development (RAD) Pada Perancangan Aplikasi Tracer Study Berbasis Android," *eDikInformatika*, vol. 7, no. 1, pp. 111-122, Oct. 2020.
- [9] Hikmat, Dr. Harry. 2010. *Monitoring dan Evaluasi Proyek*.
- [10] Katsuhiko Ogata. "Teknik kontrol otomatis", PT penerbit erlangga-SIMON. &SCHUTER (ASUA) Pte.ltd., 1997.

- [11] N. Serrano, J. Hernantes and G. Gallardo. Mobile Web Apps. *IEEE Software*, vol. 30, no. 5, 2013, pp. 22 -27.
- [12] Ganda, Yusmi P. W., Happy Flutter. Cetakan 1. Tangerang Selatan : Al Qolam, 2019.
- [13] IBM Cloud Education, “What is an application programming interface (API),” *IBM*, 19-Aug-2020. [Online]. Available: <https://www.ibm.com/cloud/learn/api>. [Accessed: 11-Feb-2022].
- [14] OCI, “What is a database?,” *Oracle*, 2022. [Online]. Available: <https://www.oracle.com/database/what-is-database/>. [Accessed: 11-Feb-2022].
- [15] “What is a flowchart?,” *ASQ*. [Online]. Available: <https://asq.org/quality-resources/flowchart>. [Accessed: 15-Feb-2022].
- [16] D. Rizky, “Jenis Flowchart Dan Simbol-Simbolnya,” *Medium*, 30-Apr-2019. [Online]. Available: <https://medium.com/dot-intern/jenis-flowchart-dan-simbol-simbolnya-ef6553c53d73>. [Accessed: 20-Feb-2022].
- [17] M. R. Adani, “Use case diagram: Pengertian, Fungsi, Teknik, Dan Contoh,” *Sekawan Media — Software House & System Integrator Indonesia*, 21-Jun-2021. [Online]. Available: <https://www.sekawanmedia.co.id/blog/use-case-diagram/>. [Accessed: 20-Feb-2022].
- [18] Lucid Software, “UML use case diagram tutorial,” *Lucidchart*. [Online]. Available: <https://www.lucidchart.com/pages/uml-use-case-diagram>. [Accessed: 24-Feb-2022].
- [19] F. Mandrelli, “UML use case diagram,” *Figma*, 2021. [Online]. Available: <https://www.figma.com/community/file/986330591099819762>. [Accessed: 24-Feb-2022].
- [20] R. Habibi and R. Aprilian, Tutorial dan Penjelasan Aplikasi E-Office Berbasis Web Menggunakan Metode RAD, Bandung: Kreatif Industri Nusantara, 2020.
- [21] Iskandaria. 2012. Contoh Pengujian Black Box
- [22] Mutiara, A. B., Awaludin, R., Muslim, A. and T. Oswari, “Testing Implementasi Website Rekam Medis Elektronik Opeltgunasys Dengan Metode Acceptance Testing,” 2014.

- [23] Sugiyono, *Memahami Penelitian Kualitatif*. Bandung, Indonesia: Alfabeta, 2012.
- [24] Riduwan, *Belajar mudah penelitian untuk guru-karyawan dan peneliti pemula*. Bandung, Indonesia: Alfabeta, 2009.

# LAMPIRAN

**Coding****Login****Aplikasi**