

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, Mei.2022
Penulis,

Rizki Juliansyah

Diperiksa dan disetujui oleh,

Pembimbing

Tanda Tangan

1. Mohamad Idris S.Si., M.Sc.,
NIP. 198610102019031016

.....

Pengujii

Tanda Tangan

1. Andika Setiawan S.Kom., M.Cs.
NIP/NRK.
2. Winda Yulita S.Pd., M.Cs.,
NIP/NRK.

.....

Disahkan oleh,

Koordinator Program Studi Teknik Informatika

Jurusan Teknologi, Produksi dan Industri

Institut Teknologi Sumatera

Ir. Hira Laksmiwati Soemitra M.Sc.
NIP. 195201091985032001

HALAMAN PERNYATAAN ORISINALITAS

Tugas Akhir dengan judul "**IMPLEMENTASI METODE RAPID APPLICATION DEVELOPMENT (RAD) DALAM PERANCANGAN DAN PEMBUATAN APLIKASI MOBILE KEBUNQ BPP LAMPUNG**" adalah karya saya sendiri, dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan benar.

Nama : Rizki Juliansyah
NIM : 14116151

Tanda Tangan :
Tanggal :

LEMBAR PERSETUJUAN

Sebagai civitas akademik Institut Teknologi Sumatera, saya yang bertanda tangan di bawah ini:

Nama : Rizki Juliansyah

NIM : 14116151

Program Studi : Teknik Informatika

Jurusan : Jurusan Teknologi, Produksi dan Industri

Jenis Karya : Tugas Akhir

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Institut Teknologi Sumatera **Hak Bebas Royalti Noneksklusif** (*Non-exclusive Royalty Free Right*) atas karya ilmiah saya yang berjudul:

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

Berserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini, Institut Teknologi Sumatera berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan skripsi saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di Lampung Selatan

Pada tanggal Mei 2022

Yang menyatakan

Rizki Juliansyah

PERSEMPAHAN

KATA PENGANTAR

Puji syukur kehadirat 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:

- 1.
- 2.

Lampung Selatan,
Mei 2022

Rizki Juliansyah

ABSTRAK

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

Rizki Juliansyah

Sektor pertanian merupakan sumber daya alam yang seharusnya dikelola dengan sebaik-baiknya. 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. Balai Pelatihan Pertanian (BPP) Lampung juga masih mengharuskan tenaga kerja datang ke lokasi untuk melakukan *monitoring* dan kontrol kondisi lahan. Ketersediaan *smartphone* di kalangan petani dapat memberikan dampak positif yaitu peningkatan produktivitas pertanian melalui penerapan Teknologi Informasi dan Komunikasi. Berdasarkan hal tersebut dilakukan perancangan dan pembuatan aplikasi KEBUNQ. *System Development Life Cycle* (SDLC) yang digunakan adalah *Rapid Application Development* (RAD). Pengujian yang dilakukan menggunakan metode *black box testing* untuk menguji fungsionalitas aplikasi, dan *User Acceptance Testing* (UAT) untuk mengetahui tingkat penerimaan pengguna. Pada penelitian ini, *black box* didapatkan satu bagian yang belum sesuai dengan yang diharapkan. Kemudian pada pengujian UAT, didapatkan hasil 89,5% yang menunjukkan tingkat penerimaan yang sangat kuat oleh pengguna.

Kata Kunci: Pertanian, *Smartphone*, *Rapid Application Development*, *Black box testing*, *User Acceptance Testing*

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

| | |
|--|-------------|
| LEMBAR PENGESAHAN | ii |
| LEMBAR PERNYATAAN | iii |
| LEMBAR PERSETUJUAN | iv |
| PERSEMBAHAN | v |
| KATA PENGANTAR | vi |
| ABSTRAK | vii |
| ABSTRACT | viii |
| DAFTAR ISI | x |
| DAFTAR TABEL | xi |
| DAFTAR GAMBAR | xiii |
| I PENDAHULUAN | 1 |
| 1.1 Latar Belakang | 1 |
| 1.2 Rumusan Masalah | 2 |
| 1.3 Tujuan Penelitian | 2 |
| 1.4 Batasan Masalah | 2 |
| 1.5 Manfaat Penelitian | 3 |
| 1.6 Sistematika Penulisan | 3 |
| II LANDASAN TEORI | 4 |
| 2.1 Tinjauan Pustaka | 4 |
| 2.2 Dasar Teori | 4 |
| 2.2.1 <i>Monitoring</i> dan Kontrol | 4 |
| 2.2.2 Aplikasi <i>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) | 6 |
| 2.2.6 <i>Database</i> | 7 |

| | | |
|---|--------------------------------------|-----------|
| 2.2.7 | <i>Flowchart</i> | 7 |
| 2.2.8 | <i>Use Case Diagram</i> | 8 |
| 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 | Skala <i>Likert</i> | 10 |
| III METODE PENELITIAN | | 11 |
| 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 |
| IV HASIL PENELITIAN DAN PEMBAHASAN | | 16 |
| 4.1 | Hasil Penelitian | 16 |
| 4.1.1 | Hasil Observasi | 16 |
| 4.1.2 | Hasil Penerapan RAD | 19 |
| 4.1.3 | Uji Lapangan | 42 |
| 4.2 | Pembahasan | 45 |
| V KESIMPULAN DAN SARAN | | 46 |
| 5.1 | Kesimpulan | 46 |
| 5.2 | Saran | 46 |
| DAFTAR PUSTAKA | | 47 |
| LAMPIRAN | | 50 |

DAFTAR TABEL

| | | |
|-----|--|----|
| 3.1 | Bobot nilai jawaban | 13 |
| 3.2 | Nilai Presentase | 14 |
| 3.3 | Nilai Kesimpulan | 15 |
| 4.1 | Pengujian Halaman <i>Login</i> | 41 |
| 4.2 | Pengujian Halaman <i>Home</i> | 41 |
| 4.3 | Kuesioner | 43 |
| 4.4 | Jumlah Hasil Kuesioner | 43 |

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 | Ilustrasi <i>Mist</i> | 17 |
| 4.2 | Ilustrasi <i>Sprinkler</i> | 17 |
| 4.3 | Ilustrasi <i>Fan</i> | 18 |
| 4.4 | Bentuk <i>Valve</i> | 18 |
| 4.5 | Ilustrasi <i>Drip</i> | 19 |
| 4.6 | Pompa | 19 |
| 4.7 | Jadwal Pengerjaan | 20 |
| 4.8 | Skema Tabel <i>Database</i> | 21 |
| 4.9 | API <i>Summary</i> Pengguna | 22 |
| 4.10 | API Data <i>Realtime</i> | 23 |
| 4.11 | API Data Kontrol | 24 |
| 4.12 | API Status Alat | 25 |
| 4.13 | <i>flowchart</i> bagian 1 (mulai) | 26 |
| 4.14 | <i>flowchart</i> bagian 2 (kontrol) | 27 |
| 4.15 | <i>flowchart</i> bagian 3 (selesai) | 27 |
| 4.16 | <i>Use Case Diagram</i> | 28 |
| 4.17 | <i>Sequence Diagram Login</i> | 28 |
| 4.18 | <i>Sequence Diagram</i> Melihat Menu / Alat | 29 |
| 4.19 | <i>Sequence Diagram</i> Melihat Detail Alat | 29 |
| 4.20 | <i>Sequence Diagram</i> Melakukan Kontrol | 30 |
| 4.21 | <i>Sequence Diagram</i> Melakukan Kontrol Automatis | 30 |
| 4.22 | Rancangan layout UI | 31 |
| 4.23 | <i>assets</i> Logo aplikasi | 31 |
| 4.24 | <i>assets</i> Ikon Sensor | 32 |
| 4.25 | <i>assets</i> Kontrol | 33 |
| 4.26 | Tampilan <i>Splash Screen</i> | 34 |
| 4.27 | Tampilan Halaman <i>Login</i> | 35 |
| 4.28 | Tampilan Halaman <i>Home</i> | 36 |
| 4.29 | Tampilan Menu <i>Drawer</i> | 37 |

| | |
|--|----|
| 4.30 Tampilan <i>Loading</i> | 38 |
| 4.31 Rancangan Alat Secara Umum | 39 |
| 4.32 Rancangan Skematik Alat Lahan Cabai | 39 |
| 4.33 Alat Terpasang di Lahan | 40 |
| 4.34 Skala penilaian | 45 |

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. Penggunaan teknologi yang tepat guna dipengaruhi oleh beberapa aspek lokal, diantaranya adalah aspek lingkungan, aspek sosial (sumber daya manusia lokal), dan aspek ekonomi masyarakat [1, 2]. 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.

Smartphone merupakan teknologi yang berkembang pesat saat ini. Dalam referensi [3] jumlah pengguna smartphone di Indonesia mencapai 170,4 juta. Jumlah petani per 2019 dalam catatan Badan Pusat Statistik (BPS) mencapai 33,4 juta orang [4]. Jumlah petani di Indonesia akan terus bertambah mengingat perekonomian nasional sangat bergantung pada sektor pertanian sesuai dengan referensi [5] yang menyatakan bahwa 14,9% dari Produk Domestik Bruto (PDB) berasal dari sektor pertanian. Berdasarkan data tersebut, ketersediaan *smartphone* di kalangan petani Indonesia dapat memberikan dampak positif yaitu peningkatan

produktivitas pertanian melalui penerapan Teknologi Informasi dan Komunikasi (TIK).

Perancangan dan pembuatan sebuah *software* atau aplikasi tidak lepas dari penerapan *Software Development Life Cycle* (SDLC). SDLC bertujuan untuk memaksimalkan output *software* yang dibuat, baik itu dari segi kualitas yang tinggi sesuai dengan ekspektasi stakeholder dan para pengguna [6] dikarenakan SDLC berperan dalam perencanaan, kontrol, transparansi, meminimalisir risiko, dan biaya dalam sebuah pengerjaan proyek [7]. Ada banyak jenis SDLC yang dapat diterapkan dengan menimbang kebutuhan dan keadaan pengembang, diantaranya *waterfall* [8], *Agile* [9], dan *Rapid Application Development* (RAD) [10]. Berdasarkan referensi [10] RAD merupakan metode pengembangan yang cepat dikarenakan pengerjaan dapat dilakukan secara paralel oleh tim lain atau anggota lain dalam tim.

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 [10] dan jumlah tim pengembangan yang kecil [11].

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan, maka rumusan masalahnya adalah sebagai berikut:

1. Bagaimana implementasi metode RAD dalam perancangan dan pembuatan aplikasi *mobile* KEBUNQ?
2. Bagaimana pengujian yang dilakukan pada aplikasi KEBUNQ?

1.3 Tujuan Penelitian

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

1.4 Batasan Masalah

Adapun batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Penelitian perancangan dan pembuatan aplikasi KEBUNQ ini dibatasi pada

pengoperasian aplikasi di sistem operasi android.

2. Lahan yang diamati dan diimplementasikan aplikasi KEBUNQ dalam penelitian ini terbatas pada lahan cabai besar, lahan cabai kecil, dan *screen* hidroponik di Balai Pelatihan Pertanian Lampung.

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 [12] dihasilkan kesimpulan bahwa penerapan metode RAD sudah dapat memberikan hasil maksimal ditunjukkan dengan terpenuhinya kebutuhan pengguna. Dalam penelitian yang dikerjakan oleh Abdul Rahman [13] dihasilkan kesimpulan bahwa aplikasi yang telah dibuat menggunakan RAD menunjukkan penerimaan yang baik dari pengguna ditunjukkan dengan persentase pengujian *User Acceptance Testing (UAT)* sebesar 91%. Dalam penelitian yang dikerjakan oleh Diah Aryani, Malabay, dan Hani Dewi Ariessanti [14] 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 dibuat.

2.2 Dasar Teori

2.2.1 Monitoring dan Kontrol

Sebuah program yang dijalankan secara berkelanjutan memerlukan pemantauan yang lebih detail terhadap proses pengumpulan data dan analisis informasi. Pemantauan ini bertujuan untuk menyempurnakan program sehingga dapat meminimalisir kesalahan. Dalam referensi [15] proses pemantauan ini disebut monitoring. Upaya untuk memonitor program agar berjalan sesuai tujuan maka diperlukannya suatu tindakan pengendalian. Dalam referensi [16] sistem ini disebut sistem kontrol.

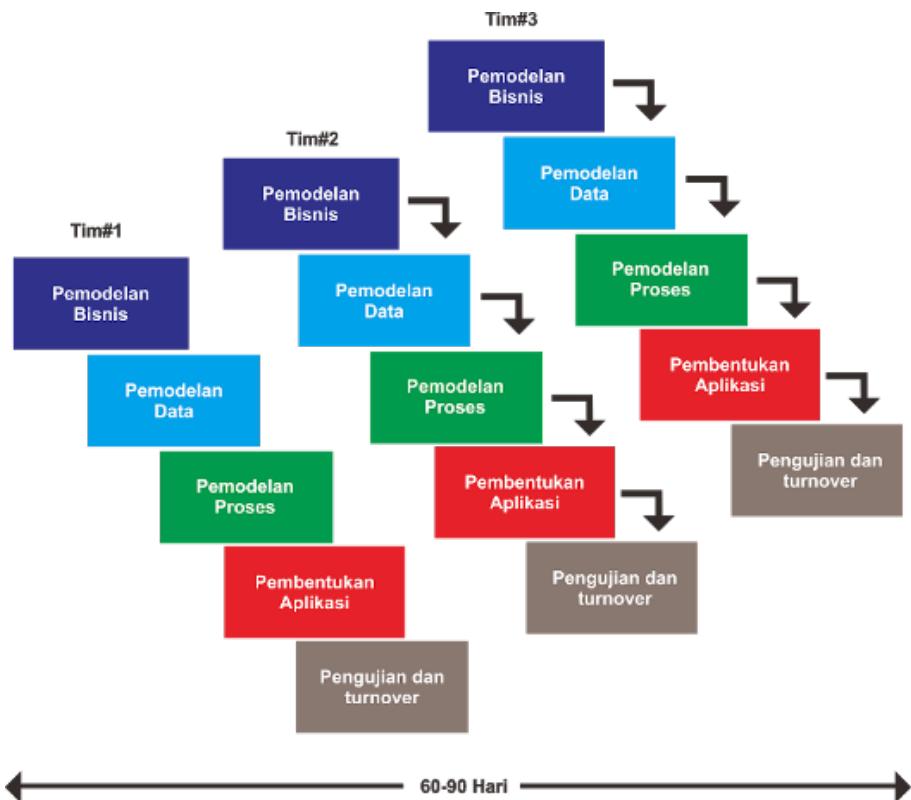
2.2.2 Aplikasi *Mobile*

Program perangkat lunak yang dirancang untuk dijalankan di *smartphone*, tablet, dan perangkat lain disebut aplikasi *mobile* [17]. 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 [10] terdapat lima tahapan yang terdapat pada model RAD, yaitu sebagai berikut:



Gambar 2.1 Gambar Ilustrasi Model RAD

Berdasarkan Gambar 2.1, dapat diperhatikan penjabaran sebagai berikut :

a. Pemodelan Bisnis

Pemodelan bisnis adalah serangkaian informasi mengenai produk yang dirancang. dalam penelitian ini, informasi tersebut dicantumkan dalam dokumen *Software Requirements Specification* (SRS) agar dapat divalidasi oleh *stakeholder*.

b. Pemodelan Data

Pemodelan data adalah kegiatan merancang dan menghubungkan sekumpulan data berdasarkan pemodelan bisnis.

c. Pemodelan Proses

Pemodelan proses adalah kegiatan mengkonversi pemodelan data menjadi serangkaian alur proses yang bertujuan untuk menggambarkan fungsi bisnis.

d. Pembuatan Aplikasi

Pembuatan aplikasi adalah implementasi fungsi bisnis dan model data kedalam bahasa pemrograman. *Output* yang dihasilkan dari pembuatan aplikasi ini adalah sebuah prototipe.

e. Pengujian dan Pergantian

Pengujian dan pergantian adalah kegiatan uji fitur yang tersedia dalam prototipe untuk mengetahui kesesuaian prototipe dengan standar yang diharapkan.

2.2.4 Flutter

Dalam referensi [18] flutter merupakan framework yang menggunakan bahasa pemrograman dart yang bersifat *open source* (terbuka) dari Google. Flutter dapat digunakan dalam pengembangan aplikasi *cross-platform* untuk penggunaan pada sistem operasi Android maupun iOS, bahkan juga dapat digunakan dalam pengembangan Web ataupun Desktop dari codebase tunggal. Beberapa kelebihan dari flutter adalah sudah terkoneksinya *Package modules* menjadikan developer tidak harus melakukan proses manual melalui terminal, sudah menggunakan konsep *Object Oriented Programming* (OOP), sistem penggunaan *library* yang mudah dengan hanya melakukan perubahan pada file puspec.yaml, cepatnya performa dan ketersediaan *hot reload* yang sangat menyingkat waktu dalam proses *debug*, adanya *state management* sangat memudahkan penggunaannya, dan support oleh IDE yang sering digunakan dikalangan developer seperti Android Studio dan Visual Studio Code.

2.2.5 Application Programming Interface (API)

Application Programming Interface (API) dapat digunakan untuk membuka akses data dan fungsionalitas dari sebuah aplikasi di perusahaan tertentu kepada pihak lain atau pihak internal di dalam perusahaan tersebut. Hal ini memungkinkan terhubungnya komunikasi antar produk satu dengan lainnya serta memanfaatkan data dan fungsionalitas melalui *interface* yang terdokumentasi. *Developer* biasa menggunakan *interface* untuk menghubungkan sebuah produk dengan layanan produk lain bahkan tanpa perlu mengetahui bagaimana API diimplementasikan. Penggunaan API telah menjadi hal tak terlepaskan dari pengembangan *software*, sampai banyak aplikasi web paling populer saat ini tidak akan mungkin tanpa API [19]. API secara tradisional merujuk ke *interface* yang

terhubung ke aplikasi lain. Dalam aplikasi lain tersebut mungkin telah dibuat dengan salah satu bahasa pemrograman tingkat rendah, seperti Javascript. API biasanya dibuat untuk HTTP serta mematuhi prinsip REST dan format JSON, menghasilkan *interface* yang ramah kepada *developer* dan mudah diakses serta dipahami oleh pengembangan *software* dan aplikasi yang dibuat dengan bahasa pemrograman yang ada saat ini.

2.2.6 Database

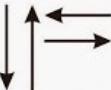
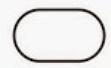
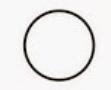
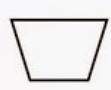
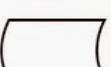
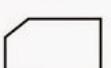
Collection data atau informasi yang terstruktur dan terorganisir, tersimpan dalam sistem komputer dikenal sebagai *database* [20]. Pengelolaan *database* dilakukan menggunakan *Database Management System* (DBMS). Data, DBMS dan aplikasi yang saling terhubung disebut sebagai sistem basis data dan familiar dengan sebutan basis data. *Database* yang paling umum beroperasi saat ini dimodelkan dengan baris dan kolom dalam serangkaian tabel digunakan untuk meningkatkan efisiensi dalam pemanfaatan kueri data. Kemudian data dengan mudah diakses untuk melakukan pengelolaan, modifikasi, dan perbaruan. Secara umum *database* tidak lepas dari penggunaan *structured query language* (SQL) untuk menulis dan mengolah data. Salah satu SQL yang sering digunakan adalah MySQL yang merupakan perangkat lunak *open source*.

2.2.7 Flowchart

Flowchart sering juga disebut dengan *process flowchart*, *process flow diagram* atau diagram alir. Flowchart merupakan gambaran terpisah dari langkah-langkah yang tersedia dalam suatu proses yang berurutan. *Flowchart* sangat umum digunakan untuk berbagai tujuan, dapat juga berupa penggambaran sebuah proses seperti proses administrasi/layanan, proses manufaktur, atau gambaran rencana proyek [21]. 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 [22], dapat dilihat simbol-simbol *flowchart* pada gambar di bawah ini.

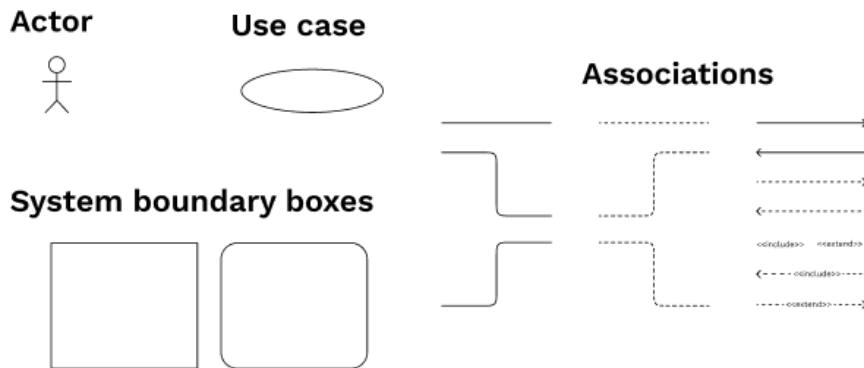
| | |
|---|---|
|  | Flow Direction symbol Yaitu simbol yang digunakan untuk menghubungkan antara simbol yang satu dengan simbol yang lain. Simbol ini disebut juga connecting line. |
|  | Terminator Symbol Yaitu simbol untuk permulaan (start) atau akhir (stop) dari suatu kegiatan |
|  | Connector Symbol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

Gambar 2.2 Simbol - simbol *flowchart*

Pada Gambar 2.2 tercantum bentuk simbol, nama, dan keterangan penggunaan simbol tersebut.

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 [23, 24]. Berdasarkan referensi [23] terdapat dua fungsi utama dari *use case* dalam sebuah sistem yang dibuat, yaitu untuk memperlihatkan urutan aktivitas proses dan untuk menggambarkan *bussines process*.



Gambar 2.3 Komponen *Use case*

Berdasarkan Gambar 2.3 ada beberapa komponen *use case* [23, 25] 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.
4. *Association* merupakan teknik yang digunakan menunjukkan interaksi antara komponen aktor dan *use case* tetentu dan digambarkan dengan garis penghubung.

2.2.9 Sequence Diagram

Sequence diagram menggambarkan hubungan dan komunikasi antar objek [26] sekaligus menunjukkan serangkaian pesan yang diperlukan oleh objek-objek yang aktif. Objek dibaca dan diurutkan dari kiri ke kanan, aktor yang menginisiasi interaksi ditaruh pada bagian paling kiri dan diikuti oleh objek setelahnya. Dimensi vertikal merepresentasikan berjalanannya waktu. Bagian paling atas menjadi titik awal dan waktu berjalan ke bawah. Garis vertikal itu disebut juga sebagai *lifeline*, pasti dimiliki setiap objek atau aktor. *Lifeline* ditampilkan dalam bentuk kotak yang juga dikenal sebagai *activation box*. ketika objek melakukan sebuah operasi, keadaan ini disebut sebagai *live activation*. Pesan antar objek digambarkan dengan sebuah panah yang diatasnya 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. Tahap pengujian atau *testing* merupakan hal wajib dalam siklus pengembangan perangkat lunak [27].

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 [28] serta sebagai pengujian perangkat lunak oleh pengguna atau klien untuk menentukan apakah perangkat lunak tersebut dapat diterima atau tidak.

2.2.12 *Skala Likert*

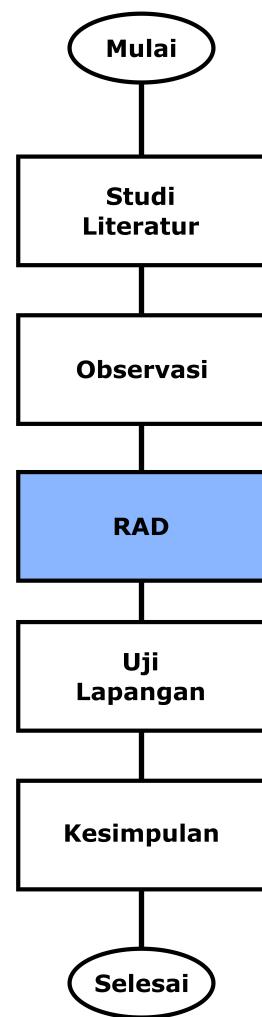
Skala *Likert* adalah skala yang digunakan untuk mengukur sikap, pendapat, dan persepsi seseorang atau sekelompok orang tentang sesuatu yang diamati. Semua pilihan jawaban diberi skor, maka responden harus menggambarkan, mendukung pernyataan (positif) atau tidak mendukung pernyataan (negatif) [29].

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 dan melakukan pengujian UAT.

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 Studio 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 Baham

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 [30] 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 [30]:

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

Keterangan :

P = Persentase

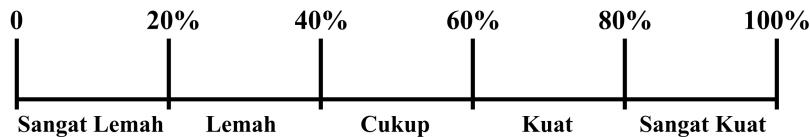
n = Jumlah responden

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



Gambar 3.2 Skala Penilaian

sudah diolah diklasifikasikan berdasarkan skala penilaian berikut [30] :

Berikut nilai kesimpulan berdasarkan referensi [30] :

Tabel 3.3 Nilai Kesimpulan

| Nilai | Pernyataan |
|------------|--------------|
| 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 pengujinya 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 Hasil Observasi

Peneliti mengamati kebutuhan sensor dan kontrol yang sering diaplikasikan pada lahan secara umum untuk kebutuhan pembuatan aplikasi KEBUNQ. Didapat data sebagai berikut.

4.1.1.1 Sensor

Beberapa sensor yang akan digunakan pada lahan pertanian yang perlu dicantumkan ke aplikasi KEBUNQ adalah sebagai berikut:

1. Sensor suhu untuk mengukur nilai suhu ruangan
2. Sensor *humidity* untuk mengukur kelembapan
3. Sensor intensitas cahaya untuk mengukur intensitas cahaya
4. Sensor suhu tanah
5. Sensor *humidity* tanah untuk mengukur kelembapan tanah
6. Sensor pH tanah untuk mengukur kandungan pH yang terdapat di tanah
7. Sensor suhu air untuk mengukur suhu pada air
8. Sensor TDS (ppm meter) untuk mengukur kadar TDS (*Total Dissolve Solid*) dalam air.
9. Sensor pH air untuk mengukur kandungan pH yang terdapat di air

4.1.1.2 Kontrol

Berikut beberapa kontrol yang biasa digunakan pada lahan pertanian yang perlu dicantumkan ke aplikasi KEBUNQ :

1. *Mist*

Digunakan untuk melakukan pengabutan, ketika menyala akan mengeluarkan kabut air. Berikut ilustrasi *mist* dapat dilihat pada Gambar 4.1.



Gambar 4.1 Ilustrasi *Mist*

2. *Sprinkler*

Digunakan untuk melakukan penyiraman, ketika menyala *sprinkler* akan menyemprotkan air. Berikut ilustrasi *sprinkler* dapat dilihat pada Gambar 4.2.



Gambar 4.2 Ilustrasi *Sprinkler*

3. *Fan*

Digunakan untuk memberikan hembusan angin, ketika menyala *fan* akan meniupkan angin. Berikut ilustrasi *fan* dapat dilihat pada Gambar 4.3.



Gambar 4.3 Ilustrasi Fan

4. *Valve*

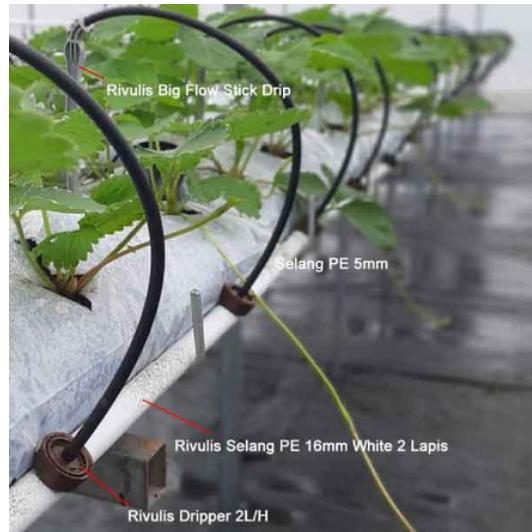
Digunakan untuk membuka alur aliran air selayaknya keran air, ketika menyala *valve* akan membuka aliran air pada selang. Berikut bentuk *valve* dapat dilihat pada Gambar 4.4.



Gambar 4.4 Bentuk Valve

5. *Drip*

Digunakan untuk melakukan pemupukan, ketika menyala *drip* akan meneteskan air yang sudah disiapkan. Berikut ilustrasi *drip* dapat dilihat pada Gambar 4.5.



Gambar 4.5 Ilustrasi Drip

6. Pompa

Digunakan untuk membantu mengalirkan air, ketika menyala pompa akan menghisap air dari sumber air yang tersedia. Berikut bentuk pompa dapat dilihat pada Gambar 4.6.



Gambar 4.6 Pompa

7. Lainnya (*Customize*)

Dibuat untuk menampung bentuk kontrol ketika terdapat kontrol selain yang tercantum pada poin 1 hingga 6.

4.1.2 Hasil Penerapan RAD

Sebelum masuk pada pembahasan per proses dalam RAD, berikut dicantumkan jadwal penggeraan sistem KEBUNQ secara menyeluruh dengan

mengimplementasikan RAD.

| Kegiatan | Minggu | | | | | | | | | | PIC |
|--|--------|---|---|---|---|---|---|---|---|----|----------------------|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | |
| Pemodelan Bisnis | | | | | | | | | | | Rizki |
| Pembuatan Dokumen SRS | | | | | | | | | | | Rizki, Ferdian |
| Pemodelan Data | | | | | | | | | | | Rizki, Ferdian, Leja |
| Persiapan Server dan penggerjaan backend | | | | | | | | | | | Leja |
| Pemodelan Proses | | | | | | | | | | | Rizki |
| Pembuatan Aplikasi | | | | | | | | | | | Rizki |
| Persiapan desain asset aplikasi | | | | | | | | | | | Rizki |
| Perancangan alat | | | | | | | | | | | Ferdian |
| Testing Aplikasi | | | | | | | | | | | Rizki, Ferdian, Leja |
| Pengujian Lapangan | | | | | | | | | | | Rizki, Ferdian, Leja |
| Penggunaan Internal | | | | | | | | | | | Ferdian |

Gambar 4.7 Jadwal Pengerjaan

Pada Gambar 4.7 dapat dilihat bahwa beberapa pengerjaan dilakukan dengan cara bersamaan dan paralel oleh anggota tim. Tedapat tiga bagian khusus yang penggerjaannya dibagi, yaitu persiapan server dan penggerjaan *backend* yang dilakukan oleh anggota tim bernama Leja, pembuatan aplikasi yang dikerjakan oleh Rizki (peneliti), dan penggerjaan perancangan alat yang dilakukan oleh anggota tim bernama Ferdian. Penggerjaan proyek ini memiliki *stakeholder* dari pihak BPP Lampung.

Kemudian berikut hasil yang didapatkan dari setiap proses dalam pemngimplementasian RAD:

4.1.2.1 Pemodelan Bisnis

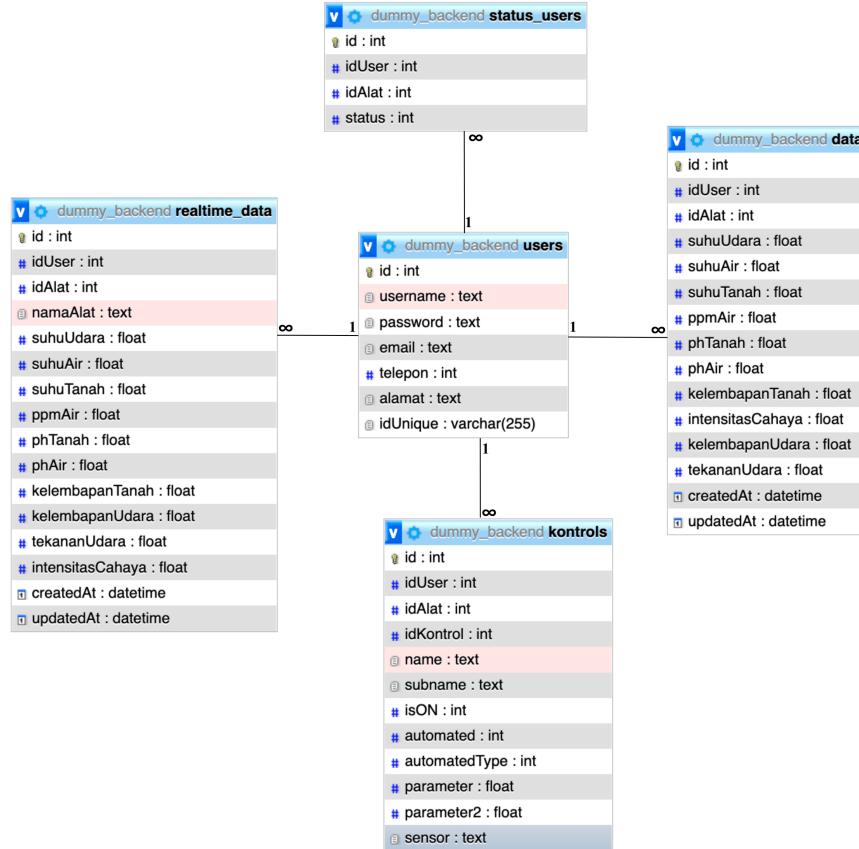
Berdasarkan hasil analisa ditentukan bahwa aplikasi KEBUNQ dirancang untuk satu jenis pengguna. Sehingga tidak terdapat level pengguna pada akses aplikasi. Berikut analisa kebutuhan pengguna

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 automatis pada suatu kontrol yang tersedia mode automatismnya

Dari proses pemodelan bisnis ini juga sekaligus dibuat SRS yang terdapat pada lampiran penelitian.

4.1.2.2 Pemodelan Data



Gambar 4.8 Skema Tabel *Database*

Setelah dilakukan tahap pemodelan bisnis, maka didapatkan skema *database* yang dirancang seperti gambar 4.8 di atas. *Database* terdiri dari lima tabel yaitu sebagai berikut:

1. Tabel data, digunakan untuk menyimpan data time series dari nilai sensor yang ada pada alat,
2. Tabel kontrols, digunakan untuk menyimpan dan mendefinisikan kontrol yang dimiliki,
3. Tabel realtime_data, digunakan untuk menyimpan data realtime sensor dan kontrol,

4. Tabel status_users, digunakan untuk menyimpan status alat apakah *online* atau *offline*,
 5. Tabel users, digunakan untuk menyimpan data user.

Dapat dilihat juga bahwa tabel user berada di tengah dan memiliki relasi *one to many* ke keempat tabel lain. Setelah skema *database* difinalisasi, maka dilakukan pembuatan *Application Programming Interface* (API) dan konfigurasi server yang dalam bagian ini penggerjaan bukan dilakukan oleh peneliti namun oleh anggota lain dalam proyek. Sehingga akhirnya didapat beberapa API (*dummy*) sebagai berikut dengan *endpoint* yang dirahasiakan demi keamanan aplikasi :

- API login

[http://47.89.212.89/dummyBackend/\(endpoint\)](http://47.89.212.89/dummyBackend/(endpoint))

Digunakan untuk melakukan *post login*.

- API *Summary Data* dari pengguna

[http://47.89.212.89/dummyBackend/data/\(endpoint\)](http://47.89.212.89/dummyBackend/data/(endpoint))

Merupakan data rangkuman dari keseluruhan data yang dimiliki seorang pengguna. Berikut contoh tampilan API *Summary Data* dari pengguna dapat dilihat pada Gambar 4.9.

Body Cookies (1) Headers (7) Test Results

Pretty Raw Preview Visualize JSON ▾

```
1 {  
2   "user": [  
3     {  
4       "email": "testing@gmail.com",  
5       "username": "testing",  
6       "password": "initesting",  
7       "telepon": 81278888,  
8       "alamat": "testing",  
9       "idUnique": "testing"  
10    }  
11  ],  
12  >   "kontrols": [...  
13  ],  
14  >   "realtimeData": [...  
15  ],  
16  >   "data": [...  
17  ]  
18 }  
19 }
```

Gambar 4.9 API Summary Pengguna

- API data *realtime*

[`http://47.89.212.89/dummyBackend/data/mobile/\(endpoint\)`](http://47.89.212.89/dummyBackend/data/mobile/(endpoint))

Merupakan API yang menyediakan bagian data *realtime* dari sebuah alat yang dimiliki pengguna. Berikut contoh tampilan API data *realtime* dapat dilihat pada Gambar 4.10.



```

1  {
2    "realtimeData": [
3      {
4        "name": "Suhu Udara",
5        "value": 26,
6        "unit": "°C",
7        "image": "/media/sensor/suhu.png"
8      },
9      {
10        "name": "Suhu Tanah",
11        "value": 31.12,
12        "unit": "°C",
13        "image": "/media/sensor/suhu_tanah.png"
14      },
15      {
16        "name": "pH Tanah",
17        "value": 32,
18        "unit": "pH",
19        "image": "/media/sensor/ph_tanah.png"
20      },
21    ],
22    "updatedAt": [
23      ...
24    ],
25    "date": "25 May 2022",
26    "time": "12:15:25"
27  }

```

Gambar 4.10 API Data *Realtime*

- API data kontrol

[http://47.89.212.89/dummyBackend/kontrol/\(endpoint\)](http://47.89.212.89/dummyBackend/kontrol/(endpoint))

Merupakan API yang berisikan data kontrol dari sebuah alat yang dimiliki pengguna. Berikut contoh tampilan API data kontrol dapat dilihat pada Gambar 4.11.



```

1  {
2      "kontrol": [
3          {
4              "isON": 0,
5              "automated": 0,
6              "parameter": 40,
7              "name": "Pompa",
8              "subname": "",
9              "sensor": "null"
10         },
11         {
12             "isON": 0,
13             "automated": 0,
14             "parameter": 50,
15             "name": "Valve",
16             "subname": "satu",
17             "sensor": "Kelembaban Tanah"
18         },
19         {
20             "isON": 0,
21             "automated": 0,
22             "parameter": 200,
23             "name": "Valve",
24             "subname": "dua",
25             "sensor": "Kelembaban Tanah"
26         }
27     ]
28 }

```

Gambar 4.11 API Data Kontrol

- API data ikon

`http://47.89.212.89/dummyBackend/(endpoint)`

(*endpoint* berada pada API data *realtime*) yang tertera pada Gambar 4.10
Merupakan data yang menyimpan lokasi *asset* ikon sensor.

- API status alat

`http://47.89.212.89/dummyBackend/status/(endpoint)`

Merupakan API yang berisikan data yang menunjukkan status keadaan alat.
Berikut contoh tampilan API status alat dapat dilihat pada Gambar 4.12.

```

1  {
2   "id": 2,
3   "idUser": 1,
4   "idAlat": 2,
5   "status": 1
6 }

```

Gambar 4.12 API Status Alat

- API update kontrol

[http://47.89.212.89/dummyBackend/kontrol/\(endpoint\)](http://47.89.212.89/dummyBackend/kontrol/(endpoint))

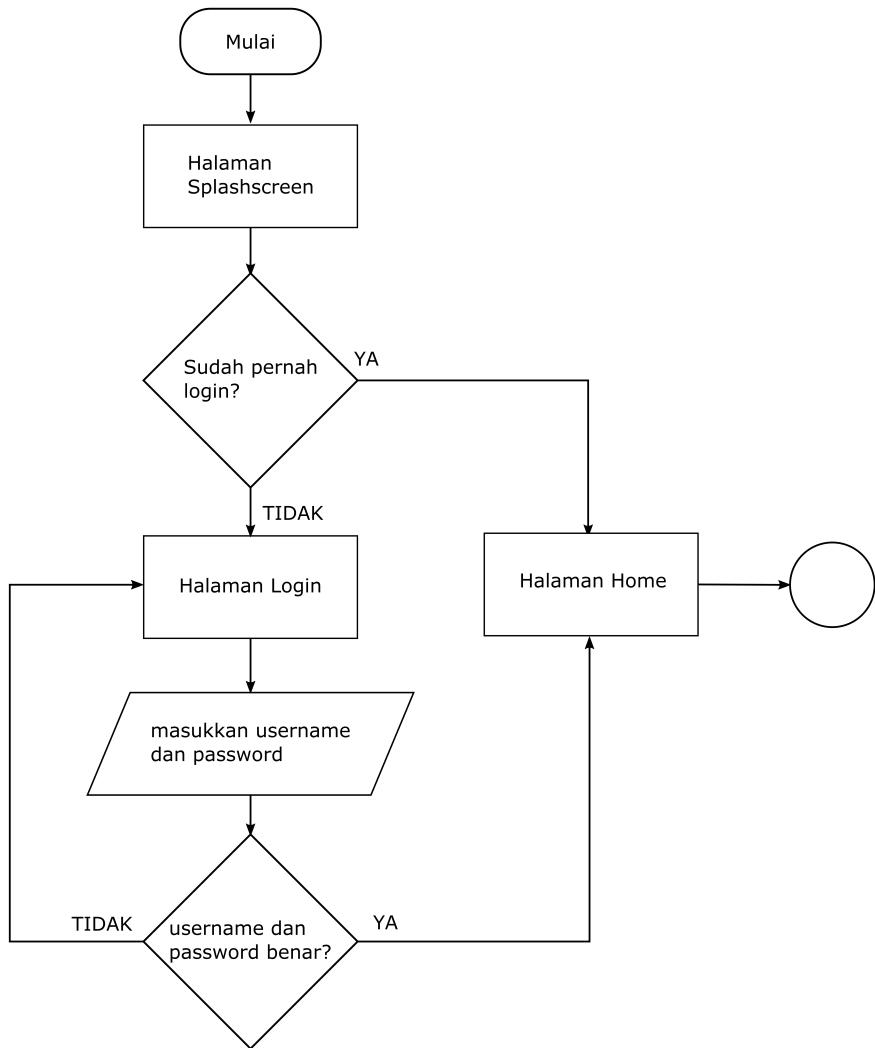
Merupakan API untuk melakukan *post* data update dari masukan pengguna.

API ini digunakan oleh peneliti dalam tahap pembuatan aplikasi *mobile* KEBUNQ sebagai penghubung antara aplikasi dengan *database server*.

4.1.2.3 Pemodelan Proses

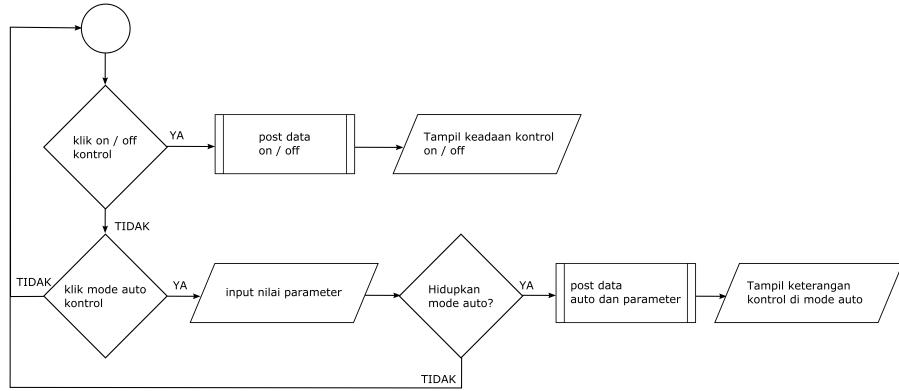
Berikut beberapa pemodelan proses aplikasi mobile KEBUNQ BPP Lampung yang dirancang setelah menganalisa tahap pemodelan bisnis dan pemodelan data.

- a. *Flowchart*

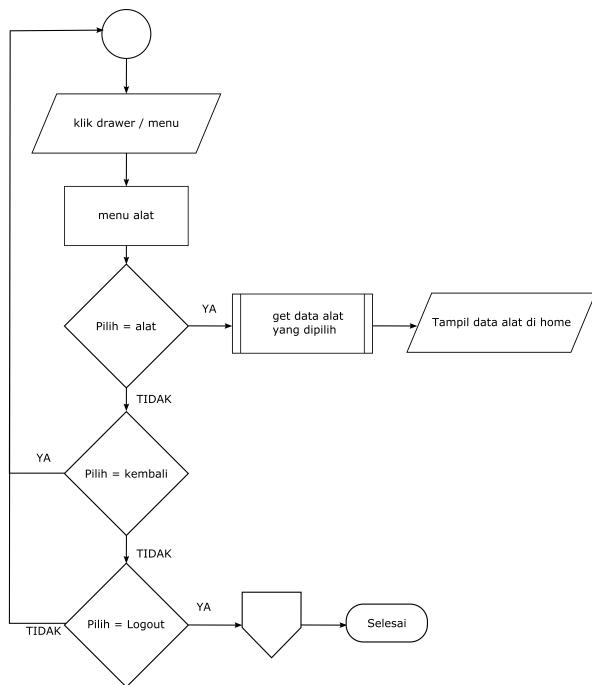


Gambar 4.13 flowchart bagian 1 (mulai)

Pada Gambar 4.13 menunjukkan proses sejak dimulai hingga masuk ke halaman *home*. Ketika dibuka, aplikasi akan menampilkan halaman *splashscreen* kemudian melakukan pengecekan apakah pengguna sudah pernah melakukan login dengan benar atau belum, jika sudah pernah dan belum melakukan logout maka akan langsung masuk ke halaman *home* dan jika belum, aplikasi akan menampilkan halaman *login*. Pada halaman *login* pengguna diharuskan memasukkan *username* dan *password* dengan benar untuk bisa masuk ke halaman *home*.

**Gambar 4.14** flowchart bagian 2 (kontrol)

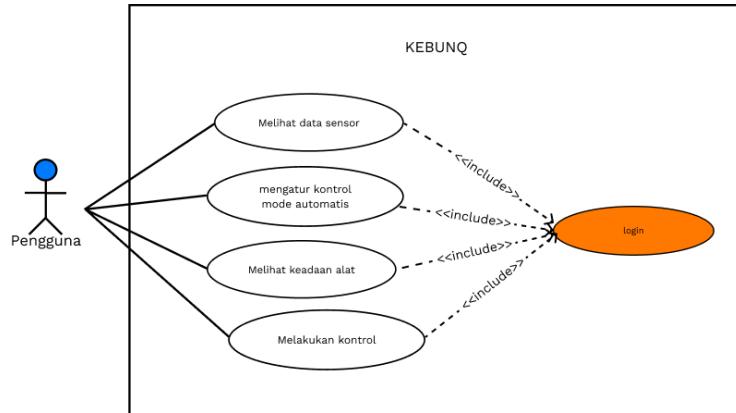
Pada Gambar 4.14 menunjukkan proses pengguna melakukan kontrol. Ketika pengguna memilih fitur kontrol langsung maka aplikasi akan mengirimkan data perintah *on* atau *off* ke server sehingga keadaan data menjadi sesuai dengan data aktual sebagaimana yang tampil pada aplikasi. Jika tidak melakukan kontrol langsung, maka pengguna bisa melakukan perintah kontrol dengan mode otomatis., dalam proses tersebut pengguna diminta memberikan masukan berupa nilai parameter dan melakukan konfirmasi. Setelah itu data akan dikirim ke *database* dan aplikasi akan menampilkan status kontrol tersebut berada pada mode otomatis.

**Gambar 4.15** flowchart bagian 3 (selesai)

Pada Gambar 4.15 menunjukkan proses yang terjadi ketika pengguna

membuka menu yang terdapat beberapa opsi pilihan, yaitu (1) memilih alat yang akan ditampilkan datanya, (2) memilih untuk kembali ke halaman *home*, dan (3) pengguna dapat melakukan *logout*.

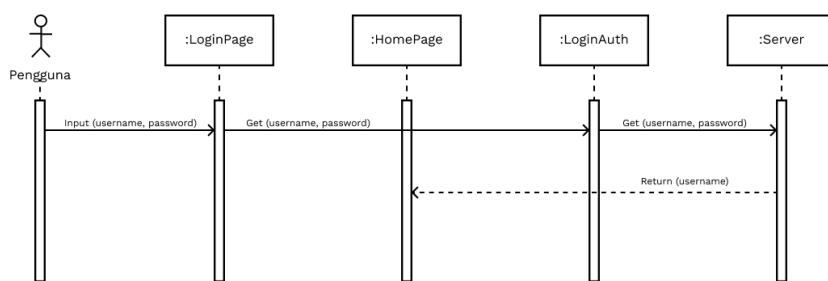
b. Use Case



Gambar 4.16 Use Case Diagram

Pada Gambar 4.16 menampilkan gambaran apa saja yang bisa dilakukan pengguna dalam aplikasi KEBUNQ, yaitu pengguna dapat melihat data sensor, melakukan kontrol secara langsung atau mode otomatis, dan melihat keadaan alat apakah *offline / online*. Namun hal tersebut dapat dilakukan pengguna dengan syarat harus melakukan login terlebih dahulu.

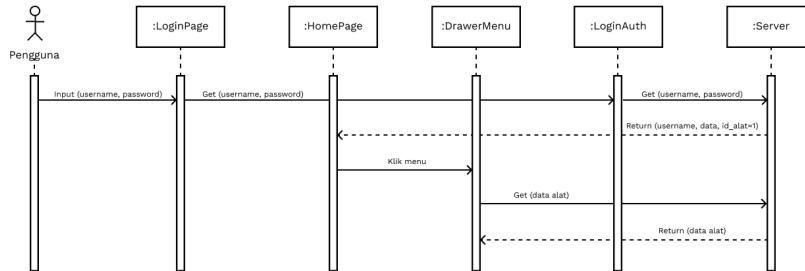
c. Sequence Diagram



Gambar 4.17 Sequence Diagram Login

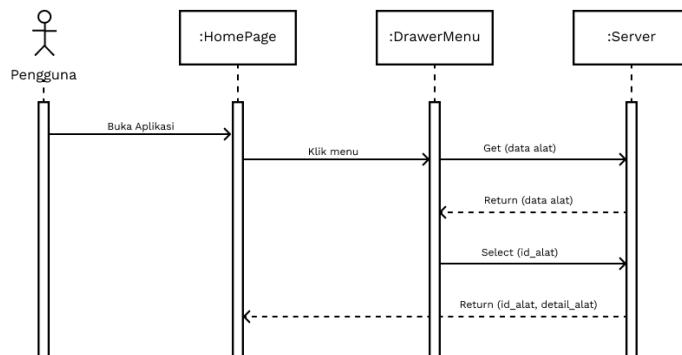
Pada Gambar 4.17 menunjukkan gambaran proses ketika pengguna melakukan *login*. Dimulai dengan pengguna memberikan *input username* dan *password* pada halaman *login*. Kemudian aplikasi melakukan *authentication* atau proses pengecekan data apakah sesuai dengan yang tersedia pada server. Ketika data tersebut sesuai, maka aplikasi akan

melakukan pengambilan data dan pada halaman *home* akan menampilkan *username* pengguna.



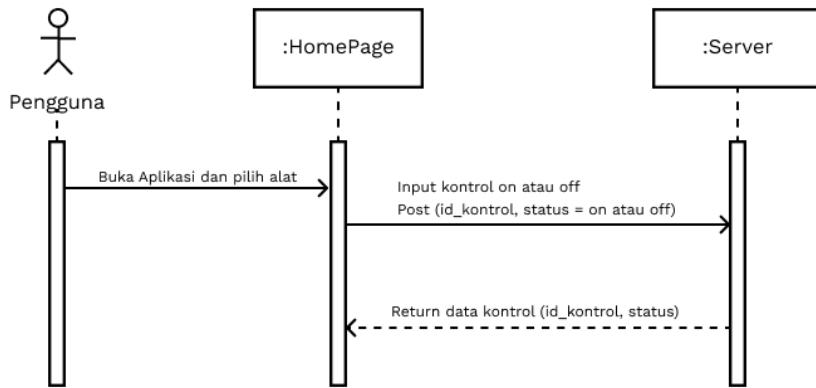
Gambar 4.18 Sequence Diagram Melihat Menu / Alat

Pada Gambar 4.18 menunjukkan gambaran proses ketika pengguna melihat menu. Setelah melewati proses *login*, pengguna memilih menu pada halaman *home* kemudian aplikasi akan melakukan pengambilan data ke server dan menampilkan hasilnya pada menu.



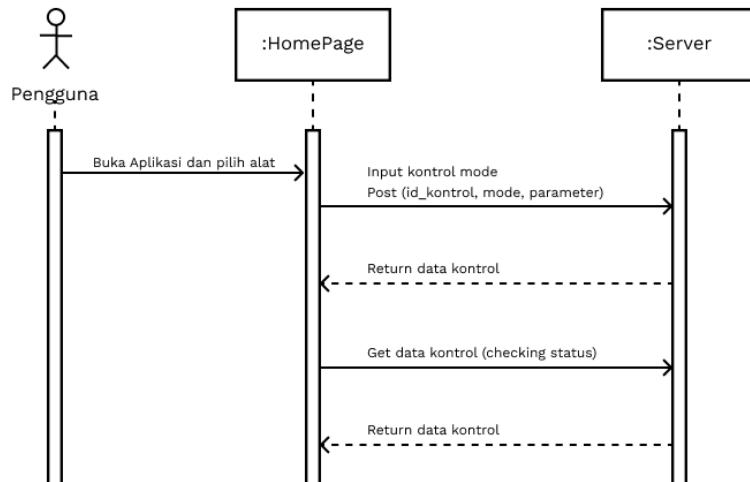
Gambar 4.19 Sequence Diagram Melihat Detail Alat

Pada Gambar 4.19 menunjukkan gambaran proses ketika pengguna melihat data yang ada pada suatu alat. Dimulai dengan pengguna membuka aplikasi dan setelah berada pada halaman *home* pengguna memilih menu sehingga tampil bagian *drawer/menu* alat yang ada, setelah itu pengguna dapat memilih alat yang berada pada menu tersebut dan aplikasi akan melakukan pengiriman data *id* yang dipilih ke server, kemudian aplikasi akan menampilkan data yang ada dari alat tersebut pada halaman *home*.



Gambar 4.20 Sequence Diagram Melakukan Kontrol

Pada Gambar 4.20 menunjukkan gambaran proses ketika pengguna melakukan kontrol. Setelah berada pada halaman *home* dan menampilkan bagian alat yang terpilih, pengguna dapat menginputkan kontrol *on/off* kemudian aplikasi akan menampilkan data sesuai dengan yang tersedia pada server.



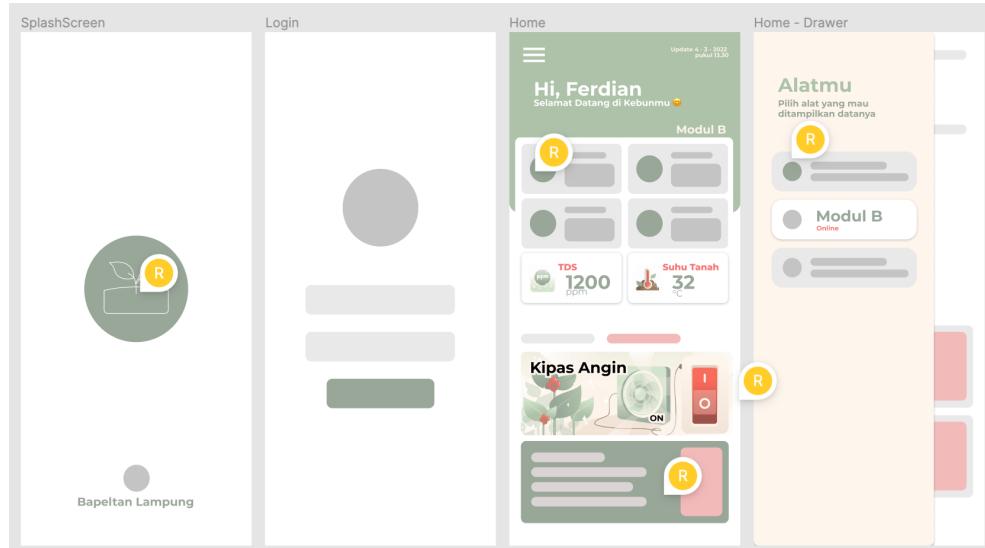
Gambar 4.21 Sequence Diagram Melakukan Kontrol Automatis

Pada Gambar 4.21 menunjukkan gambaran proses ketika pengguna melakukan kontrol otomatis. Dimulai dengan keadaan pengguna yang sudah berada pada tampilan alat yang akan dikontrol memilih kontrol mode otomatis dan memasukkan nilai parameter kontrol tersebut, kemudian aplikasi akan menampilkan data aktual sebagaimana data yang berada pada server.

4.1.2.4 Pembuatan Aplikasi

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

1. Perancangan Desain Layout *User Interface*



Gambar 4.22 Rancangan layout UI

Pada Gambar 4.22 terdapat beberapa halaman yang dibuat berupa *layout* untuk memudahkan pembuatan aplikasi. Halaman tersebut yaitu halaman *splashscreen*, halaman *login*, halaman *home* dan tampilan menu *drawer*.

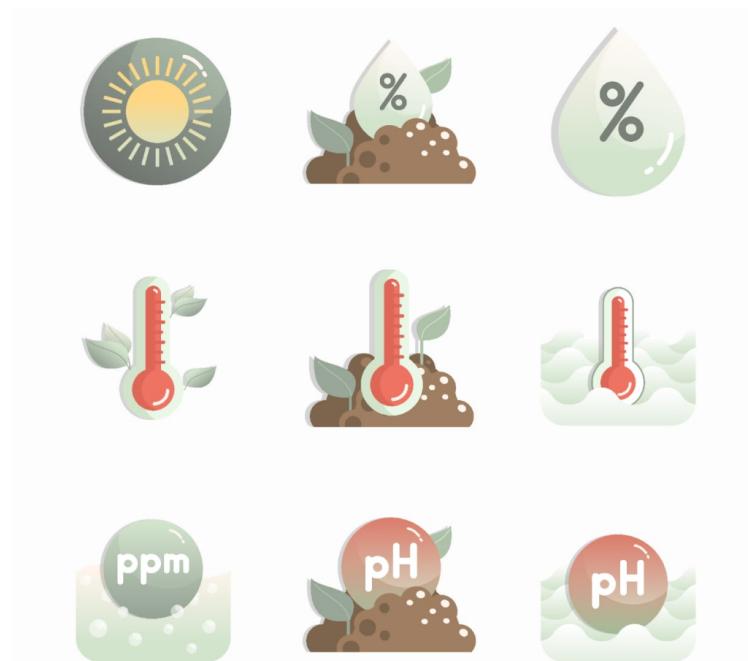
2. Pembuatan *assets*



Gambar 4.23 assets Logo aplikasi

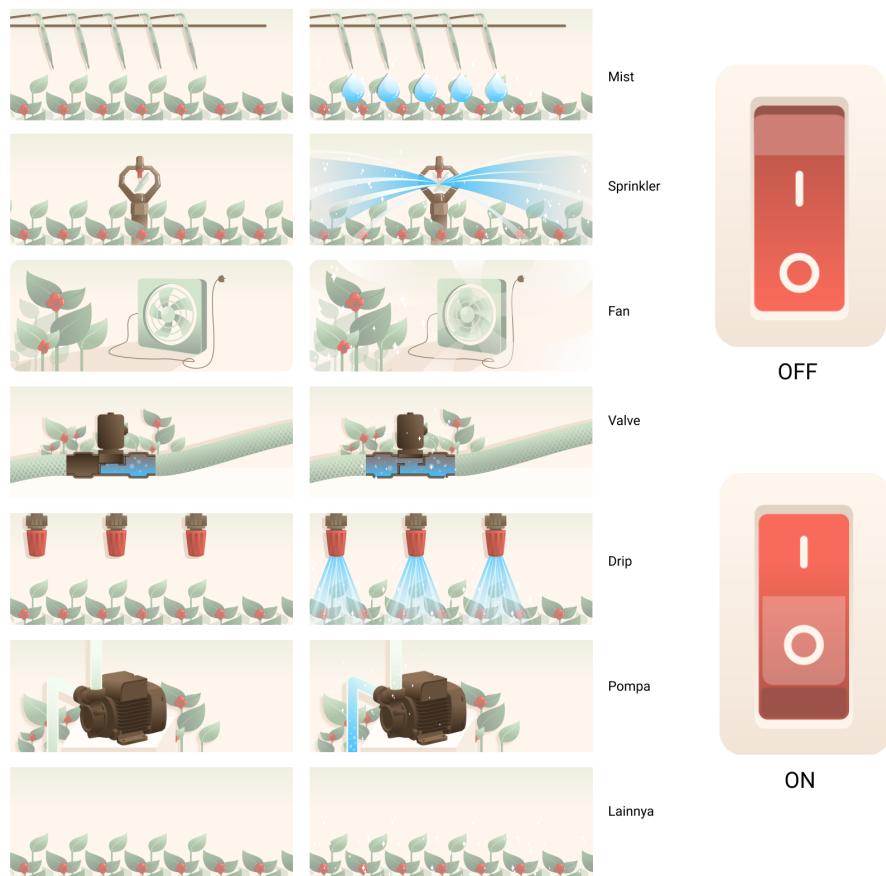
Logo aplikasi KEBUNQ pada Gambar 4.23 dirancang memiliki arti sebagai berikut :

- Bentuk koin : menunjukkan bahwa aplikasi KEBUNQ dirancang untuk mendukung program *low cost smart farming*
- Tanda sinyal : menunjukkan penggunaan aplikasi memerlukan koneksi jaringan internet untuk melakukan pemantauan dan kontrol dari jarak jauh (*online apps*)
- Tanaman : menunjukkan bahwa aplikasi KEBUNQ berhubungan erat dan tidak lepas dengan tanaman dalam penggunaannya
- pijakan tanaman dan bulatan-bulatan kecil : mewakilkan banyaknya unsur yang dapat kita pantau melalui aplikasi KEBUNQ.



Gambar 4.24 assets Ikon Sensor

Pada Gambar 4.24 tercantum *asset* ikon sensor yang akan digunakan untuk ditampilkan pada aplikasi. Terdapat 9 buah ikon sebagaimana yang tercantum dalam hasil observasi.



Gambar 4.25 *assets* Kontrol

Pada Gambar 4.25 mencantumkan *assets* yang akan digunakan pada tampilan kontrol alat, desain yang dibuat tersebut berdasarkan data hasil observasi. Pembuatan *assets* logo, ikon dan ilustrasi bertujuan untuk meningkatkan kemudahan pengguna dalam menggunakan aplikasi KEBUNQ dan menjadikan KEBUNQ sebagai aplikasi yang unik dengan menggunakan *assets* yang orisinal khusus didesain dan digunakan pada aplikasi KEBUNQ.

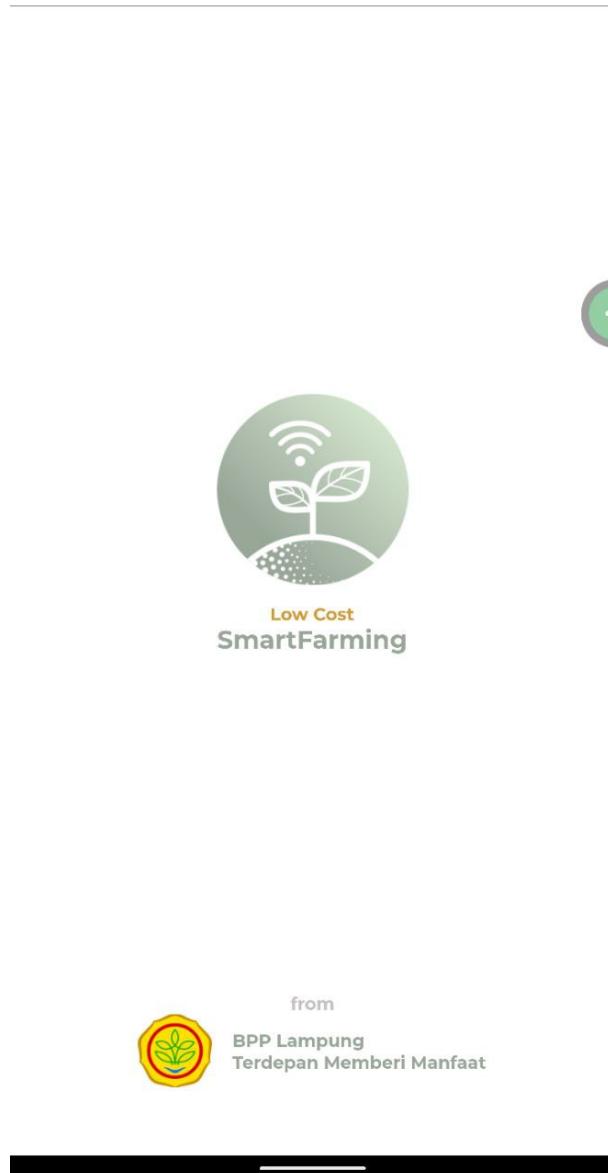
3. Pembuatan Aplikasi

Implementasi dari perancangan aplikasi KEBUNQ

- Halaman *Splash Screen*

Merupakan halaman pertama yang dilihat dengan hitungan waktu beberapa detik. Pada aplikasi KEBUNQ, *splash screen* dibuat dengan sedikit implementasi gerak animasi dan mencantumkan logo KEBUNQ dan BPP Lampung. Halaman *Splash Screen* dapat dilihat pada Gambar

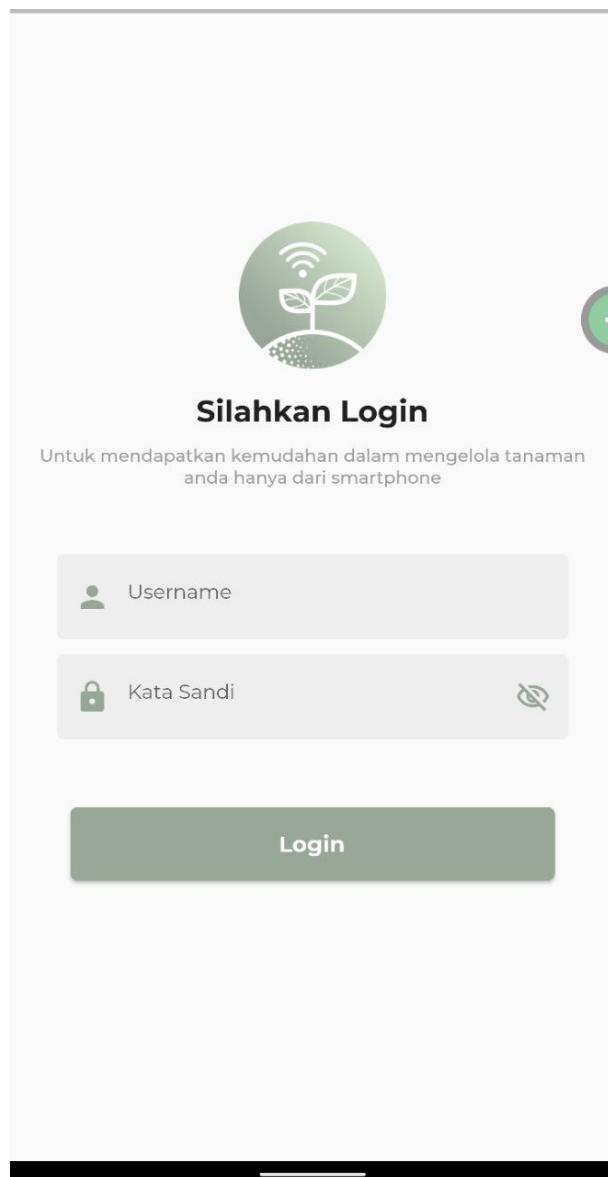
4.26.



Gambar 4.26 Tampilan *Splash Screen*

- Halaman *Login*

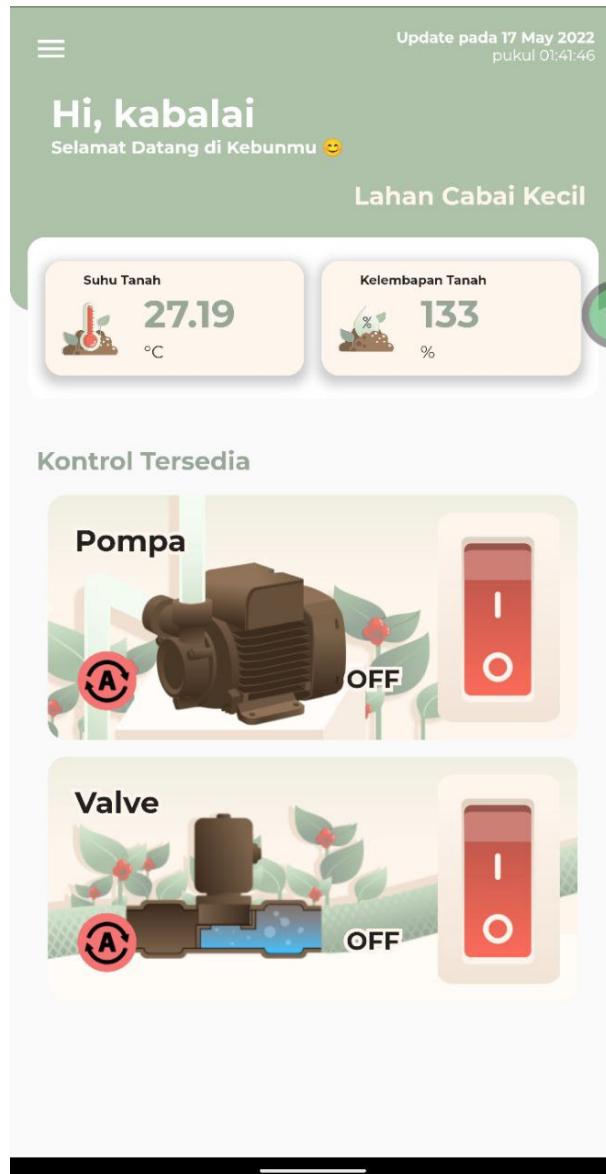
Halaman *login* berupa formulir *login* yang diharuskan untuk diisi dengan benar supaya kemudian pengguna dapat menggunakan fitur utama aplikasi KEBUNQ. Halaman login menampilkan *field username* dan *password*. Dalam proses *login* dilakukan pemeriksaan data apakah data yang dimasukkan sesuai dengan *database* atau tidak. Jika ada maka aplikasi KEBUNQ akan menampilkan halaman *Home* dan menampilkan data yang dimiliki oleh pengguna. Halaman *login* dapat dilihat pada Gambar 4.27.



Gambar 4.27 Tampilan Halaman *Login*

- Halaman *Home*

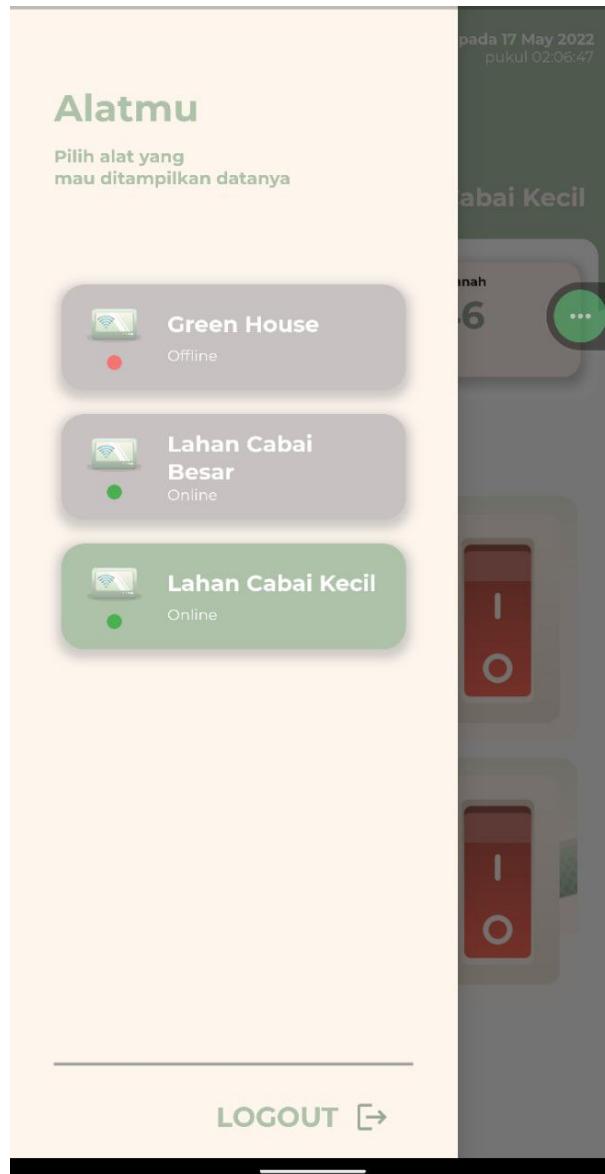
Halaman *Home* merupakan halaman utama yang digunakan pengguna untuk melihat data sensor dan kontrol yang ada. Pengguna dapat melakukan kontrol atau kendali dengan cara menyalakan atau mematikan kontrol maupun memberikan *setting* automatis pada kontrol yang dipilih. Halaman *Home* dapat dilihat pada Gambar 4.28.



Gambar 4.28 Tampilan Halaman Home

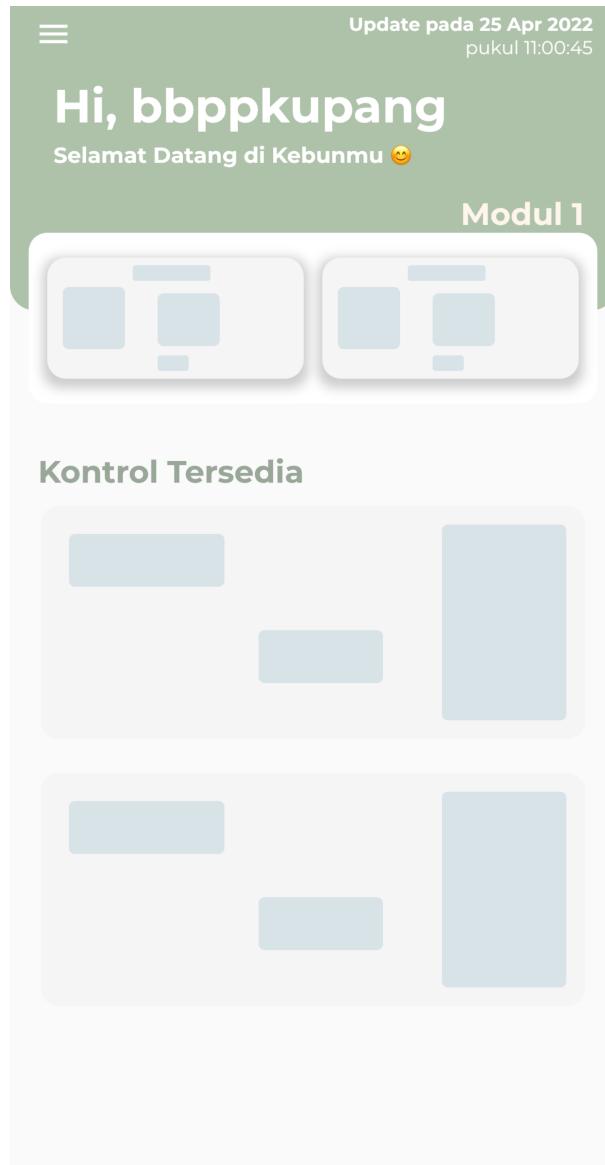
- *Drawer*

Drawer merupakan bagian tampilan Menu yang terdapat pada halaman *Home* ketika diklik/disentuh. Menu *drawer* berisikan *list* alat yang tersedia pada akun pengguna. Pada *list* alat tersebut terdapat nama alat dan status keadaan alat sekarang apakah *online* (keadaan nyala) atau *offline* (keadaan mati). Tampilan menu *drawer* dapat dilihat pada Gambar 4.29.



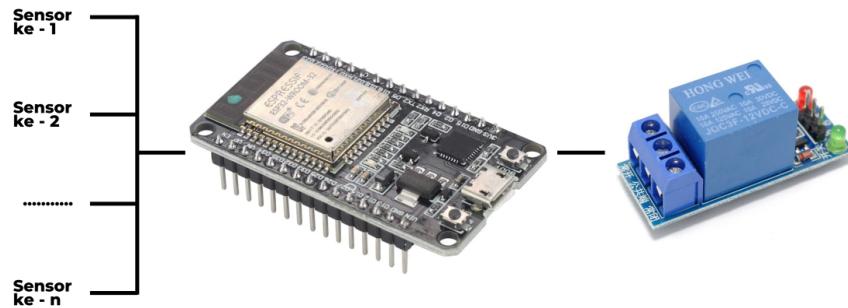
Gambar 4.29 Tampilan Menu Drawer

- Tampilan *Loading* Tampilan *loading* merupakan tampilan yang muncul membentuk *shimmer* pada layout yang dibuat ketika aplikasi KEBUNQ menunggu *loading request* data dari API. Tampilan *loading* dapat dilihat pada Gambar 4.30.



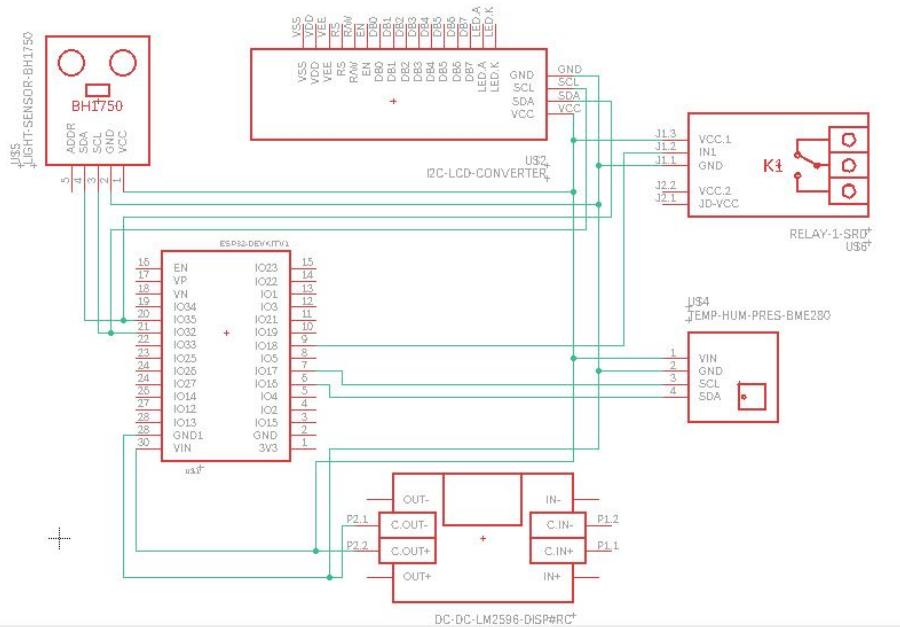
Gambar 4.30 Tampilan *Loading*

Bersamaan dengan proses pembuatan aplikasi, sebagaimana tertera pada Gambar 4.7 Halaman 20 dilakukan juga penggerjaan perancangan alat yang dikerjakan oleh anggota tim. Alat yang dibuat tersebut merupakan alat yang akan dipasang pada lahan cabai besar, lahan cabai kecil, dan *screen hidroponik*. Pada lahan cabai besar alatnya menggunakan Esp32, *soil capasitive sensor*, *one wire temperature sensor*, dan modul relay 4 *channel*. Pada lahan cabai kecil alatnya menggunakan Esp32, *soil capasitive sensor*, *one wire temperature sensor*, dan modul relay 2 *channel*. Pada lahan *screen hidroponik* alatnya menggunakan Esp32, DHT21, lux sensor BH1750, dan modul relay 2 *channel*. Berikut hubungan rancangan alat secara umum dapat dilihat pada Gambar 4.31.



Gambar 4.31 Rancangan Alat Secara Umum

Pada Gambar 4.31 dapat kita lihat ada 3 komponen utama dari alat yang dirancang yaitu *microcontroller* Esp32, sensor dan relay yang jumlahnya menyesuaikan dengan kebutuhan. Aplikasi sudah dirancang akan otomatis mendukung jumlah sensor dan kontrol yang dinamis sesuai dengan *database*. Untuk detail rancangan alat, berikut salah satu rancangan skematik dari alat yang dibuat untuk penggunaan pada lahan cabai dapat dilihat pada Gambar 4.32.



Gambar 4.32 Rancangan Skematik Alat Lahan Cabai

Pada Gambar 4.32 menunjukkan skematik dari alat yang dibuat untuk lahan cabai yang menggambarkan hubungan dan alur antar pin dari komponen yang ada.



Gambar 4.33 Alat Terpasang di Lahan

Pada Gambar 4.33 menunjukkan alat yang terpasang dan digunakan pada lahan cabai di BPP Lampung.

4.1.2.5 Pengujian dan Pergantian

Pada *testing* aplikasinya menggunakan *black box testing* untuk mencari tahu apakah aplikasi sudah seperti yang dirapkan atau belum. Berikut beberapa tes yang dilakukan dapat dilihat pada Tabel 4.1 dan Tabel 4.2

- Pengujian Halaman *Login*

Tabel 4.1 Pengujian Halaman *Login*

| No. | Skenario Pengujian | Test Case | Hasil yang Diharapkan | Hasil Pengujian | Keterangan |
|-----|---|--|-----------------------------|-----------------------------|------------|
| 1 | Mengisi <i>Username</i> dan <i>password</i> yang benar sudah terdaftar di <i>database</i> | <i>Login</i> dengan akun terdaftar yang diberikan | Berhasil <i>Login</i> | Berhasil <i>Login</i> | Sesuai |
| 2 | Mengisi <i>Username</i> yang benar dan <i>password</i> yang salah | <i>Login</i> dengan <i>password</i> yang salah | Tidak Berhasil <i>Login</i> | Tidak Berhasil <i>Login</i> | Sesuai |
| 3 | Mengisi <i>Username</i> yang salah dan <i>password</i> yang benar | <i>Login</i> dengan <i>password</i> yang salah | Tidak Berhasil <i>Login</i> | Tidak Berhasil <i>Login</i> | Sesuai |
| 4 | Mengisi <i>Username</i> dan <i>password</i> yang salah | <i>Login</i> dengan <i>username</i> dan <i>password</i> yang salah | Tidak Berhasil <i>Login</i> | Tidak Berhasil <i>Login</i> | Sesuai |

Pada Tabel 4.1 dapat dilihat dari *test case* yang dilakukan untuk melihat apakah fungsionalitas pada halaman *login* sudah sesuai atau belum dengan hasil yang diharapkan. Hasilnya adalah semua *test case* sesuai dengan yang diharapkan setelah beberapa kali perbaikan ketika ditemukan bagian yang tidak sesuai.

- Pengujian Halaman *Home*

Tabel 4.2 Pengujian Halaman *Home*

| No. | Skenario Pengujian | Test Case | Hasil yang Diharapkan | Hasil Pengujian | Keterangan |
|-----|--|--------------------------------|--|--|------------|
| 1 | Menekan <i>on</i> sebuah kontrol | klik tombol <i>on</i> | kondisi kontrol berubah menjadi <i>on</i> | kondisi kontrol berubah menjadi <i>on</i> | Sesuai |
| 2 | Menekan <i>off</i> sebuah kontrol | klik tombol <i>off</i> | kondisi kontrol berubah menjadi <i>off</i> | kondisi kontrol berubah menjadi <i>off</i> | Sesuai |
| 3 | Melakukan setting auto pada kontrol yang tersedia mode auto dan menginputkan nilai parameter | klik tombol auto | Mode auto menyala | Mode auto menyala | Sesuai |
| 4 | Melakukan setting auto pada kontrol yang tidak tersedia mode auto | klik tombol auto | Muncul keterangan mode auto tidak tersedia | Muncul keterangan mode auto tidak tersedia | Sesuai |
| 5 | Mematikan mode auto pada sebuah kontrol | klik tombol auto, klik matikan | Mode auto mati | Mode auto mati | Sesuai |
| 6 | Membuka menu drawer dan memilih alat lain | klik tombol <i>drawer</i> menu | berhasil menampilkan data alat terpilih | berhasil menampilkan data alat terpilih | Sesuai |

| No. | Skenario Pengujian | Test Case | Hasil yang Diharapkan | Hasil Pengujian | Keterangan |
|-----|--|--|--|--|------------|
| 7 | Membuka menu drawer dan pengcekan keadaan alat (<i>online / offline</i>) | menyediakan alat yang berada dalam kondisi menyala | benar menampilkan keadaan aktual | benar menampilkan keadaan aktual | Sesuai |
| 8 | Melakukan perubahan <i>database</i> sensor | mengubah value atribut sensor di <i>database</i> | Tampilan <i>home</i> dinamis mengikuti perubahan | Tampilan <i>home</i> dinamis mengikuti perubahan | Sesuai |
| 9 | Melakukan perubahan <i>database</i> kontrol | mengubah value atribut kontrol di <i>database</i> | Tampilan <i>home</i> dinamis mengikuti perubahan | Tampilan <i>home</i> dinamis mengikuti perubahan | Sesuai |
| 10 | Membuka menu <i>drawer</i> dan melakukan <i>Logout</i> | pilih ya saat konfirmasi <i>logout</i> | Aplikasi akan berhasil <i>logout</i> | Aplikasi akan berhasil <i>logout</i> | Sesuai |
| 11 | Membuka menu <i>drawer</i> dan melakukan <i>Logout</i> | pilih tidak saat konfirmasi <i>logout</i> | Aplikasi tidak berhasil <i>logout</i> | Aplikasi tidak berhasil <i>logout</i> | Sesuai |

Pada Tabel 4.2 tertera fungsionalitas pada halaman *home* dari 11 *test case* yang disediakan, hasil akhirnya menunjukkan semua sesuai dengan yang diharapkan.

4.1.3 Uji Lapangan

Pada uji lapangan selain mengamati jalannya aplikasi sekaligus dilakukan pengujian UAT untuk mengetahui tingkat penerimaan pengguna dalam menggunakan aplikasi KEBUNQ. Setelah data kuesioner dikumpulkan kemudian dilakukan perhitungan persentase dengan cara mengalikan jumlah setiap pilihan jawaban dengan 100 kemudian dibagi dengan jumlah responden. Persentase dapat dihitung dengan menggunakan rumus:

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

Keterangan :

P = Persentase

n = Jumlah responden

f = Frekuensi jawaban

Berikut kuesioner yang dilakukan, pertanyaannya dapat dilihat pada Tabel 4.3.

Tabel 4.3 Kuesioner

| No | Variabel | Kode | Pertanyaan |
|----|-----------|------|--|
| 1 | Desain | P1 | Apakah tampilan aplikasi KEBUNQ ini menarik? |
| | | P2 | Menu atau fitur aplikasi KEBUNQ ini mudah dipahami? |
| | | P3 | Apakah penggunaan warna sudah cocok dan sesuai? |
| 2 | Kemudahan | P4 | Apakah penyampaian informasi pada data alat sudah cukup lengkap |
| | | P5 | Apakah aplikasi KEBUNQ ini dapat dijadikan alat bantu kelola lahan? |
| | | P6 | Apakah penggunaan aplikasi KEBUNQ ini dapat mengurangi beban pengguna dalam mengelola lahan? |
| | | P7 | Apakah dengan adanya aplikasi KEBUNQ ini membantu pengguna dalam mengakses keadaan lahan dan mengelolanya? |
| 3 | efisiensi | P8 | Apakah pemantauan lahan lebih cepat dengan menggunakan aplikasi? |
| | | P9 | Apakah pengolahan kontrol lahan lebih cepat dengan menggunakan aplikasi? |
| | | P10 | Apakah pengolahan lahan menggunakan aplikasi KEBUNQ mempersingkat waktu dalam pengolahan lahan? |

Hasil kuesioner yang didapat kemudian dihitung jumlahnya berdasarkan setiap jawaban, perhitungan dapat dilihat pada tabel 4.4, berikut:

Tabel 4.4 Jumlah Hasil Kuesioner

| No | Variabel | Kode | Nilai | | | | |
|----|----------|------|-------|----|---|----|-----|
| | | | SS | S | N | TS | STS |
| 1 | Desain | P1 | 16 | 14 | 5 | 0 | 0 |
| | | P2 | 16 | 16 | 3 | 0 | 0 |
| | | P3 | 11 | 14 | 8 | 2 | 0 |

| No | Variabel | Kode | Nilai | | | | |
|-------|-----------|------|-------|-----|----|----|-----|
| | | | SS | S | N | TS | STS |
| 2 | Kemudahan | P4 | 9 | 21 | 4 | 1 | 0 |
| | | P5 | 19 | 15 | 1 | 0 | 0 |
| | | P6 | 14 | 12 | 9 | 0 | 0 |
| | | P7 | 20 | 14 | 1 | 0 | 0 |
| 3 | efisiensi | P8 | 18 | 11 | 6 | 0 | 0 |
| | | P9 | 17 | 13 | 5 | 0 | 0 |
| | | P10 | 16 | 12 | 7 | 0 | 0 |
| Total | | | 156 | 142 | 49 | 3 | 0 |

Dari data tertera pada Tabel 4.4, dilakukan perhitungan dengan pemberian bobot pada total setiap jawaban. Perhitungan pembobotan dapat dilihat sebagai berikut:

- Jumlah skor responden Sangat Setuju (SS) = $156 \times 5 = 780$
- Jumlah skor responden Setuju (S) = $142 \times 4 = 568$
- Jumlah skor responden Netral (N) = $49 \times 3 = 147$
- Jumlah skor responden Tidak Setuju (TS) = $3 \times 2 = 6$
- Jumlah skor responden Sangat Tidak Setuju (STS) = $0 \times 1 = 0$

Maka didapat jumlah total dari pemberian bobot adalah **1.501**

Hasil jawaban dari 35 responden tersebut kemudian dilakukan perhitungan nilai terendah dan tertinggi seperti berikut:

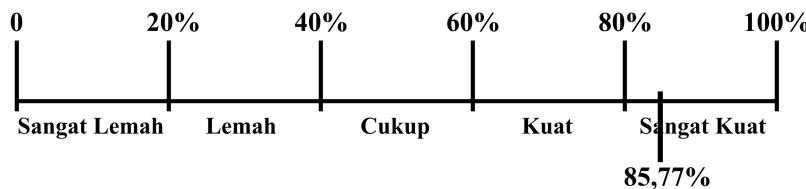
- Nilai terendah = $35 \times 10 \times 1 = 350$
- Nilai tertinggi = $35 \times 10 \times 5 = 1.750$

Hasil perhitungan nilai tertinggi yang didapat adalah 1.750. Kemudian dilakukan perhitungan persentase menggunakan persamaan sebelumnya, sebagai berikut:

$$P = \frac{1501}{1750} \times 100\% = 85,77\% \quad (\text{IV.2})$$

Berdasarkan perhitungan di atas, didapatkan bahwa tingkat penerimaan terhadap aplikasi KEBUNQ terletak pada rentang 81% - 100%, masuk dalam kategori nilai kesimpulan sangat kuat sesuai dengan yang dikemukakan oleh Riduwan dalam

referensi [30], dengan nilai perhitungan persentase 85,77%, yang berarti aplikasi KEBUNQ ini dapat diterima oleh pengguna. Hasil persentase tersebut dapat dilihat pada skala penilaian pada Gambar 4.20 berikut ini:



Gambar 4.34 Skala penilaian

4.2 Pembahasan

Pada tahap observasi didapatkan jenis-jenis sensor dan kontrol yang akan dimasukkan pada penentuan bagaimana aplikasi dirancang dan dibuat menjadi sedinamis mungkin sehingga aplikasi dapat dengan mudah menampilkan sensor dan kontrol dari alat-alat yang debuat berbeda antara alat satu dengan lainnya. Pada tahap pembuatan aplikasi KEBUNQ, peneliti menggunakan *framework Flutter* dengan bahasa pemrograman *Dart*. Dalam penelitian ini dilakukan dua metode pengujian yaitu *black box testing* dan UAT. Pada pengujian *black box* semua hasil menunjukkan kesesuaian dengan apa yang diharapkan, sedangkan pada pengujian UAT yang melibatkan 35 responden, didapatkan hasil persentase sebesar 85,77% yang menunjukkan penerimaan pengguna yang sangat kuat.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian yang dilakukan, maka dapat disimpulkan sebagai berikut:

1. Aplikasi KEBUNQ berhasil dibuat dan berjalan di android.
2. Aplikasi KEBUNQ yang dibuat dapat melakukan *monitoring* dan kontrol pada alat yang dipasang.
3. Pengimplementasian RAD mempermudah peneliti dalam melakukan manajemen proyek.
4. Hasil pengujian UAT aplikasi KEBUNQ memiliki nilai persentase 85,77% dengan keterangan penerimaan sangat kuat, hal ini merepresentasikan bahwa aplikasi dapat diterima oleh responden atau pengguna.

5.2 Saran

Berdasarkan aplikasi KEBUNQ yang dirancang dan dibuat masih sederhana perlu adanya pengembangan yang dilakukan untuk penelitian selanjutnya, sehingga peneliti memberikan saran sebagai berikut:

1. Perlu adanya peningkatan pengamanan pada aplikasi.
2. Perlu adanya pengembangan *User Interface* (UI) yang lebih baik dengan kaidah dan aturan yang ada.
3. Perlu adanya pengembangan fitur dengan mempertimbangkan aspek *User Experience* (UX).
4. Dalam penerapan RAD, benar-benar dibutuhkan tim yang cepat dan solid, sehingga ketika mengalami masalah dapat diperbaiki dengan cepat.

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] Admin dispmd, "Pengertian Teknologi Tepat Guna," *Dinas Pemberdayaan Masyarakat dan Desa*, 16-May-2018. [Online]. Available: https://dispmd.bulelengkab.go.id/informasi/detail/bank_data/pengertian-teknologi-tepat-guna-13. [Accessed: 02-Feb-2022].
- [3] 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].
- [4] A. ID, "Jumlah Petani di indonesia - grafik alinea ID," <https://data.alinea.id/>, 11-Oct-2021. [Online]. Available: <https://data.alinea.id/jumlah-petani-di-indonesia-b2cCd9Bp9c>. [Accessed: 08-Feb-2022].
- [5] Direktorat Pangan dan Pertanian, "Studi Pendahuluan Rencana Pembangunan Jangka Menengah Nasional (RPJMN) Bidang Pangan dan Pertanian 2015 – 2016", Direktorat Pangan dan Pertanian Kementerian Perancanaan Pembangunan Nasional/Badan Perencanaan Pembangunan Nasional, 2013.
- [6] C. Jessica, "Software development life cycle (SDLC): Arti, Cara Kerja, Penerapan, Dan Manfaatnya," *Glints Blog*, 17-Dec-2021. [Online]. Available: <https://glints.com/id/lowongan/sdlc-software-development-life-cycle/>. [Accessed: 27-May-2022].
- [7] F. NKD, "Pengertian, model, Dan Tahapan SDLC (software development life cycle)," *Web developer LOGIQUE's Blog*, 28-Apr-2021. [Online]. Available: <https://www.logique.co.id/blog/2021/04/28/tahapan-sdlc/>. [Accessed: 27-May-2022].

- [8] G. W. Sasmito, "Penerapan Metode Waterfall Pada Desain Sistem Informasi Geografis Industri Kabupaten Tegal," *Jurnal Informatika: Jurnal Pengembangan IT (JPIT)*, vol. 2, no.1, Jan. 2017.
- [9] I. T. Kusnadi and A. Supiandi, "Implementasi Agile Mmethode pada Sistem Informasi Penjualan Alat Olahraga Berbasis Web," *Jurnal Informatika(JURIN)*, vol. 3, no.2, pp. 28-36, Maret 2021.
- [10] Sukamto, R. A. dan Shalahudin, M.. Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek. Bandung : Informatika Bandung, 2016.
- [11] 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.
- [12] 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.
- [13] A. Rahman, "Rapid Application Development Sistem Pembelajaran Daring Berbasis Android," *Jurnal Intech*, vol. 1, no. 2, pp. 20-25, Nov. 2020.
- [14] 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.
- [15] Hikmat, Dr. Harry. 2010. *Monitoring dan Evaluasi Proyek*.
- [16] Katsuhiko Ogata. "Teknik kontrol automatik", PT penerbit erlangga-SIMON. &SCHUTER (ASUA) Pte.ltd., 1997.
- [17] N. Serrano, J. Hernantes and G. Gallardo. Mobile Web Apps. *IEEE Software*, vol. 30, no. 5, 2013, pp. 22 -27.
- [18] Ganda, Yusmi P. W., Happy Flutter. Cetakan 1. Tangerang Selatan : Al Qolam, 2019.
- [19] 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].

- [20] OCI, “What is a database?,” *Oracle*, 2022. [Online]. Available: <https://www.oracle.com/database/what-is-database/>. [Accessed: 11-Feb-2022].
- [21] “What is a flowchart?,” *ASQ*. [Online]. Available: <https://asq.org/quality-resources/flowchart>. [Accessed: 15-Feb-2022].
- [22] 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].
- [23] 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].
- [24] Lucid Software, “UML use case diagram tutorial,” *Lucidchart*. [Online]. Available: <https://www.lucidchart.com/pages/uml-use-case-diagram>. [Accessed: 24-Feb-2022].
- [25] F. Mandrelli, “UML use case diagram,” *Figma*, 2021. [Online]. Available: <https://www.figma.com/community/file/986330591099819762>. [Accessed: 24-Feb-2022].
- [26] R. Habibi and R. Aprilian, Tutorial dan Penjelasan Aplikasi E-Office Berbasis Web Menggunakan Metode RAD, Bandung: Kreatif Industri Nusantara, 2020.
- [27] Iskandaria. 2012. Contoh Pengujian Black Box
- [28] Mutiara, A. B., Awaludin, R., Muslim, A. and T. Oswari, “Testing Implementasi Website Rekam Medis Elektronik Opeltnunasys Dengan Metode Acceptance Testing,” 2014.
- [29] Sugiyono, *Memahami Penelitian Kualitatif*. Bandung, Indonesia: Alfabeta, 2012.
- [30] Riduwan, *Belajar mudah penelitian untuk guru-karyawan dan peneliti pemula*. Bandung, Indonesia: Alfabeta, 2009.

LAMPIRAN

