

BAB III

METODOLOGI PERCOBAAN

3.1 Metodologi Percobaan

Dalam penelitian, diperlukan suatu metodologi penelitian yang berperan sebagai panduan dalam menjalankan penelitian. Keberadaan metodologi penelitian ini bertujuan untuk mengatur alur penelitian agar lebih sistematis sehingga proses penelitian dapat berjalan dengan baik dan terorganisir.

1. Studi Literatur

Studi literatur dilakukan bertujuan untuk menemukan informasi mengenai perancangan aplikasi, membantu pengembang untuk memahami kebutuhan pengguna, mengetahui tren teknologi terkini, serta mendapatkan inspirasi desain dari aplikasi sukses yang sudah ada. Dengan menganalisis literatur terkait, pengembang dapat memperoleh pemahaman mendalam tentang persyaratan fungsional yang diperlukan, menghindari kesalahan yang telah terjadi pada penelitian aplikasi sebelumnya, dan bahkan menggunakan informasi tersebut sebagai dasar untuk inovasi.

2. Analisis Kebutuhan dan Konsep Awal Aplikasi

Analisis kebutuhan pengguna bertujuan untuk memastikan bahwa aplikasi yang dikembangkan dapat memenuhi harapan, kebutuhan, dan preferensi pengguna dengan sebaik mungkin. Dengan melakukan analisis ini, tim pengembangan dapat memahami secara mendalam siapa pengguna potensial aplikasi, masalah apa yang terjadi pada mereka, dan bagaimana aplikasi dapat memberikan solusi yang paling efektif dan memuaskan bagi mereka. Hal ini melibatkan pengumpulan dan evaluasi informasi tentang preferensi pengguna, kebiasaan penggunaan, tantangan yang dihadapi, serta harapan dan tujuan pengguna dalam menggunakan aplikasi.

3. Perancangan Desain Aplikasi

Perancangan desain aplikasi bertujuan untuk menciptakan UI (*User Interface*) dan UX (*User Experience*) yang intuitif, efisien, dan memberikan kesan yang baik kepada pengguna. Perancangan awal berupa desain logo aplikasi, desain *low fidelity* (*wireframe*) berupa gambar manual pada kertas dan *high fidelity* (*mockup*) dari tampilan UI aplikasi yang dibuat menggunakan software Figma.

Setelah didapat rancangan awal, kemudian dilanjut dengan mengimplementasikan desain UI tersebut di *software* android studio dalam bentuk file XML.

4. Perancangan Program Aplikasi

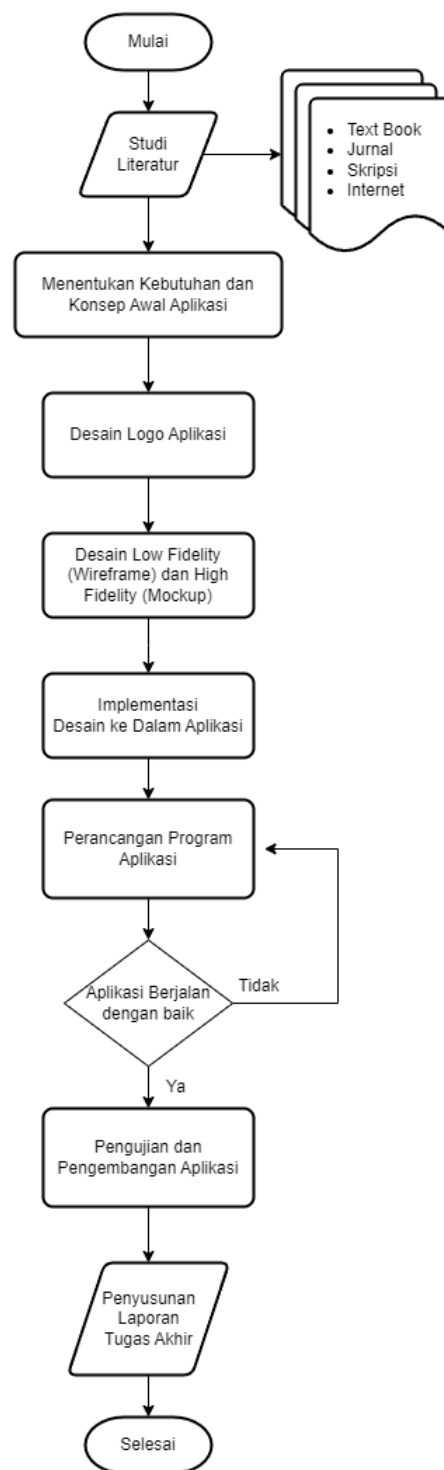
Perancangan program aplikasi bertujuan untuk membangun infrastruktur android yang baik dan terstruktur. Hal ini melibatkan proses merencanakan struktur keseluruhan aplikasi, termasuk arsitektur perangkat lunak, desain database, serta pemilihan arsitektur dan metode yang tepat untuk membangun aplikasi yang baik dan efisien. Perancangan infrastruktur aplikasi Android juga mempertimbangkan aspek keamanan, skalabilitas, dan kinerja aplikasi. Perancangan aplikasi ini menggunakan *software* android studio dengan Bahasa pemrograman Kotlin. Selain itu ditambahkan beberapa komponen untuk membangun aplikasi seperti *activity*, *fragment*, desain XML, dan komponen utama lainnya.

5. Pengujian dan Pengembangan Aplikasi

Pengujian dan pengembangan aplikasi bertujuan untuk memverifikasi dan memastikan bahwa aplikasi tersebut sesuai dengan standar kualitas serta memastikan kinerja dari aplikasi tetap optimal dan relevan. Selain itu, pada proses pengujian dan pengembangan aplikasi, pengembang akan memastikan fungsionalitas yang sesuai, dimana semua fitur beroperasi dengan baik. Selain itu, pengujian dilakukan untuk mengevaluasi performa aplikasi, memeriksa kompatibilitas dengan berbagai perangkat dan versi Android, serta memastikan kualitas antarmuka pengguna yang baik. Pengujian aplikasi dilakukan dengan metode *System Usability Scale (SUS)* untuk mengetahui keandalan aplikasi dan respon dari pengguna secara langsung.

3.2 Diagram Alir

Dalam pelaksanaan penelitian ini, penting untuk menerapkan diagram alir (flowchart) guna memfasilitasi penciptaan suatu urutan perencanaan yang lebih terstruktur dan terorganisir. Diagram alir memberikan gambaran visual tentang langkah-langkah atau proses yang terlibat dalam penelitian, mulai dari tahap awal hingga tahap akhir. Berikut merupakan diagram alir dari penelitian perancangan aplikasi Collabolio.



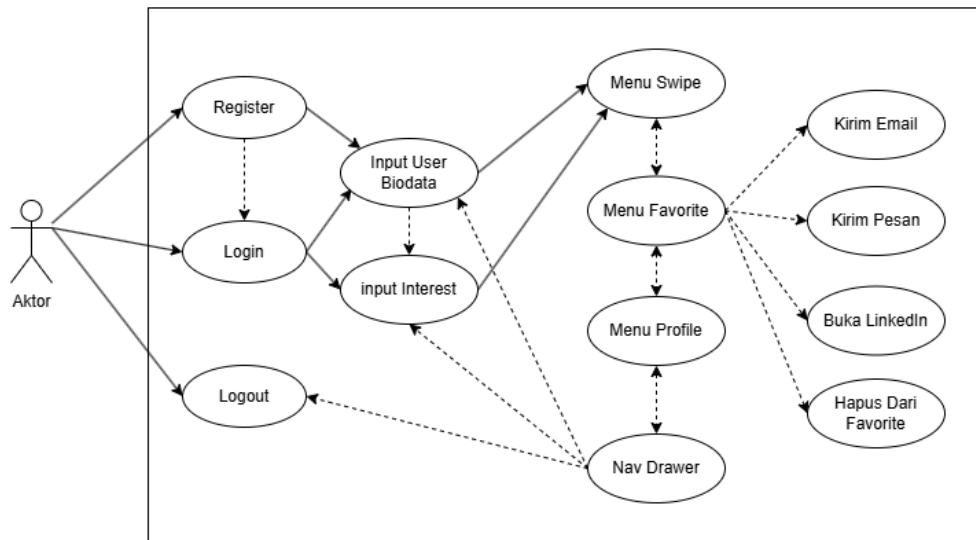
Gambar 3.1 Diagram Alir Penelitian

3.3 Skenario Aplikasi

Untuk scenario aplikasi disini menggunakan pendekatan dari *use case diagram* dan *activity diagram*.

3.3.1 Use Case Diagram

Use Case Diagram adalah salah satu jenis diagram UML (Unified Modeling Language) yang mengilustrasikan interaksi antara sistem dan aktor yang terlibat.



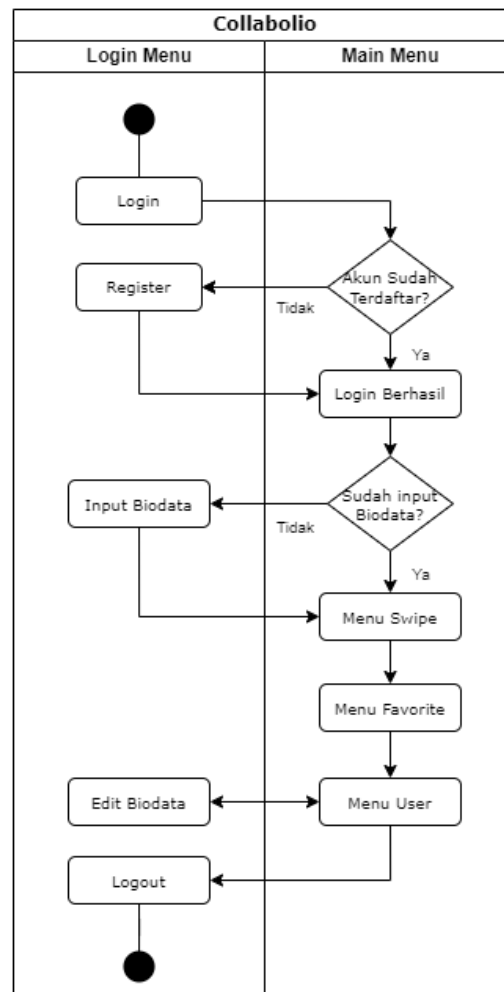
Gambar 3.2 Use Case Diagram

Berdasarkan gambar diatas, aktor disini dapat melakukan register, login dan logout. Ketika aktor berhasil melakukan register dan login, pengguna akan diminta untuk mengisi biodata, kemudian pengguna juga diminta untuk mengisi interest atau keahlian rekan yang dibutuhkan. Setelah itu, pengguna akan dibawa ke menu utama yang terdapat 3 tab tampilan yaitu menu swipe, menu *favorite*, dan menu profile. Pada menu *favorite*, pengguna dapat melakukan beberapa aksi kepada pengguna yang disukai, seperti mengirim email, mengirim pesan, membuka profile linkedIn rekan, dan menghapus rekan dari daftar *favorite*. Selain itu terdapat juga *navigation drawer* untuk melakukan pengeditan data, membuka menu *about*, dan melakukan *logout*.

3.3.2 Activity Diagram

Diagram aktivitas (*activity diagram*) adalah salah satu jenis diagram dalam UML (*Unified Modeling Language*) yang digunakan untuk menggambarkan alur kerja atau perilaku proses dalam sistem perangkat lunak. Diagram ini menunjukkan

serangkaian aktivitas, tindakan, dan keputusan yang terjadi dalam suatu proses dari awal hingga akhir. Berikut *Activity* diagram dari aplikasi Collabolio.



Gambar 3.3 *Activity Diagram*

3.4 Komponen Penelitian

Dalam penelitian ini, komponen penelitian menjadi elemen penting yang membentuk kerangka kerja Perancangan Aplikasi Collabolio Sebagai Platform Sosial-Kolaboratif Berbasis Android. Berikut komponen yang digunakan untuk penelitian.

3.4.1 Spesifikasi Alat

Untuk mendukung pengembangan aplikasi Collabolio, diperlukan alat-alat berikut dengan spesifikasi yang memadai yaitu

1. Laptop

Laptop digunakan untuk menjalankan berbagai jenis perangkat lunak seperti aplikasi, IDE (*Integrated Development Environment*), atau bahkan untuk mengakses data dan sumber daya online. Dalam konteks penelitian, laptop umumnya digunakan untuk melakukan pengolahan data, analisis, pengembangan perangkat lunak, penulisan laporan, dan tugas-tugas lainnya terkait penelitian. Spesifikasi laptop yang digunakan untuk penelitian yaitu AMD Athlon Gold 3150u dengan ram 4gb.

2. Internet

Dalam konteks penelitian, akses internet memiliki peran krusial. Internet digunakan untuk mengumpulkan informasi, data, dan literatur relevan dengan topik penelitian. Hal ini memungkinkan akses ke basis data, jurnal ilmiah, dan sumber daya online lainnya yang mendukung penelitian.

3. Android Studio IDE

Android Studio IDE merupakan komponen yang krusial dalam konteks penelitian yang berfokus pada pengembangan aplikasi Android. IDE ini menyediakan berbagai fitur yang memungkinkan peneliti untuk mendesain tampilan, merancang kode program, dan menganalisis aplikasi Android dengan efisien.

4. Perangkat Android

Perangkat android digunakan untuk melakukan uji coba atau testing aplikasi untuk mengetahui apakah aplikasi dapat berjalan dengan baik atau masih terdapat kesalahan atau bug pada aplikasi. Spesifikasi perangkat yang digunakan adalah perangkat dengan prosessor 8 core, ram 4gb, dan menggunakan versi android 10.

3.4.2 Spesifikasi Aplikasi

Bahasa pemrograman yang digunakan dalam penelitian ini adalah bahasa pemrograman Kotlin. Kemudian komponen-komponen yang ditambahkan ke dalam aplikasi yaitu firebase auth untuk proses autentikasi login dan register, firebase firestore untuk penyimpanan data secara cloud, *Card Swipe Layout* untuk tampilan utama aplikasi untuk menggeser *list* pengguna, okhttp untuk memanggil API hasil *training* data machine learning, *Recyclerview* untuk menampilkan list

data *favorite* pengguna, dan *Room Database* untuk menyimpan data pengguna yang disukai secara local.

3.5 Skenario Pengujian Aplikasi

Metode pengujian yang digunakan pada penelitian ini yaitu metode *system usability scale (SUS)*. Metode SUS bersifat *quick and dirty* dan berbasiskan pada kuesioner yang terdiri dari sepuluh pertanyaan dengan jawaban skala 1 sampai 5, yang dimulai dari Sangat Tidak Setuju sampai Sangat Setuju. Metode ini tidak membutuhkan banyak sampel sebagai sumber datanya sehingga metode SUS ini tergolong metode yang efektif, murah dan paling umum digunakan [42]. Pertanyaan yang digunakan dalam pengujian SUS ini adalah sebagai berikut.

Tabel 4.3 Tabel Pertanyaan SUS

No	Pertanyaan
Q1	"Saya berpikir akan menggunakan sistem ini lagi."
Q2	"Saya merasa sistem ini rumit untuk digunakan."
Q3	"Saya merasa sistem ini mudah digunakan."
Q4	"Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini."
Q5	"Saya merasa fitur-fitur sistem ini berjalan dengan semestinya."
Q6	"Saya merasa ada banyak hal yang tidak konsisten (tidak sesuai pada sistem ini)."
Q7	"Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat."
Q8	"Saya merasa sistem ini membingungkan."
Q9	"Saya merasa tidak ada hambatan dalam menggunakan sistem ini."
Q10	"Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini."

Kemudian dalam melakukan perhitungan skor SUS terdapat beberapa aturan sebagai berikut :

1. Pertanyaan Ganjil

Untuk setiap pertanyaan bernomor ganjil (1, 3, 5, 7, 9), nilai pengguna dikurangi dengan 1.

Rumus : $[Skor\ Pertanyaan\ Ganjil = Penilaian\ pengguna - 1]$.

2. Pertanyaan Genap

Untuk setiap pertanyaan bernomor genap (2, 4, 6, 8, 10), nilai maksimum (5) dikurangi dengan penilaian pengguna.

Rumus : $[Skor\ Pertanyaan\ Genap = 5 - Penilaian\ Pengguna]$.

3. Menghitung Skor Responden

Jumlahkan skor dari semua pertanyaan (ganjil dan genap) untuk setiap responden kemudian hasilnya dikalikan dengan 2,5 untuk mendapatkan skor responden.

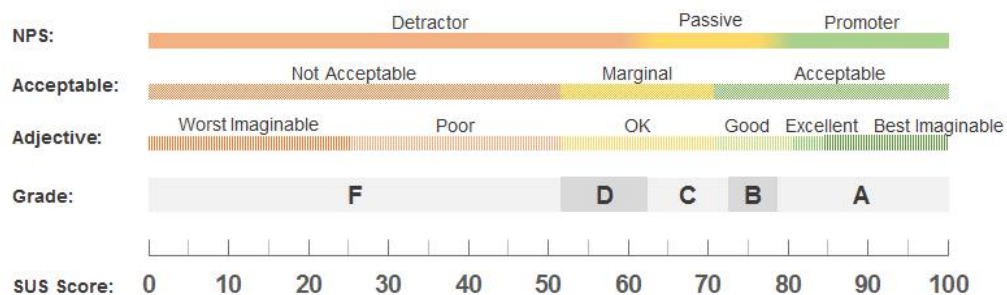
Rumus : $(Total\ Skor\ Semua\ Pertanyaan) \times 2,5 = Skor\ Responden$.

4. Menghitung Skor SUS

Untuk menghitung nilai rata-rata, jumlahkan semua skor responden dan bagi dengan jumlah responden.

Rumus : $\frac{Total\ Skor\ Responden}{Jumlah\ Responden} = Skor\ SUS$.

Setelah didapatkan hasil skor rata-rata SUS, dilanjutkan ke proses interpretasi skor SUS. Dalam menginterpretasikan hasil skor SUS, terdapat lima cara yang bisa digunakan yaitu dengan berdasarkan pada interpretasi perbandingan peringkat persentil, peringkat sifat (*adjective*), tingkat penerimaan (*acceptable*), dan NPS dari skor SUS itu sendiri [42]. Hasil akhirnya dapat di bandingkan dengan gambar skala interpretasi berikut.



Gambar 3.4 Skala Interpretasi Hasil Skor SUS [42].

3.6 Tempat Penelitian

Lokasi penelitian adalah tempat atau objek untuk diadakan suatu penelitian. Lokasi penelitian berada di rumah penulis, yaitu Kelurahan Blokan, Kecamatan Bandung, Kabupaten Serang, Provinsi Banten. Lokasi penelitian ini dipilih berdasarkan pelaksanaan penelitian yang diadakan oleh mitra Kampus Merdeka, yaitu Yayasan Dicoding Indonesia, yang dilaksanakan secara daring atau online dengan menggunakan perangkat komputer/laptop.

3.7 Pengambilan Data

Pengambilan data merupakan tahap kunci dalam proses penelitian Perancangan Aplikasi Collabolio Sebagai Platform Sosial Kolaboratif Berbasis Android. Penelitian ini membutuhkan data yang akurat mengenai fitur yang perlu di masukkan ke aplikasi, metode pemrograman yang di gunakan agar bisa berjalan secara optimal di perangkat android, serta desain UI/UX yang menarik agar pengguna dapat merasakan pengalaman yang impresif saat menggunakan aplikasi. Pengambilan data dilakukan dengan mengambil beberapa sumber yaitu referensi aplikasi Tinder, Bumble dan LinkedIn, referensi dari jurnal penelitian terkait terdahulu, panduan developer android yang disediakan oleh google serta panduan yang diberikan oleh platform Dicoding. Kemudian dilakukan juga pengujian keandalan aplikasi lewat metode *System Usability Scale* (SUS) yaitu pengujian langsung kepada beberapa responden untuk menjalankan aplikasi yang sudah dibuat dan kemudian responden akan mengisi form yang disediakan.