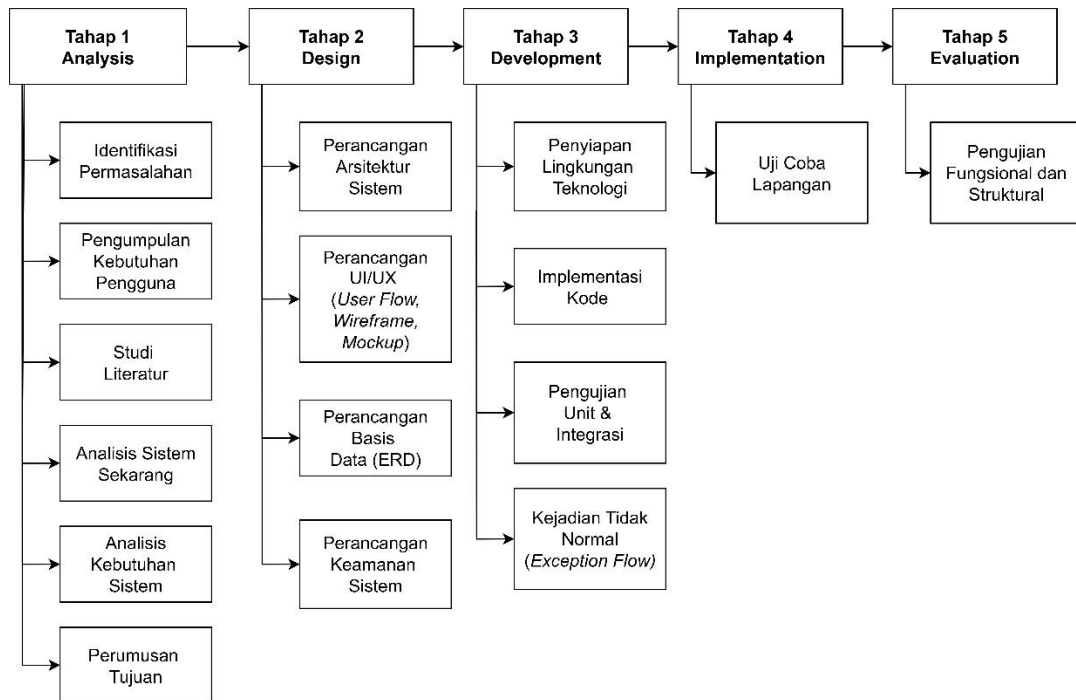


BAB III

METODE PENELITIAN

3.1 Desain Penelitian



Gambar 3.1 Desain Penelitian Metode ADDIE

Penelitian ini menggunakan metode ADDIE sebagai kerangka kerja pengembangan perangkat lunak. ADDIE (*Analysis, Design, Development, Implementation, Evaluation*) adalah model yang umum digunakan untuk mengelola proyek pengembangan dan pendidikan, dan sangat efektif dalam memastikan bahwa tiap tahap dikembangkan dengan baik. Pada penelitian ini, model ADDIE diadaptasi untuk mengatasi permasalahan dalam pengelolaan data penumpang di Bandara Soekarno-Hatta.

3.1.1 Analysis

Tahap *Analysis* merupakan langkah awal dalam metode ADDIE yang bertujuan untuk mengidentifikasi kebutuhan dan masalah yang akan diselesaikan melalui pengembangan sistem. Pada penelitian ini, tahap analisis difokuskan pada pengumpulan data dan informasi yang diperlukan untuk merancang sistem yang efektif dalam mendukung proses boarding penumpang di Bandara Internasional

Soekarno-Hatta. Dalam tahap ini, beberapa kegiatan penting dilakukan sebagai berikut:

a. Identifikasi Permasalahan

Langkah pertama adalah mengidentifikasi permasalahan terkait pengumpulan data penumpang saat boarding di bandara. Peneliti akan melakukan observasi dan diskusi informal dengan pihak terkait, seperti petugas bandara atau manajemen bandara, untuk memahami tantangan yang dihadapi dalam proses pendataan penumpang. Tujuannya adalah untuk mengetahui kendala-kendala yang ada, terutama yang berkaitan dengan akurasi dan kecepatan pendataan.

b. Pengumpulan Kebutuhan Pengguna

Setelah masalah teridentifikasi, tahap selanjutnya adalah mengumpulkan kebutuhan pengguna. Peneliti menyusun daftar kebutuhan sistem berdasarkan masukan dari pengguna, yaitu petugas yang melakukan pemindaian *Boarding pass* dan pihak manajemen yang membutuhkan data penumpang untuk analisis operasional. Fokus utama adalah pada kebutuhan akan data yang *real-time*, keakuratan informasi, dan kemudahan penggunaan sistem.

c. Analisis Sistem Sekarang

Setelah kebutuhan pengguna awal teridentifikasi, langkah selanjutnya adalah melakukan analisis mendalam terhadap sistem yang berjalan saat ini (sistem manual). Analisis ini dilakukan melalui metode observasi langsung (*direct observation*) di lapangan, yaitu di area *Security Checkpoint* (SCP) Bandara Internasional Soekarno-Hatta. Selama observasi, peneliti mengamati dan mencatat alur kerja (*workflow*) petugas *Aviation Security* (Avsec) secara rinci, mulai dari tahap instruksi kepada penumpang hingga proses validasi dokumen secara visual. Hasil dari observasi ini kemudian dipetakan untuk mengidentifikasi kelemahan fundamental dalam sistem manual, seperti titik-titik yang menyebabkan penumpukan antrean (*bottleneck*), potensi terjadinya kesalahan manusia (*human error*), dan ketiadaan perolehan data operasional. Temuan dari analisis ini menjadi dasar justifikasi untuk perancangan sistem baru dan perumusan kebutuhan fungsional yang lebih spesifik.

d. Analisis Kebutuhan Sistem

Berdasarkan hasil identifikasi masalah dan kebutuhan pengguna, peneliti menyusun spesifikasi kebutuhan sistem yang dikembangkan. Hal ini mencakup kebutuhan fungsional, seperti kemampuan aplikasi *mobile* untuk memindai *Boarding pass* secara otomatis, serta kebutuhan non-fungsional, seperti keandalan sistem dan antarmuka yang user-friendly.

e. Studi Literatur dan Analisis Konteks

Peneliti juga akan melakukan studi literatur untuk mempelajari solusi serupa yang telah diterapkan di bandara lain atau di sektor industri yang relevan. Hal ini bertujuan untuk mendapatkan wawasan mengenai praktik terbaik (*best practices*) yang dapat diadaptasi dalam pengembangan sistem ini.

f. Perumusan Tujuan Pengembangan Sistem

Berdasarkan analisis yang dilakukan, peneliti akan merumuskan tujuan dari sistem yang akan dikembangkan. Sistem ini direncanakan untuk mendukung operasional bandara dengan menyediakan data penumpang yang lebih akurat dan *real-time* melalui aplikasi *mobile* untuk pemindaian *Boarding pass* dan *Dashboard* berbasis *web* untuk *monitoring* data.

Tahap analisis ini menjadi landasan bagi tahapan selanjutnya dalam metode ADDIE, yaitu *Design*, di mana peneliti akan merancang detail sistem berdasarkan kebutuhan yang telah diidentifikasi pada tahap ini. Hasil dari tahap Analysis ini akan digunakan untuk menyusun spesifikasi desain yang lebih rinci, guna memastikan bahwa sistem yang dikembangkan dapat memenuhi kebutuhan pengguna dengan efektif.

3.1.2 Design

Pada tahap Design, seluruh hasil dari tahap analisis diterjemahkan ke dalam sebuah cetak biru (*blueprint*) teknis. Tahap ini difokuskan pada perancangan arsitektur sistem, antarmuka pengguna (UI), pengalaman pengguna (UX), dan struktur basis data. Tujuannya adalah untuk merumuskan spesifikasi rinci sebagai panduan pada tahap pengembangan (Development). Langkah-langkah perancangan yang dilakukan dalam penelitian ini meliputi:

a. Perancangan Arsitektur Sistem

Langkah pertama dalam tahap perancangan adalah memodelkan arsitektur sistem secara keseluruhan. Proses ini bertujuan untuk mendefinisikan bagaimana komponen-komponen utama yaitu aplikasi *mobile* untuk pemindaian, *server backend*, basis data, dan *dashboard monitoring* akan saling berinteraksi. Arsitektur dirancang menggunakan model *client-server*, di mana aplikasi *mobile* dan *dashboard web* bertindak sebagai *client* yang berkomunikasi dengan *server* melalui *API (Application Programming Interface)*.

b. Perancangan Antarmuka dan Pengalaman Pengguna (UI/UX)

Perancangan UI/UX dilakukan untuk memastikan sistem mudah digunakan (*user-friendly*), efisien, dan intuitif bagi pengguna target (petugas lapangan dan manajemen). Proses perancangan ini mengikuti beberapa langkah:

1. Pembuatan Alur Pengguna (*User Flow*): Memetakan alur interaksi pengguna untuk setiap skenario, seperti proses petugas melakukan *login* hingga berhasil memindai *boarding pass*, dan proses manajemen mengakses data analitik pada *dashboard*.
2. Perancangan *Wireframe*: Membuat sketsa atau kerangka dasar (tanpa warna dan detail visual) untuk setiap halaman pada aplikasi *mobile* dan *dashboard*. *Wireframe* difokuskan untuk mengatur tata letak elemen, navigasi, dan struktur informasi secara fungsional.
3. Perancangan *Mockup* (Desain Visual): Mengembangkan *wireframe* menjadi desain visual berketelitian tinggi (*high-fidelity*). Pada langkah ini, dilakukan penentuan palet warna, tipografi, ikon, dan komponen visual lainnya yang selaras dengan identitas jenama (*brand identity*) Angkasa Pura dan InJourney.

c. Perancangan Basis Data

Langkah perancangan basis data bertujuan untuk merancang struktur penyimpanan data yang logis, efisien, dan terukur. Proses yang dilakukan adalah sebagai berikut:

1. Identifikasi Entitas dan Atribut: Mengidentifikasi semua entitas data yang dibutuhkan sistem, seperti data penumpang, data penerbangan, dan data

petugas. Selanjutnya, setiap entitas dijabarkan atribut-atributnya (misalnya, nama penumpang, nomor penerbangan, dll.).

2. Pemodelan Data dengan *Entity-Relationship Diagram* (ERD): Membuat ERD untuk memvisualisasikan hubungan antar entitas data. ERD ini membantu memastikan integritas dan konsistensi data di seluruh sistem.
3. Perancangan Skema Fisik Tabel: Menerjemahkan ERD menjadi skema fisik basis data. Pada langkah ini, nama tabel, nama kolom, tipe data, dan primary key ditentukan secara rinci. Tabel utama yang dirancang adalah tabel untuk menyimpan data *Passenger Name Record* (PNR).
4. Pemilihan Sistem Manajemen Basis Data (DBMS): Berdasarkan kebutuhan akan skalabilitas, performa *query*, dan dukungan komunitas yang luas, MySQL dipilih sebagai DBMS yang akan digunakan untuk mengimplementasikan rancangan basis data ini.

d. Perancangan Keamanan Sistem

Dalam tahap perancangan, aspek keamanan sistem menjadi fokus utama untuk menjamin integritas dan kerahasiaan data. Perancangan arsitektur keamanan dilakukan melalui serangkaian langkah metodis yang terstruktur sebagai berikut:

1. Identifikasi Konteks Platform dan Risiko

Langkah awal adalah membedakan model interaksi untuk kedua platform. Aplikasi *mobile* diidentifikasi sebagai klien *stateless* yang membutuhkan mekanisme otentikasi per permintaan, sementara *dashboard* web adalah klien *stateful* yang dapat memanfaatkan sesi *browser*. Risiko yang diidentifikasi meliputi penyadapan data, akses tidak sah ke API, dan pembajakan sesi (*session hijacking*) pada web.

2. Perancangan Skema Otentikasi Pengguna.

Berdasarkan konteks platform, dirancang dua skema otentikasi yang berbeda namun sama-sama aman:

- Untuk Aplikasi Mobile (API): Dirancang skema otentikasi berbasis *Bearer Token* menggunakan Laravel Sanctum. Alur ini dipetakan mulai dari verifikasi kredensial, penerbitan *token*, hingga penyimpanan aman di sisi klien.

- Untuk Dashboard Web: Dirancang skema otentikasi berbasis sesi (*session-based*) standar web menggunakan *middleware* bawaan Laravel. Alur ini mencakup proses validasi kredensial melalui `Auth::attempt`, pembuatan sesi aman di sisi *server*, dan penggunaan *cookie* sesi di *browser* pengguna.
3. Perancangan Mekanisme Otorisasi.
Ditetapkan mekanisme otorisasi yang sesuai untuk masing-masing platform. Untuk API, setiap permintaan divalidasi menggunakan *Bearer Token*. Untuk *dashboard* web, setiap akses ke halaman yang dilindungi divalidasi melalui pemeriksaan sesi aktif oleh *middleware* auth.
 4. Perancangan Lapisan Keamanan Tambahan.
Untuk API *mobile*, dirancang lapisan verifikasi integritas menggunakan *header* guid (SHA-256). Untuk *dashboard* web, dirancang penggunaan fitur keamanan bawaan Laravel seperti *Session Regeneration* (`$request->session()->regenerate()`) setelah *login* berhasil untuk mencegah serangan *session fixation*.
 5. Perancangan Manajemen Sesi Klien.
 - Untuk Aplikasi Mobile: Sesi dikelola di sisi klien dengan masa berlaku token selama satu jam.
 - Untuk Dashboard Web: Sesi dikelola di sisi server dan diakhiri melalui proses logout yang aman, yang mencakup `Auth::logout()`, *invalidasi* sesi, dan regenerasi token CSRF.
 6. Identifikasi Kebutuhan Enkripsi Transportasi Data.
Sebagai langkah final, diidentifikasi kebutuhan mutlak penggunaan protokol HTTPS (SSL/TLS) untuk kedua platform saat implementasi di lingkungan produksi, guna mengenkripsi seluruh komunikasi dan melindungi baik *token* API maupun *cookie* sesi.

3.1.3 Development (Pengembangan)

Pada tahap Development, seluruh spesifikasi dan cetak biru (blueprint) yang telah dirumuskan pada tahap perancangan akan diimplementasikan menjadi produk

fungsional. Tahap ini merupakan proses konstruksi teknis sistem, yang mencakup penulisan kode untuk aplikasi *mobile*, *server backend*, dan *dashboard* monitoring.

Langkah-langkah pengembangan yang akan dilakukan dalam penelitian ini meliputi:

a. Penyiapan Lingkungan dan Pemilihan Teknologi

Langkah awal dalam tahap pengembangan adalah penyiapan lingkungan kerja dan finalisasi teknologi berdasarkan rancangan yang telah dibuat. Berikut pertimbangan yang diputuskan:

1. *Framework Aplikasi Mobile*: Implementasi aplikasi *mobile* akan menggunakan *framework* Flutter. Pemilihan ini didasarkan pada kebutuhan pengembangan lintas platform (*cross-platform*) yang efisien, sehingga aplikasi dapat berjalan di sistem operasi Android dan iOS dengan satu basis kode. Kemampuan Flutter dalam menciptakan antarmuka yang responsif akan digunakan untuk merealisasikan desain *Mockup* yang telah disetujui.
2. *Framework Backend dan Dashboard*: Untuk sisi *server* dan *dashboard web*, pengembangan akan memanfaatkan *framework* Laravel. Laravel dipilih karena menyediakan fondasi yang kokoh untuk membangun API, mengelola keamanan, dan berinteraksi secara efisien dengan basis data MySQL yang telah dirancang. Fitur-fitur bawaan Laravel akan dipergunakan untuk mempercepat pengembangan fitur analitik pada *dashboard*.

b. Implementasi dan Penulisan Kode (*Coding*)

Proses ini adalah inti dari tahap pengembangan, di mana rancangan diterjemahkan menjadi baris-baris kode. Pengembangan akan dibagi menjadi tiga modul utama yang dikerjakan secara paralel dan terintegrasi:

1. Pengembangan *Backend* (API): Membangun Application Programming Interface (API) sebagai jembatan komunikasi antara aplikasi *mobile*, *dashboard*, dan basis data. Proses ini mencakup pembuatan endpoint untuk proses otentikasi petugas, pengiriman data hasil pemindaian, dan pengambilan data untuk ditampilkan pada *dashboard*.

2. Pengembangan Aplikasi *Mobile*: Mengimplementasikan seluruh rancangan antarmuka (UI) dan alur pengguna (UX) menjadi aplikasi fungsional. Fitur utama yang akan dikembangkan meliputi modul *login* petugas, antarmuka kamera untuk pemindaian *boarding pass*, serta logika untuk mengirimkan data PNR yang terekstraksi ke *server*.
 3. Pengembangan *Dashboard Monitoring*: Membangun antarmuka *dashboard web* sesuai dengan *Mockup*. Proses ini mencakup pembuatan halaman visualisasi data, tabel data penumpang, dan fitur filter data penerbangan yang terhubung langsung dengan data dari *server* melalui API.
- c. Pengujian Unit dan Integrasi

Setelah penulisan kode pada setiap modul selesai, akan dilakukan serangkaian pengujian untuk memastikan fungsionalitas dan keandalan sistem.

1. Pengujian Unit (*Unit Testing*): Setiap komponen atau fungsi kecil (misalnya, fungsi validasi *login*, fungsi pengiriman data) akan diuji secara terpisah untuk memastikan bekerja tanpa eror.
2. Pengujian Integrasi (*Integration Testing*): Menguji interaksi antar modul yang telah dikembangkan. Fokus utama pada tahap ini adalah untuk memverifikasi bahwa alur data dari aplikasi *mobile* saat memindai, lalu ke *server* untuk diproses, hingga ditampilkan pada *dashboard* berjalan dengan lancar, akurat, dan sesuai dengan rancangan arsitektur sistem. Proses debugging akan dilakukan secara iteratif untuk memperbaiki setiap masalah yang ditemukan.

3.1.4 Implementation

Uji coba lapangan menggunakan metode Studi Lapangan (*Field Study*). Metode ini dipilih karena bertujuan untuk menguji aplikasi secara langsung di lokasi kerjanya yang sebenarnya. Menguji aplikasi di bandara yang ramai tentu akan memberikan hasil yang lebih nyata. Dengan metode ini, peneliti bisa melihat langsung berbagai faktor penting di lapangan, seperti kondisi cahaya, gangguan suara, kondisi internet di lapangan, dan kesibukan petugas yang dapat memengaruhi cara kerja aplikasi.

Studi lapangan ini dilaksanakan pada 7 Juni 2024 di area *Security Check Point* (SCP) Terminal 3, Bandara Internasional Soekarno-Hatta. Partisipan yang dilibatkan adalah 2 (dua) orang petugas keamanan penerbangan (AVSEC) senior. Mereka dipilih secara sengaja karena pengalamannya yang luas, sehingga diharapkan dapat memberikan penilaian dan masukan yang akurat mengenai kelebihan dan kekurangan aplikasi ini dibandingkan cara kerja manual yang selama ini mereka lakukan.

Proses pengumpulan data dilakukan dengan langkah-langkah berikut:

- a. Pengarahan Kepada Petugas
- b. Memeriksa Boarding Pass Penumpang yang memasuki *security check point*
- c. Pengukuran waktu pemindaian dan observasi terhadap kegiatan

Selama proses tersebut, peneliti menggunakan dua alat utama: *stopwatch* untuk mencatat waktu pemindaian (data kuantitatif), dan buku catatan (*field notes*) untuk menuliskan semua pengamatan, komentar, dan kendala yang muncul (data kualitatif).

Data yang sudah terkumpul kemudian dianalisis dengan dua cara. Untuk data kualitatif dari catatan lapangan, semua catatan dibaca dan dikelompokkan untuk menemukan tema atau pola-pola penting. Untuk data kuantitatif berupa catatan waktu, data tersebut dianalisis secara statistik sederhana untuk menghitung waktu rata-rata, tercepat, dan terlama dari proses pemindaian.

3.1.5 Evaluation

Tahap evaluasi akan menjadi fase final yang bertujuan untuk mengukur efektivitas, kualitas, dan keberhasilan sistem secara objektif berdasarkan tujuan penelitian yang telah ditetapkan. Proses evaluasi akan menggunakan beberapa pendekatan pengujian standar sebagai berikut:

- a. Pengujian Fungsional (*Black Box Testing*) Fungsionalitas sistem akan diuji menggunakan pendekatan Black Box Testing. Pengujian ini akan dilakukan dari perspektif pengguna untuk memverifikasi bahwa setiap fitur berjalan sesuai dengan kebutuhan fungsional yang telah dirumuskan. Prosesnya akan mencakup perancangan skenario uji dan kasus uji (test case) yang diturunkan

dari alur pengguna (*user flow*) untuk menguji kondisi normal (berhasil) maupun kondisi alternatif (gagal/eror).

- b. Pengujian Struktur (*White Box Testing*) dilakukan untuk memverifikasi struktur internal dan logika dari kode program. Pendekatan ini berfokus pada pengujian jalur-jalur logika, percabangan, dan alur data di dalam kode untuk memastikan setiap fungsi bekerja dengan benar dari sisi teknis. Metode yang akan digunakan adalah Pengujian Unit dan Pengujian Integrasi. Pengujian unit akan diterapkan pada fungsi-fungsi krusial. Untuk memastikan akurasi ekstraksi data. Pengujian integrasi akan memastikan bahwa komunikasi data antara aplikasi *mobile*, API, dan basis data berjalan sesuai dengan rancangan.
- c. User Acceptance Questionnaire (UAQ)

Untuk mengukur penerimaan pengguna secara komprehensif, sebuah kerangka kerja yang solid dan telah teruji diperlukan. Model yang paling relevan dan telah diterima secara luas dalam riset sistem informasi adalah *Technology Acceptance Model* (TAM). TAM dipilih sebagai fondasi utama karena model ini secara langsung mengukur dua faktor kognitif kunci yang terbukti menjadi prediktor kuat terhadap niat dan perilaku penggunaan teknologi:

1. *Perceived Usefulness* (Persepsi Kegunaan): Sejauh mana seorang individu percaya bahwa menggunakan suatu sistem akan meningkatkan kinerja pekerjaannya.
2. *Perceived Ease of Use* (Persepsi Kemudahan Penggunaan): Sejauh mana seorang individu percaya bahwa menggunakan suatu sistem akan bebas dari usaha (baik fisik maupun mental).

Kedua konstruk ini sangat selaras dengan tujuan penelitian yang berfokus pada peningkatan efisiensi (kegunaan) dan penyediaan alat yang praktis (kemudahan penggunaan) bagi pengguna di Bandara Soekarno-Hatta. Meskipun TAM sangat efektif dalam memberikan gambaran umum mengenai tingkat penerimaan, model ini kurang memberikan diagnosis yang mendalam mengenai aspek-aspek spesifik dari pengalaman pengguna. Untuk mengatasi keterbatasan ini, kerangka TAM akan diperkaya dan dilengkapi dengan

konstruk-konstruk yang diadaptasi dari *User Experience Questionnaire* (UEQ). UEQ adalah instrumen yang dirancang untuk mengukur pengalaman pengguna secara holistik, mencakup aspek pragmatis (berorientasi tugas) dan hedonis (tidak berorientasi tugas). Untuk konteks sistem ini, konstruk pragmatis dari UEQ yang paling relevan akan diintegrasikan, yaitu:

- *Efficiency* (Efisiensi): Apakah pengguna dapat menyelesaikan tugas mereka tanpa usaha yang tidak perlu?.
- *Perspiciuity* (Kejelasan): Seberapa mudah untuk menjadi terbiasa dengan sistem dan mempelajari cara menggunakannya?.
- *Dependability* (Keandalan): Apakah pengguna merasa memegang kendali atas interaksi dan apakah sistem bekerja secara dapat diprediksi?.

Penggunaan kerangka hibrida ini memberikan sinergi yang kuat untuk evaluasi yang holistik. TAM akan menjawab pertanyaan "apakah" pengguna menerima sistem, sementara konstruk dari UEQ akan membantu menjelaskan "mengapa" dan "bagaimana" mereka sampai pada kesimpulan tersebut.

Operasionalisasi adalah proses menerjemahkan konsep-konsep teoretis yang abstrak (seperti *Perceived Usefulness*) menjadi item-item atau pernyataan kuesioner yang konkret, spesifik, dan terukur. Proses ini dilakukan secara terpisah untuk setiap profil pengguna untuk memastikan bahwa setiap pertanyaan relevan dengan konteks tugas dan kebutuhan unik mereka.

Untuk Petugas Aviation Security (Avsec):

- *Perceived Usefulness* (Kegunaan): Konstruk ini dioperasionalkan menjadi pertanyaan yang secara langsung mengukur apakah aplikasi *mobile* berhasil mengatasi *pain points* utama mereka yang teridentifikasi dalam Bab 4.1.1. Pertanyaan akan berfokus pada persepsi peningkatan kecepatan verifikasi, efektivitas dalam mengurangi risiko *human error*, dan peningkatan produktivitas kerja secara keseluruhan.
- *Perceived Ease of Use* (Kemudahan Penggunaan): Dioperasionalkan untuk mengukur aspek-aspek yang krusial bagi petugas di lapangan. Pertanyaan akan mengevaluasi kesederhanaan antarmuka, kemudahan pengoperasian

(misalnya, dengan satu tangan), dan kecepatan kurva belajar untuk menjadi terampil dalam menggunakan aplikasi.

- *Efficiency & Dependability* (Efisiensi & Keandalan): Konstruk gabungan dari UEQ ini akan mengukur persepsi petugas mengenai kelancaran alur kerja pemindaian. Pertanyaan akan fokus pada kecepatan respons sistem, konsistensi fungsional, dan tingkat kepercayaan diri yang dirasakan petugas saat menggunakan aplikasi untuk memvalidasi penumpang.

Untuk Manajemen Operasional:

- *Perceived Usefulness* (Kegunaan): Dioperasionalkan untuk mengukur sejauh mana *dashboard* memenuhi kebutuhan strategis manajemen. Pertanyaan akan berpusat pada kemampuan *dashboard* dalam menyediakan visibilitas data operasional secara *real-time*, mendukung pengambilan keputusan yang lebih proaktif (bukan reaktif), dan fungsinya sebagai alat bantu untuk analisis pola penumpang jangka panjang.
- *Perceived Ease of Use* (Kemudahan Penggunaan): Dioperasionalkan untuk mengevaluasi pengalaman interaksi dengan *dashboard* berbasis web. Pertanyaan akan fokus pada kemudahan navigasi antar halaman, efektivitas fitur pencarian dan penyaringan data, serta tata letak informasi secara umum.
- *Perspicuity & Dependability* (Kejelasan & Keandalan): Konstruk ini akan secara spesifik mengukur seberapa mudah data yang disajikan dapat dipahami dan dipercaya. Pertanyaan akan mengevaluasi kejelasan visualisasi data (grafik dan tabel) dan keyakinan manajemen terhadap akurasi dan keandalan data yang ditampilkan pada *dashboard*.

Untuk Struktur Kuisioner mengikuti praktik terbaik dalam desain survei, setiap kuesioner (baik untuk Avsec maupun Manajemen) dibagi menjadi tiga bagian utama untuk menciptakan alur pengisian yang logis dan komprehensif bagi responden:

1. Bagian A: Data Demografi Responden: Bagian ini bertujuan untuk mengumpulkan data kontekstual dasar mengenai responden. Informasi seperti jabatan, dan lama pengalaman kerja sangat penting untuk melakukan

analisis segmentasi di kemudian hari. Misalnya, untuk mengetahui apakah ada perbedaan persepsi antara petugas senior dan junior.

2. Bagian B: Evaluasi Inti (Skala Likert): Ini adalah bagian utama kuesioner yang berisi serangkaian pernyataan terstruktur untuk mengukur setiap konstruk teoretis yang telah dioperasionalkan (Kegunaan, Kemudahan Penggunaan, Efisiensi, dll.). Responden akan diminta untuk menyatakan tingkat persetujuan mereka terhadap setiap pernyataan. Untuk Bagian B, skala pengukuran yang digunakan adalah Skala Likert 5 poin. Skala ini merupakan standar industri dalam kuesioner penerimaan teknologi dan evaluasi usability karena beberapa keunggulan: mudah dipahami oleh responden, menyediakan gradasi respons yang cukup, dan datanya dapat dianalisis secara statistik dengan mudah. Rentang skala yang digunakan adalah:

- 1 = Sangat Tidak Setuju
- 2 = Tidak Setuju
- 3 = Netral
- 4 = Setuju
- 5 = Sangat Setuju

Sebuah teknik penting yang diadaptasi dari *System Usability Scale* (SUS) adalah penggunaan pernyataan yang dirancang secara bergantian antara kalimat positif dan negatif. Misalnya, setelah sebuah pernyataan positif seperti "Saya merasa aplikasi ini mudah digunakan," akan diikuti oleh pernyataan negatif seperti "Saya merasa alur kerja di aplikasi ini rumit."

Tabel 3.1 Kuisiomer Bagian B Petugas Aviation Security

No	Aspek	Pertanyaan	Skala
1	<i>Perceived Ease of Use</i>	Saya merasa aplikasi ini sangat mudah untuk digunakan.	1-5
2		Saya merasa alur kerja untuk memindai <i>boarding pass</i> di aplikasi ini rumit.	1-5

3		Saya dapat dengan cepat menjadi terampil dalam menggunakan aplikasi ini.	1-5
4		Saya pikir saya akan membutuhkan dukungan dari orang teknis untuk dapat menggunakan sistem ini.	1-5
5	<i>Perceived Usefulness</i>	Menggunakan aplikasi ini secara signifikan mempercepat proses verifikasi penumpang dibandingkan cara manual.	1-5
6		Saya merasa aplikasi ini tidak memberikan banyak manfaat untuk efektivitas pekerjaan saya sehari-hari.	1-5
7		Aplikasi ini sangat membantu saya mengurangi kemungkinan kesalahan (<i>human error</i>) saat memeriksa <i>boarding pass</i> .	1-5
8		Saya yakin penggunaan aplikasi ini membuat pekerjaan saya menjadi lebih produktif.	1-5
9	<i>Efficiency & Dependability</i>	Saya merasa proses dari memindai hingga menampilkan hasil di aplikasi ini sangat efisien dan tanpa jeda.	1-5
10		Saya merasa ada terlalu banyak inkonsistensi dalam sistem ini.	1-5
11		Saya merasa sangat percaya diri saat menggunakan aplikasi ini untuk memvalidasi penumpang.	1-5
12		Saya mendapati sistem ini sangat merepotkan untuk digunakan.	1-5

Tidak jauh berbeda dengan kuisioner petugas Aviation Security, berikut kuisioner untuk manajemen operasional.

Tabel 3.2 Kuisisioner Bagian B Manajemen Operasional

No	Aspek	Pertanyaan	Skala
1	<i>Perceived Ease of Use</i>	Saya merasa <i>dashboard</i> ini sangat mudah untuk dinavigasi.	1-5
2		Saya merasa informasi yang disajikan dalam bentuk grafik dan tabel sulit untuk dipahami.	1-5
3		Sangat mudah bagi saya untuk menemukan data spesifik yang saya butuhkan menggunakan fitur filter yang tersedia.	1-5
4		Saya merasa perlu belajar banyak hal terlebih dahulu sebelum bisa menggunakan <i>dashboard</i> ini.	1-5
5	<i>Perceived Usefulness</i>	<i>Dashboard</i> ini memberikan saya gambaran <i>real-time</i> yang jelas dan berharga mengenai alur penumpang di SCP.	1-5
6		Saya merasa data yang ditampilkan di <i>dashboard</i> ini tidak terlalu membantu dalam pengambilan keputusan operasional.	1-5
7		Dengan <i>dashboard</i> ini, saya dapat lebih proaktif dalam mengantisipasi dan mengelola potensi kepadatan antrean.	1-5
8		<i>Dashboard</i> ini merupakan alat yang sangat berharga untuk menganalisis tren dan pola penumpang secara historis.	1-5
9	<i>Dependability & Attractiveness</i>	Saya yakin data yang ditampilkan di <i>dashboard</i> ini akurat dan dapat diandalkan.	1-5

10		Secara keseluruhan, saya merasa tampilan visual <i>dashboard</i> ini tidak menarik.	1-5
11		Saya merasa semua fungsi di dalam <i>dashboard</i> ini terintegrasi dengan sangat baik.	1-5
12		Saya tidak akan sering menggunakan <i>dashboard</i> ini jika sudah diimplementasikan secara penuh.	1-5

Data kuantitatif yang berasal dari Skala Likert (Bagian B) akan diolah menggunakan prosedur skoring yang diadaptasi dari metode *System Usability Scale* (SUS) untuk menghasilkan skor yang intuitif pada skala 0-100. Langkah-langkahnya adalah sebagai berikut:

- a. Untuk pernyataan positif (nomor 1, 3, 5, 7, 9, 11 pada kuesioner Avsec; nomor 1, 3, 5, 7, 8, 9, 11 pada kuesioner Manajemen), skor kontribusi dihitung dengan rumus: $\text{Skorkontribusi} = (\text{SkorLikert} - 1)$. Hasilnya akan berada di rentang 0 hingga 4.

$$\text{skor SUS ganjil} = \sum P_x - 1$$

- b. Untuk pernyataan negatif (nomor 2, 4, 6, 8, 10, 12 pada kuesioner Avsec; nomor 2, 4, 6, 10, 12 pada kuesioner Manajemen), skor kontribusi dihitung dengan rumus: $\text{Skorkontribusi} = (5 - \text{SkorLikert})$. Hasilnya juga akan berada di rentang 0 hingga 4.

$$\text{skor SUS genap} = \sum 5 - P_n$$

- c. Skor total per responden dihitung dengan menjumlahkan semua skor kontribusi dari 12 pertanyaan. Skor total ini akan berada di rentang 0 hingga 48.

$$(\sum \text{skor ganjil} - \sum \text{skor genap}) \times 2,5$$

- d. Skor UAQ final untuk setiap responden dihitung dengan mengalikan skor total dengan 2.5. Langkah ini mengubah rentang skor dari 0-48 menjadi 0-100, di mana 100 merepresentasikan penerimaan dan pengalaman pengguna yang paling positif.

$$\bar{x} = \frac{\sum x}{n}$$

Keterangan:

$\bar{x}$: Nilai rata-rata

$\sum x$: Jumlah skor System Usability Scale

n : Jumlah responden

Setelah skor UAQ final untuk setiap responden didapatkan, analisis selanjutnya akan menghitung skor rata-rata (*mean*) untuk setiap profil pengguna secara keseluruhan, serta skor rata-rata untuk setiap konstruk spesifik (misalnya, rata-rata skor *Perceived Usefulness*).

3.2 Alat dan Bahan

Dalam melaksanakan ini dibutuhkan alat dan bahan yang digunakan untuk membantu jalannya proses penelitian dari awal hingga mencapai tujuan penelitian. Berikut alat dan bahan yang digunakan:

3.2.1 Perangkat Lunak (Software)

- Flutter: *Framework* untuk pengembangan aplikasi *mobile*.
- Laravel: *Framework* PHP untuk pengembangan *Dashboard web* dan API.
- API (*Application Programming Interface*): Menghubungkan aplikasi *mobile* dengan sistem penyimpanan data (*database*).
- Database Management System* (DBMS): MySQL atau PostgreSQL sebagai basis data utama untuk menyimpan data penumpang.
- Editor Kode: Visual Studio Code untuk menulis dan mengelola kode aplikasi.

3.2.2 Perangkat Keras (Hardware)

- Smartphone*: Digunakan oleh petugas untuk pemindaian *Boarding pass*.

- b. *Server*: Menyimpan data yang terkumpul dari aplikasi *mobile* dan mendukung *Dashboard*.
- c. Kamera Seluler: Memanfaatkan kamera perangkat seluler untuk pemindaian kode *Boarding pass*.

3.2.3 Bahan Penelitian

Studi ini menggunakan bahan penelitian yang terdiri dari sumber sekunder berupa jurnal-jurnal ilmiah relevan sebagai landasan teoretis, dan sumber primer. Sumber primer ini adalah data yang diperoleh dari hasil observasi langsung di lapangan serta diskusi informal dengan para pemangku kepentingan untuk mengidentifikasi kebutuhan dan permasalahan secara kontekstual.