

BAB III

METODE PENELITIAN

3.1 Jenis Penelitian

Penelitian ini mengadopsi jenis penelitian dan pengembangan atau *Research and Development* (RnD). Menurut definisi yang dikemukakan oleh para ahli, RnD adalah metode penelitian yang secara sistematis digunakan untuk menghasilkan produk baru atau menyempurnakan produk yang telah ada, serta menguji validitas dan efektivitas produk tersebut. Menurut Borg & Gall (1983, sebagaimana dikutip oleh banyak sumber), tujuan utama dari RnD adalah menjembatani kesenjangan antara temuan penelitian teoretis dengan praktik di lapangan. Berbeda dengan penelitian murni yang berfokus pada pengujian teori, RnD berorientasi pada pengembangan solusi praktis dan fungsional terhadap masalah yang nyata.

Justifikasi pemilihan RnD sebagai jenis penelitian didasarkan pada keselarasan fundamental antara karakteristik metodologi ini dengan tujuan penelitian. Tujuan utama penelitian ini adalah menghasilkan sebuah prototipe sistem pembelajaran fungsional bernama SEKAPAI dan melakukan uji fungsionalitas terhadapnya. Pendekatan ini secara inheren memastikan bahwa teknologi dikembangkan untuk melayani tujuan pedagogis, bukan sebaliknya, sehingga selaras dengan rumusan masalah yang berfokus pada perancangan sistem yang mengintegrasikan kerangka kerja pedagogis *scaffolding* dengan teknologi AI.

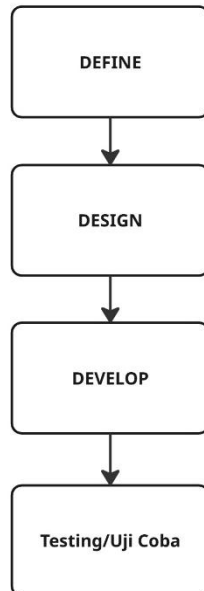
3.2 Prosedur Penelitian

Prosedur penelitian yang digunakan dalam pengembangan sistem SEKAPAI ini mengadaptasi Model Pengembangan 4-D (*Four-D Model*). Model ini pertama kali diperkenalkan oleh Thiagarajan & Semmel (1974) dan telah diakui secara luas sebagai kerangka kerja yang sistematis dalam penelitian dan pengembangan, khususnya di bidang pendidikan dan teknologi pembelajaran (Fayrus et al., 2022). Model 4-D dipilih karena memiliki alur yang logis, terstruktur, dan jelas, yang mencakup seluruh siklus pengembangan produk mulai

dari analisis kebutuhan hingga penyebaran. Model ini terdiri dari empat tahapan utama, yaitu: (1) Define (Pendefinisian), (2) Design (Perancangan), (3) Develop (Pengembangan), dan (4) Disseminate (Penyebaran). Meskipun model ini bersifat sekuensial, dalam praktiknya ia memungkinkan adanya adaptasi, seperti yang ditunjukkan oleh beberapa penelitian yang memodifikasi atau membatasi tahapannya sesuai dengan ruang lingkup dan sumber daya yang tersedia, misalnya hingga tahap *Develop* saja (Rahmatsyah & Dwiningsih, 2021).

Dalam konteks penelitian ini, dilakukan adaptasi terhadap model 4D standar. Sejalan dengan batasan masalah dan tujuan penelitian yang berfokus pada perancangan dan uji fungsionalitas prototipe, tahap *Disseminate* tidak diartikan sebagai penyebaran massal ke pengguna akhir. Sebaliknya, tahap ini diadaptasi menjadi tahap **Test (Uji Coba)**, di mana prototipe sistem yang telah dikembangkan akan diuji secara fungsional dalam lingkup terbatas untuk memvalidasi kinerjanya sesuai spesifikasi. Untuk memberikan gambaran yang jelas mengenai alur kerja penelitian, Gambar 3.1 di bawah ini mengilustrasikan model pengembangan 4D yang telah diadaptasi.

MODEL 4-D Thiagarajan



Gambar 3. 1 Alur Model Pengembangan 4D (diadaptasi dari Thiagarajan et al., 1974)

Berikut adalah penjelasan rinci mengenai setiap tahapan dalam model 4-D yang dilakukan dalam penelitian ini:

3.2.1 Tahap *Define*

Tahap *Define* merupakan fondasi dari seluruh proses pengembangan. Tahap ini bertujuan untuk menetapkan dan mendefinisikan syarat-syarat pengembangan dengan melakukan analisis mendalam terhadap permasalahan dan kebutuhan yang ada. Berdasarkan kerangka kerja yang diusulkan oleh Thiagarajan, tahap ini mencakup lima kegiatan utama. Fondasi untuk kelima kegiatan ini sebagian besar telah diletakkan dan diuraikan secara rinci pada Bab I dan Bab II.

1. *Front-end Analysis* (Analisis Awal): Kegiatan ini berfokus pada analisis masalah yang mendorong perlunya adaptasi dan penyempurnaan sebuah kerangka kerja teoretis yang ada. Sebagaimana telah diidentifikasi secara mendalam pada Bab I dan Bab II, penelitian oleh Liao et al. (2024) telah

menyajikan sebuah kerangka kerja yang menjanjikan, namun sekaligus mengungkap adanya masalah baru, yaitu risiko *over-reliance* (ketergantungan berlebih) mahasiswa pada sistem AI. Dengan demikian, masalah fundamental yang dianalisis bukanlah sekadar kesenjangan antara teori dan praktik secara umum, melainkan kebutuhan spesifik untuk mengadaptasi dan memitigasi kelemahan yang teridentifikasi dalam sebuah model teoretis yang sudah ada.

2. *Learner Analysis* (Analisis Siswa): Kegiatan ini berfokus pada identifikasi karakteristik, kebutuhan, dan tantangan yang dihadapi oleh target pengguna, yaitu siswa atau pembelajar, seperti pengetahuan awal, gaya belajar, dan motivasi mereka terhadap penggunaan teknologi. Kajian pada Bab II telah mengidentifikasi bahwa pembelajar sering kali memerlukan dukungan belajar yang bersifat personal, adaptif, dan tepat waktu, suatu kebutuhan yang sulit dipenuhi secara konsisten oleh pendidik manusia dalam skala besar karena tuntutan kognitif yang tinggi.
3. *Task Analysis* (Analisis Tugas): Kegiatan ini bertujuan mengidentifikasi dan merinci konsep-konsep utama yang akan dimuat dalam sistem, dalam hal ini adalah teori *scaffolding* dan materi pembelajaran yang akan didukung.
4. *Concept Analysis* (Analisis Konsep): Kegiatan ini melibatkan analisis terhadap konsep-konsep teoretis dan teknologis inti yang akan diimplementasikan dalam sistem. Konsep-konsep ini, yang meliputi teori *scaffolding* pedagogis, *Zone of Proximal Development* (ZPD) Vygotsky, dan kapabilitas teknis dari LLM seperti ChatGPT, telah dikaji secara ekstensif pada Tinjauan Pustaka di Bab II.
5. *Specifying Instructional Objectives* (Perumusan Tujuan Pengembangan): Kegiatan ini merupakan sintesis dari empat analisis sebelumnya, di mana hasil analisis diterjemahkan menjadi tujuan pengembangan produk yang konkret dan terukur. Tujuan ini selaras dengan Tujuan Penelitian yang telah dirumuskan pada Bab I, yaitu: (1) Menghasilkan sebuah rancangan sistem pembelajaran SEKAPAI yang terstruktur, dan (2) Melakukan uji fungsionalitas pada prototipe sistem SEKAPAI.

3.2.2 Tahap *Design*

Pada tahap *Design*, tujuan utamanya adalah untuk menyusun cetak biru (*blueprint*) dari produk yang akan dikembangkan. Tahap ini merupakan proses kreatif untuk menerjemahkan spesifikasi kebutuhan dari tahap *Define* menjadi sebuah rancangan produk awal (Okpatrioka, 2023). Kegiatan pada tahap ini meliputi:

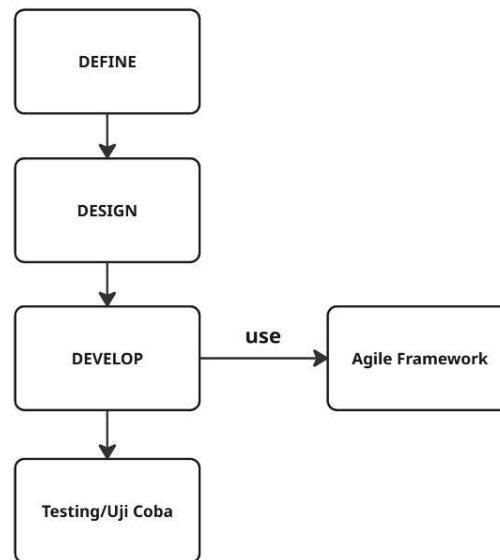
1. *Criterion-Referenced Test Construction* (Penyusunan Tes Acuan): Dalam konteks ini, dirancang skenario pengujian fungsionalitas untuk mengukur apakah fitur-fitur sistem bekerja sesuai tujuan.
2. *Media Selection* (Pemilihan Media): Menentukan format media yang paling sesuai untuk menyajikan fungsionalitas *scaffolding*, yaitu melalui antarmuka web yang interaktif dengan integrasi *chatbot*.
3. *Format Selection* (Pemilihan Format): Merancang format penyajian konten dan interaksi, seperti tata letak antarmuka, alur dialog, dan cara penyampaian umpan balik dari AI.
4. *Initial Design* (Rancangan Awal): Menghasilkan rancangan awal yang konkret berupa *wireframe* atau *mockup* antarmuka pengguna sistem SEKAPAI. Hasil dari tahap ini adalah Draf I dari produk.

3.2.3 Tahap *Develop*

Tahap *Develop* adalah tahap untuk merealisasikan rancangan yang telah dibuat menjadi sebuah produk yang siap diuji. Tahap ini memiliki dua tujuan utama: (1) menghasilkan prototipe produk, dan (2) melakukan uji formatif untuk mendapatkan masukan atau revisi (Fayrus et al., 2022). Proses ini melibatkan pembangunan sistem SEKAPAI berdasarkan *blueprint* dari tahap *Design*, yang mencakup coding (pemrograman) pada sisi *front-end* dan *back-end* serta integrasi dengan API ChatGPT. Untuk melaksanakan tahap pengembangan ini, peneliti mengadopsi pendekatan Metodologi Agile seperti disajikan pada Gambar 3.2. Agile adalah sebuah filosofi pengembangan perangkat lunak yang menekankan

pada fleksibilitas, adaptabilitas, dan kolaborasi untuk merespons perubahan secara cepat (Dudhat & Ali Abbasi, 2021).

MODEL 4-D Thiagarajan



Gambar 3. 2 Adopsi Metodologi Pengembangan *Agile*

Pendekatan pengembangan lain yang lebih tradisional dan sekuensial, seperti model Waterfall, tidak dipilih karena sifatnya yang kaku dan linear. Model Waterfall mengharuskan setiap tahapan (analisis, desain, implementasi, pengujian) diselesaikan secara penuh sebelum melangkah ke tahap berikutnya (Al-Saqqa et al., 2020). Kekakuan ini tidak cocok untuk proyek inovatif seperti SEKAPAI, di mana pemahaman terhadap solusi dan kebutuhan pengguna dapat berevolusi. Perubahan pada tahap akhir akan sangat sulit dan mahal untuk diimplementasikan. Sementara itu, Model *Prototyping*, yang juga bersifat iteratif, seringkali hanya berfokus pada penyempurnaan antarmuka pengguna (UI) dan pengumpulan umpan balik tanpa struktur yang jelas untuk menghasilkan produk yang solid secara inkremental. Model ini berisiko menghasilkan prototipe yang "dibuang" dan harus dibangun ulang dari awal, yang kurang efisien untuk skala proyek skripsi (Tetteh, 2024).

Penerapan Metodologi Agile dalam penelitian ini didasarkan pada prinsip-prinsip utamanya, yaitu pengembangan yang iteratif dan inkremental (*iterative and incremental*) (Williams, 2010). Pendekatan iteratif berarti proses pengembangan dilakukan dalam siklus-siklus pendek yang berulang, di mana setiap siklus melibatkan fase perencanaan, desain, implementasi, dan pengujian. Pendekatan inkremental berarti di akhir setiap iterasi, dihasilkan sebuah bagian kecil dari sistem yang fungsional dan berpotensi untuk dirilis (*a working increment*) (Al-Saqqa et al., 2020). Dengan cara ini, prototipe sistem SEKAPAI tidak dibangun sekaligus secara masif, melainkan tumbuh dan berkembang secara bertahap. Pendekatan ini dipilih karena sangat sesuai untuk proyek skripsi yang lingkupnya dapat berevolusi, memungkinkan peneliti untuk mendapatkan umpan balik lebih awal dan melakukan penyesuaian secara efisien tanpa harus merombak keseluruhan sistem (Atawneh, 2019).

Alur kerja iteratif yang dilakukan dalam pengembangan prototipe SEKAPAI adalah sebagai berikut:

1. Perencanaan Iterasi: Dari daftar fitur yang telah dirancang, dipilih satu atau beberapa fitur prioritas untuk dikerjakan dalam satu siklus pengembangan.
2. Pengembangan Fitur: Melakukan proses coding untuk mengimplementasikan fitur tersebut ke dalam prototipe sistem.
3. Pengujian Internal: Melakukan pengujian mandiri untuk memastikan fitur yang baru dibuat berfungsi sesuai harapan dan tidak menimbulkan masalah pada bagian sistem yang sudah ada sebelumnya.
4. Evaluasi dan Refleksi: Di akhir siklus, peneliti melakukan evaluasi terhadap hasil kerja. Temuan, bug, atau ide penyempurnaan yang muncul dari proses ini akan menjadi dasar untuk merencanakan iterasi berikutnya. Proses ini diulang secara terus-menerus hingga seluruh fitur inti yang ditetapkan dalam ruang lingkup penelitian berhasil dikembangkan, membentuk sebuah prototipe sistem SEKAPAI yang kohesif dan siap untuk diuji pada tahap selanjutnya.

Proses ini diulang secara terus-menerus hingga seluruh fitur inti yang ditetapkan dalam ruang lingkup penelitian berhasil dikembangkan, membentuk

sebuah prototipe sistem SEKAPAI yang kohesif dan siap untuk diuji pada tahap selanjutnya.

3.2.4 Tahap *Testing*/Uji Coba

Tahap keempat dan terakhir dari model 4-D adalah *Disseminate* (Penyebaran). Sesuai dengan batasan masalah dan tujuan penelitian yang berfokus pada perancangan serta validasi fungsionalitas produk, tahap ini tidak diwujudkan dalam bentuk penyebaran massal ke pengguna akhir. Sebaliknya, tahap ini diadaptasi menjadi fase uji coba produk yang sistematis, yang bertujuan untuk mengumpulkan data empiris guna menjawab rumusan masalah kedua mengenai hasil uji fungsionalitas prototipe sistem SEKAPAI.

Teknik pengujian yang digunakan untuk mengevaluasi sistem adalah Pengujian *Black-Box Testing*. *Black-box testing* adalah sebuah pendekatan pengujian perangkat lunak yang berfokus pada fungsionalitas sistem dari perspektif eksternal, tanpa perlu mengetahui struktur kode internal, arsitektur, atau detail implementasi di dalamnya (Sholehurrohman et al., 2025). Pengujian dilakukan dengan cara memberikan serangkaian *input* pada sistem dan mengamati serta memverifikasi apakah output yang dihasilkan sesuai dengan hasil yang diharapkan berdasarkan spesifikasi kebutuhan fungsional (Fahrezi et al., 2022). Pendekatan ini sering disebut juga sebagai pengujian perilaku (*behavioral testing*) karena ia menguji bagaimana sistem "berperilaku" dari sudut pandang pengguna (Khan, 2011).

Pelaksanaan *black-box testing* ini akan berfokus pada pengujian fungsional. Meskipun *black-box testing* mencakup berbagai teknik spesifik seperti *equivalence partitioning* dan *boundary value analysis* (Hardika et al., 2024), penelitian ini akan memprioritaskan validasi fungsional dasar dari setiap fitur yang telah dikembangkan.

3.3 Instrumen Penelitian

Instrumen utama yang digunakan untuk melaksanakan pengujian fungsionalitas adalah Skenario Interaksi. Skenario interaksi adalah sebuah spesifikasi yang terdiri dari serangkaian langkah interaksi, data *input*, dan kondisi

eksekusi yang dirancang untuk mengevaluasi aspek tertentu dari sebuah perangkat lunak, serta mendefinisikan hasil yang diharapkan (*expected result*) untuk memastikan sistem berfungsi sesuai dengan tujuannya (Putri et al., 2024), skenario interaksi juga dirancang untuk mengevaluasi alur pedagogis sistem secara holistik dari perspektif pengguna. Instrumen ini berupa serangkaian narasi atau deskripsi langkah-langkah yang akan disimulasikan oleh peneliti untuk menguji bagaimana sistem merespons berbagai *input* pengguna dalam sebuah konteks pembelajaran yang utuh.

Untuk mendokumentasikan pelaksanaan pengujian secara sistematis, setiap skenario interaksi akan dipecah menjadi *test case* yang lebih terstruktur dalam format tabel pengujian *black-box*. Tabel ini akan mencatat *input* spesifik yang diberikan pada setiap langkah dalam skenario dan membandingkan *output* aktual sistem dengan *output* yang diharapkan sesuai dengan alur pedagogis yang dirancang. Tabel skenario pengujian akan mencakup kolom-kolom berikut:

1. **ID Tes:** Kode unik untuk setiap skenario interaksi.
2. **Skenario Interaksi:** Penjelasan naratif dari alur yang diuji.
3. **Hasil yang Diharapkan (*Expected Output*):** Perilaku atau respons sistem yang diharapkan terjadi jika alur *scaffolding* berjalan dengan benar.
4. **Hasil Aktual (*Actual Output*):** Perilaku atau respons sistem yang sebenarnya terjadi saat pengujian dilakukan.
5. **Kesimpulan/Status:** Status hasil pengujian, diisi dengan "Berhasil" atau "Tidak Berhasil".

3.4 Teknik Analisis Data

Data yang diperoleh dari pelaksanaan uji fungsionalitas menggunakan instrumen skenario uji akan dianalisis secara deskriptif kualitatif. Pendekatan ini dipilih karena bertujuan untuk memberikan pemahaman yang mendalam dan interpretatif terhadap kinerja fungsional prototipe sistem SEKAPAI. Fokus analisis adalah pada deskripsi naratif tentang perilaku sistem selama pengujian untuk menjawab rumusan masalah kedua. Proses analisis data akan dilakukan melalui langkah-langkah berikut:

1. Klasifikasi dan Reduksi Data

Data mentah dari hasil pengujian akan diklasifikasikan terlebih dahulu. Data akan dikelompokkan berdasarkan modul utama yang diuji, yaitu SA, CA, dan FI. Proses reduksi data akan dilakukan dengan memfokuskan analisis pada temuan-temuan yang paling signifikan dari setiap modul, terutama pada skenario-skenario yang mengungkap perilaku sistem yang penting untuk dievaluasi.

2. Analisis Konten Kualitatif

Setelah data diklasifikasikan, analisis konten akan dilakukan pada catatan-catatan hasil pengujian. Analisis ini akan berfokus pada dua tema utama:

- 1) Verifikasi Fungsionalitas yang Berhasil: Mendeskripsikan secara naratif bagaimana sistem menunjukkan perilaku yang sesuai dengan hasil yang diharapkan (*expected outcome*) pada skenario-skenario yang berhasil. Analisis ini akan menyoroti fitur-fitur kunci yang telah terbukti berfungsi dengan baik dan stabil pada setiap modul.
- 2) Analisis Kegagalan (*Failure Analysis*): Untuk setiap skenario yang berstatus "Tidak Berhasil", akan dilakukan analisis kualitatif yang mendalam. Berdasarkan catatan observasi, akan diidentifikasi dan dideskripsikan pola-pola kegagalan yang muncul. Analisis akan berfokus pada sifat dari anomali yang terjadi (misalnya, *bug* pada antarmuka, respon yang tidak logis dari AI, atau kegagalan sistem dalam menangani input tertentu) untuk membangun pemahaman konseptual tentang area kelemahan dalam prototipe.

3. Penarikan Kesimpulan

Langkah terakhir adalah menarik kesimpulan secara induktif dari seluruh temuan kualitatif. Dengan mensintesis deskripsi fungsionalitas yang berhasil dan analisis pola kegagalan, akan dirumuskan sebuah kesimpulan menyeluruh mengenai kondisi fungsionalitas sistem SEKAPAI. Kesimpulan ini akan bersifat interpretatif, menjelaskan tingkat "kematangan" fungsional dari setiap modul dan sistem secara keseluruhan, serta memberikan justifikasi untuk rekomendasi perbaikan pada iterasi pengembangan selanjutnya.