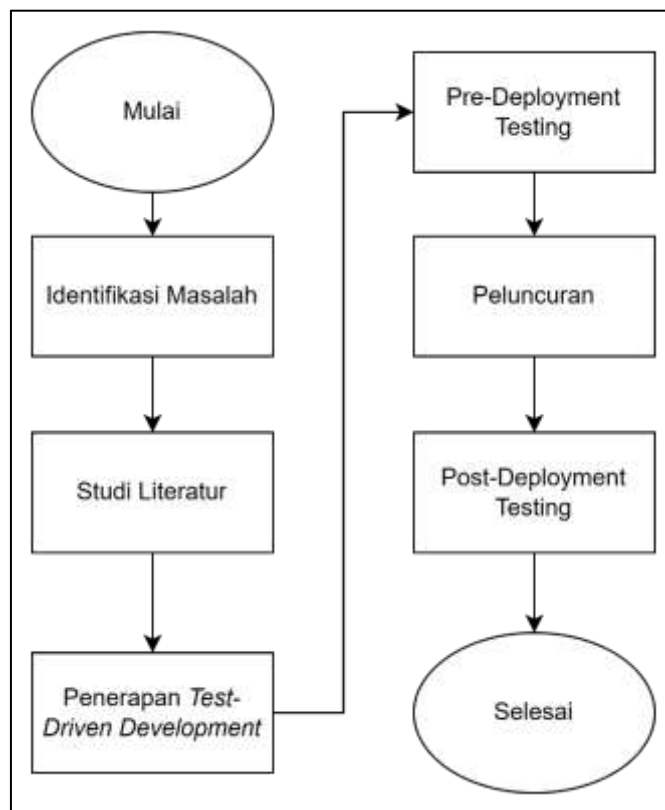


BAB III

METODE PENELITIAN

3.1 Alur Penelitian

Penelitian tugas akhir ini akan dilakukan membangun aplikasi penilaian esai singkat otomatis berbasis web dengan metode pengembangan *Test-Driven Development* (TDD) dan diuji dengan metode dua metode pengujian, yakni *System Usability Scale* (SUS) dan *Black Box*. Alur penelitian dapat dilihat pada *flowchart* pada Gambar 3.1 dibawah ini.



Gambar 3.1 *Flowchart* Penelitian

3.2 Penjabaran Langkah Penelitian

Dapat diperhatikan melalui Gambar 3.1 sebelumnya bahwa pada penelitian ini memiliki 4 fase sebagaimana pada gambar tersebut. Penjelasan setiap fase penelitian ini adalah sebagai berikut:

3.2.1 Identifikasi Masalah

Pada tahap ini, peneliti akan menyediakan alasan dilakukannya penelitian dengan mengumpulkan permasalahan nyata yang memerlukan tindakan dan dapat diselesaikan dengan disiplin teknik informatika.

3.2.2 Studi Literatur

Dalam sebuah penelitian, diperlukan pengetahuan landasan secara teoritis terhadap permasalahan yang dihadapi dan ingin ditindak. Pengetahuan teoritis ini dapat diperoleh melalui jurnal penelitian terdahulu maupun buku yang berisi materi yang dibutuhkan. Jurnal penelitian terdahulu dan buku yang digunakan akan memiliki kesimpulan terkait satu atau lebih teori yang akan diperlukan pada penelitian ini. Kemudian teori yang sudah dikumpulkan dan dijadikan landasan akan menjadi arahan kemana kemungkinan penelitian yang dilakukan akan berakhir.

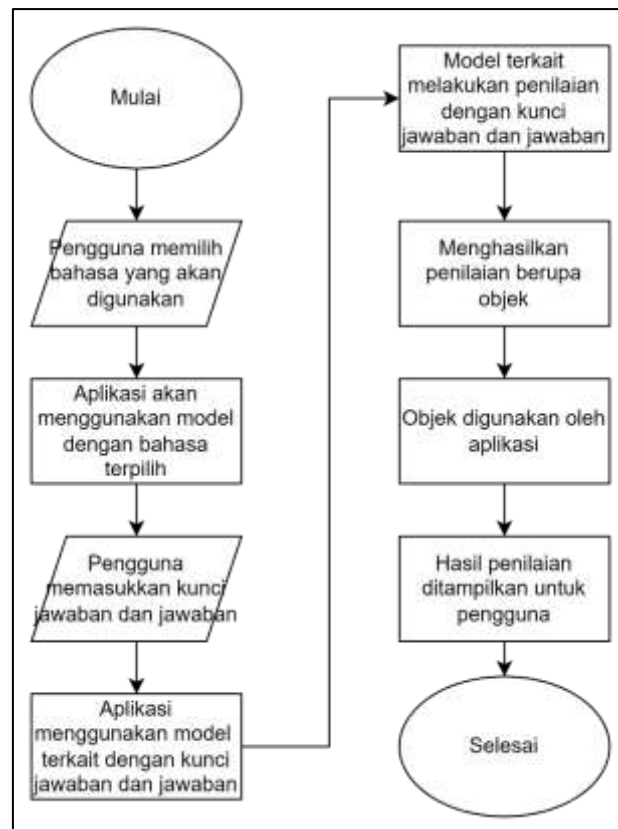
3.2.3 Penerapan *Test-Driven Development*

Penelitian ini akan dilakukan menggunakan metode siklus hidup perangkat lunak (SDLC) *Test-Driven Development* (TDD) dengan harapan menjadi solusi atas permasalahan yang ingin diselesaikan. Dengan ini, penelitian akan menekankan pengembangan melalui beberapa kebutuhan yang dijadikan pengujian atas dibuatnya fitur maupun fungsi aplikasi yang dikembangkan. Dengan metode ini, aplikasi yang dikembangkan dapat dipastikan kelayakannya dalam memenuhi semua fitur dan fungsi berdasarkan pengujian sebelumnya.

Telah diketahui sebelumnya bahwa TDD memiliki 3 fase utama yaitu membuat pengujian, membuat kode yang berhasil memenuhi pengujian tersebut, lalu *refactor* kode yang telah berhasil tersebut. Tahap lainnya seperti pemahaman terhadap fitur atau *bug* yang ingin diuji sebelum masuk ke tiga fase utama, dan tahap iterasi atau tindakan terhadap fitur atau *bug* lain yang ingin diuji dan diselesaikan apabila pengujian sebelumnya sudah berhasil tanpa error. Apabila seluruh fitur dan *bug* yang ingin ditindak sudah selesai dibangun, maka aplikasi siap memasuki tahap pengujian

3.2.4 Integrasi Aplikasi dan Model

Pada penelitian ini, aplikasi yang akan dikembangkan akan menggunakan dua model pembelajaran mesin yang telah dilatih sebelumnya. Dua model tersebut yakni dalam bahasa Indonesia dan bahasa Inggris. Alur integrasi aplikasi dengan model yang digunakan dapat dilihat pada Gambar 3.2 berikut ini.



Gambar 3.2 Alur Integrasi Aplikasi dan Model

Integrasi model pembelajaran mesin pada aplikasi digunakan dengan pengguna memilih bahasa yang akan digunakan pada penilaian yang dilakukannya, dengan ini aplikasi akan memilih model dengan bahasa yang tepat.

Selanjutnya aplikasi akan menjalankan model terkait dengan masukkan dari pengguna untuk diproses. Setelah model selesai melakukan penilaian otomatis, model akan menghasilkan objek dan mengembalikannya pada aplikasi. Kemudian aplikasi akan menggunakan objek untuk menampilkan hasil penilaian kepada pengguna untuk dilihat atau diunduh berupa PDF.

3.2.5 Pengujian

Aplikasi yang dikembangkan pada penelitian tugas akhir ini akan diuji menggunakan *System Usability Scale* (SUS) dan *Black Box* untuk memastikan apakah aplikasi penilaian esai otomatis berbasis web yang dikembangkan dapat meningkatkan proses penilaian esai singkat menjadi lebih efisien. Pengujian dilakukan dengan melakukan perbandingan subjektif antara nilai yang diusulkan dari aplikasi dengan nilai yang diperiksa oleh pengajar suatu subjek terhadap jawaban esai dari pertanyaan esai yang diberikan pengajar tersebut. Apabila hasil nilai akhir dari aplikasi dan pribadi pengajar kurang lebih sama, dapat dipastikan bahwa aplikasi dapat bekerja dengan baik.

3.3 Alat dan Bahan Tugas Akhir

Alat dan bahan yang dibutuhkan untuk mendukung jalannya penelitian tugas akhir ini. Alat dan bahan yang dimaksud dapat dilihat sebagai berikut ini:

3.3.1 Alat

Alat yang mendukung dan dibutuhkan dalam pelaksanaan penelitian tugas akhir ini adalah berikut ini:

1. *Notebook* dengan spesifikasi minimum, sistem operasi Windows 8, *processor* Intel Core i3 2330M CPU @ 2,2 GHz, memori 2GB DDR3, grafis NVIDIA GeForce GT 610 (4GB), *hardisk* 500GB. Pada tugas akhir ini digunakan komputer dengan spesifikasi minimum sistem operasi Windows 10, processor AMD Ryzen 5 2500U @ 2 GHz, memori 16GB DDR4, grafis AMD Radeon Vega 8, dan *Solid-State Drive* 512GB.
2. *Smartphone* dengan spesifikasi tipe minimum, OS Android OS v4.1.2 (Jelly Bean), CPU Dual-core 800 MHz, GPU Mali-400, Internal 4 GB, 768 MB RAM. Pada tugas akhir ini digunakan OS Android 9.0 (Pie), CPU Octa-core 2.3GHz, GPU PowerVR GE8320, Internal 128 GB, 4GB RAM.
3. *Node.js* dengan versi minimal 16.16.0. Pada tugas akhir ini menggunakan *Node.js* dengan versi 18.12.1.
4. *Python* dengan versi minimal 3.11.5.

5. *Visual Studio Code* (VSCode) untuk menulis kode program menggunakan versi 1.74.2. Versi dari VSCode dapat berubah seiring adanya pembaruan dari pembangun aplikasi VSCode tersebut sewaktu-waktu.

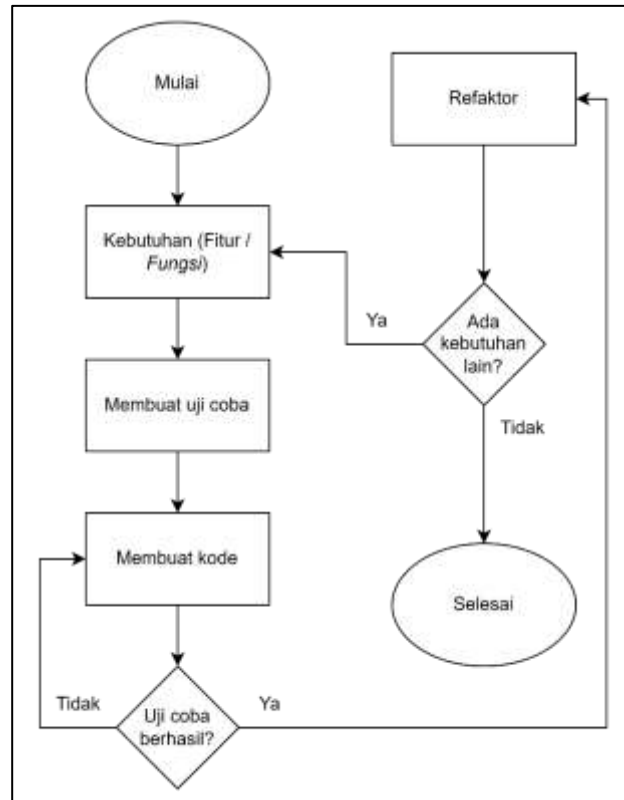
3.3.2 Bahan

Bahan yang digunakan / diperlukan untuk melakukan penelitian tugas akhir ini adalah sebagai berikut:

1. Model Terlatih untuk memproses penilaian esai singkat berbahasa Indonesia.
2. *Dataset* dari model terlatih penilaian esai singkat untuk bahasa Indonesia
3. Model Terlatih untuk memproses penilaian esai singkat berbahasa Inggris.
4. *Dataset* dari model terlatih penilaian esai singkat untuk bahasa Inggris

3.4 Metode Pengembangan

Dalam pengembangan aplikasi penilaian esai singkat otomatis berbasis web pada penelitian ini, akan digunakan salah satu metode pengembangan siklus hidup perangkat lunak *Agile*, khususnya *Test-Driven Development* (TDD). Alur pengembangan perangkat lunak menggunakan TDD pada penelitian ini dapat diperhatikan pada Gambar 3.3 dibawah ini.



Gambar 3.3 Alur Penelitian Test-Driven Development

Berdasarkan diagram alir yang dapat dilihat pada Gambar 3.3, TDD memiliki tahapan berupa kebutuhan, pembuatan uji coba, pembuatan kode, refaktor, dan peluncuran. Kemudian dibawah ini merupakan penjelasan dari masing-masing tahapan pada metode pengembangan TDD:

3.4.1 Kebutuhan Fitur / Fungsi

Tahap ini merupakan tahap pertama dari metode pengembangan TDD guna mendapatkan kebutuhan atau fungsi pada aplikasi yang akan dibangun. Pada tahap ini, pengembang diharapkan dapat mengerti kebutuhan yang diminta untuk dapat dilakukan oleh perangkat lunak yang sedang dikembangkan.

3.4.1.1 Wawancara

Dalam penelusuran kebutuhan fungsi untuk aplikasi yang akan dibangun pada tugas akhir ini, dibutuhkan informasi dari calon pengguna yaitu pengajar yang sudah biasa melakukan penilaian esai pada pelajar. Penulis melakukan wawancara kepada

sepuluh guru/dosen. Pada Tabel 3.1 dapat dilihat data narasumber, dan pada Tabel 3.2 merupakan pertanyaan dan jawaban dari wawancara yang dilakukan.

Tabel 3.1 Data Narasumber

No	Nama Responden	Peran	Lokasi
1	Zamroni Riza S.E	Guru	SMA Sapta Kharisma Jakarta
2	Dr. Dian Mustika Dewi Priyandari, S. Sos., M.M.	Dosen	Universitas Suryadarma
3	Sukma Erawan, S.Kom	Guru	SMA Negeri 68 Jakarta
4	Alan Syahwandi, S.Pd.I	Guru	SMP Negeri 3 Bogor
5	Ajeng Yeni Ratnasari, S.Pd	Guru	Pondok Pesantren Alfassalam
6	Dita Putri Bestari S.Pd	Guru	SMA Negeri 65 Jakarta
7	Ayu Nulantika S.Pd	Guru	SMA Negeri 65 Jakarta
8	Virgawaty Dewy, S.Pd	Guru	SMA Negeri 65 Jakarta
9	Ahmad Arfandi, S.Kom	Guru	SMA Negeri 65 Jakarta
10	Muhammad Lutfi, S.Pd	Guru	SMA Negeri 113 Jakarta

Tabel 3.2 Hasil Wawancara

No	Pertanyaan	Jawaban
1	Apabila ada aplikasi penilai esai otomatis, apa fungsi/fitur yang ibu/bapak harapkan dari aplikasi tersebut?	Aplikasi harus bisa menentukan kata kunci dari jawaban yang mana bisa digunakan untuk penilaian. Aplikasi juga harus bisa menilai sekaligus banyak karena apabila tidak, akan tetap memakan energi seperti menilai esai biasa
2	Apakah ada permasalahan terkait jumlah pelajar dan pengajar?	Sebagai pembuat soal, biasanya perlu waktu lama untuk mengoreksi 300 jawaban hanya untuk satu soal, maka perlu alat bantu untuk mempercepat ini.
3	Apa ada fungsi/fitur lain yang kira-kira diperlukan?	Aplikasi harus bisa menilai jawaban dari singkatan, karena tidak seluruh jawaban akan menggunakan kata formal untuk setiap kalimatnya, khususnya untuk pelajar yang masih SD atau SMP.
4	Terkait efisiensi, apa yang sekiranya dibutuhkan untuk sebuah aplikasi penilaian esai otomatis?	Aplikasi harus bisa menilai banyak jawaban sekaligus. Kalau bisa terintegrasi kepada media yang sering digunakan untuk jawaban esai seperti Google Form
5	Apakah ada fitur terkait efektifitas aplikasi yang dibutuhkan?	Aplikasi harus bisa menampilkan perbandingan jawaban pengajar dan pelajar serta bobot nilai dari

No	Pertanyaan	Jawaban
		aplikasinya. Karena nilai untuk sebuah soal esai itu tidak satu dan nol, bisa saja nilainya setengah (50% benar).
6	Aplikasi sebaiknya dapat diakses hanya pada komputer atau bebas?	Untuk utamanya pada komputer, karena pengajar biasanya menilai dari komputer sekolah, namun sebaiknya dibuat juga untuk versi ponsel untuk mengantisipasi adanya kebutuhan khusus saat menggunakan aplikasi tersebut, seperti saat ada pelajar yang ketinggalan jawabannya untuk dinilai bersama yang lain.

3.4.1.2 User Story

Tahap ini dilakukan untuk mendapatkan fungsi dan fitur utama pada aplikasi yang dikembangkan pada kebutuhan fungsional dan non-fungsional. *User story* didapatkan dengan melakukan wawancara kepada calon pengguna sistem dengan mengetahui apa yang diinginkan dan mengapa mereka menginginkan hal tersebut. *User story* dari hasil wawancara yang dilakukan dapat dilihat pada Tabel 3.3 dibawah ini:

Tabel 3.3 Kumpulan *User Story*

No	Nama	Objective	Alasan
1	Zamroni Riza S.E	Dapat melakukan penilaian secara masal	Aplikasi harus bisa dilakukan sekaligus agar proses penilaian bisa cepat
2	Alan Syahwandi, S.Pd.I		
3	Ajeng Yeni Ratnasari, S.Pd		
4	Dita Putri Bestari S.Pd	Dapat menilai esai dengan pembobotan nilai dan melihat bobot nilai setiap jawaban	Nilai kebenaran esai tidak selalu 1 atau 0, karena ada esai yang mungkin hanya benar 50 persen.
5	Ayu Nulantika S.Pd		
6	Muhammad Lutfi, S.Pd		
7	Dr. Dian Mustika Dewi Priyandari, S. Sos., M.M.	Dapat dipakai langsung dari aplikasi ujian	Aplikasi harus dapat dipakai langsung dari aplikasi yang sering digunakan ujian esai
8	Ahmad Arfandi, S.Kom	Dapat melakukan penilaian untuk satu jawaban	Aplikasi bisa dipakai untuk menilai satu esai apabila ada kebutuhan khusus
9	Virgawaty Dewy, S.Pd		
10	Sukma Erawan, S.Kom		

Berdasarkan wawancara yang dilakukan, calon pengguna membutuhkan aplikasi yang bisa melakukan penilaian esai secara masal untuk sehingga pengajar tidak perlu melakukan aktifitas yang berulang (menyalin dan menempel jawaban) untuk setiap jawaban pelajar dan penilaian esai secara tunggal dengan memasukkan satu pasangan jawaban pengajar dan pelajar untuk kasus khusus (seperti esai yang telat dikumpul oleh pelajar). Aplikasi juga diharapkan dapat menampilkan bobot nilai jawaban pelajar karena menurut narasumber nilai pada esai tidak benar atau salah seluruhnya, melainkan ada bobot pada seberapa benar jawaban esai tersebut. Kemudian untuk penilaian esai secara masal harus menggunakan tipe dokumen yang sama dari aplikasi ujian pada umumnya. Kebutuhan ini diuraikan menjadi kebutuhan teknis yang dijelaskan pada sub-bab 3.4.1.2 Kebutuhan Fungsional.

3.4.1.3 Kebutuhan Fungsional

Pada pengimplementasian aplikasi penilaian esai otomatis berbasis *website* ini, perlu beberapa kebutuhan fungsional yang harus dipenuhi untuk aplikasi dapat bekerja semestinya. Kebutuhan fungsional aplikasi berbasis *website* dapat dilihat pada Tabel 3.4 berikut ini.

Tabel 3.4 Kebutuhan Fungsional

ID	Deskripsi
F0	Aplikasi <i>website</i> dapat menentukan algoritma penilaian yang digunakan berdasarkan permintaan pengguna melalui tombol
F1A	Aplikasi <i>website</i> dapat menerima masukan dua buah jawaban berupa teks melalui <i>input form</i>
F1B	Aplikasi <i>website</i> dapat melakukan penilaian otomatis dari kedua masukan pada fungsi dengan ID F1A
F2A	Aplikasi <i>website</i> dapat menerima masukan berkas dengan ekstensi CSV, sebuah jawaban berupa teks, nama kolom nama pelajar berupa teks, dan nama kolom jawaban pelajar berupa teks.
F2B	Aplikasi <i>website</i> dapat melakukan penilaian berdasarkan masukan pada fungsi dengan ID F2A

3.4.1.4 Kebutuhan Non-Fungsional

Untuk meningkatkan kenyamanan pengguna saat menggunakan aplikasi penilaian esai otomatis berbasis *website* ini, diperlukan beberapa kebutuhan non-fungsional

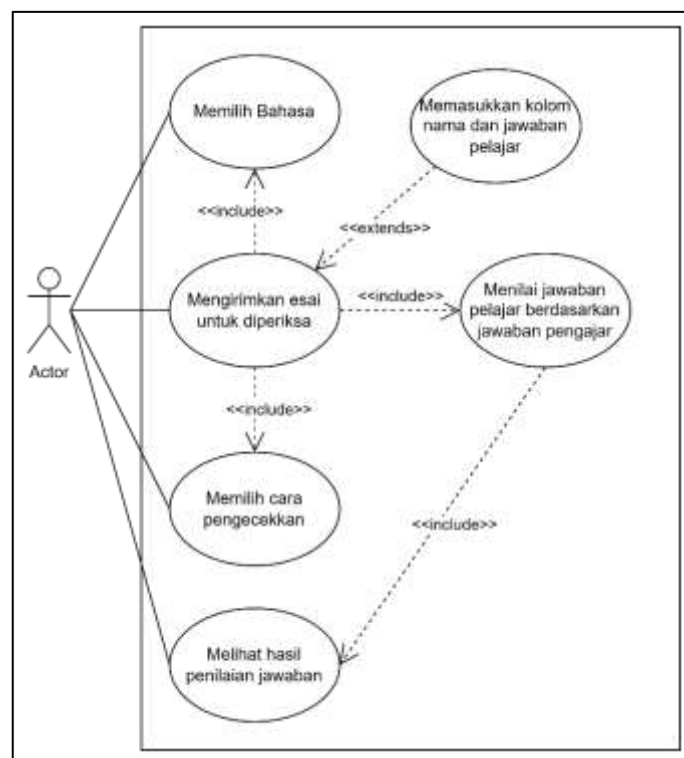
yang ingin dikembangkan. Kebutuhan ini dapat meningkatkan retensi pengguna agar lebih nyaman saat menggunakan aplikasi ini. Pada Tabel 3.5 dibawah ini dapat dilihat kebutuhan non-fungsional yang akan dipenuhi.

Tabel 3.5 Kebutuhan Non-Fungsional

ID	Deskripsi
NF1	Aplikasi <i>website</i> dapat digunakan pada berbagai perangkat, komputer dan mobile
NF2	Aplikasi <i>website</i> dapat dijalankan pada peramban web yang mendukung ES6
NF3	Aplikasi <i>website</i> memiliki tampilan yang menyesuaikan dengan ukuran layar perangkat yang digunakan (responsif)

3.4.1.5 Use Case Diagram

Use Case Diagram adalah diagram yang digunakan untuk menggambarkan sebuah interaksi yang dapat terjadi antara pengguna dan sistem. Pada Gambar 3.4 dibawah ini dapat diperhatikan interaksi pengguna pada sistem yang terjadi pada sistem yang hendak dibangun.

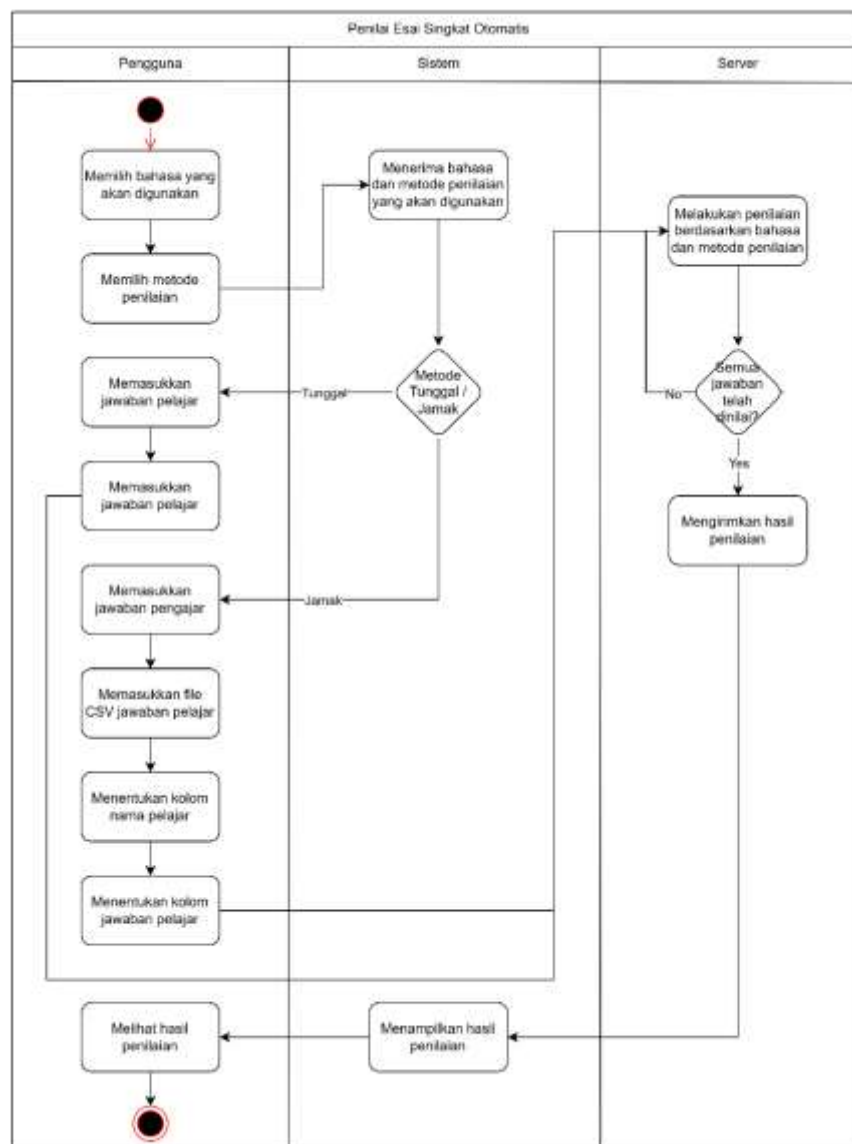


Gambar 3.4 *Use Case Diagram* Aplikasi

Pada Gambar 3.4 tersebut, aktor merupakan pengajar yang ingin melakukan pengecekan jawaban esai milik pelajar. Pengguna dapat melihat hasil jawaban yang diperiksa dengan mengirimkan esai yang ingin diperiksa, kemudian memilih bahasa dan teknik pengecekan yang diinginkan.

3.4.1.6 Activity Diagram

Activity diagram digunakan untuk menggambarkan alur pekerjaan dari aktivitas pada suatu sistem. Pada Gambar 3.5 berikut ini dapat dilihat *activity diagram* dari aplikasi yang akan dibangun pada penelitian ini.

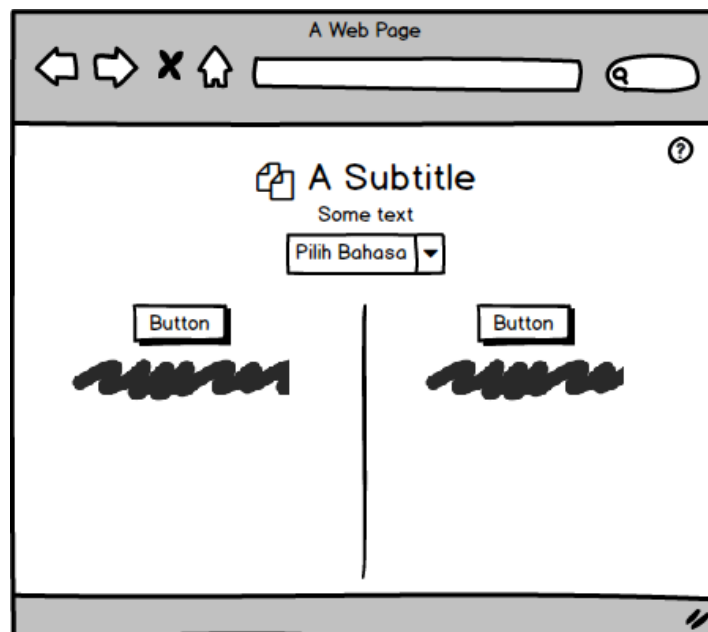


Gambar 3.5 Activity Diagram Aplikasi

Pada Gambar 3.5 tersebut dapat diperhatikan bahwa aplikasi dimulai dengan pengguna memilih bahasa dan metode penilaian yang akan digunakan. Kemudian berdasarkan pilihan tersebut, sistem akan menentukan masukkan yang dapat dilakukan oleh pengguna. Apabila semua masukkan telah diisi, kemudian sistem akan meminta *server* melakukan penilaian yang diminta oleh pengguna berdasarkan bahasa dan metode penilaian. Setelah *server* selesai melakukan penilaian, *server* akan mengirimkan hasil penilaian untuk ditampilkan pada aplikasi agar dapat dilihat oleh pengguna.

3.4.1.7 Mockup

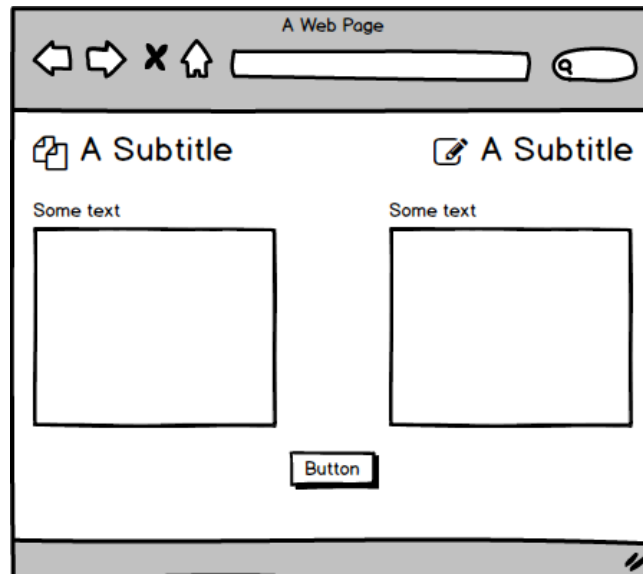
Untuk pengguna dapat menggunakan aplikasi web yang akan dibangun, perlu dibuat sebuah desain antarmuka agar dapat mengoperasikan aplikasinya. Perancangan antarmuka ini dilakukan pada tahap kebutuhan sebagai perencanaan terhadap gambaran aplikasi yang akan dikembangkan. Rancangan antarmuka aplikasi dapat dilihat sebagai berikut.



Gambar 3.6 *Mockup* Halaman Home

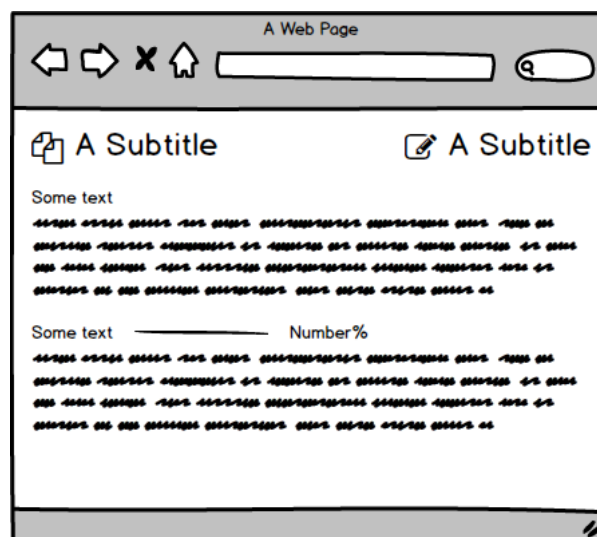
Saat pengguna mengakses halaman *website*, pengguna akan ditunjukkan halaman beranda seperti pada Gambar 3.6 dimana pengguna dapat memilih bahasa yang akan digunakan, dan metode penilaian yang akan digunakan dengan sebelah kiri

merupakan metode tunggal dan sebelah kanan merupakan metode jamak. Pengguna juga dapat mengakses halaman bantuan dengan menekan logo tanda tanya yang ada pada pojok kanan atas halaman beranda web.



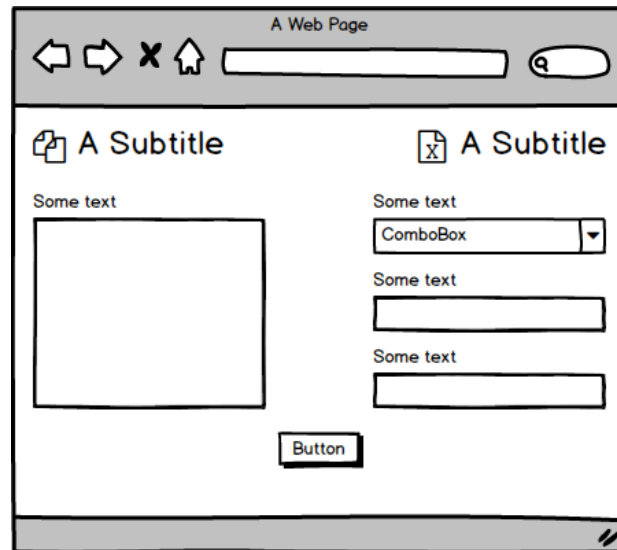
Gambar 3.7 *Mockup* Metode Penilaian Tunggal

Pada Gambar 3.7 merupakan tampilan yang akan dilihat pengguna apabila memilih metode penilaian tunggal. Pada halaman ini, pengguna sebagai pengajar dapat mengisi jawaban pengajar di sebelah kiri dan jawaban pelajar untuk dinilai di sebelah kanan.



Gambar 3.8 *Mockup* Hasil Metode Penilaian Tunggal

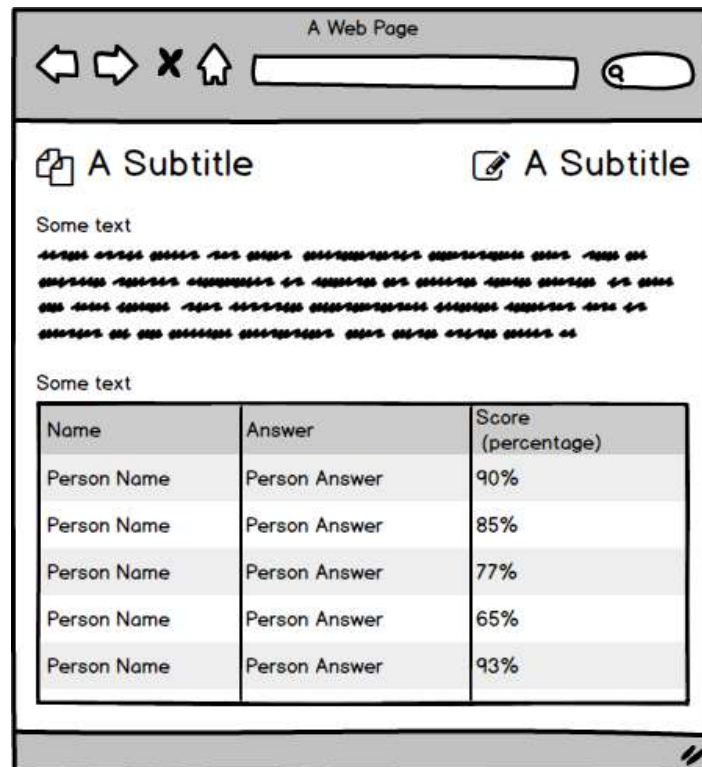
Setelah memproses penilaian otomatis, aplikasi akan mengarahkan pengguna ke halaman hasil penilaian dengan metode penilaian 1 sebagaimana ditunjukkan pada Gambar 3.8 dimana pengguna dapat melihat jawaban pengajar dan jawaban pelajar dengan persentase hasil jawabannya.



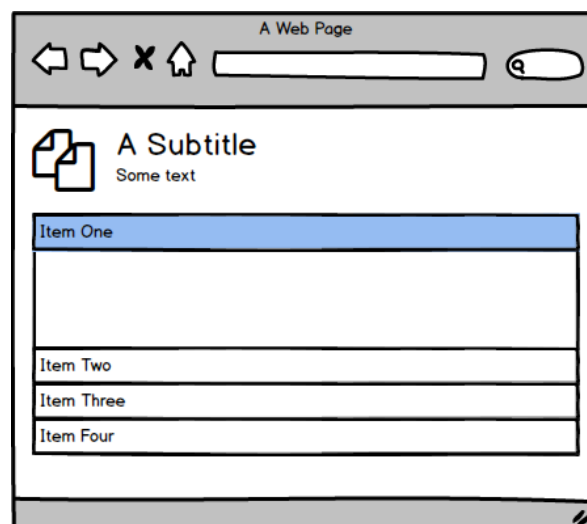
Gambar 3.9 *Mockup* Metode Penilaian Jamak

Pada Gambar 3.9 diatas merupakan halaman yang akan ditampilkan apabila pengguna memilih metode penilaian secara jamak. Pada halaman ini, pengguna dapat mengisi jawaban pengajar dan memasukkan berkas CSV pada aplikasi dan menentukan judul kolom berisi nama dan jawaban pelajar untuk diperiksa terhadap jawaban pengajar.

Gambar 3.10 merupakan halaman yang akan ditampilkan kepada pengguna untuk melihat hasil penilaian otomatis oleh aplikasi. Aplikasi akan menampilkan halaman yang akan menunjukkan jawaban pengajar pada bagian atas dan tabel berisi nama, jawaban, dan persentase nilai pelajar.



Gambar 3.10 *Mockup* Hasil Metode Penilaian Jamak



Gambar 3.11 *Mockup* Halaman Bantuan

Apabila pada halaman beranda seperti pada Gambar 3.6 pengguna memutuskan untuk menekan logo tanda tanya, maka pengguna akan diarahkan ke halaman bantuan seperti pada Gambar 3.11. Pada halaman ini, pengguna dapat melihat cara menggunakan aplikasi penilaian esai otomatis.

3.4.2 Pembuatan Uji Coba (Scenario Testing)

Pada tahap ini, pengembang membuat uji coba lengkap dengan ekspektasi berupa hasil dari pengujian tersebut. Pengembang membuat uji coba berdasarkan satu kebutuhan yang telah didapat dari tahap sebelumnya.

Aplikasi yang dikembangkan akan memiliki 3 skenario uji coba yang harus berhasil setiapnya dengan ekspektasi pada Tabel 3.6 berikut ini:

Tabel 3.6 Skenario Uji Coba

Skenario	Masukkan Jawaban Pengajar	Masukkan Jawaban Pelajar	Ekspektasi Keluaran
SC1	Teks untuk uji coba jawaban tunggal “The testing stage can influence both the coding stage (phase 5) and the solution refinement stage (phase 7)”	Teks untuk uji coba tunggal “The implementation phase and the maintenance phase are effected”	1.128299 dari 5.0
SC2	Teks untuk uji coba masal “To sumulate the behaviour of portions of the desired software product.”	Dokumen CSV berisi jawaban pelajar (“dataset.csv”)	Persentase Hasil Penilaian
SC3	Teks untuk uji coba tunggal “Fungsi karbohidrat adalah sebagai pemasok energi, dapat memperlancar proses pada pencernaan, memberikan efek kenyang dengan kandungan selulosa-nya dan penyeimbang asam dan basa dalam tubuh “	Teks untuk uji coba tunggal “Karbohidrat mempunyai peran penting untuk proses metabolisme atau proses yang berfungsi sebagai penyeimbang asam dan basa dalam tubuh. Karbohidrat dapat mencegah terjadinya ketidaksempurnaan proses oksidasi lemak. Fungsi karbohidrat yang utama adalah sebagai pemasok energi. Makanan berkarbohidrat dan serat tinggi dapat membantu memperlancar proses pada pencernaan. ”	55 dari 100

3.4.3 Pembuatan Kode (Coding)

Pada tahap ini, pengembang akan membuat fungsi pada perangkat lunak yang dikembangkan yang dapat memenuhi uji coba pada tahapan sebelumnya. Apabila ekspektasi hasil uji coba tidak dapat terpenuhi, maka kode dianggap gagal dan harus dibuat ulang hingga berhasil meraih ekspektasi yang diharapkan dari uji coba tersebut.

3.4.4 Refaktor (Refactor)

Saat kode untuk suatu fungsi berdasarkan suatu uji coba sudah berhasil dibuat, maka akan dilakukan pembersihan kode. Pembersihan disini berarti kode akan tulis sedemikian rupa agar tidak bersifat ambigu bagi pengembang lain yang mungkin akan melanjutkan pengembangan perangkat lunak yang sedang dikembangkan saat ini di masa depan. Setelah melakukan refaktorisasi kode, pengembang dapat kembali ke tahap kebutuhan dan mengulang seluruh tahapan ini dengan kebutuhan yang berbeda hingga semua kebutuhan berhasil dibuat.

3.5 Pre-Deployment Testing (Integration Testing)

Pengujian ini dibuat untuk memastikan bahwa komponen-komponen berbeda pada program berjalan semestinya berdasarkan skenario yang telah dibuat sebelumnya. Akan dibuat sebuah fungsi pada *server* untuk dapat memproses skenario pengujian dengan rancangan pada Tabel 3.7 berikut:

Tabel 3.7 *Integration Testing*

ID	Metode	Input 1	Input 2	Output
IT1	ScenarioEn(req, res)	Teks jawaban pengajar berbahasa Inggris.	Teks jawaban pelajar berbahasa Inggris.	JSON hasil penilaian tunggal berbahasa Inggris
IT2	ScenarioEnTab(req, res)	To sumulate the behaviour of portions of the desired software product.	dataset.csv	JSON hasil penilaian jamak berbahasa Inggris.
IT3	ScenarioId(req, res)	Teks jawaban pengajar berbahasa	Teks jawaban pelajar	Hasil penilaian tunggal

ID	Metode	Input 1	Input 2	Output
		Indonesia.	berbahasa Indonesia.	berbahasa Indonesia

Pada Tabel 3.7 tersebut, akan dibuat 3 fungsi yang masing-masing dapat menerima teks jawaban pengajar dalam bahasa Indonesia atau Inggris, kemudian menerima jawaban pelajar untuk diperiksa melalui teks atau berkas CSV. Fungsi tersebut selanjutnya akan memproses kedua masukan, dan menampilkan persentase hasil penilaian jawaban pada *console*.

3.6 Peluncuran

Apabila semua kebutuhan terhadap perangkat lunak yang dibangun sudah dibuat dan telah teruji bekerja dengan baik berdasarkan pengujian sebelumnya, maka akan dilakukan tahap peluncuran, yaitu kondisi dimana perangkat lunak yang dikembangkan sudah siap digunakan secara umum. Tahap ini dilakukan untuk mendapatkan timbal balik dari pengguna terkait perangkat lunak yang dikembangkan.

3.7 Post-Deployment Testing

Setelah aplikasi diluncurkan, maka akan didapatkan timbal balik dari pengguna untuk pengujian SUS dan *Blackbox*. Dengan ini dapat dilakukan perbaikan apabila terdapat *bug* yang tidak terdeteksi selama pembangunan aplikasi dan dapat diluncurkan kembali. Apabila tahap ini telah membuahkan hasil yang memuaskan, maka penelitian ini dapat dianggap selesai.

3.7.1 System Usability Scale

Seperti yang telah dilansir pada Bab 2, aplikasi yang dihasilkan penelitian ini akan diuji menggunakan SUS pada sekelompok orang untuk mendapatkan hasil kegunaan aplikasi dari penelitian yang dilakukan. Dengan itu, pada Tabel 3.8 berikut adalah daftar kuesioner yang akan diajukan kepada pengguna setelah menggunakan aplikasi yang bersangkutan.

Tabel 3.8 Kuesioner *System Usability Scale*

No	Pertanyaan
1	Saya berpikir akan menggunakan aplikasi penilaian esai otomatis berbasis web ini lagi
2	Saya merasa aplikasi penilaian esai otomatis berbasis web ini sulit digunakan
3	Saya merasa aplikasi penilaian esai otomatis berbasis web ini mudah digunakan
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan aplikasi penilaian esai otomatis berbasis web ini
5	Saya merasa fitur-fitur aplikasi penilaian esai otomatis berbasis web ini berjalan dengan semestinya
6	Saya merasa banyak hal yang tidak konsisten / tidak serasi pada aplikasi penilaian esai otomatis berbasis web ini
7	Saya merasa orang lain akan memahami cara menggunakan aplikasi penilaian esai otomatis berbasis web ini dengan cepat
8	Saya merasa aplikasi penilaian esai otomatis berbasis web ini membingungkan
9	Saya merasa tidak ada hambatan dalam menggunakan aplikasi penilaian esai otomatis berbasis web ini
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan aplikasi penilaian esai otomatis berbasis web ini

3.7.2 Black Box

Pada waktu yang bersamaan, akan dilakukan pengujian Black Box untuk memastikan bahwa aplikasi yang dikembangkan bekerja sesuai dengan yang diharapkan. Pada tahap ini, penulis mungkin akan mendapatkan kecacatan program (*bug*), namun dengan ini penulis dapat memperbaiki aplikasi terkait dan menyempurnakannya. Daftar pengujian *Black Box* dapat dilihat secara rinci pada Tabel 3.9 dibawah ini.

Tabel 3.9 Daftar Pengujian Fungsional *Black Box*

PARENT ID	ID	Pengujian	Ekspektasi
F1A	F1A-1	Aplikasi web dapat menerima masukkan teks sebagai jawaban pengajar	Aplikasi web berhasil menyimpan teks sebagai jawaban pengajar
	F1A-2	Aplikasi web dapat menerima masukkan teks sebagai jawaban pelajar	Aplikasi web berhasil menerima teks sebagai jawaban pelajar
F1B	F1B-1	Aplikasi web dapat menerima masukkan dari fungsi F1A untuk penilaian otomatis tunggal bahasa Inggris	Aplikasi web berhasil menerima masukkan dari fungsi F1A untuk digunakan pada penilaian otomatis tunggal bahasa

PARENT ID	ID	Pengujian	Ekspektasi
			Inggris
	F1B-2	Aplikasi web dapat menerima masukka dari fungsi F1A untuk penilaian otomatis tunggal bahasa Indonesia	Aplikasi berhasil menerima masukkan dari fungsi F1A untuk digunakan pada penilaian otomatis tunggal bahasa Indonesia
F0	F0	Aplikasi dapat menggunakan model yang tepat untuk melakukan penilaian berdasarkan bahasa yang dipilih pengguna	Aplikasi web berhasil menerima permintaan bahasa yang digunakan untuk penilaian dan menggunakan bahasa tersebut untuk penilaian.
F2A	F2A-1	Aplikasi web dapat menerima masukkan berupa berkas berbentuk CSV sebagai kumpulan jawaban pelajar	Aplikasi web berhasil menyimpan berkas CSV sebagai kumpulan jawaban pelajar
	F2A-2	Aplikasi web dapat menerima masukkan teks sebagai kolom nama pada kumpulan jawaban pelajar	Aplikasi web berhasil menyimpan nama kolom yang berisi nama pelajar pada berkas CSV
	F2A-3	Aplikasi web dapat menerima masukkan teks sebagai kolom jawaban pada kumpulan jawaban pelajar	Aplikasi web berhasil menyimpan nama kolom yang berisi jawaban pelajar pada berkas CSV
	F2A-4	Aplikasi web dapat mendapatkan pasangan nama dan jawaban pada kumpulan jawaban pelajar sebagaimana ditentukan pada pengujian F2A-2 dan F2A-3	Aplikasi web berhasil menentukan pasangan nama dan jawaban pada berkas CSV untuk proses penilaian masal
F2B	F2B-1	Aplikasi web dapat membuat dokumen CSV berdasarkan pengujian F1A-1, dan F1A-2 atau F2A-1 agar cocok dengan syarat penilaian pada algoritma penilaian otomatis bahasa Inggris	Aplikasi web berhasil membuat berkas CSV dari berkas CSV, nama, dan jawaban pelajar untuk diproses pada model penilaian otomatis bahasa Inggris
	F2B-3	Algoritma penilaian otomatis bahasa Inggris dapat membuat dokumen .json setelah melakukan penilaian otomatis	Algoritma penilaian berhasil membuat dokumen .json sebagai hasil penilaian otomatis model bahasa Inggris
	F2B-5	Aplikasi dapat menampilkan hasil penilaian bahasa Inggris berdasarkan data .json yang dihasilkan oleh algoritma penilaian bahasa Inggris pada	Aplikasi web berhasil menggunakan dokumen .json yang telah dihasilkan penilaian otomatis bahasa Inggris sebelumnya untuk

PARENT ID	ID	Pengujian	Ekspektasi
		halaman web	ditampilkan pada halaman web
	F2B-2	Aplikasi web dapat membuat dokumen .json berdasarkan pengujian F1A-1, dan F1A-2 atau F2A-1 agar cocok dengan syarat penilaian pada algoritma penilaian otomatis bahasa Indonesia	Aplikasi web berhasil membuat dokumen .json setelah menentukan pasangan nama dan jawaban pada berkas CSV
	F2B-4	Algoritma penilaian otomatis bahasa Indonesia dapat membuat dokumen .json setelah melakukan penilaian otomatis	Algoritma penilaian otomatis bahasa Indonesia berhasil membuat dokumen .json untuk ditampilkan pada halaman web
F2	F2	Aplikasi penilaian otomatis dapat menampilkan hasil penilaian ke halaman web berdasarkan dokumen .json yang dihasilkan oleh algoritma penilaian otomatis.	Dokumen .json yang merupakan hasil penilaian otomatis dari algoritma bahasa Indonesia maupun bahasa Inggris berhasil ditampilkan pada halaman web

Pada Tabel 3.9 tersebut dapat diperhatikan ID dan PARENT ID untuk setiap pengujian *black box*. PARENT ID melambangkan bahwa pengujian memiliki fokus utama untuk suatu pengujian yang dipecah menjadi beberapa pengujian lainnya yang dilambangkan sebagai ID.

PARENT ID F1A merupakan pengujian untuk memastikan bahwa aplikasi web dapat menerima masukan teks dari pengguna sebagai variabel pada *backend* untuk dilakukan penilaian jawaban.

PARENT ID F1B merupakan pengujian untuk memastikan bahwa masukan teks dari F1A dapat digunakan sebagai variabel penilaian pada model *machine learning* yang akan digunakan yang penentuannya merupakan pengujian dari ID F0.

Selanjutnya, PARENT ID F2A adalah sekumpulan pengujian untuk menguji fungsi penilaian jamak melalui aplikasi web yang akan dikembangkan. Pengujian F2A memastikan bahwa *backend* akan menerima berkas CSV dan rinciannya yang akan

digunakan untuk memproses penilaian jamak untuk setiap model *machine learning* terpilih yang mana pengujiannya merupakan bagian dari PARENT ID F2B.

Setelah pengujian fungsional, dilakukan juga pengujian non-fungsional pada aplikasi yang dikembangkan penelitian ini dengan pengujian sebagai berikut.

Tabel 3.10 Daftar Pengujian Non Fungsional *Black Box*

PARENT ID	ID	Pengujian	Ekspektasi
NF1	NF1-1	Aplikasi dapat dijalankan pada ponsel	Aplikasi web berhasil diakses dan digunakan pada ponsel
	NF1-2	Aplikasi dapat dijalankan pada komputer	Aplikasi web berhasil diakses dan digunakan pada komputer
NF2	NF2	Aplikasi dapat dijalankan pada peramban web yang mendukung ES6	Aplikasi web berhasil diakses dan dijalankan pada peramban web yang telah mendukung ES6
NF3	NF3	Aplikasi tidak memiliki tampilan tumpang-tindih pada resolusi yang berbeda	Aplikasi web berhasil beradaptasi pada setiap resolusi dimana perlu ditampilkan tanpa adanya komponen yang tumpang-tindih

Pada Tabel 3.10, PARENT ID NF1 merupakan pengujian untuk tampilan yang dapat berubah (*responsive*) pada ukuran layar apapun yang sifatnya non-fungsional.

Selain pengujian fungsional dan non-fungsional, dilakukan juga pengujian teks pada aplikasi yang dikembangkan. Aplikasi yang dikembangkan menggunakan model pembelajaran mesin berbahasa Indonesia dan Inggris. Maka peneliti akan melakukan pengujian untuk mengetahui apa yang akan terjadi apabila pengguna memasukkan teks pada kondisi normal hingga kondisi ekstrem. Pada Tabel 3.11 berikut ini, ID percobaan menggambarkan kondisi aksi yang dilakukan, dengan ID pertama merupakan teks jawaban pengajar, ID kedua merupakan teks jawaban pelajar, dan ID ketiga merupakan model bahasa penilaian yang digunakan. ID "ID" berarti bahasa Indonesia, ID "EN" merupakan bahasa Inggris, ID "IDEN" merupakan teks

berbahasa Indonesia dengan campuran kata berbahasa Inggris, sedangkan ID "ENID" merupakan teks berbahasa Inggris dengan campuran kata berbahasa Indonesia.

Tabel 3.11 Daftar Pengujian Teks *Black Box*

ID	Aksi	Masukkan Pengajar	Masukkan Pelajar	Ekspektasi
ID-ID-ID	Pengguna memasukkan teks berbahasa Indonesia sebagai jawaban pelajar dan jawaban pelajar saat menggunakan model penilaian berbahasa Indonesia.	saya adalah pemilik tempat rekreasi ini	pemilik tempat rekreasi ini adalah saya	Menunjukkan skor penilaian normal.
ID-IDEN-ID	Pengguna memasukkan teks berbahasa Indonesia sebagai jawaban pelajar dan teks berbahasa Indonesia dengan kata campuran berbahasa Inggris pada jawaban pelajar saat menggunakan model penilaian berbahasa Indonesia	saya adalah pemilik tempat rekreasi ini	owner tempat rekreasi ini adalah saya	Menunjukkan skor penilaian yang berbeda dari pengujian ID-ID-ID.
IDEN-IDEN-ID	Pengguna memasukkan teks berbahasa Indonesia dengan kata campuran berbahasa Inggris pada jawaban pengajar dan jawaban pelajar saat menggunakan model penilaian berbahasa Indonesia.	saya adalah owner tempat rekreasi ini	owner tempat rekreasi ini adalah saya	Menunjukkan skor penilaian yang berbeda dari pengujian ID-IDEN-ID, namun kemungkinan mirip dengan skor pengujian ID-ID-ID.
EN-EN-EN	Pengguna memasukkan teks berbahasa Inggris pada jawaban pengajar dan jawaban pelajar saat menggunakan model penilaian berbahasa	My children work at South Jakarta	South Jakarta is where my children work	Menunjukkan skor penilaian normal.

ID	Aksi	Masukkan Pengajar	Masukkan Pelajar	Ekspektasi
	Inggris.			
EN-ENID-EN	Pengguna memasukkan teks berbahasa Inggris pada jawaban pengajar dan teks berbahasa Inggris dengan kata campuran berbahasa Indonesia pada jawaban pelajar saat menggunakan model penilaian berbahasa Inggris	My children work at South Jakarta	Jakarta Selatan is where my children work	Menunjukkan skor penilaian yang berbeda dari pengujian EN-EN-EN.
ENID-ENID-EN	Pengguna memasukkan teks berbahasa Inggris dengan kata campuran berbahasa Indonesia pada jawaban pengajar dan pelajar saat menggunakan model penilaian berbahasa Inggris.	My children work at Jakarta Selatan	Jakarta Selatan is where my children work	Menunjukkan skor penilaian yang berbeda dari pengujian EN-ENID-EN, namun kemungkinan mirip dengan skor pengujian EN-EN-EN.
EN-EN-ID	Pengguna memasukkan teks berbahasa Inggris pada jawaban pengajar dan pelajar saat menggunakan model penilaian berbahasa Indonesia.	My children work at South Jakarta	South Jakarta is where my children work	Menunjukkan skor penilaian.
EN-ENID-ID	Pengguna memasukkan teks berbahasa Inggris pada jawaban pengajar dan teks berbahasa Inggris dengan kata campuran berbahasa Indonesia pada jawaban pelajar saat menggunakan model penilaian berbahasa Indonesia.	My children work at South Jakarta	Jakarta Selatan is where my children work	Menunjukkan skor penilaian yang berbeda dari pengujian EN-EN-ID.
ENID-ENID-ID	Pengguna memasukkan teks berbahasa Inggris	My children work at Jakarta	My children work at Jakarta	Menunjukkan skor penilaian yang berbeda

ID	Aksi	Masukkan Pengajar	Masukkan Pelajar	Ekspektasi
	dengan kata campuran berbahasa Indonesia pada jawaban pengajar dan pelajar saat menggunakan model penilaian berbahasa Indonesia.	Selatan	Selatan	dari pengujian EN-ENID-ID, namun mirip dengan EN-EN-ID.
ID-ID-EN	Pengguna memasukkan teks berbahasa Indonesia pada jawaban pengajar dan pelajar saat menggunakan model penilaian berbahasa Inggris.	saya adalah pemilik tempat rekreasi ini	pemilik tempat rekreasi ini adalah saya	Menunjukkan skor penilaian.
ID-IDEN-EN	Pengguna memasukkan teks berbahasa Indonesia pada jawaban pengajar dan teks berbahasa Indonesia dengan kata campuran berbahasa Inggris saat menggunakan model penilaian berbahasa Inggris.	saya adalah pemilik tempat rekreasi ini	owner tempat rekreasi ini adalah saya	Menunjukkan skor penilaian yang berbeda dari pengujian ID-ID-EN.
IDEN-IDEN-EN	Pengguna memasukkan teks berbahasa Indonesia dengan kata campuran berbahasa Inggris pada jawaban pengajar dan jawaban pelajar saat menggunakan metode penilaian berbahasa Inggris.	saya adalah owner tempat rekreasi ini	owner tempat rekreasi ini adalah saya	Menunjukkan skor penilaian yang berbeda dari pengujian ID-IDEN-EN, namun mirip dengan ID-ID-EN.

Pada Tabel 3.11 diatas dapat diperhatikan semua kombinasi kemungkinan dari kondisi normal hingga kondisi ekstrem yang mungkin dilakukan oleh pengguna aplikasi. Pengujian tersebut dilakukan untuk mengetahui apa yang akan terjadi apabila pengguna salah menggunakan bahasa untuk melakukan penilaian teksnya, menggunakan dua bahasa pada sebuah teks, dan apa pengaruhnya pada penilaian.

Dengan ini, telah dilakukan pengujian fungsional dengan total 15 pengujian dengan hasil pengujian berhasil seluruhnya. Terpenuhinya pengujian fungsional ini menunjukkan bahwa aplikasi yang dikembangkan penelitian ini dapat berfungsi dengan baik hingga fungsi terinci. Pengujian non-fungsional juga dilakukan dengan keberhasilan 4 dari 4 pengujian. Hal ini dilakukan untuk memastikan pengguna memiliki pengalaman terbaik khususnya dari sisi tampilan dan kompatibilitas peramban web yang digunakan.

Selanjutnya dilakukan juga pengujian teks untuk meneliti semua kemungkinan yang dapat terjadi saat pengguna hendak menggunakan aplikasi. Pengujian ini dilakukan untuk melihat perilaku/keluaran aplikasi saat digunakan pada kondisi normal, sub-normal, hingga abnormal. Setelah dilakukan 12 pengujian teks tersebut, hasil pengujian sesuai dengan ekspektasi awal dan didapatkan analisis seperti yang telah dijelaskan sebelumnya.