

---

# **Software Requirements Specification**

**for**

**<Bloomers>**

**Version 1.0 approved**

**Prepared by <Kelompok 3 PPL Prak I1>**

**<Universitas Airlangga>**

**<10/10/2025>**

## Table of Contents

<b>Table of Contents.....</b>	<b>1</b>
<b>List of Figures.....</b>	<b>4</b>
<b>List of Tables.....</b>	<b>5</b>
<b>1. Introduction.....</b>	<b>6</b>
1.1 Purpose.....	6
1.2 Document Conventions.....	6
1.3 Intended Audience and Reading Suggestions.....	6
1.4 Product Scope.....	6
1.5 References.....	6
<b>2. Overall Description.....</b>	<b>7</b>
2.1 Product Perspective.....	7
2.2 Product Functions.....	7
2.3 User Classes and Characteristics.....	7
2.4 Operating Environment.....	8
2.5 Design and Implementation Constraints.....	8
2.6 User Documentation.....	8
2.7 Assumptions and Dependencies.....	8
<b>3. External Interface Requirements.....</b>	<b>9</b>
3.1 User Interfaces.....	9
3.2 Hardware Interfaces.....	9
3.3 Software Interfaces.....	9
3.4 Communication Interfaces.....	10
<b>4. System Features.....</b>	<b>10</b>
4.1 Sign Up.....	10
4.1.1 Description and Priority.....	10
4.1.2 Stimulus/Response Sequences:.....	10
4.1.3 Functional Requirements:.....	10
4.2 Upload Question File.....	11
4.3 View Classification Result.....	11
4.3.1 Description and Priority.....	11
4.3.2 Stimulus/Response Sequences:.....	11
4.3.3 Functional Requirements:.....	12
4.4 Download Report.....	12
4.4.1 Description and Priority:.....	12
4.4.2 Stimulus/Response Sequences:.....	12
4.4.3 Functional Requirements:.....	12
4.5 Regenerate Question.....	12
4.5.1 Description and Priority.....	12
4.5.2 Stimulus/Response Sequences:.....	12
4.5.3 Functional Requirements:.....	13

4.6 Logout.....	13
4.6.1 Description and Priority.....	13
4.6.2 Stimulus/Response Sequences:.....	13
4.6.3 Functional Requirements:.....	13
<b>5. Other Nonfunctional Requirements.....</b>	<b>13</b>
5.1 Performance Requirements.....	13
5.2 Safety Requirements.....	14
5.3 Security Requirements.....	14
5.4 Software Quality Attributes.....	14
5.5 Business Rules.....	14
<b>6. Other Requirements.....</b>	<b>15</b>
6.1 Database Requirements.....	15
6.2 Security Requirements.....	15
6.3 Legal and Ethical Requirements.....	16
6.4 Maintainability Requirements.....	16
6.5 Performance and Scalability Requirements.....	16
6.6 Usability and Documentation Requirements.....	17
6.7 Future Development Considerations.....	17
<b>7. Artifact.....</b>	<b>18</b>
7.1 Use Case Diagram.....	18
7.2 Use Case Specification.....	18
7.2.1 Unggah File Soal.....	18
7.2.2 Melihat Hasil Klasifikasi.....	19
7.2.3 Mengunduh Laporan.....	19
7.2.4 Regenerate Soal Based On Level.....	20
7.2.5 Sign Up.....	21
7.2.6 Login.....	21
7.2.7 Logout.....	22
7.3 User Story.....	22
7.4 User Story Scenario.....	27
7.5 GUI Design.....	30
7.5.1 Sign Up page.....	30
7.5.2 Sign In page.....	31
7.5.3 Page utama.....	32
7.5.4 Page Review Soal.....	33
7.5.5 Page overview soal.....	34
7.5.6 Page Lihat History.....	35
7.6 Database.....	35
7.7 Activity Diagram.....	38
7.7.1 Activity Diagram Mengunduh Laporan.....	38
7.7.2 Activity Diagram Mengunggah Laporan.....	39

7.7.3 Activity Diagram Melihat Hasil Klasifikasi.....	40
7.7.4 Activity Diagram Melihat History Klasifikasi.....	41
7.7.5 Activity Diagram Regenerate Soal Based on Level.....	42
7.7.6 Activity Diagram Login.....	43
7.7.7 Activity Diagram Logout.....	44
7.7.8 Activity Diagram Sign Up.....	45
7.8 Sequence Diagram.....	46
7.8.1 Sequence Diagram Sign Up.....	46
7.8.2 Sequence Diagram Mengunduh Laporan.....	47
7.8.3 Sequence Diagram Mengunggah Laporan.....	48
7.8.4 Sequence Diagram Melihat Hasil Klasifikasi.....	49
7.8.5 Sequence Diagram Melihat History Klasifikasi.....	50
7.8.6 Sequence Diagram Regenerate Soal Based on Level.....	51
7.8.7 Sequence Diagram Login.....	52
7.8.7 Sequence Diagram Logout.....	53
7.9 Class Diagram.....	54
<b>Appendix A: Glossary.....</b>	<b>55</b>
<b>Appendix B: Analysis Models.....</b>	<b>56</b>
<b>Appendix C: To Be Determined List.....</b>	<b>56</b>

## List of Figures

Gambar 7.1 Use Case Diagram.....	18
Gambar 7.2 Sign Up Page.....	30
Gambar 7.3 Sign In Page.....	31
Gambar 7.4 Page Utama Upload.....	32
Gambar 7.5 Page Review Soal.....	33
Gambar 7.6 Page Overview Soal.....	34
Gambar 7.7 Page Lihat History.....	35
Gambar 7.8 Table Database.....	36
Gambar 7.9 Database File Soal.....	36
Gambar 7.10 Database Laporan.....	36
Gambar 7.11 Database Regenerate Soal.....	37
Gambar 7.12 Database Review History.....	37
Gambar 7.13 Database User.....	37
Gambar 7.14 Activity Diagram Mengunduh Laporan.....	38
Gambar 7.14 Activity Diagram Mengunggah Laporan.....	39
Gambar 7.15 Activity Diagram Melihat Hasil Klasifikasi.....	40
Gambar 7.16 Activity Diagram Melihat History Klasifikasi.....	41
Gambar 7.17 Activity Diagram Regenerate Soal Based on Level.....	42
Gambar 7.18 Activity Diagram Login.....	43
Gambar 7.19 Activity Diagram Logout.....	44
Gambar 7.20 Activity Diagram Sign Up.....	45
Gambar 7.21 Sequence Diagram Sign Up.....	46
Gambar 7.22 Sequence Diagram Mengunduh Laporan.....	47
Gambar 7.23 Sequence Diagram Mengunggah Laporan.....	48
Gambar 7.24 Sequence Diagram Melihat Hasil Klasifikasi.....	49
Gambar 7.25 Sequence Diagram Melihat History Klasifikasi.....	50
Gambar 7.26 Sequence Diagram Regenerate Soal Based on Level.....	51
Gambar 7.27 Sequence Diagram Login.....	52
Gambar 7.28 Sequence Diagram Logout.....	53
Gambar 7.29 Class Diagram.....	54

## **List of Tables**

Tabel 7.1 Use Case Specification Unggah File Soal.....	18
Tabel 7.2 Use Case Specification Melihat Hasil Klasifikasi.....	19
Tabel 7.3 Use Case Specification Mengunduh Laporan.....	19
Tabel 7.4 Use Case Specification Regenerate Soal Based On Level.....	20
Tabel 7.5 Use Case Specification Sign Up.....	21
Tabel 7.6 Use Case Specification Login.....	21
Tabel 7.7 Use Case Specification Logout.....	22
Tabel 7.8 User Story Sign Up.....	23
Tabel 7.9 User Story Login.....	23
Tabel 7.10 User Story Upload Question.....	24
Tabel 7.11 User Story View Classification Results.....	25
Tabel 7.12 User Story Regenerate Question Based on Level.....	25
Tabel 7.13 User Story Download Report.....	26
Tabel 7.14 User Story Logout.....	27
Tabel 7.15 User Story Scenario Sign Up.....	27
Tabel 7.16 User Story Scenario Login.....	28
Tabel 7.17 User Story Scenario Upload Question File.....	28
Tabel 7.18 User Story Scenario Regenerate Question to New Level.....	29
Tabel 7.19 User Story Scenario Download Report.....	29
Tabel 7.20 User Story Scenario Logout.....	30
Tabel Glosary.....	55
Tabel To Be Determined List.....	56
Tabel Lampiran Kontribusi Anggota Kelompok.....	58

## **1. Introduction**

### **1.1 Purpose**

Tujuan dari dokumen ini adalah untuk mendeskripsikan secara rinci aplikasi Bloomers, sebuah sistem klasifikasi otomatis tingkat kognitif soal berdasarkan Taksonomi Bloom. Dokumen ini merinci tujuan, fitur, antarmuka, batasan, serta kebutuhan fungsional dan non-fungsional dari sistem. Dokumen ini ditujukan untuk menjadi panduan bagi pengembang, penguji, dan pemangku kepentingan agar memiliki pemahaman yang sama mengenai perangkat lunak yang akan dibangun.

### **1.2 Document Conventions**

Dokumen ini disusun mengikuti format standar spesifikasi kebutuhan perangkat lunak. Setiap kebutuhan fungsional diberi nomor unik (contoh: REQ\_01) untuk kemudahan pelacakan dan referensi.

### **1.3 Intended Audience and Reading Suggestions**

**Developers:** Akan menggunakan dokumen ini untuk memahami spesifikasi teknis sebagai dasar perancangan dan implementasi sistem.

**Educators/Testers:** Akan menggunakan dokumen ini untuk memahami semua fitur sistem dan merancang skenario pengujian.

**Project Managers:** Akan menggunakan dokumen ini untuk memahami lingkup dan fungsionalitas produk secara keseluruhan

### **1.4 Product Scope**

Bloomers adalah aplikasi berbasis web dan desktop yang berfungsi untuk mengklasifikasikan soal ujian atau kuis ke dalam enam tingkat kognitif Taksonomi Bloom (C1-C6) secara otomatis. Sistem ini dirancang untuk membantu tenaga pendidik meningkatkan efisiensi dan objektivitas dalam mengevaluasi instrumen soal.

### **1.5 References**

*Institute of Electrical and Electronics Engineers (IEEE), Inc. (1998). IEEE Standard 1016-1998, IEEE Recommended Practice for Software Requirements Specifications. New York, NY, USA: IEEE, Inc..*

## **2. Overall Description**

### **2.1 Product Perspective**

Bloomers merupakan aplikasi baru yang dikembangkan secara mandiri dan tidak bergantung pada sistem lain. Sistem ini dirancang sebagai solusi otomatis untuk membantu tenaga pendidik melakukan klasifikasi tingkat kognitif soal berdasarkan Taksonomi Bloom. Aplikasi bekerja dengan menerima file input berisi soal, kemudian memproses teks menggunakan model Natural Language Processing (NLP) berbasis RoBERTa untuk menghasilkan klasifikasi ke dalam enam level kognitif (C1–C6). Bloomers akan beroperasi sebagai aplikasi berbasis web dan desktop, dengan antarmuka pengguna intuitif serta integrasi ke database internal yang menyimpan data pengguna, file soal, hasil klasifikasi, history aktivitas, dan laporan hasil analisis.

### **2.2 Product Functions**

Sistem Bloomers akan menyediakan fungsi-fungsi utama sebagai berikut:

- Sistem harus mengizinkan pengguna untuk membuat akun (Sign Up) dan masuk (Login).
- Sistem harus mengizinkan pengguna mengunggah file soal.
- Sistem harus dapat melakukan klasifikasi otomatis pada soal yang diunggah.
- Sistem harus menampilkan hasil klasifikasi secara rinci dan dalam bentuk ringkasan (overview).
- Sistem harus mengizinkan pengguna mengunduh laporan analisis.
- Sistem harus menyediakan fitur untuk meregenerasi soal berdasarkan level kognitif yang dipilih.
- Sistem harus mengizinkan pengguna untuk keluar dari akun (Logout).

### **2.3 User Classes and Characteristics**

**Pengajar (Educator):** Pengguna yang memiliki wewenang untuk mengakses semua fitur sistem. Mereka dapat mendaftar, login, mengelola file soal, melihat hasil

analisis untuk mengevaluasi kualitas bank soal, dan mengunduh laporan untuk keperluan dokumentasi.

## **2.4 Operating Environment**

Bloomers akan dijalankan pada lingkungan berikut:

- Hardware: Komputer atau laptop dengan minimal RAM 4 GB dan prosesor dual-core.
- Operating System: Windows 10/11, Linux Ubuntu 22.04, atau macOS Monterey ke atas.
- Browser Compatibility: Google Chrome, Mozilla Firefox, dan Microsoft Edge.
- Framework & Tools: Backend berbasis Python (Flask/Django), Frontend berbasis React, serta Database PostgreSQL atau MySQL.
- NLP Model Dependencies: Library HuggingFace Transformers, Pandas, dan Scikit-Learn.

## **2.5 Design and Implementation Constraints**

- Aplikasi dikembangkan menggunakan bahasa Python dan JavaScript (React.js).
- Model NLP yang digunakan adalah RoBERTa fine-tuned untuk klasifikasi teks berdasarkan dataset soal edukatif.
- File input hanya dapat diterima dalam format CSV, PDF, DOCX, dan TXT.
- Sistem harus beroperasi secara stateless untuk keamanan sesi pengguna.
- Proses regenerasi soal bergantung pada koneksi internet karena memerlukan model teks eksternal (text generation model).
- Database hanya dapat diakses oleh akun dengan hak admin.

## **2.6 User Documentation**

Dokumentasi pengguna akan mencakup:

- Panduan Pengguna (User Manual) dalam format PDF.
- Help Page di dalam aplikasi, berisi penjelasan fitur utama dan langkah penggunaan.
- Video Tutorial yang menjelaskan cara mengunggah file, melihat hasil klasifikasi, dan mengunduh laporan.

## **2.7 Assumptions and Dependencies**

- Diasumsikan pengguna memiliki koneksi internet stabil untuk menjalankan klasifikasi dan regenerasi soal.
- Sistem bergantung pada model NLP eksternal (RoBERTa) dan library Python tertentu.
- Keakuratan hasil klasifikasi bergantung pada kualitas dataset pelatihan model.
- Pembaruan sistem dapat mempengaruhi performa dan kompatibilitas model lama.

## **3. External Interface Requirements**

### **3.1 User Interfaces**

Antarmuka Bloomers didesain agar sederhana dan ramah pengguna:

- Halaman Login dan Registrasi: Form input sederhana dengan validasi real-time.
- Dashboard Utama: Menampilkan menu unggah file, hasil klasifikasi, history aktivitas, dan laporan.
- Halaman Klasifikasi: Menampilkan tabel hasil klasifikasi per soal serta grafik distribusi C1–C6.
- Halaman Regenerate soal: Menampilkan soal asli dan hasil regenerasi secara berdampingan.
- Halaman Laporan: Tombol unduh PDF dan tampilan ringkasan hasil.

### **3.2 Hardware Interfaces**

Bloomers tidak memerlukan perangkat keras khusus selain komputer/laptop standar. Sistem hanya bergantung pada:

- Perangkat input: Keyboard dan mouse.
- Penyimpanan lokal: Untuk menyimpan hasil laporan PDF.
- Server atau cloud storage: Untuk menyimpan file yang diunggah dan hasil klasifikasi.

### **3.3 Software Interfaces**

Operating System: Windows, Linux, atau macOS.

- Database Management System: PostgreSQL/MySQL.
- NLP Model: Terhubung ke pustaka *transformers* (HuggingFace).
- Libraries & Frameworks: Flask/Django, React.js, Pandas, Scikit-Learn, dan PyTorch.
- File Management: Mendukung input dari format CSV, PDF, DOCX, dan TXT.

### **3.4 Communication Interfaces**

Sistem akan menggunakan HTTP/HTTPS untuk komunikasi antara frontend dan backend.

- Pengunggahan file dan unduhan laporan dilakukan melalui protokol REST API.
- Semua komunikasi dienkripsi dengan SSL/TLS untuk menjaga keamanan data pengguna.
- Server mendukung integrasi melalui port standar 80 (HTTP) dan 443 (HTTPS).

## **4. System Features**

Bagian ini menjelaskan fitur-fitur utama yang disediakan oleh sistem Bloomers,yaitu aplikasi klasifikasi otomatis tingkat kognitif soal berdasarkan Taksonomi Bloom. Setiap fitur dijabarkan berdasarkan fungsi utama yang mendukung kebutuhan pengguna dalam melakukan pengelolaan dan analisis soal.

### **4.1 Sign Up**

#### **4.1.1 Description and Priority**

Fitur ini memungkinkan pengguna baru untuk mendaftarkan akun agar dapat mengakses sistem. **Priority: High**.

#### **4.1.2 Stimulus/Response Sequences:**

- Pengguna memilih menu "Sign Up".
- Sistem menampilkan form pendaftaran (Username, Email, Password).
- Pengguna mengisi data dan menekan tombol "Sign Up".
- Sistem memvalidasi dan mendaftarkan akun baru ke database.
- Sistem mengalihkan pengguna ke halaman Login.

#### **4.1.3 Functional Requirements:**

- REQ\_01: Sistem harus menyediakan form pendaftaran dengan kolom Username, Email, dan Password.
- REQ\_02: Sistem harus menyimpan akun baru ke database setelah validasi berhasil.
- REQ\_03: Sistem harus menampilkan notifikasi jika data yang dimasukkan tidak lengkap atau tidak valid

### **4.2 Upload Question File**

#### **4.2.1 Description and Priority**

Fitur ini memungkinkan pengguna untuk mengunggah file soal.

**Priority: High.**

#### **4.2.2 Stimulus/Response Sequences:**

- Pengguna memilih file dari perangkat.
- Sistem memvalidasi format file.
- Sistem menampilkan file yang diunggah dan siap untuk diproses.

#### **4.2.3 Functional Requirements:**

- REQ\_04: Sistem akan mengharuskan pengguna untuk login sebelum mengunggah file.
- REQ\_05: Sistem hanya akan menerima file dalam format CSV, PDF, DOCX, atau TXT.
- REQ\_06: Sistem akan menampilkan pesan kesalahan jika format file tidak valid.

### **4.3 View Classification Result**

#### **4.3.1 Description and Priority**

Fitur ini memungkinkan pengguna melihat hasil klasifikasi soal dalam bentuk tabel dan grafik. **Priority: High.**

#### **4.3.2 Stimulus/Response Sequences:**

- Setelah proses klasifikasi selesai, pengguna memilih menu "Lihat Hasil".

- Sistem mengambil data hasil klasifikasi.
- Sistem menampilkan hasil dalam dua tab: "Rincian Soal" dan "Overview".

#### **4.3.3 Functional Requirements:**

- REQ\_10: Sistem hanya dapat menampilkan hasil jika proses klasifikasi telah selesai.
- REQ\_11: Sistem harus menampilkan hasil dalam bentuk tabel detail dan grafik distribusi.
- REQ\_12: Sistem harus menampilkan pesan "Data belum tersedia" jika belum ada hasil klasifikasi.

### **4.4 Download Report**

#### **4.4.1 Description and Priority:**

Fitur ini memungkinkan pengguna mengunduh laporan hasil analisis dalam format PDF. **Priority: High.**

#### **4.4.2 Stimulus/Response Sequences:**

- Pengguna menekan tombol "Download" pada halaman hasil klasifikasi.
- Sistem membuat file laporan dalam format PDF.
- Sistem mengirimkan file PDF ke perangkat pengguna untuk disimpan.

#### **4.4.3 Functional Requirements:**

- REQ\_13: Sistem harus menyediakan tombol untuk mengunduh laporan.
- REQ\_14: Laporan yang dihasilkan harus dalam format PDF.
- REQ\_15: Sistem harus menampilkan pesan error jika terjadi kegagalan saat membuat laporan.

### **4.5 Regenerate Question**

#### **4.5.1 Description and Priority**

Fitur opsional yang memungkinkan pengguna mengubah redaksi soal ke level kognitif lain. **Priority: Medium.**

#### **4.5.2 Stimulus/Response Sequences:**

- Pengguna memilih opsi "Regenerate Soal".
- Sistem menampilkan level kognitif awal soal tersebut.
- Pengguna memilih level target baru (C1-C6) yang diinginkan.
- Sistem melakukan proses regenerasi teks.
- Sistem menampilkan hasil regenerasi soal di layar.

#### **4.5.3 Functional Requirements:**

- REQ\_16: Pengguna harus dapat memilih level kognitif target baru dari C1 hingga C6.
- REQ\_17: Sistem harus menampilkan soal baru hasil regenerasi kepada pengguna.
- REQ\_18: Sistem harus menampilkan pesan error jika proses regenerasi teks gagal.

### **4.6 Logout**

#### **4.6.1 Description and Priority**

Fitur ini memungkinkan pengguna untuk keluar dari sesi aktif.  
**Priority: High.**

#### **4.6.2 Stimulus/Response Sequences:**

- Pengguna memilih menu "Logout".
- Sistem menampilkan pesan konfirmasi.
- Pengguna menekan tombol "Iya".
- Sistem mengakhiri sesi dan menampilkan halaman Login.

#### **4.6.3 Functional Requirements:**

- REQ\_19: Sistem harus menyediakan mekanisme bagi pengguna yang sedang login untuk keluar.
- REQ\_20: Setelah logout, sistem harus mengakhiri sesi pengguna dan mengembalikannya ke halaman Login.

## **5. Other Nonfunctional Requirements**

### **5.1 Performance Requirements**

Waktu respons sistem untuk menampilkan hasil klasifikasi harus cepat dan tidak membuat pengguna menunggu terlalu lama. Aplikasi harus mampu menangani unggahan file dengan ukuran yang wajar tanpa mengalami penurunan performa yang signifikan.

### **5.2 Safety Requirements**

Sistem harus mencegah kehilangan data pengguna atau hasil klasifikasi dengan menerapkan:

- Mekanisme auto-save hasil proses.
- Backup database secara berkala.
- Peringatan konfirmasi sebelum pengguna menghapus file atau keluar dari sesi tanpa menyimpan hasil. Selain itu, sistem tidak akan memproses file yang terindikasi berbahaya atau mengandung malware.

### **5.3 Security Requirements**

Semua data otentikasi pengguna, terutama password, harus dienkripsi saat disimpan di database.

Sistem harus melindungi data pengguna dan file yang diunggah dari akses yang tidak sah.

### **5.4 Software Quality Attributes**

- Availability: Aplikasi web harus dapat diakses oleh pengguna setiap saat dengan downtime minimal.
- Correctness: Hasil klasifikasi yang ditampilkan harus akurat dan sesuai dengan performa model yang telah dilatih.
- Maintainability: Kode sumber harus ditulis mengikuti standar pemrograman yang baik dan diberi anotasi yang cukup untuk memudahkan pemeliharaan dan pengembangan di masa depan.
- Usability: Antarmuka pengguna harus intuitif dan mudah dipahami oleh target pengguna (pengajar) tanpa memerlukan pelatihan khusus.

## **5.5 Business Rules**

- Hanya pengguna terdaftar yang dapat mengakses sistem.
- Setiap pengguna dapat melihat hanya file dan hasil klasifikasinya sendiri.
- Admin memiliki hak untuk melihat semua aktivitas dan melakukan pemeliharaan database.
- Laporan yang diunduh bersifat final dan tidak dapat diubah setelah dihasilkan.

## **6. Other Requirements**

Bagian ini menjelaskan kebutuhan tambahan yang tidak termasuk dalam kategori fungsional dan non-fungsional, namun penting untuk menjamin keberlanjutan, keamanan, dan keandalan sistem Bloomers dalam penggunaannya.

### **6.1 Database Requirements**

Sistem Bloomers menggunakan basis data relasional untuk menyimpan seluruh data yang berkaitan dengan pengguna, file soal, hasil klasifikasi, riwayat review, regenerasi soal, dan laporan hasil analisis.

Database harus memenuhi kriteria berikut:

- Setiap tabel memiliki primary key unik dan relasi antar tabel dijaga dengan foreign key untuk menjaga integritas data.
- Data pengguna (akun, hasil klasifikasi, dan file soal) harus dapat diakses sesuai hak pengguna yang terautentikasi.
- Kapasitas database harus mampu menampung data dari banyak pengguna dan file secara bersamaan tanpa mengurangi performa sistem.
- Proses backup data dilakukan secara berkala untuk mencegah kehilangan data akibat kesalahan sistem atau pengguna.

### **6.2 Security Requirements**

Sistem harus menjamin keamanan data pengguna dan menjaga privasi setiap informasi yang disimpan.

Beberapa ketentuan keamanan yang diterapkan meliputi:

- Data sensitif seperti password disimpan dalam bentuk terenkripsi.
- Setiap pengguna hanya dapat mengakses data miliknya sendiri.

- Sistem menerapkan autentikasi login agar hanya pengguna terdaftar yang dapat masuk ke aplikasi.
- Setiap aktivitas pengguna seperti unggah file, klasifikasi, dan regenerasi tercatat dalam log aktivitas untuk keperluan audit.

### **6.3 Legal and Ethical Requirements**

Penggunaan sistem Bloomers harus sesuai dengan etika dan peraturan yang berlaku di lingkungan akademik.

Ketentuan yang diterapkan antara lain:

- Soal yang diunggah merupakan tanggung jawab pengguna, dan sistem tidak berhak mendistribusikan atau mempublikasikan file tersebut tanpa izin.
- Sistem hanya digunakan untuk kepentingan pembelajaran dan penelitian, bukan untuk kepentingan komersial.
- Penggunaan dataset dan model Natural Language Processing (NLP) mengikuti lisensi sumber terbuka (open source license).

### **6.4 Maintainability Requirements**

Sistem Bloomers dirancang agar mudah diperbaiki dan diperbarui sesuai kebutuhan.

Upaya pemeliharaan mencakup:

- Perawatan berkala terhadap database dan server untuk menjaga performa.
- Dokumentasi kode dan struktur sistem disiapkan agar pengembang lain dapat memahami alur kerja sistem.
- Peningkatan dan pengujian model NLP dilakukan secara berkala untuk menjaga akurasi klasifikasi.

### **6.5 Performance and Scalability Requirements**

Sistem harus mampu memproses file soal dari berbagai ukuran dengan waktu yang efisien.

Untuk menjamin hal tersebut, diterapkan beberapa ketentuan berikut:

- Sistem mampu menangani unggahan file dalam jumlah besar secara simultan.

- Setiap proses klasifikasi dijalankan secara otomatis dan menampilkan hasil tanpa jeda yang signifikan.
- Struktur aplikasi dibuat modular agar sistem dapat diperluas untuk menampung pengguna baru tanpa perubahan besar pada arsitektur.

## 6.6 Usability and Documentation Requirements

Sistem harus mudah digunakan oleh pengguna dengan latar belakang pendidikan yang beragam.

Untuk mendukung hal ini, sistem menyediakan:

- Antarmuka yang sederhana dan mudah dipahami.
- Panduan penggunaan (*user guide*) yang menjelaskan langkah-langkah menggunakan fitur utama seperti unggah soal, melihat hasil klasifikasi, dan mengunduh laporan.
- Informasi bantuan dalam aplikasi untuk mempermudah pengguna baru.

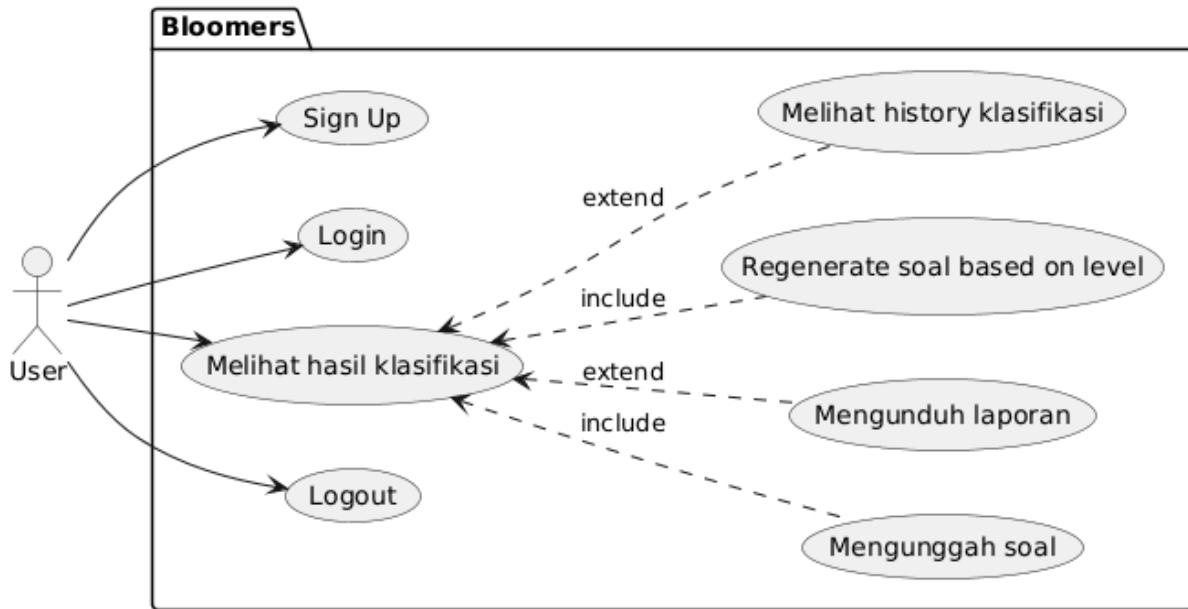
## 6.7 Future Development Considerations

Sistem **Bloomers** masih memiliki ruang untuk pengembangan lanjutan, antara lain:

- Integrasi dengan *Learning Management System (LMS)* seperti Moodle atau Google Classroom.
- Pengembangan fitur analisis tingkat kesulitan soal.
- Penambahan versi aplikasi mobile agar pengguna dapat mengakses sistem dengan lebih fleksibel.

## 7. Artifact

### 7.1 Use Case Diagram



Gambar 7.1 Use Case Diagram

Use case diagram “Bloomers” menggambarkan interaksi antara pengguna (User) dengan sistem yang berfungsi untuk mengelola proses klasifikasi soal. Pengguna dapat melakukan beberapa aktivitas utama seperti Sign Up untuk mendaftar akun baru, Login untuk masuk ke sistem, dan Logout untuk keluar setelah selesai menggunakan aplikasi. Fitur utama sistem adalah Melihat hasil klasifikasi, yang menjadi pusat dari berbagai aktivitas lain. Use case ini memiliki hubungan include dengan aktivitas Mengunggah soal dan Mengunduh laporan, karena kedua proses tersebut merupakan bagian dari langkah melihat hasil klasifikasi. Selain itu, terdapat hubungan extend dengan aktivitas Regenerate soal based on level dan Melihat history klasifikasi, yang berarti kedua fitur tersebut bersifat opsional dan hanya dijalankan ketika pengguna ingin memperbarui soal atau meninjau riwayat klasifikasi sebelumnya. Secara keseluruhan, diagram ini menunjukkan bagaimana sistem Bloomers mendukung proses manajemen soal berbasis klasifikasi secara terstruktur dan interaktif.

### 7.2 Use Case Specification

#### 7.2.1 Unggah File Soal

Tabel 7.1 Use Case Specification Unggah File Soal

Nama Use Case	Unggah File Soal
Aktor	User
Deskripsi	User mengunggah file berisi kumpulan soal agar dapat diklasifikasikan oleh Bloomers
Trigger	User menekan tombol “Upload”

Pre-condition	<ol style="list-style-type: none"> <li>1. User telah login</li> <li>2. File tersedia dalam format CSV, PDF, DOCX, atau TXT.</li> </ol>
Basic Flow	<ol style="list-style-type: none"> <li>1. User memilih file dari perangkat</li> <li>2. Sistem memvalidasi format file</li> <li>3. Sistem menyimpan file sementara</li> </ol>
Post-condition	File berhasil diunggah dan siap diproses oleh sistem
Exception Flow	Jika format tidak valid, sistem akan menampilkan pesan kesalahan

### 7.2.2 Melihat Hasil Klasifikasi

**Tabel 7.2 Use Case Specification Melihat Hasil Klasifikasi**

Nama Use Case	Melihat Hasil Klasifikasi
Aktor	User
Deskripsi	User hasil klasifikasi soal berdasarkan tingkat kognitif Taksonomi Bloom (C1-C6) setelah proses klasifikasi selesai
Trigger	User memilih menu “Lihat Hasil Klasifikasi”
Pre-condition	Soal telah berhasil diproses dan diklasifikasikan
Basic Flow	<ol style="list-style-type: none"> <li>1. User memilih menu “Lihat hasil klasifikasi”</li> <li>2. Sistem mengambil data hasil klasifikasi</li> <li>3. Sistem menampilkan tabel klasifikasi beserta grafik distribusi</li> </ol>
Post-condition	Hasil klasifikasi ditampilkan dalam bentuk tabel dan grafik distribusi
Exception Flow	Jika belum ada hasil klasifikasi, sistem menampilkan pesan “Data belum tersedia”

### 7.2.3 Mengunduh Laporan

**Tabel 7.3 Use Case Specification Mengunduh Laporan**

Nama Use Case	Mengunduh Laporan
Aktor	User
Deskripsi	User mengunduh laporan hasil klasifikasi dalam format PDF sebagai dokumentasi atau bahan evaluasi
Trigger	Pengejar memilih menu “Unduh Laporan”

Pre-condition	Hasil klasifikasi sudah ditampilkan di perangkat User
Basic Flow	<ol style="list-style-type: none"> <li>1. User memilih menu “Unduh Laporan”</li> <li>2. Sistem membuat file laporan dalam format PDF</li> <li>3. Sistem mengirimkan file PDF ke perangkat User untuk disimpan</li> </ol>
Post-condition	Laporan PDF tersimpan di perangkat User
Exception Flow	Jika terjadi kesalahan saat membuat laporan, sistem menampilkan pesan error

#### 7.2.4 Regenerate Soal Based On Level

**Tabel 7.4 Use Case Specification Regenerate Soal Based On Level**

Nama Use Case	Regenerate Soal Based On Level
Aktor	User
Deskripsi	User dapat melakukan regenerasi soal berdasarkan level kognitif Bloom tertentu (C1–C6). Setelah sistem menampilkan hasil klasifikasi soal, User memiliki opsi untuk memilih level target baru agar sistem menghasilkan redaksi soal yang sesuai dengan level tersebut.
Trigger	User memilih menu “Regenerate Soal”
Pre-condition	Hasil klasifikasi sudah ditampilkan di perangkat User
Basic Flow	<ol style="list-style-type: none"> <li>1. User memilih soal target yang ingin di regenerate</li> <li>2. User memilih menu “Regenerate Soal”.</li> <li>3. Sistem menampilkan level kognitif awal dari soal (misalnya: C3).</li> <li>4. User memilih level target yang diinginkan (C1–C6).</li> <li>5. Sistem menyiapkan template/pola soal sesuai level target.</li> <li>6. Sistem melakukan regenerasi soal dengan model text generation.</li> <li>7. Sistem menampilkan hasil regenerasi soal pada layar.</li> </ol>
Post-condition	Soal baru dengan level target Bloom tersedia di perangkat User (ditampilkan di layar atau disimpan sesuai pilihan User).
Exception Flow	Jika terjadi kesalahan pada proses text generation,

	sistem menampilkan pesan error dan menyarankan User untuk mencoba kembali.
--	--

### 7.2.5 Sign Up

**Tabel 7.5 Use Case Specification Sign Up**

Nama Use Case	Sign Up
Aktor	User
Deskripsi	User melakukan pendaftaran akun baru
Trigger	Pengejar memilih menu “Sign Up”
Pre-condition	User belum memiliki akun
Basic Flow	<ol style="list-style-type: none"> <li>1. User memilih menu “Sign Up”</li> <li>2. Sistem menampilkan form pembuatan akun baru</li> <li>3. User menngisikan data yang diperlukan</li> <li>4. User menekan tombol “Sign Up”</li> <li>5. Sistem mendaftarkan akun baru ke database</li> <li>6. Sistem mengalihkan ke halaman login</li> </ol>
Alternate Flow	<ol style="list-style-type: none"> <li>5a. Sistem menampilkan notifikasi data yang kurang lengkap atau kurang tepat</li> <li>6a. User memasukkan data yang kurang atau membenarkan data yang salah</li> <li>7a. User menekan tombol “Sign Up”+</li> <li>8a. Sistem mendaftarkan akun baru ke database</li> <li>9a. Sistem mengalihkan ke halaman login</li> </ol>
Post-condition	User telah memiliki akun
Exception Flow	Jika terjadi kesalahan saat proses Sign Up, sistem akan menampilkan pesan eror dan mengulangi proses Sign Up

### 7.2.6 Login

**Tabel 7.6 Use Case Specification Login**

Nama Use Case	Login
Aktor	User
Deskripsi	User masuk ke akun yang telah ada
Trigger	Pengejar memilih menu “Login”
Pre-condition	<ol style="list-style-type: none"> <li>1. User telah memiliki akun</li> </ol>

	2. User belum melakukan “Login”
Basic Flow	<ol style="list-style-type: none"> <li>1. User memilih menu “Login”</li> <li>2. Sistem menampilkan form login</li> <li>3. User mengisi form login dengan username/email dan password</li> <li>4. User menekan tombol “Login”</li> <li>5. Sistem memvalidasi username/email dan password</li> <li>6. Sistem mengarahkan ke halaman beranda</li> </ol>
Post-condition	User telah login dan masuk ke beranda
Exception Flow	Jika terjadi kesalahan saat proses Login, sistem akan menampilkan pesan eror dan mengulangi proses Login

### 7.2.7 Logout

**Tabel 7.7 Use Case Specification Logout**

Nama Use Case	Logout
Aktor	User
Deskripsi	User keluar dari akun yang sedang digunakan
Trigger	Pengejar memilih menu “Logout”
Pre-condition	User telah melakukan login
Basic Flow	<ol style="list-style-type: none"> <li>1. User memilih menu “Logout”</li> <li>2. Sistem menampilkan pesan “Apakah anda yakin ingin logout?”</li> <li>3. User menekan tombol “Iya”</li> <li>4. Sistem mengeluarkan akun yang sedang digunakan</li> <li>5. Sistem menampilkan halaman “Login”</li> </ol>
Alternate Flow	<ol style="list-style-type: none"> <li>3a. User menekan tombol “Tidak”</li> <li>4a. Sistem kembali ke tampilan beranda</li> </ol>
Post-condition	User telah keluar dari akun yang sedang digunakan
Exception Flow	Jika terjadi kesalahan saat proses Logout, sistem akan menampilkan pesan eror dan tidak mengeluarkan akun yang sedang digunakan

### 7.3 User Story

1. User Story

a. Sign Up

**Tabel 7.8 User Story Sign Up**

User Story	<p><b>As a new user</b></p> <p><b>I want to</b> sign up for an account by providing a username, email, and password</p> <p><b>So that I</b> can have an account</p>
Acceptance Criteria	<ul style="list-style-type: none"> <li>● The system must provide a registration form with fields for username, email, and password.</li> <li>● The system must validate all inputs, including checking for valid email format and password strength.</li> <li>● If the provided data is valid and the email is not already registered, the system should create a new user account and store it in the database.</li> <li>● Upon successful registration, the system should display a success message and redirect the user to the login page.</li> <li>● If the email is already in use or the input data is invalid, the system should display an appropriate error message.</li> </ul>

b. Login

**Tabel 7.9 User Story Login**

User Story	<p><b>As a registered teacher</b></p> <p><b>I want to</b> log in with my username/email and password</p> <p><b>So that I</b> can access the application's main dashboard and have my activity history saved for future reference.</p>
Acceptance Criteria	<ul style="list-style-type: none"> <li>● Given the teacher has a registered account, when they enter valid credentials, then the system should grant access to the main dashboard.</li> </ul>

	<ul style="list-style-type: none"> <li>Given the teacher enters invalid credentials, when they attempt to log in, then the system should display an error message.</li> <li>Given the teacher successfully logs in, when they access features, then their activity should be recorded in their session history.</li> <li>Given the teacher has been inactive for a period (e.g., 30 minutes), when the session expires, then the system should log out automatically for security.</li> </ul>
--	---

c. Upload Questions

**Table 7.10 User Story Upload Question**

User Story	<p><b>As a teacher</b></p> <p><b>I want to</b> upload a file containing a set of questions in a supported format (CSV, PDF, DOCX, or TXT)</p> <p><b>So that</b> the system can validate and prepare it for automatic cognitive classification according to Bloom's Taxonomy.</p>
Acceptance Criteria	<ul style="list-style-type: none"> <li>Given the teacher selects a valid file (CSV, PDF, DOCX, or TXT), when they upload it, then the system should successfully read and validate the file.</li> <li>Given the file is not in a supported format, when the teacher uploads it, then the system should display a format error message.</li> <li>Given the file contains invalid or unreadable data, when the system processes it, then it should notify the teacher about the issue.</li> <li>Given the file passes validation, when processing completes, then the system should confirm that the questions are ready for classification.</li> </ul>

d. View Classification Results

**Tabel 7.11 User Story View Classification Results**

User Story	<p><b>As a teacher</b></p> <p><b>I want to</b> view the results of the classification process, which should be displayed as both a detailed table and a visual distribution graph</p> <p><b>So that</b> I can easily analyze the cognitive spread (C1-C6) across all my questions.</p>
Acceptance Criteria	<ul style="list-style-type: none"> <li>● Given the classification process has completed, when the teacher accesses the results page, then the system should display a detailed table of classified questions.</li> <li>● Given classification data exists, when the teacher views the results, then the system should also show a visual graph of C1–C6 distribution.</li> <li>● Given classification data is not available, when the teacher attempts to view results, then the system should display a “No data available” message.</li> <li>● Given the results are displayed, when the teacher hovers or clicks on a cognitive level, then additional details or question examples should be shown.</li> </ul>

e. Regenerate Question Based on Level

**Tabel 7.12 User Story Regenerate Question Based on Level**

User Story	<p><b>As a teacher</b></p> <p><b>I want to</b> select a question and choose a new target cognitive level (from C1-C6)</p> <p><b>So that</b> the system can regenerate the question's phrasing to match the new, desired level of complexity</p>
Acceptance Criteria	<ul style="list-style-type: none"> <li>● Given the teacher selects a question, when they choose a new cognitive level, then the system should process</li> </ul>

	<p>and generate a modified version of the question.</p> <ul style="list-style-type: none"> <li>Given the regeneration process is successful, when it completes, then the teacher should see both the original and regenerated questions for comparison.</li> <li>Given the regeneration fails (e.g., due to insufficient context), when it occurs, then the system should notify the teacher with an appropriate error message.</li> <li>Given the regenerated question is acceptable, when the teacher confirms, then the system should allow saving or exporting the new version.</li> </ul>
--	--

#### f. Download Report

**Tabel 7.13 User Story Download Report**

User Story	<p><b>As a teacher</b></p> <p><b>I want to</b> download the classification results and analysis in a single PDF report</p> <p><b>So that I</b> have a formal document for my records or for evaluation purposes.</p>
Acceptance Criteria	<ul style="list-style-type: none"> <li>Given the classification results are available, when the teacher clicks the “Download Report” button, then the system should generate a PDF file.</li> <li>Given the report generation completes successfully, when the download starts, then the PDF should include a summary table and visual graphs.</li> <li>Given the teacher has filtered data, when they download, then the report should only include the filtered subset.</li> <li>Given the report generation fails, when it happens, then the system should display an error and allow retrying.</li> </ul>

g. Logout

**Tabel 7.14 User Story Logout**

User Story	<p><b>As a teacher</b></p> <p><b>I want to</b> log out from my account when I am finished with my work</p> <p>So that I can securely end my session and ensure my account remains private.</p>
Acceptance Criteria	<ul style="list-style-type: none"><li>● Given the teacher is logged in, when they click the “Logout” button, then the system should terminate the current session.</li><li>● Given the logout is successful, when it completes, then the system should redirect the teacher to the login page.</li><li>● Given the teacher has unsaved actions, when they attempt to log out, then the system should show a confirmation prompt.</li><li>● Given the session has expired automatically, when the teacher interacts with the system, then they should be redirected to the login screen.</li></ul>

## 7.4 User Story Scenario

a. Sign up

**Tabel 7.15 User Story Scenario Sign Up**

Scenario	
Successful registration	Given I am on the registration page When I fill in "username" with "newteacher" And I fill in "email" with " <a href="mailto:teacher@example.com">teacher@example.com</a> " And I fill in "password" with "SecurePass123" And I press "Sign Up" Then I should be redirected to the login page And I should see "Registration successful" message
Registration fails due to already registered email	Given I am on the registration page And there is an existing user with email " <a href="mailto:teacher@example.com">teacher@example.com</a> " When I fill in "username" with "anotheruser"

	<p>When I fill in "email" with "teacher@example.com"      When I fill in "password" with "AnotherPass123"      When I press "Sign Up"      When I should see "Email already registered" error</p>
Registration fails due to invalid input	<p>Given I am on the registration page      When I fill in "username" with ""      And I fill in "email" with "not-an-email"      And I fill in "password" with "123"      And I press "Sign Up"      Then I should see "Invalid username/email/password" error</p>

### b. Login

**Tabel 7.16 User Story Scenario Login**

Scenario	
Login with valid credentials	<p>Given I am on the login page      And a user exists with email "x@gmail.com" and password "123"      When I fill in "email" with "x@gmail.com"      And I fill in "password" with "123"      And I press "Login"      Then I should be redirected to the main dashboard      And I should see "Welcome" message</p>
Login fails with invalid credentials	<p>Given I am on the login page      When I fill in "email" with "teacher@example.com"      And I fill in "password" with "WrongPassword"      And I press "Login"      Then I should see "Invalid credentials" error      And I should remain on the login page</p>

### c. Upload Question File

**Tabel 7.17 User Story Scenario Upload Question File**

Scenario	
Upload valid file	<p>Given I am logged in as "teacher@example.com"      And I am on the "Upload Questions" page      When I attach the file "questions.csv" to "file_upload"      And I press "Upload"      Then the system should accept the file      And I should see "File validated successfully" message</p>
Upload unsupported file format	<p>Given I am logged in as "teacher@example.com"      And I am on the "Upload Questions" page      When I attach the file "questions.exe" to "file_upload"      And I press "Upload"</p>

	Then I should see "Unsupported file format" error
Upload file with invalid content	<p>Given I am logged in as "teacher@example.com"</p> <p>And I am on the "Upload Questions" page</p> <p>When I attach the file "corrupt_questions.csv" to "file_upload"</p> <p>And I press "Upload"</p> <p>Then I should see "File content invalid or unreadable" error</p>

d. Regenerate Question to New Level

**Tabel 7.18 User Story Scenario Regenerate Question to New Level**

Scenario	
Successfully regenerate question	<p>Given I have classification results displayed</p> <p>And one question is labeled as "C3"</p> <p>When I choose that question and select new level "C5"</p> <p>And I press "Regenerate"</p> <p>Then I should see a regenerated version of the question</p> <p>And I should see both original and new versions</p>
Regeneration fails	<p>Given I have classification results displayed</p> <p>When I choose a question and select new level "C6"</p> <p>And the system is unable to generate a new phrasing</p> <p>Then I should see "Regeneration failed, try again" message</p>

e. Download Report

**Tabel 7.19 User Story Scenario Download Report**

Scenario	
Download full report	<p>Given classification results are available</p> <p>When I press "Download Report"</p> <p>Then the system should generate a PDF</p> <p>And the PDF should include summary table and charts</p>
Report generation fails	<p>Given classification results are available</p> <p>When I press "Download Report"</p> <p>And an internal error occurs</p> <p>Then I should see "Error generating report" message</p> <p>And I should be able to retry</p>

f. Logout

**Tabel 7.20 User Story Scenario Logout**

Scenario	
Manual logout	Given I am logged in When I press the "Logout" button Then my session should terminate And I should be redirected to the login page
Session expires due to inactivity	Given I have been inactive for defined timeout When I attempt to perform an action Then I should be redirected to the login page And see "Session expired, please log in again" message

## 7.5 GUI Design

### 7.5.1 Sign Up page

Sign up to Bloomers

Create your Bloomers Account

Join Bloomers and start transforming the way you evaluate questions with AI-powered analysis.

Username

Email

Password

Forgot Password?

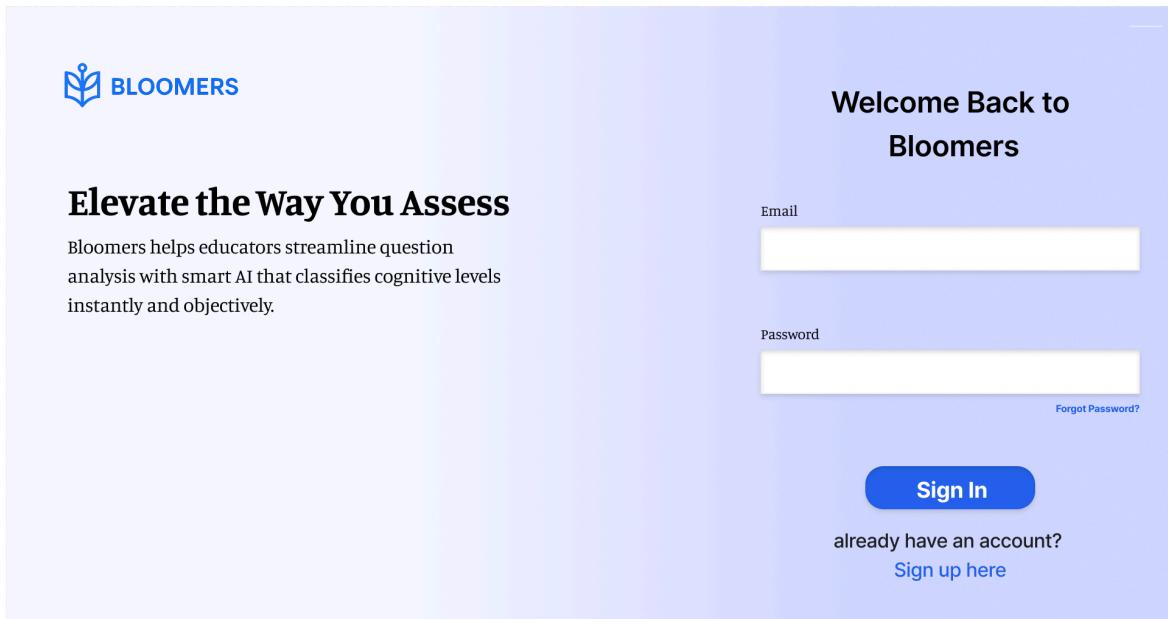
Sign up

already have an account?  
[Sign in here](#)

**Gambar 7.2 Sign Up Page**

Halaman Sign Up merupakan halaman tempat pengguna baru melakukan pendaftaran akun sebelum dapat mengakses fitur utama pada aplikasi Bloomers. Pada halaman ini tersedia form yang terdiri dari kolom Username, Email, dan Password yang harus diisi oleh pengguna. Terdapat pula tautan untuk mengarahkan pengguna ke halaman Sign In jika sudah memiliki akun, serta tombol Forgot Password untuk membantu pengguna yang lupa kata sandinya. Antarmuka halaman ini dirancang dengan tampilan sederhana dan bersih agar proses pendaftaran dapat dilakukan dengan cepat dan mudah dipahami.

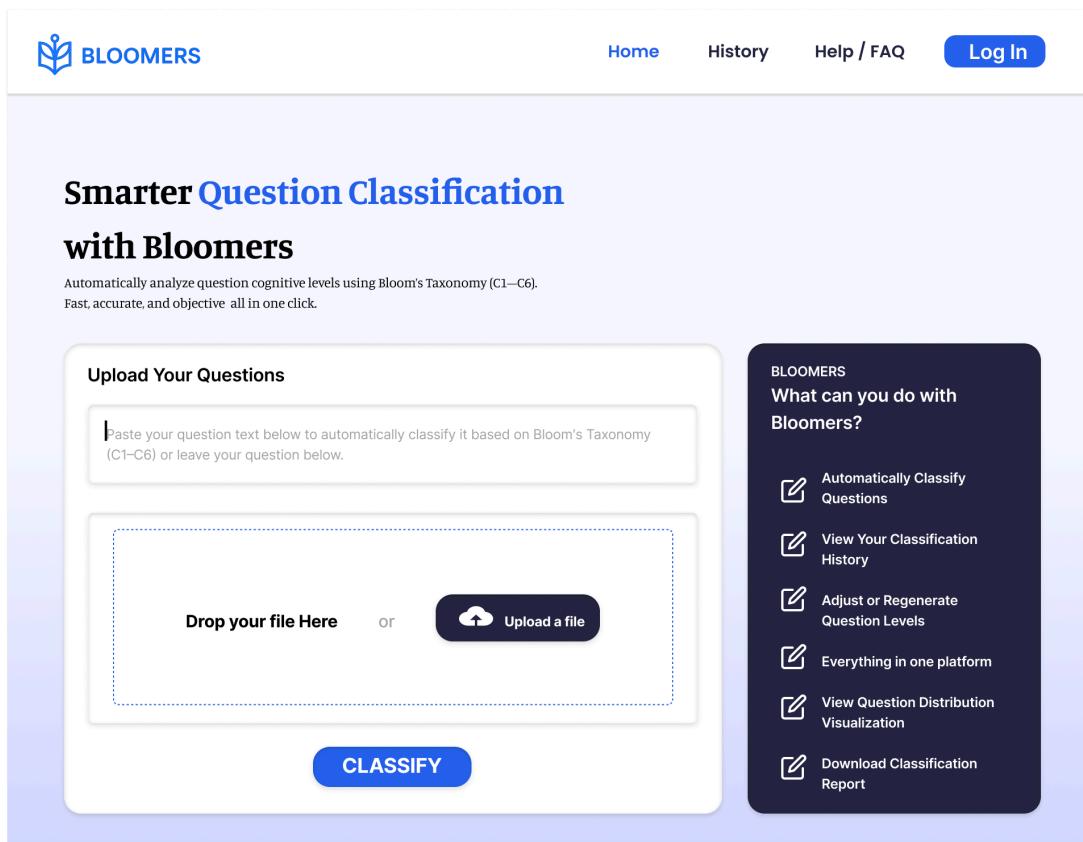
### 7.5.2 Sign In page



*Gambar 7.3 Sign In Page*

Halaman Sign In berfungsi sebagai pintu masuk bagi pengguna yang sudah memiliki akun. Pengguna perlu mengisi kolom Email atau Username dan Password, kemudian menekan tombol Login untuk masuk ke sistem. Setelah berhasil masuk, sistem akan menyimpan sesi pengguna sehingga dapat mengakses fitur-fitur personal seperti riwayat aktivitas, file yang telah diunggah, dan hasil klasifikasi sebelumnya. Desain halaman ini dibuat sederhana dan responsif, dengan fokus pada kemudahan akses serta navigasi yang jelas menuju halaman utama.

### 7.5.3 Page utama



**Gambar 7.4 Page Utama Upload**

Halaman utama merupakan pusat aktivitas pengguna untuk memulai proses klasifikasi soal. Di bagian tengah halaman terdapat area unggahan yang mendukung berbagai format file seperti PDF, DOCX, TXT, dan CSV. Pengguna dapat mengunggah file dengan menekan tombol Upload atau menggunakan metode drag and drop. Setelah file berhasil diunggah, sistem akan menampilkan informasi terkait file seperti nama file, ukuran, dan progres unggahan. Pengguna kemudian dapat menekan tombol Classify untuk memulai proses klasifikasi menggunakan model RoBERTa. Selain itu, disediakan pula tombol Delete untuk membatalkan atau mengganti file sebelum diproses. Antarmuka halaman ini menekankan kemudahan interaksi dan alur kerja yang jelas dari proses unggah hingga klasifikasi.

### 7.5.4 Page Review Soal

The screenshot shows a software interface for reviewing assignments. At the top, there's a logo for 'BLOOMERS' and a navigation bar with links for 'Home', 'History', 'Help / FAQ', and a user profile for 'Agim'. The main content area is titled 'SOAL LATIHAN UTS.pdf'. Below the title, there are two tabs: 'Question Detail' (which is selected) and 'Overview'. The 'Question Detail' tab contains three sections, each representing a question from the assignment:

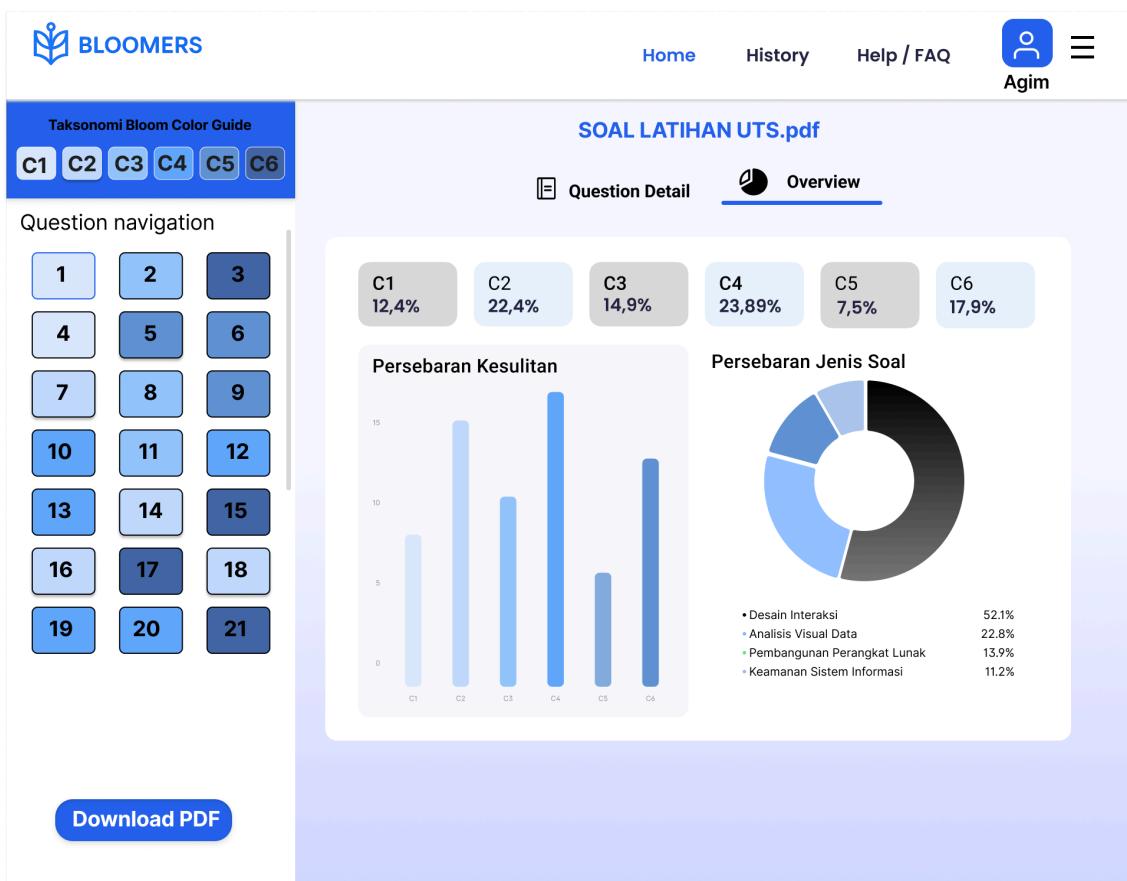
- Question 1:** The question text is: "Jelaskan secara singkat apa yang dimaksud dengan User Persona dalam proses desain interaksi." To the right is a 'Change Classification' dropdown menu set to 'Select', with options 'C2', 'C3', and 'C4' visible.
- Question 2:** The question text is: "HealthBridge merancang sistem notifikasi untuk pasien. Berdasarkan hasil wawancara, user di daerah 3T lebih familiar menggunakan WhatsApp daripada notifikasi aplikasi. Terapkan konsep User-Centered Design untuk menjelaskan mengapa WhatsApp lebih efektif digunakan pada sistem HealthBridge dibandingkan notifikasi aplikasi bawaan." To the right is a 'Change Classification' dropdown menu set to 'Select', with options 'C2', 'C3', and 'C4' visible.
- Question 3:** The question text is: "Dalam Feature Prioritization Matrix (MoSCoW), fitur Gamification untuk Donatur ditempatkan pada kategori Could Have. Menurut pendapatmu, apakah fitur ini sebaiknya tetap dipertahankan atau dihapus dari roadmap pengembangan? Berikan alasan evaluatif berdasarkan kebutuhan user dan tujuan utama platform." To the right is a 'Change Classification' dropdown menu set to 'Select', with options 'C2', 'C3', and 'C4' visible.

On the left side of the interface, there is a 'Question navigation' sidebar containing a 4x5 grid of numbered boxes from 1 to 21. A blue button at the bottom of this sidebar says 'Download PDF'.

**Gambar 7.5 Page Review Soal**

Halaman Review Soal menampilkan hasil klasifikasi setiap soal berdasarkan tingkatan Taksonomi Bloom (C1–C6). Setiap soal ditampilkan bersama label tingkat kognitifnya yang tertera di samping teks soal. Pengguna dapat meninjau hasil klasifikasi secara langsung dan memastikan kesesuaian antara isi soal dengan level Bloom yang terdeteksi. Di bagian kanan atas halaman terdapat tombol Download PDF yang memungkinkan pengguna mengunduh laporan hasil klasifikasi sebagai dokumentasi atau bahan evaluasi. Tampilan halaman ini dibuat dengan fokus pada keterbacaan dan kemudahan dalam melakukan verifikasi hasil klasifikasi.

### 7.5.5 Page overview soal



**Gambar 7.6 Page Overview Soal**

Halaman Overview Soal menyajikan ringkasan hasil klasifikasi dari satu file soal yang telah diunggah. Pada bagian kiri halaman, ditampilkan daftar seluruh soal beserta label level Bloom-nya (C1–C6). Sementara itu, bagian kanan menampilkan visualisasi hasil berupa diagram batang dan diagram lingkaran yang menggambarkan distribusi soal berdasarkan tingkat kognitif dan mata kuliah. Melalui tampilan ini, pengguna dapat dengan mudah menganalisis komposisi serta tingkat kesulitan soal secara keseluruhan. Di bagian bawah terdapat tombol Download Report untuk mengunduh hasil analisis dalam format PDF. Halaman ini dirancang dengan tampilan visual yang informatif dan membantu pengguna memahami persebaran soal secara menyeluruh.

### 7.5.6 Page Lihat History

#	File Name	Question	Date	Q1	Q2	Q3	Q4	Q5	Q6	Report	Action
1	Soal UTS.pdf	10	15/2/2024	2	2	1	3	2	0	<a href="#">Soal UTS.pdf</a>	
2	Soal apa aja.pdf	23	19/3/2024	5	7	3	5	1	2	<a href="#">Soal apa aja.pdf</a>	

*Gambar 7.7 Page Lihat History*

Halaman History berfungsi untuk menampilkan riwayat aktivitas pengguna selama menggunakan aplikasi Bloomers. Pada halaman ini, pengguna dapat melihat daftar file yang telah diunggah dan diklasifikasikan sebelumnya, lengkap dengan informasi seperti nama file, tanggal klasifikasi, jumlah soal, dan distribusi level Bloom (C1–C6). Selain itu, tersedia pula tombol aksi seperti Re-run, Download Report, dan Delete untuk memungkinkan pengguna mengulang proses klasifikasi, mengunduh kembali laporan, atau menghapus riwayat tertentu. Halaman ini mempermudah pengguna dalam melacak aktivitas yang telah dilakukan serta memantau perkembangan hasil klasifikasi dari waktu ke waktu.

### 7.6 Database

Table:

Server: localhost:3306 > Database: bloomers

Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers Designer

Filters Containing the word:

Table	Action	Rows	Type	Collation	Size	Overhead
file_soal	<span>Browse</span> <span>Structure</span> <span>Search</span> <span>Insert</span> <span>Empty</span> <span>Drop</span>	0	InnoDB	latin1_swedish_ci	32.0 KiB	-
laporan	<span>Browse</span> <span>Structure</span> <span>Search</span> <span>Insert</span> <span>Empty</span> <span>Drop</span>	0	InnoDB	latin1_swedish_ci	48.0 KiB	-
regenerate_soal	<span>Browse</span> <span>Structure</span> <span>Search</span> <span>Insert</span> <span>Empty</span> <span>Drop</span>	0	InnoDB	latin1_swedish_ci	48.0 KiB	-
review_history	<span>Browse</span> <span>Structure</span> <span>Search</span> <span>Insert</span> <span>Empty</span> <span>Drop</span>	0	InnoDB	latin1_swedish_ci	32.0 KiB	-
user	<span>Browse</span> <span>Structure</span> <span>Search</span> <span>Insert</span> <span>Empty</span> <span>Drop</span>	0	InnoDB	latin1_swedish_ci	32.0 KiB	-
5 tables	Sum	0	InnoDB	latin1_swedish_ci	192.0 KiB	0 B

Check all With selected:

Print Data dictionary

**Gambar 7.8 Table Database**

File\_Soal

MySQL returned an empty result set (i.e. zero rows).

```
SELECT * FROM `file_soal`
```

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ]

file_id	user_id	file_name	file_type
---------	---------	-----------	-----------

**Gambar 7.9 Database File Soal**

Laporan:

MySQL returned an empty result set (i.e. zero rows). (Query took 0.000 sec)

```
SELECT * FROM `laporan`
```

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [

laporan_id	user_id	review_id	teks_laporan
------------	---------	-----------	--------------

**Gambar 7.10 Database Laporan**

Regenerate\_soal:

MySQL returned an empty result set (i.e. zero rows). (Query took 0.0004 seconds.)

```
SELECT * FROM `regenerate_soal`
```

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP class ]

regen_id	laporan_id	file_id	level_awal	level_target
----------	------------	---------	------------	--------------

*Gambar 7.11 Database Regenerate Soal*

Review\_history:

MySQL returned an empty result set (i.e. zero rows). (Query took 0.0004 seconds.)

```
SELECT * FROM `review_history`
```

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ]

review_id	user_id	tanggal_review
-----------	---------	----------------

*Gambar 7.12 Database Review History*

User:

MySQL returned an empty result set (i.e. zero rows). (Query took 0.0004 seconds.)

```
SELECT * FROM `user`
```

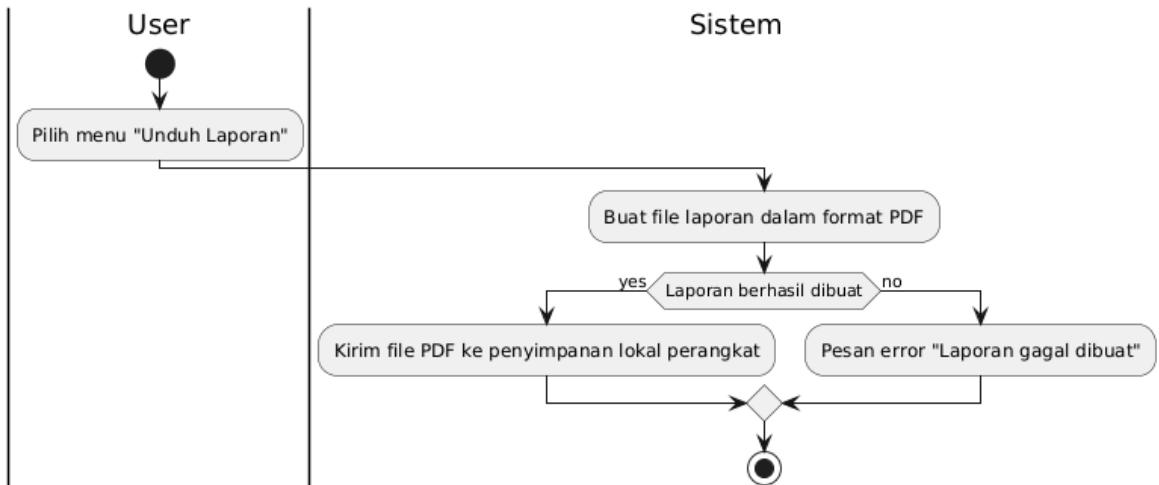
Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP class ]

user_id	username	email	password
---------	----------	-------	----------

*Gambar 7.13 Database User*

## 7.7 Activity Diagram

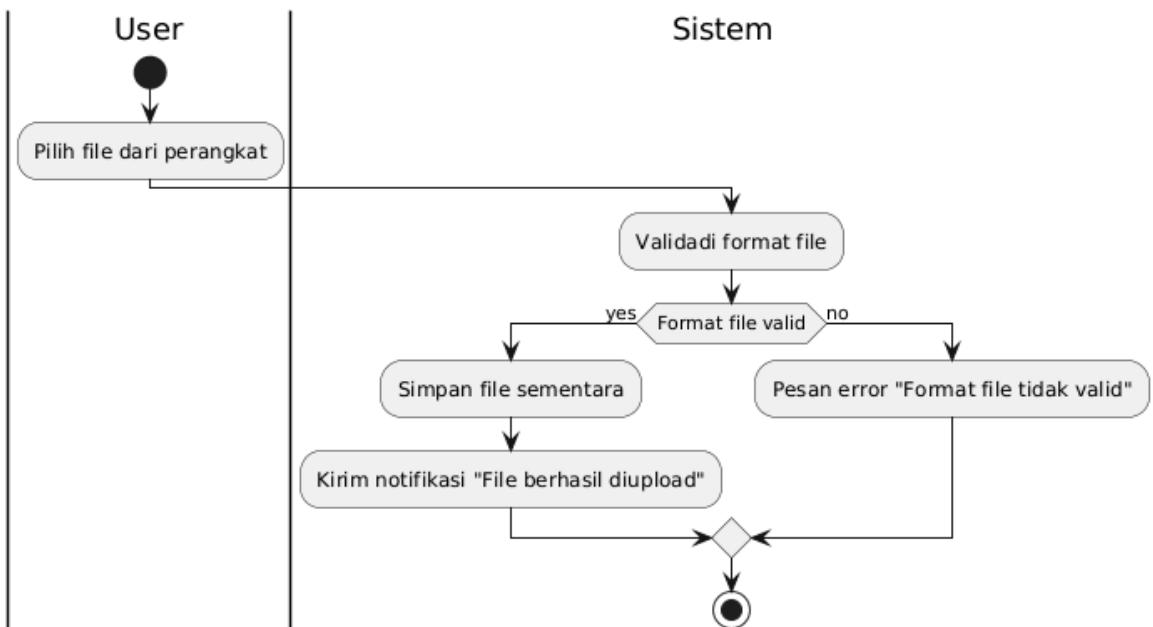
### 7.7.1 Activity Diagram Mengunduh Laporan



Gambar 7.14 Activity Diagram Mengunduh Laporan

Diagram aktivitas tersebut menggambarkan proses pengunduhan laporan oleh pengguna. Alur dimulai ketika User memilih menu “Unduh Laporan”, kemudian Sistem memproses permintaan dengan membuat file laporan dalam format PDF. Setelah proses pembuatan selesai, sistem akan melakukan pengecekan apakah laporan berhasil dibuat. Jika berhasil, sistem akan mengirim file PDF ke penyimpanan lokal perangkat pengguna. Namun, jika gagal, sistem akan menampilkan pesan kesalahan “Laporan gagal dibuat”. Proses ini memastikan pengguna mendapatkan laporan dalam format yang diinginkan atau diberi notifikasi apabila terjadi kegagalan.

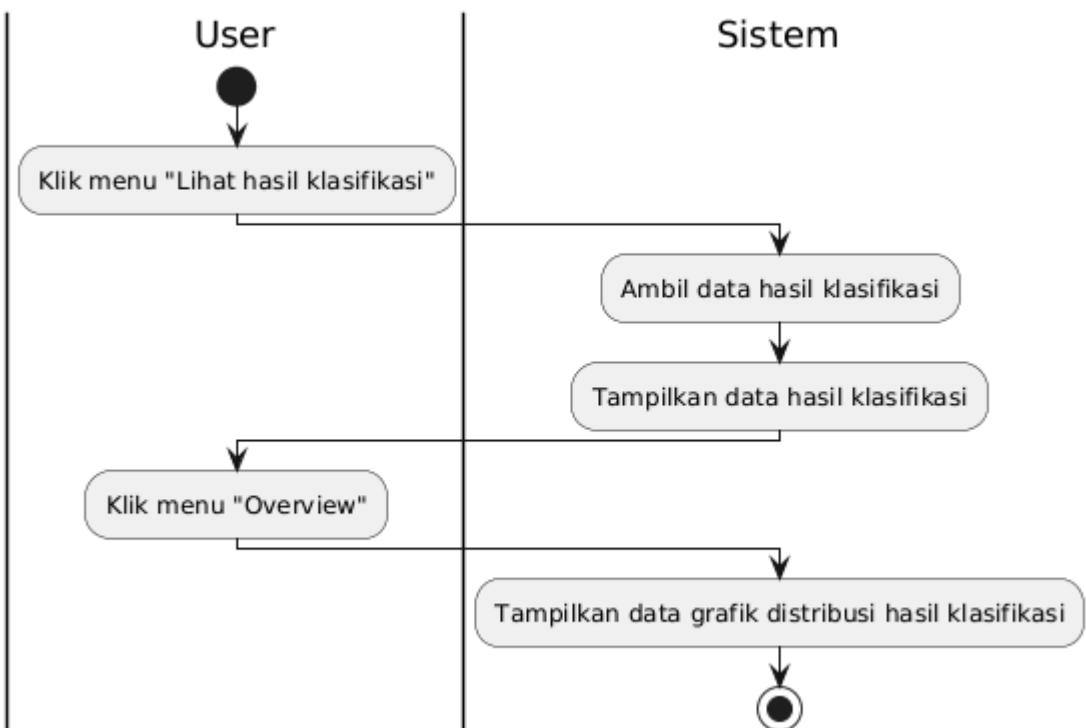
### 7.7.2 Activity Diagram Mengunggah Laporan



**Gambar 7.14 Activity Diagram Mengunggah Laporan**

Diagram aktivitas tersebut menggambarkan proses unggah file oleh pengguna. Alur dimulai ketika User memilih file dari perangkat, kemudian Sistem melakukan validasi terhadap format file yang diunggah. Jika format file valid, sistem akan menyimpan file tersebut secara sementara dan mengirimkan notifikasi bahwa file berhasil diunggah. Namun, jika format file tidak valid, sistem akan menampilkan pesan error “Format file tidak valid”. Proses ini memastikan bahwa hanya file dengan format yang sesuai standar sistem yang dapat diunggah.

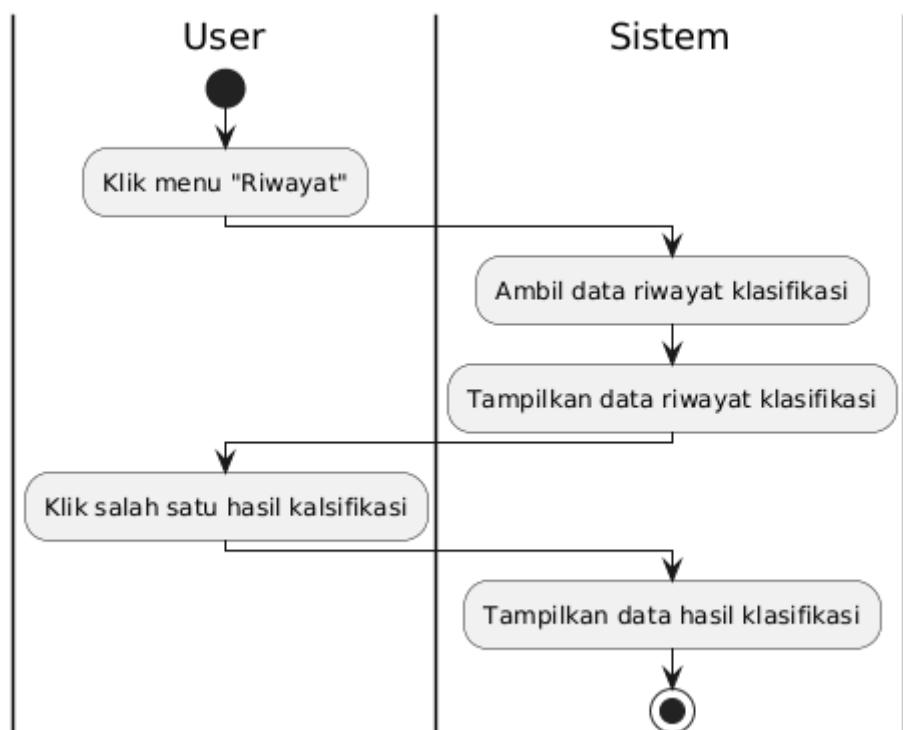
### 7.7.3 Activity Diagram Melihat Hasil Klasifikasi



**Gambar 7.15 Activity Diagram Melihat Hasil Klasifikasi**

Diagram aktivitas tersebut menggambarkan proses penayangan hasil klasifikasi data oleh pengguna. Proses dimulai ketika User mengklik menu “Lihat hasil klasifikasi”, kemudian Sistem akan mengambil data hasil klasifikasi dan menampilkannya kepada pengguna. Setelah itu, pengguna dapat memilih menu “Overview” untuk melihat ringkasan visual, dan sistem akan menampilkan grafik distribusi dari hasil klasifikasi. Alur ini memudahkan pengguna untuk memahami hasil analisis secara detail maupun dalam bentuk visual yang lebih informatif.

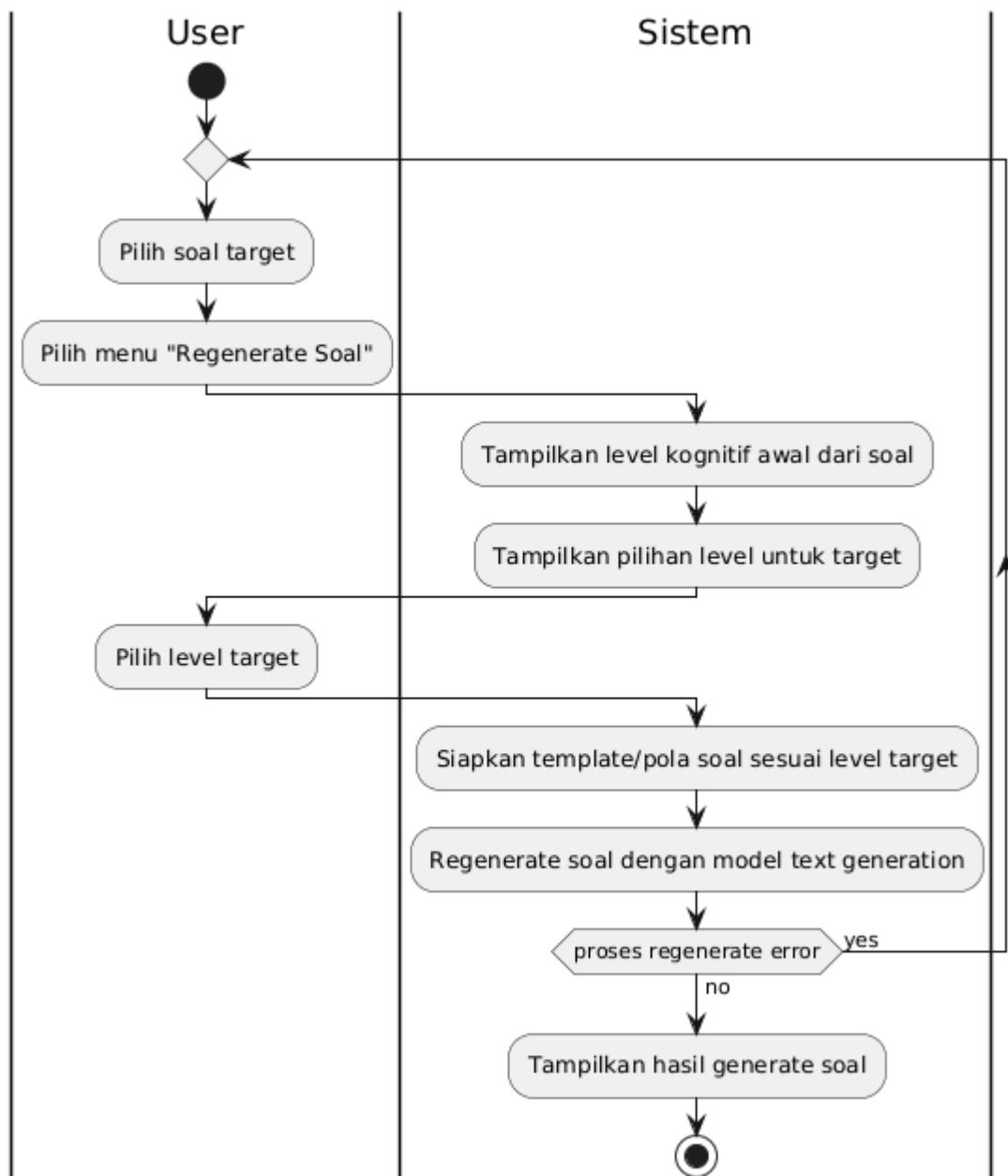
#### 7.7.4 Activity Diagram Melihat History Klasifikasi



**Gambar 7.16 Activity Diagram Melihat History Klasifikasi**

Diagram aktivitas tersebut menggambarkan proses melihat history hasil klasifikasi oleh pengguna. Alur dimulai ketika User mengklik menu “history”, kemudian Sistem akan mengambil data history hasil klasifikasi dan menampilkannya dalam daftar kepada pengguna. Setelah itu, pengguna dapat memilih salah satu hasil klasifikasi yang ingin dilihat lebih detail, dan sistem akan menampilkan data hasil klasifikasi tersebut secara lengkap. Proses ini memungkinkan pengguna untuk meninjau kembali hasil klasifikasi sebelumnya dengan mudah dan terstruktur.

### 7.7.5 Activity Diagram Regenerate Soal Based on Level

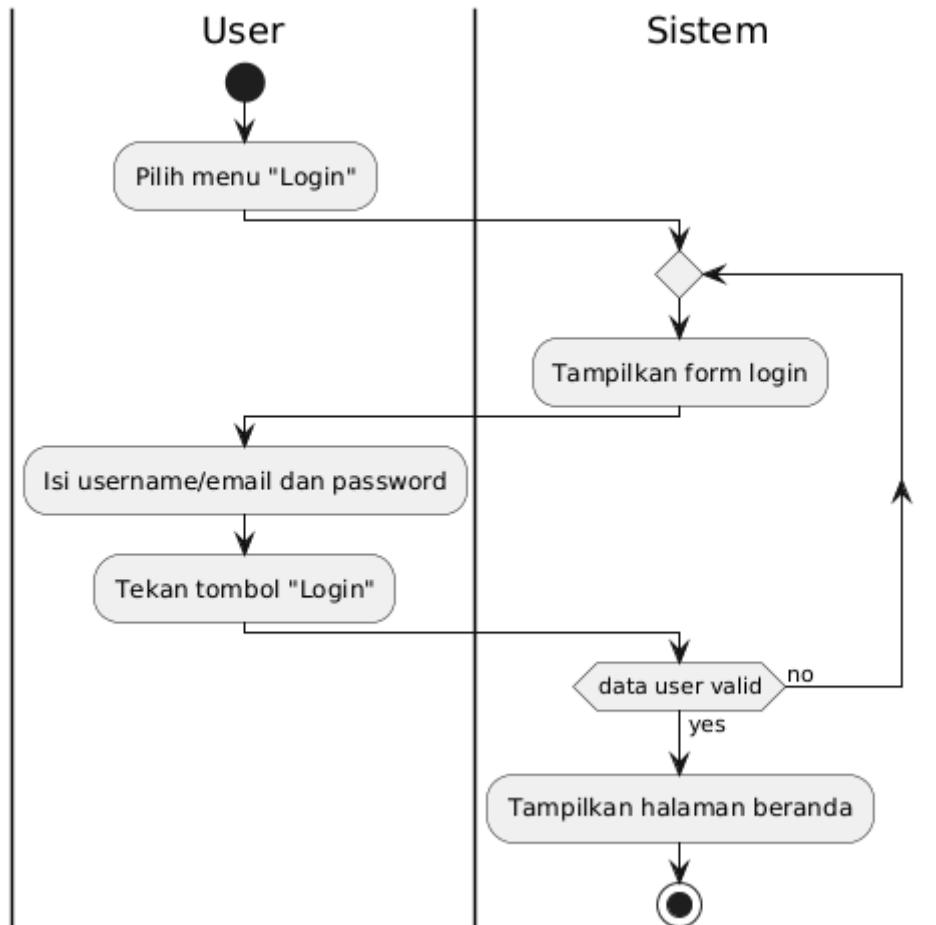


**Gambar 7.17 Activity Diagram Regenerate Soal Based on Level**

Diagram aktivitas tersebut menggambarkan proses regenerasi soal oleh pengguna. Alur dimulai ketika User memilih soal target kemudian mengklik menu “Regenerate Soal”. Selanjutnya, Sistem akan menampilkan level kognitif awal dari soal serta daftar pilihan level target yang dapat dipilih pengguna. Setelah pengguna menentukan level target, sistem akan menyiapkan template atau pola soal yang sesuai dan melakukan proses regenerasi soal menggunakan model text generation. Jika proses regenerasi mengalami error, sistem akan mengulang prosesnya; namun jika

berhasil, sistem akan menampilkan hasil generate soal baru kepada pengguna. Proses ini memungkinkan penyesuaian tingkat kesulitan soal secara otomatis dan efisien.

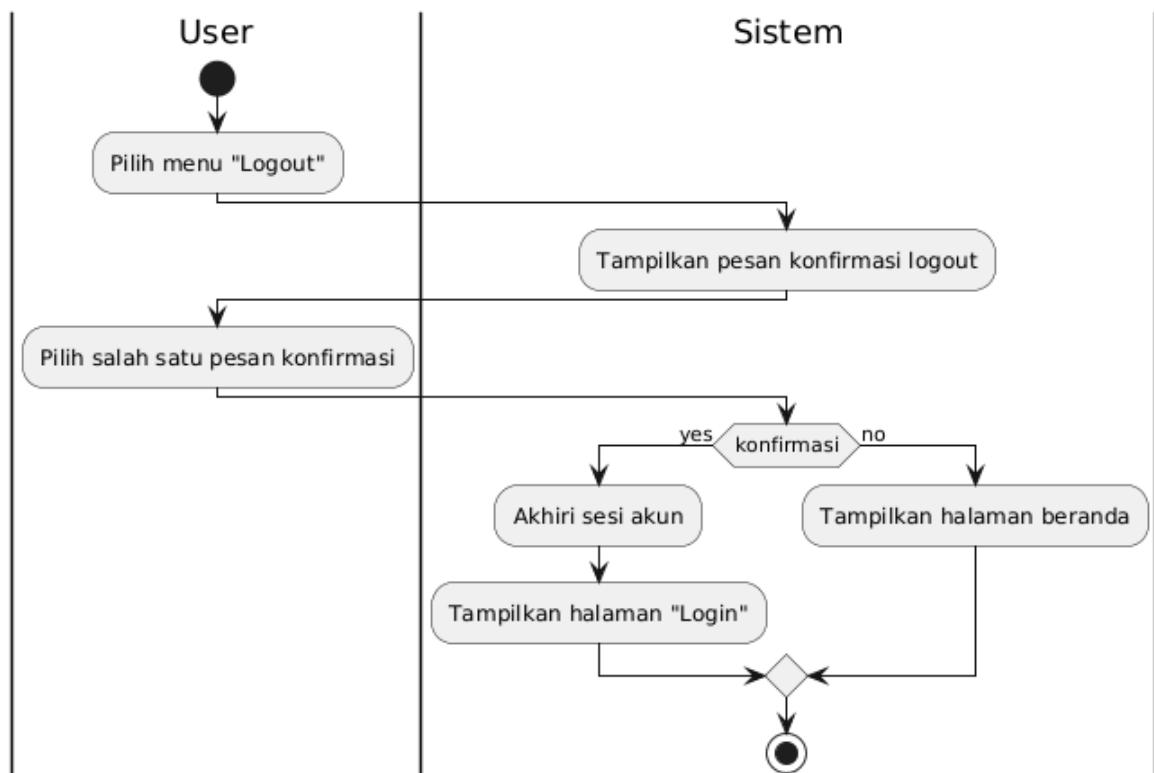
#### 7.7.6 Activity Diagram Login



*Gambar 7.18 Activity Diagram Login*

Diagram aktivitas tersebut menggambarkan proses login pengguna ke dalam sistem. Proses dimulai ketika User memilih menu “Login”, kemudian Sistem akan menampilkan form login. Pengguna lalu mengisi username atau email serta password, dan menekan tombol “Login” untuk mengirimkan data autentikasi. Sistem kemudian memvalidasi data pengguna; apabila data tidak valid, sistem akan menampilkan kembali form login untuk percobaan ulang. Jika data valid, sistem akan mengalihkan pengguna ke halaman beranda. Alur ini memastikan hanya pengguna dengan kredensial yang benar dapat mengakses sistem.

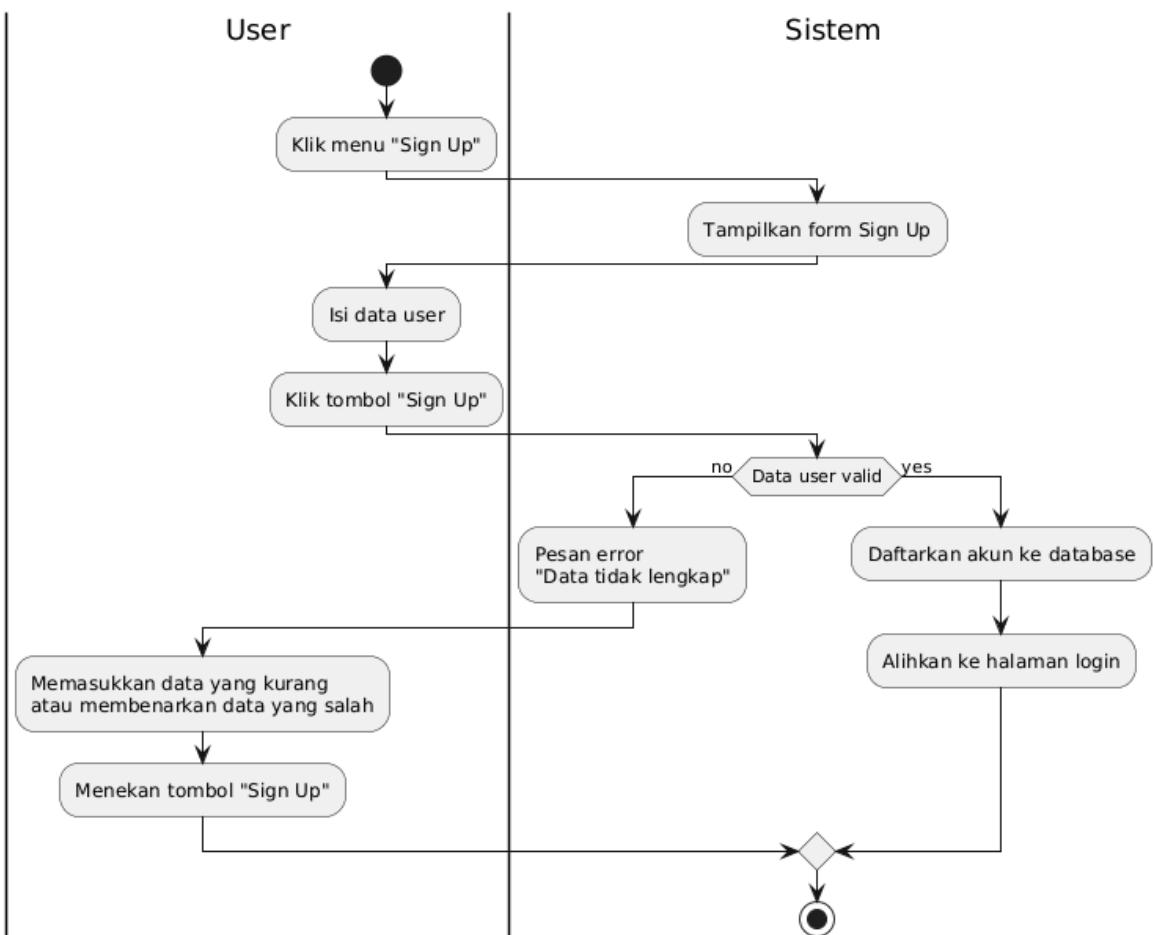
### 7.7.7 Activity Diagram Logout



**Gambar 7.19 Activity Diagram Logout**

Gambar tersebut merupakan activity diagram proses logout pada sistem. Proses dimulai ketika user memilih menu “Logout”, kemudian sistem menampilkan pesan konfirmasi logout untuk memastikan keputusan pengguna. Setelah itu, user memilih salah satu opsi konfirmasi, yaitu “ya” atau “tidak”. Jika user memilih “ya”, maka sistem mengakhiri sesi akun dan menampilkan halaman login. Sebaliknya, jika user memilih “tidak”, maka sistem akan menampilkan kembali halaman beranda. Proses berakhir setelah salah satu alur tersebut selesai dijalankan.

### 7.7.8 Activity Diagram Sign Up

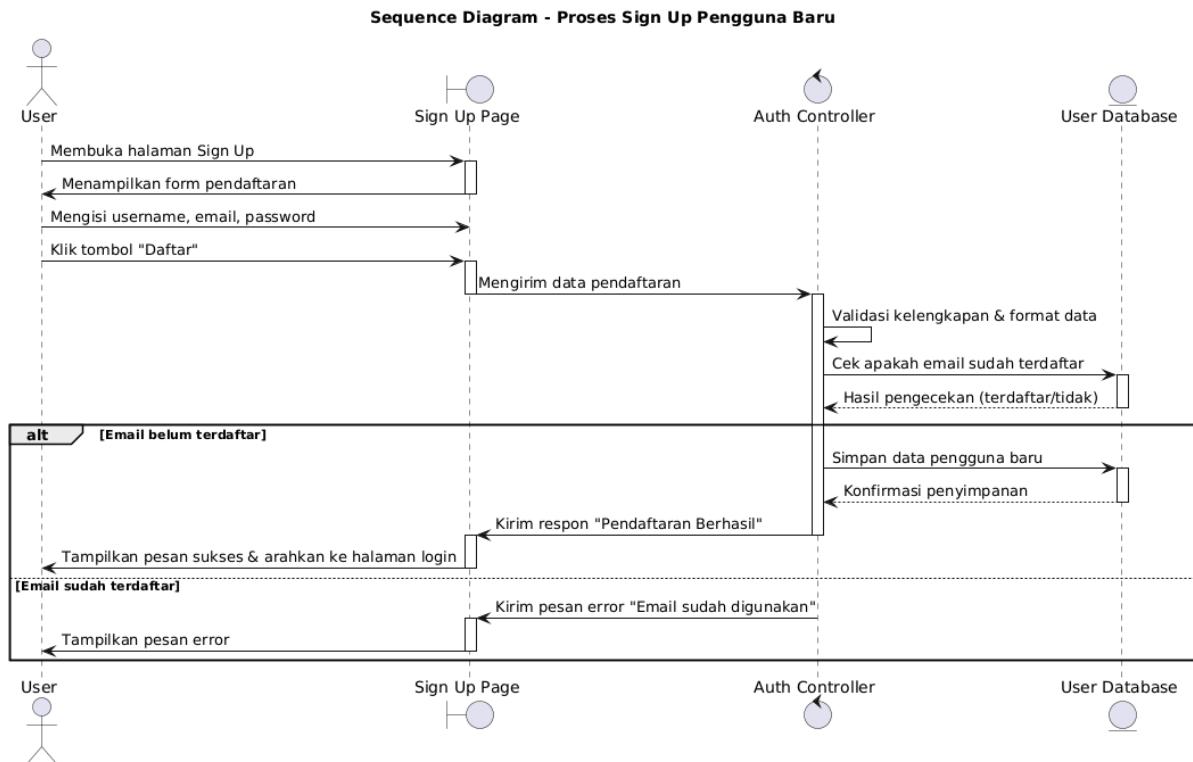


**Gambar 7.20 Activity Diagram Sign Up**

Diagram aktivitas tersebut menggambarkan proses pendaftaran akun baru (Sign Up) oleh pengguna. Proses dimulai ketika User mengklik menu “Sign Up”, kemudian Sistem akan menampilkan form pendaftaran. Pengguna selanjutnya mengisi data diri dan menekan tombol “Sign Up” untuk mengirimkan data tersebut. Sistem kemudian memvalidasi data pengguna; jika data tidak lengkap, sistem menampilkan pesan error “Data tidak lengkap”, dan pengguna dapat memperbaiki atau melengkapi data sebelum mengirim ulang. Jika data dinyatakan valid, sistem akan mendaftarkan akun ke dalam database dan mengalihkan pengguna ke halaman login. Alur ini memastikan bahwa proses pendaftaran berjalan dengan benar dan data pengguna tersimpan secara valid.

## 7.8 Sequence Diagram

### 7.8.1 Sequence Diagram Sign Up

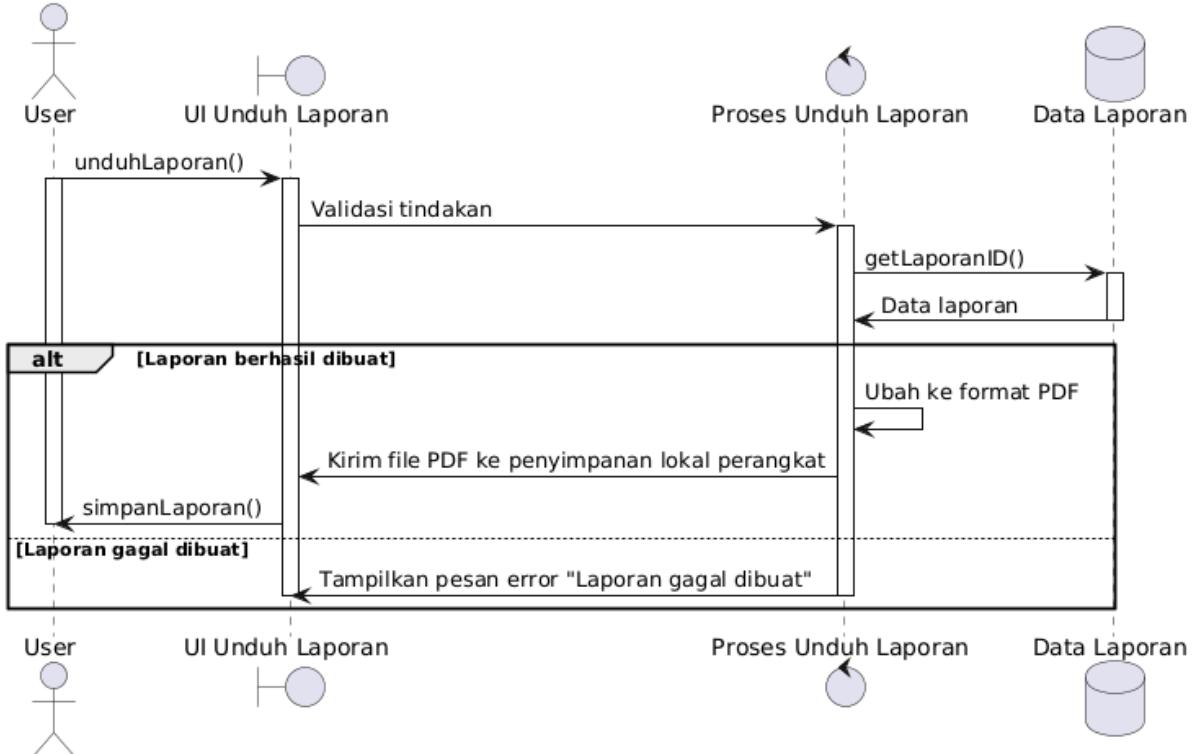


Gambar 7.21 Sequence Diagram Sign Up

Sequence Diagram “Proses Sign Up Pengguna Baru” menggambarkan alur interaksi antara User, halaman Sign Up, Auth Controller, dan User Database saat pengguna mendaftar akun baru di aplikasi Bloomers.

Proses dimulai ketika User membuka halaman Sign Up dan sistem menampilkan formulir pendaftaran. Pengguna kemudian mengisi data (username, email, dan password) dan menekan tombol “Daftar”. Halaman Sign Up mengirimkan data tersebut ke Auth Controller, yang bertugas memvalidasi kelengkapan serta format data. Setelah validasi, controller melakukan pengecekan ke database untuk memastikan apakah email sudah terdaftar. Jika email belum terdaftar, sistem akan menyimpan data pengguna baru ke database, mengirimkan respon “Pendaftaran Berhasil”, dan mengalihkan user ke halaman login. Namun, jika email sudah digunakan, controller mengirimkan pesan error ke halaman Sign Up untuk ditampilkan ke pengguna.

### 7.8.2 Sequence Diagram Mengunduh Laporan



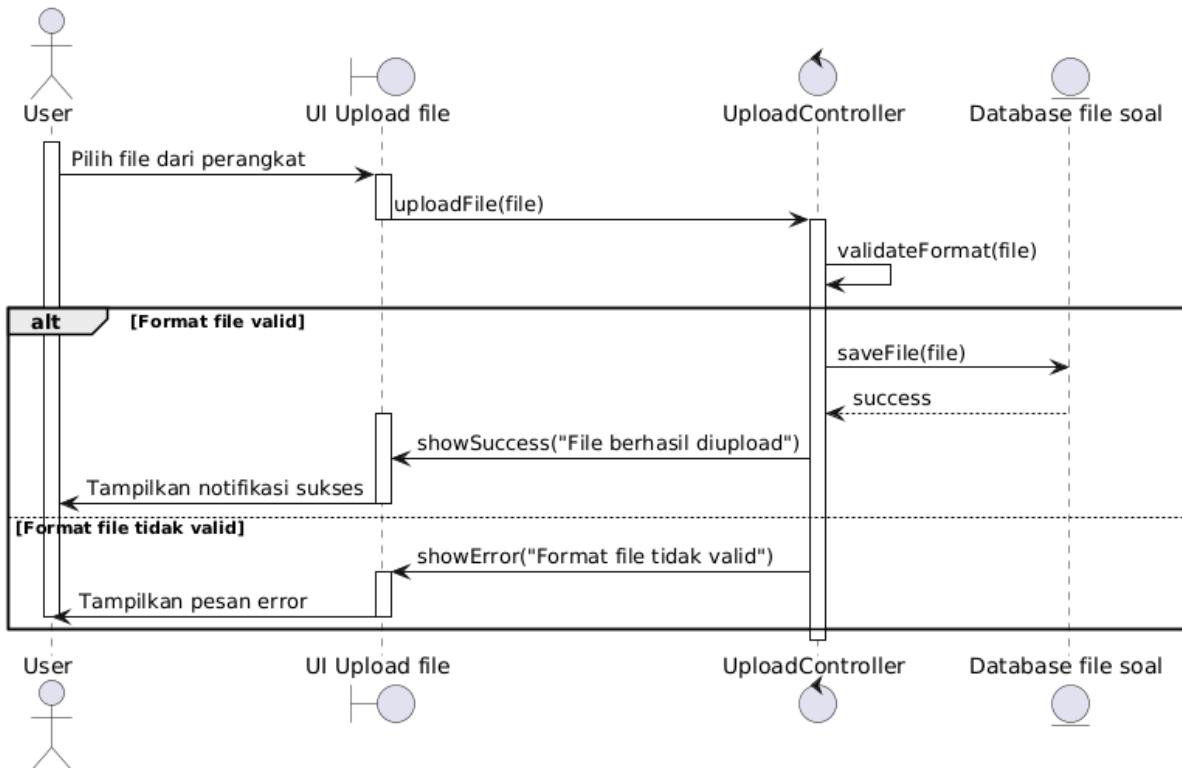
**Gambar 7.22 Sequence Diagram Mengunduh Laporan**

Sequence diagram tersebut menggambarkan alur proses unggah file soal pada aplikasi Bloomers, mulai dari interaksi pengguna hingga penyimpanan file ke database.

Proses dimulai ketika User memilih file dari perangkat melalui antarmuka UI Upload File. Setelah itu, UI mengirimkan file yang dipilih ke UploadController untuk diproses. Di dalam controller, sistem melakukan validasi format file guna memastikan bahwa file yang diunggah sesuai dengan format yang didukung (misalnya PDF, DOCX, TXT, atau CSV).

Jika format file valid, maka controller akan menyimpan file ke dalam Database file soal dan mengirimkan konfirmasi berhasil. UI kemudian menampilkan notifikasi sukses kepada pengguna bahwa file berhasil diunggah. Namun, jika format file tidak valid, controller akan mengirimkan pesan error ke UI yang kemudian ditampilkan kepada pengguna.

### 7.8.3 Sequence Diagram Mengunggah Laporan



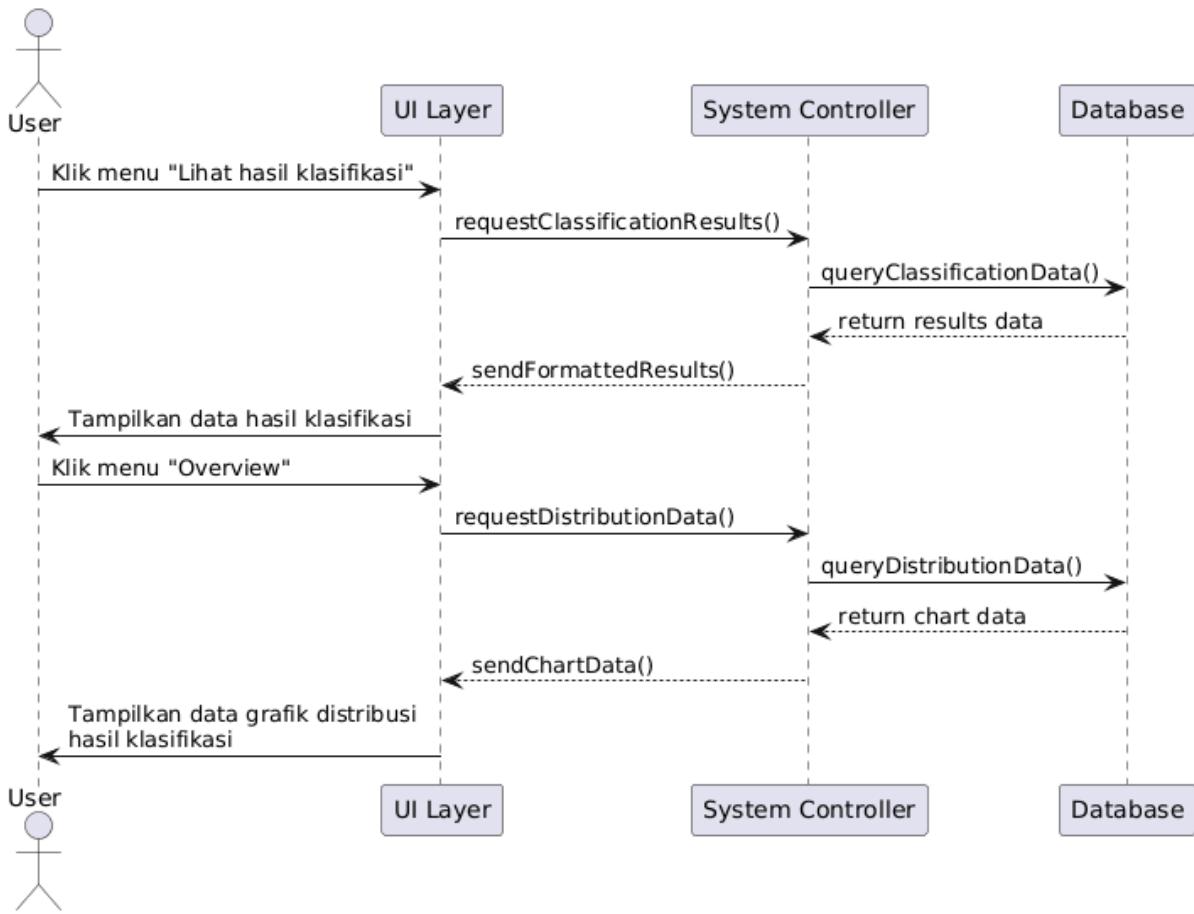
**Gambar 7.23 Sequence Diagram Mengunggah Laporan**

Sequence diagram tersebut menggambarkan proses unggah file soal oleh pengguna pada aplikasi Bloomers yang melibatkan empat komponen utama: User, UI Upload File, UploadController, dan Database File Soal.

Proses dimulai ketika User memilih file dari perangkat melalui antarmuka UI Upload File. Selanjutnya, UI mengirimkan file tersebut ke UploadController melalui fungsi uploadFile(file) untuk diproses lebih lanjut. Controller kemudian melakukan validasi format file menggunakan fungsi validateFormat(file) guna memastikan bahwa file yang diunggah sesuai dengan format yang diperbolehkan sistem (seperti CSV, PDF, DOCX, atau TXT).

Jika format file valid, maka UploadController menyimpan file ke dalam Database File Soal dengan perintah saveFile(file). Setelah penyimpanan berhasil, controller mengirimkan respon showSuccess("File berhasil diupload") ke UI untuk ditampilkan kepada pengguna sebagai notifikasi sukses. Namun, jika format file tidak valid, controller mengirimkan pesan showError("Format file tidak valid") ke UI, dan sistem akan menampilkan pesan error agar pengguna mengetahui kesalahan format.

#### 7.8.4 Sequence Diagram Melihat Hasil Klasifikasi



*Gambar 7.24 Sequence Diagram Melihat Hasil Klasifikasi*

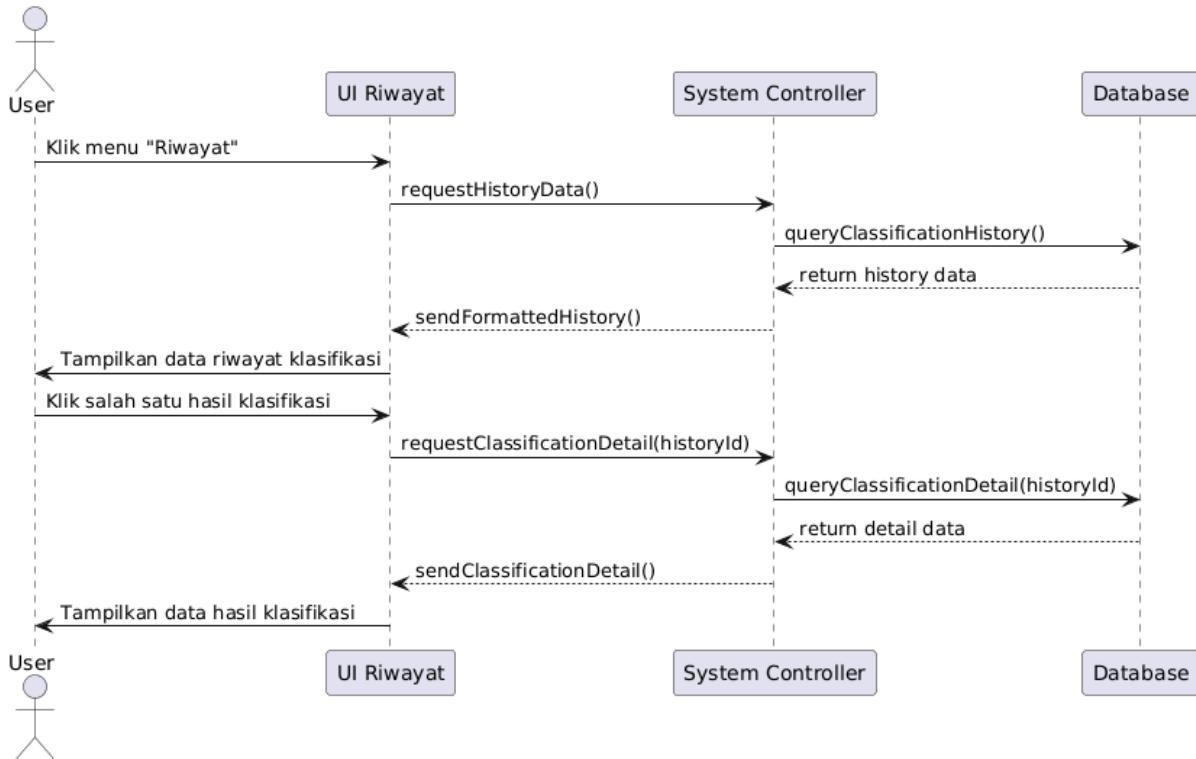
Sequence diagram tersebut menggambarkan proses penampilan hasil klasifikasi dan overview distribusi soal pada aplikasi Bloomers. Diagram ini menunjukkan bagaimana interaksi antara User, UI Layer, System Controller, dan Database berlangsung secara berurutan.

Proses pertama dimulai ketika User mengklik menu “Lihat hasil klasifikasi”. UI Layer kemudian mengirimkan permintaan `requestClassificationResults()` ke System Controller, yang akan meneruskan permintaan ke Database melalui `queryClassificationData()`. Setelah database mengembalikan data hasil klasifikasi, controller memformat data tersebut dan mengirimkannya kembali ke UI dalam bentuk yang siap ditampilkan. UI kemudian menampilkan hasil klasifikasi soal berdasarkan tingkat kognitif Bloom (C1–C6) kepada pengguna.

Selanjutnya, User mengklik menu “Overview” untuk melihat visualisasi distribusi hasil klasifikasi. Prosesnya serupa: UI mengirimkan `requestDistributionData()` ke controller, lalu controller mengambil data grafik dari

database melalui `queryDistributionData()`. Setelah data berhasil dikembalikan, controller mengirimkan hasilnya ke UI menggunakan `sendChartData()`. UI kemudian menampilkan grafik distribusi klasifikasi soal, yang memberikan gambaran persentase atau proporsi soal pada tiap level Taksonomi Bloom.

### 7.8.5 Sequence Diagram Melihat History Klasifikasi



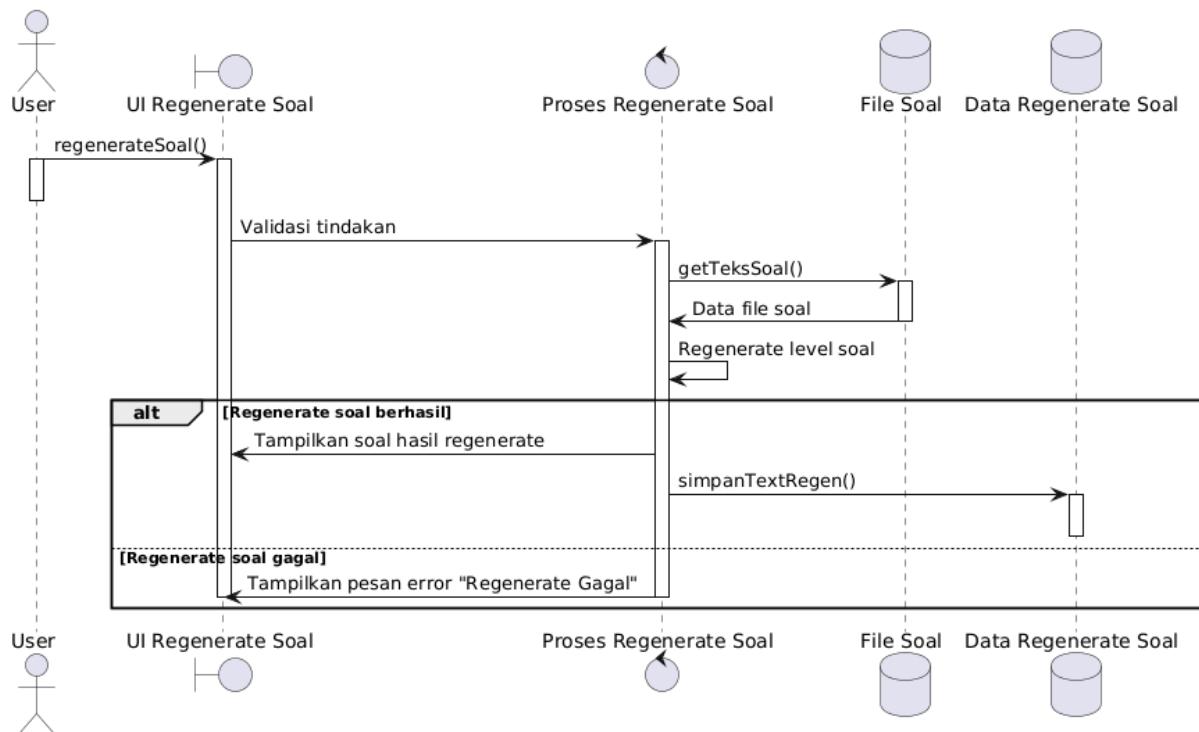
**Gambar 7.25 Sequence Diagram Melihat History Klasifikasi**

Sequence diagram tersebut menggambarkan proses penampilan history dan detail hasil klasifikasi soal pada aplikasi Bloomers. Diagram ini melibatkan empat komponen utama yaitu User, UI Layer, System Controller, dan Database yang saling berinteraksi untuk menampilkan data klasifikasi yang pernah dilakukan sebelumnya.

Proses dimulai ketika User mengklik menu “history”. UI Layer kemudian mengirimkan permintaan `requestHistoryData()` ke System Controller, yang akan meneruskan permintaan tersebut ke Database melalui `queryClassificationHistory()`. Database kemudian mengembalikan data history klasifikasi yang berisi daftar hasil klasifikasi sebelumnya. Controller memformat data tersebut dan mengirimkannya kembali ke UI dengan `sendFormattedHistory()`. Selanjutnya, UI menampilkan daftar history klasifikasi kepada pengguna.

Pada tahap berikutnya, User memilih salah satu hasil klasifikasi dari daftar history. UI mengirimkan permintaan `requestClassificationDetail(historyId)` ke controller untuk mengambil data detail klasifikasi berdasarkan ID history yang dipilih. Controller lalu melakukan query ke database menggunakan `queryClassificationDetail(historyId)`. Setelah database mengembalikan data detail, controller mengirimkannya ke UI dengan `sendClassificationDetail()`, dan UI menampilkan detail hasil klasifikasi soal kepada pengguna.

#### 7.8.6 Sequence Diagram Regenerate Soal Based on Level



**Gambar 7.26 Sequence Diagram Regenerate Soal Based on Level**

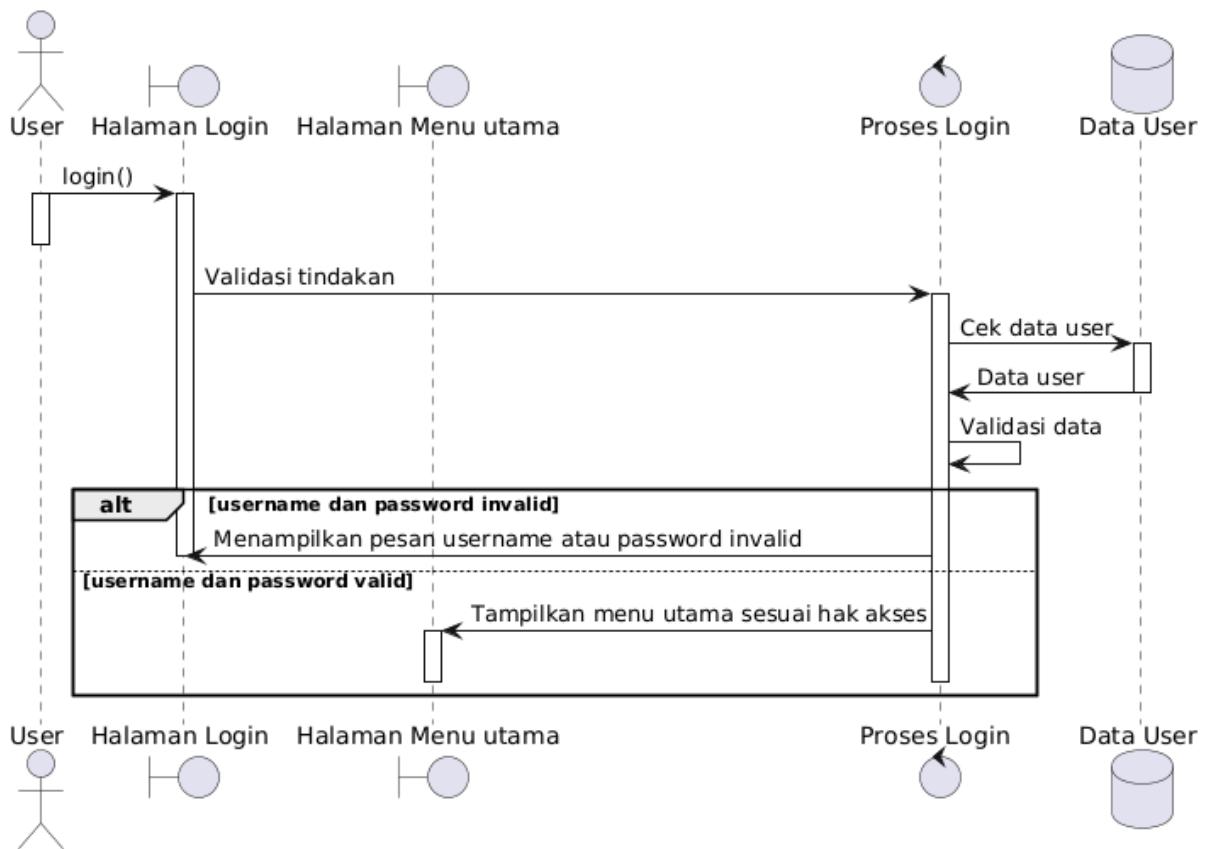
Sequence diagram tersebut menggambarkan proses regenerasi soal berdasarkan level Taksonomi Bloom pada aplikasi Bloomers. Diagram ini melibatkan empat komponen utama yaitu User, UI Regenerate Soal, Proses Regenerate Soal (controller), serta dua basis data: File Soal dan Data Regenerate Soal.

Proses dimulai ketika User melakukan aksi `regenerateSoal()` pada antarmuka UI Regenerate Soal. Setelah menerima perintah dari pengguna, UI melakukan validasi tindakan dengan mengirimkan permintaan ke Proses Regenerate Soal. Controller kemudian mengakses database File Soal melalui fungsi `getTeksSoal()` untuk mengambil teks soal yang akan diregenerasi. Setelah data soal diperoleh, sistem

melakukan proses regenerasi level soal, yaitu mengubah redaksi atau kompleksitas soal agar sesuai dengan level kognitif baru yang dipilih (misalnya dari C2 ke C4).

Apabila proses regenerasi berhasil, controller mengirimkan hasil ke UI untuk menampilkan soal hasil regenerasi kepada pengguna, dan sistem juga menyimpan hasil tersebut ke database Data Regenerate Soal menggunakan fungsi simpanTextRegen(). Namun, jika proses regenerasi gagal, sistem akan mengirimkan pesan error ke UI dengan notifikasi “Regenerate Gagal” agar pengguna mengetahui bahwa proses tidak berhasil dijalankan.

#### 7.8.7 Sequence Diagram Login

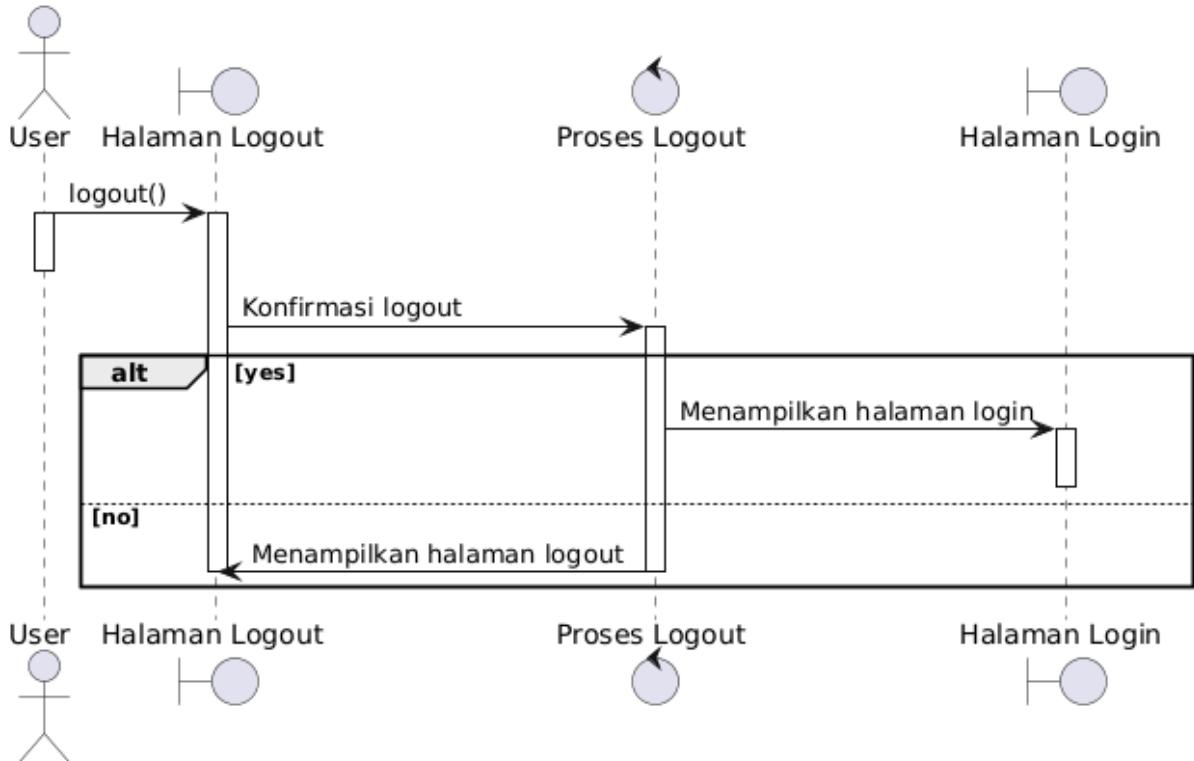


**Gambar 7.27 Sequence Diagram Login**

Gambar tersebut merupakan sequence diagram proses login pada sistem. Proses diawali ketika user melakukan login melalui halaman login dengan memasukkan username dan password. Sistem kemudian melakukan validasi tindakan dan pengecekan data user ke dalam basis data untuk memastikan kecocokan data. Jika username atau password tidak valid, sistem akan menampilkan pesan kesalahan kepada pengguna. Namun, jika data login valid, maka sistem akan menampilkan

menu utama sesuai dengan hak akses user. Proses ini memastikan bahwa hanya pengguna yang terautentikasi dengan benar yang dapat mengakses sistem.

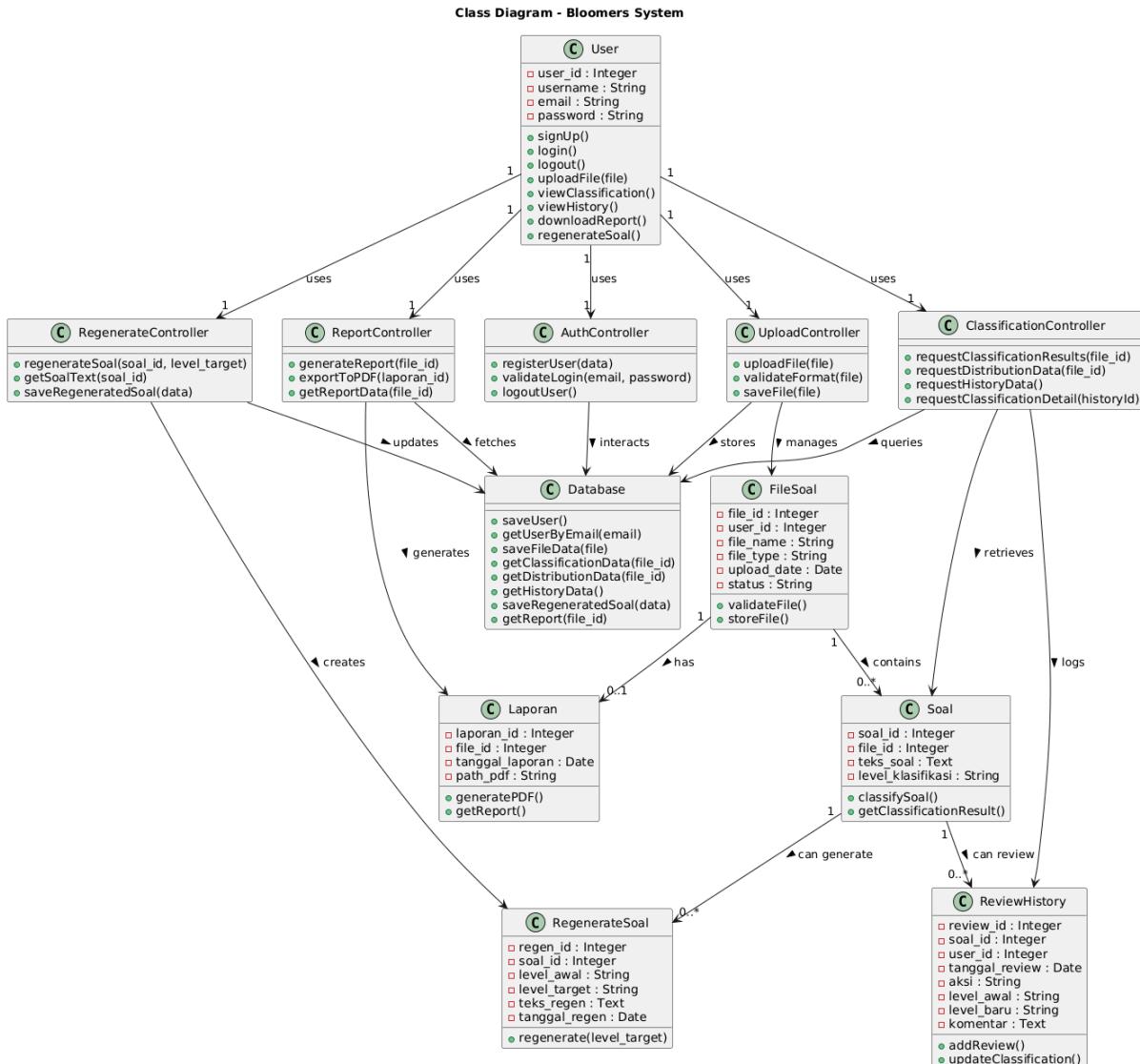
#### 7.8.7 Sequence Diagram Logout



**Gambar 7.28 Sequence Diagram Logout**

Gambar tersebut merupakan sequence diagram proses logout pada sistem. Proses dimulai ketika user melakukan aksi logout dari halaman logout. Sistem kemudian melakukan konfirmasi logout kepada user. Jika user memilih “yes”, maka sistem akan mengakhiri sesi dan menampilkan halaman login. Namun, jika user memilih “no”, maka sistem akan tetap menampilkan halaman logout tanpa melakukan perubahan. Diagram ini menggambarkan alur komunikasi antara user dan sistem untuk memastikan bahwa proses logout dilakukan secara aman dan sesuai keputusan pengguna.

## 7.9 Class Diagram



**Gambar 7.29 Class Diagram**

Class diagram sistem Bloomers menggambarkan hubungan antar kelas yang digunakan dalam proses klasifikasi tingkat kognitif soal berdasarkan Taksonomi Bloom. Diagram ini dibangun menggunakan pendekatan Model–View–Controller (MVC), yang memisahkan sistem ke dalam beberapa lapisan agar mudah dikembangkan dan dipelihara.

Kelas **User** berperan sebagai aktor utama yang berinteraksi dengan sistem melalui berbagai fungsi seperti pendaftaran (`signUp()`), login (`login()`), unggah file

soal (`uploadFile()`), melihat hasil klasifikasi (`viewClassification()`), mengunduh laporan (`downloadReport()`), dan regenerasi soal (`regenerateSoal()`).

Pada lapisan kontrol, terdapat beberapa kelas pengendali seperti `AuthController`, `UploadController`, `ClassificationController`, `RegenerateController`, dan `ReportController`. Setiap controller menangani logika proses tertentu, mulai dari autentikasi pengguna, pengunggahan file, klasifikasi soal, regenerasi, hingga pembuatan laporan.

Lapisan entitas mencakup kelas `FileSoal`, `Soal`, `RegenerateSoal`, `ReviewHistory`, dan `Laporan` yang menyimpan data hasil klasifikasi, hasil regenerasi, dan laporan yang dihasilkan sistem. Semua operasi data diatur oleh kelas `Database`, yang menangani penyimpanan dan pengambilan data dari seluruh modul.

Hubungan antar kelas menunjukkan alur proses yang terstruktur, di mana pengguna berinteraksi dengan controller, controller berkomunikasi dengan database, dan entitas menyimpan data hasil proses. Dengan struktur ini, sistem Bloomers dapat berjalan secara terintegrasi, efisien, dan mudah dikembangkan di masa mendatang.

## Appendix A: Glossary

**Tabel Glosary**

Taksonomi Bloom	Sebuah kerangka kerja yang digunakan untuk mengklasifikasikan tujuan pembelajaran ke dalam enam tingkatan kognitif.
C1-C6	Singkatan untuk enam tingkatan kognitif dalam Taksonomi Bloom, yaitu: C1 (Mengingat), C2 (Memahami), C3 (Menerapkan), C4 (Menganalisis), C5 (Mengevaluasi), dan C6 (Mencipta).
RoBERTa	(Robustly Optimized BERT Pretraining Approach) Sebuah model bahasa modern berbasis arsitektur transformer yang telah dioptimalkan untuk memahami teks secara kontekstual dan digunakan untuk klasifikasi soal.
NLP	(Natural Language Processing) Bidang kecerdasan buatan yang memungkinkan klasifikasi soal secara otomatis dengan memanfaatkan model bahasa.
TF-IDF	(Term Frequency-Inverse Document Frequency) Sebuah metode untuk mengubah teks soal menjadi representasi numerik agar dapat diproses oleh algoritma klasifikasi.
SVM	(Support Vector Machine) Salah satu algoritma klasifikasi pembelajaran mesin yang digunakan untuk menentukan level kognitif

	soal dengan mencari bidang pemisah (hyperplane) terbaik antar kelas.
Fine-tuning	Proses melatih ulang sebuah model bahasa yang sudah ada (seperti RoBERTa) menggunakan dataset spesifik (EDM2022) agar mampu melakukan klasifikasi sesuai tingkatan Taksonomi Bloom dengan lebih akurat.

## Appendix B: Analysis Models

Bagian ini menyajikan berbagai model analisis yang digunakan untuk memahami struktur dan alur kerja sistem aplikasi Bloomers secara menyeluruh. Model-model ini membantu menggambarkan hubungan antar komponen sistem, aliran data, serta desain konseptual dan logis dari basis data dan program.

Analisis dilakukan melalui beberapa representasi utama, yaitu:

1. Conceptual Framework, yang menjelaskan alur konseptual proses sistem mulai dari input hingga keluaran utama.
2. Conceptual Data Model (CDM), yang menggambarkan entitas utama dan hubungan antar entitas pada tingkat konseptual.
3. Physical Data Model (PDM), yang menunjukkan struktur tabel dan relasi antar tabel pada implementasi basis data.
4. Class Diagram, yang merepresentasikan hubungan antar kelas dalam sistem dari sisi pemrograman berorientasi objek.

Model-model tersebut digunakan sebagai dasar dalam tahap perancangan dan implementasi aplikasi agar sistem yang dibangun konsisten dengan kebutuhan fungsional dan non-fungsional yang telah ditetapkan.

## Appendix C: To Be Determined List

Tabel To Be Determined List

No.	Aspek yang Belum Ditetapkan (TBD)	Keterangan / Rencana Penetapan
1	Pelatihan model NLP (RoBERTa)	Model RoBERTa telah dipilih sebagai model utama klasifikasi, namun proses pelatihan (fine-tuning) terhadap dataset soal belum dilakukan dan akan diselesaikan pada tahap pengujian

2	Dataset pelatihan tambahan	Dataset soal pendidikan Indonesia masih dalam tahap pengumpulan dan kurasi.
3	Lingkungan deployment	Penentuan apakah sistem akan dijalankan secara lokal atau di server cloud masih dalam tahap evaluasi.
4	Jadwal backup dan maintenance	Frekuensi backup data dan perawatan sistem akan ditentukan setelah tahap implementasi stabil.
5	Evaluasi performa sistem	Pengujian waktu proses dan beban maksimum sistem akan dilakukan setelah uji coba versi beta.
6	Desain antarmuka final	Tampilan antarmuka akhir (warna, ikon, tata letak) masih dapat mengalami penyesuaian pada tahap pengujian pengguna.

**Tabel Lampiran Kontribusi Anggota Kelompok**

No	Uraian	Nala I'lma Kamila (187231003) 	Abhinaya Zaim (187231022) 	Balya Ananta Wicaksana (187231033) 	Ahmad Alvan Nurdin (187231043) 	Amir Gymnastiar (187231093) 	Raihan Arif Mustofa (187231084) 	Total
1	Pengumpulan dan preprocessing dataset (tokenisasi, normalisasi, labeling Taksonomi Bloom).	15%	15%	15%	25%	15%	15%	100%
2	Implementasi TF-IDF, Naive Bayes, dan SVM serta evaluasi performa model klasik.	15%	15%	15%	15%	15%	25%	100%

3	Fine-tuning model RoBERTa dan analisis perbandingan dengan model klasik.	15%	15%	25%	15%	15%	15%	100%
4	Pengembangan backend dan integrasi model	15%	15%	15%	15%	25%	15%	100%
5	Pembuatan antarmuka pengguna, fitur upload, dan tampilan hasil klasifikasi	25%	15%	15%	15%	15%	15%	100%
6	Visualisasi hasil klasifikasi (Plotly) dan ekspor laporan PDF	15%	15%	15%	15%	15%	15%	100%
7.	Pembuatan CDM, PDM, database, dan Class Diagram	15%	25%	15%	15%	15%	15%	100%
8.	Pembuatan User Story dan	15%	15%	25%	15%	15%	15%	100%

	User Scenario							
9.	Pembuatan Use case Diagram	15%	15%	15%	15%	25%	15%	100%
10.	Pembuatan Activity dan Sequence diagram	15%	15%	15%	25%	15%	15%	100%
11.	Pembuatan GUI	25%	15%	15%	15%	15%	15%	100%
12.	Penyusunan laporan SRS	15%	15%	15%	15%	15%	25%	100%