

SOFTWARE REQUIREMENTS SPECIFICATION (SRS)

FOR SISTEM PENDAFTARAN
TUGAS AKHIR

Disusun Oleh :

3411211036_Yoga

3411211041_Firza Rafiandi A

3411211044_Aldy Aditiya

DAFTAR ISI

DAFTAR ISI.....	i
DAFTAR GAMBAR	ii
BAB I. PENDAHULUAN.....	1
1.1 Purpose	1
1.2 Intended Audience and Reading Suggestions	2
1.3 Project Scope.....	3
1.4 References	4
BAB II. OVERAL DESCRIPTION	5
2.1 Organitations	5
2.2 Product Perspective	6
2.3 User Classes and Characteristics.....	7
2.4 Operating Environment	8
2.5 Design and Implementation Constrains (optional).....	9
2.6 Assumptions and Dependencies (optional)	9
BAB III. FUNCTIONAL REQUIREMENTS.....	11
1.1 Detailed Functional Requirements	11
3.2 Use Case Diagram.....	12
1.2 Use Case Scenario.....	13
BAB IV. NON FUNCTIONAL REQUIREMENTS	15
4.1 Performance Requirements (optional)	15
4.2 Safety Requirements (optional).....	16
4.3 Software Quality Attributes (optional).....	16
BAB V. DATA REQUIREMENTS	17
5.1 Input	17
5,2 Output.....	18
BAB VI. INTERFACE REQUIREMENTS	19
6.1 User Interface	19
6.2 Hardware Interface.....	20
6.3 Software Interface	21
6.4 Communication Interface.....	23

DAFTAR GAMBAR

Gambar 1. UseCase Diagram.....	12
--------------------------------	----

BAB I. PENDAHULUAN

1.1 Purpose

Latar belakang pembuatan dokumen SDD (Software Design Document) untuk tugas akhir sistem informasi di Universitas XYZ mungkin melibatkan beberapa faktor, seperti:

1. Kondisi Organisasi Klien: Dokumen SDD dapat mencakup deskripsi tentang organisasi klien yang menjadi fokus pengembangan sistem informasi. Ini melibatkan pemahaman tentang struktur organisasi, ukuran organisasi, industri atau bidang usaha yang terkait, proses bisnis yang ada, serta sistem informasi yang sedang digunakan. Informasi ini membantu dalam memahami konteks dan lingkungan di mana sistem informasi akan beroperasi.

2. Kebutuhan Pengembangan Organisasi: Dokumen SDD akan merinci kebutuhan pengembangan sistem informasi yang sesuai dengan kebutuhan dan tujuan organisasi klien. Ini melibatkan identifikasi kelemahan atau kekurangan dalam sistem yang ada, kebutuhan pengguna, persyaratan fungsional dan non-fungsional yang harus dipenuhi, serta tujuan bisnis yang ingin dicapai melalui pengembangan sistem informasi baru.

3. Masalah yang Diharapkan Terselesaikan: Dokumen SDD dapat mengidentifikasi masalah atau tantangan spesifik yang dihadapi oleh organisasi klien dan diharapkan dapat dipecahkan melalui pengembangan sistem informasi. Misalnya, mungkin ada masalah dalam pengelolaan data, kurangnya integrasi antar departemen, proses yang tidak efisien, atau kebutuhan untuk meningkatkan kinerja dan produktivitas organisasi. Dokumen SDD akan menjelaskan bagaimana sistem informasi yang dikembangkan diharapkan dapat mengatasi masalah tersebut.

Dalam konteks tugas akhir di Universitas XYZ, dokumen SDD akan menjadi panduan yang rinci tentang desain sistem informasi yang akan dikembangkan sebagai bagian dari proyek tugas akhir. Dokumen ini akan menjelaskan secara terperinci bagaimana sistem informasi akan memenuhi kebutuhan organisasi, mengatasi masalah yang ada, serta mencapai tujuan bisnis yang telah ditetapkan.

1.2 Intended Audience and Reading Suggestions

1. Mahasiswa:

- Memahami bagaimana menggunakan aplikasi untuk pengelolaan tugas akhir mereka.
- Mengetahui langkah-langkah yang diperlukan untuk mengajukan tugas akhir, mengunggah dokumen, dan mengikuti proses evaluasi.

2. Dosen Pembimbing 1 dan 2:

- Memahami cara menggunakan aplikasi untuk memberikan bimbingan kepada mahasiswa.
- Mengetahui proses penilaian dan evaluasi tugas akhir melalui aplikasi.

3. Dosen Penguji 1, 2, dan 3:

- Mengetahui bagaimana aplikasi mendukung proses penilaian dan evaluasi tugas akhir.
- Memahami cara mengakses dan memberikan feedback terkait tugas akhir mahasiswa.

4. Ketua Jurusan:

- Memahami bagaimana aplikasi dapat membantu dalam pemantauan dan manajemen tugas akhir di tingkat jurusan.
- Mengetahui cara melihat laporan dan statistik terkait tugas akhir mahasiswa.

5. Ketua KBK (Kelompok Bidang Keahlian):

- Memahami kontribusi aplikasi terhadap manajemen tugas akhir di tingkat bidang keahlian.
- Mengetahui cara menggunakan aplikasi untuk koordinasi antara dosen pembimbing dan penguji di dalam kelompok bidang keahlian.

6. Koordinator Tugas Akhir:

- Mengetahui seluruh proses dan fitur-fitur aplikasi dalam mengelola tugas akhir di tingkat universitas.
- Bertanggung jawab untuk menyampaikan panduan kepada pemangku kepentingan lainnya dan memastikan integrasi yang baik antara berbagai entitas yang terlibat dalam manajemen tugas akhir.

1.3 Project Scope

Perangkat lunak, juga dikenal sebagai software, adalah kumpulan program komputer, data, dan instruksi yang memberikan fungsi-fungsi yang diinginkan pada perangkat keras komputer. Lingkup perangkat lunak meliputi segala sesuatu yang berkaitan dengan desain, pengembangan, pengujian, implementasi, dan pemeliharaan program-program komputer.

Manfaat perangkat lunak meliputi:

1. Perangkat lunak memungkinkan proses-proses manual atau rutin dapat diotomatisasi, mengurangi kesalahan manusia, meningkatkan efisiensi, dan menghemat waktu.
2. Dengan menggunakan perangkat lunak yang tepat, tugas-tugas dapat diselesaikan lebih cepat dan lebih efisien, yang pada gilirannya meningkatkan produktivitas individu atau organisasi.
3. Perangkat lunak dapat dengan mudah disesuaikan dengan pertumbuhan atau perubahan kebutuhan bisnis. Hal ini memungkinkan perusahaan untuk menyesuaikan diri dengan cepat terhadap perubahan pasar atau kebutuhan pelanggan.
4. Dengan adanya perangkat lunak, kesalahan-kesalahan dapat diidentifikasi dan diperbaiki sebelum produk akhir dirilis. Hal ini membantu meningkatkan kualitas produk atau layanan yang disediakan.
5. Perangkat lunak dapat dikustomisasi sesuai kebutuhan pengguna. Pengguna dapat memilih fitur-fitur yang mereka perlukan dan mengintegrasikan perangkat lunak dengan sistem yang sudah ada.

Tujuan perangkat lunak meliputi:

1. Perangkat lunak dirancang untuk mengotomatisasi tugas-tugas yang memakan waktu, sehingga mengurangi beban kerja manusia dan meningkatkan efisiensi dalam menjalankan proses bisnis.
2. Menyediakan solusi yang memadai dan fungsional bagi pengguna.
3. Perangkat lunak dapat membantu meningkatkan kinerja sistem dengan mengidentifikasi dan menghilangkan bottlenecks, memperbaiki keandalan, dan meningkatkan waktu respons.
4. Perangkat lunak harus dirancang dengan memperhatikan keamanan data dan melindungi sistem dari ancaman keamanan seperti serangan malware atau peretasan.

5. Perangkat lunak juga dapat digunakan sebagai alat untuk mendorong inovasi dengan menciptakan solusi baru, memecahkan masalah yang kompleks, dan membuka peluang baru dalam berbagai bidang.

1.4 References

- [1] Modul Praktikum Analisis dan Perancangan Perangkat Lunak

BAB II. OVERAL DESCRIPTION

2.1 Organitations

Universitas XYZ memiliki tujuan untuk menjadi lembaga pendidikan yang unggul dalam pengelolaan tugas akhir dengan menyediakan solusi yang inovatif dan efisien. Misi universitas adalah mengembangkan aplikasi yang dapat membantu mahasiswa dan staf akademis dalam mengelola tugas akhir, memperoleh evaluasi yang optimal, dan meningkatkan kepuasan semua pihak yang terlibat. Struktur organisasi Universitas XYZ terdiri dari beberapa departemen yang memiliki peran kunci dalam pengembangan dan pengelolaan aplikasi ini.

1. Tim Pengembangan Perangkat Lunak:

- Bertanggung jawab merancang dan membangun aplikasi.
- Memastikan aplikasi memiliki antarmuka yang ramah pengguna dan sesuai dengan kebutuhan manajemen tugas akhir.

2. Tim Analisis Bisnis:

- Menganalisis kebutuhan pengguna dan menerjemahkannya menjadi persyaratan aplikasi yang jelas.
- Berkolaborasi erat dengan tim akademis dan teknis untuk memastikan aplikasi memenuhi tujuan manajemen tugas akhir universitas.

3. Tim Pemasaran:

- Bertanggung jawab mempromosikan aplikasi kepada mahasiswa, dosen, dan pemangku kepentingan lainnya.
- Mengembangkan strategi untuk meningkatkan kesadaran dan adopsi aplikasi "Manajemen Tugas Akhir" di lingkungan universitas.

4. Tim Implementasi:

- Mengelola penerapan dan integrasi aplikasi ke dalam sistem yang sudah ada di universitas.
- Berkolaborasi dengan departemen IT dan administrasi untuk memastikan proses implementasi berjalan lancar.

5. Tim Dukungan Pelanggan:

- Memberikan bantuan dan dukungan teknis kepada pengguna aplikasi.
- Memastikan responsif dan membantu dalam menanggapi masalah atau pertanyaan terkait aplikasi.

6. Mahasiswa:

- Terlibat dalam aplikasi untuk mengelola dan memantau progres tugas akhir mereka.
- Memanfaatkan platform untuk berkomunikasi dengan pembimbing, mengunggah dokumen proyek, dan memantau status evaluasi.

7. Dosen Pembimbing 1 dan 2:

- Menggunakan aplikasi untuk membimbing dan memberikan panduan kepada mahasiswa dalam tugas akhir.
- Mengakses fitur evaluasi proyek, memberikan umpan balik, dan memantau progres mahasiswa.

8. Dosen Penguji 1, 2, dan 3:

- Menggunakan aplikasi untuk meninjau dan menilai tugas akhir mahasiswa.
- Mengakses alat untuk memberikan evaluasi menyeluruh, umpan balik, dan memberikan nilai.

9. Ketua Jurusan:

- Mengawasi implementasi dan penggunaan aplikasi di tingkat jurusan.
- Memantau kinerja departemen dalam mengelola tugas akhir dan berkolaborasi dengan kepala departemen lainnya.

10. Ketua KBK (Kelompok Bidang Keahlian):

- Berkoordinasi di dalam KBK untuk memastikan aplikasi sesuai dengan kebutuhan bidang keahlian masing-masing.
- Memfasilitasi komunikasi dan kerja sama antara departemen dan tim akademis.

11. Koordinator Tugas Akhir:

- Berperan sebagai titik sentral kontak untuk penggunaan dan fungsionalitas aplikasi.
- Berkolaborasi dengan semua pihak terkait untuk memastikan proses manajemen tugas akhir berjalan efektif.

2.2 Product Perspective

Dari perspektif universitas, aplikasi "Manajemen Tugas Akhir" di Universitas XYZ diharapkan memberikan manfaat yang signifikan bagi seluruh entitas yang terlibat dalam proses akademis. Mahasiswa akan mendapatkan kemudahan dalam melacak dan memantau progres tugas akhir mereka, berkomunikasi dengan dosen pembimbing, dan memiliki akses mudah terhadap informasi terkait batas waktu dan persyaratan tugas akhir. Dosen Pembimbing

1 dan 2 akan merasakan manfaat dalam memberikan bimbingan akademis yang lebih efisien, memberikan umpan balik secara real-time, dan berkoordinasi dengan dosen penguji untuk evaluasi proyek mahasiswa. Dosen Penguji 1, 2, dan 3 akan dapat dengan lebih mudah menilai dan memberikan umpan balik pada tugas akhir mahasiswa melalui platform yang terintegrasi. Ketua Jurusan dapat memantau kinerja dan progres tugas akhir di tingkat departemen, sementara Ketua KBK dapat memastikan bahwa aplikasi mendukung kebutuhan dan spesifikasi bidang keahlian. Terakhir, Koordinator Tugas Akhir berperan sebagai pusat informasi dan bantuan teknis, memastikan penggunaan aplikasi berjalan lancar dan memberikan laporan serta analisis yang relevan untuk meningkatkan efektivitas manajemen tugas akhir di seluruh universitas. Melalui aplikasi ini, diharapkan bahwa Universitas XYZ dapat meningkatkan transparansi, koordinasi, dan efisiensi dalam manajemen tugas akhir, membawa dampak positif terhadap pengalaman akademis mahasiswa dan kinerja staf dosen.

2.3 User Classes and Characteristics

Aplikasi "Manajemen Tugas Akhir" di Universitas XYZ akan melibatkan beberapa kelas pengguna dengan karakteristik yang berbeda. Pertama, terdapat mahasiswa yang merupakan pengguna utama aplikasi ini. Mahasiswa akan menggunakan aplikasi untuk memantau dan mengelola progres tugas akhir mereka, berkomunikasi dengan dosen pembimbing, serta mengunggah dan memantau dokumen proyek. Karakteristik mahasiswa mencakup kebutuhan akan antarmuka pengguna yang ramah, kemudahan navigasi, dan akses cepat terhadap informasi terkait proyek akademis mereka.

Selanjutnya, terdapat Dosen Pembimbing 1 dan 2 yang akan menggunakan aplikasi untuk memberikan bimbingan akademis kepada mahasiswa, memberikan umpan balik secara real-time, dan memantau kemajuan proyek. Karakteristik dari dosen pembimbing mencakup kebutuhan akan alat evaluasi yang efisien, akses mudah terhadap informasi proyek mahasiswa, dan kemampuan untuk berkomunikasi secara efektif melalui platform aplikasi.

Dosen Penguji 1, 2, dan 3 juga akan menjadi pengguna aplikasi, menggunakan platform untuk meninjau dan menilai tugas akhir mahasiswa. Karakteristik dosen penguji melibatkan kebutuhan akan akses mudah terhadap dokumen proyek, alat evaluasi yang komprehensif, dan kemampuan untuk berkoordinasi dengan dosen pembimbing melalui aplikasi.

Selain itu, terdapat Ketua Jurusan yang akan memonitor kinerja dan progres tugas akhir di tingkat departemen. Karakteristik dari ketua jurusan mencakup kebutuhan akan laporan departemen yang akurat dan analisis kinerja mahasiswa.

Ketua KBK (Kelompok Bidang Keahlian) akan menggunakan aplikasi untuk memastikan bahwa aplikasi mendukung kebutuhan dan spesifikasi bidang keahlian. Karakteristik dari ketua KBK mencakup kebutuhan akan pemahaman mendalam tentang kebutuhan bidang keahlian dan kemampuan untuk memberikan umpan balik terkait pengembangan aplikasi.

Terakhir, Koordinator Tugas Akhir akan menjadi pengguna yang berperan sebagai pusat informasi dan bantuan teknis bagi mahasiswa dan dosen. Karakteristik koordinator tugas akhir mencakup kemampuan untuk memberikan panduan dan dukungan teknis kepada pengguna serta menghasilkan laporan dan analisis terkait manajemen tugas akhir.

2.4 Operating Environment

Berikut adalah Operating Environment atau lingkungan operasional yang dibutuhkan untuk menjalankan aplikasi manajemen tugas akhir di Universitas XYZ:

Perangkat Lunak:

1. Aplikasi Manajemen Tugas Akhir:

Aplikasi inti yang menyediakan antarmuka untuk mahasiswa, dosen pembimbing, dosen penguji, ketua jurusan, ketua KBK, dan koordinator tugas akhir untuk mengelola proyek tugas akhir, memberikan umpan balik, dan melacak kemajuan.

2. Sistem Database:

Sistem basis data yang dapat mendukung penyimpanan dan pengambilan data proyek tugas akhir. MySQL, Microsoft SQL Server, atau PostgreSQL dapat digunakan sesuai kebutuhan.

3. Perangkat Lunak Komunikasi:

Platform komunikasi internal untuk memfasilitasi interaksi antara mahasiswa dan dosen, serta koordinasi antara pihak terkait. Penggunaan email, pesan instan, atau platform kolaborasi seperti Microsoft Teams atau Slack dapat menjadi pilihan.

4. Sistem Manajemen Laporan:

Perangkat lunak untuk membuat dan mengelola laporan terkait manajemen tugas akhir. Ini dapat mencakup penggunaan Microsoft Excel, Google Sheets, atau aplikasi khusus untuk pembuatan laporan.

2.5 Design and Implementation Constrains (optional)

1. Batasan harus ditetapkan untuk menentukan fungsionalitas apa yang tersedia untuk setiap pengguna.
2. Batasan harus diperhitungkan untuk memastikan data dan informasi yang konsisten di seluruh sistem.
3. Batasan meliputi kemampuan sistem untuk menangani volume data yang meningkat, jumlah pengguna yang bertambah, dan kompleksitas operasional yang berkembang.
4. Batasan keamanan harus ditetapkan untuk melindungi data sensitif dan informasi perusahaan dari akses yang tidak sah dan serangan siber.
5. Batasan performa harus didefinisikan untuk memastikan bahwa PL dapat berfungsi dengan efisien dan responsif dalam kondisi operasional yang berbeda.
6. Batasan termasuk kesesuaian antarmuka untuk setiap jenis pengguna dan kemudahan penggunaan untuk mengoptimalkan produktivitas.

2.6 Assumptions and Dependencies (optional)

1. Ketergantungan pada Lingkungan Sistem Operasi:

Asumsi bahwa aplikasi manajemen tugas akhir di Universitas XYZ akan dijalankan pada lingkungan sistem operasi tertentu, misalnya Windows (XP, Vista, 7, Server 2008, dll). Diperlukan perhatian khusus terhadap kompatibilitas aplikasi dengan sistem operasi yang ditentukan.

2. Waktu Pengembangan Terbatas:

Keterbatasan waktu dalam pengembangan aplikasi manajemen tugas akhir dapat menyebabkan sejumlah fungsi mungkin tidak dapat diimplementasikan sepenuhnya. Proyek ini harus mempertimbangkan batasan waktu untuk memastikan fungsionalitas yang kritis dapat diintegrasikan dengan efektif.

3. Tidak Ada Perubahan pada File atau Database Tanpa Izin:

Asumsi bahwa pengembangan aplikasi ini tidak akan mengubah file-file atau basis data yang ada tanpa izin dari pihak berwenang, seperti ketua divisi akademik atau pengelola sistem informasi universitas. Setiap perubahan pada data mahasiswa atau struktur basis data perlu mendapatkan persetujuan sebelumnya.

4. Otomatisasi Pengelolaan Data Tugas Akhir:

Asumsi bahwa pengembangan aplikasi ini akan mengotomatisasi pengelolaan data yang terkait dengan tugas akhir, termasuk data mahasiswa, dosen pembimbing, dan data terkait tugas akhir lainnya. Otomatisasi ini diharapkan dapat meningkatkan efisiensi dan akurasi dalam manajemen data tugas akhir.

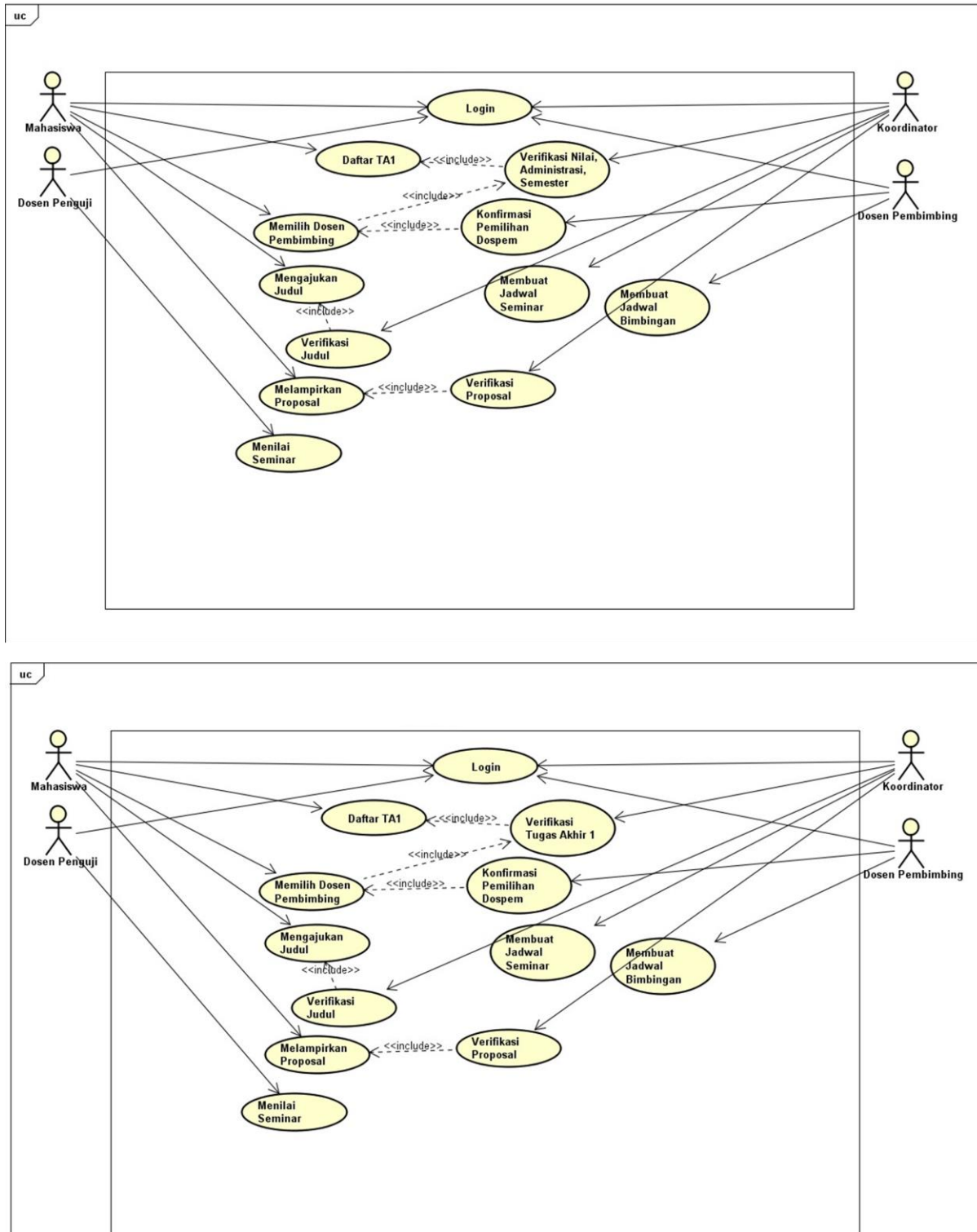
BAB III. FUNCTIONAL REQUIREMENTS

Kebutuhan Fungsional adalah kebutuhan yang berisi proses-proses apa saja / layanan apa saja yang nantinya harus disediakan oleh PL, mencakup bagaimana sistem harus bereaksi pada input tertentu dan bagaimana perilaku sistem pada situasi tertentu.

1.1 Detailed Functional Requirements

1. Sistem dapat menyediakan fungsi login yang juga berperan sebagai sistem validasi pembayaran. Note : Mahasiswa yang belum melakukan pembayaran maka tidak akan bisa masuk kedalam website akademik.
2. Sistem dapat melakukan validasi transaksi.
3. Sistem dapat mengelola data mahasiswa.
4. Sistem dapat mengelola data dosen pembimbing.
5. Sistem dapat mengelola data dosen penguji.
6. Sistem dapat melakukan pendataan terkait dengan dokumen mahasiswa.
7. Sistem dapat menampilkan jadwal dan waktu pelaksanaan kegiatan seminar.
8. Sistem dapat melakukan perhitungan kalkulasi nilai khs mahasiswa.
9. Sistem dapat melakukan perhitungan kalkulasi nilai TA mahasiswa.
10. Sistem dapat melakukan seleksi data mahasiswa yang ditujukan untuk menguji kelayakan data hasil penelitian, khs serta pembayaran. Mahasiswa yang telah melakukan point-point tersebut maka akan maju ke tahap selanjutnya.
11. Sistem dapat melakukan deteksi plagiarisme.
12. Sistem dapat menampilkan data dosen penguji dan pembimbing kepada mahasiswa

3.2 Use Case Diagram



Gambar 1. UseCase Diagram

1.2 Use Case Scenario

- A. Kebutuhan Fungsional
 - Pendaftaran TA
 - Pengajuan Judul
 - Bimbingan
 - Pengajuan Seminar
 - Pembayaran
- B. Use Case Scenario
 - 1. Pendaftaran TA

Mahasiswa	Sistem
Masuk ke sistem pendaftaran TA.	
Mengisi formulir pendaftaran dengan informasi pribadi dan detail tentang TA.	
Melampirkan berkas yang diperlukan	
	Memvalidasi data dan menyimpan pendaftaran TA

2. Pengajuan Judul

Mahasiswa	Dosen Pembimbing	Sistem
masuk ke sistem dan memilih opsi "Pengajuan Judul TA".		
memasukkan judul yang diinginkan dan menjelaskan latar belakang serta tujuan penelitian.		
		mengirim notifikasi kepada dosen pembimbing terkait.
	meninjau dan menyetujui atau menolak judul tersebut melalui sistem.	

3. Bimbingan

Mahasiswa	Sistem
Membuat janji temu melalui sistem dengan dosen pembimbing	
Mahasiswa dapat mengajukan pertanyaan, mendiskusikan perkembangan penelitian	
	Menyimpan Data Pertemuan dan Bimbingan

4. Pengajuan Seminar

Mahasiswa	Koordinator	Sistem
-----------	-------------	--------

Masuk ke sistem dan memilih opsi "Pengajuan Seminar TA"		
Mengisi formulir dengan detail jadwal yang diinginkan dan materi presentasi seminar.		
		mengirim notifikasi kepada dosen pembimbing dan pihak terkait lainnya.
	Meninjau pengajuan dan menyetujui atau menolak jadwal serta materi seminar tersebut.	

5. Pembayaran

Mahasiswa	Sistem
Masuk ke sistem pembayaran tugas akhir.	
Mahasiswa memilih opsi pembayaran yang tersedia, seperti transfer bank atau pembayaran online.	
Mahasiswa mengisi detail pembayaran, termasuk jumlah yang harus dibayarkan dan nomor referensi.	
	Memperbarui status pembayaran dan mengirimkan konfirmasi kepada mahasiswa.

BAB IV. NON FUNCTIONAL REQUIREMENTS

Kebutuhan Non Fungsional adalah kebutuhan yang menitikberatkan pada properti perilaku yang dimiliki oleh sistem. kebutuhan fungsional juga sering disebut sebagai batasan layanan atau fungsi yang ditawarkan sistem seperti batasan waktu, batasan pengembangan proses, standarisasi dan lain lain. Contoh :

1. Sistem dapat memberikan laporan mahasiswa kepada dosen
2. Sistem dapat menyimpan data dokumentasi mahasiswa meliputi makalah karya ilmiah yang akan dipublikasikan. Penyimpanan data ini dilakukan sebagai rekam jejak timeline dari kegiatan mahasiswa. Jika pada timeline waktu yang sudah ditentukan, mahasiswa belum melakukan publikasi maka akan diberikan sanksi meliputi penahanan nilai dan jika sampai dengan sidang yudisium, mahasiswa belum melakukan publikasi maka mahasiswa wajib mengambil kembali TA2 pada semester berikutnya.
3. Sistem dapat melakukan fungsi chatting yang bertujuan sebagai sarana komunikasi antar mahasiswa dengan dosen.
4. Sistem dapat memberikan transparansi informasi nilai seminar mahasiswa dengan point sebagai berikut : nilai bimbingan (40%), nilai pengujian (50%) dan nilai administrasi (10%). Transparansi nilai yang diberikan dalam bentuk indeks dengan range : 0-44,9 (E), 45-54,9 (D), 55-64,9 (C), 65-79,9 (B), 80-100 (A).
5. Sistem dapat menampilkan data waktu terbaru (kalender dan jam)

4.1 Performance Requirements (optional)

Kebutuhan yang diinginkan :

1. Kecepatan (Speed) atau responsive.
2. Skalabilitas (Scalability)
3. Ketersediaan (Availability)
4. Efisiensi (Efficiency)
5. Skalabilitas Vertikal dan Horizontal (Vertical and Horizontal Scalability)
6. Keandalan (Reliability)

4.2 Safety Requirements (optional)

Kebutuhan dari standar PL yang diinginkan organisasi sebagai berikut :

1. Pengamanan Data (Data Security)
2. Otentikasi dan Otorisasi (Authentication and Authorization)
3. Pengelolaan Keamanan (Security Management)

4.3 Software Quality Attributes (optional)

Kebutuhan dari sisi kualitas PL yang diinginkan organisasi sebagai berikut :

1. Fungsionalitas
2. Keandalan
3. Usabilitas
4. Kinerja
5. Keamanan
6. Maintainabilitas dan Scalability

BAB V. DATA REQUIREMENTS

5.1 Input

1. Proses Pengajuan Tugas Akhir oleh Mahasiswa:

- Mahasiswa mengajukan tugas akhir melalui aplikasi.
- Atribut:
 - Judul Tugas Akhir: untuk menentukan judul proyek.
 - Bidang Penelitian: untuk memilih bidang penelitian yang sesuai.
 - Abstrak: untuk memberikan ringkasan proyek.

2. Proses Penetapan Dosen Pembimbing:

- Koordinator Tugas Akhir menetapkan Dosen Pembimbing 1 dan 2 untuk mahasiswa yang mengajukan tugas akhir.
- Atribut:
 - Mahasiswa: untuk memilih mahasiswa yang akan ditetapkan.
 - Dosen Pembimbing 1: untuk menentukan dosen pembimbing utama.
 - Dosen Pembimbing 2: untuk menentukan dosen pembimbing pendamping.

3. Proses Penetapan Dosen Penguji:

- Koordinator Tugas Akhir menetapkan Dosen Penguji 1, 2, dan 3 untuk menguji tugas akhir mahasiswa.
- Atribut:
 - Mahasiswa: untuk memilih mahasiswa yang akan diuji.
 - Dosen Penguji 1, 2, dan 3: untuk menentukan dosen penguji yang akan terlibat.

4. Proses Pengajuan Perubahan Judul atau Dosen Pembimbing:

- Mahasiswa dapat mengajukan perubahan judul atau dosen pembimbing jika diperlukan.
- Atribut:
 - Mahasiswa: untuk memilih mahasiswa yang mengajukan perubahan.
 - Judul Baru (jika perubahan judul): untuk menentukan judul baru.
 - Dosen Pembimbing Baru (jika perubahan pembimbing): untuk menentukan dosen pembimbing baru.

5,2 Output

1. Laporan Progres Tugas Akhir:

- Mahasiswa dan Dosen Pembimbing 1 dan 2 memiliki akses ke laporan progres tugas akhir.
- Atribut Laporan:
 - Evaluasi Progres: untuk menunjukkan evaluasi progres mahasiswa.
 - Catatan Dosen Pembimbing: untuk catatan dan umpan balik dari dosen pembimbing.

2. Laporan Penilaian Tugas Akhir:

- Dosen Penguji 1, 2, dan 3 memberikan penilaian terhadap tugas akhir mahasiswa.
- Atribut Laporan:
 - Nilai Akhir: untuk menampilkan nilai akhir yang diberikan.
 - Catatan dan Umpan Balik: untuk memberikan komentar dan umpan balik terkait penilaian.

3. Pemberitahuan Hasil Penilaian:

- Mahasiswa menerima pemberitahuan hasil penilaian tugas akhir.
- Atribut Pemberitahuan:
 - Status Lulus/Tidak Lulus: untuk menunjukkan apakah mahasiswa lulus atau tidak lulus.
 - Nilai Akhir: untuk menampilkan nilai akhir (jika lulus).
 - Catatan dan Umpan Balik: untuk memberikan komentar dan umpan balik terkait hasil penilaian.

4. Laporan Ketersediaan Dosen Pembimbing:

- Koordinator Tugas Akhir dan Ketua KBK memiliki akses ke laporan ketersediaan dosen pembimbing.
- Atribut Laporan:
 - Status Ketersediaan: untuk menunjukkan ketersediaan dosen pembimbing.
 - Jadwal yang Dapat Diambil: untuk menampilkan jadwal yang masih dapat diambil oleh mahasiswa.

BAB VI. INTERFACE REQUIREMENTS

6.1 User Interface

Dalam sistem informasi tugas akhir di Universitas XYZ, berikut adalah karakteristik umum dari setiap antarmuka antara perangkat lunak (PL) dan pengguna:

A. Antarmuka Pengguna (User Interface):

- Interaktif: Antarmuka pengguna harus dapat berinteraksi dengan pengguna secara langsung, memberikan respons terhadap input dan tindakan pengguna.
- User-friendly: Antarmuka harus dirancang dengan baik, mudah dipahami, dan intuitif bagi pengguna.
- Navigasi yang jelas: Pengguna harus dapat dengan mudah menjelajahi berbagai fitur dan fungsionalitas sistem melalui antarmuka pengguna.
- Tampilan yang menarik: Antarmuka harus memiliki tampilan yang menarik secara visual, menggunakan elemen desain yang estetik dan konsisten.

B. Antarmuka Sistem Eksternal:

- Standar komunikasi: Antarmuka ini harus mematuhi standar komunikasi yang relevan, seperti protokol jaringan, format data, atau API yang digunakan untuk berinteraksi dengan sistem eksternal.
- Keamanan: Antarmuka ini harus menjaga keamanan data dan informasi yang dipertukarkan antara sistem informasi tugas akhir dan sistem eksternal. Hal ini mungkin melibatkan mekanisme otentikasi, enkripsi, atau pengaturan izin akses.

C. Antarmuka Basis Data:

- Konektivitas: Antarmuka ini bertanggung jawab untuk menghubungkan perangkat lunak dengan basis data yang digunakan untuk menyimpan dan mengelola data. Hal ini mencakup koneksi ke basis data, pemrosesan query, dan pengambilan dan penyimpanan data.
- Keamanan data: Antarmuka ini juga harus menjaga keamanan dan integritas data dalam basis data, melalui kontrol akses, enkripsi, dan mekanisme pengelolaan data yang sesuai.

Karakteristik antarmuka ini dapat disesuaikan dengan persyaratan dan kebutuhan sistem informasi tugas akhir di Universitas XYZ. Penting untuk mempertimbangkan pengalaman pengguna, keamanan data, dan fungsionalitas yang diperlukan saat merancang dan mengembangkan antarmuka antara PL dengan pengguna maupun sistem eksternal atau basis data yang terlibat.

6.2 Hardware Interface

Dalam sistem informasi tugas akhir di Universitas XYZ, perangkat lunak (PL) akan berinteraksi dengan beberapa komponen hardware yang mendukung operasionalnya. Berikut ini adalah contoh karakteristik antarmuka antara PL dengan komponen hardware tertentu: A. Antarmuka dengan Processor (CPU):

Deskripsi: Antarmuka ini digunakan untuk berinteraksi dengan unit pemrosesan pusat (CPU) yang menjalankan instruksi-instruksi PL.

Karakteristik:

- a. Instruksi dan Eksekusi: Antarmuka harus mampu mengirim instruksi ke CPU untuk dieksekusi. Instruksi tersebut dapat berupa operasi matematika, logika, akses memori, dan sebagainya.
 - b. Pengelolaan Sumber Daya: Antarmuka harus memperhatikan penggunaan sumber daya CPU, seperti alokasi memori, penjadwalan proses, dan pemantauan kinerja CPU.
- B. Antarmuka dengan Memori: Deskripsi: Antarmuka ini digunakan untuk berinteraksi dengan memori komputer yang menyimpan program dan data yang digunakan oleh PL.

Karakteristik:

- a. Baca dan Tulis Data: Antarmuka harus mampu membaca dan menulis data ke memori. Hal ini meliputi pengambilan instruksi PL dari memori dan penyimpanan data sementara (variabel, objek, struktur data) yang digunakan oleh PL.
- b. Pengelolaan Memori: Antarmuka harus mempertimbangkan alokasi, dealokasi, dan manajemen memori agar PL dapat berjalan dengan efisien dan tidak terjadi konflik memori.

Antarmuka dengan Perangkat Input:

Deskripsi: Antarmuka ini digunakan untuk berinteraksi dengan perangkat input seperti keyboard, mouse, atau perangkat pengetikan lainnya.

Karakteristik:

- a. Penerimaan Input: Antarmuka harus mampu menerima input dari perangkat input yang digunakan oleh pengguna, seperti membaca tombol yang ditekan pada keyboard atau gerakan mouse.
- b. Interpretasi Input: Antarmuka harus dapat menginterpretasikan input dari perangkat input menjadi instruksi atau tindakan yang dapat dipahami oleh PL. D. Antarmuka dengan Perangkat Output:

Deskripsi: Antarmuka ini digunakan untuk berinteraksi dengan perangkat output seperti monitor, printer, atau perangkat tampilan lainnya.

Karakteristik:

- a. Pengiriman Output: Antarmuka harus mampu mengirimkan output atau hasil dari PL ke perangkat output yang tepat, seperti menampilkan informasi di layar monitor atau mencetak dokumen melalui printer.
- b. Pemformatan Output: Antarmuka harus dapat memformat output agar sesuai dengan format yang diinginkan oleh perangkat output, misalnya resolusi layar monitor atau format cetakan yang diinginkan.

Penting untuk memperhatikan kompatibilitas antarmuka PL dengan komponen hardware yang digunakan dalam sistem informasi tugas akhir. Hal ini memastikan bahwa PL dapat berinteraksi dengan perangkat keras dengan baik dan mendukung fungsionalitas yang diinginkan. Selain itu, juga penting untuk memastikan bahwa antarmuka memenuhi standar komunikasi dan keamanan yang relevan untuk menjaga kinerja dan integritas sistem secara keseluruhan.

6.3 Software Interface

Dalam sistem informasi tugas akhir di Universitas XYZ, perangkat lunak (PL) akan berinteraksi dengan beberapa komponen perangkat lunak lainnya. Berikut ini adalah contoh karakteristik antarmuka antara PL dengan komponen perangkat lunak tertentu:

A. Antarmuka dengan Sistem Manajemen Basis Data (Database Management System/DBMS):

Deskripsi: Antarmuka ini digunakan untuk berkomunikasi dengan DBMS yang digunakan dalam sistem informasi tugas akhir.

Karakteristik:

- a. SQL Interface: Antarmuka harus mendukung penggunaan bahasa SQL (Structured Query Language) untuk mengirim permintaan dan mengelola data di dalam database. Contohnya adalah antarmuka dengan MySQL (versi 8.0) atau PostgreSQL (versi 13.0).
- b. Koneksi dan Autentikasi: Antarmuka harus mampu menjalin koneksi dengan DBMS dan melakukan proses autentikasi dengan menggunakan nama pengguna (username) dan kata sandi (password).
- c. Query Execution: Antarmuka harus dapat menjalankan query SQL untuk mengambil, memperbarui, atau menghapus data dari database. Ini mencakup penggunaan pernyataan SELECT, INSERT, UPDATE, DELETE, dan sejenisnya.
- d. Pengelolaan Transaksi: Antarmuka harus mendukung pengelolaan transaksi dalam database, seperti memulai, menggulirkan (rollback), atau mengkonfirmasi (commit) transaksi.

B. Antarmuka dengan Sistem Operasi:

Deskripsi: Antarmuka ini digunakan untuk berinteraksi dengan sistem operasi di mana PL dijalankan.

Karakteristik:

- a. Fungsi Sistem Operasi: Antarmuka harus memanfaatkan fungsi yang disediakan oleh sistem operasi, seperti membaca atau menulis file, mengelola proses, mengatur hak akses pengguna, dan sebagainya.
- b. Kompatibilitas Platform: Antarmuka harus dapat beroperasi dengan sistem operasi yang ditargetkan, misalnya Windows (versi 10), Linux (misalnya Ubuntu 20.04), atau macOS (versi 12.0).
- c. Interaksi dengan Layanan Sistem: Antarmuka harus dapat berinteraksi dengan layanan sistem operasi, seperti mengirim email melalui layanan email, mengatur jaringan, atau mengelola layanan terjadwal.

C. Antarmuka dengan Layanan Eksternal:

Deskripsi: Antarmuka ini digunakan untuk berkomunikasi dengan layanan eksternal yang terintegrasi dalam sistem informasi tugas akhir.

Karakteristik:

- a. Protokol Komunikasi: Antarmuka harus mendukung protokol yang sesuai untuk berkomunikasi dengan layanan eksternal, seperti protokol HTTP, protokol SOAP, atau protokol RESTful.
- b. Format Data: Antarmuka harus mendukung format data yang diterima dan dikirim oleh layanan eksternal, seperti JSON (JavaScript Object Notation) atau XML (eXtensible Markup Language).
- c. Pengelolaan Koneksi: Antarmuka harus dapat mengelola koneksi dengan layanan eksternal, termasuk membuat koneksi, mengirim permintaan, dan menerima respons.
- d. Keamanan: Antarmuka harus mempertimbangkan aspek keamanan, seperti penggunaan kunci API (Application Programming Interface) untuk mengotentikasi dan mengamankan komunikasi antara PL dan layanan eksternal.

Penting untuk merancang antarmuka yang sesuai dan kompatibel dengan komponen perangkat lunak lainnya agar dapat berinteraksi dengan baik dan mendukung fungsionalitas sistem informasi tugas akhir secara efektif.

6.4 Communication Interface

Antarmuka antara pengguna dan perangkat lunak (PL) pada sistem informasi tugas akhir di Universitas XYZ memiliki fitur-fitur berikut:

A. Antarmuka Pengguna Grafis (Graphical User Interface/GUI):

Deskripsi: Antarmuka grafis memungkinkan pengguna berinteraksi dengan PL menggunakan elemen visual seperti tombol, menu, dan formulir.

Karakteristik:

- a. Visual dan Intuitif: Antarmuka harus dirancang dengan tampilan menarik dan mudah dipahami agar pengguna dapat dengan mudah menggunakan fitur-fitur PL.
- b. Navigasi yang Mudah: Pengguna harus dapat dengan mudah menjelajahi fungsi-fungsi PL melalui menu, ikon, atau tautan yang jelas dan terorganisir.
- c. WYSIWYG (What You See Is What You Get): Antarmuka harus menampilkan tampilan yang akurat dan sesuai dengan output yang dihasilkan oleh PL.
- d. Responsif: Antarmuka harus memberikan umpan balik yang cepat dan merespons tindakan pengguna secara instan.

B. Antarmuka Masukan Pengguna (User Input Interface):

Deskripsi: Antarmuka ini memungkinkan pengguna untuk memberikan masukan atau input kepada PL.

Karakteristik:

- a. Formulir: Antarmuka harus menyediakan formulir yang mudah diisi dan meminta pengguna memberikan informasi yang jelas sesuai kebutuhan.
- b. Validasi Data: Antarmuka harus memvalidasi data untuk memastikan bahwa masukan pengguna sesuai dengan format atau aturan yang ditentukan sebelum diterima oleh PL.
- c. Kontrol Input: Antarmuka harus menyediakan kontrol input yang sesuai, seperti kotak teks, tombol, pilihan drop-down, dan pilih tanggal, untuk memudahkan pengguna memberikan masukan.

C. Antarmuka Notifikasi dan Pesan (Notification and Messaging Interface):

Deskripsi: Antarmuka ini digunakan untuk menyampaikan pesan, pemberitahuan, atau umpan balik kepada pengguna.

Karakteristik:

- a. Pemberitahuan: Antarmuka harus mampu memberikan pemberitahuan yang jelas dan informatif kepada pengguna tentang perubahan status, perkembangan, atau kejadian penting terkait tugas akhir mereka.
- b. Pesan Kesalahan: Jika terjadi kesalahan atau masalah dalam penggunaan PL, antarmuka harus dapat menampilkan pesan kesalahan yang informatif untuk membantu pengguna memahami masalah dan mengambil langkah yang tepat.
- c. Umpan Balik: Antarmuka harus memberikan umpan balik kepada pengguna tentang tindakan yang mereka lakukan, seperti konfirmasi pendaftaran tugas akhir atau status persetujuan.

Penting untuk merancang antarmuka pengguna dengan memperhatikan prinsip-prinsip usability, seperti kejelasan, konsistensi, dan kemudahan penggunaan yang baik. Dengan antarmuka yang baik, pengguna akan dapat berinteraksi dengan PL secara efisien dan meningkatkan pengalaman pengguna secara keseluruhan.