

# SOFTWARE REQUIREMENTS SPECIFICATION (SRS)

FOR MANAJEMEN ADMINISTRASI  
TUGAS AKHIR JURUSAN INFORMATIKA  
UNIVERSITAS XYZ

Kelompok:

Dwi Wahyu Kuncoro	3411211048
Rendy Adhinata	3411211052
Nabillah Nurhaliza	3411211058
Denissya Julianthy	3411211063

## Table Of Contents

Table Of Contents .....	i
Bab I Introduction .....	1
1.1 Purpose .....	1
1.2 Intended Audience and Reading Suggestions.....	1
1.3 Project Scope .....	3
1.4 References .....	4
Bab II Overall Description .....	6
2.1 Organisations .....	6
2.2 Product Perspective .....	6
2.3 User Classes and Characteristics.....	7
2.4 Operating Environment .....	8
2.5 Design and Implementation Constrains.....	9
2.6 Assumptions and Dependencies .....	10
Bab III Functional Requirements.....	12
3.1 Detailed Functional Requirements .....	12
3.2 Use Case Diagram .....	13
3.3 Use Case Scenario .....	14
Bab IV Non Functional Requirements.....	20
4.1 Performance Requirements.....	20
4.2 Safety Requirements.....	21
4.3 Software Quality Attributes .....	21
Bab V Data Requirements.....	23
4.1 Input.....	23
4.2 Output.....	23
Bab VI Interface Requirements.....	26
4.1 User Interface .....	26
4.2 Hardware Interface .....	28
4.3 Software Interface .....	29
4.4 Communication Interface .....	30

## Bab I Introduction

### 1.1 Purpose

Latar Belakang:

1. Identifikasi Kebutuhan TA: Dokumen SRS membantu mengidentifikasi dan mendokumentasikan kebutuhan pengguna, terutama terkait administrasi Tugas Akhir di Universitas XYZ. Hal ini memastikan bahwa sistem yang dikembangkan dapat memenuhi persyaratan dasar untuk mendukung proses administrasi TA secara efektif.
2. Pemahaman Bersama: Dokumen SRS membangun pemahaman yang lebih baik antara pengembang perangkat lunak dan pihak yang terlibat dalam pengelolaan administrasi TA. Pemahaman yang sama mengenai fitur-fitur yang diinginkan dan tujuan administrasi TA membantu mengurangi risiko terjadinya kesalahpahaman selama pengembangan.
3. Panduan Proyek: Dokumen SRS memberikan panduan jelas kepada tim pengembang mengenai implementasi dan pengujian sistem pengelolaan administrasi TA. Ini membantu mengatur langkah-langkah pengembangan, memastikan fokus pada kebutuhan administrasi TA, dan memberikan arah selama seluruh proses pengembangan.
4. Evaluasi dan Validasi: Dokumen SRS menjadi dasar untuk mengevaluasi dan memvalidasi sistem administrasi TA yang dikembangkan. Dengan demikian, sistem dapat dipastikan memenuhi kebutuhan yang telah ditetapkan dan sesuai dengan harapan pihak terkait di Universitas XYZ.

Tujuan:

1. Menyediakan Rujukan: Dokumen SRS menjadi rujukan utama bagi pengembang dalam merancang, mengembangkan, dan menguji sistem pengelolaan administrasi TA. Ini memastikan bahwa sistem yang dibangun sesuai dengan persyaratan yang telah ditetapkan oleh Universitas XYZ.
2. Menjelaskan Kebutuhan: Dokumen SRS secara rinci menjelaskan kebutuhan fungsional dan non-fungsional dari sistem administrasi TA. Ini mencakup deskripsi fitur-fitur, batasan sistem, interaksi pengguna, keamanan, kinerja, dan aspek lainnya, sehingga semua pihak terkait memiliki pemahaman yang sama tentang persyaratan sistem.
3. Menghindari Kesalahpahaman: Dokumen SRS membantu menghindari kesalahpahaman antara pengembang dan pihak yang terlibat dalam administrasi TA. Dengan mendokumentasikan persyaratan secara rinci, semua pihak terkait memiliki pemahaman yang jelas tentang tujuan dan spesifikasi sistem.
4. Memandu Pengembangan: Dokumen SRS memberikan panduan yang jelas bagi tim pengembang dalam merancang dan mengembangkan sistem administrasi TA. Ini membantu dalam mengarahkan langkah-langkah pengembangan, termasuk perencanaan, desain, implementasi, dan pengujian, sesuai dengan kebutuhan yang telah ditetapkan oleh Universitas XYZ.

### 1.2 Intended Audience and Reading Suggestions

Berikut adalah penjelasan lebih rinci mengenai peran dan kepentingan masing-masing pihak terkait:

1. Pengembang (Developers):

- Peran: Pembaca utama yang bertanggung jawab merancang, mengembangkan, dan mengimplementasikan sistem perangkat lunak.
  - Saran Membaca: Fokus pada kebutuhan fungsional dan non-fungsional, cakupan sistem, dan fitur-fitur yang dijelaskan dalam SRS. Perhatikan spesifikasi teknis untuk pemahaman yang jelas terkait tujuan pengembangan.
2. Manajer Proyek (Project Managers):
- Peran: Individu kunci yang terlibat dalam perencanaan dan manajemen proyek.
  - Saran Membaca: Tekankan pada bagian terkait tujuan proyek, jadwal, dan kebutuhan sumber daya. Gunakan SRS sebagai dasar untuk komunikasi proyek, memastikan kesesuaian dengan tujuan keseluruhan, dan memfasilitasi koordinasi dengan tim pengembang.
3. Pengguna (Users):
- Peran: Pengguna akhir atau calon pengguna sistem.
  - Saran Membaca: Telusuri bagian yang merinci kemampuan sistem, interaksi pengguna, dan manfaat yang diharapkan. Gunakan SRS untuk mendapatkan wawasan tentang sejauh mana sistem ini sesuai dengan kebutuhan dan harapan pengguna.
4. Penguji (Testers):
- Peran: Tim yang bertanggung jawab untuk memvalidasi dan menguji sistem yang dikembangkan.
  - Saran Membaca: Perhatikan bagian yang mendefinisikan persyaratan sistem untuk pengujian dan skenario pengujian. Gunakan SRS sebagai referensi untuk membuat rencana pengujian yang efisien dan memastikan sistem memenuhi persyaratan yang ditentukan.
5. Penulis Dokumen (Document Writers):
- Peran: Individu yang terlibat dalam membuat dokumentasi terkait proyek, seperti penulis teknis.
  - Saran Membaca: Gunakan SRS sebagai sumber utama untuk memahami kebutuhan sistem. Gunakan sebagai referensi saat membuat dokumen desain, panduan pengguna, atau materi pelatihan.

Tips Membaca untuk Semua Pembaca:

- Mulailah dengan gambaran umum SRS untuk memahami konteks dan tujuan proyek.
- Fokus pada bagian tertentu yang relevan dengan peran dan tanggung jawab Anda.
- Catat setiap ketergantungan, kendala, atau fitur kritis yang dijelaskan dalam dokumen.
- Komunikasikan dan berkolaborasi dengan pemangku kepentingan lain untuk memastikan pemahaman yang seragam mengenai tujuan proyek dan kebutuhan.
- Secara berkala kembali ke SRS sepanjang siklus proyek untuk menjaga kesesuaian dengan tujuan yang ditetapkan.

### 1.3 Project Scope

Tujuan Perangkat Lunak:

1. Peningkatan Administrasi Tugas Akhir (TA):
  - Deskripsi: Lingkup proyek mencakup pengembangan perangkat lunak untuk meningkatkan administrasi Tugas Akhir mahasiswa di Universitas XYZ.
  - Fitur Utama: Sistem ini akan menyediakan fitur-fitur yang mendukung manajemen Tugas Akhir, termasuk pencatatan progres, pengelolaan waktu, dan penyimpanan dokumen terkait.
2. Memudahkan Proses Administrasi:
  - Deskripsi: Perangkat lunak akan dirancang untuk mempermudah proses administrasi terkait pengelolaan Tugas Akhir.
  - Fitur Utama: Automatisasi tugas administratif, penjadwalan pertemuan antara mahasiswa dan pembimbing, serta integrasi dengan sistem administrasi akademik untuk keperluan pelaporan.
3. Peningkatan Kolaborasi:
  - Deskripsi: Fokus pada meningkatkan kolaborasi antara mahasiswa, dosen pembimbing, dan pihak terkait lainnya.
  - Fitur Utama: Menyediakan platform komunikasi dan kolaborasi, termasuk forum diskusi, notifikasi otomatis, dan fitur berbagi dokumen untuk memfasilitasi pertukaran informasi.
4. Pemantauan Progres secara Efektif:
  - Deskripsi: Lingkup proyek mencakup pengembangan fitur pemantauan progres Tugas Akhir.
  - Fitur Utama: Dashboard pemantauan progres, pelacakan perkembangan, dan pelaporan otomatis untuk memudahkan pemantauan oleh dosen pembimbing, mahasiswa, dan pihak terkait.
5. Meningkatkan Transparansi Proses Administrasi:
  - Deskripsi: Tujuan proyek adalah meningkatkan transparansi dalam proses administrasi Tugas Akhir.
  - Fitur Utama: Akses mudah ke informasi terkait Tugas Akhir, termasuk kebijakan, pedoman, dan status progres, untuk mahasiswa, dosen, dan pihak terkait lainnya.
6. Integrasi dengan Sistem Eksternal:
  - Deskripsi: Lingkup proyek mencakup integrasi dengan sistem administrasi universitas yang sudah ada.
  - Fitur Utama: Memastikan keterkaitan yang lancar antara sistem pengelolaan administrasi Tugas Akhir dan sistem-sistem lainnya yang digunakan oleh universitas.
7. Pelatihan dan Dukungan Pengguna:
  - Deskripsi: Menyediakan pelatihan dan dukungan pengguna untuk memastikan pemahaman yang baik dan penggunaan efektif sistem.
  - Fitur Utama: Modul pelatihan online, panduan pengguna, dan dukungan teknis yang dapat diakses oleh mahasiswa, dosen, dan staf administrasi.

Catatan Tambahan:

- Lingkup proyek ini fokus pada pengembangan sistem pengelolaan administrasi Tugas Akhir secara menyeluruh.
- Integrasi dengan sistem eksternal memastikan keterkaitan yang efisien dan pertukaran data yang lancar antara sistem TA dan sistem lainnya.

#### 1.4 References

Dokumen SRS (*Software Requirements Specification*) untuk Sistem Pengelolaan Gaji Pegawai. Berikut adalah penjelasan lebih detail tentang masing-masing dokumen referensi tersebut:

1. Panduan Pengelolaan Administrasi Tugas Akhir:
  - [Panduan Administrasi Tugas Akhir Universitas XYZ, Edisi Terbaru]
  - Fungsi: Menyediakan aturan, persyaratan, pedoman, dan prosedur terkait pengelolaan administrasi Tugas Akhir di Universitas XYZ.
2. Kebijakan dan Standar Universitas XYZ:
  - [Dokumen Kebijakan dan Standar Universitas XYZ]
  - Fungsi: Merinci kebijakan keamanan, standar presentasi, dan kebijakan privasi yang berlaku di Universitas XYZ, yang harus diperhatikan dalam pengembangan sistem administrasi Tugas Akhir.
3. Dokumen Spesifikasi Teknis Lingkungan Pengembangan:
  - [Spesifikasi Teknis Lingkungan Pengembangan Universitas XYZ]
  - Fungsi: Memberikan spesifikasi teknis tentang infrastruktur dan lingkungan pengembangan yang akan digunakan dalam mengembangkan sistem administrasi Tugas Akhir.
4. Analisis Kebutuhan Pengguna Tugas Akhir:
  - [Laporan Analisis Kebutuhan Pengguna Tugas Akhir, Edisi Terbaru]
  - Fungsi: Menggambarkan hasil analisis kebutuhan pengguna Tugas Akhir melalui wawancara, survei, atau pemetaan proses bisnis.
5. Studi Kelayakan Proyek Administrasi Tugas Akhir:
  - [Dokumen Studi Kelayakan Proyek Administrasi Tugas Akhir]
  - Fungsi: Menyajikan hasil analisis ekonomi, teknis, operasional, dan jadwal proyek untuk memastikan kelayakan dan manfaat proyek.
6. Rencana Proyek Administrasi Tugas Akhir:
  - [Rencana Proyek Administrasi Tugas Akhir Universitas XYZ]
  - Fungsi: Mengandung jadwal, sumber daya, tanggung jawab, dan risiko proyek untuk memandu pengembangan sistem administrasi Tugas Akhir.
7. Dokumen Desain Sistem Administrasi Tugas Akhir:
  - [Dokumen Desain Sistem Administrasi Tugas Akhir, Edisi Terbaru]
  - Fungsi: Menyediakan desain sistem yang rinci, termasuk arsitektur, diagram aliran data, diagram kelas, dan antarmuka pengguna.
8. Manual Pengguna Sistem Administrasi Tugas Akhir:
  - [Manual Pengguna Sistem Administrasi Tugas Akhir Universitas XYZ]

- Fungsi: Memberikan panduan pengguna yang instruktif tentang penggunaan sistem, tata cara operasional, dan solusi atas masalah umum.

Catatan Tambahan:

- Semua dokumen referensi harus memperhatikan kebijakan dan standar yang berlaku di Universitas XYZ.
- Edisi terbaru dari dokumen-dokumen referensi harus diakses dan diperbarui sesuai dengan perubahan kebijakan atau prosedur yang mungkin terjadi.

## Bab II Overall Description

### 2.1 Organisations

1. Visi: Menciptakan masa depan di mana pengelolaan administrasi Tugas Akhir menjadi lebih efisien dan inovatif melalui penerapan teknologi informasi yang canggih. Fokus pada pengalaman terbaik dalam pengelolaan administrasi Tugas Akhir, meningkatkan efisiensi, transparansi, dan kualitas proses pengelolaan gaji pegawai di perusahaan tersebut.
2. Struktur Organisasi: Struktur organisasi mencakup berbagai peran utama yang bekerja bersama untuk mencapai visi dan misi proyek pengembangan aplikasi Sistem Pengelolaan Gaji Pegawai.
  - Manajer Proyek: Bertanggung jawab atas pengelolaan keseluruhan proyek, koordinasi tim, dan pengambilan keputusan strategis.
  - Analis Sistem: Bertanggung jawab atas analisis kebutuhan pengguna dan merancang spesifikasi sistem.
  - Pengembang Perangkat Lunak: Mengimplementasikan desain sistem dan mengembangkan aplikasi.
  - Desainer Antarmuka Pengguna: Fokus pada desain antarmuka pengguna yang intuitif dan responsif.
  - Administrator Database: Menangani manajemen dan pemeliharaan database yang digunakan oleh aplikasi.
  - Tim Pemasangan dan Pelatihan: Bertanggung jawab atas instalasi, pengujian, dan pelatihan pengguna terkait sistem.
3. Tanggung Jawab dan Wewenang: Setiap posisi dalam struktur organisasi memiliki tanggung jawab dan wewenang yang terdefinisi dengan baik.
  - Manajer Proyek: Pengelolaan proyek, koordinasi tim, dan pengambilan keputusan strategis.
  - Analis Sistem: Analisis kebutuhan pengguna dan merancang spesifikasi sistem.
  - Pengembang Perangkat Lunak: Implementasi desain sistem dan pengembangan aplikasi.
  - Desainer Antarmuka Pengguna: Fokus pada estetika dan fungsionalitas antarmuka pengguna.
  - Administrator Database: Manajemen dan pemeliharaan database yang efisien.
  - Tim Pemasangan dan Pelatihan: Instalasi, pengujian, dan pelatihan pengguna terkait sistem.

Dengan struktur organisasi yang jelas dan terstruktur, setiap anggota tim dapat berperan secara efektif untuk mencapai tujuan visi dan misi proyek pengembangan aplikasi Sistem Pengelolaan Gaji Pegawai di Universitas XYZ. Struktur ini mendukung kelancaran dan kesuksesan proyek secara keseluruhan.

### 2.2 Product Perspective

1. Konteks Umum: Perangkat lunak ini beroperasi di dalam konteks administrasi TA di lingkungan universitas. Ini berintegrasi dengan sistem administrasi universitas yang sudah ada.
2. Integrasi dengan Sistem Universitas: Perangkat lunak ini harus dapat berintegrasi dengan sistem-sistem lain yang digunakan dalam administrasi universitas, termasuk sistem keuangan, sistem pendaftaran, dan sistem informasi lainnya.



3. Keterkaitan dengan Basis Data: Perangkat lunak ini akan terkait erat dengan basis data yang menyimpan informasi tentang mahasiswa, staf pengajar, keuangan, dan informasi terkait TA. Ini memastikan keakuratan dan konsistensi data.
4. Antarmuka Pengguna: Perangkat lunak menyediakan antarmuka pengguna yang intuitif dan mudah digunakan bagi berbagai pemangku kepentingan, termasuk staf administrasi, dosen, mahasiswa, dan pengguna lainnya.
5. Ketergantungan pada Infrastruktur Teknologi: Perangkat lunak memerlukan infrastruktur teknologi yang memadai, termasuk perangkat keras dan perangkat lunak yang mendukung kebutuhan aplikasi ini.
6. Keamanan dan Privasi: Keamanan data dan privasi menjadi fokus utama. Perangkat lunak harus memastikan bahwa informasi sensitif, seperti data mahasiswa dan keuangan, dilindungi dengan baik.
7. Dukungan Pelatihan: Diperlukan dukungan pelatihan untuk pengguna agar dapat menggunakan perangkat lunak dengan efektif dan memaksimalkan manfaatnya.

#### Implikasi untuk Pengguna:

- Staf Administrasi: Kemudahan dalam mengelola informasi administrasi TA.
- Dosen: Akses cepat dan mudah ke informasi TA mahasiswa.
- Mahasiswa: Aksesibilitas informasi terkait TA dan kemajuan akademis.
- Pemangku Kepentingan Lain: Informasi yang relevan dan transparansi dalam administrasi TA.

Hubungan dengan Sistem: Perangkat lunak ini berinteraksi dengan sistem administrasi universitas dan mungkin harus terintegrasi dengan beberapa sistem lain sesuai kebutuhan.

Ketergantungan dengan Produk Lain: Dalam beberapa kasus, perangkat lunak mungkin bergantung pada perangkat lunak lain atau layanan pihak ketiga untuk fungsi tertentu seperti otentikasi pengguna atau layanan keamanan.

Keterbatasan Produk: Diperlukan ketersediaan dan integritas data yang tinggi, Dukungan teknis dan pemeliharaan sistem secara teratur diperlukan.

Dengan fokus pada integrasi, antarmuka pengguna yang intuitif, dan keamanan data, perangkat lunak ini dirancang untuk mendukung efisiensi administrasi TA di lingkungan universitas XYZ.

### 7.1 User Classes and Characteristics

#### 1. Pengguna Mahasiswa TA:

- Karakteristik: Pengguna mahasiswa TA adalah mahasiswa yang tengah menyelesaikan tugas akhir. Mereka menggunakan aplikasi untuk mengunggah dokumen TA, memantau status persetujuan, dan menerima informasi terkait administrasi TA.
- Hak Akses: Pengguna mahasiswa TA memiliki hak akses untuk mengunggah dokumen TA, melacak status persetujuan, dan menerima pemberitahuan terkait administrasi TA. Mereka juga dapat mengakses informasi pribadi terkait tugas akhir mereka.

#### 2. Pengguna Dosen Pembimbing:

- Karakteristik: Pengguna dosen pembimbing adalah dosen yang membimbing mahasiswa dalam menyelesaikan tugas akhir. Mereka menggunakan aplikasi untuk menyetujui proposal, memberikan feedback, dan memberikan nilai akhir pada TA.
  - Hak Akses: Pengguna dosen pembimbing memiliki hak akses untuk menyetujui proposal TA, memberikan feedback, memberikan nilai akhir, dan melihat informasi terkait mahasiswa yang mereka bimbing.
3. Pengguna Staf Administrasi TA:
- Karakteristik: Pengguna staf administrasi TA adalah staf administrasi universitas yang bertanggung jawab atas pengelolaan administrasi tugas akhir. Mereka menggunakan aplikasi untuk mengelola jadwal ujian, mengeluarkan pengumuman, dan mengelola data administrasi terkait tugas akhir.
  - Hak Akses: Pengguna staf administrasi TA memiliki hak akses untuk mengelola jadwal ujian, mengeluarkan pengumuman, dan mengelola data administrasi tugas akhir. Mereka juga dapat mengakses informasi terkait mahasiswa dan dosen pembimbing.

## 7.2 Operating Environment

1. Hardware Platform:
  - Server: Diperlukan server dengan spesifikasi tinggi yang mampu menghosting aplikasi Sistem Pengelolaan Administrasi TA. Server harus memiliki kecepatan prosesor, kapasitas RAM, ruang penyimpanan, dan koneksi jaringan yang memadai untuk menangani beban pengguna dan volume data yang diharapkan.
  - Komputer Klien: Pegawai, dosen pembimbing, dan staf administrasi membutuhkan komputer klien yang dapat terhubung ke server dan menjalankan aplikasi. Komputer ini harus memenuhi spesifikasi minimum, termasuk prosesor, RAM, dan kapasitas penyimpanan yang cukup.
2. Operating System and Version:
 

Server: Lingkungan server dapat menggunakan sistem operasi Linux (seperti Ubuntu Server, CentOS) atau Windows Server, tergantung pada kebijakan perusahaan dan kompatibilitas aplikasi.

  - Komputer Klien: Komputer klien dapat menjalankan berbagai sistem operasi, seperti Windows (misalnya Windows 10), macOS, atau Linux (seperti Ubuntu, Fedora), sesuai dengan preferensi individu pengguna.
3. Database Management System (DBMS):
  - DBMS seperti MySQL, PostgreSQL, atau Oracle digunakan untuk menyimpan dan mengelola data terkait administrasi TA, seperti informasi mahasiswa, dokumen tugas akhir, dan catatan evaluasi.
4. Web Server:
  - Aplikasi dijalankan melalui web server seperti Apache HTTP Server, Nginx, atau Microsoft IIS. Web server bertanggung jawab untuk menangani permintaan dari klien dan mengirimkan konten aplikasi web kepada pengguna.
5. Bahasa Pemrograman dan Framework:

- Aplikasi dapat dikembangkan menggunakan berbagai bahasa pemrograman seperti Java, Python, PHP, atau JavaScript. Penggunaan framework seperti Laravel, Django, atau Spring dapat mempercepat pengembangan dan menyediakan fungsionalitas tambahan yang diperlukan.

6. Browser Web:

- Aplikasi dirancang untuk diakses melalui berbagai browser web seperti Google Chrome, Mozilla Firefox, Microsoft Edge, atau Safari. Aplikasi harus diuji dan dijamin berfungsi dengan baik di berbagai browser web utama agar dapat diakses oleh pengguna dengan berbagai preferensi.

### 7.3 Design and Implementation Constrains

Dalam mengembangkan perangkat lunak untuk Sistem Pengelolaan Administrasi Tugas Akhir (TA) di Universitas XYZ, beberapa batasan desain dan implementasi perangkat lunak perlu diperhatikan.

Berikut adalah batasan-batasan tersebut:

1. Fungsionalitas: Batasan fungsionalitas perangkat lunak akan mencakup fitur dan kemampuan yang telah ditetapkan sesuai dengan kebutuhan inti. Fitur tambahan di luar cakupan awal mungkin harus dikecualikan untuk memastikan fokus pada kebutuhan utama.
2. Waktu dan Anggaran: Proyek ini terikat oleh batasan waktu dan anggaran yang telah ditentukan. Pengerjaan harus mematuhi jangka waktu yang ditetapkan dan memastikan penggunaan anggaran yang efisien.
3. Sumber Daya: Ketersediaan sumber daya, termasuk personel pengembangan, hardware, dan perangkat lunak, akan membatasi tingkat kompleksitas dan cakupan implementasi yang dapat dicapai.
4. Integrasi dengan Sistem Lain: Jika diperlukan integrasi dengan sistem lain di universitas, batasan dapat muncul terkait dengan kompatibilitas, protokol komunikasi, dan kemampuan integrasi yang dapat diimplementasikan.
5. Keamanan: Desain dan implementasi perangkat lunak harus memenuhi standar keamanan yang ditetapkan oleh universitas. Perlindungan data sensitif dan langkah-langkah keamanan teknis harus diterapkan sesuai kebijakan keamanan yang ada.
6. Lingkungan Operasional: Batasan dapat muncul terkait dengan lingkungan operasional, seperti sistem operasi yang didukung, browser web yang kompatibel, dan ketersediaan infrastruktur jaringan yang memadai.
7. Skalabilitas: Perangkat lunak harus dirancang dengan mempertimbangkan kemampuan untuk menangani pertumbuhan dan skala yang mungkin terjadi di masa depan, baik dari segi arsitektur maupun kinerja aplikasi.
8. Persyaratan Hukum dan Regulasi: Batasan harus mematuhi persyaratan hukum, regulasi, dan kebijakan yang mengatur pengelolaan administrasi TA. Kepatuhan terhadap privasi, keamanan data, dan aturan hukum lainnya harus menjadi prioritas.

#### 7.4 Assumptions and Dependencies

Dalam menyusun Software Requirements Specification (SRS) untuk perangkat lunak pengelolaan administrasi Tugas Akhir (TA) di Universitas XYZ, beberapa asumsi dan ketergantungan perlu diperhatikan. Berikut adalah asumsi dan ketergantungan yang relevan:

1. Keterbatasan Sumber Daya:
  - Asumsi: Proyek pengembangan perangkat lunak TA dapat dilaksanakan sesuai alokasi sumber daya yang telah disetujui, termasuk anggaran, waktu, dan tenaga kerja.
  - Ketergantungan: Desain dan implementasi perangkat lunak harus memperhitungkan keterbatasan sumber daya yang mungkin mempengaruhi cakupan fitur dan jangka waktu pengembangan.
2. Perubahan Kebijakan atau Regulasi:
  - Asumsi: Kebijakan dan regulasi universitas yang berlaku pada saat penyusunan SRS tetap konsisten selama siklus pengembangan.
  - Ketergantungan: Jika ada perubahan dalam kebijakan atau regulasi universitas terkait administrasi TA, perangkat lunak harus dapat disesuaikan untuk memenuhi persyaratan baru.
3. Teknologi atau Standar Baru:
  - Asumsi: Teknologi atau standar yang relevan dengan perangkat lunak TA tetap konsisten selama siklus pengembangan.
  - Ketergantungan: Jika ada kemajuan teknologi atau perubahan standar industri yang berpengaruh pada perangkat lunak, sistem harus dapat diadaptasi untuk memanfaatkan perkembangan tersebut.
4. Perubahan Kebutuhan Pengguna:
  - Asumsi: Kebutuhan pengguna terkait administrasi TA yang ditetapkan pada awal proyek tetap stabil, kecuali ada kebutuhan yang memang perlu direvisi.
  - Ketergantungan: Jika ada perubahan dalam kebutuhan pengguna, perangkat lunak harus mampu menyesuaikan diri untuk memenuhi perubahan tersebut.
5. Penemuan Baru atau Informasi Tambahan:
  - Asumsi: Informasi yang diperoleh pada saat penyusunan SRS mencakup pemahaman yang komprehensif tentang kebutuhan dan desain sistem administrasi TA.
  - Ketergantungan: Jika ada penemuan baru atau informasi tambahan yang signifikan, perangkat lunak harus dapat direvisi sesuai dengan temuan tersebut.
6. Perubahan Prioritas Bisnis:
  - Asumsi: Prioritas bisnis terkait administrasi TA yang dinyatakan pada awal proyek tetap konsisten, kecuali ada perubahan signifikan dalam kondisi pasar atau lingkungan universitas.
  - Ketergantungan: Jika terjadi perubahan dalam prioritas bisnis, perangkat lunak harus dapat menyesuaikan fokus dan cakupan fiturnya.
7. Faktor Eksternal yang Tidak Terduga:
  - Asumsi: Tidak ada faktor eksternal yang tidak terduga yang secara signifikan mempengaruhi proyek pengembangan perangkat lunak administrasi TA.
  - Ketergantungan: Jika terjadi kejadian tidak terduga, perangkat lunak harus dapat menanggapi perubahan kondisi yang tak terduga tersebut.



### Bab III Functional Requirements

Kebutuhan Fungsional adalah kebutuhan yang berisi proses/layanan yang harus disediakan oleh perangkat lunak mencakup bagaimana system harus bereaksi pada input tertentu dan bagaimana perilaku system pada situasi tertentu.

#### 3.1 Detailed Functional Requirements

##### **Functional Requirement 1: Pengelolaan Data Mahasiswa TA**

###### **1) Pendaftaran Tugas Akhir:**

1. Deskripsi: Sistem harus menyediakan fungsionalitas pendaftaran tugas akhir bagi mahasiswa. Ini mencakup pengisian data mahasiswa, judul tugas akhir, dan pemilihan pembimbing.
2. Proses Bisnis:
  - Mahasiswa mengakses sistem informasi administrasi TA.
  - Mahasiswa mengisi formulir pendaftaran tugas akhir, menyertakan judul dan memilih pembimbing.
  - Sistem menyimpan dan memvalidasi data pendaftaran.

###### **2) Otorisasi Akses Pendaftaran :**

1. Deskripsi: Hanya mahasiswa yang bersangkutan, pembimbing, dan pihak administrasi yang memiliki hak akses untuk melihat dan memproses data pendaftaran tugas akhir.
2. Proses Bisnis:
  - Sistem harus memastikan bahwa hanya mahasiswa yang bersangkutan yang dapat melihat dan memperbarui data pendaftaran tugas akhir mereka.
  - Pembimbing memiliki hak akses untuk melihat dan menyetujui pendaftaran tugas akhir mahasiswa.

##### **Functional Requirement 2: Penilaian dan Evaluasi Tugas Akhir**

###### **1) Pengisian Nilai oleh Penguji:**

1. Deskripsi: Bagian Keuangan dapat mengunggah slip gaji sebagai bukti penggajian, kemudian pegawai dapat mengunduh slip gaji.
2. Proses Bisnis:
  - Bagian Keuangan mengunggah slip gaji yang telah dibuat untuk diserahkan ke pegawai.
  - Sistem menyimpan data slip gaji.
  - Pegawai mengunduh slip gaji.

###### **2) Kalkulasi Nilai Akhir:**

1. Deskripsi: Sistem harus dapat menghitung nilai akhir tugas akhir berdasarkan nilai yang diberikan oleh penguji.
2. Proses Bisnis:
  - Sistem secara otomatis menghitung nilai akhir tugas akhir berdasarkan kriteria penilaian yang telah ditentukan.
  - Sistem menyimpan nilai akhir tersebut untuk dilihat oleh mahasiswa dan pihak terkait.

##### **Functional Requirement 2: Pengelolaan Pengumuman dan Informasi TA**

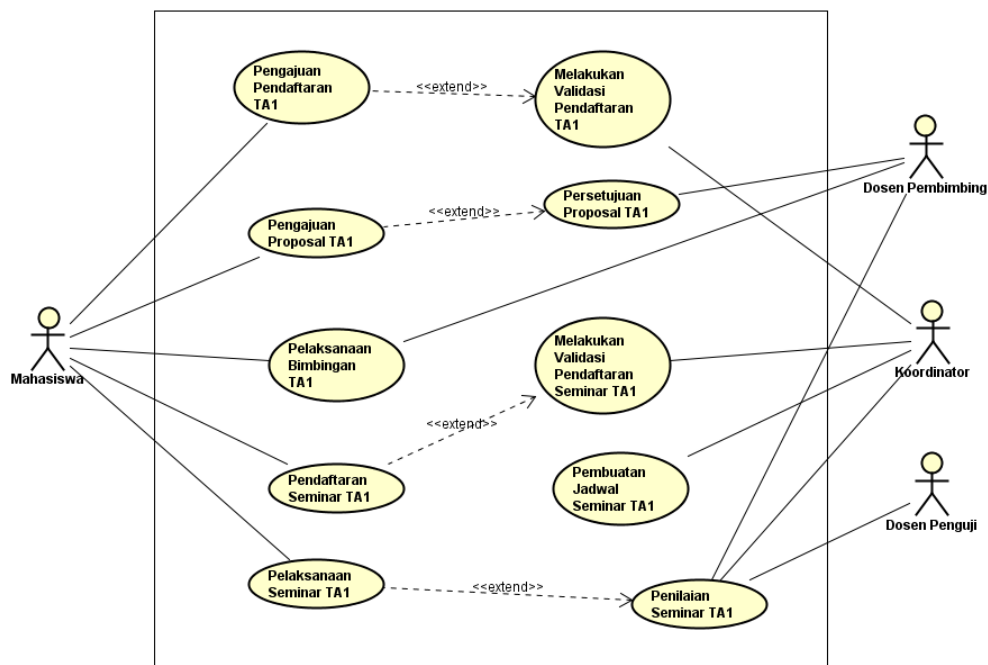
### 1) Pengumuman Hasil Tugas Akhir:

1. Deskripsi: Sistem harus memungkinkan pihak administrasi mengumumkan hasil tugas akhir kepada mahasiswa.
2. Proses Bisnis:
  - Pihak administrasi mengakses sistem informasi administrasi TA.
  - Pihak administrasi memasukkan dan mengonfirmasi hasil tugas akhir.
  - Sistem mengirimkan pengumuman hasil tugas akhir kepada mahasiswa.

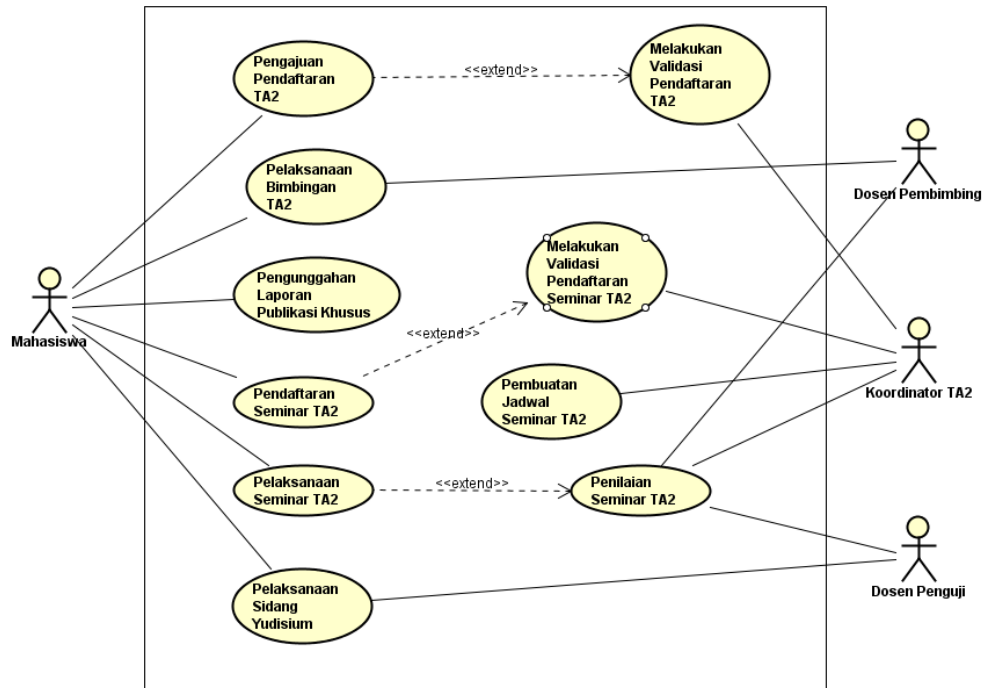
### 2) Akses Informasi TA:

1. Deskripsi: Mahasiswa harus dapat mengakses informasi terkait tugas akhir, seperti jadwal sidang, informasi pembimbing, dan panduan penulisan.
2. Proses Bisnis:
  - Mahasiswa mengakses sistem informasi administrasi TA.
  - Mahasiswa melihat jadwal sidang, informasi pembimbing, dan materi panduan penulisan yang disediakan oleh sistem.

## 3.2 Use Case Diagram



Gambar 1 Use Case Diagram Pelaksanaan TA1



Gambar 2 Use Case Diagram Pelaksanaan TA2 dan Sidang Yudisium

### 3.3 Use Case Scenario

#### Use Case 1

Nama Use Case : Pengajuan Pendaftaran TA1.  
 Aktor Utama : Mahasiswa (calon peserta TA1)  
 Tujuan : Calon peserta TA1 mendaftarkan diri untuk melaksanakan TA1.  
 Aktor Pendukung : Koordinator TA1  
 Kondisi Sebelum : Mahasiswa belum terdaftar untuk melaksanakan TA1, telah memilih dosen pembimbing dan tema proposal TA1.  
 Kondisi Setelah : Mahasiswa yang memenuhi persyaratan telah terdaftar menjadi peserta TA1.

#### Use Case 2

Nama Use Case : Melakukan Validasi Pendaftaran TA1.  
 Aktor Utama : Koordinator TA1  
 Tujuan : Koordinator TA1 melakukan validasi terhadap pendaftaran TA1.  
 Aktor Pendukung : Mahasiswa  
 Kondisi Sebelum : Koordinator belum melakukan pengecekan data mahasiswa pendaftar TA1 (meliputi biodata, jumlah sks, riwayat pembayaran, dan nilai mahasiswa).



Kondisi Setelah : Koordinator telah melakukan pengecekan data mahasiswa pendaftar TA1, koordinator telah mendapat keputusan apakah data mahasiswa valid atau tidak.

### Use Case 3

Nama Use Case : Pengajuan Proposal TA1.  
Aktor Utama : Mahasiswa (Peserta TA1)  
Tujuan : Mahasiswa peserta TA1 mengajukan proposal TA1 sehingga dapat maju ke tahap bimbingan.  
Aktor Pendukung : Dosen pembimbing  
Kondisi Sebelum : Proposal TA1 mahasiswa belum diajukan ke dosen pembimbing.  
Kondisi Setelah : Proposal TA1 mahasiswa telah diajukan ke dosen pembimbing.

### Use Case 4

Nama Use Case : Persetujuan Proposal TA1.  
Aktor Utama : Dosen Pembimbing  
Tujuan : Dosen pembimbing memberi keputusan apakah proposal TA1 yang diajukan mahasiswa disetujui atau tidak.  
Aktor Pendukung : Mahasiswa  
Kondisi Sebelum : Dosen pembimbing belum memberi keputusan mengenai kelayakan proposal TA1.  
Kondisi Setelah : Dosen pembimbing telah memberi keputusan mengenai kelayakan proposal TA1.

### Use Case 5

Nama Use Case : Melakukan bimbingan TA1.  
Aktor Utama : Mahasiswa (Peserta TA1)  
Tujuan : Mahasiswa melakukan bimbingan TA1 untuk memenuhi persyaratan seminar TA1.  
Aktor Pendukung : Dosen pembimbing  
Kondisi Sebelum : Proposal TA1 telah disetujui, mahasiswa belum mendapatkan bimbingan dosen pembimbing.  
Kondisi Setelah : Mahasiswa telah mendapatkan bimbingan dosen pembimbing.

### Use Case 6

Nama Use Case : Pendaftaran Seminar TA1.  
Aktor Utama : Mahasiswa (Peserta TA1)  
Tujuan : Calon peserta seminar TA1 mendaftarkan diri untuk melaksanakan seminar TA1.  
Aktor Pendukung : Koordinator TA1

Kondisi Sebelum	:	Mahasiswa belum menjadi peserta seminar TA1. Sudah melakukan bimbingan minimal 6x dan melampirkan surat kelayakan maju seminar.
Kondisi Setelah	:	Mahasiswa sudah menjadi peserta seminar TA1.

#### Use Case 7

Nama Use Case	:	Melakukan Validasi Pendaftaran Seminar TA1.
Aktor Utama	:	Koordinator TA1
Tujuan	:	Koordinator TA1 melakukan validasi terhadap pendaftaran Seminar TA1.
Aktor Pendukung	:	Mahasiswa
Kondisi Sebelum	:	Koordinator belum melakukan pengecekan terhadap bimbingan TA1 yang dilakukan dan surat kelayakan maju seminar.
Kondisi Setelah	:	Koordinator telah melakukan pengecekan data mahasiswa, koordinator telah mendapat keputusan apakah data mahasiswa valid atau tidak.

#### Use Case 8

Nama Use Case	:	Pembuatan Jadwal Seminar TA1.
Aktor Utama	:	Koordinator TA1
Tujuan	:	Membuat jadwal pelaksanaan seminar TA1.
Aktor Pendukung	:	Mahasiswa
Kondisi Sebelum	:	Jadwal pelaksanaan seminar TA1 belum dibuat.
Kondisi Setelah	:	Jadwal pelaksanaan seminar TA1 telah dibuat.

#### Use Case 9

Nama Use Case	:	Pelaksanaan Seminar TA1.
Aktor Utama	:	Mahasiswa (Peserta TA1)
Tujuan	:	Peserta melaksanakan seminar TA1 untuk mendapatkan nilai seminar TA1.
Aktor Pendukung	:	Koordinator TA1, dosen pembimbing, dosen penguji
Kondisi Sebelum	:	Mahasiswa telah terdaftar seminar TA1, mahasiswa telah mendapat jadwal seminar TA1.
Kondisi Setelah	:	Mahasiswa mendapatkan hasil penilaian dari total perhitungan nilai dosen pembimbing, dosen penguji, dan koordinator.

#### Use Case 10

Nama Use Case	:	Penilaian Seminar TA1.
Aktor Utama	:	Dosen Pembimbing, dosen penguji, koordinator TA1

Tujuan	: Dosen pembimbing, dosen penguji, dan koordinator TA1 melakukan penilaian terhadap seminar TA1 mahasiswa.
Aktor Pendukung	: Mahasiswa
Kondisi Sebelum	: Mahasiswa telah mengikuti seminar TA1, dosen dan koordinator belum memberikan nilai TA1 kepada mahasiswa.
Kondisi Setelah	: Dosen dan koordinator telah memberikan nilai TA1 kepada mahasiswa.

#### **Use Case 11**

Nama Use Case	: Pengajuan Pendaftaran TA2.
Aktor Utama	: Mahasiswa (calon peserta TA2)
Tujuan	: Calon Peserta TA2 mendaftarkan diri untuk melaksanakan TA2.
Aktor Pendukung	: Koordinator TA2
Kondisi Sebelum	: Mahasiswa belum terdaftar untuk melaksanakan TA2, mahasiswa sudah melaksanakan TA1.
Kondisi Setelah	: Mahasiswa yang memenuhi persyaratan telah terdaftar menjadi peserta TA2.

#### **Use Case 12**

Nama Use Case	: Melakukan Validasi Pendaftaran TA2.
Aktor Utama	: Koordinator TA2
Tujuan	: Koordinator TA2 melakukan validasi terhadap pendaftaran TA2.
Aktor Pendukung	: Mahasiswa
Kondisi Sebelum	: Koordinator belum melakukan pengecekan data mahasiswa pendaftar TA1 (meliputi biodata, jumlah sks, masa berlaku judul, riwayat pembayaran, dan nilai mahasiswa).
Kondisi Setelah	: Koordinator telah melakukan pengecekan data mahasiswa pendaftar TA2, koordinator telah mendapat keputusan apakah data mahasiswa valid atau tidak.

#### **Use Case 13**

Nama Use Case	: Melakukan bimbingan TA2.
Aktor Utama	: Mahasiswa (Peserta TA2)
Tujuan	: Memenuhi persyaratan seminar TA2.
Aktor Pendukung	: Dosen pembimbing
Kondisi Sebelum	: Mahasiswa belum mendapatkan bimbingan dari dosen pembimbing.
Kondisi Setelah	: Mahasiswa telah mendapatkan bimbingan dari dosen pembimbing.

#### **Use Case 14**

Nama Use Case	: Pengunggahan Laporan Publikasi Khusus.
Aktor Utama	: Mahasiswa (Peserta TA2)
Tujuan	: Melaporkan publikasi khusus sehingga mendapat nilai TA2 dengan indeks A tanpa mengikuti seminar TA2 jika laporannya layak.
Aktor Pendukung	: Koordinator TA2
Kondisi Sebelum	: Mahasiswa telah melakukan publikasi khusus, mahasiswa belum melaporkan publikasi khusus.
Kondisi Setelah	: Mahasiswa telah melaporkan publikasi khusus.

#### Use Case 15

Nama Use Case	: Pendaftaran Seminar TA2.
Aktor Utama	: Mahasiswa (Peserta TA2)
Tujuan	: Calon peserta TA2 mendaftarkan diri untuk melaksanakan seminar TA2.
Aktor Pendukung	: Koordinator TA2
Kondisi Sebelum	: Mahasiswa belum menjadi peserta seminar TA2. Sudah melakukan bimbingan minimal 8x, melampirkan surat kelayakan maju seminar dan surat bebas plagiasi.
Kondisi Setelah	: Mahasiswa telah menjadi peserta seminar TA2.

#### Use Case 16

Nama Use Case	: Melakukan Validasi Pendaftaran Seminar TA2.
Aktor Utama	: Koordinator Seminar TA2
Tujuan	: Koordinator TA2 melakukan validasi terhadap pendaftaran Seminar TA2.
Aktor Pendukung	: Mahasiswa
Kondisi Sebelum	: Koordinator belum melakukan pengecekan terhadap bimbingan TA2 yang dilakukan dan surat kelayakan maju seminar.
Kondisi Setelah	: Koordinator telah melakukan pengecekan data mahasiswa, koordinator telah mendapat keputusan apakah data mahasiswa valid atau tidak.

#### Use Case 17

Nama Use Case	: Pembuatan Jadwal Seminar TA2.
Aktor Utama	: Koordinator TA2
Tujuan	: Membuat jadwal pelaksanaan seminar TA2.
Aktor Pendukung	: Mahasiswa
Kondisi Sebelum	: Jadwal pelaksanaan seminar TA2 belum dibuat.
Kondisi Setelah	: Jadwal pelaksanaan seminar TA2 telah dibuat.

#### **Use Case 18**

Nama Use Case	: Pelaksanaan Seminar TA2.
Aktor Utama	: Mahasiswa (Peserta TA2)
Tujuan	: Peserta melaksanakan seminar TA2 untuk mendapatkan nilai seminar TA2.
Aktor Pendukung	: Koordinator TA2, dosen pembimbing, dosen penguji
Kondisi Sebelum	: Mahasiswa telah terdaftar seminar TA2, mahasiswa telah mendapat jadwal seminar TA2.
Kondisi Setelah	: Mahasiswa mendapatkan hasil penilaian dari total perhitungan nilai dosen pembimbing, dosen penguji, dan koordinator.

#### **Use Case 19**

Nama Use Case	: Penilaian Seminar TA2.
Aktor Utama	: Dosen Pembimbing, dosen penguji, dan koordinator TA2
Tujuan	: Dosen pembimbing, dosen penguji, dan koordinator TA2 melakukan penilaian terhadap seminar TA2 mahasiswa.
Aktor Pendukung	: Mahasiswa
Kondisi Sebelum	: Mahasiswa telah mengikuti seminar TA2, dosen dan koordinator belum memberikan nilai TA1 kepada mahasiswa.
Kondisi Setelah	: Dosen dan koordinator telah memberikan nilai TA2 kepada mahasiswa.

#### **Use Case 20**

Nama Use Case	: Pelaksanaan Sidang Yudisium.
Aktor Utama	: Mahasiswa (Peserta Sidang Yudisium)
Tujuan	: Peserta melaksanakan sidang yudisium untuk mendapatkan status kelulusan.
Aktor Pendukung	: Dosen penguji
Kondisi Sebelum	: Mahasiswa telah melaksanakan seminar TA2 dan melakukan publikasi ilmiah.
Kondisi Setelah	: Mahasiswa mendapatkan status kelulusan.

## Bab IV Non Functional Requirements

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

### 4.1 Performance Requirements

Berikut beberapa contoh kebutuhan performa yang diinginkan dalam aplikasi Administrasi TA Universitas XYZ:

1. Waktu Respons:
  - Aplikasi harus memberikan respons terhadap permintaan pengguna dalam waktu kurang dari 2 detik.
  - Selama jam kerja normal, waktu respons aplikasi tidak boleh melebihi 1 detik untuk tindakan umum seperti mengakses data administrasi tugas akhir atau menghasilkan laporan.
2. Skalabilitas:
  - Aplikasi harus mampu menangani pertumbuhan data dan pengguna di lingkungan administrasi tugas akhir universitas.
  - Kemampuan skalabilitas harus dioptimalkan untuk mengelola data administrasi tugas akhir hingga setidaknya 1000 mahasiswa secara bersamaan tanpa penurunan kinerja yang signifikan.
3. Ketersediaan:
  - Ketersediaan aplikasi harus mencapai setidaknya 99,9% setiap bulan.
  - Downtime yang dijadwalkan harus terbatas pada periode waktu yang paling tidak berdampak pada pengguna, seperti di tengah malam atau selama periode minimum penggunaan.
4. Kecepatan Pemuatan Halaman:
  - Waktu pemuatan halaman aplikasi web, terutama yang berisi informasi administrasi tugas akhir, harus kurang dari 3 detik.
  - File dokumen administrasi tugas akhir, seperti formulir pengajuan TA, harus dapat diunduh dalam waktu kurang dari 5 detik.
5. Kinerja Simultan Pengguna:
  - Aplikasi harus dapat menangani setidaknya 100 pengguna yang aktif secara bersamaan tanpa mengurangi kinerja secara signifikan.
  - Performa aplikasi tidak boleh menurun secara nyata bahkan ketika pengguna sedang mengakses informasi tugas akhir atau melakukan tindakan yang membutuhkan sumber daya komputasi yang tinggi.
6. Monitoring Kinerja:
  - Sistem harus dilengkapi dengan mekanisme pemantauan kinerja untuk memantau penggunaan sumber daya seperti CPU, RAM, dan penyimpanan.
  - Pemantauan kinerja harus dapat memberikan laporan berkala dan peringatan saat batas kinerja mendekati atau tercapai.

## 4.2 Safety Requirements

Berikut beberapa contoh kebutuhan keamanan yang diinginkan dalam aplikasi Administrasi TA Universitas XYZ:

1. Perlindungan Data:
  - Seluruh data administrasi TA, termasuk informasi pribadi mahasiswa dan dosen, harus dienkripsi saat disimpan dalam database dan saat berpindah melalui jaringan.
  - Sistem harus menggunakan standar enkripsi yang memadai dan terkini untuk melindungi kerahasiaan dan integritas data.
2. Otorisasi Akses:
  - Sistem harus memiliki mekanisme autentikasi yang kuat untuk memverifikasi identitas pengguna sebelum memberikan akses ke data administrasi TA.
  - Otorisasi akses harus diimplementasikan secara ketat, memastikan bahwa setiap pengguna hanya dapat mengakses informasi sesuai dengan peran dan tanggung jawabnya dalam administrasi TA.
3. Audit Trail:
  - Aplikasi harus mencatat log kejadian yang terkait dengan administrasi TA, termasuk perubahan data mahasiswa, pengaksesan informasi oleh pengguna, dan aktivitas sistem lainnya.
  - Informasi log harus mencakup timestamp, deskripsi kejadian, dan identifikasi pengguna yang terlibat.
4. Pemulihan Bencana:
  - Sistem harus memiliki prosedur pemulihan bencana terencana untuk melindungi data administrasi TA dan memastikan ketersediaan sistem dalam skenario bencana atau kegagalan perangkat keras.
  - Backup data secara rutin dan penyimpanan backup harus terpisah fisik dari server utama untuk menghindari risiko kehilangan data.
5. Pencegahan Serangan Keamanan:
  - Aplikasi harus dilengkapi dengan sistem perlindungan terhadap serangan siber dan ancaman keamanan lainnya.
  - Sistem harus diperbarui secara berkala untuk menanggapi ancaman keamanan baru yang mungkin muncul.

## 4.3 Software Quality Attributes

Berikut beberapa contoh kebutuhan kualitas perangkat lunak yang diinginkan dalam aplikasi Administrasi TA Universitas XYZ:

1. Usabilitas
  - Deskripsi: Antarmuka pengguna harus dirancang dengan baik dan mudah dipahami.
  - Kriteria: Pengguna harus dapat dengan mudah mengakses dan memasukkan data terkait administrasi TA, menjalankan proses manajemen TA dengan intuitif, dan menggunakan fitur-fitur sistem tanpa kesulitan.

- Metrik: Pengukuran waktu yang dibutuhkan untuk menyelesaikan tugas-tugas umum, tingkat kepuasan pengguna, dan tingkat kesalahan input.
2. (Skalabilitas):
- Deskripsi: Aplikasi harus dapat tumbuh sejalan dengan perkembangan administrasi TA dan jumlah data yang dielola.
  - Kriteria: Aplikasi harus dapat menangani pertumbuhan data dan pengguna tanpa mengurangi kinerja atau menimbulkan downtime.
  - Metrik: Kemampuan aplikasi untuk menangani peningkatan jumlah pengguna atau data tanpa pengurangan kinerja yang signifikan.
3. Maintainability (Kemudahan Pemeliharaan):
- Deskripsi: Aplikasi harus mudah untuk dikelola, diperbarui, dan disesuaikan dengan perubahan kebijakan atau kebutuhan administrasi TA.
  - Kriteria: Pembaruan perangkat lunak, penyesuaian kebijakan, atau integrasi dengan sistem lain harus dapat dilakukan tanpa gangguan besar.
  - Metrik: Waktu yang dibutuhkan untuk menerapkan pembaruan, keberhasilan implementasi perubahan, dan jumlah masalah setelah pembaruan.
- 8 Reliability (Keandalan):
- Deskripsi: Aplikasi harus memberikan kinerja yang stabil dan handal.
  - Kriteria: Sistem harus dapat diandalkan untuk memberikan informasi administrasi TA dengan benar dan konsisten.
  - Metrik: Tingkat kegagalan sistem, waktu penyediaan sistem, dan tingkat kepuasan pengguna terkait keandalan aplikasi.
- 9 Testability (Uji Kemampuan):
- Deskripsi: Aplikasi harus mudah diuji untuk memastikan fungsionalitas dan kualitas yang baik.
  - Kriteria: Pengujian unit, integrasi, dan fungsional harus dapat diimplementasikan secara menyeluruh.
  - Metrik: Persentase kode yang dicakup oleh pengujian otomatis, jumlah bug yang ditemukan selama pengujian, dan keberhasilan uji implementasi.



## Bab V Data Requirements

### 4.1 Input

#### 1. Data Mahasiswa TA:

##### a) Akses:

- Mahasiswa TA: Create, Read, Update (CRU)
- Koordinator TA dan Admin: Create, Read, Update, Delete (CRUD)

##### b) Atribut:

- Nama Mahasiswa (Teks): Informasi lengkap mengenai nama mahasiswa TA.
- NIM (Teks): Nomor Induk Mahasiswa.
- Program Studi (Teks): Program studi yang diikuti oleh mahasiswa.
- Judul TA (Teks): Judul dari tugas akhir yang sedang dikerjakan.
- Dosen Pembimbing (Teks): Nama dosen pembimbing yang ditugaskan.
- Status TA (Teks): Status pengerjaan tugas akhir (e.g., dalam pengerjaan, selesai).
- Nilai TA (Numerik): Nilai atau penilaian yang diberikan setelah selesai.

#### 2. Data Progres TA:

##### a) Akses:

- Mahasiswa TA dan Koordinator TA: Create, Read, Update (CRU)

##### b) Atribut:

- Tanggal Progres (Tanggal): Tanggal pelaporan progres oleh mahasiswa.
- Deskripsi Progres (Teks): Catatan atau deskripsi mengenai progres yang telah dicapai.
- Persentase Selesai (Numerik): Persentase progres yang telah diselesaikan.

#### 3. Data Jurnal TA:

##### a) Akses:

- Mahasiswa TA: Read (R)
- Koordinator TA dan Admin: Create, Read, Update, Delete (CRUD)

##### b) Atribut:

Tanggal Jurnal (Tanggal): Tanggal penulisan jurnal oleh mahasiswa.

Isi Jurnal (Teks): Konten atau tulisan dalam jurnal.

### 4.2 Output

#### 1. Laporan Progres Tugas Akhir Bulanan:

a. Pengguna Laporan: Dosen Pembimbing, Bagian Akademik, dan Mahasiswa Terkait.

b. Atribut Laporan:

- Nama Mahasiswa (Teks): Nama lengkap dari mahasiswa yang sedang menjalani Tugas Akhir (TA).

- Nomor Identitas Mahasiswa (Teks): Nomor identitas mahasiswa terkait dengan TA.
- Program Studi (Teks): Program studi yang diambil oleh mahasiswa.
- Judul Tugas Akhir (Teks): Judul dari Tugas Akhir yang sedang dikerjakan.
- Dosen Pembimbing (Teks): Nama dosen pembimbing yang ditetapkan untuk TA.
- Status TA (Teks): Status Tugas Akhir, misalnya "Dalam Proses," "Selesai," atau "Dibatalkan."
- Nilai TA (Numerik): Nilai atau hasil dari Tugas Akhir yang telah diselesaikan.
- Catatan Progres (Teks): Catatan atau komentar terkait dengan progres TA mahasiswa.

## 2. Laporan Kehadiran Mahasiswa:

- Pengguna Laporan: Dosen Pembimbing, Bagian Akademik, dan Mahasiswa Terkait.
- Atribut Laporan:
  - Tanggal (Tanggal): Tanggal pelaksanaan atau pertemuan progres TA.
  - Waktu Mulai (Waktu): Waktu mulai kegiatan TA atau pertemuan dengan dosen pembimbing.
  - Waktu Selesai (Waktu): Waktu selesai kegiatan atau pertemuan dengan dosen pembimbing.
  - Total Jam Kerja (Numerik): Total jam kerja atau durasi kegiatan TA pada tanggal tersebut.
  - Jumlah Jam Terlambat (Numerik): Jumlah jam terlambat dari waktu yang telah ditetapkan.
  - Jumlah Jam Lembur (Numerik): Jumlah jam lembur, jika ada.

## 3. Laporan Evaluasi Dosen Pembimbing:

- Pengguna Laporan: Mahasiswa dan Bagian Akademik.
- Atribut Laporan:
  - Nama Dosen Pembimbing (Teks): Nama lengkap dari dosen pembimbing.
  - Penilaian Kinerja (Numerik): Nilai atau penilaian kinerja dosen pembimbing dari mahasiswa.
  - Umpan Balik (Teks): Umpan balik atau komentar mahasiswa terkait dengan dosen pembimbing.

- Rekomendasi Perbaikan (Teks): Rekomendasi perbaikan atau saran dari mahasiswa untuk dosen pembimbing.

Catatan:

- Teks: Representasi data dalam bentuk teks atau kata-kata.
- Numerik: Representasi data dalam bentuk angka atau nilai numerik.
- Tanggal: Representasi data dalam bentuk tanggal.
- Waktu: Representasi data dalam bentuk waktu.

## Bab VI Interface Requirements

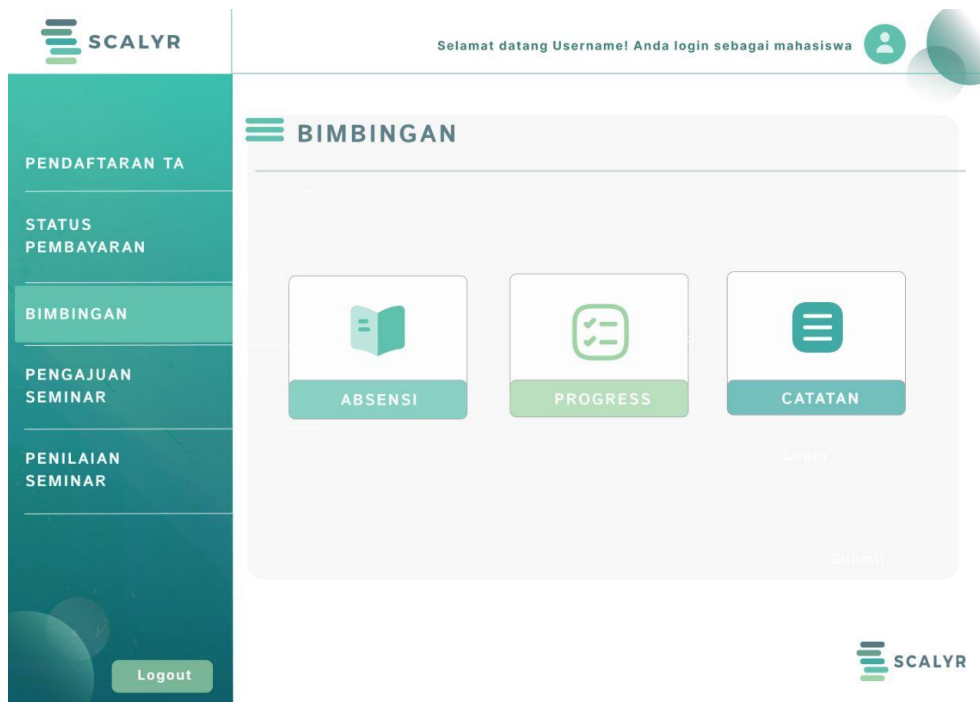
### 4.1 User Interface

The login screen features a teal background with a white arrow icon pointing right. The SCALYR logo is in the top left corner. The main heading is "Selamat Datang di Aplikasi SCALYR!". Below it, a paragraph reads: "Silahkan lakukan log-in jika sudah memiliki akun akademik SCALYR. Jika belum, daftar terlebih dahulu pada bagian akademik kampus." The login form is centered and includes fields for "Username" and "Password", a "Login" button, and a "Forgot Password" link. The SCALYR logo is also in the bottom right corner.

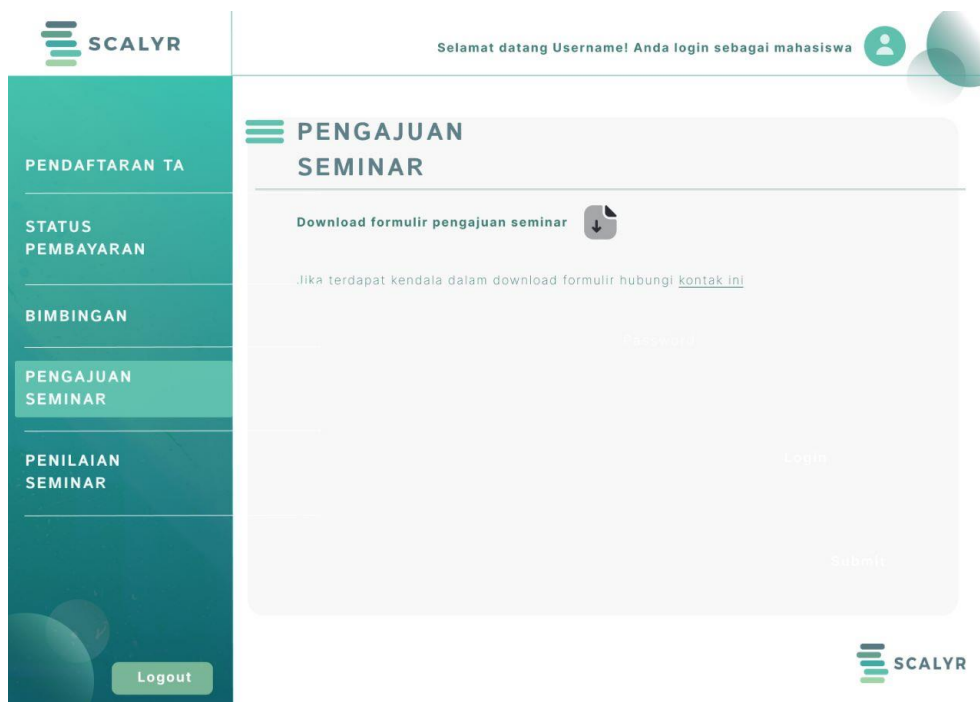
Gambar 3 Login

The registration screen has a teal sidebar on the left with the SCALYR logo and a "Logout" button. The main area is white with a teal header bar containing the SCALYR logo and a welcome message: "Selamat datang Username! Anda login sebagai mahasiswa". The registration form is titled "FORM PENDAFTARAN TA" and includes fields for "NAMA LENGKAP", "NIM", "PROGRAM STUDI" (a dropdown menu showing "Informatika"), "TANGGAL LAHIR" (with DD, MM, and YY dropdowns), "SEMESTER" (a dropdown menu showing "7"), "No. Telp/HP", and "Email". A "Submit" button is at the bottom right. The SCALYR logo is in the bottom right corner.

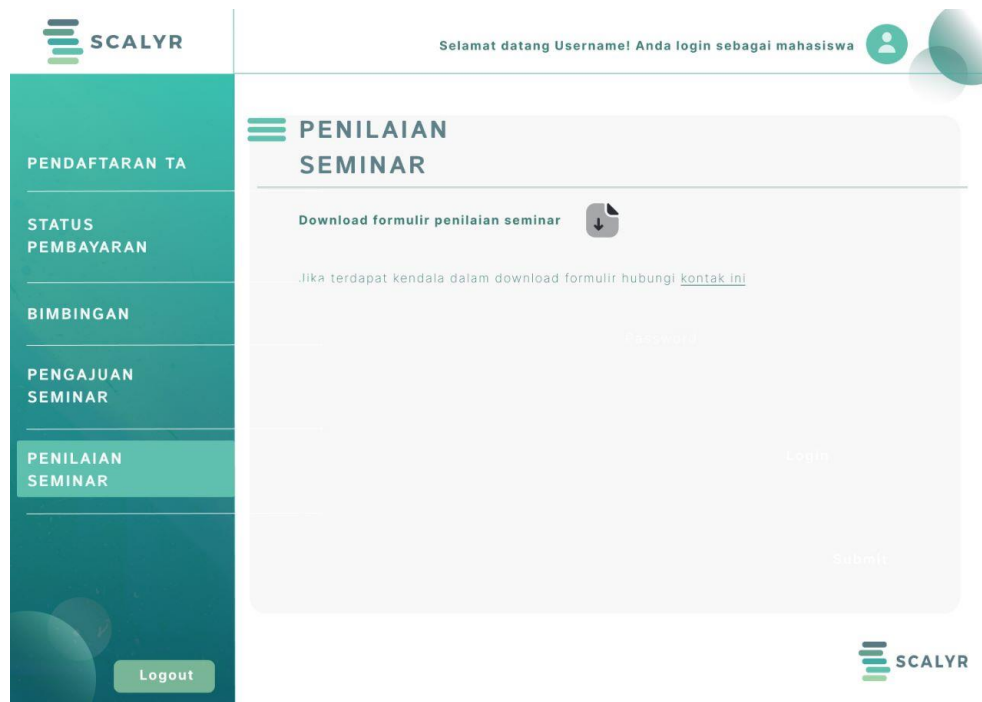
Gambar 4 Pendaftaran TA



Gambar 5 Bimbingan



Gambar 6 Pengajuan Seminar



Gambar 7 Penilaian Seminar

## 4.2 Hardware Interface

### 1. Server Database:

- Deskripsi: Aplikasi akan berinteraksi dengan server database untuk menyimpan dan mengakses data terkait administrasi Tugas Akhir (TA) mahasiswa.
- Koneksi: Koneksi yang aman dan dikonfigurasi dengan baik antara aplikasi dan server database.
- Keamanan: Perlindungan data yang kuat dan implementasi protokol keamanan untuk mengamankan data gaji dan informasi administrasi TA.

### 2. Komputer/Komputer Server:

- Deskripsi: Aplikasi pengelolaan administrasi TA dijalankan pada komputer atau komputer server yang menjadi host perangkat lunak.
- Instalasi: Proses instalasi aplikasi pada komputer/komputer server sesuai dengan persyaratan sistem aplikasi.
- Pemeliharaan: Ketersediaan perangkat keras yang memadai dan pemeliharaan yang teratur untuk memastikan kinerja aplikasi yang optimal.

### 3. Jaringan (Network):

- Deskripsi: Antarmuka jaringan memungkinkan akses aplikasi pengelolaan administrasi TA dari berbagai lokasi dan departemen di Universitas XYZ.

- Tipe Jaringan: Koneksi ke jaringan lokal dan/atau jaringan yang lebih luas untuk mendukung penggunaan aplikasi dari berbagai lokasi.
- Keamanan Jaringan: Implementasi langkah-langkah keamanan jaringan yang efektif untuk melindungi data selama pengiriman dan penerimaan.

#### 4. Perangkat Input:

- Deskripsi: Aplikasi dapat berinteraksi dengan perangkat input seperti keyboard dan pemindai sidik jari untuk memasukkan data atau autentikasi.
- Fungsionalitas Keyboard: Pengguna dapat memasukkan data gaji atau informasi TA menggunakan keyboard.
- Autentikasi Sidik Jari: Opsi autentikasi menggunakan pemindai sidik jari untuk keamanan tambahan.

#### 5. Perangkat Penyimpanan Eksternal:

- Deskripsi: Antarmuka dengan perangkat penyimpanan eksternal memungkinkan pengguna mengimpor atau mengekspor data administrasi TA.
- Tipe Perangkat: Dukungan untuk hard drive eksternal, USB, atau media penyimpanan eksternal lainnya.
- Fungsionalitas Impor/Ekspor: Kemampuan untuk mentransfer data gaji atau laporan antara aplikasi dan perangkat penyimpanan eksternal.

### 4.3 Software Interface

#### 1. Basis Data (Database Management System):

- Deskripsi: Antarmuka perangkat lunak dengan Sistem Manajemen Basis Data (Database Management System/DBMS) untuk menyimpan dan mengakses data administrasi Tugas Akhir (TA) mahasiswa.
- DBMS yang Digunakan: Contoh DBMS yang mungkin digunakan termasuk MySQL, Oracle, atau Microsoft SQL Server.
- Komunikasi dengan DBMS: Aplikasi berkomunikasi dengan DBMS untuk melakukan operasi seperti pengambilan, pembaruan, dan penyimpanan data.

#### 2. Sistem Operasi:

- Deskripsi: Antarmuka perangkat lunak dengan Sistem Operasi (Operating System/OS) yang digunakan untuk menjalankan aplikasi pengelolaan administrasi TA.
- Sistem Operasi yang Dukung: Aplikasi dapat dijalankan di berbagai sistem operasi seperti Windows, macOS, atau Linux, tergantung pada kebutuhan dan preferensi.

#### 3. Web Server:

- Deskripsi: Antarmuka perangkat lunak dengan Web Server untuk menyajikan antarmuka pengguna melalui web browser.
- Web Server yang Digunakan: Contoh web server termasuk Apache HTTP Server, Nginx, atau Microsoft IIS, tergantung pada kebutuhan dan persyaratan aplikasi.

#### 4. Bahasa Pemrograman dan Framework:

- Deskripsi: Antarmuka perangkat lunak dengan bahasa pemrograman dan framework yang digunakan dalam pengembangan aplikasi pengelolaan administrasi TA.
- Bahasa Pemrograman: Contoh bahasa pemrograman termasuk Java, C#, atau Python, tergantung pada kebutuhan aplikasi. Framework: Contoh framework termasuk Spring, .NET, atau Django, tergantung pada preferensi dan persyaratan pengembangan.

#### 5. Libraries dan Dependensi:

- Deskripsi: Antarmuka perangkat lunak dengan libraries atau dependensi lain yang mendukung fungsionalitas aplikasi.
- Contoh Libraries: Penggunaan library untuk manajemen pengiriman email, pembuatan laporan PDF, atau autentikasi pengguna sesuai kebutuhan aplikasi.
- Versi Libraries dan Dependensi: Konsistensi versi untuk memastikan kompatibilitas dan kinerja optimal aplikasi.

### 4.4 Communication Interface

Terdapat kebutuhan komunikasi untuk memungkinkan interaksi antara perangkat lunak (PL) dengan komponen lainnya, seperti pengguna, sistem basis data, perangkat keras, atau sistem eksternal.

#### 1. Komunikasi Pengguna:

- Deskripsi: Antarmuka pengguna aplikasi pengelolaan administrasi TA harus dirancang untuk memfasilitasi komunikasi dua arah antara pengguna (mahasiswa, dosen pembimbing, dan staf administrasi) dan sistem.
- Kebutuhan: Mahasiswa dapat menginput dan melihat data terkait TA, seperti judul, abstrak, dan kemajuan, Dosen pembimbing dapat memberikan umpan balik dan mengevaluasi kemajuan TA melalui antarmuka, Staf administrasi dapat memasukkan informasi terkait administrasi TA dan memberikan pemberitahuan.

#### 2. Komunikasi dengan Sistem Basis Data

- Deskripsi: Aplikasi harus dapat berkomunikasi dengan sistem basis data untuk mengelola data TA mahasiswa.
- Kebutuhan: Pengambilan data mahasiswa, dosen, dan detail TA dari sistem basis data, Memperbarui dan menyimpan informasi kemajuan TA ke dalam basis data, Penggunaan bahasa query untuk operasi membaca dan menulis data, Komunikasi dengan Sistem Eksternal.



### 3. Komunikasi dengan Sistem Eksternal

- Deskripsi: Aplikasi perlu terintegrasi dengan sistem eksternal seperti sistem akademik universitas atau sistem perpustakaan.
- Kebutuhan: Sinkronisasi data TA dengan sistem akademik untuk memastikan konsistensi informasi, Integrasi dengan sistem perpustakaan untuk memantau literatur yang digunakan dalam TA, Komunikasi dengan Perangkat Keras.

### 4. Komunikasi dengan Perangkat Keras

- Deskripsi: Aplikasi mungkin membutuhkan interaksi dengan perangkat keras seperti komputer mahasiswa, perangkat penyimpanan, atau pemindai dokumen.
- Kebutuhan: Kompatibilitas dengan berbagai jenis perangkat komputer yang digunakan oleh mahasiswa. Koneksi dengan perangkat penyimpanan eksternal jika mahasiswa perlu menyimpan data TA. Integrasi dengan pemindai dokumen untuk mengunggah dokumen TA.

### 5. Komunikasi Antar-Komponen dalam Aplikasi:

- Deskripsi: Komponen-komponen dalam aplikasi harus berkomunikasi untuk menyediakan fungsionalitas terpadu.
- Kebutuhan: Modul pengelolaan kemajuan TA berinteraksi dengan modul evaluasi oleh dosen pembimbing, Modul administrasi TA berkomunikasi dengan modul pengarsipan untuk menyimpan data secara efisien, Kesatuan informasi antar modul untuk memastikan integritas data dan pelacakan progres TA.