

# SOFTWARE REQUIREMENTS SPECIFICATION (SRS)

FOR Manajemen Administrasi  
Tugas Akhir Universitas XYZ

Disusun Oleh:  
Muhammad Farid Al  
Farisi  
3411211020  
Kelas A

# Table Of Contents

<b>Table Of Contents .....</b>	<b>1</b>
<b>Bab I Introduction .....</b>	<b>2</b>
1.1 Purpose.....	2
1.2 Intended Audience and Reading Suggestions .....	2
1.3 Project Scope .....	2
1.4 References .....	2
<b>Bab II Overall Description .....</b>	<b>3</b>
2.1 Organisations .....	3
2.2 Product Perspective .....	3
2.3 User Classes and Characteristics.....	3
2.4 Operating Environment.....	3
2.5 Design and Implementation Constrains (optional) .....	4
2.6 Assumptions and Dependencies (optional).....	4
<b>Bab III Functional Requirements.....</b>	<b>5</b>
3.1 Detailed Functional Requirements .....	5
3.2 Use Case Diagram.....	5
3.3 Use Case Scenario.....	6
<b>Bab IV Non Functional Requirements.....</b>	<b>7</b>
4.1 Performance Requirements (optional) .....	7
4.2 Safety Requirements (optional) .....	7
4.3 Software Quality Attributes (optional) .....	8
<b>Bab V Data Requirements.....</b>	<b>9</b>
4.1 Input .....	9
4.2 Output .....	10
<b>Bab VI Interface Requirements.....</b>	<b>12</b>
4.1 User Interface.....	12
4.2 Hardware Interface.....	13
4.3 Software Interface .....	13
4.4 Communication Interface.....	13

# **Bab I Introduction**

## **1.1 Purpose**

Latar belakang dan tujuan dibuatnya dokumen SRS ini adalah untuk memenuhi ujian praktikum APPL, untuk mengembangkan aplikasi Manajemen Administrasi Tugas Akhir pada Jurusan Informatika di Universitas XYZ. Dokumen ini bertujuan untuk menyediakan sistem yang efektif dan efisien dalam mengelola proses Tugas Akhir di universitas tersebut.

## **1.2 Intended Audience and Reading Suggestions**

Dokumen SRS ini ditujukan untuk berbagai jenis pembaca, termasuk:

- Tim pengembang (developers)
- Manajer proyek (project managers)
- Staf pemasaran (marketing staff)
- Pengguna (users)
- Tester (testers)
- Penulis dokumen (document writers)

## **1.3 Project Scope**

Lingkup aplikasi Manajemen Administrasi Tugas Akhir meliputi:

- Manajemen mahasiswa, dosen pembimbing, dosen penguji, ketua jurusan, ketua KBK, dan koordinator Tugas Akhir.
- Pembuatan dan pengelolaan Proposal Penelitian (TA1), dokumen laporan penelitian (TA2), makalah Publikasi Ilmiah, dan perangkat lunak yang digunakan dalam penelitian.
- Pengecekan syarat-syarat dosen pembimbing, dosen penguji, dan mahasiswa untuk mengambil Tugas Akhir.
- Pelaksanaan seminar Tugas Akhir 1 dan Tugas Akhir 2.

## **1.4 References**

- Buku Panduan Tugas Akhir Informatika Universitas XYZ.

## **Bab II Overall Description**

### **2.1 Organitations**

Organisasi yang terlibat dalam sistem ini meliputi:

- Universitas XYZ
- Jurusan Informatika
- Dosen Pembimbing 1 dan 2
- Dosen Penguji 1, 2, dan 3
- Ketua Jurusan
- Ketua KBK
- Koordinator Tugas Akhir

### **2.2 Product Perspective**

Aplikasi Manajemen Administrasi Tugas Akhir ini akan menjadi sistem yang terintegrasi dengan organisasi dan berfungsi untuk membantu mengelola seluruh proses Tugas Akhir di Jurusan Informatika. Sistem ini akan memberikan manfaat dalam efisiensi administrasi, pemantauan progress Tugas Akhir, dan pencapaian syarat-syarat yang diperlukan.

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

Pengguna yang terlibat dalam aplikasi ini adalah:

- Mahasiswa: Memiliki akses untuk mengajukan Tugas Akhir, melihat informasi pembimbing dan penguji, melaporkan progress penelitian, dan mengikuti seminar Tugas Akhir.
- Dosen Pembimbing 1 dan 2: Bertanggung jawab untuk membimbing mahasiswa dalam Tugas Akhir, memberikan penilaian, dan melaporkan kemajuan penelitian.
- Dosen Penguji 1, 2, dan 3: Bertanggung jawab untuk menguji mahasiswa dalam seminar Tugas Akhir.
- Ketua Jurusan: Bertanggung jawab dalam pengelolaan Tugas Akhir di jurusan.
- Ketua KBK: Menunjuk dosen penguji untuk mahasiswa yang akan melakukan seminar Tugas Akhir.
- Koordinator Tugas Akhir: Bertanggung jawab untuk memverifikasi syarat-syarat mahasiswa, mengatur jadwal seminar, dan memantau progres Tugas Akhir.

### **2.4 Operating Environment**

Sistem ini akan dioperasikan di lingkungan berikut:

- Platform hardware: Disesuaikan dengan persyaratan minimum sistem aplikasi.
- Sistem operasi dan versi: Disesuaikan dengan persyaratan minimum sistem aplikasi.
- Komponen software lainnya yang mendukung PL: Disesuaikan dengan persyaratan minimum sistem aplikasi.

## 2.5 Design and Implementation Constrains (optional)

Batasan desain dan implementasi sistem ini akan ditentukan lebih lanjut selama proses pengembangan.

## 2.6 Assumptions and Dependencies (optional)

Beberapa asumsi dan ketergantungan yang diperhatikan dalam dokumen ini antara lain:

- Asumsi bahwa informasi yang diberikan dalam Buku Panduan Tugas Akhir Informatika Universitas XYZ akurat dan relevan.
- Ketergantungan pada kebijakan dan prosedur yang ada di Universitas XYZ terkait 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.

### **3.1 Detailed Functional Requirements**

Berikut adalah detail kebutuhan fungsional pada PL (Manajemen Administrasi Tugas Akhir):

1. Registrasi Tugas Akhir: Mahasiswa dapat mendaftarkan diri untuk mengambil Tugas Akhir dengan mengisi formulir pendaftaran dan melampirkan berkas-berkas yang diperlukan.
2. Pemilihan Pembimbing: Mahasiswa dapat memilih Pembimbing 1 dan Pembimbing 2 untuk Tugas Akhir. Jika tidak memilih dalam waktu yang ditentukan, koordinator akan memilihkan pembimbing.
3. Penentuan Judul Penelitian: Mahasiswa dapat menentukan judul penelitian sendiri atau meminta rekomendasi judul penelitian pada Dosen Pembimbing.
4. Bimbingan Tugas Akhir 1: Mahasiswa dapat melakukan bimbingan dengan Pembimbing 1 dan Pembimbing 2 minimal 6 kali sebelum maju seminar Tugas Akhir.
5. Kelayakan Maju Seminar Tugas Akhir: Mahasiswa harus memenuhi syarat-syarat yang telah ditentukan sebelum dapat maju seminar Tugas Akhir.
6. Penentuan Dosen Penguji: Ketua KBK menentukan Dosen Penguji yang akan menguji Tugas Akhir.
7. Pengajuan Tugas Akhir: Setelah Tugas Akhir 1 selesai, mahasiswa dapat mengajukan Tugas Akhir dengan mengumpulkan dokumen laporan penelitian.
8. Penentuan Jadwal Sidang: Koordinator Tugas Akhir menentukan jadwal sidang Tugas Akhir.
9. Hasil Sidang Tugas Akhir: Dosen Penguji memberikan penilaian dan masukan pada hasil penelitian yang telah dilakukan oleh mahasiswa.

### **3.2 Use Case Diagram**

Beberapa use case yang terdapat pada diagram tersebut antara lain:

1. Registrasi Tugas Akhir: Mahasiswa dapat mendaftarkan diri untuk mengambil Tugas Akhir dengan mengisi formulir pendaftaran dan melampirkan berkas-berkas yang diperlukan.

2. Pemilihan Pembimbing: Mahasiswa dapat memilih Pembimbing 1 dan Pembimbing 2 untuk Tugas Akhir. Jika tidak memilih dalam waktu yang ditentukan, koordinator akan memilihkan pembimbing.
3. Penentuan Judul Penelitian: Mahasiswa dapat menentukan judul penelitian sendiri atau meminta rekomendasi judul penelitian pada Dosen Pembimbing.
4. Bimbingan Tugas Akhir 1: Mahasiswa dapat melakukan bimbingan dengan Pembimbing 1 dan Pembimbing 2 minimal 6 kali sebelum maju seminar Tugas Akhir.

### 3.3 Use Case Scenario

Berikut adalah penjelasan mengenai kebutuhan fungsionalitas PL yang telah dijelaskan dalam use case diagram dalam scenario use case:

1. Memahami permasalahan yang akan dibangun: Sebelum membuat skenario use case, penting untuk memahami permasalahan yang akan dibangun dan menetapkan tujuan yang jelas untuk solusi yang akan dibuat.
2. Identifikasi use case: Setelah memahami permasalahan, identifikasi use case yang diperlukan dan manfaatkan teknik analisis seperti wawancara dan observasi untuk mengumpulkan informasi tentang aktivitas pengguna.
3. Membuat skenario utama: Mulai dengan membuat skenario utama, yaitu alur penggunaan sistem yang ideal dan mencakup semua langkah yang harus dilakukan pengguna untuk mencapai tujuan mereka.
4. Menggunakan format yang benar: Gunakan format yang benar saat menulis skenario use case, termasuk langkah-langkah yang terperinci, input yang dibutuhkan, output yang dihasilkan, serta kondisi dan batasan yang relevan.
5. Melakukan validasi: Validasi skenario use case sangat penting untuk memastikan bahwa solusi yang dibangun dapat memenuhi kebutuhan pengguna dan memecahkan masalah yang ada. Lakukan uji coba dan perbaikan jika diperlukan.

## **Bab IV Non Functional Requirements**

Kebutuhan Non Fungsional adalah kebutuhan yang menitikberatkan pada properti perilaku yang dimiliki oleh sistem. Berikut adalah kebutuhan non-fungsional yang relevan untuk aplikasi Manajemen Administrasi Tugas Akhir pada Jurusan Informatika di Universitas XYZ:

1. Keamanan: Aplikasi harus memiliki sistem keamanan yang kuat untuk mencegah akses yang tidak sah dan melindungi data sensitif mahasiswa dan dosen.
2. Performa: Aplikasi harus dapat menangani beban kerja yang tinggi dan memberikan respons yang cepat pada penggunaan normal.
3. Skalabilitas: Aplikasi harus dapat diubah dan diperluas dengan mudah jika jumlah pengguna atau fitur aplikasi meningkat di masa depan.
4. Interoperabilitas: Aplikasi harus dapat berintegrasi dengan sistem lain yang digunakan di Universitas XYZ, seperti sistem manajemen akademik dan sistem manajemen perpustakaan.
5. Keterjangkauan: Aplikasi harus mudah diakses dan digunakan oleh mahasiswa dan dosen, dan biayanya harus terjangkau untuk Universitas XYZ.
6. Ketersediaan: Aplikasi harus tersedia 24 jam sehari, 7 hari seminggu, kecuali jika ada perawatan atau perbaikan yang diperlukan.
7. Usabilitas: Aplikasi harus mudah digunakan oleh pengguna yang tidak terampil dalam teknologi dan harus menyediakan antarmuka yang intuitif dan mudah dipahami.
8. Portabilitas: Aplikasi harus dapat dijalankan di berbagai perangkat dan platform, seperti desktop, laptop, tablet, dan ponsel pintar.

### **4.1 Performance Requirements (optional)**

1. Aplikasi harus mampu menangani beban kerja yang tinggi dan memberikan respons yang cepat terhadap penggunaan normal.
2. Waktu respons aplikasi harus dalam rentang yang dapat diterima, misalnya, tidak boleh lebih dari 3 detik untuk mengakses halaman atau memproses permintaan.
3. Aplikasi harus dapat melakukan pengolahan data dengan efisien, termasuk mengunduh dan mengunggah berkas dengan cepat.

### **4.2 Safety Requirements (optional)**

1. Aplikasi harus memiliki sistem keamanan yang kuat untuk melindungi data sensitif mahasiswa dan dosen.
2. Data yang dikirim antara pengguna dan aplikasi harus dienkripsi untuk mencegah akses yang tidak sah.
3. Aplikasi harus dilengkapi dengan mekanisme otentikasi yang kuat untuk mencegah akses yang tidak sah.



#### 4.3 Software Quality Attributes (optional)

1. Aplikasi harus memiliki antarmuka yang mudah dipahami dan mudah digunakan oleh pengguna yang tidak terampil dalam teknologi.
2. Aplikasi harus dapat berintegrasi dengan sistem lain yang digunakan di Universitas XYZ, seperti sistem manajemen akademik dan sistem manajemen perpustakaan.
3. Aplikasi harus mudah diakses dan digunakan oleh mahasiswa dan dosen, dan biayanya harus terjangkau untuk Universitas XYZ.
4. Aplikasi harus tersedia 24 jam sehari, 7 hari seminggu, kecuali jika ada perawatan atau perbaikan yang diperlukan.
5. Aplikasi harus memiliki dokumentasi yang lengkap dan jelas untuk memudahkan pengguna dalam memahami dan menggunakan sistem.
6. Aplikasi harus dapat dijalankan di berbagai perangkat dan platform, seperti desktop, laptop, tablet, dan ponsel pintar.
7. Aplikasi harus memiliki performa yang stabil dan tidak mengalami kegagalan sistem yang sering.

## Bab V Data Requirements

Dalam sistem Manajemen Administrasi Tugas Akhir pada Jurusan Informatika di Universitas XYZ, terdapat beberapa data yang dibutuhkan oleh pengguna. Berikut adalah rincian data yang diperlukan:

### 4.1 Input

#### a) Data Mahasiswa:

- Nama (atribut: string)
- NIM (atribut: string)
- Jurusan (atribut: string)
- Jumlah SKS (atribut: integer)
- Nilai Praktikum (atribut: array of string)
- Nilai Mata Kuliah (atribut: array of string)
- Tanda Bukti Pembayaran Administrasi TA1 (atribut: file)
- Tanda Bukti Pembayaran Administrasi TA2 (atribut: file)

#### b) Data Dosen Pembimbing:

- Nama (atribut: string)
- NIP (atribut: string)
- Jabatan Fungsional (atribut: string)
- Bidang Keahlian (atribut: string)

#### c) Data Dosen Penguji:

- Nama (atribut: string)
- NIP (atribut: string)
- Jabatan Fungsional (atribut: string)
- Bidang Keahlian (atribut: string)

#### d) Data Koordinator Tugas Akhir:

- Nama (atribut: string)
- NIP (atribut: string)
- Jabatan (atribut: string)

#### e) Data Tugas Akhir:

- Judul Penelitian (atribut: string)
- Tema Penelitian (atribut: string)
- Proposal Penelitian (atribut: file)
- Laporan Penelitian (atribut: file)
- Makalah Publikasi Ilmiah (atribut: file)
- Perangkat Lunak (atribut: file)

f) Data Publikasi Ilmiah:

- Judul Publikasi (atribut: string)
- Penulis (atribut: array of string)
- Tahun Publikasi (atribut: integer)
- Jurnal atau Konferensi (atribut: string)
- Indeks (atribut: string)

Matriks Akses Pengguna terhadap Data:

	<b>Mahasiswa</b>	<b>Dosen Pembimbing</b>	<b>Dosen Penguji</b>	<b>Koordinator Tugas Akhir</b>
Data Mahasiswa	CRUD	Read	Read	Read
Data Dosen Pembimbing	Read	CRUD	-	-
Data Dosen Penguji	Read	-	CRUD	-
Data Koordinator Tugas Akhir	-	-	-	CRUD
Data Tugas Akhir	CRUD	CRUD	-	CRUD
Data Publikasi Ilmiah	-	-	-	CRUD

## 4.2 Output

a) Laporan Penelitian TA1:

- Judul Penelitian (atribut: string)
- Nama Mahasiswa (atribut: string)
- Nama Dosen Pembimbing (atribut: string)

- Status (atribut: string)

b) Laporan Penelitian TA2:

- Judul Penelitian (atribut: string)
- Nama Mahasiswa (atribut: string)
- Nama Dosen Pembimbing (atribut: string)
- Nama Dosen Penguji (atribut: array of string)
- Status (atribut: string)

c) Surat Pengantar Publikasi Ilmiah:

- Nama Mahasiswa (atribut: string)
- Judul Penelitian (atribut: string)
- Nama Dosen Pembimbing (atribut: string)
- Tanggal Publikasi (atribut: date)

d) Surat Kelayakan Maju Seminar:

- Nama Mahasiswa (atribut: string)
- Judul Penelitian (atribut: string)
- Nama Dosen Pembimbing (atribut: string)
- Tanggal Seminar (atribut: date)

e) Surat Keterangan Bebas Plagiasi:

- Nama Mahasiswa (atribut: string)
- Judul Penelitian (atribut: string)
- Nama Dosen Pembimbing (atribut: string)
- Tanggal Pengecekan Plagiasi (atribut: date)

## **Bab VI Interface Requirements**

Dalam sistem Manajemen Administrasi Tugas Akhir pada Jurusan Informatika di Universitas XYZ, terdapat beberapa kebutuhan antarmuka yang dibutuhkan oleh pengguna. Berikut adalah rincian kebutuhan antarmuka yang diperlukan:

### **4.1 User Interface**

#### **a) Antarmuka Mahasiswa:**

1. Tampilan beranda untuk mengakses fitur-fitur terkait Tugas Akhir.
2. Formulir pendaftaran Tugas Akhir yang mencakup input data mahasiswa, pemilihan dosen pembimbing, dan unggahan berkas.
3. Halaman pengelolaan Tugas Akhir, termasuk pemilihan judul penelitian, komunikasi dengan dosen pembimbing, dan unggahan berkas terkait Tugas Akhir.
4. Notifikasi dan pesan untuk memberikan informasi penting terkait Tugas Akhir.
5. Laporan penelitian dan hasil penilaian Tugas Akhir.

#### **b) Antarmuka Dosen Pembimbing:**

1. Tampilan beranda untuk mengakses fitur-fitur terkait pembimbingan Tugas Akhir.
2. Halaman pengelolaan mahasiswa bimbingan, termasuk penugasan judul penelitian dan komunikasi dengan mahasiswa.
3. Laporan dan penilaian Tugas Akhir mahasiswa yang dibimbing.
4. Notifikasi dan pesan untuk memberikan informasi penting terkait pembimbingan Tugas Akhir.

#### **c) Antarmuka Dosen Penguji:**

1. Tampilan beranda untuk mengakses fitur-fitur terkait pengujian Tugas Akhir.
2. Halaman pengelolaan jadwal pengujian Tugas Akhir dan komunikasi dengan mahasiswa.
3. Penilaian dan masukan terhadap Tugas Akhir yang diuji.

#### **d) Antarmuka Koordinator Tugas Akhir:**

1. Tampilan beranda untuk mengakses fitur-fitur terkait koordinasi Tugas Akhir.
2. Halaman pengelolaan proses Tugas Akhir, termasuk verifikasi dan validasi persyaratan, penjadwalan seminar, dan komunikasi dengan mahasiswa, dosen pembimbing, dan dosen penguji.
3. Laporan dan informasi terkait progres Tugas Akhir mahasiswa.

#### 4.2 Hardware Interface

1. Komputer atau perangkat dengan sistem operasi yang kompatibel (misalnya, Windows, macOS, Linux).
2. Perangkat input seperti keyboard, mouse, dan touchpad untuk interaksi pengguna.
3. Perangkat output seperti monitor atau layar untuk menampilkan antarmuka pengguna.
4. Perangkat penyimpanan seperti hard drive atau SSD untuk menyimpan data dan berkas aplikasi.

#### 4.3 Software Interface

1. Basis data untuk menyimpan data pengguna, informasi Tugas Akhir, dan laporan penelitian.
2. Sistem operasi seperti Windows, macOS, atau Linux.
3. Perangkat lunak pengolah kata dan presentasi seperti Microsoft Word dan PowerPoint.
4. Perangkat lunak pengiriman pesan dan komunikasi seperti email atau platform kolaborasi.
5. Perangkat lunak browser web untuk mengakses aplikasi berbasis web.

#### 4.4 Communication Interface

1. Jaringan komputer atau internet untuk menghubungkan pengguna dengan aplikasi dan server basis data.
2. Protokol komunikasi seperti HTTP atau HTTPS untuk mengirim dan menerima data antara aplikasi dan pengguna.
3. Email atau pesan sistem internal untuk pengiriman notifikasi dan komunikasi penting antara pengguna.