

Hasil Sidang Pelanggaran Mahasiswa

SRS-xxxx

SOFTWARE REQUIREMENTS SYSTEM

<Sistem Pelacak Pelanggaran Akademik>

untuk:

Fakultas Informatika Universitas Telkom

Dipersiapkan oleh:

Muhammad Rizky Fajri 1302204043,

Program Studi S1 Rekayasa Perangkat Lunak

Fakultas Informatika

Universitas Telkom

2024

Program Studi S1 Rekayasa Perangkat Lunak

-

Fakultas Informatika

SKPL - Nomor Dokumen

Halaman

Revisi

<nomor revisi>

Tgl: <isi tanggal>

Daftar Perubahan

Revisi

Deskripsi

A

Pembuatan dokumen SRS awal untuk SiPPak

-

B

C

D

INDEX

-

A

B

C

D

TGL

Ditulis oleh

M. Rizky Fajri

Diperiksa oleh

Disetujui oleh

Daftar Halaman Perubahan

Revisi

Halaman

Isi Semula

Perubahan

A

15

Activity Diagram Pembelian Barang Tidak Sesuai dengan UC Scenario nya

Activity Diagram Pembelian Barang sudah disesuaikan dengan UC scenario

Daftar Isi

Daftar Perubahan“

Daftar Halaman Perubahan“0

Daftar Isi“@

1. •endahuluan“P

1.1 •@ujuan Penulisan Dokumen“P

1.2 •'V ær Æ–æv·W enulisan Dokumen “P

1.3• Overview“P

1.3.1• Perspective“P

1.3.2• &÷VæF y / Limitation“P

1.3.3•W6W" 6† acteristics/User Class“P

1.4•&V`erensi“P

2.”FW6• ipsi Umum Perangkat Lunak“^

2.1”FW6• ipsi Kebutuhan“^

2.2”°ebutuhan Fungsional“^

2.4”°ebutuhan Non Functional“^

3.”FW6• ipsi Detail Perangkat Lunak“p

3.1• emetaan Functional Requirement menjadi Use Case“p

3.1.1 Matriks UC“p

3.2• emodelan Analisis “p

3.2.1 Use Case Scenario untuk Register Barang“p

3.2.2• emodelan Activity Diagram untuk Register Barang“€

3.3• emodelan Class Diagram (Problem Domain)“€

4.”°ebutuhan Lain-Lain“•

4.1”°ebutuhan Data “•

4.2”°ebutuhan Antar Pengguna “•

4.3”°ebutuhan Perangkat Keras“•

4.4”°ebutuhan Perangkat Lunak“•

Pendahuluan

Tujuan Penulisan Dokumen

Dokumen SRS ini bertujuan untuk mengkomunikasikan kebutuhan sistem SiPPak (Sistem Pendataan Pelanggaran Akademik) kepada seluruh pemangku kepentingan dan tim pengembang. Dokumen ini menjadi panduan utama dalam pengembangan sistem manajemen pelanggaran akademik yang terintegrasi untuk Fakultas Informatika Universitas Telkom.

Ruang Lingkup Penulisan Dokumen

Dokumen ini mencakup spesifikasi kebutuhan fungsional dan non-fungsional sistem SiPPak, analisis aktor dan use case, pemodelan sistem, serta kebutuhan teknis untuk implementasi sistem berbasis web dengan teknologi React.js dan Express.js.

Product Overview

SiPPak adalah sistem informasi berbasis web yang dirancang untuk mengelola dan mendokumentasikan kasus pelanggaran akademik mahasiswa di lingkungan Fakultas Informatika. Sistem ini menyediakan platform terpusat untuk pencatatan, pemantauan, dan pengelolaan kasus pelanggaran mulai dari pelaporan hingga penyelesaian sidang etik.

Vision Statement: Menciptakan sistem manajemen pelanggaran akademik yang efisien, transparan, dan akuntabel untuk mendukung penegakan etika akademik di Fakultas Informatika.

Statement of Objective:

Digitalisasi proses pencatatan pelanggaran akademik

Meningkatkan efisiensi pengelolaan kasus pelanggaran

Menyediakan dokumentasi yang terstruktur dan mudah diakses

Product Perspective

SiPPak merupakan sistem standalone yang beroperasi dalam domain manajemen akademik Fakultas Informatika. Sistem ini berfungsi sebagai:

Platform digital untuk menggantikan sistem pencatatan manual

Alat bantu untuk tim kemahasiswaan dalam mengelola kasus pelanggaran

Dashboard monitoring untuk pimpinan fakultas

Sistem dokumentasi yang terintegrasi dengan workflow sidang etik

Product Boundary / Limitation

Batasan sistem SiPPak meliputi:

Hanya menangani kasus pelanggaran akademik, bukan pelanggaran pidana

Terbatas pada lingkup Fakultas Informatika

Tidak terintegrasi dengan sistem informasi akademik utama universitas

Akses terbatas pada jaringan internal fakultas untuk keamanan data

User Characteristics/User Class•

Kelas Pengguna

Deskripsi

Estimasi Jumlah

Admin

Memiliki akses penuh ke seluruh fitur sistem termasuk manajemen user dan persetujuan kasus

2-3

Staff

Staff kemahasiswaan yang mengelola data pelanggaran dan dokumentasi kasus

4-7

User

Pengguna umum yang dapat melihat dashboard dan mengelola profil

6-10

Referensi

Standar IEEE 830-1998 untuk Software Requirements Specification

Peraturan Akademik Universitas Telkom

Panduan Etika Akademik Fakultas Informatika

Dokumentasi React.js dan Express.js

Deskripsi Umum Perangkat Lunak

Deskripsi Kebutuhan

SiPPak dikembangkan untuk memenuhi kebutuhan digitalisasi proses pengelolaan pelanggaran akademik yang saat ini masih dilakukan secara manual. Sistem ini harus mampu mengakomodasi workflow mulai dari pencatatan kasus, dokumentasi proses, hingga penyimpanan hasil sidang etik dengan tetap menjaga keamanan dan kerahasiaan data.

Kebutuhan Fungsional

No.

Kode

Kebutuhan Fungsional dalam format System Perspective

1

FR-001

Sistem harus menyediakan fungsi autentikasi dan otorisasi pengguna

2

FR-002

Sistem harus dapat mengelola data mahasiswa pelanggar

3

FR-003

Sistem harus dapat mencatat dan mengelola kasus pelanggaran

4

FR-004

Sistem harus dapat mengelola status kasus (Berjalan, Ditunda, Selesai, Dibatalkan)

5

FR-005

Sistem harus dapat menyimpan dokumentasi sidang (hasil sidang, notulensi)

6

FR-006

Sistem harus menyediakan dashboard monitoring kasus

7

FR-007

Sistem harus dapat mengelola profil pengguna

8

FR-008

Sistem harus dapat mengelola user management (hanya Admin)

9

FR-009

Sistem harus dapat mengupload dan menyimpan file pendukung

10

FR-010

Sistem harus dapat melakukan ekspor laporan

11

FR-011

Sistem harus dapat melakukan persetujuan kasus (Case Approval)

12

FR-012

Sistem harus dapat menampilkan data dalam bentuk grafik dan statistik

User Story

No.

Kode

Kebutuhan Fungsional dalam format User Story

1.

FR-001

Sebagai semua pengguna, saya harus dapat melakukan login ke sistem SiPPak

2

FR-002

Sebagai Admin dan Staff, saya harus dapat mengelola data mahasiswa

3

FR-003

Sebagai Admin dan Staff, saya harus dapat membuat dan mengelola kasus pelanggaran

4

FR-004

Sebagai Admin dan Staff, saya harus dapat mengubah status kasus sesuai progress

5

FR-005

Sebagai Admin dan Staff, saya harus dapat mengupload hasil sidang dan notulensi

6

FR-006

Sebagai semua pengguna, saya harus dapat melihat dashboard monitoring

7

FR-007

Sebagai semua pengguna, saya harus dapat mengelola profil saya

8

FR-008

Sebagai Admin, saya harus dapat mengelola user dan role

9

FR-009

Sebagai Admin dan Staff, saya harus dapat mengupload file bukti pelanggaran

10

FR-010

Sebagai Admin dan Staff, saya harus dapat mengekspor laporan pelanggaran

11

FR-011

Sebagai Admin, saya harus dapat menyetujui atau menolak kasus

12

FR-012

Sebagai semua pengguna, saya harus dapat melihat statistik dalam bentuk grafik

Kebutuhan Non Functional

No.

Quality Criteria

Kode Kebutuhan

Deskripsi

1

Usability

NFR-001

Sistem harus mudah digunakan dengan antarmuka yang intuitif

2

Security

NFR-002

Sistem harus aman dengan enkripsi password dan autentikasi JWT

3

Supportability

NFR-003

Sistem harus dapat memuat halaman dalam waktu < 5 detik

4

Reliability

NFR-004

Sistem harus memiliki uptime minimal 90%

5

Scalability

NFR-005

Sistem harus dapat menangani hingga 100 pengguna concurrent

6

Maintainability

NFR-006

Kode harus terstruktur dan terdokumentasi dengan baik

Deskripsi Detil Perangkat Lunak

Pemetaan Functional Requirement menjadi Use Case•

No.

Kode FR

Nama UC

1

FR-001

Login

2

FR-002

Kelola Data Mahasiswa

3

FR-003

Kelola Kasus Pelanggaran

4

FR-004

Update Status Kasus

5

FR-005

Upload Dokumentasi Sidang

6

FR-006

View Dashboard

7

FR-007

Kelola Profil

8

FR-008

Kelola User Management

9

FR-009

Upload File

10

FR-010

Ekspor Laporan

11

FR-011

Approval Kasus

12

FR-012

View Statistik

3.1.1 Matriks UC

<Menunjukkan hubungan antara aktor dan use case dalam suatu sistem.>

Use Case / Aktor

Admin

Staff

User

Login

,

,

,

Kelola Data Mahasiswa

,

,

-

Kelola Kasus Pelanggaran

,

,

-

Update Status Kasus

,

,

-

Upload Dokumentasi Sidang

,

,

-

View Dashboard

,

,

,

Kelola Profil

,

,

,

Kelola User Management

,

-

-

Upload File

,

,

-

Ekspor Laporan

,

,

-

Approval Kasus

,

-

-

View Statistik

,

,

,

Pemodelan Analisis

3.2.1 Use Case Diagram

Use Case Diagram SiPPAK menggambarkan interaksi antara tiga aktor utama dengan sistem:

Aktor:

Admin: Memiliki akses penuh ke semua fitur sistem

Staff: Memiliki akses untuk input, edit, view, dan export data pelanggaran, serta melihat dan edit profil

User: Memiliki akses terbatas untuk melihat, mengekspor data, serta melihat dan edit profil

Use Case utama:

Login - Digunakan oleh semua aktor untuk masuk ke system

View Dashboard - Digunakan oleh semua aktor untuk melihat ringkasan dan statistik sistem

View Profile - Extends ke "Edit Profil" untuk mengelola profil pengguna (semua aktor)

Lihat Status Case - Untuk melihat status kasus pelanggaran (Staff dan Admin)

Buat Case - Extends ke "Edit Case Information" dan "Close Case"

Tutup Sidang Etik - Extends ke "Buka Sidang Etik" untuk

mengelola proses sidang

User Management - Extends ke "Add User" dan "Edit User" (khusus Admin)

Relasi Include dan Extend:

Include: Login !' semua use case lainnya (semua fitur memerlukan login)

Extend:

View Profil !' Edit Profil

Buat Case !' Edit Case Information

Buat Case !' Close Case

Tutup Sidang Etik !' Buka Sidang Etik

User Management !' Add User

User Management !' Edit User

3.2.2 Use Case Scenario

3.2.2.1 Use Case Scenario Login

Nama Use Case

Login

Deskripsi

Fungsi ini digunakan oleh pengguna untuk masuk ke dalam sistem SiPPak dengan autentikasi berbasis role

Pre-Kondisi

Pengguna memiliki akun yang terdaftar di sistem

Post-Kondisi

Pengguna berhasil masuk dan diarahkan ke dashboard sesuai role

Skenario Utama

Aktor

Sistem

1. Pengguna mengakses halaman login

2. Sistem menampilkan form login

3. Pengguna memasukkan Email/NIP dan password

4. Sistem memvalidasi kredensial

5. Sistem mengecek role pengguna

6. Sistem mengarahkan ke menu sesuai role

Skenario Eksepsional (Alternative flow)

Aktor

Sistem

1. Pengguna memasukkan kredensial salah

2. Sistem menampilkan pesan error

3. Sistem tetap menampilkan form login

3.2.2.2 Use Case Scenario Buat Case

Nama Usecase

Buat Case

Deskripsi

Fungsi untuk membuat kasus pelanggaran baru dalam sistem

Aktor

Staff, Admin

Pre-Kondisi

- Pengguna sudah login
- Pengguna memiliki akses untuk membuat case

Post-Kondisi

- Kasus baru berhasil dibuat
- Data tersimpan dalam database

Skenario Utama

Aktor

Sistem

Staff memilih menu "Buat Case"

Sistem menampilkan form input case

Staff mengisi data Pelanggaran

Sistem memvalidasi data input

Sistem menyimpan data case

Sistem menampilkan pesan sukses dan ID case

3.2.2.3 Use Case Scenario Lihat Status Case

Nama Usecase

Lihat Status Case

Deskripsi

Fungsi untuk melihat status kasus pelanggaran

Pre-Kondisi

- Pengguna sudah login
- Terdapat data case dalam sistem

Post-Kondisi

- Status case ditampilkan ke pengguna

Skenario Utama

Aktor

Sistem

Pengguna memilih menu "Lihat Status Case"

Sistem menampilkan daftar case

Pengguna memilih case tertentu

Sistem menampilkan detail status case

3.2.2.4 Use Case Scenario User Management

Nama Usecase

User Management

Deskripsi

Fungsi untuk menutup sidang etik dan memberikan keputusan final

Aktor

Admin

Pre-Kondisi

- Pengguna sudah login
- Sidang etik sudah dilaksanakan

Post-Kondisi

- Sidang etik ditutup
- Keputusan final tersimpan

Skenario Utama

Aktor

Sistem

Admin memilih "Menu User Management"

Sistem menampilkan menu

Admin mengisi hasil sidang dan keputusan final

Sistem menyimpan data hasil sidang

Sistem mengubah status case menjadi "Selesai"

3.2.2.5 Use Case Scenario Lihat Profile

Nama Usecase

Lihat Profile

Deskripsi

Fungsi ini digunakan oleh pengguna untuk melihat informasi profil mereka di sistem SiPPak

Pre-Kondisi

- Pengguna sudah login ke system
- Pengguna memiliki profil yang terdaftar

Post-Kondisi

- Informasi profil pengguna ditampilkan
- Pengguna dapat memilih untuk mengedit profil

Skenario Utama

Aktor

Sistem

Pengguna mengakses menu profil

Sistem menampilkan informasi profil pengguna

Pengguna dapat melihat data pengguna

Sistem menyediakan opsi "Edit Profil"

Pengguna dapat memilih untuk mengedit profil

Pemodelan Activity Diagram

Activity Diagram yang menggambarkan alur proses utama dalam sistem SiPPak dari login hingga logout.

Deskripsi Activity Diagram:

Mulai - Pengguna memulai proses

Login - Pengguna memasukkan kredensial

Decision: Valid? - Sistem memvalidasi kredensial

Tidak: Kembali ke form login

Ya: Lanjut ke pengecekan role

Decision: Cek Role - Sistem mengecek role pengguna dan mengarahkan ke aktivitas yang sesuai:

Admin: Dashboard Monitoring !' Kelola Kasus !' Input Pelanggaran !' Kelola Pengguna !' Persetujuan Keputusan !' Logout

Staff: Dashboard Monitoring !' Kelola Kasus !' Input Pelanggaran !' Proses Sidang !' Tutup Kasus !' Edit Profil !' Logout

User: Dashboard Monitoring !' Edit Profil! Logout

Selesai - Proses berakhir setelah logout

Pemodelan Class Diagram (Problem Domain)

Kelas Utama:

User: Menyimpan informasi pengguna sistem (NIP, email, password, role_id, nama, dll.)

Role: Menyimpan jenis peran pengguna (admin, staff, user)

Mahasiswa: Menyimpan data mahasiswa (nama, NIM, jurusan)

Violations: Menyimpan data pelanggaran akademik (id_kasus, jenis_kasus, status, hasil_sidang, dll.)

Report: Menyimpan log laporan dalam bentuk PDF

Relasi:

User N:1 Role (Setiap user memiliki satu role)

Violations N:1 Mahasiswa (Setiap violation terkait dengan satu mahasiswa)

Violations 1:N Report (Setiap Violations dapat menghasilkan laporan)

User 1:N Violations (Satu user dapat menangani banyak violations)

Kebutuhan Lain-Lain

Kebutuhan Data

No.

Nama Data

Tipe Data

Deskripsi

Sumber Data

1

NIP

VARCHAR(20)

Nomor Induk Pegawai untuk login

Input Admin

2

Email

VARCHAR(100)

Email pengguna

Input Admin

3

Password

VARCHAR(255)

Password terenkripsi

Input Pengguna

4

Role ID

INT

ID peran pengguna

Sistem

5

Nama

VARCHAR(255)

Nama lengkap pengguna

Input Admin

6

NIM

VARCHAR(20)

Nomor Induk Mahasiswa

Input Staff

7

Jurusan

VARCHAR(100)

Jurusan mahasiswa

Input Staff

8

ID Kasus

VARCHAR(20)

Kode unik kasus

Sistem (auto-generated)

9

Jenis Kasus

VARCHAR(100)

Jenis pelanggaran

Input Staff

10

Status

TINYINT

Status kasus (1–4)

Input Staff

11

Meeting Date

DATE

Tanggal sidang

Input Staff

12

Hasil Sidang

TEXT

File hasil sidang

Upload Staff

13

Notulensi

TEXT

File notulensi

Upload Staff

14

Deskripsi

TEXT

Deskripsi kasus

Input Staff

15

Foto

VARCHAR(255)

File foto bukti

Upload Staf

Kebutuhan Antar Muka Pengguna

Admin: Interface lengkap dengan semua menu dan kontrol penuh

Staff: Interface untuk data management dengan fitur CRUD yang mudah digunakan

User: Interface sederhana dengan focus pada dashboard monitoring dan profile management

Responsive Design: Interface harus dapat diakses dari berbagai ukuran layar

Intuitive Navigation: Menu navigasi yang mudah dipahami sesuai dengan role masing-masing

Kebutuhan Perangkat Keras

Server:

Processor: Intel Core i5 atau setara

RAM: Minimum 8GB

Storage: Minimum 500GB SSD

Network: Koneksi internet stabil

Client:

Processor: Intel Core i3 atau setara

RAM: Minimum 4GB

Browser: Chrome, Firefox, Safari versi terbaru

Network: Koneksi internet stabil

Kebutuhan Perangkat Lunak

Development Stack:

Frontend: React.js 18.2.0

Backend: Express.js 5.1.0

Database: MySQL (XAMPP)

Authentication: JWT (jsonwebtoken 9.0.2)

File Upload: Multer 2.0.0

Password Hashing: bcryptjs 3.0.2

Production Environment:

Web Server: Apache/Nginx

Database Server: MySQL 8.0+

Node.js Runtime: Version 18+

SSL Certificate untuk HTTPS

Dependencies:

Material-UI untuk komponen UI

Axios untuk HTTP requests

React Router untuk routing

Recharts untuk visualisasi data

PDFKit untuk generate PDF

Lampiran A: Daftar Kata-Kata Asing

SiPPak: Sistem Pendataan Pelanggaran Akademik

NIP: Nomor Induk Pegawai

NIM: Nomor Induk Mahasiswa

CRUD: Create, Read, Update, Delete

JWT: JSON Web Token

API: Application Programming Interface

PDF: Portable Document Format

HTTPS: HyperText Transfer Protocol Secure

Lampiran B: Analysis Models

Lampiran ini berisi model analisis yang mendukung dokumen SRS, termasuk Use Case Diagram lengkap, Activity Diagram, dan Class Diagram yang menunjukkan struktur data sistem.

Use Case Diagram

Activity Diagram

Class Diagram