**Dokumen Desain Perangkat Lunak (Software Design Document - SDD)**

# **Pendahuluan**

## **Tujuan**

Modul Penilaian dan KHS dirancang untuk memfasilitasi proses input, pengolahan, dan penyajian hasil studi mahasiswa. Modul ini memungkinkan dosen dan admin akademik untuk mengelola data nilai secara efisien, serta menyediakan akses bagi mahasiswa untuk memantau perkembangan akademiknya melalui tampilan interaktif dan dokumen resmi seperti KHS

## **Ruang Lingkup**

Modul Penilaian dan Kartu Hasil Studi (KHS) merupakan bagian integral dari sistem informasi akademik yang bertujuan untuk memfasilitasi pengelolaan data nilai mahasiswa secara terstruktur dan efisien. Modul ini mencakup proses input nilai oleh dosen, verifikasi oleh admin akademik, perhitungan otomatis indeks prestasi (IPS dan IPK), serta penyajian hasil studi dalam format digital maupun cetak.

Fitur utama dalam ruang lingkup modul ini meliputi:

* **Input dan pengelolaan nilai akademik** oleh dosen melalui antarmuka yang disediakan.
* **Validasi dan verifikasi data nilai** oleh admin akademik untuk memastikan keakuratan data.
* **Perhitungan otomatis IPS dan IPK** berdasarkan data nilai yang telah diverifikasi.
* **Penyajian Kartu Hasil Studi (KHS)** yang dapat diakses oleh mahasiswa melalui dashboard pribadi.
* **Export KHS** dalam format PDF atau Excel untuk keperluan administrasi atau arsip pribadi.
* **Visualisasi performa akademik** mahasiswa melalui grafik interaktif.
* **Integrasi dengan modul manajemen data dan laporan analitik** untuk memastikan konsistensi dan kemudahan pelaporan.

## **1.3 Referensi**

 **IEEE 1016-2009:** *Standard for Software Design Description*, sebagai acuan dalam penyusunan dokumen desain perangkat lunak.

 **Dokumentasi Yii2 Framework:** Panduan pengembangan aplikasi web menggunakan framework Yii2 untuk backend sistem.

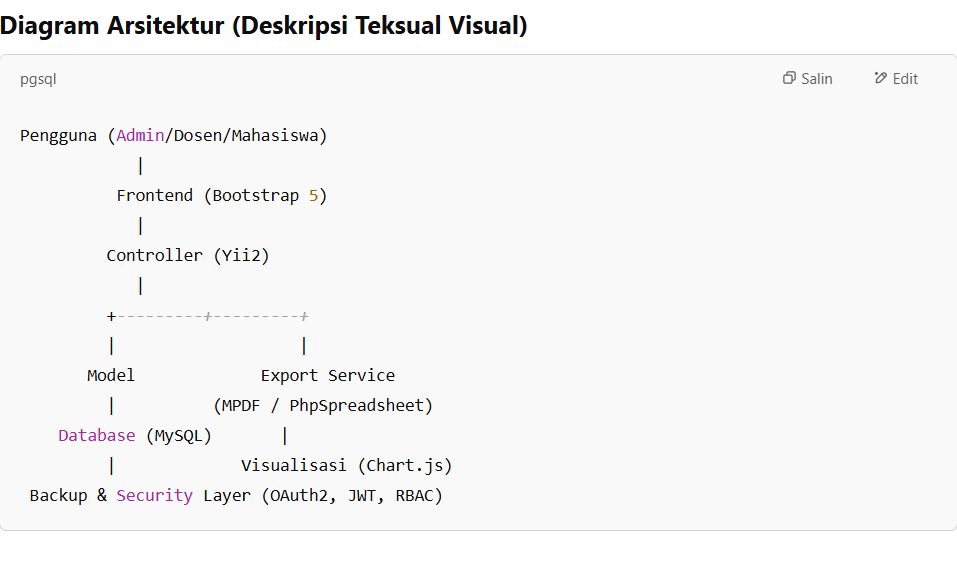
 **Panduan Bootstrap 5:** Referensi dalam perancangan antarmuka pengguna yang responsif dan intuitif.

 **MPDF Documentation:** Panduan penggunaan MPDF untuk menghasilkan dokumen KHS dalam format PDF.

 **PhpSpreadsheet Documentation:** Dokumentasi untuk pembuatan dan manipulasi file Excel sebagai format alternatif KHS.

# **Desain Arsitektur**

## **Diagram Arsitektur Sistem**



Modul Penilaian dan KHS mengikuti arsitektur sistem utama yang berbasis **MVC (Model-View-Controller)** dan terintegrasi dengan layanan serta komponen pendukung lainnya. Berikut penjelasan arsitekturnya:

**Deskripsi Komponen:**

1. **Frontend (View)**
   * **Dashboard Mahasiswa:** Menampilkan KHS, grafik performa, dan opsi unduh PDF/Excel.
   * **Dashboard Dosen/Admin:** Form input nilai, daftar mata kuliah, dan rekapitulasi nilai.
   * **Responsif:** Menggunakan Bootstrap 5 untuk tampilan yang optimal di berbagai perangkat.
2. **Controller**
   * Mengelola alur proses input nilai, verifikasi, perhitungan IPS/IPK, serta penyajian data KHS.
   * Menghubungkan antara permintaan pengguna dengan data di database.
3. **Model**
   * Mengatur logika bisnis dan pengelolaan data nilai, mahasiswa, serta mata kuliah.
   * Interaksi dengan database melalui ORM Yii2.
4. **Database (MySQL)**
   * Menyimpan data mahasiswa, mata kuliah, nilai per mata kuliah, hasil IPS/IPK, dan log aktivitas.
5. **Export Services**
   * **MPDF:** Untuk menghasilkan KHS dalam format PDF.
   * **PhpSpreadsheet:** Untuk ekspor KHS dalam format Excel.
6. **Visualization Service**
   * **Chart.js:** Untuk menampilkan grafik performa akademik mahasiswa.
7. **Security Layer**
   * **OAuth2 & JWT:** Untuk autentikasi dan manajemen sesi pengguna.
   * **RBAC:** Untuk pembatasan akses berdasarkan peran (admin, dosen, mahasiswa).
8. **Backup & Recovery**
   * Backup otomatis database untuk menjaga keamanan dan kontinuitas data

.

## Komponen Utama Sistem

**1. Frontend Interface**

* **Deskripsi:**  
  Antarmuka pengguna yang responsif untuk input nilai oleh dosen/admin serta tampilan KHS bagi mahasiswa.
* **Teknologi:**  
  Bootstrap 5, HTML, CSS, Chart.js.
* **Fungsi:**
  + Form input nilai.
  + Dashboard mahasiswa untuk melihat KHS dan grafik performa.
  + Fitur download KHS dalam format PDF/Excel.

**2. Controller**

* **Deskripsi:**  
  Pengatur alur data dari pengguna ke sistem backend.
* **Teknologi:**  
  Yii2 MVC Controller.
* **Fungsi:**
  + Memproses input nilai dari form.
  + Mengelola permintaan generate KHS.
  + Mengatur otorisasi akses pengguna sesuai peran.

**3. Model**

* **Deskripsi:**  
  Logika bisnis dan pengolahan data nilai akademik.
* **Teknologi:**  
  Yii2 Active Record ORM.
* **Fungsi:**
  + Menyimpan dan memperbarui data nilai ke database.
  + Menghitung IPS dan IPK.
  + Mengambil data untuk laporan dan visualisasi.

**4. Database**

* **Deskripsi:**  
  Penyimpanan terstruktur untuk data akademik.
* **Teknologi:**  
  MySQL.
* **Fungsi:**
  + Menyimpan data mahasiswa, mata kuliah, nilai, dan log aktivitas.
  + Mendukung indexing untuk mempercepat query nilai dan laporan.

# **Desain Modul dan Komponen**

## **Modul Autentikasi Pengguna**

* Fungsi: Login, logout, registrasi, dan pemulihan kata sandi.
* Input: Username, password, token autentikasi.
* Output: Token sesi pengguna, status autentikasi.
* Teknologi: OAuth2 untuk keamanan autentikasi, JWT untuk token berbasis sesi.
* Validasi: Captcha untuk mencegah serangan brute force.

## **Modul Manajemen Data**

* Fungsi: CRUD untuk entitas pengguna, transaksi, dan data lainnya.
* Input: Data pengguna (nama, email, peran, dsb.).
* Output: Data tersimpan dalam database, respons API.
* Teknologi: ORM Yii2 untuk interaksi database, mekanisme indexing untuk optimasi pencarian data.

## **Modul Laporan dan Analitik**

* Fungsi: Pembuatan laporan berbasis data dengan visualisasi grafik.
* Output: File PDF atau Excel, grafik statistik.
* Teknologi: Library MPDF untuk PDF, PhpSpreadsheet untuk Excel, Chart.js untuk visualisasi data.

## **Modul Integrasi API**

* Fungsi: Komunikasi dengan sistem eksternal seperti sistem pembayaran dan layanan pihak ketiga.
* Input: Data permintaan dalam format JSON.
* Output: Respons API dari layanan eksternal.
* Teknologi: Implementasi RESTful API dengan autentikasi berbasis API key.

# **Desain Basis Data**

## **Model Data**

Struktur utama basis data mencakup tabel:

1. **users** (id, nama, email, password, role, created\_at, updated\_at).
2. **transactions** (id, user\_id, amount, date, status).
3. **logs** (id, user\_id, activity, timestamp, ip\_address).
4. **api\_keys** (id, user\_id, key, created\_at, expired\_at).

## **Diagram Entity-Relationship (ERD)**

(Diagram ERD dapat disertakan sebagai gambar yang menunjukkan relasi antar tabel.) **4.3 Skema Indexing dan Optimasi**

* **Indexing pada kolom pencarian utama** untuk meningkatkan performa query.
* **Partisi tabel transaksi** berdasarkan periode untuk optimasi pengambilan data.
* **Backup otomatis setiap 24 jam** untuk memastikan pemulihan data jika terjadi kegagalan.

# **Antarmuka Pengguna**

## **Wireframe Desain**

* **Halaman Login**: Form input username dan password dengan opsi autentikasi dua faktor.
* **Dashboard**: Menampilkan ringkasan data dan laporan interaktif dengan filter.
* **Halaman CRUD**: Form input dan tabel data dengan fitur sorting dan pagination.

## **Desain Responsif**

* Menggunakan Bootstrap 5 untuk memastikan kompatibilitas dengan berbagai perangkat.
* Layout fleksibel dengan grid system untuk tampilan yang lebih dinamis.
* Penggunaan AJAX untuk mempercepat interaksi pengguna tanpa perlu reload halaman.

# **Pertimbangan Keamanan**

* **Enkripsi Data**: Menggunakan AES-256 untuk menyimpan data sensitif.
* **Kontrol Akses**: Implementasi Role-Based Access Control (RBAC) untuk membatasi hak akses pengguna.
* **Proteksi API**: Penggunaan API key dan token autentikasi untuk mencegah akses tidak sah.
* **Audit Log**: Penyimpanan semua aktivitas pengguna untuk keamanan dan kepatuhan.
* **Firewall Aplikasi Web (WAF)**: Mencegah serangan SQL Injection dan XSS.

# **Pengujian dan Validasi**

## **Pengujian Unit**

* Pengujian tiap modul sistem dengan PHPUnit.
* Simulasi skenario input valid dan tidak valid.

## **Pengujian Keamanan**

* Pengujian penetrasi untuk mengidentifikasi kerentanan sistem.
* Uji beban untuk memastikan sistem dapat menangani lonjakan pengguna.

## **Uji Fungsionalitas**

* Pengujian fitur utama sistem berdasarkan spesifikasi kebutuhan.
* Penggunaan Selenium untuk mengotomatiskan pengujian UI.

# **Lampiran**

* Diagram kelas dan urutan proses sistem.
* Dokumentasi API untuk integrasi eksternal.
* Laporan hasil pengujian dan rekomendasi perbaikan.