# ****Dokumen Desain Perangkat Lunak (Software Design Document SDD)**** Modul Presensi dan Absensi

## ****1. Pendahuluan****

### ****1.1 Tujuan****

Dokumen ini disusun untuk menjelaskan desain teknis dan arsitektural dari sistem Modul Presensi dan Absensi berbasis web. Tujuannya adalah sebagai panduan bagi tim pengembang dalam membangun sistem sesuai dengan kebutuhan pengguna, serta memastikan bahwa sistem dapat berjalan dengan baik dan efisien.

### ****1.2 Ruang Lingkup****

Sistem ini berfungsi untuk mencatat dan mengelola data kehadiran pengguna (mahasiswa secara otomatis ,mudah dan tidak ribet. Fitur utama yang disediakan antara lain:

* Membuat sistem pencatatan kehadiran yang otomatis dan mudah digunakan.
* Mengurangi kesalahan pencatatan yang sering terjadi pada metode manual.
* Menjaga data kehadiran agar tidak mudah hilang atau disalahgunakan.
* Memastikan sistem bisa diakses lewat komputer maupun smartphone.
* Menyediakan data kehadiran yang rapi dan mudah dibaca.

### ****1.3 Referensi****

* IEEE 1016-2009 – Standar Desain Perangkat Lunak.
* Dokumentasi Yii2 Framework (<https://www.yiiframework.com/doc/>).
* Panduan Bootstrap 5 (<https://getbootstrap.com/docs/5.0/>).
* Panduan OWASP Top 10 – Keamanan Aplikasi Web.

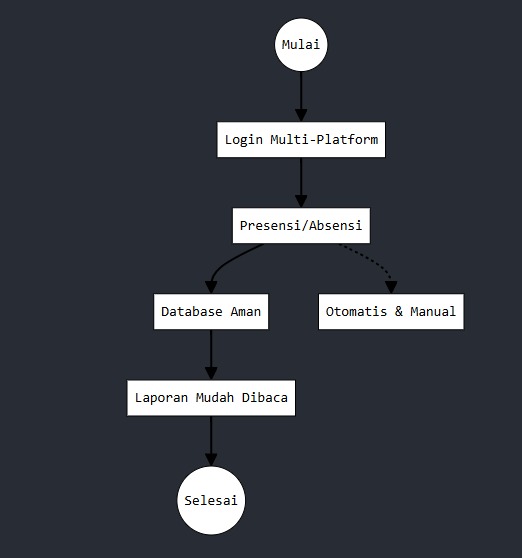
## ****2. Desain Arsitektur****

### ****2.1 Diagram Arsitektur Sistem****

Sistem dibangun menggunakan pola arsitektur **MVC (Model-View-Controller)**. Arsitektur ini memisahkan logika aplikasi, tampilan antarmuka, dan pengolahan data agar lebih mudah dalam pengembangan dan pemeliharaan.

**Komponen Teknologi yang Digunakan:**

* **Backend:** PHP dengan Yii2 Framework
* **Frontend:** HTML5, CSS3, JavaScript, Bootstrap 5
* **Database:** MySQL
* **API:** RESTful API untuk komunikasi dengan sistem lain (misalnya sistem kepegawaian atau payroll)
* **Caching (opsional):** Redis untuk mempercepat akses data tertentu



1. **Login multi-platfom** : mencakup fitur akses dari dari compute dan smarphone
2. **Presensi /absensi** : terhubung dengan opsi otomatis dan manual yang mencangkup fitur percantatan kehadiran otomatis dengan pengurangan kesalahan pencatatan
3. **Database Aman** : Mecangkup fitur keamanan data agar tidak mudah hilang atau disalahgunakan
4. **Lapuran Mudah Dibaca**: mencangkup fitur penyedian data kehadiran yang rapi dan mudah dibaca

**2.2 Komponen Utama Sistem**

1. **Model**: Bertanggung jawab atas pengelolaan data dan interaksi dengan database.

2. **View**: Menyediakan tampilan antarmuka pengguna yang responsif dan interaktif.

3. **Controller**: Mengatur aliran data antara model dan view.

4. **Database**: Struktur data yang disimpan dalam MySQL untuk memastikan konsistensi data.

5. **API Layer**: Digunakan untuk komunikasi antar sistem dan mendukung integrasi eksternal.

6. **Caching Layer**: Menggunakan Redis untuk mempercepat pengambilan data yang sering diakses.

**3. Desain Modul dan Komponen**

**3.1 Modul Autentikasi Pengguna**

1. **Login Pengguna**
   * Memasukkan username dan password untuk masuk ke sistem.
   * Verifikasi kredensial melalui database.
2. **Logout**
   * Mengakhiri sesi pengguna secara aman.
3. **Registrasi (Opsional/Admin)**
   * Pendaftaran akun baru oleh admin atau pengguna tertentu.
4. **Hak Akses Pengguna**
   * Admin: mengelola semua data dan pengguna.
   * mahasiswa: hanya bisa melakukan presensi dan melihat data sendiri.

**3.2 Modul Manajemen Data**

* **model :** mengelola dan menyimpan data ke database
* **view :** menampilkan halaman input, table data, dan laporan
* **controller :** mengatur proses CRUD dan komunikasi antar bagian
* **Databese :** menyimpan data pengguna, presensi, izin, dan laporan
* **Api Layer :** integrasi untuk mengakses/ manipulasi data dari sistem eksternal
* **Caching :** mempercepat pengambilan data rekap presensi atau laporan rutin

**3.3 Modul Laporan dan Analitik**

* **model :** mengambil dan mengelola data laporan dari databese
* **view :** menyajikan laporan dalam bentuk tabel dan grafik interaktif
* **controller :** mengatur permintaan laporan dari pengguna dan filter data
* **Databese :** menyimpan semua data presensi ,izin, dan keterlambatan
* **Api Layer :** jika laporan perlu diambil atau dikirim ke sistem lain
* **Grafik (Chart Library) :** menampilkan data dalam bentuk visual (misalnya pie chart,bar chart)

**3.4 Modul Integrasi API**

* **Format API :** RESTful API dengan JSON
* **Keamanan :** Token-based authentication (JWT/API key)
* **Transport :** HTTPS untuk komunikasi yang aman
* **Dokumentasi :** swangger/OpenAPI(untuk dokumentasi endpoint)

**4. Desain Basis Data**

**4.1 Model Data**

Struktur utama basis data mencakup tabel:

* **Id :** ID unik pengguna
* **Nama :** nama lengkap pengguna
* **Password :** password terenkripsi

**4.2 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.

**5. Antarmuka Pengguna**

**5.1 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.

**5.2 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.

**6. Pertimbangan Keamanan**

* **Firewall dan Sistem Deteksi Intrusi**: Implementasikan firewall dan sistem deteksi serta pencegahan intrusi untuk melindungi server dari serangan eksternal.​
* **Pembaruan Sistem**: Lakukan pembaruan rutin pada sistem operasi dan perangkat lunak untuk menutup celah keamanan yang diketahui.

**7. Pengujian dan Validasi**

**7.1 Pengujian Unit**

• Pengujian tiap modul sistem dengan PHPUnit.

• Simulasi skenario input valid dan tidak valid.

**7.2 Pengujian Keamanan**

• Pengujian penetrasi untuk mengidentifikasi kerentanan sistem.

• Uji beban untuk memastikan sistem dapat menangani lonjakan pengguna.

**7.3 Uji Fungsionalitas**

• Pengujian fitur utama sistem berdasarkan spesifikasi kebutuhan.

• Penggunaan Selenium untuk mengotomatiskan pengujian UI.

**8. Lampiran**

* Diagram Arsitektur Sistem
* Entity-Relationship Diagram (ERD
* Laporan hasil pengujian dan rekomendasi perbaikan.