Logo

Description automatically generatedText

Description automatically generatedIcon

Description automatically generated

LAPORAN PROJECT WORK

**SKARIGA ABSENKU: SISTEM ABSENSI PKL BERBASIS TELEGRAM BOT DENGAN LIVE LOCATION DAN DASHBOARD WEB MENGGUNAKAN REACT.JS DAN NEXT.JS**

GILANG ARDHI MAULANA  
24008/2057.063

**GURU PEMBIMBING :**

Yanuar Setyoningsih, S.Pd

Venny Meida Hersianty, S.TR.Kom

**BIDANG KEAHLIAN TEKNOLOGI INFORMASI DAN KOMUNIKASI**

**PROGRAM KEAHLIAN PENGEMBANGAN PERANGKAT LUNAK DAN GIM**

**KONSENTRASI KEAHLIAN REKAYASA PERANGKAT LUNAK**

**SMK PGRI 3 MALANG 2026**

# LEMBAR PERSETUJUAN

JUDUL : SKARIGA ABSENKU: SISTEM ABSENSI PKL BERBASIS TELEGRAM BOT DENGAN LIVE LOCATION DAN DASHBOARD WEB MENGGUNAKAN REACT.JS DAN NEXT.JS

OLEH : GILANG ARDHI MAULANA

NIS : 24008/2057.063

|  |  |
| --- | --- |
| **Mengetahui :** | |
| **Pembimbing 1** | **Pembimbing 2** |
|  |  |
| Yanuar Setyoningsih, S.Pd | Venny Meida Hersianty, S.TR.Kom |
|  |  |
| **Menyetujui :** | |
| **Kepala Bengkel TIK** | **Guru Wali** |
|  |  |
| Eko Mulyanto, S.Pd | Sri Ira Herawati, S.Pd |
|  | |
| **Kepala Bidang Keahlian TIK** | |
|  |  |
| Yanri Nur Wibowo, ST | |

# LEMBAR PENGESAHAN

Judul : SKARIGA ABSENKU: SISTEM ABSENSI PKL BERBASIS TELEGRAM BOT DENGAN LIVE LOCATION DAN DASHBOARD WEB MENGGUNAKAN REACT.JS DAN NEXT.JS

Oleh : GILANG ARDHI MAULANA

NIS : 24008/2057.063

Tanggal Ujian : ………………………….

|  |  |
| --- | --- |
| **Pembimbing :** | |
| **Pembimbing 1** | **Pembimbing 2** |
|  |  |
| Yanuar Setyoningsih, S.Pd | Venny Meida Hersianty, S.TR.Kom | |
|  |  |
| **Penguji :** | |
| **Penguji 1** | **Penguji 2** |
|  |  |
| (……………………………………………….) | (……………………………………………….) |

# UCAPAN TERIMAKASIH

Tugas akhir ini dapat tersusun berkat bantuan, bimbingan dan saran-saran serta masukan dari berbagai pihak hingga laporan ini dapat terselesaikan. Oleh karena itu pada kesempatan ini penulis ingin mengucapkan terima kasih yang sebesar-besarnya kepada :

1. Allah SWT atas Kesehatan dan kelancaran penulis.
2. Kedua Orang Tua atas doa dan dukungannya
3. Bapak Moch. Lukman Hakim, S.T selaku kepala sekolah SMK PGRI 3 Malang
4. Ibu Sri Ira Herawati, S. Pd selaku guru wali.
5. Ibu Yanuar Setyoningsih, S.Pd selaku pembimbing 1.
6. Ibu Venny Meida Hersianty, S.TR.Kom selaku pembimbing 2.
7. Bapak Yanri Nur Wibowo, ST selaku Kepala Bidang Teknik Informatika SMK PGRI 3 Malang.
8. Bapak Eko Mulyanto, S.Pd selaku Kepala Bengkel Teknik Informatika SMK PGRI 3 Malang.
9. Serta teman-teman penulis yang membantu selama penyelesaian laporan dan project.

# KATA PENGANTAR

Segala puji syukur kita panjatkan ke hadirat Allah SWT serta shalawat kepada Nabi Muhammad SAW atas limpahan rahmat-Nya, sehingga penulis dapat menyelesaikan tugas akhir berjudul **"SKARIGA ABSENKU: SISTEM ABSENSI PKL BERBASIS TELEGRAM BOT DENGAN LIVE LOCATION DAN DASHBOARD WEB MENGGUNAKAN REACT.JS DAN NEXT.JS"**. Laporan ini disusun untuk menguraikan komponen serta tahapan yang dilakukan dalam pengembangan sistem absensi tersebut.

Penulis juga mengucapkan terima kasih yang sebesar-besarnya kepada guru pembimbing atas arahan dan ilmu yang diberikan, serta kepada seluruh pihak yang telah membantu kelancaran proyek ini. Semoga laporan tugas akhir ini dapat memberikan manfaat dan kontribusi positif.

|  |  |
| --- | --- |
|  |  |
|  | Malang, |
|  | Penulis |
|  |  |
|  |  |
|  | (Gilang Ardhi Maulana)  24008/2057.063 |

# Daftar Isi

[LEMBAR PERSETUJUAN 1](#_Toc221830125)

[LEMBAR PENGESAHAN 2](#_Toc221830126)

[UCAPAN TERIMAKASIH 3](#_Toc221830127)

[KATA PENGANTAR 4](#_Toc221830128)

[Daftar Isi 5](#_Toc221830129)

[Daftar Gambar 8](#_Toc221830130)

[Daftar Tabel 9](#_Toc221830131)

[BAB I PENDAHULUAN 10](#_Toc221830132)

[1.1 Latar Belakang 10](#_Toc221830133)

[1.2 Rumusan Masalah 11](#_Toc221830134)

[1.3 Tujuan dan Manfaat 12](#_Toc221830135)

[1.4 Batasan Masalah 13](#_Toc221830136)

[1.5 Metode Penelitian 13](#_Toc221830137)

[1.6 Sistematika Penulisan 14](#_Toc221830138)

[1.7 Rencana Kegiatan 15](#_Toc221830139)

[BAB II TINJAUAN PUSTAKA 16](#_Toc221830140)

[2.1 Tinjauan Teori 16](#_Toc221830141)

[*2.2* *Sistem Absensi* 17](#_Toc221830142)

[*2.3* *Telegram Bot* 18](#_Toc221830143)

[2.4 *Node*.js 19](#_Toc221830144)

[2.5 React.js dan Next.js 20](#_Toc221830145)

[A. React JS 20](#_Toc221830146)

[B. Next JS 20](#_Toc221830147)

[2.6 *JavaScript* 21](#_Toc221830148)

[2.7 *TypeScript* 22](#_Toc221830149)

[2.8 Prisma 24](#_Toc221830150)

[2.9 Teknologi *Geolocation* dan Location Based Service 25](#_Toc221830151)

[2.10 Database 26](#_Toc221830152)

[2.11 Visual Studio Code 27](#_Toc221830153)

[2.12 Flowchart 28](#_Toc221830154)

[2.13 DFD 29](#_Toc221830155)

[2.14 ERD 31](#_Toc221830156)

[BAB III PROSES PERANCANGAN 33](#_Toc221830157)

[3.1 DFD (Data *Flow* Diagram) 33](#_Toc221830158)

[3.2 Flowchart 35](#_Toc221830159)

[3.3 ERD 39](#_Toc221830160)

[3.4 Tabel 40](#_Toc221830161)

[*3.5* Rancangan Tampilan Website / *MOCK UP* 44](#_Toc221830162)

[BAB IV HASIL DAN PEMBAHASAN 57](#_Toc221830163)

[4.1 Form Login 57](#_Toc221830164)

[4.2 Implementasi Fitur Utama 58](#_Toc221830165)

[4.2.1 Fitur Validasi Geofencing 58](#_Toc221830166)

[4.2.2 Kalkulasi Keterlambatan Otomatis 59](#_Toc221830167)

[4.2.3 Kalkulasi Keterlambatan Otomatis 59](#_Toc221830168)

[4.4 Tampilan Halaman Dashboard 60](#_Toc221830169)

[4.5 Tampilan Daftar Absensi 61](#_Toc221830170)

[4.6 Tampilan Daftar Jurnal 62](#_Toc221830171)

[4.7 Tampilan Halaman Daftar Perusahaan 63](#_Toc221830172)

[4.8 Tampilan Hasil Daftar Pengguna 64](#_Toc221830173)

[4.9 Tampilan Daftar Pengguna Tertunda 65](#_Toc221830174)

[4.10 Tampilan Halaman Mengatur Profil 66](#_Toc221830175)

[4.11 Tampilan Halaman Mengatur Kata Sandi 67](#_Toc221830176)

[4.12 Tampilan Menu Pengguna 68](#_Toc221830177)

[4.13 Implementasi Database (Prisma Schema) 69](#_Toc221830178)

[BAB V KESIMPULAN DAN SARAN 70](#_Toc221830179)

[5.1 Kesimpulan 70](#_Toc221830180)

[5.2 Saran 71](#_Toc221830181)

[DAFTAR PUSTAKA 73](#_Toc221830182)

[BAB VI BIODATA PENULIS 76](#_Toc221830183)

# Daftar Gambar

[**Gambar 1** Sistem Absensi 18](#_Toc221830062)

[**Gambar 2** Telegram Bot 19](#_Toc221830063)

[**Gambar 3** Node.js 20](#_Toc221830064)

[**Gambar 4** React.js dan Next.js 21](#_Toc221830065)

[**Gambar 5** Java Script 22](#_Toc221830066)

[**Gambar 6** Typescript 23](#_Toc221830067)

[**Gambar 7** Prisma ORM 25](#_Toc221830068)

[**Gambar 8** Geolocation 26](#_Toc221830069)

[**Gambar 9** Database 27](#_Toc221830070)

[**Gambar 10** Visual Studio Code 28](#_Toc221830071)

[**Gambar 11** Flowchart 29](#_Toc221830072)

[**Gambar 12** DFD 30](#_Toc221830073)

[**Gambar 13** ERD 32](#_Toc221830074)

[**Gambar 14** DFD Level 0 34](#_Toc221830075)

[**Gambar 15** DFD Level 1 35](#_Toc221830076)

[**Gambar 16** Flowchart User 38](#_Toc221830077)

[**Gambar 17** FLowchart Admin 39](#_Toc221830078)

[**Gambar 18** ERD 40](#_Toc221830079)

[**Gambar 19** Design Halaman Login 45](#_Toc221830080)

[**Gambar 20** Design Halaman Dashboard 46](#_Toc221830081)

[**Gambar 21** Daftar Absensi 47](#_Toc221830082)

[**Gambar 22** Design Halaman Daftar Perusahaan 47](#_Toc221830083)

[**Gambar 23** Design Halaman Daftar Jurnal 48](#_Toc221830084)

[**Gambar 24** Design Halaman Daftar Pengguna 48](#_Toc221830085)

[**Gambar 25** Design Halaman Pengguna Tertunda 49](#_Toc221830086)

[**Gambar 26** Design Halaman Pengaturan 49](#_Toc221830087)

[**Gambar 27** Design Halaman Ganti Password 50](#_Toc221830088)

[**Gambar 28** Design Halaman Menu User 51](#_Toc221830089)

[**Gambar 29** Design Halaman Daftar Akun 52](#_Toc221830090)

[**Gambar 30** Design Halaman Mulai 53](#_Toc221830091)

[**Gambar 31** Design Halaman Panduan 54](#_Toc221830092)

[**Gambar 32** Design Halaman Profil 55](#_Toc221830093)

[**Gambar 33** Design Halaman Rekap Absen 56](#_Toc221830094)

[**Gambar 34** Design Halaman Jurnal 57](#_Toc221830095)

[**Gambar 35** Form Login 58](#_Toc221830096)

[Gambar 36 Implementasi Validasi Geofencing 59](#_Toc221830097)

[**Gambar 37** Implementasi Kalkulasi Keterlabatan Otomatis 60](#_Toc221830098)

[**Gambar 38** Implementasi Kalkulasi Keterlabatan Otomatis 60](#_Toc221830099)

[**Gambar 39** Halaman Dashboard 61](#_Toc221830100)

[**Gambar 40** Halaman Daftar Absensi 62](#_Toc221830101)

[**Gambar 41** Halaman Daftar Jurnal 63](#_Toc221830102)

[**Gambar 42** Halaman Daftar Perusahaan 64](#_Toc221830103)

[**Gambar 43** Tampilan Daftar Pengguna 65](#_Toc221830104)

[**Gambar 44** Tampilan Pengguna User Tertunda 66](#_Toc221830105)

[**Gambar 45**  Halaman History Pembelian 67](#_Toc221830106)

[**Gambar 46** Halaman Admin 68](#_Toc221830107)

[**Gambar 47** Halaman Menu Pengguna 69](#_Toc221830108)

[**Gambar 48** Implementasi Database 70](#_Toc221830109)

# Daftar Tabel

[**Tabel 1** Rencana Kegiatan 14](#_Toc221827251)

[**Tabel 2** Company 40](#_Toc221827252)

[**Tabel 3** User 41](#_Toc221827253)

[**Tabel 4** Absen 42](#_Toc221827254)

[**Tabel 5** Jurnal 43](#_Toc221827255)

# PENDAHULUAN

## Latar Belakang

Di tengah pesatnya perkembangan teknologi informasi, *digitalisasi* administrasi sekolah menjadi langkah krusial untuk meningkatkan efisiensi, khususnya dalam pelaksanaan Praktik Kerja Lapangan (PKL). Namun, metode pengawasan konvensional yang masih mengandalkan absensi manual sering kali tidak efektif dan rentan terhadap manipulasi data. Guru pembimbing menghadapi kendala dalam memverifikasi keberadaan siswa secara *real-time* karena lokasi magang yang tersebar, mengakibatkan proses *absensi* dan rekapitulasi kehadiran menjadi lambat serta kurang akurat. Sebagai solusi komprehensif, dikembangkanlah sistem "Skariga Absenku" yang memanfaatkan platform Bot Telegram yang terintegrasi dengan teknologi *geolocation*. Sistem ini dirancang untuk memvalidasi posisi siswa secara otomatis menggunakan metode *geofencing*, di mana siswa hanya dapat melakukan absensi jika perangkat mereka terdeteksi berada dalam radius koordinat kantor yang telah ditentukan. Mekanisme ini secara efektif menutup celah kecurangan manipulasi lokasi, memastikan bahwa data kehadiran yang masuk adalah valid dan dapat dipertanggungjawabkan.

Tidak hanya berfokus pada validasi kehadiran, sistem ini juga menyediakan fitur pelaporan jurnal kegiatan harian yang mewajibkan penyertaan bukti foto. Seluruh data yang diinput oleh siswa akan tersimpan secara terstruktur dalam *database* dan dapat diakses langsung oleh guru pembimbing melalui . Dengan demikian, inovasi ini tidak hanya menjamin transparansi data, tetapi juga memudahkan pihak sekolah dalam memantau kedisiplinan dan kinerja siswa secara praktis, terukur, dan mendukung terciptanya lingkungan administrasi yang *paperless*.

## Rumusan Masalah

1. Bagaimana merancang dan membangun sistem absensi berbasis *Bot* Telegram yang memudahkan siswa dalam melakukan pencatatan kehadiran secara *real-time*?
2. Bagaimana menerapkan validasi lokasi dan pembatasan radius (*geofencing*) untuk memastikan keakuratan posisi siswa saat melakukan absensi?
3. Bagaimana mengelola pelaporan jurnal kegiatan harian beserta bukti foto agar dapat tersimpan secara terstruktur dan transparan dalam *database*?

## Tujuan dan Manfaat

* Tujuan dari dibuatnya web ini adalah :

1. Membangun sistem absensi berbasis Bot Telegram yang memungkinkan siswa melakukan pencatatan kehadiran (masuk dan pulang) secara *real-time*.
2. Menerapkan teknologi *geolocation* dan *geofencing* untuk memvalidasi posisi siswa agar sesuai dengan lokasi kantor magang yang telah ditentukan
3. Menyediakan fitur pelaporan jurnal kegiatan harian beserta bukti foto yang terintegrasi langsung ke dalam *database* untuk memudahkan rekapitulasi.

* Manfaat dari dibuatnya aplikasi ini adalah :

1. Mempermudah proses absensi dan pelaporan kegiatan magang secara praktis dan cepat melalui perangkat seluler tanpa perlu mengisi formulir manual.
2. Memberikan kemudahan dalam memantau kedisiplinan dan keberadaan siswa PKL secara akurat, transparan, dan efisien.
3. Meningkatkan akurasi data kehadiran, meminimalisir risiko manipulasi data, serta mendukung efisiensi pengelolaan data secara *digital*.

## Batasan Masalah

* 1. Sistem ini berjalan pada platform Telegram, sehingga pengguna wajib memiliki akun Telegram, koneksi internet yang stabil, serta mengaktifkan fitur *GPS* pada perangkat untuk validasi absensi.
  2. Pembuatan *Website* ini menggunakan beberapa *software* diantaranya:

1. *Visual Studio Code* (Sebagai *Text Editor*)
2. Node.js (Sebagai *Environment* *Server*).
3. Prisma ORM (Sebagai Manajemen *Database*).

## Metode Penelitian

Penulis akan melakukan studi literatur dan merancang alur interaksi bot yang fokus pada kemudahan penggunaan. Pada tahap implementasi, teknologi berbasis *backend* seperti Node.js dan TypeScript akan digunakan, didukung oleh pustaka Telegraf dan Prisma ORM untuk pengelolaan data. Uji coba aplikasi akan melibatkan simulasi validasi lokasi untuk mengidentifikasi kendala teknis, kemudian penulis akan menerapkan perbaikan sistem untuk memastikan bot absensi ini berjalan akurat, *responsif*, dan memenuhi kebutuhan operasional siswa PKL.

## Sistematika Penulisan

Adapun sistematika penulisan untuk menyusun Laporan Tugas Akhir adalah sebagai berikut:

**BAB I PENDAHULUAN**

Berisi mengenai Latar Belakang, Perumusan Masalah, Tujuan dan Manfaat Penelitian, Batasan Masalah, Metode Penelitian, Sistematika Penulisan dan Rencana Kegiatan.

**BAB II TINJAUAN PUSTAKA**

Dalam bab ini nantinya akan dijelaskan tentang beberapa hal yang berhubungan dengan pengembangan aplikasi dan mekanisme yang digunakan

**BAB III PROSES PERANCANGAN**

Bab ini akan menjelaskan tentang proses perancangan aplikasi mulai dari alur, method, hingga *display* program yang akan dibuat.

**BAB IV HASIL PEMBAHASAN**

Bab ini akan memaparkan tahapan penelitian mulai dari analisis, desain, hingga hasil testing dan implementasinya.

**BAB V PENUTUP**

Berisi kesimpulan dan saran yang mendukung untuk pengembangan sistem dari konsep yang telah ditunjukkan.

## Rencana Kegiatan

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Kegiatan | Bulan | | | | | | | |
| Januari | | | | Februari | | | |
| 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 |
| Bab 1 |  |  |  |  |  |  |  |  |
| Bab 2 |  |  |  |  |  |  |  |  |
| Bab 3 |  |  |  |  |  |  |  |  |
| Bab 4 |  |  |  |  |  |  |  |  |
| Bab 5 |  |  |  |  |  |  |  |  |
| Pengerjaan Project |  |  |  |  |  |  |  |  |

**Tabel 1** Rencana Kegiatan

# TINJAUAN PUSTAKA

## Tinjauan Teori

Dalam era transformasi digital saat ini, hampir seluruh aspek kehidupan terintegrasi dengan teknologi informasi dan komunikasi, termasuk dalam sektor pendidikan dan administrasi. Salah satu elemen kunci dalam perkembangan ini adalah pemanfaatan *chatbot* dan teknologi *location-based service*. Aplikasi pesan instan seperti Telegram kini telah berevolusi, tidak hanya berfungsi sebagai alat komunikasi personal, tetapi juga menjadi media strategis bagi institusi pendidikan untuk mengelola data siswa secara otomatis dan waktu nyata (*real-time*). Sistem absensi modern tidak lagi sekadar mencatat kehadiran, melainkan mencakup validasi kehadiran berbasis lokasi (*geofencing*), pelaporan jurnal kegiatan harian digital, hingga pengawasan jarak jauh. Integrasi teknologi ini menjadi solusi krusial bagi sekolah untuk memantau kedisiplinan siswa Praktik Kerja Lapangan secara akurat, transparan, dan efisien, menggantikan metode konvensional yang rentan terhadap manipulasi data.

## *Sistem Absensi*

**Gambar 1** Sistem Absensi

**Sumber**: https://share.google/aLIIapJIXbX1RrW4o

adalah metode atau prosedur yang digunakan untuk mencatat kehadiran seseorang di suatu tempat atau instansi. Dalam konteks pendidikan dan pekerjaan, absensi berfungsi sebagai indikator kedisiplinan dan dasar penilaian kinerja. Secara umum, sistem absensi terbagi menjadi dua, yaitu manual dan digital. Sistem absensi manual seringkali memiliki kelemahan seperti risiko kehilangan data, manipulasi kehadiran , dan rekapitulasi data yang memakan waktu lama. Sebaliknya, sistem absensi digital atau elektronik memanfaatkan teknologi untuk memverifikasi kehadiran secara *real-time*, menyimpan data secara terpusat di *database*, dan mengurangi potensi kecurangan. Dalam proyek ini, sistem absensi yang dibangun bersifat *mobile-based* menggunakan Telegram Bot untuk memudahkan akses siswa PKL.

## *Telegram Bot*

**Gambar 2.2** **Telegram Bot**

**Sumber**: https://share.google/YbyU763Vde90O2aHZ

**Gambar 2** Telegram Bot

Telegram *Bot* adalah akun khusus dalam platform Telegram yang dioperasikan oleh program komputer (perangkat lunak), bukan oleh manusia. *Bot* ini dirancang untuk merespons pesan secara otomatis, menjalankan perintah tertentu, dan berinteraksi dengan pengguna melalui antarmuka obrolan (*chat interface*).Dalam konteks sistem absensi, Telegram *Bot* berfungsi sebagai antarmuka utama bagi siswa untuk mengirimkan data kehadiran. Keunggulan penggunaan Telegram *Bot* meliputi:

* **Efisiensi:** Tidak perlu mengunduh aplikasi tambahan karena terintegrasi dalam aplikasi Telegram.
* **Kecepatan:** Respons otomatis (*auto-reply*) yang *real-time*.
* **Keamanan:** Mendukung *enkripsi* data end-to-end standar Telegram.

## *Node*.js

**Gambar 3** Node.js

**Gambar 2.3** Node.js

**Sumber**: https://www.tridevitsolution.com/node

Node.js adalah lingkungan waktu jalan (*runtime* *environment*) untuk JavaScript yang bersifat *open-source* dan *cross-platform*. Berbeda dengan PHP yang berjalan di sisi server web tradisional, Node.js memungkinkan pengembang menjalankan kode JavaScript di sisi server (*backend*).Dalam proyek ini, Node.js berfungsi sebagai "otak" yang menjalankan logika *bot*, memproses permintaan lokasi siswa, dan menghubungkan *bot* dengan *database*. Keunggulan Node.js meliputi:

* **Non-blocking :** Mampu menangani banyak permintaan absensi secara bersamaan tanpa membekukan sistem.
* **Menggunakan Satu Bahasa:** Memungkinkan penggunaan JavaScript/TypeScript di sisi server dan klien.

## React.js dan Next.js

**Gambar 4** React.js dan Next.js

**Gambar 2.4** React.js dan Next.js

**Sumber**:

https://share.google/HxcYN6Hwruc6WMehG

### React JS

Adalah pustaka (library) JavaScript yang dikembangkan oleh Facebook untuk membangun antarmuka pengguna. React memungkinkan pembuatan komponen *UI* yang dapat digunakan kembali, seperti komponen "Kartu Siswa" atau "Tabel Absen", yang membuat pengembangan web menjadi lebih efisien.

### Next JS

Adalah kerangka (*framework*) kerja berbasis React yang digunakan untuk membangun aplikasi web modern. Dalam sistem "Skariga Absenku", Next.js digunakan untuk membangun Website Dashboard Admin. Keunggulan Next.js:

1. Server-Side Rendering (SSR): Halaman dimuat lebih cepat karena diproses di server sebelum dikirim ke browser.
2. Routing Otomatis: Memudahkan navigasi antar halaman admin (misal: dari halaman Home ke halaman Rekap).

## *JavaScript*

**Gambar 5** Java Script

**Gambar 2.5** JavaScript

**Sumber** : https://share.google/OVxgzzPrtAiZoXJJi

JavaScript adalah bahasa pemrograman yang sangat populer digunakan untuk membuat halaman *web* menjadi lebih interaktif dan dinamis. JavaScript berjalan di sisi klien (*browser*) dan bisa melakukan berbagai hal seperti:

1. Mengubah konten halaman tanpa memuat ulang.
2. Menangani input pengguna dan menjalankan logika tertentu berdasarkan aksi pengguna, seperti klik tombol, mengisi form, atau scroll.
3. Menerapkan efek visual seperti animasi, transisi, atau tampilan pop-up.
4. Menghubungkan halaman *web* dengan server untuk mendapatkan atau mengirim data secara *real-time*.

## *TypeScript*

**Gambar 6** Typescript

**Gambar 2.6** Typescript

**Sumber**: https://worldvectorlogo.com/logo/typescript

Adalah bahasa pemrograman *open-source* yang dikembangkan oleh Microsoft. TypeScript merupakan superset sintaksis dari JavaScript, yang berarti semua kode JavaScript yang valid juga adalah kode TypeScript, namun TypeScript menambahkan fitur *static typing* (pengetikan statis).

Dalam proyek ini, TypeScript digunakan untuk meningkatkan keamanan dan kualitas kode. Keunggulan penggunaan TypeScript meliputi:

1. **Type Safety:**

Mencegah kesalahan tipe data saat proses penulisan kode (misalnya, mencegah input teks pada kolom yang seharusnya angka).

1. **IntelliSense:**

Memberikan fitur *autocomplete* yang cerdas pada kode editor (VS Code), sehingga mempercepat proses *coding*.

1. **Integrasi Prisma:**

TypeScript sangat kompatibel dengan Prisma ORM, memastikan data yang diambil dari *database* memiliki struktur yang pasti dan terhindar dari kesalahan.

## Prisma

**Gambar 7** Prisma ORM

**ambar 2. 7** Prisma ORM

**Sumber**: https://plugins.jetbrains.com/plugin/20686-prisma-orm

Prismaadalah *Object-Relational Mapper* (ORM) generasi baru yang dirancang khusus untuk Node.js dan TypeScript. Dalam arsitektur sistem "Skariga Absenku", Prisma berfungsi sebagai jembatan penghubung antara kode program (baik *Bot* Telegram maupun *Dashboard Web*) dengan *database*.

## Teknologi *Geolocation* dan Location Based Service

**Gambar 8** Geolocation

**Gambar 2. 8** Geolocation

**Sumber**: https://play.google.com/store/apps/details?id=com.positionlogic.lbsManager

*Geolocation* adalah teknologi yang digunakan untuk mengidentifikasi posisi geografis dunia nyata dari suatu objek digital, seperti perangkat ponsel pintar. Dalam pengembangan sistem absensi ini, teknologi *Geolocation* memanfaatkan sensor Global Positioning System yang terdapat pada perangkat siswa untuk mendapatkan titik koordinat yang akurat berupa latitude dan longitude.Penerapan teknologi ini dalam sistem absensi dikategorikan sebagai *Location Based Service*, yaitu layanan informasi yang dapat diakses melalui perangkat seluler dengan memanfaatkan kemampuan lokasi perangkat tersebut. Sistem ini menerapkan metode *Geofencing*, yaitu sebuah mekanisme pembatasan wilayah secara digital.

Sistem akan memvalidasi kehadiran dengan cara menghitung jarak antara titik koordinat siswa saat mengirimkan lokasi terkini melalui Telegram dengan titik koordinat kantor tempat magang. Perhitungan jarak ini menggunakan algoritma Haversine Formula untuk memastikan siswa berada dalam radius yang diizinkan saat melakukan presensi.dan diterima oleh sistem jika posisi siswa terdeteksi berada di dalam radius yang telah ditentukan.

## Database

**Gambar 9** Database

**Gambar 2. 9** Database

**Sumber**: https://blogs.powercode.id/author/andri/

*Database* atau basis data, adalah kumpulan data yang terorganisir dan dikelola sesuai dengan aturan tertentu sehingga informasi di dalamnya dapat diakses, diatur, dan diperbarui dengan mudah. Dengan pengelolaan yang baik, pengguna dapat dengan cepat mencari informasi, menghapus data yang tidak diperlukan, atau menyimpan informasi baru.Selain definisi tersebut, *database* juga dapat diartikan sebagai sistem yang digunakan untuk mengumpulkan dan menyimpan *file*, arsip, atau tabel dalam berbagai media elektronik. Sistem ini menghubungkan berbagai informasi dan memungkinkan pengelolaan data yang efisien dalam lingkungan digital.

## Visual Studio Code

**Gambar 2. 10** Visual Studio Code

**Sumber**: https://share.google/3h0aVXwtxmuOEnsJo

**Gambar 10** Visual Studio Code

Visual Studio Code (VS Code) adalah *editor* buatan Microsoft yang beroperasi pada komputer desktop dan kompatibel dengan berbagai sistem operasi. VS Code cocok untuk *JavaScript*, TypeScript, dan Node.js. Selain itu, editor ini memiliki ekosistem *extension* yang luas yang dapat mendukung berbagai bahasa pemrograman dan runtime lainnya, termasuk C++, C#, Java, Python, PHP, Go, dan .NET. Karena serbaguna dan memiliki *extension* yang beragam, Visual Studio Code menjadi pilihan populer di kalangan *developer* untuk beragam tugas pemrograman.

## Flowchart

**Gambar 2. 11** Flowchart

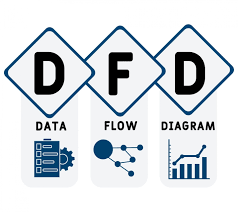
**Sumber:** https://share.google/acx8COxK4gBNjDJvs

**Gambar 11** Flowchart

*Flow*chart adalah sebuah diagram yang menggambarkan alur proses atau alur logika suatu sistem. *Flow*chart menggunakan simbol-simbol standar untuk menunjukkan aktivitas, kondisi, dan alur logika dari proses yang digambarkan. *Flow*chart bisa digunakan dalam berbagai bidang, seperti pengembangan perangkat lunak, perencanaan bisnis, manajemen proyek, desain sistem, dan lain-lain.*Flow*chart dapat membantu dalam menentukan alur proses, menemukan kesalahan atau kekurangan dalam sistem, dan meningkatkan efisiensi suatu proses.

*Flow*chart juga dapat digunakan untuk menjelaskan bagaimana suatu sistem bekerja pada orang yang tidak familiar dengan sistem tersebut. *Flow*chart dapat dibuat menggunakan *Software* khusus atau dibuat secara manual menggunakan simbol-simbol standar.

## DFD



**Gambar 12** DFD

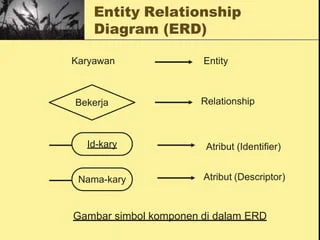
**Sumber:** https://www.shutterstock.com/id/image-vector/dfd-data-flow-diagram-acronym-business-1828668167

Secara umum, pengertian data *flow* diagram yakni gambaran dari alur informasi yang dimulai dari proses input sampai dengan proses output suatu sistem. Penggunaan DFD dilakukan karena ada banyak alur kerja sulit untuk digambarkan dengan deskripsi kata.Diagram banyak digunakan karena dianggap lebih efektif. Terlebih pada proses pengembangan (*development*), DFD mampu menjabarkan dan menganalisis suatu sistem secara lebih mendalam. Analisis inilah yang kemudian akan diterjemahkan dalam *coding*

.

Anda mungkin pernah mendengar istilah *flow*chart. Meskipun mirip, sejatinya antara DFD dan *flow*chart berbeda. *Flow*chart banyak menjelaskan alur kerja dan prosedur pada sistem *Software*, sedangkan DFD hanya menunjukkan alur data pada sistem tersebut. Penulisan data *flow* diagram dilakukan secara paralel atau serentak. Berbeda dengan *flow*chart yang harus dibuat secara berurutan. Meski demikian, keduanya sama-sama dapat dibuat secara manual ataupun menggunakan aplikasi khusus.

## ERD



**Gambar 13** ERD

**Sumber:** https://share.google/KCuCp8i27kNWsfGnc

ERD (*Entity Relationship Diagram*) atau diagram hubungan entitas adalah sebuah diagram yang digunakan untuk perancangan suatu *database* dan menunjukan relasi atau hubungan antar objek atau entitas beserta atribut-atributnya secara detail. Dengan menggunakan ERD, sistem *database* yang sedang dibentuk dapat digambarkan dengan lebih terstruktur dan terlihat rapi. Selain digunakan dalam perancangan *database*, ERD sendiri sering digunakan untuk *debugging* *database* jika terjadi masalah pada *database*. Untuk melakukan *debug* pada *database* bukanlah hal yang mudah, terlebih lagi jika *database* yang mengalami masalah memiliki banyak tabel dan memerlukan penulisan SQL yang kompleks. Dengan menggambarkan skema *database* menggunakan ERD, kamu menjadi lebih mudah untuk menemukan permasalahan yang terjadi dalam *database* dan menyelesaikan masalah dengan mudah.

# PROSES PERANCANGAN

## DFD (Data *Flow* Diagram)

DFD digunakan untuk memodelkan proses-proses yang ada dalam "Sistem Absensi Anak Magang" serta mendefinisikan input, output, dan penyimpanan data yang diperlukan.

##### DFD Level 0

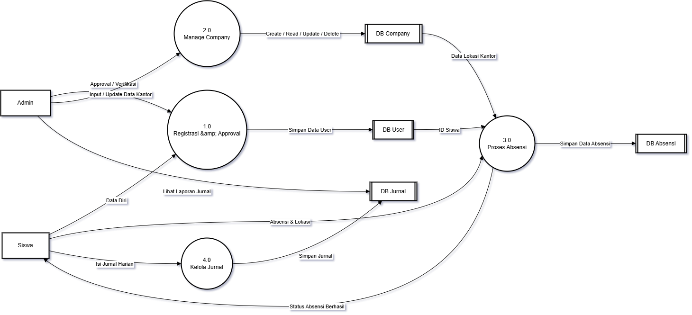
**Gambar 14** DFD Level 0

**Gambar 3. 1** DFD Level 0

DFD Level 0 atau Diagram Konteks merupakan tingkatan tertinggi dalam DFD yang menggambarkan ruang lingkup sistem secara keseluruhan.

1. **Entitas Eksternal:** Terdiri dari dua aktor utama, yaitu **Siswa** dan **Admin**.
2. **Aliran Data Siswa:** Siswa memberikan input berupa data pendaftaran, data absensi, dan data jurnal harian ke dalam sistem. Sebagai output, siswa menerima informasi akun, status, dan notifikasi absensi.
3. **Aliran Data Admin:** Admin melakukan pengelolaan data *company* dan validasi user. Sistem memberikan output berupa laporan kehadiran, laporan jurnal, serta respons atas permintaan laporan yang diajukan oleh Admin.

##### DFD Level 1



**Gambar 15** DFD Level 1

DFD Level 1 merupakan dekomposisi atau penjabaran lebih detail dari proses utama yang ada pada Level 0. Pada level ini, proses dipecah menjadi beberapa sub-proses fungsional:

1. Proses 1.0 (Manage Company): Menangani operasi *CRUD* (*Create, Read, Update, Delete*) terkait data instansi atau perusahaan tempat magang yang disimpan ke dalam Data Store: DB Company.
2. Proses 2.0 (Registrasi & Approval): Mengelola pendaftaran akun siswa dan proses validasi oleh admin. Data akun yang telah disetujui akan disimpan ke dalam Data Store: DB User.
3. Proses 3.0 (Proses Absensi): Mengolah data absensi yang masuk dari siswa dan menyimpannya ke dalam Data Store: DB Absensi. Proses ini juga menyediakan data bagi admin untuk penarikan laporan.
4. Proses 4.0 (Kelola Jurnal): Menangani input jurnal kegiatan harian siswa. Data tersebut disimpan ke dalam Data Store: DB Jurnal agar dapat dipantau oleh admin sebagai laporan progres kerja.

## Flowchart

Flowchart digunakan untuk menggambarkan alur algoritma atau urutan instruksi kerja di dalam sistem, mulai dari interaksi pengguna hingga pemrosesan data oleh admin.

##### 3.2.1 *Flow*chart User

Flowchart ini menggambarkan alur interaksi siswa atau pengguna saat menggunakan bot sistem absensi.

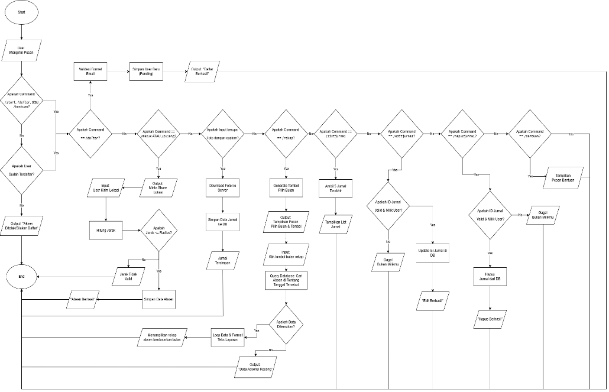
1. **Akses Pesan & Validasi:**

Sistem dimulai dengan menerima pesan dari pengguna dan melakukan pengecekan apakah pengguna sudah terdaftar di sistem. Jika belum, sistem akan meminta pengguna untuk melakukan pendaftaran.

1. **Pemrosesan Perintah (*Command*):**

Pengguna dapat mengirimkan berbagai perintah khusus, di antaranya:

* 1. **/daftar**: Untuk registrasi akun baru.
  2. **/absen**: Mengirimkan lokasi (GPS) untuk validasi jarak (radius) terhadap lokasi perusahaan. Jika berada di dalam radius, absensi berhasil disimpan.
  3. **Input Jurnal**: Mengirimkan foto dan teks deskripsi kegiatan yang kemudian disimpan ke *database*.
  4. **/rekap & /listjurnal**: Mengambil data dari *database* berdasarkan filter bulan atau tanggal untuk ditampilkan kembali kepada pengguna.
  5. **Edit/Hapus Jurnal**: Memberikan fleksibilitas bagi pengguna untuk memperbarui data jurnal harian sebelum divalidasi oleh admin.

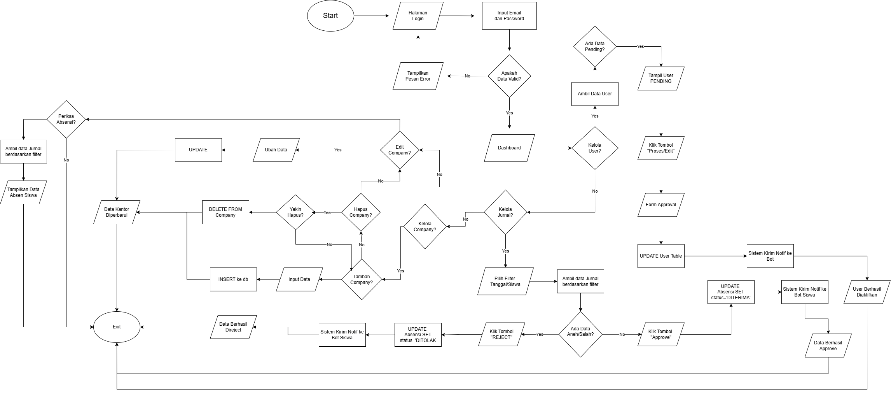


**Gambar 16** Flowchart User

##### *Flow*chart Admin

Flowchart ini berfokus pada fungsi manajerial yang dilakukan oleh admin untuk mengontrol data di sistem.

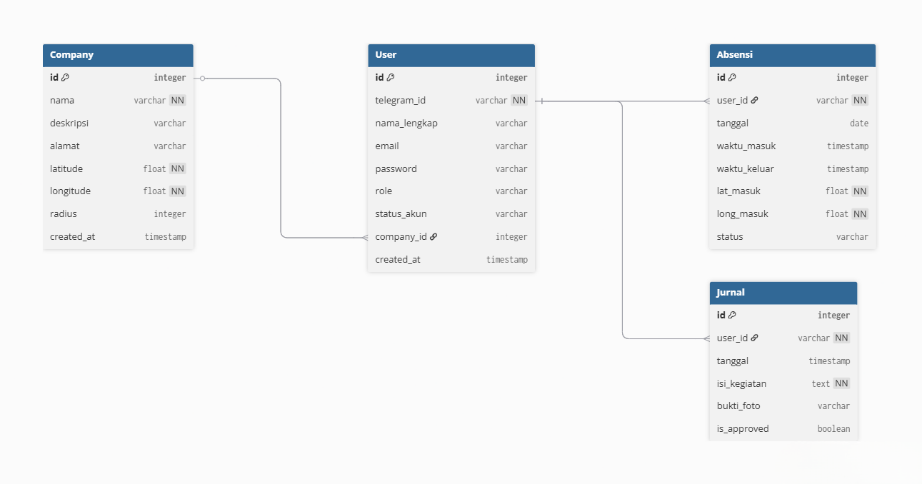
1. Manajemen Akun: Admin menerima notifikasi pendaftaran pengguna baru dan melakukan verifikasi agar pengguna tersebut dapat menggunakan fitur absensi.
2. Manajemen Perusahaan: Admin dapat menambah, mengubah, atau menghapus data lokasi perusahaan, termasuk mengatur koordinat latitude, longitude, dan radius absensi.
3. Monitoring & Laporan: Admin melakukan penarikan data dari *database* untuk menghasilkan laporan kehadiran dan laporan jurnal harian siswa secara kolektif untuk kebutuhan dokumentasi magang.



**Gambar 17** FLowchart Admin

## ERD

ERD menggambarkan struktur *database* dan keterkaitan antar entitas dalam sistem. Relasi dimulai dari Company ke User (*One-to-Many*) untuk menentukan lokasi magang siswa. Selanjutnya, User memiliki relasi *One-to-Many* terhadap Absensi dan Jurnal, di mana setiap data kehadiran dan aktivitas harian siswa terikat secara unik pada identitas akun mereka untuk memudahkan pemantauan oleh Admin.



**Gambar 18** ERD

## Tabel

Perancangan *database* sistem ini terdiri dari beberapa tabel utama yang saling berelasi untuk menyimpan data operasional. Berikut adalah spesifikasi teknisnya:

1. **Tabel Company**

Menyimpan data master instansi magang, termasuk koordinat dan *radius* sebagai acuan validasi lokasi absensi siswa.

**Tabel 2** Company

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Nama Field** | **Tipe Data** | **Value** | **Keterangan** |
| 1 | id | INT | 11 | Primary Key,AUTO\_INCREMENT |
| 2 | nama | VARCHAR | 50 | NOT NULL |
| 3 | deskripsi | VARCHAR | 50 | NUll |
| 4 | alamat | VARCHAR | 50 | NULL |
| 5 | latitude | FLOAT | - | NOT NULL |
| 6 | longitude | FLOAT | - | NOT NULL |
| 7 | radius | INT | 11 | NULL |
| 8 | created\_at | TIMESTAMP | - | CURRENT\_TIMESTAMP |

1. **Tabel User**

Menyimpan identitas siswa dan admin yang terintegrasi dengan ID Telegram unik, serta menghubungkan siswa ke instansi melalui *company\_id*.

**Tabel 3** User

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Nama Field** | **Tipe Data** | **Value** | **Keterangan** |
| 1 | id | INT | 11 | Primary Key,AUTO\_INCREMENT |
| 2 | telegram\_id | VARCHAR | 50 | NOT NULL |
| 3 | nama\_lengkap | VARCHAR | 50 | NULL |
| 4 | email | VARCHAR | 50 | NULL |
| 5 | password | VARCHAR | 50 | NULL |
| 6 | role | VARCHAR | 50 | NULL |
| 7 | status | VARCHAR | 20 | NULL |
| 8 | company\_id | INT | 20 | Foreign Key |
| 9 | created\_at | TIMESTAMP | - | CURRENT\_TIMESTAMP |

1. **Tabel Absensi**

Merekam riwayat kehadiran *real-time* (waktu dan koordinat masuk/pulang) untuk validasi posisi serta pemantauan kedisiplinan.

**Tabel 4** Absen

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Nama Field** | **Tipe Data** | **Value** | **Keterangan** |
| 1 | id | INT | 11 | Primary Key,AUTO\_INCREMENT |
| 2 | user\_id | INT | 11 | Foreign Key |
| 3 | tanggal | DATE | - | NULL |
| 4 | waktu\_masuk | TIMESTAMP | - | NULL |
| 5 | waktu\_keluar | TIMESTAMP | - | NULL |
| 6 | lat\_masuk | FLOAT | - | NOT NULL |
| 7 | long\_masuk | FLOAT | - | NOT NULL |
| 8 | status | VARCHAR | 20 | NULL |

1. **Tabel Jurnal**

Wadah dokumentasi aktivitas harian berupa teks dan bukti foto, dilengkapi status validasi untuk persetujuan laporan oleh admin.

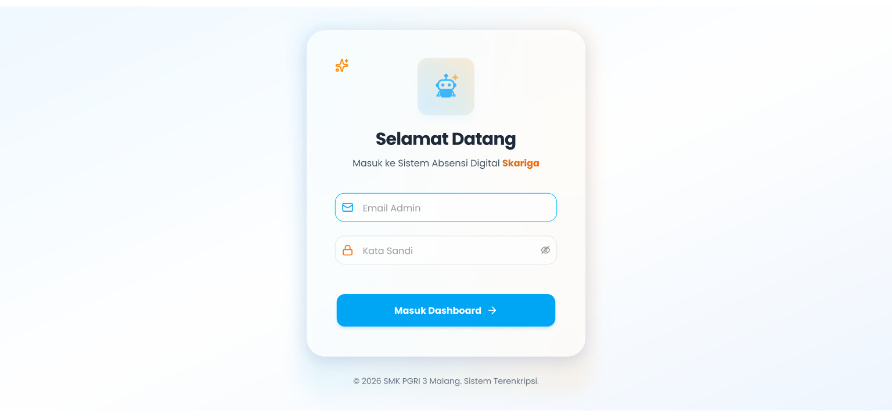
**Tabel 5** Jurnal

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Nama Field** | **Tipe Data** | **Value** | **Keterangan** |
| 1 | id | INT | 11 | Primary Key,AUTO\_INCREMENT |
| 2 | user\_id | INT | 11 | Foreign Key |
| 3 | tanggal | TIMESTAMP | - | NULL |
| 4 | Isi\_kegiatan | TEXT | - | NOT NULL |
| 5 | bukti\_foto | VARCHAR | 50 | NULL |
| 6 | is\_approved | BOOLEAN | - | NULL |

## Rancangan Tampilan Website / *MOCK UP*

Rancangan antarmuka merupakan gambaran *visual* sistem yang akan dibangun untuk memberikan representasi alur kerja kepada pengguna. Rancangan ini dibagi menjadi dua bagian utama, yaitu antarmuka berbasis *web dashboard* yang diperuntukkan bagi admin dalam mengelola data perusahaan serta memantau rekapitulasi absensi, dan antarmuka berbasis *bot* Telegram yang digunakan oleh siswa untuk melakukan registrasi, absensi berbasis lokasi, serta pengisian jurnal harian secara mandiri.

##### 3.5.1 Design Halaman Login



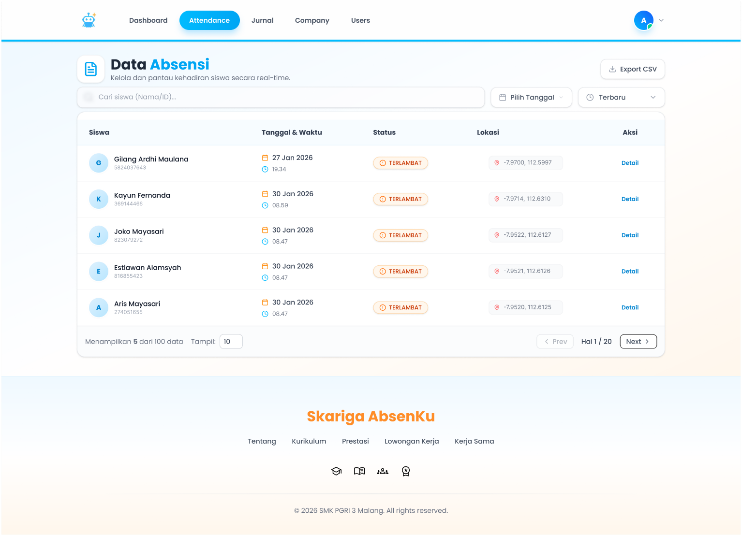
**Gambar 19** Design Halaman Login

##### 3.5.2 Design Halaman Dashboard



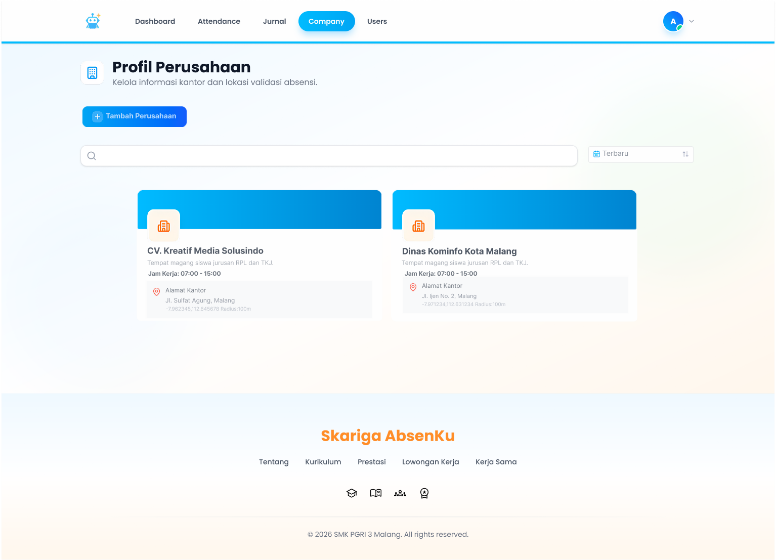
**Gambar 20** Design Halaman Dashboard

##### 3.5.3 Design Halaman Daftar Absensi



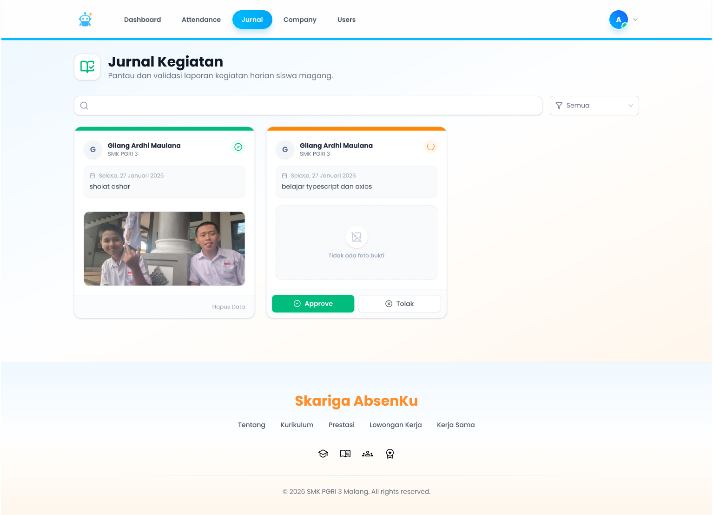
**Gambar 21** Daftar Absensi

##### 3.5.4 Design Halaman Daftar Perusahaan



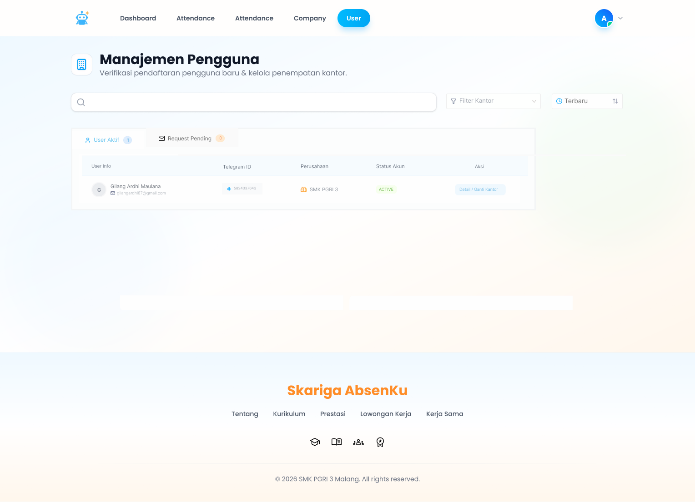
**Gambar 22** Design Halaman Daftar Perusahaan

##### 3.5.5 Design Halaman Daftar Jurnal



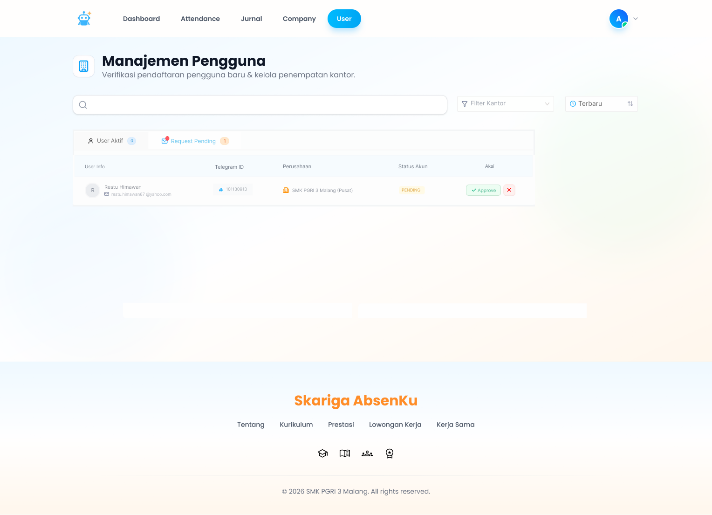
**Gambar 23** Design Halaman Daftar Jurnal

##### 3.5.6 Design Halaman Daftar Pengguna



**Gambar 24** Design Halaman Daftar Pengguna

##### 3.5.7 Design Halaman Daftar Pengguna Tertunda



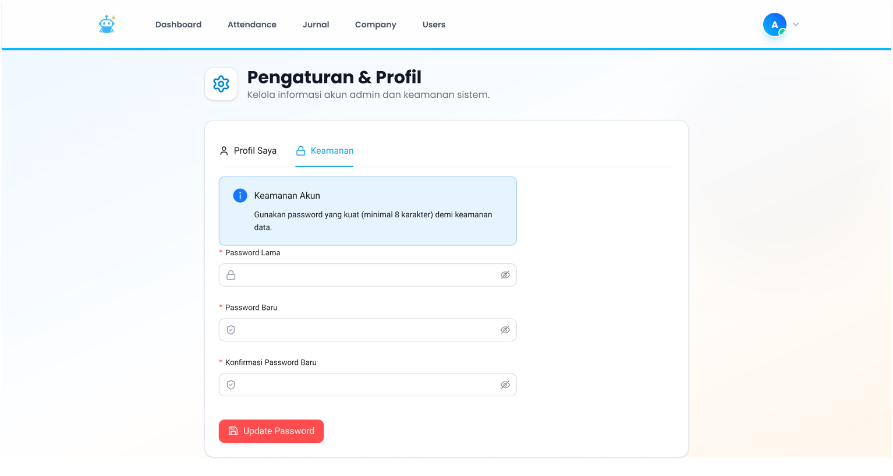
**Gambar 25** Design Halaman Pengguna Tertunda

##### 3.5.8 Design Halaman Pengaturan



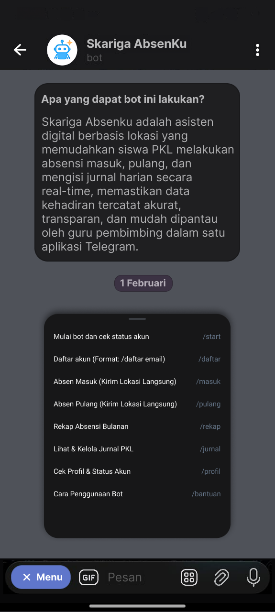
**Gambar 26** Design Halaman Pengaturan

##### 3.5.9 Design Halaman Ganti Password



**Gambar 27** Design Halaman Ganti Password

##### 3.5.10 Design Halaman Menu User



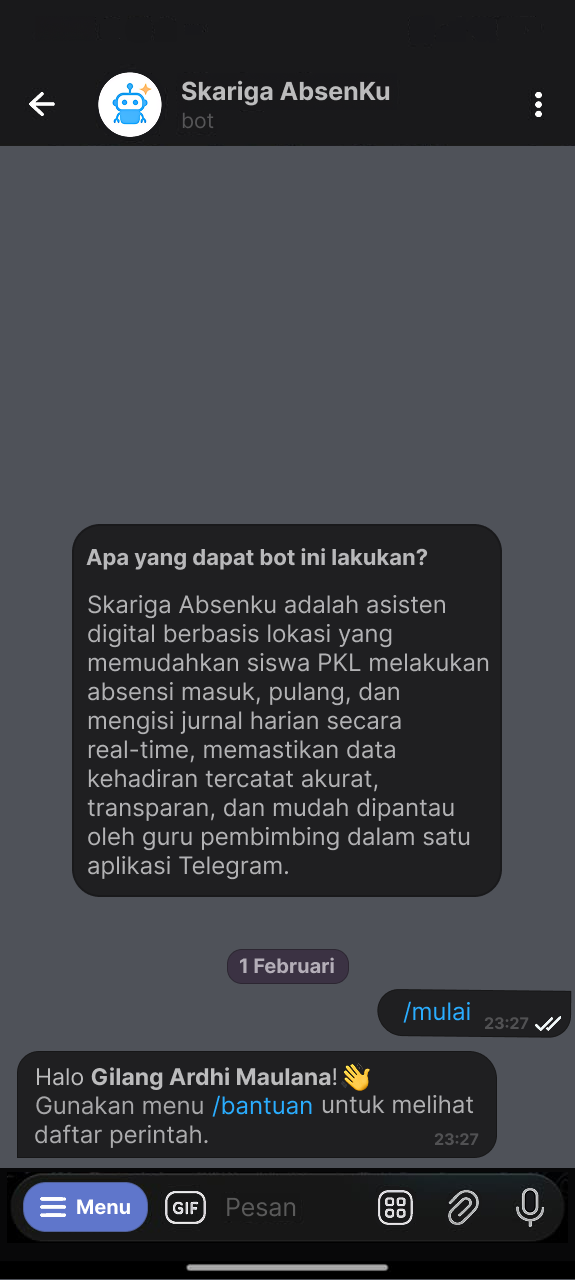
**Gambar 28** Design Halaman Menu User

##### 3.5.11 Design Halaman Daftar Akun



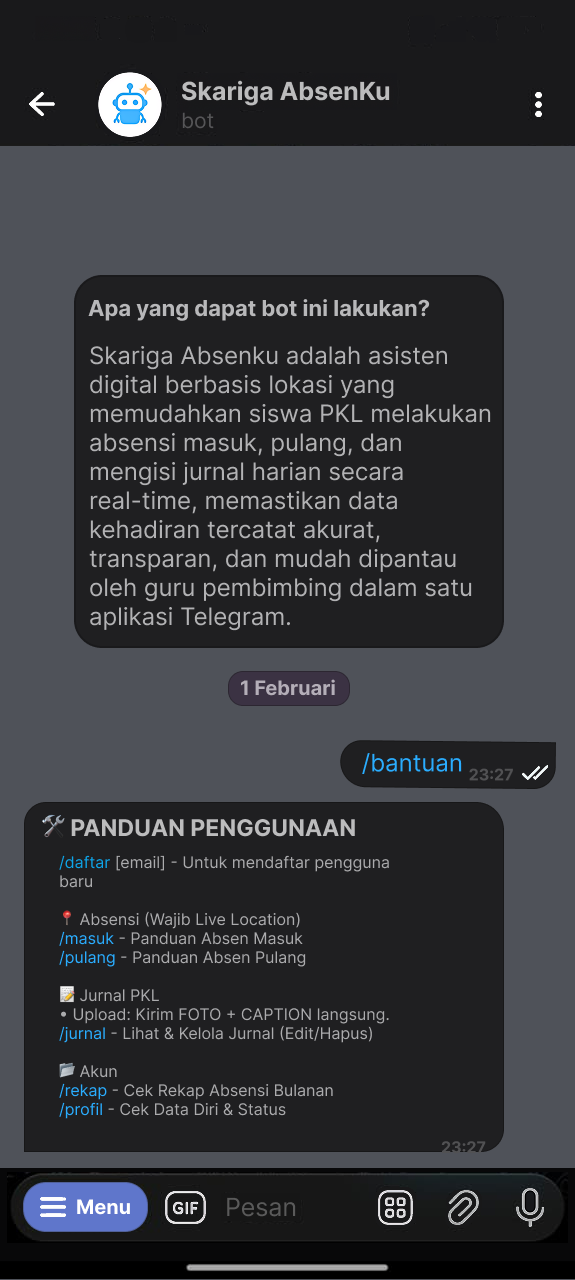
**Gambar 29** Design Halaman Daftar Akun

##### 3.5.12 Design Halaman Mulai



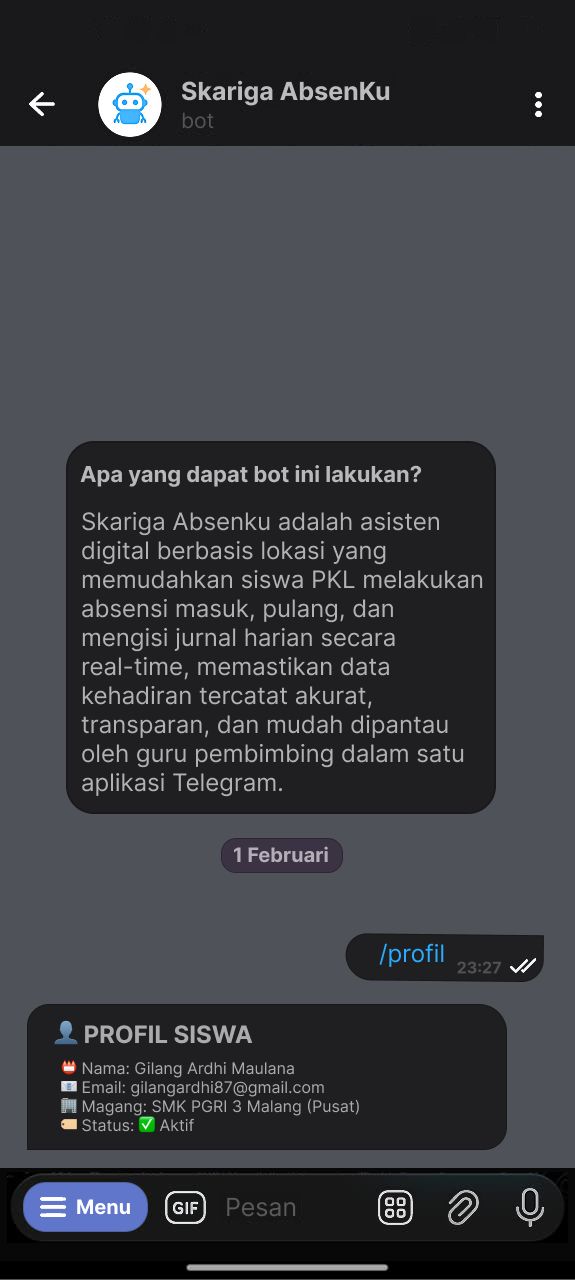
**Gambar 30** Design Halaman Mulai

##### 3.5.13 Design Halaman Panduan



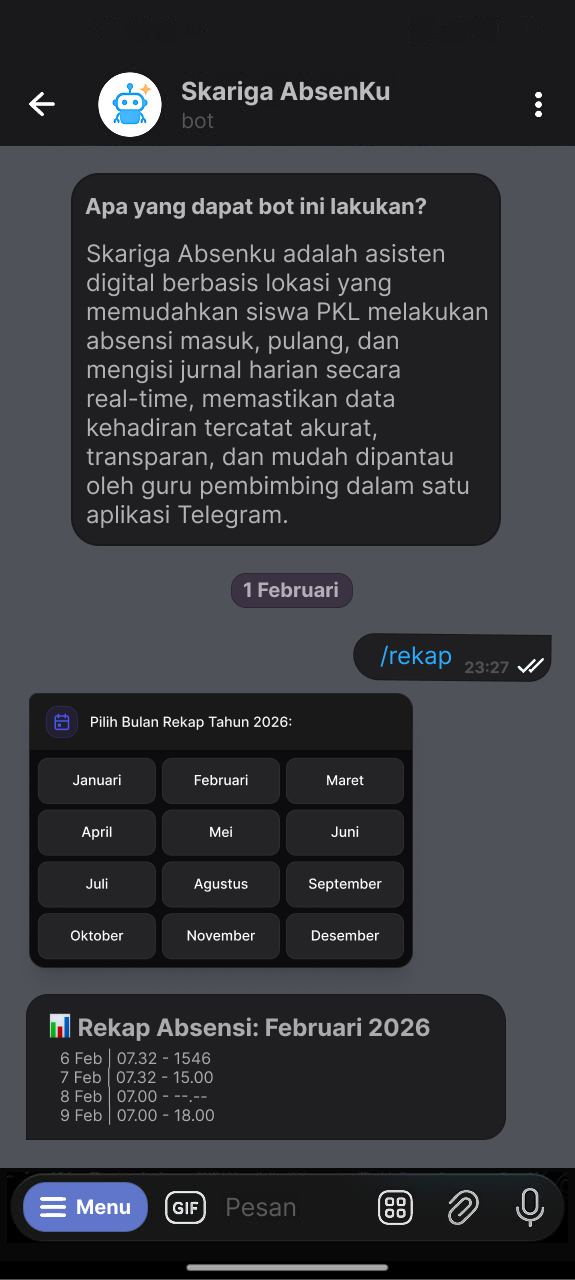
**Gambar 31** Design Halaman Panduan

##### 3.5.14 Design Halaman Profil



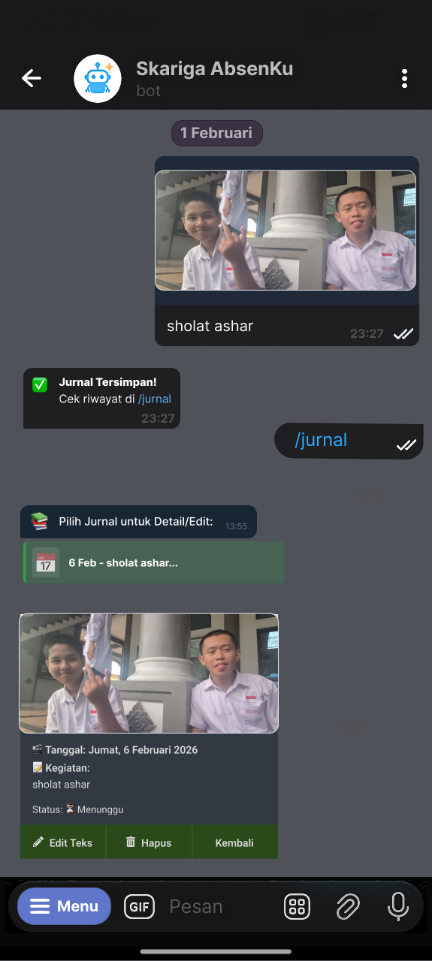
**Gambar 32** Design Halaman Profil

##### 3.5.15 Design Halaman Rekap Absen



**Gambar 33** Design Halaman Rekap Absen

##### 3.5.16 Design Halaman Jurnal

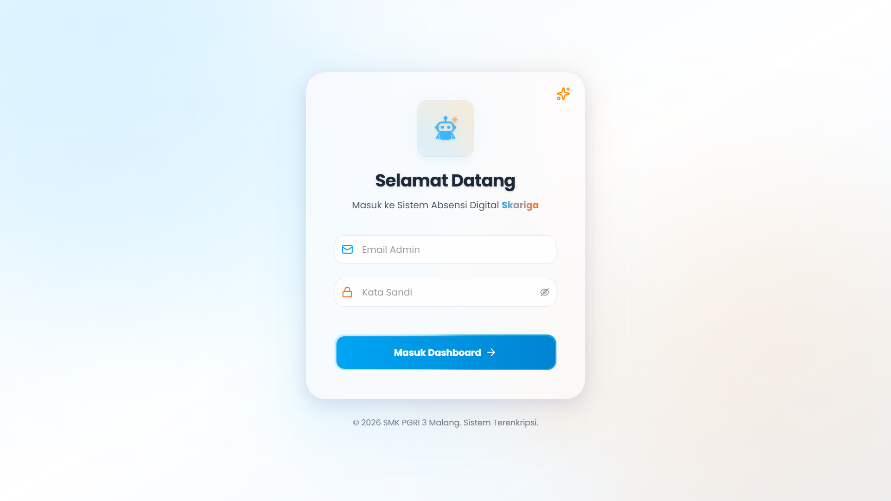


**Gambar 34** Design Halaman Jurnal

# HASIL DAN PEMBAHASAN

## Form Login

Admin masuk menggunakan email dan kata sandi untuk mendapatkan otoritas penuh dalam memverifikasi siswa, memantau absensi dan jurnal, serta mengelola data perusahaan sebagai validator lokasi.



**Gambar 35** Form Login

## Implementasi Fitur Utama

### Fitur Validasi Geofencing

Fitur ini adalah inti dari sistem "Skariga Absenku". Kode ini berfungsi untuk menghitung jarak antara lokasi siswa dan lokasi kantor magang, serta memastikan siswa mengirimkan Live Location, bukan lokasi diam atau palsu.



Gambar Implementasi Validasi Geofencing

### Kalkulasi Keterlambatan Otomatis

Sistem ini secara otomatis membandingkan waktu saat siswa absen dengan jam masuk yang sudah ditentukan oleh admin di database.



**Gambar 37** Implementasi Kalkulasi Keterlabatan Otomatis

### Kalkulasi Keterlambatan Otomatis

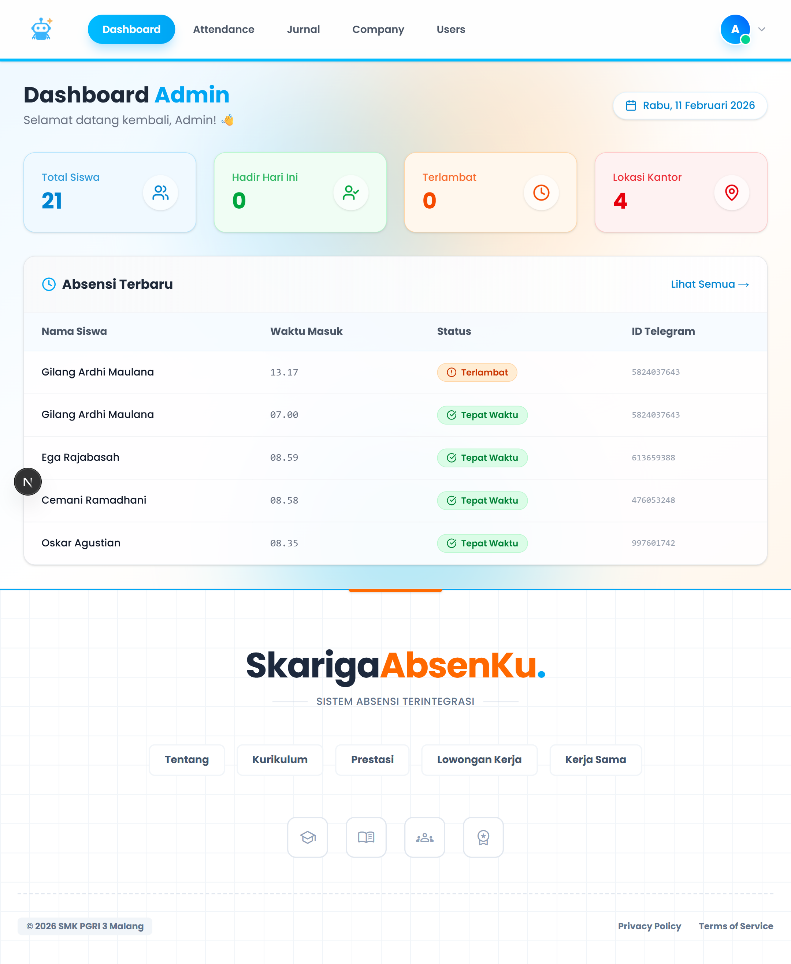
Fitur ini memungkinkan siswa mengunggah foto kegiatan magang yang langsung disimpan datanya dicatat ke basis data.



**Gambar 38** Implementasi Kalkulasi Keterlabatan Otomatis

## Tampilan Halaman Dashboard

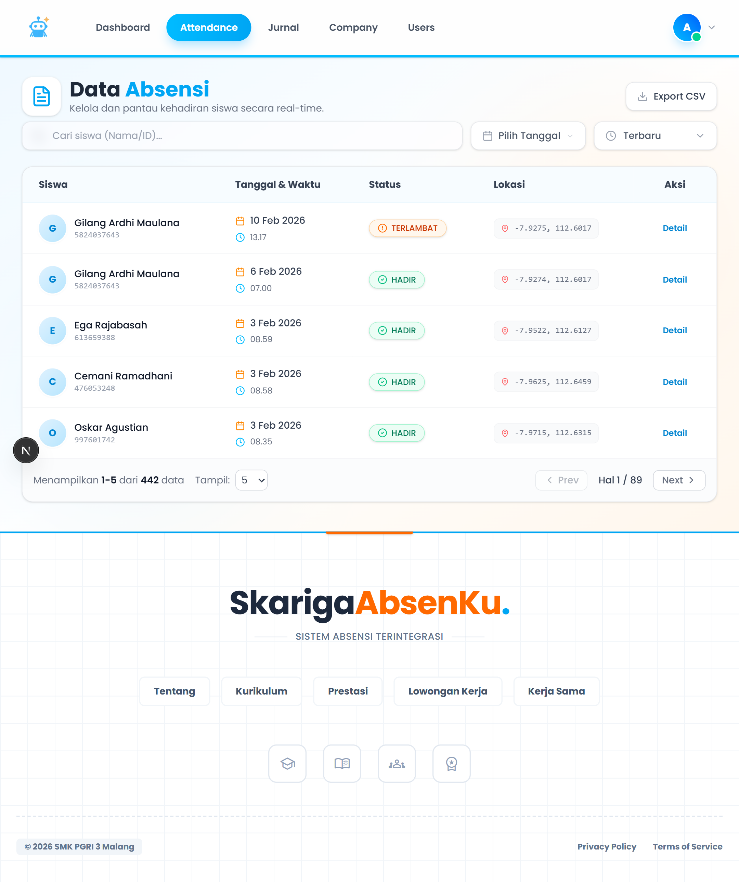
Pusat kendali visual yang menyajikan statistik *real-time* (jumlah siswa, absensi, dan jurnal) untuk mempermudah monitoring aktivitas magang secara efisien dalam satu halaman



**Gambar 39** Halaman Dashboard

## Tampilan Daftar Absensi

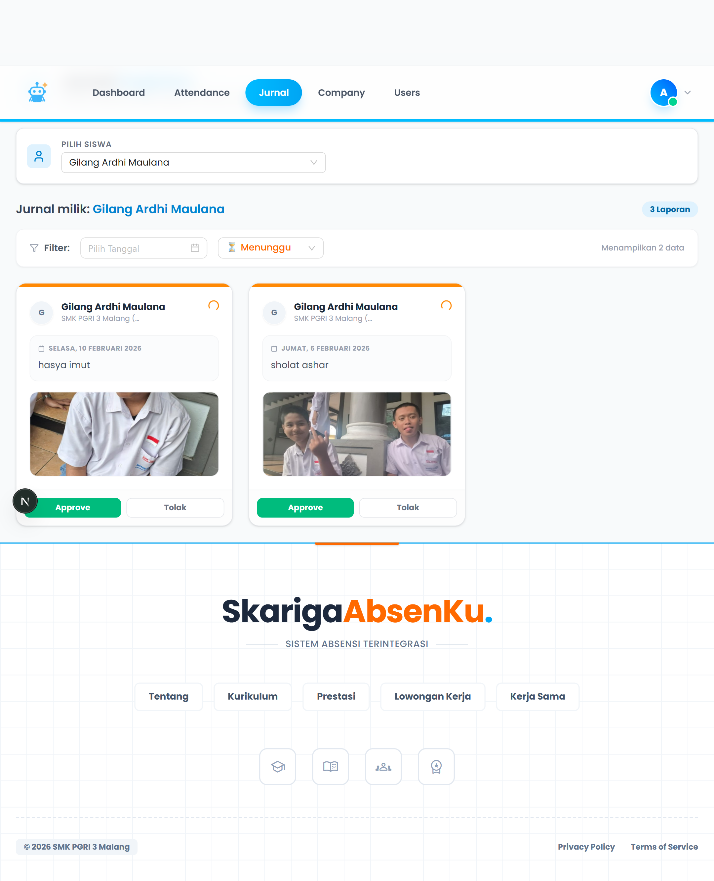
Halaman ini menyajikan riwayat kehadiran siswa secara rinci, mencakup jam masuk, jam pulang, dan keadaan kehadiran. Bagian ini berfungsi sebagai sarana pemantauan kedisiplinan yang memudahkan pengelola memeriksa kebenaran kehadiran serta letak siswa di tiap lembaga secara berurutan sesuai waktu.



**Gambar 40** Halaman Daftar Absensi

## Tampilan Daftar Jurnal

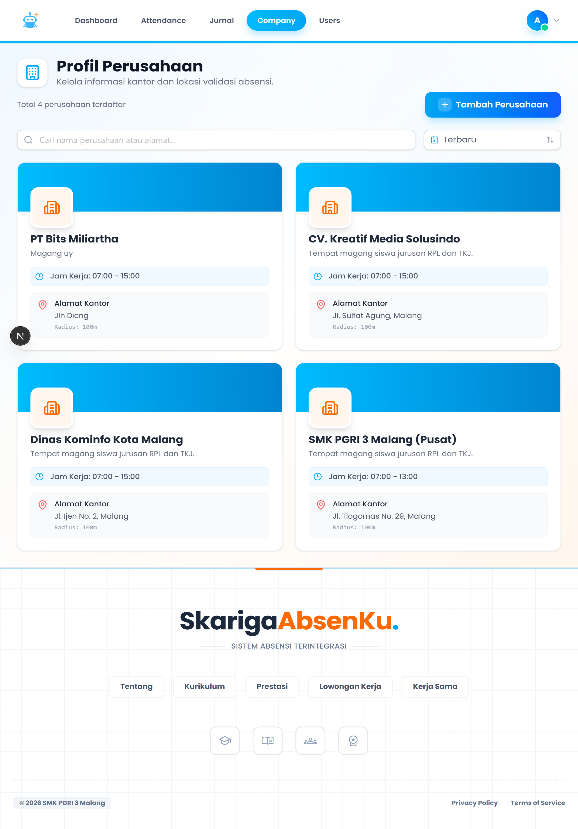
Halaman ini menyediakan rekapitulasi laporan kegiatan harian yang diunggah oleh siswa magang. Admin dapat meninjau rincian pekerjaan, tanggal pelaksanaan, serta status validasi setiap laporan guna memantau produktivitas dan kesesuaian aktivitas siswa dengan program kerja di instansi terkait.



**Gambar 41** Halaman Daftar Jurnal

## Tampilan Halaman Daftar Perusahaan

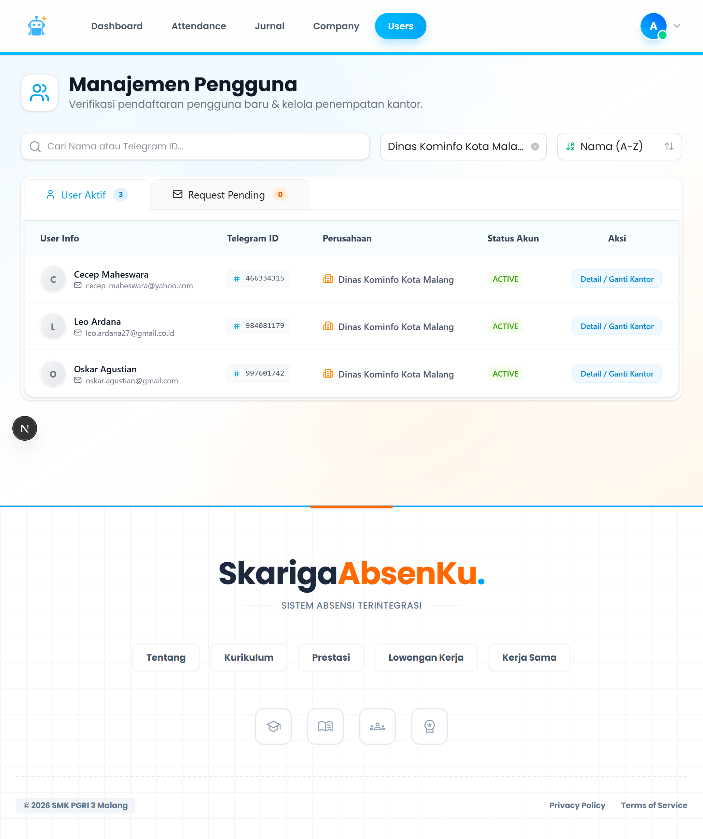
Halaman ini berfungsi untuk mengelola data mitra industri tempat siswa melaksanakan magang. Admin dapat mengatur informasi detail perusahaan, mulai dari nama, alamat, hingga titik koordinat lokasi yang digunakan sebagai basis validasi *geofencing* untuk memastikan keakuratan lokasi absensi siswa secara *real-time*.



**Gambar 42** Halaman Daftar Perusahaan

## Tampilan Hasil Daftar Pengguna

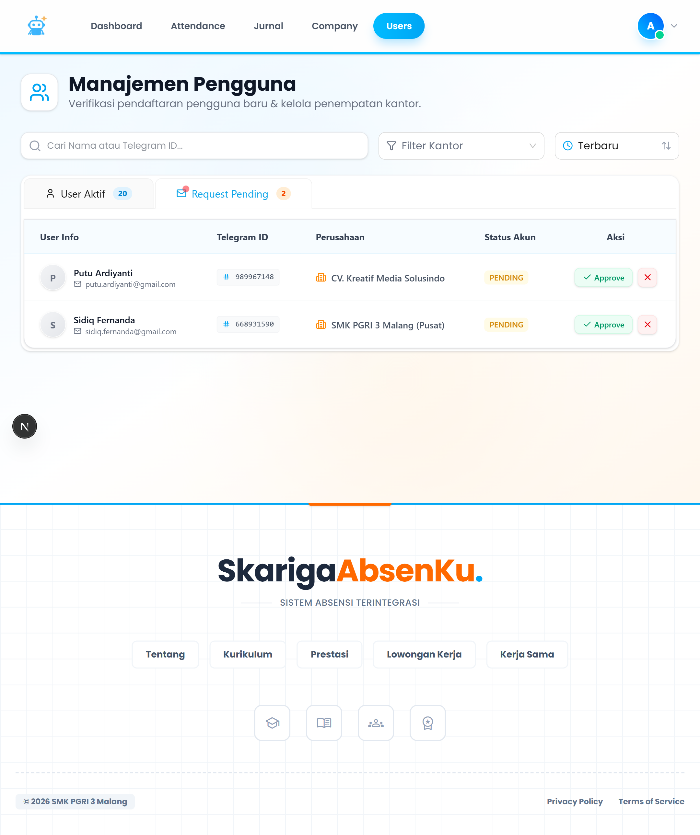
Halaman ini berfungsi sebagai pusat pengelolaan akun untuk seluruh entitas sistem, mulai dari administrator, pembimbing, hingga siswa. Admin dapat memantau profil pengguna, serta memverifikasi status akun guna memastikan distribusi hak akses berjalan sesuai dengan tanggung jawab masing-masing dalam program magang.



**Gambar 43** Tampilan Daftar Pengguna

## Tampilan Daftar Pengguna Tertunda

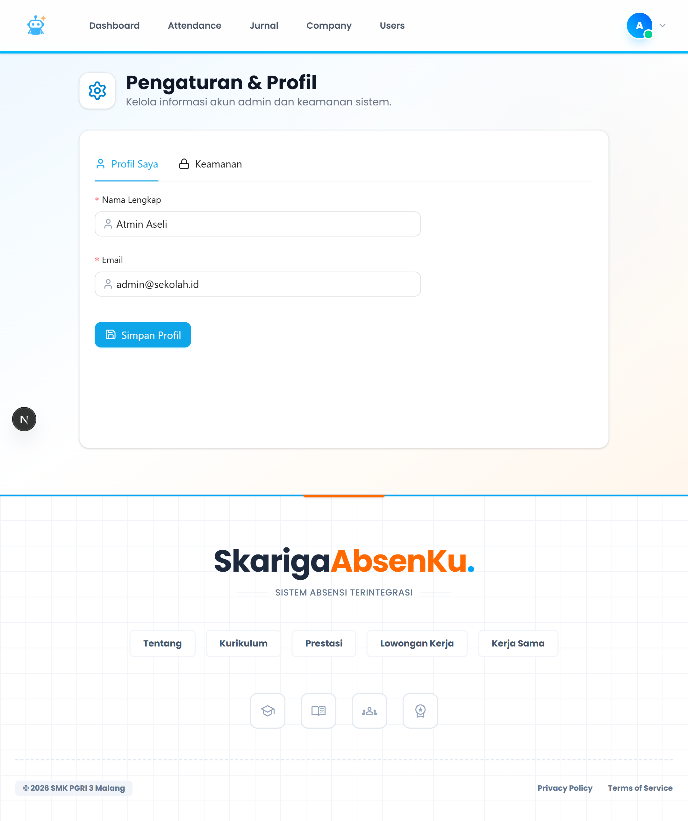
Halaman ini merupakan area validasi bagi admin untuk meninjau permohonan akun baru sebelum diberikan akses ke sistem. Admin dapat memeriksa data pendaftar dan memberikan keputusan berupa persetujuan atau penolakan akun guna memastikan bahwa hanya pengguna resmi yang dapat menggunakan fasilitas aplikasi.



**Gambar 44** Tampilan Pengguna User Tertunda

## Tampilan Halaman Mengatur Profil

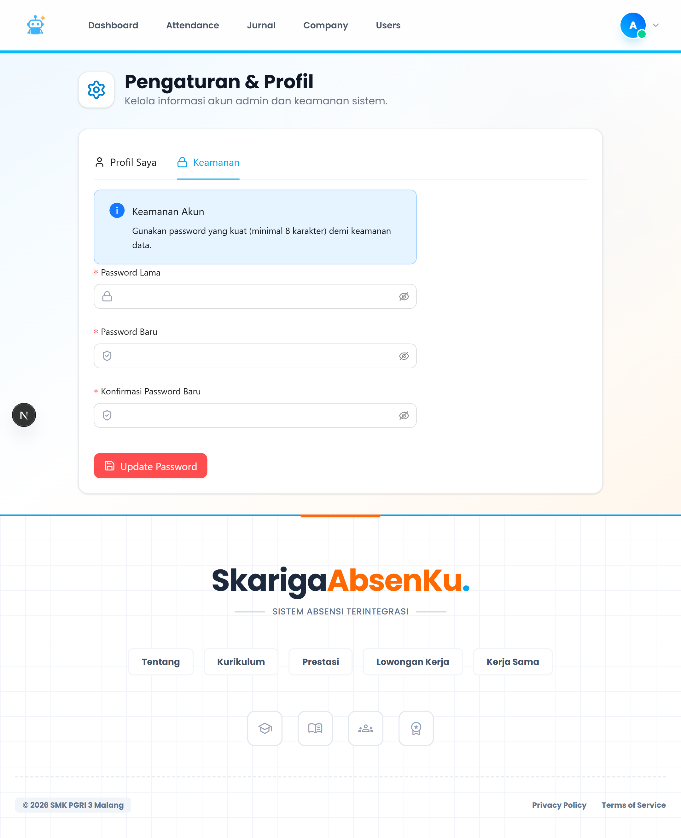
Halaman ini menyediakan fasilitas bagi pengguna untuk melakukan pembaruan data pribadi secara mandiri. Melalui antarmuka ini, pengguna dapat mengelola informasi dasar seperti foto profil, detail kontak, serta melakukan pengaturan keamanan akun melalui pembaruan kata sandi guna memastikan integritas dan akurasi identitas di dalam sistem.



**Gambar 45**  Halaman History Pembelian

## Tampilan Halaman Mengatur Kata Sandi

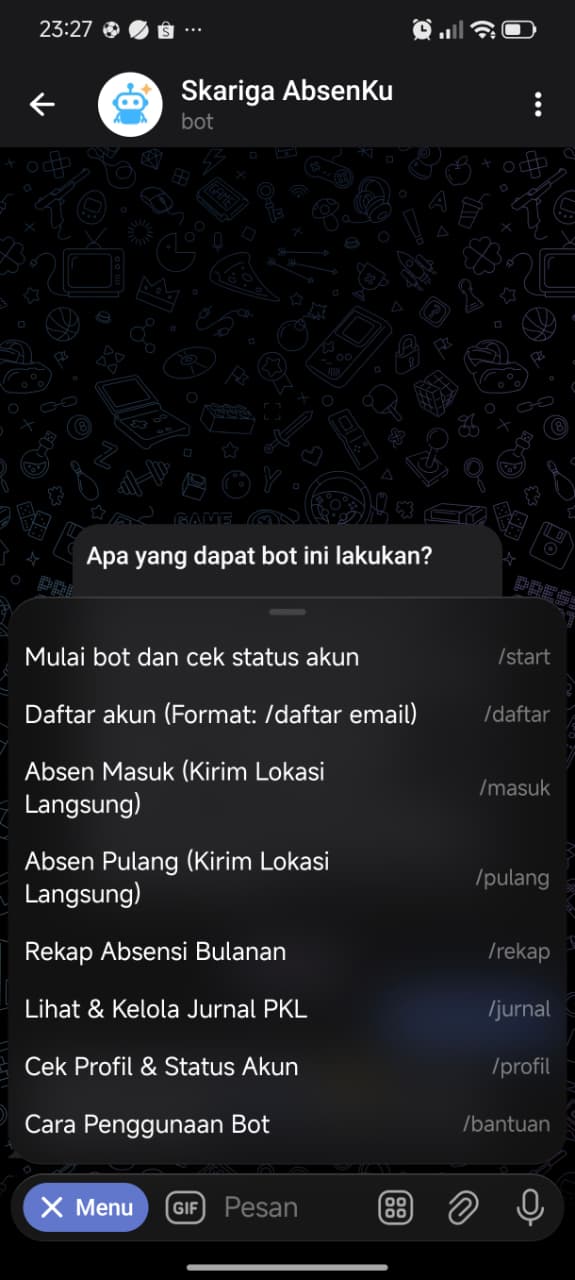
Halaman ini merupakan fitur keamanan yang memungkinkan pengguna untuk melakukan pembaruan kata sandi secara berkala. Melalui antarmuka ini, pengguna diwajibkan memvalidasi kredensial lama sebelum menetapkan kata sandi baru guna memastikan kerahasiaan akun dan mencegah upaya akses tidak sah oleh pihak lain di dalam sistem.



**Gambar 46** Halaman Admin

## Tampilan Menu Pengguna

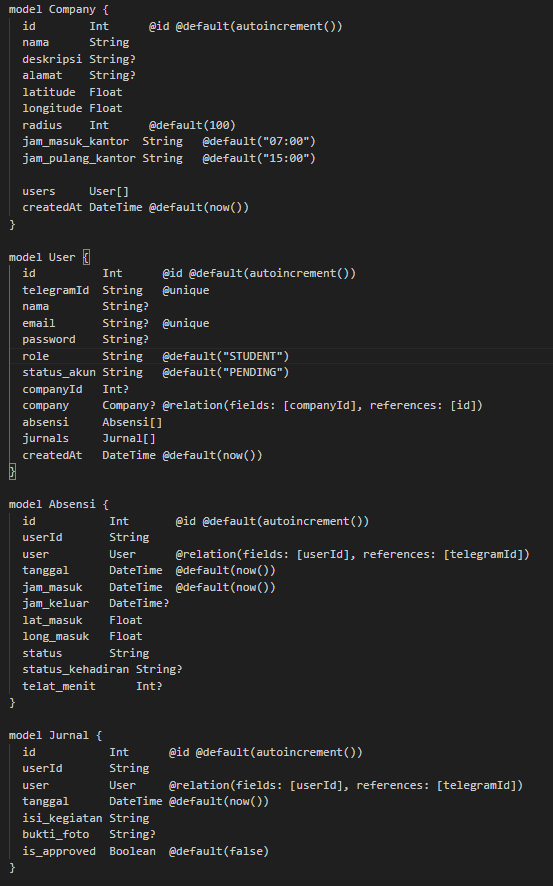
Halaman ini merupakan antarmuka interaktif berbasis pesan instan yang berfungsi sebagai pusat kendali utama bagi siswa magang. Melalui menu ini, pengguna dapat mengakses fitur absensi (masuk/pulang) dan pengisian jurnal harian secara real-time, yang dirancang untuk memberikan kemudahan akses pelaporan langsung dari perangkat seluler tanpa perlu membuka aplikasi tambahan.



**Gambar 47** Halaman Menu Pengguna

## Implementasi Database (Prisma Schema)

Bagian ini menjelaskan struktur data yang digunakan untuk menyimpan seluruh informasi dalam sistem Skariga Absenku. Database dirancang menggunakan Prisma ORM dengan engine SQLite untuk memastikan integritas data dan kemudahan dalam melakukan relasi antar tabel.



**Gambar 48** Implementasi Database

# KESIMPULAN DAN SARAN

## 5.1 Kesimpulan

Sistem asisten digital Skariga Absenku ini dirancang untuk memberikan solusi efisien dalam manajemen kehadiran dan pelaporan kegiatan harian bagi siswa yang sedang melaksanakan praktik kerja lapangan secara digital. Administrator memegang peranan sentral dalam ekosistem ini, di mana mereka memiliki otoritas penuh untuk memantau aktivitas seluruh siswa, memvalidasi laporan jurnal kegiatan, serta mengelola data koordinat perusahaan mitra guna memastikan keakuratan lokasi presensi. Dengan adanya pusat kendali ini, operasional pemantauan dapat dilakukan secara terpusat, transparan, dan dapat dipertanggungjawabkan tanpa perlu melakukan pengecekan manual yang memakan waktu.

Bagi pengguna, khususnya siswa, platform berbasis Telegram ini menyajikan pengalaman pelaporan yang sangat praktis dan responsif melalui fitur perintah interaktif. Fitur utama seperti absensi berbasis lokasi (*geofencing*) dan pengisian jurnal harian memungkinkan data terkirim secara *real-time*, sehingga meminimalkan risiko manipulasi data kehadiran. Integrasi yang kuat antara bot Telegram dengan basis data sistem memastikan bahwa setiap informasi yang dikirimkan oleh siswa dapat langsung diakses oleh guru pembimbing atau admin, memberikan kemudahan akses informasi dalam satu alur komunikasi yang ringkas.

Secara keseluruhan, sistem ini dirancang dengan fokus utama pada akuntabilitas, kemudahan akses, dan kecepatan distribusi data bagi semua pihak yang terlibat. Pemanfaatan teknologi bot ini menjadi langkah inovatif dalam mentransformasi proses administrasi magang konvensional menjadi sistem monitoring yang modern, sekaligus mendukung kedisiplinan siswa melalui instrumen pemantauan yang akurat dan mudah digunakan di perangkat seluler masing-masing.

## 5.2 Saran

Dalam proses pengembangan Website dan Bot Skariga Absenku ini, penulis menyadari bahwa masih terdapat berbagai keterbatasan yang perlu ditingkatkan melalui pengembangan dan penyempurnaan lebih lanjut di masa mendatang. Adapun saran yang dapat dipertimbangkan agar sistem ini dapat berfungsi dengan lebih optimal, menarik, dan memiliki tingkat keamanan yang lebih tinggi adalah sebagai berikut.

Penulis menyarankan adanya penambahan fitur notifikasi pengingat otomatis (*auto-reminder*) yang dikirimkan langsung oleh bot kepada siswa yang belum melakukan absensi masuk atau pengisian jurnal pada jam yang telah ditentukan. Selain itu, integrasi fitur pelaporan dalam bentuk dokumen PDF yang dihasilkan secara otomatis akan sangat membantu admin dalam menyusun rekapitulasi bulanan tanpa harus mengolah data secara manual. Penulis juga menyarankan agar aspek antarmuka (*User Interface*) pada dasbor admin diperbarui agar lebih modern dan interaktif, serta dilakukan optimalisasi pada algoritma pembacaan lokasi guna meningkatkan akurasi presensi pada area yang memiliki kendala sinyal *GPS*. Dengan penyempurnaan pada sisi visual dan teknis tersebut, diharapkan sistem ini dapat memberikan kenyamanan dan fungsionalitas yang lebih luas bagi seluruh penggunanya.

# DAFTAR PUSTAKA

**Amazon Web Services. 2024.** *Apa itu JavaScript?*

<https://aws.amazon.com/id/what-is/javascript>.

(Diakses 29 Oktober 2024).

**Biznet Gio. 2026.** Apa ituNode.js*?*

<https://www.biznetgio.com/news/apa-itu-nodejs>. (Diakses 11 Februari 2026).

**Coding Studio Indonesia. 2024.** *Data Flow Diagram (DFD)*

<https://codingstudio.id/blog/data-flow-diagram-adalah/>. (Diakses 30 Oktober 2024).

**Dicoding Blog. 2024.** Memahami *ERD*

<https://www.dicoding.com/blog/memahami-erd/>. (Diakses 30 Oktober 2024).

**Dicoding Blog. 2026.***Library JavaScript* untuk Membuat *UI.*

<https://www.dicoding.com/blog/apa-itu-react/>.

(Diakses 11 Februari 2026).

**Gramedia. 2024.***Jenis*, Fungsi, dan Simbol-Simbolnya.

<https://www.gramedia.com/literasi/flowchart/>.

(Diakses 30 Oktober 2024).

**Hostinger Tutorial.** **2026.** Mengenal Kelebihan dan Cara Kerjanya.

<https://www.hostinger.co.id/tutorial/apa-itu-typescript>. (Diakses 11 Februari 2026).

**Merdeka.com. 2024.** Mengenal *Website*

<https://www.merdeka.com/jabar/mengenal-website-pengertian-jenis-dan-cara-melindunginya-dari-gangguan-15391-mvk.html>.

(Diakses 29 Oktober 2024).

**Merdeka.com. 2024.** Pengertian *HTML* Lengkap

<https://www.merdeka.com/trending/pengertian-html-lengkap-dengan-fungsi-dan-sejarah-kemunculannya-kln.html>.

(Diakses 29 Oktober 2024).

**Next.js Documentation. 2026.** *Introduction to Next.js*

<https://nextjs.org/docs>.

(Diakses 11 Februari 2026).

**Prisma Documentation. 2026.** *What is Prisma?*

<https://www.prisma.io/docs/concepts/overview/what-is-prisma>.

(Diakses 11 Februari 2026).

**Revou.co. 2024.** *Kosakata Visual Studio Code.*

<https://revou.co/kosakata/visual-studio-code>.

(Diakses 29 Oktober 2024).

**Telegram. 2026.** *Introduction to Bots: Telegram Bot Features.*

<https://core.telegram.org/bots/features>.(Diakses 11 Februari 2026).

**Telkom University IT. 2024.** *Cascading Style Sheet (CSS)*

<https://it.telkomuniversity.ac.id/pengertian-dan-fungsi-cascading-style-sheet-css-pada-website>.

(Diakses 29 Oktober 2024).

# BIODATA PENULIS

Penulis dilahirkan di Kota Blitar, 20 Juni 2008, Penulis yang biasa dipanggil dengan nama “Gilang” ini memiliki nama lengkap Gilang Ardhi Maulana, dan merupakan anak pertama dari 3 bersaudara. Penulis telah menempuh pendidikan formal yaitu di TK Lab UM, SDN Bareng 3 Malang, dan SMP Shalahuddin Malang. Setelah lulus dari SMP Shalahuddin Malang tahun 2026, Penulis meneruskan pendidikan di SMK PGRI 3 Malang di Bidang Keahlian Teknik Informasi dan Komunikasi, Kompetensi Keahlian Rekayasa Perangkat Lunak pada tahun 2024 dan terdaftar dengan Nomor Induk 24008/2057.063 Disaat SMK Penulis mengikuti kegiatan Ekstrakurikuler Esport. Berikut ini adalah email Penulis yang bisa dihubungi, gilang.ardhi26@smk.belajar.id dan media sosial penulis yang masih aktif, Instagram @gln.hooks .