



**SUDAH DIPERIKSA
UNIT PERPUSTAKAAN**

MENGETAHUI KEPALA PERPUSTAKAAN
DIAN NUGROHO PRASETYO, SIP, M.IP
NITK. 7700010699



Plagiarism Checker X Originality Report

Similarity Found: 10%

Date: Wednesday, June 19, 2024

Statistics: 1230 words Plagiarized / 12307 Total words

Remarks: Low Plagiarism Detected - Your Document needs Optional Improvement.

RIZKA AMELIA 203200117 BAB I

PENDAHULUAN Latar Belakang Masalah Menurut Peraturan Menteri Ketenagakerjaan No. 6 Tahun 2020 tentang Pelaksanaan Pemagang di Dalam Negeri, pada Bagian I pasal 1 Ayat (1), dijelaskan bahwa pemagang merupakan bagian integral dari sistem pelatihan kerja [1]. Program magang dalam bidang Pendidikan diharapkan dapat memperkuat hubungan antara perguruan tinggi dan lembaga Pendidikan.

Sistem pelaksanaan program magang saat ini masih dinilai belum efektif karena kurangnya koordinasi di antara pihak-pihak yang terlibat, kurang terstruktur, dan kurang selektif, sehingga perlu dilakukan penyempurnaan [2]. Menurut Chen magang merupakan bagian dari kurikulum, praktikum professional yang dilaksanakan dengan mengatur agar mahasiswa mendapatkan pengalaman nyata melalui bekerja di industri. [3]Program Kuliah Kerja Lapangan (KKL) atau magang menyediakan kesempatan berharga untuk memperoleh pengalaman praktis, berkolaborasi dengan tim, memperbaiki portofolio, dan memperluas jaringan professional di dunia industri.

Kebanyakan sektor industri cenderung memprioritaskan perekrutan individu yang telah memiliki pengalaman, sehingga melalui program magang, pada siswa dapat mendekatkan diri untuk mencapai cita-cita karir mereka [4]. Universitas Alma Ata Yogyakarta adalah salah satu perguruan tinggi swasta yang ada di Kota Yogyakarta yang memiliki beberapa Fakultas, salah satunya yaitu Fakultas Komputer dan Teknik. Di Fakultas Komputer dan Teknik memiliki 2 program studi S1 yaitu Sistem Informasi (SI) dan Informatika (INF).

Pada program studi informatika mahasiswa harus mengikuti program kerja kuliah lapangan, yang bertujuan agar mahasiswa dapat menerapkan teori yang telah dipelajari

dalam praktik di lapangan sehingga dapat menambah dan pengalaman mahasiswa dalam pengelolaan TI. Pada Program Studi Informatika menyediakan beberapa opsi untuk Magang, yaitu ada Magang BUMN yang diselenggarakan oleh Badan Usaha Milik Negara dan sudah bekerjasama dengan Universitas, ada Magang Bersertifikat (MBKM) yang diselenggarakan oleh Kementrian Pendidikan dan Kebudayaan dengan penempatan pada mitra-mitra yang sudah ditetapkan, kemudian ada Magang LIPI-BRIN yang diselenggarakan oleh Lembaga Ilmu Pengetahuan Indonesia dengan penempatan divisi sesuai dengan kompetensi yang diambil.

Kemudian, yang opsi terakhir yang ditawarkan oleh Program Studi, ada Magang mandiri yang dilaksanakan pada lokasi Magang yang diusulkan oleh mahasiswa secara mandiri, dan dilakukan validasi oleh Program Studi. Pada pilihan opsi ini mahasiswa wajib menyerahkan data lokasi Magang yang diusulkan ketika proses Kartu Rencana Studi (KRS) dengan menghubungi Sekretaris program Studi [5]. Berdasarkan tabel di atas, dapat disimpulkan bahwa, pada tahun 2018 dan 2019 tidak ada mahasiswa yang mengikuti program magang MSIB dan pada tahun 2018 ada 14 mahasiswa yang mengikuti magang mandiri, pada tahun 2019 ada 43 mahasiswa yang mengikuti program magang mandiri. Sedangkan pada tahun 2020 ada 15 mahasiswa yang mengikuti program magang MSIB dan ada 54 mahasiswa mengikuti magang mandiri, yang lokasi magangnya berbeda-beda.

Persyaratan untuk bisa mengikuti program magang tersebut diantaranya ialah mahasiswa yang telah menyelesaikan sedikitnya 120 SKS, mahasiswa yang tidak memiliki nilai D yang dibuktikan dengan KHS, mahasiswa yang memiliki Nilai Indeks Prestasi Kumulatif minimal = 3,00. Dan program magang ini pelaksanaannya dilakukan minimal 4 bulan dan maksimal 6 bulan yang dimana memiliki bobot 3 sks atau yang setara dengan 160 (seratus enam puluh) menit perminggu per semester dan ditawarkan pada semester 7 (tujuh) Pada saat ini, di Universitas Alma Ata khususnya pada Program Studi Informatika ini belum memiliki sistem presensi untuk mahasiswa magang.

Sistem presensi yang digunakan masih dilakukan secara semi-manual, itu mengakibatkan proses monitoring dan mengolah data mahasiswa magang tidak berjalan dengan baik. Jadi sampai saat ini, mahasiswa program studi informatika yang sedang mengikuti praktik magang atau Magang (KKL) melakukan absensi yang hanya mengandalkan absensi dari tempat praktik magangnya tersebut. Yang dimana nantinya diakhir periode program magang, pihak Program Studi khususnya Dosen Pembimbing Lapangan akan mengambil rekapan dari logbook mingguan yang dikumpulkan mahasiswa melalui google form tersebut.

Penelitian yang relevan sebelumnya juga dilakukan oleh Ni Kadek Ita Restiani (2023)

dengan judul "Sistem Informasi Absensi Pegawai Magang Berbasis Web Pada Diskominfo Badung" yang menyatakan bahwa sistem informasi absensi berbasis website pada Diskominfo Badung menggunakan framework Codeigniter, sistem yang bertujuan untuk dapat memudahkan pengguna dalam melakukan absensi dan pengisian jurnal kegiatan. Sistem informasi yang dikembangkan dengan metode pengembangan waterfall dan dikembangkan menggunakan bahasa pemrograman PHP, dan MariaDB sebagai DBMS.

[6] Selain itu, penelitian juga pernah dilakukan oleh Indra Kristian Nygraha, Oku Irnawati (2022) dengan judul "Penerapan metode Waterfall Dalam Perancangan Sistem Absensi Online Untuk Pegawai Yayasan SD Islam Assa'addah" menyatakan bahwa Pembangunan sistem absensi berbasis online menggunakan metode waterfall, pada tahap desain perancangan database menggunakan ERD dan UML. Dan untuk pengkodean yang digunakan untuk membangun sistem absensi online berbasis website ialah MySQL dengan bahasa pemrograman PHP. Dan untuk pengujian menggunakan blackbox testing.

[7] Adapun penelitian lainnya juga sempat dilakukan oleh Muhamamd Ridwan Arid (2021) dengan judul "Aplikasi Absensi Mahasiswa/Pelajar Magang Berbasis Website (Studi Kasus : Dinas Samsat Banjar Baru)" menyatakan bahwa aplikasi presensi mahasiswa magang berbasis website ini dibentuk menggunakan bahasa pemrograman berbasis website HTML, PHP, dan CSS serta menggunakan web server berupa apache. Metode yang digunakan dalam melakukan pengujian blackbox testing yang bertujuan untuk memvalidasi inputan yang diolah suatu unit dan memfokuskan pada fungsionalisme dari perangkat lunak.

[8] Berdasarkan permasalahan dan referensi yang sudah diuraikan diatas, perlu adanya pemanfaatan teknologi sebuah sistem presensi yang digunakan sebagai wadah untuk mengelola absensi yang dianggap lebih praktis dan lebih mudah untuk merekap data mahasiswa. Sehubungan dengan adanya permasalahan tersebut maka penulis mengangkat judul dari tugas akhir yaitu dengan judul "SISTEM PRESENSI MAHASISWA MAGANG PRODI INFORMATIKA UAA BERBASIS WEBSITE MENGGUNAKAN METODE WATERFALL". Sistem ini diharapkan dapat membantu mengatasi permasalahan yang ada sehingga sistem presensi dan pengolahan data mahasiswa magang ini lebih terstruktur dan efisien.

Rumusan Masalah Berdasarkan permasalahan yang telah dijabarkan diatas, maka dapat dirumuskan permasalahan yang ada, yaitu Bagaimana Membangun Aplikasi Presensi Mahasiswa Magang di Prodi Informartika Universitas Alma Ata Yogyakarta menggunakan metode waterfall. Batasan Masalah Batasan masalah pada penelitian ini,

ialah untuk mahasiswa magang regular adalah sebagai berikut : Fitur yang diterapkan : Mahasiswa dapat mengisi logbook harian yang dijadikan sebagai acuan untuk absensi mahasiswa Mahasiswa dapat mengupload laporan akhir magang Sistem yang dibuat digunakan oleh : Admin yang berperan sebagai penanggung jawab. Mahasiswa. Dosen. Mitra.

Tujuan Penelitian Tujuan dari penelitian ini adalah untuk merancang bangun aplikasi sistem presensi mahasiswa magang prodi informatika UAA berbasis website menggunakan metode waterfall. Signifikansi Penelitian Bagi Mahasiswa Untuk mengaplikasikan ilmu yang didapatkan selama menempuh Pendidikan di Universitas Alma Ata Yogyakarta Mahasiswa dapat memperoleh pengetahuan bagaimana cara merancang dan membangun aplikasi presensi berbasis website. Bagi Universitas Universitas menerima kontribusi tenaga kerja dari mahasiswa, sehingga memungkinkan permasalahan yang ada di Universitas bisa terselesaikan. Mengetahui kemampuan mahasiswa dalam menerapkan ilmunya dan sebagai bahan evaluasi.

Sebagai referensi mahasiswa yang akan melakukan penelitian. BAB II TINJAUAN PUSTAKA Penelitian Terkait Berikut ini peneliti terdahulu yang akan digunakan sebagai acuan pada penelitian kali ini. Tabel 2. 1 Peneliti Terkait No _Judul Penelitian _Nama Peneliti _Tahun _Deskripsi _1. _Sistem Informasi Absensi Pegawai Magang Berbasis Web Pada Diskominfo Badung _Ni Kadek Ita Restiani _2023 _Sistem informasi absensi berbasis website pada Diskominfo Badung menggunakan framework Codeigniter.

Sistem yang bertujuan untuk dapat memudahkan pengguna dalam melakukan absensi dan pengisian jurnal kegiatan. Sistem informasi ini dikembangkan dengan metode pengembangan waterfall dan dikembangkan menggunakan bahasa pemrograman PHP (Hypertext Preprocessor) dengan Framework Codeigniter dan MariaDB sebagai DBMS. _2. _Rancang Bangun Aplikasi Presensi Mahasiswa Magang Berbasis Website Menggunakan QR Code _Marwanto Rahmatuloh, Adhisa Puspita Rizani, Widia Resdiana _2023 _Sistem yang dibangun menggunakan framework codeigniter versi 4 bahasa pemrograman PHP.

Dengan databasenya MySQL Sistem berhasil dibangun dapat digunakan oleh 3 user yaitu administrator, mahasiswa magang, dan penanggung jawab guna mempermudah proses presensi. _3. _Sistem informasi Presensi Mahasiswa Magang Berbasis Website Pada PT. Kimia Farma TBK _Gilang Akbar Panggulu, Ermatita, Ruth Mariana Bunga Wadu _2022 _Pada PT Kimia Farma TBK masih menggunakan sistem manual dalam proses pengelolaan presensi magang pada PT Kimia Farma TBK agar tercapainya efektivitas dan efisien dalam proses pengelolaan presensi. Penelitian ini menggunakan metode agile

dengan Model Extreme Programming (XP), serta dibantu dengan pemodelan Unified Modelling Language (UML). _ _4.

_Penerapan Metode Waterfall dalam Perancangan Sistem Absensi Online Untuk Pegawai Yayasan SD Islam Assa'addah. _Indra Kristian Nugraha, Oku Irnawati _2022

_Pembangunan sistem absensi berbasis online menggunakan metode waterfall, pada tahap desain perancangan database menggunakan ERD, UML. Dan untuk pengkodean yang digunakan untuk membangun sistem absensi online berbasis website ialah MySql dengan bahasa pemrograman PHP. Dan untuk pengujian menggunakan blackbox testing. _ _5.

_Pengembangan Sistem Presensi Untuk Work From Home (WFH) dan Work From Office (WFO) Selama Pandemi Covid-19 _Fajar Pratama Purwantoro Putra _2022 _Sistem ini berbentuk sebuah aplikasi berbasis website dan aplikasi mobile. Pengembangan sistem ini dibantu oleh beragam package dan fitur-fitur pada Laravel sehingga sistem dapat dikembangkan sesuai requirement dan ekspektasi dari project manager. Sebuah rest API juga dikembangkan agar sistem ini dapat digunakan pada aplikasi mobile oleh karyawan yang bekerja dari kantor dengan cara memindai QR Code yang dibuat administrator. _ _6.

_Perancangan Aplikasi Presensi Online dan Monev Tugas Mahasiswa Magang Diskominfo Provinsi Kalimantan Selatan Berbasis Website _Akbar, Muhammad Rapi'i _2024 _Metode pengembangan dalam membangun aplikasi presensi online mahasiswa ini didasarkan pada metode waterfall. metode pengembangan perangkat lunak yang mengusulkan pendekatan kepada perangkat lunak sistematis dan sekuensial yang mulai pada Tingkat kemajuan sistem pada seluruh analisis, design, kode, pengujian dan pemeliharaan. _ _7. _Aplikasi Absensi Mahasiswa/Pelajar Magang Berbasis Website (Studi Kasus di Dinas Samsat Banjar Baru) _Muhammad Ridwan Arif _2021 _Aplikasi presensi mahasiswa magang berbasis web ini dibentuk menggunakan bahasa pemrograman berbasis website HTML, PHP dan CSS serta menggunakan web server berupa Apache. Metode yang digunakan dalam melakukan pengujian adalah pengujian blackbox testing yang bertujuan untuk memvalidasi inputan yang diolah suatu unit dan memfokuskan pada fungsionalisme dari perangkat lunak.

_ _8. _Perancangan Sistem Presensi Online Peserta Magang Berbasis Website Pada Unit Sistem Informasi PT. Kereta Api Indonesia Divre III Palembang _Dhimas Rosanto Saputra _2020 _Sistem presensi online berbasis website untuk peserta magang di PT. KAI DIVRE III dirancang dan dibuat oleh penulis menggunakan bahasa pemrograman PHP. Dimana website ini bekerja untuk mengatur sistem presensi, mengatur pembagian shift, serta pengajuan izin berhalangan hadir bagi para peserta magang di PT. KAI DIVRE III

Palembang. _ _9.

_Perancangan sistem absensi berbasis web pada program studi PTI UNIMUDA Sorong
_Rut Chrystin Saragi Napitu, Indri Anugrah Ramadhani, Firman Firman _2020 _Jenis penelitian ini menggunakan pendekatan penelitian dan pengembangan (Research and Development) dan menggunakan model perancangan waterfall. Pada penelitian ini menggunakan 5 teknik Analisa data yaitu, mengembangkan instrumen, menentukan sampel penelitian, melakukan pengumpulan data, melakukan analisa data, dan interpretasi data. _ _10. _Perancangan Absensi Siswa Berbasis Website Berbasis PHP MySql di SMA Negeri 1 Palupuh _Gustiar, Supratman Zakir, Wedra Aprison, Zulfani Sesmiarni _2022 _Tujuan penelitian ini bagaimana merancang sebuah sistem absensi siswa berbasis website online untuk mengatasi persoalan absensi dan pendataan absensi secara digital. Metode yang digunakan adalah R & D versi ADDIE. Serta uji produk yaitu uji validitas, uji praktikalitas, dan uji efektifitas.

_ _ Secara keseluruhan, dapat disimpulkan bahwa peneliti-peneliti sebelumnya merancang aplikasi yang menunjukkan fokus dan fitur-fitur yang berbeda, sesuai dengan tujuannya masing-masing. Berdasarkan tinjauan pustaka dari sepuluh referensi penelitian di atas, perbedaan utama penelitian ini dengan penelitian-penelitian sebelumnya terletak pada fokus studi kasus dan sistem presensinya. Peneliti sebelumnya cenderung berfokus hanya pada fitur presensi dan rekapan datanya mahasiswa, yang dimana mahasiswa hanya dapat melakukan presensi.

Sebaliknya, pada penelitian ini, dosen dapat melihat rekapan data mahasiswa dan memberikan nilai akhir untuk program magangnya, sedangkan, mitra dapat menyetujui logbook harian mahasiswa dan juga memberikan nilai akhir. Selain itu, penelitian ini memiliki fokus khusus pada sistem presensi mahasiswa mahasiswa magang pada prodi informatika di Universitas Alma Ata, dengan penggunaanya yaitu mahasiswa, dosen, admin, dan mitra. Landasan Teori Program Magang Program magang adaah strategi pengembangan melalui pelatihan yang dilaksanakan di instansi industri/institusi yang relevan dalam rangka meningkatkan kompetensi professional guru.

Program magang ini terutama diperuntukkan bagi guru kejuruan dan dapat dilakukan pada periode tertentu [9]. Sistem Presensi Sistem manajemen kehadiran, yang dikenal sebagai sistem presensi, digunakan oleh lembaga atau instansi untuk secara otomatis mereka informasi kehadiran kemudian dijadikan laporan manajemen terkait kehadiran mahasiswa/pelajar [10]. Kehadiran sistem presensi sangatlah krusial dalam mengelola kehadiran mahasiswa/pelajar di suatu perusahaan. Perkembangan teknologi, seperti internet, komputer, dan smartphone, telah mengakselerasi perkembangan sistem presensi pada saat ini.

Awalnya, sistem presensi menggunakan metode konvensional seperti kertas, barcode, sidik jari, serta teknologi biometrik lainnya. Sehingga saat ini terciptanya sistem presensi yang dapat diakses melalui aplikasi web atau smartphone, memfasilitasi penggunaan sistem ini baik di kantor maupun di rumah [7]. Presensi ialah **suatu sistem formal dan terstruktur yang** menilai serta memengaruhi aspek aspek yang terkait dengan pekerjaan, perilaku, hasil kerja, termasuk absensi.

Tujuannya adalah untuk mengukur produktivitas seorang mahasiswa/pelajar dan apakah ia mampu hadir dengan efektif di masa depan, memberikan manfaat bagi mahasiswa/pelajar itu sendiri, organisasi, dan masyarakat [11]. Website merupakan **Kumpulan halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman.**

Website juga dapat dikatakan sebagai **Kumpulan halaman dalam suatu domain yang memuat tentang berbagai informasi agar dapat dibaca dan dilihat oleh pengguna internet melalui sebuah mesin pencari.** Informasi yang dapat dimuat dalam sebuah website umumnya berisi mengenai konten gambar, ilustrasi, video, dan teks untuk berbagai macam kepentingan. Biasanya untuk tampilan awal sebuah website dapat diakses melalui halaman utama (homepage) menggunakan browser dengan menuliskan URL yang tepat.

Di **dalam sebuah homepage, juga memuat beberapa halaman web turunan yang saling terhubung satu dengan yang lain.** [12] PHP (PHP Hypertext Preprocessor) PHP (Hypertext Preprocessor) adalah bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam HTML. **PHP banyak dipakai untuk memprogram situs web dinamis. PHP dapat digunakan untuk membangun sebuah CMS. Pada awalnya PHP merupakan kependekan dari Personal Home Page (Situs Personal).** PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. **Pada waktu itu PHP masih bernama Form Interpreted (FI), yang wujudnya berupa sekumpulan skrip yang digunakan untuk mengolah data** formular dari web.

[3] Metode Waterfall **Metode waterfall adalah pendekatan klasik dalam pengembangan perangkat lunak yang menggambarkan metode pengembangan linier dan berurutan. Ini terdiri dari lima hingga tujuh fase, setiap fase didefinisikan oleh tugas dan tujuan yang berbeda, di mana keseluruhan fase menggambarkan siklus hidup perangkat lunak hingga pengirimnya. Tahapan dari metode waterfall dapat dilihat pada gambar berikut** [13]: Sumber : Buku Rekayasa Perangkat Lunak Requirement, pada fase ini.

Mengumpulkan kebutuhan secara lengkap kemudian dianalisis dan didefinisikan kebutuhan yang harus dipenuhi oleh program yang akan dibangun. Fase ini harus dikerjakan secara lengkap untuk bisa menghasilkan desain yang lengkap. Design, dalam tahap ini pengembang akan menghasilkan sebuah sistem secara keseluruhan dan menentukan alur perangkat lunak hingga algoritma yang detail. Development, pada fase ini dimana seluruh desain diubah menjadi kode kode program. Kode program yang dihasilkan masih berupa modul-modul yang akan diintegrasikan menjadi sistem yang lengkap.

Testing, di tahap ini dilakukan penggabungan modul-modul yang sudah dibuat dan dilakukan pengujian ini dilakukan untuk mengetahui apakah software yang dibuat telah sesuai dengan desainnya dan fungsi pada software terdapat kesalahan atau tidak. Maintenance, yaitu instalasi dan proses perbaikan sistem sesuai yang disetujui. MySQL MySQL merupakan server yang melayani database. Untuk membuat dan mengolah database, kita dapat mempelajari pemrograman khusus yang disebut query (perintah) SQL. Database sendiri dibutuhkan jika kita ingin menginput data dari user menggunakan form HTML untuk kemudian diolah PHP agar bisa disimpan ke dalam database MySQL.

[14] Laravel Laravel adalah sebuah framework PHP yang dirilis di bawah lisensi MIT dan dibangun dengan konsep MVC (Model View Controller). Laravel adalah pengembangan website berbasis MVP yang ditulis dalam PHP yang dirancang untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan awal dan biaya pemeliharaan, serta untuk meningkatkan pengalaman bekerja dengan aplikasi dengan menyediakan sintaks yang ekspresif, jelas, dan menghemat waktu. Sebagai sebuah framework PHP, Laravel hadir sebagai platform web development yang bersifat open source.

Yang menarik dari Laravel adalah sintaks ekspresif dan elegan, serta dirancang khusus untuk memudahkan dan mempercepat proses web development. [15] UML (Unified Modeling Language) Perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek, yaitu Unified Modeling Language (UML). UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun dan dokumentasi dari sistem perangkat lunak.

UML merupakan bahasa visual untuk memodelkan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. UML hanya berfungsi untuk melakukan pemodelan. [17] Use case Diagram Menunjukkan sebuah interaksi antara satu atau lebih actor dengan sistem informasi yang akan dibuat. Diagram ini

sangat penting dalam mengatur dan memodelkan perilaku suatu sistem. [18] Tabel 2. 2 Use Case Diagram Simbol _keterangan _ _ Actor orang proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari actor adalah gambar orang, biasanya dinyatakan menggunakan kata benda di awal frase nama actor.

_ _ _ Usecase

fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau actor biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama use case. _ _ _ Asosiasi / Association Komunikasi antar actor dan use case yang berpartisipasi pada use case atau use case memiliki interaksi dengan actor. _ _ _ Ekstensi / Extend Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan dapat berdiri sendiri walaupun tanpa use case tambahan memiliki nama depan yang sama dengan use case yang ditambahkan.

_ _ _ Generalisasi / Generalization Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya. _ _ _ Menggunakan / Include Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan memerlukan use case ini untuk menjalankan fungsional atau sebagai syarat dijalankan use case ini. _ _ _ Activity Diagram Diagram yang menunjukkan arus dari aktivitas ke aktivitas dalam suatu sistem. Activity Diagram membahas pandangan dinamis suatu sistem, dan sangat penting dalam pemodelan fungsi suatu sistem dan menekankan aliran control antar objek.

Tabel 2. 3 Activity Diagram Simbol _Nama _Keterangan _ _ _ Status Awal / Initial _Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal. _ _ _ Aktivitas / Activity _Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja. _ _ _ Percabangan / Decision _Asosiasi percabangan dimana lebih dari satu aktivitas digabungkan menjadi satu. _ _ _ Penggabungan / Join _Asosiasi penggabungan dimana lebih dari satu aktivitas. _ _ _ Status Akhir / Final _Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status satu.

_ _ _ Swimlane _Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi _ _ _ Class Diagram Menunjukkan seperangkat kelas, antarmuka, dan kolaborasi dan hubungan di antara mereka. Class diagram membahas desain statis dari suatu sistem. Tabel 2. 4 Class Diagram Gambar _Nama _Keterangan _ _ _ Generalization _Hubungan dimana objek anak (descendent) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (ancestor). _ _ _ Nary Association _Upaya untuk menghindari asosiasi dengan lebih dari 2 objek. _ _ _ Class _Himpunan dari objek-objek

yang berbagi atribut serta operasi yang sama.

__ _Collaboration _Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukut bagi suatu actor. __ _Realization _Operasi yang benar-benar dilakukan oleh suatu objek. __ _Dependency _Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (independent) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri. __ _Association _Apa yang menghubungkan antara objek satu dengan objek lainnya __ _Visual Studio Code Visual Studio Code merupakan code editor gratis yang dapat digunakan di Operating System Windows, Linux dan MacOS.

Visual Studio code dapat digunakan banyak Bahasa pemrograman seperti PHP, Java, Javascript dan lain-lain. Pengembang dari tools IDE (integrated Development Environment) Visual Studio Code yaitu Microsoft. [19] Blackbox Testing Blackbox testing atau sering dikenal dengan sebutan pengujian spesifikasi fungsional merupakan metode pengujian untuk mengetahui apakah fungsi-fungsi masukan dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. Dalam pengujian ini, tester menyadari apa yang harus dilakukan oleh program tetapi tidak memiliki pengetahuan tentang bagaimana melakukannya.

[6] Kerangka Berpikir Dalam melakukan penelitian, peneliti menggambarkan kerangka berpikir pelaksanaan penelitian dengan bagan sebagai berikut : Problem (Masalah) Prodi Informatika UAA memiliki masalah dalam mengelola data mahasiswa magang. Untuk mengatasi masalah tersebut, pada penelitian ini fokus pada pengembangan aplikasi website sistem absensi mahasiswa magang pada prodi tersebut. Approach (Solusi dan Teori) Pendekatan yang digunakan adalah mengembangkan sebuah sistem presensi dengan memanfaatkan teknologi berbasis website.

Dengan demikian, solusi yang diajukan adalah rancang bangun sistem presensi mahasiswa magang yang mudah diakses, dan memudahkan penanggung jawab dari mahasiswa magang tersebut. Software Developer (Pengembangan Perangkat Lunak) Dalam mengembangkan perangkat lunak untuk Prodi Informatika UAA, pendekatan dimulai dengan desain sistem menggunakan UML, yang terdiri dari use case diagram, class diagram, dan activity diagram. Menggunakan bahasa pemrograman PHP, dengan integrasi database MySql, serta memanfaatkan kerangka kerja Laravel 9. Software Implementation Perangkat lunak yang dikembangkan akan di implementasikan di Prodi Informatika UAA.

Dengan demikian, dapat mengatasi permasalahan yang ada di prodi tersebut. Testing Dari hasil implementasi, untuk proses testingnya mencakup blackbox testing. Hasil

(Result) Hasil dari implementasi aplikasi sistem presensi mahasiswa magang Prodi Informatika UAA mencakup website yang dapat digunakan untuk pengisian logbook, mengupload laporan akhir magang, dan melihat nilai akhir magang. Aplikasi ini dapat membantu instansi untuk memantau data mahasiswa dengan lebih efisien. BAB III METODOLOGI PENELITIAN Jenis dan Desain Penelitian Pada penelitian ini, penulis menggunakan metode pendekatan penelitian Research and Development (R & D).

P. Metode R & D ini dipilih karena memungkinkan pengembangan produk baru dan pengujian keefektifan produk tersebut secara sistematis. R & D sangat cocok untuk penelitian ini karena berfokus pada penciptaan dan pengembangan sistem presensi mahasiswa magang berbasis website yang inovatif dan sesuai dengan kebutuhan pengguna. Metode Pengumpulan Data Adapun Teknik pengumpulan data yang digunakan penulis pada penelitian ini yaitu, sebagai berikut : Studi literatur Studi literatur dilakukan untuk memahami konteks masalah dan menemukan dasar teori yang relevan.

Sumber-sumber seperti buku, jurnal, dan artikel ilmiah digunakan untuk mengumpulkan informasi yang mendukung penelitian. Wawancara Data dikumpulkan melalui wawancara dengan pihak-pihak tertentu yaitu dosen dan admin (LNO) magang, guna untuk memberikan pemahaman mendalam mengenai kebutuhan pengguna terhadap sistem presensi. Observasi Dalam konteks ini, salah satu metode pengumpulan data yang digunakan dalam penelitian ini, yaitu penulis melakukan observasi langsung ke lokasi penelitian prodi informatika UAA untuk mengidentifikasi masalah dan kebutuhan yang ada dalam proses presensi mahasiswa magang. Lokasi Penelitian Penelitian ini dilakukan di Prodi Informatika Universitas Alma Ata Yogyakarta yang beralamat di Jl. Brawijaya No.99, Jadan, Tamantirto, Kec. Kasihan, Kabupaten Bantul, Daerah Istimewa Yogyakarta 55184.

Metode Pengembangan Sistem Metode pengembangan sistem dalam perancangan sistem presensi mahasiswa magang ini menggunakan metode waterfall dimana model ini menyediakan siklus alur terurut mulai dari analisis kebutuhan sistem perangkat lunak, desain, pengkodean dan sampai pengujian. Analisa Sistem Berdasarkan dengan pengumpulan data yang telah dilakukan melalui observasi dan wawancara menghasilkan permasalahan yang terdapat pada sistem monitoring presensi mahasiswa magang sehingga dapat dianalisis bahwa perlu dibangun sebuah sistem presensi yang berbasis website. Perancangan Sistem Pada tahap desain ini, akan diuraikan perancangan arsitektur perangkat lunak menggunakan Unified Modeling Language (UML).

Perancangan ini mencakup penggunaan Use Case Diagram untuk mengilustrasikan

prosedur yang ada dalam sistem yang diajukan dan melibatkan pengguna yang terkait, Activity Diagram untuk menunjukkan aliran aktivitas dari sistem yang diusulkan, dan Class Diagram untuk menggambarkan struktur statis sistem, termasuk kelas-kelas dengan atribut, metode, dan hubungan diantara kelas-kelas tersebut. Pembangunan Sistem Tahap pengkodean ini digunakan untuk membangun sistem. Dalam tahap ini, penulis menggunakan Laravel sebagai framework pengembangan, dengan bahasa pemrograman PHP, Visual Studio Code sebagai teks editor dan MySql sebagai database.

Kode program dikembangkan sesuai dengan rancangan sistem yang telah dibuat pada tahapan sebelumnya. Uji Coba Pada tahap pengujian ini menggunakan blackbox testing untuk melakukan evaluasi terhadap fungsionalitas sistem yang diajukan. Perancangan Sistem Flowchart Flowchart atau diagram alir adalah representasi grafis dari langkah-langkah, urutan, dan keputusan dalam suatu proses sistem. Penggambaran ini memanfaatkan simbol-simbol tertentu untuk menunjukkan aktivitas, kondisi, dan alur logika dari suatu proses. Pada gambar 3.1.

flowchart mahasiswa, proses dimulai dengan mahasiswa mengakses sistem, yang kemudian menampilkan halaman login di mana mahasiswa diminta untuk memasukkan email dan password mereka. Setelah masuk, sistem melakukan verifikasi login. Jika login tidak berhasil, maka mahasiswa diminta memasukkan ulang email dan password yang sesuai, dan jika berhasil, mahasiswa diarahkan ke halaman beranda yang menawarkan beberapa opsi. Pertama, halaman dashboard yang dimana menampilkan informasi-informasi seputar update harian logbook, dan juga mahasiswa dapat mengedit profil mereka guna untuk mengubah informasi jika diperlukan.

Kedua, mahasiswa dapat mengakses logbook harian untuk mencatat aktivitas mereka selama magang berlangsung. Ketiga, mahasiswa dapat mengakses fitur laporan akhir yang hanya bisa diupload sekali di periode akhir magang. Keempat, mahasiswa dapat mengunduh nilai akhir yang diberikan oleh dosen dan perusahaan magang. Terakhir, ada opsi untuk kembali ke halaman beranda jika mahasiswa tidak ingin menggunakan fitur-fitur tersebut. Proses ini berlanjut hingga mahasiswa selesai menggunakan fitur atau memilih untuk kembali ke halaman beranda, dan kemudian proses berakhir. Pada gambar 3.2

flowchart dosen, proses dimulai dengan dosen mengakses sistem, yang kemudian menampilkan halaman login dimana dosen diminta untuk memasukkan email dan password mereka. Setelah masuk, sistem melakukan verifikasi login untuk memastikan keabsahan kredensial. Jika login berhasil, dosen diarahkan ke halaman beranda yang menawarkan beberapa opsi. Pertama, dosen dapat memilih untuk melihat rekapan logbook harian yang diisi oleh mahasiswa setiap hari selama periode magang. Kedua,

terdapat fitur untuk melihat dan mengunduh laporan akhir mahasiswa yang telah diunggah ke sistem. Ketiga, dosen memiliki opsi untuk memberikan nilai pada mahasiswa magang berdasarkan kriteria yang sudah ditentukan sebelumnya.

Setelah selesai menggunakan fitur-fitur tersebut, dosen dapat kembali ke halaman beranda jika tidak ingin memilih opsi lain. **Proses ini berlanjut hingga** dosen selesai menggunakan fitur atau memilih untuk kembali ke halaman beranda, dan kemudian proses berakhir. Pada gambar 3.3. diatas, proses dimulai dari halaman login, dimana **pengguna diminta untuk memasukkan email dan password** mereka. Setelah itu, sistem akan melakukan verifikasi untuk memastikan apakah berhasil atau tidak.

Jika **login berhasil, pengguna akan diarahkan ke halaman** beranda. Di halaman beranda, terdapat beberapa fitur yang bisa dipilih. Fitur pertama adalah "Membuat akun" yang berguna bagi admin untuk membuat akun bagi semua pengguna, baik dosen maupun mahasiswa. Menu berikutnya adalah fitur "Kriteria Penilaian" yang berguna untuk dosen dan mitra memberikan nilai magang kepada mahasiswa berdasarkan kriteria yang telah di input oleh admin. Setelah itu, proses dianggap selesai. Pada gambar 3.4. flowchart mitra, proses dimulai dari halaman login, dimana **pengguna diminta untuk memasukkan** email dan password. Sistem kemudian melakukan verifikasi untuk memastikan apakah login berhasil dan tidak.

Jika **login berhasil, pengguna akan diarahkan ke halaman** beranda. Di halaman beranda, pengguna dapat memilih beberapa menu. Menu pertama adalah fitur Logbook yang memungkinkan pihak mitra untuk menyetujui logbook harian dari mahasiswa. Kemudian ada fitur Jobdesc yang berguna bagi pihak mitra untuk mengunggah file yang berisi deskripsi pekerjaan yang akan dilakukan oleh mahasiswa selama periode magang, serta melihat daftar mahasiswa yang magang. Terakhir, ada fitur Mengolah nilai akhir yang memungkinkan pihak mitra untuk memberikan penilaian akhir kepada mahasiswa magang. Setelah itu proses dianggap selesai.

Use case Diagram Diagram use case digunakan untuk memvisualisasikan interaksi antara pengguna dengan sistem. Pada fase implementasi ini, digram use case akan mengidentifikasi scenario pengguna utama dan fitur-fitur yang ada dalam aplikasi presensi. Pada gambar 3.5. di atas, sistem ini memungkinkan mahasiswa melakukan berbagai aktivitas penting. Mereka dapat mengunduh nilai akhir, mengisi logbook harian, mengedit profil, dan mengunggah laporan akhir. Sementara itu, Dosen dapat melihat rekapan laporan akhir, rekapan logbook, dan memberikan penilaian. Untuk Admin memiliki peran penting dalam mengelola akun pengguna karena memiliki otoritas membuat akun untuk semua pengguna, termasuk mahasiswa dan dosen.

Admin juga memiliki otoritas untuk mengatur dosen pembimbing lapangan. Dan yang terakhir ada Mitra dapat menyetujui logbook harian mahasiswa, mengunggah deskripsi pekerjaan (Jobdesc) yang akan dikerjakan oleh mahasiswa selama magang, dan juga memberikan penilaian akhir terhadap kinerja mahasiswa. Class Diagram Diagram kelas akan mengilustrasikan struktur objek dan keterkaitan di antara mereka dalam aplikasi. Ini membantu dalam memahami entitas-entitas kunci dalam kerangka sistem yang telah dijalankan.

Diagram kelas diatas menunjukkan entitas-entitas dalam sistem magang mahasiswa, termasuk atribut-atribut utama dan operasi-operasi dasar seperti 'create()', 'update()', dan 'delete()'. Entitas-entitas ini saling terhubung dengan hubungan yang jelas, menggambarkan bagaimana data dalam sistem saling terkait. Kelas Roles : Nama _Tipe _ _Id _int _ _User _varchar _ _ Roles ini bisa melakukan operasi 'create()' dan 'update()'. Dan memiliki relasi setiap role bisa dimiliki oleh banyak user.

Kelas Users : Nama _Tipe _ _id _int _ _name _varchar _ _email _varchar _ _email_verified_at _datetime _ _password _varchar _ _remember_token _varchar _ _role_id _int _ _ Users ini bisa melakukan operasi 'create()' dan 'update()'. Dan memiliki relasi yang dimana setiap user memiliki satu role (1:1), dihubungkan oleh role_id. Kelas Mahasiswa : Nama _Tipe _ _id _int _ _user_id _int _ _dosen_id _int _ _tanggal_lahir _date _ _magang_batch _varchar _ _perusahaan_id _int _ _gambar _varchar _ _nama_supervisor _varchar _ _no_hp_supervisor _varchar _ _ Kelas mahasiswa ini bisa melakukan operasi 'create()', 'update()', 'delete()'.

Dan memiliki relasi setiap mahasiswa memiliki satu user (1:1) dihubungkan oleh 'user_id', setiap mahasiswa memiliki satu dosen (1:1) yang dihubungkan oleh 'dosen_id', dan setiap mahasiswa terhubung dengan satu perusahaan (1:1) yang dihubungkan oleh 'perusahaan_id'. Kelas Perusahaan : Nama _Tipe _ _id _int _ _user_id _int _ _nama_perusahaan _varchar _ _alamat _varchar _ _ Kelas perusahaan bisa melakukan operasi 'create()' dan 'update()'. Dan memiliki relasi setiap perusahaan memiliki satu user (1:1) yang dihubungkan oleh 'user_id'. Kelas Dosen : Nama _Tipe _ _id _int _ _user_id _int _ _nidn _varchar _ _gambar _varchar _ _ Kelas dosen ini bisa melakukan operasi 'create()' dan 'update()'.

Dan memiliki relasi setiap dosen memiliki satu user (1:1) yang dihubungkan oleh 'user_id'. Kelas Logbook : Nama _Tipe _ _id _int _ _mhs_id _int _ _deskripsi _varchar _ _lampiran _varchar _ _Status _varchar _ _ Kelas logbook ini bisa melakukan 'create()' dan 'update()'. Dan memiliki relasi setiap logbook terkait dengan satu mahasiswa (M:1) yang dihubungkan oleh 'mhs_id'. Kelas Jobdesc : Nama _Tipe _ _id _int _ _mhs_id _int _ _file_job _varchar _ _ Kelas Jobdesc ini bisa melakukan 'create()' dan 'update()'.

Dan memiliki relasi setiap jobdesc dengan satu mahasiswa (M:1) yang dihubungkan oleh 'mhs_id'. Kelas Laporan Akhir : Nama _Tipe _ _id _int _ _mhs_id _int _ _laporan_akhir _varchar _ _ Kelas laporan akhir ini bisa melakukan 'create()' dan 'update()'. Dan memiliki relasi setiap laporan dengan satu mahasiswa (M:1) yang dihubungkan oleh 'mhs_id'. Kelas Penilaian : Nama _Tipe _ _id _int _ _mhs_id _int _ _kriteria_penilaian _int _ _nilai _float _ _ Kelas laporan akhir ini bisa melakukan 'create()' dan 'update()'.

Dan memiliki relasi setiap penilaian terkait dengan satu mahasiswa (M:1) yang dihubungkan oleh 'mhs_id' dan setiap penilaian terkait dengan satu kriteria penilaian (M:1) yang dihubungkan oleh 'kriteria_penilaian'. Kelas Kriteria Penilaian Nama _Tipe _ _id _int _ _mhs_id _int _ _kriteria_penilaian _int _ _nilai _float _ _ Kelas kriteria penilaian ini bisa melakukan 'create()' dan 'update()'. Entity Relationship Diagram (ERD) ERD adalah salah satu jenis diagram yang digunakan dalam perancangan basis data untuk menggambarkan hubungan antara entitas atau objek dalam suatu sistem.

Entity utama dalam diagram ini adalah mahasiswa, dosen, perusahaan, dan mitra. Mahasiswa memiliki atribut seperti id, tanggal_lahir, magang_batch, perusahaan_id, dosen_id, gambar, nama_supervisor, dan no_hp_supervisor. Setiap mahasiswa dapat melakukan beberapa jobdesc, logbook, dan laporan_akhir, yang masing-masing memiliki atribut terkait. Jobdesc memiliki atribut id, mhs_id, dan file_job, sedangkan logbooks memiliki atribut id, mhs_id, deskripsi, lampiran, dan status. Laporan_akhir memiliki atribut id, mhs_id, dan laporan_akhir. Dosen memiliki atribut id, user_id, nidn, dan gambar. Setiap dosen dapat melakukan penilaian terhadap mahasiswa. penilaian memiliki atribut seperti tanggal, nilai_akhir, kriteria_id, dan dosen_id.

Setiap penilaian terhubung dengan kriteria yang memiliki atribut_id, nama_kriteria, bobot, dan jenis. Entitas user menyimpan informasi pengguna dengan atribut id, email, email_verified_at, password, remember_token, name, dan role_id. Setiap user bisa memiliki satu peran (role) yang ditentukan oleh entitas roles yang memiliki atribut id dan name. Perusahaan memiliki atribut id, user_id, nama_perusahaan, dan alamat. Perusahaan terkait dengan mahasiswa melalui atribut perusahaan_id pada entitas mahasiswa.

Kardinalitas yang ditampilkan dalam diagram menunjukkan hubungan antar entitas sebagai berikut : Setiap mahasiswa bisa memiliki satu dosen (1:1) Setiap dosen bisa memiliki beberapa penilaian (1:M) Setiap penilaian terkait dengan satu kriteria (M:1) Setiap mahasiswa dapat melakukan beberapa jobdesc, logbook, dan laporan_akhir (1:M) Setiap mahasiswa bisa memiliki satu role dan satu perusahaan (1:1) Setiap user bisa menjadi dosen atau perusahaan (1:1) Activity Diagram Diagram ini bermanfaat untuk

memvisualisasikan urutan langkah-langkah yang diambil oleh sistem dan pengguna dalam menjalankan fungsi tertentu. Diagram Activity Login Diagram activity login pada gambar 3.8, diatas dapat dijelaskan sebagai berikut : Terdapat 4 pengguna dalam sistem yang diusulkan, yaitu admin, mahasiswa, dosen, dan mitra Pengguna memulai dengan melakukan login melalui formular login, dimana actor menginputkan username dan password yang kemudian diverifikasi di database. Jika proses login tidak berhasil, pengguna akan diarahkan kembali ke formular login. Namun, jika proses login berhasil, mereka akan diarahkan ke halaman menu. Diagram Activity Mahasiswa dan Dosen / Gambar 3.

9 Activity Diagram Mahasiswa dan Dosen Pada gambar 3.9 Aktivitas Diagram mahasiswa dan dosen diatas, proses dimulai Ketika mahasiswa dan dosen melakukan login ke dalam sistem. Setelah login berhasil, sistem menampilkan menu dashboard yang tersedia untuk kedua pengguna. Mahasiswa kemudian memilih menu yang diinginkan, yang terdiri dari tiga opsi : logbook, laporan akhir, dan penilaian akhir. Jika mahasiswa memilih untuk mengisi logbook, sistem akan menerima data logbook yang diinput oleh mahasiswa, dan dosen akan dapat melihat rekapan logbook tersebut.

Apabila mahasiswa memilih untuk mengunggah laporan akhir, sistem akan menyimpan laporan tersebut sehingga dosen dapat melihat laporan tersebut. Pada menu penilaian akhir, dosen memiliki wewenang untuk memberikan nilai berdasarkan evaluasi yang telah dilakukan. Setelah nilai diberikan, Sistem akan menampilkan nilai akhir tersebut kepada mahasiswa. Proses ini memastikan interaksi yang efektif antara mahasiswa dan dosen dalam pengelolaan logbook, laporan akhir, dan penilaian akhir. Diagram Activity Mitra dan Mahasiswa Pada gambar 3.10 Activity Diagram Mitra dan Mahasiswa, proses dimulai Ketika mitra dan mahasiswa melakukan login ke dalam sistem.

Setelah berhasil login, sistem akan menampilkan menu dashboard yang tersedia untuk kedua pengguna. Jika mahasiswa mengisi logbook, sistem akan menyimpan data logbook tersebut dan menampilkannya pada tampilan mitra untuk disetujui. Setelah mitra menyetujui logbook, sistem akan memvalidasi dan menampilkan status bahwa logbook telah disetujui. Kemudian, pada menu jobdesc mitra dapat mengunggah dekripsi pekerjaan (jobdesc) yang akan dikerjakan oleh mahasiswa. Sistem akan menampilkan form jobdesc tersebut sehingga mahasiswa dapat melihat dan memahami tugas-tugas yang akan mereka kerjakan selama periode magang. Selanjutnya, pada menu upload nilai akhir, sistem akan menampilkan form penilaian kepada mitra.

Mitra kemudian mengisi dan menyimpan penilaian tersebut. Setelah penilaian diberikan, mahasiswa dapat melihat dan mengunduh nilai akhir yang telah diberikan oleh mitra. Alur Penelitian Penelitian ini mengikuti alur penelitian yang dimulai dengan identifikasi

masalah dalam pengembangan sistem presensi untuk mahasiswa magang prodi informatika UAA berbasis website. Tahap awal melibatkan studi literatur dari berbagai sumber, seperti jurnal, buku, dan internet untuk memahami secara mendalam konteks dan tantangan yang dihadapi. Pengumpulan data dilakukan melalui wawancara dan observasi memastikan pemahaman yang komprehensif.

Kemudian, setelah itu, mengidentifikasi metode apa yang sesuai untuk memecahkan masalah, membimbing langkah-langkah selanjutnya dalam pembuatan sistem dan pengujian. Hasil akhir dari penelitian ini diharapkan sebuah sistem presensi mahasiswa magang yang berbasis website dapat menciptakan sebuah solusi yang dapat membantu memonitoring data mahasiswa secara optimal. Pada gambar 3.11. Penelitian ini dimulai dengan identifikasi masalah, di mana peneliti menentukan isu atau tantangan utama yang akan dipecahkan. Langkah selanjutnya adalah melakukan studi literatur untuk mendapatkan pemahaman yang lebih mendalam tentang topik tersebut.

Studi literatur ini mencakup tinjauan terhadap jurnal-jurnal ilmiah, skripsi, dan sumber-sumber dari internet yang relevan. Setelah memiliki landasan teori cukup, peneliti melanjutkan dengan pengumpulan data melalui wawancara dan observasi untuk mendapatkan informasi langsung dari sumber yang terkait. Data yang telah dikumpulkan digunakan sebagai dasar dalam tahapan perancangan sistem, yang melibatkan pembuatan diagram UML untuk memodelkan sistem, pengkodean dengan PHP, dengan pengaturan database.

Implementasi dari sistem yang telah dirancang dilakukan dalam bentuk website. Setelah website selesai dibangun, dilakukan pengujian menggunakan metode blackbox testing untuk memastikan semua fitur berjalan sesuai dengan yang diharapkan. Hasil dari pengujian ini dievaluasi untuk mengidentifikasi dan memperbaiki kelemahan atau bug yang ditemukan. Akhirnya, penelitian ini disimpulkan dengan merangkum temuan utama, efektivitas solusi yang diterapkan, dan rekomendasi untuk penelitian atau pengembangan lebih lanjut.

Perancangan Wireframe Wireframe adalah representasi visual dasar antarmuka pengguna (UI) suatu aplikasi atau situs web. Ini merupakan langkah awal dalam proses desain, di mana struktur dan tata letak elemen-elemen penting dalam antarmuka diperjelas tanpa memperhatikan detail desain visual. Gambar 3. 12 Tampilan Wireframe Login Mahasiswa, Dosen dan Admin, Mitra Pada gambar 3.12, tampilan wireframe untuk halaman login bagi Mahasiswa, Dosen, Admin, dan Mitra dirancang dengan kesederhanaan dan kemudahan pengguna, dimana fitur utamanya adalah memasukkan email dan kata sandi.

Setiap halaman login diawali dengan header yang menampilkan nama aplikasi untuk menampilkan identitas visual yang konsisten. Di bagian tengah, terdapat dua field input utama: satu untuk "Email" dan satu lagi untuk "Kata Sandi". Di bawah field input, terdapat tombol "Masuk" yang digunakan untuk masuk ke dalam sistem. Selain itu, ada tombol "Lupa Kata Sandi" yang berfungsi untuk membantu pengguna yang mungkin lupa kata sandi user. / Pada gambar 3.13, tampilan wireframe untuk profil mahasiswa dirancang agar informatif dan mudah digunakan, dengan fitur utama yang mencakup informasi pribadi dan data terkait magang.

Di bagian samping kiri, terdapat area untuk foto profil mahasiswa, memberikan identitas visual yang jelas. Di samping kanan, terdapat beberapa field yang menampilkan informasi penting: "Nama", "Email", "Tanggal Lahir", "Magang Batch", "Nama Perusahaan", "Gambar profil", "Nama Supervisor", dan "Nomor HP Supervisor". Setiap field informasi bisa diedit memungkinkan mahasiswa untuk memperbarui datanya sesuai kebutuhan. Tata letak yang rapi dan terstruktur memudahkan navigasi dan memastikan semua informasi penting mudah diakses dan dilihat. Gambar 3.

14 Tampilan Wireframe Mahasiswa Isi Logbook Pada gambar 3.14, tampilan wireframe untuk halaman logbook mahasiswa dirancang dengan dua panel utama untuk memfasilitasi pencatatan harian dan pengelolaan rekapan logbook. Pada panel sebelah kiri, terdapat form untuk pengisian logbook harian. Form ini mencakup field teks untuk deskripsi kegiatan yang dilakukan, serta fitur unggah file yang mendukung berbagai jenis berkas seperti gambar dan PDF, memungkinkan mahasiswa untuk menyertakan bukti visual atau dokumen terkait.

Panel sebelah kanan, menampilkan form rekapan logbook yang berisi catatan kegiatan dari hari sebelumnya. Setiap entri pada rekapan logbook ini dapat diedit oleh mahasiswa selama belum disetujui oleh mitra. Setelah logbook disetujui oleh mitra, entri tersebut menjadi terkunci dan tidak dapat diubah lagi. Gambar 3. 15 Tampilan Wireframe Mahasiswa Upload Laporan Akhir Pada gambar 3.15, tampilan wireframe untuk halaman upload laporan akhir mahasiswa dirancang dengan struktur yang jelas dan informatif untuk memastikan penggunaan oleh mahasiswa.

Di bagian paling atas halaman, ditampilkan informasi penting yaitu "Nama Mahasiswa" dan "Tempat Magang", memberikan identifikasi yang jelas tentang pemilik laporan dan lokasi magangnya. Di bawah informasi ini, terdapat form unggah laporan akhir yang dilengkapi dengan field untuk memilih dan mengunggah file laporan dalam format yang ditentukan, yaitu PDF. / Gambar 3. 16 Tampilan Wireframe Mahasiswa Penilaian Akhir Pada gambar 3.16. diatas, tampilan wireframe untuk halaman nilai akhir mahasiswa dirancang untuk memberikan informasi yang lengkap dan mudah diakses mengenai

hasil akademik mahasiswa.

Di bagian utama halaman, terdapat tabel atau daftar yang menampilkan secara penuh semua nilai akhir mahasiswa, termasuk detail seperti nilai yang diperoleh. Dibagian atas tabel atau daftar nilai tersebut, terdapat tombol yang jelas dan mudah ditemukan dengan label "Download PDF". / Gambar 3. 17 Tampilan Wireframe Dosen Dashboard Pada gambar 3.17. diatas, menampilkan wireframe untuk dashboard dosen yang bisa melihat mahasiswa tersebut mengikuti magang batch pada periode berapa, bisa melihat siapa saja mahasiswa yang belum mengisi logbook, dan juga bisa melihat total mahasiswa magang dan total mitra secara keseluruhan. / Gambar 3. 18 Tampilan Wireframe Dosen Rekap Logbook Pada gambar 3.18, menampilkan wireframe dosen untuk rekap logbook mahasiswa. Menu ini, dosen bisa melihat rekap logbook mahasiswa yang telah mengisi logbook harian.

/ Gambar 3. 19 Tampilan Wireframe Dosen Detail Rekap Logbook Pada gambar 3.19 diatas, menampilkan tampilan wireframe dosen untuk detail rekap logbook mahasiswa, yang dimana pada halaman ini ada jumlah hadir dan tidak hadir dari mahasiswa yang diambil dari data logbook harian yang sudah di isi sebelumnya oleh mahasiswa. Dan ada juga tampilan rekap logbook yang di blok perminggu. / Gambar 3. 20 Tampilan Wireframe Dosen Laporan Akhir Pada gambar 3.20 diatas, dosen bisa melihat data rekap laporan akhir mahasiswa yang telah mengisi mengupload laporan akhir.

/ Gambar 3. 21 Tampilan Wireframe Nilai Akhir Pada gambar 3.20 diatas, Dosen melihat daftar mahasiswanya kemudian bisa memberikan dan mengolah nilai magang kepada mahasiswanya yang sudah dipilih sebelumnya. / Gambar 3. 22 Tampilan Wireframe Dosen Detail Penilaian Pada gambar 3.22 diatas, menampilkan halaman dosen untuk detail penilaian untuk mahasiswa yang sudah dipilih pada halaman sebelumnya. Pada halaman ini Dosen bisa menginput langsung nilai yang akan diberikan untuk mahasiswanya dan sistem akan otomatis melakukan perhitungan. / Gambar 3. 23 Tampilan Wireframe Dosen Riwayat Pada gambar 3.

23 diatas, menampilkan halaman Dosen untuk menu Riwayat yang berguna untuk Dosen dapat melihat dan memfilter daftar mahasiswanya berdasarkan dengan batch yang dipilih. / Gambar 3. 24 Tampilan Mitra Jobdesc Pada gambar 3.24. diatas, menampilkan mitra dapat melihat daftar nama mahasiswa dan mengupload jobdesc masing masing mahasiswa magang. / Gambar 3. 25 Tampilan Wireframe Mitra Penilaian Akhir Pada gambar 3.25 diatas, Pada mitra dapat melakukan dan menginputkan nilai untuk mahasiswa yang magang di mitra tersebut.

BAB IV

HASIL DAN PEMBAHASAN Analisis Masalah Identifikasi Masalah Dari hasil observasi dan wawancara di Prodi Studi Informatika, identifikasi masalah utama yang dihadapi di Program Studi Informatika yaitu, belum memiliki sistem presensi untuk mahasiswa magang, yang dimana proses absensi masih dilakukan secara semi-manual, itu mengakibatkan proses monitoring dan mengolah data mahasiswa magang tidak berjalan dengan baik. Metode Penyelesaian Masalah Dalam mengatasi tantangan yang dihadapi di Prodi Informatika UAA yaitu dengan metode Waterfall menjadi solusi yang tepat.

Dengan menggunakan metode waterfall tersebut proses pembuatan sistem dilakukan secara terstruktur dan terencana. Analisis kebutuhan Fitur Kebutuhan Perangkat Keras Dalam menganalisis kebutuhan perangkat keras untuk sistem presensi magang mahasiswa berbasis website dengan menggunakan metode waterfall, beberapa aspek teknis harus diperhatikan. Pertama, diperlukan server yang mampu menangani aplikasi web dan database yang baik, menjamin ketersediaan sistem secara terus-menerus dan kinerja yang cepat.

Kedua, dibutuhkan server database untuk menyimpan dan mengelola data mahasiswa dengan aman dan efisien. Selain itu, jaringan yang stabil diperlukan untuk menghubungkan server dengan user, memastikan aksesibilitas sistem dari berbagai Lokasi. Dengan mempertimbangkan kebutuhan perangkat keras ini, pengembangan sistem presensi dapat dilakukan dengan efektif dan memastikan kinerja optimal di Prodi Informatika UAA. Kebutuhan Fitur Dalam menganalisis kebutuhan fitur untuk sistem presensi mahasiswa magang di prodi Informatika, langkah penting yang harus dilakukan adalah wawancara dengan berbagai pihak internal yang terlibat.

Dari wawancara ini, diperoleh wawasan yang berharga mengenai kendala yang dihadapi oleh tim internal, kebutuhan informasi yang mereka miliki, dan proses kerja sistem. Hasil wawancara ini memungkinkan identifikasi fitur-fitur yang dibutuhkan dengan lebih akurat sesuai dengan kebutuhan instansi. Berdasarkan hasil analisis tersebut, beberapa fitur yang diperlukan adalah : Fitur Buat Akun Admin memiliki fitur untuk membuat akun bagi semua pengguna (mahasiswa, dosen, dan mitra) karena sistem ini tidak mendukung register mandiri oleh pengguna.

Logbook Mahasiswa mengisi logbook harian yang berfungsi sebagai absensi dan catatan aktivitas magang. Dosen bisa melihat rekapan logbook tersebut, dan mitra dapat menyetujui ataupun menolak logbook tersebut. Laporan Akhir Mahasiswa mengunggah laporan magang pada akhir periode magang, dan dosen dapat melihat dan mendownload laporan tersebut. Penilaian Akhir Mahasiswa dapat melihat dan mendownload nilai akhir magang yang diberikan oleh dosen dan mitra. Sedangkan

dosen dan mitra dapat mengolah dan menginput nilai akhir magang mahasiswa. Jobdesc Mitra mengunggah jobdesc pekerjaan yang harus dilakukan oleh mahasiswa selama program magang berlangsung. Perancangan Basis Data Struktur Tabel / Gambar 4.

1 Tabel Dosen Pada gambar 4.1, tabel yang ditampilkan merupakan tabel dosen, berikut penjelasan rinci setiap kolom dalam tabel : id : ditunjukkan dengan tipe data 'BIGINT' dengan Panjang 20. Kolom ini diatur auto-increment, yang berarti nilai akan otomatis bertambah unik untuk setiap catatan baru. Kolom ini juga unsigned, yang berarti hanya dapat menyimpan angka positif. user_id : kolom ini juga bertipe 'BIGINT' dengan Panjang 20. Kolom ini juga unsigned dan kemungkinan berfungsi sebagai kunci asing yang menghubungkan ke tabel pengguna, mengidentifikasi pengguna mana yang memiliki catatan tersebut. nidn : kolom dengan tipe 'VARCHAR' dengan Panjang 255.

Kolom ini dimaksudkan untuk menyimpan pengenal unik, mungkin nomor identifikasi akademik. gambar : kolom ini juga bertipe 'VARCHAR' dengan Panjang 255. Kolom ini digunakan untuk menyimpan data gambar atau path/URL ke file gambar. created_at : kolom 'TIMESTAMP' ini mencatat tanggal dan waktu Ketika catatan dibuat. Nilai default kolom ini adalah 'NULL', yang berarti timestamp akan secara otomatis diatur ke waktu saat ini ketika catatan baru dimasukkan. update_at : kolom 'TIMESTAMP' lainnya yang mencatat tanggal dan waktu ketika catatan terakhir diperbaharui. Nilai default kolom ini juga 'NULL', secara otomatis memperbaharui setiap kali catatan diubah. / Gambar 4. 2 Tabel Users Pada gambar 4.2

didasar, menampilkan bagian dari basis data yang berhubungan dengan informasi users. Berikut penjelasan rinci setiap kolom : id : dengan tipe data 'BIGINT' dan Panjang 20. Kolom ini diatur untuk auto-increment, sehingga setiap rekaman baru akan mendapatkan nilai unik secara otomatis. Kolom ini juga unsigned, artinya hanya menyimpan nilai positif. name : kolom ini bertipe 'VARCHAR' dengan Panjang 255. Kolom ini digunakan untuk menyimpan nama pengguna. email : kolom ini bertipe 'VARCHAR' dengan Panjang 255. Kolom ini digunakan untuk menyimpan alamat email pengguna. email_verified_at : kolom dengan tipe "TIMESTAMP".

Kolom ini digunakan untuk menyimpan tanggal dan waktu ketika email pengguna telah diverifikasi. password : kolom ini bertipe 'VARCHAR' dengan Panjang 255. Kolom ini digunakan untuk menyimpan kata sandi pengguna yang telah di-hash untuk keamanan. remember_token : kolom ini bertipe 'VARCHAR' dengan Panjang 100. Kolom ini digunakan untuk menyimpan token yang digunakan untuk mengingat sesi pengguna, biasanya untuk fitur 'ingat saya' pada sistem login. created_at : kolom 'TIMESTAMP' yang mencatat tanggal dan waktu ketika catatan pengguna dibuat. Nilai default kolom

ini adalah 'NULL', yang berarti timestamp akan secara otomatis diatur ke waktu saat ini ketika catatan baru ditambahkan.

updated_at : kolom 'TIMESTAMP' lainnya yang mencatat tanggal dan waktu ketika catatan pengguna terakhir diperbaharui. Nilai default kolom ini juga 'NULL', dan akan diperbaharui setiap kali catatan diubah. role_id : kolom ini bertipe 'BIGINT' dengan panjang 20 dan unsigned. Kolom ini kemungkinan merupakan kunci asing yang menghubungkan ke tabel peran (roles), mengidentifikasi peran atau level akses pengguna. / Gambar 4. 3 Tabel Mahasiswa Pada gambar 4.3 diatas, menampilkan tabel mahasiswa yang berisi bagian skema basis data yang berhubungan dengan data mahasiswa. berikut penjelasan secara rincinnya : id : dengan tipe data 'BIGINT' dan panjang karakter 20.

Kolom ini diatur untuk auto-increment, sehingga setiap rekaman baru akan mendapatkan nilai unik secara otomatis. Kolom ini juga unsigned, artinya hanya menyimpan nilai positif. user_id : kolom ini bertipe 'BIGINT' dengan panjang 20. Kolom ini unsigned dan kemungkinan berfungsi sebagai kunci asing yang menghubungkan ke tabel pengguna, mengidentifikasi pengguna mana yang terkait dengan data mahasiswa ini. dosen_id : kolom ini bertipe 'BIGINT' dengan panjang 20. Kolom ini unsigned dan kemungkinan berfungsi sebagai kunci asing yang menghubungkan ke tabel dosen, mengidentifikasi dosen pembimbing lapangan dari mahasiswa tersebut. tanggal_lahir : kolom ini bertipe 'DATE'.

Kolom ini digunakan untuk menyimpan tanggal lahir mahasiswa. magang_batch : kolom ini bertipe 'VARCHAR' dengan panjang 255. Kolom ini digunakan untuk menyimpan periode batch mahasiswa. perusahaan_id : kolom ini bertipe 'BIGINT' dengan panjang 20. Kolom ini unsigned dan kemungkinan berfungsi sebagai kunci asing yang menghubungkan ke tabel perusahaan, mengidentifikasi perusahaan tempat mahasiswa melakukan magang. gambar : kolom ini bertipe 'VARCHAR' dengan panjang 255. Kolom ini digunakan untuk menyimpan data gambar atau path/URL ke file gambar mahasiswa. nama_supervisor : kolom ini bertipe 'VARCHAR' dengan panjang 255.

Kolom ini digunakan untuk menyimpan nama supervisor atau pembimbing dari perusahaan tempat mahasiswa magang. no_hp_supervisor : kolom ini bertipe 'VARCHAR' dengan panjang 255. Kolom ini digunakan untuk menyimpan nomor telepon supervisor dari perusahaan. created_at : kolom 'TIMESTAMP' yang mencatat tanggal dan waktu ketika catatan mahasiswa dibuat. Nilai default kolom ini adalah 'NULL', yang berarti timestamp akan secara otomatis diatur ke waktu saat ini ketika catatan baru ditambahkan. update_at : kolom 'TIMESTAMP' lainnya yang mencatat tanggal dan waktu ketika catatan mahasiswa terakhir diperbarui. Nilai default kolom ini juga 'NULL', dan

akan diperbarui setiap kali catatan diubah. / Gambar 4. 4 Tabel Logbooks Pada gambar 4.4

dias, menampilkan tabel logbooks yang berisi bagian skema basis data yang berhubungan dengan data logbooks mahasiswa. berikut penjelasan secara rincinya : id : dengan tipe data 'BIGNIT' dan panjang karakter 20. Kolom ini diatur untuk auto-increment, sehingga setiap rekaman baru akan mendapatkan nilai unik secara otomatis. Kolom ini juga unsigned, artinya hanya menyimpan nilai positif. mhs_id : dengan tipe data 'BIGNIT' dan panjang karakter 20. Kolom ini bersifat unsigned yang artinya, hanya bisa menyimpan nilai positif. Kolom ini tidak memiliki nilai default, sehingga harus diisi setiap kali ada data baru.

deksripsi : kolom ini bertipe 'TEXT' yang digunakan untuk menyimpan teks dengan panjang yang tidak ditentukan. Kolom ini memiliki nilai default. lampiran : kolom ini bertipe 'VARCHAR' dengan panjang maksimal 255 karakter. Kolom ini digunakan untuk menyimpan teks dengan panjang terbatas. status : kolom ini bertipe 'VARCHAR' dengan panjang karakter minimal 255. Kolom ini digunakan untuk menyimpan teks dengan panjang terbatas. Kolom ini memiliki nilai default "Menunggu Persetujuan". created_at : kolom ini bertipe data 'TIMESTAMP'. Kolom ini digunakan untuk menyimpan informasi waktu pembuatan data. Kolom ini bisa menerima nilai NULL. update_at : kolom ini bertipe data 'TIMESTAMP'.

Kolom ini digunakan untuk menyimpan informasi waktu pembaruan data. Kolom ini bisa menerima nilai NULL. / Gambar 4. 5 Tabel Laporan Akhir Pada gambar 4.5 diatas, menampilkan tabel laporan akhir yang berisi bagian skema basis data yang berhubungan dengan data laporan akhir. Berikut penjelasan secara rincinya : id : kolom ini bertipe data 'BIGNIT' dengan panjang 20 karakter. Kolom ini bersifat unsigned, yang berarti hanya dapat menyimpan nilai positif. Selain itu, kolom ini memiliki property auto_increment, yang mengindikasikan bahwa nilainya akan otomatis bertambah setiap kali ada data baru yang dimasukkan ke dalam tabel.

Kolom ini berfungsi sebagai primary key untuk mengidentifikasi setiap entri secara unik. mhs_id : kolom ini juga bertipe data 'BIGNIT' dengan panjang 20 karakter dan bersifat unsigned. Kolom ini digunakan untuk menyimpan ID mahasiswa yang terkait dengan laporan akhir. Tidak ada nilai default yang ditetapkan untuk kolom ini, sehingga nilai harus diisi setiap kali ada data baru. laporan_akhir : kolom ini bertipe data 'VARCHAR' dengan panjang maksimal 255 karakter. Kolom ini digunakan untuk menyimpan teks yang berisi laporan akhir. Tidak ada nilai default yang ditetapkan untuk kolom ini. created_at : kolom ini bertipe 'TIMESTAMP'.

Kolom ini digunakan untuk mencatat waktu kapan data laporan akhir tersebut dibuat. Kolom ini dapat menerima nilai NULL, yang berarti kolom ini tidak wajib diisi saat data baru ditambahkan. update_at : kolom ini juga bertipe data 'TIMESTAMP'. Kolom ini digunakan untuk mencatat waktu kapan data laporan akhir tersebut diperbarui. Kolom ini juga dapat menerima nilai NULL, yang berarti kolom ini tidak wajib diisi saat data baru ditambahkan atau diperbarui. / Gambar 4. 6 Tabel Jobdesc Pada gambar 4.6 diatas, menampilkan tabel jobdesc yang berisi bagian skema basis data yang berhubungan dengan mitra.

Berikut penjelasan secara rincinya : id : kolom ini bertipe data 'BIGNIT' dengan panjang 20 karakter dan bersifat unsigned, yang berarti hanya bisa menyimpan nilai positif. Kolom ini memiliki property auto_increment, yang berarti nilainya akan otomatis bertambah setiap kali ada data baru yang dimasukkan ke dalam tabel. Kolom ini berfungsi sebagai primary key untuk mengidentifikasi setiap entri secara unik. mhs_id : kolom ini juga bertipe data 'BIGNIT' dengan panjang 20 karakter dan bersifat unsigned. Kolom ini digunakan untuk menyimpan ID mahasiswa yang terkait dengan deskripsi pekerjaan tersebut.

Kolom ini tidak memiliki nilai default, sehingga harus diisi setiap kali ada data baru. file_job : kolom ini bertipe data 'VARCHAR' dengan panjang maksimal 255 karakter. Kolom ini digunakan untuk menyimpan nama file yang berisi deskripsi pekerjaan mahasiswa. Kolom ini tidak memiliki nilai default. created_at : kolom ini bertipe data 'TIMESTAMP'. Kolom ini digunakan untuk menyimpan informasi waktu pembuatan atau deskripsi pekerjaan. Kolom ini bisa menerima nilai NULL, yang berarti kolom ini tidak wajib diisi saat data baru ditambahkan. updated_at : kolom ini juga bertipe data 'TIMESTAMP'. Kolom ini digunakan untuk menyimpan informasi waktu pembaruan data deskripsi pekerjaan.

Kolom ini bisa menerima NULL, yang berarti kolom ini tidak wajib diisi saat data baru ditambahkan atau diperbarui. / Gambar 4. 7 Tabel Penilaian Pada gambar 4.7 diatas, menampilkan tabel penilaian, yang berfungsi untuk menyimpan data penilaian terhadap mahasiswa. Berikut penjelasannya secara rinci : id : kolom ini bertipe data 'BIGNIT' dengan panjang 20 karakter dan bersifat unsigned, yang berarti hanya bisa menyimpan nilai positif.

Kolom ini memiliki property auto_increment, yang berarti nilainya akan otomatis bertambah setiap kali ada data baru yang dimasukkan ke dalam tabel. Kolom ini berfungsi sebagai primary key untuk mengidentifikasi setiap entri secara unik. mhs_id : kolom ini juga bertipe 'BIGNIT' dengan panjang 20 karakter dan bersifat unsigned. Kolom ini digunakan untuk menyimpan ID mahasiswa yang dinilai. Kolom ini tidak

memiliki nilai default, sehingga harus diisi setiap kali ada data baru. kriteria_penilaian_id : kolom ini bertipe data 'BIGINT' dengan panjang 20 karakter dan bersifat unsigned. Kolom ini digunakan untuk menyimpan ID kriteria penilaian yang digunakan untuk menilai mahasiswa.

Kolom ini tidak memiliki nilai default, sehingga harus diisi setiap kali ada data baru. nilai : kolom ini bertipe data 'DECIMAL' dengan panjang 5 digit dan 2 digit di belakang koma. Kolom ini digunakan untuk menyimpan nilai penilaian yang diberikan kepada mahasiswa. Kolom ini tidak memiliki nilai default, sehingga harus diisi setiap kali ada data baru. created_at : kolom ini bertipe data 'TIMESTAMP' yang digunakan untuk menyimpan informasi waktu pembuatan data penilaian. Kolom ini bisa menerima nilai NULL, yang berarti kolom ini tidak wajib diisi saat data baru ditambahkan.

updated_at : kolom ini juga bertipe data 'TIMESTAMP'. Kolom ini digunakan untuk menyimpan informasi pembaruan data penilaian. Kolom ini bisa menerima nilai NULL, yang berarti kolom ini tidak wajib diisi saat data baru ditambahkan atau diperbarui. / Gambar 4. 8 Tabel Kriteria Penilaian Pada gambar 4.8 diatas, menampilkan tabel kriteria penilaian yang berfungsi untuk menyimpan data kriteria penilaian yang digunakan dalam prose penilaian terhadap mahasiswa. Berikut penjelasannya secara rinci : id : kolom ini bertipe data 'BIGINT' dengan panjang 20 karakter dan bersifat unsigned, yang berarti hanya bisa menyimpan nilai positif.

Kolom ini memiliki peran property auto_increment yang berarti nilainya akan otomatis bertambah setiap kali ada data baru yang dimasukkan ke dalam tabel. Kolom ini fungsinya sebagai primary key untuk mengidentifikasi setiap entri secara unik. nama_kriteria : kolom ini bertipe data 'VARCHAR' dengan panjang maksimal 255 karakter. Kolom ini digunakan untuk menyimpan nama kriteria penilaian. Bobot : kolom ini bertipe data 'DECIMAL' dengan panjang 5 digit dan 2 digit di belakang koma. Kolom ini digunakan untuk menyimpan bobot kriteria penilaian, yang menunjukkan seberapa penting kriteria tersebut dalam penilaian keseluruhan. Jenis : kolom ini bertipe data 'ENUM' yang berisi nilai-nilai tetap yaitu 'internal' dan 'eksternal'.

Kolom ini digunakan untuk menentukan jenis kriteria penilaian, apakah kriteria tersebut bersifat internal dan eksternal. Kolom ini tidak memiliki nilai default, sehingga harus diisi setiap kali ada data baru. created_at : kolom ini bertipe data 'TIMESTAMP'. Kolom ini digunakan untuk menyimpan informasi waktu pembuatan data kriteria penilaian. Kolom ini bisa menerima nilai NULL yang berarti kolom ini tidak wajib diisi saat data baru ditambahkan. updated_at : kolom ini bertipe data 'TIMESTAMP'. Kolom ini digunakan untuk menyimpan informasi waktu pembaruan data kriteria penilaian. Kolom ini bisa menerima nilai NULL yang berarti kolom ini tidak wajib diisi saat data baru ditambahkan.

atau diperbarui. / Gambar 4. 9 Tabel Perusahaan Pada gambar 4.9

dias, menampilkan tabel perusahaan yang merupakan bagian dari basis data dalam sistem yang berfungsi untuk menyimpan informasi mengenai perusahaan yang terdaftar dalam sistem. Berikut penjelasannya secara rinci ; id : yang bertipe data 'BIGINT' dengan panjang 20 karakter, bersifat unsigned dan diatur sebagai auto_increment. Kolom ini berfungsi sebagai primary key untuk tabel 'perusahaan' secara otomatis menambahkan nilai unik setiap kali ada entri baru ditambahkan. User_id : bertipe data 'BIGINT' dengan panjang 20 karakter dan bersifat unsigned.

Kolom ini mengacu pada 'id' pengguna dari tabel 'users' mengindikasikan hubungan antara perusahaan dengan pengguna yang mengelolanya. Kolom ini tidak boleh bernilai null dan bertindak sebagai foreign key dalam konteks database relasional.

nama_perusahaan : bertipe data 'VARCHAR' dengan panjang maksimal 255 karakter, berfungsi untuk menyimpan nama perusahaan. Kolom ini tidak boleh bernilai null.

Alamat : yang memiliki tipe data 'VARCHAR' dengan panjang maksimal 255 karakter, yang berfungsi untuk menyimpan alamat perusahaan dan juga tidak boleh bernilai null.

/ Gambar 4. 10 Tabel Roles Pada gambar 4.10 diatas, menampilkan tabel roles yang berfungsi untuk menyimpan informasi tentang peran pengguna dalam sistem. Berikut penjelasannya secara rinci : id : yang bertipe data 'BIGINT' dengan panjang 20 karakter.

Kolom ini bersifat unsigned, tidak mengizinkan nilai null, dan diatur sebagai auto_increment. Fungsinya adalah sebagai primary key untuk tabel 'roles' yang secara otomatis memberikan nilai unik pada setiap entri baru yang ditambahkan. name : yang bertipe data 'VARCHAR' dengan panjang maksimal 255 karakter. Kolom ini menyimpan nama dari peran pengguna dan tidak boleh bernilai null, tetapi tidak memiliki nilai default. created_at : yang bertipe data 'TIMESTAMP' kolom ini mencatat waktu saat data peran pertama kali dibuat, danizinkan untuk bernilai null. updated_at : bertipe data 'TIMESTAMP' yang mencatat waktu terakhir kali data peran diperbarui.

Kolom ini juga diizinkan untuk bernilai null. Struktur Database Pada gambar 4.11 diatas, menampilkan stuktur database sistem yang ditampilkan terdiri dari beberapa tabel yang saling berhubungan dan digunakan untuk mengelola informasi terkait mahasiswa, dosen, perusahaan, dan kegiatan magang. Berikut penjelasan secara rinci mengenai relasinya : users dan roles : tabel 'users' memiliki kolom 'role_id' yang berhubungan dengan kolom 'id' pada tabel 'roles'.

Ini adalah relasi many-to-one, dimana setiap pengguna memiliki satu peran (role), tetapi satu peran dapat dimiliki oleh banyak pengguna. mahasiswa dan users : tabel 'mahasiswa' memiliki kolom 'user_id' yang berhubungan dengan kolom 'id' pada tabel

'users'. Ini adalah one-to-one, dimana setiap mahasiswa terkait dalam satu pengguna. dosen dan users : tabel 'dosen' memiliki kolom 'user_id' yang berhubungan dengan kolom 'id' pada tabel 'users'. Ini juga merupakan relasi one-to-one, dimana setiap dosen terkait dengan satu pengguna. perusahaan dan users : tabel 'perusahaan' memiliki kolom 'user_id' yang berhubungan dengan kolom 'id' pada tabel 'users'.

Ini adalah relasi one-to-one, dimana setiap perusahaan terkait dengan satu pengguna. mahasiswa dan users : tabel 'mahasiswa' memiliki kolom 'dosen_id' yang berhubungan dengan kolom 'id' pada tabel 'dosen'. Ini adalah relasi many-to-one, dimana setiap mahasiswa memiliki satu dosen pembimbing, tetapi satu dosen dapat membimbing banyak mahasiswa. logbooks dan mahasiswa : tabel 'logbooks' memiliki kolom 'mhs_id' yang berhubungan dengan kolom 'id' pada tabel 'mahasiswa'.

ini adalah relasi many-to-one, dimana setiap logbook terkait dengan satu mahasiswa, tetapi satu mahasiswa dapat memiliki banyak logbook. laporan_akhir dan mahasiswa : tabel 'laporan_akhir' memiliki kolom 'mhs_id' yang berhubungan dengan kolom 'id' pada tabel 'mahasiswa'. ini adalah relasi one-to-one, dimana setiap laporan akhir terkait dengan satu mahasiswa. jobdesc dan mahasiswa : tabel 'jobdesc' memiliki kolom 'mhs_id' yang berhubungan dengan kolom 'id' pada tabel 'mahasiswa'. ini adalah relasi one-to-one, dimana setiap deskripsi pekerjaan terkait dengan satu mahasiswa.

penilaian dan mahasiswa : tabel 'penilaian' memiliki kolom 'mhs_id' yang berhubungan dengan kolom 'id' pada tabel 'mahasiswa'. ini adalah relasi many-to-one, dimana setiap penilaian terkait dengan satu mahasiswa, tetapi satu mahasiswa dapat memiliki banyak penilaian. penilaian dan kriteria_penilaian : tabel 'penilaian' memiliki kolom 'kriteria_penilaian_id' yang berhubungan dengan kolom 'id' pada tabel 'kriteria_penilaian'.

Ini adalah relasi many-to-one, dimana setiap penilaian menggunakan satu kriteria penilaian, tetapi satu kriteria penilaian dapat digunakan dalam banyak penilaian. Perancangan Sistem Pada gambar 4.12 diatas, menampilkan halaman login untuk mahasiswa, dosen, mitra, dan juga admin. yang dimana user harus memasukkan email address dan password mereka untuk mengakses sistem. Dan juga ada tombol "Masuk" yang harus diklik pengguna setelah memasukkan kredensial mereka. Selain itu, dibagian bawah terdapat tautan untuk memulihkan sandi yang terlupa dengan label "Lupa kata sandi". Tampilan Admin / Gambar 4.

13 Tampilan Beranda Admin Pada gambar 4.13 diatas, tampilan beranda admin menampilkan pesan pengingat "Selamat Datang Kembali" / Gambar 4. 14 Tampilan Tambah Akun Pada gambar 4.14 diatas, tampilan tambah akun untuk semua pengguna.

yang dimana pada halaman ini menampilkan rekapan akun yang telah dibuatkan oleh admin yang isinya ada nama, email, beserta rolesnya. Dan juga akun tersebut dapat diedit dan di delete oleh admin. / Gambar 4. 15 Tampilan Tambah Akun Pada gambar 4.15 diatas, halaman ini berguna untuk admin jika ingin membuatkan akun baru kepada pengguna yang rolesnya dapat dipilih sesuai dengan pengguna yang ingin dibuatkan akun. Pada gambar 4.16 diatas, tampilan tambah akun mahasiswa yang di isi kan oleh admin sesuai dengan data mahasiswa, yang dimana nama kolomnya ada nama mahasiswa, email, roles, tanggal lahir, magang batch, nama supervisor, no.

telepon supervisor, perusahaan, input dosen pembimbing lapangan, dan foto profil untuk mahasiswa. / Gambar 4. 17 Tampilan Tambah Akun Mitra Pada gambar 4.17 diatas, tampilan tambah akun mitra yang di isikan oleh admin jika ingin membuatkan akun untuk mitra atau perusahaan tempat mahasiswa tersebut magang. Yang dimana kolomnya ada nama perusahaan, email, roles, dan juga alamat perusahaan tersebut. Pada gambar 4.18 diatas, tampilan tambah akun dosen yang harus di isikan oleh admin jika ingin membuat akun untuk dosen, yang isi kolomnya ada nama dosen, email, roles, NIDN, dan juga foto profil buat akun dosen.

Yang dimana kolom tersebut semuanya wajib di isi. Pada gambar 4.19 diatas, tampilan kriteria penilaian pada halaman ini admin menginput kriteria penilaian untuk di isi oleh mitra dan dosen yang nantinya akan di kirimkan ke masing-masing mahasiswa. Yang dimana kriteria tersebut ada 2 jenis internal dan eksternal, pada gambar diatas dapat dilihat kriteria penilaiannya beserta bobot yang telah di input oleh admin. Pada gambar 4.20 diatas, pada halaman **ini admin dapat menambahkan** kriteria penilaian jika suatu waktu kriteria yang telah di input sebelumnya berubah atau ingin ditambah.

Pada gambar diatas dapat dilihat ada kolom nama kriteria, bobot penilaian, dan juga jenis penilaian yang dimana kolom tersebut tidak boleh kosong. Tampilan Mahasiswa / Gambar 4. 21 Tampilan Beranda Mahasiswa Pada gambar 4.21 diatas, tampilan beranda mahasiswa yang menampilkan informasi seputar "Selamat datang kembali" dan update informasi logbook mahasiswa. Pada gambar 4.22 diatas, tampilan pertama dan yang paling penting yaitu halaman jobdesc mahasiswa untuk jobdesc yang menampilkan sebuah file dalam bentuk PDF yang dapat di download oleh mahasiswa untuk mengetahui jobdesc apa saja yang diberikan oleh perusahaan tempat magang.

Pada gambar 4.23 diatas, tampilan logbook mahasiswa. Pada kolom sebelah kiri, terdapat kotak yang di bagian paling atasnya menampilkan tanggal dan hari untuk mengisi logbook harian. Logbook ini hanya dapat diisi satu kali sehari. Sebelum mengirimkan (submit) logbook, mahasiswa harus melampirkan bukti pengerjaan dalam bentuk PDF, PNG, dan format lainnya. Di kolom sebelah kanan, terdapat rekapan data

logbook mahasiswa yang telah diisi sebelumnya, yang dibagi perminggu. Jika logbook tersebut belum disetujui oleh perusahaan, maka mahasiswa dapat mengedit logbook tersebut. Namun, jika sudah disetujui oleh perusahaan, mahasiswa tidak dapat mengubahnya lagi dan logbook tersebut akan dikirimkan ke rekapan tampilan dosen.

/ Gambar 4. 24 Tampilan Laporan Akhir Mahasiswa Pada gambar 4.24 diatas, tampilan laporan akhir mahasiswa yang menampilkan nama mahasiswa dan nama perusahaan tempat magang dan sebuah form yang berguna untuk mahasiswa meng upload laporan akhir magang. / Gambar 4. 25 Tampilan Nilai Magang Mahasiswa Pada gambar 4.25 diatas, tampilan nilai magang mahasiswa yang dimana pada halaman ini ada penjelasan kriteria, bobot, jenis, beserta nilai magang mahasiswa yang diberikan oleh dosen dan perusahaan. Dan pada halaman ini juga mahasiswa dapat mendownload dalam bentuk PDF nilai yang telah diberikan tersebut. Tampilan Dosen / Gambar 4. 26 Tampilan Beranda Dosen Pada gambar 4.26 diatas, tampilan beranda dosen yang menampilkan informasi-informasi seputar periode magang, total perusahaan, informasi logbook harian mahasiswa, dan juga total mahasiswa magang pada periode yang sedang berlangsung. Pada gambar 4.27, tampilan rekapan logbook mahasiswa ditampilkan dosen dimana halaman ini menampilkan seluruh logbook mahasiswa yang dilengkapi dengan button "Lihat Detail". Pada gambar 4.28 diatas, tampilan detail logbook mahasiswa, lanjutan dari halaman sebelumnya jika di klik button "Lihat Detail" maka akan menampilkan halaman ini, yang dimana pada halaman ini ada kalender guna untuk merekap kehadiran logbook mahasiswa dan disebelah kanannya **rekapan jumlah kehadiran dan** tidak hadir. Dan untuk tampilan bawahnya ada rekapan keterangan isi logbook dari mahasiswa. Pada gambar 4.29 diatas, tampilan rekapan laporan akhir mahasiswa, pada halaman ini menampilkan rekapan nama mahasiswa yan telah mengupload laporan akhirnya dan juga ada tanggal yang sesuai dengan waktu mahasiswa mengupload laporan akhir tersebut.

Dan juga filenya yang bentuk PDF bisa di download. Pada gambar 4.30 diatas, menampilkan Riwayat rekapan mahasiswa, pada halaman ini bisa memfilter mahasiswa magang berdasarkan batch magangnya. dan juga menampilkan nama perusahaan tempat mahasiswa tersebut magang. Pada gambar 4.31 diatas, tampilan semua daftar mahasiswa yang sudah di filter sesuai dosen pembimbing lapagannya, yang nantinya akan diberikan nilai magang oleh masing masing dosennya. Pada gambar 4.32 diatas, tampilan beri nilai yang pada halamannya sebelumnya jika diklik beri nilai akan muncul halaman ini.

Pada halaman ini menampilkan kriteria penilaian yang sebelumnya sudah di input oleh admin dan dosen tinggal mengisi sesuai bobot yang telah ditentukan. Pada gambar 4.33 diatas, tampilan detail penilaian mahasiswa yang telah diberikan oleh mitra dan dosen,

dimana pada halaman ini menampilkan rincian penilaian yang memuat jenis, bobot, dan nilai akhirnya. Tampilan Mitra Pada gambar 4.34 diatas, tampilan beranda mitra menampilkan pesan "Selamat Datang Kembali" dan menampilkan informasi seputar total logbook mahasiswa yang belum dan sudah disetujui oleh mitra. Pada gambar 4.35 diatas, tampilan mitra logbook pada halaman ini menampilkan data logbook semua mahasiswa yang sedang magang diperusahaan tersebut. Pada gambar 4.36 diatas, tampilan mitra detail logbook ini menampilkan halaman yang berguna untuk menyetujui logbook mahasiswa.

Nantinya setelah disetujui akan dikirimkan langsung ke tampilan dosen. Pada gambar 4.37 diatas, tampilan mitra jobdesc menampilkan form untuk mitra mengupload jobdesc pekerjaan untuk mahasiswa. Dan tampilan kolom dibawah menampilkan data rekapan mahasiswa magang beserta jobdesc yang dapat didownload dan dilihat ulang oleh mitra. Pada gambar 4.38 diatas, tampilan mitra penilaian menampilkan rekapan daftar mahasiswa yang akan diberikan penilaian setelah program magang yang dijalankan oleh mahasiswa berakhir. Pada gambar 4.39 diatas, tampilan mitra beri nilai sama halnya dengan tampilan dosen beri nilai, pada halaman ini mitra akan menginput nilai untuk mahasiswa magangnya sesuai dengan kriteria penilain yang telah di inputkan oleh admin.

Implementasi Pada Pengkodean Logbook Mahasiswa Dalam metode waterfall, implementasi sistem presensi mahasiswa magang ini menggunakan framework Laravel merupakan tahap ketiga setelah analisis kebutuhan dan perancangan sistem. Pada tahap ini, tim pengembang memastikan bahwa kode yang dihasilkan mematuhi standar pengkodean Laravel, memperhatikan prinsip-prinsip desain yang baik, dan mengikuti praktik keamanan yang disarankan. Dengan menggunakan Laravel, mereka dapat memanfaatkan fitur-fitur bawaan yang mendukung pengembangan, seperti migrasi database dan sistem routing yang fleksibel, untuk membangun sistem presensi yang stabil, skalabel, dan mudah dikelola.

Model Logbook Pada gambar 4.40 diatas, kode di atas merupakan skrip migrasi Laravel yang digunakan untuk membuat table 'logbooks' di database. Skrip migrasi ini mendefinisikan struktur tabel serta aturan relasionalisnya. Berikut adalah penjelasan lebih rinci : Namespace dan Penggunaan Kelas : Kode ini menggunakan beberapa namespace dari Laravel, yaitu 'Illuminate\Database\Migrations\Migration', 'Illuminate\Database\Schema\Blueprint', dan 'Illuminate\Support\Facades\Schema'. Membuat tabel 'logbooks' : Pada metode 'up()', digunakan fungsi 'Schema::create' untuk membuat tabel 'logbooks' dengan kolom-kolom sebagai berikut : 'id' : Kolom auto increment sebagai primary key.

'mhs_id' : Kolom integer yang tidak bernilai negatif, mengacu pada id mahasiswa.
'deskripsi' : Kolom teks untuk menyimpan deskripsi logbook. 'lampiran' : Kolom string untuk menyimpan informasi lampiran. 'status' : Kolom string dengan nilai default 'Menunggu Persetujuan' 'timestamps()' : Menambahkan dua kolom 'created_at' dan 'updated_at'. Relasi dengan tabel 'mahasiswa' : Kolom 'mhs_id' diatur sebagai foreign key yang mengacu pada kolom 'ide' di tabel 'mahasiswa'. Jika data mahasiswa dihapus, maka semua logbook yang terkait juga akan dihapus ('onDelete('cascade')').

Menghapus tabel 'logbooks' : Pada metode 'down()', digunakan fungsi 'Schema: dropIfExists' untuk menghapus tabel 'logbooks' jika tabel tersebut ada. Controller Logbook Pada gambar 4.41, kode diatas merupakan bagian dari controller dalam aplikasi Laravel bernama 'MahasiswaController'. Controller ini menangani beberapa fungsi yang berkaitan dengan logbook mahasiswa. Berikut adalah penjelasan lebih rinci mengenai fungsi-fungsi yang ada didalamnya : Namespace dan Penggunaan Kelas : Kode ini menggunakan beberapa namespace dari Laravel, seperti 'App\Models', 'Carbon\Carbon', dan 'Illuminate\Http\Request', serta facade 'Auth' dan 'Storage'. Fungsi 'beranda()' : Fungsi ini mengembalikan tampilanview 'home'. Ini berfungsi sebagai halaman beranda dari aplikasi. Fungsi 'logbook()' : Fungsi ini digunakan untuk menampilkan logbook mahasiswa.

Pertama, mengambil data pengguna yang sedang login menggunakan 'auth()->user()'. Kemudian, mengambil id mahasiswa dari pengguna tersebut. Melakukan query ke tabel 'logbooks' untuk mendapatkan semua logbook milik mahasiswa yang sedang login, diurutkan berdasarkan tanggal pembuatan secara menurun. Menggunakan 'Carbon' untuk mendapatkan tanggal hari ini dalam format lokal Mengembalikan view 'mahasiswa.logbook.index'. dengan data 'logbook' dan 'hariIni'.

Fungsi 'logbookStore(Request \$request)' : Fungsi ini digunakan untuk menyimpan logbook baru ke dalam database. Melakukan validasi terhadap data yang diterima dari request. Validasi mencakup : 'deskripsi' : Wajib diisi, minimal 20 karakter. 'lampiran' : Wajin berupa file dengan tipe tertentu (jpeg, png, jpg, gif, svg, pdf, doc, docx, xls, xlsx) dan ukuran maksimal 2MB. Mengambil data pengguna yang sedang login dan id mahasiswa dari pengguna tersebut. Memeriksa apakah logbook untuk hari ini sudah ada atau belum untuk mencegah duplikasi entri logbook dalam satu hari. Jika sudah ada logbook untuk hari ini, mengembalikan pesan error.

Jika belum ada, membuat instance baru dari model 'logbook' dengan mengisi 'mhs_id' dan 'deskripsi' dari data yang sudah divalidasi. Jika ada file lampiran yang diunggah, menyimpannya di folder 'lampiran' dengan nama file yang unik berdasarkan tanggal dan waktu. Menyimpan data logbook ke database. Mengembalikan redirect ke route

'mahasiswa.logbook.index' dengan pesan sukses. Pada gambar 4.42 diatas, menunjukkan kode PHP dalam file 'LogbookController.php' yang merupakan bagian dari aplikasi berbasis Laravel. Kode ini mendefinisikan sebuah kelas 'LogbookController' yang memperluas (extends) kelas 'Controller'.

Kelas ini memiliki dua metode : 'index' dan 'store'. Metode 'index' mengembalikan tampilan 'mahasiswa.logbook.index'. Metode 'store' menerima permintaan ('Request \$request'), memvalidasi bahwa kolom 'deskripsi' harus diisi, kemudian mengambil ID **pengguna yang sedang login** dengan 'Auth::user()-> id' dan membuat entri logbook baru di basis data menggunakan model 'Logbook', dengan mengisi kolom 'mhs_id' dan 'deskripsi' sesuai dengan data yang diterima dari permintaan. Setelah itu, pengguna akan diarahkan kembali ke halaman sebelumnya dengan pesan sukses yang menunjukkan bahwa logbook telah berhasil dibuat.

View Logbook Pada gambar 4.43 diatas, menunjukkan kode bagian dari tampilan (view) dalam aplikasi Laravel **yang digunakan untuk menampilkan** data logbook mahasiswa. Layout dimulai dengan kolom dan kartu yang menampilkan nama pengguna yang sedang login. Data logbook dikelompokkan berdasarkan minggu menggunakan Carbon. Setiap minggu ditampilkan dalam bentuk panel accordion dengan rentang tanggal yang sesuai. Di dalam setiap panel, daftar logbook ditampilkan dengan detail seperti tanggal, deskripsi, lampiran, dan status. Jika logbook memiliki lampiran, ditampilkan gambar atau tautan ke file tersebut.

Status logbook ditampilkan dengan badge, dan ada tombol untuk mengedit logbook. Ketika tombol edit ditekan, sebuah modal terbuka yang berisi form untuk mengedit deskripsi dan lampiran logbook. Seluruh iterasi logbook diakhiri dengan penutupan tag yang sesuai. Pengujian No _Fitur _Tujuan _Langkah-langkah _Harapan _Hasil __ _Login _Memastikan fungsionalitas login dengan benar. _Admin, Dosen, Mitra, dan Mahasiswa **memasukkan email dan password yang** benar. Klik tombol "Masuk". _User (Admin, Dosen, Mitra, dan Mahasiswa) akan diarahkan ke tampilan beranda. _Berhasil __ _2.

_Beranda _Memastikan tampilan dan fungsionalitas beranda _Menampilkan card informasi yang sesuai dengan user yang sedang login. Login sebagai admin, dosen, mahasiswa, dan mitra perhatikan apakah semua menu dan fitur pada beranda dapat diakses. Klik setiap menu untuk memastikan fungsi bekerja. _Semua menu dan fitur pada beranda dapat diakses dan berfungsi dengan baik _Berhasil __ _3. _CRUD _Memastikan operasi CRUD berjalan dengan baik _Tambahkan data baru Lihat data yang ditambahkan Edit data yang telah ditambahkan Hapus data yang telah ditambahkan Verifikasi bahwa semua perubahan (tambah, edit, hapus) tercermin dengan benar pada sistem. _Data dapat ditambahkan, dilihat, diedit, dan dihapus

dengan benar. _Berhasil _4. _Logbook Harian _Memastikan fitur logbook harian mahasiswa berfungsi _Mahasiswa mengisi logbook harian.

Mitra menyetujui logbook Dosen melihat rekapan logbook _Logbook harian dapat diisi oleh mahasiswa, disetujui oleh mitra, dan direkap oleh dosen. _Berhasil _5. _Laporan Akhir _Memastikan fitur laporan akhir berjalan dengan benar _Mahasiswa mengunggah laporan akhir. Dosen melihat dan mendownload laporan akhir _Laporan akhir dapat diunggah oleh mahasiswa dan di download oleh dosen _Berhasil _6. _Jobdesc _Memastikan fitur jobdesc berjalan dengan benar _Mitra menginput jobdesc mahasiswa. Mahasiswa mendownload jobdesc _Jobdesc dapat diinput oleh mitra dan didownload oleh mahasiswa _Berhasil _7.

_Kriteria Penilaian _Memastikan fitur kriteria penilaian berjalan dengan benar _Admin menginput kriteria penilaian beserta jenis dan bobot _Kriteria penilaian dapat diinput oleh admin _Berhasil _8. _Nilai akhir _Memastikan fitur nilai akhir berjalan dengan benar _Dosen dan mitra memberikan nilai magang sesuai kriteria penilaian yang telah diinput oleh admin. Mahasiswa mendownload nilai tersebut. _Nilai magang dapat diberikan oleh dosen dan mitra sesuai kriteria, dan mahasiswa dapat mendownload nilai tersebut.

_Berhasil _ _BAB V

KESIMPULAN DAN SARAN Kesimpulan Berdasarkan penelitian yang dilakukan, dapat disimpulkan bahwa **penerapan metode waterfall dalam pengembangan** Sistem Presensi Mahasiswa Magang Prodi Informatika UAA Berbasis Website telah membawa dampak yang signifikan. Metode waterfall, dengan pendekatan tahap demi tahap yang terstruktur, memungkinkan perencanaan yang matang, pengembangan yang terarah, dan pengujian yang komprehensif. Hal ini membantu dalam mengatasi masalah yang dihadapi oleh program studi, seperti pengelolaan data kehadiran mahasiswa yang kurang efektif.

Dengan pendekatan ini, Prodi Informatika UAA dapat mengimplementasikan fitur-fitur penting dalam sistem presensi, seperti login untuk mahasiswa, dosen, mitra, dan admin. Pengelolaan logbook harian yang memungkinkan persetujuan oleh mitra dan direview oleh dosen, pengunggahan laporan akhir oleh mahasiswa yang dapat diakses dan diunduh oleh dosen, manajemen deskripsi pekerjaan oleh mitra yang dapat diunduh oleh mahasiswa, serta sistem kriteria penilaian yang diinput oleh admin dan diisi oleh dosen dan mitra. Fitur-fitur dalam sistem presensi memberikan nilai tambah yang signifikan bagi Prodi Informatika UAA.

Sistem login yang terpisah memudahkan setiap pengguna mengakses fitur yang relevan

dengan peran mereka, sementara pengelolaan logbook harian membantu memantau aktivitas mahasiswa magang yang lebih efisien. Pengunggahan dan akses laporan akhir secara digital mempercepat proses penilaian dan dokumentasi, dan manajemen deskripsi pekerjaan serta kriteria penilaian yang terintegrasi mempermudah kolaborasi antara mahasiswa, dosen, mitra, dan admin. Implementasi **metode waterfall dalam pengembangan** sistem presensi membantu prodi informatika UAA meningkatkan efisiensi operasional, mengoptimalkan pengelolaan magang, dan memberikan pengalaman yang lebih baik bagi semua pihak yang terlibat.

Saran Untuk meningkatkan fungsionalitas sistem, disarankan untuk mengembangkan fitur notifikasi otomatis yang memberitahu mahasiswa dan dosen mengenai jadwal penting, dan pengingat untuk pengunggah laporan akhir. Ini akan memastikan semua pihak selalu mendapatkan informasi terbaru tanpa harus memeriksa sistem secara manual. Disarankan juga untuk mengintegrasikan sistem presensi ini dengan sistem akademik lainnya di UAA. Integrasi ini akan mempermudah pengelolaan data mahasiswa secara keseluruhan dan memungkinkan sinkronisasi informasi yang lebih baik antara berbagai sistem.

Untuk memastikan semua pengguna dapat memanfaatkan sistem dengan optimal, diperlukan pelatihan khusus bagi mahasiswa, dosen, mitra, dan admin. Evaluasi berkala terhadap sistem perlu dilakukan untuk mengidentifikasi kekurangan dan area yang perlu ditingkatkan. Feedback dari pengguna sangat penting dalam proses evaluasi ini, sehingga perbaikan dan pengembangan sistem dapat dilakukan sesuai kebutuhan nyata. Dengan saran, sistem ini, diharapkan sistem presensi mahasiswa magang Prodi Informatika UAA dapat terus berkembang dan memberikan manfaat yang lebih besar bagi semua pihak yang terlibat.

INTERNET SOURCES:

<1% - <https://eprints2.undip.ac.id/id/eprint/18335/2/BAB%20I.pdf>

<1% -

https://ayotumandang.pacitankab.go.id/web/wp-content/uploads/2022/10/Permen_6_2020.pdf

<1% -

https://www.academia.edu/91894963/ANALISIS_EFEKTIFITAS_PROGRAM_MAGANG_UNTUK_SINKRONISASI_LINK_AND_MATCH_PERGURUAN_TINGGI_DENGAN_DUNIA_INDUSTRI_Studi_Terhadap_Program_Magang_Mahasiswa_Program_Studi_Administrasi_Bisnis_Politeknik_Negeri_Madiun

<1% -

http://repository.politanisamarinda.ac.id/id/eprint/612/4/F201500328_YULIANTO_MAGANG.pdf

<1% -

<https://fitk.uin-malang.ac.id/wp-content/uploads/2020/09/Pedoman-Kuliah-Kerja-Lapangan-Magang.pdf>

<1% -

<https://jogja.jpnn.com/jogja-terkini/4023/daftar-jurusan-di-universitas-alma-ata-yogyakarta-dan-akreditasinya>

<1% -

<http://fkt.almaata.ac.id/wp-content/uploads/2023/11/PANDUAN-KULIAH-KERJA-LAPANGAN-INF-2021.pdf>

<1% -

<https://www.kompas.com/edu/read/2023/01/03/124747271/593-lowongan-magang-kampus-merdeka-msib-buat-mahasiswa-cek-di-sini>

<1% -

<https://news.ddtc.co.id/perlakuan-pajak-atas-imbalan-tertentu-menurut-se-24-2018-15260>

<1% -

http://repository.pnb.ac.id/8369/3/RAMA_57401_2015323089_0013048502_0006078502_part.pdf

<1% - <https://ejournal.bsi.ac.id/ejurnal/index.php/ijse/article/download/11091/5168>

<1% -

https://www.academia.edu/60906621/Sistem_Absensi_Online_Berbasis_Web_Dengan_QR_Code_Secara_Real_Time_Menggunakan_Algoritma_Vigenere_Cipher

<1% -

https://www.researchgate.net/publication/349277543_Sistem_Informasi_Presensi_Mahasiswa_Berbasis_RFID_Menggunakan_Metode_Rapid_Application_Development

<1% -

https://www.researchgate.net/publication/359906841_Strategi_Kuliah_Sambil_Bekerja_Oleh_Mahasiswa_Studi_Kasus_Mahasiswa_Bekerja_di_Universitas_Negeri_Padang

<1% - <https://conference.upnvj.ac.id/index.php/senamika/article/download/2237/1699>

<1% - <https://ejournal.unis.ac.id/index.php/jutis/article/view/2376/1648>

<1% - <http://journal.iaincurup.ac.id/index.php/arcitech/article/download/9637/pdf>

<1% -

<https://dspace.uui.ac.id/bitstream/handle/123456789/37783/17523094.pdf?sequence=1>

<1% - <https://accurate.id/teknologi/laravel/>

<1% - <https://dspace.uui.ac.id/handle/123456789/37783>

<1% -

<https://fti.uniska-bjm.ac.id/wp-content/uploads/2023/08/DATA-MAHASISWA-SIDANG-S>

KRIPSI-SESI-2.pdf

<1% - <https://www.ekrut.com/media/tahapan-metode-waterfall>

<1% - <https://id.scribd.com/document/469701323/Skripsi>

<1% -

[https://repository.binadarma.ac.id/1537/1/Laporan%20KKP%20Januari%202020%20\(Diki%20Wahyudi%20171420082\).pdf](https://repository.binadarma.ac.id/1537/1/Laporan%20KKP%20Januari%202020%20(Diki%20Wahyudi%20171420082).pdf)

<1% -

<http://download.garuda.kemdikbud.go.id/article.php?article=2346751&val=8273&title=Sistem%20Informasi%20Pemesanan%20Kamar%20MESS%20PT%20KAI%20Persero%20Divre%20III%20Sumatera%20Selatan>

<1% -

<https://sumsel.antaranews.com/berita/745566/kai-divre-iii-palembang-sediakan-75968-seat-sambut-idul-adha-dan-liburan-sekolah>

<1% - <https://jurnal.unived.ac.id/index.php/jmi/article/download/5280/4251/>

<1% - <https://garuda.kemdikbud.go.id/documents/detail/3484496>

<1% -

<http://download.garuda.kemdikbud.go.id/article.php?article=3484496&val=30408&title=Perancangan%20Absensi%20Siswa%20berbasis%20Web%20Berbasis%20PHP%20MySQL%20di%20SMA%20Negeri%201%20Palupuh>

<1% - http://repository.upi.edu/98562/4/s_ind_0703686_chapter3.pdf

<1% - http://repository.upi.edu/111127/2/S_PKN_1900339_Chapter1.pdf

<1% - <https://e-journals.unmul.ac.id/index.php/JAR/article/viewFile/885/795>

<1% -

https://www.researchgate.net/publication/362066933_PENGEMBANGAN_PROFESI_GURU_DALAM_MENINGKATKAN_KINERJA_GURU

<1% - https://repository.dinamika.ac.id/id/eprint/2575/5/BAB_III.pdf

<1% -

<https://mediaindonesia.com/teknologi/531273/website-adalah-pengertian-jenis-dan-fungsi>

<1% -

https://repository.bsi.ac.id/repo/files/269557/download/File_10-Bab-II-Landasan-Teori-dikonversi.pdf

1% - <https://www.codehero.id/blog/pengertian-website>

<1% - <https://www.sekawanmedia.co.id/blog/website-adalah/>

<1% - <https://solmet.kemdikbud.go.id/?p=2550>

<1% - <https://www.rumahweb.com/journal/belajar-php-dasar/>

<1% -

<https://it.telkomuniversity.ac.id/metode-waterfall-dalam-pengembangan-perangkat-lunak/>

<1% - <https://journals.unisba.ac.id/index.php/JRTI/article/download/3311/1681/23842>

<1% - <https://journals.usm.ac.id/index.php/jisl/article/download/5107/2640>

<1% - <https://ids.ac.id/yuk-cari-tahu-6-tahapan-metode-waterfall-untuk-pengembangan-software/>

<1% - <https://jurnal.pn-balebandung.go.id/index.php/tekno/article/download/14/14/99>

<1% - <https://adoc.pub/siklus-hidup-perangkat-lunak-swdlc-software-development-life.html>

<1% - <http://eprints.umg.ac.id/6067/6/BAB%20III.pdf>

<1% - <https://s3.amazonaws.com/elexmedia/preview/9786020486222.pdf>

<1% - <https://idcloudhost.com/blog/pengertian-dan-keunggulan-framework-laravel/>

<1% - <https://layanancoding.com/blog/apa-itu-framework-laravel>

<1% - <https://socs.binus.ac.id/2018/12/13/kelebihan-menggunakan-laravel-web-development/>

<1% - <https://medium.com/komandro-ccit-ftui/tutorial-laravel-introduction-installation-4279407a396c>

<1% - <https://www.kompasiana.com/valentinowahyupratama2164/664eff98c925c4146028b1a2/unified-modelling-language-uml-definisi-fungsi-dan-jenisnya>

<1% - <https://media.neliti.com/media/publications/496212-none-d79cbaf9.pdf>

<1% - <https://www.sekawanmedia.co.id/blog/apa-itu-uml/>

<1% - <https://socs.binus.ac.id/2019/11/26/uml-diagram-use-case-diagram/>

<1% - <https://medium.com/d3ti2019-09/use-case-diagram-dan-activity-diagram-cca5b73e64cf>

<1% - https://repository.nusamandiri.ac.id/index.php/unduh/item/16280/File_10-Daftar-Simbol.pdf

<1% - <https://www.materidosen.com/2017/04/use-case-diagram-lengkap-studi-kasus.html>

<1% - <http://www.waskhas.com/2020/02/pengertian-use-case-diagram-dan.html>

<1% - <https://repository.atmaluhur.ac.id/bitstream/handle/123456789/588/10.%20DAFTAR%20SIMBOL.pdf?sequence=11>

<1% - <http://repository.unama.ac.id/1293/3/Bab%202.pdf>

<1% - https://repository.nusamandiri.ac.id/repo/files/7017/download/File_9%20Daftar%20Simbol.pdf

<1% - <https://blog.myskill.id/istilah-dan-tutorial/activity-diagram-menggambarkan-aliran-pros>

es-dengan-jelas/

<1% -

<https://bintanginspirasi.com/2019/08/pengertian-flowmap-diagram-konteks-uml-use-case-class-diagram-dan-activity-diagram%E2%94%82berserta-sumber.html>

<1% -

<https://medium.com/@nisawidyapangestika/mengenal-materi-activity-diagram-pemodelan-perangkat-lunak-bagian-4-ffe766e8af92>

<1% -

<https://repository.uinjkt.ac.id/dspace/bitstream/123456789/68334/1/SAKINAH-FST.pdf>

<1% -

https://repository.bsi.ac.id/repo/files/249896/download/FILE_9_DAFTAR-SIMBOL.pdf

<1% -

<https://psti.unisayogya.ac.id/2022/03/08/pengenalan-visual-studio-code-aplikasi-koding-paling-populer-saat-ini/>

<1% -

<https://manusiakomputer.medium.com/jenis-pengujian-sistem-testing-2fd0e5fb0565>

<1% - <https://medium.com/skyshidigital/pengujian-sistem-52940ee98c77>

<1% - <https://journal.udn.ac.id/index.php/eduscotech/article/download/104/117/>

<1% - <https://repository.unej.ac.id/handle/123456789/62094>

<1% - <https://www.mendeley.com/catalogue/d66abb50-aa13-346e-b77f-26a9d71aff8c/>

<1% - http://repository.upi.edu/21048/7/T_BIO_1302228_Chapter4.pdf

<1% - <https://suteki.co.id/aplikasi-sistem-informasi-akademik/>

<1% - http://repository.upi.edu/82508/4/S_PLS_046173_Chapter3.pdf

<1% - <http://eprints.umpo.ac.id/6883/5/BAB%20III.pdf>

<1% - <https://si.almaata.ac.id/skripsi/>

<1% -

https://www.researchgate.net/publication/356061373_Perancangan_aplikasi_sistem_presentation_karyawan_berbasis_web_di_PT_PWS_Reinsurance_Broker_Indonesia

<1% - <https://revou.co/kosakata/class-diagram>

<1% - <https://crocodic.com/php-laravel-vs-php-native/>

<1% - <https://asana.com/id/resources/what-is-a-flowchart>

<1% - <https://itbox.id/blog/activity-diagram-adalah/>

<1% -

<https://ichi.pro/id/sdlc-fase-model-siklus-hidup-pengembangan-perangkat-lunak-234877250586705>

<1% - <https://www.gabuttech.com/2023/10/cara-membuat-form-login-multi-user.html>

<1% -

http://repository.unika.ac.id/31905/5/19.N4.0003-Yehuda%20Joy%20Muljanto-BAB%20IV_a.pdf

<1% - <https://www.sekawanmedia.co.id/blog/use-case-diagram/>

<1% - <https://accurate.id/marketing-manajemen/entity-relationship-diagram/>
<1% - <https://medium.com/@myskill.id/creating-user-flow-and-user-journey-map-bf77824fab3b>
<1% - <https://www.hostnic.id/blog/tutorial/cara-menghubungkan-form-login-ke-menu-utama-html-tutorial/>
<1% - https://www.researchgate.net/publication/377315399_KUALITATIF_MEMAHAMI_KARAKTERISTIK_PENELITIAN_SEBAGAI_METODOLOGI
<1% - <https://educhannel.id/artikel/METODOLOGI-PENELITIAN/identifikasi-masalah-penelitian.html>
<1% - <https://dspace.uui.ac.id/bitstream/handle/123456789/15201/05.4%20bab%204.pdf>
<1% - <https://blog.rumahweb.com/wireframe-adalah/>
<1% - <https://dibimbing.id/blog/detail/tahapan-desain-thinking-dan-aplikasinya-dalam-ui-ux>
<1% - <https://tekno.kompas.com/read/2023/09/17/16150007/mengenal-tombol-tombol-pada-keyboard-dan-fungsinya>
<1% - <https://tambahpinter.com/pengertian-field-dalam-basis-data/>
<1% - <https://dosen.pancabudi.ac.id/panduan/Tutorial%20Portal%20Dosen.html?ProfilDosen.html>
<1% - <https://jurnal.umk.ac.id/index.php/biner/article/download/12683/4669>
<1% - <https://terralogiq.com/metode-waterfall/>
<1% - <https://www.hostinger.co.id/tutorial/perbedaan-mariadb-vs-mysql>
<1% - <https://codeascetic.com/merancang-database-sistem-informasi-akademik/>
<1% - <https://www.localstartupfest.id/faq/perbedaan-char-dan-varchar/>
<1% - <https://kantinit.com/database/tipe-data-mysql-yang-harus-kamu-ketahui/>
<1% - <https://jayjay.co/tipe-data-mysql-dan-penjelasan>
<1% - <https://bahasaweb.com/tipe-field-kolom-mysql/>
<1% - <https://kodekreasi.com/macam-macam-tipe-data-pada-mysql/>
<1% - <https://stackoverflow.com/programming/707874/differences-between-index-primary-unique-fulltext-in-mysql>
<1% - <https://www.otodidakkuy.com/2023/01/section-12-quiz-database-programming-with-sql-oracle-bahasa-indonesia.html>

<1% -

<https://www.kelasringan.com/tutorial-mysql-dasar-mengenal-tipe-data-pada-mysql/>

<1% - <https://zahiraccounting.com/id/blog/jurnal-pengeluaran-kas/>

<1% - <https://www.bee.id/blog/metode-perpetual/>

<1% -

<http://repository.umy.ac.id/bitstream/handle/123456789/15233/Bab%204.pdf?sequence=8>

<1% - <https://igun.uk/cara-membuat-auto-increment-di-mysql/>

<1% - <https://dosenit.com/sql/keyword-not-null-sql-fungsi-dan-contohnya>

<1% -

<https://www.erickunto.com/2023/08/rubrik-penilaian-definisi-manfaat-kesalahan-umum-dan-panduan-praktis-menyusunnya.html>

<1% -

http://repository.umy.ac.id/bitstream/handle/123456789/20879/07_Bab%204.pdf?sequence=8

<1% - <https://www.root93.co.id/2016/01/cara-menambahkan-auto-increment-dan.html>

<1% -

<https://learn.microsoft.com/id-id/sql/t-sql/statements/alter-table-column-constraint-transact-sql?view=sql-server-ver16>

<1% -

<https://sites.google.com/view/mediapembelajaranbasisdata/materi/materi-2-struktur-hirarki-dan-abstraksi-basis-data>

<1% - <https://midteknologi.com/blog/relasi-database/>

<1% - <http://ccc.komunitas.uksw.edu/2014/05/entity-relational-diagram-erd.html>

<1% - <http://repo.darmajaya.ac.id/6010/8/9.%20BAB%204.pdf>

<1% -

<https://www.ultraviolet-developer.com/sistem-pendukung-keputusan-menggunakan-metode-saw-berbasis-website-oop-php-dan-mysql/>

<1% - <https://www.hadirr.com/blog/rumus-absensi-karyawan-excel/>

<1% -

<https://biztechacademy.id/membangun-aplikasi-web-modern-dengan-paket-dan-fitur-laravel/>

<1% -

<https://inet.detik.com/telecommunication/d-6881721/arti-html-lengkap-dengan-penjelasan-fungsi-struktur-komponen-dan-contoh>

<1% - <https://kelasprogrammer.com/cara-membuat-session-di-codeigniter/>

<1% -

<https://smkmugaweleri.sch.id/cara-mudah-mengatasi-masalah-duplicate-entry-0-pada-kunci-utama-database/>

<1% -

<https://mti.binus.ac.id/2018/06/08/synthetic-minority-over-sampling-technique-smote-algorithm-for-handling-imbalanced-data/>

<1% -

<https://pusatinformasi.mitrakm.kemdikbud.go.id/hc/en-us/articles/5626252849049-Cara-Registrasi-Akun-Mitra>

<1% - http://repository.upi.edu/51565/6/S_PGSD_1602431_Chapter%205.pdf

<1% - <https://www.domainesia.com/berita/metode-waterfall/>