**PERANCANGAN SISTEM INFORMASI ALUMNI PENDIDIKAN**

**TEKNOLOGI INFORMASI FAKULTAS KEGURUAN**

**DAN ILMU PENDIDIKAN UNIVERSITAS**

**MUHAMMADIYAH KENDARI**



**IRMA**

**NIM. 21916040**

# PROPOSAL PENELITIAN

***Diajukan Sebagai Salah Satu Syarat Untuk Mengikuti Seminar Proposal***

**PROGRAM STUDI PENDIDIKAN TEKONLOGI INFORMASI**

**FAKULTAS KEGURUAN DAN ILMU PENDIDIKAN**

**UNIVERSITAS MUHAMMADIYAH KENDARI**

**KENDARI**

**2023**

# HALAMAN PERSETUJUAN PEMBIMBING

Telah diperiksa dan disetujui oleh pembimbing I dan pembimbing II untuk mengikuti ujian seminar proposal pada Program Studi Pendidikan Teknologi Informasi Fakultas Keguruan Dan Ilmu Pendidikan Universitas Muhammadiyah Kendari.

|  |  |  |
| --- | --- | --- |
| Nama | : | Irma |
| NIM | : | 21916040 |
| Judul | : | Perancangan Sistem Informasi Alumni Pendidikan Teknologi Informasi Fakultas Keguruan Dan Ilmu Pendidikan Universitas Muhammadiyah Kendari |

Menyetujui,

Pembimbing I Pembimbing II

# Muhammad Akbar Yasin, S.T., M.T Halima, S.Ag., M.Pd

# NIDN. 0907099101 NIDN. 0922059102

# Mengetahui,

# Ketua Program Studi Pendidikan Teknologi Informasi Fakultas Keguruan dan Ilmu Pendidikan

**Universitas Muhammadiyah Kendari**

**Zila Razilu,S.Pd., M.Pd**

**NIDN. 0922059102**

**DAFTAR ISI**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **HALAMAN JUDUL …………………………………………………………………………** | | | | i |
| **HALAMAN PERSETUJUAN ………………………………………………………………** | | | | ii |
| **DAFTAR ISI …………………………………………………………………………………** | | | | iii |
| **DAFTAR GAMBAR ………………………………………………………………………...** | | | | v |
| **DAFTAR TABEL ……………………………………………………………………………** | | | | vii |
| **BAB I PENDAHULUAN ……………………………………………………………………** | | | | **1** |
|  | 1. Latar Belakang …………………………………………………………………….. | | | 1 |
|  | 1. Rumusan Masalah …………………………………………………………..……. | | | 4 |
|  | 1. Batasan Masalah …………………………………………………………….…... | | | 4 |
|  | 1. Tujuan ……………………………………………………………………………... | | | 5 |
|  | 1. Manfaat ……………………………………………………………………………. | | | 5 |
| **BAB II LANDASAN TEORI ………………………………………………………………..** | | | | **6** |
|  | 1. Sistem Informasi Alumni ……………………………………………………… | | | 6 |
|  |  | 1. Sistem Informasi …………………………………………………………… | | 6 |
|  |  | 1. Sistem Informasi Alumni ………………………………………………….. | | 8 |
|  | 1. *Website* ………………………………………………………………………… | | | 9 |
|  | 1. *Framework* *Codegniter* ………………………………………………………. | | | 11 |
|  | 1. *Framework* *Boostrap* …………………………………………………………... | | | 11 |
|  | 1. *HTML* ………………………………………………………………………….. | | | 14 |
|  | 1. *PHP* ……………………………………………………………………………. | | | 16 |
|  | 1. *MYSQL* ………………………………………………………………………… | | | 19 |
|  | 1. Model *Waterfall ……………………………………………….……………………….* | | | 21 |
|  | 1. *UML* …………………………………………………………………………… | | | 23 |
|  |  | 1. *Use* *Case* *Diagram* ………………………………………………………… | | 24 |
|  |  | 1. *Activity* *Diagram* …………………………………………………………... | | 25 |
|  |  | 1. *Swim Lane Diagram ……………………………………………………………….* | | 26 |
|  |  | 1. *Sitemap Diagram …………………………………………………………………* | | 26 |
|  |  | 1. *Sequens* *Diagram* …………………………………………………………... | | 27 |
|  | 1. Penelitian Relevan ……………………………………………………………... | | | 28 |
| **BAB III METODE PENELITIAN ………………………………………………………….** | | | | **31** |
|  | 1. Jenis Penelitian …………………………………………………………………. | | | 31 |
|  | 1. Prosedur Penelitian ……………………………………………………………... | | | 31 |
|  | | | 1. Analisis …………....……………………………………………………….. | 31 |
|  | | | 1. Desain Sistem ……………………………………………………………… | 32 |
|  | | | 1. *Implementation* ……………………………………………………………. | 58 |
|  | | | 1. *Testing* ... …………………………………………………………………... | 58 |
|  | 1. Tempat dan Waktu Penelitian …...……………………………………………... | | | 60 |
|  | 1. Alat dan Bahan Penelitian ...……………………………………………………. | | | 60 |
|  | 1. Analisis dan Perancangan Perangkat Lunak …………………………………… | | | 62 |
| **DAFTAR PUSTAKA ………………………………………………………………………..** | | | | 65 |

**DAFTAR GAMBAR**

|  |  |
| --- | --- |
| Gambar 2.1 Hubungan antara *model, view,* dan *controller* …………………………...……… | 13 |
| Gambar 2.2 Sejarah *PHP* …………………………………………………………………….. | 18 |
| Gambar 2.3 Model *Waterfall* ………………………………………………………………. | 21 |
| Gambar 2.4 *Sequence* Diagram …………………………………………………………….… | 28 |
| Gambar 3.1 *Profile* Prodi PTI FKIP UMK ……………………………………………………. | 32 |
| Gambar 3.2 *Use case diagram* menu pendaftaran alumni ………………..…………………. | 33 |
| Gambar 3.3 *Use case diagram* menu profile alumni ………………...………………………. | 33 |
| Gambar 3.4 *Use case diagram* menu informasi jurusan …………………...………………… | 34 |
| Gambar 3.5 *Use case diagram* menu lowongan kerja ………………...……………………... | 34 |
| Gambar 3.6 *Activity diagram login user ……………………..……………………………………..* | 35 |
| Gambar 3.7 *Activity diagram admin menu* pendaftaran alumni …………………………….... | 36 |
| Gambar 3.8 *Activity diagram admin menu profile* alumni …………………………………… | 37 |
| Gambar 3.9 *Activity diagram admin menu* informasi jurusan ………………...……………... | 38 |
| Gambar 3.10 *Activity diagram admin* lowongan pekerjaan ………………………………..… | 38 |
| Gambar 3.11 *Acitivity diagram* alumni menu *profile ……………………...………………………* | 39 |
| Gambar 3.12 *Activity diagram* Kaprodi *menu profile* alumni ………………………………... | 40 |
| Gambar 3.13 *Activity diagram log out user …………………………………………………………* | 41 |
| Gambar 3.14 *Swim lane sistem* sedang berjalan …………………………...………………… | 41 |
| Gambar 3.15 *Swim lane sistem* yang diajukan ……………………………………………….. | 42 |
| Gambar 3.16 *Sitemap* …………………………..…………………………………………….. | 43 |
| Gambar 3.17 *Sequens diagram login user …………………………………………………………..* | 44 |
| Gambar 3.18 *Sequens diagram admin menu profile* alumni …………………………………. | 45 |
| Gambar 3.19 *Sequens diagram admin menu* informasi jurusan ……………………………… | 46 |
| Gambar 3.20 *Sequens diagram admin menu* informasi jurusan ……………………………… | 47 |
| Gambar 3.21 *Sequens diagram admin menu* lowongan kerja ………………………………... | 48 |
| Gambar 3.22 *Sequens diagram* Kaprodi *menu profile* alumni ……………………………….. | 49 |
| Gambar 3.23 *Sequens diagram* alumni *menu profile ………………………………………………* | 49 |
| Gambar 3.24 *ER Diagram ……………………………………………………………………………..* | 53 |
| Gambar 3.25 *Mockup login user ……………………………………...………………………………* | 54 |
| Gambar 3.26 *Mockup admin menu profile* alumni …………………………………………… | 55 |
| Gambar 3.27 *Mockup* *Admin* *menu* lowongan kerja ..……………...………………………… | 55 |
| Gambar 3.28 *Mockup* Kaprodi menu profile alumni *………..………………………….………….* | 56 |
| Gambar 3.29 *Mockup* Kaprodi menu lowongan kerja *…………………………………….……….* | 56 |
| Gambar 3.30 *Mockup* alumni menu *profile ……………………..………………………………….* | 57 |
| Gambar 3.31 *Mockup* alumni menu lowongan kerja *……………………...……………………….* | 58 |

**DAFTAR TABEL**

|  |  |
| --- | --- |
| Table 2.1 Simbol *use case diagram* …………………………………………………………... | 24 |
| Table 2.2 Simbol *activity diagram* ……………………………………………………………. | 25 |
| Table 2.3 Simbol *sequens diagram* ………….………………………………………………... | 27 |
| Table 2.4 *Data alumni PTI ……………..………………………………………………………………* | 31 |
| Table 3.1 *User ……………………………………………………………………………………………* | 50 |
| Tabel 3.2 Data\_*admin* ……………..…………………………….…………………………... | 50 |
| Tabel 3.3 Data\_kaprodi ………………………...……………………………………………... | 50 |
| Tabel 3.4 Data\_alumni ……………...………………………………………………………… | 51 |
| Tabel 3.5 Informasi\_jurusan …………………………..……………………………………… | 52 |
| Tabel 3.6 Lowongan\_pekerjaan ……………………………….…………………….……….. | 52 |
| Table 3.7 Kisi-kisi uji *black box ……………….……………………….…………………………….* | 59 |
| Tabel 3.8 Rencana kerja sistem informasi alumni …………………………….…………….. | 60 |

**BAB I**

**PENDAHULUAN**

1. **Latar belakang**

Teknologi informasi terus berkembang pesat dan mengikuti kebutuhan manusia di berbagai bidang kehidupan. Teknologi informasi adalah suatu perangkat yang digunakan seseorang untuk mengolah data dan mendapat informasi yang berpengaruh terhadap lingkungan masyarakat serta memberikan dampak positif dan negatif tergantung bagaimana seseorang menggunakannya, (Novi Yona Sidratul Munti & Dwi Asril Syaifuddin, 2020). Kemajuan teknologi informasi menjadi salah satu pertimbangan dalam memenuhi kebutuhan, kualitas yang ditawarkan menjadi nilai tambah dan mempermudah kegiatan manusia. Kolaborasi antara manusia dan teknologi memiliki pengaruh pada berbagai aspek kehidupan terlebih pendidikan, saat ini dunia pendidikan telah mengambil bagian dalam memanfaatkan teknologi informasi untuk peningkatan kualitas pendidikan. Dalam dunia pendidikan pemanfaatan teknologi informasi adalah hal yang perlu diterapkan salah satunya pemanfaatan teknologi komputer, selain untuk media pembelajaran, alat bantu pengolah data, bisa juga dimanfaatkan sebagai media penyimpanan informasi. Teknologi dapat meningkatkan kualitas sumber daya manusia yang membentuk individu lebih produktif di dunia kerja.(Budiman & Atmaja Darmawan, 2020).

Data dan informasi pada sebuah instansi merupakan hal yang sangat penting sebagai *asset* administrasi lembaga. Permasalahan yang kerap kali dihadapi adalah pada data yang tercecer bahkan hilang. Bukan hanya penyimpanan dalam bentuk fisik namun penyimpanan berkas pada komputer, laptop atau *smartphone* yang mana kurang ter managemen secara kompleks dan alat-alat yang rentan terhadap kerusakan bisa kapan saja terhapus atau hilang, mengakibatkan data tidak ter managemen, kurang terstruktur secara baik. Sehingga penyimpanan *online* berbasis *website* pada lembaga tepatnya program studi Pendidikan Teknologi Informasi sangat diperlukan.

Program studi (prodi) Pendidikan Teknologi Informasi (PTI) merupakan salah satu program studi di bawah naungan Fakultas Keguruan dan Ilmu Pendidikan (FKIP) Universitas Muhammadiyah Kendari (UMK). Pendidikan Teknologi Informasi di UMK berdiri pada tahun 2018 sehingga terhitung prodi yang masih cukup muda. Meskipun masih muda, Pendidikan Teknologi Informasi selalu berupaya untuk menghasilkan mahasiswa-mahasiswa yang kompeten, namun belum tersedianya sistem informasi untuk alumni maka akan mengalami kendala dalam memperoleh informasi terbaru mengenai alumni atau dalam berbagai informasi.

Berdasarkan hasil wawancara penulis lakukan 02 Maret 2023 dengan ketua program studi Pendidikan Teknologi Informasi bahwa data-data alumni yang ada masih belum lengkap, karena memang sistem pengisian data alumni masih secara konvensional memanfaatkan *excel* dan *google drive* kemudian data *file* yang didapatkan disimpan dalam komputer, pihak program studi akan terkendala dalam mendapatkan data informasi terbaru dari alumni itu sendiri, karena harus melakukan proses pengisian data seperti sebelumnya. Prodi PTI masih memiliki sedikit alumni karena termasuk prodi yang terhitung cukup muda. Berdasarkan pengalaman dari program studi lain di UMK yang telah memiliki banyak alumni, bahwa dalam penelusuran alumni merupakan hal yang cukup susah karena tidak adanya wadah. Dengan telah diketahui kendala yang tentu akan dihadapi kedepan maka prodi PTI perlu memiliki sistem alumni sebagai antisipasi dari masalah yang akan dihadapi sehingga dalam penelusuran alumni dan berbagai informasi lain akan lebih mudah.

Berdasarkan permasalahan di atas penulis menawarkan suatu sistem informasi alumni yang dapat menginformasikan data secara akurat dari para alumni yang dapat diakses dimana pun dan kapan pun. Sebuah perancangan sistem informasi alumni untuk penyimpanan data dan informasi terlebih era saat ini merupakan hal yang perlu dimiliki setiap instansi khususnya program studi. Penyimpanan *excel* dapat hilang atau pada *google drive* dapat terhapus namun *website* dengan dikunjungi maka akan ditemukan dengan penyimpanan *database* yang lebih ter managemen dan aman. Berdasarkan penelitian terdahulu yang telah dilakukan oleh (Robbani, 2019) bahwa dengan adanya *sistem* informasi alumni dapat membantu pihak Program Studi melakukan pendataan alumni dan memberi informasi kegiatan pada alumni. Dari sistem tersebut dan sistem yang akan peneliti buat memiliki perbedaan yaitu pada *form* galeri dan *form* prestasi, dimana sistem yang akan peneliti buat *form* *galeri* dan *form* prestasi akan berada dalam satu form yang sama yaitu *profile* alumni.

Sistem informasi alumni ini, diharapkan dapat memudahkan program studi dalam mengelola data dan informasi alumni, memudahkan komunikasi antara alumni dengan program studi, serta untuk contoh program studi lain dalam membuat sistem informasi alumni. Sistem informasi alumni memberikan informasi efektif dan efisien. Contohnya mahasiswa yang telah lulus dari Program Studi dapat mengakses datanya atau data-data alumni yang lain tanpa harus datang mencari *file* dahulu ke program studi, mereka hanya perlu mengakses data-data yang diinginkan dengan media internet. begitu pula dengan pihak program studi ketika data alumni akan diberikan pada Universitas, dengan menggunakan fitur laporan pada sistem maka akan mendapatkan informasi terbaru alumni secara keseluruhan dalam bentuk *file*. Dalam statuta Universitas Muhammadiyah Kendari bahwa informasi mengenai alumni dibutuhkan sebagai nilai tambah program studi itu sendiri. Informasi alumni dibutuhkan perguruan tinggi dalam proses akreditasi.

Berdasarkan uraian latar belakang diatas, maka peneliti tertarik untuk melakukan penelitian dengan judul: “Perancangan Sistem Informasi Alumni Pendidikan Teknologi Informasi Fakultas Keguruan dan Ilmu Pendidikan Universitas Muhammadiyah Kendari”.

1. **Rumusan Masalah**

Berdasarkan latar belakang di atas maka rumusan masalah pada penelitian ini adalah bagaimanakah merancang Sistem Informasi Alumni Pendidikan Teknologi Informasi Fakultas Keguruan dan Ilmu Pendidikan Universitas Muhammadiyah Kendari?

1. **Batasan Masalah**

Adapun batasan masalah yang terdapat dalam penelitian ini adalah sebagai berikut:

1. *Website* yang dirancang dan dibangun adalah Sistem Informasi Alumni Pendidikan Teknologi Informasi Fakultas Keguruan dan Ilmu Pendidikan Universitas Muhammadiyah Kendari.
2. Pengembangan *website* menggunakan *framework* *Codegniter* yang berbasis *PHP* dan menggunkan *framework* *Bootstrap*.
3. Pengembangan *website* ini menggunakan model *waterfall* yaitu *analysis*, Desain, implementasi/*coding* dan *testing.*
4. Pengembangan *website* ini menggunakan 5 diagram yang terdapat pada *Unified Modeling Language* (UML) yaitu*,* *use case diagram*, *activity diagram, swim lane diagram, sitemaps* dan *sequence diagram*.
5. Cakupan modul yang akan dikembangkan adalah profil alumni, informasi Program Studi dan lowongan pekerjaan.
6. **Tujuan**

Merancang Sistem Informasi Alumni Pendidikan Teknologi Informasi Fakultas Keguruan dan Ilmu Pendidikan Universitas Muhammadiyah Kendari pada Program Studi Pendidikan Teknologi Informasi Universitas Muhammadiyah Kendari.

1. **Manfaat**

Sesuai permasalahan dan tujuan penelitian yang telah disebutkan pada latar belakang beserta tujuan, manfaat yang diharapkan dapat diperoleh dari penelitian sistem ini adalah sebagai berikut:

1. Manfaat teoritis

Sebagai referensi bagi kepentingan ilmu pengetahuan, terutama bagi prosedur tindakan dalam merancang Sistem Informasi Alumni Pendidikan Teknologi Informasi Fakultas Keguruan dan Ilmu Pendidikan Universitas Muhammadiyah Kendari, dan dapat menjadi rujukan untuk penelitian berikutnya yang relevan dengan penelitian ini.

1. Manfaat praktis
   1. Dapat mempermudah dalam rekapitulasi dan informasi mengenai alumni untuk *staff*, kaprodi dan mahasiswa.
   2. Sistem Informasi Alumni Pendidikan Teknologi Informasi FKIP UMK ini dapat menjadi nilai tambah akreditas program studi.
   3. Dapat melihat akuntabilitas alumni.
2. Manfaat bagi peneliti
3. Meningkatkan kontribusi pada Program Studi.
4. Meningkatkan profil akademik.
5. Meningkatkan keterampilan penelitian.
6. Memberikan wawasan untuk pengembangan program.
7. Memperdalam pemahaman tentang alumni.

**BAB II**

**LANDASAN TEORI**

1. **Sistem Informasi Alumni**
2. **Sistem informasi**

Menurut (Banyal et al., 2021) Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. Secara umum, system merupakan sekumpulan elemen perangkat lunak dan perangkat keras yang saling berinteraksi serta bekerja sama untuk memproses, menyimpan, dan mentransmisikan informasi dalam suatu organisasi atau jaringan untuk mencapai tujuan bersama. Sistem memiliki batasan yang jelas, input, proses, dan output yang terdefinisi. dalam realisasinya mengilustrasikan gabungan manusia saling berkolaborasi sesuai aturan yang terstruktur membentuk kesatuan untuk menjalankan fungsi dalam mencapai tujuan. Elemen, batasan, lingkungan luar, masukan, keluaran, pengolahan, dan sasaran sistem merupakan karakteristik yang ada dalam sistem. Sistem dapat didefinisikan dengan dua pendekatan yaitu a) pendekatan prosedur yaitu kumpulan prosedur-prosedur yang memiliki tujuan tertentu, b) pendekatan komponen yaitu kumpulan dari komponen yang saling berhubungsatu dengan yang lain membentuk satu kesatuan untuk mencapai tujuan tertentu.

Informasi adalah data yang diproses, disimpan, dan ditransmisikan melalui jaringan teknologi informasi seperti internet, intranet, atau komputer. Informasi menggambarkan kumpulan data yang sudah diproses menjadi wujud spesifik sesuai dengan kebutuhan. Data dan informasi memiliki tingkatan yang berbeda. Data berada di tingkat manajemen, sedangkan informasi berada di tingkat transaksi. Data berkaitan erat dengan informasi. Secara sederhana, data yang sudah diproses memerankan wujud yang mempunyai nilai disebut informasi. Menurut (Banyal et al., 2021) Informasi berupa data yang telah di olah menjadi bentuk yang lebih berarti dan berguna bagi penerimanya untuk mengambil keputusan masa kini maupun masa yang akan datang.

Sistem informasi adalah kumpulan komponen dan adanya informasi yang berkaitan untuk mencapai suatu tujuan tertentu sistem informasi adalah kombinasi dari teknologi informasi dan aktivitas orang yang menggunakan teknologi itu untuk mendukung operasi dan manajemen. Sistem informasi membantu memecahkan masalah bisnis, meningkatkan efisiensi dan produktivitas, dan mempermudah pengambilan keputusan. Membentuk suatu gabungan berbagai elemen yang ada di organisasi atau perusahaan berkaitan dengan proses penciptaan dan peredaran informasi. Dalam hal ini, salah satu elemen kecil dalam pola organisasi adalah teknologi informasi. Kemampuan sistem informasi yang ada di lembaga, organisasi maupun perusahaan terletak pada hubungan antar elemen-elemen yang ada untuk dapat dihasilkan dan diedarkan informasi yang bernilai. Sistem informasi terdiri dari beberapa komponen.

Komponen-komponen sistem informasi menurut Jogiyanto (2005:12) dalam penelitian yang dilakukan oleh (Moshinsky, 1959) terdiri dari *Hardware, Software*, Data, *Prosedur*, dan Manusia. *Hardware* adalah bagian fisik dari komputer, termasuk perangkat keras seperti *motherboard*, *prosesor*, memori, *hard drive*, dan lain-lain. Ini berbeda dengan *software*, yang merupakan program komputer dan data yang menjalankan tugas pada komputer. Beberapa jenis *software* termasuk sistem operasi, aplikasi, dan perangkat lunak pemrograman. Data prosedur adalah serangkaian langkah yang terstruktur dan sistematis yang mengikuti alur logika tertentu untuk menyelesaikan suatu tugas atau memecahkan suatu masalah. Ini sering digunakan dalam bidang teknologi informasi untuk memastikan bahwa tugas-tugas komputer atau aplikasi dieksekusi dengan benar dan efisien. Data prosedur dapat ditemukan dalam bentuk panduan, manual, atau alur kerja yang dipersiapkan untuk memandu pengguna melalui tugas tertentu. kemudian komponen terakhir menurut Jogiyanto yaitu manusia yang berperan dalam menggunakan, membuat, atau memelihara sistem informasi. Mereka mungkin bekerja sebagai pengguna akhir, pengembang *software*, *administrator* *sistem*, atau profesional TI lainnya. Peran manusia dalam sistem informasi sangat penting karena mereka bertanggung jawab untuk memastikan sistem bekerja dengan benar dan memenuhi kebutuhan organisasi atau individu yang menggunakannya. Manusia juga dapat mempengaruhi bagaimana sistem informasi berkembang dan beradaptasi dengan perubahan lingkungan bisnis atau teknologi.

1. **Sistem Informasi Alumni**

Alumni merupakan siswa ataupun mahasiswa yang telah menyelesaikan jenjang pendidikan dengan segala aturannya pada sebuah institusi pembelajaran. Alumni yaitu salah satu aset berharga yang harus dijaga karena mewakili perguruan tinggi di dunia nyata. Keberadaaan Alumni pada saat ini menjadi salah satu bagian penilaian akreditasi sebuah lembaga pendidikan (Robbani, 2019).

Sistem Informasi Alumni adalah aplikasi atau platform yang digunakan untuk mengelola dan mengkoordinasikan informasi tentang alumni universitas, atau lembaga pendidikan. Sistem informasi alumni merupakan salah satu contoh dari suatu sistem informasi yang di rancang untuk dapat membantu pekerjaan dari suatu instansi pendidikan khususnya diperguruan tinggi dalam mengolah data alumni sampai memberikan data tersebut secara lengkap lewat tersediannya layanan informasi berbasis web. Data-data tersebut meliputi modul pendaftaran, modul profil, modul laporan, informasi perkembangan program studi dan informasi lowongan pekerjaan. suatu sistem informasi dirancang untuk dapat membantu pekerjaan perguruan tinggi khususnya program studi baik dalam mengolah data sampai memberikan data secara lengkap lewat tersedianya layanan informasi sehingga lebih efesien terhadap program studi dan mahasiswa.

1. ***Website***

*Website* adalah suatu halaman *web* yang saling terhubung dan file-filenya saling terkait. *Web* terdiri dari *page* atau halaman, dan kumpulan halaman yang dinamakan *homepage*. *Homepage* berada pada posisi teratas, dengan halaman-halaman yang terkait berada di bawahnya. Biasanya setiap halaman di bawah *homepage* disebut *child page* dan berisi *hyperlink* ke halaman lain dalam suatu web. *Hyperlink* adalah metode yang digunakan untuk menghubungkan halaman satu dengan halaman lainnya yang bisa ditemukan dalam artikel, presentasi, Word, dan sejenisnya. Biasanya, *hyperlink* dibuat dalam bentuk teks atau kata kemudian ditautkan dengan suatu *link* atau lokasi halaman yang dituju. *Website* dibuat pertama kali oleh Tim Berners Lee pada pada tahun 1990. (Bekti, 2015) mengemukakan bahwa: “*Website* adalah kumpulan halaman-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”.

Struktur *website* merupakan bagian penting dari desain dan organisasi sebuah *website* dan mempengaruhi bagaimana pengguna menavigasi dan menggunakan situs. Struktur *website* membantu menentukan bagaimana halaman-halaman yang membentuk *website* terkait satu sama lain, sehingga memudahkan pengguna untuk menemukan informasi dan melakukan tugas yang diinginkan.

*Website* sendiri adalah hasil akhir dari struktur *website* yang baik. *Website* yang memiliki struktur yang jelas dan mudah dipahami akan lebih mudah dinavigasi dan lebih efektif dalam memenuhi tujuannya, seperti memberikan informasi, menjual produk, atau memfasilitasi komunikasi antar pengguna. Dengan kata lain, struktur website membentuk dasar dari sebuah *website*, dan *website* adalah hasil akhir dari struktur tersebut. Kedua hal tersebut saling mempengaruhi satu sama lain dan bekerja sama untuk menciptakan pengalaman pengguna yang baik dan efektif.

Struktur *website* yang baik memudahkan pengguna untuk menemukan informasi yang mereka cari dan memahami bagaimana bagian-bagian dari *situs* terkait. Berikut adalah beberapa elemen umum dari struktur *website* :

1. Halaman utama: Halaman utama dari sebuah *website*, seringkali halaman pertama yang dilihat pengguna ketika mereka mengunjungi situs.
2. Menu navigasi: Daftar tautan ke bagian dan halaman berbeda dari sebuah website, biasanya terletak di bagian atas atau samping halaman.
3. *Footer*: Bagian di bagian bawah dari sebuah *website* yang biasanya berisi tautan ke halaman penting, seperti Halaman Tentang Kami, Hubungi Kami, dan Kebijakan Privasi.
4. *Header*: Bagian di bagian atas dari sebuah *website* yang sering berisi logo, menu navigasi, dan informasi penting lainnya.
5. Area konten: Bagian utama dari sebuah *website* di mana konten ditampilkan, seperti artikel, posting blog, deskripsi produk, dan gambar.
6. *Sidebar*: Kolom di samping sebuah *website* yang sering berisi informasi tambahan, seperti tautan ke konten terkait, posting terbaru, atau iklan.
7. Jejak roti bakar: Bantuan navigasi yang menunjukkan lokasi pengguna dalam *website*, biasanya ditampilkan sebagai serangkaian tautan yang dipisahkan oleh panah.
8. Bilah pencarian: Fitur yang memungkinkan pengguna untuk mencari konten tertentu dalam sebuah *website*.

*Website* dapat dibuat oleh individu atau organisasi untuk berbagai tujuan, seperti mempromosikan produk atau layanan, berbagi informasi, atau memfasilitasi komunikasi antar pengguna. *Website* biasanya dihosting pada server dan dapat diakses oleh pengguna dari seluruh dunia dengan mengetikkan alamat *URL* yang sesuai.

1. ***Framework Codedgniter***

*Framework* atau dalam bahasa Indonesia dapat diartikan sebagai “kerangka kerja” merupakan kumpulan dari fungsi-fungsi/prosedur-prosedur dan class-class untuk tujuan tertentu yang sudah siap digunakan sehingga bisa lebih mempermudah dan mempercepat pekerjaan seorang programer, tanpa harus membuat fungsi atau class dari awal (Bin Tahir et al., 2019). Pengertian *framework* menurut Naista dalam penelitian yang dilakukan oleh (Agus Efendi, 2020) adalah suatu struktur konseptual dasar yang digunakan untuk memecahkan atau menangani suatu masalah yang kompleks. Singkatnya, *framework* merupakan wadah atau kerangka kerja dari sebuah website yang akan dibangun. Dengan menggunakan kerangka tersebut waktu yang digunakan dalam membuat website lebih singkat dan memudahkan dalam melakukan perbaikan.

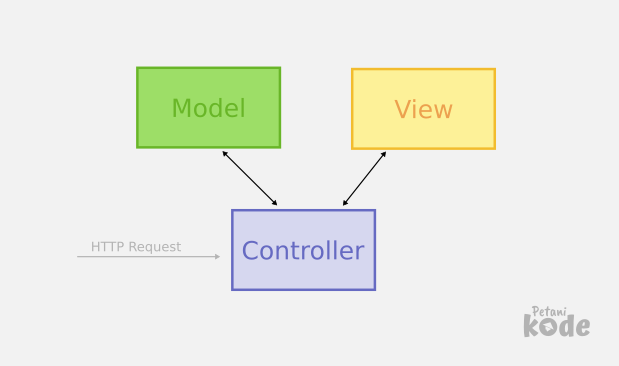
CodeIgniter merupakan aplikasi open source yang berupa *framework* dengan model MVC (*Model, View, Controller*) untuk membangun website dinamis dengan menggunakan PHP (Binarso Yusi Ardi, Sarwoko Eka Adi, 2012).

*Model* merupakan struktur data. Secara spesifik class model akan mengandung fungsi kode yang akan membantu dalam segala proses yang berhubungan dengan database seperti memasukkan, mengedit mendapatkan dan menghapus data dalam sebuah database (Prabowo, 2015). *Model* mengelola basis data (RDBMS) seperti MySQL ataupun Oracle RDMS. *Model* berhubungan dengan *databases* sehingga biasanya dalam model akan berisi class ataupun fungsi untuk membuat (create) melakukan pembaruan(update), menghapus data (delete), mencari data (search), dan mengambil data (select), pada databases. Selain itu juga model akan berhubungan dengan perintah-perintah query sebagai tindak lanjut dari fungsi-fungsi *(create, update, delete* dan *select).*

*View* merupakan informasi yang disampaikan ke pengguna. Sebuah view biasanya berupa halaman web, view juga bisa berupa sebuah fragmen halaman seperti *header* dan *footer*. View juga bisa berupa halaman RSS atau jenis halaman web yang lain (Prabowo, 2015). *View* bisa berupa halaman *html, css, rss, avascript jquery, ajax,* dan lain lain. karena metode yang dipakai merupakan *MVC* sehingga dalam *view* tidak boleh terdapat pemrosesan data ataupun pengaksesan yang berhubungan dengan *database*. Sehingga view hanya menampilkan data-data hasil dari *model* dan *controller*. Selain itu, ketika model berubah, view secara otomatis menggambar ulang bagian layar yang terkena perubahan untuk menunjukkan perubahan tersebut.

*Controller*  merupakan sebuah perantara antara model dan *view* dan semua sumber yang dibutuhkan untuk memproses permintaan *HTTP* dalam membuat halaman *web* (Prabowo, 2015). Maksudnya ialah karena model tidak dapat berhubungan langsung dengan view ataupun sebaliknya jadi, *controller* inilah yang digunakan sebagai jembatan dikeduanya. Sehingga tugas *controller* ialah sebagai pemrosesan data atau alur *logic* program, menyediakan variable yang akan ditampilkan di view, pemanggil model sehingga model dapat mengakses *databases, error handling*, validasi atau *check* terhadap suatu inputan. *Controller* bertanggung jawab untuk pemetaan aksi pengguna akhir terhadap respon aplikasi. Sebagai contoh, ketika pengguna mengklik tombol atau memilih item menu, *controller* bertanggung jawab untuk menentukan bagaimana aplikasi seharusnya merespon.

Berikut adalah ilustrasi dari konsep *MVC* pada *Codegniter* yang ditunjukkan pada Gambar 1.



Gambar 2.1 Hubungan antara *model, view,* dan *controller*

1. ***Framework* *Bootstrap***

Bootstrap adalah sebuah *framework* yang dibuat menggunakan bahasa *HTML* dan *CSS*, namun juga menyediakan efek *javascript* yang dibangun dengan menggunakan *jquery*. *Bootstrap* diciptakan untuk mempermudah proses desain *web* bagi berbagai tingkat pengguna, mulai dari *level* pemula hingga yang sudah berpengalaman. Cukup bermodalkan pengetahuan dasar mengenai *HTML* dan *CSS*, maka dapat menggunakan *bootstrap*. Kelebihan *Bootstrap* adalah adanya dokumentasi super komplit di *website* nya. *Bootstrap* telah menyediakan kumpulan komponen *class* *interface* dasar yang telah dirancang sedemikian rupa untuk menciptakan tampilan yang menarik, bersih dan ringan. Selain itu, *bootstrap* juga memiliki fitur *grid* yang berfungsi untuk mengatur *layout* yang bisa digunakan dengan sangat mudah dan cepat. Kita juga diberi keleluasaan dalam mengembangkan tampilan *website* yang menggunakan bootstrap yaitu dengan mengubah tampilan *bootstrap* dengan menambahkan *class* dan *CSS* sendiri (Sanjaya & Hesinto, 2018)**.**

Untuk menggunakan *Bootstrap*, harus memasukkan file *CSS* dan *JavaScript* ke halaman *web*. Ini bisa dilakukan dengan mengunduh *file* dari *situs web* *Bootstrap* atau mengunduh melalui *package manager* seperti *npm*. Dengan menggunakan *Bootstrap*, *developer* dapat mempercepat proses pengembangan halaman *web* dan memastikan tampilan yang konsisten di berbagai perangkat.

1. ***HTML***

*Hypertext Markup Language (HTML)* adalah bahasa standar yang digunakan untuk menampilkan halaman *web* (Priyanto Hidayatullah, 2015) dalam penelitian yang dilakukan oleh (Endra & Aprilita, 2018). Fungsi-fungsi yang dapat dilakukan dengan bahasa programan *HTML* adalah : (1) Mengatur serta mendesain tampilan isi halaman *website*, (2) Membuat tabel pada halaman *website*, (3) Mempublikasikan halaman *website* secara *online*, (4) Membuat *form* yang dapat menjadi input serta menangani *registrasi* dan transaksi *via* *website*, (5) Menampilkan area gambar pada *browser*.

*HTML* menggunakan tag (label) untuk menandai bagian-bagian tertentu dari halaman web dan menentukan bagaimana konten akan ditampilkan di browser *web*. *Tag* diawali dengan tanda kurung siku "<" dan ditutup dengan tanda kurung siku ">". *HTML* menentukan struktur dokumen *web*, seperti bagian *header*, navigasi, main content, dan bagian *footer*. Struktur ini membantu *browser web* dan mesin pencari mengerti bagaimana halaman web harus ditampilkan dan diterjemahkan. Setiap bagian dari halaman web disebut elemen. Beberapa elemen *HTML* yang paling umum meliputi heading, paragraf, gambar, *tabel*, dan tautan. Beberapa tag *HTML* memiliki atribut tambahan yang memodifikasi bagaimana elemen tersebut ditampilkan. Misalnya, tag "*img*" memiliki atribut "*src*" yang menentukan lokasi gambar yang ingin ditampilkan. *HTML* memiliki beberapa tag yang memiliki makna semantik tertentu, seperti tag "*header*" yang menandakan bagian *header* dari halaman web, atau tag "*nav*" yang menandakan bagian navigasi. Ini membantu mesin pencari dan pembaca layar untuk memahami konteks konten.

*TAG* digunakan untuk menentukan tampilan dari dokumen *HTML*. *TAG* ada yang berpasangan dengan format berikut :

<BEGIN TAG> *isi dokumen* </END TAG>

Tag ada juga yang tidak berpasangan dengan format berikut ini :

<TAG> *isi dokumen*

Namun perlu diperhatikan bahwa tag <p> lebih baik diakhiri dengan tag </p> karena untuk kedepannya, *HTML* mengharuskan penggunaan *end tag.*

contoh : setiap dokumen *HTML* diawali dan diakhiri dengan tag *HTML*.

<html>

<head>

. . ….. . .

</head>

<body>

. . ….. . .

</body>

</html>

Elemen Head kepala dari dokumen *HTML*. *tag* <*head*> dan *tag* </*head*> terletak dianatara tag <*html*> dan tag </*html*>. elemen title merupakan judul dari dokumen *HTML* yang ditampilkan pada judul jendela browser. *tag* <*title*> dan tag </*title*> terletak diantara tag <head> dan tag </head>. elemen Body menampilkan isi dokumen *HTML*. *tag* <body> dan tag </body> terletak dibawah tag <head> dan tag </head>. elemen body mempunyai atribut-atribut yang menspesifikasikan khususnya warna dan latar belakang dokumen yang akan ditampilkan pada *browser*.

*HTML* banyak memberi kemudahan pada programmar yaitu mengontrol struktur dan konten halaman web, membuat halaman *web* yang *responsive* dan interaktif, mendukung aksesibilitas kemudian dengan bahasa markup sehingga lebih sederhana dan mudah dipelajari, bahkan bagi pemula.

1. ***PHP***

PHP adalah sebuah bahasa pemrograman yang di desain agar dapat disisipkan dengan mudah ke halaman *HTML*. *PHP* memberikan solusi sangat murah (karena gratis digunakan) dan dapat berjalan di berbagai jenis platform. Pada awalnya memang *PHP* berjalan di *sistem* *UNIX* dan variannya, namun kini dapat berjalan dengan lancar di lingkungan sistem operasi *Windows*. Suatu nilai tambah yang luar biasa karena proses pengembangan program berbasis *web* dapat dilakukan lintas sistem operasi (Bachtiar, 2010). Sebagian besar sintaks *PHP* mirip dengan bahasa C, java, dan perl. Namun, PHP terdapat beberapa fungsi yang lebih spesifik. *PHP* digunakan untuk merancang yang sifatnya dinamis dan dapat bekerja secara *otomatis* (Mediana & Nurhidayat, 2018). Karena *PHP* merupakan suatu serverside-scripting maka sintaks dan perintah-perintah *PHP* akan diesksekusi diserver kemudian hasilnya akan dikirimkan ke browser dengan format *HTML*. Dengan demikian kode *program* yang ditulis dalam suatu *PHP* tidak akan terlihat oleh *user* sehingga keamanan halaman *web* lebih dapat terjamin (Saputra & Ari Waluyo, 2020).

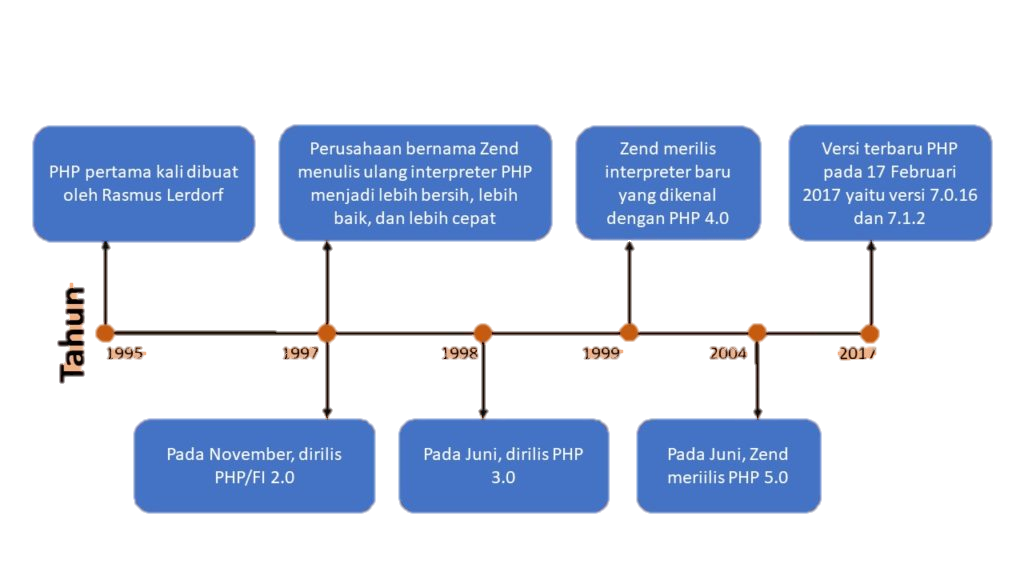
Penemu bahasa pemrograman ini adalah Rasmus Lerdorf, yang bermula dari keinginan sederhana Lerdorf untuk mempunyai alat bantu dalam memonitor pengunjung yang melihat situs web pribadinya. Inilah sebabnya pada awal pengembangannya, *PHP* merupakan singkatan dari Personal *Home Page tools*, sebelum akhirnya menjadi Page Hypertext Preprocesso (Bachtiar, 2010). PHP bersifat dinamis yaitu halaman web yang dapat membentuk suatu tampilan - tampilan berdasarkan permintaan terupdate, seperti menampilkan isi basis data di halaman web. PHP dapat dijalankan pada berbagai macam sistem operasi seperti *Windows, Linux*, dan *Mac Os.* Selain *Apache, PHP* juga mendukung beberapa *web server* lain, seperti *Microsoft ISS*, *Caudium*, dan *PWS*. *PHP* dapat memanfaatkan database untuk menghasilkan halaman *web* yang dinamis. Sistem manajemen database yang sering digunakan bersama *PHP* adalah *MYSQL*. Namun, *PHP* juga mendukung *sistem* manajemen *Database Oracle, Microsoft Access, Interbase, d-Base*, dan PostgreSQL (Novendri, 2019).

Aplikasi-aplikasi yang dibangun oleh *PHP* pada umumnya akan memberikan hasil pada *web browser* tetapi prosesnya secara keseluruhan dijalankan di *server*, pada prinsipnya server akan bekerja apabila ada permintaan dari client. Dalam hal ini client menggunakan kode-kode *PHP* untuk mengirimkan permintaan ke *server* (Usada et al., 2012).

Beberapa kelebihan *PHP* sebagai berikut

1. *PHP* berbasis *server side* *scripting*
2. *Command line scripting* pada *PHP*
3. *PHP* dapat membuat aplikasi *desktop*
4. Digunakan untuk berbagai macam *platform OS*
5. Mendukung berbagai macam *web server*
6. *Object oriented* *programming* atau *procedural*
7. *Output file PHP* pada *XHTML, HTML* dan *XML*
8. Mendukung banyak *RDMS (database)*
9. Mendukung banyak komunikasi
10. Pengolahan teks yang sangat baik

Adapun sejarah *PHP* sebagai berikut



Gambar 2.2 Sejarah *PHP*

1. ***MYSQL***

*MySql* merupakan aplikasi data *server*. Perkembangannya disebut juga *Sql* yang merupakan kepanjangan dari *Structured Query Language*. SQL merupakan bahasa terstruktur yang khusus digunakan untuk mengolah database. SQL pertama kali didefinisikan oleh *American National Standards Institute (*ANSI*)* pada tahun 1986. *SQL* merupakan konsep dari pengoperasian *database*, terutama untuk pemilihan atau menyeleksi dan memasukan data, yang kemungkinan pengoperasian data dikerjakan dengan sangat mudah secara otomatis. *Sql* dengan bahasa terstruktur yang digunakan untuk mengolah *database*. Mysql termasuk jenis *RDBMS* (*Relational Database Management System)* yang diditribusikan secara gratis dibawah licen*si GPL (General Public License). Database* besar seperti *Mysql, PostgreSQL* dan *SQL Server* sudah menggunakan *SQL* untuk mengolah *database* nya. *SQL* yang di pergunakan *software database* tersebut adalah sama kecuali sedikit perbedaan di beberapa tempat.

*MYSQL sistem manajemen database* yang bersifat *relational*. Artinya, data yang dikelola dalam *database* yang akan diletakkan pada beberapa tabel yang terpisah sehingga manipulasi data akan jauh lebih cepat. *MYSQL* dapat digunakan untuk mengelola database mulai dari yang kecil sampai dengan yang sangat besar. (Andi, Wahana Komputer, 2014:h,73). *SQL* juga dapat diartikan sebagai antar muka standar untuk sistem manajemen relasional, termasuk sistem yang beroperasi pada komputer pribadi.

*MySQL* menggunakan bahasa *SQL* untuk mengakses database nya. Lisensi *Mysql* adalah *FOSS* *License* *Exception* dan ada juga yang versi komersial nya. *Tag Mysql* adalah “*The World's most popular open source database*”. *MySQL* tersedia untuk beberapa *platform*, di antara nya adalah untuk versi *windows* dan versi *linux*. Untuk melakukan administrasi secara lebih mudah terhadap *Mysql*, anda dapat menggunakan *software* tertentu, di antara nya adalah *phpmyadmin* dan *mysql yog*. *MySQL* dapat dikatakan lebuh unggul dibandingkan *database server* lainnya dalam *query* data. Hal ini terbukti untuk *query* yang dilakukan oleh *single user*, kecepatan *query My SQL* dapat sepuluh kali lebih cepat dari *PostgreSQL* dan lima kali lebih cepat dibandingkan *Interbase*.

*SQL* memungkinkan seorang pengguna untuk mengetahui dimana lokasinya, atau bagaimana informasi tersebut disusun. *SQL* lebih mudah digunakan dibandingkan dengan bahasa pemrograman, tetapi rumit dibandingkan software lembar kerja dan pengolah data. Sebuah pernyataan *SQL* yang sederhana daoat menghasilkan set permintaan untuk informasi yang tersimpan pada komputer yang berbeda diberbagai lokasi yang tersebar, sehingga membutuhkan waktu dan sumber daya komputasi yang banyak. *SQLinte* dapat digunakan untuk ivestigasi interaktif, atau pembuatan laporan ad hoc atau disisipkan dalam program aplikasi. *SQL* juga merupakan bahasa pemrograman yang dirancang khusus untuk mengirimankan suatu perintah *query* (pengaksesan data berdasarkan pengalamatan tertentu) terhadap sebuah database. Kebanyakan *software database* mengimplementasikan *SQL* secara sedikit berbeda, tapi seluruh *database* *SQL* mendukung subset standar yang ada.Jadi, *SQL* adalah permintaan yang melekat pada suatu *database* atau *SMBD* tertentu. Dengan kata lain, *SQL* adalah perintah atau bahasa yang melekat di dalam SMBD. Sebagai suatu bahasa permintaan, *SQL* didukung oleh *SMBD*, seperti *MySQL Server, MySQL, PostgreSQL, Interbase*, dan *Oracle*. Selain itu *SQL* juga didukung oleh *database* bukan *server*, seperti *MS Acess* maupun *Paradox*.(Surniawan, Wahana Eri Mardiani, 2014:h,25,26).

1. **Model *Waterfall***

Metode air terjun atau yang sering disebut metode *waterfall* adalah suatu metodologi pengembangan perangkat lunak atau *software* yang menggunakan pendekatan secara *linear* dan berurutan dalam proses pengembangannya. Model *waterfall* pertama kali diperkenalkan oleh Winston Royce sekitar tahun 1970 sehingga sering dianggap kuno, namun merupakan model yang paling banyak dipakai didalam *Software Engineering* (*SE*). Saat ini model *waterfall* merupakan model pengembangan perangkat lunak yang sering digunakan. Model ini melakukan pendekatan secara sistematis dan urut mulai dari level kebutuhan sistem lalu menuju ke tahap analisis, desain, coding, testing dan *maintenance*. Disebut dengan waterfall karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan (Watung & Sinsuw, 2014). Adapun tahapan-tahapan tersebut dapat dilihat sebagai berikut

Implementationtion

Design

Analysis

Testing

Gambar 2.3 Model *waterfall*

1. *Analysis*

Tahap ini penulis melakukan analisa terhadap sistem yang akan dibangun berupa kebutuhan *software*, *hardware* dan melakukan pengumpulan data-data. Proses pengumpulan data di lakukaan dengan melakukan *observasi*, wawancara, dokumentasi, dan studi pustaka. Data yang penulis kumpulkan untuk pembuatan sistem adalah berupa data dokumen alumni serta data mengenai Program Studi Pendidikan Teknologi Informasi FKIP UMK.

1. *Desain*

Pada proses ini penulis menerjemahkan kebutuhan ke sebuah perancangan *software* yang dapat diperkirakan sebelum dibuat *coding*. Penulis merancang desain sistem nya seperti menjadi *Use Case Diagram*, *Activity Diagram*, *Swimlane Diagram, Mockup,* dan *Sequence* *Diagram* untuk menentukan perangkat keras (*hardware*) dan sistem persyaratan serta dalam mendefinisikan arsitektur sistem secara keseluruhan sehingga akan memberi kemudahan dalam membuat sistem.

1. *Implementation/coding*

Untuk dapat dimengerti oleh mesin, dalam hal ini adalah komputer, maka desain sebelumnya harus diubah menjadi bentuk yang dapat dimengerti oleh mesin, yaitu ke dalam bahasa pemrograman melalui proses *coding*. Tahap ini merupakan implementasi dari tahap *design* yang secara teknis nantinya dikerjakan oleh *programmer* (Watung & Sinsuw, 2014).

1. *testing*

Pada proses ini penulis menguji kemampuan dan keefektifan pada sistem yang telah dibuat. Semua fungsi-fungsi *software* harus diujicobakan, agar *software* bebas dari kesalahan (*error*). Dilakukan hal tersebut guna mengetahui kekurangan dan kelemahan pada aplikasi ini sehingga dapat dilakukan perbaikan sebelum dapat di implementasikan nantinya.

1. ***UML***

Menurut Sugiarti (2013:30) “*UML* adalah salah satu bentuk *langguage* atau bahasa pencetusnya, *UML* di definisikan sebagai bahasa visual untuk menjelaskan memberikan spesifikasi, merancang, membuat model dan mendokumentasikan aspek-aspek dari sebuah *system*”. *UML* mengilustrasikan prosedur perincian yang dipakai untuk menciptakan serta mengarsipkan *software. UML* menggambarkan salah satu alat dan metodologi yang mendukung untuk mengembangkan sistem berorientasi objek. . *UML* menyediakan serangkaian gambar dan diagram yang sangat baik. Beberapa diagram memfokuskan diri pada ketangguhan teori object oriented dan sebagian lagi memfokuskan pada detail rancangan dan konstruksi. Semua dimaksudkan sebagai sarana komunikasi antar tim pengembang maupun dengan pengguna (Rosa dan Shalahuddin, 2018)

1. ***Use* *Case Diagram***

*Use case* diagram merupakan kumpulan *actor* dan *use case* yang digunakan untuk mempresentasikan fungsionalitas aplikasi. *use case* yaitu teknik yang digunakan dalam pengembangan sebuah software atau sistem informasi untuk menangkap kebutuhan fungsional dari sistem yang bersangkutan. *Actor* adalah orang, proses, atau sistem lain yang berinteraksi dengan sistem yang akan dibuat. *Use case* diagram berfungsi untuk menggambarkan fasilitas menu yang diberikan kepada *user*. *User* adalah orang yang berhak memakai sebuah aplikasi. Pada sistem di dalam gambar *use case, use case* telah menjelaskan mengenai sistem, lingkungan pada sistem, dan hubungan pada sistem lingkungannya.

*Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut.

Simbol-simbol yang digunakan dalam *use case diagram* yaitu:

Table 2.1 Simbol *use case diagram*

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Gambar** | **Nama** | **Keterangan** |
| 1. |  | *Actor* | Menspesifikasikan himpunan kepada pemakai untuk memainkan pada saat interaksi  dengan use case |
| 2. |  | *Dependency* | Hubungan disaat perubahan yang terjadi kepada suatu elemen mandiri dan akan menjadi masalah di saat elemen yang bergantung pada elemen yang tidak mandiri  (*independent*) |
| 3. |  | *Generalization* | Hubungan yang dimana letak antara objek kecil atau disebut anak yang berbagi perilaku dan struktur data dari objek yang  ada diatasnya objek induk |
| 4. |  | *Include* | Memspesifikasi bahwa *use case* diagram sumber secara eksplisit |
| 5. |  | *Exclude* | Memspesifikasi bahwa *use case* target memperlebar perilaku dari use case sumber kepada  sebuah titik yang diberi. |
| 6. |  | *Association* | Apa yang menghubungkan  antara objek satu Dengan objek lainnya. |

1. ***Activity* *Diagram***

Aktivitas merupakan kumpulan aksi- aksi. Aksi-aksi melakukan langkah sekali saja tidak boleh dipecah menjadi beberapa langkah lagi. *Activity* *diagram* menggambarkan rangkaian aliran dari aktivitas, digunakan untuk mendeskripsikan aktivitas yang dibentuk dalam satu operasi sehingga dapat juga untuk aktivitas lainnya. *Activity* diagram memperlihatkan serangkaian aktivitas yang terhubung dan melibatkan beberapa objek dalam sistem. Setiap aktivitas ditunjukkan sebagai sebuah node yang terhubung dengan node lainnya menggunakan garis. Node-node ini menunjukkan tindakan-tindakan yang diambil, seperti proses bisnis, operasi sistem, atau interaksi dengan objek lain.

Simbol-simbol yang digunakan dalam *activity diagram* yaitu:

Table 2.2 Simbol *activity diagram*

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Gambar** | **Nama** | **Keterangan** |
| 1. |  | *Start point* | Menggambarkan awal dari aktivitas |
| 2. |  | *End point* | Menggambarkan akhir dari aktivitas |
| 3. |  | *Activity* | Menggambarkan suatu proses atau kegiatan sistem |
| 4. |  | *Fork* | Menggambarkan aktivitas yang dimulai dengan sebuah aktivitas dan diikuti oleh dua atau lebih aktivitas yang harus dikerjakan. |
| 5. |  | *Decision* | Menggambarkan eputusan atau pilihan |
| 6. |  | *State Transition* | Menggambarkan aliran perpindahan control antara state. |
| 7. |  | *Swimlane* | Menggambarkan pemisahan atau pengelompokkan aktivitas berdasarkan *actor*. |

1. ***Swim Lane* *Diagram***

*Swim Lane Diagram* adalah elemen visual yang digunakan dalam diagram alir proses yang menggambarkan siapa bekerja pada subset tertentu dari sebuah proses. Jenis diagram yang digunakan untuk menggambarkan proses bisnis atau operasional secara visual. Diagram ini dapat membantu pengguna untuk memahami proses yang ada di dalam lembaga dan interaksi antara departemen atau individu yang terlibat dalam proses tersebut.

1. ***Sitemap* *Diagram***

*Sitemap* adalah sebuah diagram herarki yang menunjukkan informasi arsitektur dan hubungan antara halaman dari situs web atau aplikasi. Sitemap dihasilkan pada fase define (memahami kebutuhan pengguna), setelah membuat user persona. Sitemap memiliki beberapa kegunaan yaitu diantarnya sebagai struktur navigasi, membantu untuk mengindentifikasi dimana konten akan ditempatkan, menentukan taksonomi (pengelompokan informasi dari suatu web/app) dan menunjukkan hubungan antara halaman yang berbeda (Amini et al., 2021)

Menurut Hakim (2009) dalam penelitian yang dilakukan oleh (Pratiwi et al., n.d.) beberapa hal yang perlu diperhatikan dalam membuat *sitemap* yaitu :

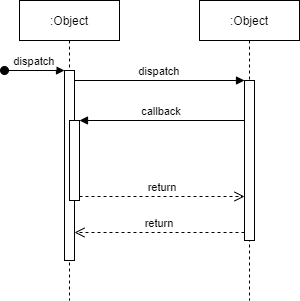
1. Tujuan Navigasi Sitemap secara harfiah bertindak sebagai peta dari sebuah situs, diterima ketika pengunjung menulusuri dan mendapatkan link jika hilang diantara ribuan halaman di sebuah situs maka dapat dengan mudah melihat pada sitemap.
2. Menyampaikan Tema Situs Sitemap harus dapat memberikan suatu gambaran umum daripada sebuah situs.
3. Optimasi Tujuan Pembuatan Situs
4. Organisasi dan Relevans
5. ***Sequence* *Diagram***

Sequence diagram merupakan diagram yang menjelaskan interaksi objek berdasarkan urutan waktu. Sequence diagram menggambarkan urutan atau tahapan yang harus dilakukan pengguna yang menjelaskan interaksi objek yang disusun dalam suatu urutan waktu. Diagram ini secara Khusus berasosiasi dengan use case. Sequence Diagram memperlihatkan tahap demi tahap apa yang sebenarnya terjadi untuk menghasilkan sesuatu didalam use case. *Sequence* diagram biasa digunakan untuk menggambarkan skenario atau rangkaian langkah – langkah yang dilakukan sebagai respons dari sebuah event untuk menghasilkan output tertentu.

Simbol-simbol yang digunakan dalam *sequens diagram* yaitu:

Table 2.3 Simbol *sequens diagram*

|  |  |  |
| --- | --- | --- |
| **No** | **Simbol** | **Keterangan** |
| 1 | Aktor | 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 aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya  dinyatakan menggunakan kata benda di awal frase nama aktor. |
| 2 | Garis Hidup/ *Lifeline* | Menyatakan kehidupan suatu objek. |
| 3 | Objek | Menyatakan objek yang berinteraksi pesan. |
| 4 | Waktu Aktif | Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya. |



Gambar 2.4 *Sequence* Diagram

1. **Penelitian relevan**

Penelitian yang relevan merupakan acuan bagi peneliti dalam membuat penelitian. Tujuan dari penelitian yang relevan untuk mencari perbandingan dari penelitian sebelumnya dengan penelitian yang akan dilakukan agar kedepannya tidak terjadi kesamaan (*plagiarisme*). Berikut ini merupakan beberapa penelitian yang dilakukan oleh peneliti terdahulu yang dapat dijadikan sebagai landasan atau gambaran untuk penelitian yang sekarang sedang dilakukan.

1. Sistem informasi alumni program studi sistem informasi UIN Sultan Syarif Kasim Riau, penelitian dilakukan oleh (Robbani, 2019), merupakan mahasiswa Program Studi Sistem Informasi, Fakultas Sains dan Teknologi UIN Suska Riau. Sistem alumni ini dapat membantu pihak Program Studi melakukan pendataan alumni dan memberi informasi kegiatan serta lowongan pekerjaan pada Program studi Sistem Informasi. Sistem ini menggunakan menggunakan *PHP* versi 7, *sublime teks editor* dan *Paket Software Xampp Control Panel.*
2. Sistem informasi alumni program studi informatika universitas teknologi sumbawa berbasis web, penelitian dilakukan oleh (Rodianto et al., 2020), merupakan mahasiswa Teknik Informatika, Universitas Teknologi Sumbawa. Sistem Informasi Alumni ini berbasis web menggunakan bahasa pemrograman *PHP* dengan *framework Codeigniter* dan *Database MySql* dan telah diuji secara fungsional bahwa sistem dapat digunakan untuk mengelola data alumni diantaranya, mampu menyimpan data-data identitas alumni, memetakan data alumni, mencetak data alumni, dan mengelola informasi seputar alumni.
3. Perancangan sistem informasi pendataan alumni fakultas ekonomi universitas methodist indonesia berbasib web, penelitian dilakukan oleh (Sebayang et al., 2018), merupakan mahasiswa D-III Komputerisasi Akuntansi Universitas Methodist Indonesia. Sistem alumni yang dibagun dapat membantu pihak universitas untuk mengolahan dan mengkoordinir alumninya, dapat pula melakukan registrasi secara efisien dan mudah untuk mendaftar. Meningkatkan efektivitas dan produktivitas alumni dan pihak program studi dalam mengelolah informasi. Sistem ini memiliki beberapa menu diantaranya admin, kegiatan, daftar alumni, forum *question*, berita, info loker, forum answer, pengumuman dan buku tamu.

Berdasarkan dari ketiga hasil penelitian tersebut, perbedaan yang akan dilakukan oleh penulis dengan judul “Perancangan *sistem* informasi alumni Pendidikan Teknologi Informasi Fakultas Keguruan dan Ilmu Pendidikan Universitas Muhammadiyah Kendari” ialah sebagai berikut.

1. Penelitian pertama oleh (Robbani, 2019) terdapat perbedaan yaitu pada *form* galeri dan *form* prestasi, sistem yang akan penulis buat *form* galeri dan *form* prestasi akan berada dalam satu form yang sama yaitu *profile* alumni serta pada sistem yang dibuat menggunakan *PHP* versi 8, *visual studio code teks editor* dan *Paket Software Laragon Control Panel.*
2. Penelitian kedua oleh (Rodianto et al., 2020) sistem yang akan penulis buat memiliki forum komunikasi, berita terkini program studi dan info lowongan pekerjaan serta menggunakan *framework laravel.*
3. Penelitian terakhir oleh (Sebayang et al., 2018) terdapat perbedaan yaitu sistem yang akan penulis buat tidak memiliki menu buku tamu.

**BAB III**

**METODOLOGI PENELITIAN**

1. **Jenis penelitian**

Peneliti menggunakan metode *Waterfall* untuk pengembangan sistem. Metode *waterfall* salah satu model proses pengembangan perangkat lunak atau *software* *development life cycle (SDLC)* yang mengikuti tahapan-tahapan secara linear dan berurutan. Adapun tahapan-tahapan model *waterfall* dapat dilihat pada gambar 2.3

1. **Prosedur penelitian**

Untuk memudahkan peneliti dan meningkatkan pemahaman tentang kebutuhan dalam melakukan penelitian maka sesuai dengan gambar 2.3 model *waterfall*, maka prosedur dari perancangan sistem ini sebagai berikut:

1. Analysis
2. Kebutuhan *software* dan *hadware* sistem

Untuk merancang sistem penulis membutuhkan *sorfware* diantaranya *Visual studio code, laragon, codegniter, bootstrap, web browser, draw io dan xampp.* Kemudian *Hadware* yaitu laptop *Asus.*

1. Dokumen alumni Program Studi Pendidikan Teknologi Informasi FKIP UMK.

Table 2.4 Data alumni Program Studi Pendidikan Teknologi Informasi

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Stambuk** | **Nama** | Tgl Lulus | Prodi |
| 1 | 21816001 | Harun Rasid | 14/11/2022 | PTI |
| 2 | 21816012 | Yusril | 14/11/2022 | PTI |
| 3 | 21816035 | Waode Sri Wulandari | 14/11/2022 | PTI |
| 4 | 21816056 | Tri Wahyudi | 14/11/2022 | PTI |
| 5 | 21816049 | Fitrah Warahma | 14/11/2022 | PTI |
| 6 | 21816031 | Nurjiani | 28/11/2022 | PTI |
| 7 | 21816033 | Sitti Aisa Anini | 28/11/2022 | PTI |

1. Data program studi Pendidikan Teknologi Informasi FKIP UMK.

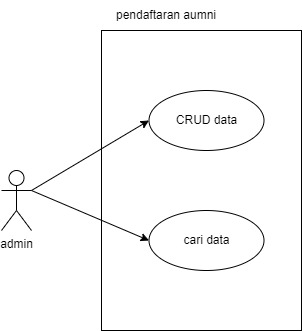


Gambar 3.1 *Profile* Prodi PTI FKIP UMK

1. Desain Sistem

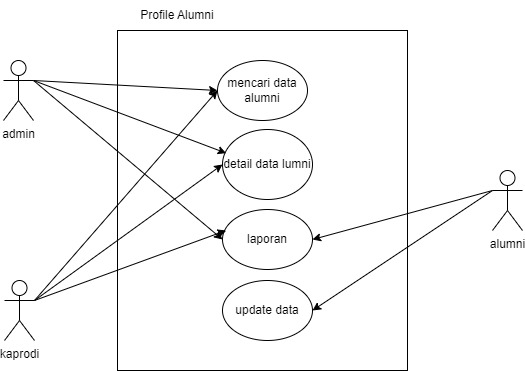
Desain sistem dapat didefinisikan sebagai penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi. Pada proses desain ini sistem dilakukan perancangan sesuai spesifikasi kebutuhan sebelumnya. Desain sistem meliputi *Unified Modeling Language (UML)* diantaranya *use case diagram, activity diagram, swim lane dan sequens diagram,* kemudianbasis data(*database*), dan *mockup*.

* + 1. *Unified Modeling Language (UML)*
       - 1. *Uce* *case diagram*
* *Use case diagram menu* pendaftaran alumni



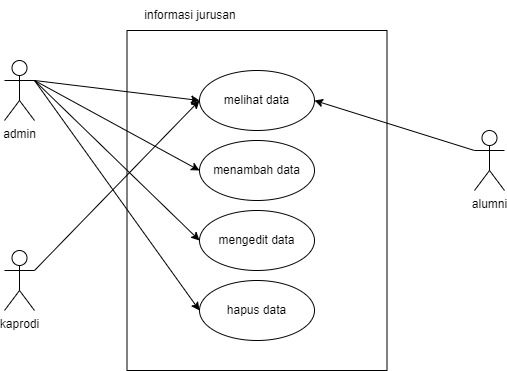
Gambar 3.2 *use case diagram* menu pendaftaran alumni

* use case diagram menu profile alumni



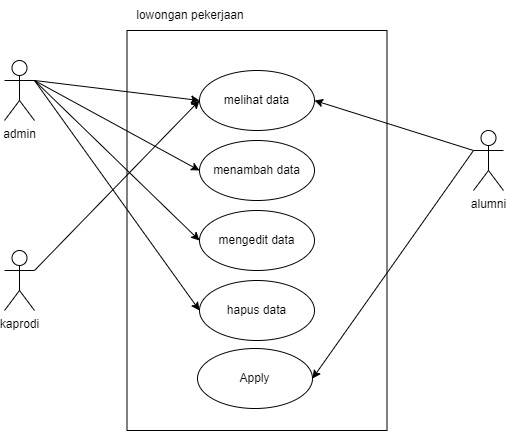
Gambar 3.3 *use case diagram* menu profile alumni

* *use case* menu informasi jurusan



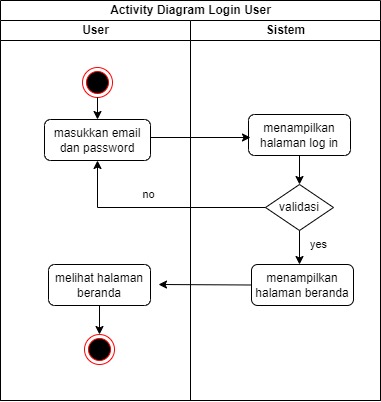
Gambar 3.4 *use case diagram* menu informasi jurusan

* use case diagram menu lowongan kerja



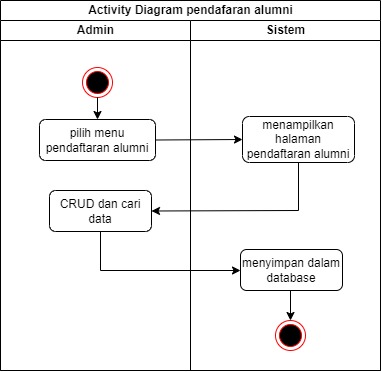
Gambar 3.5 *use case diagram* menu lowongan kerja

* + - * 1. *Activity* *diagram*
* *Activity diagram login user*

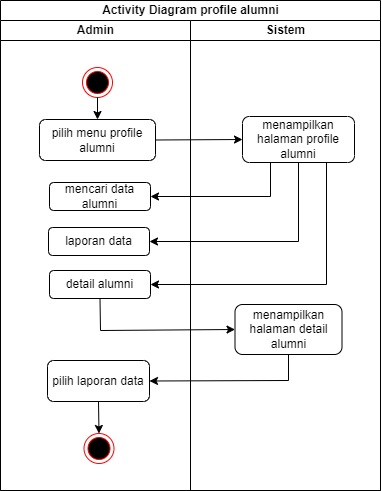


Gambar 3.6 *activity diagram login user*

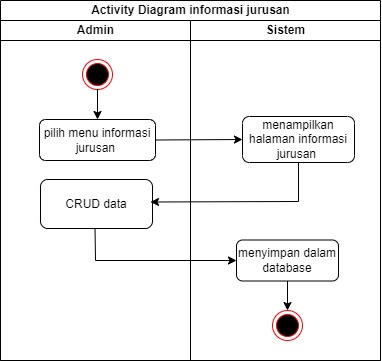
* *Activity diagram admin*



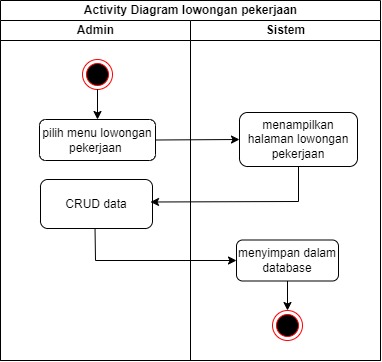
Gambar 3.7 *activity diagram admin menu* pendaftaran alumni



Gambar 3.8 *activity diagram admin menu profile* alumni

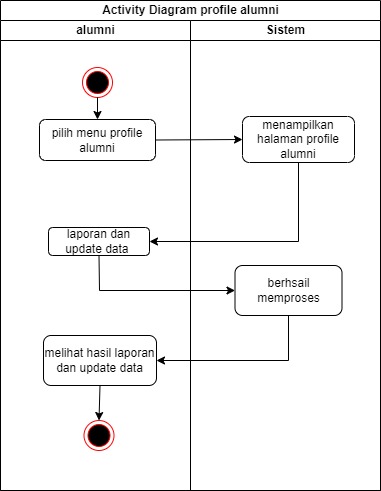


Gambar 3.9 *activity diagram admin menu* informasi jurusan



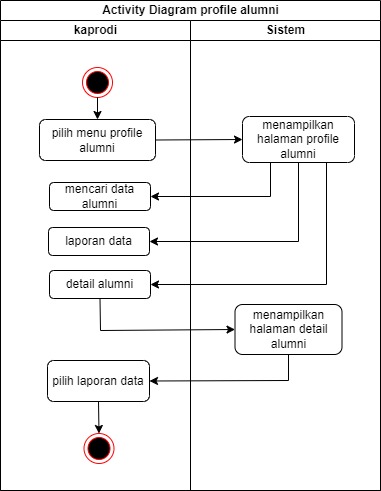
Gambar 3.10 *activity diagram admin* lowongan pekerjaan

* *Activity diagram* alumni



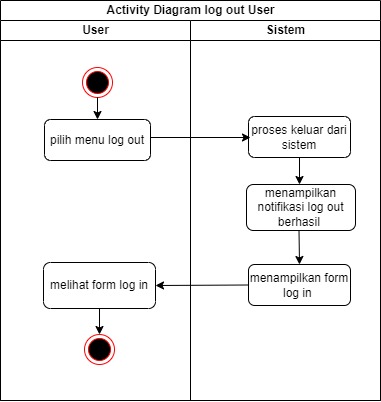
Gambar 3.11 *acitivity diagram* alumni menu *profile*

* *Activity diagram* Kaprodi



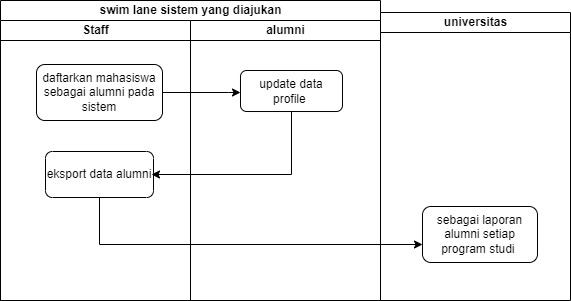
Gambar 3.12 *activity diagram* Kaprodi *menu profile* alumni

* *Activity diagram log out* *user*



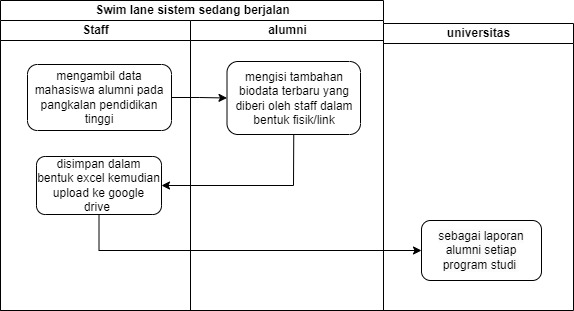
Gambar 3.13 *activity diagram log out user*

* + - * 1. *Swim* *lane diagram*
* *Swim lane sistem* yang diajukan



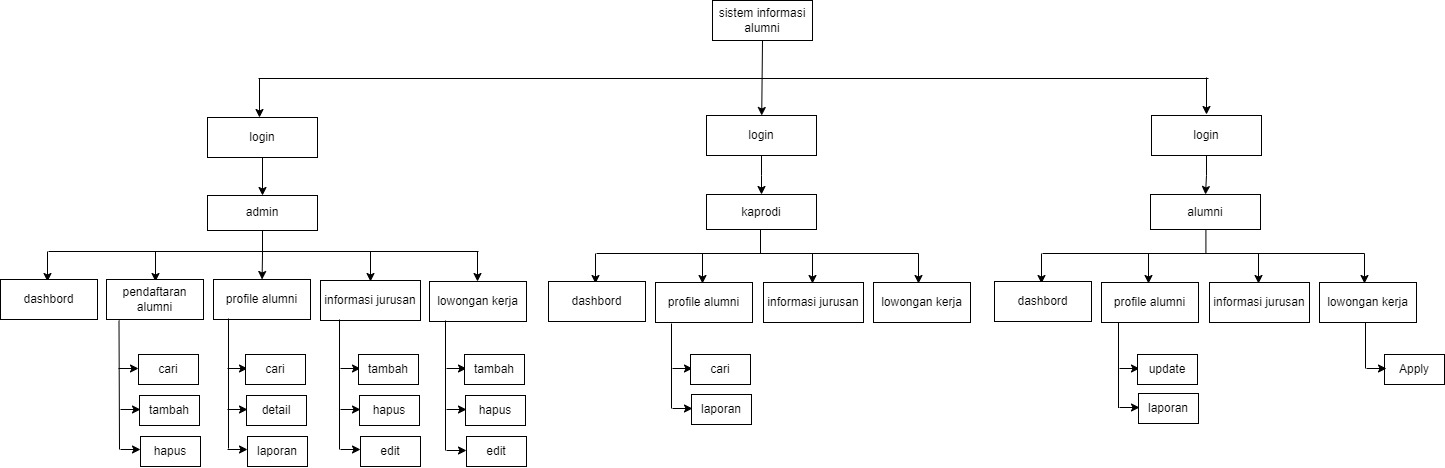
Gambar 3.14 *swim lane sistem* sedang berjalan

* *Swim lane sistem* sedang berjalan



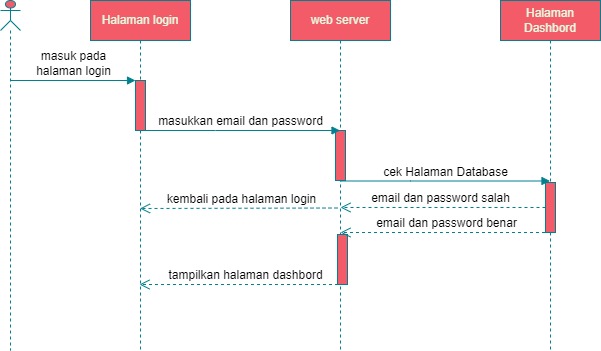
Gambar 3.15 *swim lane sistem* yang diajukan

* + - * 1. *Sitemap* *diagram*



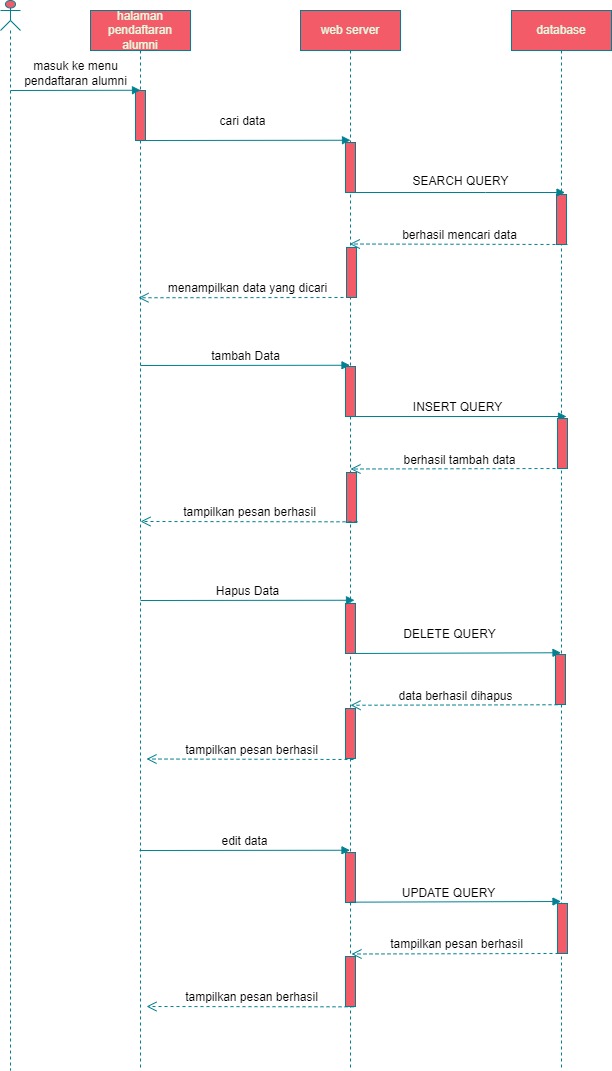
Gambar 3.16 *Sitemap*

* + - * 1. *Sequence* *diagram*
* *Sequens diagram login user*

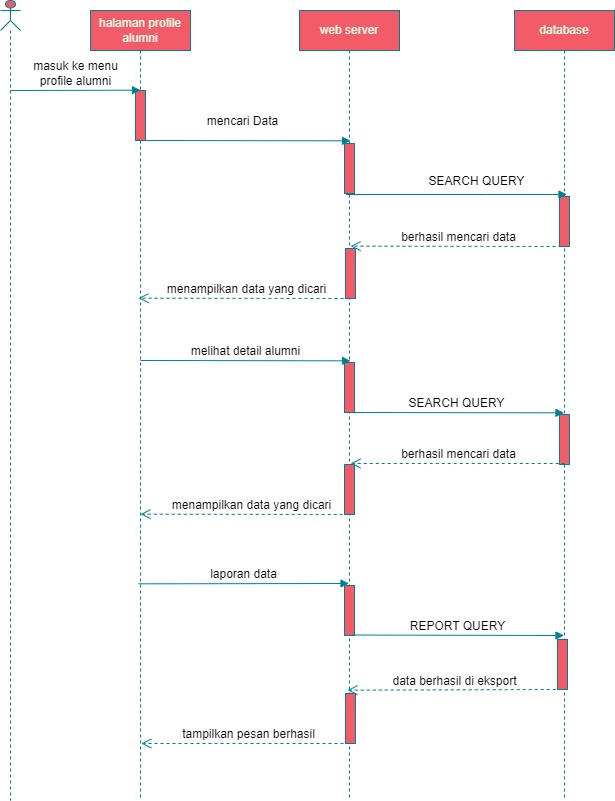


Gambar 3.17 *sequens diagram login user*

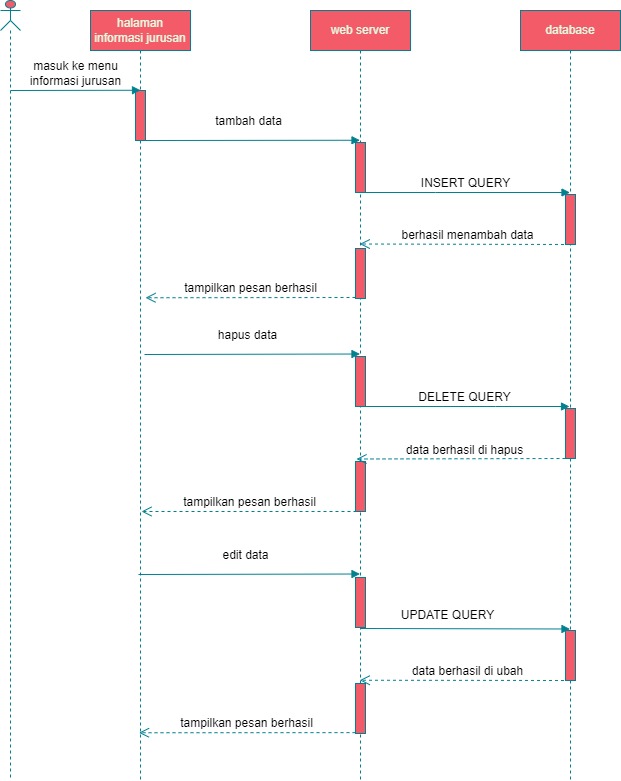
* *Sequens diagram admin*



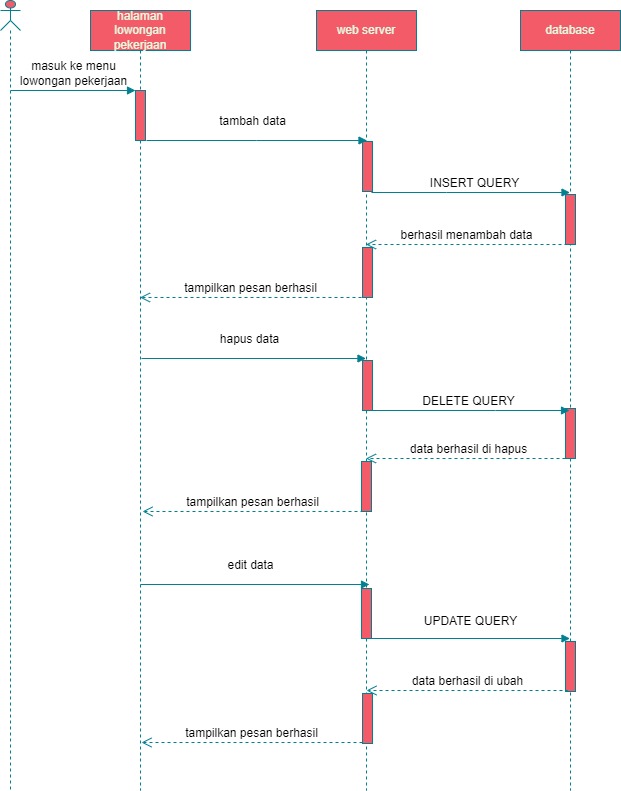
Gambar 3.18 *sequens diagram admin menu profile* alumni



Gambar 3.19 *sequens diagram admin menu* profile

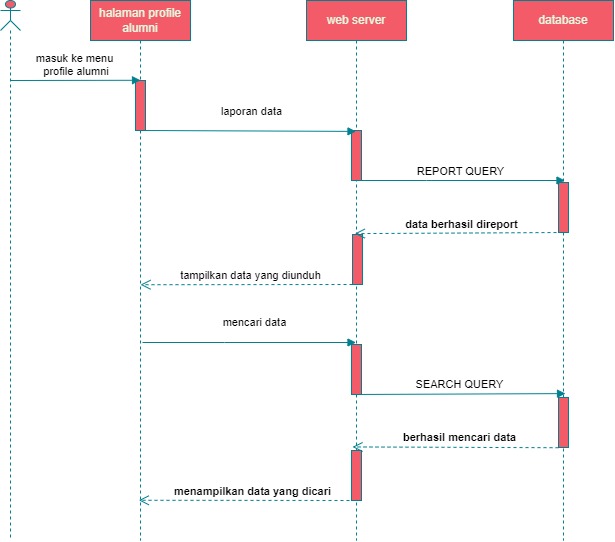


Gambar 3.20 *sequens diagram admin menu* informasi jurusan



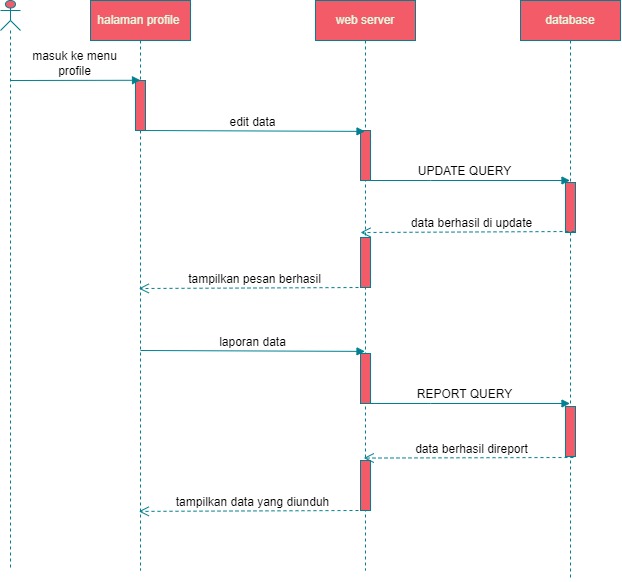
Gambar 3.21 *sequens diagram admin menu* lowongan kerja

* *Sequens diagram* Kaprodi



Gambar 3.22 *sequens diagram* Kaprodi *menu profile* alumni

* *Sequens diagram* alumni



Gambar 3.23 *sequens diagram* alumni *menu profile*

* + 1. Desain basis data (*database*)

Pada tahap ini digunakan untuk menentukan isi dan pengaturan data yang dibutuhkan sehingga mendukung berbagai rancangan sistem.

Berikut tabel *database* yang akan di gunakan untuk membangun Sistem Informasi ini.

Table 3.1 *user*

|  |  |  |
| --- | --- | --- |
| Nama | Tipe data | Keterangan |
| Id | Int(11) |  |
| nama | Varchar(50) |  |
| Email | Varchar(50) |  |
| Password | Varchar(100) |  |
| Role | Enum (Kaprodi, staff, Alumni) |  |
| Status | Enum (aktif, non aktif,) |  |

Tabel 3.2 data\_*admin*

|  |  |  |
| --- | --- | --- |
| Nama | Tipe data | Keterangan |
| Id | Int(11) | *Primary Key, Auto Increment* |
| User\_id | Int(11) |  |
| Nama\_admin | Varchar(50) |  |

Tabel 3.3 data\_kaprodi

|  |  |  |
| --- | --- | --- |
| Nama | Tipe data | Keterangan |
| Id | Int(11) | *Primary Key, Auto Increment* |
| User\_id | Int(11) |  |
| Nama\_kaprodi | Varchar(200) |  |
| NIDN | Int(10) |  |
| Jabatan | Varchar(100) |  |

Tabel 3.4 data\_alumni

|  |  |  |
| --- | --- | --- |
| Nama | Tipe data | Keterangan |
| Id | Int(11) | *Primary Key, Auto Increment* |
| User\_id | Int(11) |  |
| Nama | Varchar(50) |  |
| NIM | Int(10) |  |
| Nomor\_alumni | Varchar(25) |  |
| Jenis\_kelamin | Enum(‘perempuan’, ‘laki-laki’) |  |
| Angkatan\_alumni | Enum(‘2018’, ‘2019’, ‘2020’, 2021’, ’2022’, ’2023’, ’2024’) |  |
| Program\_studi | Varchar(50) |  |
| Agama | Varchar(20) |  |
| Tempat\_lahir | Varchar(50) |  |
| Tanggal\_lahir | date |  |
| Alamat | Varchar(200) |  |
| Password | Varchar(50) |  |
| Email | Varhar(30) |  |
| Telp | Int(20) |  |
| Tahun\_masuk | Enum(‘2018’, ‘ 2019, ‘ 2020’, ‘2021’, ‘2022’, ‘2023’, ‘2024’, ‘2025’) |  |
| Tahun\_tamat | Enum(‘2023’, ‘ 2024’, ‘ 2025’, ‘2026’) |  |
| IPK | Int(10) |  |
| Foto | Text |  |
| JSTA | Text |  |
| S2 | Int(50) |  |
| Pekerjaan | Varchar(50) |  |
| Tempat\_pekerjaan | Varchar(100) |  |
| Pengalaman\_kerja | Teks |  |

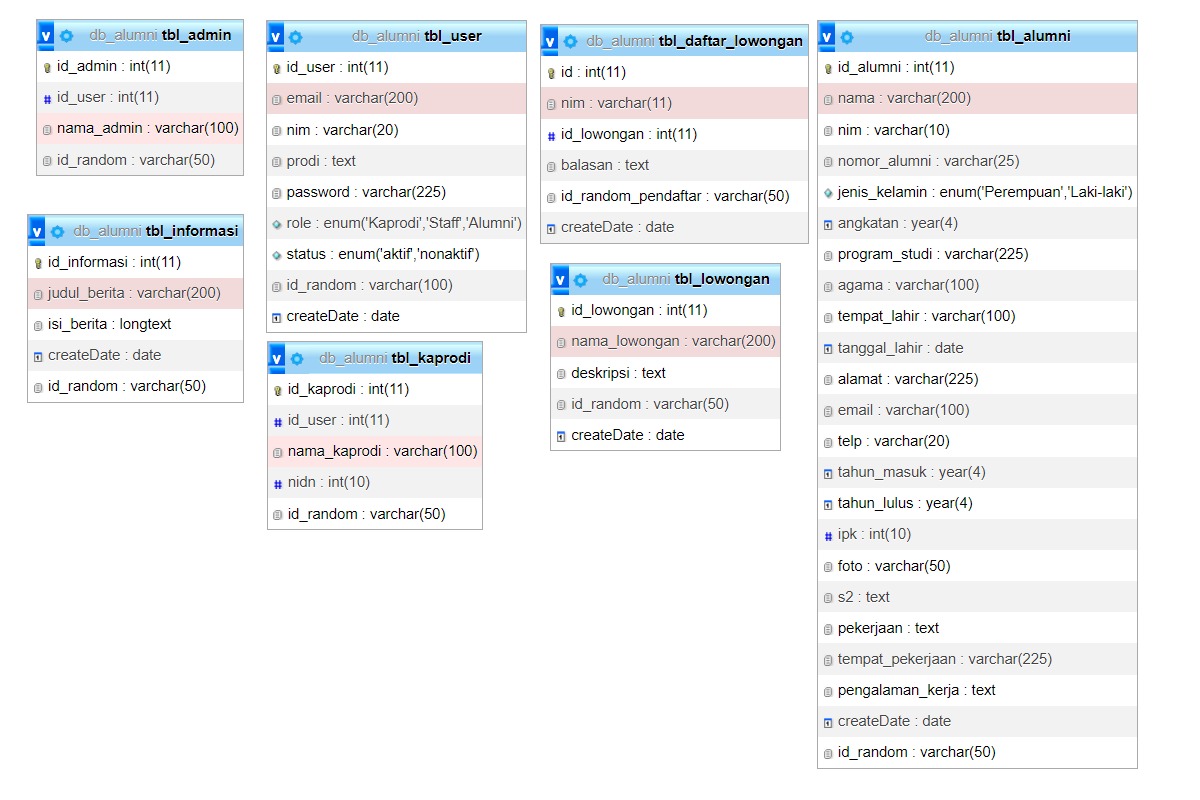
Tabel 3.5 informasi\_jurusan

|  |  |  |
| --- | --- | --- |
| Nama | Tipe data | Keterangan |
| Judul\_berita | Varchar(500) |  |
| Isi\_berita | text |  |

Tabel 3.6 lowongan\_pekerjaan

|  |  |  |
| --- | --- | --- |
| Nama | Tipe data | Keterangan |
| Judul\_loker | Varchar(500) |  |
| Deskripsi | text |  |
| requirement | text |  |

Perancangan database di perlukan sebuah ER Diagram dalam perancangannya untuk menghubungkan antara beberapa tabel, berikut ini Gambar 3.23 ER Diagram pada dalam sistem ini :



Gambar 3.24 ER Diagram

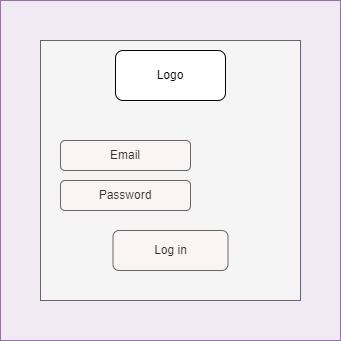
* + 1. *Mock Up*

*Mock Up* adalah sebuah media visual atau preview dari sebuah konsep desain “datar” yang diberikan efek visual sehingga hasilnya sangat menyerupai wujud nyata, *mockUp* dapat memberikan gambaran nyata dari sebuah konsep desain bagaimana konsep itu akan terlihat nantinya jika sudah diaplikasikan menjadi atau kedalam benda nyata. *MockUp* juga dapat menambahkan nilai visual dari sebuah konsep desain (Rizkita et al., 2018)**.** *Mockup* dapat mempermudah *stakeholder* karena menggunakan gambaran *high-fidelity,* yaitu rencana desain yang sudah menampilkan informasi secara rinci dan menunjukkan dengan jelas bagaimana sebuah situs atau aplikasi akan dibuat. Oleh karena itu, *mockup* adalah cara sederhana agar *stakeholder* dapat dengan mudah melihat dan memahami produk final yang akan dibuat. *Stakeholder* juga bisa memberikan masukan dan kritik mengenai desain tersebut sebelum melaju ke proses selanjutnya.

Berikut ini merupakan desain *Mockup* :

* Tampilan *Mockup log in*

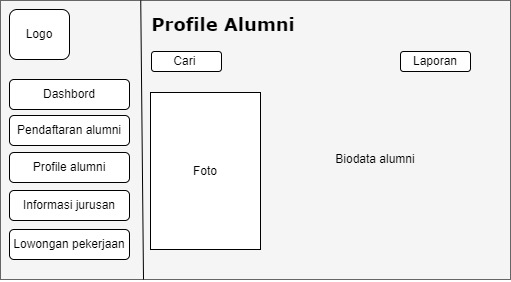
Tampilan ini merupakan halaman fungsi *log in* yang sama untuk semua *user* yaitu *admin*, kapodi dan alumni, dengan memasukkan *email* dan *password* yang telah disediakan oleh *admin*.



Gambar 3.25 *mockup login user*

* Tampilan *mockup admin menu profile* alumni

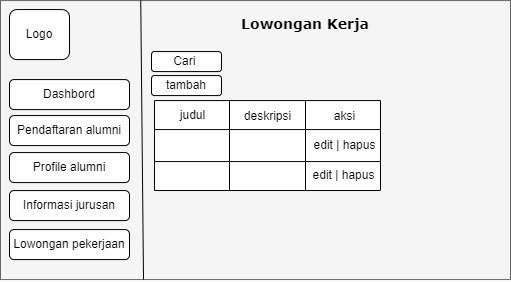
Tampilan *mockup admin* yaitu *staff* *program* *studi* pada *menu* *profie* ini digunakan untuk melihat dan mendapatkan *file* data alumni terbaru yang telah dimasukkan pada menu pendaftaran alumni dan hasil *update* alumni dengan menggunakan fungsi laporan.



Gambar 3.26 *mockup admin menu profile* alumni

* Tampilan *mockup* *Admin* *menu* lowongan kerja

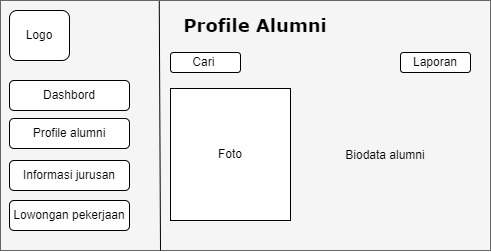
Tampilan *menu* lowongan kerja ini digunakan untuk mengelola informasi lowongan kerja dengan menambah, edit dan hapus informasi lowongan kerja. pada halaman ini juga terdapat informasi dari alumni yang *apply* lowongan kerja.



Gambar 3.27 *mockup admin menu* lowongan kerja

* Tampilan *mockup* Kaprodi *menu profile* alumni

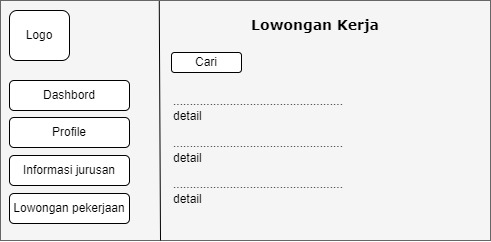
Tampilan *mockup* Kaprodi yaitu ketua program studi pada menu *profie* ini digunakan untuk melihat dan mendapatkan *file* data alumni terbaru yang telah dimasukkan oleh *admin* dan hasil *update* alumni dengan menggunakan fungsi laporan.



Gambar 3.28 *mockup* Kaprodi *menu profile* alumni

* Tampilan *mockup* Kaprodi *menu* lowongan kerja

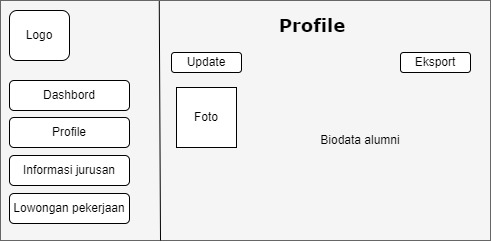
Tampilan *mockup* Kaprodi pada menu lowongan kerja menampilan informasi lowongan kerja yang telah dimasukkan oleh admin.



Gambar 3.29 *Mockup* Kaprodi *menu* lowongan kerja

* Tampilan *mockup* Alumni *menu profile*

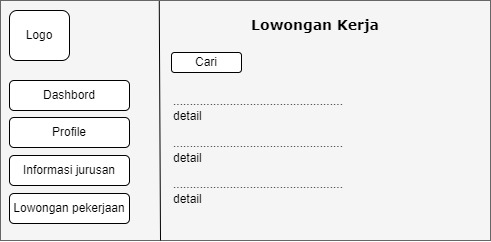
Tampilan *mockup* alumni menampilan data alumni yang telah di masukkan oleh admin kemudian dapat di *update*. pada *menu* ini alumni juga dapat mengunduh data *profile* mereka dengan menggunakan fungsi *eksport*.



Gambar 3.30 *Mockup* alumni menu *profile*

* Tampilan *mockup* Alumni *menu* lowongan kerja

Tampilan *mockup* alumni pada menu lowongan kerja menampilan informasi lowongan kerja serta alumni dapat *apply* lowongan kerja yang ada. Ketika alumni *apply* lowongan kerja maka admin akan menerima informasi tersebut sehingga admin atau staff Prodi dapat membantu dan mempermudah alumni dalam proses pekerjaan.



Gambar 3.31 *Mockup* alumni menu lowongan kerja

1. Implementation / penulisan kode

Penulisan kode program merupakan tahap penulis memasukan script dan kode pemrograman ke dalam sebuah software untuk menghasilkan aplikasi yang akan dibuat (Riyadh Arridha et al., 2020). Sistem ini menggunakan bahasa pemrograman *PHP* yang disusun melalui *framework* *Laravel* dan Boostrap lalu dihubungkan kedalam *server* My*SQL* sebagai penyimpanan data. *Software* yang digunakan yaitu *visual studio code* dan *laragon*.

Berikut *scipt* dasar untuk menampilkan “hello word”

<?php

echo "Hello World!";

?>

1. *Testing* / pengujian sistem

*Testing* merupakan proses pengujian atau verifikasi perangkat lunak untuk memastikan sistem berfungsi dengan baik dan memenuhi persyaratan bisnis atau teknik yang ditentukan sebelumnya.Untuk metode pengujian yang dilakukan pada penelitian ini menggunakan pengujian *Black Box*. Dengan pengujian *Black Box* akan diketahui kesalahan maupun kekurangan pada saat melakukan pengujian rancangan sehingga dapat diperbaiki sistem yang belum berfungsi dengan benar.

Berikut kisi-kisi uji *black box* dalam pengujian sistem berdasarkan perancangannya

Table 3.7 kisi-kisi uji *black box*

|  |  |  |  |
| --- | --- | --- | --- |
| **Activitas** | **Action** | **Expected Result** | **Keterangan** |
| Mengisi email dan password | Klik login | Sistem akan menerima masuk pada halaman dashboard dan menampilkan menampilkan pesan “berhasil ” . | Berhasil |
| Melihat menu pendaftaran alumni | Klik menu pendaftaran alumni | Sistem akan menerima dan menampilkan halaman pendaftaran alumni | Berhasil |
| Melihat menu profile alumni | Klik menu profile alumni | Sistem akan menerima dan menampilkan halaman profile alumni | Berhasil |
| Melihat menu informasi jurusan | Klik menu informasi jurusan | Sistem akan menerima dan menampilkan halaman informasi jurusan | Berhasil |
| Melihat menu lowongan pekerjaan | Klik menu lowongan pekerjaan | Sistem akan menerima dan menampilkan halaman lowongan pekerjaan | Berhasil |
| Keluar dari sistem | Klik menu logout | Sistem akan menerima keluar dari sistem dan menampilkan pesan “berhasil”. | Berhasil |

1. **Tempat dan waktu penelitian**

Penelitian ini dilaksanakan di Program Studi Pendidikan Teknologi Informasi Fakultas Keguruan dan Ilmu pendidikan Universitas Muhammadiyah Kendari pada bulan Maret sampai Mei 2023.

Adapun rencana kerja yang Penulis buat untuk pembuatan pembuatan Sistem Informasi Alumni ini ditampilkan pada Tabel berikut :

Tabel 3.8 rencana kerja sistem informasi alumni

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Fase | Uraian Kegiatan | Maret | | | | April | | | |
| M1 | M2 | M3 | M4 | M1 | M2 | M3 | M4 |
| Analysis | Wawancara sistem berjalan |  |  |  |  |  |  |  |  |
| Perencanaan sistem yang diajukan |  |  |  |  |  |  |  |  |
| Desain | Perancangan *UML* |  |  |  |  |  |  |  |  |
| Perancangan tabel |  |  |  |  |  |  |  |  |
| Perancangan *mockup* |  |  |  |  |  |  |  |  |
| Implementation | *Coding* |  |  |  |  |  |  |  |  |
| Testing | Proses *testing* |  |  |  |  |  |  |  |  |

1. **Alat dan bahan penelitian**
2. Alat Penelitian

Penulis dalam perancangan sistem membutuhkan beberapa alat diantaranya perangkat keras (*hadware*) dan perangkat lunak (*sorfware*).

Peralatan *hardware* maupun *software* dipilih disesuaikan dengan kebutuhan sebagai berikut:

* *Hadware*

Laptop *Asus*, sistem *model* : *VivoBook* 14\_ASUS Laptop X441MA\_X441MA, *Memory* : 4,00 GB , *Processor* : *Intel*(R) *Celeron*(R) N4000 *CPU* @ 1.*10GHz* 1.10 *GHz*, sistem operasi : *Microsoft Windows* 10 *Home Single Language* 64 *bit*.

* *Sorftware*

|  |  |  |
| --- | --- | --- |
| *Laragon* | : | Mengakses MySQL dan Apace dalam pengelolaan database |
| *Visual studio code* | : | *Teks editor* untuk menulis kode program |
| *Codegniter* | : | *Framework* atau kerangka kerja dalam membangun *back end* *website* |
| *Bootstrap* | : | *Framework* atau kerangka kerja dalam membangun tampilan (*front end*) *website* |
| *Web browser* | : | *Browser* yang digunakan yaitu *google chrome* untuk menampilkan hasil rancang sistem |
| *Draw io* | : | Aplikasi untuk membuat desain perancangan *website*. |

*Xampp*  : *Sublime text*

1. Bahan Penelitian

Bahan yang digunakan dalam penelitian ini yaitu :

* Wawancara

Penulis melakukan wawancara secara langsung pada ketua Program Studi Pendidikan Teknologi Informasi FKIP UMK mengenai sistem informasi alumni yang sedang berjalan.

* Dokumentasi

Dokumentasi merupakan tahapan untuk mengumpulkan data dengan menggunakan sumber-sumber tertulis seperti mengenai data dari profile Program Studi dan data alumni.

* Studi Literatur

Studi literatur dalam penelitian ini yaitu mengumpulkan data jurnal, buku, skripsi yang relevan yang menunjang penulis membangun sistem informasi alumni lebih baik kedepannya.

1. **Analisis dan Perancangan Perangkat Lunak**
2. Identifikasi masalah

Permasalahan yang terjadi di Program Studi Pendidikan Teknologi Informasi yaitu sistem penginputan data alumni masih dilakukan secara konvensional mengguakan *excel* dan *google drive*. Pihak Program Studi akan terkendala dalam mendapatkan data dan informasi terbaru dari alumni itu sendiri karena harus melakukan proses pengisian data secara konvensional seperti sebelumnya, sehingga kurang mendapatkan informasi secara cepat dan tepat serta memperlambat proses kerja Program Studi dalam mengolah informasi.

1. *Analysis* sistem
2. Analysis sistem berjalan

*Analysis* sistem sedang berjalan adalah suatu metode untuk mempelajari dan memahami sistem yang sedang berjalan dalam situasi nyata. Metode ini dilakukan dengan mengumpulkan data dan informasi tentang sistem melalui wawancara, observasi, dan pengumpulan dokumen. Analisis ini dilakukan agar memahami bagaimana sebuah sistem bekerja, mengidentifikasi masalah atau kelemahan yang ada, dan menemukan cara untuk meningkatkan kinerjanya.

Pada tahap ini merupakan proses bagaimana sistem berjalan secara manual saat ini.

* Program studi mengambil data mahasiswa alumni di Pangkalan Data Pendidikan Tinggi
* disimpan dalam bentuk *excel* ke dalam *google drive*
* menghubungi masing-masing alumni untuk mengambil data tambahan dan terbaru dari mahasiswa alumni diantarannya tempat tanggal lahir, alamat, agama, pendidikan (apabila menlanjutkan pendidikan S2), pekerjaan, tempat bekerja dan nomor handphone.
* menyimpan semua data dalam satu file *excel* secara konvensional kemudian di upload ke *google drive*.
* memberi data alumni file excel tersebut kepada Universitas ketika diminta data alumni setiap tahunnya.

1. *Analysis* sistem ajuan

Berdasarkan analisa sistem yang sedang berjalan secara manual sehingga belum bisa memenuhi kebutuhan dalam pengolahan data secara cepat dan efektif. Maka langkah selanjutnya adalah merancang sistem informasi alumni yang bertujuan untuk memenuhi kebutuhan Program Studi Pendidikan Teknologi Informasi dalam melakukan dalam proses management data.

Berikut adalah prosedur sistem informasi alumni yang diajukan pada Program Studi Pendidikan Teknologi Informasi.

* Administrator log in ke sistem
* Kemudian administrator daftrakan mahsiswa yang telah terdaftar sebagai alumni
* Setelah terdaftar kemudian alumni log in ke sistem menggunakan email dan password yang diberikan oleh administrator
* Dalam sisem alumni dapat update profile terbaru diantaranya riwayat pendidikan (apabila lanjut *study*), pekerjaan dan data lainnya pada profile. kemudian dapat melakukan proses komunikasi dengan admin atau alumni lain serta dapat melihat informasi terbaru jurusan hingga melihat lowongan pekerjaan.
* Ketika permintaan dari Universitas mengenai data alumni program studi maka administrator dengan menggunakan fitur eksport akan mengahsilkan file data alumni program studi PTI dalam bentuk excel.

1. Kategori level /actor pengguna
2. Staff Program Studi Pendidikan Teknologi Informasi
3. Ketua Program Studi Pendidikan Teknologi Informasi
4. Alumni Program Studi Pendidikan Teknologi Informasi
5. Kebutuhan data

Data yang dibutuhkan dalam merancang Sistem Informasi Alumni Program Studi Pendidikan Teknologi Informasi FKIP UMK ialah :

1. Data mahasiswa alumni di Program Studi Pendidikan Teknologi Informasi.
2. Data profile Program Studi Pendidikan Teknologi Informasi FKIP UMK .

**DAFTAR PUSTAKA**

Agus Efendi, U. (2020). Sistem Informasi Pkpri ( Pusat Koperasi Pegawai Republik Indonesia ). *Jurnal Insand Comtech*, *5*(1), 21–39.

Amini, T. N. A., Fabroyir, H., & Akbar, R. J. (2021). Desain dan Evaluasi Antarmuka Mobile App MyITS Alumni pada Platform Android dan Ios Melalui Pendekatan User-Centered Design. *Jurnal Teknik ITS*, *10*(2), 133–139. https://doi.org/10.12962/j23373539.v10i2.63024

Bachtiar. (2010). *Rancang Bangun Sistem Pendukung Keputusan ( Spk ) Penjurusan Program Studi Rancang Bangun Sistem Pendukung Keputusan ( Spk ) Penjurusan*.

Banyal, N. A., Angriani, L., & Surianti, S. (2021). Aplikasi Rancang Bangun Sistem Informasi Alumni Perguruan Tinggi Di Kota Jayapura Berbasis Web (Studi Kasus Stmik Umel Mandiri). *Jurnal Ilmiah Matrik*, *23*(1), 94–99. https://doi.org/10.33557/jurnalmatrik.v23i1.1215

Bin Tahir, T., Rais, M., & Hs, M. A. (2019). Aplikasi Point OF Sales Menggunakan Framework Laravel Point OF Sales Appilaction using Laravel Framework. *Jurnal Informatika Dan Komputer) p-ISSN*, *2*(2), 2355–7699. http://dx.doi.org/10.33387/jiko

Binarso Yusi Ardi, Sarwoko Eka Adi, B. N. (2012). Pembangunan Sistem Informasi Alumni Berbasis Web Pada Program Studi Teknik Informatika Universitas Diponogoro. *Journal of Informatics and Technology*, *1*(1), 72–84. https://ejournal3.undip.ac.id/index.php/joint/article/view/434

Budiman, M. A., & Atmaja Darmawan, I. D. M. B. (2020). Pengembangan Sistem Informasi Alumni Program Studi Teknik Informatika Fakultas Mipa Universitas Udayana. *JELIKU (Jurnal Elektronik Ilmu Komputer Udayana)*, *9*(1), 91. https://doi.org/10.24843/jlk.2020.v09.i01.p09

Endra, R. Y., & Aprilita, D. S. (2018). E-Report Berbasis Web Menggunakan Metode Model View Controller Untuk Mengetahui Peningkatan Perkembangan Prestasi Anak Didik. *Explore: Jurnal Sistem Informasi Dan Telematika*, *9*(1). https://doi.org/10.36448/jsit.v9i1.1028

Fachreza, A., Sultonuddin, R., & Pratama, F. I. (2021). Rancang Bangun Sistem E-Learning untuk Sarana Belajar pada Wahidev.Com. *Jurnal Informatika Dan Rekayasa Perangkat Lunak*, *3*(1), 74. https://doi.org/10.36499/jinrpl.v3i1.4040

Mediana, D., & Nurhidayat, A. I. (2018). Rancang Bangun Aplikasi Helpdesk (A-Desk) Berbasis Web Menggunakan Framework Laravel ( Studi Kasus di PDAM Surya Sembada Kota Surabaya ). *Jurnal Manajemen Informatika*, *8*(2), 75–81. http://ejournal.ukrida.ac.id/ojs/index.php/TIK/article/view/1495/1617

Moshinsky, M. (1959). No Titleیلیب. *Nucl. Phys.*, *13*(1), 104–116.

Novendri. (2019). Pengertian Web. *Lentera Dumai*, *10*(2), 46–57.

Novi Yona Sidratul Munti, & Dwi Asril Syaifuddin. (2020). Analisa Dampak Perkembangan Teknologi Informasi Dan Komunikasi Dalam Bidang Pendidikan. *Jurnal Pendidikan Tambusai*, *4*(2), 1799–1805. https://jptam.org/index.php/jptam/article/view/655

Prabowo, D. (2015). WEBSITE E-COMMERCE MENGGUNAKAN MODEL VIEW CONTROLLER ( MVC ) DENGAN FRAMEWORK CODEIGNITER Studi Kasus : Toko Miniatur. *Jurnal Ilmiah DASI*, *16*(1), 23–29.

Pratiwi, P. I. H., Kom, S., Pd, M., Pembimbing, M. T. I., Pajar, I. I. H., Kom, M., Studi, P., Informasi, S., Widya, S., Dharma, C., Yamin, J. L. M., & Samarinda, N. (n.d.). *SISTEM INFORMASI TRACER STUDY ALUMNI PADA MTs NEGERI MODEL SAMARINDA BERBASIS WEB DAN SMS GATEWAY Muhammad Yovi Iqbal*.

Riyadh Arridha, Titing Magfirah, Andi Roy, Abd Ganiyu M. Natsir, & Risna Zullidya Nengsih. (2020). Rancang Bangun Sistem Informasi Data Alumni Pada Politeknik Negeri Fakfak. *Jurnal Informasi, Sains Dan Teknologi*, *3*(2), 35–44. https://doi.org/10.55606/isaintek.v3i2.34

Rizkita, N., Rosely, E., & Nugroho, H. (2018). Aplikasi Pendaftaran dan Transaksi Pasien Klinik Hewan di Bandung Berbasis Web. *EProceedings of Applied Science*, *4*(3), 1512–1520.

Robbani, M. A. (2019). *Sistem Informasi Alumni Program Studi Sistem*. *5*(2), 153–160.

Rodianto, Abduh Robbani, M., & Dery Sofya, N. (2020). Sistem Informasi Alumni Program Studi Informatika Universitas Teknologi Sumbawa Berbasis Web. *Jurnal Informatika, Teknologi Dan Sains*, *2*(1), 59–68. https://doi.org/10.51401/jinteks.v2i1.559

Sanjaya, R., & Hesinto, S. (2018). Rancang Bangun Website Profil Hotel Agung Prabumulih Menggunakan Framework Bootstrap. *Jurnal Teknologi Dan Informasi*, *7*(2), 57–64. https://doi.org/10.34010/jati.v7i2.758

Saputra, D., & Ari Waluyo. (2020). Perancangan Sistem Informasi Alumni Berbasis Website dengan Menggunakan PHP MYSQL di Politeknik Dharma Patria Kebumen. *Jurnal E-Komtek (Elektro-Komputer-Teknik)*, *4*(2), 191–199. https://doi.org/10.37339/e-komtek.v4i2.406

Sebayang, R., Hutapea, M. I., & Simamora, R. J. (2018). Perancangan Sistem Informasi Pendataan Alumni Fakultas Ekonomi Universitas Methodist Indonesia Berbasis Web. *Jurnal Managemen Informatika & Komputerisasi Akutansi*, *2*(1), 1–11.

Watung, I. A., & Sinsuw, A. A. E. (2014). Perancangan Sistem Informasi Data Alumni Fakultas Teknik Unsrat Berbasis Web. *Jurnal Teknik Informatika*, *3*(1). https://doi.org/10.35793/jti.3.1.2014.3922