**RANCANG BANGUN SISTEM PEMINJAMAN SARANA DAN**

**PRASARANA BERBASIS *WEBSITE* DI YAYASAN**

**PENDIDIKAN AN-NADHIR LOSARANG**

**TUGAS AKHIR**

Sebagai salah satu syarat untuk menyelesaikan

Program Diploma III Teknik Informatika

Politeknik Negeri Indramayu

**Oleh:**

**ALYAA ADELYA**

**NIM.1903033**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**JURUSAN TEKNIK INFORMATIKA  
POLITEKNIK NEGERI INDRAMAYU**

**FEBRUARI 2022**

**HALAMAN PENGESAHAN**

Tugas Akhir ini diajukan oleh :

Nama : ALYAA ADELYA

NIM : 1903033

Program Studi : Diploma III Teknik Informatika

Judul : Rancang Bangun Sistem Peminjaman Sarana dan Prasarana

Berbasis *Website* Di Yayasan Pendidikan An-Nadhir Losarang

Pembimbing : I. Eka Ismantohadi, S.Kom., M.Eng

NIP. 198107092021211005 ………………..

: II. Kurnia Adi Cahyanto, M.Kom ………………..

NIP. 198503022018031001

Telah berhasil dipertahankan dihadapan dewan penguji pada tanggal 2022 dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Ahli Madya Program Studi Diploma III Teknik Informatika, Jurusan Teknik Informatika, Politeknik Negeri Indramayu.

DEWAN PENGUJI

Ketua Penguji :

Anggota :

Penguji I

Anggota :

Penguji II

|  |
| --- |
| Indramayu, Agustus 2022  Ketua Jurusan Teknik Informatika |
|  |
| Iryanto, S.Si.,M.Si NIP. 199008012019031014 |

**PERNYATAAN**

Dengan ini saya menyatakan dengan sesungguhnya bahwa Tugas Akhir ini adalah asli hasil karya saya sendi serta **Tugas Akhir** ini tidak terdapat karya yang pernah diajukan untuk memperoleh gelar **Ahli Madya** di suatu Perguruan Tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dirujuk dalam naskah ini dan disebutkan dalam daftar pustaka.

|  |
| --- |
| Indramayu, Agustus 2022  Yang menyatakan |
| Materai 10000 |
| **Alyaa Adelya**  NIM. 1903033 |

**ABSTRAK**

Peminjaman sarana dan prasarana biasanya dilakukan secara manual dengan mengisi kertas form sebagai bukti bahwa sarana atau prasarana sedang dipinjam. Begitupun dengan pendataan sarana prasarana tersebut yang seringkali tidak tercatat dibuku pendataan atau bisa saja data hilang tersebut hilang. Demikian juga dengan sebuah yayasan pendidikan di Losarang Indramayu. Oleh karena itu dibutuhkan sebuah sistem untuk memudahkan proses peminjaman sarana prasarana. Sistem yang dapat menyimpan data dengan efektif. Serta sistem bisa diakses dimanapun dan kapanpun.

***ABSTRACT***

**KATA PENGANTAR**

*Alhamdulillah*, puji dan syukur kepada Allah SWT yang telah melimpahkan Rahmat dan Karunia-Nya, sehingga saya selaku penulis dapat menyelesaikan penulisan serta pembuatan Tugas Akhir ini dengan baik. Shalawat beserta salam semoga senantiasa tercurahkan kepada Rasul kita Nabi Muhammad SAW beserta keluarga, sahabat dan umatnya hingga akhir zaman.

Tugas Akhir merupakan salah satu tugas wajib yang harus dikerjakan oleh mahasiswa Diploma III khususnya yang menempuh pendidikan di Politeknik Negeri Indramayu sebagai persyaratan utama untuk bisa dinyatakan lulus sebagai Ahli Madya Diploma III.

Penulis menyadari bahwa Tugas Akhir ini dapat selesai dengan lancar dan tepat waktu berkat bantuan, arahan, kerjasama serta dukungan dari berbagai pihak, baik secara langsung maupun tidak langsung. Oleh karena itu penulis menyampaikan banyak terima kasih kepada :

1. Allah SWT yang telah melimpahkan segala nikmat dan karunia-Nya, serta memberikan kemudahan bagi penulis dalam melaksanakan pembuatan Tugas Akhir ini.
2. Kedua orang tua yang senantiasa mendo’akan, memberikan dukungan, semangat, motivasi serta memberikan restunya kepada penulis dalam pembuatan Tugas Akhir ini.
3. Bapak Casiman Sukardi, S.T., M.T selaku Direktur Politeknik Negeri Indramayu.
4. Bapak Iryanto, S.Si., M.Si selaku Ketua Jurusan Teknik Informatika Politeknik Negeri Indramayu.
5. Bapak Eka Ismantohadi, S.Kom., M.Eng selaku pembimbing utama
6. Bapak Kurnia Adi Cahyanto, S.T., M.Kom., selaku pembimbing pendamping.
7. Seluruh teman Teknik Informatika angkatan 2019 dan semua pihak yang tidak dapat penulis sebutkan sati persatu yang telah banyak membantu dalam pembuatan Tugas Akhir ini.

Penulis sudah berusaha untuk membuat laporan Tugas Akhir ini dengan sebaik mungkin, namun penulis sadar bahwa tidak ada kesempurnaan di dunia ini karena kesempurnaan hanya milik Allah, oleh karena itu kritik dan saran yang membangun sangat penulis harapkan. Akhir kata, semoga laporan Tugas Akhir ini bisa bermanfaat bagi almamater dan teman-teman mahasiswa lainnya.

*Wassalamu’alaikum Warahmatullahi Wabarakatuh.*

Indramayu, Agustus 2022

Penulis

**DAFTAR ISI**

**DAFTAR GAMBAR**

**DAFTAR TABEL**

**DAFTAR LAMPIRAN**

**BAB I**

**PENDAHULUAN**

1. **Latar Belakang**

Yayasan Pendidikan An-Nadhir adalah sebuah lembaga pendidikan pengajaran yang memadukan ilmu umum dan ilmu terapan yang berlandaskan ilmu agama islam yang yang berfaham *Ahlusunnah Wal Jama’ah ala Nahdlatul Ulama*. Pendidikan di Yayasan An-Nadhir mengkolaborasikan media fikir dan zikir, IPTEK dan IMTAK, intelektual dan moral serta teori dan aplikasi dengan tetap dalam bingkai pendidikan dengan nilai-nilai islami, moralitas dan nasionalisme.

Pada Yayasan Pendidikan An-Nadhir terdapat beberapa jenjang diantaranya adalah SMP (Sekolah Menengah Pertama) dan SMK (Sekolah Menengah Kejuruan). Karena terdapat beberapa jenjang di lingkungan yang sama, Yayasan Pendidikan An-Nadhir memiliki banyak fasilitas penunjang kegiatan seperti laboratorium, perlengkapan presentasi, perlengkapan olahraga dan lain-lain. Kelengkapan dan ketersediaan peralatan yang baik menjadi aset bagi sebuah yayasan pendidikan untuk dapat menjalankan kegiatan pembelajaran dan praktikum siswa dengan optimal.

Data sarana dan prasarana yang terdapat pada Yayasan Pendidikan An-Nadhir tersimpan secara tidak rapih dalam bentuk dokumen *softfile* yang berbeda-beda, seperti *file* pdf ataupun *file* excel. Kendala yang dihadapi dalam pendataan sarana dan prasarana adalah data-data yang tercecer sehingga bisa menghasilkan informasi yang tidak valid terkait ketersediaan sarana dan prasarana atau bahkan data tersebut bisa saja hilang. Pada proses peminjaman sarana dan prasarana masih menggunakan cara konvensional seperti, siswa-siswi yang hendak meminjam perlu mengisi *form* peminjaman yang masih berbentuk selembar kertas. Penggunaan *form* yang masih menggunakan media kertas memiliki kelemahan yaitu mudah robek, basah, ataupun hilang.

Berdasarkan permasalahan tersebut, diperlukan sebuah sistem yang menyediakan proses pengelolaan data dan peminjaman sarana dan prasarana. Dengan tujuan dapat menyimpan data-data sarana prasarana dan data peminjaman sarana prasarana dengan akurat Berdasarkan analisis kebutuhan sistem, maka penulis memilih membangun sebuah sistem peminjaman sarana dan prasrana berbasis *website.*

1. **Rumusan Masalah**

Berdasarkan latar belakang yang diuraikan, maka diperoleh suatu rumusan permasalahan yang menjadi dasar pembuatan sistem tersebut, yakni sebagai berikut:

1. Bagaimana rancangan sistem peminjaman sarana dan prasarana berbasis *website* pada Yayasan Pendidikan An-Nadhir?
2. Bagaimana melakukan kegiatan peminjaman sarana dan prasarana pada Yayasan Pendidikan An-Nadhir?
3. Bagaimana melakukan pendataan peminjaman sarana dan prasarana pada Yayasan Pendidikan An-Nadhir?
4. **Batasan Masalah**

Untuk memfokuskan pembahasan, dapat diperoleh beberapa batasan masalah, di antaranya:

1. Sistem peminjaman ini dibuat untuk Yayasan Pendidikan An-Nadhir.
2. *Platform* yang digunakan pada sistem peminjaman berbasis *website* dengan menggunakan bahasa pemprograman PHP dan *database* MySQL.
3. Sistem ini menangani kegiatan peminjaman dan pendataan sarana prasarana pada Yayasan Pendidikan An-Nadhir.
4. Pengguna yang berinteraksi dalam sistem ini adalah admin dan peminjam.
5. **Tujuan**

Tujuan dari pembuatan sistem yang dibuat meliputi:

1. Merancang dan membangun sistem peminjaman yang dapat memberikan informasi terkait ketersedian sarana dan prasarana di Yayasan Pendidikan An-Nadhir.
2. Membuat kegiatan peminjaman sarana dan prasarana di Yayasan Pendidikan An-Nadhir.
3. Membuat pencatatan dan pendataan peminjaman sarana dan prasarana di Yayasan Pendidikan An-Nadhir.
4. **Manfaat**

Manfaat yang dapat diperoleh dengan dibuatnya sistem ini adalah :

1. Sistem ini dapat memudahkan pihak yayasan melakukan pendataan dan pencatatan peminjaman sarana dan prasarana.
2. Sistem ini mempermudah masyarakat yayasan untuk melakukan kegiatan peminjaman sarana dan prasarana.
3. Pembuatan aplikasi ini dapat menambah wawasan bagi penulis sekaligus menambah ilmu pengetahuan dan kemampuan berfikir secara detail.
4. **Sistematika Penulisan**

Dalam sistematika penulisan laporan tugas akhir ini dibagi menjadi beberapa bagian bab yang di mana setiap bagian memiliki pembahasan yang berbeda-beda tetapi saling terkait antara satu dengan lainnya. Adapun urutan penulisan laporan ini sebagai berikut:

**BAB I PENDAHULUAN**

Bab pendahuluan berisi latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, sistematika penulisan laporan. Sehingga permasalahan tersebut memiliki titik fokus.

**BAB II LANDASAN TEORI**

Bab ini menjelaskan tentang landasan teori atau teori-teori yang disusun berdasarkan sumber referensi yang valid berkaitan dengan masalah yang diteliti. Sumber yang dijadikan sebagai referensi teori-teori dasar berasal dari buku, jurnal penelitian ataupun website resmi.

**BAB III METODE PELAKSANAAN**

Bab metodologi pelaksanaan menjelaskan tentang tahapan serta metode pelaksanaan yang digunakan dalam pembuatan tugas akhir Rancang Bangun Sistem Peminjaman Sarana Dan Prasarana Berbasis *Website* Di Yayasan An-Nadhir. Pada bagian ini dibuat diagram alir (*flowchart*), rancangan antarmuka (*interface*), rancangan *database* (ERD) dan rancangan sistem berupa UML (*Unified Modeling Language*) untuk menjelaskan tahapan alur proses dan sistem yang akan dibuat.

**BAB IV HASIL DAN PEMBAHASAN**

Pada bab ini membahas hasil-hasil dari pembuatan Sistem Peminjaman Sarana dan Prasarana Berbasis *Webiste* Di Yayasan An-Nadhir berdasarkan perancangan yang telah dipaparkan, mulai dari tahap analisis, desain, implementasi desain, implementasi *database*, *black box,* hasil pengujian, dan pengujian kuisoner.

**BAB V PENUTUP**

Bab ini berisi tentang kesimpulan dan saran yang didapatkan dari hasil tugas akhir atau pembuatan aplikasi.

**BAB II**

**LANDASAN TEORI**

1. **Rancang Bangun**

Rancang bangun adalah penggambaran, perencanaan, dan, pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam suatu kesatuan yang ituh dan berfungsi. (Fauzi, 2018)

1. **Sistem**

Sistem dapat diartikan sebagai sebuah tatanan yang terdiri atas sejumlah komponen fungsional (dengan tugas / fungsi khusus) yang saling berhubungan dan secara bersama-sama bertujuan untuk memenuhi suatu proses atau pekerjaan tertentu. Sistem juga merupakan kumpulan elemen-elemen yang saling terkait dan bekerja sama untuk memproses masukan (*input*) yang dilanjutkan kepada mengolah masukan tersebut sampai menghasilkan keluaran (*output*) yang diinginkan.(Sinaga, 2020)

1. **Peminjaman**

Peminjaman adalah sebuah kegiatan yang dilakukan sseorang dengan memberikan barang, uang, dan tempat untuk dipakai oleh orang lain atau organisasi tertentu dengan jaminan harus mengembalikan sesuai dengan kesepakatan durasi waktu awal serta sesuatu yang dipinjam tersebut harus dkembali seperti awal dipinjam baik berupa jumlah dan kondisinya. Jika jumlah dan kondisi barang yang telah dipinjam mengalami perubahan maka peminjam harus bertanggung jawab atas hal tersebut. (Estiningrat, 2022)

1. **Sarana**

Menurut KBBI Edisi V Tahun 2016, sarana adalah segala sesuatu yang dapat dipakai sebagai alat dalam mencapai maksud atau tujuan; alat; media.

1. **Website**

*Website* adalah sistem dengan informasi yang disajikan dalam bentu teks, gambar, suara, dan lainnya yang tersimpan dalam sebuah *server web* internet yang disajikan dalam bentuk *hypertext*. (Listyanto, 2019)

1. **Laravel**

Laravel adalah sebuah *framework* *web* berbasis PHP yang *open source* dan tidak berbayar, diciptakan oleh Taylor Otwell dan diperuntukkan untuk pengembangan aplikasi *web* yang menggunakan pola MVC Struktur pola MVC pada laravel sedikit berbeda pada struktur pola MVC pada umumnya. Dalam laravel terdapat *routing* yang menjembatani antara *request* dari *user* dan *controller*. Jadi *controller* tidak langsung menerima *request* tersebut. (Sari, 2020)

1. **PHP (*Hypertext Preprocessor)***

PHP (Hypertext Preprocessor) yaitu bahasa pemrograman web *server-side* yang bersifat *open source*. PHP juga merupakan salah satu bahasa pemrograman *open source* yang dapat digunakan pada berbagai sistem operasi seperti Linux, Unix, macintosh, maupun Windows. Padadasarnya PHP dirancang untuk pembuatan jenis web dinamis, dinamis berarti halaman website yang ditampilkan dibuat saat halaman itu diminta oleh client, mekanisme ini menyebabkan informasi yang diterima client selalu yang terbaru. Salah satu kelebihan lainyang dimiliki PHP antara lain dapat terhubung pada beberapa database salah satunya MySql. (Yusuf, 2017)

1. **Mysql**

MySQL adalah sebuah implementasi dari sistem manajemen basis data relasional yang didistribusikan secara gratis. Setiap pengguna dapat secara bebas menggunakan MySQL, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam basisdata yang telah ada sebelumnya SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian basisdata, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. (Lestanti, 2016)

1. **XAMPP**

XAMPP merupakan software web server yang berguna dalam pengembangan website yang berguna dalam pengembangan website yang didalamnya sudah tersedia database server MySQL dan dapat mendukung pemograman PHP, XAMPP merupakan software gratis, dapat dijalankan di sistem operasi Windows, Linux maupun Mac OS. Softwre XAMPP bisa didapatkan gratis pada website apachefriends sesuai dengan operating system yang digunakan. (Prsetia, 2019)

1. **Figma**

Figma adalah salah satu perangkat lunak alat desain yang memiliki kelebihan *web based*. Jika dilihat dari kegunaan aplikasi ini, dapat didefinisikan bahwa figma adalah suatu alat untuk membantu pembuatan desain grafis dan prototyping. Figma memiliki banyak keunggulan yaitu, berbasis *web* yang artinya dapat diakses oleh perangkat apapun yang dapat diakses oleh perangkat apapun yang dapat menjalankan *web browser*. Melihat dari *website* resmi figma yaitu figma.com, selain berbasis *web* figma tersedia dengan dua pilihan gratis dan berbayar serta fitur-fitur yang lengkap. Figma mempunyai fitur selain untuk membuat sebuah UI *design* dan *prototyping* namun juga dapat melakukan *collaborate team* dan *commenting.* Berikut ini adalah kelebihan figma adalah sebagai berikut: (Pradita, 2021)

1. Figma gratis untuk beberapa orang, namun untuk menggunakan bersama dengan tim ada biaya $12 per bulan per *editor.*
2. Siapa pun dapat membuka *file* figma tanpa perlu melakukan instalasi. Mereka pun bisa langsung meninjau dan menulis komentar secara gratis.
3. Kecepatan dan performa bisa jauh lebih cepat saat mengedit dan melihat *file*.
4. *Platform* bersifat agnostik, sehingga dapat dijalankan di sistem operasi dan *browser* apapun.
5. Kolaborasi bisa berjalan secara *real-time*, demikian juga ketika melakukan pembaruan *file*.

Logo Figmadapat dilihat pada Gambar 2.2.

1. ***Flowchart***

*Flowchart* atau sering disebut dengan diagram alir merupakan suatu jenis diagram yang merepresentasikan algoritma atau langkah-langkah instruksi yang berurutan dalam sistem.seorang analis sistem menggunakan *flowchart* sebagai bukti dokumentasi untuk menjelaskan gambaran logis sebuah sistem yang akan dibangun kepada *programmer*. Dengan begitu, *flowchart* dapat membantu untuk memberikan solusi terhadap masalah yang bisa saja terjadidalam membangun sistem. Pada dasarnya, *flowchart* digambarkan dengan menggunakan simbol-simbol. Setiap simbol mewakili suatu proses tertentu. Sedangkan untuk menghubungkan satu proses ke proses selanjutnya digambarkan dengan menggunakan garis penghubung. (Rosaly, 2019)

Dalam pembuatan sebuah *flowchart* digunakan berbagai macam simbol untuk menggambarkan alur sebuah program. Simbol-simbol beserta penjelasan yang digunakan dalam pembuatan *flowchart* sebuah program dapat dilihat pada Tabel 2.1.

Tabel 2.1 Tabel Simbol Flowchart

|  |  |  |
| --- | --- | --- |
| **Simbol** | **Nama** | **Keterangan** |
|  | *Process* | Merupakan proses perhitungan / proses pemberian harga awal. |
|  | *Decision* | Menunjukkan perbandingan pernyataan, penyeleksian data yang memberikan piihan untuk langkah selanjutnya. |
|  | *Input / Output Data* | Menunjukkan proses *input* / *output* data, parameter, dan informasi. |
|  | *Predefined Process (Sub Program)* | Menunjukkan permulaan sub program / proses menjalankan sub program. |
|  | *Terminator* | Menunjukkan permulaan / akhir program (*start/end*). |
|  | *Manual Input* | Simbol *manual input* adalah simbol berfungsi untuk pemasukan data secara manual *on-line* keyboard |
|  | *Document* | Simbol *document* menggambarkan input dan output baik itu dalam proses manual, mekanik atau komputer. |
|  | *On-Page Connector* | Adalah penghubung bagian – bagian *flowchart* yang berada pada satu halaman. |
|  | *Direct data* | Simbol yg menyatakan *input* berasal dari *drum magne*tic atau *output* disimpan ke *drum magnetic*. |
|  | *Display* | Berfungsi untuk menyatakan peralatan *output* yang digunakan yaitu layar, *plotter*, *printer* dan sebagainya. |
|  | *Flow Line* | Menunjukkan arah aliran program. |

(Rosaly, 2019)

1. ***Unified* *Modeling* *Language* (UML)**

*Unified* *Modeling* *Language* atau disingkat UML adalah sebuah bahasa yang berdasarkan grafik/gambar untuk memvisualisasi, menspesifikasikan, membangun, dan mendokumentasikan dari sebuah sistem pengembangan software berbasis *Object Oriented*. UML sendiri juga memberikan standar penulisan sebuah sistem blueprint, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema database, dan komponen-komponen yang diperlukan dalam sistem software. Diagram *Unified Modelling Language* (UML) antara lain sebagai berikut (Zufria, 2013):

1. *Use Case Diagram*

*Use case* adalah pemodelan untuk menggambarkan *behavior* / kelakuan sistem yang akan dibuat. Use case diagram menggambarkan sebuah interaksi antara satu atau lebih sistem dengan sistem yang akan dibuat. Diagram juga lebih mudah karena penggambaran dilakukan dengan simbol– simbol yang sangat membantu pemahaman rancangan tersebut. (Nurfitria, 2021) Penjelasan simbol –simbol dari use case diagram dapat dilihat pada Tabel 2.2.

Tabel 2.2 Tabel Simbol Use Case

|  |  |  |
| --- | --- | --- |
| **Simbol** | **Nama** | **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 simbolnya adalah gambar orang, aktor belum tentu orang. |
|  | *Extend* | Relasi *use case* tambahan ke sebuah *use case* di mana *use case* yang ditambahkan dapat berdiri sendiri walau tanpa *use case* tambahan itu. |
|  | *Association* | Apa yang menghubungkan antara objek satu dengan objek lainnya. |
|  | *Use Case* | Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama *use case.* |
|  | *Generalization* | Hubungan generalisasi dan spesialisasi (umum - khusus) antara dua buah *use case* di mana fungsi yang satu adalah fungsi yang lebih umum dari lainnya. |
|  | *Include* | Relasi *use case* tambahan ke sebuah *use case* di mana *use case* yang ditambahkan memerlukan *use case* ini untuk menjalankan fungsinya atau sebagai syarat dijalankan *use case* ini. |

(Nurfitria, 2021)

1. *Class* Diagram

*Class* diagram menggambarkan struktur statis *class* didalam sistem *class* merepresentasikan sesuatu yang ditangani oleh sistem *class* data behubungan dengan yang lain. *Class* memiliki tiga area pokok yaitu (Nurfitria, 2021):

1. Nama, kelas harus mempunyai sebuah nama.
2. Atribut, adalah kelengkapan yang melekat pada kelas. Nilai dari suatu kelas hanya bisa diproses sebatas atribut yang dimiliki.
3. Operasi, adalah proses yang dapat dilakukan oleh sebuah kelas, baik pada kelas itu sendiri ataupun kepada kelas lainnya.

Adapun penjelasan simbol–simbol dari *class diagram* dapat dilihat pada Tabel 2.3.

Tabel 2.3 Tabel Simbol Class Diagram

|  |  |  |
| --- | --- | --- |
| **Simbol** | **Nama** | **Keterangan** |
|  | *Class* | Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama. |
|  | *Generalization* | Hubungan di mana 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. |
|  | *Realization* | Operasi yang benar-benar dilakukan oleh suatu objek. |
|  | *Dependency* | Hubungan di mana perubahan yang terjadi pada suatu elemen mandiri (independent) akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri. |
|  | *Association* | Apa yang menghubungkan antara objek satu dengan objek lainnya. |
|  | *Collaboration* | Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu Actor |

(Nurfitria, 2021)

1. *Activity Diagram*

*Activity Diagram* menunjukkan aktivitas sistem dalam bentuk kumpulan aksi-aksi, bagaimana masing-masing aksi tersebut dimulai, keputusan yang mungkin terjadi hingga berakhirnya aksi. *Activity diagram* juga dapat menggambarkan proses lebih dari satu aksi dalam waktu bersamaan. Diagram *activity* adalah aktivitas-aktivitas, objek, *state*, transisi *state* dan *event*. Dengan kata lain kegiatan diagram alur kerja menggambarkan perilaku sistem untuk aktivitas. (Nugraha, 2020) Penjelasan simbol–simbol dari activity diagram dapat dilihat pada Tabel 2.4.

Tabel 2.4 Tabel Simbol Activity Diagram

|  |  |
| --- | --- |
| **Simbol** | **Keterangan** |
|  | *Swimlane*, pembagian activity diagram untuk menunjukkan siapa melakukan apa. |
|  | *Start* *Point*, diletakkan pada pojok kiri atas dan merupakan awal aktivitas. |
|  | *End* *Point*, akhir aktivitas. |
|  | *Decision* *Points*, menggambar kan pilihan untuk pengambilan keputusan, tru atau false. |
|  | *Activities*, menggambar kan suatu proses/kegiat an bisnis. |
|  | *Fork*/percaban gan, digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabung kan dua kegiatan paralel menjadi satu. |
|  | Joiin (penggabungan) atau rake, digunakan untuk menunjukkan adanya dekomposisi. |

(Nugraha, 2020)

1. *Sequence Diagram*

Diagram *sequence* menggambarkan kelakuan objek pada *use case* denga mendeskripsikan waktu hidup objek dan *massage* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar diagram *sequence* maka harus diketahui objek-objek yang terlibat dalam sebuah use case beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat diagram *sequence* juga dibutuhkan untuk melihat skenario yang ada pada *use case*. (Simatupang, 2019) Penjelasan tentang symbol-simbol dari *sequence diagram* dapat dilihat pada Tabel 2.5.

Tabel 2.5 Tabel Simbol Sequence Diagram

|  |  |  |
| --- | --- | --- |
| **Simbol** | **Nama** | **Keterangan** |
|  | *Lifeline* | *Lifeline* mengindikasikan keberadaan sebuah *object* dalam basis waktu. Notasi untuk *Lifeline* adalah garis putus – putus *vertical* yang ditarik darisebuah *object*. |
|  | *Activation* | Activation dinotasikan sebagai sebuah kotak segi empat yang digambar pada sebuah obyek yang akan melakukan sebuah aksi. |
|  | *Message* | *Message*, digambarkan dengan anak panah horizontal antara *Activation Message* mengindikasikan komunikasi antara *object – object*. |
|  | *Object* | *Object* merupakan instance dari sebuah class dan dituliskan tersusun secara *horizontal*. Degambarkan sebagai sebuah *class* (kotak) dengan nama *object* di dalamnya yang diawali dengan sebuah titik koma. |
|  | *Actor* | *Actor* juga dapat berkomunikasi. |

(Simatupang, 2019)

1. ***Entity Relationship Diagram*** (**ERD)**

*Entity Relationship Diagram* (ERD) adalah model teknik pendekatan yang menyatakan atau menggambarkan hubungan suatu model. Di dalam hubungan ini tersebut dinyatakan yang utama dari ERD adalah menunjukan objek data (*Entity*) dan hubungan (*Relationship*), yang ada pada *Entity* berikutnya. (Fridayanthie, 2016) Simbol-simbol dalam *Entity Relationship Diagram* (ERD) dapat dilihat pada Tabel 2.6.

Tabel 2.6 Tabel Simbol ERD

|  |  |  |
| --- | --- | --- |
| **Simbol** | **Nama** | **Keterangan** |
|  | Entitas | Persegi panjang menyatakan himpunan entitas adalah orang, kejadian, atau berada di mana data akan dikumpulkan. |
|  | Atribut | Atribut merupakan informasi yang diambil tentang sebuah entitas. |
|  | Relasi | Belah ketupat menyatakan himpunan relasi merupakan hubungan antar entitas. |
|  | Link | Garis sebagai penghubung antar himpunan, relasi, dan himpunan entitas dengan atributnya. |

(Fridayanthie, 2016)

Kardinalitas Relasi adalah suatu hubungan antara beberapa entitas. contoh relasi antar mahasiswa dengan mata kuliah atau pelajaran yang diambil, yang mana pada tiap-tiap mahasiswa dapat mengambil beberapa mata kuliah serta setiap mata kuliah itu juga bisa diambil atau lebih dari 1 mahasiswa. Himpunan relasi antar entitias pemetaan kardiniliat terdiri dari: (Ropianto, 2021)

1. Relasi satu-ke-satu *(one-to-one)*

Pada relasisatu-ke-satu *(one-to-one)* atau dengan pemakaian angka (1 *to* 1) ini yang berarti setiap entitas pada himpunan entitas A berhubungan dengan paling banyak dengan satu entitas pada himpunan entintas B, dan begitu juga sebaliknya setiap entitas pada himpunan entitas B berhubungan paling banyak dengan satu entitas pada himpunan entitas A.

1. Relasi satu-ke-banyak *(one-to-many)*

Pada relasisatu-ke-banyak *(one-to-many)* atau bisa disimbolkan denga (1 to N) ini yang berarti setiap entitas pada himpunan A dapat berhubungan dengan banyak entitas pada himpunan entitas B, tetapi tidak sebaliknya, di mana setiap entitas pada himpunan entitas B berhubungan banyak dengan satu entitas pada himpunan entitas A.

1. Relasi banyak-ke-banyak *(many-to-many).*

Pada relasi banyak-ke-banyak *(one-to-many)* atau bisa disimbolkan denga (N to N) yang berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan B, dan demikian juga sebaliknya, di mana setiap entitas pada himpunan entitas B dapat berhubungan banyak entitas pada himpunan entitas A.

**BAB III**

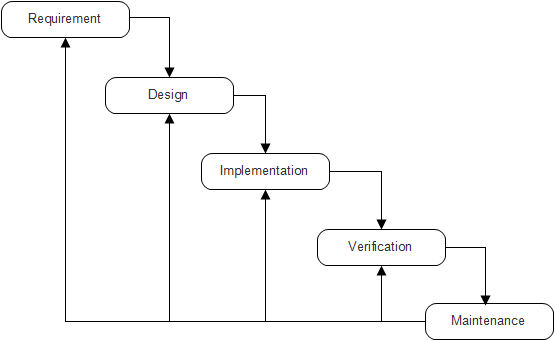
**METODE PELAKSANAAN**

1. **Metodologi Pelaksanaan**

Metodologi penelitian berisi tahap-tahap yang digunakan dalam penelitian ini agar terstruktur dengan baik. Dengan sistematika ini proses penelitian dapat dipahami dan diikuti oleh pihak lain. Penelitian yang dilakukan untuk merancang sistem diperoleh dari pengamatan data-data yang ada.

Dalam pembuatan Rancang Bangun Sistem Peminjaman Sarana dan Prasara Berbasis *Website* di Yayasan Pendidikan An-Nadhir ini penulis menggunakan metode *waterfall*. Metode *waterfall* ini adalah salah satu metode pengembangan perangkat lunak di mana proses pengembangan perangkat lunak dibagi ke dalam beberapa tahapan yaitu *requirement*, *design*, *implementation*, *verification* dan *maintenance*. Dalam metode *waterfall* proses pengembangan dilakukan secara berurutan dari tahap pertama sampai yang paling akhir.

1. **Desain Penelitian**



Gambar 3. 1 Metode Waterfall

Dari Gambar 3.1 di atas kita bisa melihat bahwa tahapan – tahapan dari Rancang Bangun Sistem Peminjaman Sarana dan Prasara Berbasis *Website* di Yayasan Pendidikan An-Nadhir, adapun penjelasan dari setiap tahap adalah :

1. *Requirement Analysis*

Pada tahap ini penulis melakukan indentfikasi kebutuhan terkait sistem peminjaman.

1. *System Design*

Tahap perancangan aplikasi merupakan tahap untuk merancang bagaimana sistem peminjaman yang dibuat berdasarkan rancangan sistem dengan uml, flowchart, entity relationship diagran, rancangan database hingga perancangan pada desain *interface.*

1. *Implementation*

Tahap Implementasi aplikasi ini merupakan tahap publikasi aplikasi yang sudah jadi dan siap untuk diimplementasikan kepada siswa-siswi Yayasan Pendidikan An-Nadhir dan staff bagian sarana prasarana.

1. *Verification*

Setelah implementasi, maka tahap selanjutnya adalah *verification*. Dalam tahap *verification* ini dilakukan dua hal untuk mengembangkan sistem, yaitu *integration* dan *testing*. Semua unit program diintegrasikan ke dalam sistem. Kemudian, akan dilakukan *testing* pada sistem untuk memastikan bahwa tidak ada kegagalan dan error dalam sistem. Setiap unit kecil diuji coba apakah ada yang mengalami *error*. Jika masih ada yang *error* maka akan kembali ke *step* sebelumnya.

1. *Operation and Maintenance*

Tahap *operation* *and* *maintenance* ini adalah akhir dalam Rancang Bangun Sistem Peminjaman Sarana dan Prasara Berbasis *Website* di Yayasan Pendidikan An-Nadhir. Jika semua tahapan sudah diselesaikan dan sudah menjadi satu sistem yang terintegrasi, maka akan dilakukan penerapan operasional dan pemeliharaan. Pemeliharaan ini termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah testing sebelumnya. Di tahapan ini lebih mengutamakan *maintenance*, karena uji coba ini menentukan apakah sistem akan berhasil atau tidak memenuhi kebutuhan.

1. **Analisa Kebutuhan Sistem**

Dalam pembuatan Rancang Bangun Sistem Peminjaman Sarana dan Prasara Berbasis *Website* di Yayasan Pendidikan An-Nadhir terdapat kebutuhan baik itu dalam hal *software* maupun *hardware*. Berikut adalah kebutuhan *software* dan *hardware* tersebut:

1. Kebutuhan *Software*

Kebutuhan *software* dalam pembuatan Rancang Bangun Sistem Peminjaman Sarana dan Prasara Berbasis *Website* di Yayasan Pendidikan An-Nadhir dapat dilihat pada Tabel 3.1.

Tabel 3.1 Tabel Daftar Kebutuhan *Software*

|  |  |  |
| --- | --- | --- |
| **No** | **Jenis *Software*** | **Kebutuhan *Software*** |
| 1. | Sistem Operasi | Microsoft Windows 10 |
| 2. | Bahasa Pemograman | HTML,PHP,CSS, *Javascript,* |
| 3. | Pengolah *Software* | Visual Studio Code |
| 4. | *Framework* | Laravel 8 |
| 5. | Penyimpanan Data | *Database* MySQL |
| 6. | Desain Aplikasi | Figma, Draw.io |
|  | *Web Browser* | Google Chrome |

1. Kebutuhan *Hardware* (Perangkat Keras)

Kebutuhan *hardware* dalam pembuatan Rancang Bangun Sistem Peminjaman Sarana dan Prasarana Berbasis *Website* di Yayasan An-Nadhir dapat dilihat pada Tabel 3.2.

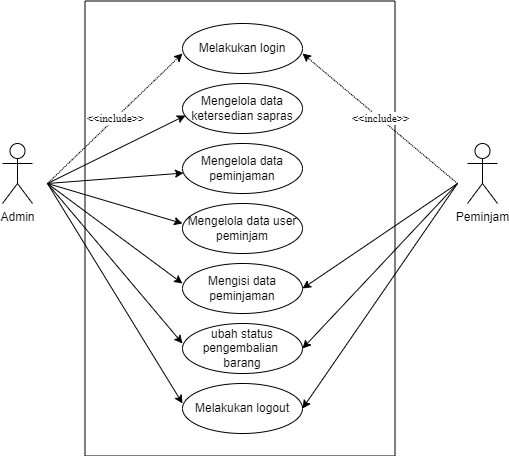
Tabel 3.2 Tabel Daftar Kebutuhan Hardware

|  |  |  |
| --- | --- | --- |
| **No** | **Jenis *Hardware*** | **Kebutuhan *Hardware*** |
| 1 | Laptop | LCD 14 inch |
| 2 | Processor | AMD Ryzen 3 |
| 3 | RAM | 8 GB |
| 4 | Harddisk | 1000 GB |

1. **Perancangan Sistem dengan *Unified Modeling Language*(UML)**

UML dapat diaplikasikan untuk merancang perangkat lunak, untuk menjabarkan sistem secara rinci untuk analisa kebutuhan sistem, dan untuk mendokumentasikan sistem. Perancangan sistem dengan UML pada Rancang Bangun Sistem Peminjaman Sarana dan Prasarana Berbasis *Website* di Yayasan Pendidikan An-Nadhir, meliputi *use case diagram, activity diagram, sequence diagram,* dan *class diagram.*

1. *Use Case Diagram*

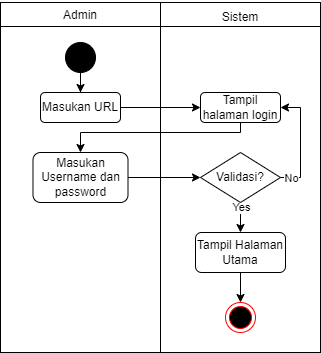
*Use case diagram* merupakan salah satu dari *Unfied Modelling Language* (UML) yang tidak menggambarkan banyak detail, namun bisa mendeskripsikan gambar tingkat tinggi dari relasi antara *use case, actor,* dan sistem. Secara umum *use case diagram* bisa digunakan untuk mewakili tujuan interaksi sistem dengan pengguna dan dapat mendefinisikan persyaratan fungsional suatu sistem. *Use case diagram* pada Rancang Bangun Sistem Peminjaman Sarana dan Prasarana Berbasis *Website* Di Yayasan Pendidikan An-Nadhir Losarang memiliki dua aktor yaitu aktor peminjam dan aktor admin. Aktor admin dapat mengkonfirmasi peminjaman sarana dan prasara yang dilakukan oleh aktor peminjam dan admin juga memiliki hak akses dalam pengelolaan data, seperti data sarana dan prasarana, data peminjaman sarana dan prasaran, data pengembalian sarana dan prasaran, serta data *user*. Berikut adalah Berikut *use case diagram* dari Rancang Bangun Sistem Peminjaman Sarana dan Prasarana Berbasis *Website* Di Yayasan Pendidikan An-Nadhir Losarang dapat dilihat pada gambar 3.2 di bawah ini.

Gambar 3. 2 Use Case Diagram

1. *Activity Diagram*

*Activity diagram* adalah bagian penting dari *Unified Modelling Language* (UML) yang menggambarkan aspek dinamis dari sistem dengan cara menunjukkan aliran pesan dari satu aktivitas ke aktivitas lainnya. Penggambaran aliran aktivitas dari sistem dilakukan secara urut pada *activity diagram.* Berikut *activity diagram* dari Rancang Bangun Sistem Peminjaman Sarana dan Prasarana Berbasis *Website* Di Yayasan Pendidikan An-Nadhir Losarang.

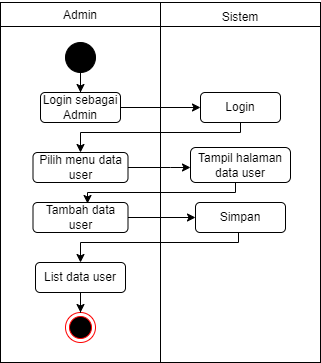
1. *Activity Diagram Login* Admin

*Activity diagram login* admin merupakan *activity diagram* yang menjelaskan aktivitas *login* admin supaya dapat memiliki akses ke dalam sebuah sistem. Berikut ini *activity diagram* dapat dilihat pada gambar 3.3.

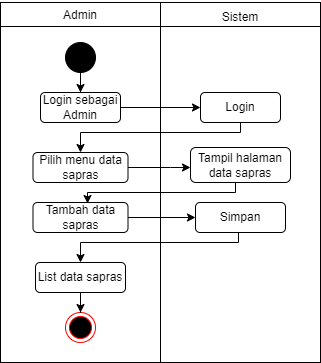
Gambar 3. 3 Activity Diagram Login Admin

1. *Activity Diagram* Data *User*

*Activity diagram* data *user* menggambarkan langkah-langkah admin dapat menambahkan data *user.* Berikut ini *activity diagram* data *user* yang dapat dilihat pada gambar 3.4.

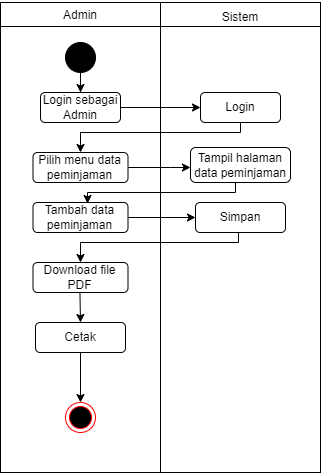
Gambar 3. 3 Activity Diagram Data User

1. *Activity Diagram* Data SAPRAS

Pada data SAPRAS, admin dapat menambahkan data SAPRAS. Berikut ini *activity diagram* dataSAPRASyang dapat dilihat pada gambar 3.5.

Gambar 3. 4 Activity Diagram Data SAPRAS

1. *Activity Diagram* Data Peminjaman

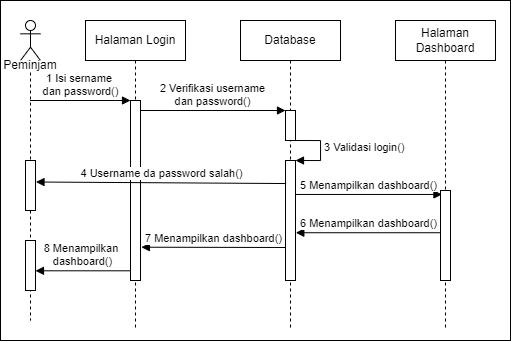
*Activity diagram* data peminjaman merupakan langkah-langkah admin untuk melihat dan menambahkan data peminjaman sarana prasarana. Berikut ini merupakan *activity diagram* data peminjaman yang dapat dilihat pada gambar 3.5.

Gambar 3. 4 Activity Diagram Data Peminjaman

1. *Sequence Digram*

*Sequence diagram* merupakan gambaran antar objek di dalam dan di sekitar sistem berupa *message.* Tujuan dari *sequence diagram* adalah mendefinisikan urutan kejadian yang dapat menghasilkan *output* yang diinginkan. Pada Sistem Peminjaman Sarana dan Prasarana Berbasis *Website* Di Yayasan Pendidikan An-Nadhir terdiri dari *sequence diagram login* peminjam, *sequence diagram* data *user, sequence diagram* data peminjaman pada *user* admin, *sequence diagram* data SAPRAS, dan *sequence diagram* data peminjaman pada *user* peminjam.

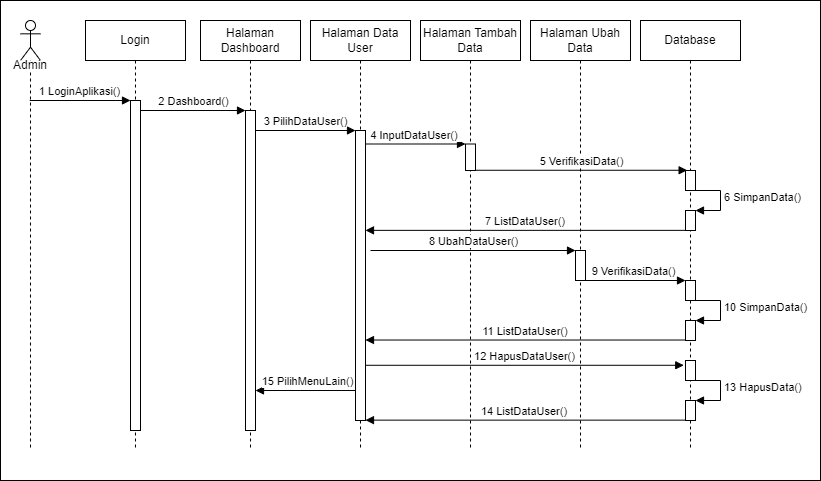
1. *Sequence Diagram Login* Peminjam

*Sequence diagram login* peminjam menggambarkan urutan proses *login* peminjam. Dimulai dari peminjam memasukkan *username* dan *password* pada halaman *login,* kemudian *database* melakukan validasi *username* dan *password.* Jika *username* dan *password* salah maka *user* kembali memasukkan *username* dan *password,* tetapi jika benar maka akan menampilkan halaman *dashboard.* Berikut ini sequence diagram login peminjam yang dapat dilihat pada gambar 3.5.

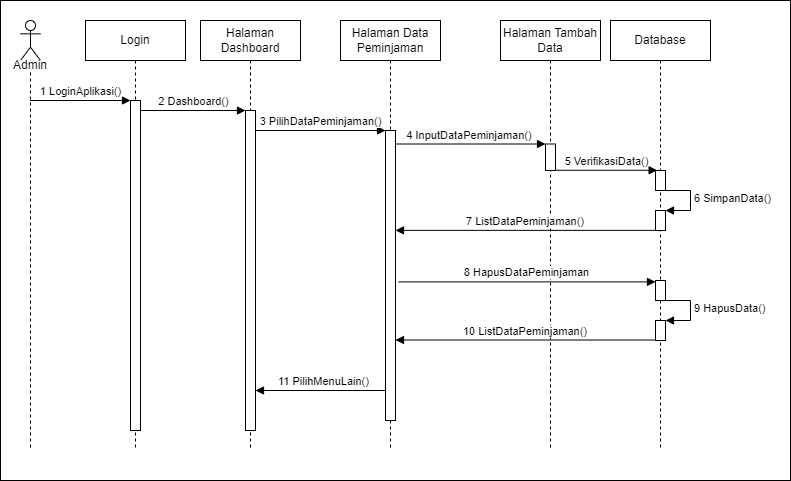
Gambar 3. 5 Sequence Diagram Login Peminjam

1. *Sequence Diagram* Data *User*

Pada *sequence diagram* data *user* menggambarkan langkah-langkah aktor admin yang dapat melakukan tambah, ubah, dan hapus data. Proses kerja dimulai dari admin yang sudah melakukan *login,* lalu memilih halaman data *user.* Kemudian admin dapat menambah data pada halaman tambah data, lalu data diverifikasi dan disimpan pada *database.* Setelah itu admin ditampilkan list data yang tersimpan dan admin bisa melakukan ubah data. Setelah admin melakukan ubah data, maka data akan terubah dan tersimpan pada *database.* Admin juga dapat melakukan hapus data, maka data yang tersimpan di *database* akan terhapus. Berikut ini merupakan *sequence diagram* data *user* yang dapat dilihat pada gambar 3.6.

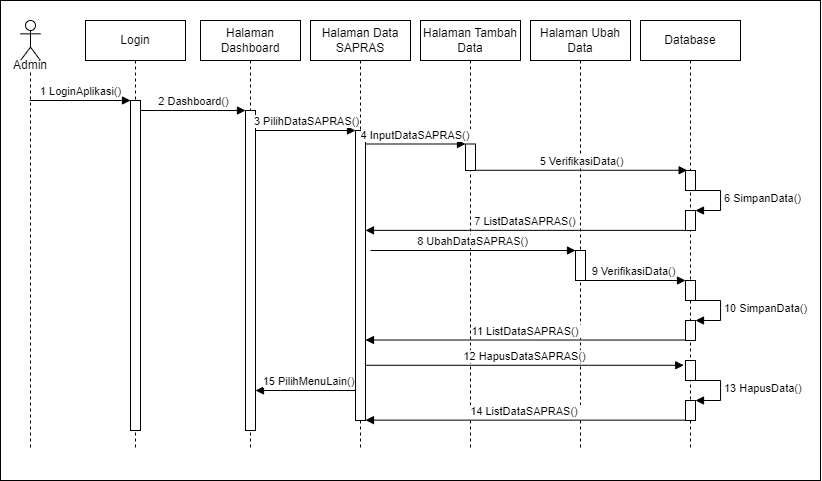
Gambar 3. 6 Sequence Diagram Data User

1. *Sequence Diagram* Data Peminjaman Pada *User* Admin

Pada gambar 3.7 menjelaskan *sequence diagram* data peminjaman pada *user* admin merupakan urutan kerja aktor admin menambah dan menghapus data peminjaman. Urutan kerja dimulai dari admin yang sudah melakukan *login,* lalu memilih halaman data peminjaman*.* Kemudian admin dapat menambah data pada halaman tambah data, lalu data diverifikasi dan disimpan pada *database.* Setelah itu admin juga dapat melakukan hapus data.

Gambar 3. 7 Sequence Diagram Data Peminjamam

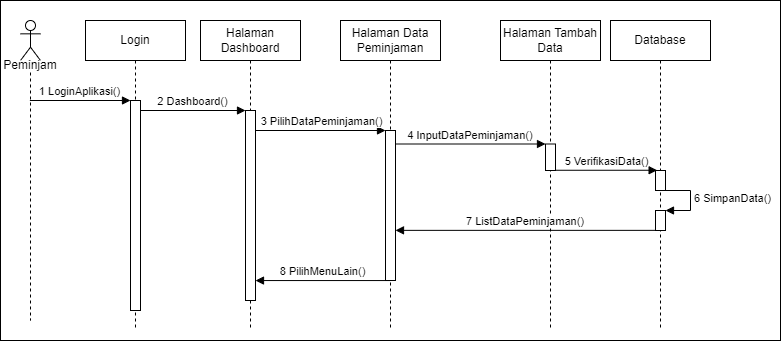
1. *Sequence Diagram* Data SAPRAS

Pada *sequence diagram* dataSAPRASmenggambarkan langkah-langkah aktor admin yang dapat melakukan tambah, ubah, dan hapus data. Proses kerja dimulai dari admin yang sudah melakukan *login,* lalu memilih halaman data SAPRAS*.* Kemudian admin dapat menambah data pada halaman tambah data, lalu data diverifikasi dan disimpan pada *database.* Setelah itu admin ditampilkan list data yang tersimpan dan admin bisa melakukan ubah data. Setelah admin melakukan ubah data, maka data akan terubah dan tersimpan pada *database.* Admin juga dapat melakukan hapus data, maka data yang tersimpan di *database* akan terhapus. Berikut ini merupakan *sequence diagram* dataSAPRASyang dapat dilihat pada gambar 3.8.

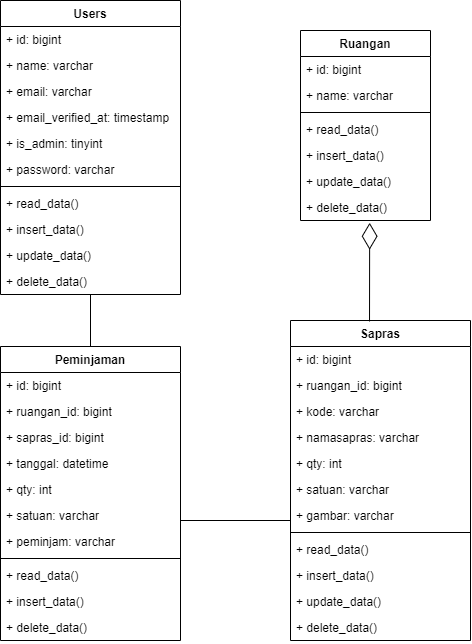
Gambar 3. 8 Sequence Diagram Data SAPRAS

1. *Sequence Diagram* Data Peminjaman Pada *User* Peminjam

*Sequence diagram* data peminjaman pada *user* peminjam menjelaskan langkah-langkah aktor peminjam melakukan tambah data peminjaman. Proses kerja dimulai dari peminjam yang sudah melakukan *login*, memilih halaman *form* peminjaman. Kemudian peminjam mengisi *form* tersebut sesuai dengan kebutuhan, maka data diverifikasi dan disimpan di *database.* Setelah itu peminjam akan ditampilkan list data peminjaman yang sudah tersimpan. Berikut ini merupakan *sequence diagram* data peminjaman pada *user* peminjam yang dapat dilihat pada gambar 3.9.

Gambar 3. 9 Sequence Diagram Data Peminjaman Pada User Peminjam

1. *Class Diagram*

*Class diagram* atau diagram kelas adalah gambaran mengenai sistem atau perangkat lunak serta relasi-relasi yang terkandung di dalamnya. Berikut ini *class diagram* pada Sistem Peminjaman Sarana dan Prasarana Berbasis *Website* Di Yayasan Pendidikan An-Nadhir yang dapat dilihat pada gambar 3.10.

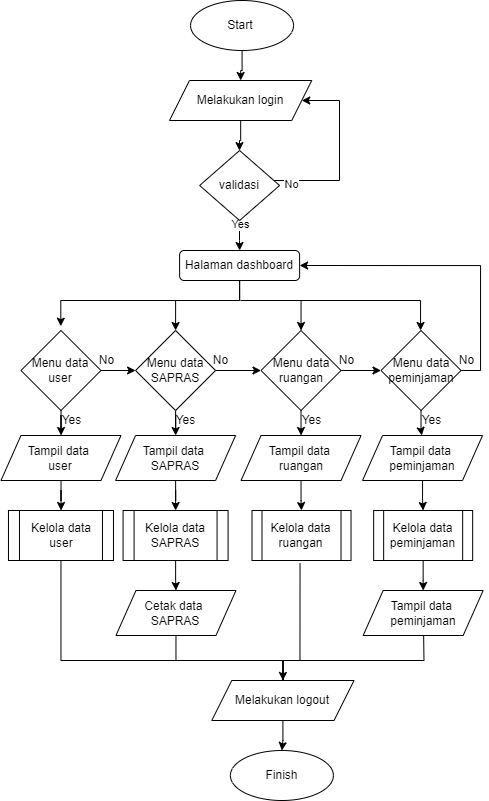
Gambar 3. 10 *Class Diagram*

1. ***Flowchart***

*Flowchart* merupakan diagram yang menampilkan langkah-langkah untuk melakukan sebuah proses dari sebuah program. Pada Sistem Peminjaman Sarana dan Prasarana Berbasis *Website* Di Yayasan Pendidikan An-Nadhir dibagi menjadi dua yaitu, *flowchart* program dan *flowchart* sistem.

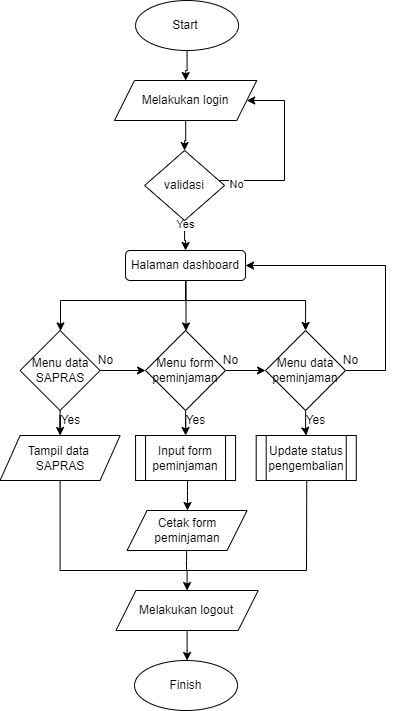
1. *Flowchart* Program

*Flowchart* program pada Sistem Peminjaman Sarana dan Prasarana Berbasis *Website* Di Yayasan Pendidikan An-Nadhir akan digambarkan menjadidua *flowchart* program, diantaranya sebagai berikut.

1. *Flowchart* Program Admin

Gambar 3. 11 *Flowchart* Program Admin

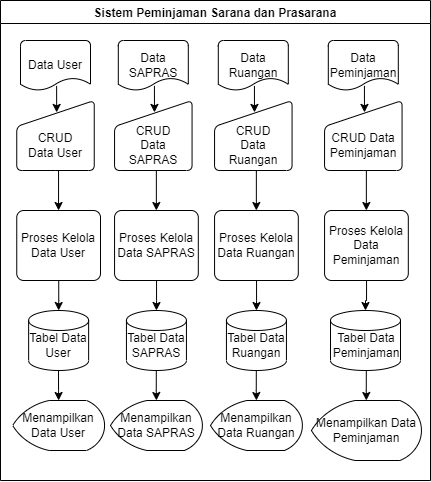
Pada gambar 3.11 merupakan langkah-langkah admin dalam menjalankan sistem mulai dari *login,* memilih menu, hingga *logout.*

1. *Flowchart* Program Peminjam

Gambar 3. 12 *Flowchart* Program Peminjam

Pada gambar 3.12 merupakan langkah-langkah *user* peminjam dalam menjalankan sistem mulai dari *login,* memilih menu, hingga *logout.*

1. *Flowchart* Sistem

Berikut ini merupakan *flowchart* sistem pada Sistem Peminjaman Sarana dan Prasarana Berbasis *Website* Di Yayasan Pendidikan An-Nadhir dapat dilihat pada gambar 3.13.

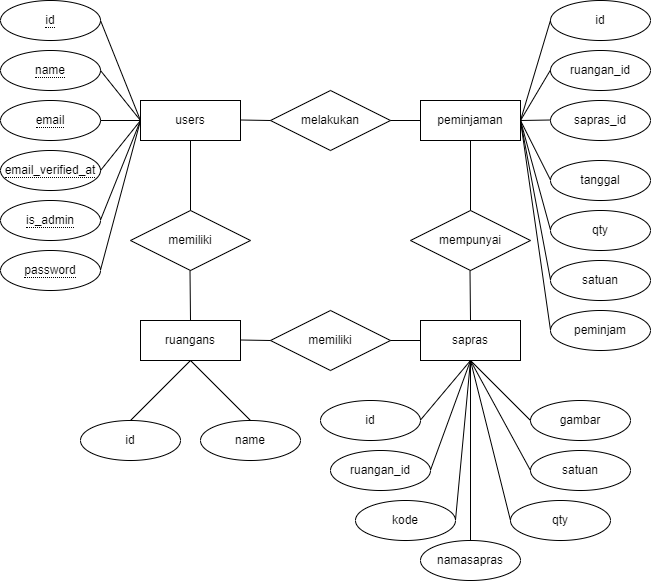
Gambar 3.13 *Flowchart* Sistem

1. **Perancangan *Database***

Berikut adalah perancangan *database* pada Sistem Peminjaman Sarana dan Prasarana Berbasis *Website* Di Yayasan An-Nadhir Losarang.

1. *Entity Relationship Diagram* (ERD)

*Entity Relationship Diagram* (ERD) merupakan diagram perangcangan basis data dan menunjukan relasi antara entitas beserta atribut-atributnya. ERD teridir dari entitas, atribut, dan relasi. Pada Sistem Peminjaman Sarana dan Prasarana Berbasis *Website* Di Yayasan An-Nadhir terdapat empat entitas diantaranya adalah entitas *users,* entitas sapras, entitas ruangan, dan entitas peminjaman Berikut ini ERD dari Sistem Peminjaman Sarana dan Prasarana Di Yayasan An-Nadhir dapat dilihat pada gambar 3.14.



Gambar 3.14 *Entity Relationship Diagram* (ERD)

1. Rancangan Tabel

Rerancangan tabel-tabel pada *database* dalam pembuatan Sistem Peminjaman Sarana dan Prasarana Berbasis *Website* Di Yayasan Pendidikan An-Nadhir Losarang diantaranya sebagai berikut.

1. Tabel *Users*

Tabel *users* adalah tabel yang digunakan untuk menyimpan data dari akun-akun *user* baik admin maupun peminjam. Berikut merupakan tabel *users* yang dapat dilihat pada tabel 3.3.

Tabel 3.3 Tabel *Users*

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Field** | **Tipe Data** | **Keterangan** |
| 1 | id\* | Bigint(20) | Menyimpan id user |
| 2 | name | varchar(255) | Menyimpan nama user |
| 3 | email | varchar(255) | Menyimpan email user |
| 4 | is\_admin | tiny(1) | Menyimpan roles user |
| 5 | password | varchar(255) | Menyimpan password |
| 6 | created\_at | timestamp | Menyimpan waktu dibuatnya user |
| 7 | update\_at | timestamp | Menyimpan waktu diubahnya user |

1. Tabel SAPRAS

Tabel SAPRAS adalah tabel yang digunakan untuk menyimpan data-data sarana dan prasarana. Berikut ini tabel SAPRAS yang dapat dilihat pada tabel 3.4.

Tabel 3.4 Tabel SAPRAS

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Field** | **Tipe Data** | **Keterangan** |
| 1 | id\* | bigint(20) | Menyimpan id sapras |
| 2 | ruangan\_id\* | bigint(20) | Menyimpan id ruangan |
| 3 | kode | varchar(255) | Menyipan kode sapras |
| 4 | namasapras | varchar(255) | Menyimpan nama sapras |
| 5 | qty | int(11) | Menyimpan stok sapras |
| 6 | satuan | varchar(255) | Menyimpan satuan sapras |
| 7 | gambar | varchar(255) | Menyimpan gambar |
| 8 | created\_at | timestamp | Menyimpan waktu dibuatnya sapras |
| 9 | update\_at | timestamp | Menyimpan waktu diubahnyaa sapras |

1. Tabel Ruangan

Tabel ruangan merupakan tabel yang digunakan untuk menyimpan data ruangan. Berikut tabel ruangan yang dapat dilihat pada tabel 3.5.

Tabel 3.5 Tabel Ruangan

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Field** | **Tipe Data** | **Keterangan** |
| 1 | id\* | bigint(20) | Menyimpan id ruangan |
| 2 | name | varchar(255) | Menyimpan nama ruanan |
| 3 | created\_at | timestamp | Menyimpan waktu dibuatnya data ruangan |
| 4 | update\_at | timestamp | Menyimpan waktu diubahnya data ruangan |

1. Tabel Peminjaman

Tabel peminjaman merupakan tabel untuk menyimpan data-dat peminjaman sarana dan prasarana. Berikut tabel peminjaman yang dapat dilihat pada tabel 3.6.

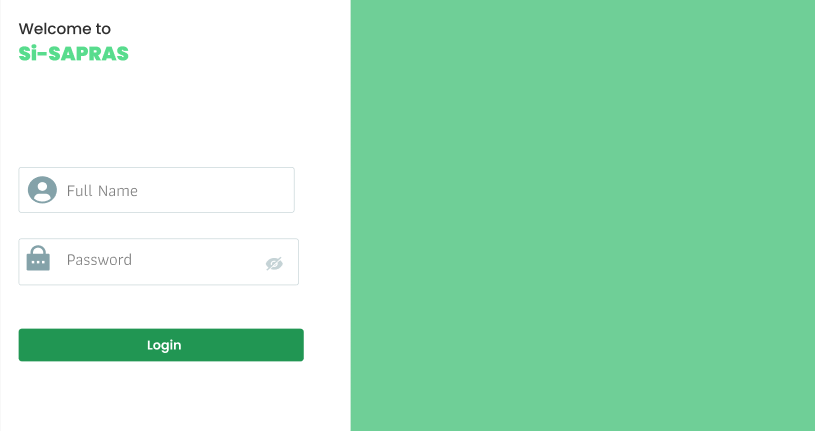
Tabel 3.6 Tabel Peminjaman

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Field** | **Tipe Data** | **Keterangan** |
| 1 | id\* | bigint(20) | Menyimpan id sapras |
| 2 | ruangan\_id\* | bigint(20) | Menyimpan id ruangan |
| 3 | sapras\_id\* | bigint(20) | Menyimpan id sapras |
| 4 | tanggal | datetime | Menyimpan tanggal peminjaman |
| 5 | qty | int(11) | Menyimpan kuantitas sapras |
| 6 | satuan | varchar(255) | Menyimpan satuan sapras |
| 7 | peminjam | varchar(255) | Menyimpan nama peminjam |
| 8 | created\_at | timestamp | Menyimpan waktu dibuatnya peminjaman |
| 9 | update\_at | timestamp | Menyimpan waktu diubahnya peminjaman |

1. **Perancangan Antar Muka (*Interface*)**

Perancangan antar muka atau *interface* merupakan sebuah tahapan dalam mendesi tampilan suatu sistem. Berikut ini rancangan antar muka Sistem Peminjaman Sarana dan Prasarana Berbasis *Website* Di Yayasan An-Nadhir.

1. Rancangan Halaman *Login*

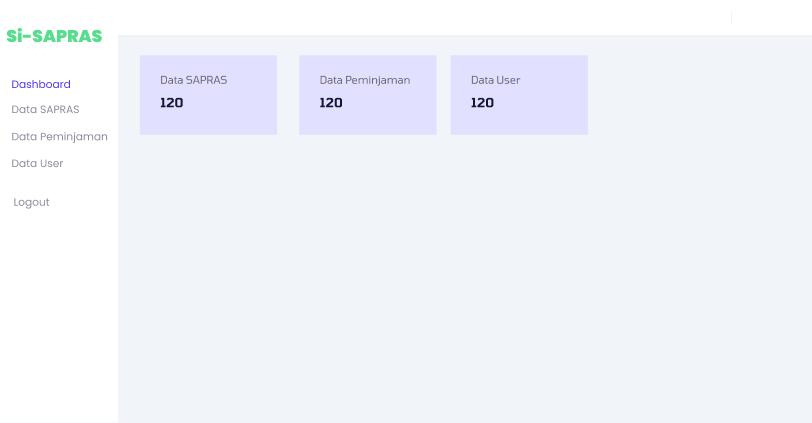
Halaman *login* merupakanhalaman dimana user admin maupun user peminjam harus mengisi *username* dan *password* dengan benar. Ketika *user* bisa melakukan *login*, maka user dapat mengakses halaman berikutnya. Rancangan halaman *login* dapat dilihat pada gambar 3.15.

Gambar 3.15 Rancangan Halaman *Login*

1. Rancangan Halaman Admin

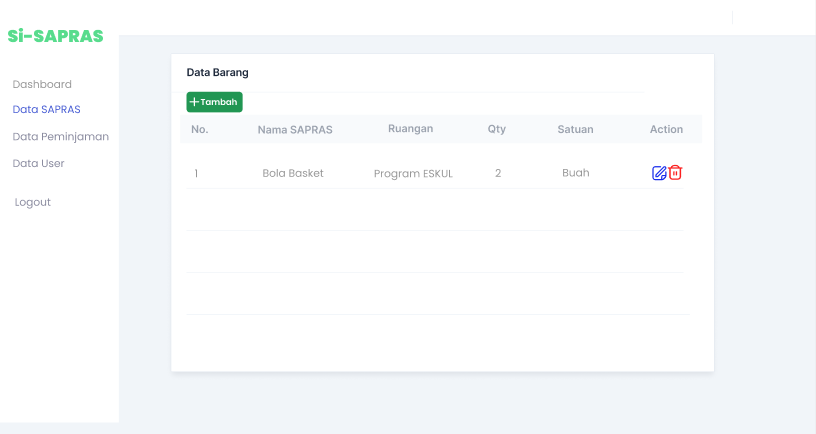
Halaman admin adalah halaman yang mengelola data-data seperti data SAPRAS, data *user,* data peminjaman, dan lainnya. Berikut adalah rancangan halaman admin Siste Peminjaman Sarana dan Prasarana Berbasis *Website* Di Yayasan Pendidikan An-Nadhir Losarang.

1. Rancangan Halaman *Dashboard*  Admin

Pada rancangan halaman *dashboard* admin terdapat tampilan informasi dari total data SAPRAS, data peminjaman, dan data *user.* Berikut ini rancangan halaman *dashboard* admin yang dapat dilihat pada gambar 3.16.

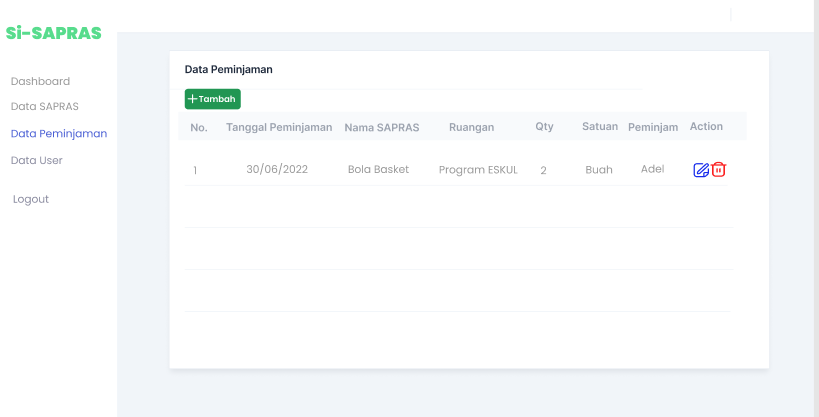
Gambar 3.16 Rancangan Halaman *Dashboard* Admin

1. Rancangan Halaman Data SAPRAS

Pada rancangan halaman data SAPRAS menampilkan data-data SAPRAS yang ada pada Yayasan Pendidikan An-Nadhir Losarang. Pada halaman ini, admin dapat menambahkan, mengubah, dan menghapus data SAPRAS. Berikut rancangan halaman data SAPRAS dapat dilihat pada gambar 3.17.

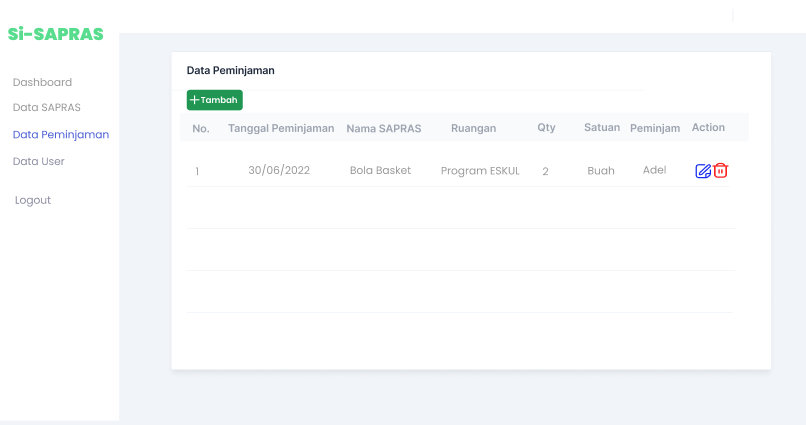
Gambar 3.17 Rancangan Halaman Data SAPRAS

1. Rancangan Halaman Data Peminjaman

Pada rancangan halaman data peminjaman menampilkan data-data peinjaman. Pada halaman ini, admin dapat menambahkan dan menghapus data peminjaman. Berikut rancangan halaman data peminjaman dapat dilihat pada gambar 3.18.

Gambar 3.18 Rancangan Halaman Data Peminjaman

1. Rancangan Halaman Data *User*

Pada rancangan halaman data *user* menampilkan data-data *user* baik *user* admin maupun *user* peminjam. Pada halaman ini, admin dapat menambahkan, mengubah, dan menghapus data *user*. Berikut rancangan halaman data *user* dapat dilihat pada gambar 3.19.

Gambar 3.19 Rancangan Halaman Data *User*

1. Rancangan Halaman Peminjam

Rancangan halaman peminjam merupakan dimana *user* peminjam dapat melakukan proses peminjaman SAPRAS. Berikut rancangan halaman peminjam Sistem Peminjaman Sarana dan Prasarana Berbasis *Website* Di Yayasan Pendidikan An-nadhir Losarang.

1. Rancangan Halaman *Landing Page*

Rancangan halaman *landing page* merupakan halaman utama yang berisikan informasi mengenai Sistem Peminjaman Sarana dan Prasarana. Berikut adalah rancangan halaman *landing page* dapat dilihat pada gambar 3.20.

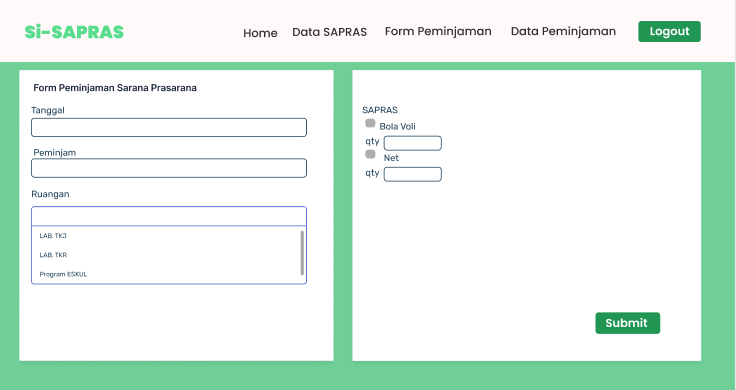
Gambar 3.20 Rancangan Halaman *Landing Page*

1. Rancangan Halaman Data SAPRAS

Pada rancangan halaman data SAPRAS terdiri dari tabel data SAPRAS. Rancangan halaman data SAPRAS dapat dilihat pada gambar 3.21.

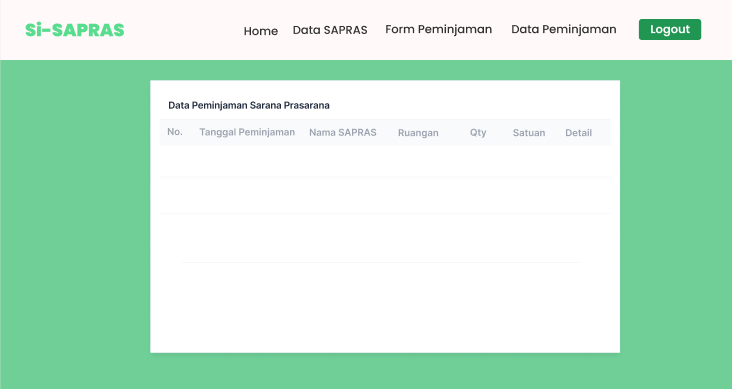
Gambar 3.21 Rancangan Halaman Data SAPRAS

1. Rancangan Halaman *Form* Peminjaman

Pada rancangan halaman *form* peminjaman, peminjam dapat mengisi *form* tersebut sesuai dengan kebutuhan. Berikut adalah rancangan halaman *form* peminjaman yang dapat dilihat pada gambar 3.22.

Gambar 3.22 Rancangan Halaman *Form* Peminjaman

1. Rancangan Halaman Data Peminjaman

Setelah mengisi *form* peminjaman hasil data tersebut maka akan tersimpan pada halaman data peminjaman. Rancangan halaman data peminjaman dapat dilihat pada gambar 3.23 sebagai berikut.

Gambar 3.23 Rancangan Halaman DataPeminjaman

**BAB IV**

**HASIL DAN PEMBAHASAN**

1. **Hasil**

Tugas akhir ini menghasilkan sebuah sistem peminjaman sarana dan prasarana berbasis *website* untuk Yayasan Pendidikan An-Nadhir Losarang Indramayu. Sistem ini digunakan untuk membantu pengguna dalam melakukan proses peminjaman sarana dan prasarana. Sistem ini digunakan oleh 2 level *user* yaitu peminjam dan admin.

Selain itu, sistem ini juga bisa digunakan untuk melakukan pendataan peminjaman sarana dan prasarana. Supaya data para peminjam yang melakukan peminjaman dapat tercatat secara rapih dan dapat dicetak oleh admin sistem.

1. **Pembahasan**

Setelah melalui proses pengumpulan data, analisis kebutuhan pengembangan sistem serta perancangan sistem, maka penulis mengimplementasikan semua sistem ini untuk terciptanya sistem peminjaman sarana dan prasarana berbasis *website* di Yayasan Pendidikan An-Nadhir Losarang.

1. **Implementasi**
2. **Implementasi Database**
3. **Implementasi Desain Interface dan program**

**BAB V**

**PENUTUP**

**DAFTAR PUSTAKA**

Estiningrat, S. (2022). Rancang Bangun Aplikasi Website Peminjaman Barang Di Universitas Dinamika Bagian Kemahasiswaan. *Doctoral dissertation, Universitas Dinamika*.

Fauzi, H. (2018). Rancang Bangun Aplikasi Simpan Pinjam Uang Mahasiswa Fakultas Teknik Universitas Asahan Berbasis Web. *JTI(Jurnal Teknologi Informasi) Vol.2, No.1. Juni 2018* .

Fridayanthie, E. W. (2016). Rancang bangun sistem informasi permintaan atk berbasis intranet (studi kasus: kejaksaan negeri rangkasbitung). *Jurnal khatulistiwa informatika*.

Lestanti, S. (2016). Sistem Pengarsipan Dokumen Guru Dan Pegawai Menggunakan Metode Mixture Modelling Berbasis Web. *Antivirus: Jurnal Ilmiah Teknik Informatika*.

Listyanto, H. (2019). Listyanto, H. (2019). RANCANG BANGUN APLIKASI E-MARKETPLACE BUKU BERBASIS WEBSITE. *Doctoral dissertation, University of Technology Yogyakarta*.

Nugraha, W. (2020). Pemodelan diagram uml sistem pembayaran tunai pada transaksi e-commerce. *JTIK (Jurnal Teknik Informatika Kaputama)*.

Nurfitria, A. (2021). Sistem Informasi Pemesanan Ojek/Transportasi Online Berbasis Web . *Journal of Student Development Information System (JoSDIS)* .

Nurlaila, H. (2014). Rancang Bangun Sistem Informasi Koperasi Berbasis Web Pada Koperasi Warga Baru MTS N 17 Jakarta. *Studia Informatika: Jurnal Sistem Informasi*.

Pradita, A. (2021). Evaluasi usability dan perancangan user interface menggunakan heuristic evaluation dan metode lean ux dengan standar iso 13407 studi kasus: RS Syarif Hidayatullah. *Bachelor's thesis, Fakultas Sains Dan Teknologi UIN Syarif Hidayatullah Jakarta*.

Prayitno, A. (2015). Pemanfaatan Sistem Informasi Perpustakaan Digital Berbasis Website Untuk Para Penulis. *IJSE – Indonesian Journal on Software Engineering*.

Prsetia, A. (2019). Aplikasi electronic commerce sebagai media penjualan produk makanan ringan business development center kabupaten pringsewu. *Jurnal TAM (Technology Acceptance Model)*.

Ropianto, M. (2021). ERD & Praktikum DBMS. *Academia.edu*.

Rosaly, R. (2019). Pengertian Flowchart Beserta Fungsi dan Simbol-simbol Flowchart yang Paling Umum Digunakan.

Sari, D. P. (2020). Implementasi Framework Laravel pada Sistem Informasi Penyewaan Kamera (Studi Kasus di Rumah Kamera Semarang). *Jurnal Informatika dan Rekayasa Perangkat Lunak*.

Simatupang, J. (2019). Perancangan Sistem Informasi Pemesanan Tiket Bus Pada Po. Handoyo Berbasis Online. *Jurnal Intra Tech*.

Sinaga, J. (2020). Perancangan Sistem Informasi Manajemen Bengkel Motor Pada Karya Motor Sukses Dengan Microsoft Visual Basic.Net 2010. *STMIK GICI BATAM*.

Yusuf, D. (2017). Sistem Peminjaman Barang Di Perusahaan Menggunakan Teknologi RFID. *Jurnal SIGMA*.

Zufria, I. (2013). Pemodelan Berbasis UML (Unified Modeling Language) dengan Strategi Teknik Orientasi Objek User Centered Design (UCD) dalam Sistem Administrasi Pendidikan. Pemodelan Berbas. UML.

**LAMPIRAN-LAMPIRAN**