**JUDUL SKRIPSI**

SKRIPSI

Diajukan untuk Memenuhi Salah Satu Syarat Kelulusan

Program Pendidikan Sarjana

Oleh:

………….

…………..

****

PROGRAM STUDI SISTEM INFORMASI

FAKULTAS ILMU KOMPUTER UNIVERSITAS SUBANG

SUBANG

……………

# HALAMAN PENGESAHAN

**JUDUL SKRIPSI**

……………..

………………

Subang, ………….

|  |  |  |
| --- | --- | --- |
|  | Menyetujui, |  |
| **…………………………**  Pembimbing I |  | **…………………….**  Pembimbing II |
|  | Mengetahui |  |
| **Dr. Tepi Peirisal,S.Sos.,M.Si**  Dekan Fakultas Ilmu Komputer |  | **Bagus Ali Akbar,SI.,M.Kom**  Ketua Program Studi |

PROGRAM STUDI SISTEM INFORMASI

FAKULTAS ILMU KOMPUTER UNIVERSITAS SUBANG

SUBANG

………..

# PERNYATAAN

Dengan ini saya menyatakan Skripsi Tugas Akhir dengan judul “JUDUL SKRIPSI” ini beserta isinya adalah benar-benar karya saya sendiri, dan saya tidak melakukan penjiplakan atau pengutipan dengan cara-cara tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan. Atas pernyataan ini, saya siap menanggung risiko/sanksi yang dijatuhkan kepada saya apabila kemudian ditemukan adanya pelanggaran terhadap etika keilmuan dalam karya saya ini, atau ada klaim dari pihak lain terhadap keaslian karya saya ini.

Subang, ……………

Yang membuat pernyataan

Nama Client

# ABSTRAK

# *ABSTRACK*

# KATA PENGANTAR

Puji dan syukur senantiasa penulis panjatkan kehadirat Allah Subhanahu wa ta’ala. Yang telah memberikan nikmat yang sangat luar biasa kepada kita semua, khususnya kepada penulis yang dapat menyelesaikan tugas akhir ini dengan sebaik mungkin. Tugas akhir ini merupakan sebuah penelitian yang dilakukan dalam mengkaji hasil pembelajaran suatu permasalahan yang dapat diidentifikasi ketika dalam pelaksanaan penelitian. Skripsi ini diajukan sebagai salah satu syarat dalam menempuh jenjang sarjana (S1) di Program Studi Sistem Informasi Fakultas Ilmu Komputer Universitas Subang.

Skripsi ini berisi mengenai JUDUL SKRIPSI dengan memanfaatkan teknologi informasi sehingga dapat menyelesaikan permasalahan dan membuat “JUDUL SKRIPSI.”.

Penulis sadar bahwa penulisan skripsi ini masih jauh dari kata sempurna, hal tersebut dikarenakan keterbatasan kemampuan dan pengetahuan penulis. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun dari semua pihak. Walaupun demikian, penulis berharap skripsi dapat memberikan manfaat bagi semua pihak yang berkepentingan.

Subang, …………….

Nama Client

# UCAPAN TERIMA KASIH

Dengan mengucap rasa bersyukur alhamdulilah pembuatan tugas akhir ini dapat terselesaikan dengan baik. Dalam pembuatan tugas akhir ini begitu banyak pihak-pihak yang membantu penulis dalam pembuatan tugas akhir ini.

Tak adil rasanya apabila pihak-pihak yang membantu penulis tidak disebutkan dalam lembaran tugas akhir ini. Untuk itu penulis mengucapkan banyak terimakasih kepada semua pihak yang telah membantu penulis dalam melaksanakan penelitian tugas akhir di Universitas Subang.

Berikut pihak-pihak yang telah membantu penulis dalam melaksanakan penelitian tugas akhir di Universitas Subang.

1. Allah Subhanahu Wa Ta’ala yang telah memberikan ilmu dan kesehatan terhadap penulis dalam mengerjakan tugas akhir ini.
2. Bapak Dr.Drs.Ir.H.A.Moeslihat Komara, M.Si selaku rektor Universitas Subang.
3. Bapak Dr.Tepi Peirisal,S.Sos.,M.Si selaku Dekan Fakultas Ilmu Komputer.
4. Bapak Bagus Ali Akbar,SI.M.Kom selaku ketua Program Studi Sistem Informasi.
5. ……………….. selaku dosen pembimbing pertama atas segala masukan, saran, bimbingan, kesabaran beliau serta waktu yang diberikan sehingga penulis dapat menyelesaikan tugas akhir ini.
6. …………….. selaku dosen pembimbing kedua atas segala masukan, saran, bimbingan, kesabaran beliau serta waktu yang diberikan sehingga penulis dapat menyelesaikan tugas akhir ini.
7. Para Dosen Fakultas Ilmu Komputer yang memberikan masukan terhadap penulis sehingga dapat terselesaikannya tugas akhir ini.
8. …………………. yang telah mengizinkan penulis untuk melaksanakan penelitian.
9. Keluarga tercinta yang telah memberikan dukungan moral ataupun material dan selalu mendoakan penulis dalam menyelesaikan tugas akhir ini.
10. Semua rekan seperjuangan fakultas ilmu komputer angkatan …………….
11. Semua pihak yang telah membantu penulis secara langsung atau tidak langsung dalam menyelesaikan tugas akhir ini.

Semoga Allah Subhanahu Wa Ta’ala membalas semua kebaikan mereka yang telah membantu penulis dalam menyelesaikan tugas akhir ini.

# DAFTAR ISI

[HALAMAN PENGESAHAN i](#_Toc140677538)

[PERNYATAAN ii](#_Toc140677539)

[ABSTRAK iii](#_Toc140677540)

[*ABSTRACK* iv](#_Toc140677541)

[KATA PENGANTAR v](#_Toc140677542)

[UCAPAN TERIMA KASIH vi](#_Toc140677543)

[DAFTAR ISI viii](#_Toc140677544)

[DAFTAR GAMBAR xi](#_Toc140677545)

[DAFTAR TABEL xii](#_Toc140677546)

[DAFTAR LAMPIRAN xix](#_Toc140677547)

[BAB I PENDAHULUAN 1](#_Toc140677548)

[1.1. Latar Belakang 1](#_Toc140677549)

[1.2. Identifikasi Masalah 1](#_Toc140677550)

[1.3. Batasan Masalah 1](#_Toc140677551)

[1.4. Maksud dan Tujuan 1](#_Toc140677552)

[1.4. 1. Maksud Penelitian 1](#_Toc140677553)

[1.4. 2. Tujuan Penelitian 1](#_Toc140677554)

[1.5. Manfaat Penelitian 1](#_Toc140677555)

[1.6. Metodologi Penelitian 1](#_Toc140677556)

[1.6. 1. Tahap Pengumpulan Data 1](#_Toc140677557)

[1.6. 2. Metode Pengembangan Sistem 2](#_Toc140677558)

[1.7. Waktu Penelitian dan Lokasi Penelitian 2](#_Toc140677559)

[1.8. Sistematika Penelitian 2](#_Toc140677560)

[BAB II LANDASAN TEORI 4](#_Toc140677561)

[2.1. Sistem Informasi 4](#_Toc140677562)

[2.2. TEORI TAMBAHAN 4](#_Toc140677563)

[2.3. TEORI TAMBAHAN 4](#_Toc140677564)

[2.4. TEORI TAMBAHAN 4](#_Toc140677565)

[2.5. TEORI TAMBAHAN 4](#_Toc140677566)

[2.6. Metode RUP (*Rational Unified Process*) 4](#_Toc140677567)

[2.6. 1. Fase RUP 4](#_Toc140677568)

[2.7. Website 4](#_Toc140677569)

[*2.7. 1.* *Hypertext Preprocessoer* 4](#_Toc140677570)

[*2.7. 2.* *Framework* 4](#_Toc140677571)

[*2.7. 3.* *Codeigneter* 4](#_Toc140677572)

[2.8. Basis Data 4](#_Toc140677573)

[*2.8. 1.* *Database Management System (DBMS)* 4](#_Toc140677574)

[*2.8. 2.* *MySql* 4](#_Toc140677575)

[*2.9.* *UML (Unified Modeling Language)* 4](#_Toc140677576)

[*2.9. 1.* *Usecase Diagram* 4](#_Toc140677577)

[*2.9. 2.* *Activity Diagram* 5](#_Toc140677578)

[*2.9. 3.* *Roubutness Diagram* 5](#_Toc140677579)

[*2.9. 4.* *Class Diagram* 5](#_Toc140677580)

[*2.9. 5.* *Squence Diagram* 5](#_Toc140677581)

[BAB III ANALISIS KEBUTUHAN SISTEM 6](#_Toc140677582)

[3.1. Profile Perusahaan 6](#_Toc140677583)

[3.1. 1. Visi dan Misi Perusahaan 6](#_Toc140677584)

[3.1. 2. Tugas dan Fungsi 6](#_Toc140677585)

[3.1. 3. Struktur Organisasi 6](#_Toc140677586)

[3.2. Analisa Sistem Berjalan 6](#_Toc140677587)

[3.3. Analisa Kebutuhan Sistem 6](#_Toc140677588)

[3.2. 1. Kebutuhan Informasi 6](#_Toc140677589)

[3.2. 2. Kebutuhan Perangkat Lunak 6](#_Toc140677590)

[3.2. 3. Kebutuhan Perangkat Keras 7](#_Toc140677591)

[3.4. Kebutuhan Fungsional 7](#_Toc140677592)

[3.5. Kebutuhan Non Fungsional 8](#_Toc140677593)

[*3.6.* Definisi *Usecase* 8](#_Toc140677594)

[*3.7.* *Usecase* 9](#_Toc140677595)

[*3.7. 1.* *Skenario Usecase* 9](#_Toc140677596)

[*3.7. 2.* *Activity Diagram* 9](#_Toc140677597)

[*3.8.* *Roubutness Diagram* 9](#_Toc140677598)

[BAB IV PERANCANGAN SISTEM 10](#_Toc140677599)

[4.1. Perancangan Sistem 10](#_Toc140677600)

[4.1. 1. Tujuan Perancangan Sistem 10](#_Toc140677601)

[4.1. 2. Perancangan Prosedur yang Diusulkan 10](#_Toc140677602)

[*4.2.* Perancangan *Class Diagram* 11](#_Toc140677603)

[*4.3.* Perancangan *Squence Diagram* 12](#_Toc140677604)

[4.4. Perancangan Database 12](#_Toc140677605)

[4.5. Relasi Antar Tabel 14](#_Toc140677606)

[4.6. Perancangan Antarmuka 14](#_Toc140677607)

[BAB V IMPLEMENTASI DAN PENGUJIAN 16](#_Toc140677608)

[5.1. Implementasi 16](#_Toc140677609)

[5.1. 1. Implementasi Perangkat Keras 16](#_Toc140677610)

[5.1. 2. Implementasi Perangkat Lunak 16](#_Toc140677611)

[5.2. Implementasi Basis Data 16](#_Toc140677612)

[5.3. Implementasi Antarmuka 17](#_Toc140677613)

[5.3. 1. Implementasi Tampilan Antarmuka 17](#_Toc140677614)

[5.4. Pengujian 18](#_Toc140677615)

[5.4. 1. Rencana Pengujian 18](#_Toc140677616)

[5.4. 2. Kategori Pengujian 18](#_Toc140677617)

[BAB VI KESIMPULAN DAN SARAN 21](#_Toc140677618)

[6.1. Kesimpulan 21](#_Toc140677619)

[6.2. Saran 21](#_Toc140677620)

[DAFTAR PUSTAKA 22](#_Toc140677621)

# DAFTAR GAMBAR

# DAFTAR TABEL

1. *Usecase Diagram*

|  |  |  |  |
| --- | --- | --- | --- |
| NO | GAMBAR | NAMA | KETERANGAN |
| 1 |  | *Actor* | Seseorang atau sesuatu yang berinteraksi dengan sistem yang sedang kita kembangkan. |
| 2 |  | *Dependency* | Umumnya penggunaan *dependency* digunakan untuk menunjukkan operasi pada suatu *class* yang menggunakan *class* yang lain. |
| 3 |  | *Generalization* | Relasi *generalization* sepadan dengan sebuah relasi *inheritance* pada konsep berorientasi objek. |
| 4 |  | *Include* | Relasi cakupan memungkinkan suatu *Use Case* untuk menggunakan fungsionalitas yang disediakan oleh *Use Case* yang lainnya. |
| 5 |  | *Extend* | Memungkinkan suatu *Use Case* memiliki kemungkinan untuk memperluas fungsional yang disediakan *Use Case* yang lainnya. |
| 6 |  | *Association* | Melambangkan tipe-tipe *relationship* dan juga dapat menampilkan hukum-hukum multiplisitas pada sebuah *relationship* (Contoh: *One-to-one, one-to-many, many-to-many*). |
| 7 |  | *System* | Menspesifikasikan paket yang menampilkan sistem secara terbatas. |
| 8 |  | *Use Case* | Peringkat Tertinggi dari fungsional yang dimiliki sistem. |
| 9 |  | *Collaboration* | Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan prilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi). |
| 10 |  | *Note* | Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi. |

1. *Activity Diagram*

|  |  |  |  |
| --- | --- | --- | --- |
| NO | GAMBAR | NAMA | KETERANGAN |
| 1 |  | *Activity* | memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain. |
| 2 |  | *Action* | State dari sistem yang mencerminkan eksekusi dari suatu aksi. |
| 3 |  | *Initial Node* | bagaimana objek dibentuk atau di awali |
| 4 |  | *Activity Final Node* | bagaimana objek dibentuk dan dihancurkan. |
| 5 |  | *Fork Node* | suatu aliran yang pada tahap tertentu berubah menjadi beberapa aliran. |

1. *Roubutness Diagram*

|  |  |  |  |
| --- | --- | --- | --- |
| NO | GAMBAR | NAMA | KETERANGAN |
| 1. |  | Actor | Ini adalah konsep yang sama dengan aktor pada diagram use case UML. |
| 2. |  | *Boundary*  *(Interface)* | Ini mewakili elemen perangkat lunak seperti layar, laporan, halaman HTML, atau antarmuka sistem yang berinteraksi dengan aktor. Juga disebut elemen antarmuka. |
| 3. |  | *Control*  *(Process)* | Ini berfungsi sebagai perekat antara elemen batas dan elemen entitas, menerapkan logika yang diperlukan untuk mengelola berbagai elemen dan interaksinya. Juga dikenal sebagai elemen proses atau hanya sebagai pengontrol. Penting untuk dipahami bahwa Anda dapat memutuskan untuk mengimplementasikan pengontrol dalam desain Anda sebagai sesuatu selain objek - banyak pengontrol yang cukup sederhana untuk diimplementasikan sebagai metode entitas atau kelas batas misalnya. |
| 4. |  | *Entity*  *(Domain)* | Ini adalah tipe entitas yang biasanya ditemukan dalam model konseptual Anda, seperti Student dan Seminar . |
| 5. |  | *Usercase* | Karena kasus penggunaan dapat memanggil kasus penggunaan lain, Anda harus dapat menggambarkannya pada diagram kekokohan Anda. |

1. *Class Diagram*

|  |  |  |  |
| --- | --- | --- | --- |
| NO | GAMBAR | NAMA | KETERANGAN |
| 1 |  | *Generalization* | hubungan dimana objek anak (descendent) berbabagi perilaku dan struktur data dari objek yang ada di atasnya. |
| 2 |  | *Nary Association* | upaya untuk menghindari asosiasi dengan lebih 2 objek. |
| 3 |  | *Class* | himpunan dari objek-objek yang berbagi atribut serta operasi yang sama. |
| 4 |  | *Colaboration* | deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor. |
| 5 |  | *Realization* | operasi yang benar-benar dilakukan oleh suatu objek. |
| 6 |  | *Dependency* | hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (independent) akan mempengaruhi elemen yang bergantung padanyan elemen yang tidak mandiri. |
| 7 |  | *Association* | apa yang menghubungkan antara objek satu dengan objek lainnya. |

1. *Squence Diagram*

|  |  |
| --- | --- |
| SIMBOL | KETERANGAN |
| Aktor  nama aktor  Nama aktor | orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar 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 fase nama aktor. |
| garis hidup/lifeline  l  l | menyatakan kehidupan suatu objek. |
| Objek  Nama objek : nama kelas | menyatakan objek yang berinterakasi pesan. |
| Waktu aktif | menyatakan objek yang dalam keadaan aktif dan berinteraksi pesan. |
| pesan tipe *call*  1 : nama\_metode() | menyatakan suatu objek memanggil operasi/ metode yang ada pada objek lain atau dirinya sendiri,  1: nama\_metode()  arah panah mengarah pada objek yang memiliki operasi/metode, karena ini memanggil operasi/metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi. |
| Pesan tipe *sen*  1 : masukan | menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim. |
| Pesan tipe *return*  1 : keluaran  - - - - - - - - - - -> | menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian. |

# DAFTAR LAMPIRAN

# BAB I PENDAHULUAN

## Latar Belakang

## Identifikasi Masalah

## Batasan Masalah

## Maksud dan Tujuan

### Maksud Penelitian

### Tujuan Penelitian

## Manfaat Penelitian

## Metodologi Penelitian

### Tahap Pengumpulan Data

Penulis melakukan kajian dengan menggunakan metode penelitian, pengumpulan data dan menganalisis secara langsung ……………….., adapun metodologi yang digunakan adalah sebagai berikut :

1. Metode wawancara

Metode ini dilakukan dengan cara melakukan tanya jawab secara tatap muka dengan pegawai yang mengelola data yang terkait

1. Metode observasi

Metode ini dilakukan dengan cara mengamati langsung mengenai ………………………...

1. Studi pustaka

Studi pustaka adalah metode pengumpulan data dengan mencari informasi lewat pencarian melalui berbagai situs di internet, buku, majalah, koran, dan literatur lainnya yang bertujuan untuk membentuk sebuah landasan teori maupun untuk memecahkan berbagai masalah penelitian.

### Metode Pengembangan Sistem

Dalam pengembangan sistem adalah metode *RUP* (*Rational* *Unified Process*). Pengembangan sistem dengan metode *RUP* dilakukan secara *iterative* (berulang) dan *incremental* (bertahap dengan progres menarik). *RUP* memiliki empat tahap yaitu *inception*, *elaboration*, *construction*, dan *transition.*

## Waktu Penelitian dan Lokasi Penelitian

## Sistematika Penelitian

BAB I: Pendahuluan

Pada bab ini dibahas tentang latar belakang masalah, rumusan masalah, batasan masalah, tujuan masalah, maksud dan tujuan masalah, manfaat penelitian , metode penelitian,metodologi pengembangan sistem, jadwal penelitian dan sistematika penulisan.

BAB II: Landasan Teori

Pada bab ini dibahas mengenai teori dasar yang mendukung penulisan skripsi, mencakup metode atau teknik yang digunakan, teori tentang permasalahan, uraian singkat perangkat implementasi yang dipakai, dan kerangka penyelesaian masalah.

BAB III: Analisis Kebutuhan Sistem

Pada bab ini dibahas tentang hasil pendefinisian kebutuhan dari permasalahan yang dijadikan topik skripsi.

BAB IV: Perancangan

Pada bab ini berisi penjelasan mengenai perancangan perangkat lunak, perancangan *database*, perancangan antarmuka serta perancangan perangkat lunak yang akan dibangun.

BAB V: Implementasi dan Pengujian

Bab ini berisi penjelasan tentang perancangan implementasi berdasarkan pada hasil perancangan dan pengujian program aplikasi atau kinerja Sistem Informasi.

BAB VI: Kesimpulan dan Saran

Berisi kesimpulan dan saran dari sistem yang dibangun tentang kasus Tugas Akhir atau Skripsi.

# BAB II LANDASAN TEORI

## Sistem Informasi

## TEORI TAMBAHAN

## TEORI TAMBAHAN

## TEORI TAMBAHAN

## TEORI TAMBAHAN

## Metode RUP (*Rational Unified Process*)

### Fase RUP

## Website

### *Hypertext Preprocessoer*

### *Framework*

### *Codeigneter*

## Basis Data

### *Database Management System (DBMS)*

### *MySql*

## *UML (Unified Modeling Language)*

### *Usecase Diagram*

Menurut Satzinger dalam Triandini dan I Gede Suartika (2012:17) mendeskripsikan bahwa, “*Use Case* adalah sebuah kegiatan yang dilakukan oleh sistem, biasanya dalam menaggapi permintaan dari pengguna sistem”.

Menurut Rosa A.S dan M. Shalahuddin (2014:155) mendefinisikan bahwa, “*Use* *case* atau diagram *use* *case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat”.

Dari beberapa pendapat diatas penulis menyimpulkan *Use* *Case Diagram* adalah kegiatan interaksi yang saling berkaitan antara sistem dan aktor.

### *Activity Diagram*

Menurut Rosa A.S dan M. Shalahuddin (2014:161) mendeskripsikan bahwa, “*Activity* Diagram adalah menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis”.

Menurut Nugroho (2010:62) mendeskripsikan bahwa, “Diagram aktivitas (*Activity* *Diagram*) sesungguhnya merupakan bentuk khusus dari state machine yang bertujuan memodelkan komputasi-komputasi dan aliran-aliran kerja yang terjadi dalam sistem/perangkat lunak yang sedang dikembangkan”.

Berdasarkan beberapa pendapat diatas penulis menyimpulkan, *Activity Diagram* adalah menggambarkan aliran aktivitas dalam sistem yang sedang dirancang.

### *Roubutness Diagram*

*Robustness* adalah kemampuan sebuah sistem komputer dalam menghadapi error ketika eksekusi program sedang dijalankan. Penerapan robustness ini bisa termasuk dalam hal apa saja, salah satu contohnya adalah algoritma sebuah program.

### *Class Diagram*

Menggambarkan struktur objek sistem. Diagram ini menunjukan *class object* yang menyusun sistem dan juga hubungan antara *class object* tersebut.

### *Squence Diagram*

Secara grafis menggambarkan bagaimana objek berinteraksi dengan satu sama lain melalui pesan pada sekuensi sebuah *use case* atau operasi.

# BAB III ANALISIS KEBUTUHAN SISTEM

## Profile Perusahaan

### Visi dan Misi Perusahaan

### Tugas dan Fungsi

### Struktur Organisasi

## Analisa Sistem Berjalan

## Analisa Kebutuhan Sistem

### Kebutuhan Informasi

Kebutuhan informasi menjelaskan kondisi atau kemampuan sistem sesuai dengan spesifikasi yang diinginkan oleh pemakai, meliputi kebutuhan informasi, kebutuhan aplikasi atau proses pengolahan data untuk menghasilkan informasi tersebut dan kebutuhan perangkat keras.

|  |  |  |
| --- | --- | --- |
| **No** | **Aktor** | **Informasi yang dibutuhkan** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

### Kebutuhan Perangkat Lunak

Kebutuhan Perangkat Lunak mendeskripsikan sistem secara umum dan menjelaskan kebutuhan-kebutuhan dari aplikasi yang dibangun.

### Kebutuhan Perangkat Keras

Kebutuhan Perangkat Keras merupakan Komponen fisik yang memiliki spesifik atau kriteria tertentu agar dapat menjalankan sistem dengan baik. Kebutuhan perangkat keras yang dibutuhkan untuk menjalankan sistem sebagai berikut.

## Kebutuhan Fungsional

Kebutuhan Fungsional merupakan kebutuhan inti dari sistem dimana sistem yang akan dibuat harus memenuhi kebutuhan yang telah ditentukan dari hasil observasi dan pengindentifikasian dari sebuah masalah yang di angkat topiknya.

|  |  |
| --- | --- |
| **No SRS** | **Deskripsi** |
| **AKTOR** | |
| SRS-F-001 |  |
| SRS-F-002 |  |
| SRS-F-003 |  |
| SRS-F-004 |  |
| SRS-F-005 |  |
| **AKTOR** | |
| SRS-F-006 |  |
| SRS-F-007 |  |
| SRS-F-008 |  |
| **AKTOR** | |
| SRS-F-009 |  |
| SRS-F-010 |  |

## Kebutuhan Non Fungsional

**K**ebutuhan Non Fungsional adalah kebutuhan pendukung atau kebutuhan pelengkap dari suatu sistem yang akan dibuat.

|  |  |
| --- | --- |
| **NO SRS** | **Deskripsi** |
| SRS-NF-500 | Sistem ini dirancang menggunakan *CodeIgneter* dan Mysql |
| SRS-NF-600 | Sistem ini menggunakan Visual studio code sebagai Text editor |

## Definisi *Usecase*

|  |  |
| --- | --- |
| **No *Usecase*** | **Deskripsi** |
| **AKTOR** | |
| UC-01 |  |
|  |  |
|  |  |
|  |  |
|  |  |
| **AKTOR** | |
|  |  |
|  |  |
|  |  |
| **AKTOR** | |
|  |  |
|  |  |
|  |  |
|  |  |

## *Usecase*

### *Skenario Usecase*

1. *Usecase …….*
2. *Usecase …….*
3. *Usecase …….*
4. *Usecase …….*
5. *Usecase …….*
6. *Usecase …….*
7. *Usecase …….*
8. *Usecase …….*

### *Activity Diagram*

1. *Activity Diagram ……..*
2. *Activity Diagram ……..*
3. *Activity Diagram ……..*
4. *Activity Diagram ……..*
5. *Activity Diagram ……..*
6. *Activity Diagram ……..*

## *Roubutness Diagram*

# BAB IV PERANCANGAN SISTEM

## Perancangan Sistem

Perancangan menghasilkan suatu model implementasi atau penggambaran sistem informasi yang akan dibuat untuk memenuhi kebutuhan-kebutuhan informasi untuk pemakai yang sudah di definisikan pada tahap analisis kebutuhan.

Sistem yang dirancang berupa aplikasi berbasis web yang akan membantu dalam proses ……………..

### Tujuan Perancangan Sistem

Tujuan perancangan adalah memberikan gambaran secara *detail* sistem yang akan dibangun meliputi perancangan *Interface*, *Database* dan Perancangan arsitektur perangkat lunak sehingga memberikan spesifikasi lengkap kepada pengguna.

### Perancangan Prosedur yang Diusulkan

Dari hasil analisis yang telah dilakukan sebelumnya, maka terdapat solusi dengan merancang prosedur baru yaitu membuat sebuah …………………. yang berfungsi untuk memastikan kegiatan …………. sesuai dengan prosedur sebagai berikut :

1. **AKTOR** mengelola **MENU** sesuai hak akses.
2. **AKTOR** mengelola **MENU** sesuai hak akses.
3. **AKTOR** mengelola **MENU** sesuai hak akses.
4. **AKTOR** mengelola **MENU** sesuai hak akses.
5. **AKTOR** mengelola **MENU** sesuai hak akses.

## Perancangan *Class Diagram*

## Perancangan *Squence Diagram*

*Sequence Diagram* biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respon dari sebuah aktivitas untuk menghasilkan output. Berikut ini adalah *Sequence Diagram* skenario …………………………..

1. Rancangan *Squnece Diagram …………*
2. Rancangan *Squnece Diagram …………*
3. Rancangan *Squnece Diagram …………*
4. Rancangan *Squnece Diagram …………*
5. Rancangan *Squnece Diagram …………*
6. Rancangan *Squnece Diagram …………*
7. Rancangan *Squnece Diagram …………*
8. Rancangan *Squnece Diagram …………*
9. Rancangan *Squnece Diagram …………*
10. Rancangan *Squnece Diagram …………*
11. Rancangan *Squnece Diagram …………*

## Perancangan Database

Berikut ini adalah perancangan basis data pada …………………………….

1. Tabel

Nama Tabel :

Fungsi : untuk mengelola

Primary Key :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field Name | Data Type | Length | PK | Not Nul | A.I |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

1. Tabel

Nama Tabel :

Fungsi : untuk mengelola

Primary Key :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field Name | Data Type | Length | PK | Not Nul | A.I |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

1. Tabel

Nama Tabel :

Fungsi : untuk mengelola

Primary Key :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field Name | Data Type | Length | PK | Not Nul | A.I |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

1. Tabel

Nama Tabel :

Fungsi : untuk mengelola

Primary Key :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field Name | Data Type | Length | PK | Not Nul | A.I |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

1. Tabel

Nama Tabel :

Fungsi : untuk mengelola

Primary Key :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field Name | Data Type | Length | PK | Not Nul | A.I |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

## Relasi Antar Tabel

## Perancangan Antarmuka

Perancangan antarmuka merupakan tampilan atau desain dari sistem yang akan di bangun dan bertujuan untuk menyampaikan informasi yang dibutuhkan oleh pengguna.

1. Antarmuka …………..
2. Antarmuka …………..
3. Antarmuka …………..
4. Antarmuka …………..
5. Antarmuka …………..
6. Antarmuka …………..
7. Antarmuka …………..
8. Antarmuka …………..

# BAB V IMPLEMENTASI DAN PENGUJIAN

## Implementasi

Implementasi merupakan tahapan dimana hasil perancangan diterapkan kedalam bahasa pemrograman berdasarkan perancangan yang telah dilakukan sebelumnya. Pada proses implementasi ini dijelaskan mengenai sistem yang dirancang dan bagaimana cara penggunaannya. Pada tahap ini dilakukan sebagai akhir dari desain ………………….. Setelah dilakukannya implementasi maka tahap selanjutnya yaitu pengujian terhadap sistem yang di bangun sehingga sistem siap digunakan.

### Implementasi Perangkat Keras

Perangkat keras yang akan di gunakan dalam mengimplementasikan ……………. adalah sebagai berikut :

### Implementasi Perangkat Lunak

Perangkat Lunak yang akan di gunakan dalam mengimplementasikan ………………….. adalah sebagai berikut :

## Implementasi Basis Data

Pembuatan *Database* dilakukan dengan menggunakan aplikasi *MySQL*, berikut adalah implementasi basis data dari …………….. :

Nama basis data :

1. **Database keseluruhan**
2. **Tabel…………**
3. **Tabel…………**
4. **Tabel…………**
5. **Tabel…………**
6. **Tabel…………**
7. **Tabel…………**

## Implementasi Antarmuka

|  |  |  |
| --- | --- | --- |
| **Menu** | **Deskripsi** | **Nama File** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

### Implementasi Tampilan Antarmuka

Untuk memperjelas implementasi tampilan antarmuka berikut adalah tampilan antarmuka …………………….. :

1. Antarmuka …………………
2. Antarmuka …………………
3. Antarmuka …………………
4. Antarmuka …………………
5. Antarmuka …………………
6. Antarmuka …………………

## Pengujian

Pengujian perangkat lunak ini menggunakan metode *blackbox* *testing* yaitu menguji perangkat lunak dari segi spesifikasi fungsionalitas tanpa menguji desain dan kode program. Pengujian dilakukan untuk mengetahui apakah fungsi-fungsi, masukan dan keluaran dari perangkat lunak sesuai dengan yang dibutuhkan.

### Rencana Pengujian

Sebelum melakukan pengujian ada tahapan yang dilakukan sebelumnya, yakni dengan membuat rencana penguji. Adapun rencana pengujian yang dibuat adalah sebagai berikut :

|  |  |  |  |
| --- | --- | --- | --- |
| **No SRS** | **Nama *Usecase*** | **Butir Uji** | **Kasus Uji** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

### Kategori Pengujian

Kategori keberhasilan pengujian dalam perangkat lunak pada sistem yang telah memenuhi dan sesuai dengan hasil yang diharapkan :

* + - * 1. **Berhasil**

Kategori berhasil jika dalam sebuah pengujian kasus uji yang diuji telah memenuhi dan sesuai dengan yang diharapkan.

* + - * 1. **Tidak Berhasil**

Kategori tidak jika dalam sebuah pengujian kasus uji tidak sesuai dengan hasil yang diharapkan.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **UAT** | **No. SRS** | **Deskripsi** | **Berhasil** | **Tidak** |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |
|  |  |  | √ |  |

# BAB VI KESIMPULAN DAN SARAN

## Kesimpulan

## Saran

# DAFTAR PUSTAKA