**SKPL-002**

SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

*Monitoring Business*

untuk:

<PT Asahimas Chemical>

Dipersiapkan oleh:

Tim IT Kelompok 2

IF–41–02

Program Studi S1 Teknik Informatika – Fakultas Informatika

Universitas Telkom

Jalan Telekomunikasi Terusan Buah Batu, Bandung

Indonesia

|  |  |  |  |
| --- | --- | --- | --- |
| Program Studi S1 Teknik Informatika  -  Fakultas Informatika | Nomor Dokumen | | Halaman |
| *SKPL-002* | |  |
| Revisi | *<nomor revisi>* | *Tgl: <isi tanggal>* |
|  | 1 | *A* | 28/03/2019 |
|  | 2 | *B* | 11/04/2019 |
|  | 3 | *C* | 25/04/2019 |
|  | 4 | *D* | 03/05/2019 |

# Daftar Perubahan

|  |  |
| --- | --- |
| Revisi | Deskripsi |
| A | Perubahan pada bagian use case scenario |
| B | Perubahan pada kebutuhan fungsional,kebutuhan non-fungsional |
| C | Perubahan pada bab 2 dan 4 |
| D | Perubahan pada bab 1 |
| E |  |
| F |  |
| G |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| INDEX | - | A | B | C | D | E | F | G |
| TGL |  | 28/03/2019 | 11/04/2019 | 25/04/2019 | 03/05/2019 |  |  |  |
| Ditulis oleh |  | Satria Amien Thoat | Budi Nugroho | Firdausi Nuzula Zamzami | Intan Maharani |  |  |  |
| Diperiksa oleh |  | Atik Zilziana | Atik  Zilziana | Bu Widowati | Bu Widowati |  |  |  |
| Disetujui oleh |  | Atik Zilziana | Atik  Zilziana | Bu Widowati | Bu Widowati |  |  |  |

# Daftar Halaman Perubahan

|  |  |  |  |
| --- | --- | --- | --- |
| Halaman | Revisi | Halaman | Revisi |
| 17-30  13-14  12-14 dan 32-33  10-11 | Perubahan pada bagian use case scenario  Perubahan pada kebutuhan fungsional,kebutuhan non-fungsional  Perubahan pada bab 2 dan 4  Perubahan pada bab 1 |  |  |

# Daftar Isi

[Daftar Perubahan 3](#_Toc8782963)

[Daftar Halaman Perubahan 5](#_Toc8782964)

[Daftar Isi 6](#_Toc8782965)

[Daftar Tabel 8](#_Toc8782966)

[Daftar Gambar 9](#_Toc8782967)

[1. Pendahuluan 10](#_Toc8782968)

[1.1 Tujuan Penulisan Dokumen 10](#_Toc8782969)

[1.2 Ruang Lingkup / Cakupan Dokumen 10](#_Toc8782970)

[1.3 Definisi, Singkatan, dan Akronim 11](#_Toc8782971)

[1.4 Referensi 11](#_Toc8782972)

[2. Deskripsi Global Perangkat Lunak 12](#_Toc8782973)

[2.1 Statement of Objective Perangkat Lunak 12](#_Toc8782974)

[2.2 Perspektif dan Fungsi Perangkat Lunak 12](#_Toc8782975)

[2.3 Profil dan Karakteristik Pengguna 12](#_Toc8782976)

[Table 1. Profil dan Otoritas User. 13](#_Toc8782977)

[2.4 Lingkungan Operasi 13](#_Toc8782978)

[2.5 Batasan Perangkat Lunak / Sistem 13](#_Toc8782979)

[2.6 Asumsi dan Dependensi 14](#_Toc8782980)

[3. Deskripsi Rinci Perangkat Lunak 15](#_Toc8782981)

[3.1 Deskripsi Kebutuhan 15](#_Toc8782982)

[3.1.1 Kebutuhan Fungsional 15](#_Toc8782983)

[Table 2. Kebutuhan Fungsional 15](#_Toc8782984)

[3.1.2 Kebutuhan Non-Fungsional 16](#_Toc8782985)

[Table 3. Kebutuhan Non-Fungsional 16](#_Toc8782986)

[3.2 Pemodelan Analisis 17](#_Toc8782987)

[3.2.1 Usecase Diagram 17](#_Toc8782988)

[GAMBAR 1. Use Case Diagram. 17](#_Toc8782989)

[3.2.1.1 Usecase Scenario #1 17](#_Toc8782990)

[Table 4. Use case scenario registrasi. 17](#_Toc8782991)

[3.2.1.2 Usecase Scenario #2 19](#_Toc8782992)

[Table 5. Use case scenario login. 19](#_Toc8782993)

[3.2.1.3 Usecase Scenario #3 20](#_Toc8782994)

[Table 6. Use case scenario kelola data owner. 20](#_Toc8782995)

[3.2.1.4 Use Case Scenario #4 23](#_Toc8782996)

[Table 7. Use case scenario kelola data cabang. 23](#_Toc8782997)

[3.2.1.5 Usecase Scenario #5 25](#_Toc8782998)

[Table 8. Use case scenario kelola arus kas. 25](#_Toc8782999)

[3.2.1.6 Usecase Scenario #6 27](#_Toc8783000)

[Table 9. Use case scenario lihat hasil. 27](#_Toc8783001)

[3.2.1.7 Usecase Scenario #7 30](#_Toc8783002)

[Table 10. Use case scenario input data keuangan. 30](#_Toc8783003)

[3.2.2 Class Diagram: 31](#_Toc8783004)

[GAMBAR 2. Class diagram. 32](#_Toc8783005)

[4. Kebutuhan Antarmuka Eksternal 32](#_Toc8783006)

[4.1 Antarmuka Pengguna 32](#_Toc8783007)

[4.2 Antarmuka Perangkat Keras 33](#_Toc8783008)

[4.3 Antarmuka Perangkat Lunak 33](#_Toc8783009)

[4.4 Antarmuka Komunikasi 33](#_Toc8783010)

[5. Requirements Lain 34](#_Toc8783011)

[GAMBAR 3. Entity Relationship Diagram. 35](#_Toc8783012)

[GAMBAR 4. Flow Map. 36](#_Toc8783013)

# Daftar Tabel

[Table 1. Profil dan Otoritas User. 12](#_Toc8781997)

[Table 2. Kebutuhan Fungsional 14](#_Toc8781998)

[Table 3. Kebutuhan Non-Fungsional 15](#_Toc8781999)

[Table 4. Use case scenario registrasi. 16](#_Toc8782000)

[Table 5. Use case scenario login. 18](#_Toc8782001)

[Table 6. Use case scenario kelola data owner. 19](#_Toc8782002)

[Table 7. Use case scenario kelola data cabang. 22](#_Toc8782003)

[Table 8. Use case scenario kelola arus kas. 24](#_Toc8782004)

[Table 9. Use case scenario lihat hasil. 26](#_Toc8782005)

[Table 10. Use case scenario input data keuangan. 29](#_Toc8782006)

# Daftar Gambar

[GAMBAR 1. Use Case Diagram. 17](#_Toc8782031)

[GAMBAR 2. Class diagram. 32](#_Toc8782032)

[GAMBAR 3. Entity Relationship Diagram. 35](#_Toc8782033)

[GAMBAR 4. Flow Map. 36](#_Toc8782034)

# Pendahuluan

Dokumen ini berisi penjelasan pemakaian dan penulisan dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) untuk aplikasi Monitoring Business. Dokumen ini selanjutnya akan menggunakan istilah SKPL. Dan dokumen ini dapat digunakan bagi pengembang perangkat lunak sebagai acuan teknis pengembangan perangkat lunak pada versi selanjutnya.

## Tujuan Penulisan Dokumen

Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) merupakan spesifikasi

kebutuhan perangkat lunak yang akan dikembangkan. Dokumen ini digunakan oleh

pengembang perangkat lunak sebagai acuan teknis pengembangan perangkat lunak

pada tahap selanjutnya sesuai dengan kebutuhan yang diinginkan oleh pengguna dan tujuan perangkat lunak itu sendiri. Dokumen ini juga digunakan oleh pengguna dalam mencatat semua spesifikasi kebutuhan dari perangkat lunak yang dikembangkan dan harapan yang diinginkan oleh pengguna.

## Ruang Lingkup / Cakupan Dokumen

*Monitoring Business* adalah perangkat lunak untuk me*monitoring* kegiatan dan keuangan bisnis dalam satu usaha untuk banyak cabang. Perangkat lunak ini dibuat menggunakan prinsip *Object Oriented* yang didalamnya terdapat fitur registrasi, login/logout, dan menu (menu pencarian, menu laporan, menu statistik bisnis perbulan). Sasaran pengguna yang dituju adalah pembisnis yang memiliki beberapa cabang dan kesulitan dalam memantau dan mengolah data bisnis nya dalam berbagai cabang yang ia punya, lebih tepatnya PT Asahimas Chemical.

Ruang lingkup dalam pembangunan aplikasi *Monitoring Business,* antara lain :

1. Owner melakukan registrasi untuk membuat akun di aplikasi ini.
2. Bila user sudah memiliki akun, user dapat melakukan login ke dalam aplikasi ini.
3. Owner dapat mengelola informasi bisnisnya dan mengelola data owner.
4. Owner dapat mengelola data owner.
5. User dapat mengelola data cabang.
6. Owner mendaftarkan akun untuk setiap cabang yang ia punya.
7. Setiap cabang menginput data keuangan ke sistem informasi Monitoring Business.
8. Owner dapat memberikan saran pengelolaan arus kas bagi masing-masing cabang yang ia miliki berdasarkan laporan keuangan yang diterima dari setiap cabang.
9. Owner dapat melihat statistik bulanan setiap cabangnya, data laba rugi, dan data arus kas.
10. Sistem mengolah perhitungan laba rugi dan Owner dapat melihat hasil perhitungan laba rugi tersebut.
11. Cabang dapat mengelola data cabangnya
12. Jika Owner sudah selesai monitoring, owner bisa logout dari aplikasi.

## Definisi, Singkatan, dan Akronim

Monitoring Business adalah sebuah aplikasi untuk memonitoring kegiatan dan keuangan bisnis dalam satu usaha untuk banyak cabang.

Berikut adalah daftar definisi dan istilah penting yang digunakan dalam dokumen SKPL ini :

*● SRS : Software Requirements Specification*

* *SKPL :* Spesifikasi Kebutuhan Perangkat Lunak adalahdokumen yang berisi spesifikasi kebutuhan perangkat lunak.
* *IEEE : Institute of Electrical and Electronics Engineering* adalah Standar internasional untuk pengembangan dan perancangan produk.
* DFD : *Data Flow Diagram* adalah suatu diagram yang menggunakan notasi-notasi untuk menggambarkan arus dari data sistem yang penggunanya sangat membantu untuk memahami sistem secara logika, terstruktur dan jelas.
* ERD : *Entity Relationship Diagram* adalahdiagram dan notasi yang digunakan untuk merepresentasikan struktur data statis pada perangkat lunak.

## Referensi

* *Slide* perkuliahan APPL
* [www.google.com](http://www.google.com)
* <https://manajemenkeuangan.net/contoh-sop-pengelolaan-keuangan/>
* <https://moeslimar.files.wordpress.com/2014/01/skpl-sinm.pdf>

# Deskripsi Global Perangkat Lunak

## Statement of Objective Perangkat Lunak

Perangkat lunak *Monitoring Business* merupakan perangkat lunak yang berfungsi memudahkan pembisnis atau suatu perusahaan dalam memonitong kegiatan bisnisnya. Owner dapat melihat peluang dari setiap cabangnya yang paling menguntungkan. Peluang tersebut dapat dilihat dari statistik bulanan dan saran pada setiap cabangnya. Dari setiap cabang yang dimiliki, owner dapat mengelola segala data keuangan baik pengeluaran maupun pendapatan secara detail.

## Perspektif dan Fungsi Perangkat Lunak

Perangkat lunak *Monitoring Business* ini merupakan perangkat lunak berbasis aplikasi web yang dapat digunakan untuk mempermudah pengelolaan keuangan bisnis dan me*monitoring* bisnis dari beberapa cabang setiap bulannya. Aplikasi web kami membutuhkan localhost, MySQL dan HTML 5.

Adapun fungsi-fungsi yang dimiliki oleh perangkat lunak ini adalah :

1. Owner

* Melakukan registrasi akun
* Melakukan login untuk memonitoring bisnisnya
* Mengelola data owner yang berisi informasi bisnis dan akun owner
* Mendaftarkan cabang yang dimiliki dan mengelola data cabang
* Melihat hasil monitoring bisnis setiap cabang baik statistik bulanan, data laba rugi, dan data arus kas.
* Melihat hasil laba rugi yang sudah melewati proses perhitungan yang dilakukan oleh sistem.
* Mengelola arus kas dan memasukkan saran untuk masing-masing bisnisnya sesuai laporan keuangan yang diterima.

2. Cabang

* Melakukan login untuk melakukan kegiatan pengiriman data dan informasi yang berhubungan dengan bisnis.
* Mengelola data cabangnya.
* Menginput data keuangan setiap bulannya ke sistem.

## Profil dan Karakteristik Pengguna

Pengguna aplikasi ini adalah seluruh pembisnis yang ingin me*monitoring* kegiatan dan keuangan bisnis dalam satu usaha untuk banyak cabang secara online. Owner adalah masyarakat yang melakukan interaksi dengan aplikasi dan siap untuk melakukan kegiatan pengelolaan keuangan bisnis dan *monitoring* cabangnya. Cabang adalah member yang telah didaftarkan oleh owner dan siap untuk mengelola segala data keuangan baik pengeluaran maupun pendapatan secara detail.

#### Table 1. Profil dan Otoritas User.

|  |  |
| --- | --- |
| *User* | *Otoritas* |
| *Owner* | * Registrasi * Login * Kelola data owner * Kelola data cabang * Lihat hasil * Perhitungan laba rugi * Mengelola arus kas dan memasukkan saran untuk masing-masing bisnisnya sesuai laporan keuangan yang diterima. |
| *Cabang* | * Login * Kelola Data Cabang * Input Data Keuangan |

## Lingkungan Operasi

Minimum requirement platform perangkat keras dari pengoperasian perangkat lunak berbasis aplikasi web yaitu device yang mampu untuk running web browser serta memerlukan akses localhost menggunakan PHP web server yang memerlukan XAMPP untuk menjalankan servernya, HTML 5 sebagai bahasa pemograman untuk membangun web, dan MySQL untuk menyimpan database.

## Batasan Perangkat Lunak / Sistem

Batasan-batasan yang digunakan pada pengembangan perangkat lunak ini adalah :

* Kesimpulan bisnis hanya akan akurat jika laporan yang diinputkan oleh masing masing cabang sesuai.
* Owner hanya bisa memonitoring maksimal 5 cabang usaha.
* Setiap cabang hanya bisa dimiliki oleh satu owner.
* Owner harus terdaftar sebagai member di aplikasi ini.
* Owner yang mendaftarkan cabang.
* Laporan pada web otomatis terupdate setelah input data keuangan di proses Penginputan Data Keuangan.
* Aplikasi ini untuk owner yang sudah memiliki bisnis.
* Owner harus memiliki minimal 2 cabang.
* Statistik keuangan tiap bulan akan selalu tercatat secara akurat dan selalu update.
* Data diri dan data cabang harus sesuai dengan keadaan.
* Owner hanya bisa menginputkan data diri, data cabang dan melihat laporan yang sudah diolah sistem(laporan laba rugi, laporan arus kas, dan statistik keuangan setiap bulannya).

## Asumsi dan Dependensi

Perangkat lunak ini hanya bisa dipakai pada device seperti PC yang terdapat Web Browser yang dapat melakukan akses web. Berubahnya spesifikasi dan aturan penggunaan perangkat lunak ini harus disetujui oleh pemilik perangkat lunak.

# Deskripsi Rinci Perangkat Lunak

## Deskripsi Kebutuhan

### Kebutuhan Fungsional

#### Table 2. Kebutuhan Fungsional

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Kode Kebutuhan | Fungsi | Deskripsi |
| 1. | FR-01 | Registrasi | Owner dapat mendaftarkan akun bisnisnya pada aplikasi Monitoring Business. |
| 2. | FR-02 | Login | User dapat melakukan login setelah owner melakukan registrasi dan Cabang telah didaftarkan oleh Ownernya. |
| 3. | FR-03 | Kelola data owner | Owner menginputkan informasi bisnis yang dimiliki, dan dapat mengelola akun bisnisnya. |
| 4. | FR-04 | Kelola data cabang | * Owner mendaftarkan cabang yang dimiliki dan mengelola seluruh data cabang yang dimiliki * Cabang dapat mengelola data cabang sesuai id cabangnya. |
| 5. | FR-05 | Perhitungan laba rugi | Sistem melakukan perhitungan laba rugi sesuai laporan keuangan yang dikirim oleh cabang dan hasil perhitungan laba rugi dilihat oleh Owner |
| 6. | FR-06 | Kelola arus kas | Owner melihat hasil perhitungan arus kas yang diproses oleh sistem dan memberikan saran untuk masing-masing cabang sesuai dengan laporan keuangan yang diterima Owner setiap bulannya. |
| 7. | FR-07 | Lihat hasil | Owner melihat laporan laba rugi, laporan arus kas, dan statistik keuangan setiap bulannya |
| 8 | FR-08 | Input data keuangan | Cabang menginputkan data keuangan setiap bulannya ke sistem dan akan diterima oleh Owner |

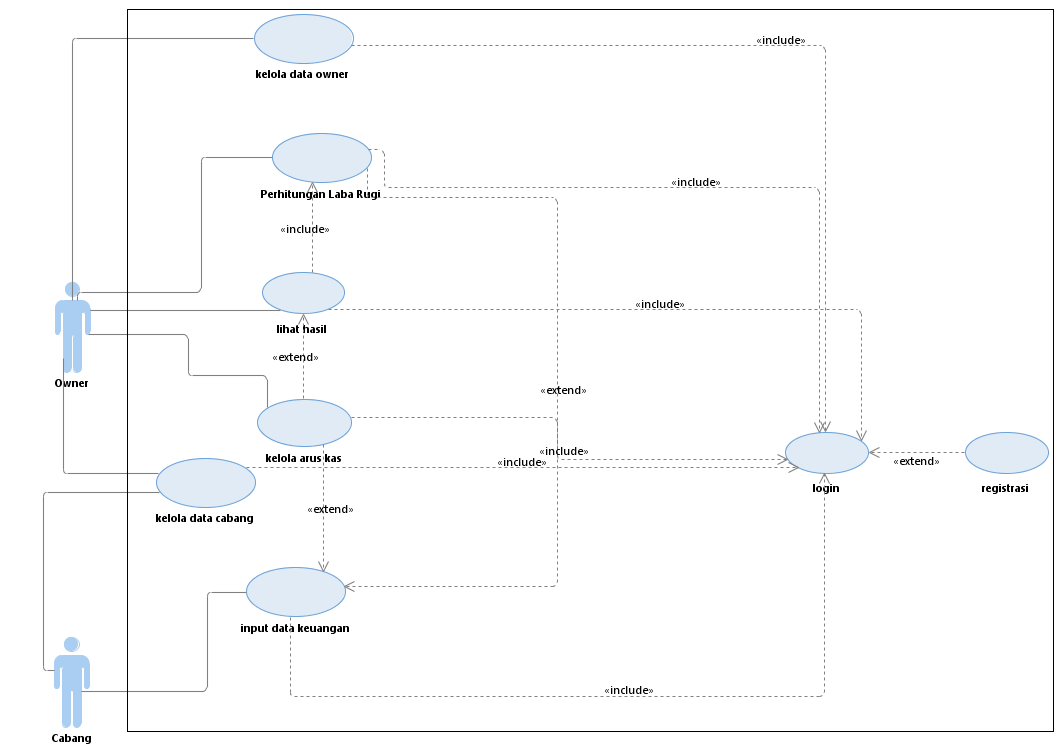
### Kebutuhan Non-Fungsional

#### Table 3. Kebutuhan Non-Fungsional

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Quality | Kode Kebutuhan | Deskripsi |
| 1. | Security | NFR-01 | Fungsi ini digunakan untuk keamanan data dan aplikasi. |
| 2. | Performansi | NFR-02 | Sistem tersedia setiap hari nya tanpa henti dan respon cepat. |
| 3. | Akurasi | NFR-03 | Owner mendapatkan rekomendasi bisnis yang akurat sesuai data yang sudah diolah berdasarkan inputan cabang. |
| 4. | Safety | NFR-05 | Fungsi ini digunakan sebagai tolak ukur bahwa aplikasi Monitoring Business tidak membahayakan fisik user dalam menggunakan aplikasi ini. |

## Pemodelan Analisis

### Usecase Diagram



#### GAMBAR 1. Use Case Diagram.

#### Usecase Scenario #1

#### Table 4. Use case scenario registrasi.

|  |  |  |
| --- | --- | --- |
| Nama Use Case | Registrasi | |
| Deskripsi | Fungsi ini digunakan oleh Owner untuk melakukan registrasi akun owner dan cabang | |
| Pre-Kondisi | Owner belum terdaftar  Owner berada pada menu registrasi | |
| Post-Kondisi | Owner telah terdaftar dan data register owner tersimpan di database Owner | |
| Skenario Utama |  | |
|  | Owner | Sistem |
|  | 1. Sistem menampilkan Login page |
| 2. Owner mengeklik link register here |  |
|  | 3. Sistem menampilkan form register |
| 4. Owner melengkapi form register |  |
| 5. Owner mengeklik tombol daftar. |  |
|  | 6. Data diri register owner tersimpan di dalam database owner |
| Skenario Eksepsional (Alternative flow) |  | |
|  | Owner | Sistem |
|  | 1. Jika username sudah ada, tampil notif username doesn’t exist. |
|  | 2. Jika konfirmasi password tidak sesuai, tampil notif password lemah. |
|  | 3. Jika email tidak valid, tampil notif maaf email anda tidak valid. |

#### Usecase Scenario #2

#### Table 5. Use case scenario login.

|  |  |  |
| --- | --- | --- |
| Nama Use Case | Login | |
| Deskripsi | Fungsi ini digunakan oleh owner atau cabang untuk menginputkan username dan password sehingga owner atau cabang bisa menggunakan sistem *Monitoring Business*. | |
| Pre-Kondisi | Owner atau Cabang belum login | |
| Post-Kondisi | Owner atau Cabang berhasil login dan menuju halaman page cabang atau owner | |
| Skenario Utama |  | |
|  | Owner, Cabang | Sistem |
| 1. User membuka monitoring bisnis |  |
| 2. user berada pada halaman login, lalu menginputkan username dan password. |  |
| 3. User mengeklik tombol login. |  |
|  | 4.sistem memproses data yang di input |
|  | 5. Sistem melakukan eksekusi validasi login user dengan data yang tersimpan di dalam database. |
|  | 6. sistem mengarahkan ke landing page owner atau cabang. |
| 7. cabang/owner berada pada landing page owner atau cabang. |  |
| Skenario Eksepsional (Alternative flow) |  | |
|  | Owner, Cabang | Sistem |
| 1. User submit dengan memasukkan username dan password | 2. inputan tidak sesuai dengan database, maka muncul notif gagal login. |

#### Usecase Scenario #3

#### Table 6. Use case scenario kelola data owner.

|  |  |  |
| --- | --- | --- |
| Nama Use Case | Kelola Data Owner | |
| Deskripsi | Fungsi ini digunakan oleh owner untuk mengelola data akun dan data bisnisnya | |
| Pre-Kondisi | * Owner sudah melakukan registrasi akun * Owner sudah melakukan login * Owner berada di page kelola dataOwner * Sistem menampilkan page kelola data Owner * Data Owner tersimpan di database | |
| Post-Kondisi | Owner menambahkan dataOwner  Owner menghapus dataOwner  Owner melihat dataOwner  Owner mengedit dataOwner  Database Owner terisi dan terupdate | |
| Skenario Utama |  | |
|  | Owner | Sistem |
|  | 1. Sistem menampilkan landing page keloladataOwner dengan mengaktifkan controller kelola data Owner |
| 2. Owner mengeklik tampilan Kelola data Owner pada navbar |  |
|  | 3. Sistem menampilkan menu Kelola dataOwner |
| 4. Jika Owner ingin menambahkan data Owner maka owner harus mengeklik tombol tambah |  |
|  |  | 5. Sistem menampilkan form tambah data dan owner harus mengisi form tambah dataOwner dan memilih tombol input dan data tersimpan di database dataOwner |
|  | 6. Jika Owner ingin menghapus data owner maka owner harus mengeklik tombol hapus |  |
|  |  | 7. Sistem akan mengaktifkan controller cari data hapus |
|  | 8. Owner memasukkan nama bisnis yang ingin dihapus |  |
|  |  | 9. Sistem akan menghapus data Owner yang tersimpan dengan mengaktifkan controller hapus dan data di dalam database dataOwner terhapus |
|  |  | 10. Jika owner memilih tombol view maka sistem akan mengaktifkan controller view dan menampilkan data yang telah terisi dan terupdate di dalam database dataOwner |
|  | 11. Jika Owner ingin mengedit data Owner maka Owner harus memilih tombol edit dan menginputkan nama bisnis yang ingin diedit |  |
|  |  | 12. Sistem akan mengaktifkan controller cari data edit |
|  | 13. Owner melengkapi dataOwner dengan mengisi form edit dataOwner |  |
|  |  | 14. Sistem akan mengupdate data dengan mengaktifkan controller edit dan dataOwner yang telah teredit tersimpan di database dataOwner |
|  |  | 15. Sistem akan memvalidasi data yang telah diinputkan, diedit, dihapus oleh Owner |
|  |  | 16. Sistem menyimpan data yang sudah dipilih oleh owner tersimpan di dalam database |
| Skenario Eksepsional (Alternative flow) |  | |
|  | Owner | Sistem |
| 1. Jika Owner memilih tombol button silang |  |
|  | 2. Sistem akan menredirect ke landing page Owner |
|  | 3. Jika Owner telah menambahkan data maka sistem akan mengirimkan pesan “Data Owner berhasil ditambahkan” |
|  | 4. Jika Owner ingin menghapus dataOwner maka sistem akan mengirimkan pesan”Apakah anda ingin menghapus data!” |
|  | 5. Jika data berhasil dihapus maka sistem akan mengirimkan pesan “Data berhasil dihapus” |
|  | 6. Jika Owner ingin mengedit dataOwner maka sistem akan mengirimkan pesan “apakah anda ingin mengedit data!” |
|  | 7. Jika data sudah berhasil diedit maka sistem akan mengirim pesan “Data berhasil diedit” |

#### Use Case Scenario #4

#### Table 7. Use case scenario kelola data cabang.

|  |  |  |
| --- | --- | --- |
| Nama Use Case | Kelola Data Cabang | |
| Deskripsi | Fungsi ini digunakan oleh owner dan cabang untuk mengelola data cabangnya. | |
| Pre-Kondisi | * Cabang atau Owner sudah melakukan login * Cabang atau Owner berada pada menu kelola data cabang masing-masing. * Owner sudah mendaftarkan bisnisnya * Owner dapat mendaftarkan cabangnya selama cabang masih kurang dari 5 | |
| Post-Kondisi | Data cabang tersimpan dalam database. | |
| Skenario Utama |  | |
|  | Cabang,Owner | Sistem |
| 1. Owner berada pada menu kelola data cabang milik owner |  |
| 2. Owner dapat memilih button view, delete, add, edit pada kelola data cabang |  |
|  | 3. Sistem menampilkan halaman berdasarkan button yang dipilih owner |
| 4. User meng-klik tombol submit |  |
|  |  | 6. sistem memproses data yang telah di input |
|  |  | 7. Data cabang tersimpan dalam database |
| Skenario Eksepsional (Alternative flow) |  | |
|  | Owner,Cabang | Sistem |
|  | 1. Jika cabang klik tombol back pada halaman edit dan view maka sistem akan menampilkan halaman kelola data cabang cabang. |
|  |  | 2.jika Owner klik tombol back pada halaman edit, view, delete, dan add, lalu sistem akan menuju halaman kelola data cabang owner. |
|  |  | 3. cabang hanya bisa mengakses button atau halaman view & edit |

#### Usecase Scenario #5

#### Table 8. Use case scenario kelola arus kas.

|  |  |  |
| --- | --- | --- |
| Nama Use Case | Kelola Arus Kas | |
| Deskripsi | Fungsi ini digunakan oleh owner untuk melakukan segala klarifikasi segala hal mengenai arus kas yang didapat dari setiap cabangnya. | |
| Pre-Kondisi | * Owner sudah melakukan login * Owner berada pada menu keuangan * Database keuangan sudah terisi | |
| Post-Kondisi | * saran pengelolaan arus kas masuk ke database * owner melihat arus kas cabang yang dipilih | |
| Skenario Utama |  | |
|  | Owner |  |
|  | 1. Sistem menampilkan Menu Keuangan Page |
| 2. Owner mengeklik button kelola arus |  |
|  | 3. Sistem memproses kelola arus kas |
|  | 4. Sistem mengambil data cabang dari database cabang |
|  | 5. Sistem mengambil data keuangan dari database keuangan |
|  | 6 Sistem menampilkan Kelola Keuangan Page |
| 7 Owner klik kelola |  |
|  | 8. Sistem mengambil id Cabang yang diklik owner |
|  | 9. Sistem mengambil id keuangan yang diklik owner |
|  | 10. Sistem mengambil data keuangan berdasarkan id cabang dan id keuangan dari database keuangan |
|  | 11. Sistem menampilkan Saran Cabang Page |
| 12. Owner menuliskan sarannya dan klik submit |  |
|  | 13. Sistem melanjutkan proses pemberian saran |
|  | 14. Sistem menyimpan saran pengelolaan ke database arus kas |
|  | 15. Sistem menampilkan Sukses Page |
| 16. Owner melihat sarannya sudah berhasil tersimpan |  |
| Skenario Eksepsional (Alternative flow) |  | |
|  | Owner | Sistem |
|  | 1. Jika owner ingin mengedit data perhitungan arus kas, owner mengeklik tombol yang bergambar pena |
|  | 2. Jika owner ingin menghapus data perhitungan arus kas, owner mengeklik tombol yang bergambar silang |
|  | 3. Jika owner mengeklik tombol cancel maka sistem akan menampilkan halaman landing page |

#### Usecase Scenario #6

#### Table 9. Use case scenario lihat hasil.

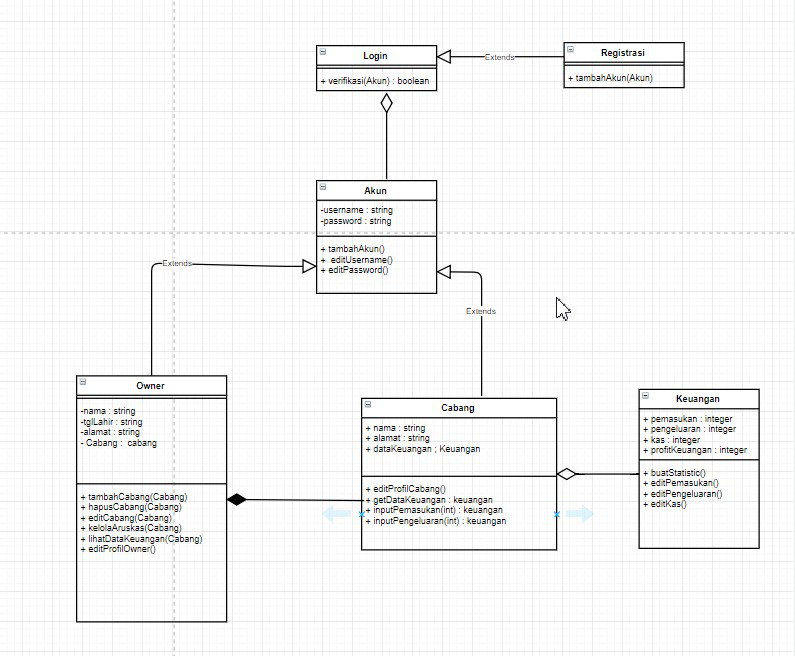
|  |  |  |
| --- | --- | --- |
| Nama Use Case | Lihat Hasil | |
| Deskripsi | Fungsi ini digunakan oleh owner untuk melihat hasil keuangan bisnis yang ada disetiap cabangnya. | |
| Pre-Kondisi | * Owner sudah melakukan login * Owner sudah berada pada menu lihat hasil * Laporan keuangan sudah ada di database keuangan | |
| Post-Kondisi | Owner melihat data keuangan cabang yang dipilih  Laba Rugi Cabang disimpan dalam database Laba Rugi  Owner melihat statistik keuangan perbulannya | |
| Skenario Utama |  | |
|  | Owner |  |
|  | 1.Menu Keuangan Controller/sistem mengaktifkan Menu Lihat Data Page |
| 2. Owner klik lihat data cabang button |  |
|  | 3. Menerima action lihat data cabang klik |
|  | 4. Sistem menampilkan menu cari cabang |
| 5. Owner memasukan nama cabang lalu klik submit |  |
|  |  | 6. Sistem mendapatkan mendapatkan inputan nama cabang |
|  |  | 7. Sistem mengambil data cabang dari database cabang sesuai nama cabang inputan user |
|  |  | 8. Sistem memverifikasi nama cabang yang diinputkan |
|  |  | 9. Sistem menerima Id Cabang dari nama cabang inputan user |
|  |  | 10. Sistem mengambil data keuangan cabang dari database cabang dan database keuangan |
|  |  | 11 Sistem melakukan perhitungan laba rugi |
|  |  | 12. Sistem menyimpan data laba rugi ke dalam database laba rugi |
|  |  | 13. Sistem menampilkan data cabang beserta laba ruginya |
|  | 14. Owner melihat data keuangan cabang beserta laba ruginya |  |
| Skenario Eksepsional (Alternative flow) |  | |
|  | Owner | Sistem |
| 1. Owner klik cancel |  |
|  | 2. Sistem mengembalikan owner ke menu lihat hasil |
|  | 3. Jika Owner salah menginputkan nama cabang maka sistem akan menampilkan notif “data yang di inputkan salah” |
| 4. Owner klik lihat semua data |  |
|  | 5. Sistem menampilkan semua data cabang beserta keuangan dan laba rugi yang diambil dari use case perhitungan arus kas |

#### Usecase Scenario #7

#### Table 10. Use case scenario input data keuangan.

|  |  |  |
| --- | --- | --- |
| Nama Use Case | Input Data Keuangan | |
| Deskripsi | Fungsi ini digunakan oleh cabang untuk menginputkan data keuangan cabangnya setiap bulan. | |
| Pre-Kondisi | Cabang sudah melakukan login as “Cabang” | |
| Post-Kondisi | Owner mendapat alert “data keuangan cabang ... telah diterima” dan dapat dilihat pada menu lihat hasil | |
| Skenario Utama |  | |
|  | Cabang | Sistem |
|  | 1. Sistem menampilkan landing page |
| 2. Cabang mengeklik menu input data keuangan |  |
| 3. Cabang mengunggah file data keuangan ke sistem |  |
| 4. Cabang mengeklik tombol submit |  |
|  |  | 5. Sistem menyimpan data keuangan dalam database dan otomatis terkirim ke Owner |
| Skenario Eksepsional (Alternative flow) |  | |
|  | Owner | Sistem |
|  | 1. Data form yang tidak terisi, sistem akan menampilkan highlight required dibawah kolom form yang tidak diisi |
|  | 2. File yang diinputkan berukuran maks 10mb, dan berformat doc atau docs, bila Cabang menginputkan file diatas cakupan tersebut atau tidak sesuai format, sistem akan menampilkan alert “maaf, data yang anda masukkan terlalu besar atau format yang and unggah salah” |

### Class Diagram:



#### GAMBAR 2. Class diagram.

# Kebutuhan Antarmuka Eksternal

## Antarmuka Pengguna

Pemakai berinteraksi dengan sistem perangkat lunak *Monitoring Business* melalui antarmuka pemakai. Perangkat lunak ini berbasis web yang dapat diakses melalui smartphone maupun desktop. Sistem perangkat lunak ini akan menerima masukan data keuangan setiap bulannya yang berasal dari setiap cabangnya sesuai keadaan saat itu, kemudian owner akan menerima output data yang telah dikelola sistem berupa file laporan laba rugi, laporan arus kas, dan statistik bisnis bulanan.

## Antarmuka Perangkat Keras

Minimum requirement perangkat keras yang dibutuhkan:

● Device yang mampu untuk running web browser

## Antarmuka Perangkat Lunak

Minimum requirement perangkat lunak yang dibutuhkan:

*●* HTML 5

* PHP WebServer yang dijalankan dengan localhost XAMPP
* MySQL

## Antarmuka Komunikasi

Minimum requirement untuk mengakses Sistem *Monitoring Business* :

● Jaringan 3G,4G dan stabil

● SSL

● HTTP

# Requirements Lain

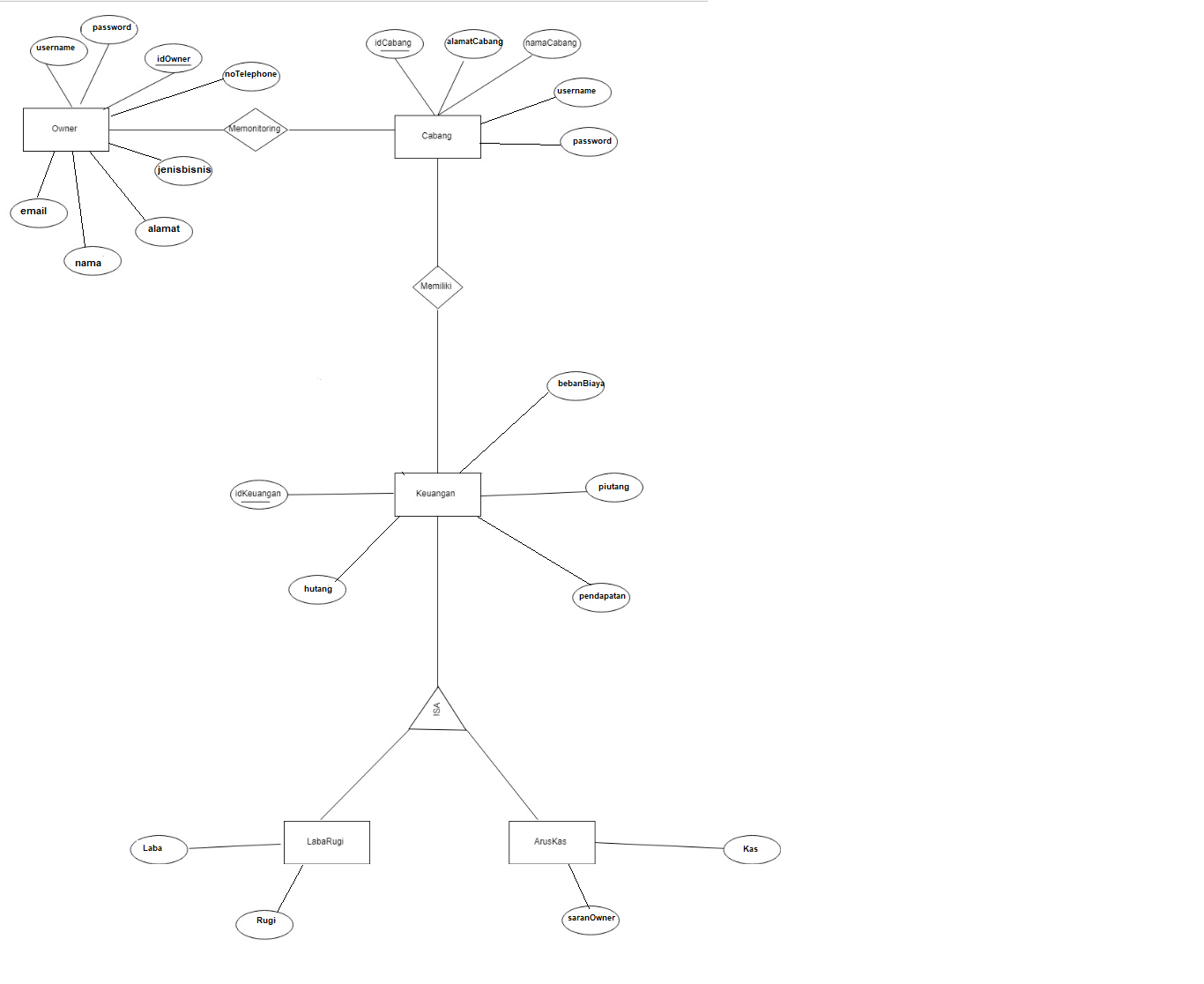
**Lampiran A: Daftar Kata-Kata Sukar**

Berikut adalah daftar kata-kata sukar yang digunakan dalam dokumen SKPL ini :

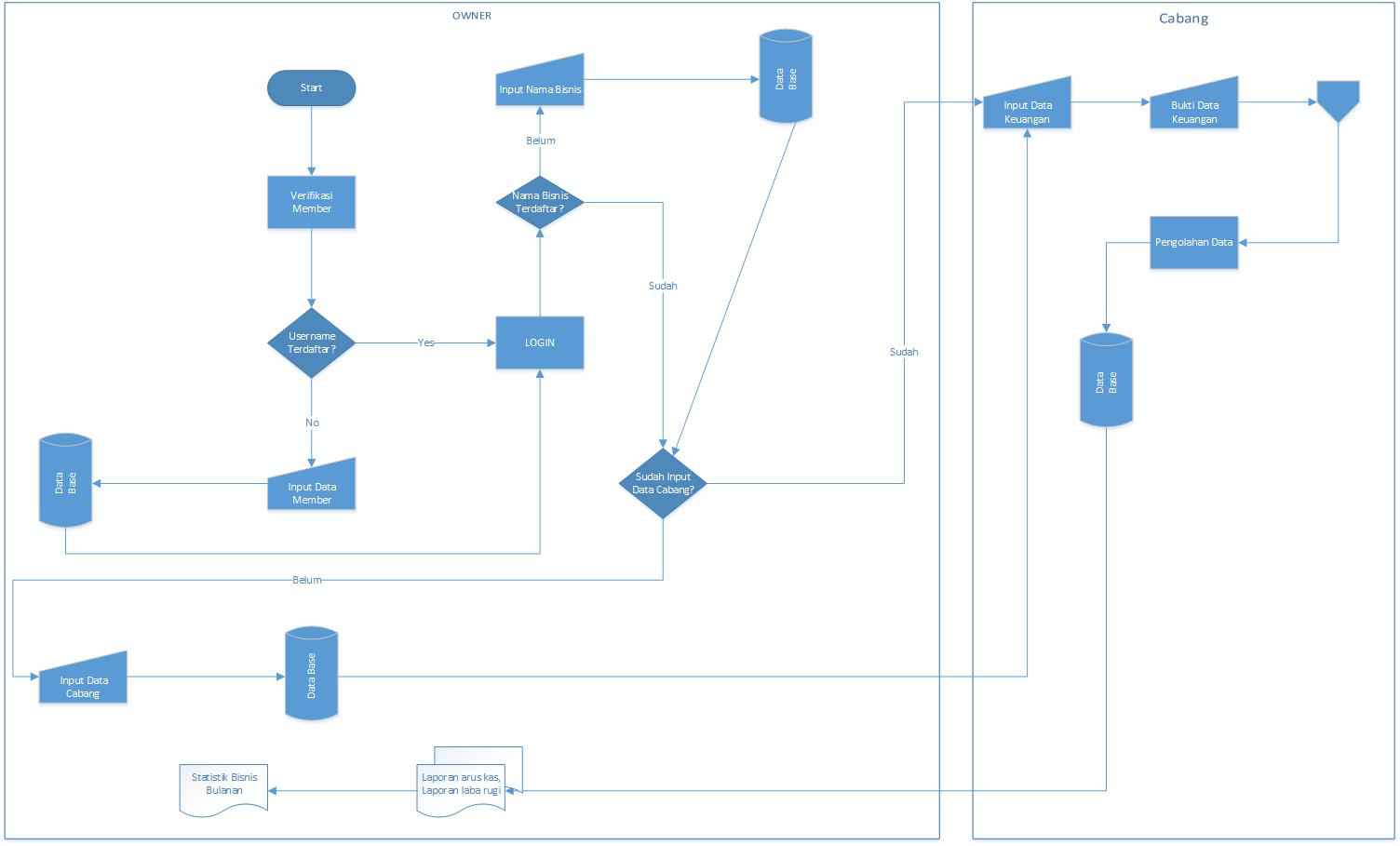
*● SRS : Software Requirements Specification*

* *SKPL :* Spesifikasi Kebutuhan Perangkat Lunak adalahdokumen yang berisi spesifikasi kebutuhan perangkat lunak.
* *IEEE : Institute of Electrical and Electronics Engineering* adalah Standar internasional untuk pengembangan dan perancangan produk.
* DFD : *Data Flow Diagram* adalah suatu diagram yang menggunakan notasi-notasi untuk menggambarkan arus dari data sistem yang penggunanya sangat membantu untuk memahami sistem secara logika, terstruktur dan jelas.
* ERD : *Entity Relationship Diagram* adalahdiagram dan notasi yang digunakan untuk merepresentasikan struktur data statis pada perangkat lunak.

**Lampiran B: Analysis Models**



#### GAMBAR 3. Entity Relationship Diagram.



#### GAMBAR 4. Flow Map.