

## **USULAN TUGAS AKHIR**

### **RANCANG BANGUN SISTEM INFORMASI ASSESSMENT RISIKO TAHANAN BERBASIS WEB MENGGUNAKAN METODE RATIONAL UNIFIED PROCESS (STUDI KASUS : LEMBAGA PEMASYARAKATAN KELAS IIB SELONG)**



Oleh :

**ANGGIN RISNA WIDIANINGSIH**

**F1D 016 010**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS TEKNIK  
UNIVERSITAS MATARAM  
2020**

## USULAN TUGAS AKHIR

### RANCANG BANGUN SISTEM INFORMASI ASSESSMENT RISIKO TAHANAN BERBASIS WEB MENGGUNAKAN METODE RATIONAL UNIFIED PROCESS (STUDI KASUS : LEMBAGA PEMASYARAKATAN KELAS IIB SELONG)

Telah diperiksa dan disetujui oleh Tim Pembimbing :

1. Pembimbing Utama



**Nadiyah Agitha, S.Kom., M.M.T.**  
**NIP: 19860813 201803 2 001**

Tanggal: 27 April 2020

2. Pembimbing Pendamping



**Royana Afwani, S.T., M.T.**  
**NIP: 19850707 201404 2 001**

Tanggal: 25 April 2020

Mengetahui,  
Ketua Program Studi Teknik Informatika  
Fakultas Teknik  
Universitas Mataram



**Prof. Dr. Eng. I Gede Pasek Suta Wijaya, S.T., M.T.**  
**NIP: 19731130 200003 1 001**

## USULAN TUGAS AKHIR

### RANCANG BANGUN SISTEM INFORMASI ASSESSMENT RISIKO TAHANAN BERBASIS WEB MENGGUNAKAN METODE RATIONAL UNIFIED PROCESS (STUDI KASUS : LEMBAGA PEMASYARAKATAN KELAS IIB SELONG)

Oleh:

**ANGGIN RISNA WIDIANINGSIH**  
**F1D016010**

Telah diperiksa dan disetujui oleh Tim Penguji :

1. Penguji I



**Moh. Ali Albar, S.T., M.Eng.**  
**NIP: 19831125 201504 1 002**

Tanggal: 25 April 2020

2. Penguji II



**Ir. Sri Endang Anjarwani, M.Kom.**  
**NIP: 19660403 200604 2 001**

Tanggal: 27 April 2020

3. Penguji III



**Andy Hidayat Jatmika, S.T., M.Kom.**  
**NIP: 19831209 201212 1 001**

Tanggal: 25 April 2020

Mengetahui,  
Ketua Program Studi Teknik Informatika  
Fakultas Teknik  
Universitas Mataram



**Prof. Dr. Eng. I Gede Pasek Suta Wijaya, S.T., M.T.**  
**NIP: 19731130 200003 1 001**

## DAFTAR ISI

|                                                             |      |
|-------------------------------------------------------------|------|
| HALAMAN SAMPUL .....                                        |      |
| LEMBAR PENGESAHAN .....                                     | ii   |
| DAFTAR ISI.....                                             | iv   |
| DAFTAR GAMBAR .....                                         | vi   |
| DAFTAR TABEL .....                                          | viii |
| ABSTRAK .....                                               | ix   |
| BAB I PENDAHULUAN .....                                     | 1    |
| 1.1 Latar Belakang .....                                    | 1    |
| 1.2 Rumusan Masalah .....                                   | 3    |
| 1.3 Batasan Masalah.....                                    | 3    |
| 1.4 Tujuan Penelitian.....                                  | 4    |
| 1.5 Manfaat Penelitian.....                                 | 4    |
| 1.6 Sistematika Penulisan.....                              | 5    |
| BAB II TINJAUAN PUSTAKA DAN LANDASAN TEORI .....            | 6    |
| 2.1 Tinjauan Pustaka .....                                  | 6    |
| 2.2 Dasar Teori.....                                        | 10   |
| 2.2.1 Sejarah Lembaga Pemasyarakatan Kelas IIB Selong ..... | 10   |
| 2.2.2 Visi dan Misi .....                                   | 11   |
| 2.2.3 Struktur Organisasi.....                              | 11   |
| 2.2.4 Alur <i>Assessment</i> .....                          | 11   |
| 2.2.5 Sistem Informasi .....                                | 12   |
| 2.2.6 RUP .....                                             | 12   |
| 2.2.7 UML .....                                             | 16   |
| 2.2.7.1 <i>Use Case Diagram</i> .....                       | 16   |
| 2.2.7.2 <i>Entity Relationship Diagram (ERD)</i> .....      | 17   |
| 2.2.7.3 <i>Class Diagram</i> .....                          | 19   |
| 2.2.7.4 <i>Sequence Diagram</i> .....                       | 19   |
| 2.2.7.5 <i>Activity Diagram</i> .....                       | 20   |
| 2.2.8 <i>Code Igniter</i> .....                             | 21   |
| 2.2.9 XAMPP .....                                           | 21   |
| 2.2.10 <i>Bussiness Modelling Canvas</i> .....              | 21   |

|                                                   |    |
|---------------------------------------------------|----|
| 2.2.11 ISO 2510 .....                             | 23 |
| BAB III METODE PENELITIAN.....                    | 26 |
| 3.1 Metode Pengembangan Sistem .....              | 26 |
| 3.1.1 Pengumpulan Data .....                      | 27 |
| 3.1.1.1 Observasi .....                           | 27 |
| 3.1.1.2 Wawancara .....                           | 27 |
| 3.1.2 Studi Literatur .....                       | 28 |
| 3.1.3 Pengembangan Sistem dengan metode RUP ..... | 28 |
| 3.1.3.1 Tahap <i>Inception</i> .....              | 28 |
| 3.1.3.2 Tahap <i>Elaboration</i> .....            | 36 |
| 3.1.3.3 Tahap <i>Construction</i> .....           | 73 |
| 3.1.3.4 Tahap <i>Transition</i> .....             | 73 |
| DAFTAR PUSTAKA .....                              | 76 |
| LAMPIRAN.....                                     | 78 |

## DAFTAR GAMBAR

|                                                                                           |    |
|-------------------------------------------------------------------------------------------|----|
| Gambar 2.1 <i>Fishbone</i> diagram .....                                                  | 9  |
| Gambar 2.2 Struktur Organisasi Lapas Selong .....                                         | 10 |
| Gambar 2.3 Arsitektur <i>rational unified process</i> .....                               | 12 |
| Gambar 2.4 Simbol entitas.....                                                            | 17 |
| Gambar 2.5 Simbol relasi.....                                                             | 17 |
| Gambar 2.6 Simbol atribut.....                                                            | 18 |
| Gambar 3.1 Tahapan pengembangan sistem .....                                              | 25 |
| Gambar 3.2 <i>Business modelling</i> sistem infomasi <i>assessment</i> risiko tahanan ... | 28 |
| Gambar 3.3 <i>Use case diagram</i> sistem.....                                            | 33 |
| Gambar 3.4 Proses <i>login</i> .....                                                      | 36 |
| Gambar 3.5 Proses menambahkan <i>user</i> .....                                           | 36 |
| Gambar 3.6 Mengubah <i>user</i> .....                                                     | 37 |
| Gambar 3.7 Menghapus <i>user</i> .....                                                    | 37 |
| Gambar 3.8 Mencetak laporan bulanan .....                                                 | 38 |
| Gambar 3.9 <i>Activity diagram</i> proses mengubah <i>profile</i> .....                   | 38 |
| Gambar 3.10 Proses melakukan <i>assessment</i> .....                                      | 39 |
| Gambar 3.11 <i>Class diagram</i> .....                                                    | 40 |
| Gambar 3.12 Proses <i>login</i> Kalapas, Kasi, Kasubs, dan assessor .....                 | 41 |
| Gambar 3.13 Proses menambah <i>user</i> .....                                             | 42 |
| Gambar 3.14 Proses mengubah <i>user</i> .....                                             | 43 |
| Gambar 3.15 Proses menghapus <i>user</i> .....                                            | 44 |
| Gambar 3.16 Proses memverifikasi hasil <i>assessment</i> .....                            | 44 |
| Gambar 3.17 Proses melakukan <i>assessment</i> .....                                      | 45 |
| Gambar 3.18 Proses mengubah data <i>assessment</i> .....                                  | 46 |
| Gambar 3.19 Proses melihat laporan bulanan.....                                           | 46 |
| Gambar 3.20 Proses mencetak laporan bulanan .....                                         | 47 |
| Gambar 3.21 Proses melihat grafik bulanan .....                                           | 48 |
| Gambar 3.22 Desain <i>database</i> .....                                                  | 49 |
| Gambar 3.23 Rancangan tampilan halaman <i>login</i> .....                                 | 56 |
| Gambar 3.24 Rancangan tampilan halaman beranda admin.....                                 | 56 |
| Gambar 3.25 Rancangan tampilan halaman data <i>user</i> .....                             | 57 |

|                                                                      |    |
|----------------------------------------------------------------------|----|
| Gambar 3.26 Rancangan tampilan halaman tambah <i>user</i> .....      | 57 |
| Gambar 3.27 Rancangan tampilan halaman edit <i>user</i> .....        | 58 |
| Gambar 3.28 Rancangan tampilan halaman hapus <i>user</i> .....       | 58 |
| Gambar 3.29 Rancangan tampilan halaman data tahanan .....            | 59 |
| Gambar 3.30 Rancangan tampilan halaman tambah data tahanan.....      | 59 |
| Gambar 3.31 Rancangan tampilan halaman edit data tahanan .....       | 60 |
| Gambar 3.32 Rancangan tampilan hapus data tahanan.....               | 60 |
| Gambar 3.33 Rancangan tampilan halaman beranda Kalapas .....         | 61 |
| Gambar 3.34 Rancangan tampilan menekan tombol <i>profile</i> .....   | 61 |
| Gambar 3.35 Rancangan tampilan halaman mengubah <i>profile</i> ..... | 62 |
| Gambar 3.36 Rancangan tampilan halaman kelola assessor.....          | 62 |
| Gambar 3.37 Rancangan tampilan halaman melihat data tahanan .....    | 63 |
| Gambar 3.38 Rancangan tampilan halaman laporan kasi .....            | 63 |
| Gambar 3.39 Rancangan tampilan halaman cetak data .....              | 64 |
| Gambar 3.40 Rancangan tampilan halaman grafik Kalapas .....          | 64 |
| Gambar 3.41 Rancangan tampilan halaman beranda kasi .....            | 65 |
| Gambar 3.42 Rancangan tampilan halaman laporan kasi .....            | 65 |
| Gambar 3.43 Rancangan tampilan halaman grafik kasi.....              | 66 |
| Gambar 3.44 Rancangan tampilan halaman beranda kasubsi .....         | 66 |
| Gambar 3.45 Rancangan tampilan halaman verifikasi .....              | 67 |
| Gambar 3.46 Rancangan tampilan halaman laporan kasubsi.....          | 67 |
| Gambar 3.47 Rancangan tampilan halaman grafik kasubsi .....          | 68 |
| Gambar 3.48 Rancangan tampilan halaman beranda assessor.....         | 68 |
| Gambar 3.49 Rancangan tampilan halaman <i>assessment</i> .....       | 69 |
| Gambar 3.50 Rancangan tampilan halaman <i>assessment</i> .....       | 69 |
| Gambar 3.51 Rancangan tampilan halaman <i>assessment</i> .....       | 70 |
| Gambar 3.52 Rancangan tampilan halaman <i>assessment</i> .....       | 70 |
| Gambar 3.53 Rancangan tampilan halaman laporan assessor .....        | 71 |
| Gambar 3.54 Rancangan tampilan halaman grafik assessor .....         | 71 |

## DAFTAR TABEL

|                                                                             |    |
|-----------------------------------------------------------------------------|----|
| Tabel 2.1 Tabel simbol <i>use case diagram</i> .....                        | 16 |
| Tabel 2.2 Tabel relasi <i>class diagram</i> .....                           | 18 |
| Tabel 2.3 Tabel simbol <i>sequence diagram</i> .....                        | 19 |
| Tabel 2.4 Tabel simbol <i>activity diagram</i> .....                        | 19 |
| Tabel 2.5 Kriteria interpretasi skor .....                                  | 24 |
| Tabel 3.1 Rincian biaya.....                                                | 29 |
| Tabel 3.2 Kebutuhan fungsional sistem .....                                 | 30 |
| Tabel 3.3 Kebutuhan non-fungsional sistem .....                             | 32 |
| Tabel 3.4 Kebutuhan perangkat keras.....                                    | 32 |
| Tabel 3.5 Kebutuhan perangkat lunak .....                                   | 32 |
| Tabel 3.6 Validasi kesesuaian <i>use case</i> dengan kebutuhan sistem ..... | 34 |
| Tabel 3.7 Tabel <i>user</i> .....                                           | 49 |
| Tabel 3.8 Tabel <i>assesment</i> .....                                      | 50 |
| Tabel 3.9 Tabel melakukan.....                                              | 50 |
| Tabel 3.10 Tabel tahanan .....                                              | 51 |
| Tabel 3.11 Tabel riwayat hukum .....                                        | 51 |
| Tabel 3.12 Tabel data fisik.....                                            | 51 |
| Tabel 3.13 Tabel catatan .....                                              | 52 |
| Tabel 3.14 Tabel <i>assessment</i> khusus.....                              | 52 |
| Tabel 3.15 Tabel terdapat.....                                              | 52 |
| Tabel 3.16 Tabel gejala .....                                               | 53 |
| Tabel 3.17 Tabel memiliki .....                                             | 53 |
| Tabel 3.18 Tabel keluarga.....                                              | 53 |
| Tabel 3.19 Tabel keluarga umum.....                                         | 54 |
| Tabel 3.20 Tabel keluarga inti .....                                        | 54 |
| Tabel 3.21 Tabel identitas .....                                            | 54 |
| Tabel 3.22 Tabel interaksi sosial.....                                      | 55 |
| Tabel 3.23 Validasi fitur sistem .....                                      | 72 |

## ABSTRAK

Lembaga Pemasyarakatan Kelas IIB Selong masih menggunakan sistem manual untuk melakukan *assessment* risiko terhadap tahanan sehingga menjadikan proses nya lebih lama, beresiko terjadi kesalahan inputan nilai dan harus dilakukan penghitungan kembali untuk memastikan apakah hasil yang diperoleh sudah sesuai atau tidak. Pada penelitian ini, dirancang dan dibangun sebuah Sistem Informasi *Assessment* Risiko Tahanan Berbasis *Website* di Lembaga Pemasyarakatan Kelas IIB Selong yang dapat membantu mengklasifikasikan tahanan berdasarkan tingkat risiko rendah, menengah, atau tinggi sehingga dapat menghasilkan perhitungan dan informasi yang sesuai dengan pedoman *assessment*. Untuk mendukung pengembangan *website*, digunakan metode *Rational Unified Process* (RUP) yang merupakan metode pengembangan perangkat lunak yang berorientasi objek. Metode *Rational Unified Process* (RUP) dipilih karena dapat menjelaskan pengembangan sistem secara *real*, memiliki dokumentasi sistem yang jelas dan memiliki fleksibilitas dalam proses pengembangan.

**Kata Kunci :** Metode *Rational Unified Process*, RUP, Sistem Informasi, Lembaga Pemasyarakatan

# **BAB 1**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Pelaksanaan sistem pemasyarakatan saat ini memiliki banyak tantangan yang harus dihadapi oleh jajaran petugas pemasyarakatan. Hal ini disebabkan oleh adanya perubahan strategis baik dalam aspek jenis kejahatan, tingkat pengulangan pidana, tuntutan masyarakat dan proses penegakan hukum yang semakin diharapkan lebih baik implementasinya. Berdasarkan data yang diperoleh dari Sistem *Database* Pemasyarakatan jumlah penghuni Lembaga Pemasyarakatan (Lapas) dan Rumah Tahanan (Rutan) di Indonesia pada tahun 2012 hingga tahun 2018 yaitu sejumlah 248.462 penghuni. Jumlah penghuni tersebut lebih besar dua kali lipat dari kapasitas ideal yang dapat ditampung yaitu sejumlah 125.145 penghuni atau persentase jumlah penghuni lapas terhadap kapasitas adalah 199%. Berdasarkan data tersebut dapat disimpulkan bahwa terjadi peningkatan penghuni (Tahanan dan Warga Binaan Pemasyarakatan) secara sangat signifikan di Lapas dan Rutan, kurang lebih 124.000 penghuni[1].

Data jumlah penghuni tersebut menjadi faktor pencetus utama munculnya berbagai permasalahan dari segi sosial dan psikologi. Program pembinaan yang diselenggarakan di rutan seperti keterampilan kerja, pembinaan intelektual, perawatan atau rehabilitasi medis dan sosial tidak dapat berjalan secara efektif. Hal tersebut disebabkan oleh masalah *over* kapasitas penghuni, keadaan ini juga semakin diperparah dengan permasalahan lain yang muncul dari segi kesehatan tahanan, baik kesehatan fisik maupun kesehatan mental.

Kompleksitas tahanan dan segala permasalahan yang terjadi di dalam rutan ini perlu segera ditindaklanjuti, untuk itulah perlu dilakukan pelaksanaan *assessment* yang mencakup semua aspek diri setiap Warga Binaan Pemasyarakatan. Instrumen *assessment* yang digunakan diharapkan dapat dipakai tidak hanya oleh tenaga ahli tertentu (Psikolog) namun juga oleh petugas yang diberikan pelatihan khusus. *Assessment* yang dilakukan bertujuan mengklasifikasikan tahanan secara objektif berupa hasil

pengelompokkan klasifikasi risiko tahanan (rendah, menengah, dan tinggi), sehingga rutan dapat secara efektif memberikan pelayanan keamanan dan kesehatan yang tepat bagi tahanan selama perkaranya disidangkan.

Pada Lapas Kelas IIB Selong, proses *assessment* dilakukan secara manual oleh pegawai yang terlatih dengan melakukan wawancara terhadap tahanan dan mencatat hasil *assessment*. Setelah dilakukan wawancara, maka pegawai akan menilai setiap indikator yang ada pada lembar *assessment* sesuai dengan pedoman. Dengan menggunakan sistem manual untuk melakukan *assessment* risiko terhadap tahanan menjadikan proses nya lebih lama, beresiko terjadi kesalahan inputan nilai dan harus dilakukan penghitungan kembali untuk memastikan apakah hasil yang diperoleh sudah sesuai atau tidak.

Seiring dengan berkembangnya teknologi informasi, terdapat banyak metode yang dapat digunakan dalam pengembangan sistem, salah satunya yaitu metode *Rational Unified Process* (RUP). Kelebihan dari metode *Rational Unified Process* (RUP) yaitu mendukung proses perulangan dalam proses pengembangan perangkat lunak yang fokus pada arsitektur (*architecture-centric*) dan lebih diarahkan berdasarkan penggunaan kasus (*use case driven*). RUP merupakan proses rekayasa perangkat lunak dengan pendefinisian yang lebih baik (*well defined*) dan penstrukturan yang baik (*well structured*).

Berdasarkan uraian di atas, maka pada tugas akhir ini penulis akan merancang dan membangun sebuah Sistem Informasi *Assessment* Risiko Tahanan Berbasis *Website* di Lapas Kelas IIB Selong yang dapat membantu mengklasifikasikan tahanan berdasarkan tingkat risiko rendah, menengah, atau tinggi sehingga dapat menghasilkan perhitungan dan informasi yang sesuai dengan pedoman *assessment*. Untuk mendukung pengembangan *website*, penulis menggunakan metode *Rational Unified Process* (RUP) yang merupakan metode pengembangan perangkat lunak yang berorientasi objek. Metode *Rational Unified Process* (RUP) dipilih karena dapat menjelaskan pengembangan sistem secara *real*, memiliki dokumentasi sistem yang jelas dan memiliki fleksibilitas dalam proses pengembangan[2].

## **1.2 Rumusan Masalah**

Berdasarkan latar belakang yang telah dipaparkan, maka rumusan masalah dalam tugas akhir ini adalah bagaimana mengimplementasikan metode *Rational Unified Process* (RUP) dalam proses pengembangan Sistem Informasi *Assessment* Risiko Tahanan?

## **1.3 Batasan Masalah**

Penelitian ini memiliki batasan-batasan masalah untuk memberikan lingkup penelitian agar lebih terfokus ketika pengerjaan. Adapun batasan masalah yang diberikan sebagai berikut:

1. Sistem Informasi *Assessment* Risiko Tahanan berupa sistem berbasis *website* dengan Bahasa pemrograman PHP *framework CodeIgniter*.
2. Metode pengembangan yang digunakan untuk merancang dan membangun Sistem Informasi *Assessment* Risiko Tahanan yaitu metode *Rational Unified Process* (RUP).
3. Pengguna Sistem Informasi *Assessment* Risiko Tahanan yaitu Admin, Assessor, Kepala Seksi Bimbingan Napi/Anak Didik Dan Kegiatan Kerja, Kepala Subseksi Registrasi Dan Bimbingan Kemasyarakatan dan Kepala Lembaga Pemasyarakatan.
4. *Output* yang dihasilkan berupa hasil klasifikasi (ringan, menengah, tinggi) berdasarkan tingkat risiko tahanan.

## **1.4 Tujuan**

Tujuan dari penelitian ini adalah mengimplementasikan Metode *Rational Unified Process* (RUP) dalam pengembangan Sistem Informasi *Assessment* Risiko Tahanan.

## **1.5 Manfaat**

Manfaat dari penelitian tugas akhir ini adalah sebagai berikut:

1. Memudahkan pegawai dalam melakukan proses *assessment* tahanan sehingga hasil klasifikasi risiko tahanan diperoleh dengan cepat dan sesuai dengan pedoman *assessment*.
2. Kepala Lembaga Pemasyarakatan dapat melihat laporan bulanan yang berupa hasil klasifikasi berdasarkan tingkat risiko tahanan.

## 1.6 Sistematika Penulisan

Laporan tugas akhir ini disajikan dengan sistematika penulisan sebagai berikut:

**BAB 1 : PENDAHULUAN**

Bab ini menjelaskan tentang latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat serta sistematika penulisan yang digunakan untuk menyusun laporan tugas akhir.

**BAB 2 : TINJAUAN PUSTAKA DAN DASAR TEORI**

Bab ini membahas tentang teori-teori yang berkaitan langsung dengan judul yang diangkat yang mendukung pembahasan dan permasalahan. Teori- teori yang mendukung laporan penelitian ini bersumber dari buku, jurnal, *e-book* dan dari situs- situs *web* resmi yang berkaitan dari pembahasan dan permasalahan yang diangkat.

**BAB 3 : METODOLOGI PERANCANGAN**

Bab ini membahas tentang metodologi yang digunakan dalam merancang Sistem Informasi *Assessment* Risiko Tahanan.

**BAB 4 : PEMBAHASAN**

Bab ini membahas tentang analisis perangkat lunak, meliputi analisis masalah, analisis metode, analisis kebutuhan sistem, serta perancangan sistem yang terdiri dari perancangan diagram alir (*flowchart*).

**BAB 5 : PENUTUP**

Bab ini berisi kesimpulan dan saran-saran berdasarkan hasil pembahasan yang diperoleh.

## **BAB II**

### **TINJAUAN PUSTAKA DAN LANDASAN TEORI**

#### **2.1 Tinjauan Pustaka**

Pembuatan Sistem Informasi dengan menggunakan metode pengembangan *Rational Unified Process* (RUP) dan Sistem Informasi yang berkaitan dengan Lembaga Pemasarakatan pernah dilakukan sebelumnya selama beberapa tahun terakhir. Penelitian-penelitian sebelumnya yang akan dijadikan sebagai rujukan ketika pelaksanaan pembuatan sistem informasi ini yaitu sebagai berikut:

Penelitian yang berjudul “Analisis dan Pengembangan Sistem Ujian Akhir Semester berbasis Komputer di SMK Negeri 1 Magelang”. Produk yang dihasilkan adalah sistem informasi berbasis web yaitu sistem ujian akhir semester berbasis komputer. Sistem ini dikembangkan dengan menggunakan *framework* Laravel 5.5. Target pengguna dari produk ini adalah siswa dan guru. Untuk mendapatkan produk yang sesuai dengan kebutuhan pengguna, maka dalam pengembangan perangkat lunak ini digunakan metode *Rational Unified Process* (RUP) sebagai model pengembangan perangkat lunak. Model pengembangan RUP dipilih karena memiliki sifat *iterative* dan *incremental* sehingga mampu mengakomodasi perubahan kebutuhan perangkat lunak. Selain itu, RUP merupakan panduan yang dibuat untuk memaksimalkan penggunaan *Unified Modelling Language*. Penelitian ini menghasilkan sebuah sistem informasi untuk mempermudah dan meningkatkan kualitas pelaksanaan ujian akhir semester. Berdasarkan hasil pengujian, sistem memenuhi kelayakan untuk semua aspek pengujian standar ISO 25010. Diantaranya nilai *functional suitability* sebesar 1 (semua fitur telah terimplementasikan), nilai *usability* sebesar 85.33% (sangat layak) dan nilai Alpha Cronbach sebesar 0.87 (*excellent*), nilai *reliability*: (1) *successful sessions* sebesar 100%, (2) *successful pages* sebesar 100%, dan (3) *successful hits* sebesar 99.99%, nilai *performance efficiency*: (1) *page speed* sebesar 98.04%, (2) YSlow sebesar 90.97%, dan (3) waktu respon sebesar 1.13 detik, dan nilai *maintainability* sebesar 100 (sangat mudah dirawat)[3].

Penelitian yang berjudul “Rancang Bangun Sistem Informasi Administrasi PKM Universitas Lampung Berbasis Web Menggunakan Metode *Rational Unified Process* (RUP)”. Penelitian ini bertujuan untuk membuat sebuah sistem informasi administrasi PKM yang dapat digunakan untuk melayani proses pendaftaran, mengunggah dokumen laporan kemajuan, dan melihat pengumuman terkait PKM, menyediakan halaman edit progres peserta PKM untuk administrator menyeleksi peserta PKM yang lolos ke tahap selanjutnya, menyimpan arsip data mahasiswa peserta PKM dan dokumen PKM mahasiswa Universitas Lampung pada *database* yang terdapat pada sistem, menyediakan fitur cetak halaman pengesahan untuk mengurangi kesalahan administrasi dalam proses melengkapi berkas pendaftaran PKM dan fitur laporan data PKM. Pada penelitian ini menggunakan metode pengembangan *Rational Unified Process* (RUP) karena metode ini dapat merumuskan model arsitektur *enterprise* ke dalam aktifitas-aktifitas pengembangan sistem informasi untuk kebutuhan organisasi, dalam hal ini adalah arsitektur sistem informasi perguruan tinggi. Metode RUP lebih mudah dalam menghasilkan desain sistem informasi yang diinginkan organisasi. RUP dapat menjelaskan pengembangan sistem secara *real*, memiliki dokumentasi sistem yang jelas dan memiliki fleksibilitas dalam proses pengembangan. Dengan menggunakan metode *Rational Unified Process* (RUP) Sistem informasi administrasi PKM Universitas Lampung berhasil dibangun dengan pengujian *equivalence partitioning* dan versi pengujian alpha dengan tingkat keberhasilan 91,41%[4].

Penelitian yang berjudul “Perbandingan Antara Metode RUP dan *Prototype* Dalam Aplikasi Penerimaan Siswa Baru Berbasis Web” bertujuan untuk membuat sebuah sistem informasi dengan menggunakan dua metode pengembangan sistem yaitu RUP dan *Prototype*. Penelitian ini membandingkan antara kedua metode perangkat lunak tersebut untuk menguji keakuratan sistem yang akan dibangun. Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa metode RUP lebih fleksibel terhadap perubahan pada setiap proses, sedangkan untuk *prototype* kurang fleksibel terhadap perubahan[5].

Penelitian yang berjudul “*A comparison between two software engineering process, RUP and waterfall models*” bertujuan untuk membandingkan antara dua model pengembangan perangkat lunak yaitu RUP dan *waterfall*. Pada penelitian ini dijelaskan tentang kelebihan dan kekurangan dari kedua model pengembangan perangkat lunak tersebut. Kelebihan dari model RUP adalah fleksibel jika *user* ingin melakukan perubahan pada setiap tahapan pengembangan, sedangkan untuk *waterfall* tidak dapat terlalu banyak melakukan revisi atau perubahan. Kelemahan RUP adalah prosesnya yang bisa jadi terlalu banyak karena dimungkinkan adanya perubahan, sedangkan *waterfall* menggunakan tahapan pasti sehingga mudah digunakan[6].

Penelitian yang berjudul “*A comparison between three SDLC models waterfall model, spiral model, and incremental/iterative model*” bertujuan untuk membandingkan ketiga model SDLC yaitu *waterfall*, *spiral*, dan *incremental/iterative* atau RUP. Perbedaan ketiga model dari segi *flexibility*, *model waterfall* merupakan model yang paling tidak fleksibel, model *spiral* agak fleksibel, dan model RUP yang paling fleksibel. Dari segi jaminan keberhasilan, model *waterfall* kurang dijamin keberhasilannya, sedangkan model *spiral* dan RUP jaminan keberhasilannya tinggi. Dari segi pengujian, model *waterfall* lambat dalam melakukan pengujian, sedangkan untuk model *spiral* dan RUP cepat dalam melakukan pengujian bahkan dilakukan di akhir setiap iterasi untuk model RUP[7].

Penelitian yang berjudul “Analisa Sistem Informasi Pemberian Remisi Bagi Narapidana Pada Lembaga Pemasyarakatan Kelas IIB Sungailiat”. Penelitian ini bertujuan untuk membuat sebuah sistem informasi pemberian remisi bagi narapidana yang dapat digunakan untuk mengelola pendataan narapidana, pendataan data remisi, pendataan usulan remisi, pendataan usulan perubahan, pendataan surat keputusan, pendataan surat lepas, dan laporan remisi. Penelitian ini menghasilkan suatu sistem yang dapat mempermudah dalam proses pemberian remisi, menghemat waktu, meningkatkan efektifitas kerja, serta dapat memperkecil kemungkinan terjadinya kesalahan dalam menentukan remisi[8].

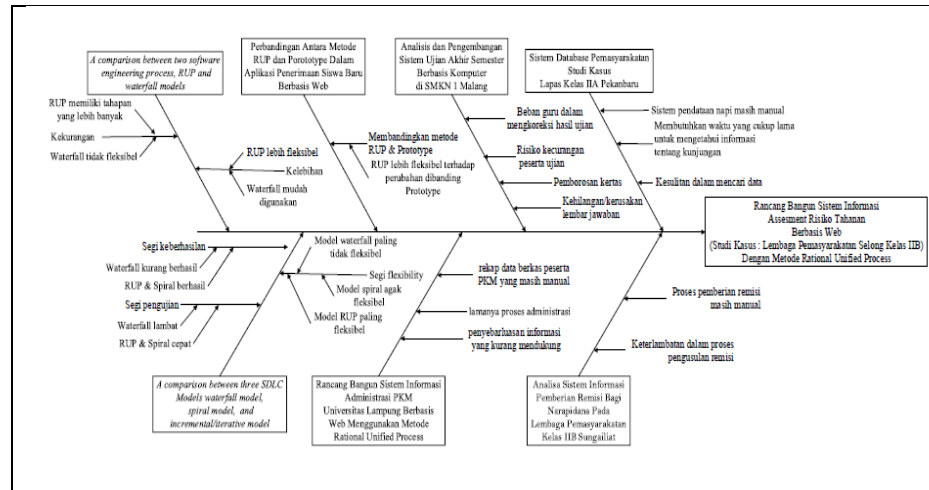
Penelitian yang berjudul “Sistem *Database* Pemasyarakatan Studi Kasus Lapas Kelas IIA Pekanbaru”. Sistem informasi ini dikembangkan dengan menggunakan Bahasa pemrograman PHP. Dalam penelitian ini sistem yang dikembangkan berfungsi untuk mengolah data narapidana, data pembinaan narapidana, dan data pengunjung di Lembaga Pemasyarakatan Kelas IIA Pekanbaru. Semua data tersebut diproses dan disimpan dalam *database* sehingga mempermudah petugas lembaga pemasyarakatan dalam merekapitulasi data narapidana, memberikan informasi tentang data pembinaan narapidana serta informasi data pengunjung dengan cepat[9].

Berdasarkan tinjauan pustaka di atas, penulis akan merancang dan membangun Sistem Informasi *Assessment* Risiko Tahanan dengan menggunakan metode *Rational Unified Process* (RUP). Metode RUP digunakan karena metode ini lebih dapat menerima perubahan pada setiap tahapan pengembangan dibandingkan dengan metode lain seperti *prototype*, ataupun metode SDLC lainnya. Pada penelitian ini, bentuk pengujian yang akan digunakan yaitu ISO 25010 dengan aspek *functional suitability* dan *usability*. Kedua aspek tersebut dipilih karena sudah memenuhi kebutuhan untuk pengembangan sistem. Studi kasus ini diambil karena belum ada penelitian tentang *assessment* risiko tahanan sehingga dibuatlah usulan tugas akhir dengan judul tersebut. Tujuan dari penelitian ini yaitu untuk mempermudah pegawai dalam mengklasifikasikan tingkat risiko tahanan sehingga dapat menghasilkan perhitungan dan informasi yang sesuai dengan pedoman *assessment*.

Adapun kelebihan yang ditawarkan sistem yaitu :

1. Proses pengolahan hasil *assessment* dilakukan melalui sistem sehingga mempercepat perolehan hasil klasifikasi tingkat risiko tahanan dan sesuai dengan pedoman *assessment*.
2. Memudahkan pegawai dalam mengolah informasi berdasarkan hasil klasifikasi tingkat risiko tahanan.

Gambar 2.1 merupakan *fishbone* diagram dari tinjauan pustaka dan usulan sistem yang akan dikembangkan:



Gambar 2.1 *Fishbone* diagram

## 2.2 Dasar Teori

Teori-teori dasar atau umum yang digunakan dalam penelitian ini yaitu sebagai berikut:

### 2.2.1 Sejarah Lembaga Pemasyarakatan Kelas IIB Selong

Lembaga Pemasyarakatan Kelas IIB Selong merupakan salah satu Unit Pelaksana Teknis Pemasyarakatan Kantor Wilayah Kementerian Hukum dan HAM Provinsi Nusa Tenggara Barat. Rumah Tahanan Negara Kelas IIB Selong terletak di jalan TGKH.M.Zainuddin Abdul Majid No. 199 Selong Lombok Timur Nusa Tenggara Barat.

Lembaga Pemasyarakatan Kelas IIB Selong dibangun diatas tanah seluas 1 Ha dahulunya adalah Lembaga Pemasyarakatan Selong yang merupakan bangunan peninggalan penjajahan Belanda. Berdasarkan Keputusan Menteri Kehakiman RI No: M.03.UM.01.06 Tahun 1983 tentang penetapan Lembaga Pemasyarakatan tertentu sebagai Rumah Tahanan Negara maka Lembaga Pemasyarakatan Selong berubah status menjadi Rumah Tahanan Negara Kelas IIB Selong dan disertifikat pada tanggal 11 April 1989 dengan Nomor : 12/1989. Status Rumah Tahanan Negara Kelas

IIB Selong tersebut kembali diubah menjadi Lembaga Pemasyarakatan Kelas IIB Selong berdasarkan Keputusan Menteri Hukum Dan HAM RI No: M.HH-05.0T.01.03 Tahun 2019 tentang perubahan status Rumah Tahanan Negara Kelas IIB Selong menjadi Lembaga Pemasyarakatan Kelas IIB Selong.

### 2.2.2 Visi dan Misi

Adapun visi dan misi yang dimiliki oleh Lembaga Pemasyarakatan Kelas IIB Selong yaitu sebagai berikut:

Visi: Mewujudkan pelayanan prima dan mendukung masyarakat memperoleh kepastian hukum.

Misi: Rutan terpercaya dalam memberikan pelayanan, pembinaan, perawatan, serta pengamanan bagi Warga Binaan Pemasyarakatan

### 2.2.3 Struktur Organisasi

Organisasi merupakan tempat berkumpulnya orang-orang untuk melakukan kerjasama guna mencapai suatu tujuan tertentu yang terdiri dari komponen-komponen tertentu. Dengan adanya struktur organisasi di suatu instansi, kita dapat melihat hubungan, peran dan kerjasama antar bagian. Adapun struktur organisasi Lapas Kelas IIB Selong dapat dilihat pada Gambar 2.2.



Gambar 2.2 Struktur organisasi Lapas Selong

### 2.2.4 Alur Assessment

Proses *assessment* risiko tahanan dilakukan oleh pegawai yang telah mengikuti bimbingan teknis/pelatihan untuk mengklasifikasikan tingkat

risiko tahanan (rendah, menengah, tinggi). Kemudian hasil *assessment* risiko tahanan tersebut akan diverifikasi oleh Kepala Subseksi Registrasi dan Bimbingan Kemasyarakatan. Setelah hasil *assessment* risiko tahanan diverifikasi, selanjutnya akan dilaporkan kepada Kepala Seksi Bimbingan Napi/Anak Didik Dan Kegiatan Kerja serta Kepala Lembaga Pemasyarakatan Kelas IIB Selong.

#### **2.2.5 Sistem Informasi**

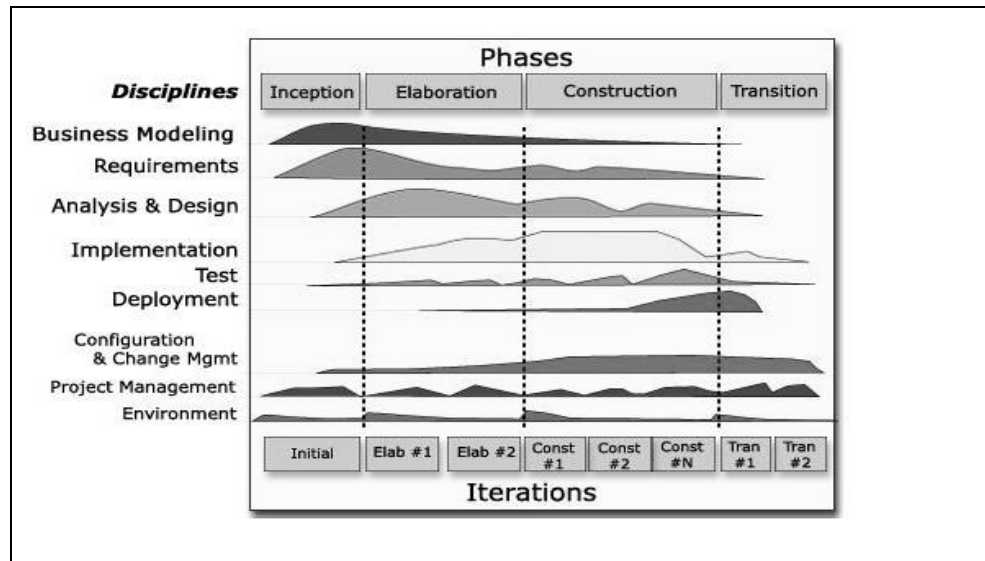
Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. Sistem Informasi adalah sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi serta menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.

Sistem informasi merupakan komponen yang saling bekerja sama untuk mengumpulkan, mengolah, menyimpan dan menyebarkan informasi untuk mendukung pengambilan keputusan, koordinasi, pengendalian, analisis masalah dan visualisasi dalam sebuah organisasi[10].

#### **2.2.6 RUP**

*Rational Unified Process* (RUP) merupakan suatu metode rekayasa perangkat lunak yang dikembangkan dengan mengumpulkan berbagai *best practises* yang terdapat dalam industri pengembangan perangkat lunak. Ciri utama metode ini adalah menggunakan *use-case driven* dan pendekatan iteratif untuk siklus pengembangan perangkat lunak. RUP menggunakan konsep *object oriented*, dengan aktifitas yang berfokus pada pengembangan model dengan menggunakan *Unified Model Language* (UML)[11].

Proses pengembangan perangkat lunak RUP digambarkan dalam 2 dimensi, yaitu dimensi horizontal dan dimensi vertikal. Dimensi horizontal yang merepresentasikan waktu dan tahapan pengembangan, sedangkan dimensi vertikal merepresentasikan *workflows* (alur kerja) pada setiap tahapan yang perlu dilakukan. Model proses pengembangan perangkat lunak RUP dapat dilihat pada Gambar 2.3.



Gambar 2.3 Arsitektur *rational unified process*.

Metode pengembangan *Rational Unified Process* (RUP) memiliki empat tahap, yaitu *Inception*, *Elaboration*, *Construction*, dan *Transition*. Adapun penjelasan empat tahap pengembangan RUP yaitu sebagai berikut:

#### 1. *Inception*

Tahap ini memodelkan proses bisnis yang dibutuhkan dan mendefinisikan kebutuhan sistem yang akan dibuat. Fokus kegiatan yaitu memahami ruang lingkup proyek seperti biaya, waktu, kebutuhan dan resiko. Tahap ini berfokus pada dua proses yaitu *business modelling* dan *requirements*.

#### 2. *Elaboration*

Tahap ini lebih difokuskan pada perencanaan arsitektur sistem. Tahap ini juga dapat mendeteksi apakah arsitektur sistem yang diinginkan dapat dibuat atau tidak. Melakukan analisis dan desain sistem serta implementasi sistem yang berfokus pada sistem *prototype*. Tahap *elaboration* memiliki fokus pada proses *analysis* dan *design*.

#### 3. *Construction*

Tahap *construction* berfokus pada pengembangan komponen dan fitur-fitur sistem. Melakukan implementasi dan pengujian sistem yang berfokus pada implementasi perangkat lunak pada kode program (*coding*).

#### 4. *Transition*

Pada tahap *transition* lebih mengarah pada *deployment* atau instalasi sistem agar dapat dimengerti oleh pengguna. Aktivitas pada tahap ini

termasuk pelatihan pengguna, pemeliharaan dan pengujian sistem apakah sudah memenuhi harapan pengguna.

RUP memiliki sembilan *core workflows* yang merepresentasikan kegiatan pengembangan. Terbagi menjadi dua kategori yaitu *Core Process Workflows* dan *Core Supporting Workflows*. *Core Process Workflows* terdiri dari *business modelling, requirement, analysis & design, implementation, test, dan deployment*. *Core Supporting Workflows* terdiri dari *project management, configuration & change management, dan environment*.

*Core Process Workflow* merupakan kegiatan yang bersifat pokok. Penjelasan lebih lengkap yaitu sebagai berikut:

1. *Business Modelling*

Dalam *business modelling* terdapat mendokumentasikan proses bisnis menggunakan kasus penggunaan bisnis (*Business Use Case*). Model tersebut menjamin pemahaman bersama di antara pelaku yang terlibat dalam pengembangan sistem. *Business Modelling* digunakan untuk menemukan dan menganalisis persyaratan sistem, memahami tujuan dari organisasi target.

2. *Requirements*

*Requirements* untuk menggambarkan apa yang harus dilakukan oleh sistem. Pengembang dan pengguna harus memiliki pemahaman yang sama terhadap kebutuhan sistem, sehingga sistem dapat dibangun sesuai keinginan dan kebutuhan pengguna. *Requirements* diperoleh dengan menganalisa kebutuhan yang didapat dari *Business Modelling*. *Requirements* yang baik mampu menggambarkan kebutuhan pengguna secara tepat.

3. *Analysis & Design*

*Analysis dan design* merupakan kegiatan menunjukkan bagaimana sistem akan direalisasikan dalam tahap implementasi. Menghasilkan model desain yang berfungsi sebagai abstraksi kode program. *Analysis dan design* yang baik dapat menggambarkan arsitektur sistem yang terstruktur sesuai kebutuhan sistem. Salah satu standar model desain yang digunakan adalah UML.

#### 4. *Implementation*

*Implementation* merupakan tahap mengintegrasikan hasil pengkodean menjadi sebuah sistem yang dapat dieksekusi atau digunakan. *Implementation* yang baik akan menghasilkan sistem sesuai kebutuhan pengguna, sistem yang mudah dikelola, dan dapat digunakan kembali atau dikembangkan.

#### 5. *Test*

*Test* merupakan kegiatan pengujian kualitas perangkat lunak untuk mendeteksi kesalahan pada setiap proses pengembangan. Karakteristik RUP menerapkan konsep perulangan, yang berarti melakukan proses pengujian terhadap keseluruhan proses pengembangan secara berulang. Standar pengujian kualitas perangkat lunak memuat fungsi pemeriksaan dan pembuatan sistem pelaporan yang bermanfaat untuk menilai efektivitas dan kelengkapan sebuah perangkat lunak. Terdapat beberapa standar pengujian kualitas perangkat lunak seperti McCall dan ISO 9126[12].

#### 6. *Deployment*

*Deployment* merupakan kegiatan menyebarkan rilis produk dan mengirimkan produk ke pengguna. Kegiatan *deployment* yang baik tidak hanya melakukan rilis produk, tetapi memberikan dukungan (misal dokumentasi penggunaan) kepada pengguna. Memberikan kemudahan pengguna untuk mengakses dan menggunakan perangkat lunak.

*Core Supporting Workflow* merupakan kegiatan yang bersifat pelengkap. Dapat tidak dikerjakan jika memang tidak diperlukan. Penjelasan lebih detail yaitu sebagai berikut:

##### 1. *Project Management*

*Project management* adalah metode untuk mengelola risiko dan mengatasi kendala melalui perencanaan, penjadwalan, penempatan staff, pelaksanaan, dan pemantauan proyek.

##### 2. *Configuration & Change Management*

*Configuration & change management* adalah kegiatan mengontrol produk yang melibatkan banyak orang pada suatu proyek. Hal ini membantu

menghindari kebingungan dan memastikan bahwa produk yang dihasilkan tidak saling bertentangan.

### 3. *Environment*

*Environment* adalah kegiatan untuk menyediakan lingkungan pengembangan perangkat lunak, berupa proses dan alat-alat yang dibutuhkan. Hal ini memberikan panduan, *template*, dan alat yang diperlukan untuk proses pengembangan perangkat lunak

Pada penelitian ini, hanya digunakan *Core Process Workflows*. Hal ini didasarkan oleh pengumpulan kebutuhan pada tahap *bussiness modelling* yaitu tidak adanya *maintenance* sistem ke depannya dikarenakan sistem ini hanya berjalan secara internal saja.

#### 2.2.7 UML

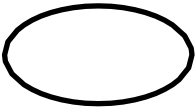
*Unified Modeling Language* (UML) adalah sebuah bahasa yang berdasarkan grafik atau gambar untuk memvisualisasi, menspesifikasikan, membangun, dan pendokumentasian dari sebuah sistem pengembangan *software* berbasis *object oriented*.

Dengan *Unified Modeling Language* (UML) sebuah sistem dapat dilihat dari struktur data yang diberikan, tingkah laku apa saja dari suatu objek hingga aktivitas – aktivitas yang ada di dalamnya. *Unified Modeling Language* (UML) sendiri sangat membantu sekali bagi para pengembang dalam membuat rancangan sebelum program tersebut dibuat dan membantu dalam pembuatan dokumentasi[13]. Pada *Unified Modeling Language* (UML) itu sendiri terdapat beberapa diagram yaitu sebagai berikut :

##### 2.2.7.1 *Use Case Diagram*

*Use Case* diagram merupakan pemodelan untuk mendeskripsikan perilaku (*behavior*) dari sistem informasi yang akan dibuat. Dapat juga dipahami sebagai cara untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi yang ada. Simbol-simbol yang digunakan pada *use case diagram* dapat dilihat pada Tabel 2.1 berikut.

Tabel 2.1 Tabel simbol *use case diagram*.

| Simbol                                                                            | Nama               | Deskripsi                                                                                                                                                                                                                       |
|-----------------------------------------------------------------------------------|--------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|  | <i>Actor</i>       | Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.                                 |
|  | <i>Use Case</i>    | 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 <i>use case</i> .                                |
|  | <i>Association</i> | Komunikasi antara <i>use case</i> dan aktor yang berpartisipasi pada <i>use case</i> atau <i>use case</i> yang berinteraksi dengan aktor.                                                                                       |
| ---<<extend>>---                                                                  | <i>Extend</i>      | Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu. Mirip dengan inheritance pada pemrograman berorientasi objek. |
| ---<<include>>---                                                                 | <i>Include</i>     | Relasi <i>use case</i> tambahan dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya.                                                                                             |

### 2.2.7.2 Entity Relationship Diagram (ERD)

*Entity Relationship Diagram* (ERD) merupakan suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi. ERD untuk memodelkan struktur data dan hubungan antar data, untuk menggambarkannya digunakan beberapa notasi dan simbol[9].

Adapun elemen-elemen diagram ER adalah sebagai berikut:

### 1. Entitas (*Entity*)

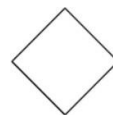
Entitas adalah sesuatu atau objek dalam dunia nyata yang dapat dibedakan dengan objek lainnya, sebagai contoh mahasiswa, dosen, dan departemen. Entitas terdiri atas beberapa atribut contohnya atribut dari entitas mahasiswa adalah nim, nama, dan alamat, selain itu pada setiap entitas harus memiliki satu buah atribut yang disebut dengan *primary key*. Simbol dari entitas adalah seperti Gambar 2.4.



Gambar 2.4 Simbol entitas

### 2. Relasi (*Relationship*)

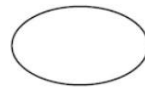
Relasi adalah hubungan antara satu atau lebih entitas, contohnya yaitu relasi antara entitas mahasiswa dan mata kuliah dimana setiap mahasiswa bisa mengambil beberapa mata kuliah. Kardinalitas menentukan kejadian suatu entitas untuk satu kejadian pada entitas yang berhubungan, kardinalitas terdiri dari *one to one*, *many to many*, dan *one to many*. Contohnya Mahasiswa bisa mengambil banyak mata kuliah. Adapun simbol dari relasi dalam diagram-ER adalah seperti pada Gambar 2.5 berikut:



Gambar 2.5 Simbol relasi

### 3. Atribut

Atribut adalah ciri umum yang mengidentifikasi entitas secara unik dan dapat mendeskripsikan karakteristik dari suatu entitas. Isi dari suatu atribut mempunyai sesuatu yang dapat mengidentifikasikan isi dari elemen satu sama lain. Terdapat dua jenis atribut yaitu *identifier* yang digunakan untuk menentukan suatu entitas secara unik, dan yang kedua yaitu *descriptor (nonkey)* digunakan untuk menspesifikasikan karakteristik dari entitas yang tidak unik. Simbol dari atribut adalah seperti Gambar 2.6.

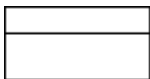


Gambar 2.6 Simbol atribut

### 2.2.7.3 Class Diagram

*Class diagram* adalah diagram yang menggambarkan struktur dan deskripsi kelas serta hubungan antar kelas. *Class diagram* terdiri dari 3 bagian utama yaitu nama, atribut, dan operasi/*method*. Kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem. Pada *class diagram* terdapat beberapa simbol dan beberapa cara penulisan diantaranya adalah seperti pada Tabel 2.2.

Tabel 2.2 Tabel relasi *class diagram*

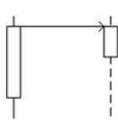
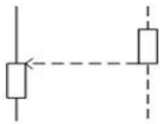
| Simbol                                                                              | Nama               | Keterangan                                                                                                                                                                                                              |
|-------------------------------------------------------------------------------------|--------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|   | <i>Class</i>       | <i>Class</i> adalah sebuah objek yang menggambarkan sebuah keadaan nyata. <i>Class</i> memiliki: Nama kelas, atribut, dan <i>method</i> .                                                                               |
|  | <i>Association</i> | Garis yang menghubungkan antara dua kelas atau lebih dan menunjukkan bahwa kelas yang terhubung menunjukkan sebuah relasi seperti : <i>one-to-one</i> (1 – 1), <i>one-to-many</i> (1 – M), <i>many-to-many</i> (M – M). |
|  | <i>Aggregation</i> | Menujukkan sebuah bagian relasi agregasi.                                                                                                                                                                               |

### 2.2.7.4 Sequence Diagram

*Sequence diagram* merupakan interaksi antar obyek di dalam dan di sekitar sistem (termasuk pengguna, *display* dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence diagram* terdiri atas dimensi vertikal (waktu) dan dimensi horizontal (obyek-obyek yang terkait). *Sequence diagram* biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respon dari sebuah *event* untuk menghasilkan *output* tertentu. Diawali dari apa yang men-*trigger*

aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan *output* apa yang dihasilkan. Simbol-simbol dalam *sequence diagram* dapat dilihat pada Tabel 2.3 berikut.

Tabel 2.3 Tabel simbol *sequence diagram*.

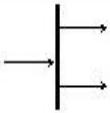
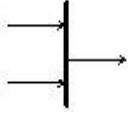
| Simbol                                                                             | Nama                  | Keterangan                                                                                             |
|------------------------------------------------------------------------------------|-----------------------|--------------------------------------------------------------------------------------------------------|
|   | <i>Actor</i>          | Menggambarkan seseorang atau sesuatu (seperti perangkat, sistem lain) yang berinteraksi dengan sistem. |
|   | <i>Object Message</i> | Menggambarkan pesan/hubungan antar obyek yang menunjukkan urutan kejadian yang terjadi.                |
|  | <i>Return Message</i> | Menggambarkan pesan / hubungan antar obyek, yang menunjukkan urutan kejadian yang terjadi.             |

#### 2.2.7.5 Activity Diagram

*Activity diagram* adalah cara untuk menggambarkan aktivitas – aktivitas yang terdapat pada suatu sistem yang akan dibuat. Dengan *activity diagram*, alur dari kerja suatu sistem dapat dibuat dari awal hingga akhir. *Activity diagram* memperlihatkan siapa saja yang terlibat baik aktor maupun sistem serta memberikan keterangan proses dari tiap bagian tersebut. Pada *activity diagram* terdapat beberapa simbol yang dapat dilihat pada Tabel 2.4.

Tabel 2. 4 Tabel simbol *activity diagram*

| Simbol                                                                              | Nama                       | Keterangan                                    |
|-------------------------------------------------------------------------------------|----------------------------|-----------------------------------------------|
|  | <i>Activity</i>            | Menunjukkan sebuah aktivitas yang dilakukan   |
|  | <i>Initial node</i>        | Awal dari aktivitas suatu sistem atau program |
|  | <i>Activity final node</i> | Akhir dari suatu aktivitas                    |

|                                                                                   |                       |                                                               |
|-----------------------------------------------------------------------------------|-----------------------|---------------------------------------------------------------|
|  | <i>Decision</i>       | Menunjukkan sebuah pilihan                                    |
|  | <i>Line connector</i> | Menghubungkan antara simbol yang satu dan yang lainnya        |
|  | <i>Fork</i>           | Menunjukkan adanya percabangan secara parallel dari aktivitas |
|  | <i>Join</i>           | Menunjukkan adanya penggabungan aktivitas                     |

### 2.2.8 *CodeIgniter*

*CodeIgniter* merupakan aplikasi *open source* yang berupa *framework* PHP dengan model MVC (*Model, View, Controller*) untuk membangun *website* dinamis dengan menggunakan PHP. *CodeIgniter* memudahkan *developer* untuk membuat aplikasi *web* dengan cepat mudah dibandingkan dengan membuatnya dari awal.

### 2.2.9 XAMPP

XAMPP adalah perangkat lunak bebas, yang mendukung banyak *system* operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai *server* yang berdiri sendiri (*localhost*), yang terdiri atas program Apache HTTP Server, MySQL *database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMP merupakan singkatan dari X (empat *system* operasi apapun), Apache, MySQL, PHP dan Perl. Program ini tersedia dalam GNU *General Public License* dan bebas, merupakan *web server* yang mudah digunakan yang dapat melayani tampilan halaman *web* yang dinamis[14].

### 2.2.10 *Business Modelling Canvas*

Bisnis model *canvas* adalah model bisnis yang berisi 9 blok area aktivitas bisnis, dimana memiliki tujuan menentukan strategi untuk membangun sebuah bisnis yang kuat, bisnis model *canvas* ini memiliki ciri

khas dengan 9 blok model yang jika di disatukan akan menjadi satu kesatuan bisnis. Adapun sembilan blok tersebut yaitu:

a. *Customer Segments*

Blok ini menjelaskan siapa yang akan dilayani atau menggunakan produk ini. dikarenakan pelanggan adalah kunci dari sebuah bisnis. Setelah tahu siapa target pasarnya maka bisa langsung menuju tahap selanjutnya dalam *business modelling* ini.

b. *Value Propositions*

Blok ini menjelaskan bagaimana cara menggabungkan antara produk dan layanan juga mengatasi masalah target bisnis dan bagaimana solusi yang akan dibuat.

c. *Channels*

Blok ini menjelaskan bagaimana produk ini bisa dipromosikan ke segmen pelanggan agar bisa segmen pelanggan tahu akan produk yang akan dibuat.

d. *Customer Relationships*

Blok ini menjelaskan bagaimana cara membangun segmen pelanggan agar bisa nyaman menggunakan aplikasi tersebut. Agar hubungan dengan segmen pelanggan berjalan lebih baik.

e. *Revenue Streams*

Blok ini menjelaskan penghasilan yang akan didapat dari produk tersebut dari pendapatan. Pendapatan ada 2 jenis yaitu pendapatan satu kali pembayaran dan ada juga yang berkelanjutan atau bersifat berulang.

f. *Key Activities*

Blok ini menjelaskan hal terpenting yang akan dilakukan oleh pemilik produk agar bisnisnya berjalan lancar dan sesuai dengan tujuan utamanya. *key activities* dapat dikatakan sebagai pemecah masalah dan apa yang akan dibuat.

g. *Key Resources*

Blok ini menjelaskan sumber daya apa aja yang sangat dibutuhkan untuk produk tersebut agar sebuah model bisnis dapat berfungsi dengan sangat baik. Sumber yang dimaksud adalah sumber finansial, intelektual atau manusia.

h. *Key Partnership*

Blok ini menjelaskan siapa mitra yang akan membantu membuat model tersebut. Agar bisa mengoptimalkan model bisnis dan mengurangi resiko yang ada. Kemitraan sangatlah penting sebagai landasan penting dari berbagai model bisnis.

i. *Cost Structure*

Blok ini menjelaskan semua biaya yang dikeluarkan untuk membuat sebuah produk tersebut. Untuk menjelaskan mengenai biaya-biaya terpenting agar lebih mudah harus tahu sumber daya utama, aktifitas-aktifitas kunci.

Dari pembahasan di atas dapat disimpulkan bisnis model *canvas* sangat membantu seseorang untuk memulai bisnis dimana membuat suatu produk harus jelas dan kuat dimana pemilik bisnis harus memiliki 9 blok tersebut agar bisnisnya berjalan dengan baik dan mengurangi tingkat kegagalan dalam berbisnis[15].

### 2.2.11 ISO 2510

ISO 25010 adalah standar kualitas perangkat lunak yang baru, menggantikan ISO 9126. Standar kualitas perangkat lunak ini dibuat oleh *International Organization for Standardization (ISO)* dan *International Electrotechnical Commission*. Oleh karena itu, dalam penelitian ini menggunakan standar ISO 25010 sebagai standar dalam pengujian perangkat lunak. ISO 25010 memiliki 8 karakteristik yaitu *functional suitability*, *reliability*, *performance efficiency*, *usability*, *security*, *compatibility*, *maintainability*, dan *portability*. Namun pada penelitian ini akan diambil 2 karakteristik ISO 25010 yaitu *functional suitability*, dan *usability*. Adapun penjelasan 2 karakteristik ISO 25010 yaitu sebagai berikut[16]:

1. *Functional Suitability*

Karakteristik sejauh mana sistem telah memenuhi kebutuhan pengguna. Terbagi menjadi 3 sub karakteristik yaitu:

- a) *Functional Completeness*, sejauh mana fungsi yang telah diimplementasikan memenuhi tugas dan tujuan pengguna secara spesifik.
- b) *Functional Correctness*, sejauh mana fungsi yang diimplementasikan menghasilkan *output* yang benar.

c) *Functional Appropriateness*, sejauh mana fungsi yang diimplementasikan mampu memfasilitasi penyelesaian tugas dan tujuan tertentu.

Pengujian *functional suitability* menggunakan metode *black-box testing*. *Black-box testing* disebut juga pengujian perilaku, berfokus pada persyaratan fungsionalitas perangkat lunak. Pengujian karakteristik *functional suitability* Sistem Informasi Assessment Risiko Tahanan ini menggunakan *test case* yang dinilai dengan skala Guttman menggunakan pilihan jawaban ya-tidak. Pilihan jawaban ya bernilai 1 dan pilihan jawaban tidak bernilai 0. Hasil pengujian *functional suitability* dihitung menggunakan rumus:

$$X = \frac{I}{P}$$

Keterangan:

P = Jumlah fitur yang dirancang

I = Jumlah fitur yang berhasil diimplementasikan

Interpretasi pengukuran yang digunakan yaitu nilai yang mendekati 1 mengindikasikan jumlah fitur yang berhasil diimplementasikan. Dalam pengujian produk ini dikatakan baik dalam karakteristik *functional suitability* jika nilai X mendekati 1.

## 2. Usability

*Usability* adalah karakteristik sejauh mana sebuah produk atau sistem dapat digunakan oleh pengguna tertentu untuk mencapai tujuan dengan efektif, efisien, dan kepuasan tertentu dalam konteks pengguna. Pada pengujian *usability*, digunakan kuesioner *USE Questionnaire* yang terdiri dari sejumlah pertanyaan yang terbagi menjadi 4 kriteria yaitu *Usefulness*, *ease of use*, *ease of learning* dan *satisfaction*. Skala yang digunakan pada kuesioner adalah skala *Likert*. Untuk keperluan analisis kuantitatif maka jawaban pada skala *likert* dapat diberi skor sebagai berikut:

- Sangat Setuju (ST) diberi skor 5
- Setuju (S) diberi skor 4
- Netral (N) diberi skor 3
- Tidak Setuju (TS) diberi skor 2

- Sangat Tidak Setuju (STS) diberi skor 1

Data hasil pengujian *usability* dianalisis dengan menghitung jawaban berdasarkan skor setiap jawaban dari responden. Rumus perhitungan skor pengujian *usability* yaitu:

$$Skor\ total = (JSS \times 5) + (JS \times 4) + (JN \times 3) + (JTS \times 2) + (JSTS \times 1)$$

Keterangan:

JSS = jumlah responden menjawab Sangat Setuju

JS = jumlah responden menjawab Setuju

JN = jumlah responden menjawab Netral

JTS = jumlah responden menjawab Tidak Setuju

JSTS = jumlah responden menjawab Sangat Tidak Setuju

Kemudian mencari persentase skor untuk mendapatkan kriteria interpretasi skor hasil pengujian *usability* dengan rumus:

$$P\ skor = Skor\ total \div r \times 5 \times 100\%$$

Keterangan:

Skor total = skor total hasil jawaban responden

i = jumlah pertanyaan

r = jumlah responden

Setelah mendapatkan hasil berupa nilai kuantitatif dari perhitungan sebelumnya, nilai dikonversi menjadi nilai kualitatif berskala 5 dengan skala *Likert*. Konversi persentase ke pernyataan seperti dalam Tabel 2.5.

Tabel 2.5 Kriteria interpretasi skor.

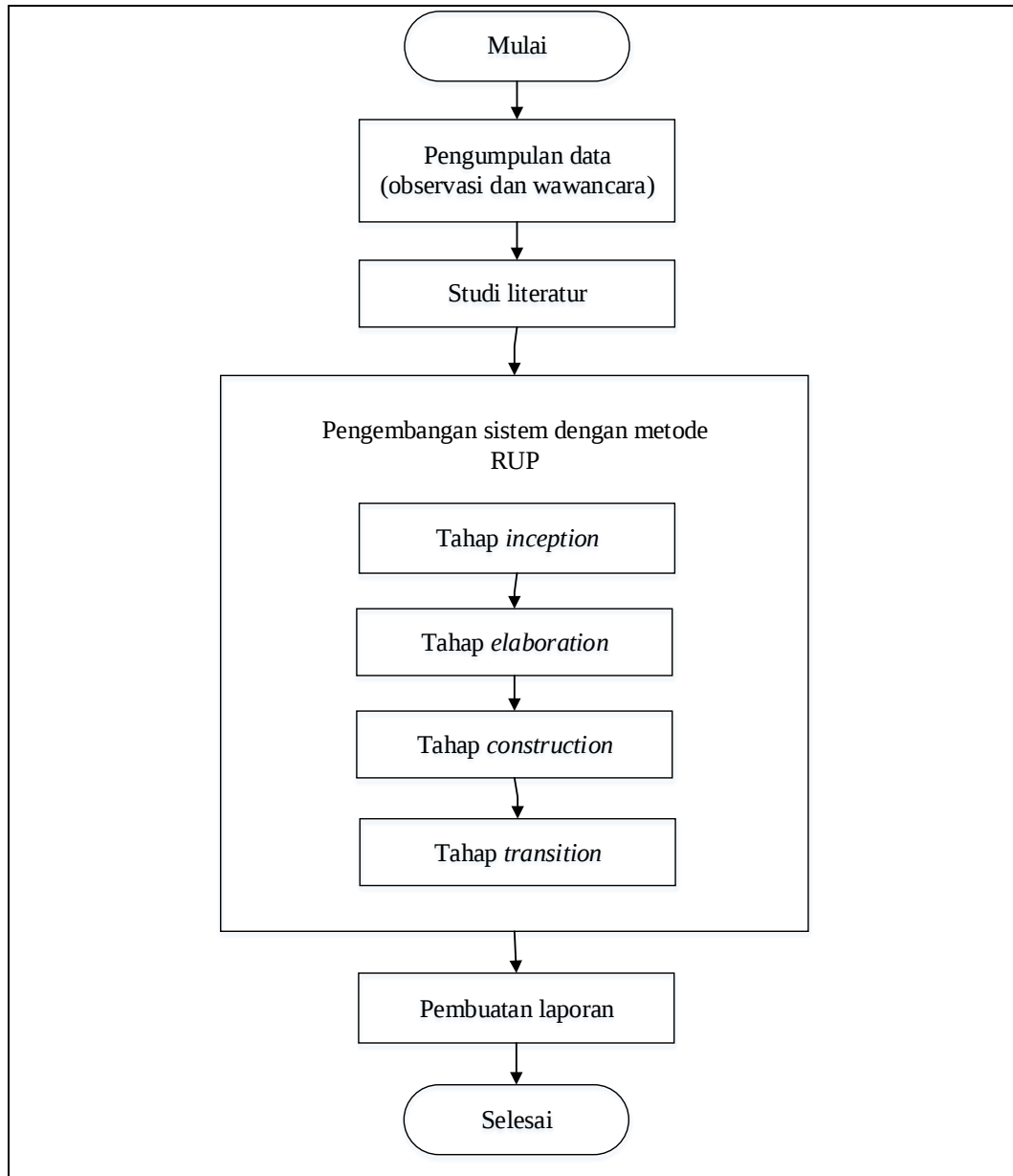
| No | Persentase Pencapaian | Interpretasi       |
|----|-----------------------|--------------------|
| 1  | 0% - 20%              | Sangat Tidak Layak |
| 2  | 21% - 40%             | Kurang Layak       |
| 3  | 41% - 60%             | Cukup Layak        |
| 4  | 61% - 80%             | Layak              |
| 5  | 81% - 100%            | Sangat Layak       |

## BAB III

### METODE PENELITIAN

#### 3.1 Metode Pengembangan Sistem

Dalam pengembangan sistem menggunakan beberapa tahapan yang mengikuti metode RUP seperti yang terlihat pada Gambar 3.1 berikut.



Gambar 3.1 Tahapan pengembangan sistem[16].

Penjelasan mengenai tahapan pengembangan sistem berdasarkan pada Gambar 3.1 yaitu sebagai berikut.

### **3.1.1 Pengumpulan data**

Pengumpulan data pada tugas akhir ini dilakukan dengan dua cara yaitu observasi dan wawancara.

#### **3.1.1.1 Observasi**

Pada observasi yang dilakukan, penulis mengamati secara langsung bagaimana proses pengelolaan data tahanan, proses melakukan *assessment* risiko tahanan dan pengolahan hasil *assessment* risiko tahanan yang dilakukan oleh pegawai di Lembaga Pemasyarakatan Kelas IIB Selong. Hasil yang didapat dari observasi yang dilakukan yaitu sebuah form *assessment* risiko tahanan yang akan digunakan pada proses *assessment* risiko tahanan.

#### **3.1.1.2 Wawancara**

Pada wawancara yang dilakukan di Lembaga Pemasyarakatan Kelas IIB Selong dengan Bapak Nasruddin selaku Kepala Seksi Bimbingan Napi/Anak Didik Dan Kegiatan Kerja, dijelaskan tentang proses *assessment* risiko tahanan masih dilakukan secara manual. Proses yang cukup lama dengan jumlah pegawai terlatih yang sedikit menjadi penghambat dalam melakukan proses *assessment* risiko tahanan yang harus dilakukan sesegera mungkin. Perhitungan *assessment* risiko sangat penting dan sensitif yang membutuhkan ketelitian dan membutuhkan hasil perhitungan yang sesuai dengan pedoman *assessment*. Dengan menggunakan sistem manual untuk melakukan *assessment* risiko terhadap tahanan menjadikan proses nya lebih lama, beresiko terjadi kesalahan inputan nilai dan harus dilakukan penghitungan kembali untuk memastikan apakah hasil yang diperoleh sudah sesuai atau tidak. Penulis menawarkan untuk melakukan penelitian pengembangan sebuah sistem informasi yang dapat membantu mengklasifikasikan tahanan berdasarkan tingkat risiko rendah, menengah, atau tinggi sehingga dapat menghasilkan perhitungan dan informasi yang sesuai dengan pedoman *assessment*.

Berdasarkan wawancara yang dilakukan, diperoleh informasi berupa proses penentuan klasifikasi *assessment* memiliki indikator berupa *assessment* umum dan *assessment* khusus. Beberapa aspek yang termasuk ke dalam *assessment* umum yaitu identitas tahanan, latar belakang keluarga, interaksi sosial, pelanggaran hukum yang pernah dilakukan, dan total persepsi tahanan terhadap pasal yang

dituduhkan. Adapun beberapa aspek yang termasuk ke dalam *assessment* khusus yaitu penyalahgunaan narkoba, minuman keras, kesehatan mental dan fisik, serta kebutuhan penanganan khusus. Hasil klasifikasi *assessment* umum dan *assessment* khusus berupa kesimpulan klasifikasi dengan rincian sebagai berikut:

1. Ringan, dengan rentang nilai 0-23.
2. Menengah, dengan rentang nilai 24-48.
3. Tinggi, dengan rentang nilai >48.

Rumus yang digunakan untuk penentuan hasil klasifikasi *assessment* risiko tahanan yaitu:

$$Skor\ total = AU + AK$$

Keterangan:

AU = total skor *Assessment* Umum

AK = total skor *Assessment* Khusus

Contohnya, total skor untuk *assessment* umum yaitu 12 dan total skor untuk *assessment* khusus yaitu 12. Maka, total skor untuk klasifikasi *assessment* risiko tahanan yaitu 24. Berdasarkan total skor tersebut, dapat disimpulkan bahwa tahanan termasuk ke dalam klasifikasi risiko menengah.

### **3.1.2 Studi literatur**

Pada penelitian ini, penulis melakukan studi literatur dengan cara mengumpulkan data dari berbagai skripsi, jurnal maupun sumber lain yang berkaitan dengan penelitian ini. Dilakukannya studi literatur bertujuan untuk melihat kekurangan serta kelebihan yang ada pada skripsi, jurnal maupun sumber lainnya yang nantinya digunakan sebagai acuan untuk mengembangkan sistem yang akan dibangun.

### **3.1.3 Pengembangan sistem dengan metode RUP**

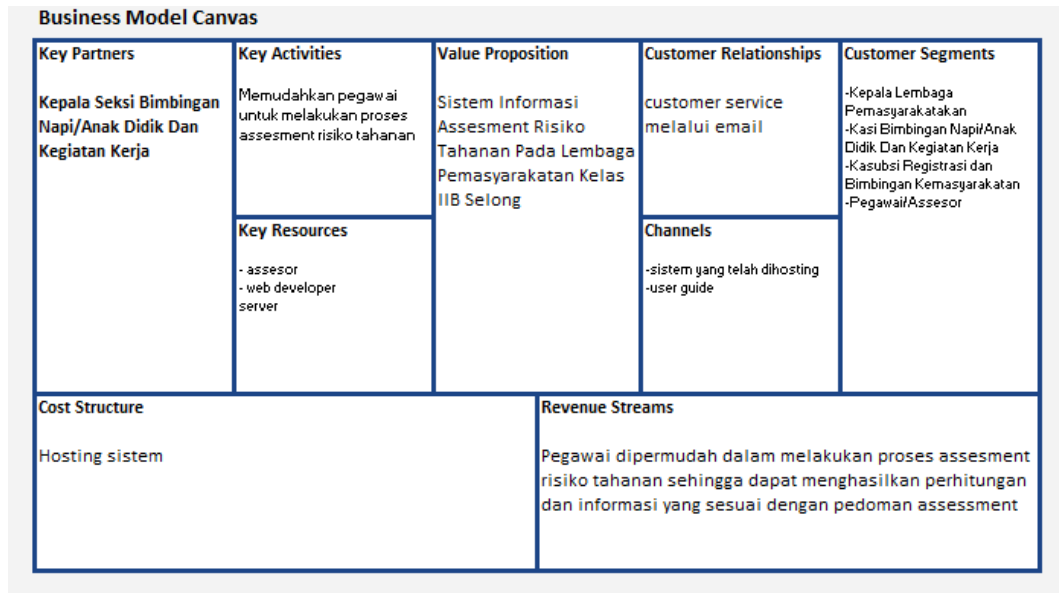
Pada pengembangan sistem dengan metode RUP terdapat 4 tahapan yaitu tahap *inception*, *elaboration*, *construction*, dan *transition*.

#### **3.1.3.1 Tahap *Inception***

Pada tahap ini yang dilakukan adalah membuat *business modeling*, *requirement* kebutuhan, analisis dan desain, serta *test inception*. Berikut ini merupakan penjelasan dari masing-masing proses pada tahap *inception*.

### 3.1.3.1.1 Business Modelling

Bisnis model *canvas* yang digunakan pada tahap awal untuk memulai perancangan sistem yaitu seperti yang terdapat pada Gambar 3.2.



Gambar 3.2 *Business modelling* sistem infomasi *assessment* risiko tahanan.

#### 1. Value Proposition

*Value Proposition* dalam penelitian yang akan dilakukan yaitu membuat sebuah *website* untuk Lembaga Pemasyarakatan Kelas IIB Selong yang akan membantu dalam proses *assessment* risiko tahanan yaitu proses pengolahan hasil *assessment*, mempermudah pegawai dalam mengolah informasi berdasarkan hasil klasifikasi tingkat risiko tahanan serta pelaporan hasil *assessment* risiko tahanan.

#### 2. Key Activities

*Key Activities* dalam penelitian yang akan dilakukan adalah memudahkan pegawai/assessor untuk melakukan proses *assessment* risiko tahanan. Salah satu contohnya yaitu proses pengolahan hasil klasifikasi *assessment* yang dapat langsung ditentukan tanpa menggunakan perhitungan manual.

#### 3. Key Resource

*Key Resource* dalam penelitian yang akan dilakukan adalah *web developer* untuk membangun sistem dan *server* untuk menyimpan data pengguna ataupun segala aktivitas yang dilakukan dalam sistem serta assessor yang akan melakukan proses *assesment*.

#### 4. *Key Partners*

*Key Partners* dalam penelitian yang akan dilakukan adalah Kepala Seksi Bimbingan Napi/Anak Didik Dan Kegiatan Kerja dari Lembaga Pemasyarakatan Kelas IIB Selong. Kerja sama dengan Kepala Seksi Bimbingan Napi/Anak Didik Dan Kegiatan Kerja terkait memudahkan pengembang dalam menganalisis kebutuhan *website* yang akan dibangun dan menyediakan sesuatu jika dibutuhkan.

#### 5. *Channels*

*Channels* dalam penelitian yang akan dilakukan adalah sebuah sistem yang telah di *hosting* untuk dapat digunakan sesuai dengan fungsionalitas yang telah ditentukan. Misalnya, melakukan *input data*, proses *assessment* dan sebagainya.

#### 6. *Customer Segments*

*Customer Segments* dalam penelitian yang akan dilakukan adalah mempermudah pegawai dalam mengolah informasi berdasarkan hasil klasifikasi tingkat risiko tahanan, mempermudah Kasubsi dalam melakukan verifikasi hasil *assessment* risiko tahanan, serta mempermudah Kasi dan Kalapas dalam melihat pelaporan hasil *assessment* risiko tahanan.

#### 7. *Customer Relationship*

*Customer Relationship* dalam penelitian yang akan dilakukan adalah adanya *customer service* untuk membantu pengguna jika mengalami kesulitan ataupun memiliki pertanyaan dengan mengirim email ke *customer service*.

#### 8. *Cost Structure*

*Cost Structure* dalam penelitian yang akan dilakukan adalah biaya *hosting website* yang akan dibangun agar dapat bekerja sesuai dengan yang dibutuhkan.

Tabel 3.1 merupakan rincian dari biaya *hosting website*:

Tabel 3.1 Rincian biaya.

| No | Nama          | Biaya                |
|----|---------------|----------------------|
| 1. | <i>Server</i> | Rp. 500.000,00/tahun |
| 2. | <i>Domain</i> | Rp. 125.000,00/tahun |

#### 9. *Revenue Streams*

*Revenue Streams* dalam penelitian yang akan dilakukan adalah dengan mulai digunakannya sistem informasi *assessment* risiko tahanan, maka akan

mempermudah pegawai dalam mengklasifikasikan tingkat risiko tahanan sehingga dapat menghasilkan perhitungan dan informasi yang sesuai dengan pedoman *assessment*.

#### 3.1.3.1.2 *Requirement*

Kebutuhan sistem dibedakan menjadi kebutuhan fungsional dan non fungsional. Kebutuhan fungsional terdiri dari kebutuhan perangkat keras (*hardware*) dan perangkat lunak (*software*).

##### a. **Kebutuhan fungsional sistem**

Kebutuhan Fungsional adalah kebutuhan yang berisi proses-proses apa saja/layanan apa saja yang nantinya harus disediakan oleh sistem, mencakup bagaimana sistem harus bereaksi pada *input* tertentu dan bagaimana perilaku sistem pada situasi tertentu.

Pada pemaparan kebutuhan fungsional sistem terdapat 27 aksi yang akan dilakukan oleh 5 aktor yaitu Admin, Kepala Lembaga Pemasyarakatan, Kepala Seksi Bimbingan Napi/Anak Didik dan Kegiatan Kerja, Kepala Subseksi Registrasi dan Bimbingan Kemasyarakatan, dan pegawai/assessor. Masing-masing aktor memiliki hak akses yang berbeda seperti yang terlihat pada Tabel 3.2 berikut.

Tabel 3.2 Kebutuhan fungsional sistem.

| No | Kebutuhan Fungsional    | Aktor   |
|----|-------------------------|---------|
| 1  | Menambah <i>user</i>    | Admin   |
| 2  | Mengubah <i>user</i>    |         |
| 3  | Menghapus <i>user</i>   |         |
| 4  | Menambah data tahanan   |         |
| 5  | Merubah data tahanan    |         |
| 6  | Menghapus data tahanan  |         |
| 7  | Melihat data assessor   | Kalapas |
| 8  | Melihat data tahanan    |         |
| 9  | Mengubah <i>profile</i> |         |
| 10 | Melihat grafik bulanan  |         |

|    |                                                                                                                   |          |
|----|-------------------------------------------------------------------------------------------------------------------|----------|
| 11 | Melihat laporan bulanan berupa nomer registrasi tahanan, nama tahanan, dan hasil <i>assessment</i> risiko tahanan |          |
| 12 | Mencetak laporan bulanan                                                                                          |          |
| 13 | Mengubah <i>profile</i>                                                                                           | Kasi     |
| 14 | Melihat grafik bulanan                                                                                            |          |
| 15 | Melihat laporan bulanan berupa nomer registrasi tahanan, nama tahanan, dan hasil <i>assessment</i> risiko tahanan |          |
| 16 | Mencetak laporan bulanan                                                                                          |          |
| 17 | Mengubah <i>profile</i>                                                                                           | Kasubsi  |
| 18 | Melihat grafik bulanan                                                                                            |          |
| 19 | Mencetak laporan bulanan                                                                                          |          |
| 20 | Melihat laporan bulanan berupa nomer registrasi tahanan, nama tahanan, dan hasil <i>assessment</i> risiko tahanan |          |
| 21 | Melakukan verifikasi hasil <i>assessment</i>                                                                      |          |
| 22 | Mengubah <i>profile</i>                                                                                           | Assessor |
| 23 | Melakukan <i>assessment</i>                                                                                       |          |
| 24 | Mengubah data <i>assessment</i>                                                                                   |          |
| 25 | Melihat laporan bulanan berupa nomer registrasi tahanan, nama tahanan, dan hasil <i>assessment</i> risiko tahanan |          |
| 26 | Mencetak laporan bulanan                                                                                          |          |
| 27 | Melihat grafik bulanan                                                                                            |          |

**b. Kebutuhan non-fungsional sistem**

Kebutuhan non-fungsional merupakan kebutuhan yang menitikberatkan pada properti perilaku yang dimiliki oleh sistem. Kebutuhan non-fungsional juga sering disebut sebagai batasan layanan atau fungsi yang ditawarkan sistem seperti batasan waktu, batasan pengembangan proses, standarisasi dan lain

lain. Kebutuhan non-fungsional sistem terdiri dari 2 kebutuhan yaitu seperti yang terdapat pada Tabel 3.3 berikut.

Tabel 3.3 Kebutuhan non-fungsional sistem.

| No | Kebutuhan Non-Fungsional                                                                                                                           |
|----|----------------------------------------------------------------------------------------------------------------------------------------------------|
| 1  | Tampilan disesuaikan dengan fungsi yang sudah dibuat.                                                                                              |
| 2  | Sistem dapat dijalankan di semua <i>web browser</i> seperti <i>Google Chrome</i> , <i>Mozilla Firefox</i> , <i>UC Browser</i> dan lain sebagainya. |

### 1) Kebutuhan perangkat keras (*Hardware*)

Kebutuhan akan perangkat keras (*hardware*) untuk mengimplementasikan sistem yaitu seperti yang terdapat pada Tabel 3.4 berikut.

Tabel 3.4 Kebutuhan perangkat keras.

| No | Keterangan       | Spesifikasi   | Fungsi                                                                                             |
|----|------------------|---------------|----------------------------------------------------------------------------------------------------|
| 1  | <i>Processor</i> | Intel Core i3 | Untuk memproses data yang diterima dari <i>input</i> sehingga menghasilkan <i>output</i>           |
| 2  | <i>Ram</i>       | 4 GB          | Sebagai pendukung bagi <i>processor</i> dalam <i>computing</i>                                     |
| 3  | <i>Hard disk</i> | 1 TB          | Sebagai media penyimpanan data dan program pada komputer yang bersifat permanen selama tidak rusak |
| 4  | <i>Monitor</i>   | Led 14 Inchi  | Untuk melihat tampilan program                                                                     |

### 2) Kebutuhan perangkat lunak (*Software*)

Kebutuhan akan perangkat lunak (*software*) pada pengembangan dan pengimplementasian sistem yaitu seperti yang terdapat pada Tabel 3.5 berikut.

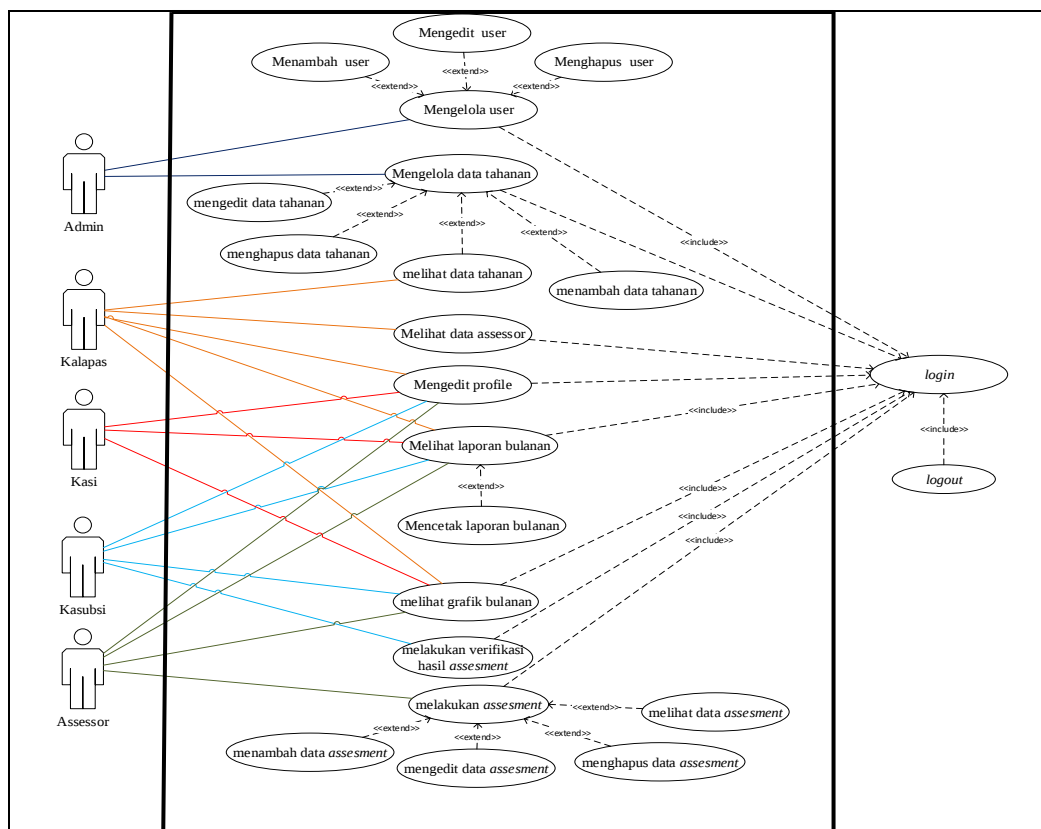
Tabel 3.5 Kebutuhan perangkat lunak.

| No | Keterangan     | Spesifikasi | Fungsi                                                                                                                                                   |
|----|----------------|-------------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1  | Sistem operasi | Windows 10  | Perangkat lunak sistem yang mengatur sumber daya dari perangkat keras dan perangkat lunak, serta sebagai jurik ( <i>daemon</i> ) untuk program komputer. |

|   |                      |               |                                                                                                                                                                                                                                                               |
|---|----------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 2 | <i>Xampp</i>         | Version 3.2.1 | Sebagai <i>server</i> yang berdiri sendiri ( <i>localhost</i> ), yang terdiri dari beberapa program diantaranya <i>Apache HTTP Server</i> , <i>MySQL database</i> , dan penerjemah bahasa yang ditulis dengan bahasa pemrograman <i>PHP</i> dan <i>Perl</i> . |
| 3 | <i>Sublime Text</i>  | Version 3     | <i>Text editor</i> yang akan digunakan untuk membuat <i>code</i> program dari aplikasi yang akan dibuat                                                                                                                                                       |
| 4 | <i>Google Chrome</i> | 70.0.3538.110 | Sebagai <i>browser</i> untuk menjalankan sistem                                                                                                                                                                                                               |

### 3.1.3.1.3 Analysis and Design inception

Analisis aplikasi pada tahap *inception* dijelaskan dengan menggunakan *use case diagram* seperti pada Gambar 3.3 berikut.



Gambar 3.3 Use case diagram sistem.

Berdasarkan Gambar 3.3 dapat diketahui bahwa pada sistem terdapat 5 aktor yang berperan yaitu Admin, Kepala Lembaga Pemasyarakatan (Kalapas), Kepala

Seksi Bimbingan Napi/Anak Didik Dan Kegiatan Kerja, Kepala Subseksi Registrasi dan Bimbingan Kemasyarakatan dan assessor, dimana masing-masing aktor tersebut memiliki hak akses yang berbeda. Admin memiliki 6 hak akses yaitu dapat menambah *user*, mengubah *user*, menghapus *user*, menambah data tahanan, mengubah data tahanan, dan menghapus data tahanan. Kepala Lembaga Pemasyarakatan (Kalapas) memiliki 6 hak akses yaitu dapat mengubah *profile*, melihat data assessor, melihat data tahanan, melihat grafik bulanan, melihat laporan bulanan, dan mencetak laporan bulanan. Untuk Kepala Seksi Bimbingan Napi/Anak Didik Dan Kegiatan Kerja memiliki 4 hak akses yaitu dapat mengubah *profile*, melihat grafik bulanan, melihat laporan bulanan, dan mencetak laporan bulanan. Kepala Subseksi Registrasi dan Bimbingan Kemasyarakatan memiliki 5 hak akses yaitu dapat mengubah *profile*, melihat grafik bulanan, melihat laporan bulanan, mencetak laporan bulanan, dan melakukan verifikasi hasil *assessment*. Sedangkan untuk assessor memiliki 9 hak akses yaitu dapat mengubah *profile*, melihat data tahanan, melihat data *assessment*, menambah data *assessment*, mengedit data *assessment*, menghapus data *assessment*, melihat grafik bulanan, melihat laporan bulanan, dan mencetak laporan bulanan. Semua akses yang diberikan dapat dilakukan jika sudah melakukan *login* ke sistem.

#### 3.1.3.1.4 *Test inception*

Pada tahap ini, pengujian bertujuan untuk memvalidasi kebutuhan sistem dan *use case* yang telah dirancang. Pengujian melibatkan 3 orang penguji yang berperan sebagai pengguna sistem yaitu Kalapas, Kasi dan assessor. Pengujian dilakukan dengan mencocokkan *use case* berdasarkan kebutuhan fungsional sistem untuk Admin, Kalapas, Kasi, Kasubsi dan assessor. Proses validasi dimulai dengan peneliti memberikan tabel isian yang kemudian diisi dan di validasi oleh masing-masing penguji pada lapas. Hasil validasi dapat dilihat pada Tabel 3.6 berikut.

Tabel 3.6 Validasi kesesuaian *use case* dengan kebutuhan sistem.

| No | Kebutuhan fungsional sistem                               | Use Case |         |      |         |          |
|----|-----------------------------------------------------------|----------|---------|------|---------|----------|
|    |                                                           | Admin    | Kalapas | Kasi | Kasubsi | Assessor |
| 1  | Mengelola <i>user</i> (menambah, mengubah, dan menghapus) | ✓        |         |      |         |          |

|   |                                                            |   |   |   |   |   |
|---|------------------------------------------------------------|---|---|---|---|---|
| 2 | Mengubah <i>profile</i>                                    |   | ✓ | ✓ | ✓ | ✓ |
| 3 | Melihat grafik bulanan                                     |   | ✓ | ✓ | ✓ | ✓ |
| 4 | Melihat laporan bulanan                                    |   | ✓ | ✓ | ✓ | ✓ |
| 5 | Mencetak laporan bulanan                                   |   | ✓ | ✓ | ✓ | ✓ |
| 6 | Melakukan verifikasi hasil <i>assessment</i>               |   |   |   | ✓ |   |
| 7 | Mengelola data tahanan (menambah, mengubah, dan menghapus) | ✓ |   |   |   |   |
| 8 | Melakukan <i>assessment</i>                                |   |   |   |   | ✓ |
| 9 | Mengubah data <i>assessment</i>                            |   |   |   |   | ✓ |

Berdasarkan validasi yang dilakukan diperoleh hasil bahwa kebutuhan dan *use case* sistem yang dirancang telah sesuai dengan yang diinginkan oleh pengguna. Maka proses dapat dilanjutkan menuju tahap selanjutnya yaitu tahap *elaboration*.

### 3.1.3.2 Tahap *Elaboration*

Tahap *elaboration* merupakan lanjutan dari tahap *inception* di mana pada tahap *inception*, *analysis and design* akan dilengkapi dengan menambahkan *activity diagram*, *sequence diagram*, *class diagram* dan desain *database*. Dilanjutkan dengan tahap *implementation* yaitu *prototype* yang berupa *digital prototype* dari desain *interface* sistem.

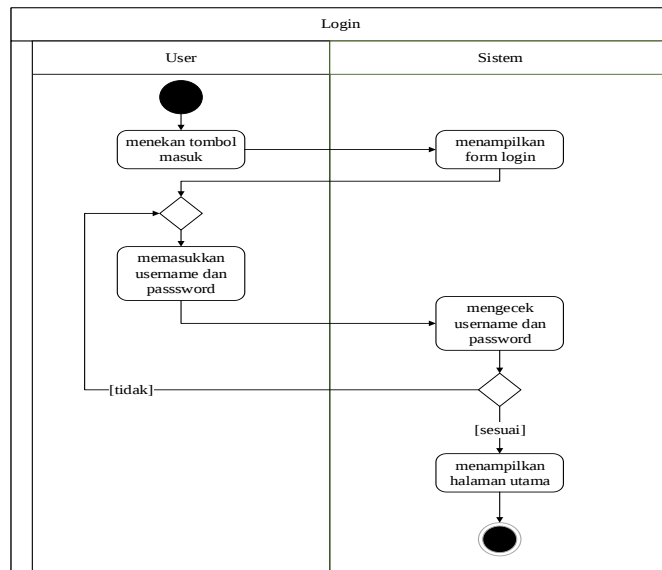
#### 3.1.3.2.1 *Analysis and Design*

Analisis dan desain pada tahap *elaboration* yaitu perancangan *activity diagram*, *sequence diagram*, *class diagram* dan desain *database*.

##### 1. *Activity Diagram*

Berikut ini adalah *Activity Diagram* dari Sistem Informasi *Assessment* Risiko Tahanan pada Lembaga Pemasyarakatan Kelas IIB Selong:

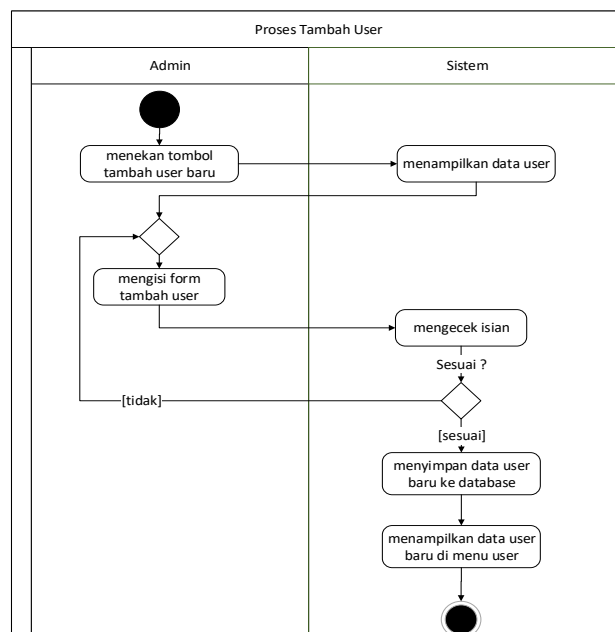
### a) Proses Login



Gambar 3.4 Proses login.

Pada Gambar 3.4 merupakan proses *login* yang dapat dilakukan oleh *user*. *User* dapat melakukan aktivitas *login* untuk masuk ke sistem dengan memasukkan *username* serta *password* yang telah terdaftar pada sistem, kemudian sistem akan melakukan pengecekan. Jika masukan *user* benar atau sesuai, maka *user* akan diarahkan ke halaman utama dan jika salah akan diarahkan untuk melakukan pengisian ulang.

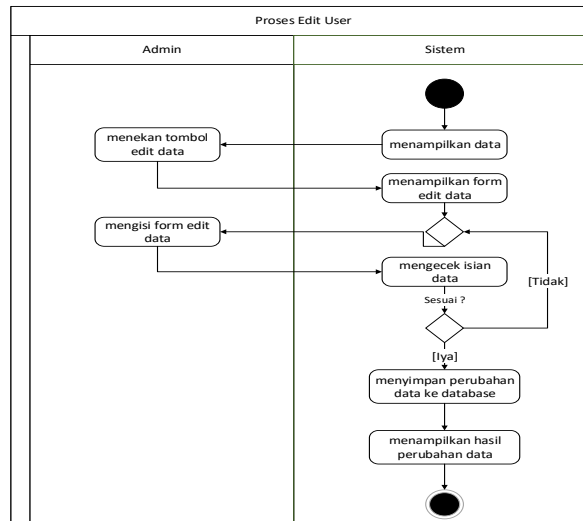
### b) Proses menambahkan user



Gambar 3.5 Proses menambahkan user.

Gambar 3.5 merupakan *activity diagram* dari proses menambahkan *user*. Aktivitas menambah *user* baru dapat dilakukan oleh admin. Admin harus mengisi *form* tambah *user* untuk selanjutnya dicek oleh sistem dan disimpan datanya ke *database*.

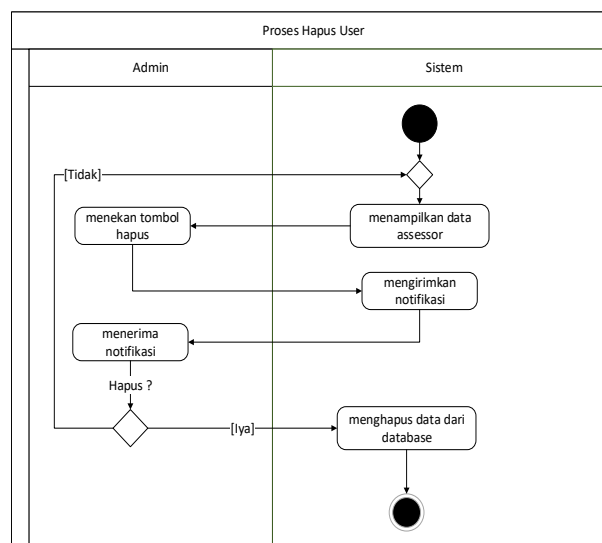
**c) Proses mengubah *user***



Gambar 3.6 Mengubah *user*.

Gambar 3.6 merupakan *activity diagram* dari proses mengubah *user*. Aktivitas mengubah data assessor dan berkas dapat dilakukan oleh admin melalui menu data *user* dengan mengisi *form* terlebih dahulu untuk dicek dan selanjutnya disimpan ke *database*.

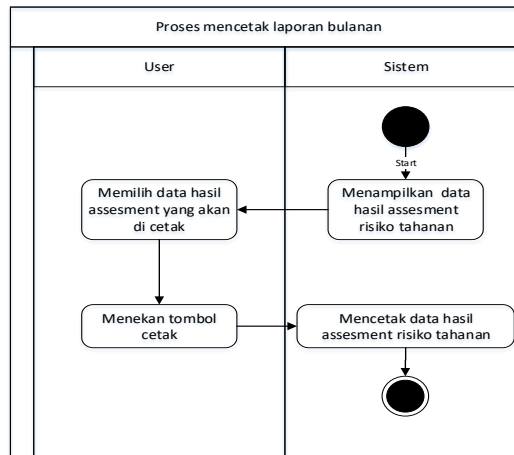
**d) Proses menghapus *user***



Gambar 3.7 Menghapus *user*.

Gambar 3.7 merupakan *activity diagram* dari proses menghapus *user*. Aktivitas menghapus *user* dapat dilakukan oleh admin dengan menekan tombol hapus dan kemudian sistem akan menghapus data dari *database*.

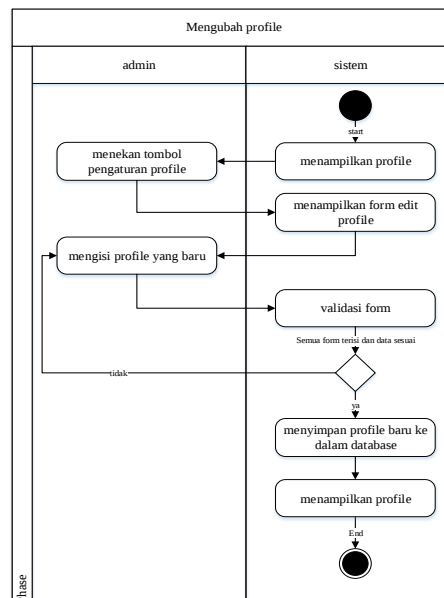
**e) Proses mencetak laporan bulanan**



Gambar 3.8 Mencetak laporan bulanan.

Gambar 3.8 merupakan *activity diagram* dari proses mencetak laporan bulanan. *User* dapat mencetak laporan bulanan untuk data hasil *assessment* risiko tahanan dengan menekan tombol cetak pada data tersebut, maka hasil pemeriksaan akan langsung tercetak.

**f) Proses mengubah *profile***

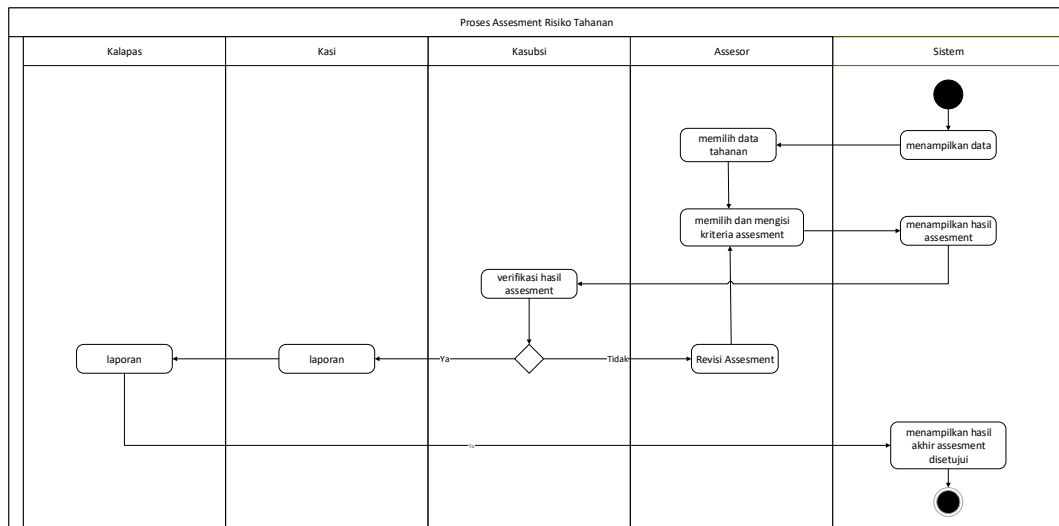


Gambar 3.9 Activity diagram proses mengubah *profile*

Gambar 3.9 merupakan *activity diagram* dari proses mengubah *profile*. Aktivitas mengubah *profile* dapat dilakukan oleh seluruh *user* dengan terlebih

dahulu sistem akan menampilkan halaman *profile* dari *user*, setelah itu sistem akan menampilkan form ubah *profile*. Dalam form tersebut *user* akan mengisi data yang baru seperti *username*, NIP, dan *password*. Setelah dilakukan validasi kesesuaian data maka *profile user* yang baru tersebut dapat disimpan dalam *database*.

**g) Proses melakukan *assessment***

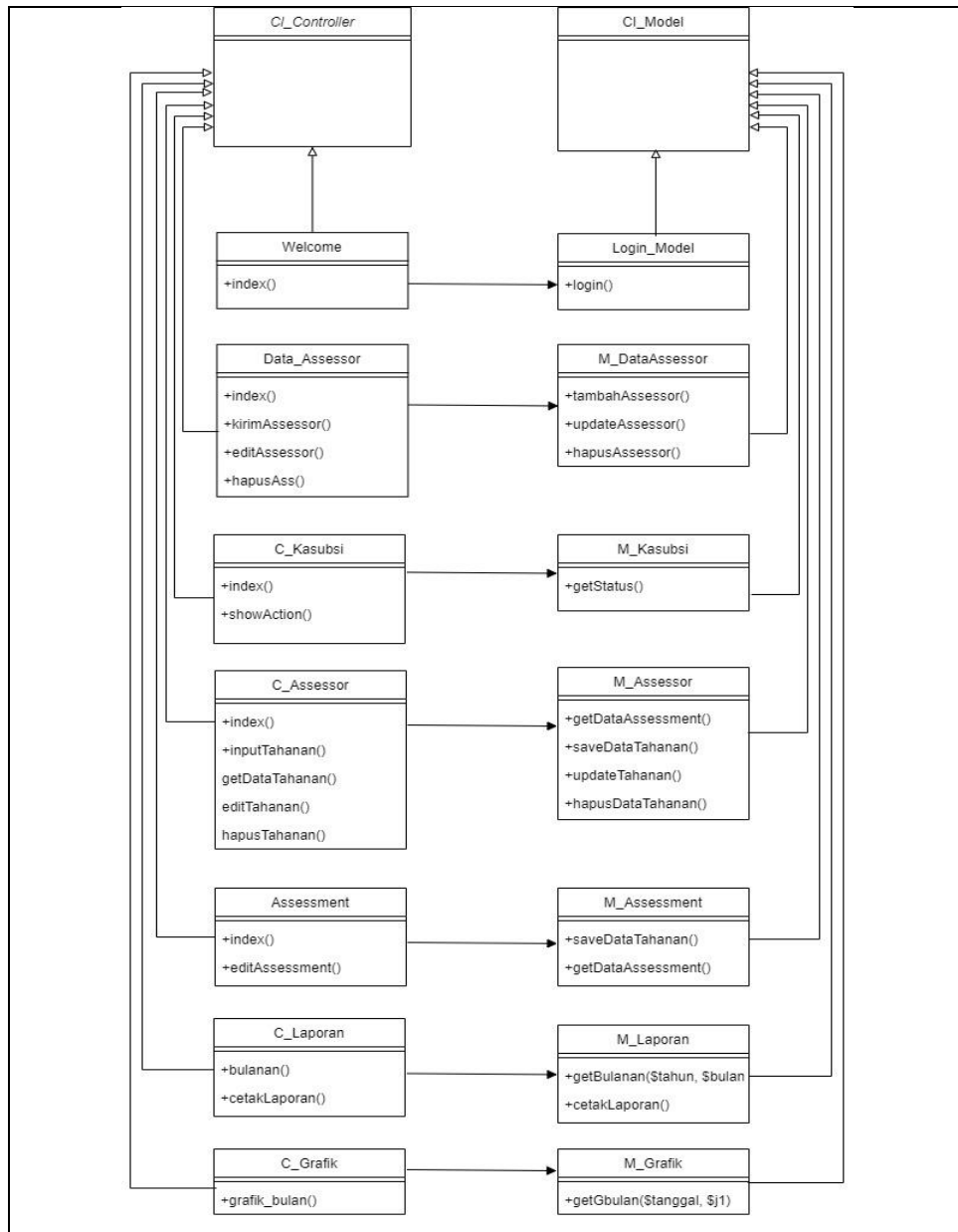


Gambar 3.10 Proses melakukan *assessment*.

Pada Gambar 3.10 merupakan *activity diagram* dari proses melakukan *assessment* risiko tahanan. Pada diagram tersebut terdapat 4 aktor di dalamnya. Proses melakukan *assessment* risiko tahanan dimulai dari sistem akan menampilkan data tahanan, kemudian assesor akan memilih data tahanan yang akan dilakukan *assessment*. Selanjutnya assesor akan mengisi setiap kriteria yang terdapat pada form *assessment*. Jika *assessment* telah selesai dilakukan, maka hasil *assessment* akan diverifikasi oleh Kepala Subsi Registrasi Dan Bimbingan Kemasyarakatan. Apabila hasil *assessment* sudah sesuai maka akan dilaporkan ke Kepala Seksi Bimbingan Napi/Anak Didik Dan Kegiatan Kerja serta Kepala Lembaga Pemasyarakatan. Selanjutnya, sistem akan menampilkan hasil *assessment* yang telah disetujui. Akan tetapi, jika masih terdapat kesalahan maka hasil *assessment* akan direvisi kembali oleh assesor.

## 2. Class Diagram

Keterhubungan antar kelas yang terdapat pada sistem dapat dilihat pada Gambar 3.11 berikut.



Gambar 3.11 Class diagram.

Berdasarkan Gambar 3.11, dapat diketahui bahwa sistem yang akan dibangun menggunakan pola arsitektur *Models Views Controllers* (MVC), sehingga dilakukan perancangan kelas – kelas yang ada pada sistem yang terdiri dari kelas *Controllers* dan *Models*. Pada *Controllers* terdiri dari *Welcome*, *Data\_User*, *C\_Kasubsi*, *C\_Assessor*, *Assessment*, *C\_Laporan*, dan *C\_Grafik*. Sedangkan pada model terdiri dari *Login\_model*, *M\_DataUser*, *M\_Kasubsi*,

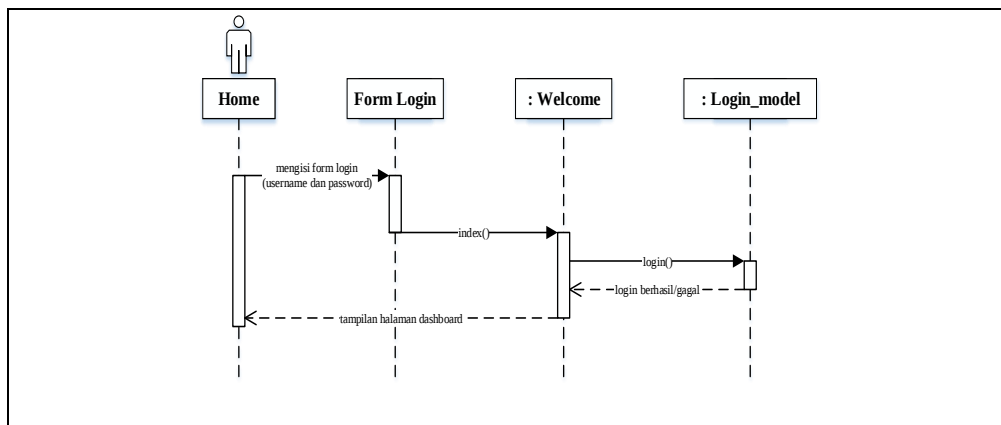
M\_Assessor, M\_Assessment, M\_Laporan, dan M\_Grafik. *Controller* Welcome berhubungan dengan *controller* Data\_Assessor, C\_Kasubsi, C\_Assessor, Assessment, C\_Laporan, C\_Grafik, dan *model* Login\_model. *Controller* Data\_User berhubungan dengan *model* M\_DataUser. *Controller* C\_Kasubsi berhubungan dengan *model* M\_Kasubsi. *Controller* C\_Assessor berhubungan dengan *model* M\_Assessor. *Controller* Assessment berhubungan dengan *model* M\_Assessment. *Controller* C\_Laporan berhubungan dengan *model* M\_Laporan. *Controller* Assessment berhubungan dengan *model* M\_Assessment. *Controller* C\_Grafik berhubungan dengan *model* M\_Grafik.

### 3. Sequence Diagram

Berikut adalah *Sequence Diagram* pada Sistem Informasi Assessment Risiko Tahanan:

#### a) Proses Login

Semua proses dapat dilakukan dengan melakukan *login* terlebih dahulu. *Sequence diagram* proses *login* untuk Kepala Lembaga Pemasyarakatan, Kepala Seksi Bimbingan Napi/Anak Didik dan Kegiatan Kerja, Kasubsi Registrasi dan Bimbingan Kemasyarakatan, serta pegawai/assessor yaitu seperti yang terdapat pada Gambar 3.12.

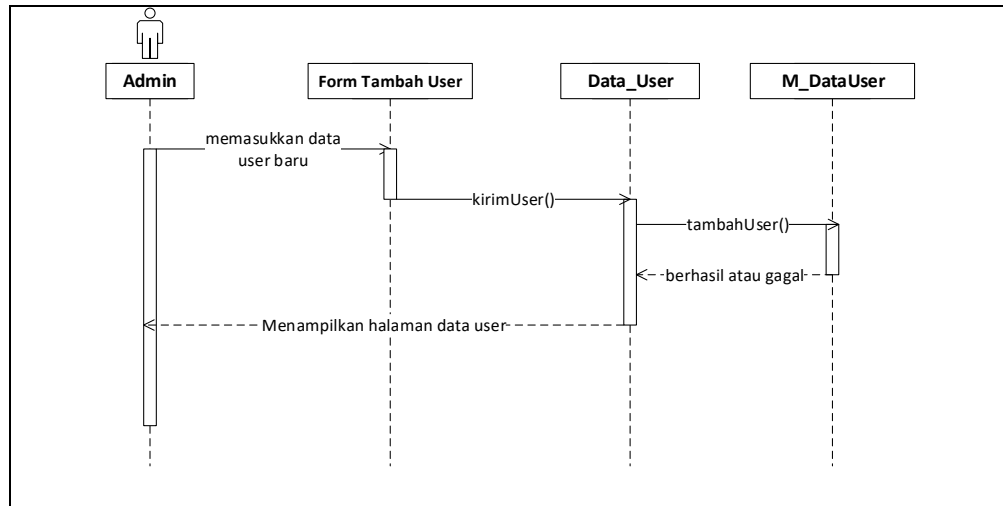


Gambar 3.12 Proses *login* Kalapas, Kasi, Kasubsi, dan assessor.

Pada Gambar 3.12 merupakan *sequence diagram* dari proses *login*. Proses *login* dimulai ketika *user* memasukan *username* dan *password* pada *form login* yang terdapat pada halaman utama. Kemudian hasil *input form* tersebut akan dikirim ke *class* `Welcome` di *Controller* untuk diperiksa dan dilakukan validasi apakah *username* dan *password* yang dimasukkan sesuai dengan yang ada pada *database* atau tidak dengan menggunakan fungsi

`login()`. Ketika mengambil data dari *database* maka kelas `Login_model` yang berada pada *Model* akan dipanggil. Jika ada dan sesuai dengan data yang ada pada *database*, maka akan diarahkan ke halaman *dashboard*.

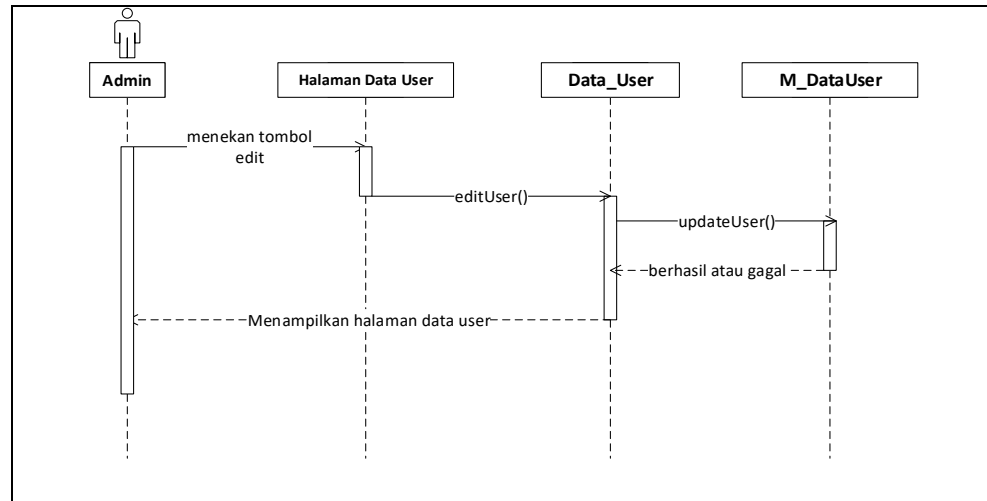
**b) Proses menambah user**



Gambar 3.13 Proses menambah user.

Pada Gambar 3.13 merupakan *sequence diagram* dari proses menambah user. Proses menambah assessor dimulai ketika admin memasukkan data user baru pada *view form* tambah assessor. Hasil *input* data user tersebut akan dikirim ke kelas `Data_User` yang merupakan *controller* untuk diolah dengan menggunakan fungsi `kirimUser()`. Kemudian data tersebut selanjutnya akan ditambahkan ke dalam *database* dengan menggunakan *class* `M_DataUser` dan fungsi yang ada di dalamnya yaitu fungsi `tambahUser()`. Ketika data sudah berhasil ditambahkan ke dalam *database*, maka selanjutnya admin akan diarahkan ke halaman data user.

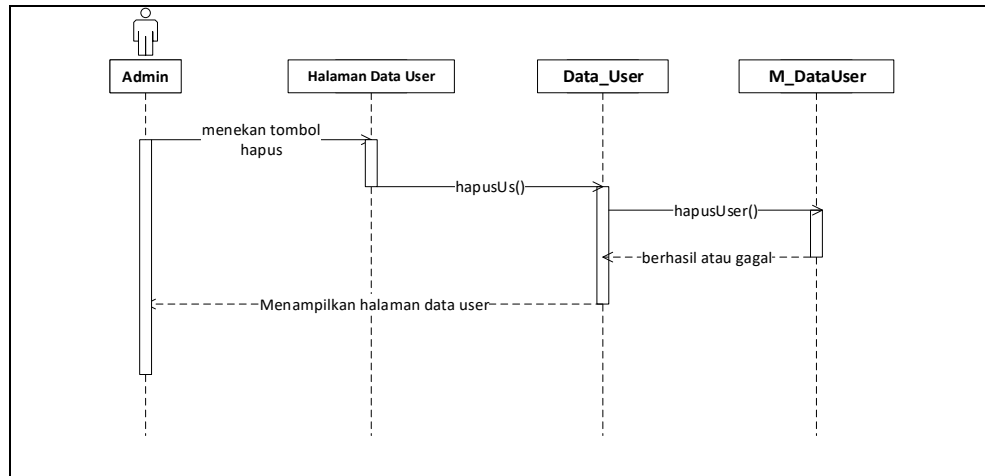
c) **Proses mengubah user**



Gambar 3.14 Proses mengubah user.

Pada Gambar 3.14 merupakan *sequence diagram* dari proses mengubah user. Proses mengubah data user dimulai ketika admin menekan tombol edit pada view halaman data user, kemudian memasukkan data user yang baru ke dalam halaman data user. Hasil *input* data pegawai yang baru tersebut akan dikirim ke kelas `Data_User` yang merupakan *controller* untuk diolah dengan menggunakan fungsi `editUser()`. Kemudian data tersebut selanjutnya akan ditambahkan ke dalam *database* dengan menggunakan *class* `M_DataUser` dan fungsi yang ada di dalamnya yaitu fungsi `updateUser()`. Ketika data yang sudah diubah berhasil ditambah ke dalam *database*, maka selanjutnya admin akan diarahkan ke halaman data user.

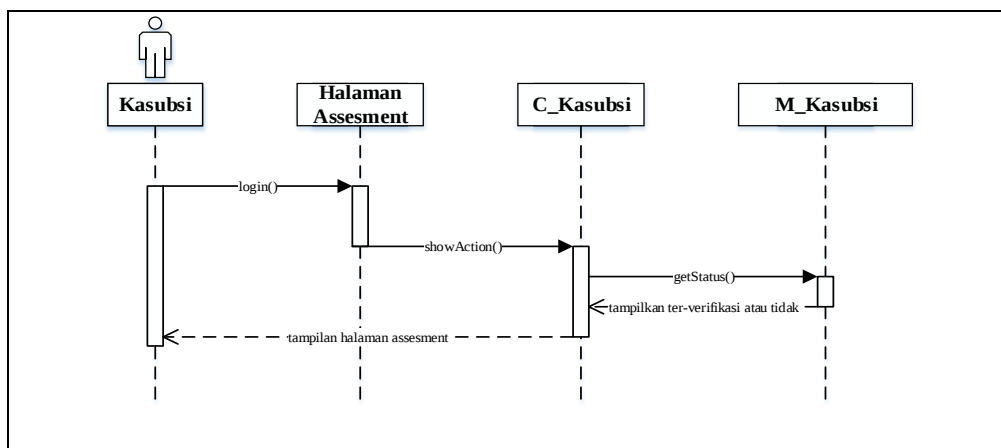
d) **Proses menghapus user**



Gambar 3.15 Proses menghapus user.

Pada Gambar 3.15 merupakan *sequence diagram* dari proses menghapus user. Proses menghapus user dimulai ketika menekan tombol hapus pada view halaman data user. Fungsi yang digunakan untuk menghapus data user ini yaitu fungsi `hapusUs()` yang berada pada *class* `Data_User` yang terdapat dalam *controller*. Kemudian data tersebut selanjutnya akan dihapus dari dalam *database* dengan menggunakan *class* `M_DataUser` dan fungsi yang ada di dalamnya yaitu fungsi `hapusUser()`. Ketika data yang sudah dipilih berhasil dihapus dari dalam *database*, maka selanjutnya admin akan diarahkan ke halaman data user.

e) **Proses memverifikasi hasil assessment**

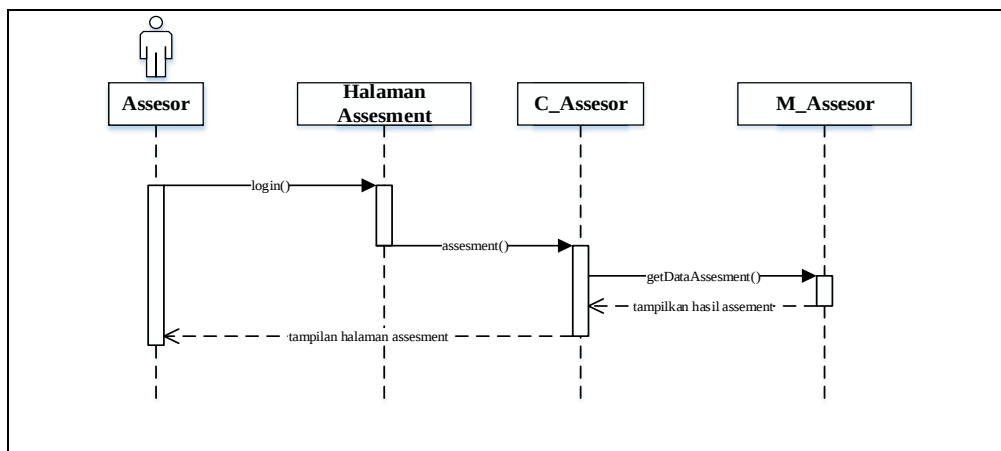


Gambar 3.16 Proses memverifikasi hasil assessment.

Pada Gambar 3.16 merupakan *sequence diagram* dari proses memverifikasi hasil assessment. Proses verifikasi hasil assessment dimulai ketika Kasubsi login kemudian akan ditampilkan halaman *assessment* yang

berisi data tahanan yang telah dilakukan *assessment* yang terdapat pada *database*. Fungsi yang digunakan untuk memverifikasi hasil *assessment* ini yaitu fungsi `showAction()` yang berada pada *class* `C_Kasubsi` yang terdapat dalam *controller*. Kemudian untuk dapat mengakses *database* agar data yang telah diverifikasi dapat ditampilkan pada sistem dengan menggunakan *class* `M_Kasubsi` dan fungsi yang ada di dalamnya yaitu fungsi `getStatus()`. Ketika data yang terverifikasi berhasil diakses dari dalam *database*, maka selanjutnya Kasubsi akan diarahkan ke halaman *assessment*.

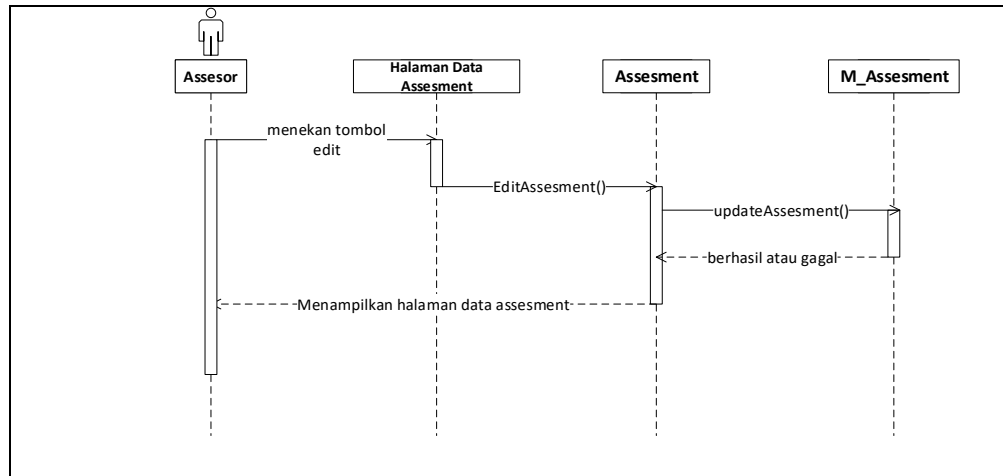
**f) Proses melakukan *assessment***



Gambar 3.17 Proses melakukan *assessment*.

Pada Gambar 3.17 merupakan *sequence diagram* dari proses melakukan *assessment*. Proses melakukan *assessment* dimulai ketika Assesor login kemudian akan ditampilkan halaman *assessment*. Fungsi yang digunakan untuk melakukan *assessment* ini yaitu fungsi `assesment()` yang berada pada *class* `C_Assessor` yang terdapat dalam *controller*. Kemudian untuk dapat mengakses *database* agar data yang telah di *assessment* dapat ditampilkan pada sistem menggunakan *class* `M_Assessor` dan fungsi yang ada di dalamnya yaitu fungsi `getDataAssesment()`. Ketika data yang terverifikasi berhasil diakses dari dalam *database*, maka selanjutnya Kasubsi akan diarahkan ke halaman *assessment*.

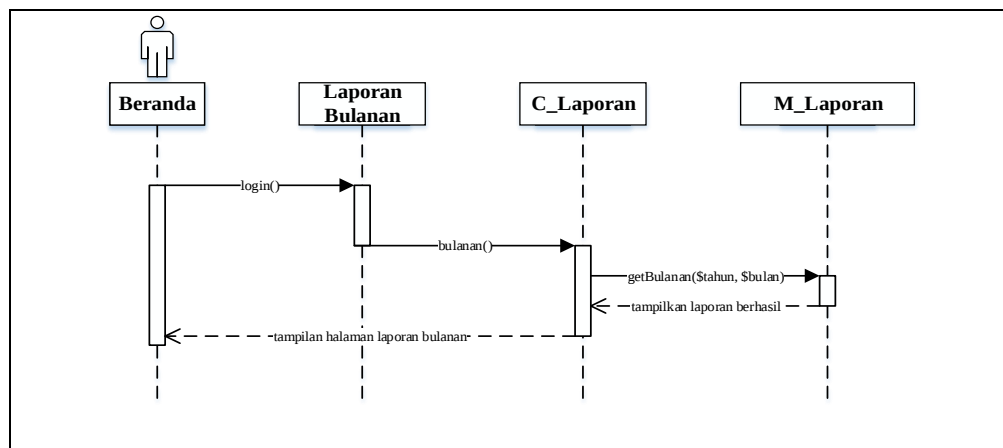
g) **Proses mengubah data *assessment***



Gambar 3.18 Proses mengubah data *assessment*.

Pada Gambar 3.18 merupakan *sequence diagram* dari proses mengubah data *assessment*. Proses mengubah data *assessment* dimulai ketika Assessor menekan tombol edit pada *view* halaman data *assessment*, kemudian memasukkan data assessor yang baru ke dalam halaman data assessor. Hasil *input* data *assessment* yang baru tersebut akan dikirim ke kelas *Assesment* yang merupakan *controller* untuk diolah dengan menggunakan fungsi `editAssesment()`. Kemudian data tersebut selanjutnya akan diperbaharui dalam *database* dengan menggunakan *class* *M\_Assessment* dan fungsi yang ada di dalamnya yaitu fungsi `updateAssesment()`. Ketika data sudah berhasil diperbarui dalam *database*, maka selanjutnya assessor akan diarahkan ke halaman data assessor.

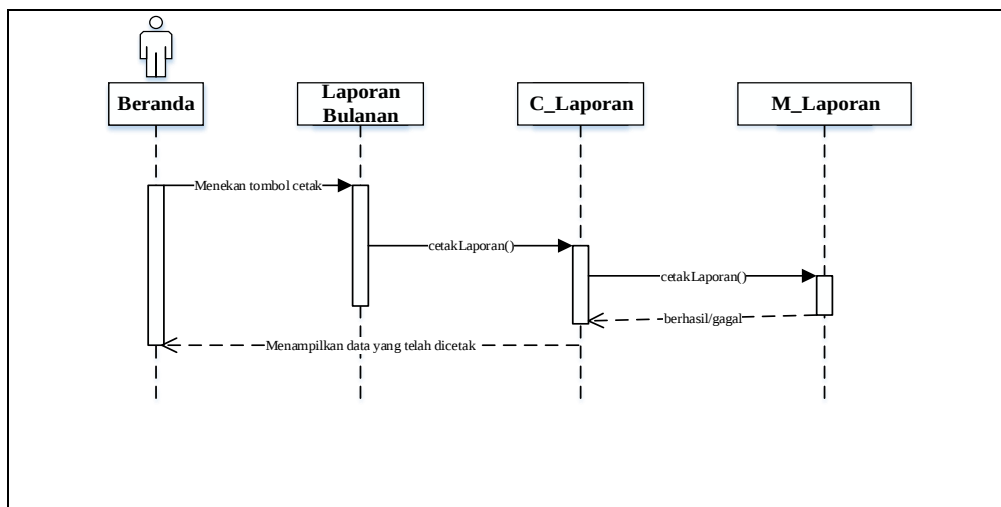
h) **Proses melihat laporan bulanan**



Gambar 3.19 Proses melihat laporan bulanan.

Pada Gambar 3.19 terdapat proses laporan bulanan dimulai ketika *user login* kemudian akan muncul laporan bulanan yang berisi data hasil *assessment* risiko tahanan yang terdapat pada *database*. Akan memanggil sebuah fungsi di *C\_Laporan* dengan yaitu fungsi *bulanan()*. Selanjutnya dipanggil fungsi yang berada pada *M\_Laporan*, yaitu fungsi *getBulanan(\$tahun, \$bulan)*. Fungsi ini digunakan untuk mengakses *database* agar data hasil *assessment* risiko tahanan dapat ditampilkan pada sistem dengan kelas mengakses kelas *view*.

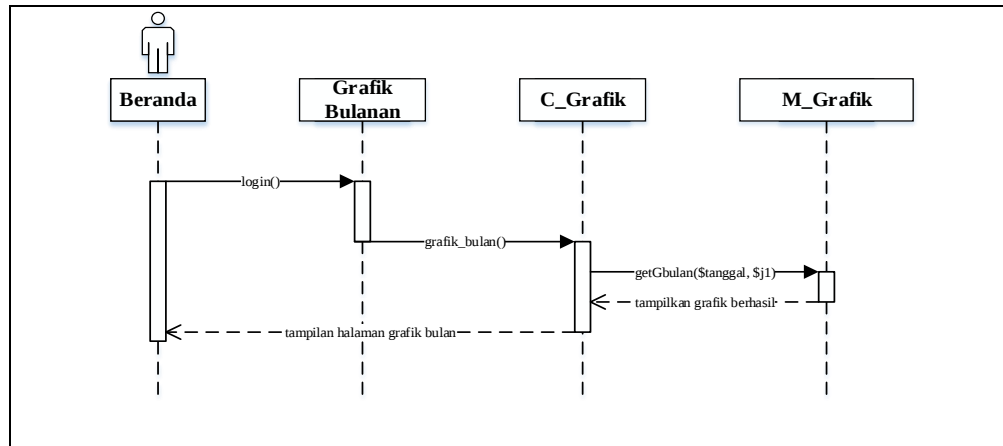
**i) Proses mencetak laporan bulanan**



Gambar 3.20 Proses mencetak laporan bulanan.

Pada Gambar 3.20 terdapat proses mencetak laporan bulanan dimulai ketika *user* menekan tombol cetak pada *view* halaman laporan, kemudian akan memanggil sebuah fungsi di *C\_Laporan* dengan fungsi *cetakLaporan()*. Selanjutnya dipanggil fungsi yang berada pada *M\_Laporan*, yaitu fungsi *cetakLaporan()*.

j) **Proses melihat grafik bulanan**

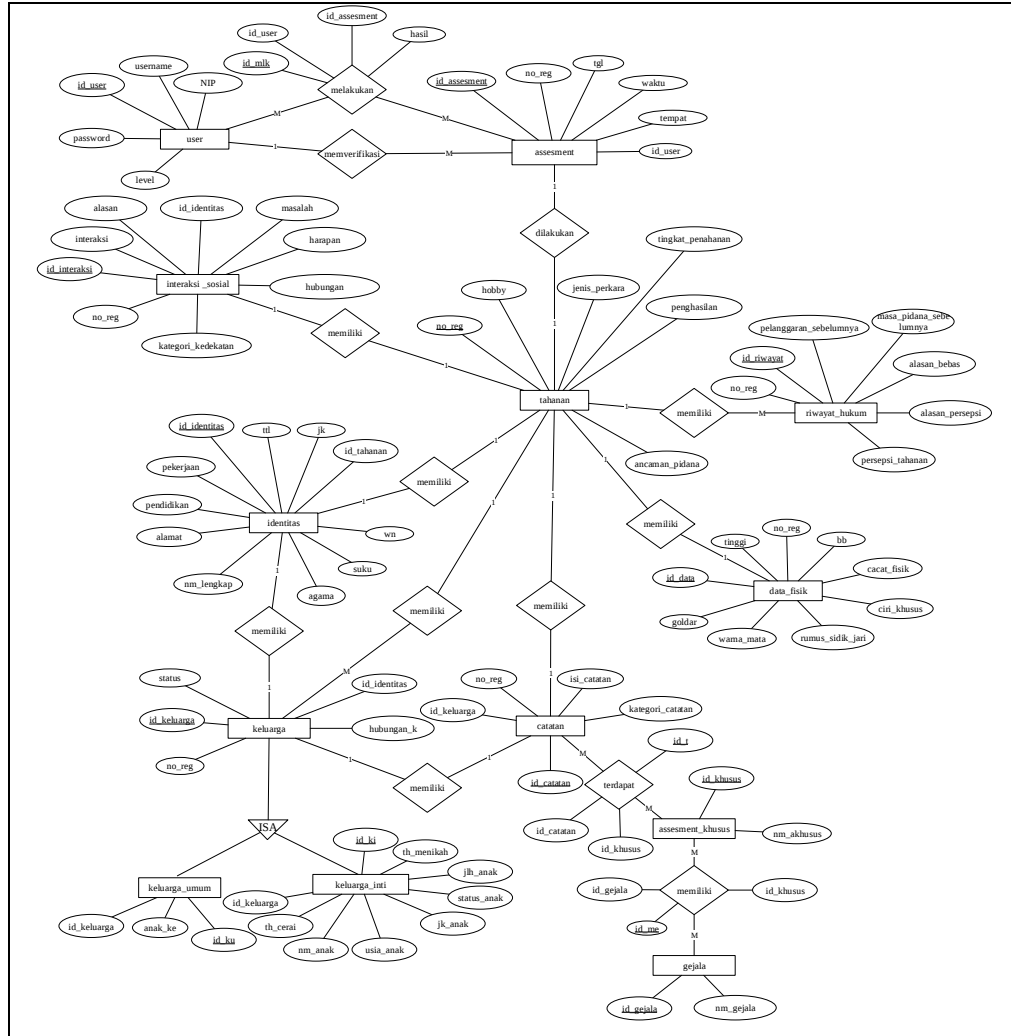


Gambar 3.21 Proses melihat grafik bulanan.

Pada Gambar 3.21 merupakan *sequence diagram* dari proses melihat grafik bulanan. Proses menampilkan grafik bulanan dimulai ketika *user* melakukan *login* kemudian akan muncul data hasil *assessment* risiko tahanan dalam bentuk grafik. Pada kelas `C_Grafik` akan dipanggil fungsi `grafik_bulanan()`. Selanjutnya dipanggil fungsi yang berada pada `M_Grafik`, yaitu fungsi `getGbulan($tanggal, $j1)`. Fungsi ini digunakan untuk mengakses *database* agar data hasil *assessment* risiko tahanan dapat ditampilkan pada sistem dengan kelas mengakses kelas *view*.

#### 4. Desain Database

Desain *Database* sistem yang akan dibangun dapat dilihat pada Gambar 3.22 berikut.



Gambar 3.22 Desain *database*.

Berdasarkan ERD pada Gambar 3.22 maka didapatkan tabel-tabel *database* sebagai berikut:

1. Tabel *user*

Tabel *user* merupakan tabel yang digunakan untuk menampung data pengguna yang melakukan *login*. Adapun struktur tabel *user* dapat dilihat pada Tabel 3.7.

Tabel 3.7 Tabel *user*.

| No | Field Name | Type | Width |
|----|------------|------|-------|
| 1  | id_user    | Int  | 5     |

|   |          |         |    |
|---|----------|---------|----|
| 2 | NIP      | Int     | 18 |
| 3 | Level    | Int     | 1  |
| 4 | Username | Varchar | 30 |
| 5 | password | Varchar | 30 |

## 2. Tabel *assessment*

Tabel *assessment* merupakan tabel yang digunakan untuk menampung data hasil *assessment* risiko tahanan. Adapun struktur tabel *assessment* dapat dilihat pada Tabel 3.8.

Tabel 3.8 Tabel *assessment*.

| No | Field Name    | Type    | Width |
|----|---------------|---------|-------|
| 1  | id_assessment | Int     | 5     |
| 2  | no_reg        | Varchar | 10    |
| 3  | id_user       | Int     | 5     |
| 4  | tgl           | Date    |       |
| 5  | Waktu         | Varchar | 10    |
| 6  | Tempat        | Varchar | 30    |

## 3. Tabel melakukan

Tabel melakukan merupakan tabel yang diperoleh dari kardinalitas *many to many* (M to M) dari tabel *user* dan tabel *assessment*. Adapun struktur tabel melakukan dapat dilihat pada Tabel 3.9.

Tabel 3.9 Tabel melakukan.

| No | Field Name    | Type    | Width |
|----|---------------|---------|-------|
| 1  | id_mlk        | Int     | 5     |
| 2  | id_user       | Int     | 5     |
| 3  | id_assessment | Int     | 5     |
| 4  | Hasil         | Varchar | 10    |

## 4. Tabel tahanan

Tabel tahanan merupakan tabel yang digunakan untuk menampung data *assessment* risiko tahanan. Adapun struktur tabel tahanan dapat dilihat pada Tabel 3.10.

Tabel 3.10 Tabel tahanan.

| <b>No</b> | <b>Field Name</b> | <b>Type</b> | <b>Width</b> |
|-----------|-------------------|-------------|--------------|
| 1         | no_reg            | Int         | 5            |
| 2         | Id_riwayat        | Int         | 5            |
| 3         | Hobby             | Varchar     | 30           |
| 4         | Jenis_perkara     | Varchar     | 30           |
| 5         | Tingkat_penahanan | Varchar     | 30           |
| 6         | Penghasilan       | Varchar     | 30           |
| 7         | Ancaman_pidana    | Varchar     | 30           |

#### 5. Tabel riwayat hukum

Tabel riwayat hukum merupakan tabel yang digunakan untuk menampung data riwayat hukum tahanan. Adapun struktur tabel riwayat hukum dapat dilihat pada Tabel 3.11.

Tabel 3.11 Tabel riwayat hukum.

| <b>No</b> | <b>Field Name</b>      | <b>Type</b> | <b>Width</b> |
|-----------|------------------------|-------------|--------------|
| 1         | Id_riwayat             | Int         | 5            |
| 2         | Pelanggaran_sebelumnya | Int         | 5            |
| 3         | Masa_pidana_sebelumnya | Varchar     | 30           |
| 4         | Alasan_bebas           | Varchar     | 30           |
| 5         | Alasan_persespsi       | Varchar     | 30           |
| 6         | Persepsi_tahanan       | Varchar     | 30           |

#### 6. Tabel data fisik

Tabel data fisik merupakan tabel yang digunakan untuk menampung data fisik tahanan. Adapun struktur tabel data fisik dapat dilihat pada Tabel 3.12.

Tabel 3.12 Tabel data fisik.

| <b>No</b> | <b>Field Name</b> | <b>Type</b> | <b>Width</b> |
|-----------|-------------------|-------------|--------------|
| 1         | id_data           | Int         | 5            |
| 2         | no_reg            | Int         | 5            |
| 3         | Tinggi            | Int         | 3            |
| 4         | Bb                | Int         | 3            |
| 5         | cacat_fisik       | Varchar     | 30           |
| 6         | ciri_khusus       | Varchar     | 30           |

|   |                  |         |    |
|---|------------------|---------|----|
| 7 | rumus_sidik_jari | Varchar | 30 |
| 8 | warna_mata       | Varchar | 30 |
| 9 | Goldar           | Varchar | 3  |

#### 7. Tabel catatan

Tabel catatan merupakan tabel yang digunakan untuk menampung catatan tambahan seperti catatan pendidikan, pekerjaan, masalah ekonomi, dan keluarga. Adapun struktur tabel catatan dapat dilihat pada Tabel 3.13.

Tabel 3.13 Tabel catatan.

| No | Field Name       | Type    | Width |
|----|------------------|---------|-------|
| 1  | id_catatan       | Int     | 5     |
| 2  | no_reg           | Int     | 5     |
| 3  | id_keluarga      | Varchar | 30    |
| 4  | Isi_catatan      | Text    |       |
| 5  | Kategori_catatan | Varchar | 30    |

#### 8. Tabel *assessment* khusus

Tabel *assessment* khusus merupakan tabel yang digunakan untuk menampung *assessment* khusus. Adapun struktur tabel *assessment* khusus dapat dilihat pada Tabel 3.14.

Tabel 3.14 Tabel *assessment* khusus.

| No | Field Name | Type    | Width |
|----|------------|---------|-------|
| 1  | id_khusus  | Int     | 5     |
| 2  | nm_akhusus | Varchar | 30    |

#### 9. Tabel terdapat

Tabel terdapat merupakan merupakan tabel yang diperoleh dari kardinalitas *many to many* (M to M) dari tabel *assessment* khusus dan tabel catatan. Adapun struktur tabel terdapat dapat dilihat pada Tabel 3.15.

Tabel 3.15 Tabel terdapat.

| No | Field Name | Type | Width |
|----|------------|------|-------|
| 1  | id_t       | Int  | 5     |
| 2  | Id_khusus  | Int  | 5     |
| 3  | Id_catatan | Int  | 5     |

10. Tabel gejala

Tabel gejala merupakan tabel yang digunakan untuk menampung data-data gejala. Adapun struktur tabel gejala dapat dilihat pada Tabel 3.16.

Tabel 3.16 Tabel gejala.

| No | Field Name | Type    | Width |
|----|------------|---------|-------|
| 1  | id_gejala  | Int     | 5     |
| 2  | nm_gejala  | Varchar | 30    |

11. Tabel memiliki

Tabel memiliki melakukan merupakan tabel yang diperoleh dari kardinalitas *many to many* (M to M) dari tabel gejala dan tabel *assessment* khusus. Adapun struktur tabel memiliki dapat dilihat pada Tabel 3.17.

Tabel 3.17 Tabel memiliki.

| No | Field Name | Type | Width |
|----|------------|------|-------|
| 1  | id_me      | Int  | 5     |
| 2  | id_gejala  | Int  | 5     |
| 3  | id_khusus  | Int  | 5     |

12. Tabel keluarga

Tabel keluarga merupakan tabel yang digunakan untuk menampung data keluarga tahanan. Adapun struktur tabel keluarga dapat dilihat pada Tabel 3.18.

Tabel 3.18 Tabel keluarga.

| No | Field Name   | Type    | Width |
|----|--------------|---------|-------|
| 1  | id_keluarga  | Int     | 5     |
| 2  | no_reg       | Int     | 5     |
| 3  | id_identitas | Int     | 5     |
| 4  | Status       | Varchar | 30    |
| 5  | hubungan_k   | Varchar | 30    |

13. Tabel keluarga umum

Tabel keluarga umum merupakan tabel yang digunakan untuk menampung data keluarga umum. Adapun struktur tabel keluarga umum dapat dilihat pada Tabel 3.19.

Tabel 3.19 Tabel keluarga umum.

| <b>No</b> | <b>Field Name</b> | <b>Type</b> | <b>Width</b> |
|-----------|-------------------|-------------|--------------|
| 1         | id_ku             | Int         | 5            |
| 2         | Id_keluarga       | Int         | 5            |
| 3         | Anak_ke           | Int         | 5            |

14. Tabel keluarga inti

Tabel keluarga inti merupakan tabel yang digunakan untuk menampung data keluarga inti. Adapun struktur tabel keluarga inti dapat dilihat pada Tabel 3.20.

Tabel 3.20 Tabel keluarga inti.

| <b>No</b> | <b>Field Name</b> | <b>Type</b> | <b>Width</b> |
|-----------|-------------------|-------------|--------------|
| 1         | id_ki             | Int         | 5            |
| 2         | id_keluarga       | Int         | 5            |
| 3         | th_menikah        | Int         | 5            |
| 4         | jlh_anak          | Int         | 5            |
| 5         | status_anak       | Varchar     | 30           |
| 6         | jk_anak           | Varchar     | 3            |
| 7         | usia_anak         | Int         | 3            |
| 8         | nm_anak           | Varchar     | 30           |
| 9         | th_cerai          | Int         | 5            |

15. Tabel identitas

Tabel identitas merupakan tabel yang digunakan untuk menampung identitas dari tahanan dan keluarga tahanan. Adapun struktur tabel identitas dapat dilihat pada Tabel 3.21.

Tabel 3.21 Tabel identitas.

| <b>No</b> | <b>Field Name</b> | <b>Type</b> | <b>Width</b> |
|-----------|-------------------|-------------|--------------|
| 1         | id_identitas      | Int         | 5            |

|    |            |         |    |
|----|------------|---------|----|
| 2  | no_reg     | Int     | 5  |
| 3  | Nm_lengkap | Int     | 5  |
| 4  | Alamat     | Text    |    |
| 5  | Ttl        | Varchar | 30 |
| 6  | Jk         | Varchar | 3  |
| 7  | Wn         | Varchar | 10 |
| 8  | Suku       | Varchar | 30 |
| 9  | Agama      | Varchar | 30 |
| 10 | Pendidikan | Varchar | 30 |
| 11 | Pekerjaan  | Varchar | 30 |

#### 16. Tabel interaksi sosial

Tabel interaksi sosial merupakan tabel yang digunakan untuk menampung data interaksi sosial yang dilakukan oleh tahanan. Adapun struktur tabel interaksi sosial dapat dilihat pada Tabel 3.22.

Tabel 3.22 Tabel interaksi sosial.

| No | Field Name         | Type    | Width |
|----|--------------------|---------|-------|
| 1  | id_interaksi       | Int     | 5     |
| 2  | Id_identitas       | Int     | 5     |
| 3  | No_reg             | Int     | 5     |
| 4  | Masalah            | Text    |       |
| 5  | Alasan             | Text    |       |
| 6  | Harapan            | Varchar | 30    |
| 7  | Hubungan           | Varchar | 30    |
| 8  | Kategori_kedekatan | Varchar | 30    |
| 9  | Interaksi          | Varchar | 30    |

#### 3.1.3.2.2 Implementation

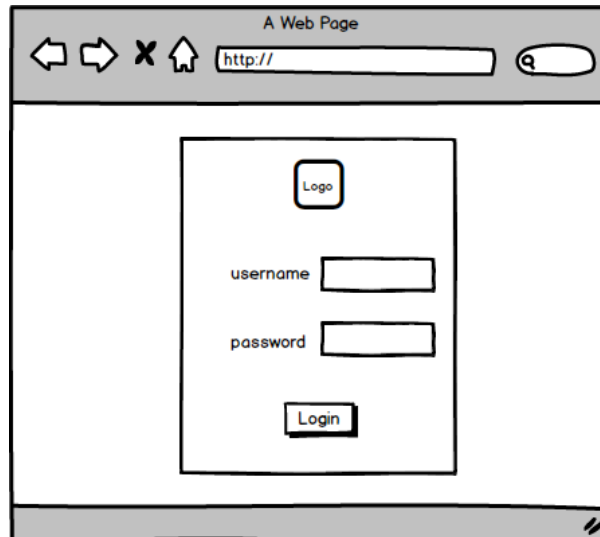
Pada tahap ini yang dilakukan adalah pembuatan *design prototype* berupa desain *interface* yang disesuaikan dengan *use case*.

##### 1. Admin

Desain *interface* untuk admin yaitu sebagai berikut:

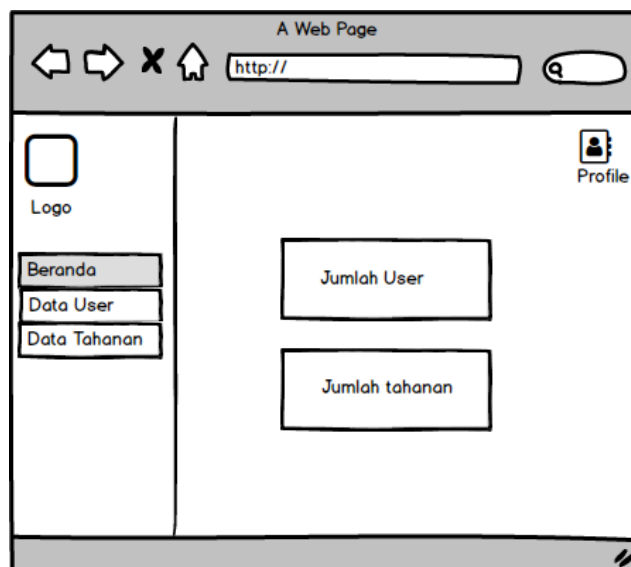
a. *Login*

Pada halaman awal sistem terdapat menu untuk *login* ke sistem jika sudah memiliki akun seperti yang terlihat pada Gambar 3.23.



Gambar 3.23 Rancangan tampilan halaman *login*.

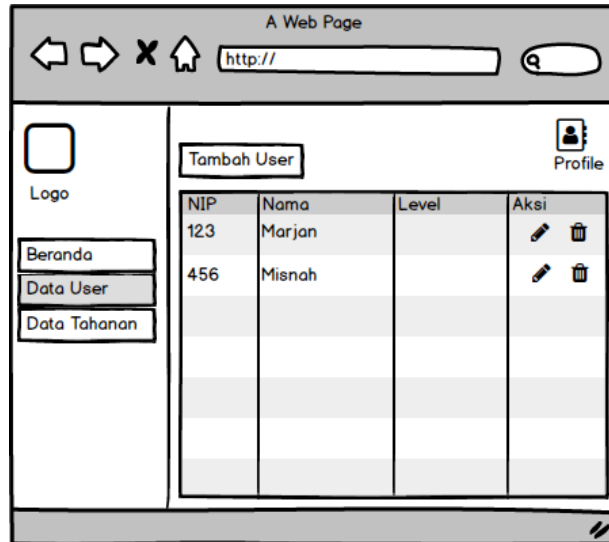
Jika *login* sebagai admin, maka akan dialihkan ke beranda untuk admin dimana terdapat informasi yang ditampilkan yaitu jumlah *user* dan jumlah tahanan seperti pada Gambar 3.24.



Gambar 3.24 Rancangan tampilan halaman beranda admin.

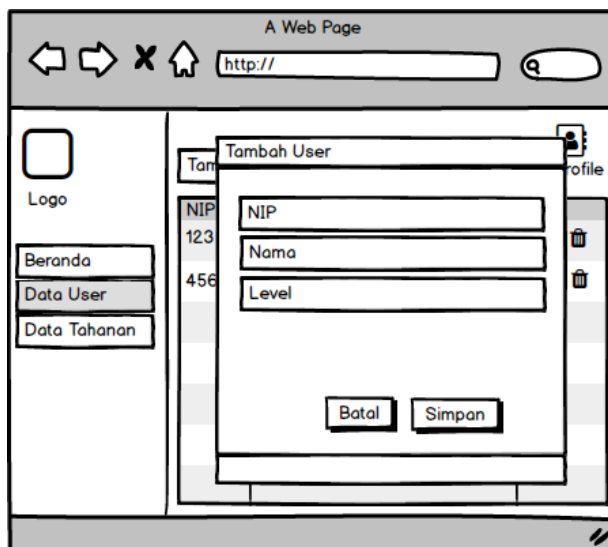
b. Data User

Pada menu data *user*, admin dapat mengelola (menambah, mengubah, dan menghapus) data *user* seperti pada Gambar 3.25.



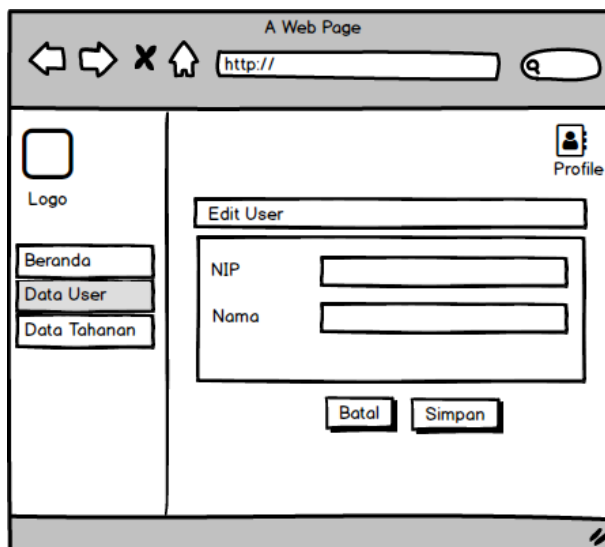
Gambar 3.25 Rancangan tampilan halaman data *user*.

Jika mengklik tombol tambah *user*, maka akan dialihkan ke halaman untuk menambah data *user* seperti pada Gambar 3.26.



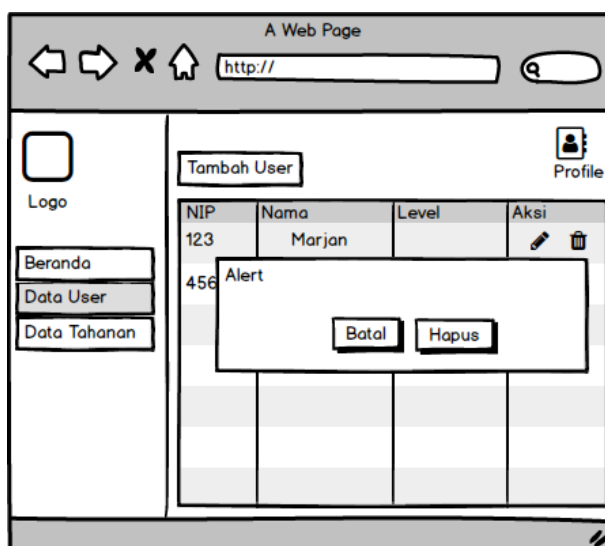
Gambar 3.26 Rancangan tampilan halaman tambah *user*.

Jika mengklik tombol edit, maka akan dialihkan ke halaman untuk mengelola mengubah data *user* seperti pada Gambar 3.27.



Gambar 3.27 Rancangan tampilan halaman edit *user*.

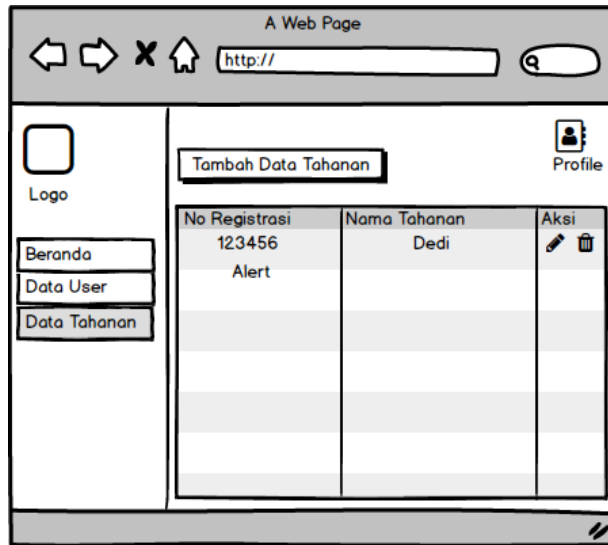
Jika mengklik tombol hapus, maka akan ditampilkan sebuah pesan konfirmasi untuk menghapus data *user* seperti pada Gambar 3.28.



Gambar 3.28 Rancangan tampilan halaman hapus *user*.

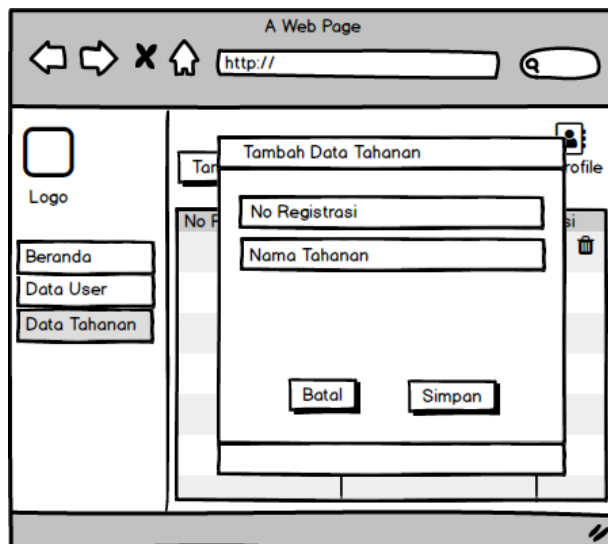
#### c. Data Tahanan

Pada menu data tahanan, admin dapat mengelola (menambah, mengubah, dan menghapus) data tahanan seperti pada Gambar 3.29.



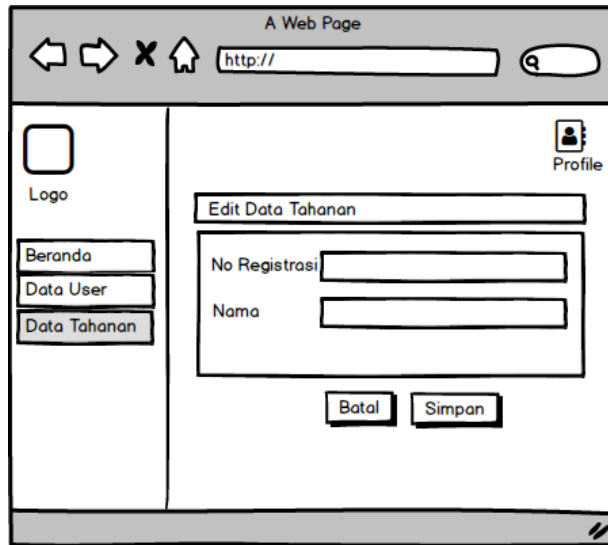
Gambar 3.29 Rancangan tampilan halaman data tahanan.

Jika mengklik tombol tambah data tahanan, maka akan dialihkan ke halaman untuk menambah data tahanan seperti pada Gambar 3.30.



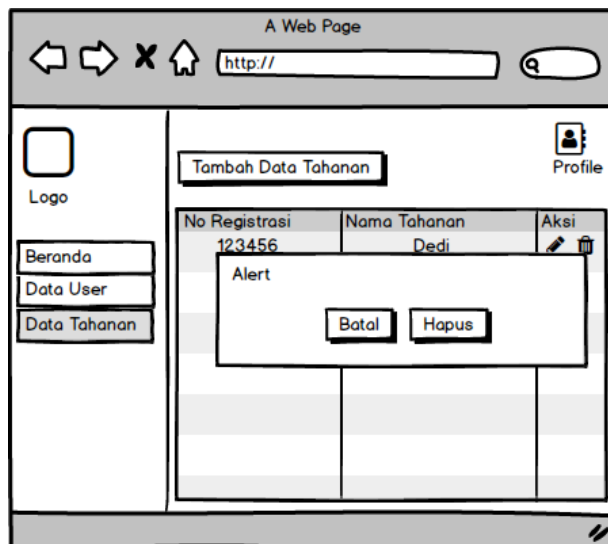
Gambar 3.30 Rancangan tampilan halaman tambah data tahanan.

Jika mengklik tombol edit, maka akan dialihkan ke halaman untuk mengubah data tahanan seperti pada Gambar 3.31.



Gambar 3.31 Rancangan tampilan halaman edit data tahanan.

Jika mengklik tombol hapus, maka akan ditampilkan pesan konfirmasi untuk menghapus data tahanan seperti pada Gambar 3.32.

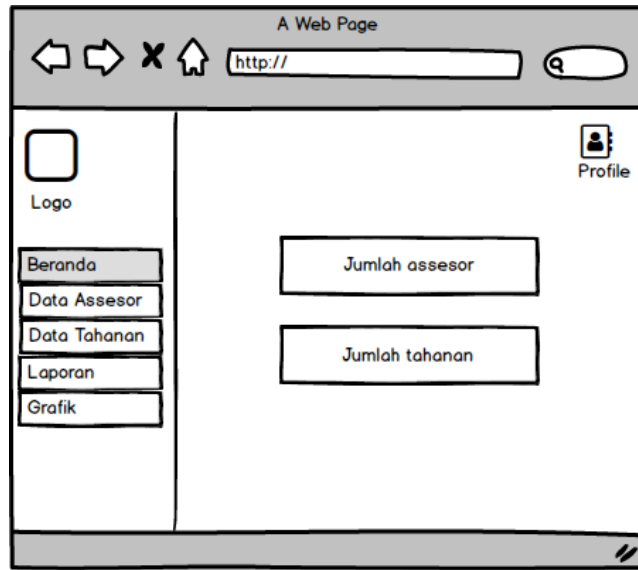


Gambar 3.32 Rancangan tampilan hapus data tahanan.

## 2. Kalapas

### a. Beranda

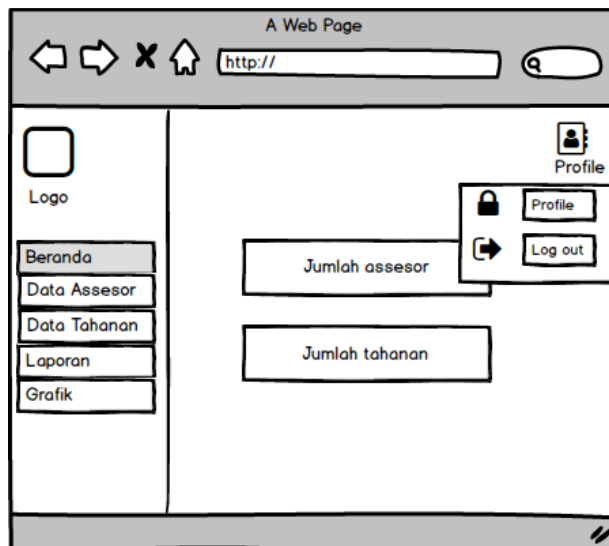
Jika *login* sebagai Kalapas, maka akan dialihkan ke beranda untuk Kalapas dimana terdapat informasi yang ditampilkan yaitu jumlah assessor seperti pada Gambar 3.33.



Gambar 3.33 Rancangan tampilan halaman beranda Kalapas.

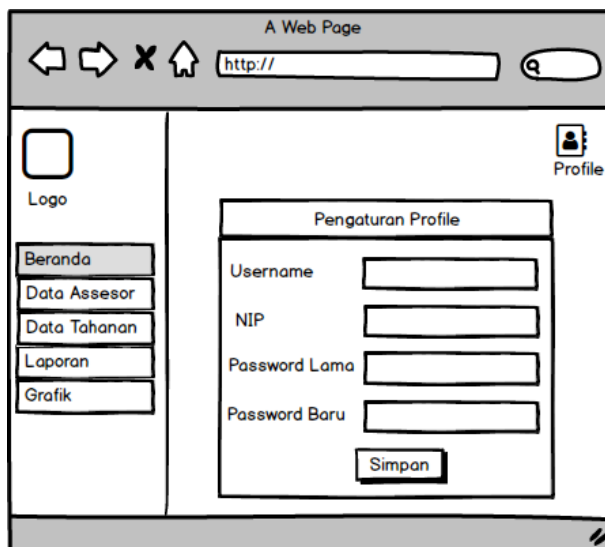
b. *Profile*

Pada halaman *profile* terdapat dua menu yaitu pengaturan *profile* dan *logout* seperti pada Gambar 3.34.



Gambar 3.34 Rancangan tampilan menekan tombol *profile*.

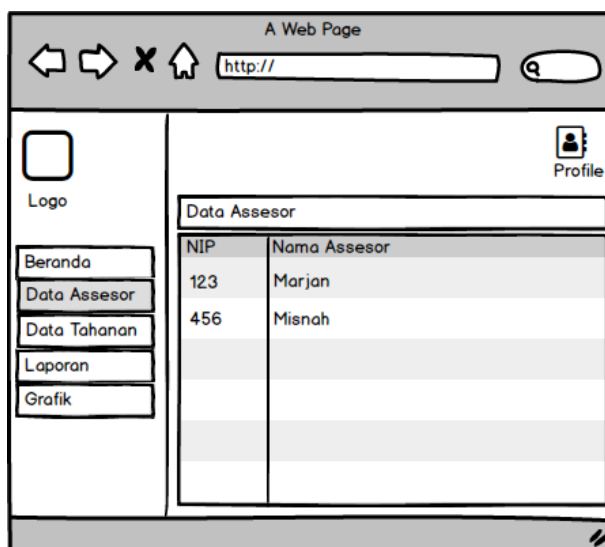
Jika menekan tombol pengaturan *profile*, maka akan dialihkan ke halaman untuk mengubah *profile* seperti pada Gambar 3.35.



Gambar 3.35 Rancangan tampilan halaman mengubah *profile*.

c. Data Assesor

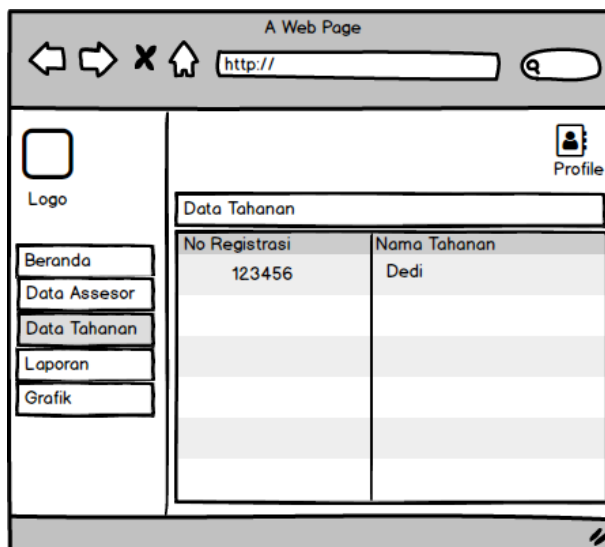
Jika memilih menu data assesor, maka akan dialihkan ke halaman untuk melihat data assesor seperti pada Gambar 3.36.



Gambar 3.36 Rancangan tampilan halaman kelola assesor.

d. Data Tahanan

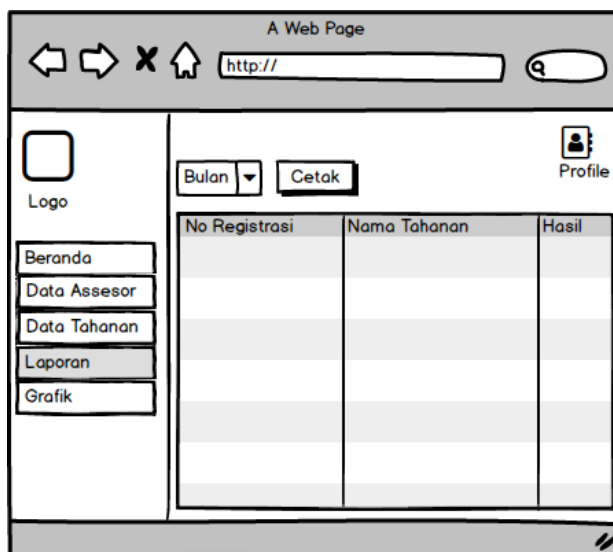
Jika memilih menu data tahanan, maka akan dialihkan ke halaman untuk melihat data tahanan seperti pada Gambar 3.37.



Gambar 3.37 Rancangan tampilan halaman melihat data tahanan.

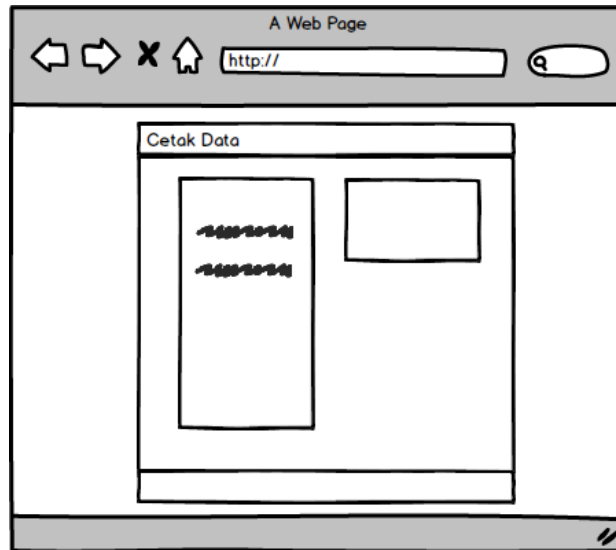
e. Laporan

Jika memilih menu laporan, maka akan dialihkan ke halaman untuk melihat laporan bulanan dari hasil *assessment* risiko tahanan seperti pada Gambar 3.38.



Gambar 3.38 Rancangan tampilan halaman laporan kasi.

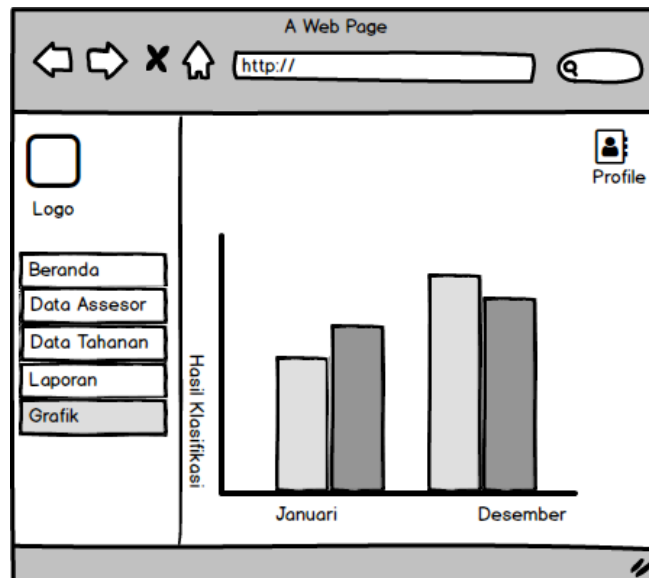
Jika mengklik tombol cetak, maka akan dialihkan ke halaman untuk mencetak laporan bulanan dari hasil *assessment* risiko tahanan seperti pada Gambar 3.39.



Gambar 3.39 Rancangan tampilan halaman cetak data.

f. Grafik

Jika memilih menu grafik, maka akan dialihkan ke halaman untuk melihat grafik bulanan yang tersusun dalam bentuk diagram batang seperti pada Gambar 3.40.

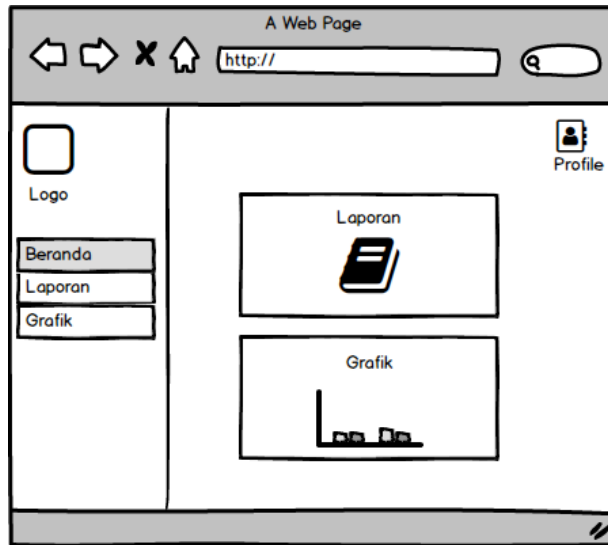


Gambar 3.40 Rancangan tampilan halaman grafik Kalapas

3. Kasi

a. Beranda

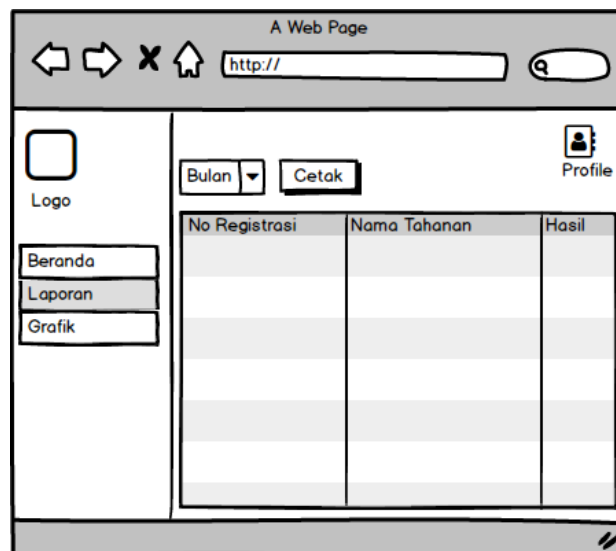
Jika *login* sebagai kasi, maka akan dialihkan ke beranda untuk kasi dimana terdapat dua menu yang dapat dilihat oleh kasi yaitu laporan dan grafik seperti pada Gambar 3.41.



Gambar 3.41 Rancangan tampilan halaman beranda kasi.

b. Laporan

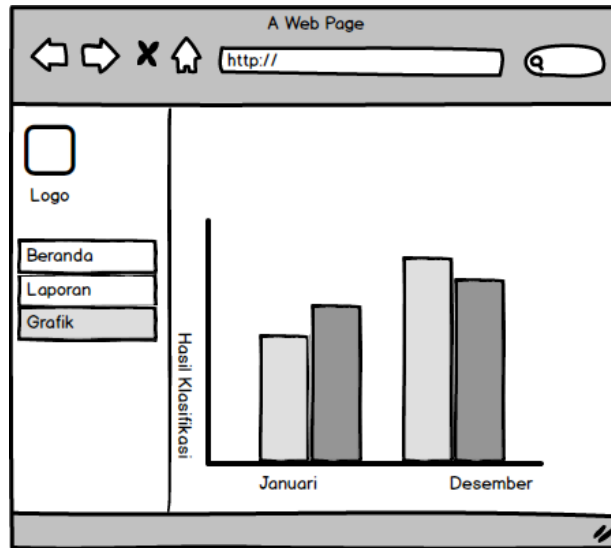
Jika memilih menu laporan, maka akan dialihkan ke halaman untuk melihat laporan bulanan dari hasil *assessment* risiko tahanan seperti pada Gambar 3.42.



Gambar 3.42 Rancangan tampilan halaman laporan kasi.

c. Grafik

Jika memilih menu grafik, maka akan dialihkan ke halaman untuk melihat grafik bulanan yang tersusun dalam bentuk diagram batang seperti pada Gambar 3.43.

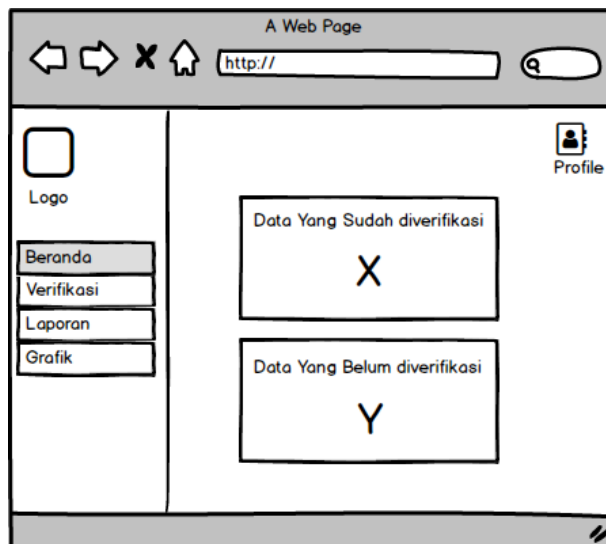


Gambar 3.43 Rancangan tampilan halaman grafik kasi.

#### 4. Kasubsi

##### a. Beranda

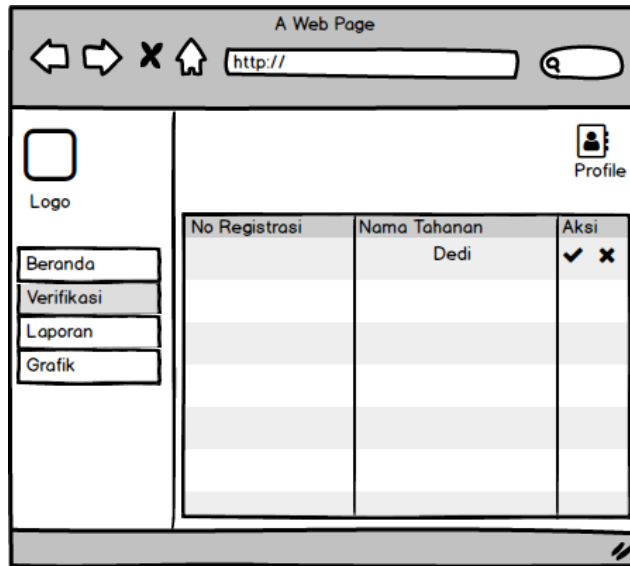
Jika *login* sebagai kasubsi, maka akan dialihkan ke beranda untuk kasubsi dimana terdapat dua informasi yang ditampilkan yaitu jumlah data tahanan yang sudah diverifikasi dan jumlah data tahanan yang belum diverifikasi seperti pada Gambar 3.44.



Gambar 3.44 Rancangan tampilan halaman beranda kasubsi.

##### b. Verifikasi

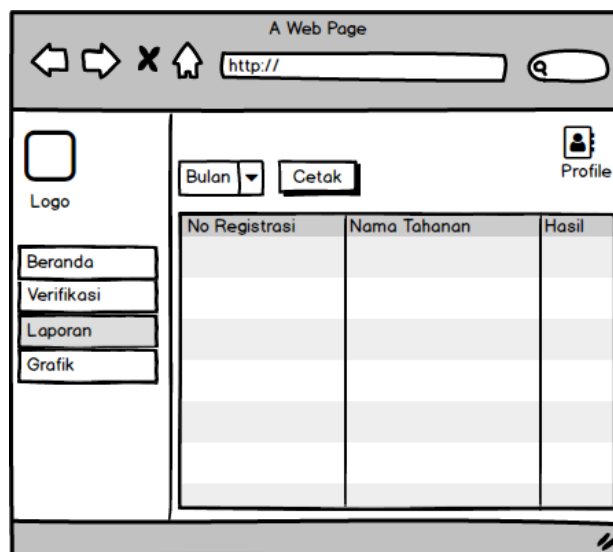
Jika memilih menu verifikasi, maka akan dialihkan ke halaman untuk melakukan verifikasi terhadap data tahanan seperti pada Gambar 3.45.



Gambar 3.45 Rancangan tampilan halaman verifikasi.

c. Laporan

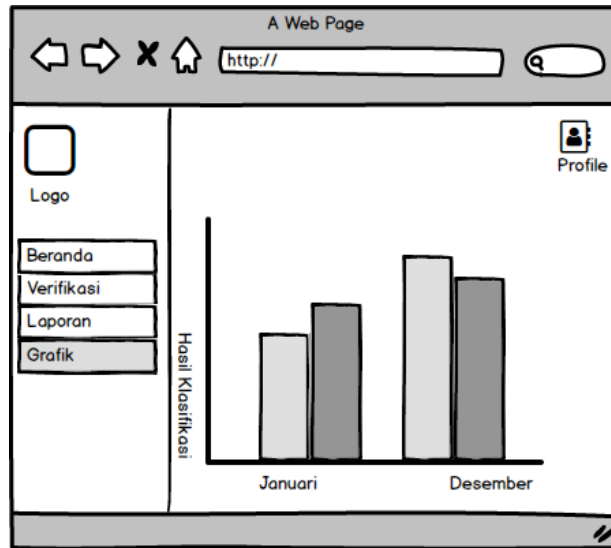
Jika memilih menu laporan, maka akan dialihkan ke halaman untuk melihat laporan bulanan dari hasil *assessment* risiko tahanan seperti pada Gambar 3.46.



Gambar 3.46 Rancangan tampilan halaman laporan kasubsi.

d. Grafik

Jika memilih menu grafik, maka akan dialihkan ke halaman untuk melihat grafik bulanan yang tersusun dalam bentuk diagram batang seperti pada Gambar 3.47.

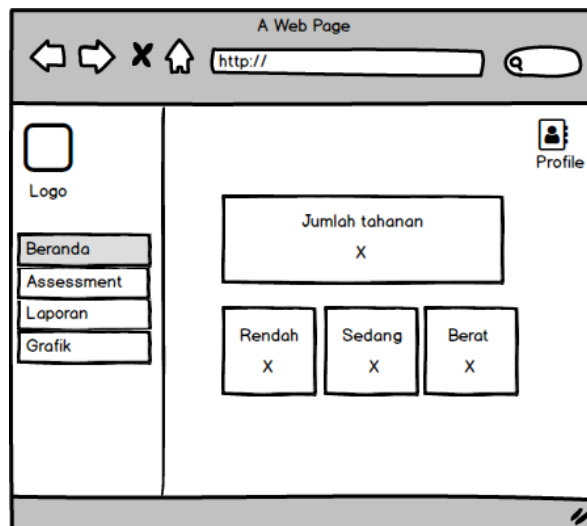


Gambar 3.47 Rancangan tampilan halaman grafik kasubsi.

## 5. Assessor

### a. Beranda

Jika *login* sebagai assessor, maka akan dialihkan ke beranda untuk assessor dimana terdapat dua informasi yang ditampilkan yaitu jumlah tahanan dan jumlah tahanan berdasarkan hasil klasifikasi *assessment* risiko tahanan (rendah, sedang, atau berat) seperti pada Gambar 3.48.



Gambar 3.48 Rancangan tampilan halaman beranda assessor.

### b. Menu *Assessment*

Jika memilih menu *assessment*, maka akan dialihkan ke halaman untuk melakukan *assesment* dimana terdapat *form* yang harus diisi sesuai dengan

ketentuan yang dibuat seperti pada Gambar 3.49, Gambar 3.50, Gambar 3.51, dan Gambar 3.52.

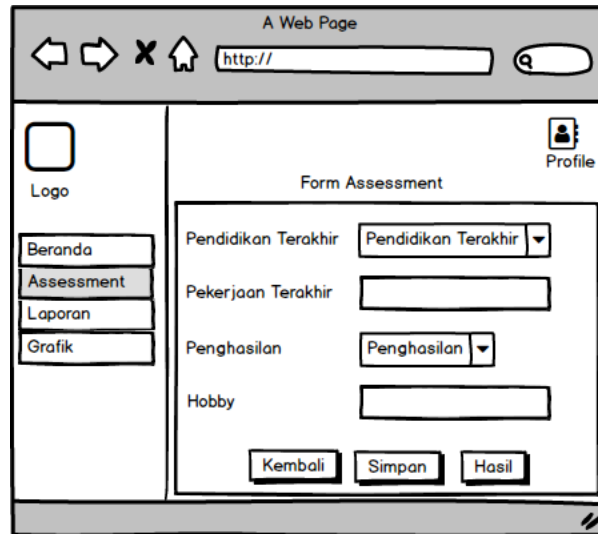
The wireframe shows a web browser window titled "A Web Page". The address bar contains "http://". On the left is a navigation menu with a "Logo" placeholder and four links: "Beranda", "Assessment", "Laporan", and "Grafik". The "Assessment" link is highlighted. On the right, there is a "Profile" icon. The main content area displays a table with three columns: "No Registrasi", "Nama Tahanan", and "Ket". The first row contains the values "123456", "Dedi", and "Ass". Below this row are four empty rows.

| No Registrasi | Nama Tahanan | Ket |
|---------------|--------------|-----|
| 123456        | Dedi         | Ass |
|               |              |     |
|               |              |     |
|               |              |     |
|               |              |     |

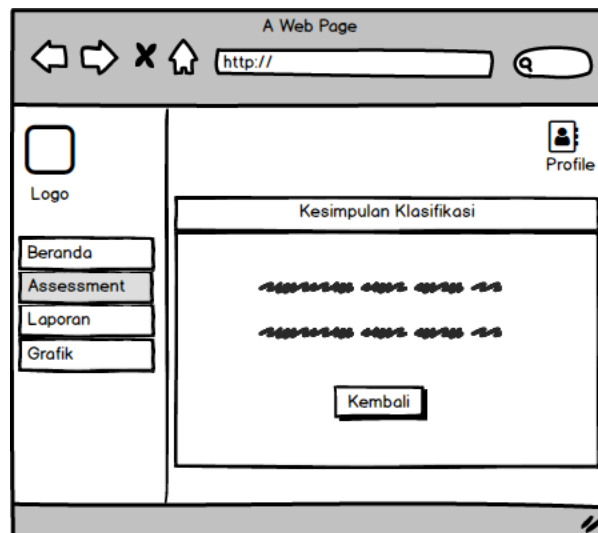
Gambar 3.49 Rancangan tampilan halaman *assessment*.

The wireframe shows a web browser window titled "A Web Page". The address bar contains "http://". On the left is a navigation menu with a "Logo" placeholder and four links: "Beranda", "Assessment", "Laporan", and "Grafik". The "Assessment" link is highlighted. On the right, there is a "Profile" icon. The main content area is titled "Form Assessment" and contains the following fields: "Tanggal Assessment" with a date input field showing "MM/DD/YYYY" and a calendar icon; "Waktu Assessment" with a text input field; "Nama Asesor" with a text input field; "Nama Tahanan" with a dropdown menu showing "Nama Tahanan"; and a "Selanjutnya" button at the bottom right.

Gambar 3.50 Rancangan tampilan halaman *assessment*.



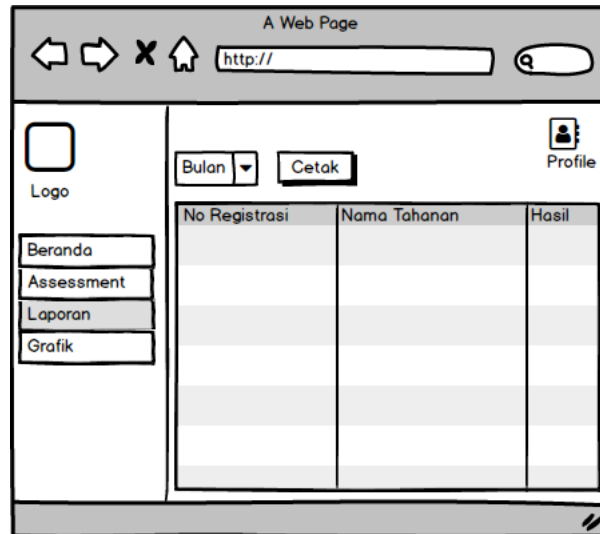
Gambar 3.51 Rancangan tampilan halaman *assessment*.



Gambar 3.52 Rancangan tampilan halaman *assessment*.

#### c. Laporan

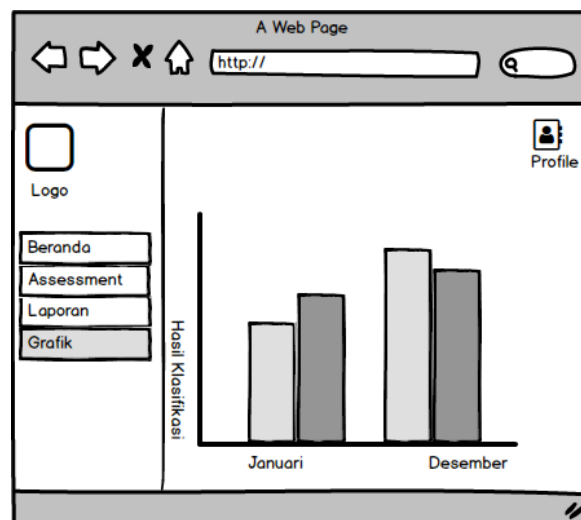
Jika memilih menu laporan, maka akan dialihkan ke halaman untuk melihat laporan bulanan dari hasil *assessment* risiko tahanan seperti pada Gambar 3.53.



Gambar 3.53 Rancangan tampilan halaman laporan assessor.

d. Grafik

Jika memilih menu grafik, maka akan dialihkan ke halaman untuk melihat grafik bulanan yang tersusun dalam bentuk diagram batang seperti pada Gambar 3.54.



Gambar 3.54 Rancangan tampilan halaman grafik assessor.

### 3.1.3.2.3 Test Elaboration

Pada tahap *elaboration*, pengujian dibutuhkan untuk menguji fitur yang telah dirancang pada *use case diagram* dengan implementasi desain *interface* sistem apakah telah sesuai dengan keinginan pengguna atau tidak. Pengujian dilakukan oleh 3 orang penguji yaitu Kalapas, Kasi dan assessor. Proses validasi fitur dimulai dengan peneliti menampilkan rancangan *interface* sistem seperti pada Gambar 3.23 sampai 3.54, setelah itu penguji memberikan pernyataan setuju atau

tidak setuju pada tabel isian yang telah disediakan. Hasil validasi dapat dilihat pada Tabel 3.23 berikut.

Tabel 3.23 Validasi fitur sistem.

| No | Pengguna | Fitur                                                                                                             | Validasi |              |
|----|----------|-------------------------------------------------------------------------------------------------------------------|----------|--------------|
|    |          |                                                                                                                   | Setuju   | Tidak Setuju |
| 1  | admin    | Menambah <i>user</i>                                                                                              | ✓        |              |
|    |          | Mengubah <i>user</i>                                                                                              | ✓        |              |
|    |          | Menghapus <i>user</i>                                                                                             | ✓        |              |
|    |          | Menambah data tahanan                                                                                             | ✓        |              |
|    |          | Mengubah data tahanan                                                                                             | ✓        |              |
|    |          | Menghapus data tahanan                                                                                            | ✓        |              |
| 2  | Kalapas  | Melihat data assessor                                                                                             | ✓        |              |
|    |          | Melihat data tahanan                                                                                              | ✓        |              |
|    |          | Mengubah <i>profile</i>                                                                                           | ✓        |              |
|    |          | Melihat grafik bulanan                                                                                            | ✓        |              |
|    |          | Melihat laporan bulanan berupa nomer registrasi tahanan, nama tahanan, dan hasil <i>assessment</i> risiko tahanan | ✓        |              |
|    |          | Mencetak laporan bulanan                                                                                          | ✓        |              |
| 3  | Kasi     | Mengubah <i>profile</i>                                                                                           | ✓        |              |
|    |          | Melihat grafik bulanan                                                                                            | ✓        |              |
|    |          | Melihat laporan bulanan berupa nomer registrasi tahanan, nama tahanan, dan hasil <i>assessment</i> risiko tahanan | ✓        |              |
|    |          | Mencetak laporan bulanan                                                                                          | ✓        |              |
| 4  | Kasubsi  | Mengubah <i>profile</i>                                                                                           | ✓        |              |
|    |          | Melihat grafik bulanan                                                                                            | ✓        |              |
|    |          | Mencetak laporan bulanan                                                                                          | ✓        |              |
|    |          | Melihat laporan bulanan berupa nomer registrasi tahanan, nama tahanan, dan hasil <i>assessment</i> risiko tahanan | ✓        |              |
|    |          | Melakukan verifikasi hasil <i>assessment</i>                                                                      | ✓        |              |
| 5  | Assessor | Mengubah <i>profile</i>                                                                                           | ✓        |              |

|  |  |                                                                                                                   |   |  |
|--|--|-------------------------------------------------------------------------------------------------------------------|---|--|
|  |  | Melakukan <i>assessment</i>                                                                                       | ✓ |  |
|  |  | Mengubah data <i>assessment</i>                                                                                   | ✓ |  |
|  |  | Melihat laporan bulanan berupa nomer registrasi tahanan, nama tahanan, dan hasil <i>assessment</i> risiko tahanan | ✓ |  |
|  |  | Mencetak laporan bulanan                                                                                          | ✓ |  |
|  |  | Melihat grafik bulanan                                                                                            | ✓ |  |

### 3.1.3.3 Tahap *construction*

Tahap ini adalah tahap implementasi sistem dimana pada tahap ini sistem sudah mulai dibangun. Tahapannya yaitu sebagai berikut.

- Membangun antar muka atau *interface* sistem menggunakan *framework codeigniter*.
- Pembuatan *database* sistem menggunakan MySQL.
- Pengkodean sistem dengan menggunakan konsep *Model View Controller* (MVC).
- Pengujian sistem oleh masing-masing aktor untuk mengetahui apakah tampilan dan fungsional sistem telah sesuai dengan kebutuhan.

### 3.1.3.4 Tahap *transition*

Tahap *transition* adalah tahap terakhir dalam metode RUP, dimana pada tahap ini dilakukan beberapa aktivitas yaitu sebagai berikut.

- Melakukan instalasi sistem yang telah selesai dibangun untuk kemudian dilakukan pengujian terakhir oleh masing-masing aktor sebelum sistem benar-benar digunakan.
- Pengembang mendampingi aktor guna untuk memberikan *user guide* secara langsung kepada setiap aktor sehingga masing-masing aktor lebih cepat dapat memahami proses dalam sistem.
- Untuk mendokumentasikan hasil proses pengujian ini dilakukan dengan menggunakan *black box testing* yaitu berupa kuesioner tentang bagaimana tampilan dan fungsional sistem yang diisi oleh masing-masing aktor setelah menggunakan sistem.

- d. Sistem yang telah selesai dilakukan pengujian dan telah sesuai dengan kebutuhan akan diserahkan kepada instansi untuk dapat digunakan sebagaimana mestinya.

## DAFTAR PUSTAKA

- [1] Gloria, *Standar Asesmen Dan Klasifikasi Tahanan*. Direktorat Jenderal Pemasyarakatan, 2018.
- [2] S. Ahmad Khoiruddin, “Pengembangan Sistem Tracer Study Jurusan Ilmu Komputer Menggunakan Metode Rational Unified Process (RUP),” *Univ. Lampung*, 2015.
- [3] A. A. D. I. Nugroho, “Analisis dan Pengembangan Sistem Ujian Akhir Semester Berbasis Komputer di SMK Negeri 1 Magelang,” *Univ. Negeri Yogyakarta*, 2018.
- [4] A. T. Ulfa, A. Hijriani, and R. Andrian, “Rancang Bangun Sistem Informasi Administrasi PKM Universitas Lampung Berbasis Web Menggunakan Metode Rational Unified Process (RUP),” *J. Komputasi*, vol. 6, no. 1, pp. 8–16, 2018.
- [5] F. Mubarak and I. Hadijah, “Perbandingan Antara Metode RUP dan Prototype Dalam Aplikasi Penerimaan Siswa Baru Berbasis Web,” *Citec J.*, vol. Vol. 2, No, no. ISSN: 2354-5771, pp. 114–127, 2015.
- [6] M. Zaminkar and M. R. Reshadinezhad, “A Comparison Between Two Software Engineering Processes, RUP And Waterfall Models,” *Int. J. Eng. Res. Technol.*, vol. 2, no. 7, pp. 1348–1352, 2013.
- [7] A. Alshamrani and A. Bahattab, “A Comparison Between Three SDLC Models Waterfall Model , Spiral Model , and Incremental / Iterative Model,” *Int. J. Comput. Sci. Issues*, vol. 12, no. 1, pp. 106–111, 2015.
- [8] Sukandi, “Analisa Sistem Informasi Pemberian Remisi Bagi Narapidana Pada Lembaga Pemasyarakatan Kelas IIB Sungailiat,” *STMIK Atma Luhur*, 2015.
- [9] Y. Irawan and U. Rahmalisa, “Sistem Database Pemasyarakatan Studi Kasus Lapas Kelas II A Pekanbaru,” *J. Technopreneursh. Inf. Syst.*, vol. 2, no. 19, pp. 59–67, 2019.
- [10] R. Lalu Hary Aditia, A. Royana, and Z. Ariyan, “Rancang Bangun Sistem

Informasi Monitoring Dan Hasil Belajar Siswa Berbasis Mobile Di SMPN 15 Mataram,” *Univ. Mataram*, 2018.

- [11] Triwahyuni and Saputra, “Architecture E-Mall Using RUP (Rational Unified Process) Methods,” *Cogito Smart J.*, vol. 1, no. IISN : 2477-8079, 2015.
- [12] S. P. Roger, *Rekayasa Perangkat Lunak (Pendekatan Praktisi) Edisi 7 : Buku 1*. Yogyakarta: Andi, 2012.
- [13] D. Puspita, *Perancangan Sistem Informasi Pengelolaan Data Tamu dan Pengunjung di Museum Geologi*. Bandung: Universitas Komputer Indonesia Bandung, 2012.
- [14] A. Alan Nur, *Jago PHP dan MySQL*. Bekasi: DUNIA KOMPUTER, 2011.
- [15] S. Young, “Penerapan Metode SDLC RUP (Rational Unified Process) Dalam Pembuatan Sistem Informasi E-Kos Berbasis Web Pada Kota Malang,” *Univ. Muhammadiyah Malang*, 2018.
- [16] P. Ririn, A. Sri Endang, and A. Royana, “Penerapan Metode Rational Unified Process ( RUP ) Dalam Pengembangan Sistem Informasi Medical Check Up Pada Citra Medical Centre,” *Univ. Mataram*, 2020.

## LAMPIRAN

Lampiran 1. Pengujian validasi kebutuhan fungsional sistem.

### LEMBAR PENGUJIAN VALIDASI KEBUTUHAN FUNGSIONAL SISTEM DENGAN USE CASE DALAM PENGEMBANGAN SISTEM INFORMASI ASSESSMENT RISIKO TAHANAN PADA LEMBAGA PEMASYARAKATAN KELAS II B SELONG

#### A. Identitas Responden

Nama : HERDIANTO, A.M.I. IPSH MS  
Pekerjaan : PNS (Kalapas)  
Instansi : Lapas Kelas II B Selong

#### B. Petunjuk Pengisian

Berilah tanda centang (✓) pada kolom Admin jika kebutuhan fungsional sistem sesuai dengan *use case* Admin, Kalapas jika kebutuhan fungsional sistem sesuai dengan *use case* Kalapas, Kasi jika kebutuhan fungsional sistem sesuai dengan *use case* kasi, Kasubsi jika kebutuhan fungsional sistem sesuai dengan *use case* kasubsi, dan Assessor jika kebutuhan fungsional sistem sesuai dengan *use case* assessor.

#### C. Lembar Pengujian

| No | Kebutuhan fungsional sistem                                | Use Case |         |      |         |          |
|----|------------------------------------------------------------|----------|---------|------|---------|----------|
|    |                                                            | Admin    | Kalapas | Kasi | Kasubsi | Assessor |
| 1  | Mengelola <i>user</i> (menambah, mengubah, dan menghapus)  | ✓        |         |      |         |          |
| 2  | Mengubah <i>profile</i>                                    |          | ✓       | ✓    | ✓       | ✓        |
| 3  | Melihat grafik bulanan                                     |          | ✓       | ✓    | ✓       | ✓        |
| 4  | Melihat laporan bulanan                                    |          | ✓       | ✓    | ✓       | ✓        |
| 5  | Mencetak laporan bulanan                                   |          | ✓       | ✓    | ✓       | ✓        |
| 6  | Melakukan verifikasi hasil <i>assessment</i>               |          |         |      | ✓       |          |
| 7  | Mengelola data tahanan (menambah, mengubah, dan menghapus) | ✓        |         |      |         |          |
| 8  | Melakukan <i>assessment</i>                                |          |         |      |         | ✓        |
| 9  | Mengubah data <i>assessment</i>                            |          |         |      |         | ✓        |

Berdasarkan hasil dari proses validasi yang dilakukan, didapatkan kesimpulan bahwa *use case* yang dirancang telah sesuai / tidak sesuai dengan kebutuhan pengguna.

• Jika tidak sesuai dengan kebutuhan pengguna, maka *usecase* yang harus ditambah :

.....  
.....  
.....



Selang, 20 Februari 2020

Responden

*[Handwritten signature]*

HEADINTO, ANJ. IP.SH MS.

**LEMBAR PENGUJIAN VALIDASI KEBUTUHAN FUNGSIONAL SISTEM  
DENGAN USE CASE DALAM PENGEMBANGAN SISTEM INFORMASI  
ASSESSMENT RISIKO TAHANAN PADA LEMBAGA PEMASYARAKATAN  
KELAS II B SELONG**

**A. Identitas Responden**

Nama : M. Fauzan Asim  
Pekerjaan : 1 PNS (Assessor)  
Instansi : Lapas Kelas II B Selong

**B. Petunjuk Pengisian**

Berilah tanda centang (✓) pada kolom Admin jika kebutuhan fungsional sistem sesuai dengan *use case* Admin, Kalapas jika kebutuhan fungsional sistem sesuai dengan *use case* Kalapas, Kasi jika kebutuhan fungsional sistem sesuai dengan *use case* kasi, Kasubsi jika kebutuhan fungsional sistem sesuai dengan *use case* kasubsi, dan Assessor jika kebutuhan fungsional sistem sesuai dengan *use case* assessor.

**C. Lembar Pengujian**

| No | Kebutuhan fungsional sistem                                | Use Case |         |      |         |          |
|----|------------------------------------------------------------|----------|---------|------|---------|----------|
|    |                                                            | Admin    | Kalapas | Kasi | Kasubsi | Assessor |
| 1  | Mengelola <i>user</i> (menambah, mengubah, dan menghapus)  | ✓        |         |      |         |          |
| 2  | Mengubah <i>profile</i>                                    |          | ✓       | ✓    | ✓       | ✓        |
| 3  | Melihat grafik bulanan                                     |          | ✓       | ✓    | ✓       | ✓        |
| 4  | Melihat laporan bulanan                                    |          | ✓       | ✓    | ✓       | ✓        |
| 5  | Mencetak laporan bulanan                                   |          | ✓       | ✓    | ✓       | ✓        |
| 6  | Melakukan verifikasi hasil <i>assessment</i>               |          |         |      | ✓       |          |
| 7  | Mengelola data tahanan (menambah, mengubah, dan menghapus) | ✓        |         |      |         |          |
| 8  | Melakukan <i>assessment</i>                                |          |         |      |         | ✓        |
| 9  | Mengubah data <i>assessment</i>                            |          |         |      |         | ✓        |

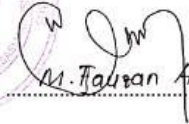
Berdasarkan hasil dari proses validasi yang dilakukan, didapatkan kesimpulan bahwa *use case* yang dirancang telah sesuai / ~~tidak sesuai~~ dengan kebutuhan pengguna.

\*Jika tidak sesuai dengan kebutuhan pengguna, maka *usecase* yang harus ditambah :

.....  
.....  
.....



Selong, ..... 20 februari ..... 2020  
Responden

  
M. Fauzan Azim

**LEMBAR PENGUJIAN VALIDASI KEBUTUHAN FUNGSIONAL SISTEM  
DENGAN USE CASE DALAM PENGEMBANGAN SISTEM INFORMASI  
ASSESMENT RISIKO TAHANAN PADA LEMBAGA PEMASYARAKATAN  
KELAS II B SELONG**

**A. Identitas Responden**

Nama : Nasrullah  
Pekerjaan : PHS (Kasi)  
Instansi : Lapas Kelas II B Selong

**B. Petunjuk Pengisian**

Berilah tanda centang (✓) pada kolom Admin jika kebutuhan fungsional sistem sesuai dengan *use case* Admin, Kalapas jika kebutuhan fungsional sistem sesuai dengan *use case* Kalapas, Kasi jika kebutuhan fungsional sistem sesuai dengan *use case* kasi, Kasubsi jika kebutuhan fungsional sistem sesuai dengan *use case* kasubsi, dan Assessor jika kebutuhan fungsional sistem sesuai dengan *use case* assessor.

**C. Lembar Pengujian**

| No | Kebutuhan fungsional sistem                                | Use Case |         |      |         |          |
|----|------------------------------------------------------------|----------|---------|------|---------|----------|
|    |                                                            | Admin    | Kalapas | Kasi | Kasubsi | Assessor |
| 1  | Mengelola <i>user</i> (menambah, mengubah, dan menghapus)  | ✓        |         |      |         |          |
| 2  | Mengubah <i>profile</i>                                    |          | ✓       | ✓    | ✓       | ✓        |
| 3  | Melihat grafik bulanan                                     |          | ✓       | ✓    | ✓       | ✓        |
| 4  | Melihat laporan bulanan                                    |          | ✓       | ✓    | ✓       | ✓        |
| 5  | Mencetak laporan bulanan                                   |          | ✓       | ✓    | ✓       | ✓        |
| 6  | Melakukan verifikasi hasil <i>assessment</i>               |          |         |      | ✓       |          |
| 7  | Mengelola data tahanan (menambah, mengubah, dan menghapus) | ✓        |         |      |         |          |
| 8  | Melakukan <i>assessment</i>                                |          |         |      |         | ✓        |
| 9  | Mengubah data <i>assessment</i>                            |          |         |      |         | ✓        |

Berdasarkan hasil dari proses validasi yang dilakukan, didapatkan kesimpulan bahwa *use case* yang dirancang telah sesuai / tidak sesuai dengan kebutuhan pengguna.

\*Jika tidak sesuai dengan kebutuhan pengguna, maka *usecase* yang harus ditambah :

.....  
.....  
.....



Selong, 20 februari 2020  
Responden

*Nasrudin*

Lampiran 2. Pengujian validasi fitur.

**LEMBAR PENGUJIAN VALIDASI FITUR SISTEM DENGAN  
IMPLEMENTASI DESAIN DALAM PENGEMBANGAN SISTEM INFORMASI  
ASSESSMENT RISIKO TAHANAN PADA LEMBAGA PEMASYARAKATAN  
KELAS II B SELONG**

**A. Identitas Responden**

Nama : HERDIANTO, A.Md IP.SH MSi  
Pekerjaan : PNS ( Kalapas )  
Instansi : Lapas Kelas II B Selong

**B. Petunjuk Pengisian**

Berilah tanda centang (✓) pada kolom Setuju jika fitur sistem sesuai dengan kebutuhan dan Tidak Setuju jika fitur sistem tidak sesuai dengan kebutuhan.

**C. Lembar Pengujian**

| No | Pengguna | Fitur                                                                                                             | Validasi |              |
|----|----------|-------------------------------------------------------------------------------------------------------------------|----------|--------------|
|    |          |                                                                                                                   | Setuju   | Tidak Setuju |
| 1  | admin    | Menambah user                                                                                                     | ✓        |              |
|    |          | Mengubah user                                                                                                     | ✓        |              |
|    |          | Menghapus user                                                                                                    | ✓        |              |
|    |          | Menambah data tahanan                                                                                             | ✓        |              |
|    |          | Mengubah data tahanan                                                                                             | ✓        |              |
|    |          | Menghapus data tahanan                                                                                            | ✓        |              |
| 2  | Kalapas  | Melihat data assessor                                                                                             | ✓        |              |
|    |          | Melihat data tahanan                                                                                              | ✓        |              |
|    |          | Mengubah profile                                                                                                  | ✓        |              |
|    |          | Melihat grafik bulanan                                                                                            | ✓        |              |
|    |          | Melihat laporan bulanan berupa nomer registrasi tahanan, nama tahanan, dan hasil <i>assessment</i> risiko tahanan | ✓        |              |
|    |          | Mencetak laporan bulanan                                                                                          | ✓        |              |
| 3  | Kasi     | Mengubah profile                                                                                                  | ✓        |              |
|    |          | Melihat grafik bulanan                                                                                            | ✓        |              |

|   |          |                                                                                                                   |   |  |
|---|----------|-------------------------------------------------------------------------------------------------------------------|---|--|
|   |          | Melihat laporan bulanan berupa nomer registrasi tahanan, nama tahanan, dan hasil <i>assessment</i> risiko tahanan | ✓ |  |
|   |          | Mencetak laporan bulanan                                                                                          | ✓ |  |
| 4 | Kasubsi  | Mengubah <i>profile</i>                                                                                           | ✓ |  |
|   |          | Melihat grafik bulanan                                                                                            | ✓ |  |
|   |          | Mencetak laporan bulanan                                                                                          | ✓ |  |
|   |          | Melihat laporan bulanan berupa nomer registrasi tahanan, nama tahanan, dan hasil <i>assessment</i> risiko tahanan | ✓ |  |
|   |          | Melakukan verifikasi hasil <i>assessment</i>                                                                      | ✓ |  |
| 5 | Assessor | Mengubah <i>profile</i>                                                                                           | ✓ |  |
|   |          | Melakukan <i>assessment</i>                                                                                       | ✓ |  |
|   |          | Mengubah data <i>assessment</i>                                                                                   | ✓ |  |
|   |          | Melihat laporan bulanan berupa nomer registrasi tahanan, nama tahanan, dan hasil <i>assessment</i> risiko tahanan | ✓ |  |
|   |          | Mencetak laporan bulanan                                                                                          | ✓ |  |
|   |          | Melihat grafik bulanan                                                                                            | ✓ |  |

Berdasarkan hasil dari proses validasi yang dilakukan, didapatkan kesimpulan bahwa fitur sistem yang dirancang telah sesuai / ~~tidak sesuai~~ dengan kebutuhan pengguna.

\*Jika tidak sesuai dengan kebutuhan pengguna, maka fitur yang harus ditambah :

.....  
 .....  
 .....



Selong, 20 februari ..... 2020

Responden

*[Signature]*  
 HERDIANTO, A.Md. IP. SH. MSI

**LEMBAR PENGUJIAN VALIDASI FITUR SISTEM DENGAN  
IMPLEMENTASI DESAIN DALAM PENGEMBANGAN SISTEM INFORMASI  
ASSESSMENT RISIKO TAHANAN PADA LEMBAGA PEMASYARAKATAN  
KELAS II B SELONG**

**A. Identitas Responden**

Nama : M. Fauzan Azim  
Pekerjaan : PNS  
Instansi : LAPAS kelas II B Selong

**B. Petunjuk Pengisian**

Berilah tanda centang (✓) pada kolom **Setuju** jika fitur sistem sesuai dengan kebutuhan dan **Tidak Setuju** jika fitur sistem tidak sesuai dengan kebutuhan.

**C. Lembar Pengujian**

| No | Pengguna | Fitur                                                                                                             | Validasi |              |
|----|----------|-------------------------------------------------------------------------------------------------------------------|----------|--------------|
|    |          |                                                                                                                   | Setuju   | Tidak Setuju |
| 1  | admin    | Menambah <i>user</i>                                                                                              | ✓        |              |
|    |          | Mengubah <i>user</i>                                                                                              | ✓        |              |
|    |          | Menghapus <i>user</i>                                                                                             | ✓        |              |
|    |          | Menambah data tahanan                                                                                             | ✓        |              |
|    |          | Mengubah data tahanan                                                                                             | ✓        |              |
|    |          | Menghapus data tahanan                                                                                            | ✓        |              |
| 2  | Kalapas  | Melihat data assessor                                                                                             | ✓        |              |
|    |          | Melihat data tahanan                                                                                              | ✓        |              |
|    |          | Mengubah <i>profile</i>                                                                                           | ✓        |              |
|    |          | Melihat grafik bulanan                                                                                            | ✓        |              |
|    |          | Melihat laporan bulanan berupa nomer registrasi tahanan, nama tahanan, dan hasil <i>assessment</i> risiko tahanan | ✓        |              |
|    |          | Mencetak laporan bulanan                                                                                          | ✓        |              |
| 3  | Kasi     | Mengubah <i>profile</i>                                                                                           | ✓        |              |
|    |          | Melihat grafik bulanan                                                                                            | ✓        |              |

|   |          |                                                                                                                   |   |  |
|---|----------|-------------------------------------------------------------------------------------------------------------------|---|--|
|   |          | Melihat laporan bulanan berupa nomer registrasi tahanan, nama tahanan, dan hasil <i>assessment</i> risiko tahanan | ✓ |  |
|   |          | Mencetak laporan bulanan                                                                                          | ✓ |  |
| 4 | Kasubsi  | Mengubah <i>profile</i>                                                                                           | ✓ |  |
|   |          | Melihat grafik bulanan                                                                                            | ✓ |  |
|   |          | Mencetak laporan bulanan                                                                                          | ✓ |  |
|   |          | Melihat laporan bulanan berupa nomer registrasi tahanan, nama tahanan, dan hasil <i>assessment</i> risiko tahanan | ✓ |  |
|   |          | Melakukan verifikasi hasil <i>assessment</i>                                                                      | ✓ |  |
| 5 | Assessor | Mengubah <i>profile</i>                                                                                           | ✓ |  |
|   |          | Melakukan <i>assessment</i>                                                                                       | ✓ |  |
|   |          | Mengubah data <i>assessment</i>                                                                                   | ✓ |  |
|   |          | Melihat laporan bulanan berupa nomer registrasi tahanan, nama tahanan, dan hasil <i>assessment</i> risiko tahanan | ✓ |  |
|   |          | Mencetak laporan bulanan                                                                                          | ✓ |  |
|   |          | Melihat grafik bulanan                                                                                            | ✓ |  |

Berdasarkan hasil dari proses validasi yang dilakukan, didapatkan kesimpulan bahwa fitur sistem yang dirancang telah sesuai / ~~tidak sesuai~~ dengan kebutuhan pengguna.

\*Jika tidak sesuai dengan kebutuhan pengguna, maka fitur yang harus ditambah :

.....  
 .....  
 .....

Selong, 20 februari 2020  
 Responden  
  
 M. Fauzan Azim

**LEMBAR PENGUJIAN VALIDASI FITUR SISTEM DENGAN  
IMPLEMENTASI DESAIN DALAM PENGEMBANGAN SISTEM INFORMASI  
ASSESMENT RISIKO TAHANAN PADA LEMBAGA PEMASYARAKATAN  
KELAS II B SELONG**

**A. Identitas Responden**

Nama : Nasruddin  
Pekerjaan : PNS (Kasi)  
Instansi : Lapas Kelas II B Selong

**B. Petunjuk Pengisian**

Berilah tanda centang (✓) pada kolom **Setuju** jika fitur sistem sesuai dengan kebutuhan dan **Tidak Setuju** jika fitur sistem tidak sesuai dengan kebutuhan.

**C. Lembar Pengujian**

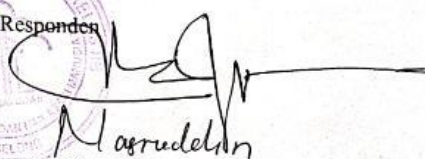
| No | Pengguna | Fitur                                                                                                             | Validasi |              |
|----|----------|-------------------------------------------------------------------------------------------------------------------|----------|--------------|
|    |          |                                                                                                                   | Setuju   | Tidak Setuju |
| 1  | admin    | Menambah <i>user</i>                                                                                              | ✓        |              |
|    |          | Mengubah <i>user</i>                                                                                              | ✓        |              |
|    |          | Menghapus <i>user</i>                                                                                             | ✓        |              |
|    |          | Menambah data tahanan                                                                                             | ✓        |              |
|    |          | Mengubah data tahanan                                                                                             | ✓        |              |
|    |          | Menghapus data tahanan                                                                                            | ✓        |              |
| 2  | Kalapas  | Melihat data assessor                                                                                             | ✓        |              |
|    |          | Melihat data tahanan                                                                                              | ✓        |              |
|    |          | Mengubah <i>profile</i>                                                                                           | ✓        |              |
|    |          | Melihat grafik bulanan                                                                                            | ✓        |              |
|    |          | Melihat laporan bulanan berupa nomer registrasi tahanan, nama tahanan, dan hasil <i>assessment</i> risiko tahanan | ✓        |              |
|    |          | Mencetak laporan bulanan                                                                                          | ✓        |              |
| 3  | Kasi     | Mengubah <i>profile</i>                                                                                           | ✓        |              |
|    |          | Melihat grafik bulanan                                                                                            | ✓        |              |

|   |          |                                                                                                                   |   |  |
|---|----------|-------------------------------------------------------------------------------------------------------------------|---|--|
|   |          | Melihat laporan bulanan berupa nomer registrasi tahanan, nama tahanan, dan hasil <i>assessment</i> risiko tahanan | ✓ |  |
|   |          | Mencetak laporan bulanan                                                                                          | ✓ |  |
| 4 | Kasubsi  | Mengubah <i>profile</i>                                                                                           | ✓ |  |
|   |          | Melihat grafik bulanan                                                                                            | ✓ |  |
|   |          | Mencetak laporan bulanan                                                                                          | ✓ |  |
|   |          | Melihat laporan bulanan berupa nomer registrasi tahanan, nama tahanan, dan hasil <i>assessment</i> risiko tahanan | ✓ |  |
|   |          | Melakukan verifikasi hasil <i>assessment</i>                                                                      | ✓ |  |
| 5 | Assessor | Mengubah <i>profile</i>                                                                                           | ✓ |  |
|   |          | Melakukan <i>assessment</i>                                                                                       | ✓ |  |
|   |          | Mengubah data <i>assessment</i>                                                                                   | ✓ |  |
|   |          | Melihat laporan bulanan berupa nomer registrasi tahanan, nama tahanan, dan hasil <i>assessment</i> risiko tahanan | ✓ |  |
|   |          | Mencetak laporan bulanan                                                                                          | ✓ |  |
|   |          | Melihat grafik bulanan                                                                                            | ✓ |  |

Berdasarkan hasil dari proses validasi yang dilakukan, didapatkan kesimpulan bahwa fitur sistem yang dirancang telah sesuai / tidak sesuai dengan kebutuhan pengguna.

\*Jika tidak sesuai dengan kebutuhan pengguna, maka fitur yang harus ditambah :

.....  
 .....  
 .....

Selong, 20 februari 2020  
 Responden  
  
 Masrudin