

LAMPIRAN

Lampiran A ***Software Requirements Specification***

PENGEMBANGAN APLIKASI PENGELOLAAN PRESENSI DAN PERIZINAN
ASRAMA MAHASISWA KAMPUS POLBAN BERBASIS *WEB*

Oleh:
Kelompok TA 208

ICHWAN LATIF FAJARI	NIM: 181511046
IRFAN SISWARA	NIM: 181511048
RIZQA FAUZIYYAH NABILA	NIM: 181511065

Revision History

<i>Date</i>	<i>Version</i>	<i>Description</i>	<i>Author</i>
25/04/2021	1.0	Pembuatan Dokumen SRS Pertama Secara Keseluruhan (<i>Introduction</i>)	Rizqa Fauziyyah N
04/05/2021	2.0	Penambahan bagian Overall Description dan Specific Requirement	Rizqa Fauziyyah N
10/05/2021	3.0	Penambahan <i>System Sequence Diagram</i>	Ichwan Latif
12/05/2021	4.0	Penambahan <i>Operation Contract</i> dan <i>User Interface</i>	Irfan Siswara
31/05/2021	5.0	Mengubah <i>Use Case Diagram</i> , Mengubah SSD dan <i>Operation Contract</i> .	Rizqa Fauziyyah N Irfan Siswara
13/06/2021	6.0	Penambahan <i>System Sequence Diagram</i> , <i>Operation Contract</i> , <i>User Interface</i> , <i>Functional Requirement</i> , <i>User Story</i> , dan <i>Requirement Traceability</i>	Rizqa Fauziyyah N Irfan Siswara
30/06/2021	7.0	Perubahan <i>System Sequence Diagram</i> , <i>Use Case Diagram</i> , <i>Operation Contract</i> dan	Rizqa Fauziyyah N Irfan Siswara
17/07/2021	8.0	<i>Balancing document</i>	Rizqa Fauziyyah N

Software Requirements Specification

I. Introduction

Bagian ini menjelaskan tujuan dibuatnya dokumen *Software Requirement Specification* (SRS), cakupan aplikasi, definisi, akronim, dan singkatan kata penting, referensi serta deskripsi umum dari bagian dokumen SRS ini.

I.1 Purpose

Tujuan dari penulisan dokumen ini untuk memberikan gambaran mengenai perangkat lunak yang dibangun. Gambaran perangkat lunak yang dijelaskan berupa gambaran umum dan penjelasan secara menyeluruh. Dokumen ini menjadi bahan acuan dalam proses pengembangan dan sebagai bahan evaluasi di akhir pengembangan. Dokumen ini diharapkan dapat memudahkan pengembang dalam mengembangkan aplikasi sehingga dapat terarah dan tidak menimbulkan ketidakjelasan dan ambiguitas pada aplikasi pengelolaan presensi dan perizinan asrama Polban ini.

I.2 Scope

Aplikasi pengelolaan presensi dan perizinan adalah aplikasi yang ditujukan bagi asrama Polban. Aplikasi bertujuan untuk dapat mengidentifikasi keberadaan mahasiswa dengan mengakses lokasi saat pengisian presensi kehadiran. Aplikasi juga mendukung pengajuan perizinan dan approval secara online sehingga dapat mengefisiensikan waktu dalam proses persetujuan pengelola asrama dan Wakil Direktur (Wadir) 3.

Gambaran umum pengelolaan presensi kehadiran yaitu aplikasi dapat mengidentifikasi keberadaan mahasiswa yang sedang melakukan presensi. Aplikasi mampu mengukur titik koordinat mahasiswa dengan koordinat asrama sehingga dapat menentukan status kehadiran mahasiswa tersebut sedang berada dalam

lingkungan asrama atau tidak. Terdapat perekapan data presensi yang dapat diakses oleh pengelola asrama dan Wadir 3 selaku stakeholder yang terlibat pada aplikasi.

Gambaran umum pengelolaan perizinan pada aplikasi ini yaitu dapat melakukan pengajuan izin dan pemberian *approval*. Aplikasi mampu mengirimkan notifikasi kepada pengelola asrama melalui *e-mail* saat mahasiswa mengajukan perizinan, begitupun mahasiswa akan menerima notifikasi *e-mail* jika perizinan yang diajukan telah ditindaklanjuti oleh Wadir 3. Terdapat perekapan data perizinan yang dapat diakses oleh pengelola asrama dan pihak manajemen (Wadir 3). Aplikasi yang dibangun berlaku untuk asrama gedung A, B, dan C.

I.3 Definitions, Acronyms and Abbreviations

I.3.1 Definitions

Definisi	Deskripsi
<i>Approval</i>	<i>Approval</i> adalah persetujuan yang akan diberikan oleh pengelola asrama dan Wadir 3 kepada mahasiswa terhadap pengajuan izin yang dilakukan.
Geolocation	Geolocation adalah identifikasi lokasi geografis suatu objek menggunakan perangkat yang tersambung ke internet.
Notifikasi	Notifikasi adalah pemberitahuan atau kabar tentang pengajuan izin yang masuk dan hasil <i>approval</i> dari pengajuan izin tersebut.
SRS	SRS adalah dokumen yang menjelaskan tentang berbagai kebutuhan yang harus dipenuhi oleh suatu perangkat lunak.

I.3.2 Acronyms

Akronim	Deskripsi
Polban	Politeknik Negeri Bandung
Wadir	Wakil Direktur

I.3.3 Abbreviations

Singkatan	Deskripsi
API	<i>Application Programming Language</i>
GPS	<i>Global Positioning System</i>
IEEE	<i>Institute of Electrical and Electronics Engineers</i>
JSON	<i>JavaScript Object Notation</i>
PC	<i>Personal Computer</i>
SRS	<i>Software Requirement Specification</i>

I.4 References

Template: Software Requirements Specification, Rational Software Corporation 1987-2001.

Larman, Craig. (2004): Applying UML and Patterns: An Introduction to Object – Oriented Analysis and Design and Iterative Development, Third Edition, Addison Wesley Professional.

I.5 Technology to be Used

Teknologi pengembangan aplikasi yang akan digunakan telah dipaparkan pada dokumen laporan tugas akhir sub bab II.2.2 Teknologi Pendukung.

I.6 Overview

Dokumen ini menjelaskan tiga poin utama, yaitu:

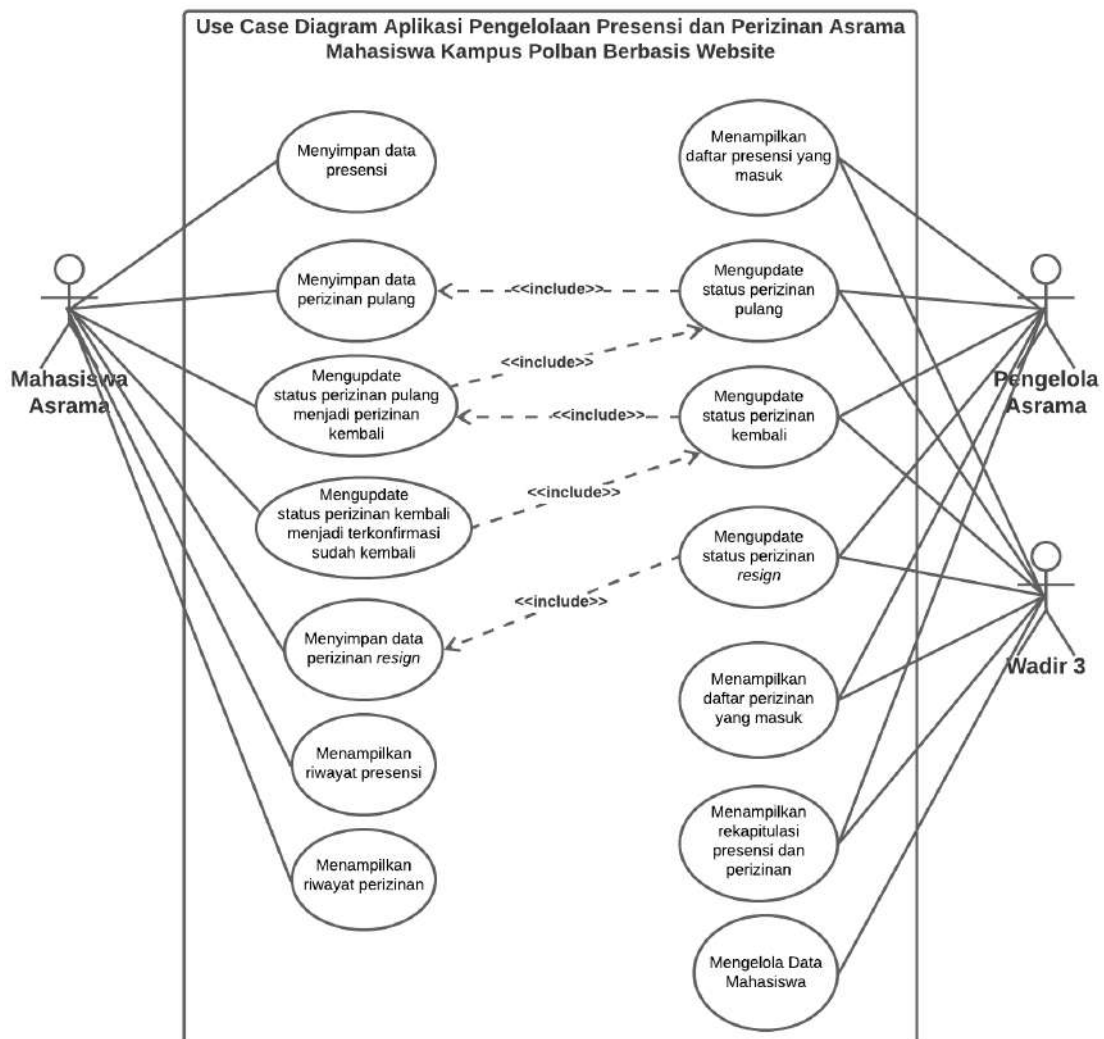
1. Bagian 1 *Introduction*: Merupakan pengantar dari SRS yang berisi tujuan pembuatan dokumen, ruang lingkup aplikasi, definisi istilah, singkatan dan akronim yang terdapat pada dokumen ini, referensi yang digunakan dan deskripsi umum dokumen SRS ini.
2. Bagian 2 *Overall Description*: Menjelaskan tentang gambaran secara umum produk yang dibuat terdiri atas, *use case diagram*, arsitektur diagram, *database design*, batasan, ketergantungan, dan asumsi yang diambil dalam pengembangan aplikasi.
3. Bagian 3 *Specific Requirements*: Menjelaskan terkait persyaratan spesifik yang mencakup persyaratan antarmuka atau diluar aplikasi, persyaratan fungsional aplikasi, persyaratan kinerja aplikasi baik persyaratan statik atau dinamik, batasan desain aplikasi, atribut sistem perangkat lunak yang meliputi *access security*, *responsibility*, *performance*, dan persyaratan spesifik yang dikelompokkan berdasarkan fitur.

II. Overall Description

Bagian ini menjelaskan gambaran umum secara keseluruhan yang mempengaruhi produk aplikasi yang dibuat. Bagian ini berisi penggambaran *use case model*, *architecture diagram*, *database design*, batasan, asumsi, dan ketergantungan.

II.1 Use Case Model Survey

Berikut merupakan penggambaran interaksi antara aktor dengan aplikasi pengelolaan presensi dan perizinan asrama Polban yang digambarkan pada Gambar II.1 menggunakan *modeling tools use case diagram*. *Use case diagram* adalah UML yang digunakan pada tahap analisis untuk menggambarkan hubungan antar aktor dengan aplikasi. Setiap bulatan yang ditulis, merupakan tindakan dari aplikasi terhadap aksi aktor. Adapun aksi aktor ditulis pada *use case story* di bagian *stakeholder interest*.



Gambar II.1 Use case diagram

Penggambaran *use case diagram* aplikasi pengelolaan presensi dan perizinan asrama Polban dirancang menggunakan *tools draw.io*. Berikut merupakan deskripsi setiap *case* pada *use case diagram* aplikasi pengelolaan presensi dan perizinan asrama Polban sebagai berikut:

1. Menyimpan Data Presensi [UC-01]

Menyimpan data presensi kehadiran adalah respon aplikasi terhadap kegiatan aktor mahasiswa asrama yaitu mengisi presensi kehadiran. Presensi kehadiran diisi oleh mahasiswa asrama. Setiap mahasiswa diharuskan mengisi presensi kehadiran setiap hari berdasarkan waktu yang

telah ditentukan. Pengisian dilakukan melalui perangkat mahasiswa masing-masing dengan mengaktifkan fitur GPS pada perangkat yang digunakan. Aplikasi akan mengambil data lokasi keberadaan mahasiswa ketika melakukan presensi. GPS membantu aplikasi untuk dapat menemukan titik koordinat mahasiswa secara lebih akurat. Koordinat yang telah didapatkan divalidasi dengan perhitungan formula Haversine untuk membuktikan mahasiswa tersebut benar berada di lingkungan asrama atau tidak. Setelah koordinat berhasil divalidasi dan diidentifikasi, maka aplikasi mencatat kehadiran mahasiswa tersebut beserta dengan status kehadirannya. Setelah selesai, data akan ditampilkan pada riwayat presensi.

2. Menyimpan Data Perizinan Pulang [UC-02]

Menyimpan data perizinan pulang adalah bentuk respon aplikasi terhadap kegiatan aktor mahasiswa asrama yaitu mengajukan perizinan pulang. Mahasiswa asrama dapat mengajukan perizinan melalui aplikasi dengan jenis izin pulang ke rumah/bermalam di tempat lain. Aplikasi akan menyimpan perizinan yang diajukan dan menampilkannya pada riwayat perizinan. Perizinan harus mendapat persetujuan dari pengelola asrama dan Wadir 3. Oleh karena itu, mahasiswa diharuskan mengajukan perizinan beberapa hari sebelum hari izin yang diajukan melalui aplikasi. Agar mendapat persetujuan izin dari pengelola dan Wadir 3, aplikasi mendukung adanya pengiriman notifikasi melalui *e-mail* kepada pengelola dan Wadir 3.

3. Mengupdate Status Perizinan Pulang [UC-03]

Mengupdate status perizinan pulang adalah bentuk respon aplikasi terhadap kegiatan aktor pengelola dan Wadir 3 yaitu memberikan *approval* terhadap perizinan pulang yang diajukan mahasiswa. Pengajuan izin yang masuk ditandai dengan adanya notifikasi *e-mail* pada perangkat pengelola. Pengelola memberikan *approval*. Kemudian hasil *approval* akan dikirim kepada Wadir 3 melalui *e-mail*. Hal tersebut sebagai pertanda bahwa perizinan sudah ditindaklanjuti pengelola dan saatnya Wadir 3 untuk

memberikan persetujuan. Hasil *approval* yang dilakukan Wadir 3 akan dikirim ke mahasiswa yang mengajukan izin sebelumnya, melalui pengiriman *e-mail* kembali.

4. **Mengupdate Status Perizinan Pulang Menjadi Perizinan Kembali [UC-04]**
 Mengupdate status perizinan pulang menjadi perizinan kembali adalah bentuk respon aplikasi terhadap kegiatan aktor mahasiswa asrama yaitu mengajukan perizinan untuk dapat kembali ke asrama. Pengajuan izin tersebut ditujukan kepada pengelola dan Wadir 3 untuk ditindaklanjuti. Mahasiswa harus mengajukan izin ketika akan kembali ke asrama melalui aplikasi. Mahasiswa akan diminta untuk mengisi *form* pengajuan izin kembali ke asrama. Pengajuan izin akan dikirim dan ditandai dengan adanya notifikasi *e-mail* pada perangkat pengelola.

5. **Mengupdate Status Perizinan Kembali [UC-05]**
 Mengupdate status perizinan kembali adalah bentuk respon aplikasi terhadap kegiatan aktor pengelola dan Wadir 3 yaitu memberikan *approval* terhadap perizinan kembali ke asrama yang diajukan mahasiswa. Pengajuan izin yang masuk ditandai dengan adanya notifikasi *e-mail* pada perangkat pengelola. Pengelola memberikan *approval*. Kemudian hasil *approval* akan dikirim kepada Wadir 3 melalui *e-mail*. Hal tersebut sebagai pertanda bahwa perizinan sudah ditindaklanjuti pengelola dan saatnya Wadir 3 untuk memberikan persetujuan. Hasil *approval* yang dilakukan Wadir 3 akan dikirim ke mahasiswa yang mengajukan izin sebelumnya, melalui pengiriman *e-mail* kembali.

6. **Mengupdate Status Perizinan Kembali Menjadi Terkonfirmasi Sudah Kembali [UC-06]**
 Mengupdate status perizinan kembali menjadi terkonfirmasi sudah kembali adalah bentuk respon aplikasi terhadap kegiatan aktor mahasiswa asrama yaitu melakukan konfirmasi melalui aplikasi bahwa dirinya telah kembali

dan berada di asrama. Pengelola dan Wadir 3 akan menerima pemberitahuan melalui *e-mail* bahwa mahasiswa yang bersangkutan telah kembali dan berada di asrama. Sebelumnya, aplikasi akan melakukan validasi keberadaan mahasiswa. Ketika mahasiswa mengkonfirmasi kepulangannya melalui aplikasi, aplikasi akan mengecek keberadaan mahasiswa.

7. Menyimpan Data Perizinan *Resign* [UC-07]

Menyimpan data perizinan *resign* adalah bentuk respon aplikasi terhadap kegiatan aktor mahasiswa asrama yaitu mengajukan perizinan *resign*. Mahasiswa asrama dapat mengajukan perizinan melalui aplikasi dengan mengisi *form* izin *resign*. Aplikasi akan menyimpan perizinan yang diajukan dan menampilkannya pada riwayat perizinan. Perizinan harus mendapat persetujuan dari pengelola asrama dan Wadir 3. Oleh karena itu, mahasiswa diharuskan mengajukan perizinan beberapa hari sebelum hari izin yang diajukan melalui aplikasi. Agar mendapat persetujuan izin dari pengelola dan Wadir 3, aplikasi mendukung adanya pengiriman notifikasi melalui *e-mail* kepada pengelola dan Wadir 3.

8. Mengupdate Status Perizinan *Resign* [UC-08]

Mengupdate status perizinan *resign* adalah bentuk respon aplikasi terhadap kegiatan aktor pengelola dan Wadir 3 yaitu memberikan *approval* terhadap perizinan *resign* yang diajukan mahasiswa. Pengajuan izin yang masuk ditandai dengan adanya notifikasi *e-mail* pada perangkat pengelola. Pengelola memberikan *approval*. Kemudian hasil *approval* akan dikirim kepada Wadir 3 melalui *e-mail*. Hal tersebut sebagai pertanda bahwa perizinan sudah ditindaklanjuti pengelola dan saatnya Wadir 3 untuk memberikan persetujuan. Hasil *approval* yang dilakukan Wadir 3 akan dikirim ke mahasiswa yang mengajukan izin sebelumnya, melalui pengiriman *e-mail* kembali.

9. Menampilkan Riwayat Presensi [UC-09]

Menampilkan riwayat presensi adalah bentuk respon aplikasi terhadap kegiatan aktor yaitu mengakses atau melihat riwayat presensi yang telah dilakukan. Selain itu, riwayat presensi ditampilkan ketika aktor tersebut selesai mengisi presensi kehadiran. Riwayat presensi dapat diakses oleh mahasiswa asrama.

10. Menampilkan Riwayat Perizinan [UC-10]

Menampilkan riwayat perizinan adalah bentuk respon aplikasi terhadap kegiatan aktor yaitu mengakses atau melihat riwayat perizinan yang telah dilakukan. Selain itu, riwayat perizinan ditampilkan ketika aktor tersebut selesai mengajukan perizinan melalui aplikasi. Riwayat perizinan dapat diakses oleh mahasiswa asrama. Pada riwayat perizinan, mahasiswa dapat mengubah status izin menjadi sudah kembali, terhadap status izin yang sebelumnya dimiliki. Dengan pengubahan status izin tersebut akan diteruskan kepada pengelola dan Wadir 3 untuk ditindaklanjuti.

11. Menampilkan Daftar Perizinan yang Masuk [UC-11]

Menampilkan daftar perizinan yang masuk adalah bentuk respon aplikasi terhadap kegiatan aktor yaitu mengakses halaman data perizinan yang masuk melalui aplikasi. Daftar perizinan yang masuk dapat diakses oleh pengelola dan Wadir 3.

12. Menampilkan Rekapitulasi Presensi dan Perizinan [UC-12]

Menampilkan rekapitulasi data presensi dan perizinan adalah bentuk respon aplikasi terhadap kegiatan aktor yaitu mengakses halaman rekap data. Perekapitan dapat diakses oleh pengelola asrama dan Wadir 3 melalui aplikasi. Aplikasi akan menampilkan rekap data perizinan berdasarkan *range* tanggal yang diinputkan. Selain itu, rekap dapat diunduh ke perangkat pengguna yang digunakan.

13. Mengelola Data Mahasiswa [UC-13]

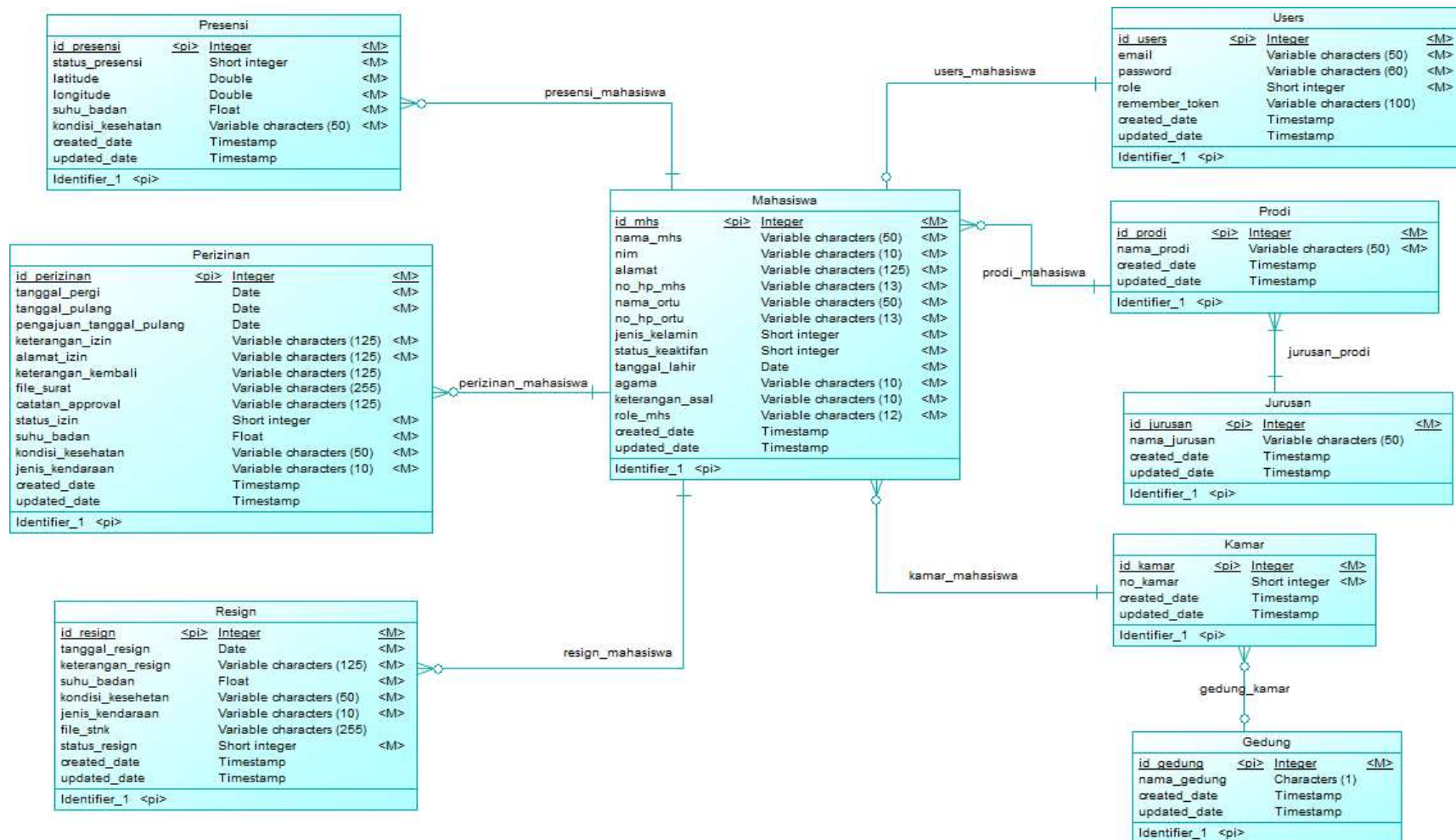
Mengelola data mahasiswa adalah respon aplikasi terhadap kegiatan aktor dalam upaya untuk mengelola data mahasiswa. Pengelolaan yang dapat dilakukan adalah menampilkan detail data mahasiswa yang dipilih, menambahkan data mahasiswa (*import* dan *insert*), dan mengubah data mahasiswa. Pengelolaan hanya dapat dilakukan oleh Wadir 3 sebagai *super admin*.

14. Menampilkan Daftar Presensi yang Masuk [UC-14]

Menampilkan daftar presensi yang masuk adalah bentuk respon aplikasi terhadap kegiatan aktor yaitu mengakses halaman data presensi yang masuk melalui aplikasi. Daftar presensi akan menampilkan data kehadiran seluruh mahasiswa setiap hari. Daftar presensi yang masuk dapat diakses oleh pengelola asrama dan Wadir 3.

II.2 Architecture Diagram and Database Design

Penggambaran arsitektur diagram dijelaskan pada subbab *product perspective*. Adapun gambaran rancangan *database* yang digunakan pada aplikasi, digambarkan menggunakan *conceptual data model*. *Conceptual data model* adalah skema penggambaran *database* secara konseptual, hasil dari analisis dan pengumpulan data. Hasil analisis terhadap kebutuhan aplikasi, dibuatlah *conceptual data model* untuk menyatakan dan memetakan tiap data yang digunakan. Penggambaran skema *database* ini mengacu pada pembentukan *domain model* dan hasil analisis yang telah dilakukan terkait penunjang fungsi aplikasi yang diimplementasi. Penggambaran *conceptual data model* dirancang menggunakan *tools* PowerDesigner yang ditunjukkan pada Gambar 2.2.



Gambar II.3 Conceptual data model

Database pada aplikasi pengelolaan presensi dan perizinan asrama Polban ini memiliki sembilan entitas yang saling berelasi dan dua entitas tambahan untuk mendukung penggunaan teknologi yang digunakan. Entitas tersebut adalah:

1. Users

Entitas Users berfungsi untuk menampung semua data akun pengguna. Entitas ini terdiri atas atribut *id_users*, *e-mail*, *password*, *role*, *remember_token*, *created_at*, dan *updated_at*. Entitas Users berelasi dengan entitas Mahasiswa (*one to one*). Data pada entitas ini digunakan ketika *user* akan masuk ke aplikasi.

Tabel II.4 Tabel entitas Users

Atribut	Tipe Data
<i>id_users</i>	Integer
<i>e-mail</i>	Varchar (50)
<i>password</i>	Varchar (255)
<i>role</i>	Short Integer
<i>remember_token</i>	Varchar (100)
<i>created_at</i>	Timestamp
<i>updated_at</i>	Timestamp

2. Mahasiswa

Entitas Mahasiswa berfungsi untuk menampung semua data mahasiswa. Entitas ini terdiri atas atribut *id_mhs*, *nama_mhs*, *nim*, *alamat*, *no_hp_mhs*, *nama_ortu*, *no_hp_ortu*, *jenis_kelamin*, *status_keaktifan*, *tanggal_lahir*, *agama*, *keterangan_asal*, *role_mhs*, *created_at*, dan *updated_at*. Entitas mahasiswa berelasi dengan entitas Presensi (*one to many*), entitas Perizinan (*one to many*), entitas Resign (*one to one*), entitas Users (*one to one*), entitas Prodi (*many to one*), dan entitas Kamar (*many to one*).

Tabel II.5 Tabel entitas Mahasiswa

Atribut	Tipe Data
<i>id_mhs</i>	Integer
<i>nama_mhs</i>	Varchar (50)
<i>nim</i>	Varchar (10)
<i>alamat</i>	Varchar (125)
<i>no_hp_mhs</i>	Varchar (13)
<i>nama_ortu</i>	Varchar (50)

no_hp_ortu	Varchar (13)
jenis_kelamin	Short Integer
status_keaktifan	Short Integer
tanggal_lahir	Date
agama	Varchar (10)
keterangan_asal	Varchar (10)
role_mhs	Varchar (12)
created_date	Timestamp
updated_date	Timestamp

3. Perizinan

Entitas Perizinan berfungsi untuk menampung semua data Perizinan. Entitas ini terdiri atas atribut id_perizinan, tanggal_pergi, tanggal_pulang, pengajuan_tanggal_kembali, keterangan_izin, alamat_izin, keterangan_kembali, surat_pendukung, catatan_approval, status_izin, suhu_badan, kondisi_kesehatan, jenis_kendaraan, created_date dan updated_date. Entitas perizinan berelasi dengan entitas Mahasiswa (*many to one*).

Tabel II.6 Tabel entitas Perizinan

Atribut	Tipe Data
id_perizinan	Integer
tanggal_pergi	Date
tanggal_pulang	Date
pengajuan_tanggal_pulang	Date
keterangan_izin	Varchar (125)
alamat_izin	Varchar (125)
keterangan_kembali	Varchar (125)
file_surat	Varchar (255)
catatan_approval	Varchar (125)
status_izin	Short Integer
suhu_badan	Float
kondisi_kesehatan	Varchar (50)
jenis_kendaraan	Varchar (10)
created_date	Timestamp
updated_at	Timestamp

4. Presensi

Entitas Presensi berfungsi untuk menampung semua data presensi. Entitas ini terdiri atas atribut `id_presensi`, `status_presensi`, `latitude`, dan `longitude`, `suhu_badan`, `kondisi_kesehatan`, `created_date`, dan `updated_date`. Entitas Presensi berelasi dengan entitas Mahasiswa (*many to one*).

Tabel II.7 Tabel entitas Presensi

Atribut	Tipe Data
<code>id_presensi</code>	Integer
<code>status_presensi</code>	Char (1)
<code>latitude</code>	Double
<code>longitude</code>	Double
<code>suhu_badan</code>	Float
<code>kondisi_kesehatan</code>	Varchar (50)
<code>created_date</code>	Timestamp
<code>updated_date</code>	Timestamp

5. Resign

Entitas Resign berfungsi untuk menampung semua data mahasiswa *resign*. Entitas ini terdiri atas atribut `id_resign`, `tanggal_resign`, `keterangan_resign`, `suhu_badan`, `kondisi_kesehatan`, `jenis_kendaraan`, `file_stnk`, `status_resign`, `created_date`, dan `updated_date`. Entitas Resign berelasi dengan entitas Mahasiswa (*many to one*).

Tabel II.8 Tabel entitas Resign

Atribut	Tipe Data
<code>id_resign</code>	Integer
<code>tanggal_resign</code>	Date
<code>keterangan_resign</code>	Varchar (125)
<code>suhu_badan</code>	Float
<code>kondisi_kesehatan</code>	Varchar (50)
<code>jenis_kendaraan</code>	Varchar (10)
<code>file_stnk</code>	Varchar (255)
<code>status_resign</code>	Short Integer
<code>created_date</code>	Timestamp
<code>updated_date</code>	Timestamp

6. Kamar

Entitas Kamar berfungsi untuk menampung semua data kamar asrama. Entitas ini terdiri atas atribut `id_kamar`, `no_kamar`, `created_date`, dan `updated_date`. Entitas Kamar berelasi dengan entitas Mahasiswa (*one to many*).

Tabel II.9 Tabel entitas Kamar

Atribut	Tipe Data
<code>id_kamar</code>	Integer
<code>no_kamar</code>	Short Integer
<code>created_date</code>	Timestamp
<code>updated_date</code>	Timestamp

7. Gedung

Entitas Gedung berfungsi untuk menampung semua data gedung asrama. Entitas ini terdiri atas atribut `id_gedung`, `nama_gedung`, `created_date`, dan `updated_date`. Entitas Gedung berelasi dengan entitas Kamar (*one to many*).

Tabel II.10 Tabel entitas Gedung

Atribut	Tipe Data
<code>id_gedung</code>	Integer
<code>nama_gedung</code>	Char (1)
<code>created_date</code>	Timestamp
<code>updated_date</code>	Timestamp

8. Prodi

Entitas Prodi berfungsi untuk menampung semua data prodi di kampus Polban. Entitas ini terdiri atas atribut `id_prodi`, `nama_prodi`, `created_date`, dan `updated_date`. Entitas Prodi berelasi dengan entitas Mahasiswa (*one to many*), dan entitas Jurusan (*many to one*).

Tabel II.11 Tabel entitas Prodi

Atribut	Tipe Data
id_prodi	Integer
nama_prodi	Varchar (50)
created_date	Timestamp
updated_date	Timestamp

9. Jurusan

Entitas Jurusan berfungsi untuk menampung semua data jurusan di kampus Polban. Entitas ini terdiri atas atribut id_jurusan, nama_jurusan, created_date, dan updated_date. Entitas jurusan berelasi dengan entitas Prodi (*many to one*).

Tabel II.12 Tabel entitas Jurusan

Atribut	Tipe Data
id_jurusan	Integer
nama_jurusan	Varchar (50)
created_date	Timestamp
updated_date	Timestamp

II.3 Assumptions, Constraint, and Dependencies

Terdapat asumsi, batasan, dan ketergantungan untuk aplikasi presensi dan perizinana asrama ini, yaitu:

II.3.1 Assumption

Adapun asumsi untuk aplikasi pengelolaan presensi dan perizinana asrama ini, yaitu:

1. Perangkat yang digunakan memiliki fitur GPS yang dapat membaca lokasi dengan baik.
2. GPS dalam keadaan aktif saat melakukan pengisian presensi dan konfirmasi kembali ke asrama melalui aplikasi.
3. Akses lokasi ke perangkat disetujui oleh mahasiswa asrama yang muncul pada layar aplikasi.
4. Seluruh pengguna (mahasiswa asrama, pengelola asrama, dan Wadir 3) memiliki akun asrama yang terdaftar pada aplikasi.

5. Seluruh pengguna (mahasiswa asrama, pengelola asrama, dan Wadir 3) memiliki akun *e-mail* aktif Polban.

II.3.2 *Constraint*

Adapun *constraint* pada aplikasi pengelolaan presensi dan perizinana asrama ini, yaitu:

1. Aplikasi tidak dapat menangkap koordinat keberadaan saat pengisian presensi dan konfirmasi kembali dengan akurasi tinggi, selain menggunakan perangkat yang mendukung GPS aktif (Laptop atau PC tanpa GPS tidak disarankan).
2. Aplikasi tidak dapat menangkap koordinat keberadaan saat pengisian presensi dan konfirmasi kembali, jika akses izin lokasi perangkat tidak diizinkan.
3. Perangkat harus tetap terhubung dengan koneksi internet yang stabil saat menggunakan aplikasi. Jika tidak, aplikasi akan terhenti dan tidak dapat diakses.

II.3.3 *Dependencies*

Adapun ketergantungan untuk aplikasi presensi dan perizinan asrama ini, yaitu:

1. Presensi kehadiran dapat mendeteksi lokasi pengguna dengan lebih akurat ketika GPS pada perangkat dalam keadaan aktif.
2. Presensi kehadiran tidak dapat mengakses lokasi perangkat tanpa izin yang diberikan mahasiswa pada akses permisi yang muncul pada layar aplikasi.

III. *Specific Requirements*

Bagian ini menjelaskan terkait persyaratan spesifik yang mencakup detail penjelasan mengenai *use case*, penetapan persyaratan berdasarkan *use case*, hasil analisis *requirement* menggunakan *system sequence diagram* dan *operation contract*. Selain itu, menjelaskan terkait antar muka pengguna, antar muka perangkat lunak dan keras, antar muka proses komunikasi dan persyaratan atribut aplikasi yang meliputi *access security* dan *responsibility*.

III.1 *Use Case Reports*

Seluruh bulatan *use case* yang ada pada *use case diagram* dibuatkan *use case story*. Hal tersebut dilakukan untuk mengetahui sejauh mana setiap *use case* memiliki tanggung jawab, dan untuk memudahkan dalam proses perancangan karena isinya yang *detail* menggambarkan *step by step* proses. Berikut merupakan perincian *use case story* dari setiap *use case* berdasarkan penggambaran yang tertera pada *use case diagram* sebelumnya.

Terdapat UML *System Sequence Diagram* (SSD) yang merupakan pemodelan yang bertujuan untuk menggambarkan interaksi antar aktor dengan sistem. Penggambaran SSD pada subbab ini menggunakan *tools draw.io*, sedangkan *style* perincian *use case story* menggunakan *fully dressed style*. Terdapat pula *operation contract* yang digunakan untuk mengetahui kapan pembentukan objek di dalam *method*. Pembuatan *operation contract* dilakukan setelah menggambarkan SSD.

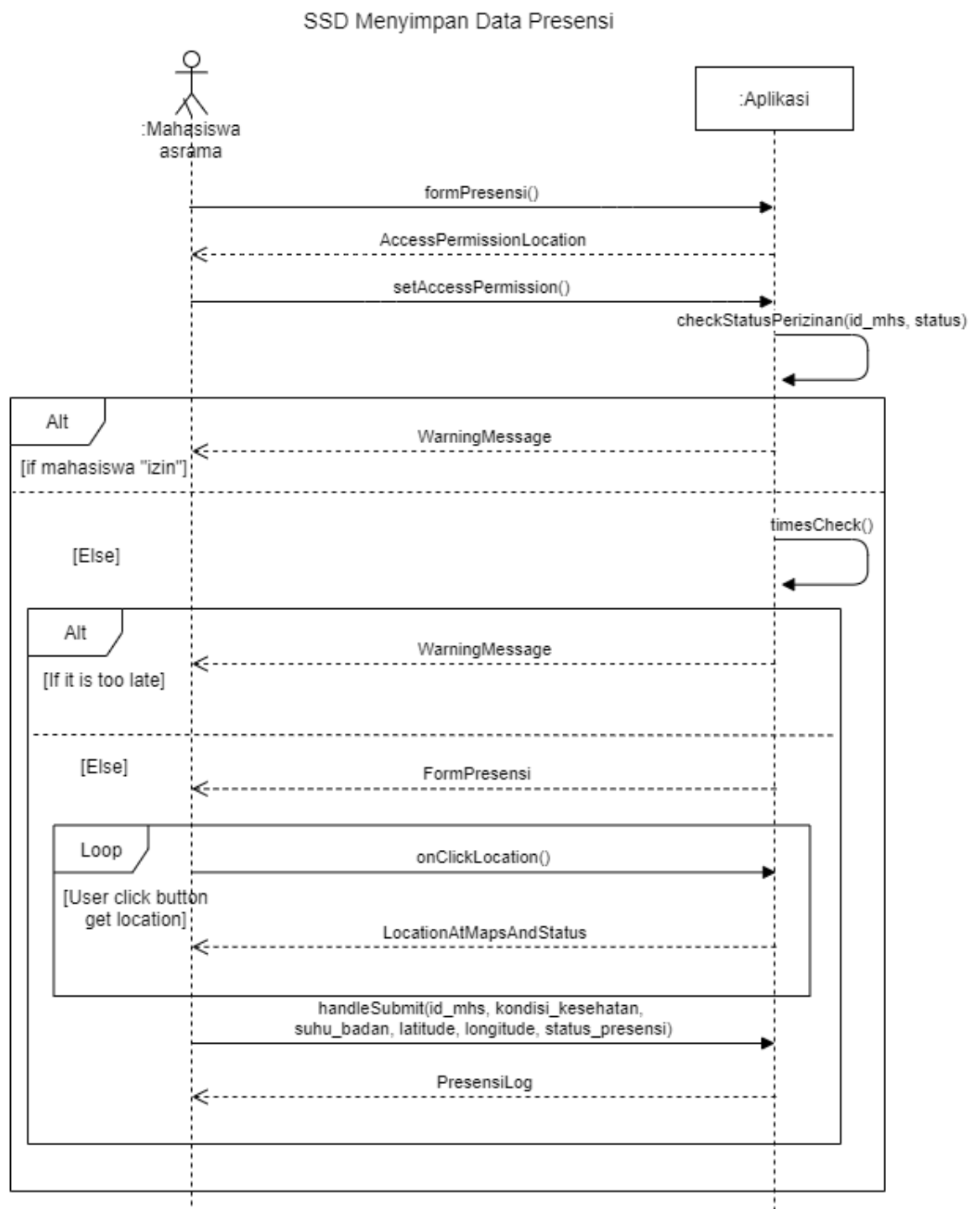
III.1.1 *Use Case : Menyimpan Data Presensi*

Tabel III.1 UC-01 menyimpan data presensi

Use Case ID	UC-01
Use Case Name	Menyimpan data presensi
Scope	System under design
Level	User goal
Primary Actor	Mahasiswa asrama
Stakeholders and interests	Mahasiswa asrama Interest: - Ingin mengisi presensi kehadiran.
Preconditions	

	<ol style="list-style-type: none"> 1. Mahasiswa asrama telah mengakses url website dengan benar pada perangkat melalui <i>browser</i>. 2. Mahasiswa asrama telah terdaftar dan terverifikasi sebagai <i>user</i> pada aplikasi. 3. GPS pada perangkat pengguna telah diaktifkan. 4. Aplikasi menampilkan halaman <i>dashboard</i> mahasiswa asrama.
Success Guarantee	Data presensi baru berhasil muncul pada riwayat presensi.
Main Success Scenario	<ol style="list-style-type: none"> 1. Mahasiswa asrama mengakses halaman pengisian presensi kehadiran. 2. Aplikasi meminta akses lokasi pada perangkat pengguna. 3. Aplikasi mengecek status perizinan mahasiswa. 4. Aplikasi mengecek ketepatan waktu presensi. 5. Aplikasi menampilkan halaman <i>form</i> presensi. 6. Mahasiswa asrama mengisi <i>form</i> berisi kondisi kesehatan, suhu badan, dan koordinat lokasi. 7. Menekan tombol ambil lokasi. 8. Aplikasi menampilkan <i>maps</i> keberadaan mahasiswa. 9. Aplikasi menampilkan status keberadaan, sedang berada di lingkungan asrama atau di luar asrama berdasarkan perhitungan dengan penetapan radius 50 meter dari titik asrama. 10. Ulangi poin 7-9 selama mahasiswa menekan tombol untuk cek lokasi. 11. Mahasiswa asrama mensubmit presensi. 12. Aplikasi menampilkan riwayat presensi.
Extensions	<ol style="list-style-type: none"> 3a. Jika status perizinan mahasiswa sedang izin, maka aplikasi menampilkan pesan peringatan. 4a. Jika mahasiswa asrama mengakses presensi di luar pukul 16.00-20.00 maka aplikasi menampilkan pesan peringatan tidak dapat mengisi presensi.
Special Requirements	-
Technology and Data Variations List	GPS yang terdapat pada perangkat mahasiswa asrama.
Frequency of Occurrence	Dilakukan satu kali setiap hari pada rentang waktu 16.00-20.00.
Miscellaneous	-

III.1.1.1 System Sequence Diagram : Menyimpan Data Presensi [SSD-01]



Gambar III.1 SSD-01 menyimpan data presensi

III.1.1.2 **Operation Contract : Menyimpan Data Presensi [OC-01]**

1. formPresensi()

Tabel III.2 OC-01 form presensi

Operation	formPresensi()
Cross Reference	Use Cases: Menyimpan data presensi
Precondition	<i>Instance</i> Mahasiswa sedang berjalan
Postcondition	<i>Instance</i> Presensi telah dibuat

2. checkStatusPerizinan(id_mhs, status)

Tabel III.3 OC-01 cek status presensi

Operation	checkStatusPerizinan(id_mhs)
Cross Reference	Use Cases: Menyimpan data presensi
Precondition	<i>Instance</i> Mahasiswa sedang berjalan <i>Instance</i> Presensi sedang berjalan
Postcondition	<i>Instance</i> Perizinan telah dibuat <i>Instance</i> Perizinan berasosiasi dengan <i>instance</i> Mahasiswa berdasarkan id

3. timesCheck()

Tabel III.4 OC-01 cek waktu

Operation	timesCheck()
Cross Reference	Use Cases: Menyimpan data presensi
Precondition	<i>Instance</i> Mahasiswa sedang berjalan <i>Instance</i> Presensi sedang berjalan
Postcondition	Presensi.waktu menjadi currentTime

4. onClickLocation()

Tabel III.5 OC-01 klik lokasi

Operation	onClickLocation()
Cross Reference	Use Cases: Menyimpan data presensi
Precondition	<i>Instance</i> Mahasiswa sedang berjalan <i>Instance</i> Presensi sedang berjalan
Postcondition	Presensi.latitude menjadi latitude Presensi.longitude menjadi longitude Presensi.suhu_badan menjadi suhu_badan Presensi.kondisi_kesehatan menjadi kondisi_kesehatan

5. handleSubmit(id_mhs, kondisi_kesehatan, suhu_badan, latitude, longitude, status_presensi)

Tabel III.6 OC-01 *handle submit*

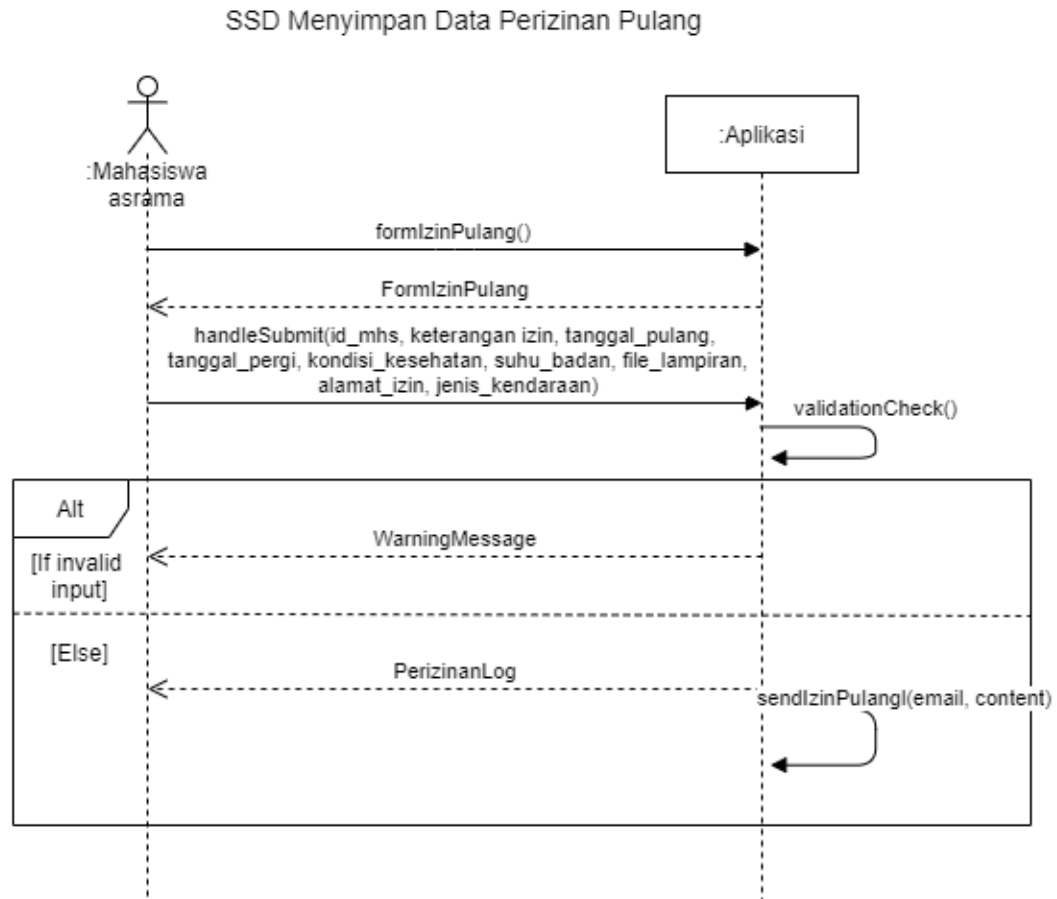
Operation	handleSubmit (id_mhs)
Cross Reference	Use Cases: Menyimpan data presensi
Precondition	Instance Mahasiswa sedang berjalan Instance Presensi sedang berjalan
Postcondition	Presensi.isComplete menjadi true

III.1.2 Use Case : Menyimpan Data Perizinan Pulang

Tabel III.7 UC-02 menyimpan data perizinan pulang

Use Case ID	UC-02
Use Case Name	Menyimpan data perizinan pulang
Scope	System under design
Level	User goal
Primary Actor	Mahasiswa
Stakeholders and interests	Mahasiswa asrama Interest: - Ingin mengajukan perizinan pulang.
Preconditions	1. Mahasiswa asrama telah mengakses url <i>website</i> dengan benar pada perangkat melalui <i>browser</i> . 2. Mahasiswa asrama telah terdaftar dan terverifikasi sebagai <i>user</i> pada aplikasi. 3. Aplikasi menampilkan halaman <i>dashboard</i> mahasiswa asrama.
Success Guarantee	Data perizinan baru berhasil muncul di riwayat perizinan.
Main Success Scenario	1. Mahasiswa asrama mengakses halaman perizinan pulang. 2. Aplikasi menampilkan halaman <i>form</i> perizinan pulang. 3. Mahasiswa asrama mengisi seluruh isian pada <i>form</i> berisi keterangan izin, tanggal pulang, tanggal pergi, kondisi kesehatan, suhu badan, file lampiran, alamat pulang, dan jenis kendaraan. 4. Mahasiswa mensubmit <i>form</i> . 5. Aplikasi melakukan validasi <i>form</i> . 6. Aplikasi menampilkan riwayat perizinan. 7. Aplikasi mengirim notifikasi pengajuan izin kepada <i>e-mail</i> pengelola asrama.
Extensions	5a. Jika <i>form</i> tidak valid (<i>field</i> kosong atau data yang diisikan salah), maka aplikasi menampilkan pesan peringatan.
Special Requirements	-
Technology and Data Variations List	-
Frequency of Occurrence	Sewaktu-waktu saat mahasiswa asrama mengajukan perizinan pulang.
Miscellaneous	-

III.1.2.1 System Sequence Diagram : Menyimpan Data Perizinan Pulang [SSD-02]



Gambar III.2 SSD-02 menyimpan data perizinan pulang

III.1.2.2 Operation Contract : Menyimpan Data Perizinan Pulang [OC-02]

1. formIzinPulang()

Tabel III.8 OC-02 form izin pulang

Operation	formIzinPulang()
Cross Reference	Use Cases: Menyimpan data perizinan pulang
Precondition	Instance Mahasiswa sedang berjalan
Postcondition	Instance Perizinan telah dibuat

2. handleSubmit(id_mhs, keterangan izin, tanggal_pulang, tanggal_pergi, kondisi_kesehatan, suhu_badan, file_lampiran, alamat_izin, jenis_kendaraan)

Tabel III.9 OC-02 *handle submit*

Operation	handleSubmit(id_mhs, keterangan izin, tanggal_pulang, tanggal_pergi, kondisi_kesehatan, suhu_badan, file_lampiran, alamat_izin, jenis_kendaraan)
Cross Reference	Use Cases: Menyimpan data perizinan pulang
Precondition	<i>Instance</i> Mahasiswa sedang berjalan <i>Instance</i> Perizinan sedang berjalan
Postcondition	Perizinan.tanggal_pergi menjadi tanggal_pergi Perizinan.tanggal_pulang menjadi tanggal_pulang Perizinan.keterangan_izin menjadi keterangan_izin Perizinan.surat_pendukung menjadi surat_pendukung Perizinan.alamat_izin menjadi alamat_izin Perizinan.suhu_badan menjadi suhu_badan Perizinan.kondisi_kesehatan menjadi kondisi_kesehatan Perizinan.jenis_kendaraan menjadi jenis_kendaraan Perizinan.isComplete menjadi true

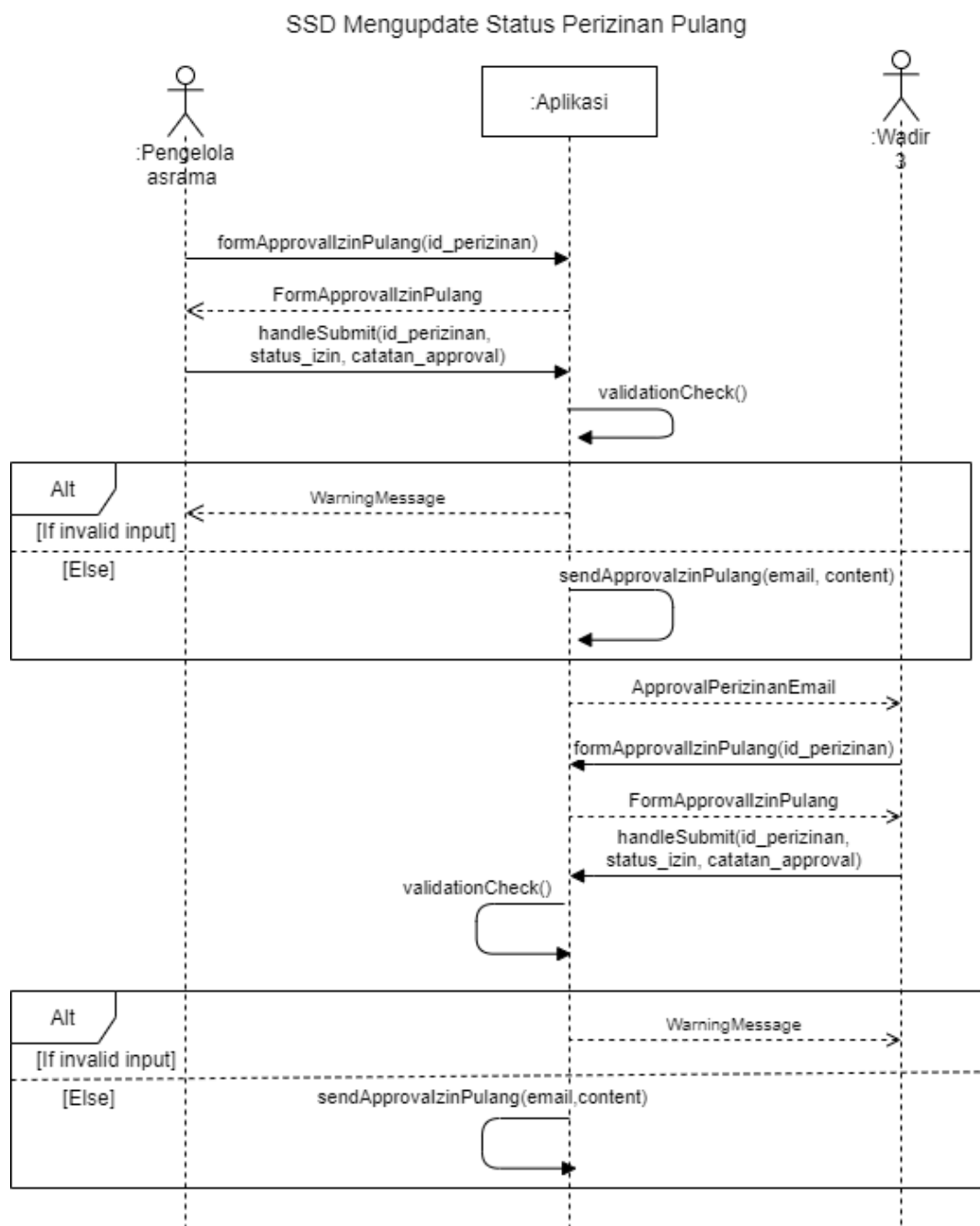
III.1.3 Use Case : Mengupdate Status Perizinan Pulang

Tabel III.10 UC-03 mengupdate status perizinan pulang

Use Case ID	UC-03
Use Case Name	Mengupdate status perizinan pulang
Scope	System under design
Level	User goal
Primary Actor	Pengelola asrama dan Wadir 3
Stakeholders and interests	Pengelola asrama dan Wadir 3 Interest: - Ingin memberikan <i>approval</i> terhadap perizinan pulang yang diajukan mahasiswa asrama.
Preconditions	1. Pengelola asrama dan Wadir 3 telah mengakses url <i>website</i> dengan benar pada perangkat melalui <i>browser</i> . 2. Pengelola asrama dan Wadir 3 telah terdaftar dan terverifikasi sebagai <i>user</i> pada aplikasi. 3. Pengelola asrama menerima <i>e-mail</i> perizinan dari mahasiswa asrama.
Success Guarantee	Data status di perizinan telah terupdate di <i>database</i> .
Main Success Scenario	1. Pengelola asrama menekan tombol yang tersedia untuk mengarah ke aplikasi pada isi <i>e-mail</i> . 2. Aplikasi menampilkan halaman <i>approval</i> perizinan. 3. Pengelola asrama mengisi <i>form approval</i> berisi status izin dan catatan <i>approval</i> . 4. Pengelola asrama mensubmit <i>form approval</i> . 5. Aplikasi melakukan validasi <i>form</i> . 6. Aplikasi mengirim hasil <i>approval</i> melalui <i>e-mail</i> kepada Wadir 3 untuk memberikan <i>approval</i> selanjutnya. 7. Mengulangi poin 1-6 untuk dilakukan oleh Wadir 3. 8. Aplikasi mengirim hasil <i>approval</i> melalui <i>e-mail</i> kepada mahasiswa asrama yang mengajukan, untuk diberitahu hasil <i>approval</i> perizinannya.
Extensions	5a. Jika <i>form</i> tidak valid (beberapa <i>field</i> kosong/data yang dientrikan salah), maka aplikasi akan menampilkan pesan peringatan.
Special Requirements	-

Technology and Data Variations List	-
Frequency of Occurrence	Sewaktu-waktu saat pengelola dan Wadir 3 memberikan <i>approval</i> perizinan pulang dari mahasiswa.
Miscellaneous	-

III.1.3.1 System Sequence Diagram : Mengupdate Status Perizinan Pulang[SSD-03]



Gambar III.3 SSD-03 mengupdate status perizinan pulang

III.1.3.2 Operation Contract : Mengupdate Status Perizinan Pulang[OC-03]

1. formApprovalIzinPulang(id_perizinan)

Tabel III.11 OC-03 form *approval* izin pulang

Operation	formApprovalIzinPulang (id_perizinan)
Cross Reference	Use Cases: Mengupdate status perizinan pulang
Precondition	<i>Instance</i> Mahasiswa sedang berjalan
Postcondition	<i>Instance</i> Perizinan telah dibuat

2. handleSubmit(id_perizinan, status_izin, catatan_approval)

Tabel III.12 OC-03 *handle submit*

Operation	handleSubmit(id_perizinan, status)
Cross Reference	Use Cases: Mengupdate status perizinan pulang
Precondition	<i>Instance</i> Mahasiswa sedang berjalan <i>Instance</i> Perizinan sedang berjalan
Postcondition	Perizinan.status_izin menjadi status_izin Perizinan.catatan_approval menjadi catatan_approval Perizinan.isComplete menjadi true

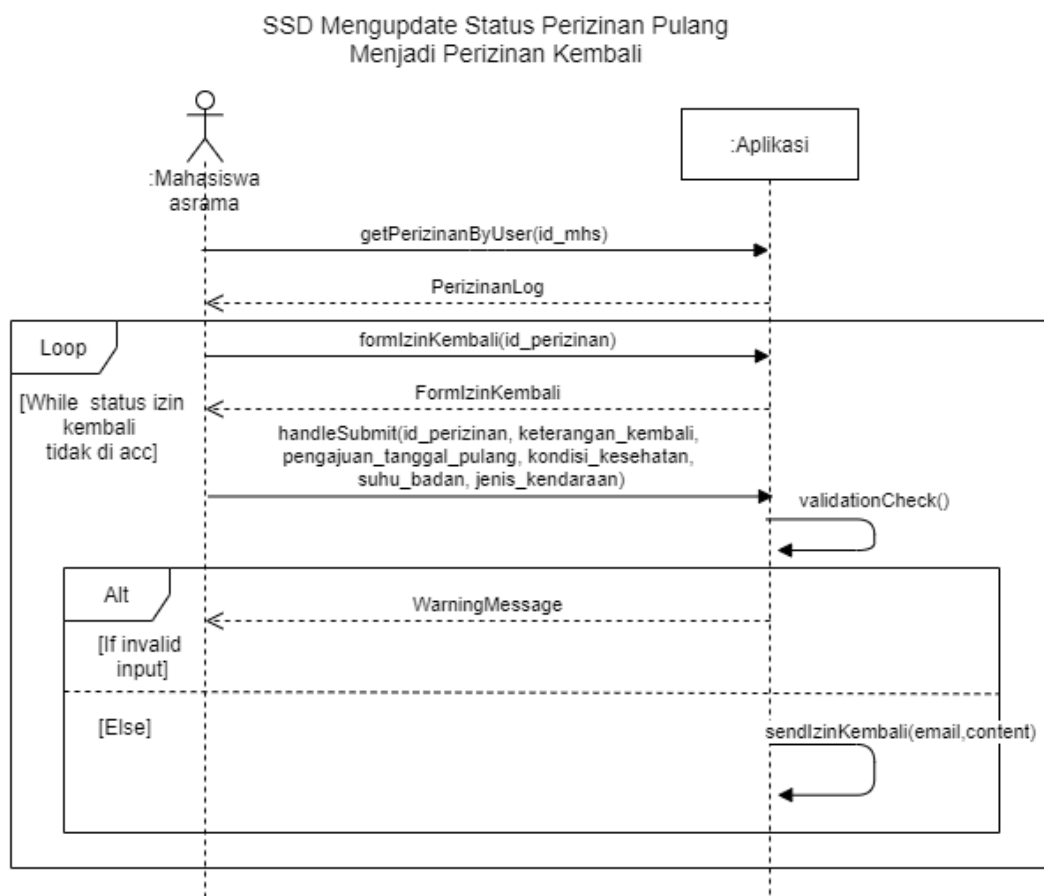
III.1.4 Use Case : Mengupdate Status Perizinan Pulang Menjadi Perizinan Kembali

Tabel III.13 UC-04 mengupdate status perizinan pulang menjadi perizinan kembali

Use Case ID	UC-04
Use Case Name	Mengupdate status perizinan pulang menjadi perizinan kembali
Scope	System under design
Level	User goal
Primary Actor	Mahasiswa asrama
Stakeholders and interests	Mahasiswa asrama Interest: - Ingin mengajukan perizinan kembali ke asrama, setelah masa izinnya.
Preconditions	1. Mahasiswa telah mengakses url <i>website</i> dengan benar pada perangkat melalui <i>browser</i> . 2. Mahasiswa telah terdaftar dan terverifikasi sebagai <i>user</i> pada aplikasi. 3. Mahasiswa menerima <i>e-mail</i> hasil <i>approval</i> perizinan pulang dari Wadir 3. 4. Status mahasiswa masih dalam perizinan pulang.
Success Guarantee	Data status di perizinan telah terupdate di <i>database</i> .
Main Success Scenario	1. Mahasiswa asrama mengakses riwayat perizinan. 2. Aplikasi menampilkan riwayat perizinan. 3. Mahasiswa asrama menekan tombol pengajuan perizinan kembali ke asrama pada izin yang diajukan. 4. Aplikasi menampilkan halaman <i>form</i> perizinan kembali ke asrama.

	5. Mahasiswa asrama mengisi seluruh isian pada <i>form</i> berisi alasan kembali, pengajuan tanggal kembali, kondisi kesehatan, suhu badan, dan jenis transportasi. 6. Mahasiswa asrama mensubmit <i>form</i> . 7. Aplikasi melakukan validasi <i>form</i> . 8. Aplikasi mengirim hasil pengajuan izin kembali melalui <i>e-mail</i> kepada pengelola asrama.
Extensions	7a. Jika <i>form</i> tidak valid (beberapa <i>field</i> kosong/data yang dientrikan salah), maka aplikasi akan menampilkan pesan peringatan. 3a. Lakukan langkah 3-8 untuk status izin kembali ke asrama yang ditolak.
Special Requirements	-
Technology and Data Variations List	-
Frequency of Occurrence	Sewaktu-waktu saat mahasiswa asrama mengajukan perizinan kembali ke asrama.
Miscellaneous	-

III.1.4.1 System Sequence Diagram : Mengupdate Status Perizinan Pulang Menjadi Perizinan Kembali [SSD-04]



Gambar III.4 SSD-04 mengupdate status perizinan pulang menjadi kembali

III.1.4.2 Operation Contract : Mengupdate Status Perizinan Pulang Menjadi Perizinan Kembali [OC-04]

1. getPerizinanByUser(id_mhs)

Tabel III.14 OC-04 *get perizinan by user*

Operation	getPerizinanByUser(id_mhs)
Cross Reference	Use Cases: Mengupdate status perizinan pulang menjadi perizinan kembali
Precondition	<i>Instance</i> Mahasiswa sedang berjalan <i>Instance</i> Perizinan sedang berjalan
Postcondition	<i>Instance</i> Perizinan berasosiasi dengan <i>instance</i> Mahasiswa berdasarkan id

2. handleSubmit(id_perizinan, keterangan_kembali, pengajuan_tanggal_pulang, kondisi_kesehatan, suhu_badan, jenis_kendaraan)

Tabel III.15 OC-04 *handle submit*

Operation	handleSubmit(id_perizinan)
Cross Reference	Use Cases: Mengupdate status perizinan pulang menjadi perizinan kembali
Precondition	<i>Instance</i> Mahasiswa sedang berjalan <i>Instance</i> Perizinan sedang berjalan
Postcondition	Perizinan.keterangan_kembali menjadi keterangan_kembali Perizinan.tanggal_pulang menjadi tanggal_pulang Perizinan.alamat_izin menjadi alamat_izin Perizinan.suhu_badan menjadi suhu_badanS Perizinan.kondisi_kesehatan menjadi kondisi_kesehatan Perizinan.jenis_kendaraan menjadi jenis_kendaraan Perizinan.isComplete menjadi true

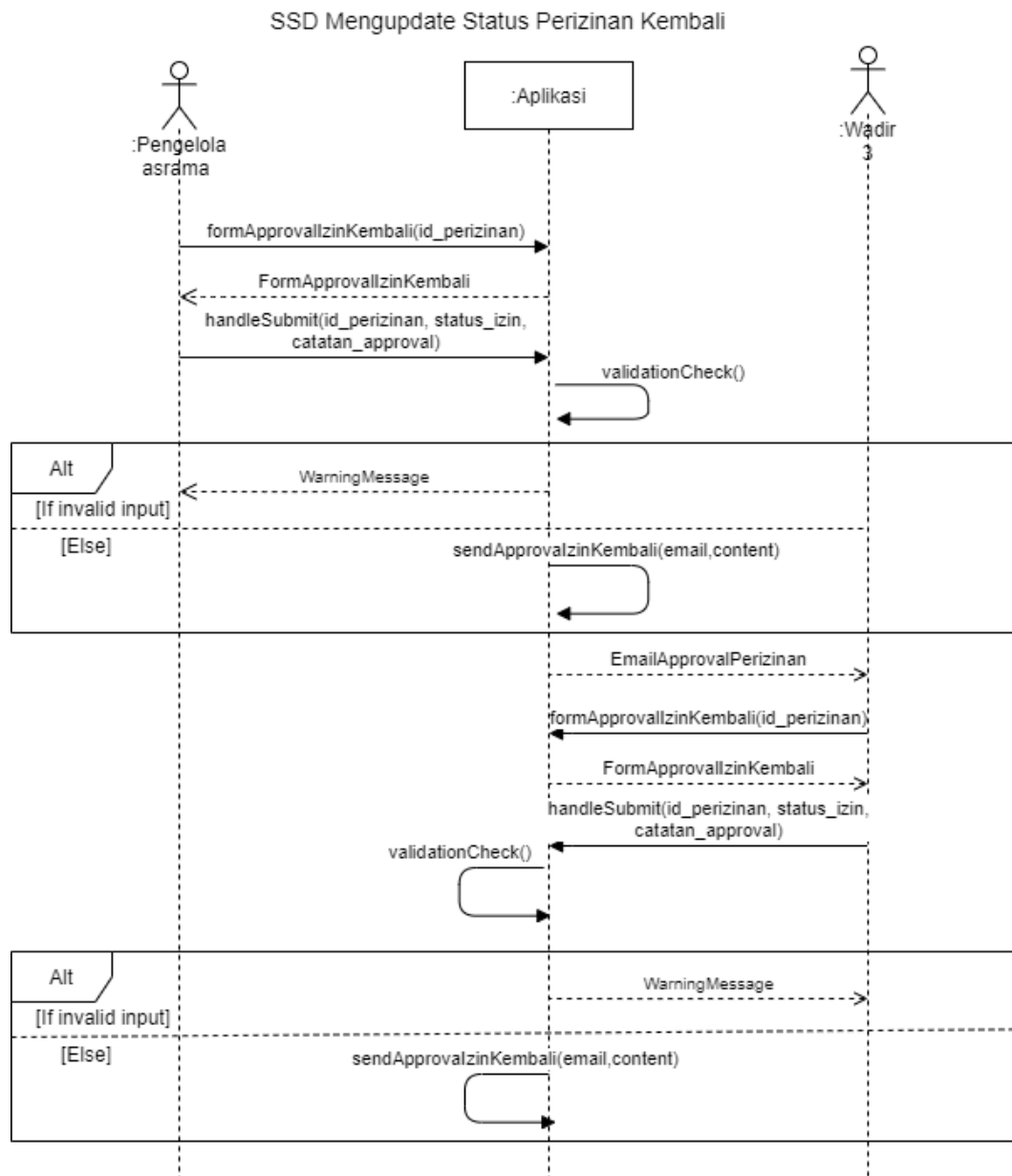
III.1.5 Use Case : Mengupdate Status Perizinan Kembali

Tabel III.16 UC-05 mengupdate status perizinan kembali

Use Case ID	UC-05
Use Case Name	Mengupdate status perizinan kembali
Scope	System under design
Level	User goal
Primary Actor	Pengelola asrama dan Wadir 3
Stakeholders and interests	Pengelola asrama dan Wadir 3 Interest: - Ingin memberikan <i>approval</i> terhadap perizinan kembali ke asrama yang diajukan mahasiswa asrama.
Preconditions	1. Pengelola asrama dan Wadir 3 telah mengakses url <i>website</i> dengan benar pada perangkat melalui <i>browser</i> . 2. Pengelola asrama dan Wadir 3 telah terdaftar dan terverifikasi sebagai <i>user</i> pada aplikasi. 3. Pengelola asrama menerima <i>e-mail</i> perizinan kembali ke asrama dari mahasiswa asrama.
Success Guarantee	Data status di perizinan telah terupdate di <i>database</i> .
Main Success Scenario	1. Pengelola menekan tombol yang tersedia untuk mengarah ke aplikasi pada isi <i>e-mail</i> .

	<ol style="list-style-type: none"> 2. Aplikasi menampilkan halaman <i>approval</i> perizinan kembali ke asrama. 3. Pengelola mengisi <i>form approval</i> berisi status izin dan catatan <i>approval</i>. 4. Pengelola mensubmit <i>form approval</i>. 5. Aplikasi melakukan validasi <i>form</i>. 6. Aplikasi mengirim hasil <i>approval</i> melalui <i>e-mail</i> kepada Wadir 3 untuk memberikan <i>approval</i> selanjutnya. 7. Mengulangi poin 1-5 yang dilakukan oleh Wadir 3, setelah selesai dilanjutkan ke poin 8. 8. Aplikasi mengirim hasil <i>approval</i> melalui <i>e-mail</i> kepada mahasiswa asrama yang mengajukan, untuk diberitahu hasil <i>approval</i> perizinannya.
Extensions	5a. Jika <i>form</i> tidak valid (beberapa <i>field</i> kosong/data yang dientrikan salah), maka aplikasi akan menampilkan pesan peringatan.
Special Requirements	-
Technology and Data Variations List	-
Frequency of Occurrence	Sewaktu-waktu saat pengelola dan Wadir 3 memberikan <i>approval</i> perizinan kembari ke asrama dari mahasiswa.
Miscellaneous	-

III.1.5.1 System Sequence Diagram : Mengupdate Status Perizinan Kembali [SSD-05]



Gambar III.5 SSD-05 mengupdate status perizinan kembali

III.1.5.2 Operation Contract : Mengupdate Status Perizinan Kembali [OC-05]

1. formApprovalIzinKembali(id_perizinan)

Tabel III.17 OC-05 form *approval* izin kembali

Operation	formApprovalIzinKembali(id_perizinan)
Cross Reference	Use Cases: Mengupdate status perizinan kembali
Precondition	<i>Instance</i> Mahasiswa sedang berjalan
Postcondition	<i>Instance</i> Perizinan telah dibuat

2. handleSubmit(id_perizinan, status_izin, catatan_approval)

Tabel III.18 OC-05 *handle submit*

Operation	handleSubmit(id_perizinan)
Cross Reference	Use Cases: Mengupdate status perizinan kembali
Precondition	<i>Instance</i> Mahasiswa sedang berjalan <i>Instance</i> Perizinan sedang berjalan
Postcondition	Perizinan.status_izin menjadi status_izin Perizinan.catatan_approval menjadi catatan_approval Perizinan.isComplete menjadi true

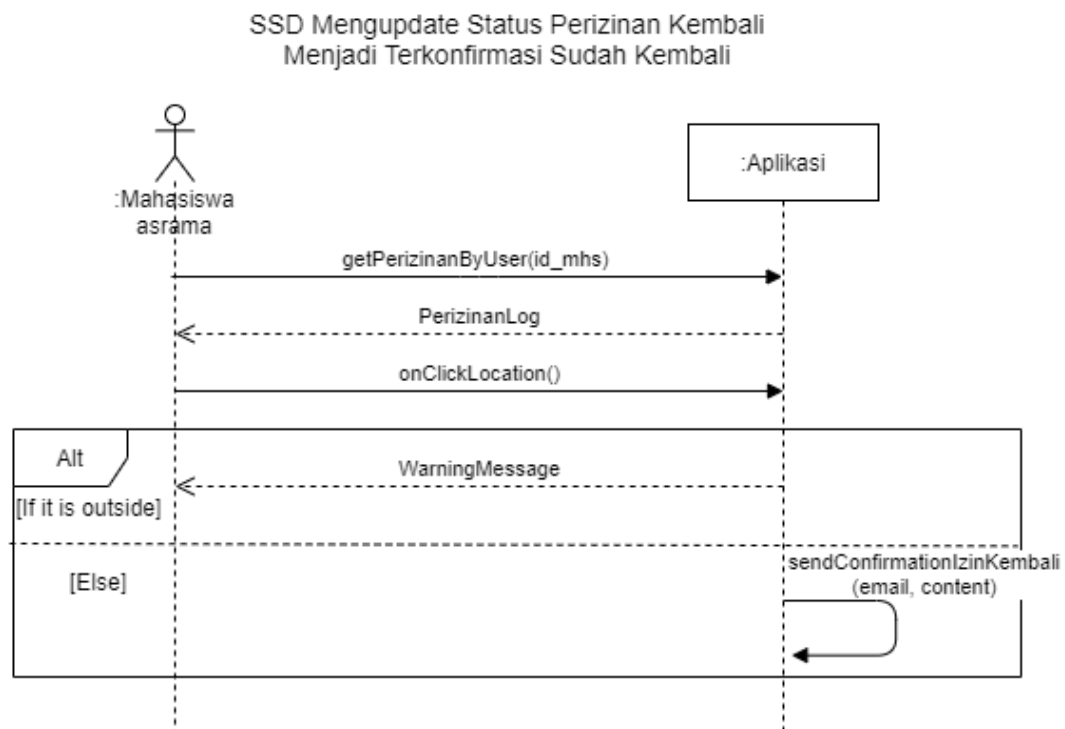
III.1.6 Use Case : Mengupdate Status Perizinan Kembali Menjadi Terkonfirmasi Sudah Kembali

Tabel III.19 UC-06 mengupdate status perizinan kembali menjadi terkonfirmasi

Use Case ID	UC-06
Use Case Name	Mengupdate status perizinan kembali menjadi terkonfirmasi sudah kembali
Scope	System under design
Level	User goal
Primary Actor	Mahasiswa asrama
Stakeholders and interests	<ol style="list-style-type: none"> 1. Mahasiswa asrama telah mengakses url <i>website</i> dengan benar pada perangkat melalui <i>browser</i>. 2. Mahasiswa asrama telah terdaftar dan terverifikasi sebagai <i>user</i> pada aplikasi. 3. Mahasiswa asrama menerima <i>e-mail</i> hasil <i>approval</i> perizinan kembali ke asrama dari Wadir 3.
Preconditions	Mahasiswa asrama Interest: - Ingin memberikan konfirmasi keberadaan bahwa mahasiswa sudah kembali ke asrama.
Success Guarantee	Data status di perizinan telah terupdate.
Main Success Scenario	<ol style="list-style-type: none"> 1. Mahasiswa asrama mengakses riwayat perizinan. 2. Aplikasi menampilkan riwayat perizinan. 3. Mahasiswa asrama menekan tombol konfirmasi sudah kembali ke asrama pada izin yang diajukan. 4. Aplikasi mengirim <i>e-mail</i> pemberitahuan kepada pengelola dan Wadir 3, bahwa mahasiswa sudah berada di asrama.

Extensions	3a. Jika mahasiswa teridentifikasi berada di luar radius asrama, maka aplikasi menampilkan pesan peringatan berada di luar asrama dan tidak bisa mengupdate status menjadi sudah berada di asrama.
Special Requirements	-
Technology and Data Variations List	GPS yang terdapat pada perangkat mahasiswa asrama.
Frequency of Occurrence	Sewaktu-waktu saat mahasiswa asrama memberikan konfirmasi dirinya sudah berada kembali di asrama.
Miscellaneous	-

III.1.6.1 System Sequence Diagram : Mengupdate Status Perizinan Kembali Menjadi Terkonfirmasi Sudah Kembali [SSD-06]



Gambar III.6 SSD-06 mengupdate status perizinan kembali menjadi terkonfirmasi

III.1.6.2 Operation Contract : Mengupdate Status Perizinan Kembali Menjadi Terkonfirmasi Sudah Kembali [OC-06]

1. `getPerizinanByUser(id_mhs)`

Tabel III.20 OC-06 *get perizinan by user*

Operation	<code>getPerizinanByUser(id_mhs)</code>
Cross Reference	Use Cases: Mengupdate status perizinan kembali menjadi terkonfirmasi sudah kembali
Precondition	<i>Instance Mahasiswa sedang berjalan</i>

	<i>Instance</i> Perizinan sedang berjalan
Postcondition	<i>Instance</i> Perizinan berasosiasi dengan <i>instance</i> Mahasiswa berdasarkan id

2. onClickLocation()

Tabel III.21 OC-06 klik lokasi

Operation	onClickLocation()
Cross Reference	Use Cases: Mengupdate status perizinan kembali menjadi terkonfirmasi sudah kembali
Precondition	<i>Instance</i> Mahasiswa sedang berjalan <i>Instance</i> Presensi sedang berjalan
Postcondition	Perizinan.status_izin menjadi status_izin Presensi.isComplete menjadi true

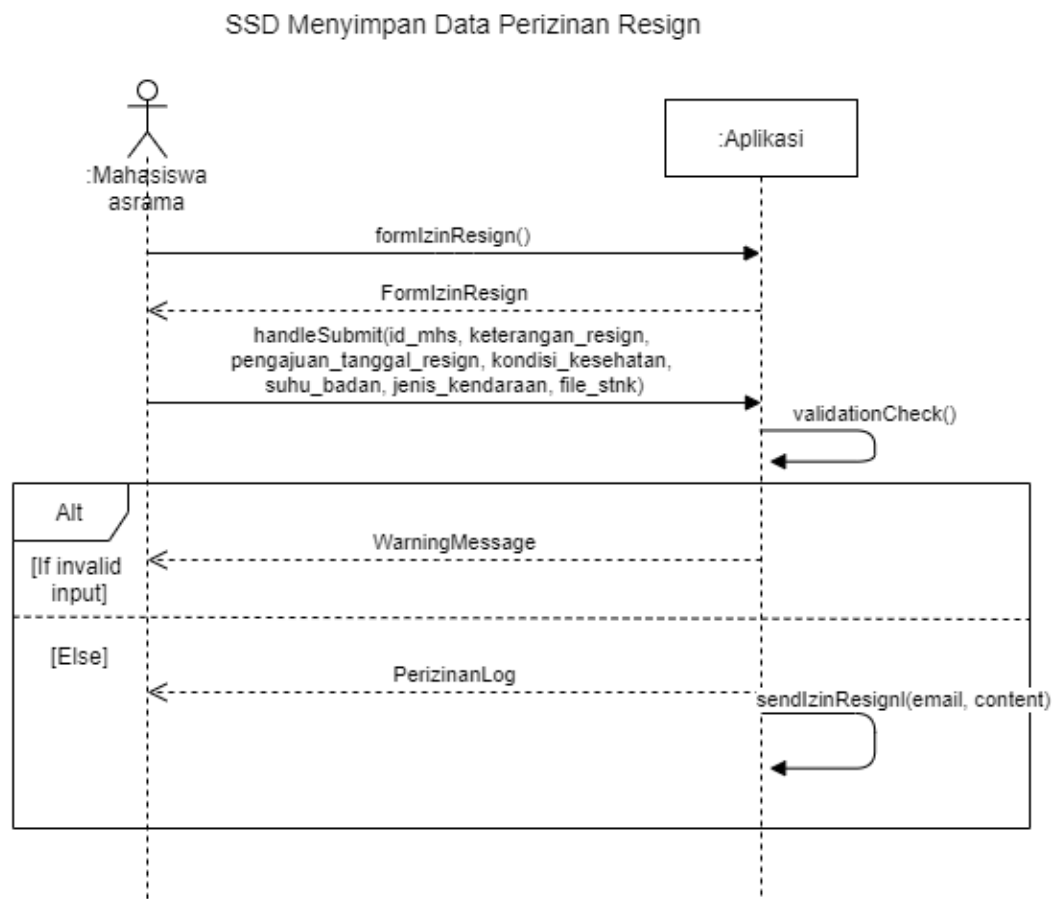
III.1.7 Use Case : Menyimpan Data Perizinan Resign

Tabel III.22 UC-07 menyimpan data perizinan *resign*

Use Case ID	UC-07
Use Case Name	Menyimpan data perizinan <i>resign</i>
Scope	System under design
Level	User goal
Primary Actor	Mahasiswa asrama
Stakeholders and interests	Mahasiswa asrama Interest: - Ingin mengajukan perizinan <i>resign</i> dari asrama.
Preconditions	1. Mahasiswa asrama telah mengakses url <i>website</i> dengan benar pada perangkat melalui <i>browser</i> . 2. Mahasiswa asrama telah terdaftar dan terverifikasi sebagai <i>user</i> pada aplikasi. 3. Aplikasi menampilkan halaman <i>dashboard</i> mahasiswa asrama.
Success Guarantee	Data perizinan <i>resign</i> tersimpan di <i>database</i> .
Main Success Scenario	1. Mahasiswa asrama mengakses halaman perizinan <i>resign</i> . 2. Aplikasi menampilkan halaman <i>form</i> perizinan <i>resign</i> . 3. Mahasiswa asrama mengisi seluruh isian pada <i>form</i> berisi keterangan <i>resign</i> , pengajuan tanggal <i>resign</i> , kondisi kesehatan, suhu badan, jenis kendaraan, dan file STNK. 4. Mahasiswa mensubmit <i>form</i> . 5. Aplikasi melakukan validasi <i>form</i> . 6. Aplikasi menyimpan hasil pengajuan <i>resign</i> ke <i>database</i> . 7. Aplikasi menampilkan riwayat perizinan. 8. Aplikasi mengirim notifikasi pengajuan <i>resign</i> melalui <i>e-mail</i> ke pengelola asrama.
Extensions	5a. Jika <i>form</i> tidak valid (beberapa <i>field</i> kosong/data yang diinputkan salah), maka aplikasi akan menampilkan pesan peringatan.
Special Requirements	-
Technology and Data Variations List	-

Frequency of Occurrence	Sewaktu-waktu saat mahasiswa mengajukan perizinan <i>resign</i> .
Miscellaneous	-

III.1.7.1 System Sequence Diagram : Menyimpan Data Perizinan Resign [SSD-07]



Gambar III.7 SSD-07 menyimpan data perizinan pulang

III.1.7.2 Operation Contract : Menyimpan Data Perizinan Resign [OC-07]

1. formIzinResign()

Tabel III.23 OC-07 form izin *resign*

Operation	formIzinResign ()
Cross Reference	Use Cases: Menyimpan data perizinan resign
Precondition	Instance Mahasiswa sedang berjalan
Postcondition	Instance Resign telah dibuat

2. `handleSubmit(id_mhs, keterangan_resign, pengajuan_tanggal_resign, kondisi_kesehatan, suhu_badan, jenis_kendaraan, file_stnk)`

Tabel III.24 OC-07 *handle submit*

Operation	<code>handleSubmit(id_mhs)</code>
Cross Reference	Use Cases: Menyimpan data perizinan resign
Precondition	<i>Instance</i> Mahasiswa sedang berjalan <i>Instance</i> Resign sedang berjalan
Postcondition	Resign.tanggal_resign menjadi tanggal_resign Resign.keterangan_resign menjadi keterangan_resign Resign.jenis_kendaraan menjadi jenis_kendaraan Resign.suhu_badan menjadi suhu_badan Resign.kondisi_kesehatan menjadi kondisi_kesehatan Resign.file_stnk menjadi file_stnk Resign.keterangan_resign menjadi resignNote

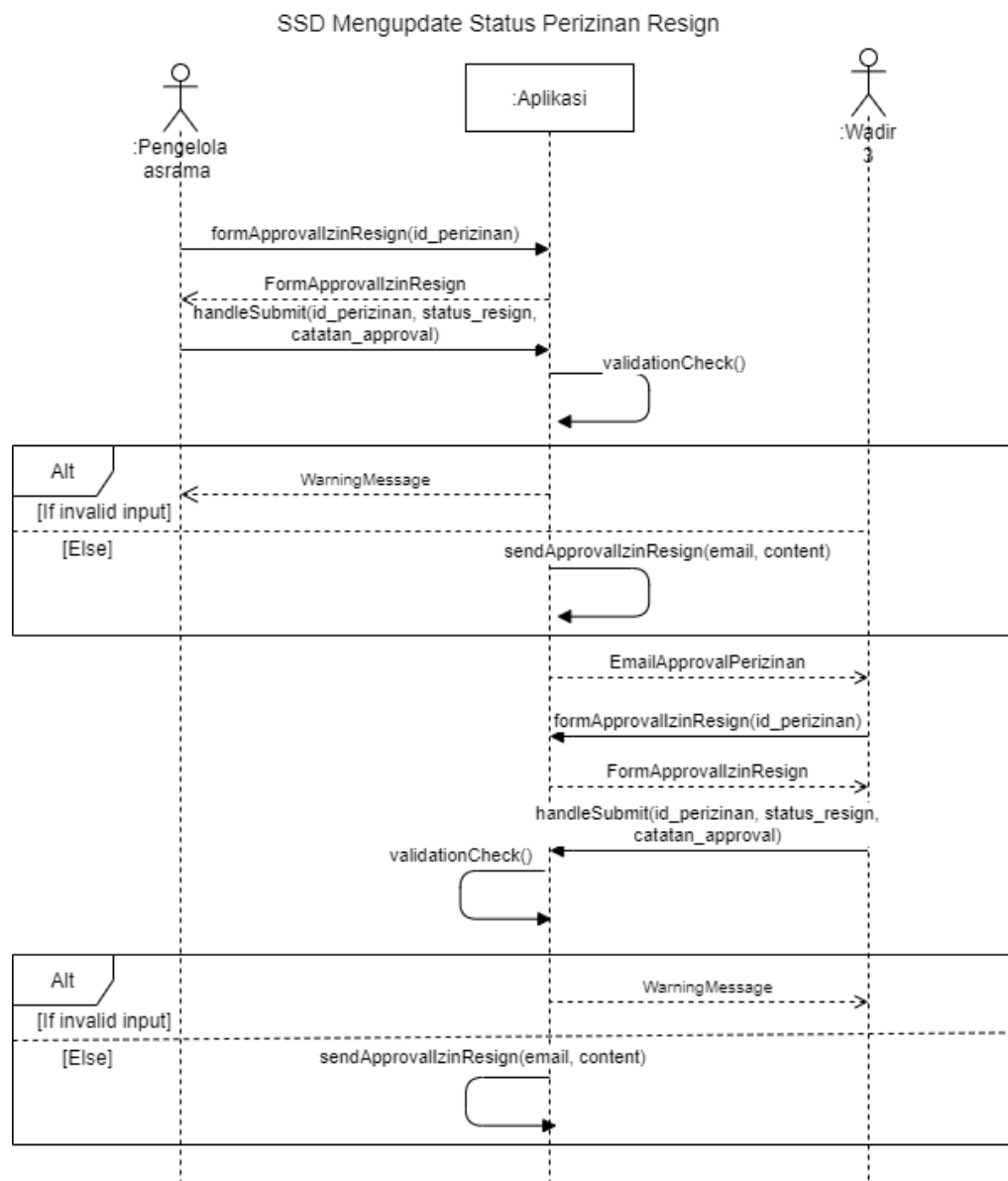
III.1.8 Use Case : Mengupdate Status Perizinan *Resign*

Tabel III.25 UC-08 mengupdate status perizinan *resign*

Use Case ID	UC-08
Use Case Name	Mengupdate status perizinan <i>resign</i>
Scope	System under design
Level	User goal
Primary Actor	Pengelola dan Wadir 3
Stakeholders and interests	Pengelola asrama dan Wadir 3 Interest: - Ingin memberikan <i>approval</i> terhadap perizinan <i>resign</i> yang diajukan mahasiswa asrama.
Preconditions	<ol style="list-style-type: none"> 1. Pengelola asrama dan Wadir 3 telah mengakses url <i>website</i> dengan benar pada perangkat melalui <i>browser</i>. 2. Pengelola asrama dan Wadir 3 telah terdaftar dan terverifikasi sebagai <i>user</i> pada aplikasi. 3. Pengelola asrama menerima <i>e-mail</i> perizinan dari mahasiswa asrama.
Success Guarantee	Data status di perizinan <i>resign</i> telah terupdate di <i>database</i> .
Main Success Scenario	<ol style="list-style-type: none"> 1. Pengelola menekan tombol yang tersedia untuk mengarah ke aplikasi pada isi <i>e-mail</i>. 2. Aplikasi menampilkan halaman <i>approval</i> perizinan <i>resign</i>. 3. Pengelola mengisi <i>form approval</i> berisi status izin dan catatan <i>approval</i>. 4. Pengelola mensubmit <i>form approval</i>. 5. Aplikasi melakukan validasi <i>form</i>. 6. Aplikasi mengirim hasil <i>approval</i> melalui <i>e-mail</i> kepada Wadir 3 untuk memberikan <i>approval</i> selanjutnya. 7. Mengulangi poin 1-6 untuk dilakukan oleh Wadir 3. 8. Aplikasi mengirim hasil <i>approval</i> melalui <i>e-mail</i> kepada mahasiswa asrama yang mengajukan <i>resign</i>, untuk diberitahu hasil <i>approval</i> perizinannya.

Extensions	5a. Jika <i>form</i> tidak valid (beberapa <i>field</i> kosong/data yang dientrikan salah), maka aplikasi akan menampilkan pesan peringatan.
Special Requirements	-
Technology and Data Variations List	-
Frequency of Occurrence	Sewaktu-waktu saat pengelola dan Wadir 3 memberikan <i>approval</i> perizinan <i>resign</i> dari mahasiswa.
Miscellaneous	-

III.1.8.1 System Sequence Diagram : Mengupdate Status Perizinan Resign [SSD-08]



Gambar III.8 SSD-08 mengupdate status perizinan *resign*

III.1.8.2 Operation Contract : Mengupdate Status Perizinan Resign [OC-08]

1. formApprovaIzinResign(id_perizinan)

Tabel III.26 OC-08 form approval izin resign

Operation	formApprovaIzinResign (id_perizinan)
Cross Reference	Use Cases: Mengupdate status perizinan resign
Precondition	<i>Instance</i> Mahasiswa sedang berjalan <i>Instance</i> Resign sedang berjalan
Postcondition	<i>Instance</i> Resign berasosiasi dengan instance Mahasiswa berdasarkan id

2. handleSubmit(id_perizinan, , status_resign, catatan_approval)

Tabel III.27 OC-08 handle submit

Operation	handleSubmit(id_perizinan, status)
Cross Reference	Use Cases: Mengupdate status perizinan resign
Precondition	<i>Instance</i> Mahasiswa sedang berjalan <i>Instance</i> Resign sedang berjalan
Postcondition	Resign.status_resign menjadi status_resign Resign.catatan_approval menjadi catatan_approval Resign.isComplete menjadi true

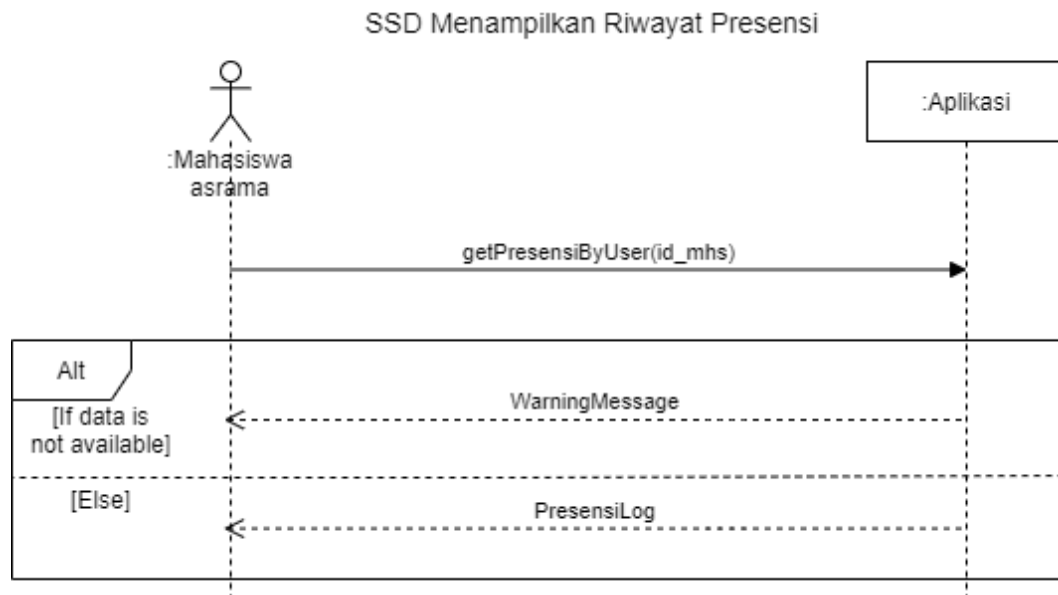
III.1.9 Use Case : Menampilkan Riwayat Presensi

Tabel III.28 UC-09 menampilkan riwayat presensi

Use Case ID	UC-09
Use Case Name	Menampilkan riwayat presensi
Scope	System under design
Level	User goal
Primary Actor	Mahasiswa asrama
Stakeholders and interests	Mahasiswa asrama Interest: - Ingin mengakses data riwayat presensi pribadi.
Preconditions	1. Mahasiswa asrama telah mengakses url <i>website</i> dengan benar pada perangkat melalui <i>browser</i> . 2. Mahasiswa asrama telah terdaftar dan terverifikasi sebagai <i>user</i> pada aplikasi. 3. Aplikasi menampilkan halaman <i>dashboard</i> mahasiswa.
Success Guarantee	Data riwayat presensi berhasil ditampilkan.
Main Success Scenario	1. Mahasiswa mengakses halaman riwayat presensi. 2. Aplikasi menampilkan halaman berisi data riwayat presensi kehadiran.
Extensions	2.a Jika data riwayat presensi tidak ditemukan, aplikasi menampilkan pesan peringatan.
Special Requirements	-
Technology and Data Variations List	-
Frequency of Occurrence	Saat mahasiswa asrama mengakses langsung halaman riwayat presensi.

Miscellaneous	-
---------------	---

III.1.9.1 System Sequence Diagram : Menampilkan Riwayat Presensi [SSD-09]



Gambar III.9 SSD-09 menampilkan riwayat presensi

III.1.9.2 Operation Contract : Menampilkan Riwayat Presensi [OC-09]

1. getPresensiByUser(id_mhs)

Tabel III.29 OC-09 *get presensi by user*

Operation	getPresensiByUser(id_mhs)
Cross Reference	Use Cases: Menampilkan riwayat presensi
Precondition	<i>Instance</i> Mahasiswa sedang berjalan <i>Instance</i> Presensi sedang berjalan
Postcondition	<i>Instance</i> Presensi berasosiasi dengan <i>instance</i> Mahasiswa berdasarkan id

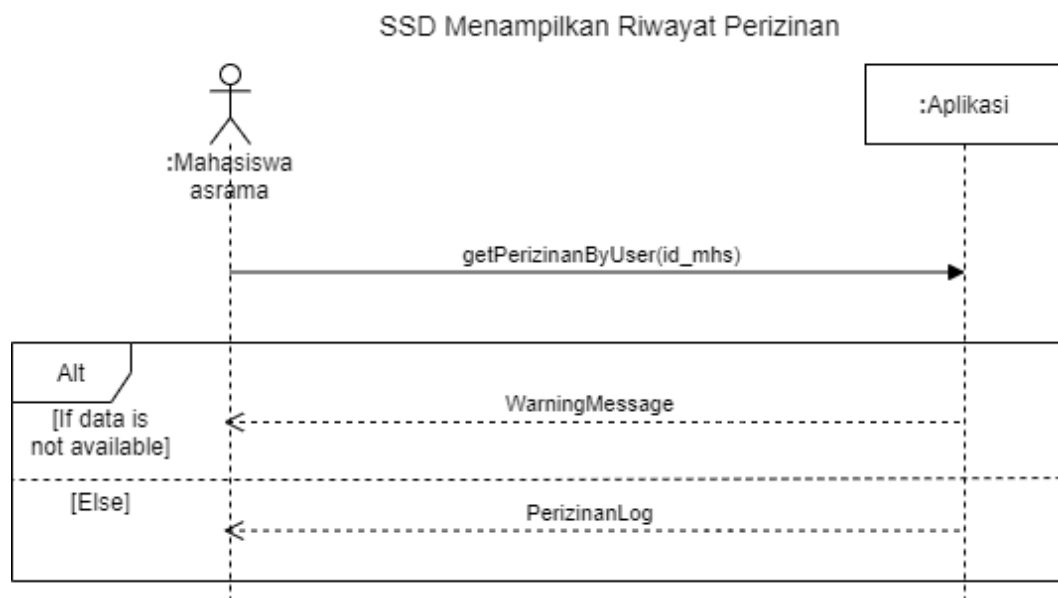
III.1.10 Use Case : Menampilkan Riwayat Perizinan

Tabel III.30 UC-10 menampilkan riwayat presensi

Use Case ID	UC-10
Use Case Name	Menampilkan riwayat perizinan
Scope	System under design
Level	User goal
Primary Actor	Mahasiswa asrama
Stakeholders and interests	Mahasiswa asrama Interest: - Ingin mengakses data riwayat perizinan pribadi.
Preconditions	1. Mahasiswa asrama telah mengakses url <i>website</i> dengan benar pada perangkat melalui <i>browser</i> .

	2. Mahasiswa asrama telah terdaftar dan terverifikasi sebagai <i>user</i> pada aplikasi. 3. Aplikasi menampilkan halaman <i>dashboard</i> mahasiswa.
Success Guarantee	Data riwayat perizinan berhasil ditampilkan.
Main Success Scenario	1. Mahasiswa mengakses halaman riwayat perizinan. 2. Aplikasi menampilkan halaman berisi data riwayat perizinan.
Extensions	2.a Jika data riwayat perizinan tidak ditemukan, aplikasi menampilkan pesan peringatan.
Special Requirements	-
Technology and Data Variations List	-
Frequency of Occurrence	Saat mahasiswa asrama mengakses langsung halaman riwayat perizinan.
Miscellaneous	-

III.1.10.1 System Sequence Diagram : Menampilkan Riwayat Perizinan [SSD-10]



Gambar III.10 SSD-10 menampilkan riwayat perizinan

III.1.10.2 Operation Contract : Menampilkan Riwayat Perizinan [OC-10]

1. `getPerizinanByUser(id_mhs)`

Tabel III.31 OC-10 *get* presensi *by user*

Operation	<code>getPerizinanByUser(id_mhs)</code>
Cross Reference	Use Cases: Menampilkan riwayat perizinan
Precondition	<i>Instance</i> Mahasiswa sedang berjalan <i>Instance</i> Perizinan sedang berjalan
Postcondition	<i>Instance</i> Perizinan berasosiasi dengan <i>instance</i> Mahasiswa berdasarkan id

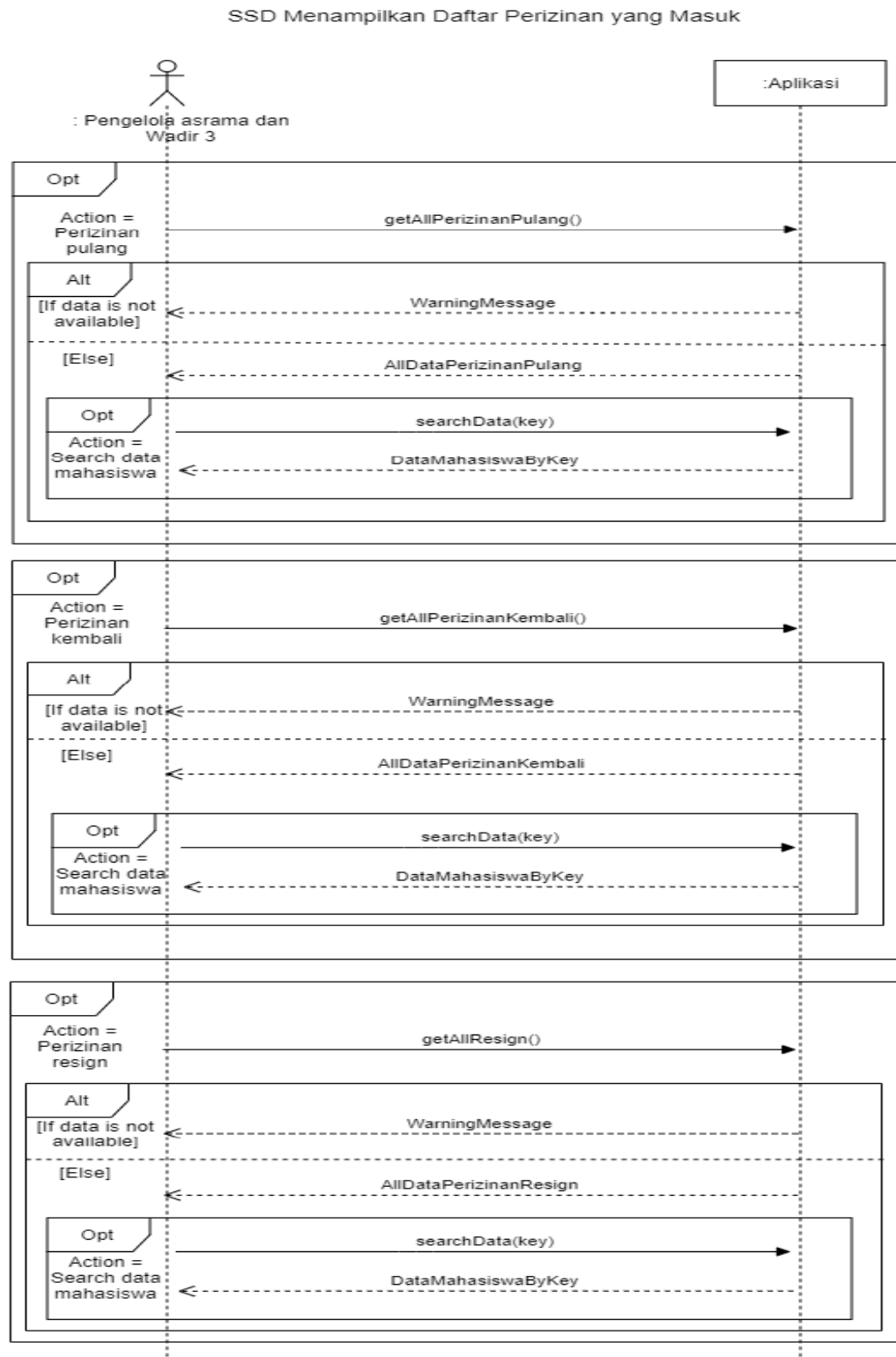
III.1.11 Use Case : Menampilkan Daftar Perizinan yang Masuk

Tabel III.32 UC-11 menampilkan daftar perizinan yang masuk

Use Case ID	UC-11
Use Case Name	Menampilkan daftar perizinan yang masuk
Scope	System under design
Level	User goal
Primary Actor	Pengelola asrama dan Wadir 3
Stakeholders and interests	Pengelola asrama dan Wadir 3 Interest: - Ingin mengakses daftar perizinan yang masuk.
Preconditions	<ol style="list-style-type: none"> 1. Pengelola asrama atau Wadir 3 telah mengakses url <i>website</i> dengan benar pada perangkat melalui <i>browser</i>. 2. Pengelola asrama atau Wadir 3 telah terdaftar dan terverifikasi sebagai <i>user</i> pada aplikasi. 3. Aplikasi menampilkan halaman <i>dashboard</i>.
Success Guarantee	Daftar perizinan yang masuk berhasil ditampilkan.
Main Success Scenario	<ol style="list-style-type: none"> 1. Pengelola atau Wadir 3 mengakses halaman daftar perizinan. 2. Aplikasi menampilkan tabel daftar perizinan.
Extensions	<p>1a. Jenis perizinan pulang.</p> <p>1a.1 Pengelola atau Wadir 3 mengakses halaman daftar perizinan pulang.</p> <p>2a. Aplikasi menampilkan tabel daftar perizinan pulang.</p> <p>2a.1 Jika data perizinan tidak ada, aplikasi menampilkan pesan peringatan.</p> <p>2a.2 Pengelola atau Wadir 3 melakukan pencarian data mahasiswa sesuai kata kunci yang diinputkan (nama).</p> <p>2a.3 Aplikasi menampilkan data hasil pencarian.</p>

Extensions	<p>1b. Jenis perizinan kembali ke asrama.</p> <p>1b.1 Pengelola atau Wadir 3 mengakses halaman daftar perizinan kembali ke asrama.</p> <p>2b. Aplikasi menampilkan tabel daftar perizinan kembali ke asrama.</p> <p>2b.1 Jika data perizinan tidak ada, aplikasi menampilkan pesan peringatan.</p> <p>2b.2 Pengelola atau Wadir 3 melakukan pencarian data mahasiswa sesuai kata kunci yang diinputkan (nama).</p> <p>2b.3 Aplikasi menampilkan data hasil pencarian.</p> <p>1c. Jenis perizinan <i>resign</i>.</p> <p>1c.1 Pengelola atau Wadir 3 mengakses halaman daftar perizinan <i>resign</i>.</p> <p>2c. Aplikasi menampilkan tabel daftar perizinan <i>resign</i>.</p> <p>2c.1 Jika data perizinan tidak ada, aplikasi menampilkan pesan peringatan.</p> <p>2c.2 Pengelola atau Wadir 3 melakukan pencarian data mahasiswa sesuai kata kunci yang diinputkan (nama).</p> <p>2c.3 Aplikasi menampilkan data hasil pencarian.</p>
Special Requirements	-
Technology and Data Variations List	-
Frequency of Occurrence	Saat pengelola asrama atau Wadir 3 mengakses daftar perizinan mahasiswa yang masuk.
Miscellaneous	-

III.1.11.1 System Sequence Diagram : Menampilkan Daftar Perizinan yang Masuk [SSD-11]



Gambar III.11 SSD-11 menampilkan daftar perizinan yang masuk

III.1.11.2 *Operation Contract* : Menampilkan Daftar Perizinan yang Masuk [OC-11]

1. getAllPerizinanPulang()

Tabel III.33 OC-11 *get all* perizinan pulang

Operation	getAllPerizinanPulang()
Cross Reference	Use Cases: Menampilkan daftar perizinan yang masuk
Precondition	Tidak ada
Postcondition	<i>Instance</i> Perizinan telah dibuat

2. getAllPerizinanKembali()

Tabel III.34 OC-11 *get all* perizinan kembali

Operation	getAllPerizinanKembali()
Cross Reference	Use Cases: Menampilkan daftar perizinan yang masuk
Precondition	Tidak ada
Postcondition	<i>Instance</i> Perizinan telah dibuat

3. getAllPerizinanResign()

Tabel III.35 OC-11 *get all* perizinan resign

Operation	getAllPerizinanResign()
Cross Reference	Use Cases: Menampilkan daftar perizinan yang masuk
Precondition	Tidak ada
Postcondition	<i>Instance</i> Resign telah dibuat

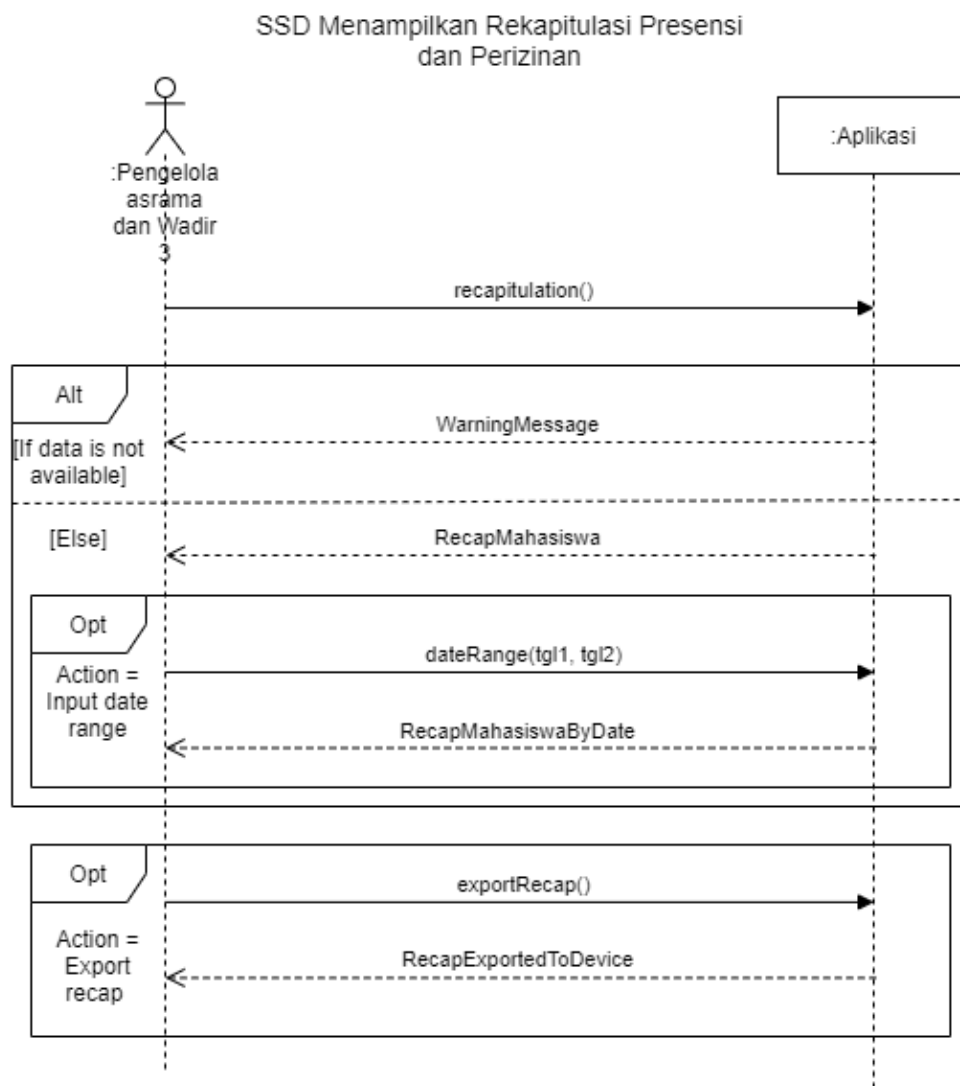
III.1.12 *Use Case* : Menampilkan Rekapitulasi Presensi dan Perizinan

Tabel III.36 UC-12 menampilkan rekapitulasi presensi dan perizinan

Use Case ID	UC-12
Use Case Name	Menampilkan rekapitulasi presensi dan perizinan
Scope	System under design
Level	User goal
Primary Actor	Pengelola asrama dan Wadir 3
Stakeholders and interests	Pengelola asrama dan Wadir 3 Interest: - Ingin mengakses rekapitulasi data presensi dan perizinan mahasiswa asrama.
Preconditions	1. Pengelola asrama atau Wadir 3 telah mengakses url <i>website</i> dengan benar pada perangkat melalui <i>browser</i> . 2. Pengelola asrama atau Wadir 3 telah terdaftar dan terverifikasi sebagai <i>user</i> pada aplikasi. 3. Aplikasi menampilkan halaman <i>dashboard</i> .
Success Guarantee	Rekapitulasi data berhasil ditampilkan.
Main Success Scenario	1. Pengelola asrama atau Wadir 3 mengakses halaman rekapitulasi data. 2. Aplikasi menampilkan halaman rekapitulasi data.

Extensions	2.a Jika data rekapitulasi tidak ada, aplikasi menampilkan pesan peringatan. 2b. Pengelola atau Wadir 3 menginput rentang tanggal. 2b.1 Aplikasi menampilkan rekap data sesuai rentang tanggal yang diinputkan. 2c. Pengelola atau Wadir 3 melakukan unduh file rekap 2c.1 Aplikasi mengunduh file ke perangkat pengguna
Special Requirements	-
Technology and Data Variations List	-
Frequency of Occurrence	Saat pengelola asrama atau Wadir 3 mengakses rekapitulasi data.
Miscellaneous	-

III.1.12.1 System Sequence Diagram : Menampilkan Rekapitulasi Presensi dan Perizinan [SSD-12]



Gambar III.12 SSD-12 menampilkan rekapitulasi presensi dan perizinan

III.1.12.2 Operation Contract : Menampilkan Rekapitulasi Presensi dan Perizinan [OC-12]

1. recapitulation()

Tabel III.37 OC-12 rekapitulasi

Operation	recapitulation()
Cross Reference	Use Cases: Menampilkan rekapitulasi presensi dan perizinan
Precondition	Tidak ada
Postcondition	<i>Instance</i> Presensi telah dibuat <i>Instance</i> Perizinan telah dibuat

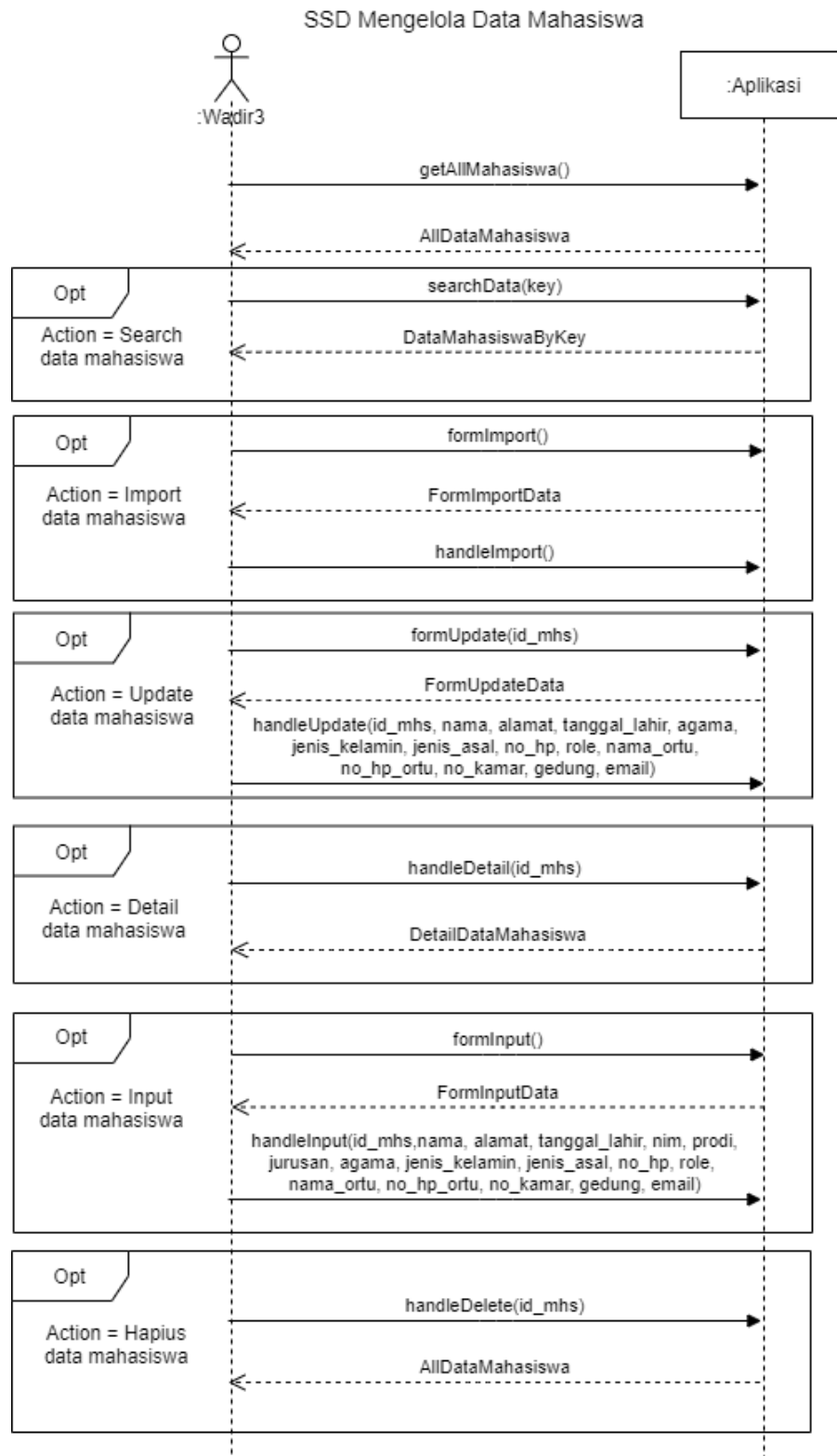
III.1.13 Use Case : Mengelola Data Mahasiswa

Tabel III.38 UC-13 mengelola data mahasiswa

Use Case ID	UC-13
Use Case Name	Mengelola data mahasiswa
Scope	System under design
Level	User goal
Primary Actor	Wadir 3
Stakeholders and interests	Wadir 3 Interest: - Ingin mengelola data mahasiswa (<i>read</i> , <i>insert</i> , <i>import</i> , dan <i>update</i>).
Preconditions	1. Wadir 3 telah mengakses url <i>website</i> dengan benar pada perangkat melalui <i>browser</i> . 2. Wadir 3 telah terdaftar dan terverifikasi sebagai <i>user</i> pada aplikasi. 3. Aplikasi menampilkan halaman <i>dashboard</i> .
Success Guarantee	Kelola data berhasil dilakukan (<i>read</i> , <i>insert</i> , <i>import</i> , dan <i>update</i>)
Main Success Scenario	1. Wadir 3 mengakses halaman kelola data mahasiswa. 2. Aplikasi menampilkan tabel data mahasiswa untuk dikelola. 3. Wadir 3 melakukan aksi untuk kelola data.
Extensions	2a. Wadir 3 melakukan pencarian data mahasiswa 2a.1 Aplikasi menampilkan data hasil pencarian 3a. Wadir 3 memilih <i>import</i> data mahasiswa. 3a.1 Aplikasi menampilkan <i>form import</i> data. 3a.2 Wadir 3 mengunggah <i>file import</i> . 3a.3 Aplikasi meng <i>import file</i> . 3b. Wadir 3 memilih mengubah data mahasiswa. 3b.1 Aplikasi menampilkan <i>form update</i> data. 3b.2 Wadir 3 mengubah data pada <i>form</i> berisi nama, alamat, tanggal lahir, agama, jenis kelamin, jenis asal, No HP, <i>role</i> , nama orang tua, no HP orang tua, no kamar, gedung, <i>e-mail</i> . 3b.3 Aplikasi mengupdate data yang diubah. 3c. Wadir 3 memilih lihat detail salah satu mahasiswa. 3c.1 Aplikasi menampilkan halaman detail mahasiswa yang dipilih. 3d. Wadir 3 memilih tambah data mahasiswa.

	<p>3d.1 Aplikasi menampilkan halaman <i>form input</i> data.</p> <p>3d.2 Wadir 3 mengisi <i>form</i> berisi nama, alamat, tanggal lahir, NIM, prodi, jurusan, agama, jenis kelamin, jenis asal, No HP, <i>role</i>, nama orang tua, no HP orang tua, no kamar, gedung, <i>e-mail</i>.</p> <p>3d.3 Aplikasi menyimpan data yang ditambahkan.</p> <p>3e. Wadir 3 menekan tombol hapus data mahasiswa pada data mahasiswa yang sudah dipilih.</p> <p>3e.1 Aplikasi menghapus data mahasiswa.</p>
Special Requirements	-
Technology and Data Variations List	-
Frequency of Occurrence	Sewaktu-waktu saat Wadir 3 melakukan kelola data mahasiswa.
Miscellaneous	-

III.1.13.1 System Sequence Diagram : Mengelola Data Mahasiswa [SSD-13]



Gambar III.13 SSD-13 mengelola data mahasiswa

III.1.13.2 Operation Contract : Mengelola Data Mahasiswa [OC-13]

1. getAllMahasiswa()

Tabel III.39 OC-13 *get all* mahasiswa

Operation	getAllMahasiswa()
Cross Reference	Use Cases: Mengelola data mahasiswa
Precondition	Tidak ada
Postcondition	<i>Instance</i> Mahasiswa telah dibuat

2. formImport()

Tabel III.40 OC-13 form *import*

Operation	formImport()
Cross Reference	Use Cases: Mengelola data mahasiswa
Precondition	<i>Instance</i> Mahasiswa sedang berjalan
Postcondition	<i>Instance</i> User telah dibuat

3. handleImport()

Tabel III.41 OC-13 *handle import*

Operation	handleImport()
Cross Reference	Use Cases: Mengelola data mahasiswa
Precondition	<i>Instance</i> Mahasiswa sedang berjalan <i>Instance</i> User sedang berjalan
Postcondition	Mahasiswa.nama_mhs menjadi nama_mhs Mahasiswa.nim_mhs menjadi nim_mhs Mahasiswa.jenis_kelamin menjadi jenis_kelamin Mahasiswa.tanggal_lahir menjadi tanggal_lahir Mahasiswa.agama menjadi agama Mahasiswa.alamat menjadi alamat Mahasiswa.no_hp_mhs menjadi no_hp_mhs Mahasiswa.nama_ortu menjadi nama_ortu Mahasiswa.no_hp_ortu menjadi no_hp_ortu Mahasiswa.status_keaktifan menjadi status_keaktifan Mahasiswa.role menjadi role User.email menjadi email User.password menjadi password

4. formUpdate(id_mhs)

Tabel III.42 OC-13 form *update*

Operation	formUpdate(id_mhs)
Cross Reference	Use Cases: Mengelola data mahasiswa
Precondition	<i>Instance</i> Mahasiswa sedang berjalan
Postcondition	<i>Instance</i> User telah dibuat

5. handleUpdate(id_mhs)

Tabel III.43 OC-13 *handle update*

Operation	handleUpdate(id_mhs)
Cross Reference	Use Cases: Mengelola data mahasiswa
Precondition	<i>Instance</i> Mahasiswa sedang berjalan <i>Instance</i> User sedang berjalan
Postcondition	Mahasiswa.nama_mhs menjadi nama_mhs Mahasiswa.nim_mhs menjadi nim_mhs Mahasiswa.jenis_kelamin menjadi jenis_kelamin Mahasiswa.tanggal_lahir menjadi tanggal_lahir Mahasiswa.agama menjadi agama Mahasiswa.alamat menjadi alamat Mahasiswa.no_hp_mhs menjadi no_hp_mhs Mahasiswa.nama_ortu menjadi nama_ortu Mahasiswa.no_hp_ortu menjadi no_hp_ortu Mahasiswa.status_keaktifan menjadi status_keaktifan User.email menjadi email User.password menjadi password Mahasiswa.role menjadi role

6. handleDetail(id_mhs)

Tabel III.44 OC-13 *handle detail*

Operation	handleDetail(id_mhs)
Cross Reference	Use Cases: Mengelola data mahasiswa
Precondition	<i>Instance</i> Mahasiswa sedang berjalan
Postcondition	<i>Instance</i> User telah dibuat

7. formInput()

Tabel III.45 OC-13 *form input*

Operation	formInput()
Cross Reference	Use Cases: Mengelola data mahasiswa
Precondition	<i>Instance</i> Mahasiswa sedang berjalan
Postcondition	<i>Instance</i> User telah dibuat

8. handleInput(id_mhs)

Tabel III.46 OC-13 *handle input*

Operation	handleInput(id_mhs, nama, alamat, tanggal_lahir, agama, jenis_kelamin, jenis_asal, no_hp, role, nama_ortu, no_hp_ortu, no_kamar, gedung, email)
Cross Reference	Use Cases: Mengelola data mahasiswa
Precondition	<i>Instance</i> Mahasiswa sedang berjalan <i>Instance</i> User sedang berjalan
Postcondition	Mahasiswa.nama_mhs menjadi nama_mhs Mahasiswa.nim_mhs menjadi nim_mhs Mahasiswa.jenis_kelamin menjadi jenis_kelamin

	Mahasiswa.tanggal_lahir menjadi tanggal_lahir Mahasiswa.agama menjadi agama Mahasiswa.alamat menjadi alamat Mahasiswa.no_hp_mhs menjadi no_hp_mhs Mahasiswa.nama_ortu menjadi nama_ortu Mahasiswa.no_hp_ortu menjadi no_hp_ortu Mahasiswa.status_keaktifan menjadi status_keaktifan Mahasiswa.role menjadi role User.email menjadi email User.password menjadi password
--	--

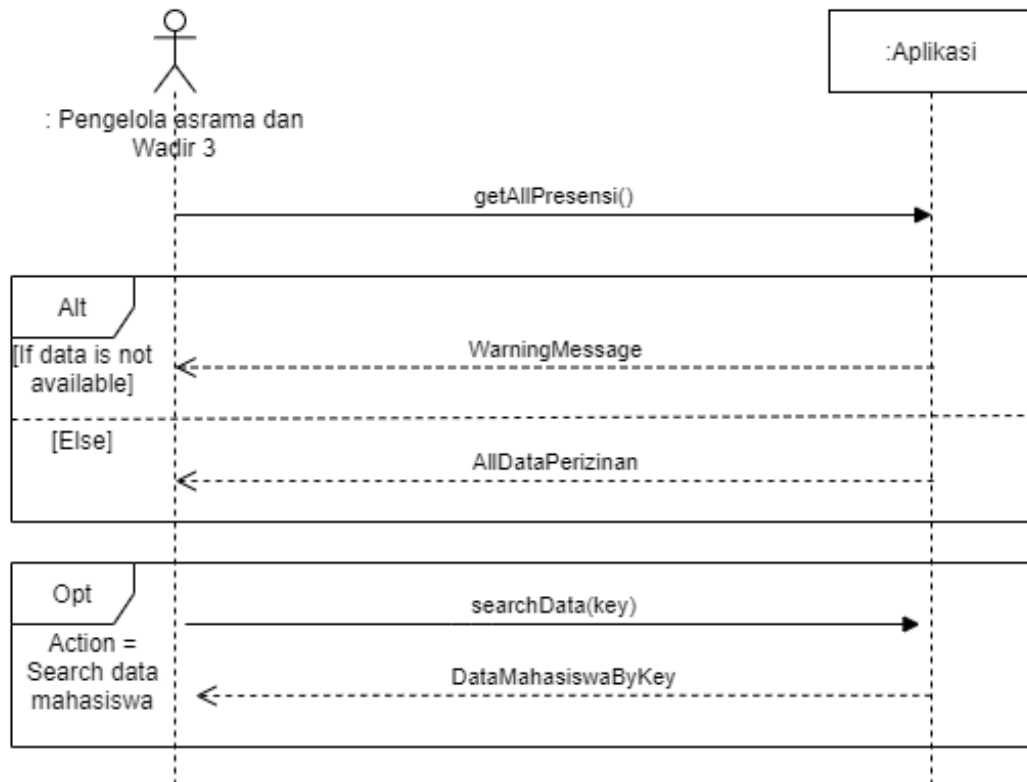
III.1.14 Use Case : Menampilkan Daftar Presensi yang Masuk

Tabel III.47 UC-14 menampilkan daftar presensi yang masuk

Use Case ID	UC-14
Use Case Name	Menampilkan daftar presensi yang masuk
Scope	System under design
Level	User goal
Primary Actor	Pengelola asrama dan Wadir 3
Stakeholders and interests	Pengelola asrama dan Wadir 3 Interest: - Ingin mengakses daftar presensi yang masuk.
Preconditions	1. Pengelola asrama atau Wadir 3 telah mengakses url <i>website</i> dengan benar pada perangkat melalui <i>browser</i> . 2. Pengelola asrama atau Wadir 3 telah terdaftar dan terverifikasi sebagai <i>user</i> pada aplikasi. 3. Aplikasi menampilkan halaman <i>dashboard</i> .
Success Guarantee	Daftar perizinan yang masuk berhasil ditampilkan.
Main Success Scenario	1. Pengelola atau Wadir 3 mengakses halaman daftar presensi. 2. Aplikasi menampilkan tabel daftar presensi yang masuk.
Extensions	2.a Jika data presensi tidak ada, aplikasi menampilkan pesan peringatan. 2b. Pengelola atau Wadir 3 melakukan pencarian data mahasiswa 2b.1 Aplikasi menampilkan data hasil pencarian -
Special Requirements	-
Technology and Data Variations List	-
Frequency of Occurrence	Saat pengelola asrama atau Wadir 3 mengakses daftar presensi mahasiswa yang masuk.
Miscellaneous	-

III.1.14.1 *System Sequence Diagram* : Menampilkan Daftar Presensi yang Masuk [SSD-14]

SSD Menampilkan Daftar Presensi yang Masuk



Gambar III.14 SSD-14 menampilkan daftar presensi yang masuk

III.1.14.2 *Operation Contract* : Menampilkan Daftar Presensi yang Masuk [OC-14]

1. getAllPresensi()

Tabel III.48 OC-14 *get all* presensi

Operation	getAllPresensi()
Cross Reference	Use Cases: Menampilkan daftar presensi yang masuk
Precondition	Tidak ada
Postcondition	<i>Instance</i> Presensi telah dibuat

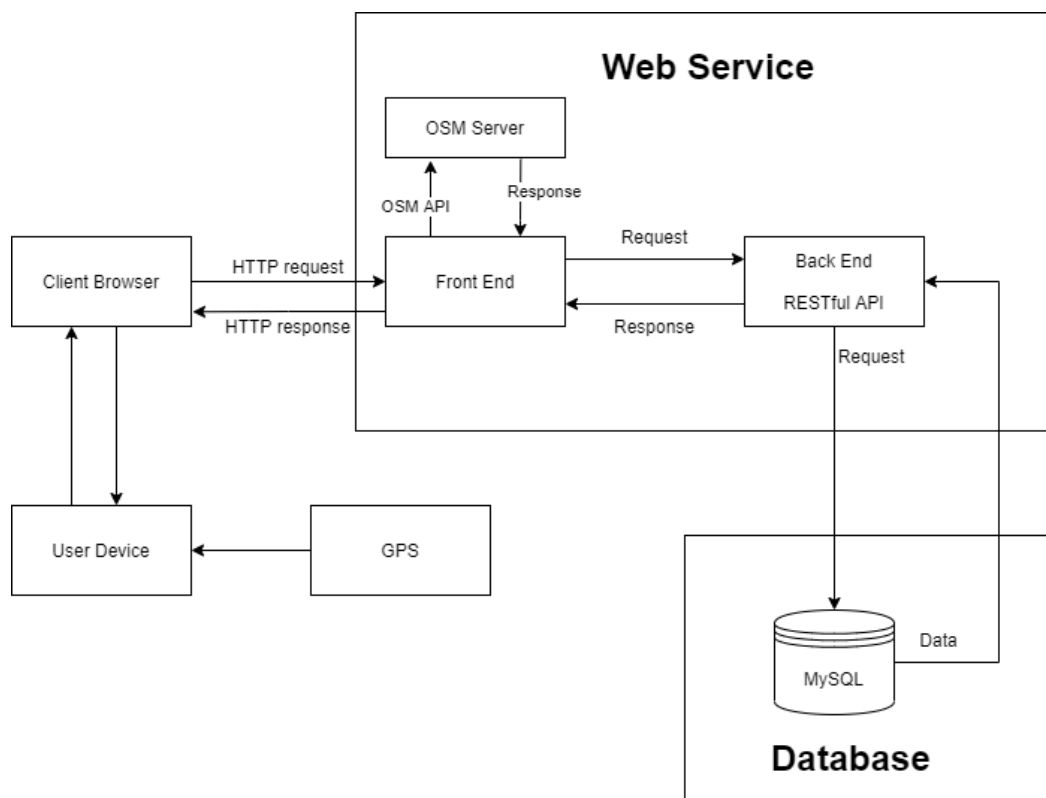
III.2 *Supplementary Requirements*

Pada bagian ini dijelaskan mengenai tiga bagian pembahasan, yaitu:

1. *Functional requirement* tambahan. Pada sub bagian ini dijelaskan mengenai *product perspective*, *product function*, dan *user characteristic*.
2. *Supporting requirement*. Pada sub bagian ini dijelaskan mengenai *access security* dan *responsibility*.
3. *Interface requirement*. Pada sub bagian ini dijelaskan mengenai detail antar muka yang terdiri dari *hardware interface*, *software interface*, *user interface*, dan *communication interface* dari aplikasi yang dikembangkan.

III.2.1 Product Perspective

Berikut merupakan penggambaran perspektif produk aplikasi yang dituangkan pada Gambar III.1 menggunakan tools draw.io.



Gambar III.15 Arsitektur diagram

Penggambaran perspektif produk Aplikasi pengelolaan presensi dan perizinan asrama kampus Polban merupakan aplikasi berbasis *web* yang digunakan secara

online dan terhubung ke GPS aktif pada perangkat mahasiswa asrama. Perangkat yang digunakan harus memiliki GPS aktif untuk mendukung penangkapan koordinat lokasi dengan hasil lebih akurat. Presensi kehadiran pada aplikasi ini menerapkan akses lokasi menggunakan Geolocation API dan GPS. Komunikasi antara *server* dan *client* dilakukan melalui *web service* dari API aplikasi yang dibuat dengan menggunakan *format* data JSON.

III.2.2 *Product Function*

Berdasarkan kebutuhan aplikasi yang dikembangkan berangkat dari hasil analisis sistem berjalan, permasalahan yang ditemukan, hasil analisis potensi terkomputerisasi, dan usulan penyelesaian masalah, maka terdapat penetapan *requirement* atau kebutuhan aplikasi. Aplikasi pengelolaan presensi dan perizinan asrama kampus Polban ditujukan bagi mahasiswa asrama yang menetap di gedung A (putri), B (putra), dan C (program tertentu). Aplikasi ini dapat berfungsi sebagai perangkat lunak yang dapat melakukan pencatat presensi kehadiran dengan memanfaatkan lokasi sebagai bentuk validasi, pengajuan perizinan mahasiswa (izin pulang, izin kembali dan izin *resign*), pemberian *approval* perizinan, perekapan data presensi dan perizinan, serta pengelolaan data mahasiswa meliputi *create* data, *read*, *update*, dan *import* data dalam bentuk *file*. Aplikasi melakukan deteksi lokasi menggunakan Geolocation API dan GPS pada perangkat pengguna. Melakukan penentuan status keberadaan dengan menerapkan formula Haversine. Terdapat penggunaan Laravel mail *notification* untuk menangani proses *approval* izin. Adapun *functional requirment* dari aplikasi ini adalah :

1. Aplikasi dapat melakukan validasi akun pengguna dan *role user*.
[REQ-F-01]
2. Aplikasi dapat menampilkan *dashboard* pengguna berdasarkan *role user*.
[REQ-F-02]
3. Aplikasi dapat menyimpan presensi kehadiran yang dilakukan mahasiswa asrama. [REQ-F-03]

4. Aplikasi dapat mengambil koordinat lokasi dengan Geolocation API dan GPS pada saat pengisian presensi dan saat pengubahan status izin menjadi kembali oleh mahasiswa asrama. **[REQ-F-04]**
5. Aplikasi dapat menampilkan *access location permission* saat pertama kali mahasiswa asrama mengakses aplikasi. **[REQ-F-05]**
6. Aplikasi dapat menghitung jarak dua lokasi (keberadaan mahasiswa dan titik lokasi asrama) dengan formula Haversine saat pengisian presensi dan saat pengubahan status izin menjadi kembali (sudah kembali ke asrama dari masa izinnya) yang dilakukan oleh mahasiswa asrama. **[REQ-F-06]**
7. Aplikasi dapat menampilkan status dan *maps* keberadaan mahasiswa asrama berdasarkan hasil perhitungan radius dengan penetapan radius 50 meter dari titik asrama pada pengisian presensi kehadiran. **[REQ-F-07]**
8. Aplikasi dapat mengakomodir cek lokasi sebelum *submit* presensi oleh mahasiswa asrama. **[REQ-F-08]**
9. Aplikasi dapat menampilkan riwayat presensi kehadiran pribadi mahasiswa asrama. **[REQ-F-09]**
10. Aplikasi dapat melakukan pengecekan status perizinan pada tabel perizinan. Jika mahasiswa asrama berstatus izin, maka status presensi secara otomatis terisi menjadi izin. **[REQ-F-10]**
11. Aplikasi dapat melakukan pengecekan waktu presensi. Jika mahasiswa asrama mengakses halaman presensi di luar pukul 16.00-20.00 WIB, maka tidak dapat melakukan pengisian presensi kehadiran. **[REQ-F-11]**
12. Aplikasi dapat menyimpan pengajuan perizinan pulang oleh mahasiswa asrama. **[REQ-F-12]**
13. Aplikasi dapat menyimpan pengajuan perizinan kembali oleh mahasiswa asrama. **[REQ-F-13]**
14. Aplikasi dapat menyimpan pengajuan perizinan *resign* oleh mahasiswa asrama. **[REQ-F-14]**
15. Aplikasi dapat mengupdate status hasil *approval* perizinan pulang oleh pengelola asrama dan Wadir 3. **[REQ-F-15]**

16. Aplikasi dapat mengupdate status hasil *approval* perizinan kembali oleh pengelola asrama dan Wadir 3. **[REQ-F-16]**
17. Aplikasi dapat mengupdate status hasil *approval* perizinan *resign* oleh pengelola asrama dan Wadir 3. **[REQ-F-17]**
18. Aplikasi dapat mengupdate status perizinan pulang menjadi terkonfirmasi sudah kembali ke asrama yang dilakukan oleh mahasiswa asrama. **[REQ-F-18]**
19. Aplikasi dapat menampilkan riwayat perizinan pribadi mahasiswa asrama. **[REQ-F-19]**
20. Aplikasi dapat mengakomodir pengiriman *e-mail* sebagai notifikasi kepada mahasiswa asrama, pengelola asrama, dan Wadir 3. Pengiriman *e-mail* terjadi pada proses pengajuan perizinan pulang, kembali, *resign*, dan pemberian *approval* terhadap seluruh perizinan. **[REQ-F-20]**
21. Aplikasi dapat menampilkan halaman daftar perizinan yang masuk beserta statusnya kepada pengelola asrama dan Wadir 3. **[REQ-F-21]**
22. Aplikasi dapat mengakomodir pencarian nama mahasiswa di halaman daftar presensi dan perizinan yang masuk. **[REQ-F-22]**
23. Aplikasi dapat menampilkan halaman daftar seluruh mahasiswa asrama di Wadir 3. **[REQ-F-23]**
24. Aplikasi dapat mengakomodir pencarian nama mahasiswa pada daftar mahasiswa asrama di Wadir 3. **[REQ-F-24]**
25. Aplikasi dapat menampilkan halaman detail data mahasiswa asrama di Wadir 3. **[REQ-F-25]**
26. Aplikasi dapat mengupdate data mahasiswa oleh Wadir 3. **[REQ-F-26]**
27. Aplikasi dapat menyimpan data mahasiswa hasil *input* oleh Wadir 3. **[REQ-F-27]**
28. Aplikasi dapat mengakomodir *import* data mahasiswa berupa *format* file .xls dan .xlsx maksimum 2 MB oleh Wadir 3. **[REQ-F-28]**
29. Aplikasi dapat menampilkan rekapitulasi data presensi dan perizinan di pengelola asrama dan Wadir 3. **[REQ-F-29]**

30. Aplikasi dapat menampilkan data rekapitulasi sesuai rentang tanggal yang diinputkan di pengelola asrama dan Wadir 3. **[REQ-F-30]**
31. Aplikasi dapat mengakomodir unduh *file* rekapitulasi dengan file yang terunduh berformat .pdf. **[REQ-F-31]**
32. Aplikasi dapat menampilkan daftar presensi setiap hari yang dapat diakses oleh pengelola asrama dan Wadir 3. **[REQ-F-32]**
33. Aplikasi dapat membuat akun mahasiswa saat Wadir 3 menambah data mahasiswa atau melakukan *import* data mahasiswa, dengan NIM sebagai *default password*. **[REQ-F-33]**
34. Aplikasi dapat mengakomodir pengajuan izin kembali ke asrama jika izin kembali ke asrama sebelumnya ditolak. **[REQ-F-34]**
35. Aplikasi dapat mengakomodir unggah file pada pengajuan izin pulang dengan format .pdf/.jpg/.png. **[REQ-F-35]**
36. Aplikasi dapat mengakomodir unggah file pada pengajuan izin *resign* dengan format .pdf/.jpg/.png. **[REQ-F-36]**
37. Aplikasi dapat mengakomodir penghapusan data mahasiswa. **[REQ-F-37]**

Selain fungsi utama dari aplikasi, terdapat kebutuhan non fungsional (*nonfunctional requirement*) untuk menunjang berjalannya aplikasi, yang terdapat pada subbab *quality attribute*.

III.2.3 User Characteristic

Terdapat tiga jenis pengguna yang dibedakan berdasarkan *interest* dan karakteristiknya terhadap aplikasi. Karakter pengguna digambarkan berdasarkan *role user*:

1. Mahasiswa Asrama [USER-01]

Mahasiswa asrama adalah mahasiswa aktif kampus Polban yang datanya tercatat pada tahun tersebut sebagai penghuni asrama. Mahasiswa asrama terdiri atas mahasiswa tingkat pertama dan mahasiswa tingkat kedua yang dinobatkan sebagai pengurus asrama dan keduanya menetap di asrama. Mahasiswa asrama harus dapat mengoperasikan *smartphone* atau laptop atau PC dan perangkat sejenis lainnya yang akan digunakan untuk mengakses aplikasi secara paham. Memiliki perangkat GPS dan *browser* yang memadai penggunaan aplikasi. Selain dapat mengoperasikan beberapa perangkat secara paham, pengguna dengan *role* ini perlu memahami cara kerja aplikasi, terutama saat akan melakukan pengisian presensi dan pengajuan perizinan. Pengguna ini secara keseluruhan memiliki peran dengan aplikasi yaitu dapat melakukan presensi kehadiran, dapat mengajukan perizinan pulang, kembali, dan *resign*, serta dapat menerima persetujuan atau penolakan izin yang diajukan.

2. Pengelola Asrama [USER-02]

Pengelola asrama adalah staf pegawai yang mendiami asrama dan bertugas untuk memantau serta mengelola kegiatan asrama mahasiswa kampus Polban. Pengelola asrama harus dapat mengoperasikan *smartphone* atau laptop atau PC dan perangkat sejenis lainnya yang akan digunakan untuk mengakses aplikasi. Selain dapat mengoperasikan beberapa perangkat dengan baik, pengelola perlu memahami cara kerja aplikasi, terutama untuk proses pemberian *approval* atau persetujuan pengajuan izin yang akan masuk. Pengelola secara keseluruhan memiliki peran dengan aplikasi yaitu memberikan status perizinan mahasiswa sebelum dilakukan perizinan oleh Wadir 3. Selain itu, pengelola dapat mengakses dan mengunduh rekapitulasi data presensi dan perizinan mahasiswa.

3. Wakil Direktur (Wadir) 3 [USER-03]

Wadir 3 adalah pihak manajemen Polban bagian Wakil Direktur (Wadir) 3 bidang kemahasiswaan yang memiliki tanggung jawab terhadap asrama Polban. Wadir 3 sebagai *super admin* yang dapat mengelola data mahasiswa. Wadir 3 ini harus dapat mengoperasikan *smartphone* atau laptop atau PC dan perangkat sejenis lainnya yang akan digunakan untuk mengakses aplikasi. Selain dapat mengoperasikan beberapa perangkat dengan baik, Wadir 3 perlu memahami cara kerja aplikasi. Cara kerja tersebut terutama untuk proses pemberian *approval* atau persetujuan pengajuan izin yang akan masuk. Pengelola secara keseluruhan memiliki peran dengan aplikasi yaitu memberikan status perizinan mahasiswa setelah dilakukan perizinan oleh pengelola asrama. Wadir 3 dapat mengakses dan mengunduh rekapitulasi data presensi dan perizinan mahasiswa. Selain itu, haru mampu melakukan kelola data mahasiswa yang meliputi tambah, ubah, lihat detail, dan *import* data mahasiswa.

III.2.4 Access Security

1. Aplikasi menerapkan validasi *role user*, *authentication* dan enkripsi token pada pengelolaan akses pengguna. [REQ-NF-01]
2. Aplikasi menerapkan kemampuan untuk *reset password* akun pengguna menggunakan konfirmasi melalui *e-mail*. [REQ-NF-02]

III.2.5 Responsibility

Aplikasi mendukung akses penggunaan dengan *smartphone* sehingga tampilan bersifat responsif. [REQ-NF-03]

III.2.6 Interface

Berikut merupakan penjelasan lebih rinci terkait antar muka pada aplikasi, terdiri dari *user interface*, *hardware interface*, *software interface*, dan *communication interface*.

III.2.6.1 *User Interface*

Aplikasi pengelolaan presensi dan perizinana asrama mahasiswa kampus Polban berbasis *web* ini memiliki rancangan tampilan pada setiap halaman yang disajikan, berikut merupakan *user interface* yang diperlukan pada aplikasi ini yaitu:

1. Halaman *Login* Pengguna [UI-01]

Pada halaman ini akan terdapat *form login* pengguna dan logo Polban sebagai identitas bahwa aplikasi yang digunakan adalah aplikasi milik Polban. *Form* berisi *text fields e-mail* dan *password* yang harus diisi. Selain itu terdapat *button login* untuk pengguna masuk ke aplikasi dan *button reset password*.

2. Halaman *Dashboard* Pengguna

Halaman *dashboard* adalah halaman utama yang akan ditampilkan pertama kali ketika pengguna berhasil masuk ke aplikasi. Setiap *role user* memiliki tampilan *dashboard* yang berbeda sesuai dengan otorisasi yang diberikan.

- a. *Dashboard* Mahasiswa Asrama [UI-02]

Dashboard mahasiswa menampilkan data terkait nama gedung yang ditempati, jumlah kehadiran, alfa, dan izin setiap bulan. Data tersebut ditampilkan dalam sebuah *card view Typografi* yang digunakan yaitu ukuran huruf pada angkayang tertera lebih terlihat jelas dibanding keterangannya. Hal tersebut dipilih agar pengguna dapat dengan mudah menangkap jumlah angka yang dimaksud dan menyesuaikan dengan estetika.

- b. *Dashboard* Pengelola Asrama dan Wadir 3 [UI-03]

Dashboard pengelola dan Wadir 3 menampilkan data asrama, data presensi, dan data perizinan secara ringkas. Data asrama berisi data jumlah mahasiswa asrama, jenis gedung, dan jumlah pengurus. Pada data presensi berisi jumlah mahasiswa status hadir, alfa, dan izin setiap bulan. Data perizinan berisi jumlah mahasiswa dengan status belum disetujui, disetujui, dan ditolak. Data tersebut ditampilkan

dalam sebuah *card view* yang disusun sedemikian rupa agar pengguna dapat dengan mudah menangkap jumlah angka yang dimaksud dan menyesuaikan dengan estetika. Selain itu, pada *card* yang berisi data presensi dan data perizinan terdapat *text button* “lihat detail” yang mengarah pada daftar rekap presensi dan perizinan.

3. Halaman Pengisian Presensi Kehadiran [UI-04]

Halaman ini menampilkan *form* presensi yang diisi oleh mahasiswa. Halaman ini terdiri atas keterangan waktu presensi, data *latitude* dan *longitude*, *text fields* isian suhu badan dan kondisi kesehatan, *button* untuk melakukan *get lokasi*, *mini maps* untuk menampilkan keberadaan mahasiswa, dan *submit button* untuk mahasiswa menyimpan data presensi yang sudah dilakukan.

4. Halaman Riwayat Presensi Kehadiran [UI-05]

Halaman ini menampilkan data riwayat presensi pribadi mahasiswa. Halaman menampilkan data dalam bentuk *list* dan pemberian aksesoris warna pada setiap jenis status yang berbeda – beda. Selain itu pada halaman ini terdapat *pagination*. Halaman riwayat presensi kehadiran menampilkan data tanggal, waktu presensi, koordinat, dan status.

5. Halaman Pengajuan Perizinan Pulang [UI-06]

Halaman ini menampilkan *form* isian yang terdiri atas keterangan atau alasan izin dengan tipe *text fields*, *datepicker* untuk mengisi tanggal mulai izin dan tanggal selesai, tanggal kembali ke asrama, kondisi kesehatan, suhu badan, dan alamat tujuan, serta *button to choose file* dan *submit button* untuk menyimpan pengajuan izin pulang.

6. Halaman Pengajuan Perizinan Kembali [UI-07]

Halaman ini menampilkan *form* isian yang terdiri atas keterangan atau alasan izin kembali ke asrama dengan tipe *text fields*, *datepicker* untuk mengisi tanggal pengajuan kembali ke asrama, isian suhu badan, kondisi kesehatan, dan jenis kendaraan yang digunakan. Selain itu, *button to choose file* dan *submit button* untuk menyimpan pengajuan izin kembali ke asrama.

7. Halaman Pengajuan Perizinan *Resign* [UI-08]

Halaman ini menampilkan *form* isian yang terdiri atas keterangan atau alasan *resign* dari asrama dengan tipe *text fields*, *datepicker* untuk mengisi tanggal pengajuan *resign*, isian suhu badan, kondisi kesehatan, dan jenis kendaraan yang digunakan, dan melampirkan *file* foto STNK jika menggunakan kendaraan pribadi. Selain itu, *submit button* untuk menyimpan pengajuan izin *resign*.

8. Halaman Riwayat Perizinan [UI-09]

Halaman ini menampilkan data riwayat perizinan pribadi mahasiswa. Halaman menampilkan data dalam bentuk *list* dan pemberian aksesoris warna pada setiap jenis status yang berbeda – beda. Selain itu pada halaman ini terdapat *pagination* dan *button* untuk mahasiswa melakukan konfirmasi kepulangan dan pengajuan izin kembali ke asrama. *Button* akan berubah warna ketika mahasiswa berhasil melakukan konfirmasi kepulangan, hal tersebut dilakukan agar mahasiswa mudah mengenali setiap status dari perizinannya.

9. Halaman Daftar Perizinan yang Masuk [UI-10]

Halaman ini terbagi menjadi tiga halaman, yaitu daftar perizinan pulang, kembali, dan *resign*. Secara keseluruhan, aplikasi menampilkan data yang sama pada setiap halaman, sedikit berbeda pada jenis tanggal yang ditampilkan. Halaman menampilkan data dalam bentuk *list* dan *pagination*. Terdapat *text field* pencarian dan *button* untuk memulai pencarian data

mahasiswa. Selain itu pada halaman ini terdapat *button* untuk melakukan *approval* setiap izin. *Button* akan berubah warna ketika pengelola asrama atau Wadir 3 berhasil melakukan aksi, hal tersebut terjadi agar dapat mudah mengenali setiap status dari aksi yang dilakukannya.

10. Halaman *Approval* Perizinan Pulang, Kembali, dan *Resign* [UI-11]

Halaman ini terbagi menjadi tiga halaman, yaitu *approval* perizinan pulang, kembali, dan *resign*. Secara keseluruhan, aplikasi menampilkan data yang sama pada setiap halaman, sedikit berbeda pada jenis tanggal yang ditampilkan. *Form* terdiri atas *radio button* untuk memilih status *approval* apakah disetujui atau ditolak. Terdapat *text field* yang dapat diisi dengan catatan atau alasan penolakan dari pengelola asrama atau Wadir 3. Selain itu, *button* untuk mengunduh lampiran yang disertakan. Pada bagian akhir terdapat *submit button* yang digunakan untuk menyimpan hasil *approval*.

11. Halaman Rekapitulasi Presensi dan Perizinan [UI-12]

Halaman ini menampilkan daftar mahasiswa yang disajikan ke dalam bentuk *list*. *List* tersebut berisi data mahasiswa dan keterangan jumlah presensi dengan status alfa, hadir, izin, dan status keaktifan. Terdapat *pagination* untuk menjangkau data lain agar dapat ditampilkan. Terdapat pula *text field* pencarian dan *button* untuk memulai pencarian data mahasiswa. Selain itu pada halaman ini terdapat *datepicker* yang digunakan untuk menginputkan rentang tanggal yang diinginkan untuk ditampilkan.

12. Halaman *Import* Data Mahasiswa [UI-13]

Halaman ini menampilkan *form* yang digunakan Wadir 3 untuk dapat *mengimport* data mahasiswa ke aplikasi. *Form* terdiri *choose file button* sebagai tempat Wadir 3 untuk mengunggah *file* data mahasiswa untuk di *import*. Pada bagian akhir terdapat *submit button* yang digunakan untuk menyimpan hasil *import*.

13. Halaman Daftar Mahasiswa [UI-14]

Halaman ini menampilkan daftar mahasiswa yang disajikan ke dalam bentuk *list*. *List* tersebut berisi data mahasiswa dan *button* info untuk melihat *detail* data mahasiswa serta *button* edit untuk mengubah data mahasiswa. Terdapat *pagination* untuk menjangkau data lain agar dapat ditampilkan. Terdapat pula *text field* pencarian dan *button* untuk memulai pencarian data mahasiswa. Selain itu pada halaman ini terdapat *add button* untuk memudahkan Wadir 3 ketika ingin menambah baru data mahasiswa.

14. Halaman *Insert* Data Mahasiswa [UI-15]

Halaman ini menampilkan *form* yang digunakan Wadir 3 untuk dapat menambah data mahasiswa. *Form* terdiri atas *text field* untuk mengisi data isian seperti nama, NIM, dan alamat. Terdapat *datepicker* untuk memilih tanggal lahir, *dropdown list* untuk memilih prodi, jurusan, dan agama, nomor kamar, jabatan atau *role* serta terdapat *radio button* untuk memilih jenis kelamin. Pada bagian akhir terdapat *submit button* yang digunakan untuk menyimpan data yang diinputkan.

15. Halaman *Update* Data Mahasiswa [UI-16]

Halaman ini menampilkan *form* yang digunakan Wadir 3 untuk dapat mengubah data mahasiswa. *Form* terdiri atas *text field* untuk mengisi *password*, *dropdown list* untuk memilih nomor kamar, status keaktifan, dan jabatan atau *role*. Pada bagian akhir terdapat *submit button* yang digunakan untuk menyimpan data yang diinputkan.

16. Halaman Rincian Data Mahasiswa [UI-17]

Halaman ini menampilkan data *detail* dari mahasiswa yang dipilih. Disajikan serupa dengan tabel, kolom bagian kiri digunakan untuk menampilkan keterangan, dan kolom kanan untuk menampilkan isi data dari setiap keterangan, misalnya kolom kiri terdiri atas keterangan nama, NIM, alamat dan kolom kanan terdiri atas isi data yaitu nama mahasiswa, nim

mahasiswa, dan alamat mahasiswa. Selain itu, halaman ini menyediakan *edit button* untuk memudahkan Wadir 3 ketika ingin mengubah data dari mahasiswa yang sedang dilihat.

17. Halaman *Reset Password* [UI-18]

Pada halaman ini terdapat *form reset password* pengguna. *Form* berisi *text fields e-mail, password* baru dan konfirmasi *password* baru yang harus diisi. Selain itu terdapat *button submit* untuk pengguna mengubah data *password*.

18. Halaman Daftar Presensi yang Masuk [UI-19]

Halaman ini menampilkan data seluruh riwayat presensi yang telah dilakukan setiap mahasiswa asrama. Data yang ditampilkan berupa data harian yang akan berubah setiap harinya, sesuai presensi yang dilakukan. Halaman ini menampilkan tabel dalam menyajikan data. Kolom data yang ditampilkan pada tabel meliputi nama, NIM, gedung, jenis, waktu, suhu badan, kondisi kesehatan, dan status presensi. Terdapat pula *bar* pencarian untuk mencari data mahasiswa.

III.2.6.2 *Hardware Interface*

Pada aplikasi ini terdapat *hardware interface* yang dibutuhkan, yaitu perangkat keras yang digunakan dapat berupa PC, Laptop, *smartphone*, dan tablet yang mendukung GPS. Antar muka perangkat keras tersebut digunakan pengguna untuk mengakses aplikasi. GPS diakses melalui perangkat pengguna. GPS ini akan digunakan pada saat proses pencatatan kehadiran yang mengharuskan pengguna untuk melakukan mengaktifkan GPS sehingga sistem pada aplikasi dapat dengan mudah mengidentifikasi keberadaan pengguna secara lebih akurat.

III.2.6.3 *Software Interface*

Software interface merupakan antar muka dari perangkat lunak yang berhubungan dengan aplikasi. Dalam hal ini yang digunakan pada aplikasi adalah penggunaan OSM (*Open Street Map*) API. OSM API adalah API yang digunakan untuk

melakukan pemanggilan terhadap tampilan *maps* milik OSM. *Maps* ditampilkan ketika terdapat *request* pengambilan lokasi pengguna yang kemudian ditampilkan pada *maps*, hal tersebut berjalan pada proses pengisian presensi.

III.2.6.4 Communication Interface

Communication interface merupakan antar muka yang digunakan untuk melakukan komunikasi antar *side*. Dalam hal ini komunikasi terjadi antara *client* dengan *server side*. Komunikasi antara *server* dan *client* dilakukan menggunakan API protokol HTTP. Protokol HTTP meliputi HTTP *request* dan *response*. HTTP *request* yang digunakan meliputi *method* GET, PUT, dan POST. HTTP *response* meliputi *response code*. Aplikasi ini dapat diakses menggunakan perangkat yang terhubung dengan jaringan internet.

IV. Supporting Information

IV.1 Requirement Traceability

Requirement traceability adalah penjelasan yang digambarkan menggunakan tabel yang berisi daftar *functional* dan *nonfunctional requirement*, serta status dari *requirement* tersebut untuk memastikan semua *requirement* telah terpenuhi. Berikut merupakan jenis verifikasi pada tabel *requirement traceability*, yaitu:

1. Inspeksi

Inspeksi adalah pemeriksaan yang dilakukan terhadap suatu produk (biasanya kode program) yang dihasilkan apakah sesuai dengan standar dan aturan yang telah ditetapkan.

2. Analisis

Analisis adalah pemeriksaan spesifikasi yang dilakukan terhadap produk yang dihasilkan dengan menggunakan model matematika, simulasi, algoritma pengujian, perhitungan, grafik dan sebagainya.

3. Demonstrasi

Demonstrasi adalah pemeriksaan fungsional yang merupakan persyaratan spesifikasi dengan mengamati hasil kualitatif operasi atau melalui latihan yang dilakukan dalam kondisi tertentu.

Requirement traceability untuk aplikasi pengelolaan presensi dan perizinan asrama mahasiswa Polban ditunjukkan pada Tabel IV.1.

Tabel IV.1 *Requirement traceability*

Kode	Functional Requirement	Verifikasi		
		Inspeksi	Analisis	Demonstrasi
[REQ-F-01]	Aplikasi dapat melakukan validasi akun pengguna dan <i>role user</i> .	✓	-	✓
[REQ-F-02]	Aplikasi dapat menampilkan <i>dashboard</i> pengguna berdasarkan <i>role user</i> .	✓	-	✓
[REQ-F-03]	Aplikasi dapat menyimpan presensi kehadiran yang dilakukan mahasiswa asrama.	✓	-	
[REQ-F-04]	Aplikasi dapat mengambil koordinat lokasi dengan Geolocation API dan GPS pada saat pengisian presensi dan saat pengubahan status izin menjadi kembali oleh mahasiswa asrama.	✓	-	✓
[REQ-F-05]	Aplikasi dapat menampilkan <i>access location permission</i> saat pertama kali mahasiswa asrama mengakses aplikasi.	✓	-	✓
[REQ-F-06]	Aplikasi dapat menghitung jarak dua lokasi (keberadaan mahasiswa dan titik lokasi asrama) dengan formula Haversine saat pengisian presensi dan saat pengubahan status izin menjadi kembali (sudah kembali ke asrama dari masa izinnya) yang dilakukan oleh mahasiswa asrama.	✓	✓	✓
[REQ-F-07]	Aplikasi dapat menampilkan status dan <i>maps</i> keberadaan mahasiswa asrama berdasarkan hasil perhitungan di pengisian presensi kehadiran.	✓	-	✓

Kode	Functional Requirement	Verifikasi		
		Inspeksi	Analisis	Demonstrasi
[REQ-F-08]	Aplikasi dapat mengakomodir cek lokasi sebelum <i>submit</i> presensi oleh mahasiswa asrama.	✓	-	✓
[REQ-F-09]	Aplikasi dapat menampilkan riwayat presensi kehadiran pribadi mahasiswa asrama.	✓	-	✓
[REQ-F-10]	Aplikasi dapat melakukan pengecekan status perizinan pada tabel perizinan. Jika mahasiswa asrama berstatus izin, maka status presensi secara otomatis terisi menjadi izin.	✓	-	✓
[REQ-F-11]	Aplikasi dapat melakukan pengecekan waktu presensi. Jika mahasiswa asrama mengakses halaman presensi di luar jam presensi, maka tidak dapat mengisi presensi kehadiran.	✓	-	✓
[REQ-F-12]	Aplikasi dapat menyimpan pengajuan perizinan pulang oleh mahasiswa asrama.	✓	-	✓
[REQ-F-13]	Aplikasi dapat menyimpan pengajuan perizinan kembali oleh mahasiswa asrama.	✓	-	✓
[REQ-F-14]	Aplikasi dapat menyimpan pengajuan perizinan <i>resign</i> oleh mahasiswa asrama.	✓	-	✓
[REQ-F-15]	Aplikasi dapat mengupdate status hasil <i>approval</i> perizinan pulang oleh pengelola asrama dan Wadir 3.	✓	-	✓
[REQ-F-16]	Aplikasi dapat mengupdate status hasil <i>approval</i> perizinan kembali oleh pengelola asrama dan Wadir 3.	✓	-	✓
[REQ-F-17]	Aplikasi dapat mengupdate status hasil <i>approval</i> perizinan resign oleh pengelola asrama dan Wadir 3.	✓	-	✓
[REQ-F-18]	Aplikasi dapat mengupdate status perizinan pulang menjadi terkonfirmasi sudah kembali ke asrama yang dilakukan oleh mahasiswa asrama.	✓	-	✓

Kode	Functional Requirement	Verifikasi		
		Inspeksi	Analisis	Demonstrasi
[REQ-F-19]	Aplikasi dapat menampilkan riwayat perizinan pribadi mahasiswa asrama.	✓	-	✓
[REQ-F-20]	Aplikasi dapat mengakomodir pengiriman <i>e-mail</i> sebagai notifikasi kepada mahasiswa asrama, pengelola asrama, dan Wadir 3. Pengiriman <i>e-mail</i> terjadi pada proses pengajuan perizinan pulang, kembali, <i>resign</i> , dan pemberian <i>approval</i> terhadap seluruh perizinan.	✓	-	✓
[REQ-F-21]	Aplikasi dapat menampilkan halaman daftar perizinan yang masuk beserta statusnya kepada pengelola asrama dan Wadir 3.	✓	-	✓
[REQ-F-22]	Aplikasi dapat mengakomodir pencarian nama mahasiswa di halaman daftar presensi dan perizinan yang masuk.	✓	-	✓
[REQ-F-23]	Aplikasi dapat menampilkan halaman daftar seluruh mahasiswa di Wadir 3.	✓	-	✓
[REQ-F-24]	Aplikasi dapat mengakomodir pencarian nama mahasiswa pada daftar mahasiswa asrama di Wadir 3.	✓	-	✓
[REQ-F-25]	Aplikasi dapat menampilkan halaman detail data mahasiswa asrama di Wadir 3.	✓	-	✓
[REQ-F-26]	Aplikasi dapat mengupdate data mahasiswa oleh Wadir 3.	✓	-	✓
[REQ-F-27]	Aplikasi dapat menyimpan data mahasiswa hasil input oleh Wadir 3.	✓	-	✓
[REQ-F-28]	Aplikasi dapat mengakomodir <i>import</i> data mahasiswa berupa <i>format file .xlsx</i> dan <i>.xlsx</i> maksimum 2 MB oleh Wadir 3.	✓	-	✓

Kode	Functional Requirement	Verifikasi		
		Inspeksi	Analisis	Demonstrasi
[REQ-F-29]	Aplikasi dapat menampilkan rekapitulasi data presensi dan perizinan di pengelola asrama dan Wadir 3.	✓	-	✓
[REQ-F-30]	Aplikasi dapat menampilkan data rekapitulasi sesuai rentang tanggal yang diinputkan di pengelola asrama dan Wadir 3.	✓	-	✓
[REQ-F-31]	Aplikasi dapat mengakomodir unduh file rekapitulasi.	✓	-	✓
[REQ-F-32]	Aplikasi dapat menampilkan riwayat presensi per hari seluruh mahasiswa, yang dapat diakses oleh pengelola asrama dan Wadir 3.	✓	-	✓
[REQ-F-33]	Aplikasi dapat membuat akun mahasiswa saat Wadir 3 menambah data mahasiswa atau melakukan <i>import</i> data mahasiswa.	✓	-	✓
[REQ-F-34]	Aplikasi dapat mengakomodir pengajuan izin kembali ke asrama jika izin kembali ke asrama sebelumnya ditolak.	✓	-	✓
[REQ-F-34]	Aplikasi dapat mengakomodir pengajuan izin kembali ke asrama jika izin kembali ke asrama sebelumnya ditolak.	✓	-	✓
[REQ-F-35]	Aplikasi dapat mengakomodir unggah file pada pengajuan izin pulang dengan format .pdf/.jpg/.png	✓	-	✓
[REQ-F-36]	Aplikasi dapat mengakomodir unggah file pada pengajuan izin <i>resign</i> dengan format .pdf/.jpg/.png.	✓	-	✓
[REQ-F-37]	Aplikasi dapat mengakomodir hapus data mahasiswa.	✓	-	✓

Kode	Functional Requirement	Verifikasi		
		Inspeksi	Analisis	Demonstrasi
[REQ-NF-01]	Aplikasi menerapkan validasi <i>role user</i> , authentication dan enkripsi token pada pengelolaan akses pengguna.	✓	-	✓
[REQ-NF-02]	Aplikasi menerapkan kemampuan untuk <i>reset password</i> akun pengguna menggunakan konfirmasi melalui <i>e-mail</i> .	✓	-	✓
[REQ-NF-03]	Aplikasi mendukung akses penggunaan dengan <i>smartphone</i> sehingga tampilan bersifat responsif.	✓	-	✓

V. Concern / Queries / Doubts if any

N/A.

Dokumen SRS ini akan menjadi acuan dalam tahap perancangan, implementasi, dan pengujian aplikasi.