

**APLIKASI E-PLUGIN BERBASIS *ANDROID***  
**(STUDI KASUS ORGANISASI POLTEK HARBER LINUX *USER GROUP*-  
INDONESIA (PLUGIN))**



**TUGAS AKHIR**

Diajukan Sebagai Salah Satu Syarat Untuk Menempuh Mata Kuliah Tugas Akhir  
Pada Program Studi D IV Teknik Informatika

Oleh :

Nama : Akhmad Tantowi

NIM : 17090031

**PROGRAM STUDI D IV TEKNIK INFORMATIKA**

**POLITEKNIK HARAPAN BERSAMA**

**TEGAL**

**2023**

## HALAMAN PERNYATAAN

Yang bertanda tangan dibawah ini :

Nama : Akhmad Tantowi

NIM : 17090031

Adalah mahasiswa Program Studi D IV Teknik Informatika Politeknik Harapan Bersama, dengan ini saya menyatakan bahwa laporan Tugas Akhir yang berjudul :

**“APLIKASI E-PLUGIN BERBASIS *ANDROID* (STUDI KASUS ORGANISASI POLTEK HARBER LINUX *USER GROUP*- INDONESIA (PLUGIN) )”**

Merupakan hasil pemikiran sendiri secara orisinil dan saya susun secara mandiri dengan tidak melanggar kode etik hak karya cipta. Pada laporan Tugas Akhir ini juga bukan merupakan karya yang pernah diajukan untuk memperoleh gelar akademik tertentu di suatu Perguruan Tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila dikemudian hari ternyata Laporan Tugas Akhir ini terbukti melanggar kode etik karya cipta atau merupakan karya yang dikategorikan mengandung unsur plagiarisme, maka saya bersedia untuk melakukan penelitian baru dan menyusun laporannya sebagai laporan Tugas Akhir, sesuai dengan ketentuan yang berlaku. Demikian pernyataan ini saya buat dengan sebenarnya dan sesungguhnya.

Tegal, Agustus 2023

Yang membuat pernyataan,



Akhmad Tantowi

NIM. 17090031

## HALAMAN REKOMENDASI

Pembimbing Tugas Akhir memberikan rekomendasi kepada :

Nama : Akhmad Tantowi  
NIM : 17090031  
Program Studi : DIV Teknik Infomatika  
Judul Tugas Akhir : Aplikasi E-Plugin Berbasis Android (Studi Kasus  
Organisasi Politeknik Harber Linux User Group -  
Inddonesia)

Mahasiswa tersebut telah dinyatakan selesai melaksanakan bimbingan dan dapat mengikuti Ujian Tugas Akhir pada tahun akademik 2020/2021.

Pembimbing I,



M. Nishom., M.Kom

NIPY.09.017.337

Tegal, 24 Juli 2021

Pembimbing II,



Romi Muharyono, S. Ag

NIPY.-

## HALAMAN PENGESAHAN

Nama : Akhmad Tantowi  
NIM : 17090031  
Program Studi : D IV Teknik Informatika  
Judul Tugas Akhir : Aplikasi E-Plugin Berbasis *Android* (Studi Kasus Organisasi Politeknik Harber Linux *User Group*–Indonesia)

Dinyatakan LULUS/TIDAK LULUS setelah dipertahankan di hadapan Dewan Penguji Tugas Akhir Program Studi D IV Teknik Informatika Politeknik Harapan Bersama.

Tegal, Agustus 2023

Dewan Penguji

Nama Tanda Tangan

1. Ketua : Muhammad Fikri Hidayattullah, S.T., M.Kom.
2. Anggota I : Taufiq Abidin, M.Kom.
3. Anggota II : Romi Muharyono, S. Ag.

1.   
2.   
3. 

Mengetahui

Ketua Program Studi D IV Teknik Informatika



Slamet Widodo, S.Pd., M. Eng

NIDN. 08.015.222

## ABSTRAK

Pendidikan kini menjadi modal penting untuk meningkatkan dan mengembangkan kualitas sumber daya manusia. Sebagai upaya untuk meningkatkan kualitas pendidikan, penerapan teknologi meningkat secara drastis dalam segala bidang, termasuk juga bidang pendidikan. Untuk mengatasi hal tersebut dibuatlah sebuah aplikasi yang bernama e-plugin berbasis *android* studi kasus organisasi poltek harber linux *User group* Indonesia (PLUGIN). Penelitian ini menggunakan alur penelitian *Waterfall*, dimana metode pengumpulan data dengan mengidentifikasi masalah dan observasi. setelah data terkumpul dilakukan perancangan sistem yang meliputi perancangan proses, perancangan struktur menu, dan perancangan *Interface*. selanjutnya pembuatan sistem dengan menggunakan bahasa pemrograman *Kotlin*. setelah aplikasi dihasilkan maka dilakukan uji program dengan *Black-Box Testing*. Hasil dari penelitian ini adalah sebuah aplikasi e-plugin pada komunitas Plugin yang dapat digunakan untuk membantu organisasi dalam penyampaian pembelajaran secara cepat dan terpusat.

## KATA PENGANTAR

Dengan memanjatkan puji syukur kehadirat *Allah* SWT, Tuhan Yang Maha Pengasih dan Maha Penyayang yang telah melimpahkan segala rahmat, hidayah dan inayah-Nya sehingga terselesaikannya Laporan Tugas Akhir yang berjudul “Aplikasi E-Plugin Berbasis *Android* (Studi Kasus Organisasi Politeknik Harber Linux *User Group-Indonesia*)”.

Tugas Akhir merupakan suatu kewajiban yang harus dilaksanakan untuk memenuhi salah satu syarat kelulusan dalam mencapai derajat Sarjana Sains Terapan pada Program Studi D IV Teknik Informatika Politeknik Harapan Bersama. Selama melaksanakan penelitian dan kemudian tersusun dalam laporan Tugas Akhir ini, banyak pihak yang telah memberikan bantuan, dukungan dan bimbingannya.

Pada kesempatan ini, tak lupa penulis mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Bapak Nizar Suhendra, S.E., MPP selaku Direktur Politeknik Harapan Bersama Tegal.
2. Bapak Slamet Wiyono, M.kom selaku Ketua Program Studi D IV Teknik Informatika Politeknik Harapan Bersama.
3. Bapak M Nishom, M.Kom selaku dosen pembimbing I.
4. Bapak Romi Muharyono, S. Ag selaku dosen pembimbing II.
5. Keluarga dan teman-teman yang telah mendukung, membantu serta mendoakan penyelesaian laporan Tugas Akhir ini.

Tegal, Agustus 2023  
Yang membuat pernyataan,

Akhmad Tantowi  
NIM. 17090031

## DAFTAR ISI

HALAMAN PERNYATAAN .....	ii
HALAMAN REKOMENDASI .....	iii
HALAMAN PENGESAHAN.....	iv
ABSTRAK .....	v
KATA PENGANTAR .....	vi
DAFTAR ISI.....	vii
DAFTAR GAMBAR .....	x
DAFTAR TABEL.....	xiv
DAFTAR LAMPIRAN .....	xv
BAB I .....	1
PENDAHULUAN.....	1
1.1. Latar Belakang.....	1
1.2. Perumusan Masalah .....	3
1.3. Pembahasan Masalah.....	4
1.4. Tujuan Penelitian .....	4
1.5. Manfaat Penelitian .....	4
1.6. Metodologi Penelitian.....	5
1.6.1. Bahan Penelitian .....	5
1.6.2. Alat Penelitian .....	5
1.6.3. Alur Penelitian .....	6
1.7. Sistematika Penulisan .....	9
BAB II.....	12
TINJAUAN PENULISAN.....	12
BAB III.....	19

LANDASAN TEORI .....	19
3.1. Komunitas Plugin.....	19
3.2. Aplikasi .....	19
3.3. Android .....	19
3.4. Kotlin .....	23
3.5. Unified Modeling Language(UML) .....	24
1.1.1. Use Case Diagram .....	24
3.6. Activity Diagram .....	27
BAB IV .....	28
PERANCANGAN DAN DESAIN .....	28
4.1. Perancangan Sistem .....	28
4.2. Perancangan UML .....	28
4.2.1. Use Case Diagram .....	28
4.2.2. Activity Diagram .....	29
4.2.3. Sequence Diagram .....	40
4.2.4. Class Diagram.....	48
4.3. Desain Interface .....	50
4.3.1. Desain Interface Member.....	50
4.2.2. Desain Interface Mentor.....	62
BAB V .....	76
HASIL .....	76
5.1. Hasil Penelitian .....	76
5.2. Implementasi Antarmuka.....	76
5.3. Pengujian Black-Box Testing .....	98
BAB VI .....	110
PENUTUP .....	110
6.1. Kesimpulan .....	110



6.2. Saran .....	110
DAFTAR PUSTAKA .....	111
LAMPIRAN.....	113

## DAFTAR GAMBAR

Gambar 1. 1 Model Waterfall SDLC .....	6
Gambar 4. 1 Use Case Diagram Mentor .....	25
Gambar 4. 2 Use Case Diagram bember .....	27
Gambar 4. 3 Activity Diagram Login .....	28
Gambar 4. 4 Activity Diagram Login .....	28
Gambar 4. 5 Activity Diagram Show Card .....	29
Gambar 4. 6 Activity Diagram Batch .....	29
Gambar 4. 7 Activity Diagram Theory .....	30
Gambar 4. 8 Activity Diagram Task .....	30
Gambar 4. 9 Activity Diagram Info .....	31
Gambar 4. 10 Activity Diagram Event .....	31
Gambar 4. 11 Activity Diagram Absent .....	32
Gambar 4. 12 Activity Diagram Report .....	32
Gambar 4. 13 Activity Diagram User .....	33
Gambar 4. 14 Activity Diagram User Edit .....	33
Gambar 4. 15 Activity Diagram mentor absent .....	34
Gambar 4. 16 Activity Diagram Mentor Task .....	35
Gambar 4. 17 Activity Diagram Profile .....	36
Gambar 4. 18 Activity Diagram Profile Edit .....	36
Gambar 4. 19 Activity Diagram Log-out .....	37
Gambar 4. 20 Sequence Diagram Login .....	37

Gambar 4. 21 Sequence Diagram Member Absent.....	38
Gambar 4. 22 Sequence Diagram Member Card .....	38
Gambar 4. 23 Sequence Diagram Member Batch.....	39
Gambar 4. 24 Sequence Diagram Member Theory .....	39
Gambar 4. 25 Sequence Diagram Member Task .....	40
Gambar 4. 26 Sequence Diagram Member Event.....	40
Gambar 4. 27 Sequence Diagram Member Profile .....	41
Gambar 4. 28 Sequence Diagram Member Report .....	41
Gambar 4. 29 Sequence Diagram Mentor Task Beginner .....	42
Gambar 4. 30 Sequence Diagram Mentor Task Status .....	42
Gambar 4. 31 Sequence Diagram Mentor Task .....	43
Gambar 4. 32 Sequence Diagram Mentor Present Beginner .....	43
Gambar 4. 33 Sequence Diagram Mentor Present Status .....	44
Gambar 4. 34 Sequence Diagram Mentor absent.....	44
Gambar 4. 35 Sequence Diagram Mentor User .....	45
Gambar 4. 36 Sequence Diagram Logout.....	45
Gambar 4. 37 Class Diagram .....	46
Gambar 4. 38 Desain Layout Halaman Login .....	47
Gambar 4. 39 Desain Layout Halaman Home .....	48
Gambar 4. 40 Desain Layout Halaman Show Qr code .....	49
Gambar 4. 41 Desain Layout Halaman Show E card.....	50
Gambar 4. 42 Desain Layout Halaman Theory .....	51

Gambar 4. 43 Desain Layout Halaman Task .....	52
Gambar 4. 44 Desain Layout Halaman Info .....	53
Gambar 4. 45 Desain Layout Halaman All Event.....	54
Gambar 4. 46 Desain Layout Halaman Detail Event .....	55
Gambar 4. 47 Desain Layout Halaman Absen.....	56
Gambar 4. 48 Desain Layout Report.....	57
Gambar 4. 49 Desain Layout Halaman Account .....	58
Gambar 4. 50 Desain Layout Halaman Login .....	59
Gambar 4. 51 Desain Layout Halaman Absent.....	60
Gambar 4. 52 Desain Layout Halaman Theory Absent .....	61
Gambar 4. 53 Desain Layout Halaman Scan Member.....	63
Gambar 4. 54 Desain Layout Halaman Task .....	64
Gambar 4. 55 Desain Layout Halaman Task Week .....	65
Gambar 4. 56 Desain Layout Halaman Task Status .....	66
Gambar 4. 57 Desain Layout Halaman Task Result .....	67
Gambar 4. 58 Desain Layout Halaman Mentor .....	68
Gambar 5. 1 Tampilan Halaman Login.....	73
Gambar 5. 2 Tampilan halaman Home .....	74
Gambar 5. 3 Tampilan Halaman Show Qr code .....	75
Gambar 5. 4 Tampilan Halaman Show E-Card .....	76
Gambar 5. 5 Tampilan Halaman Theory.....	77
Gambar 5. 6 Tampilan Halaman Task .....	77

Gambar 5. 7 Tampilan Halaman Info .....	78
Gambar 5. 8 Tampilan Halaman All Event.....	79
Gambar 5. 9 Tampilan Halaman Detail Event .....	80
Gambar 5. 10 Tampilan Halaman Absent.....	81
Gambar 5. 11 Tampilan Halaman Report .....	82
Gambar 5. 12 Tampilan Halaman Account.....	83
Gambar 5. 13 Tampilan Halaman Login Mentor.....	84
Gambar 5. 14 Tampilan Halaman Absent.....	85
Gambar 5. 15 Tampilan Halaman Theory Absent .....	86
Gambar 5. 16 Tampilan Halaman Detail Absent .....	87
Gambar 5. 17 Tampilan Halaman pindai member .....	88
Gambar 5. 18 Tampilan Halaman Task .....	89
Gambar 5. 19 Tampilan Halaman Task Week .....	90
Gambar 5. 20 Tampilan Halaman Task Status.....	91
Gambar 5. 21 Tampilan Halaman Task Result .....	92
Gambar 5. 22 Tampilan Halaman Profile Mentor .....	93
Gambar 5. 23 Pengujian Halaman Login .....	95
Gambar 5. 24 Pengujian Halaman Pengumpulan Tugas.....	96

## DAFTAR TABEL

Tabel 2. 1 Perbandingan Jurnal.....	14
Tabel 3. 1 Simbol Use Case Diagram.....	21
Tabel 3. 2 Simbol Activity Diagram.....	23
Tabel 3. 3 Simbol Sequence Diagram.....	24
Tabel 3. 4 Simbol Class Diagram.....	25
Tabel 5. 1 Pengujian halaman Login.....	94
Tabel 5. 2 Pengujian Halaman Pengumpulan Tugas.....	95

## DAFTAR LAMPIRAN

Surat Kesepakatan Bimbingan Tugas Akhir .....	A-1
Lembar Bimbingan Tugas Akhir .....	B-1

## **BAB I**

### **PENDAHULUAN**

#### **1.1. Latar Belakang**

Pendidikan kini menjadi modal penting untuk meningkatkan dan mengembangkan kualitas sumber daya manusia. Dalam rangka meningkatkan kualitas pendidikan, diperlukan banyak terobosan baru, seperti pengembangan kurikulum, metode pembelajaran, bahan ajar dan fasilitas pembelajaran. Perkembangan dunia pendidikan yang sangat dinamis membutuhkan banyak inovasi Pendidikan [1].

Adopsi teknologi telah meningkat secara dramatis di semua bidang, termasuk pendidikan. Perkembangan teknologi di bidang teknologi juga secara otomatis mempengaruhi cara pandang terhadap proses pendidikan. Ketersediaan teknologi berupa perangkat keras dan perangkat lunak merupakan salah satu faktor yang mempengaruhi proses pembelajaran. Sumber daya manusia, waktu dan tempat pembelajaran juga dapat disesuaikan dengan ketersediaan teknologi. Siswa dapat menggunakan teknologi untuk mengakses pengetahuan untuk memperoleh informasi yang berkaitan dengan mata pelajaran.

Teknologi pendidikan harus mampu menjembatani keinginan yang baik antara guru dan siswa. Pesatnya perkembangan teknologi telah melahirkan banyak perangkat yang dapat dengan mudah digunakan untuk menjembatani kesenjangan antara guru dan siswa. Salah satu produk teknologi tersebut adalah



*smartphone*. *Smartphone* merupakan teknologi yang sudah menjadi alat komunikasi bagi setiap masyarakat. Hal ini dapat menimbulkan inovasi dalam penggunaan media oleh *smartphone* alias *mobile learning* [2].

Poltek Harber *Linux User Group*-Indonesia (PLUG-IN) adalah organisasi studi *Linux* dan pengembangan aplikasi di Politeknik Harapan Bersama Tegal. Fokus komunitas ini adalah mempelajari sistem operasi *Linux* dan pengembangan perangkat lunak seperti pengembangan aplikasi berbasis *mobile* dan *web*, desain *UI/UX*, serta jaringan komputer. Meskipun PLUG-IN telah melaksanakan beberapa program kerja, organisasi ini masih belum memiliki sistem keorganisasian yang terintegrasi dengan baik. Beberapa aspek yang perlu diperbaiki adalah sistem absensi, sistem penyampaian materi, informasi kas, dan informasi Acara.

Pada masa sekarang ini perangkat seluler tidak lagi sebuah barang yang langka. Berdasarkan persentase yang diperoleh dari *website* Portal Statistik, menunjukkan bahwa lebih dari 90% pengguna *smartphone* di Indonesia menggunakan *smartphone* dengan sistem operasi *Android* dibandingkan dengan *smartphone* dengan sistem operasi lainya [3].

Dengan data di lapangan, tahun lalu pengguna *smartphone android* meningkat lima kali. Secara tidak langsung, dipastikan banyak aplikasi yang direncanakan untuk *smartphone android*, termasuk aplikasi milik golongan pendidikan, namun masih banyak pengguna yang terkadang masih sulit untuk memahami dan menggunakan aplikasi yang ada [4].

Berdasarkan permasalahan yang telah dipaparkan maka dibuatlah (Aplikasi *e-plugin* berbasis *android*) yang dapat digunakan untuk membantu organisasi plugin dalam penyampaian pembelajaran secara cepat dan terpusat.

## **1.2. Perumusan Masalah**

Bagaimana membuat aplikasi *e-plugin* pada organisasi Plugin berbasis *android* untuk menyampaikan materi pembelajaran agar terpusat, untuk mempermudah monitoring kehadiran, dan tugas serta dapat menyampaikan informasi dalam satu aplikasi.

## **1.3. Pembahasan Masalah**

Dalam pelaksanaan penelitian ditetapkan beberapa batasan masalah sebagai berikut :

1. Aplikasi ini hanya dapat digunakan pada *smartphone* berbasis *Android* dengan spesifikasi sistem operasi minimum 5.0 (*Lollipop*)
2. Aplikasi hanya dapat mengirim jawaban tugas berupa *link url*
3. Aplikasi ini hanya digunakan oleh anggota plugin
4. Mentoring tugas, presensi dan perpindahan *level* anggota masih dilakukan secara manual oleh mentor.

#### 1.4. Tujuan Penelitian

Tujuan pada penelitian ini yaitu untuk menghasilkan aplikasi *e-plugin* di organisasi Plug-in berbasis *android*, yang dapat menjadikan aplikasi untuk memudahkan pengurus organisasi dalam hal pemantauan anggota, pemberian materi, dan pemberian tugas.

#### 1.5. Manfaat Penelitian

Manfaat yang diharapkan dari penelitian ini adalah:

##### 1. Manfaat bagi Pengguna

- a. Memudahkan pengurus organisasi PLUG-IN dalam menghimpun kehadiran anggota.
- b. Memudahkan anggota dalam melakukan absensi serta mempermudah dalam anggota untuk mengakses materi, pengumpulan tugas, dan mendapatkan informasi.
- c. Melihat perkembangan kemampuan intelektual setiap anggota PLUG-IN.

##### 2. Manfaat bagi Mahasiswa

- a. Menambah wawasan pengetahuan dan pengalaman mengenai pembuatan aplikasi berbasis *mobile*.
- b. Menerapkan ilmu yang telah didapatkan selama perkuliahan khususnya tentang *mobile programming* untuk diimplementasikan menjadi aplikasi yang bermanfaat bagi mahasiswa dan masyarakat.

### 3. Manfaat bagi Politeknik

Dengan adanya penelitian dan pembuatan Aplikasi *e-plugin* berbasis *android* dapat menjadi bahan dan kerangka acuan bagi mahasiswa dalam menyusun laporan Tugas Akhir (TA) serta tambahan pustaka mengenai penggalan permasalahan dalam lingkungan institusi Politeknik Harapan Bersama Tegal yang membutuhkan pemecahan masalah.

## 1.6. Metodologi Penelitian

### 1.6.1. Bahan Penelitian

Pada bahan penelitian yang digunakan dalam proses pembangunan aplikasi ini adalah sebagai berikut :

1. Data Anggota organisasi PLUG-IN
2. Data Keuangan organisasi PLUG-IN
3. Data Kehadiran organisasi PLUG-IN
4. Data Penilaian organisasi PLUG-IN
5. Data Acara organisasi PLUG-IN
6. Data Materi organisasi PLUG-IN
7. Data Tugas organisasi PLUG-IN

### 1.6.2. Alat Penelitian

Pada penelitian ini peralatan utama dan juga peralatan pendukung yang di gunakan pada saat perancangan *system* adalah :

## 1. *Software*

*Software* yang digunakan dalam pembuatan aplikasi ini diantaranya:

- a. *Linux Operating system*
- b. *Android studio*
- c. *Postman*
- d. *Figma*
- e. *Git*

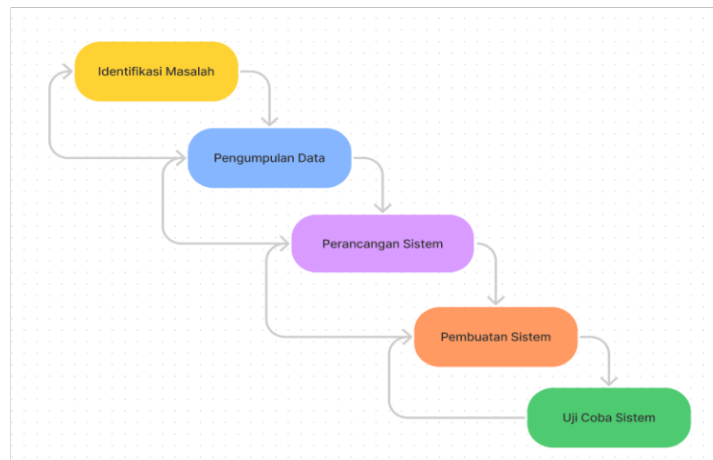
## 2. Hardware

*Hardware* atau perangkat keras yang digunakan dalam penelitian ini adalah :

- a. Laptop Thinkpad
- b. Memori 8,00 Gb
- c. *HDD 500 Gb*
- d. *Smartphone Android*

### **1.6.3. Alur penelitian**

Metode yang digunakan dalam pembangunan aplikasi ini adalah model *Waterfall*. Alur pengembangan aplikasi dimulai dari identifikasi masalah, metode pengumpulan data, analisis data, perancangan sistem, pembuatan sistem hingga tahap pengujian sistem. Alur *Waterfall* terdapat pada Gambar 1.1:



Gambar 1. 1 Model *Waterfall SDLC*

## 1. Identifikasi Masalah

Masalah yang diidentifikasi dalam penelitian ini adalah organisasi PLUG-IN saat ini belum mempunyai sistem pembelajaran *online* yang terintegrasi dengan baik. Sistem pembelajaran tersebut diantaranya seperti sistem absensi, sistem informasi, sistem pemberian materi, sistem informasi kas, dan informasi Acara.

## 2. Metode Pengumpulan Data

Pengumpulan data dalam penelitian ini dilakukan dengan dua metode yaitu observasi dan wawancara.

### a. Observasi

Dilakukan pengamatan dan mempelajari sumber-sumber yang dapat digunakan untuk pembuatan Aplikasi *E-plugin*, antara lain:

- Mengamati dan mempelajari proses pembelajaran pada organisasi PLUG-IN

- Mengambil data dari organisasi PLUG-IN.

#### b. Wawancara

Metode ini digunakan untuk mendapatkan informasi atau data yang diperlukan dengan menanyakan sumber yang relevan seperti pengurus plugin.

### 3. Analisa Data

Pada tahap ini dilakukan pengumpulan data dari hasil observasi dan wawancara untuk dijadikan acuan atas permasalahan yang diperoleh selama pembuatan sistem, dari mana sistem tersebut dapat dihasilkan.

### 4. Perancangan Sistem

Pada tahap ini dibutuhkan sebuah perancangan sistem dengan membuat skets yang menampilkan poin-poin penting menggunakan bahasa permodelan *UML*, karena untuk dokumentasi sistem yang akan dibuat sehingga memudahkan dalam membangun rancangan sistem.

### 5. Pembuatan Sistem

Aplikasi ini dibangun dengan bahasa pemrograman *Kotlin* dengan menggunakan *tools android studio* untuk kepentingan *development* aplikasi berbasis *mobile* dengan *platform android*.

### 6. Uji Coba Sistem

Pengujian adalah bagian penting dari pengembangan perangkat lunak. Pengujian dilakukan untuk memastikan kualitas dan juga untuk menemukan titik lemah dari perangkat lunak. Tujuan dari pengujian ini adalah untuk memastikan bahwa perangkat lunak yang dibangun sesuai

dengan apa yang telah dievaluasi dari segi spesifikasi, analisis, desain dan pengkodean perangkat lunak. Penelitian ini diuji menggunakan *Black-Box Testing*.

### **1.7. Sistematika Penulisan**

Sistematika yang digunakan dalam penyusunan laporan penelitian ini sebagai berikut:

#### **1. Bagian Awal Laporan**

Berupa Halaman Judul, Halaman Pernyataan, Halaman Rekomendasi, Halaman Pengesahan, Abstrak, Kata Pengantar, Daftar Isi, Daftar Tabel, Daftar Gambar, Daftar Lampiran.

#### **2. Bagian Isi Laporan**

##### **BAB I : PENDAHULUAN**

Dalam bab ini membahas tentang latar belakang diambilnya judul “APLIKASI *E-PLUGIN* BERBASIS *ANDROID* (Studi Kasus Organisasi Politeknik Harber *Linux User Group-Indonesia*)”, perumusan masalah, pembatasan masalah, tujuan, manfaat, metodologi penelitian dan sistematika penulisan.

##### **BAB II: TINJAUAN PUSTAKA**

Bab ini menjelaskan penelitian yang masalah dan solusinya dilakukan dan dibahas secara rinci, termasuk rincian konteks penelitian, tujuan, metode dan hasil.



### **BAB III: LANDASAN TEORI**

Dalam bab ini, kami membahas secara singkat teori yang relevan dan saling melengkapi untuk penulisan laporan terkait penelitian.

### **BAB IV: PERANCANGAN DAN DESAIN**

Pada bab ini terdapat penjelasan rinci mengenai perancangan dan perancangan penelitian yang dilakukan dari sistem yang dibangun berdasarkan data yang dihasilkan dalam perancangan antarmuka pengguna dan perancangan *UML*, perancangan ini meliputi: *Use Case Diagram*, *Activity Diagram*, *sequence diagram*, dan *Class Diagram*.

### **BAB V: HASIL DAN PEMBAHASAN**

Bab ini menjelaskan hasil yang diperoleh dari penelitian. Selain pengujian aplikasi, dilakukan untuk mengetahui bahwa aplikasi mampu menyelesaikan masalah yang dihadapi sesuai dengan yang diharapkan.

### **BAB VI: PENUTUP**

Bab ini menjelaskan tentang kesimpulan, yaitu kesimpulan dari pembahasan penelitian yang dilakukan, hasil akhir dan sekaligus jawaban atas pertanyaan yang beredar. Selain itu, saran juga disertakan sebagai pedoman dan pendapat yang mungkin berguna untuk penelitian sejenis dalam pengembangan penelitian lainnya

lebih lanjut mengenai *APLIKASI E-PLUGIN BERBASIS ANDROID* (Studi Kasus Organisasi Politeknik Harber *Linux User Group-Indonesia*).

### **DAFTAR PUSTAKA**

Daftar pustaka yang menjadi acuan penulisan laporan tugas akhir.

### **3. Bagian Akhir Laporan**

LAMPIRAN

## **BAB II**

### **TINJAUAN PUSTAKA**

Akhirudin Pulungan dan Alfa Saleh pada penelitiannya telah merancang aplikasi absensi berbasis *android* dengan judul “Perancangan Aplikasi Absensi Menggunakan *Qr code* Berbasis *Android*” yang memanfaatkan *Qr code* sebagai alat untuk membantu menandai kehadiran siswa, terdapat dua tampilan utama dimana ada tampilan untuk halaman guru sebagai *User* yang melakukan absensi, dan halaman siswa yang digunakan sebagai *User* yang di absen. Dimana pada halaman guru terdapat fitur laporan untuk melihat absensi siswa-siswi, dan pada halaman siswa terdapat fitur lihat absensi yang berguna untuk melihat hasil absensi [1].

Andi Lala dan Husni Rifqo dalam penelitiannya telah merancang aplikasi tentang pendaftaran anggota organisasi intra kampus pada Universitas Muhammadiyah Bengkulu. Dengan judul “Sistem Informasi Organisasi Intrakampus Pada Universitas Muhammadiyah Bengkulu”, Dimana tujuannya untuk mempermudah mahasiswa Universitas Muhammadiyah Bengkulu dalam melakukan pendaftaran masuk ke organisasi, dan bertujuan juga untuk mempermudah dalam pengelolaan data anggota mahasiswa yang telah menjadi anggota organisasi [3].

Abdur Rouf pada penelitiannya yang berjudul “Pembuatan Aplikasi Manajemen Organisasi Swara (Amora) Berbasis Desktop” menjelaskan tentang pengelolaan keanggotaan organisasi dan kegiatan program kerja yang diadakan

dalam penggunaan teknologi sangat penting untuk tetap up to date dalam pengelolaan data suatu organisasi, terutama di era digital saat ini. Perkembangan teknologi informasi sangat pesat. Setiap kali teknologi baru muncul, itu meningkatkan nilai manfaat dari teknologi sebelumnya. Berdasarkan semua kemungkinan manfaat teknologi komputer, peneliti beralih ke pembuatan aplikasi yang akan digunakan untuk mengelola anggota organisasi dan kegiatan dalam agenda. Pembuatan aplikasi ini menggunakan pemrograman *Java* dengan *database MySQL*. Pembuatan aplikasi ini mengikuti langkah-langkah *SDLC (System Development Life Cycle)* menggunakan model *Waterfall*, karena pengembangan aplikasi dilakukan secara linier. Langkah-langkah yang dilakukan untuk menerapkan dalam penelitian ini adalah analisis kebutuhan sistem, perancangan sistem dan pengkodean. Penelitian ini berhasil membuat SWARA Organization Management Application (AMORA) berbasis desktop yang dapat mengotomatisasi proses pencatatan data anggota organisasi, departemen dan kalender organisasi untuk memudahkan pengelolaan data organisasi *Event* [4].

Adrian Putranda Rispurwadi, R. Rizal Isnanto, dan Rinta Kridalukmana melakukan penelitian untuk membuat system informasi dan aplikasi UKM Bola Basket dengan judul “Pengembangan Aplikasi Android Pada UKM Basket Universitas Diponegoro”, yang dimana selama ini belum ada sistem informasi bagi organisasi kemahasiswaan bola basket di Universitas Diponegoro seperti aplikasi *mobile* yang menyediakan informasi, lokasi lapangan, dan *reminder* untuk latihan bola basket. Berdasarkan hal tersebut maka suatu aplikasi sistem informasi harus dibuat pada Sistem Operasi *Android*. Perancangan sistem informasi *mobile*

organisasi kemahasiswaan bola basket Universitas Diponegoro berbasis *android*, merupakan aplikasi *client server* yang dibuat dengan menggunakan *Java*, *PHP*, *jQuery* dan *MySQL* untuk database. Pengambilan koordinat lintang dan bujur akan memanfaatkan fitur *GPS* pada perangkat *mobile* dan akan ditampilkan pada *Google Maps API*. *JSON* akan mengumpulkan data jadwal ke perangkat seluler, terus menerima peringatan dialog dan penerima siaran untuk fitur alarm. Hasil dari perancangan ini adalah terwujudnya sebuah aplikasi yang mempermudah pengguna menemukan informasi tentang UKM Basket Universitas Diponegoro, dapat mempermudah pengguna dalam mencari lokasi lapangan menggunakan fitur *GPS* [5].

D.S Ramdan dan Syamardy Bangun Putra melakukan penelitian dengan judul “Perancangan dan Implementasi Sistem Informasi Manajemen Sistem Informasi Berbasis Web Untuk UKM (Unit Operasi Mahasiswa) di Politeknik TEDC Bandung”. Penelitian ini dilakukan untuk mengatasi masalah pengelolaan data yang tidak terintegrasi ke dalam penyampaian informasi UKM di Politeknik TEDC Bandung. Berdasarkan hasil survei terhadap mahasiswa di Politeknik TEDC Bandung, dapat disimpulkan bahwa mahasiswa di kampus mengalami kesulitan dalam mengumpulkan informasi tersebut. Hasil dari penelitian ini menghasilkan Sistem Informasi Pengelolaan Data Unit Kegiatan Mahasiswa (UKM) berbasis web yang memberikan kemudahan bagi pengguna untuk mengelola data dan informasi mahasiswa UKM kapan saja, dimana saja dan dimana saja [6].

Shuddha Chowdhury dan K. M. Salahuddin melakukan penelitian yang berjudul “*A Literature Review of Factors Influencing Implementation of*

*Management Information Systems in Organizations*”. Penelitian ini dilakukan untuk menentukan faktor-faktor dan membuat diskusi tentang Penerapan Sistem Informasi Manajemen (SIM) yang tepat dapat meningkatkan kinerja, produktivitas, dan kerja organisasi secara efisien. Tiga faktor memainkan peran utama ketika organisasi ingin menerapkan SIM di organisasinya. Faktor tersebut adalah faktor organisasi, faktor teknologi, dan faktor manajemen. Faktor-faktor ini saling terkait, dan bekerja secara terintegrasi. Ketika sebuah organisasi memutuskan untuk menerapkan SIM di organisasinya, mereka perlu melihat dengan cermat apakah tujuan penerapan SIM mereka selaras dengan tujuan organisasi mereka atau tidak. Manajemen puncak mereka harus mendukung rencana proyek, dan mereka juga melengkapi diri mereka dengan pengetahuan IT yang dibutuhkan. Mereka harus mengakui bahwa MIS yang dirancang dengan baik dapat membantu mereka secara signifikan dalam proses pengambilan keputusan. Mereka dapat mencapai tujuan organisasi mereka secara keseluruhan dan mereka juga dapat mengelola orang-orang mereka dengan dukungan Sistem Informasi Manajemen. Mereka mengalokasikan sumber daya yang diperlukan untuk implementasi MIS. Selain itu, mereka melatih penggunaanya dengan pengetahuan yang diperlukan sehingga efisiensi organisasi mereka akan meningkat secara signifikan. Sebagai kesimpulan, ketiga faktor ini bersama-sama memainkan peran penting dalam implementasi SIM, dan organisasi akan memperoleh banyak manfaat dengannya [7].

Wahyudi pada penelitiannya yang berjudul “Pengembangan Aplikasi Sistem *E-Ticketing* Konser Musik *Denganseating* Number Pada *Website Motikdong.Com*”, melakukan pengembangan sistem pada penjualan tiket konser

musik dan media streaming yang berbasis *website* dimana Di Tengah wabah covid19 ini, salah satu usaha yang paling berdampak adalah hiburan, yang salah satu diantaranya adalah konser musik. Motikdong.com yang konsennya adalah menjual tiket konser musik sangatlah berdampak dari peristiwa covid19 ini, oleh karena itu penulis akan melakukan penelitian untuk menemukan solusi agar penjualan tiket bisa tetap dilakukan oleh pihak motikdong.com dan dapat memberikan solusi atas usaha penjualan tiket konser di tengah wabah covid19 ini. Penelitian ini bertujuan untuk mengimplementasikan penjualan tiket konser, dimana pihak motikdong dan client nya agar bisa menyelenggarakan konser musik secara streaming pada *website* motikdong.com [8].

Tabel 2. 1 Perbandingan Jurnal

No	Judul	Keterangan	Perbedaan
1.	Perancangan Aplikasi Absensi Menggunakan <i>Qr code</i> Berbasis <i>Android</i>	1. Menggunakan bahasa pemrograman <i>java</i> 2. Pengolahan datanya menggunakan <i>Firestore</i>	1. Menggunakan bahasa pemrograman <i>Kotlin</i> 2. pengolahannya menggunakan <i>Rest API</i>
2.	Sistem Informasi Organisasi Interaksi kampus Pada Universitas	1. <i>Platform</i> yang menjadi studi kasus berbasis <i>Website</i> 2. Aplikasi ini hanya	1. <i>Platform</i> yang menjadi studi kasus berbasis <i>android</i>

	Muhammadiyah Bengkulu	untuk pendaftaran pada organisasi	
3.	Pembuatan Aplikasi Manajemen Organisasi Swara (Amora) Berbasis <i>Desktop</i>	1. <i>Platform</i> yang menjadi studi kasus berbasis <i>desktop</i>	1. <i>Platform</i> yang menjadi studi kasus berbasis <i>android</i>
4.	Pengembangan aplikasi <i>Android</i> Pada UKM Basket Universitas Diponegoro	1. Studi kasus yang terjadi pada ukm bola basket 2. Studi kasus yang terjadi berbasis <i>website</i>	1. Studi kasus yang terjadi pada organisasi belajar 2. Studi kasus yang terjadi berbasis <i>android</i>
5.	Perancangan dan Implementasi Aplikasi Sistem Informasi Pengelolaan Data UKM (Unit Kegiatan	1. <i>Platform</i> yang digunakan pada studi kasus yang terjadi berbasis <i>website</i> 2. Menggunakan bahasa pemrograman <i>PHP</i>	1. <i>Platform</i> yang digunakan pada studi kasus yang terjadi berbasis <i>android</i> 2. Menggunakan bahasa pemrograman <i>Kotlin</i>



	Mahasiswa) Berbasis Web di Politeknik TEDC Bandung		
6.	<i>A Literature Review of Factors Influencing Implementation of Management Information Systems in Organizations</i>	1. Penelitian tentang penyelarasan pada suatu organisasi	1. Pembuatan sistem untuk mengontrol keorganisasian
7.	Pengembangan Aplikasi Sistem E- <i>Ticketing</i> Konser Musik <i>Denganseating</i> Number Pada <i>Website</i> <i>Motikdong.Com</i>	1. Penelitian tentang Penjualan dan <i>streaming</i> <i>Event</i> 2. <i>Platform</i> yang digunakan pada penelitian tersebut berbasis <i>website</i>	1. Penelitian tentang sistem keorganisasian 2. <i>Platform</i> yang digunakan pada penelitian tersebut berbasis <i>android</i> 3. menggunakan <i>payment gateway</i> pada pembayaran <i>Event</i>

## BAB III

### LANDASAN TEORI

#### 3.1. Komunitas Plugin

Plugin adalah organisasi yang berfokus di bidang teknologi informasi dan *development* program aplikasi baik itu aplikasi berbasis *website*, *android*, maupun jaringan. Pada komunitas plugin terbagi menjadi empat divisi, yang pertama ada divisi *website* yang berfokus pada pengembangan teknologi baik itu *frontend* maupun *backend*. Kedua divisi *mobile* yang berfokus pada pengembangan teknologi berbasis aplikasi *android*, yang ketiga ada divisi *networking* atau jaringan, yang berfokus pada pengembangan suatu jaringan dan server. Dan yang terakhir ada divisi *UI/UX* yaitu divisi yang berfokus pada riset untuk kenyamanan pengguna yang bertujuan untuk di implementasikannya suatu ide atau gagasan dari desain yang ada.

#### 3.2. Aplikasi

Aplikasi adalah penggunaan dalam perangkat komputasi instruksi (*instructions*) atau instruksi (*instructions*) yang disusun sedemikian rupa sehingga komputer dapat memproses masukan (*input*) menjadi keluaran (*output*). Sedangkan menurut Kamus Besar Bahasa Indonesia, aplikasi adalah aplikasi yang mendesain suatu sistem untuk mengolah data dengan menggunakan aturan-aturan atau regulasi bahasa pemrograman tertentu. Meskipun pengembangan aplikasi sering disalahartikan sebagai kegiatan di mana seorang programmer melakukan pengkodean, proses penulisan kode, arti

sebenarnya dari pengembangan aplikasi adalah serangkaian proses yang dilakukan dari desain aplikasi hingga aplikasi selesai dan siap. menggunakan. Selama pengembangan aplikasi, proses yang berbeda perlu dijalankan untuk membuat aplikasi yang dirancang dengan baik dan terstruktur. Proses pengembangan aplikasi terdiri dari 5 tahapan utama, meskipun dalam setiap tahapan tersebut terdapat tahapan yang berbeda yang dapat diimplementasikan tergantung pada kebutuhan pengembangan. Fase-fase ini meliputi: pengumpulan informasi, perencanaan, pengembangan, penggunaan dan pemeliharaan. Aplikasi dapat diklasifikasikan menjadi beberapa kelas, antara lain:

1. Aplikasi perusahaan (*enterprise*)
2. Aplikasi infrastruktur perusahaan
3. Aplikasi informasi kerja
4. Aplikasi media dan hiburan
5. Aplikasi pendidikan
6. Aplikasi pengembangan media
7. Aplikasi rekayasa produk

### **3.3. *Android***

*Android* adalah sistem operasi *open source*, didasarkan pada satu *kernel Linux* yang dimodifikasi dan dimiliki oleh *Google*. *Google* bertanggung jawab atas pengembangan bersama dengan *Open Hand Alliance (OHA)*. Sistem operasi awalnya dianggap sebagai *platform* untuk digital kamera ketika pengembangan oleh *Android Inc.* dimulai pada tahun 2005. Namun, *Google*

membeli perusahaan dan tiga tahun kemudian *smartphone* pertama yang menjalankan *Android* dijual. Pada 2015, hampir 83% pasar ponsel menggunakan *Android* dan 1,5 miliar 1 aplikasi diunduh oleh pengguna *Android* setiap bulan [10].

### 1. *Arsitektur*

Sistem operasi *Android* adalah tumpukan komponen perangkat lunak dan seperti yang ditunjukkan pada gambar 2.1, the *Arsitektur Android* terdiri dari empat lapisan: *Linux kernel*, *Libraries* dan *Android runtime*, Kerangka Aplikasi dan Aplikasi.

### 2. *Kernel Linux*

*Android* didasarkan pada kernel *Linux* 2.6 yang dimodifikasi untuk berjalan di telepon. Seperti banyak sistem operasi yang berfungsi sebagai penyimpanan internal, protokol internet, proses manajemen, manajemen perangkat, dan layanan inti lainnya. OS *Android* berinteraksi dengan perangkat keras perangkat pada lapisan ini dan modul dan *driver* sebagian besar ditulis dalam bahasa pemrograman C.

### 3. *Library dan Runtime Android*

Di atas *Linux*, lapisan kernel adalah pustaka asli dan modul runtime *Android*. *Library native* merupakan library yang digunakan oleh aplikasi dan merupakan tautan penting yang menghubungkan aplikasi ke fitur yang ditawarkan *Android*. Perpustakaan ditulis dalam C atau C ++ bahasa dan dipanggil melalui antarmuka Java yang memungkinkan mengakses varietas yang berbeda data atau audio di telepon.

*Android* runtime adalah kombinasi dari satu set library native dan ART (*Android Run Time*) dan telah menggantikan Dalvik Virtual Machine. Java Core Library menyediakan paling banyak fungsi esensial dan ART digunakan untuk menjalankan aplikasi. Dalvik VM adalah Java Virtual Machine yang dimodifikasi dan berbasis register dan dioptimalkan untuk berjalan di ponsel karena memiliki daya dan kebutuhan memori yang rendah. Selain itu DVM mengizinkan aplikasi untuk dijalankan dalam mesinnya sendiri. Ini menghasilkan sistem yang dapat diandalkan sejak itu DVM memungkinkan beberapa mesin virtual dibuat secara bersamaan menyediakan isolasi, manajemen memori dan dukungan untuk threading. Mesin Virtual Dalvik menjalankan file .dex yang merupakan transformasi dari file .class dan .jar dan dijalankan dengan sumber daya yang lebih rendah. Namun, di *Android* 5.0, DVM sepenuhnya digantikan oleh ART sekarang runtime yang dikelola dan secara khusus dibuat untuk *Android*.

Transformasi ART bytecode, kompilasi ke instruksi asli dan kemudian dieksekusi di lingkungan runtime. Ini menggunakan .dex file yang sama dengan Dalvik tetapi menggantikan. Odex - file dengan Executable dan Executable Formatable Linkable (ELF). ELF ini adalah satu-satunya file yang dieksekusi ketika aplikasi dikompilasi yang menghasilkan penghapusan overhead eksekusi aplikasi yang berbeda. ART menggunakan kompilasi sebelumnya (AOH) yang mengkompilasi seluruh aplikasi ke mesin kode ketika aplikasi diinstal. Ini antara lain meningkatkan efisiensi eksekusi, alokasi memori dan mengurangi konsumsi daya.

#### 4. Kerangka Kerja Aplikasi

Kerangka kerja aplikasi menawarkan abstraksi dari perpustakaan asli yang digabungkan dengan kemampuan Dalvik. Selain itu, ia menyediakan antarmuka pemrograman aplikasi (API) dan layanan tingkat tinggi lainnya dalam bentuk kelas Java. Satu blok penting dari lapisan kerangka kerja aplikasi adalah Manajer Aktivitas yang menangani siklus hidup aplikasi.

#### 5. Aplikasi

Lapisan aplikasi adalah lapisan teratas dalam arsitektur dan merupakan lapisan tempat aplikasi berada. Aplikasi dapat diinstal sebelumnya dan menyediakan fungsionalitas dasar telepon sebagai membuat panggilan telepon atau menjelajahi internet tetapi lapisan aplikasi juga menangani yang sudah diunduh atau dalam pengembangan. Aplikasi ini ditulis sepuluh di Jawa dan selanjutnya dikompilasi ke dalam kode mesin untuk diinstal pada smartphone.

### 3.4. *Kotlin*

*Kotlin* adalah bahasa pemrograman berbasis Java Virtual Machine (JVM). *Kotlin* adalah bahasa pemrograman pragmatis untuk *Android* yang menggabungkan bahasa berorientasi objek (OO) dan fungsional. *Kotlin* juga merupakan bahasa pemrograman interoperable yang memungkinkan untuk digabungkan dalam sebuah proyek dengan bahasa pemrograman Java. Bahasa pemrograman ini juga dapat digunakan untuk mengembangkan aplikasi desktop, web, dan backend.

*Kotlin* awalnya dikembangkan oleh JetBrains, perusahaan di balik IntelliJ IDEA. Setelah melalui banyak pengembangan, JetBrains telah merilis *Kotlin* open source dan perkembangannya sekarang semakin maju. Google sepenuhnya mendukung *Kotlin* untuk pengembang aplikasi *Android*



### 3.5. Unified Modelling Language (UML)


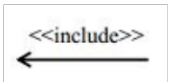
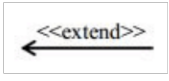



UML adalah bahasa visual untuk pemodelan dan komunikasi pada sistem menggunakan diagram dan teks pendukung. UML dimaksudkan untuk menentukan, mendeskripsikan, membangun, dan mendokumentasikan sistem perangkat lunak.

#### 1.1.1. Use Case Diagram


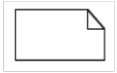
*Use Case Diagram* merupakan pemodelan untuk kelakuan system informasi yang akan dibuat. Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan informasi yang akan dibuat. Simbol-simbol yang ada pada *Use Case Diagram* terdapat pada Tabel 3.1:

Tabel 3. 1 Simbol *Use Case Diagram*

No	Simbol	Nama	Keterangan
1.		<i>Actor</i>	Menentukan peran yang dimainkan pengguna saat berinteraksi dengan use case.
2.		<i>Dependency</i>	Perubahan yang terjadi pada elemen independen akan

			mempengaruhi hubungan elemen yang bergantung pada elemen dependen.
3.		<i>Generalization</i>	Hubungan di mana objek turunan berbagi perilaku dan struktur data objek di atas objek induk (leluhur).
4.		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
5.		<i>Extend</i>	Menentukan perilaku kasus penggunaan target untuk memperluas kasus penggunaan sumber pada titik tertentu.
6.		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7.		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8.		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.








9.		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemennya (sinergi).
10.		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

### 3.6. Activity Diagram

*Activity Diagram* atau diagram aktivitas menggambarkan alur kerja (workflow) atau sistem atau aktivitas menu dalam perangkat lunak. Simbol-simbol dalam diagram aktivitas ditunjukkan pada Tabel 3.2 :

Tabel 3. 2 Simbol *Activity Diagram*

No	Simbol	Nama	Keterangan
1.		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2.		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3.		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4.		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5.		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

## BAB IV

### RANCANGAN DAN DESAIN

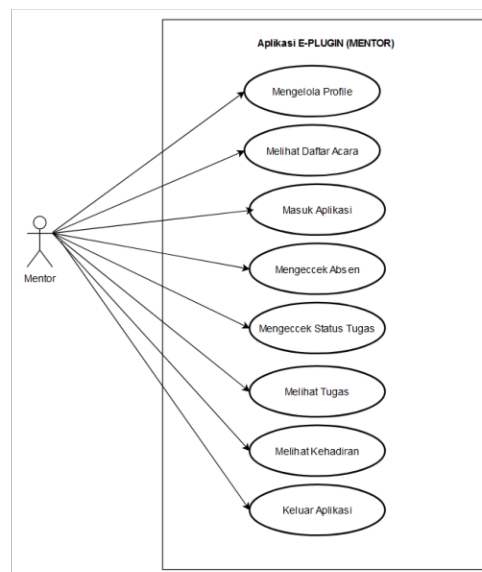
#### 4.1. Perancangan Sistem

Perancangan sistem ini bertujuan untuk mengetahui hubungan antar komponen pendukung sistem yang akan dirancang untuk memberikan gambaran kepada pengguna sistem tentang informasi yang dihasilkan oleh perancangan sistem. Tata letak ini dirancang dalam *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, dan *Class Diagram*.

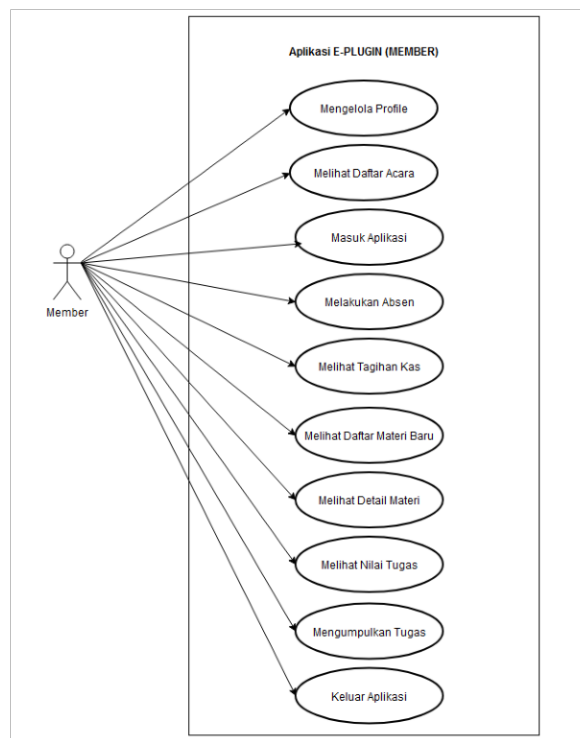
#### 4.2. Perancangan UML

##### 4.2.1. Use Case Diagram

Pada perancangan aplikasi e-plugin terdapat 2 aktor yaitu sebagai mentor dan member. Kegiatan dari aktor telah digambarkan ke dalam *Use Case Diagram* pada Gambar 4.1 dan Gambar 4.2 sebagai berikut:



Gambar 4. 1 *Use Case Diagram Mentor*



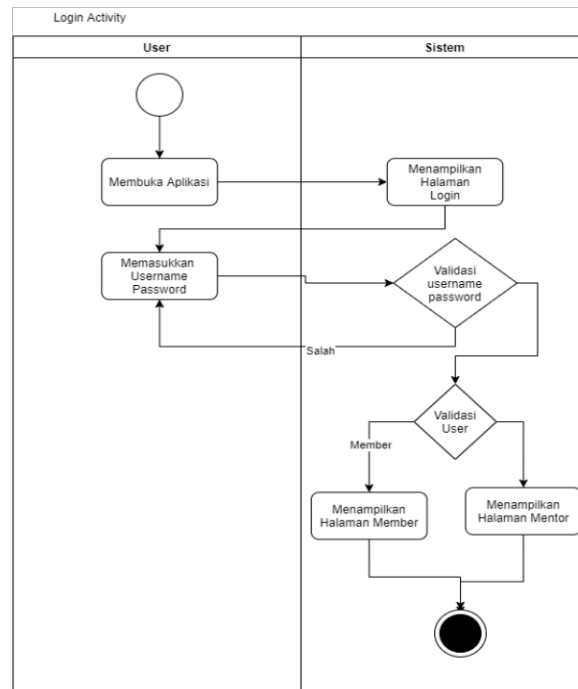
Gambar 4. 2 *Use Case Diagram Member*

#### 4.2.2. *Activity Diagram*

Diagram aktivitas adalah teknik untuk menggambarkan logika *prosedural*, proses bisnis, alur kerja dalam banyak kasus. Diagram aktivitas memiliki peran yang mirip dengan diagram alur. Berikut adalah diagram cara kerja setiap *use case*.

##### 1. *Activity Diagram Login*

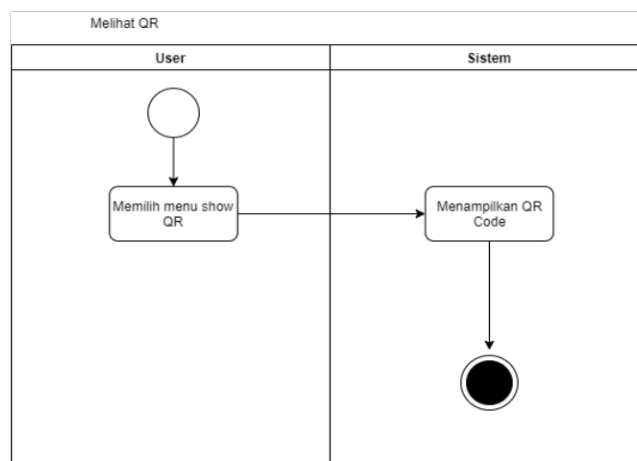
*Activity Diagram Login* menjelaskan aktivitas dari proses *User* masuk pada sistem. Sistem akan mengecek apakah *email* dan *password* yang dimasukan ada atau tidak, jika benar makan akan menuju aktivitas. Selanjutnya. *Activity Diagram Login* terdapat pada Gambar 4.3



Gambar 4.3 Activity Diagram Login

## 2. Activity Diagram Show QR

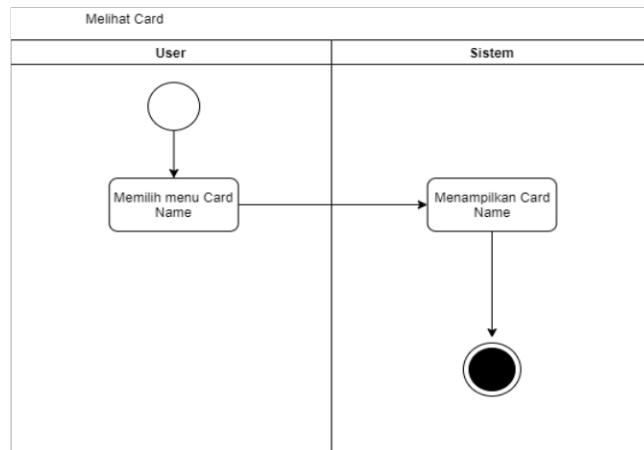
Activity Diagram Show QR menjelaskan aktivitas User menampilkan QR Code. Activity Diagram Show QR terdapat pada Gambar 4.4.



Gambar 4.4 Activity Diagram Login

### 3. Activity Diagram Show Card

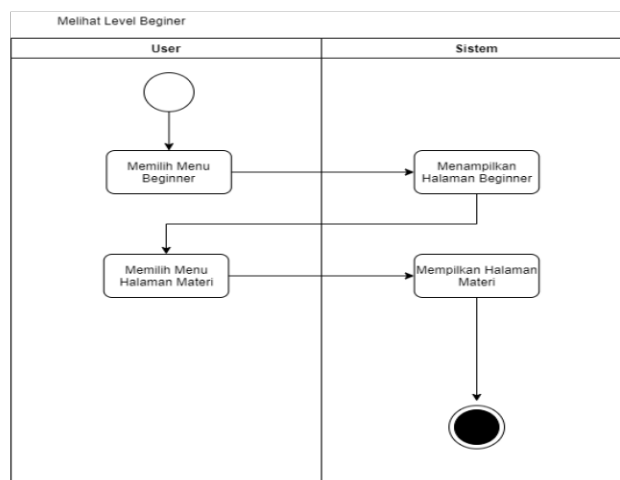
*Activity Diagram Show Card* menjelaskan aktivitas dari proses untuk menampilkan *card*. *Activity Diagram Show Card* terdapat pada Gambar 4.5.



Gambar 4. 5 *Activity Diagram Show Card*

### 4. Activity Diagram Batch

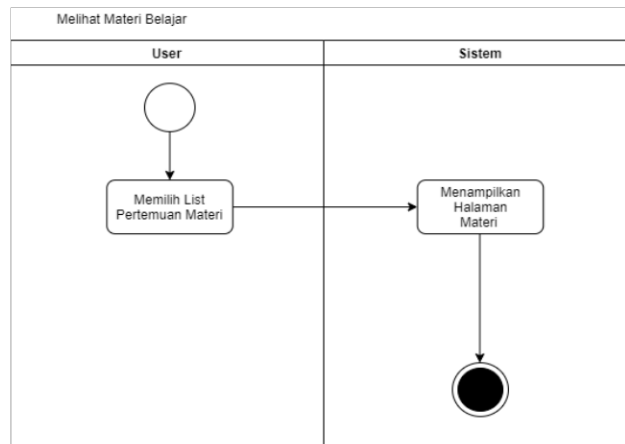
*Activity Diagram Batch* menjelaskan aktivitas dari proses untuk melihat *Batch*. *Activity Diagram Batch* terdapat pada Gambar 4.6.



Gambar 4. 6 *Activity Diagram Batch*

## 5. Activity Diagram Theory

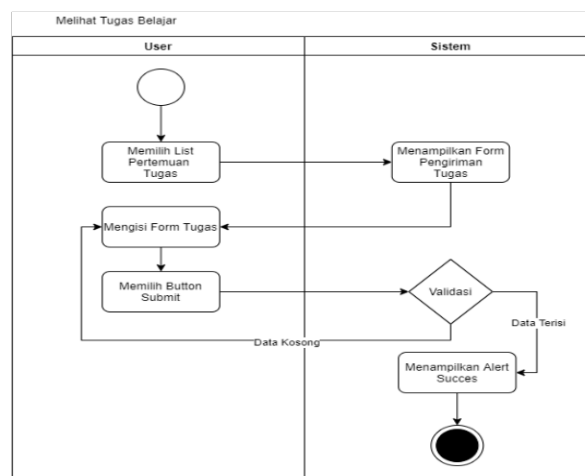
*Activity Diagram Theory* menjelaskan aktivitas dari proses untuuk melihat *Theory*. *Activity Diagram Theory* terdapat pada Gambar 4.7.



Gambar 4. 7 Activity Diagram Theory

## 6. Activity Diagram Task member

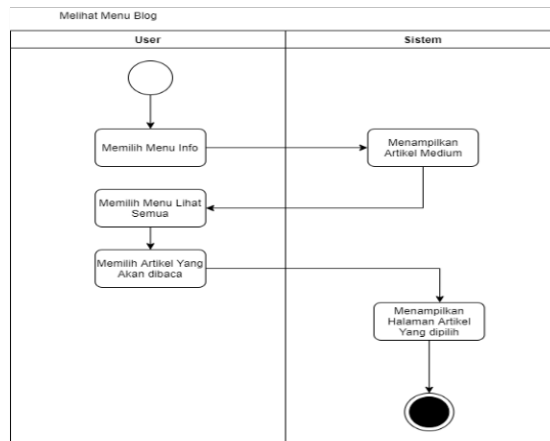
*Activity Diagram task* menjelaskan aktivitas dari proses *User* mengisi atau mengumpulkan tugas. *Activity Diagram task* terdapat pada Gambar 4.8.



Gambar 4. 8 Activity Diagram Task

## 7. Activity Diagram Info

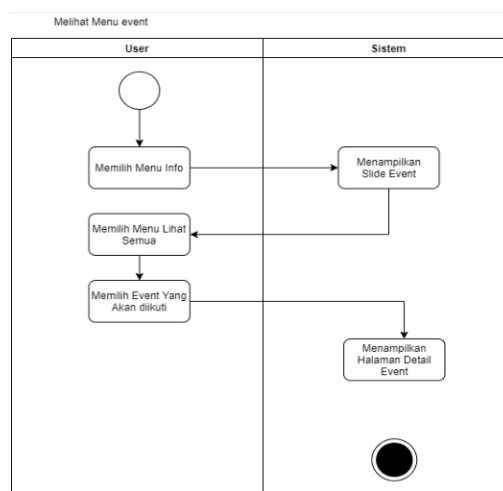
*Activity Diagram* info menjelaskan aktivitas dari proses untuk melihat info menu artikel. *Activity Diagram* info terdapat pada Gambar 4.9.



Gambar 4. 9 Activity Diagram Info

## 8. Activity Diagram Event

*Activity Diagram Event* menjelaskan aktivitas dari proses untuk melihat Event. *Activity Diagram Event* terdapat pada Gambar 4.10.

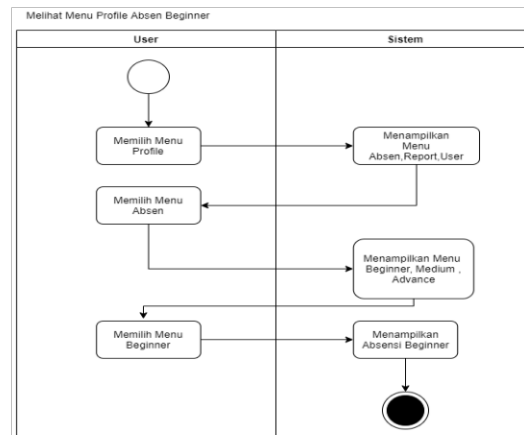


Gambar 4. 10 Activity Diagram Event



### 9. Activity Diagram Absent

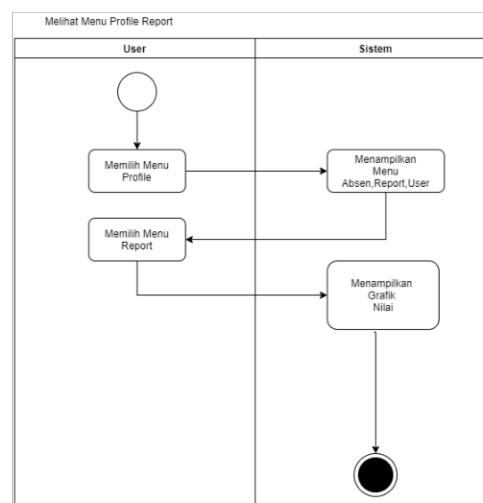
*Activity Diagram absent* menjelaskan aktivitas proses *User* melakukan *absent* pada system. *Activity Diagram absent* terdapat pada Gambar 4.11.



Gambar 4. 11 *Activity Diagram Absent*

### 10. Activity Diagram Report

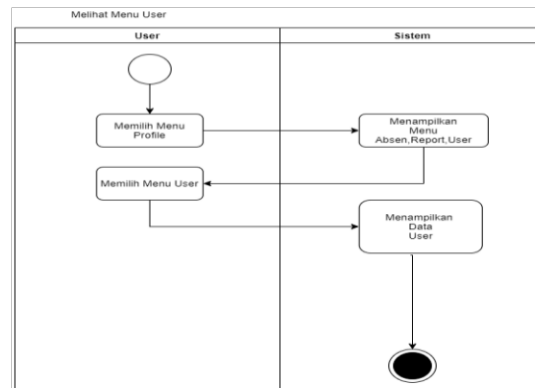
*Activity Diagram Report* menjelaskan aktivitas dari proses melihat halaman *Report*. *Activity Diagram Report* terdapat pada Gambar 4.12.



Gambar 4. 12 *Activity Diagram Report*

## 11. Activity Diagram User

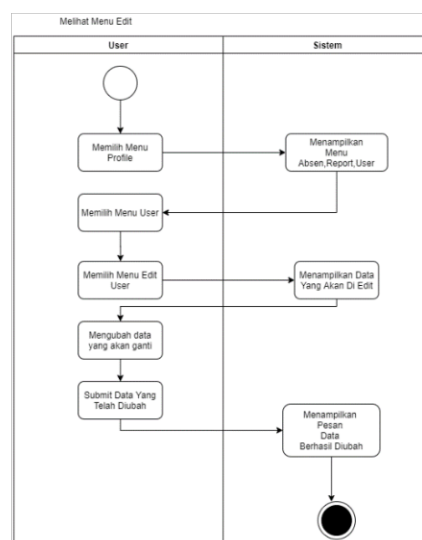
*Activity Diagram User* menjelaskan aktivitas dari proses untuk melihat detail data *User*. *Activity Diagram User* terdapat pada Gambar 4.13.



Gambar 4. 13 *Activity Diagram User*

## 12. Activity Diagram User Edit

*Activity Diagram User edit* menjelaskan aktivitas proses untuk merubah/edit data *User*. *Activity Diagram User edit* terdapat pada Gambar 4.14.

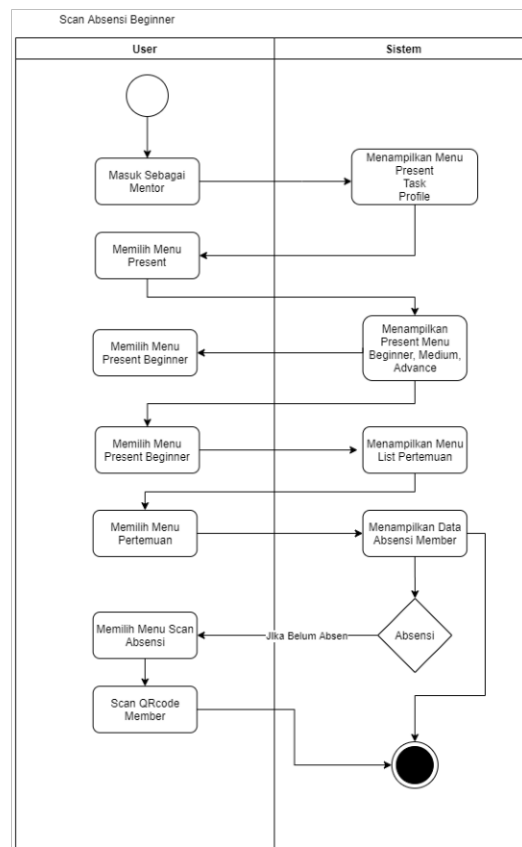


Gambar 4. 14 *Activity Diagram User Edit*

### 13. Activity Diagram Mentor Absent

*Activity Diagram* mentor *absent* menjelaskan aktivitas dari proses *User* melakukan *absent* kepada member dan melihat data member yang hadir.

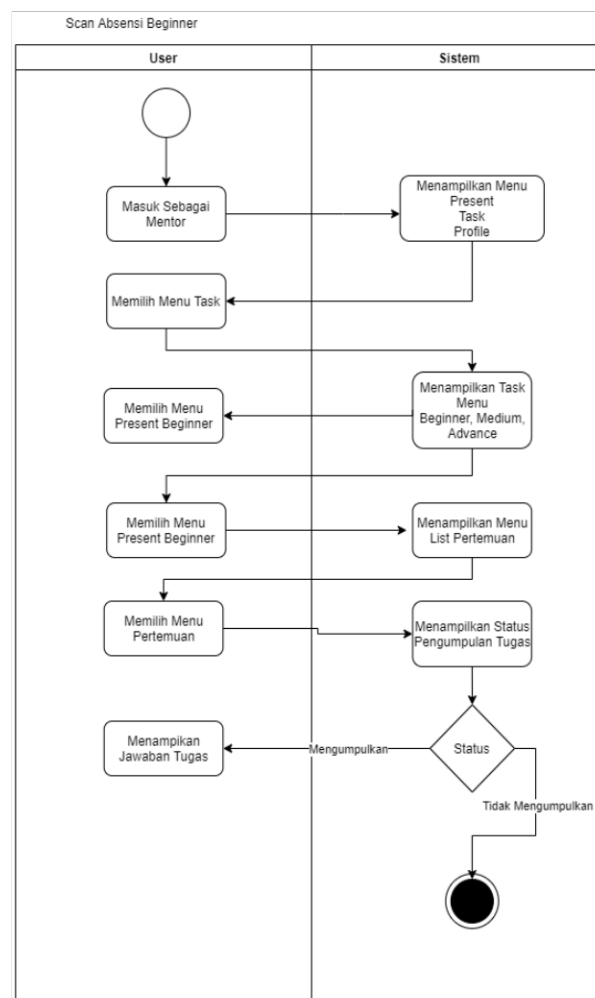
*Activity Diagram* mentor *absent* terdapat pada Gambar 4.15.



Gambar 4. 15 *Activity Diagram* mentor *absent*

#### 14. Activity Diagram Mentor Task

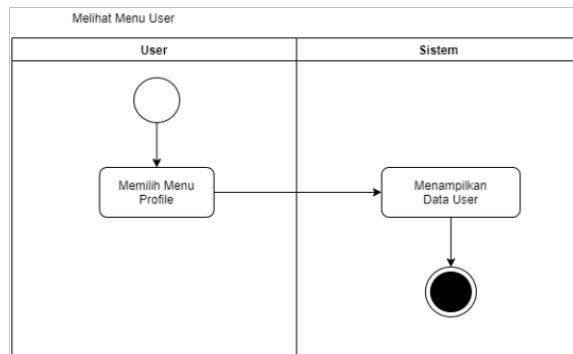
*Activity Diagram* mentor task menjelaskan aktivitas dari proses untuk mentor melihat *list* data pengumpulan tugas. *Activity Diagram* mentor task terdapat pada Gambar 4.16.



Gambar 4. 16 Activity Diagram Mentor Task

### 15. Activity Diagram Profile

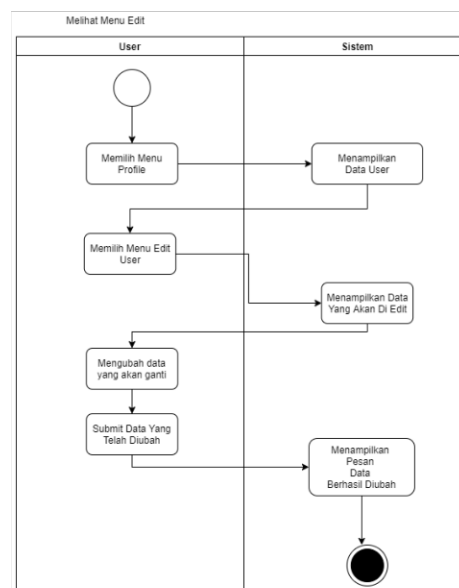
*Activity Diagram* profile menjelaskan aktivitas dari proses untuk melihat detail data *User*. *Activity Diagram* profile terdapat pada Gambar 4.17.



Gambar 4. 17 *Activity Diagram* Profile

### 16. Activity Diagram Profile Edit

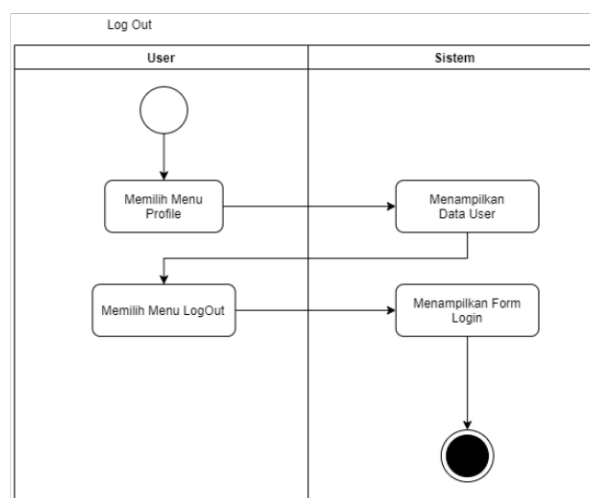
*Activity Diagram* profile edit menjelaskan aktivitas proses *User* melakukan edit data *User*. *Activity Diagram* profile edit terdapat pada Gambar 4.18.



Gambar 4. 18 *Activity Diagram* Profile Edit

### 17. Activity Diagram Log-out

*Activity Diagram* log-out menjelaskan aktivitas dari proses *User* melakukan log-out atau keluar pada sistem. Sistem akan menghapus session ketika *User* melakukan log-out. *Activity Diagram* log-out terdapat pada Gambar 4.19.

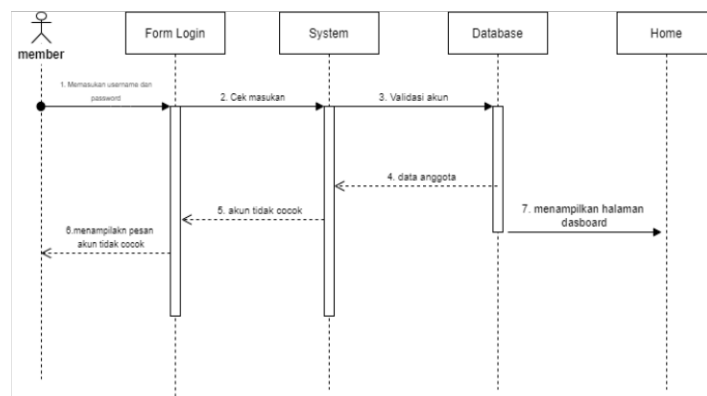


Gambar 4. 19 *Activity Diagram* Log-out

### 4.2.3. Sequence Diagram

#### 1. Sequence Diagram Login

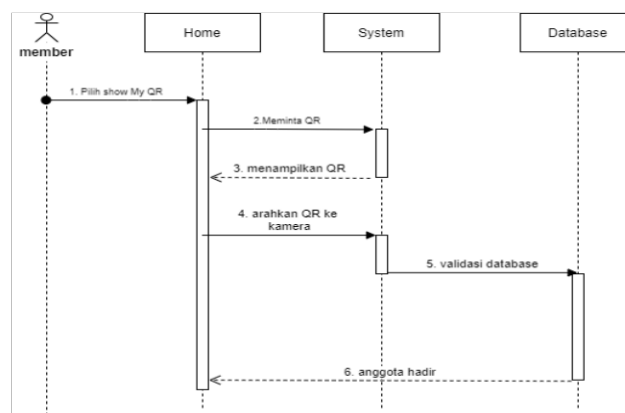
*Sequence Diagram Login* menjelaskan alur aktor ketika melakukan sign-in atau masuk pada aplikasi. *Sequence Diagram Login* terdapat pada Gambar 4.20



Gambar 4. 20 *Sequence Diagram Login*

#### 2. Sequence Diagram Member Absent

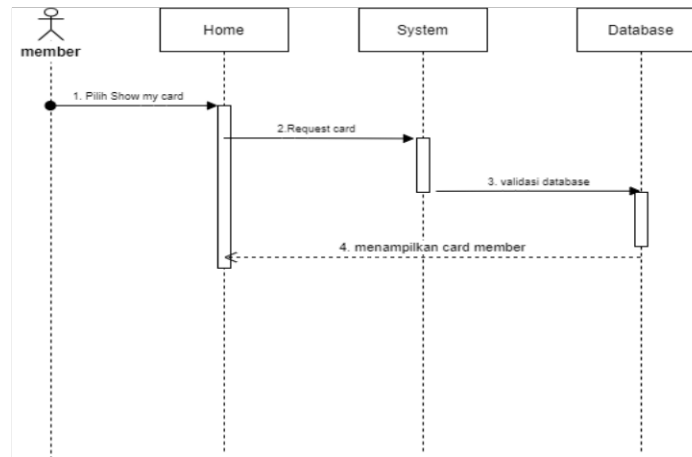
*Sequence Diagram member absent* menjelaskan alur aktor ketika akan melakukan *absent*. *Sequence Diagram member absent* terdapat pada Gambar 4.21



Gambar 4. 21 *Sequence Diagram Member Absent*

### 3. *Sequence Diagram Member Card*

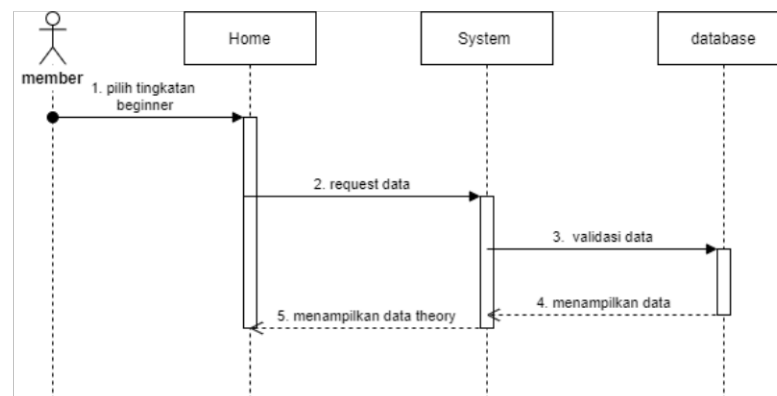
*Sequence Diagram member card* menjelaskan alur aktor ketika melihat kartu/card. *Sequence Diagram member card* terdapat pada Gambar 4.22.



Gambar 4. 22 *Sequence Diagram Member Card*

### 4. *Sequence Diagram Member Batch*

*Sequence Diagram member Batch* menjelaskan alur aktor ketika melihat *Batch*. *Sequence Diagram member Batch* terdapat pada Gambar 4.23

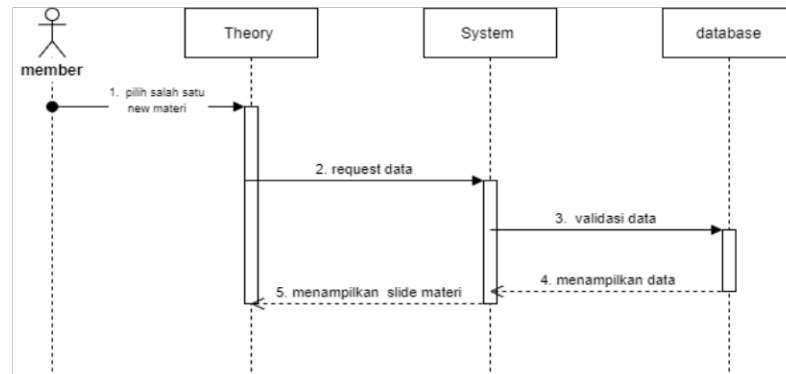


Gambar 4. 23 *Sequence Diagram Member Batch*



### 5. Sequence Diagram Member Theory

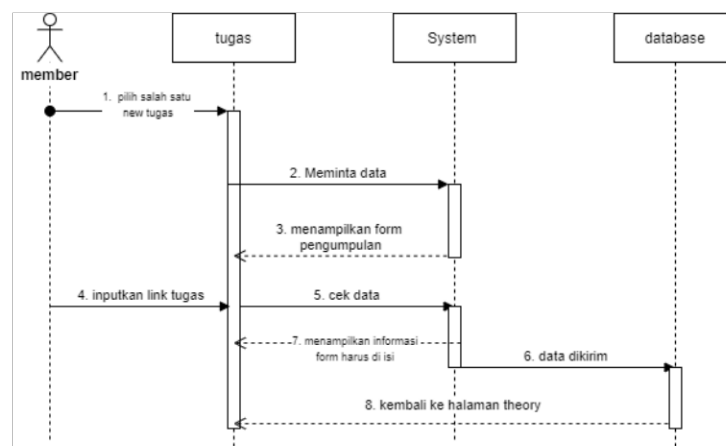
*Sequence Diagram member Theory* menjelaskan alur aktor ketika melihat *Theory*. *Sequence Diagram member Theory* terdapat pada Gambar 4.24.



Gambar 4. 24 *Sequence Diagram Member Theory*

### 6. Sequence Diagram Member Task

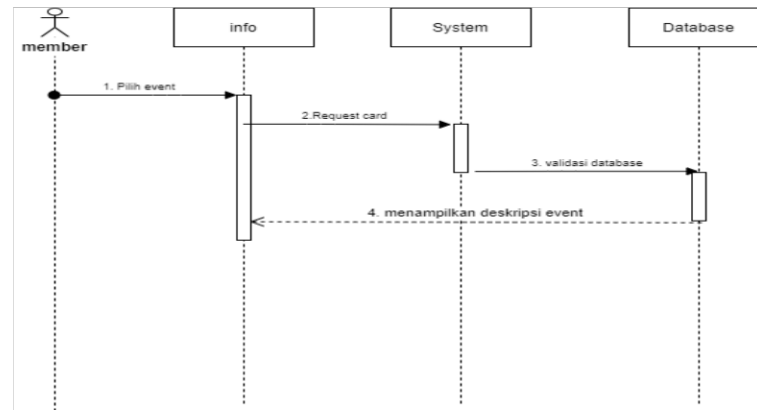
*Sequence Diagram member task* menjelaskan alur aktor ketika melakukan pengumpulan tugas. *Sequence Diagram member task* terdapat pada Gambar 4.25.



Gambar 4. 25 *Sequence Diagram Member Task*

### 7. Sequence Diagram Member Event

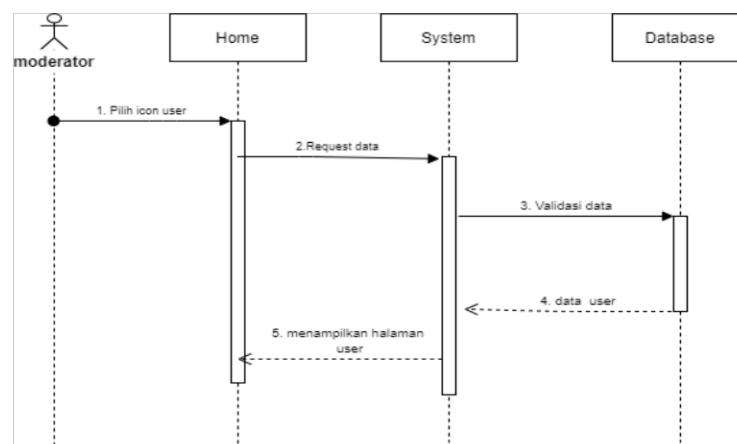
*Sequence Diagram member Event* menjelaskan alur aktor ketika melihat *Event*. *Sequence Diagram member Event* terdapat pada Gambar 4.26.



Gambar 4. 26 *Sequence Diagram Member Event*

### 8. Sequence Diagram Member Profile

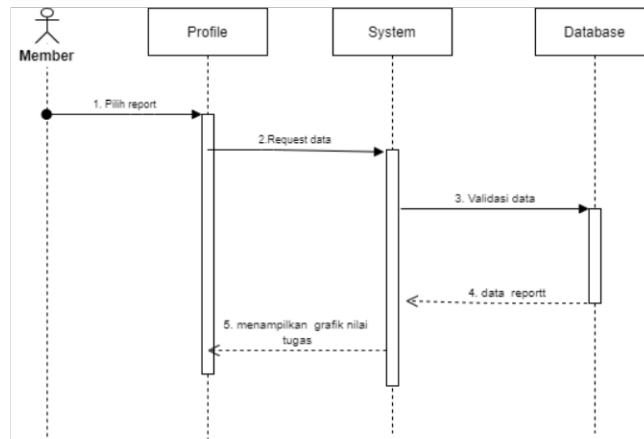
*Sequence Diagram member profile* menjelaskan alur aktor ketika melihat profile. *Sequence Diagram member profile* terdapat pada Gambar 4.27.



Gambar 4. 27 *Sequence Diagram Member Profile*

### 9. Sequence Diagram Member Report

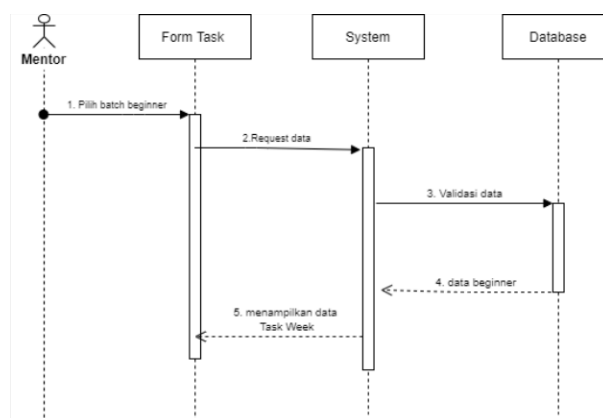
*Sequence Diagram* member *Report* menjelaskan alur aktor ketika melihat data *Report*. *Sequence Diagram* member *Report* terdapat pada Gambar 4.28



Gambar 4. 28 *Sequence Diagram* Member *Report*

### 10. Sequence Diagram Mentor Task Beginner

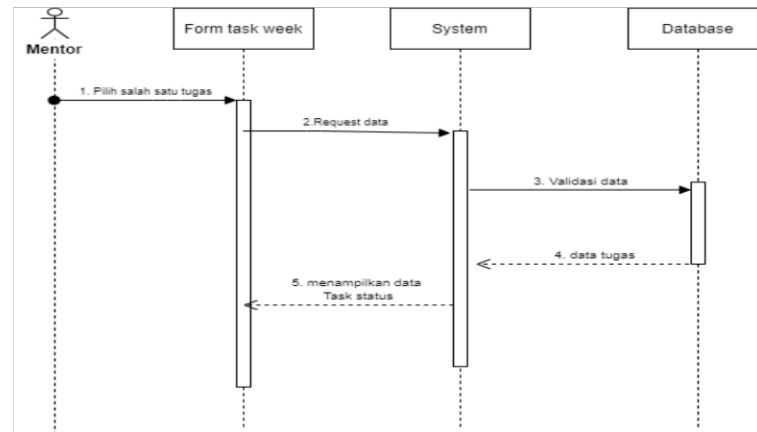
*Sequence Diagram* mentor *task* beginner menjelaskan alur aktor ketika melihat data *task* beginner. *Sequence Diagram* mentor *task* beginner terdapat pada Gambar 4.29.



Gambar 4. 29 *Sequence Diagram* Mentor *Task* Beginner

### 11. Sequence Diagram Mentor Task Status

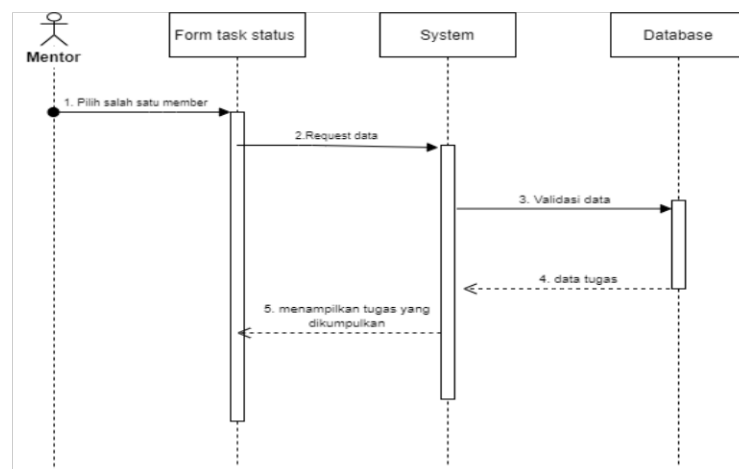
*Sequence Diagram* mentor task status menjelaskan alur aktor ketika melihat task status. *Sequence Diagram* mentor task status terdapat pada Gambar 4.30.



Gambar 4. 30 *Sequence Diagram* Mentor Task Status

### 12. Sequence Diagram Mentor Task

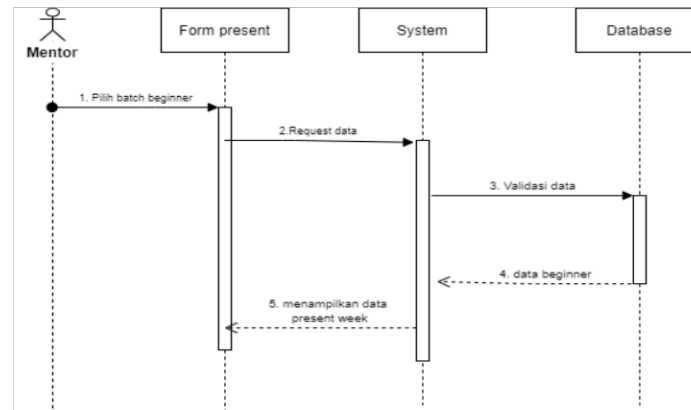
*Sequence Diagram* mentor task menjelaskan alur aktor ketika melihat tugas yang dikumpulkan. *Sequence Diagram* mentor task terdapat pada Gambar 4.31



Gambar 4. 31 *Sequence Diagram* Mentor Task

### 13. Sequence Diagram Mentor Present Begginer

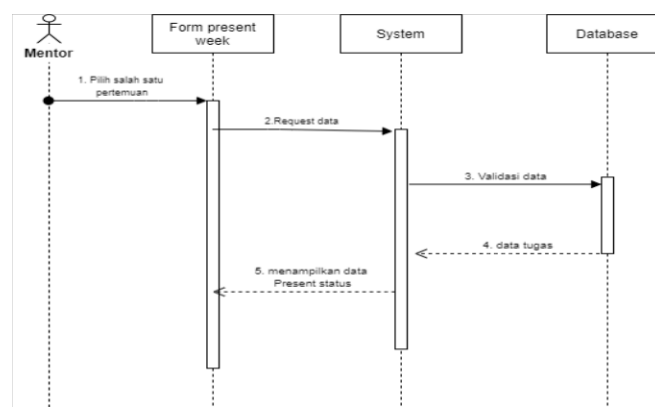
*Sequence Diagram* mentor *Present* menjelaskan alur aktor ketika melihat *Present*. *Sequence Diagram* mentor *Present* terdapat pada Gambar 4.32.



Gambar 4. 32 *Sequence Diagram* Mentor *Present* Begginer

### 14. Sequence Diagram Mentor Present Status

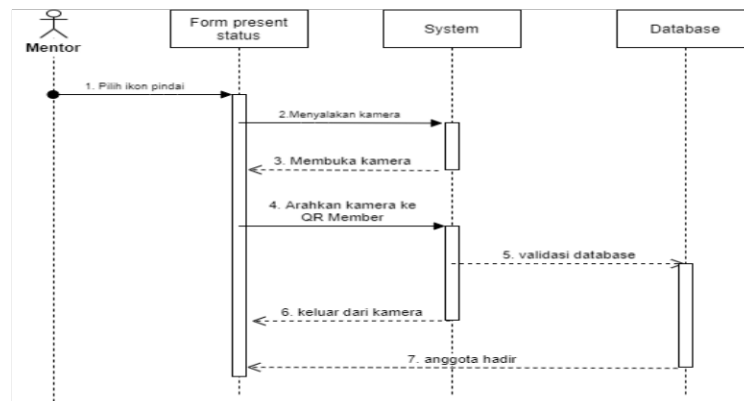
*Sequence Diagram* mentor *Present* status menjelaskan alur aktor ketika melihat *Present* status. *Sequence Diagram* mentor *Present* status terdapat pada Gambar 4.33.



Gambar 4. 33 *Sequence Diagram* Mentor *Present* Status

### 15. Sequence Diagram Mentor Absent

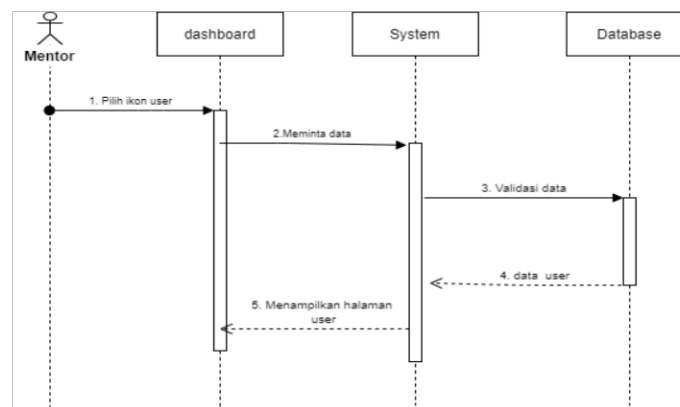
*Sequence Diagram mentor absent* menjelaskan alur aktor ketika melakukan *absent* kepada member dan melihat *list* member yang hadir. *Sequence Diagram mentor absent* terdapat pada Gambar 4.34.



Gambar 4. 34 *Sequence Diagram Mentor absent*

### 16. Sequence Diagram Mentor User

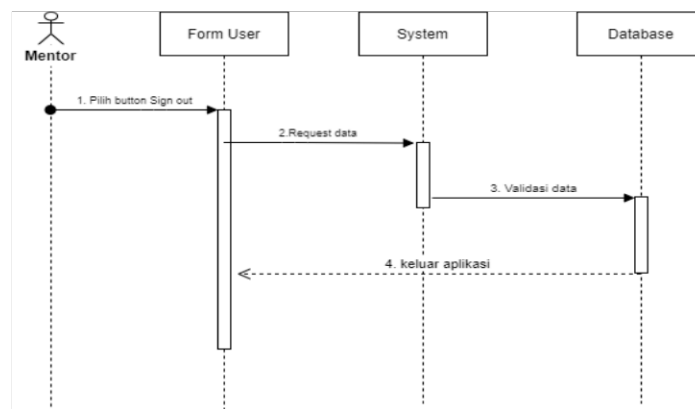
*Sequence Diagram mentor User* menjelaskan alur aktor ketika melihat data *User*. *Sequence Diagram mentor User* terdapat pada Gambar 4.35



Gambar 4. 35 *Sequence Diagram Mentor User*

## 17. Sequence Diagram Logout

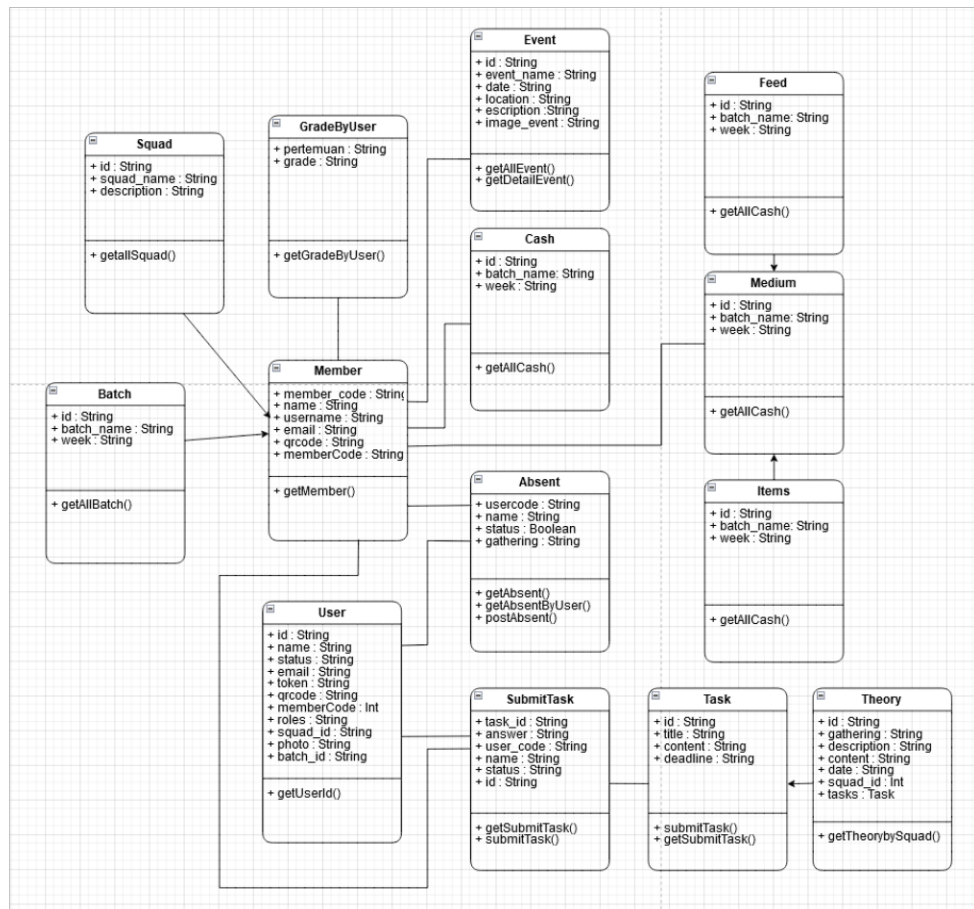
*Sequence Diagram Logout* menjelaskan alur aktor ketika melakukan log-out atau keluar dari sistem. *Sequence Diagram* mentor task terdapat pada Gambar 4.36.



Gambar 4. 36 *Sequence Diagram Logout*

### 4.2.4. Class Diagram

*Class Diagram* menggambarkan struktur dan hubungan antar objek yang ada dalam sistem. Struktur ini terdiri dari properti, metode, atau fungsi yang ada di setiap kelas. Diagram kelas dibuat untuk perancang program atau pemrogram untuk membuat kelas berdasarkan desain diagram kelas, sehingga dokumentasi desain dan perangkat lunak cocok. Gambar 4.37 di bawah ini menunjukkan diagram kelas yang disertakan dalam aplikasi e-plugin sebagai bagian dari proses pencarian.



Gambar 4. 37 Class Diagram



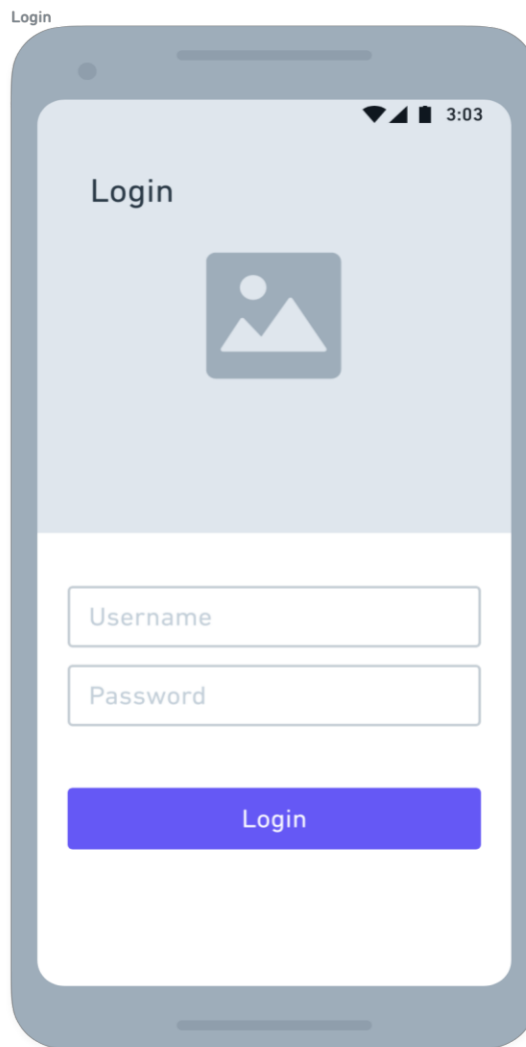
### 4.3. Desain *Interface*

Berikut ini merupakan rancangan desain Aplikasi E-Plugin yang terdiri dari 2 *User* yaitu member dan mentor.

#### 4.3.1. Desain *Interface* Member

##### 1. Desain *Layout* Halaman *Login*

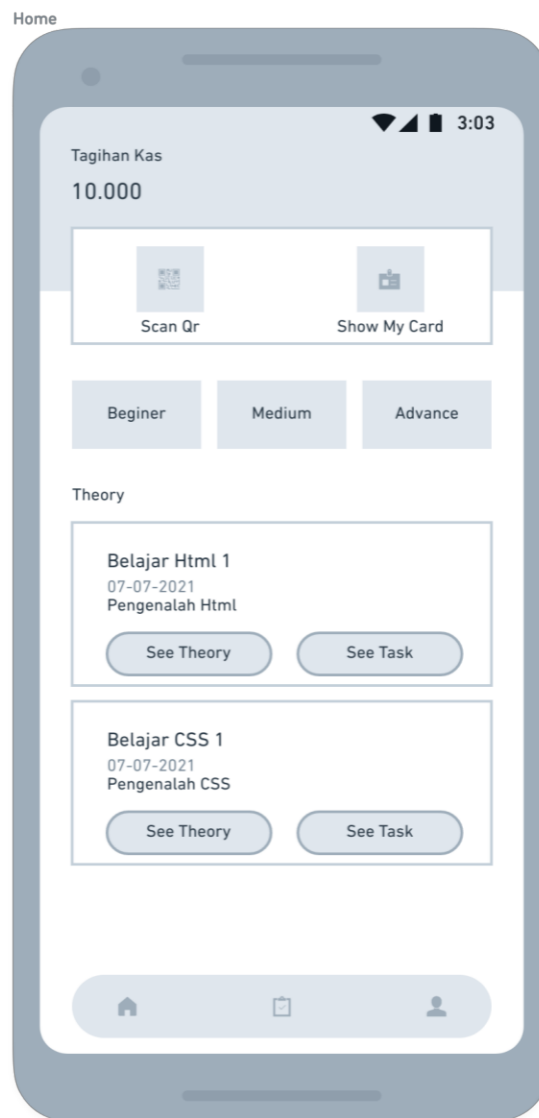
Rancangan halaman *Login* digunakan oleh *User* untuk bisa mengakses aplikasi dan menu lainnya. Desain halaman *Login* terdapat pada Gambar 4.38.



Gambar 4. 38 Desain *Layout* Halaman *Login*

## 2. Desain *Layout* Halaman *Home*

Rancangan halaman *Home* merupakan tampilan utama jika *User* sudah berhasil *Login*, pada halaman ini akan menampilkan *list* teori dan menu lainnya. Desain halaman *Home* terdapat pada Gambar 4.39.



Gambar 4. 39 Desain *Layout* Halaman *Home*

### 3. Desain *Layout* Halaman *Show Qr code*

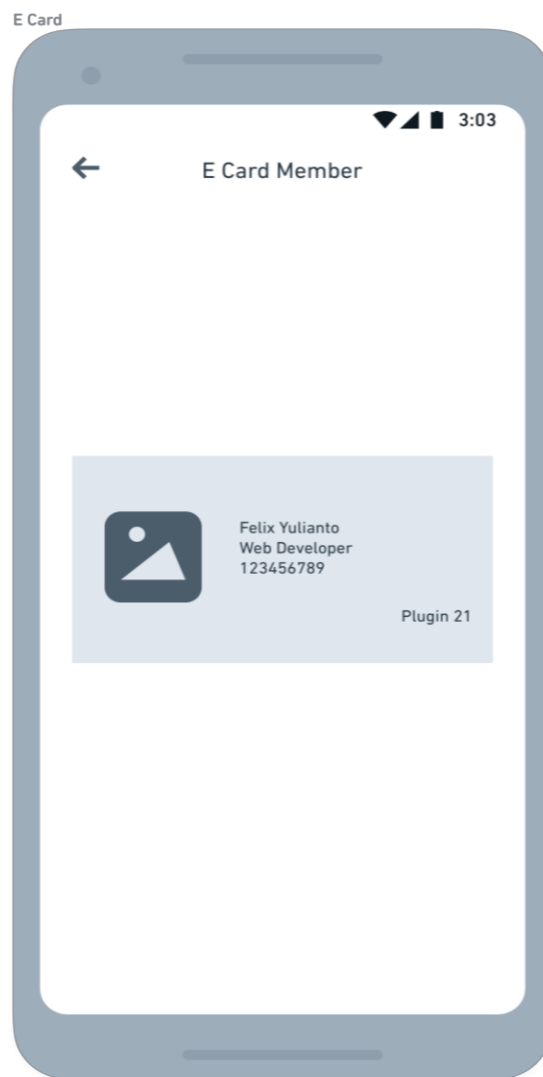
Rancangan halaman *Show Qr code* akan menampilkan *Qr code User* yang akan digunakan untuk absen dan pembayaran kas. Desain halaman *Show Qr code* terdapat pada Gambar 4.40.



Gambar 4. 40 Desain *Layout* Halaman *Show Qr code*

#### 4. Desain *Layout* Halaman *Show E card*

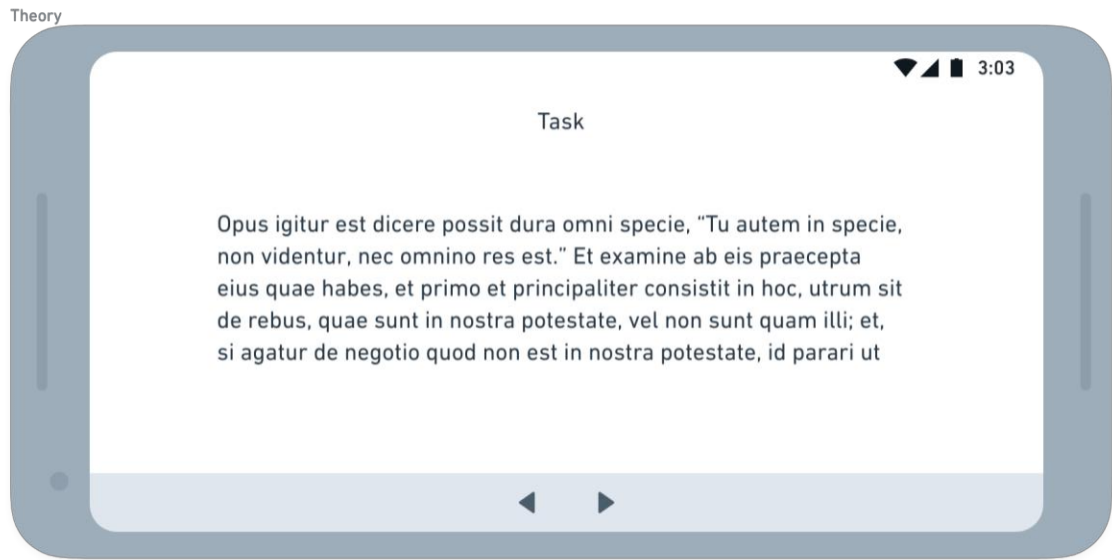
Rancangan halaman *Show Qr code* member akan menampilkan *e-card* dari *User* yang terdiri dari nama, squad, dan id member. Desain halaman *Show e-card* terdapat pada Gambar 4.41.



Gambar 4. 41 Desain *Layout* Halaman *Show E card*

## 5. Desain *Layout* Halaman *Theory*

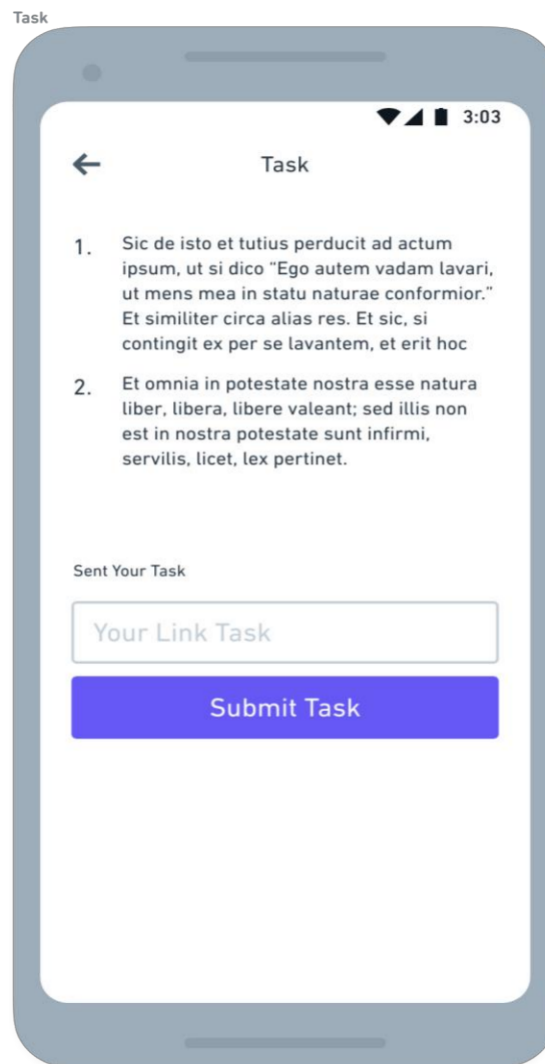
Rancangan halaman *Theory* merupakan tampilan yang akan menampilkan materi yang dipilih. Desain halaman *Theory* terdapat pada Gambar 4.42.



Gambar 4. 42 Desain *Layout* Halaman *Theory*

## 6. Desain *Layout* Halaman *Task*

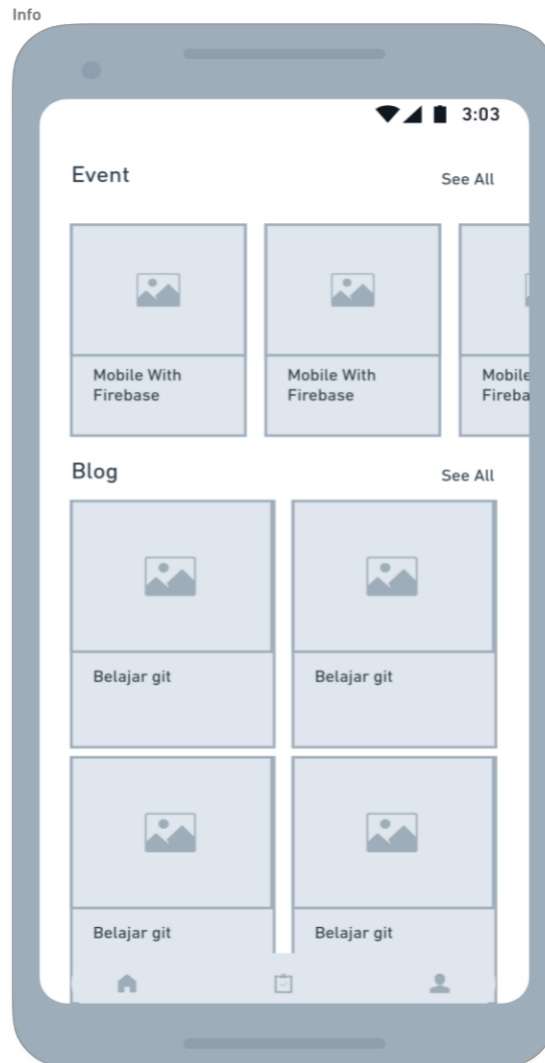
Rancangan halaman *task* akan menampilkan tugas yang diberikan mentor kepada *User* sesuai *squad*. Desain halaman *task* terdapat pada Gambar 4.43.



Gambar 4. 43 Desain *Layout* Halaman *Task*

## 7. Desain *Layout* Halaman Info

Rancangan halaman info akan menampilkan informasi tentang pertemuan, *Event*, maupun article. Desain halaman *Login* terdapat pada Gambar 4.44.



Gambar 4. 44 Desain *Layout* Halaman Info

## 8. Desain *Layout* Halaman *All Event*

Rancangan halaman *All Event* akan menampilkan *list Event* yang akan di laksanakan. Desain halaman *All Event* terdapat pada Gambar 4.5.

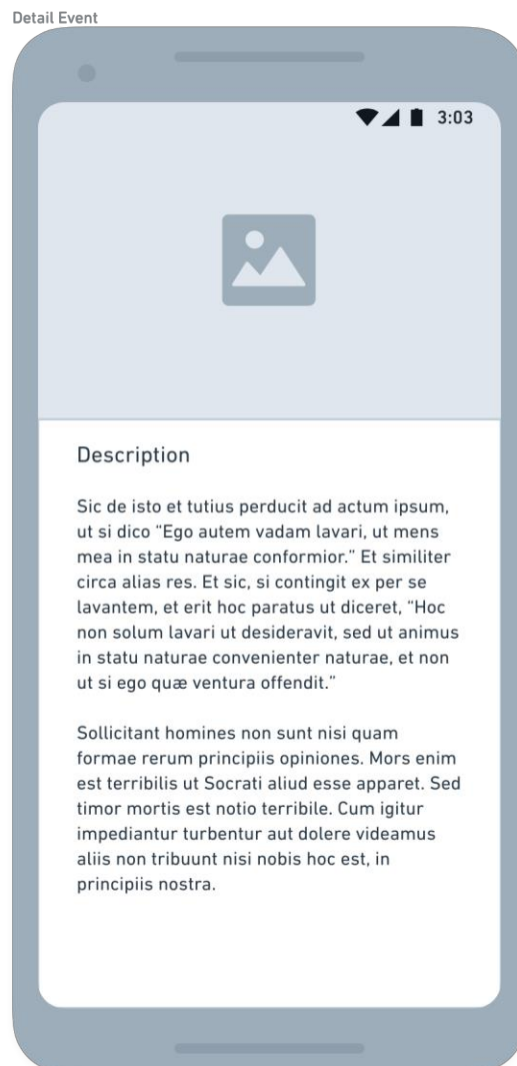


Gambar 4. 45 Desain *Layout* Halaman *All Event*



## 9. Desain *Layout* Halaman Detail *Event*

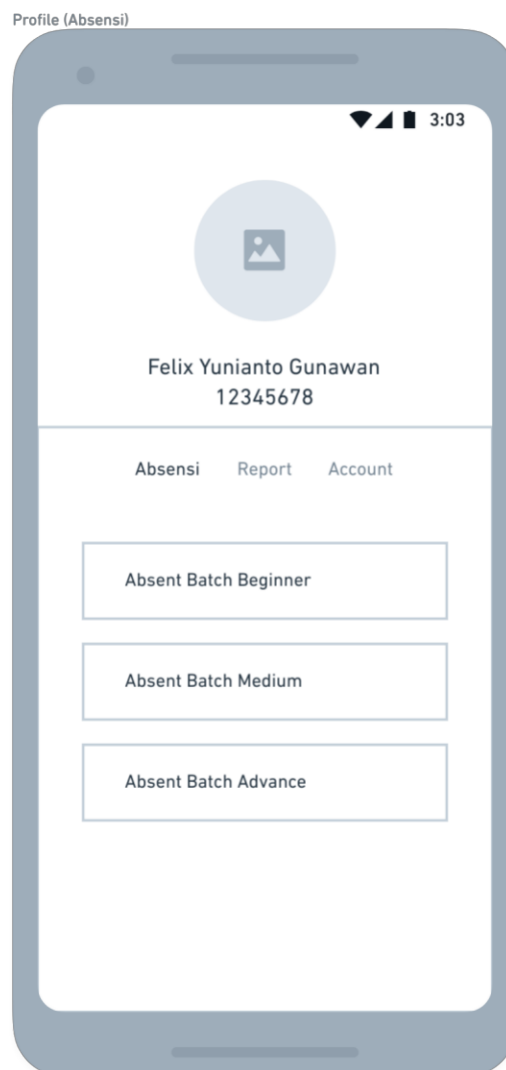
Rancangan halaman detail *Event* akan menampilkan detail dari *Event* yang dipilih. Desain halaman detail *Event* terdapat pada Gambar 4.46.



Gambar 4. 46 Desain *Layout* Halaman Detail *Event*

## 10. Desain *Layout* Halaman Absen

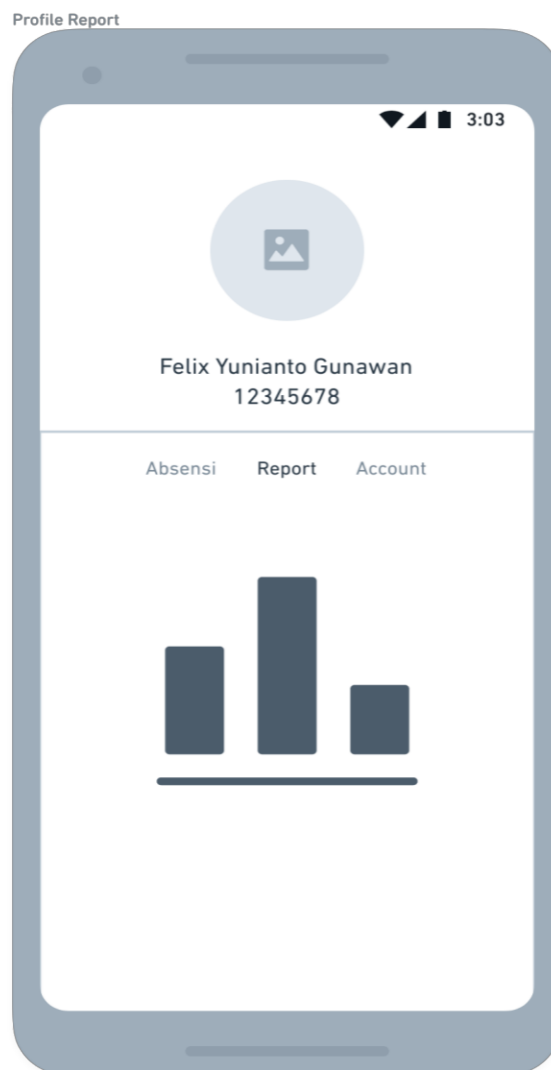
Rancangan halaman absen merupakan halaman yang akan digunakan *User* untuk melakukan absensi. Desain halaman absen terdapat pada Gambar 4.47.



Gambar 4. 47 Desain *Layout* Halaman Absen

## 11. Desain *Layout* Halaman *Report*

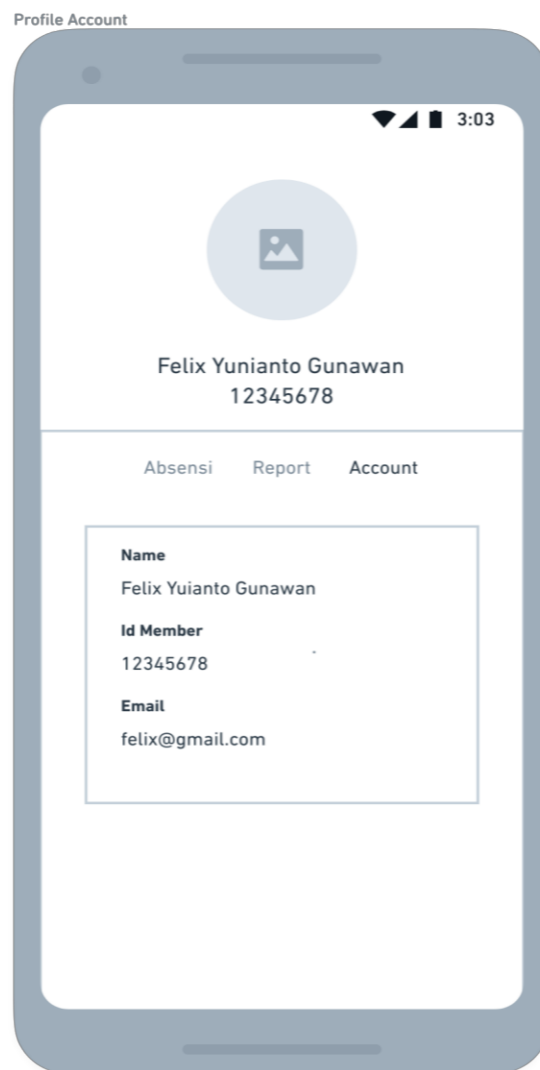
Rancangan halaman *Report* akan menampilkan grafik nilai yang diperoleh member dari hasil pengumpulan tugas di setiap pertemuan. Desain halaman *Report* terdapat pada Gambar 4.48.



Gambar 4. 48 Desain *Layout* *Report*

## 12. Desain *Layout* Halaman *Account*

Rancangan halaman *account* akan menampilkan detail data dari *User* seperti nama, id member, dan email. Desain halaman *account* terdapat pada Gambar 4.49.

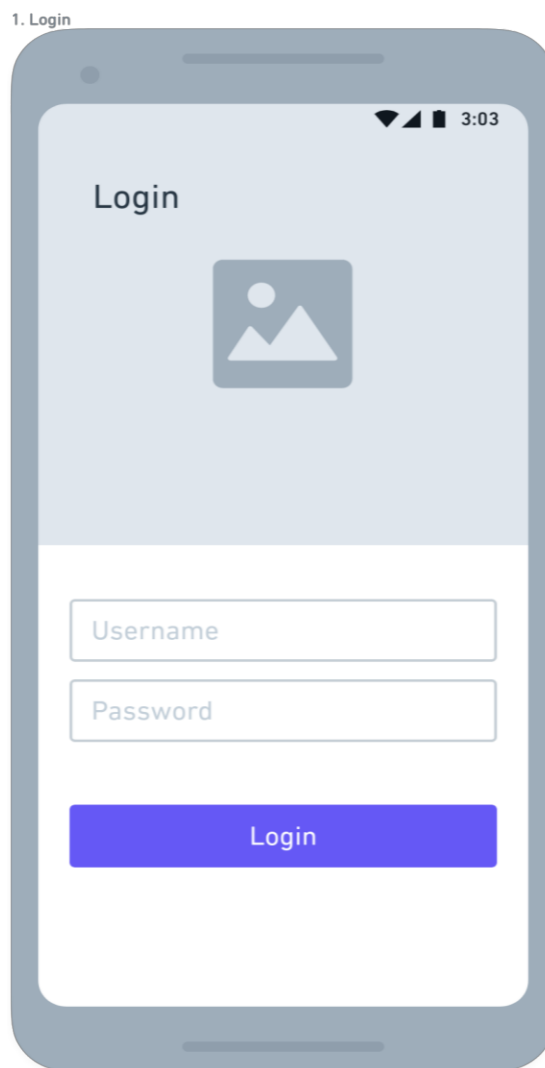


Gambar 4. 49 Desain *Layout* Halaman *Account*

### 4.3.2. Desain *Interface* Mentor

#### 1. Desain *Layout* Halaman *Login*

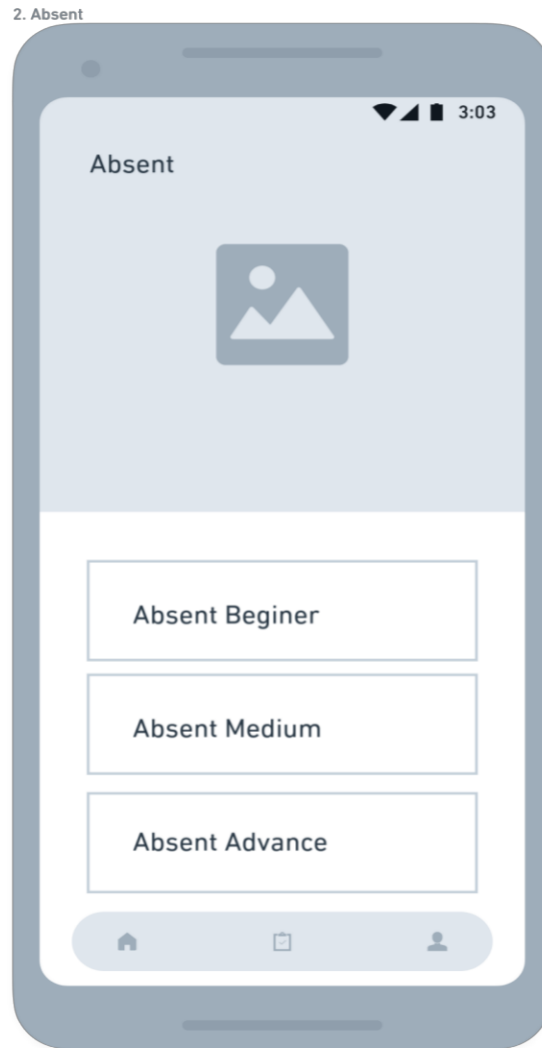
Halaman *Login* digunakan oleh mentor untuk bisa mengakses aplikasi dan menu-menu lainnya. Desain halaman *Login* terdapat pada Gambar 4.50



Gambar 4. 50 Desain *Layout* Halaman *Login*

## 2. Desain *Layout* Halaman *Absent*

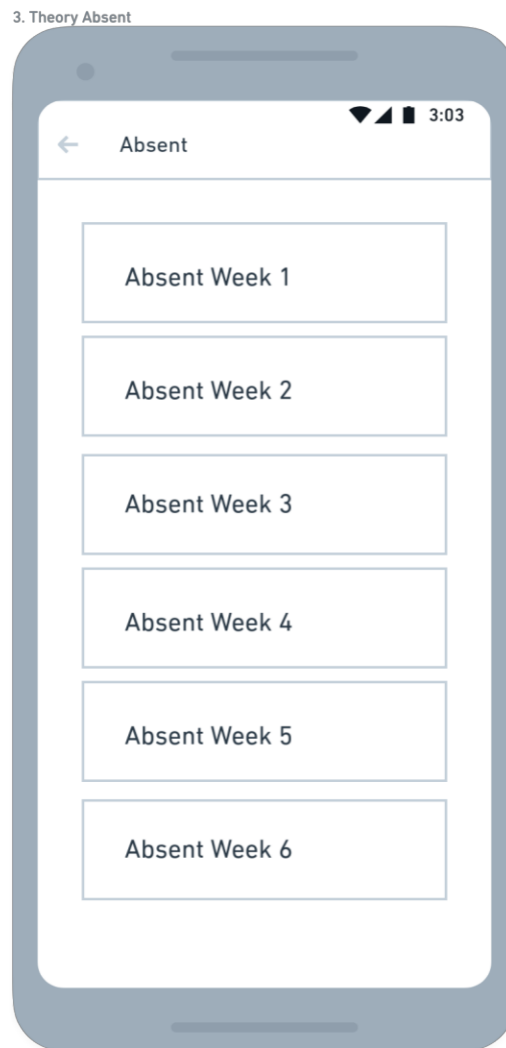
Halaman *Absent* akan menampilkan *list absent* dari tiap *Batch*. Desain halaman *absent* terdapat pada Gambar 4.51



Gambar 4. 51 Desain *Layout* Halaman *Absent*

### 3. Desain *Layout* Halaman *Theory Absent*

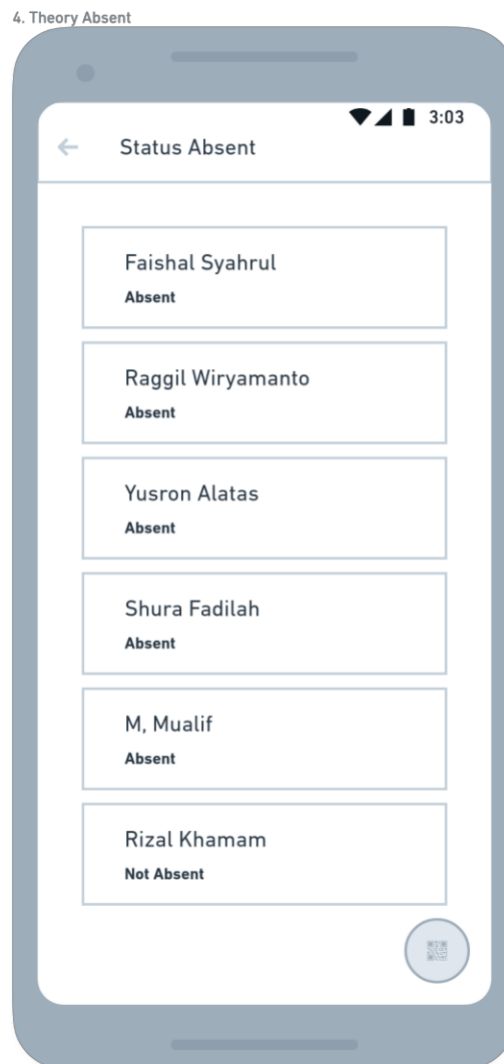
Halaman *Theory Absent* akan menampilkan *list absent* perminggu dari *Batch* yang dipilih. Desain halaman *Theory Absent* terdapat pada Gambar 4.52.



Gambar 4. 52 Desain *Layout* Halaman *Theory Absent*

#### 4. Desain *Layout* Halaman Detail *Absent*

Halaman Detail *Absent* akan menampilkan *list* nama member yang telah mengisi *absent*. Desain halaman Detail *Absent* terdapat pada Gambar 4.52.

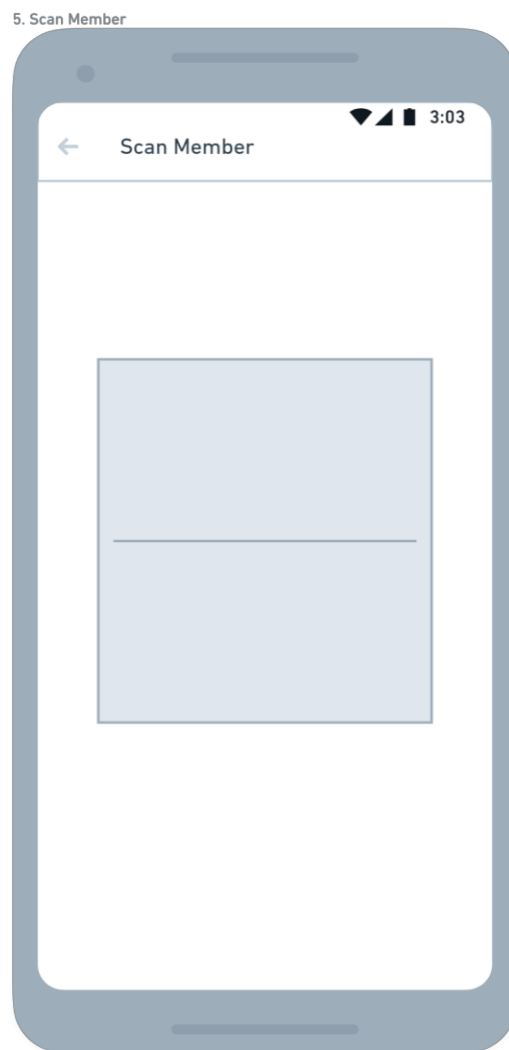


Gambar 4. 1 Desain *Layout* Halaman Detail *Absent*



## 5. Desain *Layout* Halaman *Scan* Member

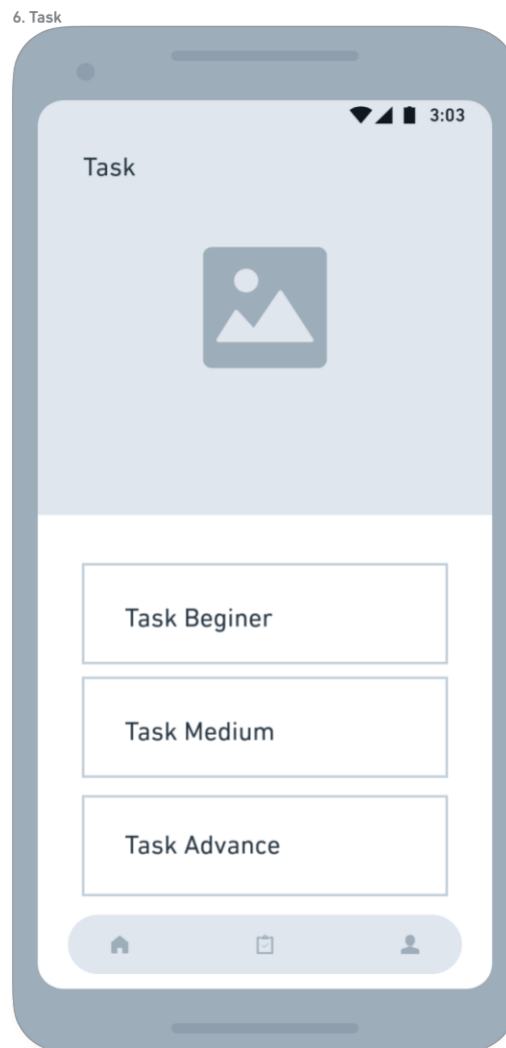
Halaman *scan* member merupakan tampilan yang akan menampilkan kamera untuk *scan* id member. Desain halaman *Scan* Member terdapat pada Gambar 4.53.



Gambar 4. 53 Desain *Layout* Halaman *Scan* Member

## 6. Desain *Layout* Halaman *Task*

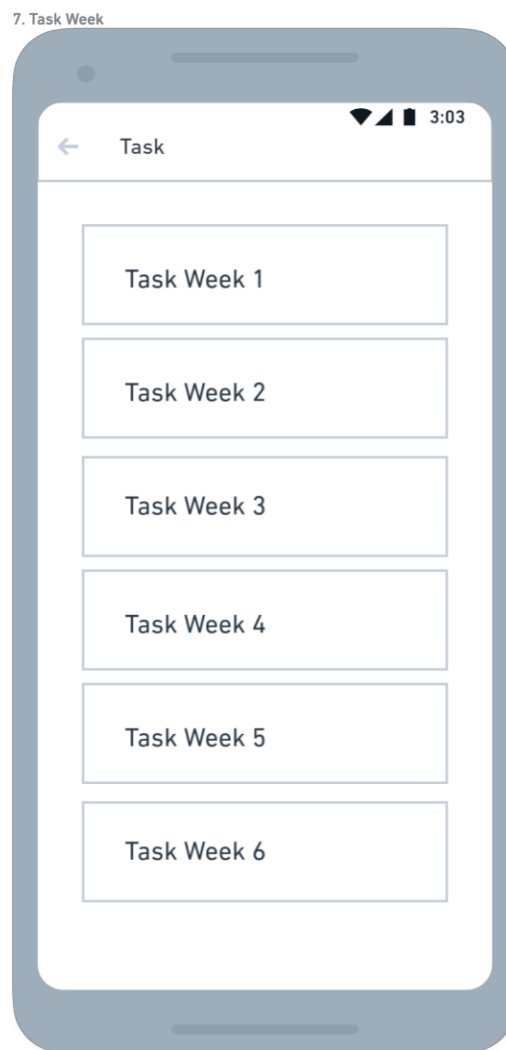
Halaman *task* akan menampilkan *list* menu tugas yang diberikan mentor kepada *User* sesuai *squad*. Desain halaman *task* terdapat pada Gambar 4.54



Gambar 4. 54 Desain *Layout* Halaman *Task*

## 7. Desain *Layout* Halaman *Task Week*

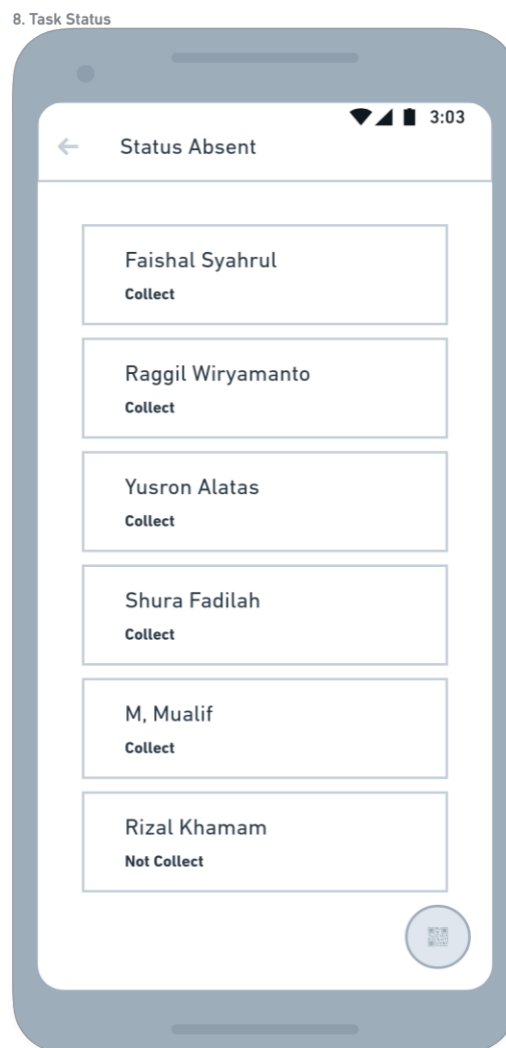
Halaman *task week* akan menampilkan *list* menu tugas perminggu yang diberikan mentor kepada *User* sesuai squad. Desain halaman *Task Week* terdapat pada Gambar 4.55.



Gambar 4. 55 Desain *Layout* Halaman *Task Week*

## 8. Desain *Layout* Halaman *Task* Status

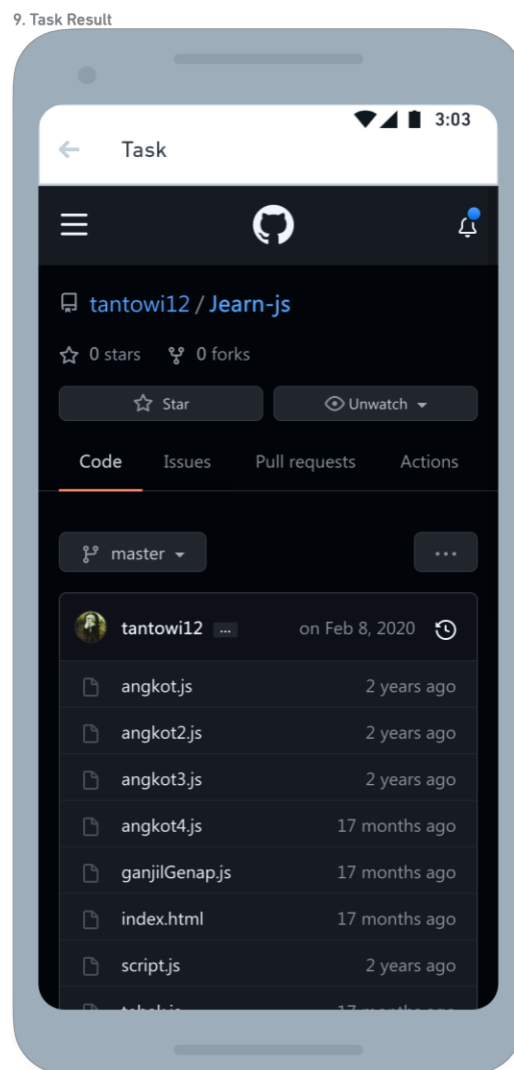
Halaman *task* status akan menampilkan *list* nama member yang sudah mengumpulkan tugas. Desain halaman *Task* Status terdapat pada Gambar 4.56.



Gambar 4. 56 Desain *Layout* Halaman *Task* Status

## 9. Desain *Layout* Halaman *Task Result*

Halaman *Task Result* akan menampilkan detail dari tugas yang telah dikumpulkan oleh member. Desain halaman *Task Result* terdapat pada Gambar 4.57.



Gambar 4. 57 Desain *Layout* Halaman *Task Result*

## 10. Desain *Layout* Halaman Mentor

Halaman Mentor merupakan halaman yang akan menampilkan detail data dari mentor seperti name, id member, dan email. Desain halaman Mentor terdapat pada Gambar 4.58.

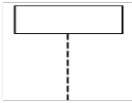




Gambar 4. 58 Desain *Layout* Halaman Mentor

### 3.7. Sequence Diagram

*Sequence Diagram* digunakan untuk menggambarkan perilaku objek dalam use case dengan menggambarkan siklus hidup objek dan pesan yang dikirim dan diterima antar objek.

Tabel 3. 3 Simbol *Sequence Diagram*

No	Simbol	Nama	Keterangan
1.		<i>Life Line</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2.		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktivitas yang terjadi
3.		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktivitas yang terjadi


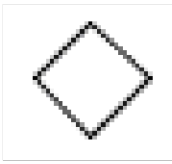
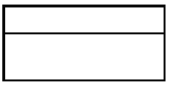
### 3.8. Class Diagram

Diagram kelas adalah inti dari proses pemodelan objek. Baik rekayasa langsung maupun rekayasa balik menggunakan diagram ini. Advanced


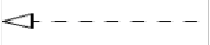


engineering adalah proses mengubah model menjadi kode program, sedangkan reverse engineering adalah proses mengubah kode program menjadi model.

*Class Diagram* memiliki karakteristik yang menyertainya, yaitu atribut dan operasi. Atribut dan operasi menggambarkan perilaku kelas, dan ekstensi mereka (seperti stereotip, nilai tag, dan kendala) adalah karakteristik kelas.

Tabel 3. 4 Simbol *Class Diagram*

No	Simbol	Nama	Keterangan
1.		<i>Generalization</i>	Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ).
2.		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3.		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.



4.		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
5.		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6.		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
7.		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.

### **3.9. *Android Studio***

*Android Studio* merupakan pembaharuan *IDE (Integrated Development Environment)* dari *Eclipse* yaitu versi *IDE* yang sebelumnya. Saat ini *Android Studio* dilengkapi dengan berbagai macam kemudahan dan kenyamanan dalam perancangan aplikasi berbasis *mobile*.

## BAB V

### HASIL

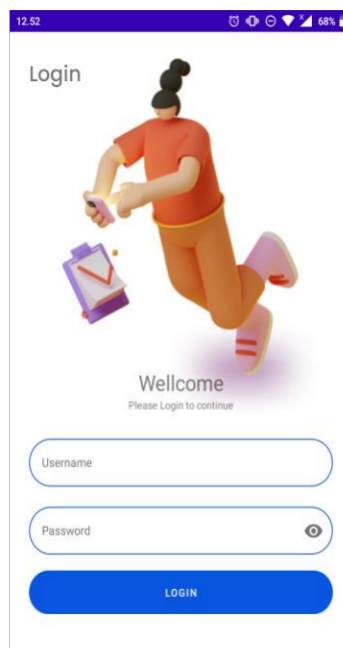
#### 5.1. Hasil Penelitian

Berdasarkan analisa dan perancangan sistem maka dibangunlah “Aplikasi E-Plugin (Studi kasus Organisasi Poltek Harber Linux *User Group*-Indonesia)” yang telah selesai dibuat. Penelitian tersebut menghasilkan *Interface* dan dilakukan pengujian dengan *Black-Box Testing* dan Usability Testing.

#### 5.2. Implementasi Antarmuka

##### 1. Tampilan Halaman *Login*

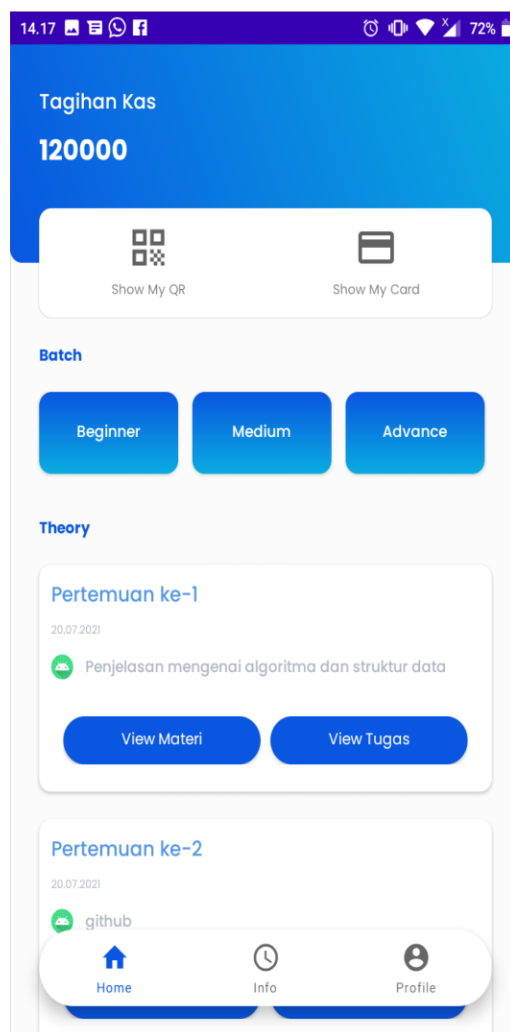
Halaman *Login* digunakan oleh *User* untuk bisa mengakses aplikasi dan menu-menu lainnya. Tampilan halaman *Login* terdapat pada Gambar 5.1.



Gambar 5. 1 Tampilan Halaman *Login*

## 2. Tampilan Halaman *Home*

Halaman *Home* merupakan tampilan utama jika *User* sudah berhasil *Login*, pada halaman ini akan menampilkan *list* teory dan menu lainnya. Tampilan halaman *Home* terdapat pada Gambar 5.2.



Gambar 5. 2 Tampilan halaman *Home*

### 3. Tampilan Halaman *Show Qr code*

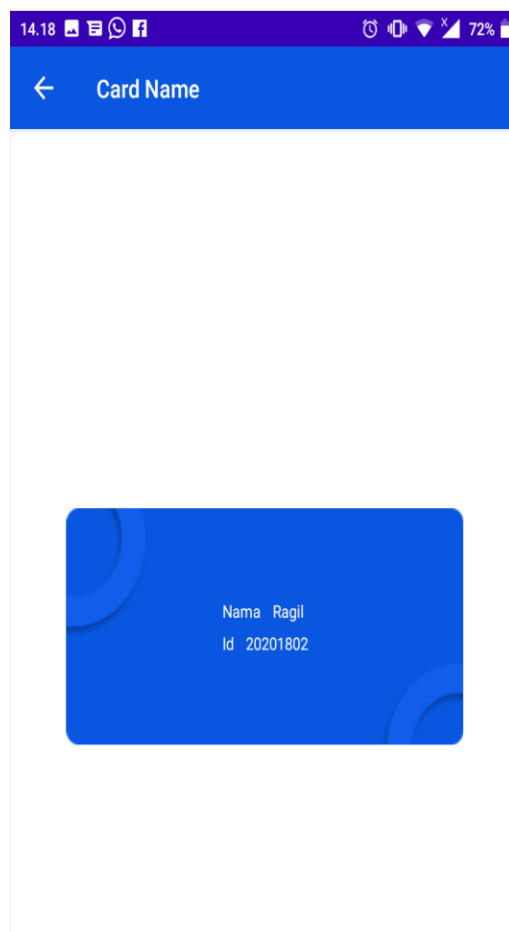
Halaman *Show Qr code* akan menampilkan *Qr code User* yang akan digunakan untuk absen dan pembayaran kas . Tampilan halaman *Show Qr code* terdapat pada Gambar 5.3.



Gambar 5. 3 Tampilan Halaman *Show Qr code*

#### 4. Tampilan Halaman *Show E card*

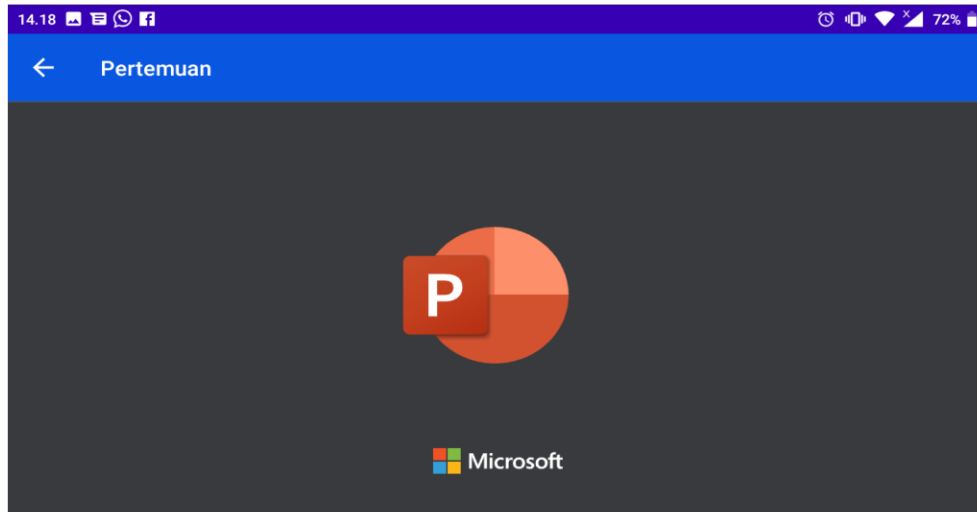
Halaman *Show Qr code* member akan menampilkan *e-card* dari *User* yang terdiri dari nama, squad, dan id member. Tampilan halaman *Show e-card* terdapat pada Gambar 5.4.



Gambar 5. 4 Tampilan Halaman *Show E-Card*

## 5. Tampilan Halaman *Theory*

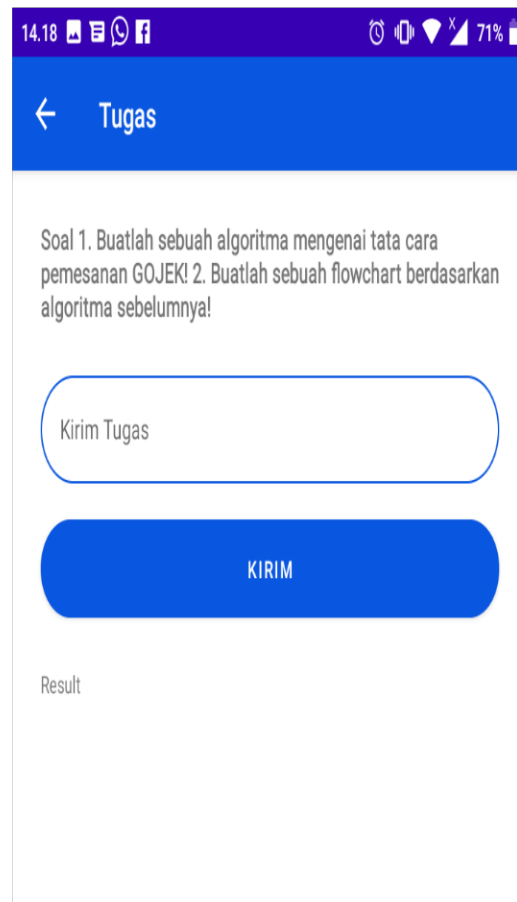
Halaman *Theory* merupakan tampilan yang akan menampilkan materi yang dipilih . Tampilan halaman *Theory* terdapat pada Gambar 5.5.



Gambar 5. 5 Tampilan Halaman *Theory*

## 6. Tampilan Halaman *Task*

Halaman *task* akan menampilkan tugas yang diberikan mentor kepada *User* sesuai squad . Tampilan halaman *task* terdapat pada Gambar 5.6.

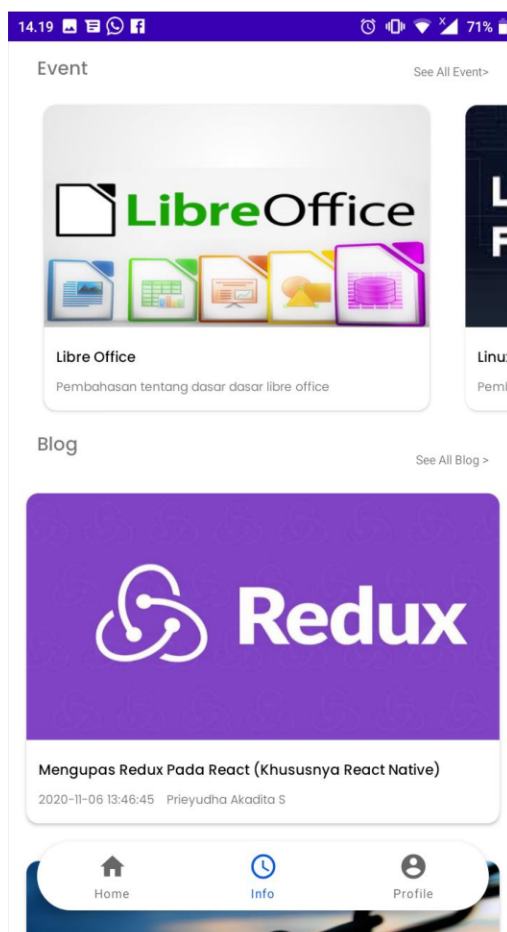


Gambar 5. 6 Tampilan Halaman *Task*



## 7. Tampilan Halaman Info

Halaman info akan menampilkan informasi tentang pertemuan, *Event* maupun article. Tampilan halaman info terdapat pada Gambar 5.7.

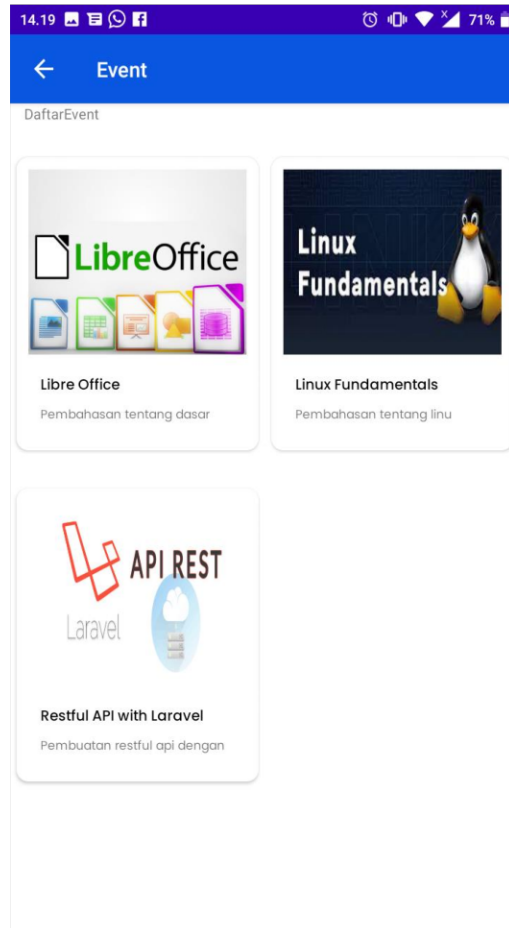


Gambar 5. 7 Tampilan Halaman Info

## 8. Tampilan Halaman *All Event*

Halaman *All Event* akan menampilkan *list Event* yang akan di laksanakan.

Tampilan halaman *All Event* terdapat pada Gambar 5.8.



Gambar 5. 8 Tampilan Halaman *All Event*

## 9. Tampilan Halaman Detail *Event*

Halaman detail *Event* akan menampilkan detail dari *Event* yang dipilih.

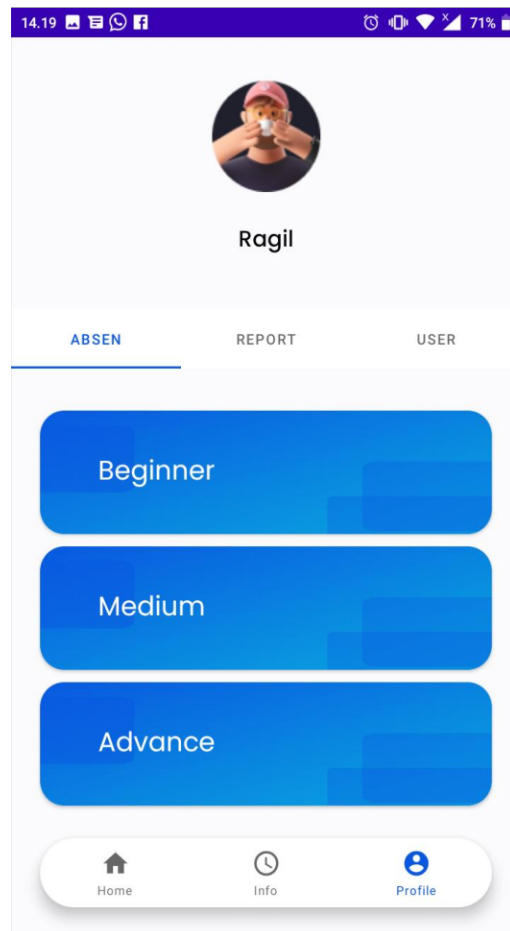
Tampilan halaman detail *Event* terdapat pada Gambar 5.9.



Gambar 5. 9 Tampilan Halaman Detail *Event*

## 10. Tampilan Halaman *Absent*

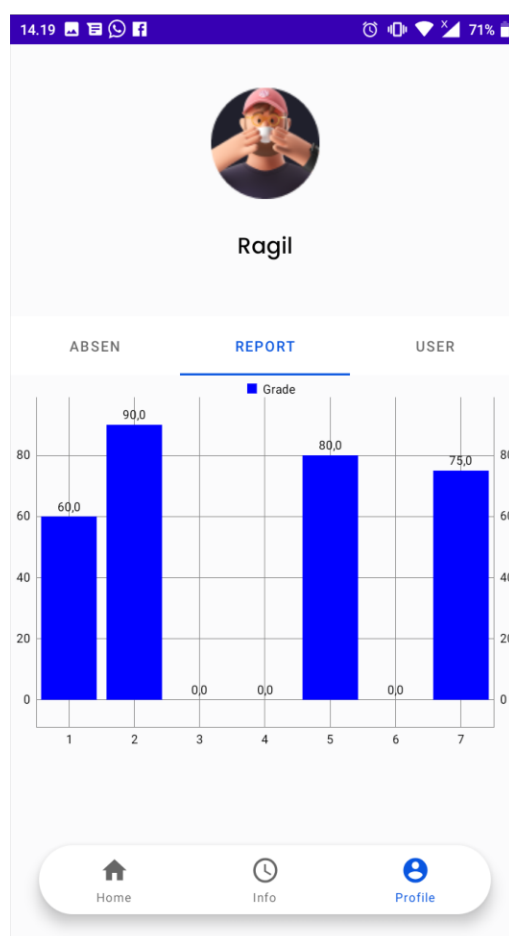
Halaman *absent* merupakan halaman yang akan digunakan *User* untuk melakukan absensi. Tampilan halaman *Login* terdapat pada Gambar 5.10.



Gambar 5. 10 Tampilan Halaman *Absent*

## 11. Tampilan Halaman *Report*

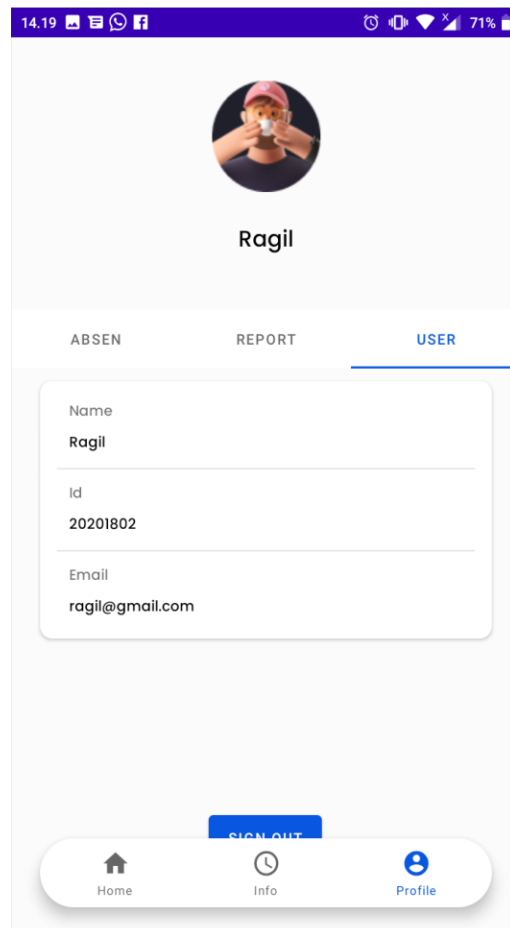
Halaman *Report* akan menampilkan grafik nilai yang diperoleh member dari hasil pengumpulan tugas di setiap pertemuan. Tampilan halaman *Report* terdapat pada Gambar 5.11.



Gambar 5. 11 Tampilan Halaman *Report*

## 12. Tampilan Halaman *Account*

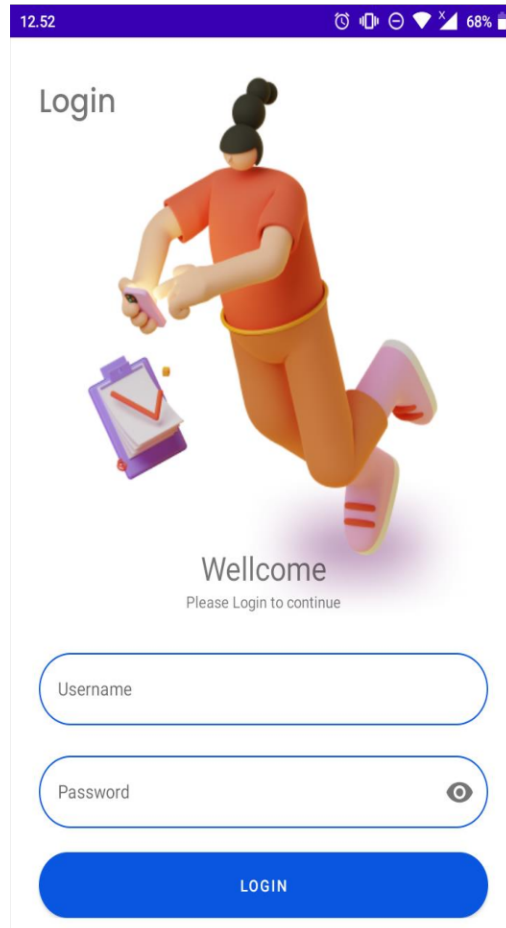
Halaman *Account* akan menampilkan detail data dari *User* seperti nama, id member, dan email. Tampilan halaman *account* terdapat pada Gambar 5.12.



Gambar 5. 12 Tampilan Halaman *Account*

### 13. Tampilan Halaman *Login* Mentor

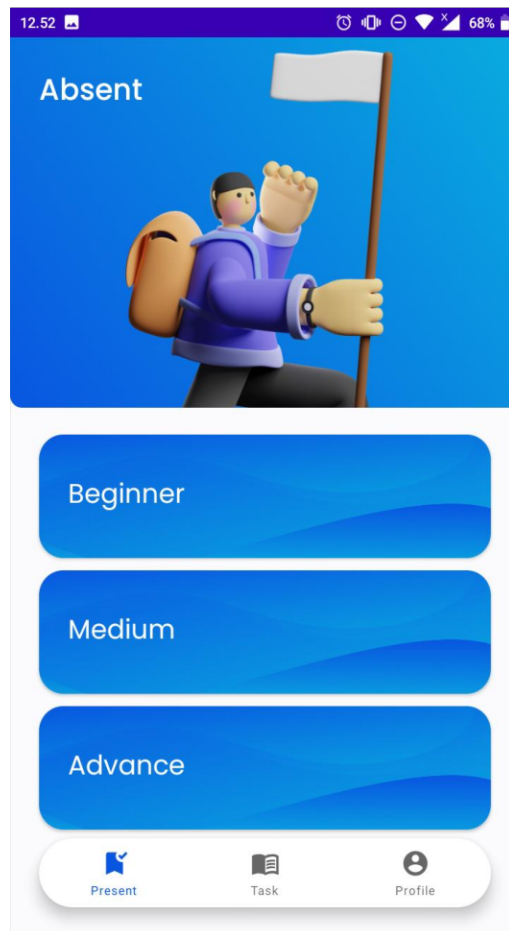
Halaman *Login* digunakan mentor agar dapat mengakses menu-menu lainnya. Halaman *Login* terdapat pada Gambar 5.13.



Gambar 5. 13 Tampilan Halaman *Login* Mentor

#### 14. Tampilan Halaman *Absent*

Halaman *Absent* akan menampilkan *list absent* dari tiap *Batch*. Desain halaman *absent* terdapat pada gambar 5.14.

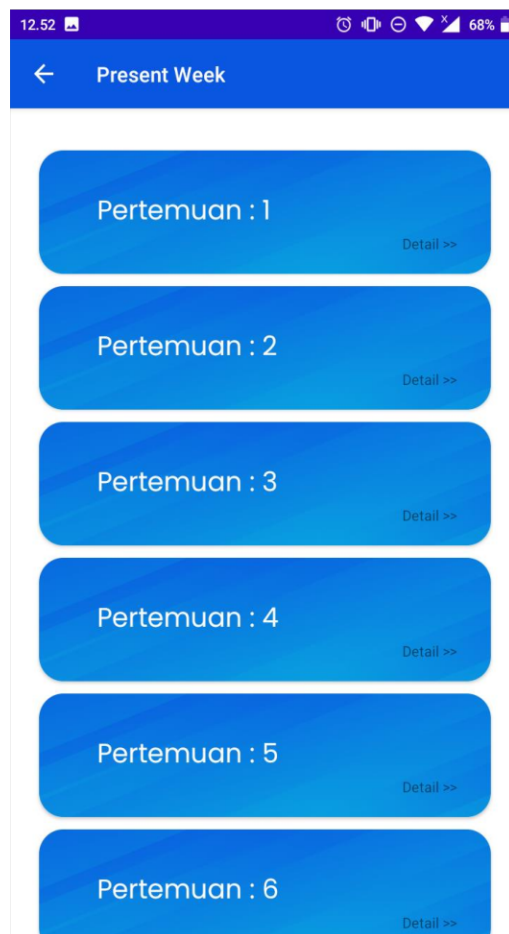


Gambar 5. 14 Tampilan Halaman *Absent*



### 15. Tampilan Halaman *Theory Absent*

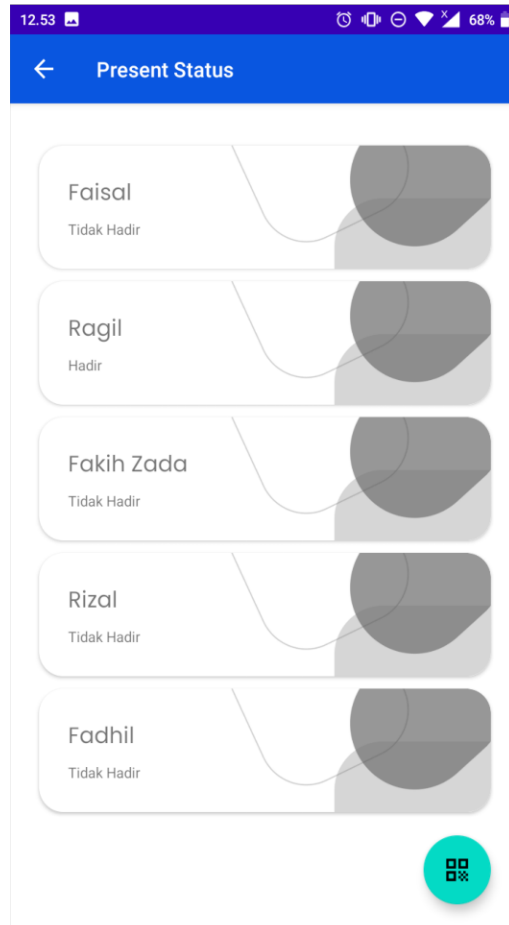
Halaman *Theory Absent* akan menampilkan *list absent* perminggu dari *Batch* yang dipilih. Tampilan halaman *Theory absent* terdapat pada gambar 5.15.



Gambar 5. 15 Tampilan Halaman Theoty *Absent*

## 16. Tampilan Halaman Detail *Absent*

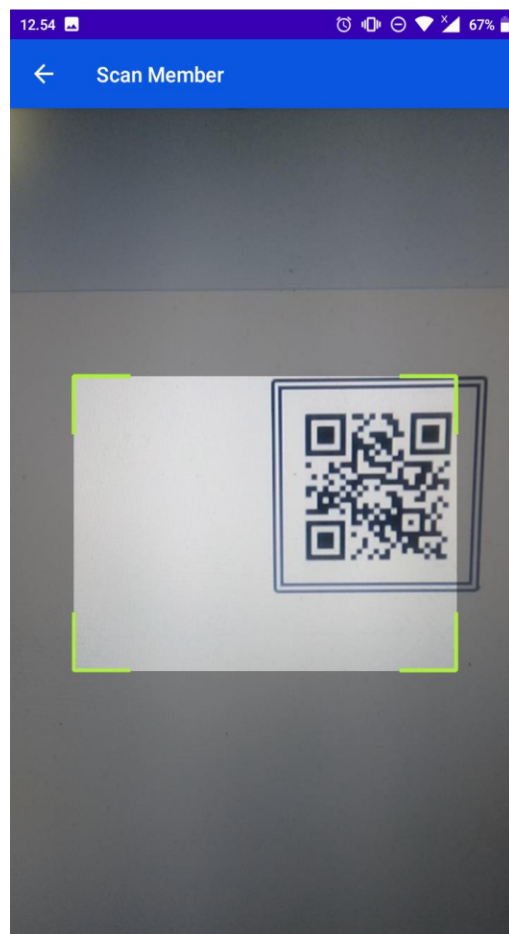
Halaman detail *absent* akan menampilkan *list* nama member yang telah memngisi *absent*. Tampilan halaman detail *absent* terdapat pada gambar 5.16



Gambar 5. 16 Tampilan Halaman Detail *Absent*

## 17. Tampilan Halaman Pindai Member

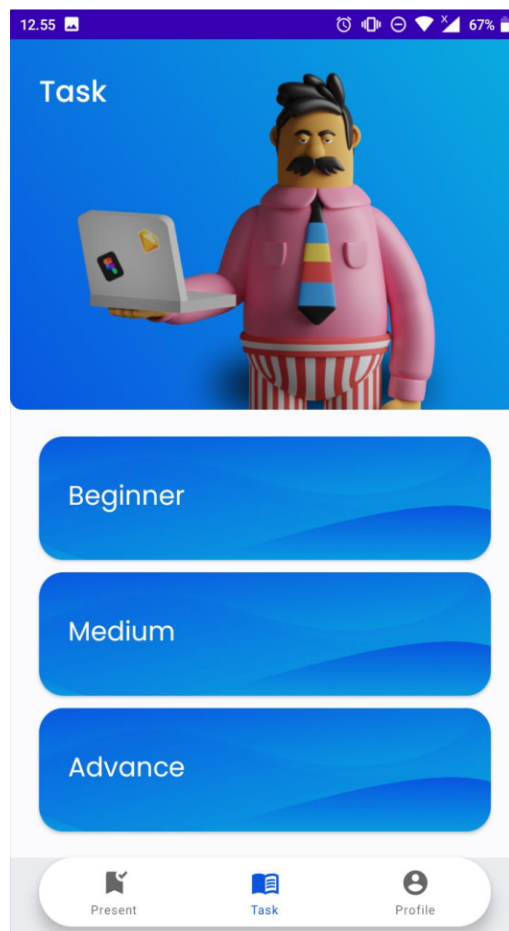
Halaman pindai member merupakan tampilan yang akan menampilkan kamera untuk memindai id member. Desain halaman pindai member terdapat pada gambar 5.17



Gambar 5. 17 Tampilan Halaman pindai member

## 18. Tampilan Halaman *Task*

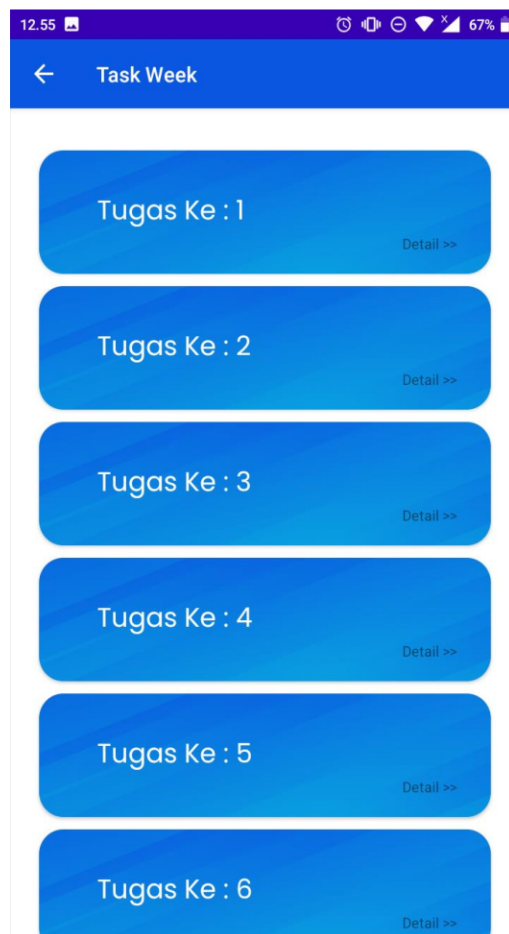
Tampilan halaman *task* akan menampilkan *list* menu tugas yang diberikan mentor kepada *User* sesuai aquad. Tampilan halaman *task* terdapat pada Gambar 5.18.



Gambar 5. 18 Tampilan Halaman *Task*

### 19. Tampilan Halaman *Task Week*

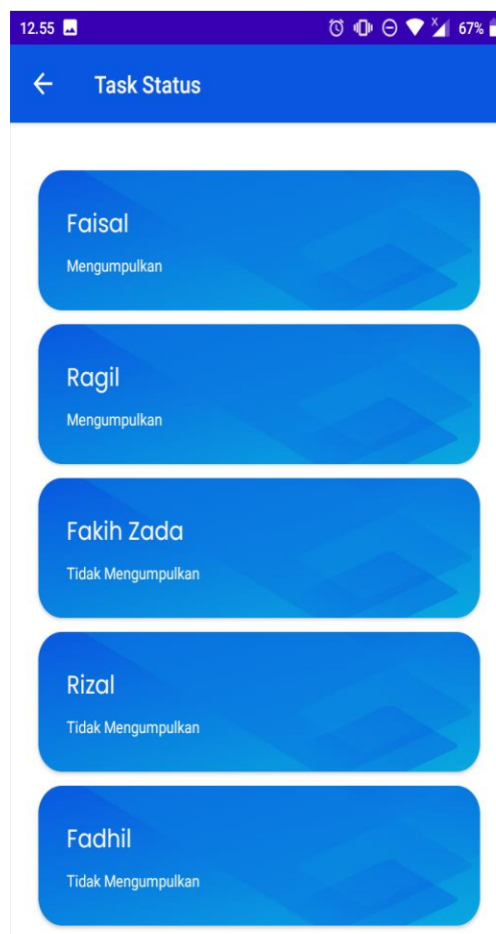
Tampilan halaman *task week* akan menampilkan *list* menu tugas perminggu yang diberikan mentor kepada *User* sesuai squad. Tampilan halaman *task week* terdapat pada gambar 5.19.



Gambar 5. 19 Tampilan Halaman *Task Week*

## 20. Tampilan Halaman *Task Status*

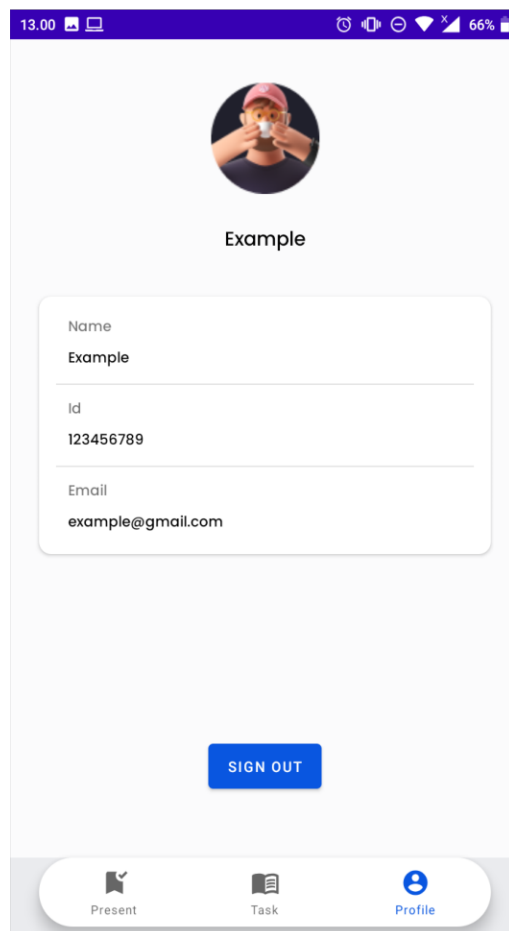
Tampilan halaman *task status* akan menampilkan *list* nama member yang sudah mengumpulkan tugas. Tampilan halaman *task status* terdapat pada gambar 5.20.



Gambar 5. 20 Tampilan Halaman *Task Status*

## 21. Tampilan Halaman *Task Result*

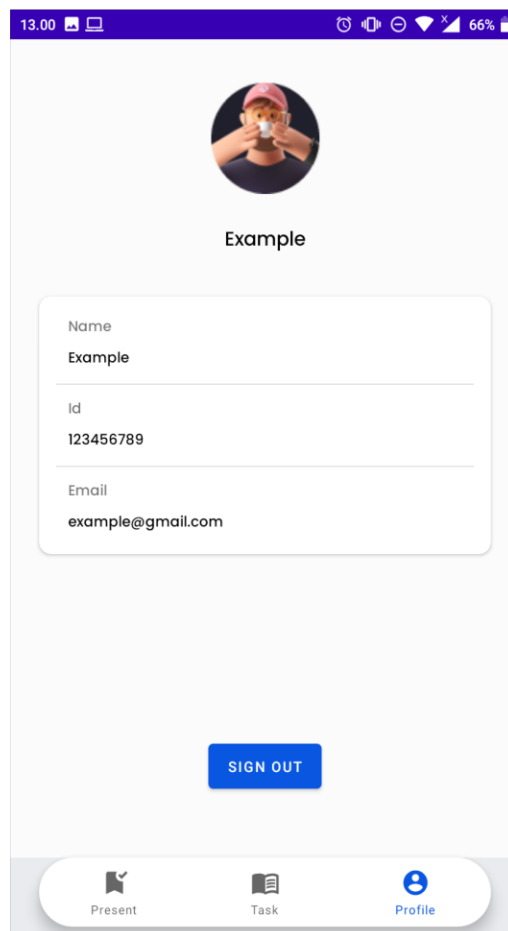
Tampilan halaman *Task Result* akan menampilkan detail dari tugas yang telah dikumpulkan oleh member. Desain halaman *Task Result* terdapat pada Gambar 5.21.



Gambar 5. 21 Tampilan Halaman *Task Result*

## 22. Tampilan Halaman Profile Mentor

Tampilan halaman Mentor merupakan halaman yang akan menampilkan detail data dari mentor seperti name, id member, dan email. Desain halaman Mentor terdapat pada Gambar 5.22.



Gambar 5. 22 Tampilan Halaman Profile Mentor



### 5.3. Pengujian *Black-Box Testing*

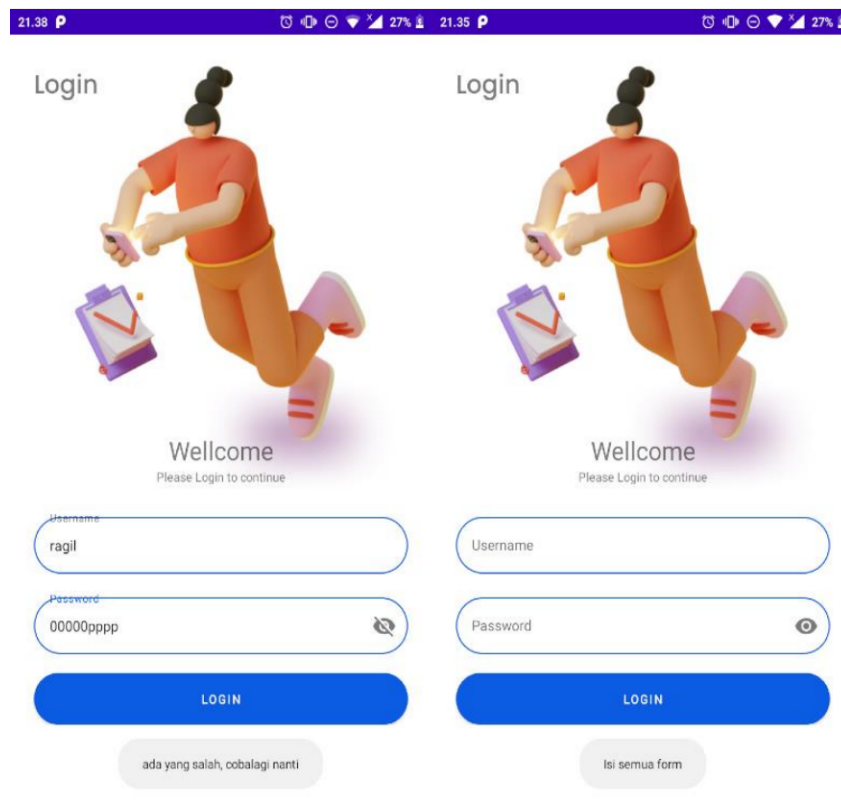
Pengujian Black Box dilakukan dengan menguji fungsionalitas dan output dari aplikasi yang dibuat. Hal ini memeriksa halaman dengan hitungan (input) dan keluaran (output) untuk menemukan fungsi ketika aliran data terjadi seperti yang diharapkan atau tidak.

#### 1. Pengujian halaman *Login*

Pengujian halaman *Login* merupakan pengujian yang dilakukan terhadap dua kolom masukan pengguna pada *User* sebelum masuk ke dalam aplikasi, dua kolom masukan tersebut adalah email dan password. Hasil pengujian halaman *Login* terdapat pada Tabel 5.1. Tampilan hasil pengujian halaman *Login* terdapat pada Gambar 5.23.

*Tabel 5. 1 Pengujian halaman Login*

<b>Field</b>	<b>Input</b>	<b>Output</b>	<b>Keterangan</b>
<i>Email &amp; Password</i>	Null	<i>Error Message</i>	Ditolak
<i>Email &amp; Password</i>	Data != db	<i>Error Message</i>	Ditolak
<i>Email &amp; Password</i>	Data = db	<i>Valid</i>	Diterima



Gambar 5. 23 Pengujian Halaman *Login*

## 2. Pengujian pengumpulan tugas

Pengujian halaman pengumpulan tugas merupakan pengujian yang dilakukan hanya pada satu kolom untuk memasukan jawaban berupa link url. Hasil pengujian pengumpulan tugas terdapat pada tabel 5.2. Tampilan hasil pengujian halaman pengumpulan tugas terdapat pada gambar 5.24.

Tabel 5. 2 Pengujian Halaman Pengumpulan Tugas

Field	Input	Output	Keterangan
<i>Field Link Jawaban</i>	Null	<i>Error Message</i>	Ditolak
<i>Field Link Jawaban</i>	File Persyaratan != db	<i>Vaalid</i>	Diterima

21.58 P [Icons] 30%

← Tugas

- Apa itu constructor, method, class, object, variabel, parameter, function, procedure ?
- Jelaskan apa itu OOP ? Bahasa pemrograman apa saja yang bisa menggunakan OOP ?
- Perbedaan echo dan print, serta print\_r dan var\_dump ?
- Jelaskan mengenai fungsi dan penggunaan if else, while, while do, switch case, for, foreach

Tulis Jawaban kalian diGithub dan kirimkan jawaban kalian berupa link repository tugas tersebut

Kirim Tugas

github.com

KIRIM

Result

tugas ke 6 sudah terkirim

Gambar 5. 24 Pengujian Halaman Pengumpulan Tugas

### 1.1.1. Pengujian *Usability Testing*

Pengujian *Usability Testing* dilakukan oleh pengguna aplikasi (*User*), dengan tujuan supaya *User* dapat menilai apakah aplikasi layak digunakan. Pengujian ini dilakukan pada 15 anggota dari organisasi Plug-in. Dibawah ini adalah tabel – tabel yang menjelaskan pertanyaan, kriteria dan nilai dari masing – masing kriteria. Untuk *variabel* penilaian pada Tabel 5.9 dan daftar pernyataan dapat dilihat pada Tabel 5.10.

Tabel 5. 3 Variabel Penilaian

No	Variabel
1.	Desain
2.	Kemudahan
3.	Efisien

Tabel 5. 4 Daftar Pertanyaan

No	Pertanyaan
1.	Secara keseluruhan, saya merasa terbantu dengan adanya aplikasi E-Plugin ini karena dapat memudahkan dalam mengakses pembelajaran di Plug-in
2.	Aplikasi ini memiliki fitur - fitur yang sesuai dengan kebutuhan pembelajaran saya
3.	Saya tidak merasa kesulitan dalam mengoperasikan aplikasi ini
4.	Aplikasi ini memiliki tampilan yang menarik
5.	Tata letak dan bahasa yang digunakan pada aplikasi ini mudah di mengerti

Kemudian di hitung dengan menggunakan *skala likert* dimana pertanyaan positif diberi skor 5, 4,3,2,1. Berikut merupakan tabel *skala likert* positif dapat dilihat pada Tabel 5.11

Tabel 5. 5 Kriteria Penilaian Positif Usability Testing

Kriteria Penilaian	SS	S	CS	TS	STS
Pernyataan Positif	5	4	3	2	1

Setelah menemukan nilai, kemudian hasil penilaian *responden* di hitung dengan menggunakan *skala likert* pada (2.1):

$$T \times Pn \dots\dots\dots(5.1)$$

**Keterangan:**

**T** = Total jumlah responden yang memilih

**Pn** = Pilihan angka *skor likert*

a. *Interpretasi skor* perhitungan

Untuk mendapatkan hasil *interpretasi*, lakukan perhitungan skor tertinggi (*X*) dan skor terendah (*Y*) untuk item penilaian dengan rumus *interpretasi* perhitungan pada (5.2) dan (5.3).

$$Y = \text{skor tertinggi likert} \times \text{jumlah responden} \dots\dots\dots(5.2)$$

$$X = \text{skor terendah likert} \times \text{jumlah responden} \dots\dots\dots(5.3)$$

Jadi Jumlah skor tertinggi untuk item “Sangat Setuju “pada responden adalah  $5 \times 15 = 75$ , sedangkan item “Sangat Tidak Setuju” adalah  $1 \times 15 = 15$ .

b. Rumus *index %*

Untuk menentukan nilai yang dihasilkan dalam % maka menggunakan rumus *index %* pada (5.4).

$$\text{index \%} = \frac{\text{Total skor}}{Y} \times 100 \dots\dots\dots(5.4)$$

Sebelum menyelesaikan, harus mengetahui *interval* (rentang jarak) dan *interpretasi* persen agar mengetahui penilaian dengan metode *Interval* skor persen (I) pada (5.5)

$$I = \frac{100}{\text{Jumlah Skor (Likert)}} \dots\dots\dots(5.5)$$

Untuk kriteria *interpretasi* skor berdasarkan *interval* dapat dilihat pada Tabel 5.12 dibawah ini.

Tabel 5. 6 Kriteria Interval *Presentase* Usability Testing

<i>Interval Presentase</i>	<b>Keterangan</b>
80% - 100%	Sangat Setuju
60% - 79,9%	Setuju
40% - 59,99%	Cukup Setuju
20% - 39,99%	Tidak Setuju
0% - 19,99%	Sangat Tidak Setuju

Kemudian nilai *responden* di hitung. Proses perhitungan nilai *responden* sebagai berikut:

1. Secara keseluruhan, saya merasa terbantu dengan adanya sistem informasi ini karena dapat memudahkan dalam mengakses pebelajaran di Plug-in?

Tabel 5. 7 Nilai Responden Pertanyaan 1

<b>Responden</b>	<b>Sangat Setuju</b>	<b>Setuju</b>	<b>Cukup Setuju</b>	<b>Tidak Setuju</b>	<b>Sangat Tidak Setuju</b>
1	1	0	0	0	0
2	0	1	0	0	0

3	1	0	0	0	0
4	1	0	0	0	0
5	1	0	0	0	0
6	0	1	0	0	0
7	1	0	0	0	0
8	0	1	0	0	0
9	0	1	0	0	0
10	0	1	0	0	0
11	1	0	0	0	0
12	1	0	0	0	0
13	1	0	0	0	0
14	1	0	0	0	0
15	0	0	0	0	0
<b>Jumlah</b>	<b>9</b>	<b>5</b>	<b>0</b>	<b>0</b>	<b>0</b>
<b>Nilai</b>	<b>5</b>	<b>4</b>	<b>3</b>	<b>2</b>	<b>1</b>
Rumus : $T \times P_n$	45	20	0	0	0
<b>Total Skor</b>	45 + 20 + 0 + 0 + 0 = 65				

Jadi, jika total skor penilaian pada pertanyaan No.1 diperoleh angka 65, maka penilaian *interpretasi responden* terhadap aplikasi pada pernyataan ini adalah hasil yang dihasilkan dengan menggunakan rumus  $Index \% =$

$$\frac{Total\ skor}{Y} \times 100 = \frac{65}{100} \times 100 = 65\%, \text{ berada dalam kategori "Setuju".}$$

2. Aplikasi ini memiliki fitur - fitur yang sesuai dengan kebutuhan pembelajaran saya

Tabel 5. 8 Nilai Responden Pertanyaan 2

Responden	Sangat Setuju	Setuju	Cukup Setuju	Tidak Setuju	Sangat Tidak Setuju
1	1	0	0	0	0
2	0	1	0	0	0
3	1	0	0	0	0
4	1	0	0	0	0
5	1	0	0	0	0
6	0	1	0	0	0
7	1	0	0	0	0
8	0	1	0	0	0
9	0	1	0	0	0
10	0	1	0	0	0
11	1	0	0	0	0
12	1	0	0	0	0
13	0	0	0	0	0
14	0	0	0	0	0
15	1	0	0	0	0
<b>Jumlah</b>	<b>8</b>	<b>5</b>	<b>0</b>	<b>0</b>	<b>0</b>
<b>Nilai</b>	<b>5</b>	<b>4</b>	<b>3</b>	<b>2</b>	<b>1</b>
Rumus : $T \times Pn$	40	20	0	0	0
<b>Total Skor</b>	40 + 20 + 0 + 0 + 0 = 60				



Jadi, jika total skor penilaian pada pertanyaan No. 2 diperoleh angka 60, maka penilaian *interpretasi responden* terhadap aplikasi pada pernyataan ini adalah hasil yang dihasilkan dengan menggunakan rumus  $Index \% = \frac{Total\ skor}{Y} \times 100 = \frac{60}{100} \times 100 = 60\%$ , berada dalam kategori “Setuju”.

3. Saya tidak merasa kesulitan dalam mengoperasikan aplikasi ini

Tabel 5. 9 Nilai Responden Pertanyaan 3

Responden	Sangat Setuju	Setuju	Cukup Setuju	Tidak Setuju	Sangat Tidak Setuju
1	1	0	0	0	0
2	0	1	0	0	0
3	1	0	0	0	0
4	1	0	0	0	0
5	1	0	0	0	0
6	0	1	0	0	0
7	1	0	0	0	0
8	0	1	0	0	0
9	0	1	0	0	0
10	0	1	0	0	0
11	1	0	0	0	0
12	1	0	0	0	0
13	0	0	0	0	0
14	0	0	0	0	0
15	1	0	0	0	0

<b>Jumlah</b>	<b>8</b>	<b>5</b>	<b>0</b>	<b>0</b>	<b>0</b>
<b>Nilai</b>	<b>5</b>	<b>4</b>	<b>3</b>	<b>2</b>	<b>1</b>
Rumus : $T \times Pn$	40	20	0	0	0
<b>Total Skor</b>	40 + 20 + 0 + 0 + 0 = 60				

Jadi, jika total skor penilaian pada pertanyaan No.3 diperoleh angka 60, maka penilaian *interpretasi responden* terhadap aplikasi pada pernyataan ini adalah hasil yang dihasilkan dengan menggunakan rumus  $Index \% = \frac{Total\ skor}{Y} \times 100 = \frac{60}{100} \times 100 = 60\%$ , berada dalam kategori “Setuju”.

4. Aplikasi ini memiliki tampilan yang menarik

Tabel 5. 10 Nilai Responden Pertanyaan 4

<b>Responden</b>	<b>Sangat Setuju</b>	<b>Setuju</b>	<b>Cukup Setuju</b>	<b>Tidak Setuju</b>	<b>Sangat Tidak Setuju</b>
1	1	0	0	0	0
2	0	1	0	0	0
3	1	0	0	0	0
4	1	0	0	0	0
5	1	0	0	0	0
6	0	1	0	0	0
7	1	0	0	0	0
8	0	1	0	0	0
9	0	1	0	0	0
10	0	1	0	0	0

11	1	0	0	0	0
12	1	0	0	0	0
13	1	0	0	0	0
14	1	0	0	0	0
15	1	0	0	0	0
<b>Jumlah</b>	<b>10</b>	<b>5</b>	<b>0</b>	<b>0</b>	<b>0</b>
<b>Nilai</b>	<b>5</b>	<b>4</b>	<b>3</b>	<b>2</b>	<b>1</b>
Rumus : $T \times Pn$	50	20	0	0	0
<b>Total Skor</b>	50 + 20 + 0 + 0 + 0 = 70				

Jadi, jika total skor penilaian pada pertanyaan No.4 diperoleh angka 70, maka penilaian *interpretasi responden* terhadap aplikasi pada pernyataan ini adalah hasil yang dihasilkan dengan menggunakan rumus *Index %* =  $\frac{\text{Total skor}}{Y} \times 100 = \frac{70}{100} \times 100 = 70\%$ , berada dalam kategori “Setuju”.

5. Tata letak dan bahasa yang digunakan pada aplikasi ini mudah di mengerti

Tabel 5. 11 Nilai Responden Pertanyaan 5

<b>Responden</b>	<b>Sangat Setuju</b>	<b>Setuju</b>	<b>Cukup Setuju</b>	<b>Tidak Setuju</b>	<b>Sangat Tidak Setuju</b>
1	1	0	0	0	0
2	0	1	0	0	0
3	1	0	0	0	0
4	1	0	0	0	0
5	1	0	0	0	0

6	0	1	0	0	0
7	1	0	0	0	0
8	0	1	0	0	0
9	0	1	0	0	0
10	0	1	0	0	0
11	1	0	0	0	0
12	1	0	0	0	0
13	1	0	0	0	0
14	1	0	0	0	0
15	0	0	0	0	0
<b>Jumlah</b>	<b>9</b>	<b>5</b>	<b>0</b>	<b>0</b>	<b>0</b>
<b>Nilai</b>	<b>5</b>	<b>4</b>	<b>3</b>	<b>2</b>	<b>1</b>
Rumus : $T \times Pn$	45	20	0	0	0
<b>Total Skor</b>	25 + 20 + 0 + 0 + 0 = 65				

Jadi, jika total skor penilaian pada pertanyaan No.5 diperoleh angka 60, maka penilaian *interpretasi responden* terhadap aplikasi pada pernyataan ini adalah hasil yang dihasilkan dengan menggunakan rumus  $Index \% =$

$$\frac{\text{Total skor}}{Y} \times 100 = \frac{65}{100} \times 100 = 65\%, \text{ berada dalam kategori "Setuju".}$$

## **BAB VI**

### **PENUTUP**

#### **6.1. Kesimpulan**

Berdasarkan penelitian yang telah dilakukan, maka dapat diambil kesimpulan sebagai berikut:

1. Dengan adanya aplikasi yang telah dibuat, pengurus dapat memantau kehadiran, dan kedisiplinan anggota pada saat pengumpulan tugas di setiap pertemuan.
2. Pengurus terbantu dengan adanya aplikasi ini dalam menyampaikan materi dan melakukan absensi pada setiap pertemuan.
3. Aplikasi telah selesai dibuat dan telah dilakukan pengujian *Black-Box Testing*, serta fungsi dalam aplikasi dapat berjalan dengan baik.
4. Pengujian aplikasi e-plugin menggunakan *Black-Box Testing* menunjukkan bahwa aplikasi mampu menampilkan informasi kesalahan pada fitur login, dan pengumpulan tugas serta dapat menampilkan halaman sesuai dengan fitur yang dipilih selanjutnya jika form yang diisi sudah sesuai dengan ketentuan.

#### **6.2. Saran**

Berikut adalah beberapa saran untuk pengembangan lebih lanjut terhadap penelitian tugas akhir ini.

1. Pembaharuan secara berkala demi terwujudnya aplikasi yang berfungsi secara optimal untuk kebutuhan organisasi.
2. Aplikasi ini dapat dikembangkan menjadi multi *platform*.

## DAFTAR PUSTAKA

- [1] D. Akhiruddin Pulungan, "Perancangan Aplikasi Absensi Menggunakan *Qr code* Berbasis *Android*," Jurnal FTIK, 2020.
- [2] Y. T. Widayati, "Aplikasi Teknologi QR (*Quick Response*) Code Implementasi Yang Univversal," KOMPUTAKI, vol. Vol.3, no. No.1, 2017.
- [3] H. R. Andilala, "Sistem Informasi Organisasi Intra Kampus Pada Universitas Muhammadiyah Bengkulu," Jurnal Sistem Informasi, vol. Vol.11, no. No.1, 2019.
- [4] Rouf, "Pembuatan Aplikasi Manajemen Organisasi Swara (*Amora*) Berbasis Desktop," Technomedia Journal, vol. Vol.4, no. No.1, 2019.
- [5] D. Andrian Putranda Rspuwandi, "Pengembangan Aplikasi *Android* Pada UKM Basket Universitas Diponegoro.," Jurnal Teknolofi dan Sistem Komputer, vol. Vol. 4, no. No.2, 2016.
- [6] B. P. S. D.S. Ramdan, "Perancangan dan Implementasi Aplikasi Sistem Informasi Pengelolaan Data UKM (Unit Kegiatan Mahasiswa) Berbasis Web di Politeknik TEDC Bandung," TEDC, vol. Vol14, no. No.1, 2020.
- [7] S. C. & K. Salahuddin, "A Literature Review of Factors Influencing Implementation of Management Information Systems in Organizations," International Journal of Business and Management, vol. Vol.12, no. No.8, 2017.116

- [8] Wahyudi, ".Pengembangan Sistem Informasi Penjualan Tiket Konser Musik Dengan Media Live Streaming Pada *Website* Motikdong.Com," Jurnal Aksara Public, vol. Vol.4, no. No.2, 2020.
- [9] H. Jagiyanto, Penelitian Aplikasi dan perkembangannya, Yogyakarta: Andi Ofset, 2001.
- [10] H. S. Nazrudin, pemrograman Aplikasi *Mobile* Smartphone dan Tablet PC Berbasis *Android*, Bandung: Informatika.
- [11] T. R. & V. G. Utomo, "Rancang Bangun Aplikasi *Mobile* Untuk Notifikasi Jadwal Kuliah Berbasis *Android*," Stmik Provisi Semarang, 2014.
- [12] N. M. G. S. & W. A. B, "Analisis Performa Aplikasi *Android* Pada Bahasa Pemrograman Java dan Analisis Performa Aplikasi *Android* Pada Bahasa Pemrograman Java dan *Kotlin*," 9th Industrial Research Workshop and National Seminar (IRONS), pp. 319-324, 2018.
- [13] P. P. Widodo, Menggunakan UML Unified Modeling Language, Bandung: Informatika, 2011.
- [14] O. N. & Y. Praharsari, "Aplikasi *Mobile* Web Geographic Information System (Webgis) Pariwisata Di Kabupaten Rote Ndao," Jurnal Magister Sistem Informasi, 2014.

# LAMPIRAN



## SURAT KESEPAKATAN BIMBINGAN TUGAS AKHIR

Kami yang bertanda tangan dibawah ini:

### Pihak Pertama

Nama : Akhmad Tantowi  
NIM : 17090031  
Program Studi : D IV Teknik Informatika

### Pihak Kedua

Nama : M. Nishom, M.Kom  
Status : Dosen  
NIDN : 0619048701  
Jabatan/Fungsional : Asisten Ahli  
Pangkat/Golongan : III / B

Pada hari Senin tanggal 11 Januari 2021 telah terjadi kesepakatan bahwa Pihak Kedua Bersedia menjadi Pembimbing I untuk Tugas Akhir Pihak Pertama. Pihak Pertama wajib melakukan bimbingan Tugas Akhir sekurang-kurangnya 1 (satu) kali dalam 1 (satu) minggu, adapun waktu dan tempat pelaksanaan disepakati antar pihak.

Tegal, 11 Januari 2021

Pihak Pertama



Akhmad Tantowi

Pihak Kedua



M. Nishom, M.Kom

Mengetahui,

Ketua Program Studi Teknik Informatika



Slamet Wiyono, S.Pd, M.Eng

08.015.222

## SURAT KESEPAKATAN BIMBINGAN TUGAS AKHIR

Kami yang bertanda tangan dibawah ini:

### Pihak Pertama

Nama : Akhmad Tantowi  
NIM : 17090031  
Program Studi : D IV Teknik Informatika


### Pihak Kedua

Nama : Romi Muharyono, S.Ag  
Status : Dosen  
NIDN : -  
Jabatan/Fungsional : Praktisi  
Pangkat/Golongan : -


Pada hari Senin tanggal 11 Januari 2021 telah terjadi kesepakatan bahwa Pihak Kedua Bersedia menjadi Pembimbing II untuk Tugas Akhir Pihak Pertama. Pihak Pertama wajib melakukan bimbingan Tugas Akhir sekurang-kurangnya 1 (satu) kali dalam 1 (satu) minggu, adapun waktu dan tempat pelaksanaan disepakati antar pihak.

Tegal, 11 Januari 2021

Pihak Pertama

  
Akhmad Tantowi

Pihak Kedua

  
Romi Muharyono, S.Ag

Mengetahui,  
Ketua Program Studi Teknik Informatika

  
Slamet Wiyono, S.Pd, M.Eng  
08.015.222






**D IV TEKNIK INFORMATIKA  
POLITEKNIK HARAPAN BERSAMA**

**LEMBAR BIMBINGAN TUGAS AKHIR**

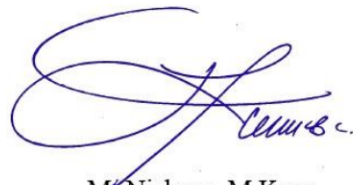
**Nama** : Akhmad Tantowi  
**NIM** : 17090031  
**No. Ponsel** : 0857-8648-1288  
**Judul TA** : Aplikasi E - PLUGIN Berbasis Android  
**Dosen Pembimbing I** : M. Nishom, M.Kom

No	Tanggal	Pemeriksaan	Perbaikan Yang Perlu Dilakukan	Paraf Pembimbing
1	29 / 1 / 2021	<ul style="list-style-type: none"> <li>Konsep Aplikasi</li> </ul>	<ul style="list-style-type: none"> <li>Konsep Aplikasi</li> <li>Latar Belakang</li> </ul>	
2	2 / 3 / 2021	<ul style="list-style-type: none"> <li>Latar Belakang</li> </ul>	<ul style="list-style-type: none"> <li>Perbaikan Latar Belakang</li> <li>Laporan BAB I</li> <li>Ganti Judul "Aplikasi e-plugin Berbasis Android"</li> </ul>	
3	25 / 3 / 2021	<ul style="list-style-type: none"> <li>Laporan BAB I</li> </ul>	<ul style="list-style-type: none"> <li>Use Case Diagram</li> </ul>	
4	16 / 4 / 2021	<ul style="list-style-type: none"> <li>Use Case Diagram</li> </ul>	<ul style="list-style-type: none"> <li>Penataan Use Case Diagram</li> <li>Pembuatan Aplikasi</li> </ul>	

5	17 / 6 / 2021	<ul style="list-style-type: none"> <li>• Aplikasi</li> </ul>	<ul style="list-style-type: none"> <li>• Archive Materi</li> <li>• Perbaikan Pengiriman Tugas Member</li> </ul>	
6	14 / 7 / 2021	<ul style="list-style-type: none"> <li>• Penyempurnaan Aplikasi</li> </ul>	<ul style="list-style-type: none"> <li>• Penulisan Rumusan Masalah</li> <li>• Perbaikan Font</li> <li>• Penyempurnaan Aplikasi</li> </ul>	
7	19 / 3 / 2021	<ul style="list-style-type: none"> <li>• Laporan</li> </ul>	<ul style="list-style-type: none"> <li>• Kerapihan Laporan</li> <li>• Abstrak Laporan</li> <li>• Unit Testing</li> </ul>	

Tegal, ..... 2021

Dosen Pembimbing I



M. Nishom. M. Kom

09.017.337

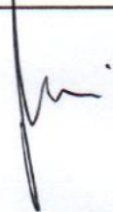





**D IV TEKNIK INFORMATIKA  
POLITEKNIK HARAPAN BERSAMA**

**LEMBAR BIMBINGAN TUGAS AKHIR**

**Nama** : Akhmad Tantowi  
**NIM** : 17090031  
**No. Ponsel** : 0857-8648-1288  
**Judul TA** : Aplikasi E - PLUGIN Berbasis Android  
**Dosen Pembimbing II** : Romi Muharyono, S.Ag

No	Tanggal	Pemeriksaan	Perbaikan Yang Perlu Dilakukan	Paraf Pembimbing
1	29/1 2021	- Mockup - Laporan BAB 1	- fungsi Aplikasi - feedback	
2	13/1 2021	- Konsep aplikasi - feedback aplikasi	- Transisi - Ganti Judul	
3	9/1 2021	- Transisi UI - Judul	- Aplikasi e-plugin Berbasis Android	
4	25/5 2021	- Aplikasi	- pembenahan bahasa	

5.	14/7			
6.	17/7	- Penggunaan bahasa di aplikasi	- Pembetulan bahasa dan menu	
7.	19/7	- Penambahan Atur aplikasi	- Penambahan arsip materi	
8.		- Testing Aplikasi	- penyempurnaan arsip aplikasi	

--	--	--	--	--

Tegal, ...29 Juli..... 2021

Dosen Pembimbing II



Romi Munaryono, S.Ag

