BAB 3
ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Sistem

Analisis sistem dapat didefinisikan sebagai penguraian dari suatu sistem informasi yang utuh ke dalam bagian-bagian komponenya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapakan sehingga dapat diusulkan perbaikan-perbaikanya. Di dalam tahap analisis sistem terdapat langkah-langkah yang harus dilakukan, antara lain :

1. *Identify*, yaitu mengidentifikasi masalah.
2. *Understand*, yaitu memahami kerja dari sistem yang ada.
3. *Analyze*, yaitu menganalisis sistem.
4. *Report*, yaitu membuat laporan analisis

Analisis sistem merupakan tahap untuk mempelajari interaksi sistem yang terdiri dari pelaku proses dalam sistem, prosedur, data serta informasi yang terkait. Analisis dilakukan terhadap sistem yang sedang berjalan sebagai dasar perancangan atau perbaikan sistem lama. Dalam analisis sistem dilakukan penguraian dari suatu sistem informasi yang utuh ke dalam bagian- bagian komponen dengan tujuan untuk mengindentifikasi dan mengevaluasi permasalahan- permasalahan sehingga ditentukan kelemahan-kelemahan yang diharapkan dapat diusulkan perbaikannya.

Bab ini akan menguraikan proses analisis sistem dan penerapan aplikasi sebagai sistem yang dapat membantu dalam proses merekomendasikan perencanaan *travelling* dengan menggunakan metode Case Based Reasoning (CBR) sebagai metode pendukung untuk menyelesaikan permasalahan pada penelitianini.

3.1.1 Analisis Masalah

Kegiatan *travelling* di Indonesia saat ini mulai digemari oleh banyak orang, mulai dari anak muda sampai orang tua dan dari berbagai profesi. Tujuan *travelling* mereka pun bervariasi, mulai dari dalam kota sampai dengan ke luar pulau di Indonesia. Namun sebagian besar masih bermasalah dalam hal perencanaan kegiatan *travelling* dan mengatur keuangan ketika *travelling*.

Membuat perencanaan yang matang sebelum melakukan kegiatan *travelling* sangat diperlukan untuk kelancaran kegiatan dan sesuai dengan kemampuan keuangan. Perencanaan dibutuhkan untuk membantu memaksimalkan waktu liburan untuk pergi ketempat-tempat yang ingin dikunjungi, sehingga kegiatan *travelling* lebih terarah dan waktu tidak terbuang untuk memikirkan destinasi selanjutnya.

Selain itu pengontrolan pengeluaran ketika kegiatan *travelling* berlangsung pun sangat dibutuhkan agar pengeluaran selama travelling bisa terkontrol dan bisa dievaluasi untuk kegiatan *travelling* berikutnya. Untuk memperoleh informasi *travelling* yang lengkap dan sesuai dengan keuangan masih sulit untuk dicari dan diperlukan usaha yang lebih untuk mencari dari berbagai sumber yang tersebar di internet.

3.1.2 Analisis Aturan Bisnis

Analisis aturan bisnis berisi tentang pemaparan proses bisnis perangkat lunak get trip. Adapun aturan bisnis yang ada yaitu:

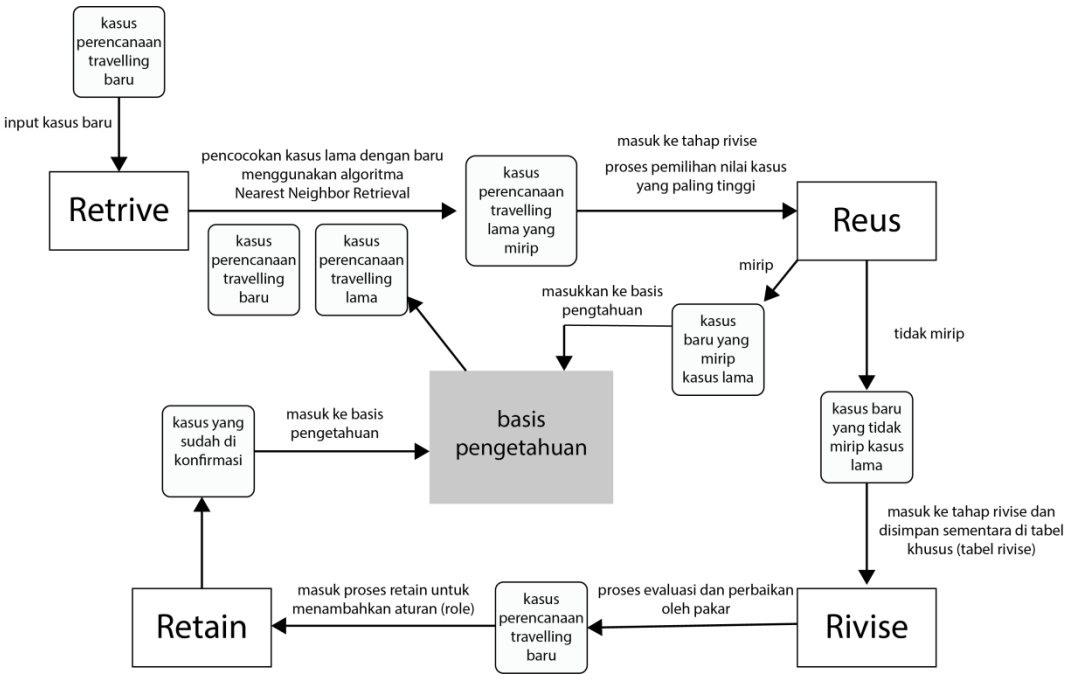
1. Pengguna harus memiliki akun Get Trip untuk dapat membuat perencanaan, menyimpan pengeluaran.
2. Simpan pengeluaran dapat dilakukan bila pengguna sudah membuat perencanaan dan dapat menambah perencanaan lainnya di luar perencanaan yang suah dibuat.
3. Riwayat perjalanan pengguna dapat diperoleh bila pengguna sudah selesai menyimpan semua pengeluaran perjalannanya.

4. Informasi riwayat perjalanan pengguna lainnya dapat diakses oleh pengguna tanpa harus memiliki akun Get Trip.
5. Setiap penyimpanan pengeluaran, pengguna wajib mengambil foto dan lokasi kegiatan travelling.
6. Tanggal pembuatan perencanaan minimal pada tanggal saat pembuatan perencanaan.
7. Lama waktu konfirmasi pendaftaran adalah 7 hari, setelah itu data pengguna akan dihapus.

3.1.3 Analisis Metode Case Based Reasoning (CBR)

Dalam penelitian ini telah diterapakan suatu metode untuk mengatasi ketidakpastian dengan sistem penalaran berbasis kasus (case-based reasoning). Yang menjadi basis pengetahuan pada case-based reasoning adalah fakta-fakta berupa kasus-kasus sebelumnya yang pernah ada dan serangkaian alur untuk memeriksa, menghitung, serta menyimpulkan suatu solusi dari permasalahan yang diberikan. Tahapan pada case-based reasoning ada 4 yaitu: retrieve, reuse, revise dan retain.

Pencarian solusi kasus di lakukan pada tahapan retrieve dan reuse. Tahapan retrieve melakukan pencarian tingkat kemiripan kasus inputan perencanaan travelling yang diberikan pengguna dengan kasus-kasus yang ada pada knowledge base untuk mencari kasus yang memiliki persamaan (similaritas) tertinggi. Hanya kasus yang memiliki persamaan (similaritas) dengan basis kasus yang akan diproses di tahapan reuse untuk disarankan solusi bagi kasus baru. Tahapan revise dan retain merupakan tahapan dimana solusi yang diberikan pada kasus inputan pengguna sebelumnya di revisi dan diputuskan untuk dijadikan knowledge baru atau tidak. Untuk lebih jelasnya dalam tahapan metode case based reasoning ini dapat dilihat pada gambar dibawah ini.



Gambar 3. 1 Tahapan Metode Case Based Reasonig

3.1.3.1 Retrieval

Retrieval yang digunakan dalam penelitian ini adalah menelusuri kembali kasus (riwayat perencanaan travelling) lama yang paling menyerupai/relevan dengan kasus baru. Pencocokan dilakukan dengan cara menginputkan data perencanaan travelling. Jika pencocokan nilai bobot kasus sama atau hampir sama dengan basis kasus, maka akan dilakukan tahap reuse yaitu melakukan perhitungan similarity dari bobot tiap atribut yang menghasilkan kemungkinan perencanaan travelling solusi dari kasus baru. Asumsi pada penelusuran ini adalah kasus yang mirip akan memiliki solusi yang mirip. Nilai bobot ini diberikan sebagai gambaran seberapa penting atribut tersebut dalam penentuan solusi dari kasus yang sama. Atribut yang dimiliki oleh tiap-tiap kasus adalah budget liburan, asal keberangkatan liburan, jenis liburan dan lama liburan. Nilai bobot mulai dari angka 0 sampai dengan angka 1 dimana angka 0 merupakan nilai terendah dan angka 1 merupkan nilai tertinggi. Berikut ini adalah tabel data atribut dan nilainya.

Tabel 3. 1 Data Atribut Dan Nilainya

NO.	Nama Atribut	Nilai Atribut
1	Budget Liburan	a. BL1 : Budget dari Rp.0 – Rp. 500.000 b. BL2 : Budget dari Rp. 500.001 – Rp. 1.000.000 c. BL3 : Budget dari Rp. 1.000.001 – Rp. 1.500.000 d. BL4 : Budget dari Rp. 1.500.001 – Rp. 2.000.000 e. BL5 : Budget dari Rp. 2.000.001 – Rp. 2.500.000 f. BL6 : Budget dari Rp. 2.500.001 – Rp. 3.000.000 g. BL7 : Budget lebih dari Rp. 3.000.000
2	Asal Keberangkatan Liburan	Kota dari traveller ketika memulai perjalanan travellingnya
3	Jenis Liburan	a. Warisan Budaya b. Lingkungan Alam c. Kota Metropolitan d. Wisata Religi
4	Lama Liburan	a. HR1 : 1 Hari b. HR2 : 2 hari c. HR3 : 3 hari d. HR4 : 4 Hari e. HR5 : 5 hari f. HR6 : lebih dari 5 hari

Dalam penelitian ini pembobotan budget liburan, asal keberangkatan liburan, jenis liburan dan lama liburan dihitung menggunakan metode pembobotan Rank Order Centroid (ROC) dimana pendefinisian bobot berdasarkan pada atribut yang paling berpengaruh untuk menentukan jenis rekomendasi, budget liuran >= asal liburan >= jenis liburan >= lama liburan . Berikut ini adalah perhitungan nilai bobot menggunakan metode ROC.

Tabel 3. 2 Perhitungan Nilai Bobot Atribut

Nama Atribut (k)	Tingkat Kepentingan	Persamaan ROC $Wk = 1/k \sum_{n=1}^{k} \left(\frac{1}{i}\right)$	Bobot (W)
Budget Liburan	1	$W1 = \frac{(\frac{1}{1}+\frac{1}{2}+\frac{1}{3}+\frac{1}{4})}{4} = \frac{1+0,5+0,3+0,25}{4}$	0,52
Asal Liburan	2	$W2 = \frac{(\frac{1}{2}+\frac{1}{3}+\frac{1}{4})}{4} = \frac{0,5+0,3+0,25}{4}$	0,27
Jenis Liburan	3	$W3 = \frac{(\frac{1}{3}+\frac{1}{4})}{4} = \frac{0,3+0,25}{4}$	0,15
Lama Liburan	4	$W4 = \frac{(\frac{1}{4})}{4} = \frac{0,25}{4}$	0,06

Dari hasil perhitungan maka di peroleh nilai-nilai bobot untuk ke empat atribut. Berikut ini adalah tabel bobot atribut.

Tabel 3. 3 Bobot Atribut

Nama Atribut	Bobot	Persentase (bobot x 100%)
Budget	0,52	52%
Asal Liburan	0,27	27%
Jenis Liburan	0,15	15%
Lama Liburan	0,06	6%

Dalam pencarian Nearest Neighbor, kedekatan fitur-fitur kasus juga akan dihitung. Nilai kedekatan biasanya berada pada nilai 0 sampai dengan 1

dimana nilai 0 artinya kedua kasus tidak mirip sedangkan nilai 1 artinya kedua kasus mirip. Penelusuranan pada aplikasi ini menggunakan teknik Similarity (problem,case) pada algoritma k-nearest neighbor sebagai berikut :

$$\text{Similarity (problem,case)} = \frac{(S1*W1)+(S2*W2)+(S3*W3)+(S4*W4)}{W1+W2+W3+W4}$$

Keterangan:

S1 : *similarity* (nilai kemiripan) budget liburan yaitu 1 (sama) dan 0 (beda)

S2 : *similarity* (nilai kemiripan) asal keberangkatan liburan yaitu 1 (sama) dan 0 (beda)

S3 : *similarity* (nilai kemiripan) jenis liburan yaitu 1 (sama) dan 0 (beda)

S4 : *similarity* (nilai kemiripan) lama liburan yaitu 1 (sama) dan 0 (beda)

W1 : bobot budget liburan yaitu 0,52

W2 : bobot asal keberangkatan liburan yaitu 0,27

W3 : bobot jenis liburan yaitu 0,15

W4 : bobot lama liburan yaitu 0,06

Berikut ini contoh kasus perencanaan travelling, dimana sudah terdapat 4 buah kasus sebagai base pengetahuan dapat dilihat pada Tabel 3.4 Contoh base Pengetahuan Perencanaan Travelling, dan 1 buah kasus baru yang akan dicari kemiripannya dapat dilihat pada Tabel 3.5 Contoh Kasus baru .

Tabel 3. 4 Contoh Base Pengetahuan Perencanaan Travelling

No	Id Kasus	Judul	Jenis Liburan	Asal	Budget Rp	Lama Liburan /hari
1	1	Trip to pangandaran	Lingkungan Alam	Bandung	500.000	2
2	2	Trip to papandayan	Lingkungan Alam	Jakarta	200.000	2
3	3	Trip to yogyakarta	Warisan Budaya	Bandung	1.200.000	4
4	4	Trip to Lombok	Lingkungan Alam	Bandung	2.500.000	4

Tabel 3. 5 Contoh Kasus Baru

No	Id Kasus	Judul	Jenis Liburan	Asal	Budget Rp	Lama Liburan /hari
1	X		Lingkungan Alam	Bandung	2.200.000	3

1. Perhitungan Kasus 1

Tabel 3. 6 Nilai Similarity Atribut Kasus X dan 1

Id Kasus : X	Similarity (X,1)	Id Kasus : 1
Lingkungan Alam	1	Lingkungan Alam
Bandung	1	Bandung
BL5 : Rp. 2.200.000	0	BL1 : Rp. 500.000
HR3 : 3 Hari	0	HR2: 2 Hari

Bobot Atribut :

Budget Liburan : 0,52 Asal Keberangkatan Liburan : 0,27

Jenis Liburan : 0, 15 Lama Liburan : 0,06

$$\text{Similarity (X,1)} = \frac{(S1*W1)+(S2*W2)+(S3*W3)+(S4*W4)}{W1+W2+W3+W4}$$

$$= \frac{(0*0{,}52)+(1*0{,}27)+(1*0{,}15)+(0*0{,}06)}{0{,}52+0{,}27+0{,}15+0{,}06}$$

$$= \frac{0+0{,}27+0{,}15+0}{1}$$

$$= \frac{0{,}42}{1} = 0{,}42$$

2. Perhitungan Kasus 2

Tabel 3. 7 Nilai Similarity Atribut Kasus X dan 2

Id Kasus : X	Similarity (X,1)	Id Kasus : 2
Lingkungan Alam	1	Lingkungan Alam
Bandung	0	Jakarta
BL5 : Rp. 2.200.000	0	BL1 : Rp. 200.000
HR3 : 3 Hari	0	HR2: 2 Hari

Bobot perencanaan travelling X :

Budget Liburan : 0,52 Asal Keberangkatan Liburan : 0,27

Jenis Liburan : 0, 15 Lama Liburan : 0,06

$$\text{Similarity (X,2)} = \frac{(S1*W1)+(S2*W2)+(S3*W3)+(S4*W4)}{W1+W2+W3+W4}$$

$$= \frac{(0*0{,}52)+(1*0{,}27)+(0*0{,}15)+(0*0{,}06)}{0{,}52+0{,}27+0{,}15+0{,}06}$$

$$= \frac{0+0{,}27+0+0}{1}$$

$$= \frac{0{,}27}{1} = 0{,}27$$

3. Perhitungan kasus 3

Tabel 3. 8 Nilai Similarity Atribut Kasus X dan 3

Id Kasus : X	Similarity (X,1)	Id Kasus : 3
Lingkungan Alam	1	Warisan Budaya
Bandung	1	Bandung
BL5 : Rp. 2.200.000	0	BL3 : Rp.1. 200.000
HR3 : 3 Hari	0	HR4: 4 Hari

Bobot perencanaan travelling X :

Budget Liburan : 0,52 Asal Keberangkatan Liburan : 0,27

Jenis Liburan : 0, 15 Lama Liburan : 0,06

$$\text{Similarity (X,3)} = \frac{(S1*W1)+(S2*W2)+(S3*W3)+(S4*W4)}{W1+W2+W3+W4}$$

$$= \frac{(0*0{,}52)+(1*0{,}27)+(1*0{,}15)+(0*0{,}06)}{0{,}52+0{,}27+0{,}15+0{,}06}$$

$$= \frac{0+0{,}27+0{,}15+0}{1}$$

$$= \frac{0{,}42}{1} = 0{,}42$$

4. Perhitungan kasus 4

Tabel 3. 9 Nilai Similarity Atribut Kasus X dan 4

Id Kasus : X	Similarity (X,1)	Id Kasus : 4
Lingkungan Alam	1	Lingkungan Alam
Bandung	1	Bandung
BL5 : Rp. 2.200.000	1	BL5 : Rp .2.500.000
HR3 : 3 Hari	0	HR4: 4 Hari

Bobot perencanaan travelling X :

Budget Liburan : 0,52 Asal Keberangkatan Liburan : 0,27

Jenis Liburan : 0, 15 Lama Liburan : 0,06

$$\text{Similarity (X,4)} = \frac{(S1*W1)+(S2*W2)+(S3*W3)+(S4*W4)}{W1+W2+W3+W4}$$

$$= \frac{(1*0{,}52)+(1*0{,}27)+(1*0{,}15)+(0*0{,}06)}{0{,}52+0{,}27+0{,}15+0{,}06}$$

$$= \frac{0{,}52+0{,}27+0{,}15+0}{1}$$

$$= \frac{0{,}94}{1} = 0{,}94$$

3.1.3.2 Reuse

Dari hasil perhitungan di tahap retrive kemudian diperoleh nilai-nilai kemiripan tiap kasus yang ditunjukan pada tabel berikut ini

Tabel 3. 10 Nilai Similarity Perbandingan Kasus

Id Kasus	Nilai Similarity dengan kasus baru (X)	Persentase (nilai similarity x 100%)
1	0,42	42%
2	0,27	27%
3	0,42	42%
4	0,94	94%

Nilai-nilai tersebut kemudian dibandingkan dengan jumlah nilai bobot dengan nilai bobot asal yaitu 0,79 dan nilai kemiripan yang lebih tinggi atau sama akan menjadi solusi yang akan diberikan. Dari tabel diatas nilai kemiripan yang lebih dari 0,79 ditunjukan pada id kasus 4 yaitu, Judul : trip to lombok, Asal Keberangkatan Liburan : Bandung, Jenis Liburan : Wisata Alam, Budget : Rp. 2.500.000 dan Lama Liburan : 4 hari. Selanjutnya pengguna sendiri yang menafsirkan solusi tersebut dan mengambil keputusan. Dalam hal ini proses *case retrieve* dikerjakan oleh komputer, namun *case reasoning*-nya diserahkan pada pengguna [**17**].

3.1.3.3 Revise

Pada contoh ini id kasus 4 sudah menghasilkan solusi dengan tingkat kemiripan 94%, jadi rekomendasi yang dihasilkan dapat langsung diberikan. Tetapi jika ternyata setelah dilakukan proses perhitungan dan tidak ada kasus yang mirip dengan kasus baru tersebut maka dilakukan proses revise. Informasi berupa masukan rencana pada kasus baru yang tidak ditemukan kemiripannya dengan basis pengetahuan (rule) tersebut ditampung pada suatu tabel khusus

(tabel revise) yang selanjutnya akan dievaluasi dan diperbaiki kembali oleh pakar untuk menemukan solusi yang tepat.

3.1.3.4 Retain

Pada tahap ini pakar mulai menambah aturan dengan memasukkan data kasus baru yang sudah ditemukan solusinya tersebut ke dalam basis pengetahuan yang nantinya dapat digunakan untuk kasus berikutnya yang memiliki permasalahan yang sama.

3.1.4 Analisis Kebutuhan Non Fungsional

Analisis kebutuhan non-fungsional merupakan analisis yang dibutuhkan untuk dapat menentukan spesifikasi dari kebutuhan sistem. Spesifikasi ini meliputi elemen atau perangkat-perangkat yang dibutuhkan untuk sistem yang akan dibangun sampai sistem tersebut dapat diimplementasikan. Analisis kebutuhan ini juga menentukan spesifikasi masukan yang diperlukan sistem, keluaran yang akan dihasilkan sistem dan proses yang dibutuhkan untuk mengolah masukan sehingga dapat menghasilkan suatu keluaran yang diinginkan. Kebutuhan non-fungsional terbagi menjadi beberapa analisis yaitu analisis perangkat keras, analisis perangkat lunak dan analisis pengguna.

3.1.4.1 Analisis Perangkat Keras (Hardware)

Komponen perangkat keras yang digunakan untuk membuat perangkat lunak menggunkan laptop dengan spesifikasi sebagai berikut :

1. Processor Intel Core I3 – 2350M, 2.3GHz.
2. VGA Nvidia Geforce 610M
3. RAM DDR3 6GB.
4. Harddisk 500GB.

Spesifikasi perangkat keras yang disarankan untuk menjalankan perangkat lunak di platform mobile sebagai berikut :

1. Processor 830 MHz ARMv6
2. Memory Internal : Free Space 512 MB
3. Memory Eksternal : Free Space 512 MB
4. RAM 512 MB
5. Layar 3,5 in
6. OS Android 4.2 Jelly Bean

Spesifikasi perangkat keras yang disarankan untuk menjalankan perangkat lunak di platform website sebagai berikut :

1. Processor Intel Dual Core 2.3 Ghz
2. RAM 1 GB
3. Harddisk dengan free space 1 GB
4. Monitor 15 in
5. Keyboard dan Mouse

Spesifikasi perangkat keras yang digunakan untuk implementasi di platform mobile mengunakan smartphone Android, dengan spesifikasi sebagai berikut :

1. Processor Quad-core 1.2 GHz.
2. Memory Internal : 8 GB
3. RAM 1 GB.
4. Memory Eksternal : 16 GB.
5. OS Android 4.4.2 Kitkat.
6. Layar IPS LCD capacitive touchscreen, 16M colors (720 x 1280 pixel, 4.7 inches (~312 ppi pixel density)).

Spesifikasi perangkat keras yang digunakan untuk implementasi di platform website mengunakan laptop, dengan spesifikasi sebagai berikut :

1. Processor Intel Core I3 – 2350M, 2.3GHz.
2. VGA Nvidia Geforce 610M

3. RAM DDR3 6GB.
4. Harddisk 500GB

3.1.4.2 Analisis Perangakat Lunak (Software)

Komponen perangkat lunak yang digunakan untuk membuat aplikasi dan simulasi program adalah sebagai berikut :

1. Sistem Operasi Microsoft Windows 8.1 Profesional 64-bit.
2. Browser Google Chrome
3. Java Runtime Environment 1.8
4. Java Development Kit 1.8
5. IDE Android Studio
6. Android Development Tools (ADT) 23.0.6
7. Android SDK 4.4 (Kitkat)
8. Adobe Illustrator CS 6
9. Microsoft Visio 2010
10. Star UML 5.0.2

3.1.4.3 Analisis Pengguna

Suatu aplikasi akan berjalan dengan optimal apabila ditunjang oleh perangkat pikir yang memiliki kemampuan dalam menjalankan aplikasi yang bersangkutan. Perangkat lunak ini akan digunakan oleh 2 jenis pengguna yaitu administrator dan pengguna umum. Karakteristik pengguna yang dibutuhkan adalah sebagai berikut :

Tabel 3. 11 Karakteristik Pengguna

Pengguna	Karakteristik yang dibutuhkan
Administrator	1. Mengerti dalam menggunakan komputer 2. Mengerti cara mengoperasikan perangkat lunak get trip.
Pengguna Umum	1. Mampu mengoperasikan *Smartphone* Android.

	2. Menguasai Bahasa Indonesia atau Bahasa Inggris. 3. Mampu melakukan kegiatan *travelling.* 4. Terbiasa menggunakan aplikasi-aplikasi mobile social media.

3.1.5 Analisis Kebutuhan Fungsional

Analisis sistem yang dilakukan menggunakan *tools* UML, adapun tahapan analisis sistem menggunakan UML meliputi *use case diagram, Use Case Scenario, activity diagram,* dan *class diagram.* Analisis kebutuhan fungsional di perangkat lunak get trip meliputi 2 platform yaitu : platform mobile android untuk antar muka ke pengguna dan platform website untuk antar muka ke administrator. Berikut ini adalah spesifikasi kebutuhan perangkat lunak fungsional dan non fungsional di platform mobile dan website perangkat lunak get trip.

Tabel 3. 12 Spesifikasi Perangkat Lunak Fungsional di Platform Mobile

Spesifikasi Kebutuhan Perangkat Lunak Fungsional di Platform Mobile	
SKPL-F	Keterangan
001	Sistem menyediakan Sign Up
002	Sistem menyediakan Login
003	Sistem mampu membuat perencanaan
004	Sistem mampu memperbarui data perencanaan
005	Sistem mampu menghapus data perencanaan
006	Sistem mampu membuat rekomendasi
007	Sistem mampu menambah data pengeluaran
008	Sistem mampu memperbarui data pengeluaran

009	Sistem mampu menghapus data pengeluaran
010	Sistem mampu menambah lokasi
011	Sistem mampu mengambil foto melalui kamera
012	Sistem mampu mengambil data lokasi melalui API Google
013	Sistem mampu mencari data riwayat
014	Sistem mampu menampilkan data riwayat pengguna
015	Sistem mampu menampilkan profil pengguna
016	Sistem mampu menampilkan tips travelling

Tabel 3. 13 Spesifikasi Perangkat Lunak Non Fungsional di Platform Mobile

Spesifikasi Kebutuhan Perangkat Lunak Non Fungsional di Platform Mobile	
SKPL-NF	Keterangan
001	Sistem bisa diakses selama 24 jam tanpa berhenti
002	sistem dapat dijalankan di versi minimum android 2.2 (Froyo)
003	Sistem dapat dijalankan optimal di smartphone yang memiliki kamera
004	sistem menyediakan 2 bahasa, yaitu bahasa inggris dan bahasa indonesia
005	Sistem menyediakan 2 mata uang, yaitu IDR dan USD
006	Sistem hanya dapat digunakan oleh satu akun di satu smartphone
007	Sistem mampu menyimpan banyak riwayat perjalanan pengguna

Tabel 3. 14 Spesifikasi Perangkat Lunak Fungsional di Platform Website

Spesifikasi Kebutuhan Perangkat Lunak Fungsional di Platform Website	
017	Sistem mampu menampilkan data pengguna
018	Sistem mampu menampilkan data riwayat travelling
019	Sistem mampu membuat tips travelling
020	Sistem mampu memperbarui tips travelling
021	Sistem mampu menghapus tips travelling

Spesifikasi Kebutuhan Perangkat Lunak Non Fungsional di Platform Website	
008	Sistem dapat diakses 24 jam tanpa berhenti
009	Sistem hanya dapat digunakan oleh pengguna administrator
010	Sistem mampu mengelola banyak pengguna

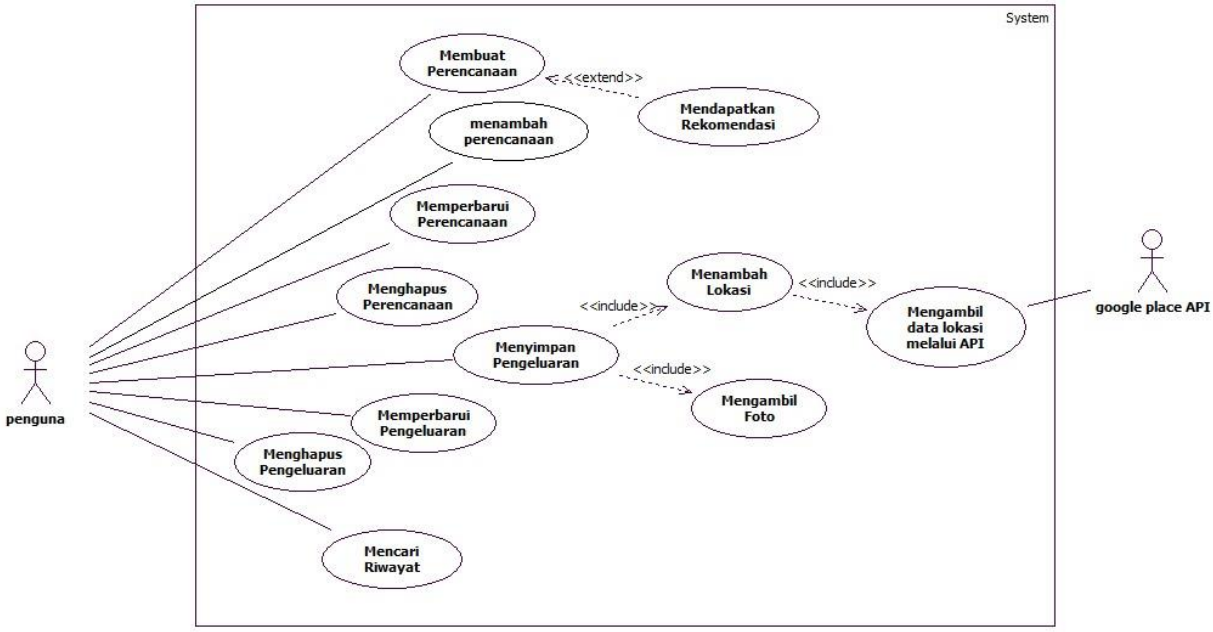
3.1.5.1 Analisis Kebutuhan Fungsional di Platform Mobile

Analisis kebutuhan fungsional perangkat lunak get trip di platform mobile android dilakukan menggunakan *tools* UML, adapun tahapan analisis sistem menggunakan UML meliputi *use case diagram, Use Case Scenario, activity diagram*, dan *class diagram.* Analisis kebutuhan perangkat lunak get trip di platform mobile android akan dijelaskan sebagai berikut :

3.1.5.1.1 Use Case Diagram

Use Case Diagram merupakan konstruksi untuk mendeskripsikan hubungan - hubungan yang terjadi antar *actor* dengan aktifitas yang terdapat pada sistem. Sasaran pemodelan *use case* diantaranya adalah mendefinisikan kebutuhan fungsional dan operasional sistem dengan mendefinisikan *scenario* penggunaan

yang disepakati antara pemakai dan pengembang. Dari analisis pengguna aplikasi yang ada maka *use case diagram* untuk perangkat lunak get trip di platform mobile dapat dilihat pada gambar 3.2 Use Case Diagram Perangkat Lunak Get Trip di Platform Mobile Android.



Gambar 3. 2 Use Case Diagram Perangkat Lunak Get Trip di Platform Mobile Android

3.1.5.1.2 Use Case Scenario

Use Case Scenario menjelaskan skenario dari setiap proses bisnis yang digambarkan pada use case diagram. Berdasarkan use case diagram pada Gambar 3. 1 Use Case Diagram Perangkat Lunak Get Trip di Platform Mobile Android, maka use case scenario untuk aplikasi yang akan dibangun akan dijelaskan sebagai berikut :

1. Use Case Scenario Membuat Perencanaan

Use case scenario membuat perencanaan menggambarkan langkah-langkah pengguna untuk membuat perencanaan *travelling*. Use case scenario membuat perencanaan dapat dilihat pada Tabel 3.15 Use Case Scenario Membuat perencanaan.

Tabel 3. 15 Use Case Scenario Membuat Perencanaan

Use Case Name	**Membuat Perencanaan**	
Related Requirements	SKPL-F-01	
Goal In Context	Pengguna membuat perencanaan travelling	
Precondition	Pengguna belum membuat perencanaan travelling	
Successful End Condition	Pengguna berhasil menyimpan data perencanaan travelling	
Failed End Condition	Pengguna gagal menyimpan data perencanaan travelling	
Actors	Pengguna	
Trigger	Pengguna mengetap tombol buat rencana	
Main Flow	Step	Action
	1	Pengguna mengetap tombol buat perencanaan
	2	Sistem menampilkan *form* buat perencanaan travelling
	3	Pengguna mengisi *form* buat perencanaan travelling
	4	Pengguna mengetap tombol selesai
	5	Sistem melakukan validasi data perencanaan travelling
	6	Sistem menampilkan halaman perencanaan saya
Extension	Step	Branching Action
	5.1	Sistem tidak melakukan validasi data perencanaan travelling
	5.2	Data perencanaan travelling ditolak

2. Use Case Scenario Menambah Perencanaan

Use case scenario menambah perencanaan menggambarkan langkah-langkah pengguna untuk menambah perencanaan *travelling*. Use case scenario memperbarui perencanaan dapat dilihat pada Tabel 3.16 Use Case Scenario Menambah Perencanaan.

Tabel 3. 16 Use Case Scenario Menambah Perencanaan

Use Case Name	**Menambah Perencanaan**	
Related Requirements	SKPL-F-02	
Goal In Context	Pengguna menambah perencanaan travelling	
Precondition	Pengguna sudah membuat perencanaan travelling	
Successful End Condition	Pengguna berhasil menambah data perencanaan travelling	
Failed End Condition	Pengguna gagal menambah data perencanaan travelling	
Actors	Pengguna	
Trigger	Pengguna mengetap tombol buat rencana	
Main Flow	Step	Action
	1	Pengguna mengetap tombol tambah perencanaan
	2	Sistem menampilkan *form* tambah perencanaan travelling
	3	Pengguna mengisi *form* tambah perencanaan travelling
	4	Pengguna mengetap tombol selesai
	5	Sistem melakukan validasi data perencanaan travelling
	6	Sistem menampilkan halaman perencanaan saya
Extension	Step	Branching Action
	5.1	Sistem tidak melakukan validasi data perencanaan travelling
	5.2	Data perencanaan travelling ditolak

3. Use Case Scenario Memperbarui Perencanaan

Use case scenario memperbarui perencanaan menggambarkan langkah-langkah pengguna untuk memperbarui perencanaan *travelling*. Use case scenario memperbarui perencanaan dapat dilihat pada Tabel 3.17 Use Case Scenario Memperbarui Perencanaan.

Tabel 3. 17 Use Case Scenario Memperbarui Perencanaan

Use Case Name	**Memperbarui Perencanaan**	
Related Requirements	SKPL-F-03	
Goal In Context	Pengguna mengubah data perencanaan travelling	
Precondition	Menampilkan data perencanaan travelling	
Successful End Condition	Pengguna berhasil mengubah data perencanaan travelling	
Failed End Condition	Pengguna gagal memperbarui data perencanaan travelling	
Actors	Pengguna	
Trigger	Pengguna mengetap tombol perbarui	
Main Flow	Step	Action
	1	Pengguna mengetap tombol perbarui
	2	Sistem menampilkan *form* perbarui perencanaan travelling
	3	Pengguna mengisi *form* perbarui perencanaan travelling
	4	Pengguna mengetap tombol selesai
	5	Sistem melakukan validasi data perencanaan travelling yang baru
	6	Data perencanaan travelling yang baru berhasil disimpan
	7	Sistem menampilkan halaman perencanaan saya
Extension	Step	Branching Action
	5.1	Sistem tidak melakukan validasi data perencanaan travelling yang baru
	5.2	Data perencanaan travelling yang baru ditolak

4. Use Case Scenario Menghapus Perencanaan

Use case scenario menghapus perencanaan menggambarkan langkah-langkah pengguna untuk menghapus perencanaan *travelling*. Use case scenario menghapus perencanaan dapat dilihat pada Tabel 3.18 Use Case Scenario Mengahapus Perencanaan.

Tabel 3. 18 Use Case Scenario Menghapus Perencanaan

Use Case Name	**Menghapus Perencanaan**	
Related Requirements	SKPL-F-04	
Goal In Context	Pengguna menghapus perencanaan travelling	
Precondition	Menampilkan data perencanaan travelling	
Successful End Condition	Pengguna berhasil menghapus data perencanaan travelling	
Failed End Condition	Pengguna gagal menghapus data perencanaan travelling	
Actors	Pengguna	
Trigger	Pengguna mengetap tombol hapus	
Main Flow	Step	Action
	1	Pengguna mengetap tombol hapus
	2	Sistem menampilkan pesan konfirmasi penghapusan data perencanaan travelling
	3	Pengguna melakukan konfirmasi penghapusan data perencanaan travelling
	4	Data perencanaan travelling berhasil di hapus
	5	Sistem menampilkan halaman perencanaan saya
Extension	Step	Branching Action
	3.1	Pengguna membatalkan penghapusan data perencanaan travelling

5. Use Case Scenario Mendapatkan Rekomendasi

Use case scenario mendapatkankan rekomendasi menggambarkan langkah-langkah pengguna untuk mendapatkan rekomendasi perencanaan *travelling*. Use case scenario mendapatkan rekomendasi dapat dilihat pada Tabel 3.19 Use Case Scenario Mendapat Rekomendasi.

Tabel 3. 19 Use Case Scenario Mendapat Rekomendasi

Use Case Name	**Mendapatkan Rekomendasi**
Related Requirements	SKPL-F-05
Goal In Context	Pengguna mendapat rekomendasi perencanaan travelling
Precondition	Menampilkan data perencanaan travelling
Successful End Condition	Pengguna mendapat rekomendasi perencanaan travelling
Failed End Condition	Pengguna gagal mendapat rekomendasi perencanaan travelling

Actors	Pengguna	
Trigger	Pengguna mengetap tombol rekomendasi	
Main Flow	Step	Action
	1	Pengguna mengetap tombol rekomendasi
	2	Sistem menampilkan data rekomendasi perencanaan travelling
	3	Pengguna memilih data rekomendasi perencanaan travelling
	4	Sistem menampilkan data rekomendasi yang dipilih
	5	Pengguna mengetap tombol selesai
	6	Sistem melakukan validasi data perencanaan travelling
	7	Sistem menampilkan halaman perencanaan saya
Extension	Step	Branching Action
	2.1	Data rekomendasi perencanaan travelling tidak ditemukan

6. Use Case Scenario Menyimpan Pengeluaran

Use case scenario menyimpan pengeluaran menggambarkan langkah-langkah pengguna untuk menambah data pengeluaran selama travelling. Use case scenario menambah pengeluaran dapat dilihat pada Tabel 3.20 Use Case Scenario Menyimpan Pengeluaran.

Tabel 3. 20 Use Case Scenario Menambah Pengeluaran

Use Case Name	**Menyimpan Pengeluaran**	
Related Requirements	SKPL-F-06	
Goal In Context	Pengguna menambah pengeluaran travelling	
Precondition	Pengguna sudah membuat perencanaan travelling	
Successful End Condition	Pengguna berhasil menambah data pengeluaran	
Failed End Condition	Pengguna gagal menyimpan data pengeluaran	
Actors	Pengguna	
Trigger	Pengguna mengetap tombol tambah pengeluaran	
Main Flow	Step	Action
	1	Pengguna mengetap tombol tambah pengeluaran
	2	Sistem menampilkan *form* tambah pengeluaran
	3	Pengguna mengisi *form* tambah pengeluaran
	4	Pengguna menambahkan foto

	Include:: Mengambil Foto	
	5 **Include:: Menambah Lokasi**	Pengguna menambahkan lokasi
	6	Pengguna mengetap tombol selesai
	7	Sistem melakukan validasi data tambah pengeluaran
	8	Sistem menampilkan halaman riwayat saya
Extension	Step	Branching Action
	7.1	Sistem tidak melakukan validasi data tambah pengeluaran
	7.2	Data tambah pengeluaran ditolak

7. Use Case Scenario Memperbarui Pengeluaran

Use case scenario memperbarui pengeluaran menggambarkan langkah-langkah pengguna untuk memperbarui data pengeluaran selama travelling. Use case scenario memperbarui pengeluaran dapat dilihat pada Tabel 3.21 Use Case Scenario Memperbarui Pengeluaran

Tabel 3. 21 Use Case Scenario Memperbarui Pengeluaran

Use Case Name	**Memperbarui Pengeluaran**	
Related Requirements	SKPL-F-07	
Goal In Context	Pengguna memperbarui pengeluaran travelling	
Precondition	Menampilkan data pengeluaran	
Successful End Condition	Pengguna berhasil memperbarui data pengeluaran	
Failed End Condition	Pengguna gagal memperbarui data pengeluaran	
Actors	Pengguna	
Trigger	Pengguna mengetap tombol perbarui	
Main Flow	Step	Action
	1	Pengguna mengetap tombol perbarui
	2	Sistem menampilkan *form* perbarui pengeluaran travelling
	3	Pengguna mengisi *form* perbarui pengeluaran travelling

	4	Pengguna mengetap tombol selesai
	5	Sistem melakukan validasi data pengeluaran travelling
	6	Data pengeluaran travelling yang baru berhasil disimpan
	7	Sistem menampilkan halaman riwayat saya
Extension	Step	Branching Action
	5.1	Sistem tidak melakukan validasi data pengeluaran travelling yang baru
	5.2	Data pengeluaran travelling yang baru ditolak

8. Use Case Scenario Menghapus Pengeluaran

Use case scenario menghapus pengeluaran menggambarkan langkah-langkah pengguna untuk menghapus data pengeluaran selama travelling. Use case scenario menghapus pengeluaran dapat dilihat pada Tabel 3.22 Use Case Scenario Menghapus Pengeluaran.

Tabel 3. 22 Use Case Scenario Menghapus Pengeluaran

Use Case Name	**Menghapus Pengeluaran**	
Related Requirements	SKPL-F-08	
Goal In Context	Pengguna menghapus pengeluaran travelling	
Precondition	Menampilkan data pengeluaran travelling	
Successful End Condition	Pengguna berhasil menghapus data pengeluaran travelling	
Failed End Condition	Pengguna gagal menghapus data pengeluaran travelling	
Actors	Pengguna	
Trigger	Pengguna mengetap tombol hapus	
Main Flow	Step	Action
	1	Pengguna menekan tombol hapus
	2	Sistem menampilkan pesan konfirmasi penghapusan data pengeluaran travelling
	3	Pengguna melakukan konfirmasi penghapusan data pengeluaran travelling
	4	Data pengeluaran travelling berhasil di hapus
	5	Sistem menampilkan halaman riwayat saya
Extension	Step	Branching Action
	3.1	Pengguna membatalkan penghapusan data pengeluaran travelling

9. Use Case Scenario Menambah Lokasi

Use case scenario menambah lokasi menggambarkan langkah-langkah pengguna untuk menambah data lokasi travelling. Use case scenario menambah lokasi dapat dilihat pada Tabel 3.23 Use Case Scenario Menambah Lokasi.

Tabel 3. 23 Use Case Scenario Menambah Lokasi

Use Case Name	**Menambah Lokasi**	
Related Requirements	SKPL-F-09	
Goal In Context	Pengguna menambah lokasi travelling	
Precondition	Pengguna akan menambah pengeluaran travelling	
Successful End Condition	Pengguna berhasil menambah data lokasi	
Failed End Condition	Pengguna gagal menambah data lokasi	
Actors	Pengguna	
Trigger	Pengguna mengetap tombol tambah lokasi	
Main Flow	Step	Action
	1 **Include:: Mengambil Data Lokasi**	Pengguna mengetap tombol tambah lokasi
	2	Sistem menampilkan halaman peta lokasi
	3	Pengguna menambahkan lokasi
	4	Pengguna mengetap tombol selesai
	5	Sistem melakukan validasi data tambah lokasi
	6	Sistem menampilkan halaman tambah pengeluaran
Extension	Step	Branching Action
	5.1	Sistem tidak melakukan validasi data tambah lokasi
	5.2	Data tambah lokasi ditolak

10. Use Case Scenario Mengambil Foto

Use case scenario mengambil foto menggambarkan langkah-langkah pengguna untuk mengambil foto. Use case scenario mengambil foto dapat dilihat pada Tabel 3.24 Use Case Scenario Mengambil Foto.

Tabel 3. 24 Use Case Scenario Mengambil Foto

Use Case Name	**Mengambil Foto**	
Related Requirements	SKPL-F-10	
Goal In Context	Pengguna mengambil foto travelling	
Precondition	Pengguna akan menambah pengeluaran travelling	
Successful End Condition	Pengguna berhasil mengambil foto travelling	
Failed End Condition	Pengguna gagal menambah data lokasi	
Actors	Pengguna	
Trigger	Pengguna mengetap tombol ambil foto	
Main Flow	Step	Action
	1	Pengguna mengetap tombol ambil foto
	2	Sistem menampilkan halaman pengambilan foto
	3	Pengguna mengambil foto
	4	Pengguna mengetap tombol simpan
	5	Sistem melakukan validasi data foto
	6	Sistem menampilkan halaman tambah pengeluaran
Extension	Step	Branching Action
	4.1	Pengguna menekan tombol batal
	5.1	Sistem tidak melakukan validasi data foto

11. Use Case Scenario Mengambil Data Lokasi Melalui Google Place API

Use case scenario mengambil data lokasi melalui API menggambarkan langkah-langkah sistem untuk mengambil data lokasi melalui API. Use case scenario mencari riwayat dapat dilihat pada Tabel 3.25 Use Case Scenario Mengambil Data Lokasi Melalui Google Place API.

Tabel 3. 25 Use Case Scenario Mengambil Data Lokasi Melalui Google Place API

Use Case Name	**Mengambil Data Lokasi Melalui Google Place API**
Related Requirements	SKPL-F-11
Goal In Context	Sistem berhasil mengambil data lokasi
Precondition	Pengguna akan menambah pengeluaran travelling
Successful End Condition	Sistem berhasil menampilkan halaman peta lokasi
Failed End Condition	Sistem gagal menampilkan halaman peta lokasi
Actors	Pengguna
Trigger	Pengguna mengetap tombol tambah lokasi

Main Flow	Step	Action
	1	Sistem mengirimkan permintaan data lokasi ke google place API
	2	Google place API melakukan validasi permintaan
	3	Goole place API mengirimkan data lokasi ke sistem
	4	Sistem menampilkan halaman peta lokasi
Extension	Step	Branching Action
	2.1	Google place API tidak melakukan validasi permintaan
	4.1	Sistem gagal menampilkan halaman peta lokasi

12. Use Case Scenario Mencari Riwayat

Use case scenario mencari riwayat menggambarkan langkah-langkah pengguna untuk mencari riwayat travelling pengguna lain. Use case scenario mencari riwayat dapat dilihat pada Tabel 3.26 Use Case Scenario Mencari Riwayat.

Tabel 3. 26 Case Scenario Mencari Riwayat

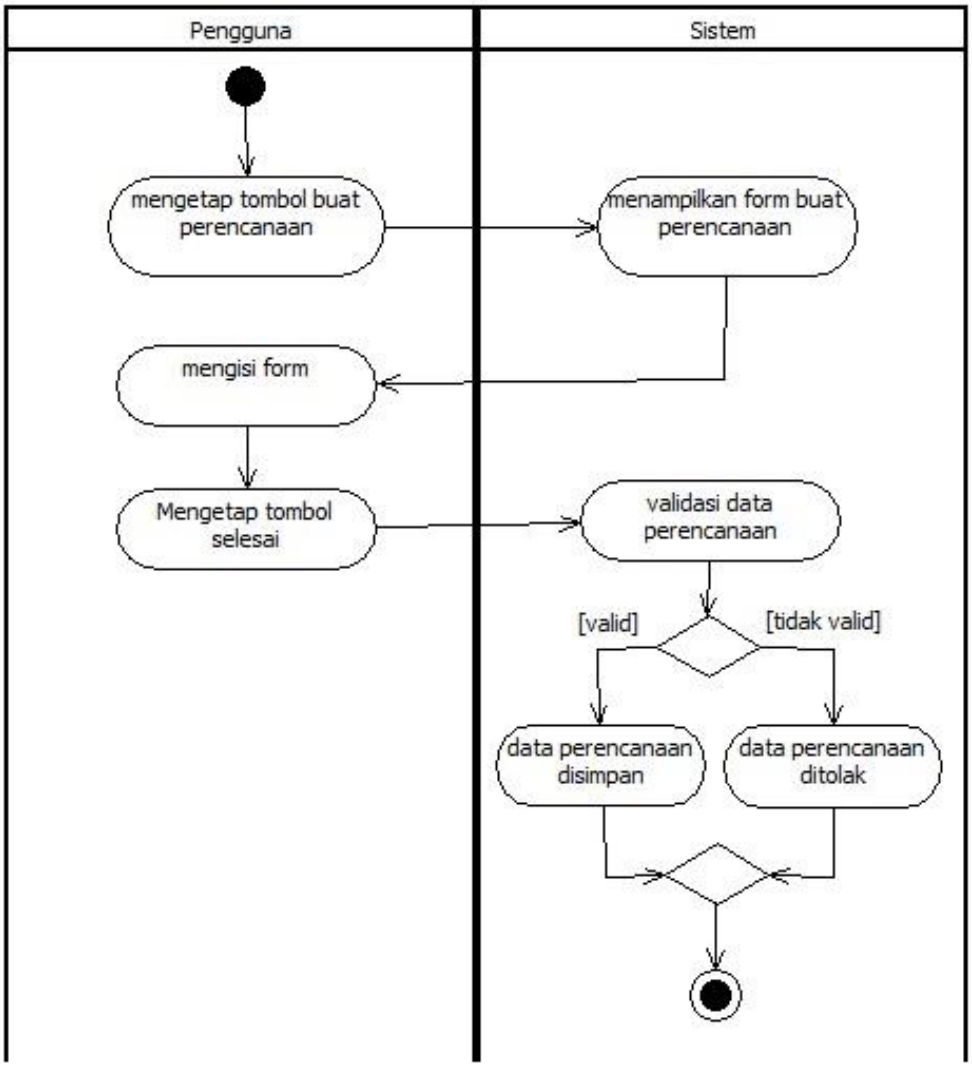
Use Case Name	**Mencari Riwayat**	
Related Requirements	SKPL-F-12	
Goal In Context	Melakukan pencarian riwayat pada database.	
Precondition	Pengguna berada di halaman riwayat	
Successful End Condition	Data riwayat ditemukan dan ditampilkan.	
Failed End Condition	Data riwayat tidak ditemukan.	
Actors	Pengguna	
Trigger	User mengetap tombol cari	
Main Flow	Step	Action
	1	Pengguna mengetap tombol cari
	2	Sistem menampilkan *field* pencarian data riwayat.
	3	Pengguna memasukkan data riwayat.
	4	Pengguna mengetap tombol cari riwayat.
	5	Sistem melakukan pengecekan pada *database.*
	6	Sistem menampilkan data riwayat.
Extension	Step	Branching Action
	5.1	Sistem tidak melakukan pengecekan data riwayat.
	5.2	Data riwayat tidak ditemukan dalam *database.*

3.1.5.1.3 Activity Diagram

Activity Diagram menggambarkan proses bisnis dan urutan aktifitas dalam sebuah proses, dan digunakan pada business modeling untuk memperlihatkan urutan aktifitas proses bisnis. *Activity Diagram* sangat bermanfaat untuk memahami proses dari sistem secara keseluruhan. *Activity Diagram* dibuat berdasarkan sebuah atau beberapa *use case* pada *use case diagram.* Berikut ini penjelasan dari masing-masing *activity diagram.*

1. Activity Diagram Membuat Perencanaan

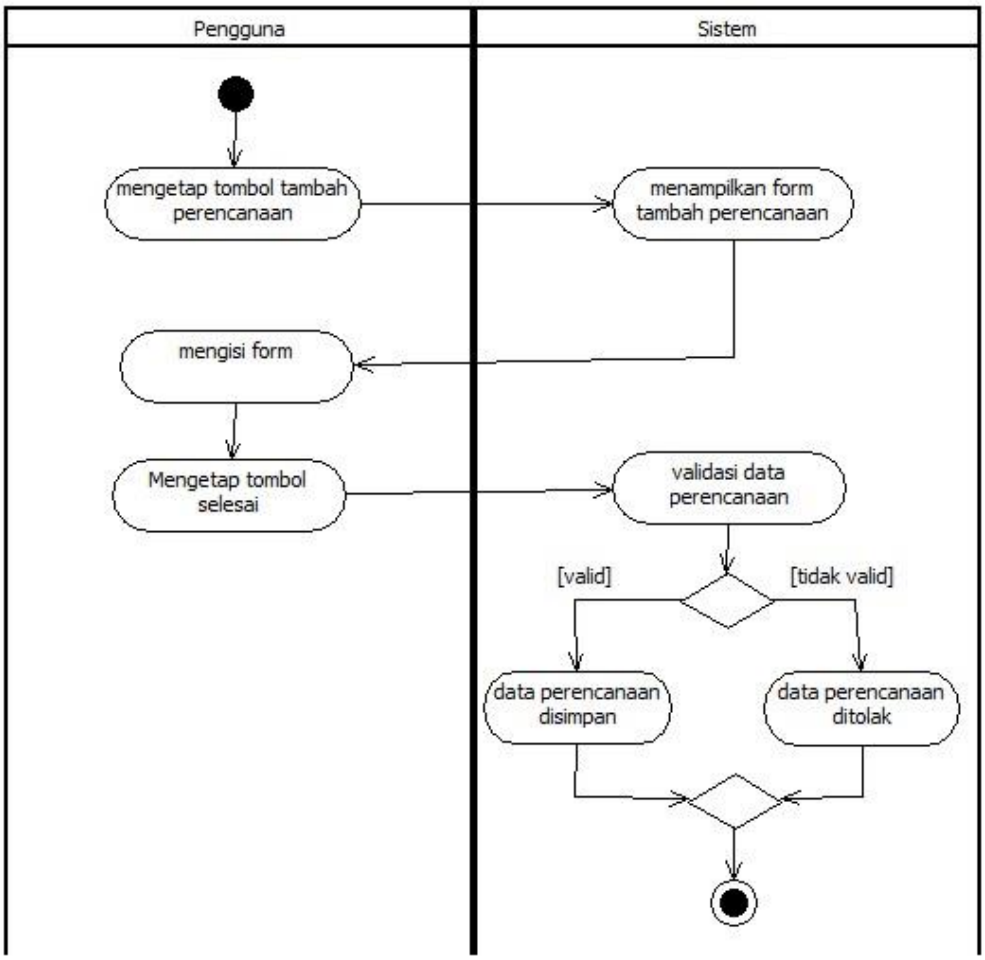
Proses membuat perencanaan pada aplikasi get trip dapat dilihat pada gambar 3.3 Activity diagram membuat perencanaan.



Gambar 3. 3 Activity Diagram Membuat Perencanaan

2. Activity Diagram Menambah Perencanaan

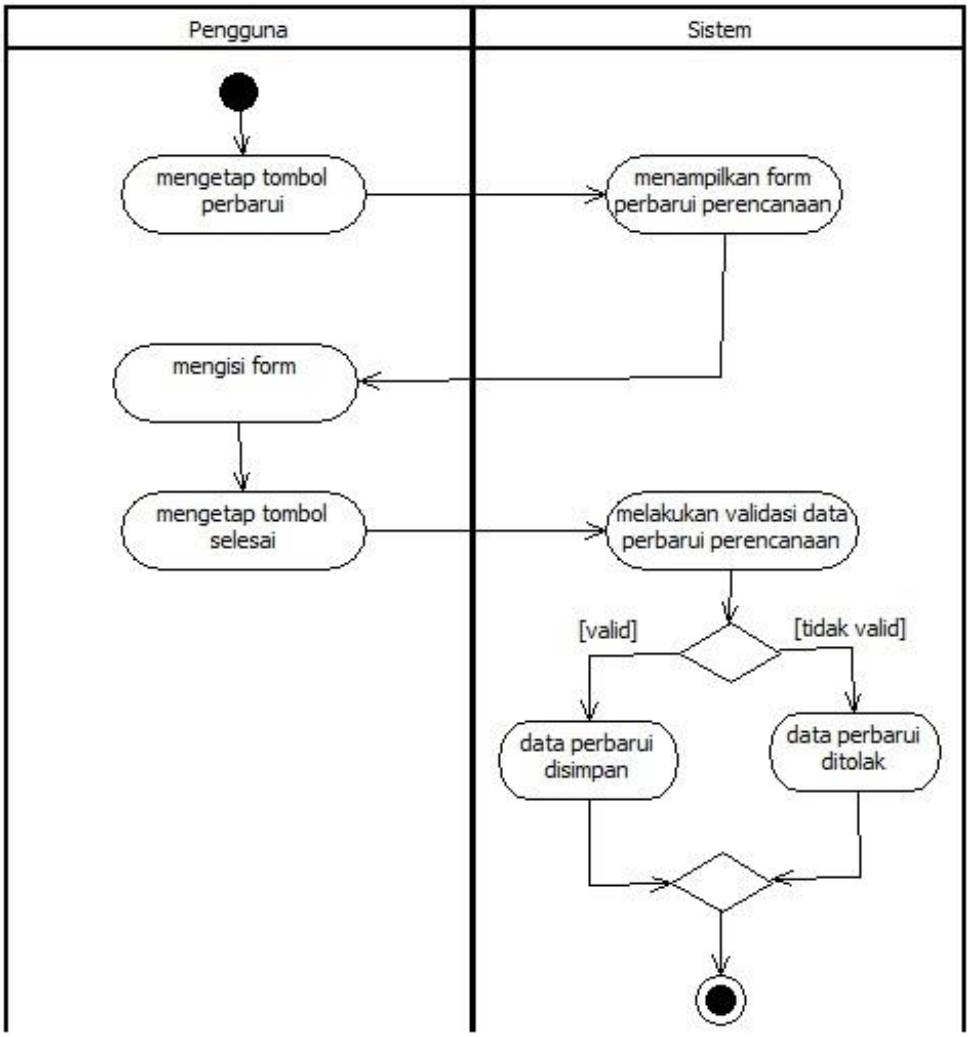
Proses menambah perencanaan pada aplikasi get trip dapat dilihat pada pada gambar 3.4 Activity diagram menambah perencanaan.



Gambar 3. 4 Activity Diagram Menambah Perencanaan

3. Activity Diagram Memperbarui Perencanaan

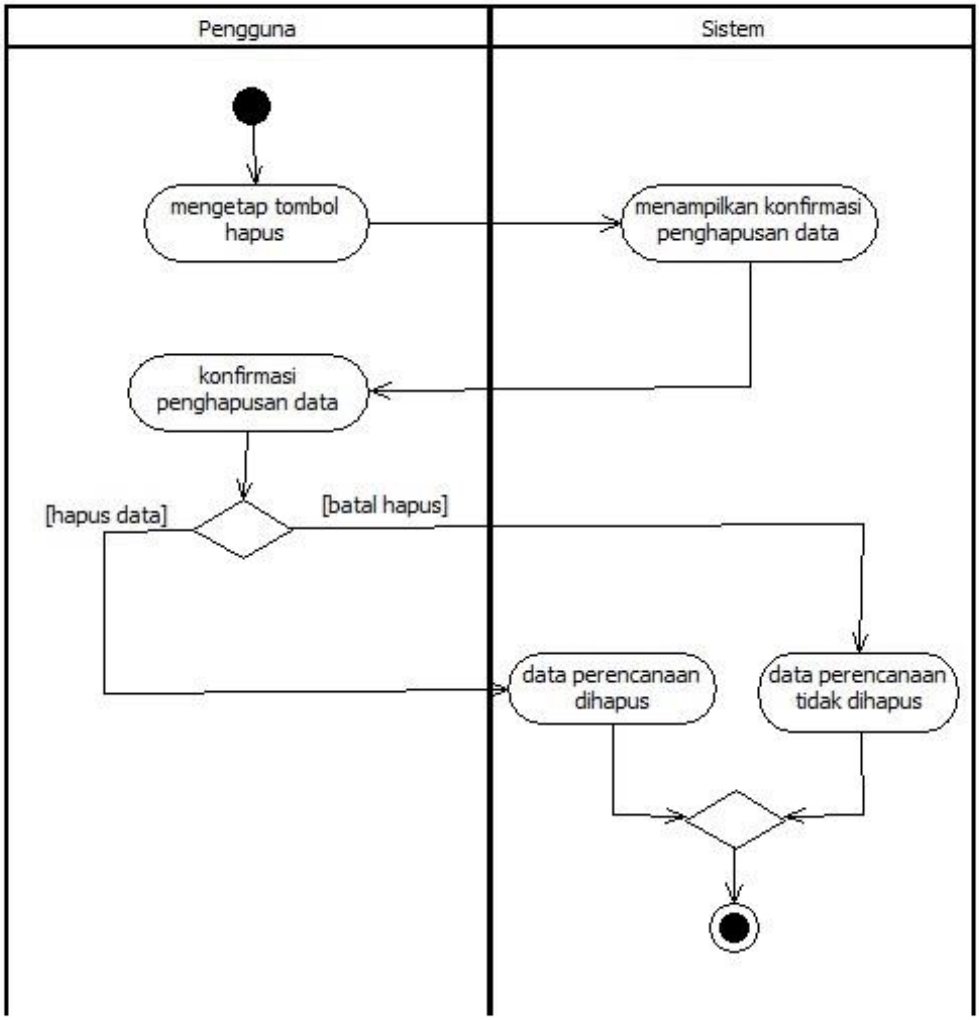
Proses memperbarui perencanaan pada aplikasi get trip dapat dilihat pada pada gambar 3.5 Activity diagram memperbarui perencanaan.



Gambar 3. 5 Activity Diagram Memperbarui Perencanaan

4. Activity Diagram Menghapus Perencanaan

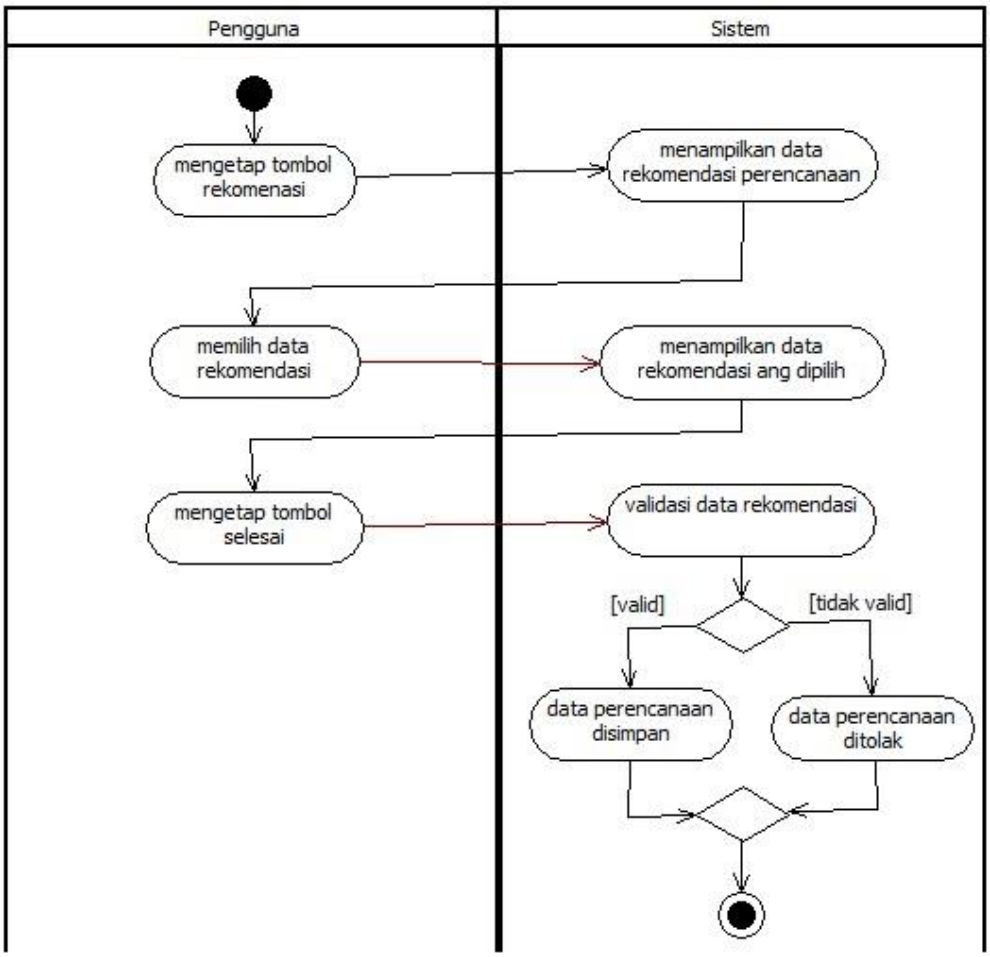
Proses menghapus perencanaan pada aplikasi get trip dapat dilihat pada pada gambar 3.6 Activity diagram menghapus perencanaan.



Gambar 3. 6 Activity Diagram Menghapus Perencanaan

5. Activity Diagram Mendapatkan Rekomendasi

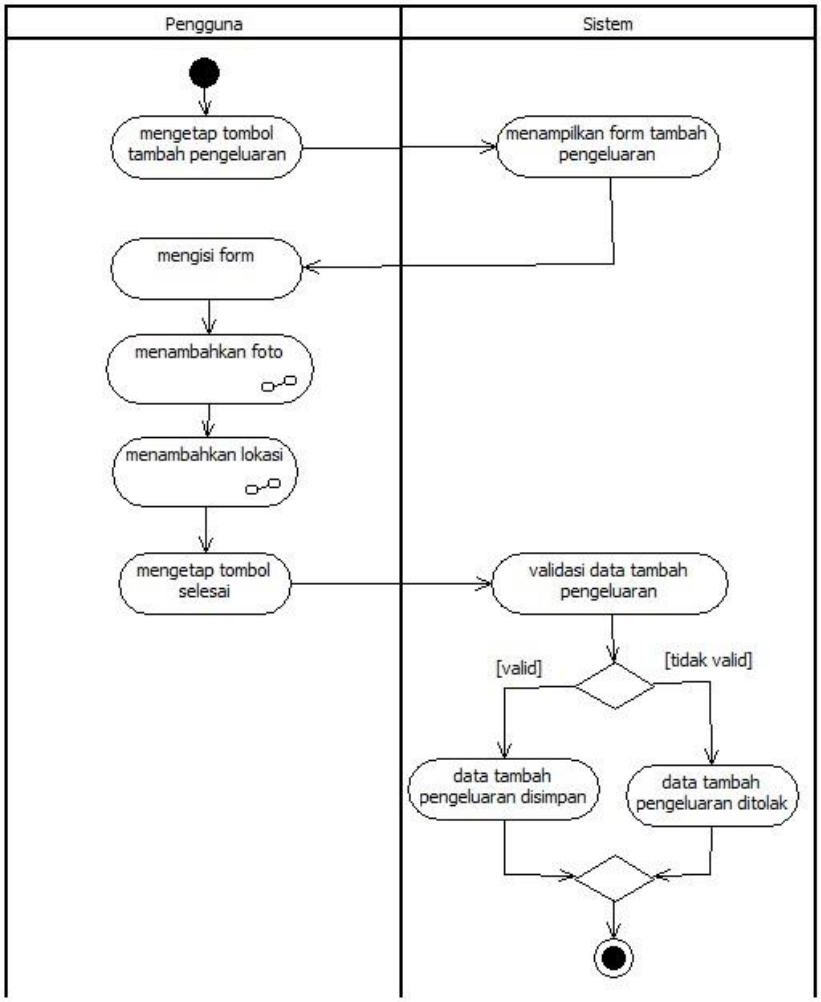
Proses mendapatkan rekomendasi pada aplikasi get trip dapat dilihat pada pada gambar 3.7 Activity diagram mendapatkan rekomendasi.



Gambar 3. 7 Activity Diagram Menghapus Perencanaan

6. Activity Diagram Menyimpan Pengeluaran

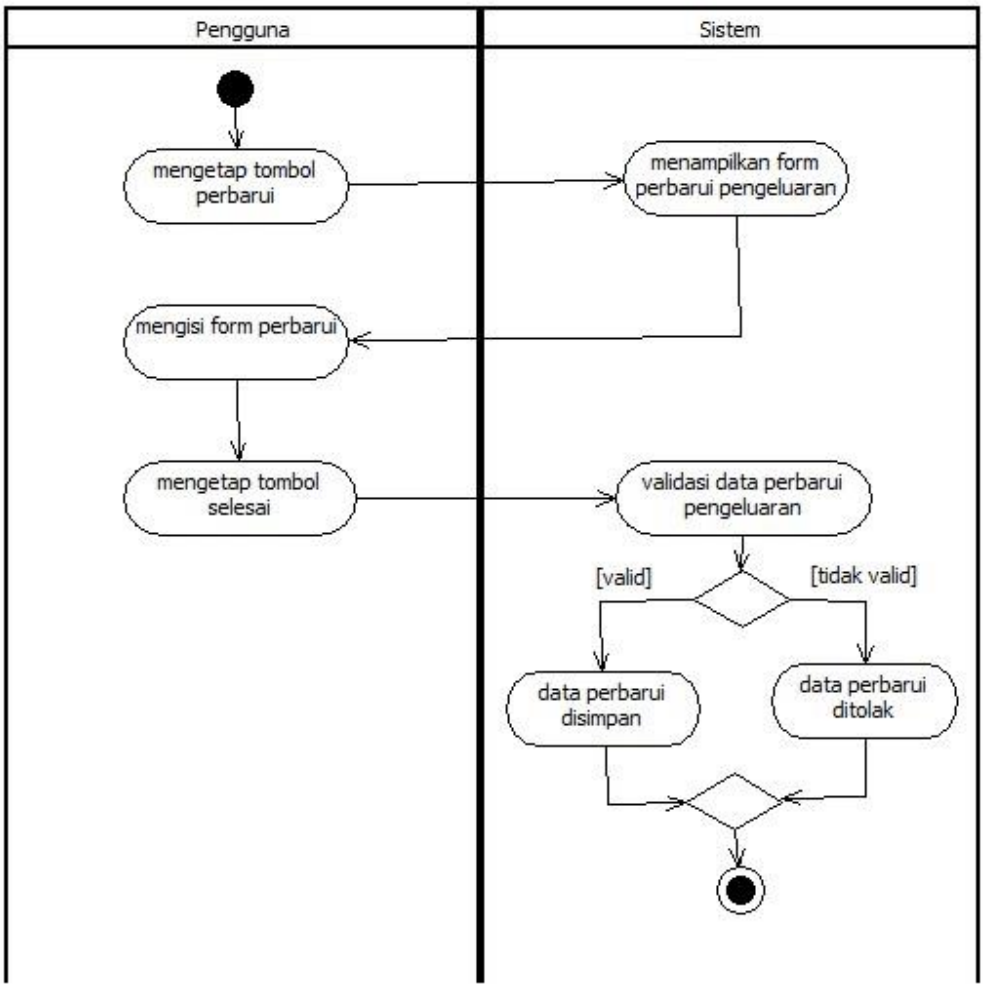
Proses menambah pengeluaran pada aplikasi get trip dapat dilihat pada gambar 3.8 Activity diagram menyimpan pengeluaran.



Gambar 3. 8 Activity Diagram Menyimpan Pengeluaran

7. Activity Diagram Memperbarui Pengeluaran

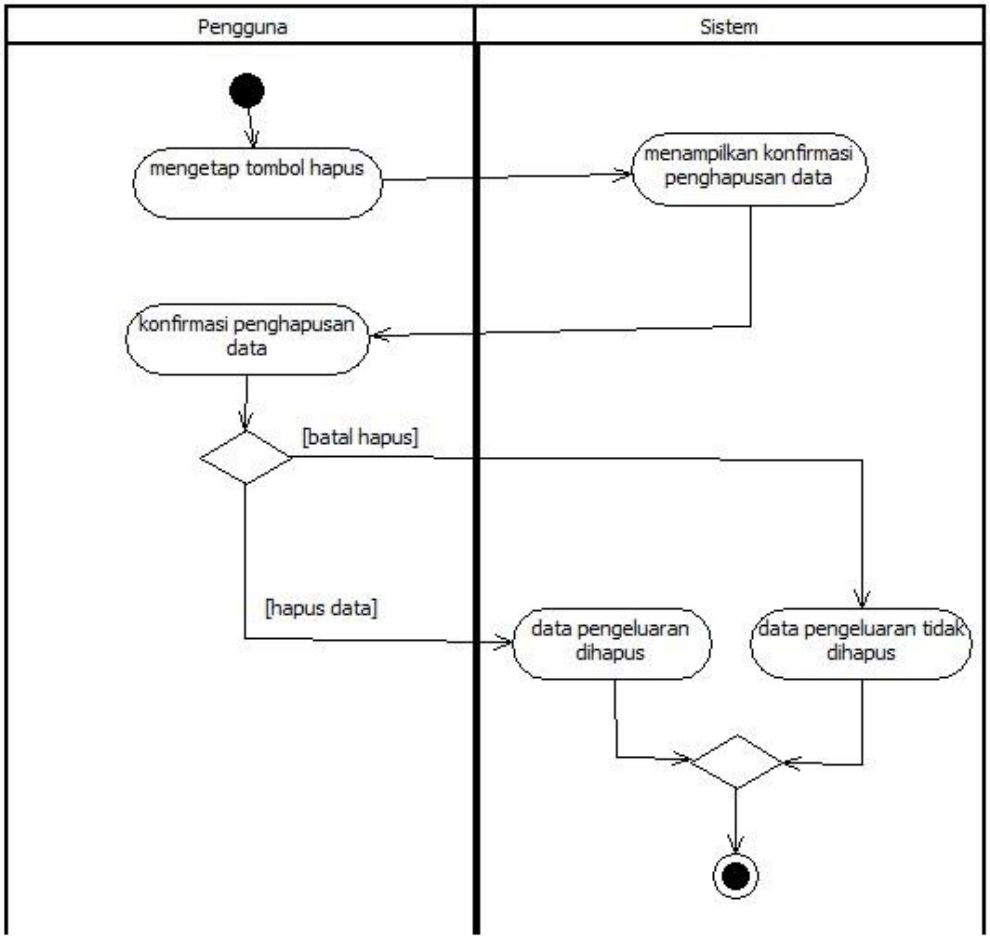
Proses memperbarui pengeluaran pada aplikasi get trip dapat dilihat pada pada gambar 3.9 Activity diagram memperbarui pengeluaran.



Gambar 3. 9 Activity Diagram Memperbarui Pengeluaran

8. Activity Diagram Menghapus Pengeluaran

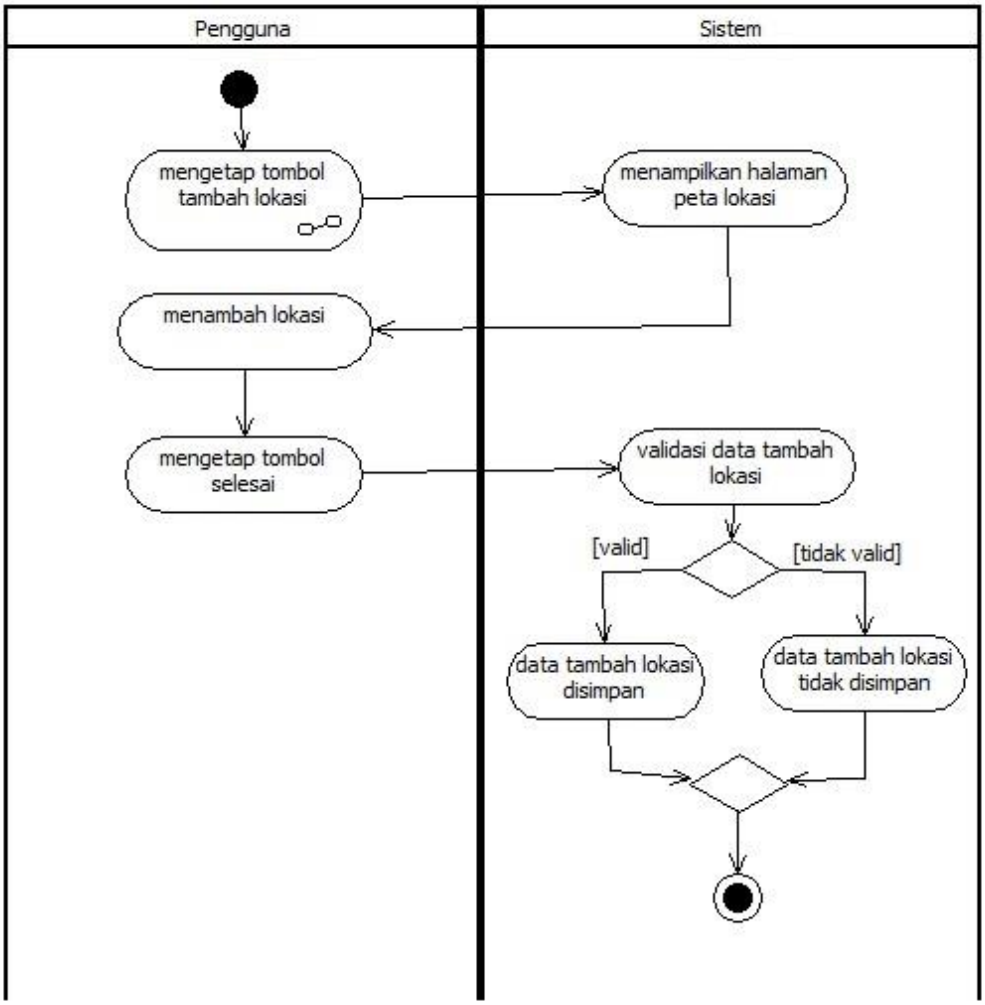
Proses menghapus pengeluaran pada aplikasi get trip dapat dilihat pada pada gambar 3.10 Activity diagram menghapus pengeluaran.



Gambar 3. 10 Activity Diagram Menghapus Pengeluaran

9. Activity Diagram Menambah Lokasi

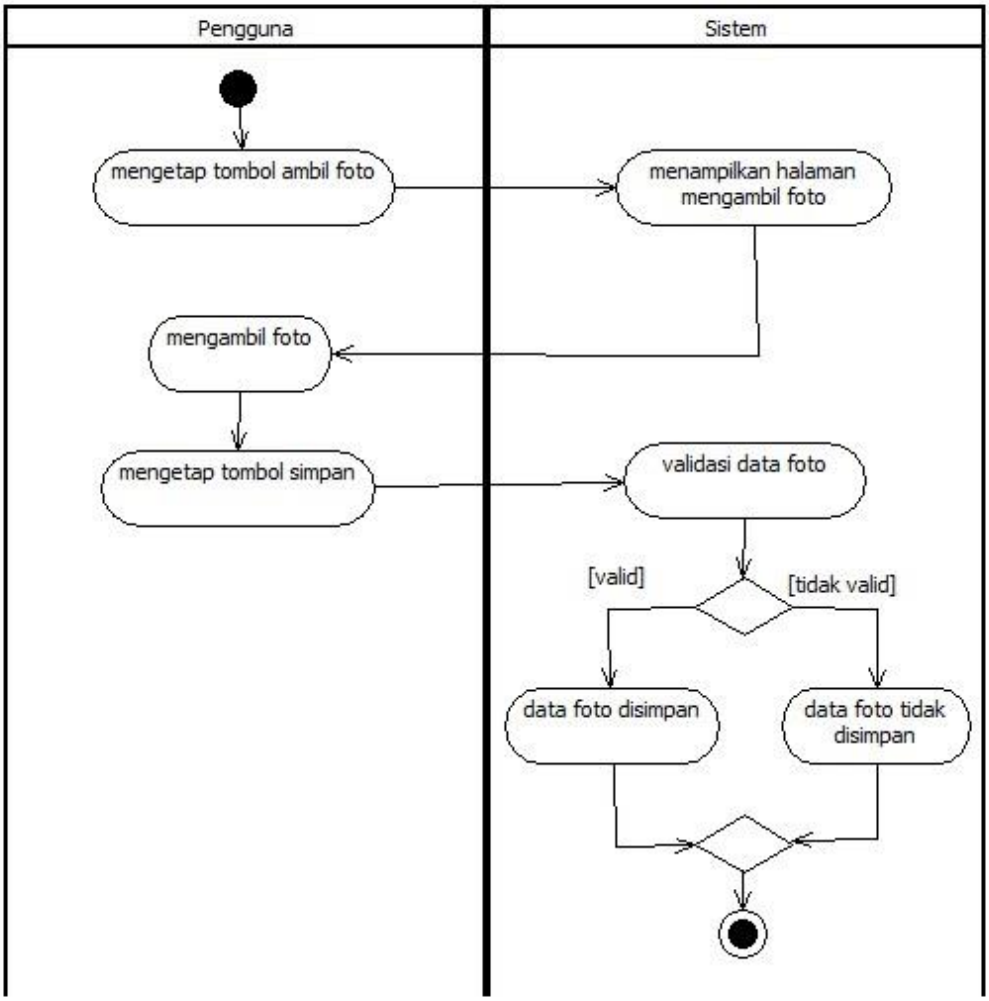
Proses menambah lokasi pada aplikasi get trip dapat dilihat pada pada gambar 3.11 Activity diagram menambah lokasi.



Gambar 3. 11 Activity Diagram Menambah Lokasi

10. Activity Diagram Mengambil Foto

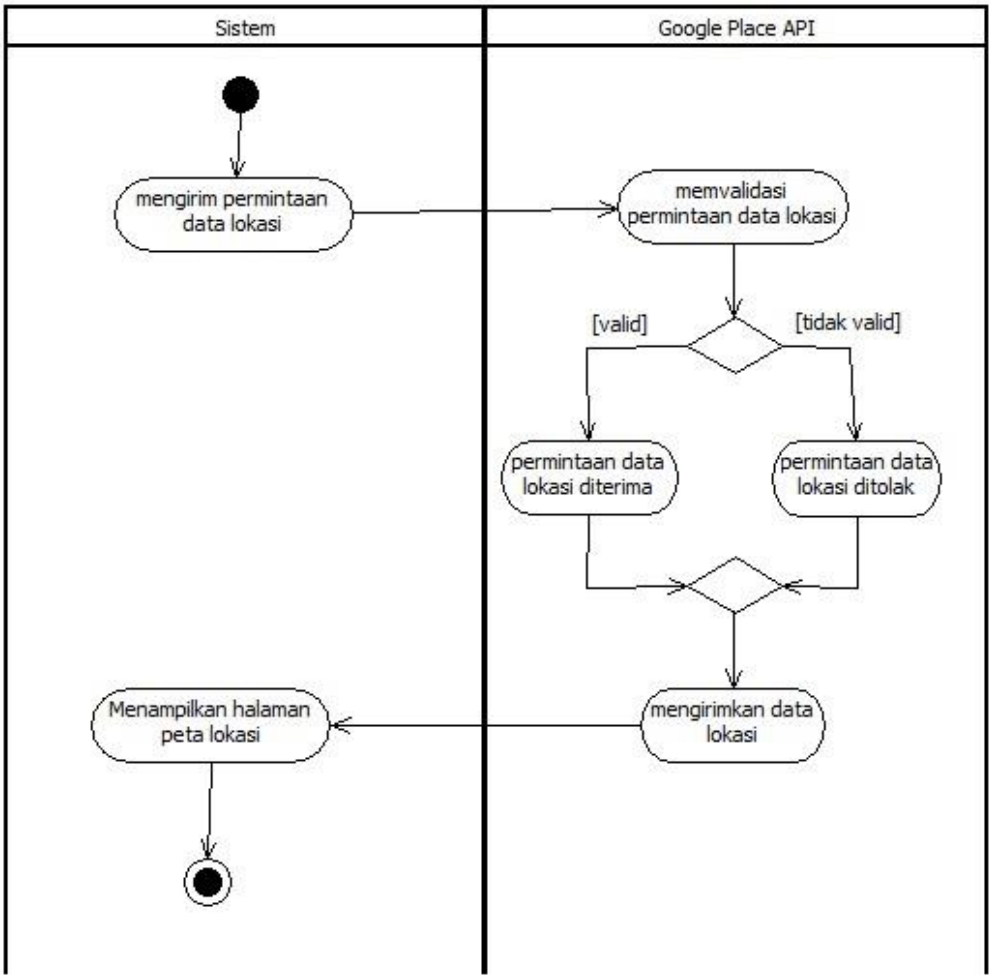
Proses mengambil foto pada aplikasi get trip dapat dilihat pada pada gambar 3.12 Activity diagram mengambil foto.



Gambar 3. 12 Activity Diagram Mengambil Foto

11. Activity Diagram Mengambil Data Lokasi Melalui Google Place API

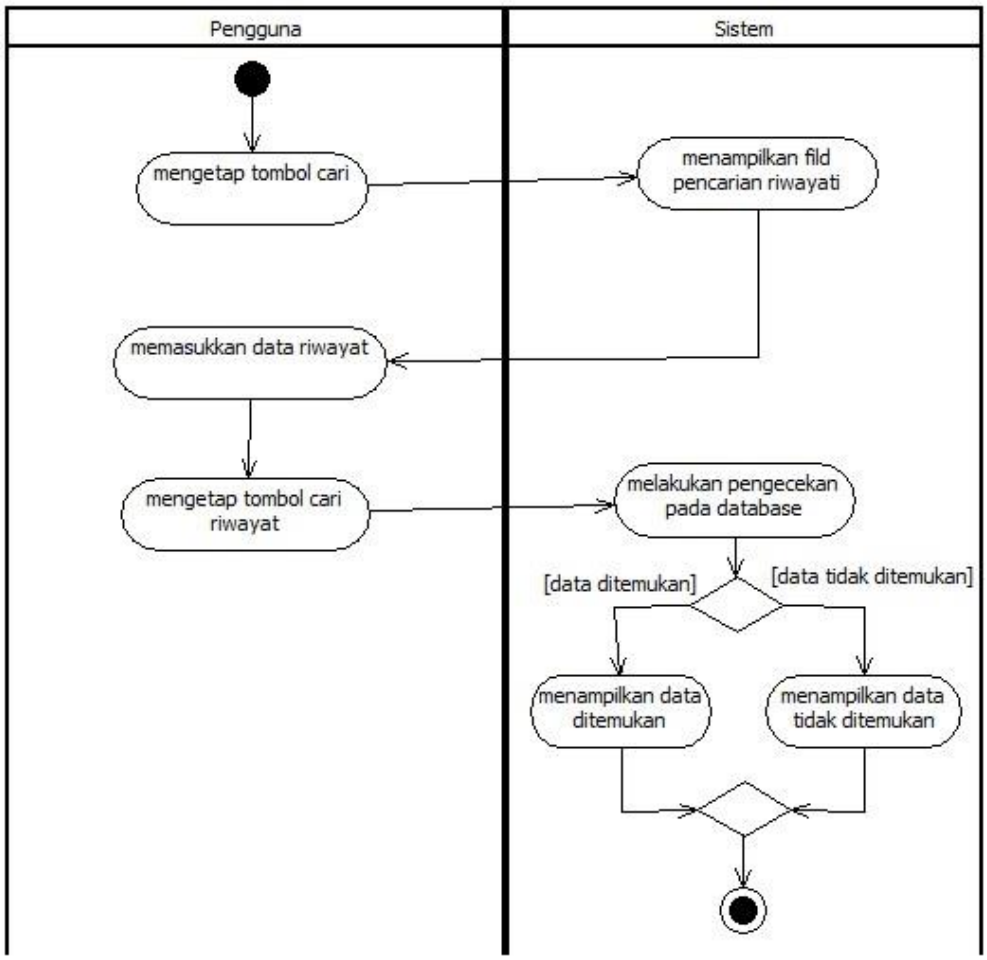
Proses mengambil data lokasi melalui google place API pada aplikasi get trip dapat dilihat pada pada gambar 3.13 Activity diagram mengambil data lokasi melalui google place API.



Gambar 3. 13 Activity Diagram Mengambil Data Lokasi Melalui Google Place API

12. Activity Diagram Mencari Riwayat

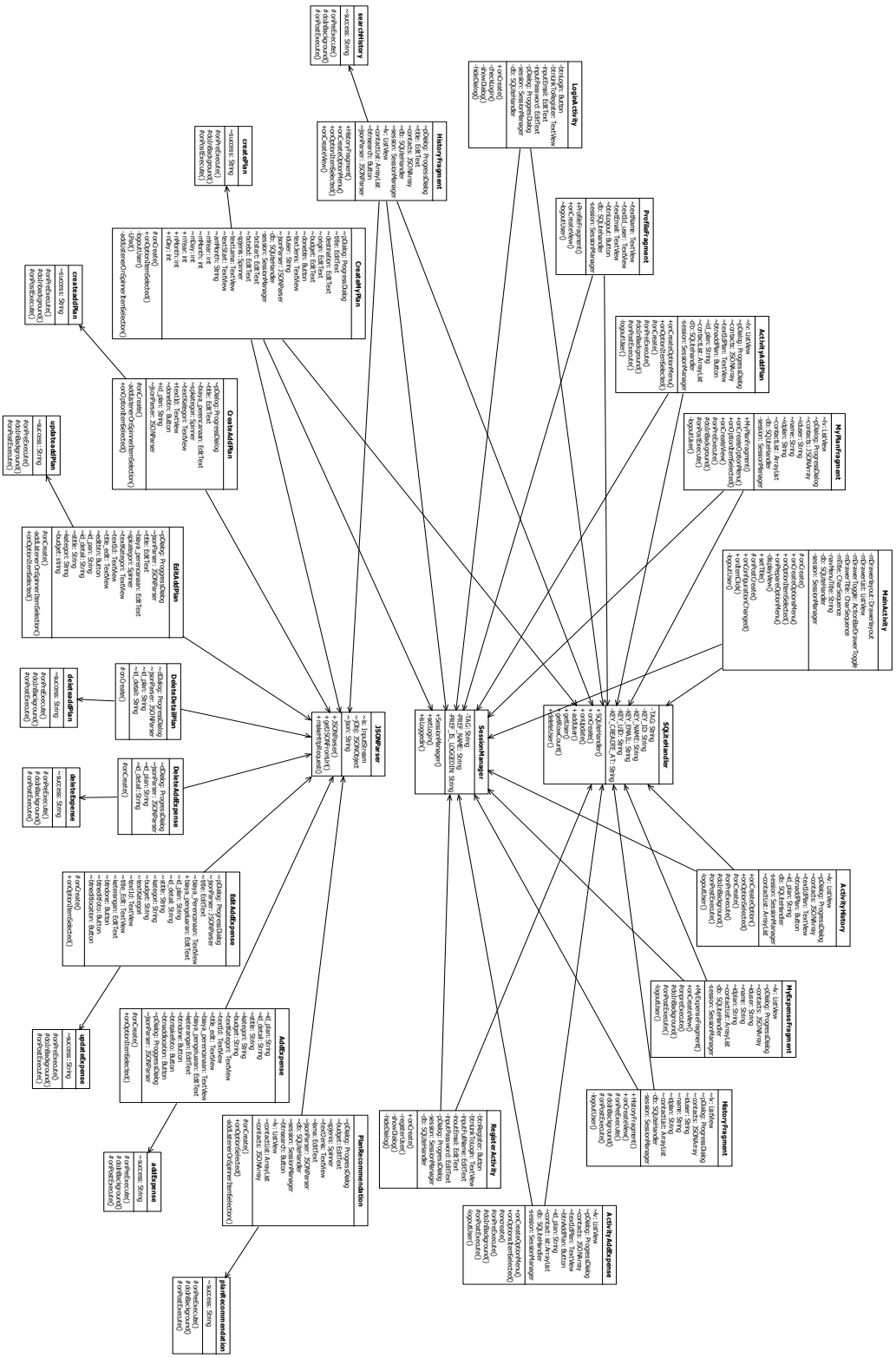
Proses mencari riwayat pada aplikasi get trip dapat dilihat pada pada gambar 3.14 Activity diagram mencari riwayat.



Gambar 3. 14 Activity Diagram Mencari Riwayat

3.1.5.1.4 Class Diagram

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun perangkat lunak get trip. Gambar 3.15 Class Diagram Platform Mobile Perangkat Lunak Get Trip, menunjukan struktur class satu yang berelasi dengan class lainnya.



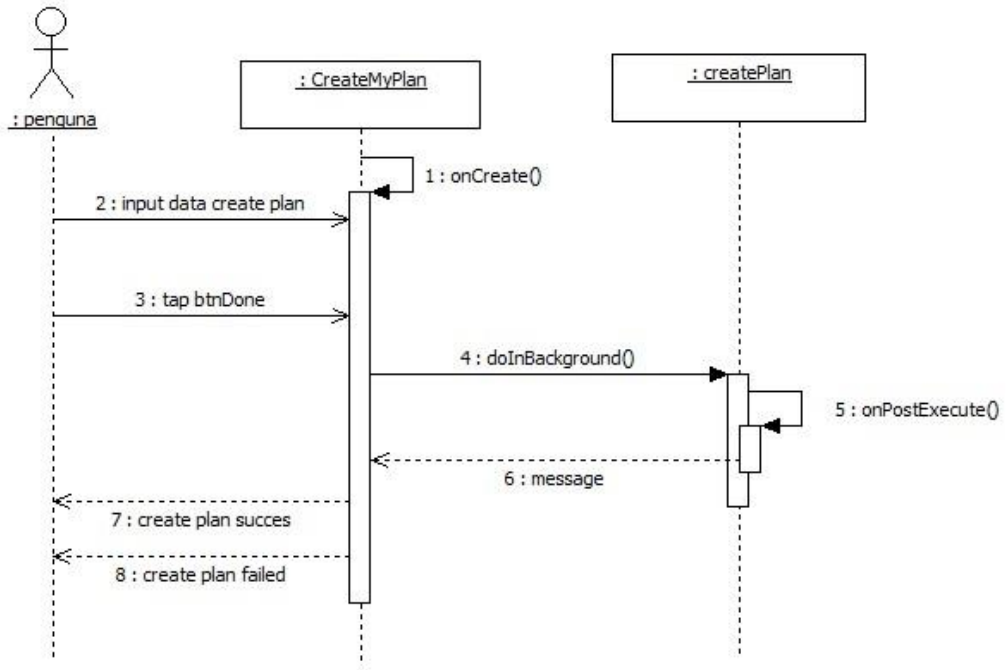
Gambar 3. 15 Class Diagram Platform Mobile Perangkat Lunak Get Trip

3.1.5.1.5 Sequence Diagram

Sequence diagram merupakan gambaran interaksi masing-masing objek pada setiap *use case* dalam urutan waktu. Interaksi ini berupa pengiriman serangkaian data antar objek-objek yang saling berinteraksi. Berikut ini adalah sequence diagram yang dapat digambarkan, diantaranya adalah :

1. Sequence Diagram Membuat Perencanaan

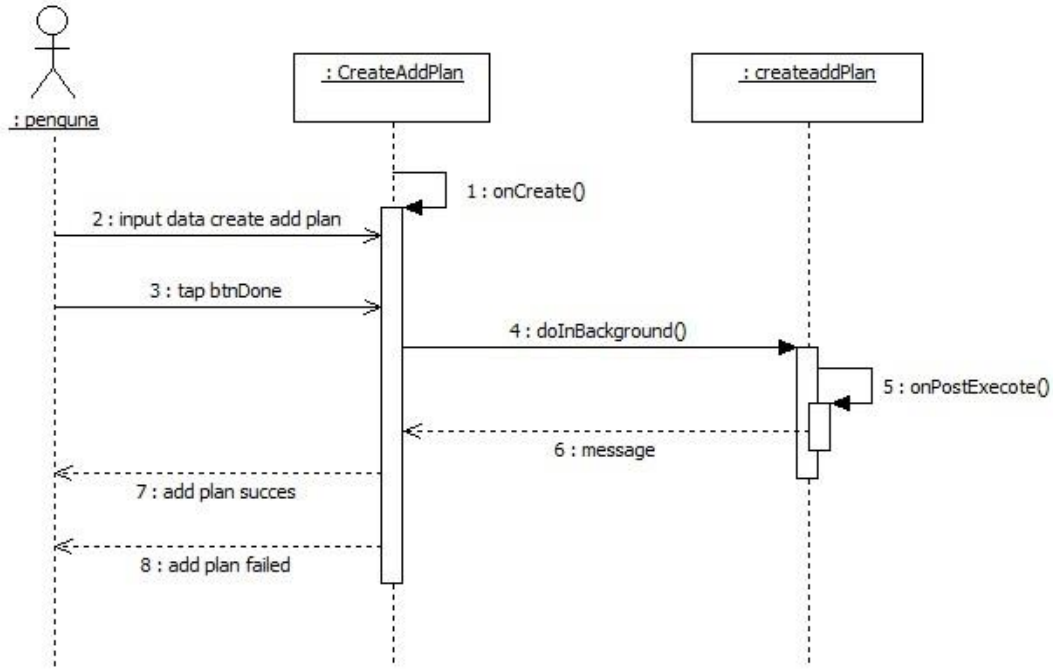
Berikut ini adalah interaksi antara aktor dengan objek-objek yang ada pada proses membuat perencanaan travelling dapat dilihat pada gambar 3.16 Sequence diagram membuat perencanaan.



Gambar 3. 16 Sequence Diagram Membuat Perencanaan

2. Sequence Diagram Menambah Perencanaan

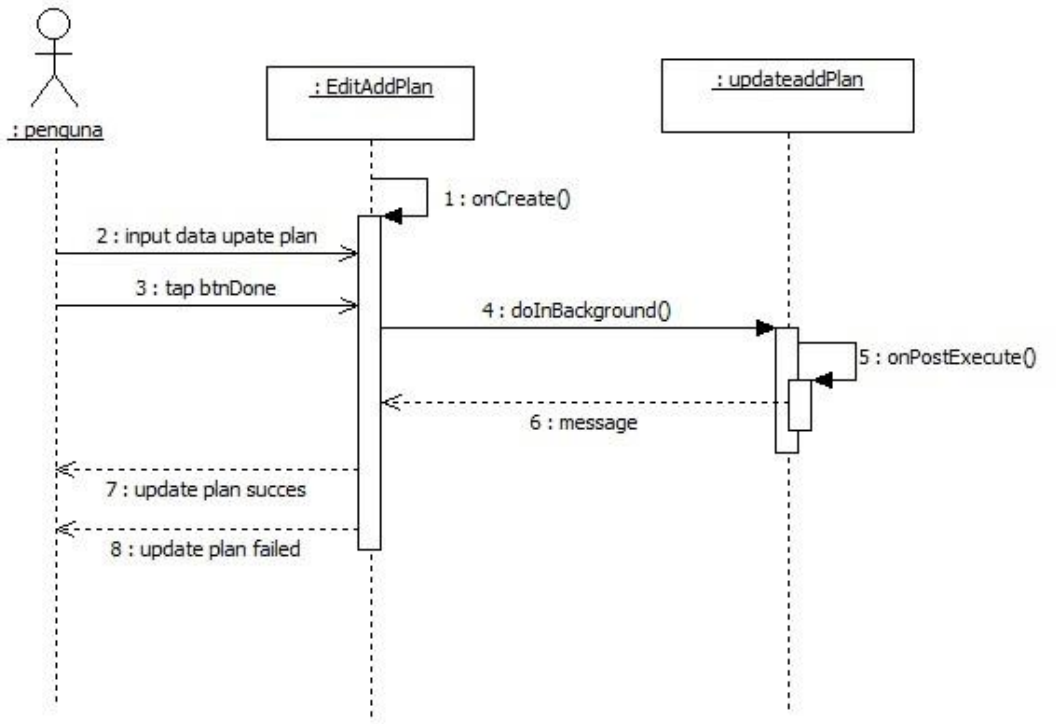
Berikut ini adalah interaksi antara aktor dengan objek-objek yang ada pada proses menambah perencanaan travelling dapat dilihat pada gambar 3.17 Sequence diagram menambah perencanaan.



Gambar 3. 17 Sequence Diagram Menambah Perencanaan

3. Sequence Diagram Memperbarui Perencanaan

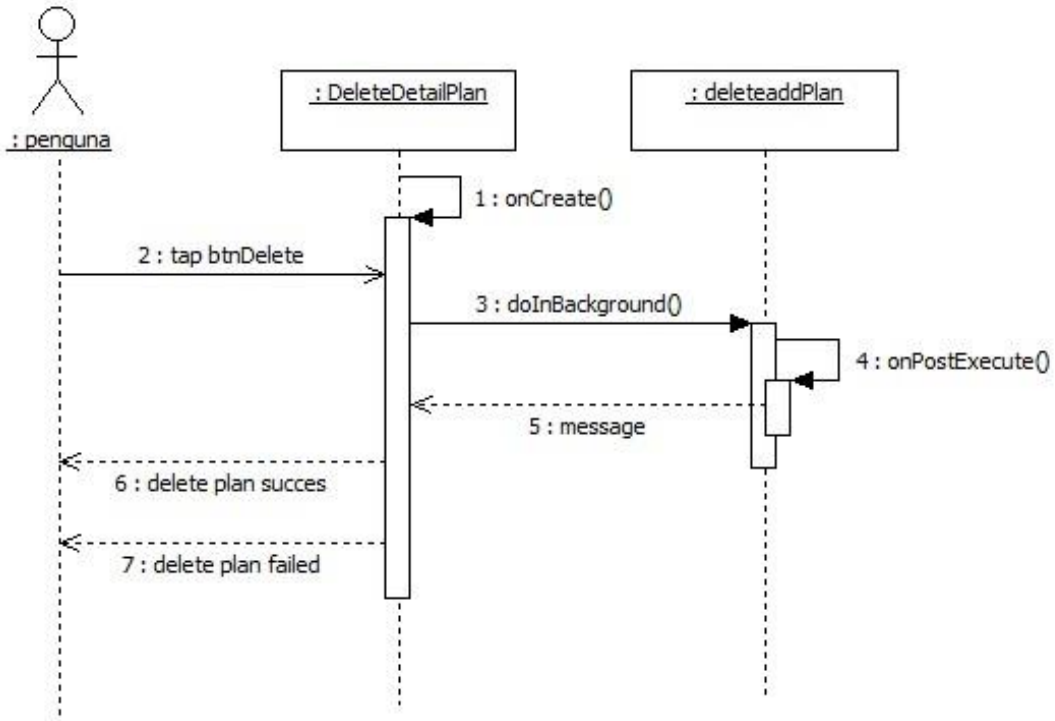
Berikut ini adalah interaksi antara aktor dengan objek-objek yang ada pada proses memperbarui perencanaan travelling dapat dilihat pada gambar 3.18 Sequence diagram memperbarui perencanaan.



Gambar 3. 18 Sequence Diagram Memperbarui Perencanaan

4. Sequence Diagram Menghapus Perencanaan

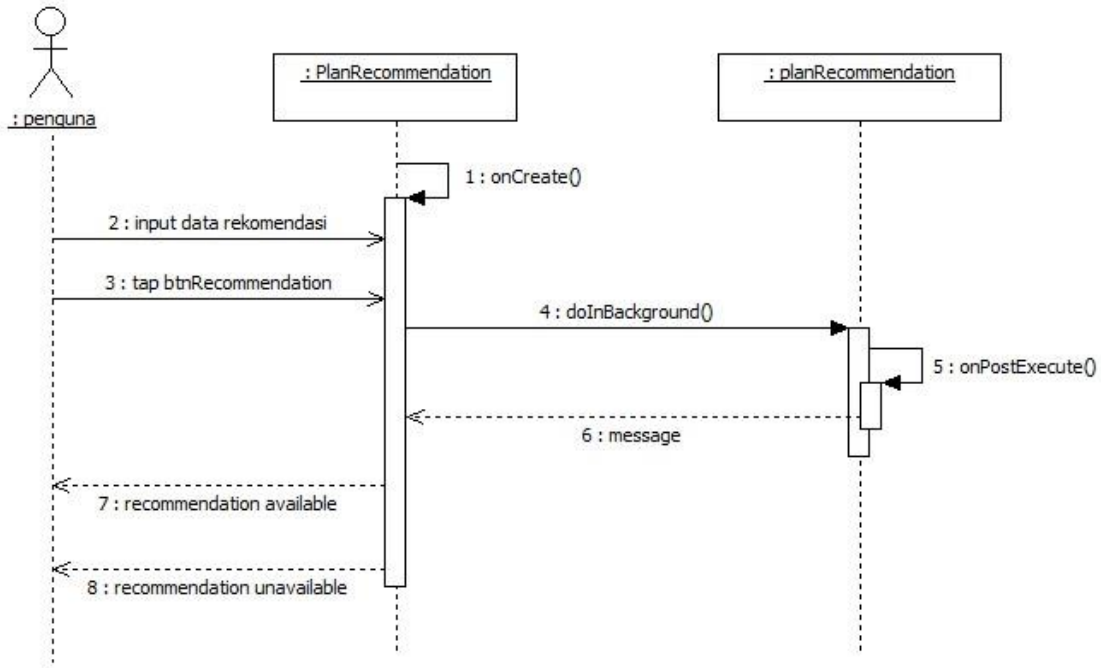
Berikut ini adalah interaksi antara aktor dengan objek-objek yang ada pada proses menghapus perencanaan travelling dapat dilihat pada gambar 3.19 Sequence diagram menghapus perencanaan.



Gambar 3. 19 Sequence Diagram Menghapus Perencanaan

5. Sequence Diagram Mendapatkan Rekomendasi

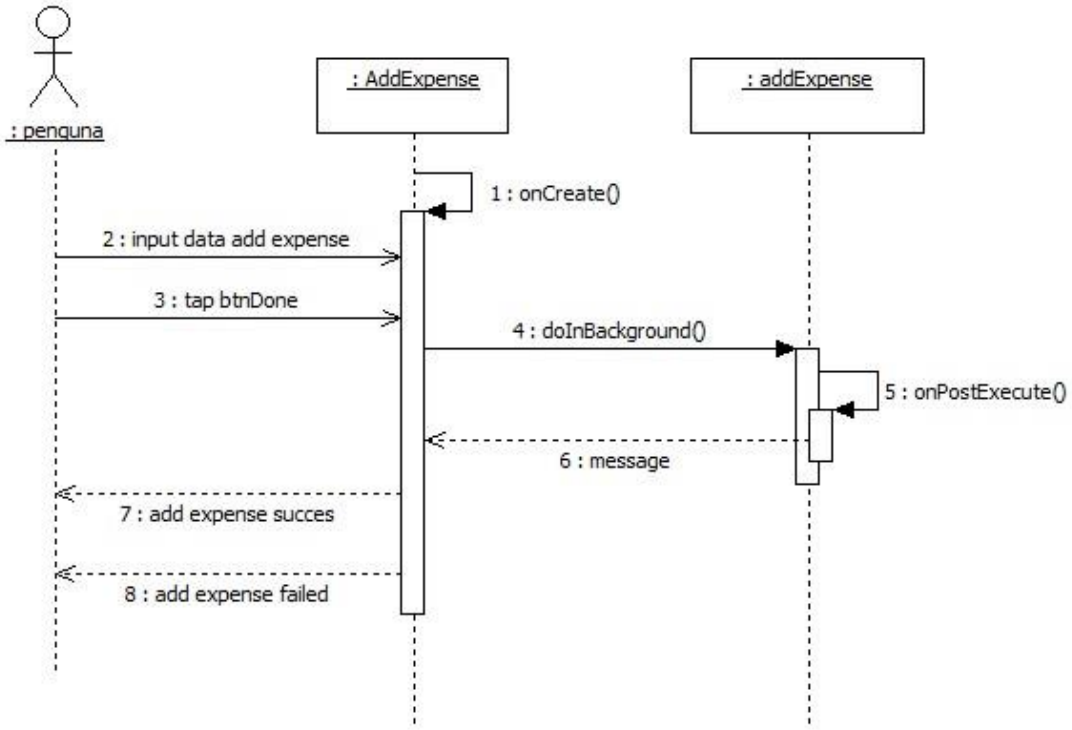
Berikut ini adalah interaksi antara aktor dengan objek-objek yang ada pada proses mendapatkan rekomendasi travelling dapat dilihat pada gambar 3.20 Sequence diagram mendapatkan rekomendasi.



Gambar 3. 20 Sequence Diagram Mendapatkan Rekomendasi

6. Sequence Diagram Menyimpan Pengeluaran

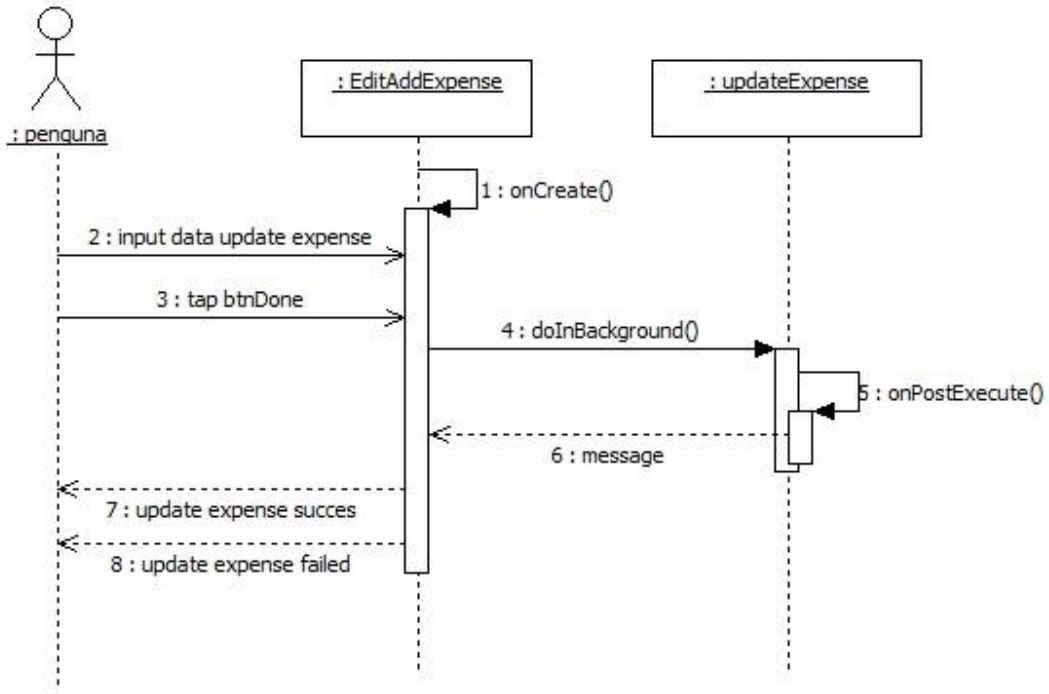
Berikut ini adalah interaksi antara aktor dengan objek-objek yang ada pada proses menyimpan pengeluaran travelling dapat dilihat pada gambar 3.21 Sequence diagram menyimpan pengeluaran.



Gambar 3. 21 Sequence Diagram Menyimpan Pengeluaran

7. Sequence Diagram Memperbarui Pengeluaran

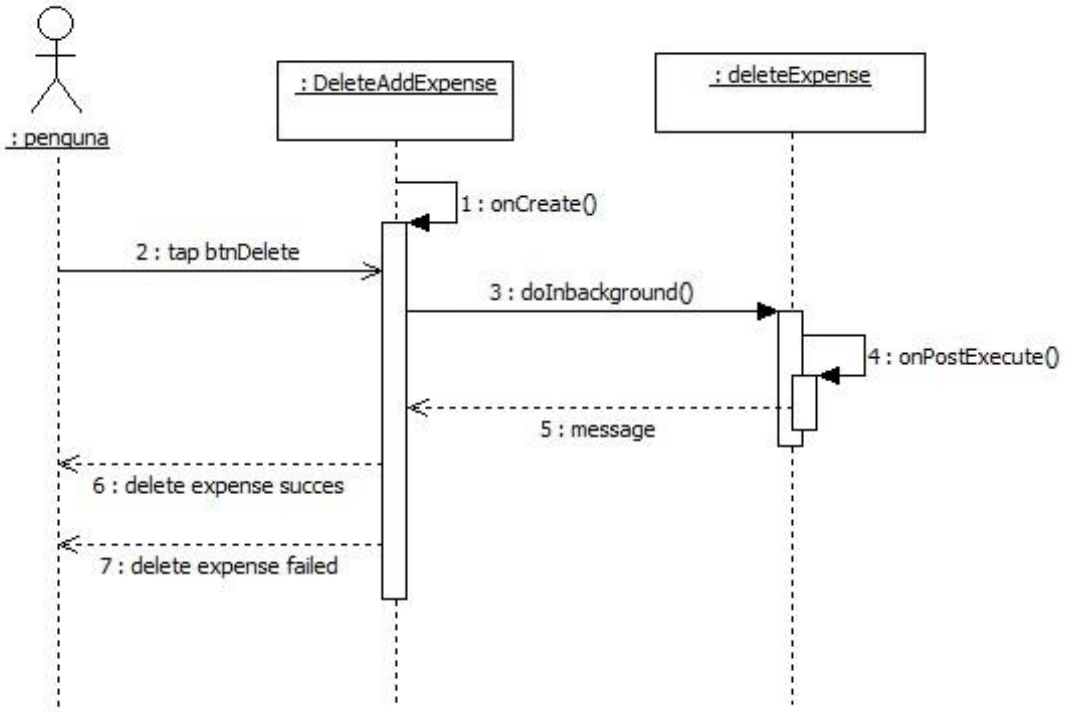
Berikut ini adalah interaksi antara aktor dengan objek-objek yang ada pada proses memperbarui pengeluaran travelling dapat dilihat pada gambar 3.22 Sequence diagram memperbarui pengeluaran.



Gambar 3. 22 Sequence Diagram Memperbarui Pengeluaran

8. Sequence Diagram Menghapus Pengeluaran

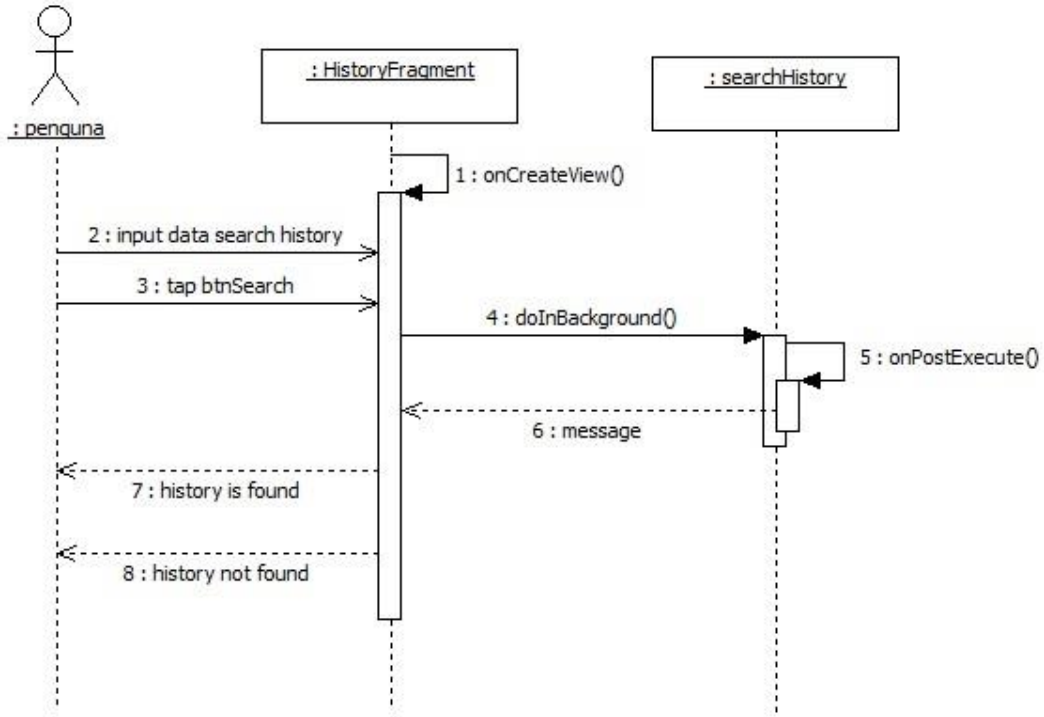
Berikut ini adalah interaksi antara aktor dengan objek-objek yang ada pada proses menghapus pengeluaran travelling dapat dilihat pada gambar 3.23 Sequence diagram menghapus pengeluaran.



Gambar 3. 23 Sequence Diagram Menghapus Pengeluaran

9. Sequence Diagram Mencari Riwayat

Berikut ini adalah interaksi antara aktor dengan objek-objek yang ada pada proses mencari riwayat travelling dapat dilihat pada gambar 3.24 Sequence diagram mencari riwayat.



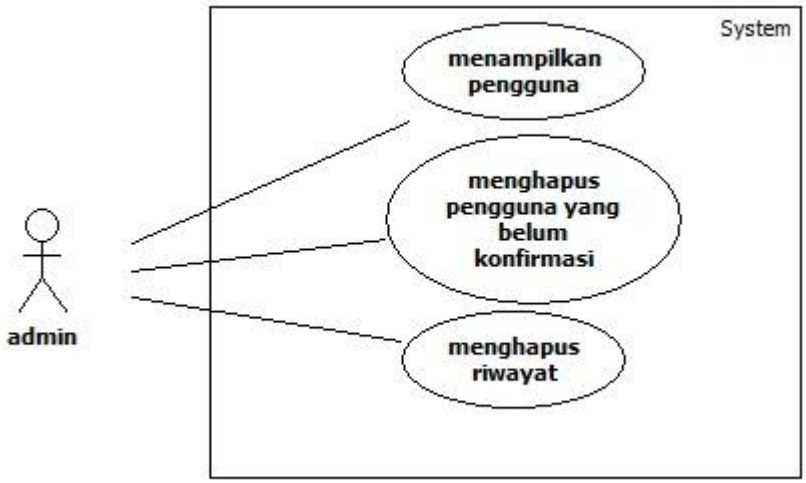
Gambar 3. 24 Sequence Diagram Mencari Riwayat

3.1.5.2 Analisis Kebutuhan Fungsional di Platform Website

Analisis kebutuhan fungsional perangkat lunak get trip di platform website dilakukan menggunakan *tools* UML, adapun tahapan analisis sistem menggunakan UML meliputi *use case diagram, Use Case Scenario, activity diagram*, dan *class diagram.* Analisis kebutuhan fungsional perangkat lunak get trip di platform website akan dijelaskan sebagai berikut :

3.1.5.2.1 Use Case Diagram

Use Case Diagram merupakan konstruksi untuk mendeskripsikan hubungan - hubungan yang terjadi antar *actor* dengan aktifitas yang terdapat pada sistem. Sasaran pemodelan *use case* diantaranya adalah mendefinisikan kebutuhan fungsional dan operasional sistem dengan mendefinisikan *scenario* penggunaan yang disepakati antara pemakai dan pengembang. Dari analisis pengguna aplikasi yang ada maka *use case diagram* untuk perangkat lunak get trip di platform website dapat dilihat pada gambar 3.25 Use case diagram perangkat lunak get trip di platform website.



Gambar 3. 25 Use Case Diagram Perangkat Lunak Get rip di Platform Website

3.1.5.2.2 Use Case Scenario

Use Case Scenario menjelaskan skenario dari setiap proses bisnis yang digambarkan pada use case diagram. Berdasarkan use case diagram pada Gambar 3. 28 Use Case Diagram Perangkat Lunak Get Trip di Platform Website, maka *use case scenario* untuk aplikasi yang akan dibangun akan dijelaskan sebagai berikut :

1. Use Case Scenario Menampilkan Pengguna

Use case scenario menampilkan pengguna menggambarkan langkah-langkah admin untuk melihat data pengguna. Use case scenario menampilkan pengguna dapat dilihat pada Tabel 3.27 Use case Scenario Menampilkan Pengguna.

Tabel 3. 27 Use Case Scenario Menampilkan Pengguna

Use Case Name	**Menampilkan Pengguna**	
Related Requirements	SKPL-F-13	
Goal In Context	Admin melihat data pengguna	
Precondition	Admin belum melihat data pengguna	
Successful End Condition	Admin berhasil melihat data pengguna	
Failed End Condition	Admin gagal melihat data pengguna	
Actors	Admin	
Trigger	Admin mengklik tombol pengguna	
Main Flow	Step	Action
	1	Admin mengklik tombol pengguna
	2	Sistem mencari data pengguna yang sudah konfirmasi
	3	Sistem menampilkan data pengguna
Extension	Step	Branching Action
	2.1	Sistem tidak menemukan data pengguna yang sudah konfirmasi

2. Use Case Scenario Menghapus Pengguna Yang Belum Konfirmasi

Use case scenario menghapus pengguna yang belum konfirmasi menggambarkan langkah-langkah admin untuk menghapus pengguna yang belum konfirmasi. Use case scenario menghapus pengguna yang belum konfirmasi dapat dilihat pada Tabel 3.28 Use Case Scenario Menghapus Pengguna Yang Belum Konfirmasi.

Tabel 3. 28 Use Case Scenario Menghapus Pengguna Yang Belum Konfirmasi

Use Case Name	**Menghapus Pengguna Yang Belum Konfirmasi**	
Related Requirements	SKPL-F-14	
Goal In Context	Admin memperbarui pengguna yang belum konfirmasi	
Precondition	Pengguna yang belum konfirmasi belum dihapus	
Successful End Condition	Admin berhasil menghapus pengguna yang belum konfirmasi	
Failed End Condition	Admin gagal menghapus pengguna yang belum konfirmasi	
Actors	Admin	
Trigger	Admin mengklik tombol hapus	
Main Flow	Step	Action
	1	Admin mengklik hapus
	2	Sistem menampilkan pesan konfirmasi penghapusan pengguna yang belum konfirmasi
	3	Admin melakukan konfirmasi penghapusan pengguna yang belum konfirmasi
	4	Data pengguna berhasil di hapus
	5	Sistem menampilkan halaman pengguna
Extension	Step	Branching Action
	3.1	Admin membatalkan penghapusan data pengguna

3. Use Case Scenario Menghapus Riwayat

Use case scenario menghapus riwayat menggambarkan langkah-langkah admin untuk menghapus riwayat *travelling*. Use case scenario memperbarui riwayat dapat dilihat pada Tabel 3.29 Use Case Scenario Menghapus Riwayat.

Tabel 3. 29 Use Case Scenario Menghapus Riwayat

Use Case Name	**Menghapus Riwayat**
Related Requirements	SKPL-F-14
Goal In Context	Admin menghapus riwayat travelling
Precondition	Riwayat travelling akan dihapus
Successful End Condition	Admin berhasil menghapus data riwayat travelling
Failed End Condition	Admin gagal menghapus data riwayat travelling
Actors	Admin
Trigger	Admin mengklik tombol hapus riwayat

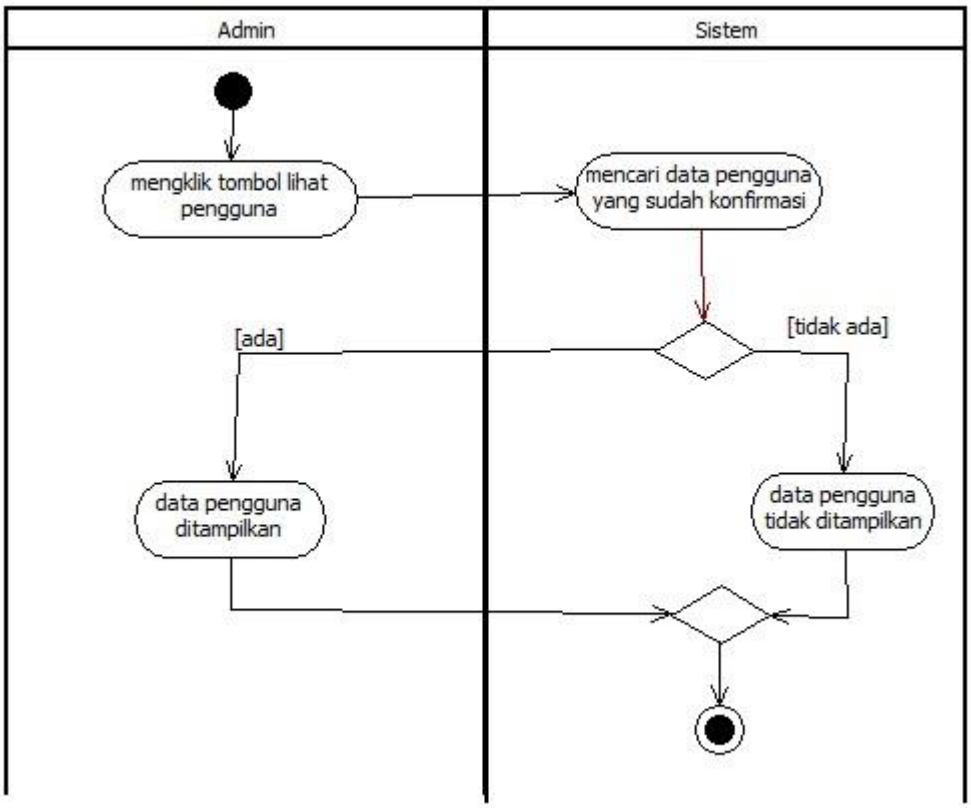
Main Flow	Step	Action
	1	Admin mengklik tombol hapus riwayat
	2	Sistem menampilkan pesan konfirmasi penghapusan data riwayat travelling
	3	Admin melakukan konfirmasi penghapusan data riwayat travelling
	4	Data riwayat travelling berhasil di hapus
	5	Sistem menampilkan halaman riwayat travelling
Extension	Step	Branching Action
	3.1	Admin membatalkan penghapusan data pengeluaran travelling

3.1.5.2.3 Activity Diagram

Activity Diagram menggambarkan proses bisnis dan urutan aktifitas dalam sebuah proses, dan digunakan pada business modeling untuk memperlihatkan urutan aktifitas proses bisnis. *Activity Diagram* sangat bermanfaat untuk memahami proses dari sistem secara keseluruhan. *Activity Diagram* dibuat berdasarkan sebuah atau beberapa *use case* pada *use case diagram.* Berikut ini penjelasan dari masing-masing *activity diagram.*

1. Activity Diagram Menampilkan Pengguna

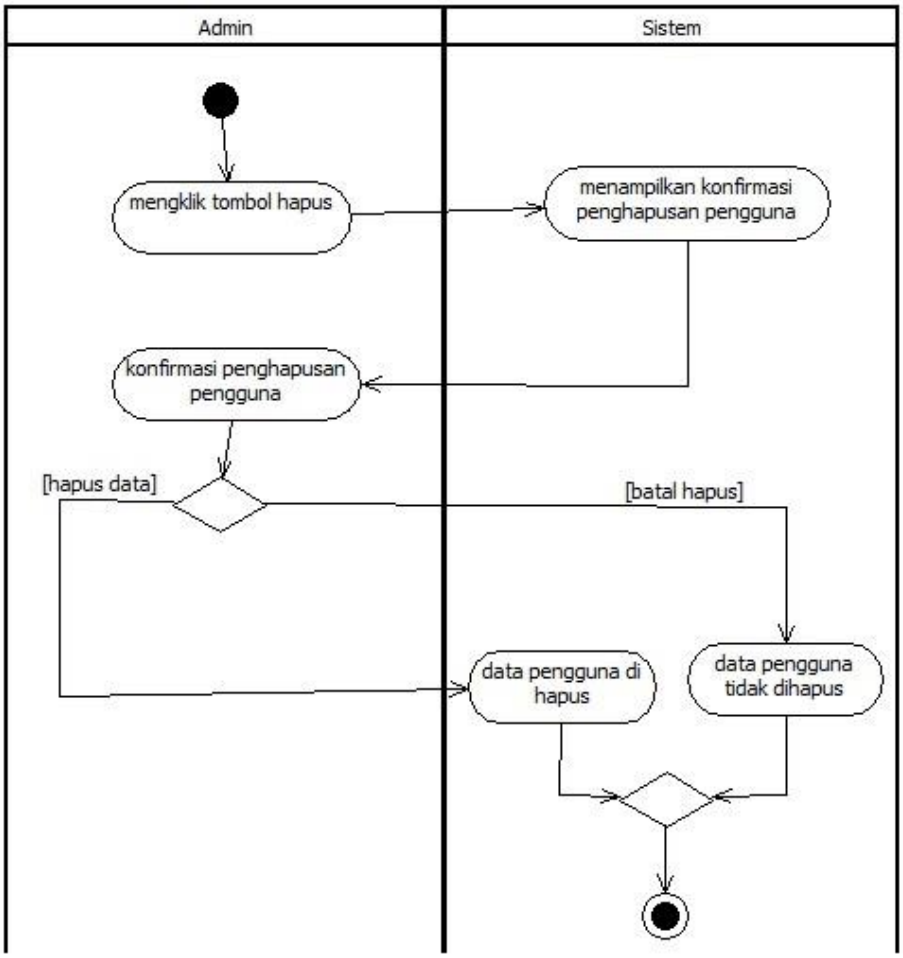
Proses menampilkan pengguna pada aplikasi get trip dapat dilihat pada pada gambar 3.26 Activity diagram menampilkan pengguna.



Gambar 3. 26 Activity Diagram Menampilkan Pengguna

2. Activity Diagram Menghapus Pengguna Yang Belum Konfirmasi

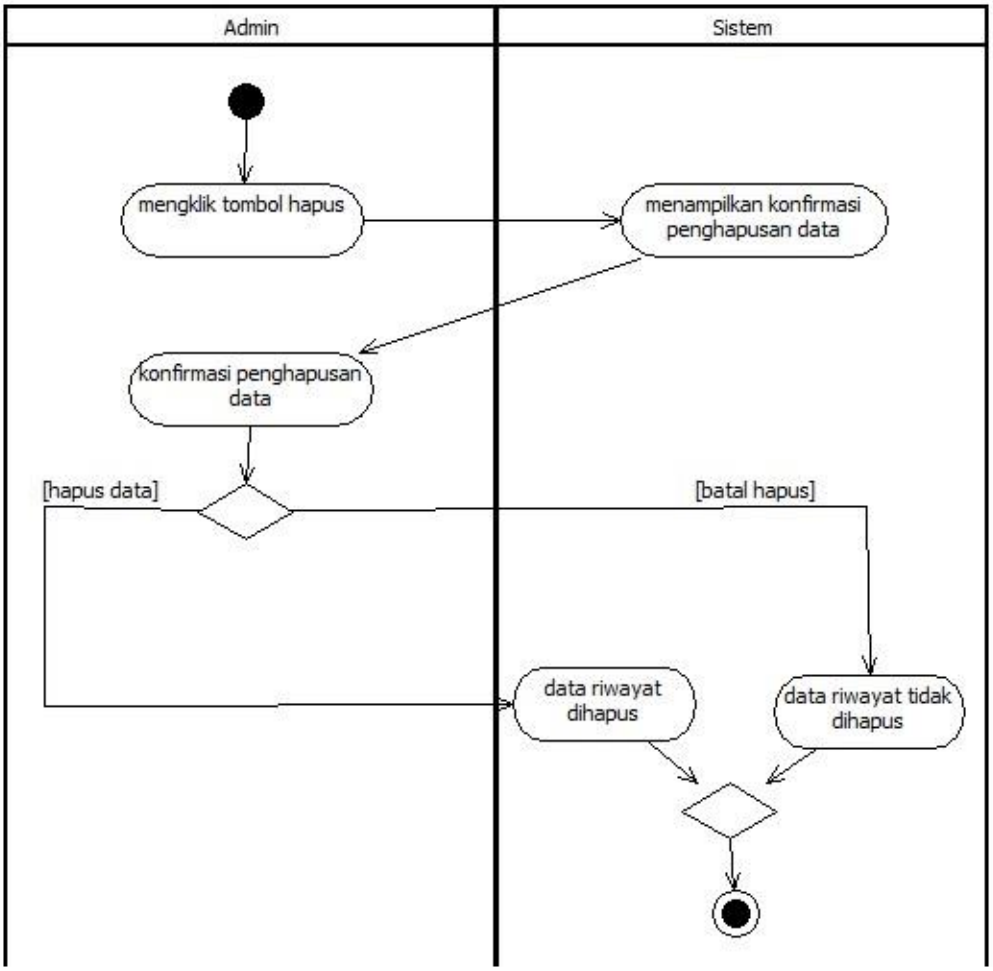
Proses menghapus pengguna yang belum konfirmasi pada aplikasi get trip dapat dilihat pada pada gambar 3.27 Activity diagram menghapus pengguna yang belum konfirmasi.



Gambar 3. 27Activity Diagram Menghapus Pengguna Yang Belum Konfirmasi

3. Activity Diagram Menghapus Riwayat

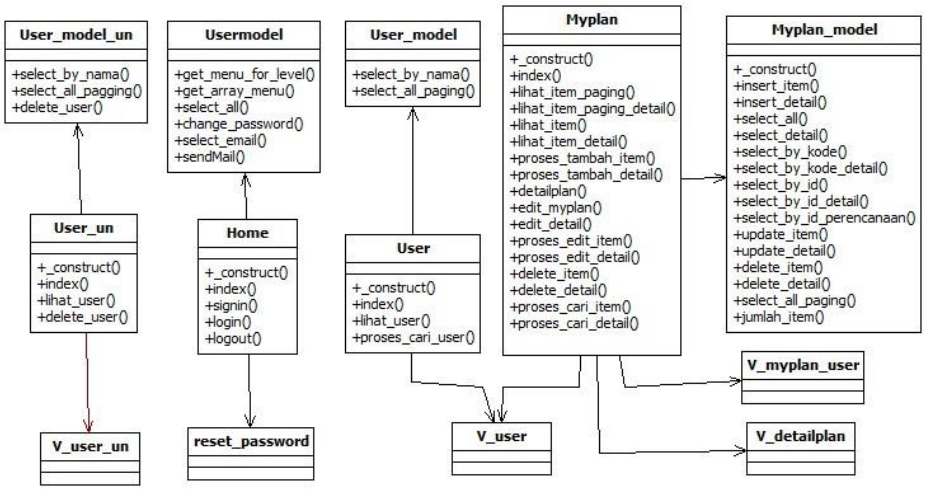
Proses menghapus riwayat pada aplikasi get trip dapat dilihat pada pada gambar 3.28 Activity diagram menghapus riwayat.



Gambar 3. 28Activity Diagram Menghapus Riwayat

3.1.5.2.4 Class Diagram

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun perangkat lunak get trip. Gambar 3.29 Class Diagram Platform Website Perangkat Lunak Get Trip, menunjukan struktur class satu yang berelasi dengan class lainnya.



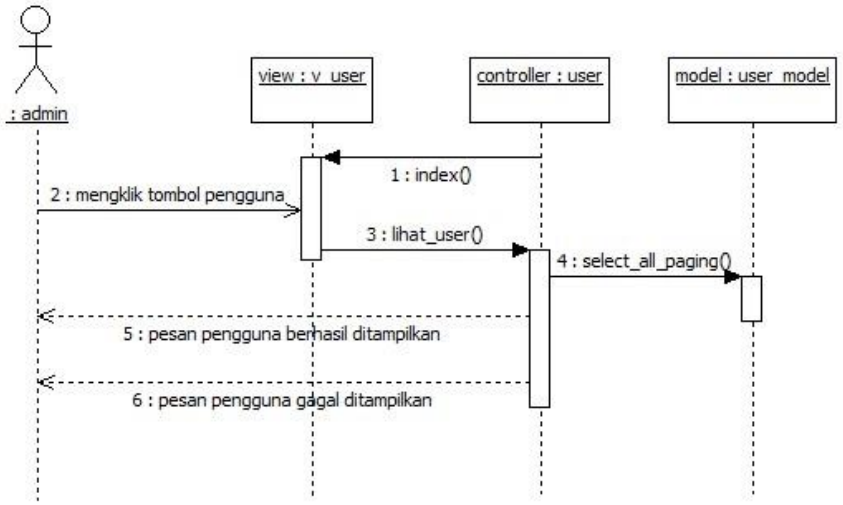
Gambar 3. 29 Class Diagram Platform Website Perangkat Lunak Get Trip

3.1.5.2.5 Sequence Diagram

Sequence diagram merupakan gambaran interaksi masing-masing objek pada setiap use case dalam urutan waktu. Interaksi ini berupa pengiriman serangkaian data antar objek-objek yang saling berinteraksi. Berikut ini adalah *sequence diagram* yang dapat digambarkan, diantaranya adalah

1. Sequence Diagram Menampilkan Pengguna

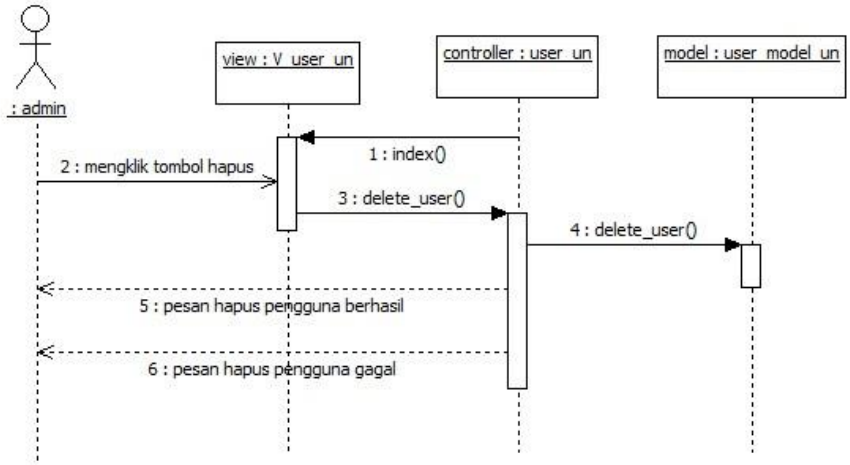
Berikut ini adalah interaksi antara aktor dengan objek-objek yang ada pada proses menampilkan pengguna dapat dilihat pada gambar 3.30 Sequence diagram menampilkan pengguna.



Gambar 3. 30 Sequence Diagram Menampilkan Pengguna

2. Sequence Diagram Menghapus Pengguna Yang Belum Konfirmasi

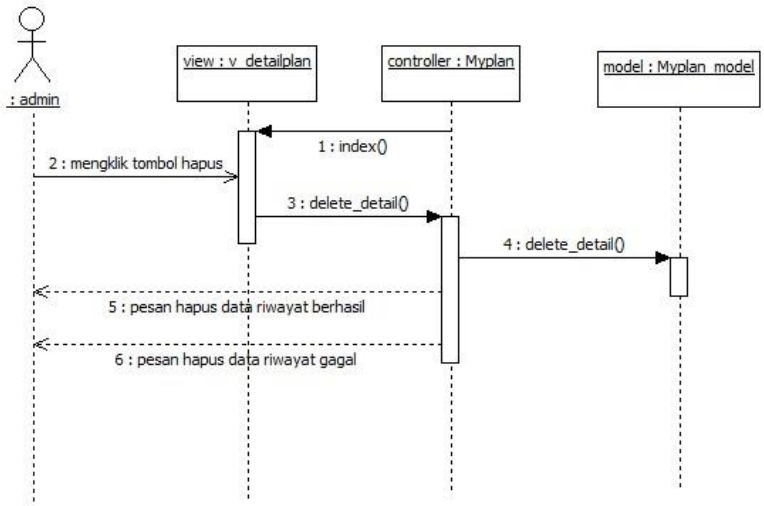
Berikut ini adalah interaksi antara aktor dengan objek-objek yang ada pada proses menghapus pengguna yang belum konfirmasi dapat dilihat pada gambar 3.31 Sequence diagram menghapus diagram menampilkan pengguna.



Gambar 3. 31 Sequence Diagram Menghapus Pengguna Yang Belum Konfirmasi

3. Sequence Diagram Menghapus Riwayat

Berikut ini adalah interaksi antara aktor dengan objek-objek yang ada pada proses menghaps riwayat dapat dilihat pada gambar 3.32 Sequence diagram menghapus riwayat.



Gambar 3. 32 Sequence Diagram Menghapus Riwayat

3.2 Perancangan Sistem

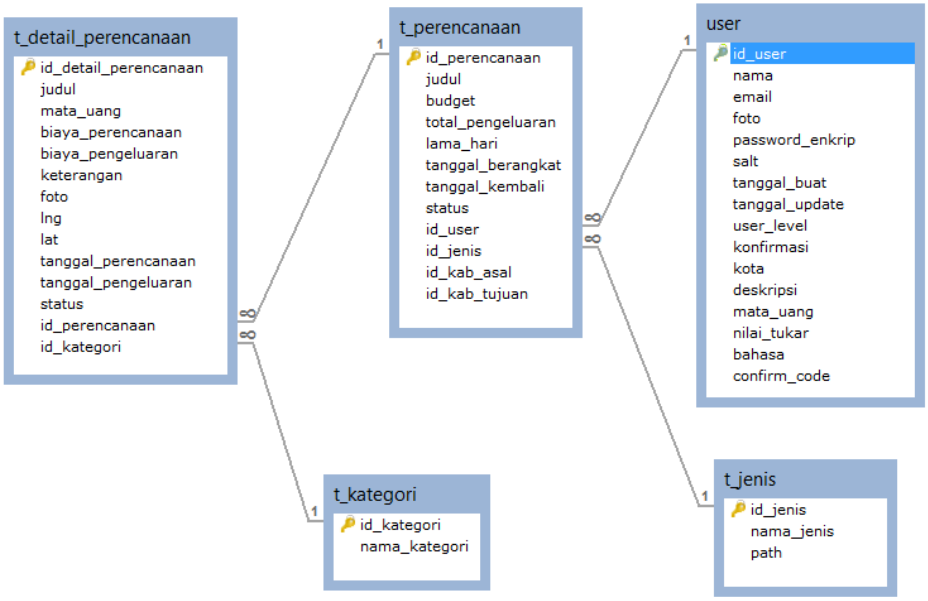
Perancangan sistem merupakan penggambaran, perencanaan, dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terdapat dalam perangkat lunak get trip.

3.2.1 Perancangan Data

Perancangan data merupakan tahapan untuk memetakan model konseptual ke model basis data yang akan dipakai. Perancangan basis data terbagi menjadi dua yaitu skema relasi dan perancangan struktur tabel.

3.2.1.1 Diagram Relasi

Diagram relasi merupaka rangkaian hubungan antara dua tabel atau lebih pada sistem basis data. Gambar 3.33 Diagram Relasi Perangkat Lunak Get Trip, merupakan gambaran rangkaian basis data pada perangkat lunak get trip.



Gambar 3. 33 Diagram Relasi Perangkat Lunak Get Trip

3.2.1.2 Struktur Tabel

Struktur tabel menggambarkan detail tabel yang berisi field, tipe data, panjang data, dan keterangann lainnya. Tabel yang terdapat pada basis data perangkat lunak get trip adalah tabel user, t_perencanaan, t_detail_perencaaan, t_jenis, t_kabupaten dan t_kategori yang dapat dilihat pada tabel-tabel dibawah ini.

1. Tabel user

Berikut ini adalah struktur tabel dari user yang ada pada perangkat lunak get trip, dapat dilihat pada tabel 3.30 Struktur tabel user.

Tabel 3. 30 Struktur Tabel User

Nama Field	Tipe Data	Panjang	Null	Ekstra	Kunci
id_user	int	11	no	Auto increment	Primary key
nama	varchar	50	no		
email	varchar	100	no		
foto	varchar	100	no		
password_enkrip	varchar	100	no		
salt	varchar	10	no		
tanggal_buat	date		no		
tanggal_update	date		no		
user_level	int	1	no		
konfirmasi	int	1	no		
kota	varchar	100	no		
deskripsi	text		no		

mata_uang	varchar	4	no		
nilai_tukar	double		no		
bahasa	varchar	15	no		
confirm_code	varchar	100	no		

2. Tabel t_perencanaan

Berikut ini adalah struktur tabel dari t_perencanaan yang ada pada perangkat lunak get trip, dapat dilihat pada tabel 3.31 Struktur tabel t_perencanaan.

Tabel 3. 31 Struktur Tabel t_perencanaan

Nama Field	Tipe Data	Panjang	Null	Ekstra	Kunci
id_perencanaan	int	11	no	Auto increment	Primary key
id_user	int	11	no		Foreign key reference tabel user
id_jenis	int	1	no		Foreign key reference tabel t_jenis
id_kab_asal	varchar	100	no		
id_kab_tujuan	varchar	100	no		
judul	varchar	50	no		
tujuan	varchar	50	no		
asal	varchar	50	no		
budget	double		no		
total_pengeluaran	double		no		

lama_hari	int	3	no		
tanggal_berangkat	date		no		
tanggal_kembali	date		no		
status	varchar	20	no		

3. Tabel t_detail_perencanaan

Berikut ini adalah struktur tabel dari t_detail_perencanaan yang ada pada perangkat lunak get trip, dapat dilihat pada tabel 3.32 Struktur tabel t_detail_perencanaan.

Tabel 3. 32 Stuktur Tabel t_detail_perencanaan

Nama Field	Tipe Data	Panjang	Null	Ekstra	Kunci
id_detail_perencanaan	int	11	no	Auto increment	Primary key
id_perencanaan	int	11	no		Foreign key reference tabel t_perencanaan
id_kategori	int	1			Foreign key reference tabel t_kategori
judul	varchar	100	no		
mata_uang	varchar	4	no		
biaya_perencanaan	double	-	no		
biaya_pengeluaran	double	-	no		
keterangan	text	-	no		

foto	varchar	100	no		
lng	decimal	10,6	no		
lat	decimal	10,6	no		
tanggal_perencanaan	date	-	no		
tanggal_pengeluaran	date	-	no		
status	varchar	20	no		

4. Tabel t_jenis

Berikut ini adalah struktur tabel dari t_jenis yang ada pada perangkat lunak get trip, dapat dilihat pada tabel 3.33 Struktur tabel t_jenis.

Tabel 3. 33 Struktur Tabel t_jenis

Nama Field	Tipe Data	Panjang	Null	Ekstra	Kunci
id_jenis	int	1	no	Auto increment	Primary key
nama_jenis	varchar	30	no		
path	varchar	50	no		

5. Tabel t_kategori

Berikut ini adalah struktur tabel dari t_kategori yang ada pada perangkat lunak get trip, dapat dilihat pada tabel 3.34 Struktur tabel t_kategori.

Tabel 3. 34 Struktur Tabel t_kategori

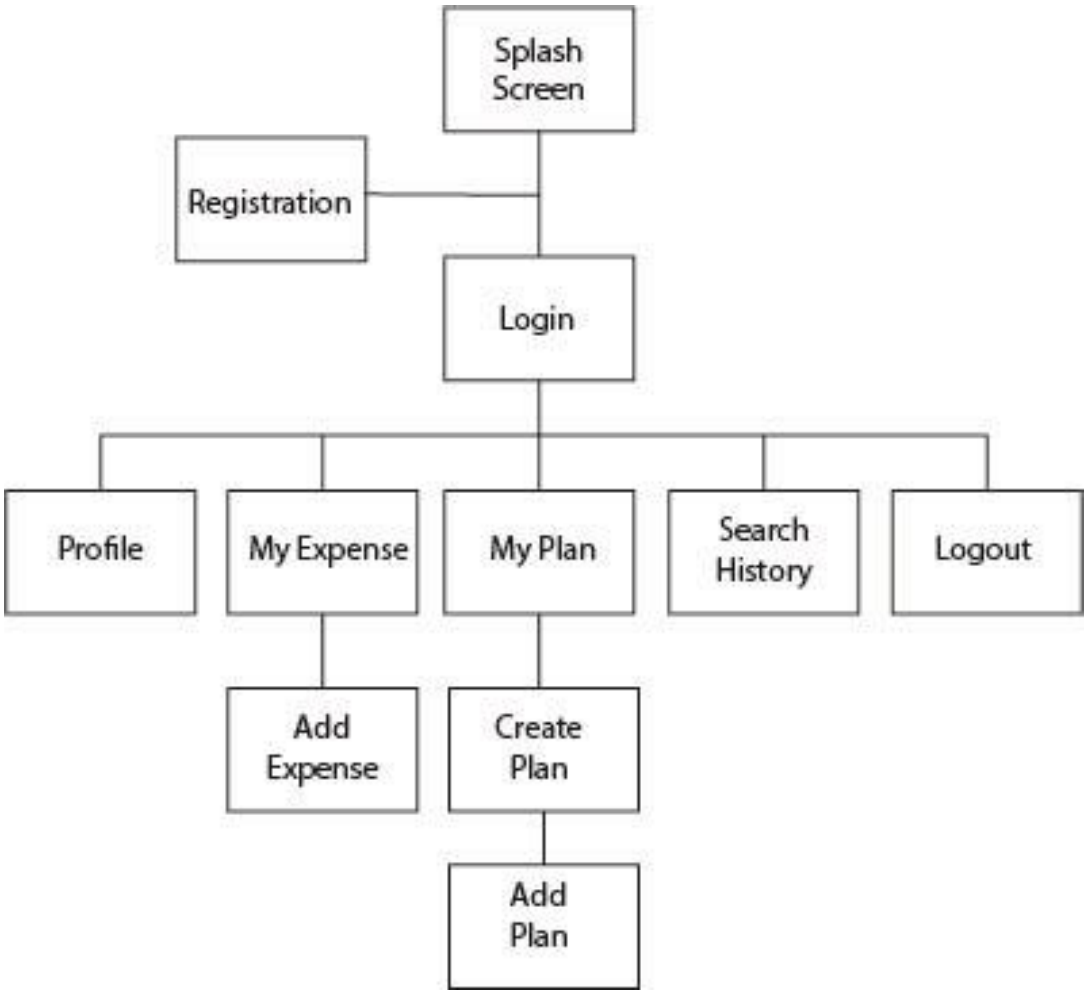
Nama Field	Tipe Data	Panjang	Null	Ekstra	Kunci
id_kategori	int	1	no	Auto increment	Primary key
nama_kategori	varchar	30	no		

3.2.2 Struktur Menu

Perancangan struktur menu merupakan gambaran jalur pemakaian aplikasi sehingga aplikasi yang mudah dipahami dan mudah digunakan. Perancangan struktur menu dari perangkat lunak get trip dapat dilihat sebagai berikut.

1. Struktur Menu Platform Mobile

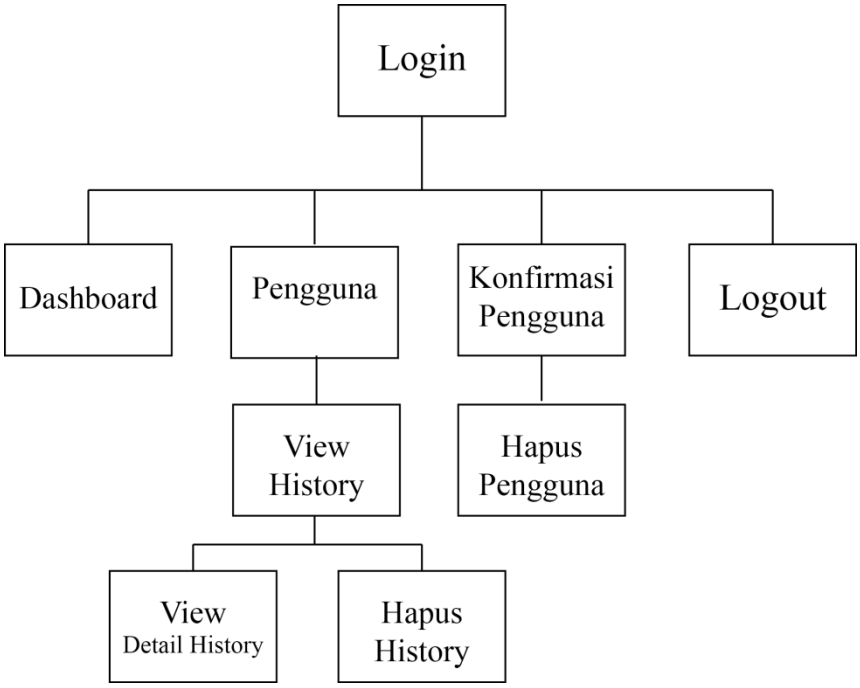
Berikut ini adalah gambar 3.34 Struktur menu platform mobile di perangkat lunak get trip.



Gambar 3. 34 Struktur Menu Platform Mobile Perangkat Lunak Get Trip

2. Struktur Menu Platform Website

Berikut ini adalah gambar 3.35 Struktur menu platform website di perangkat lunak get trip.



Gambar 3. 35 Struktur Menu Platform Website Perangkat Lunak Get Trip

3.2.3 Perancangan Antar Muka

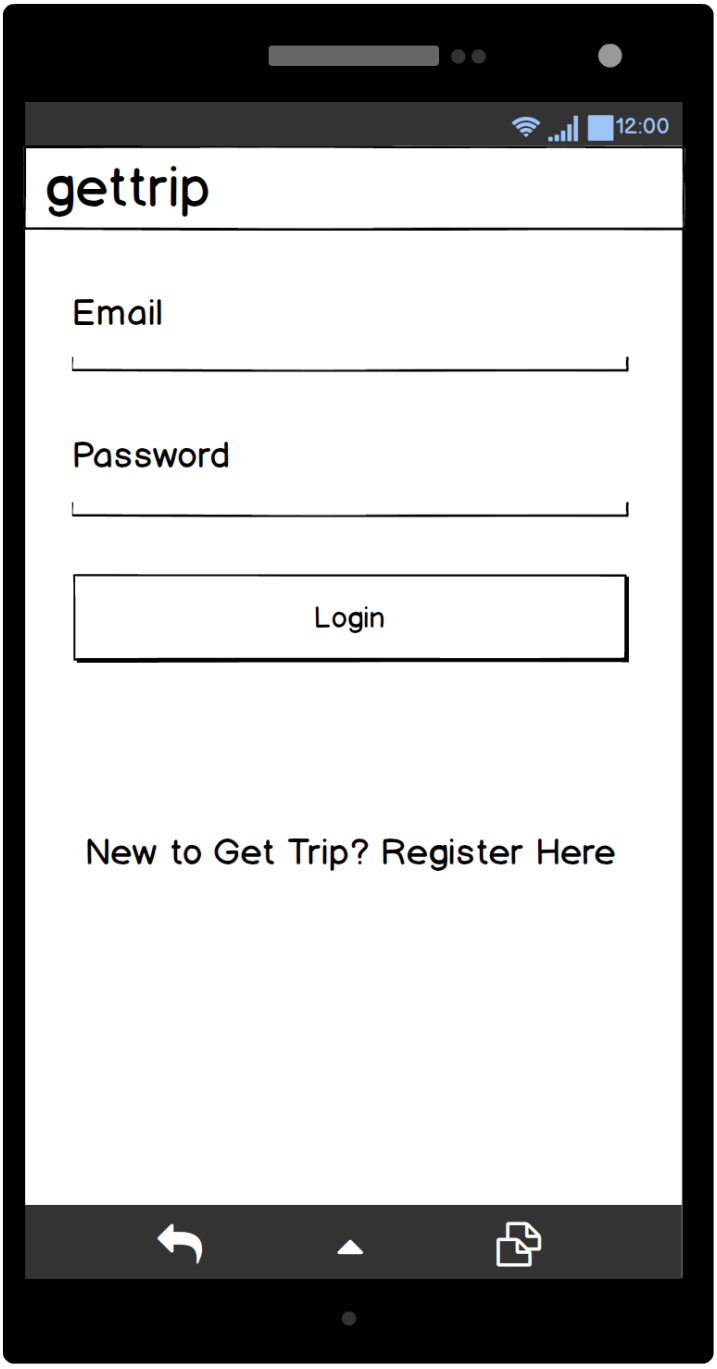
Perancangan antarmuka mendeskripsikan rencana tampilan dari setiap tampilan yang akan digunakan pada perangkat lunak get trip terdiri dari perancangan form dan pesan.

3.2.3.1 Perancangan Form Platform Mobile

Perancangan antarmuka form platform mobile pada perangkat lunak get trip dapat dilihat pada gambar-gambar dibawah ini.

1. Form Login

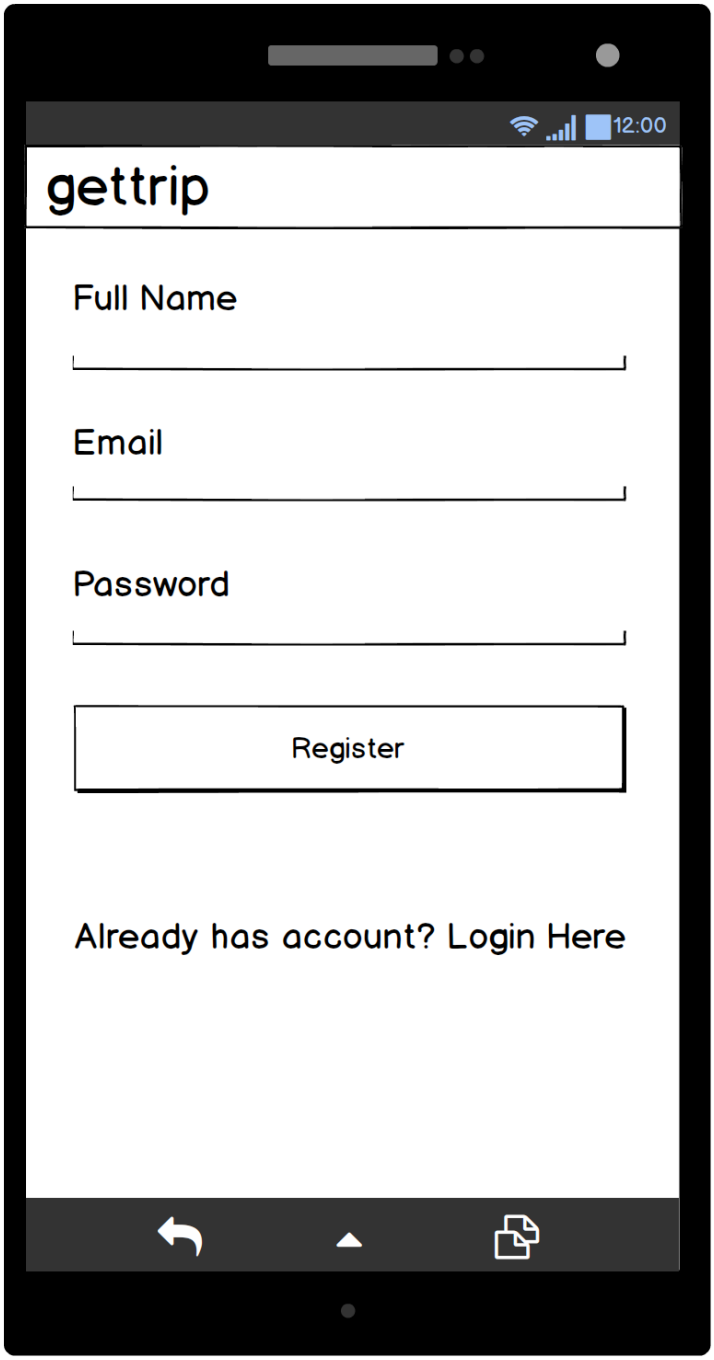
Berikut ini merupakan tampilan form login pada perangkat lunak get trip yang dapat dilihat pada Gambar 3.36 P01 Form Login.



Gambar 3. 36 P01 Form Login

2. Form Registrasi

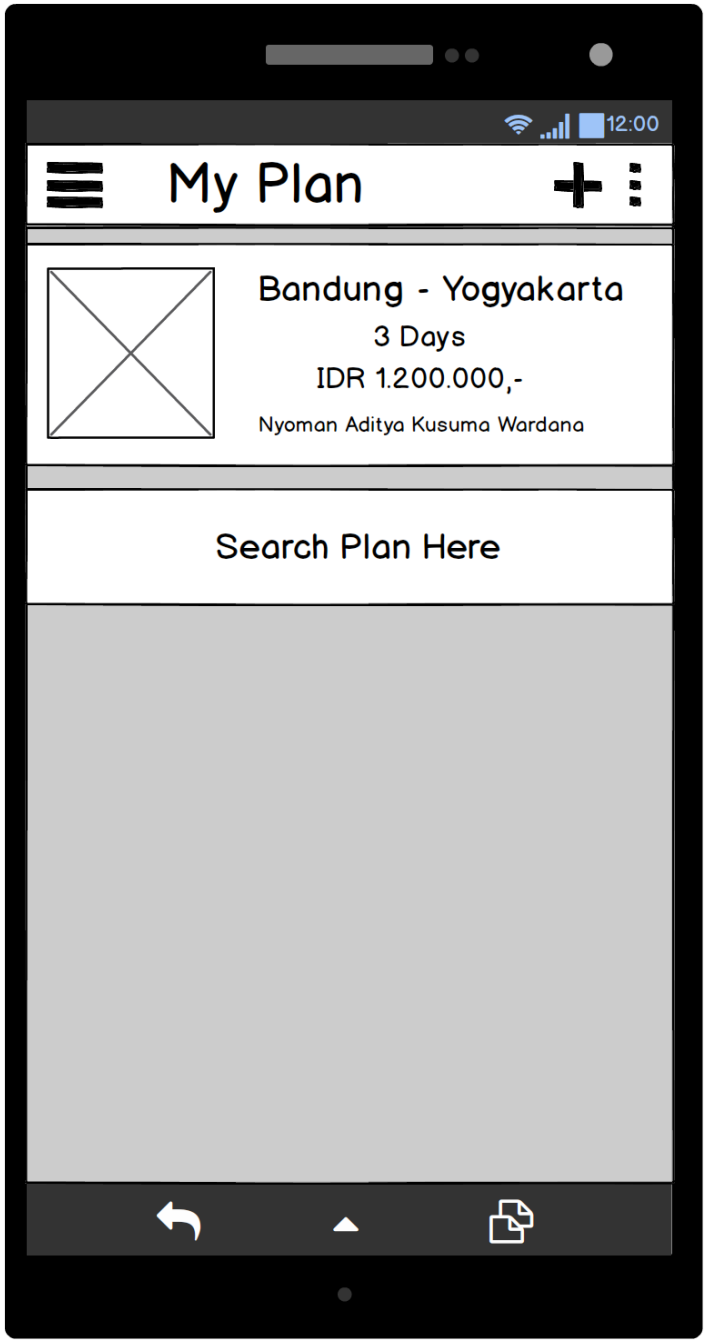
Berikut ini merupakan tampilan form registrasi pada perangkat lunak get trip yang dapat dilihat pada Gambar 3.37 P02 Form Registrasi.



Gambar 3. 37 P02 Form Registrasi

3. Form My Plan

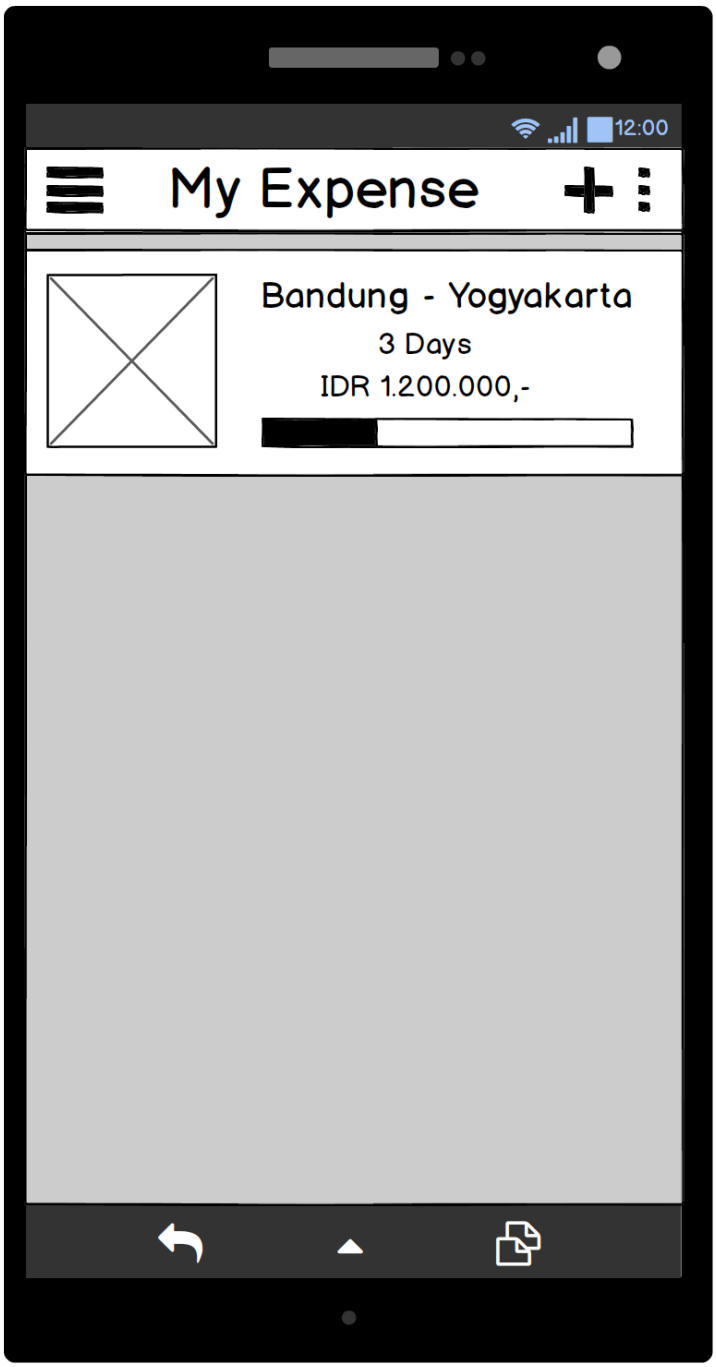
Berikut ini merupakan tampilan form my plan pada perangkat lunak get trip yang dapat dilihat pada Gambar 3.38 P03 Form My Plan.



Gambar 3. 38 P03 Form My Plan

4. Form My Expense

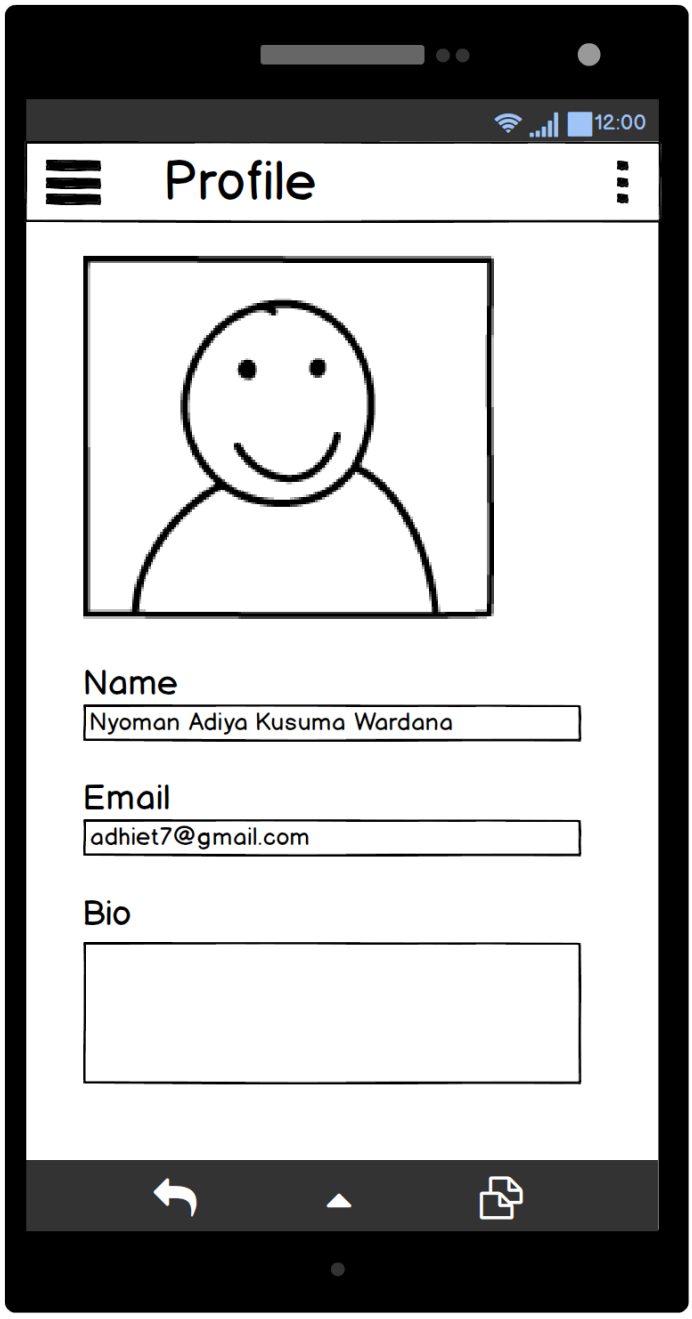
Berikut ini merupakan tampilan form my expense pada perangkat lunak get trip yang dapat dilihat pada Gambar 3.39 P04 Form My Expense



Gambar 3. 39 P04 Form My Expense

5. Form Profile

Berikut ini merupakan tampilan form profile pada perangkat lunak get trip yang dapat dilihat pada Gambar 3.40 P05 Form Profile.



Gambar 3. 40 P05 Form Profile

6. Form Search history

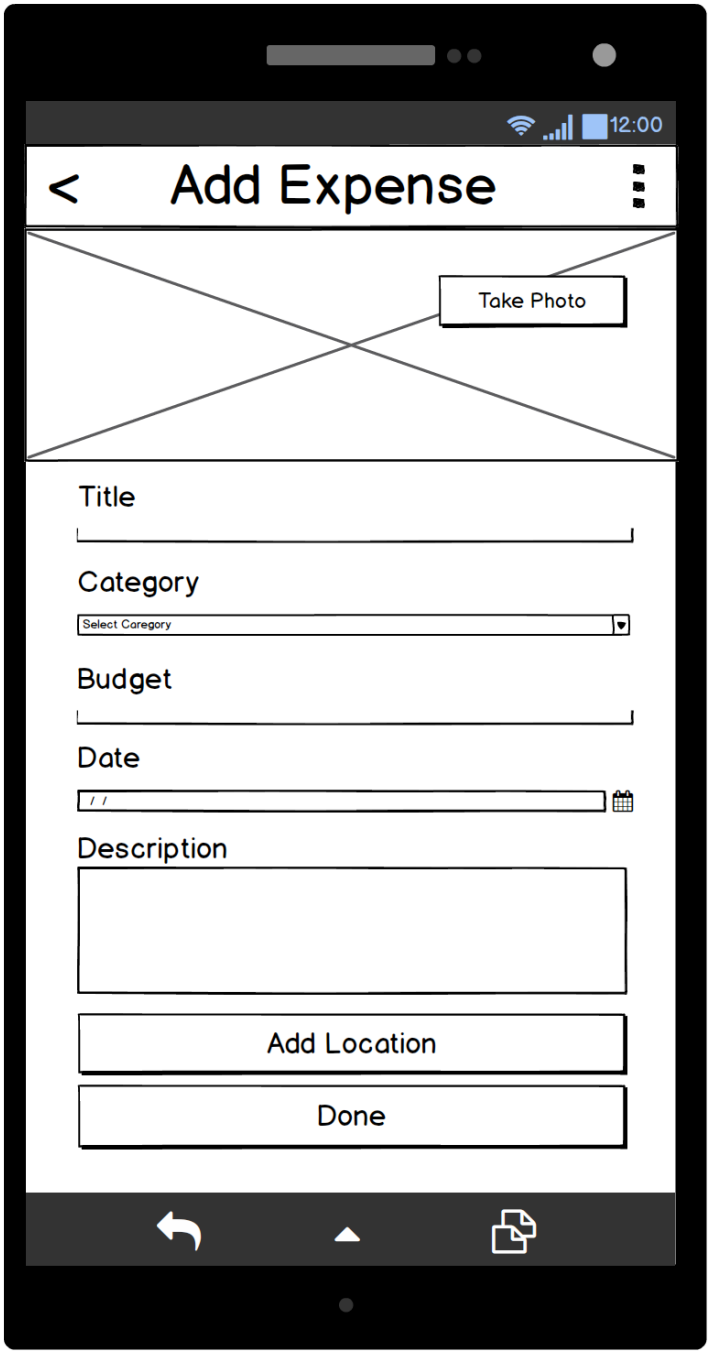
Berikut ini merupakan tampilan form search history pada perangkat lunak get trip yang dapat dilihat pada Gambar 3.41 P06 Form Search History.



Gambar 3. 41 P06 Form Search History

7. Form Add Expense

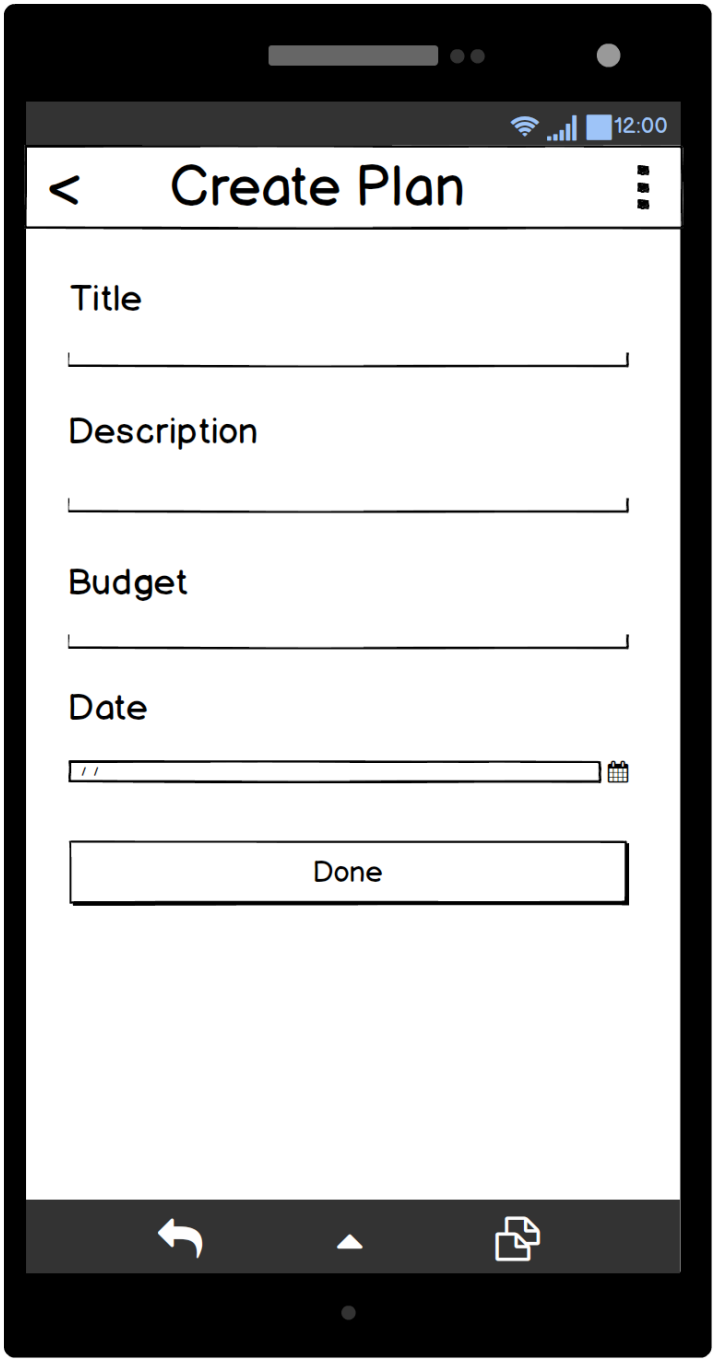
Berikut ini merupakan tampilan form add expense pada perangkat lunak get trip yang dapat dilihat pada Gambar 3.42 P07 Form Add Expense.



Gambar 3. 42 P07 Form Add Expense

8. Form Create Plan

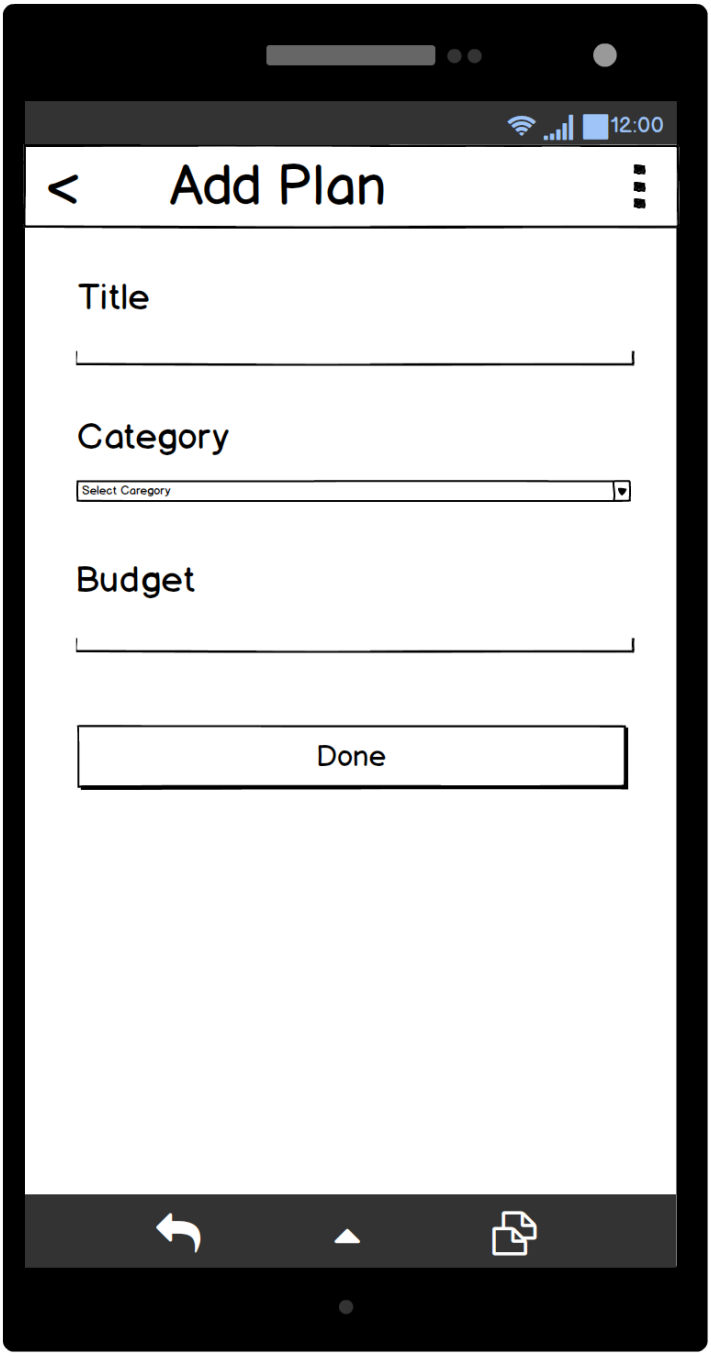
Berikut ini merupakan tampilan form create plan pada perangkat lunak get trip yang dapat dilihat pada Gambar 3.43 P08 Form Create Plan.



Gambar 3. 43 P08 Form Create Plan

9. Form Add Plan

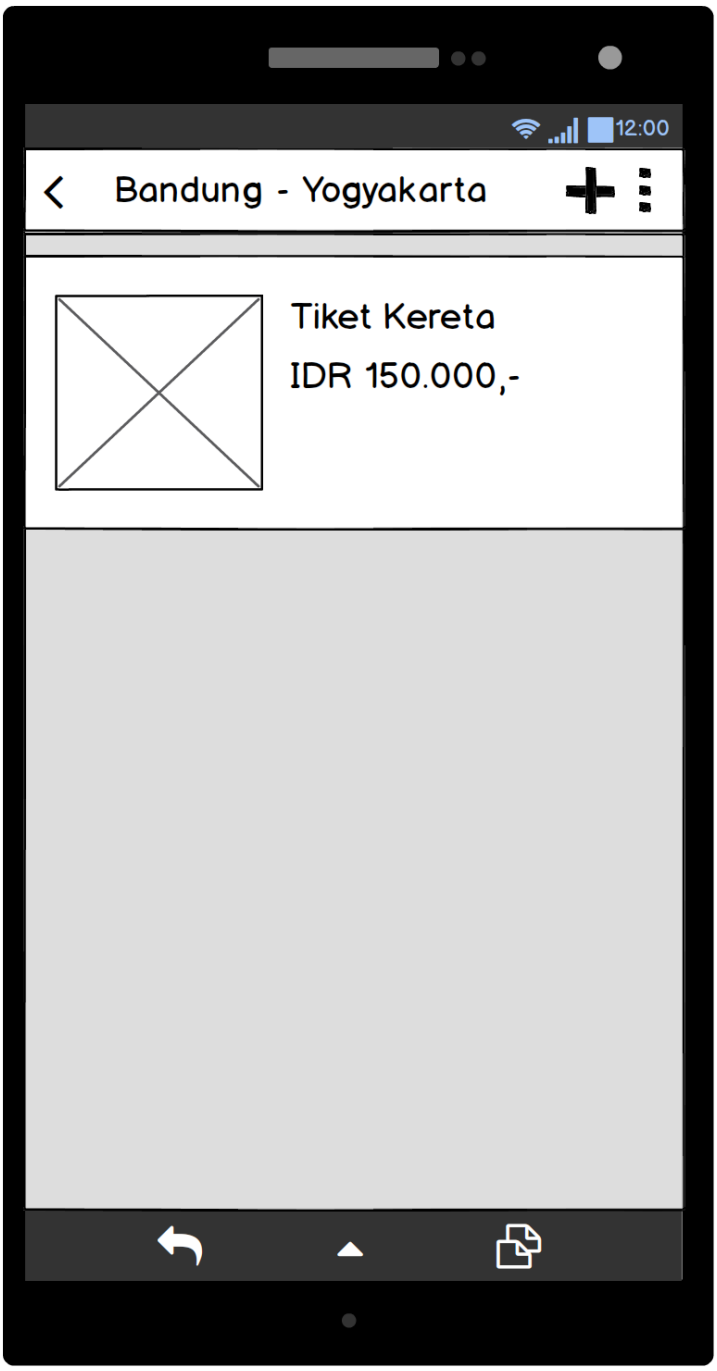
Berikut ini merupakan tampilan form add plan pada perangkat lunak get trip yang dapat dilihat pada Gambar 3.44 P09 Form Add Plan.



Gambar 3. 44 P09 Form Add Plan

10. Form Detail Planning

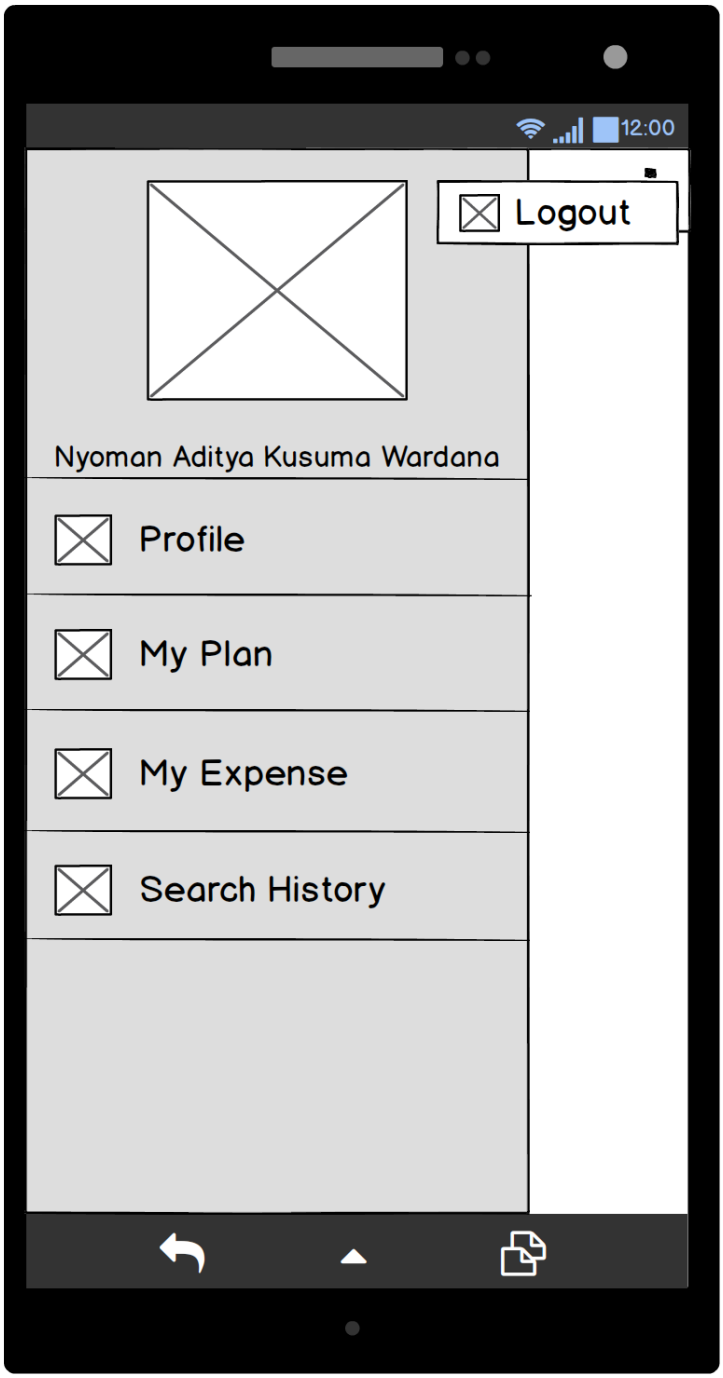
Berikut ini merupakan tampilan form detail planning pada perangkat lunak get trip yang dapat dilihat pada Gambar 3.45 P10 Form Detail Planning.



Gambar 3. 45 P10 Form Detail Planning

11. Form Drawer Menu

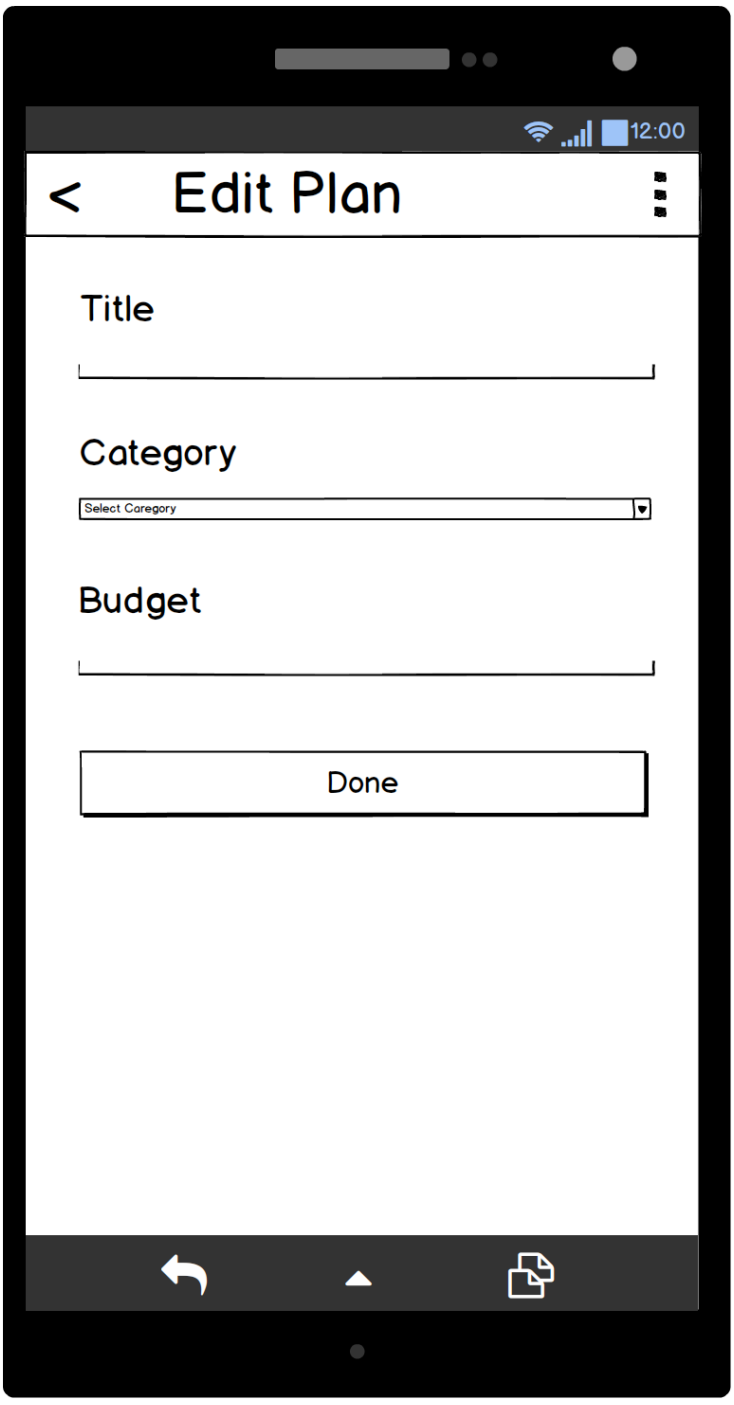
Berikut ini merupakan tampilan form drawer menu pada perangkat lunak get trip yang dapat dilihat pada Gambar 3.46 P11 Form Drawer Menu.



Gambar 3. 46 P11 Form Drawer Menu

12. Form Edit Plan

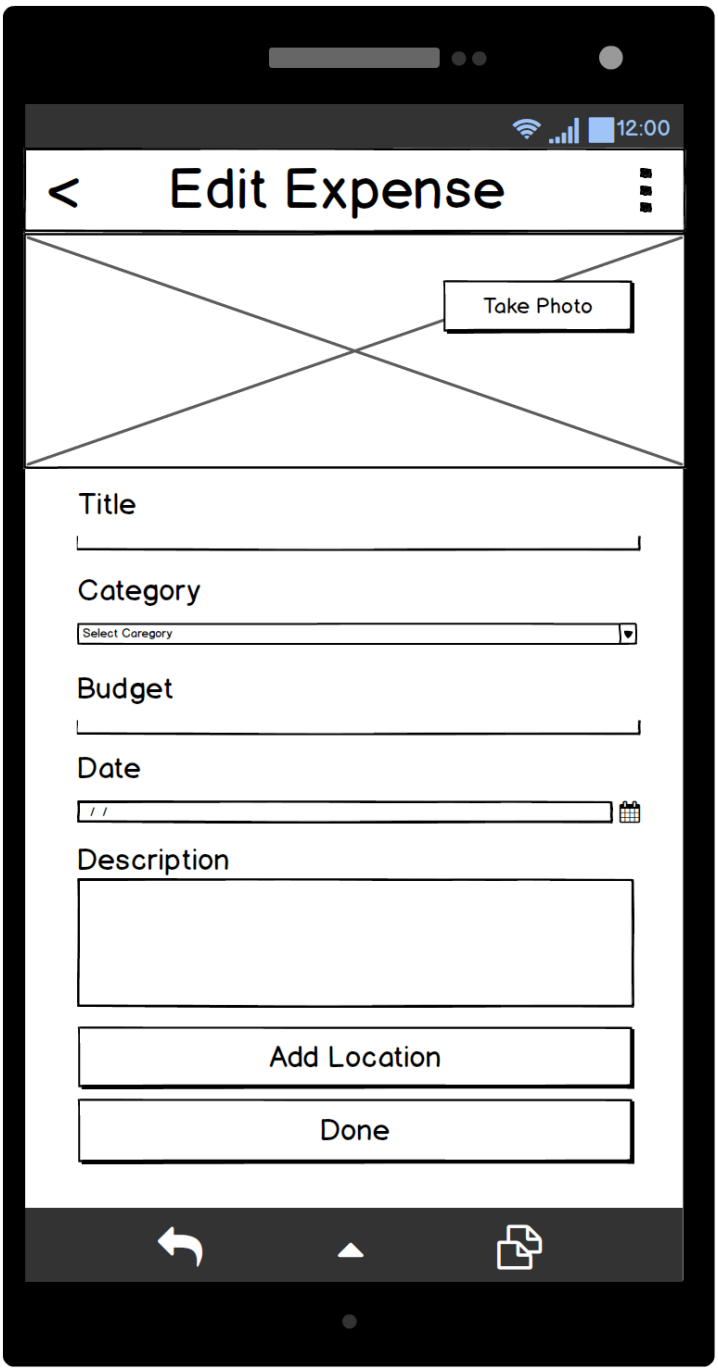
Berikut ini merupakan tampilan form drawer menu pada perangkat lunak get trip yang dapat dilihat pada Gambar 3.47 P12 Form Edit Plan.



Gambar 3. 47 P12 Form Edit Plan

13. Form Edit Expense

Berikut ini merupakan tampilan form drawer menu pada perangkat lunak get trip yang dapat dilihat pada Gambar 3.48 P11 Form Edit Expense.



Gambar 3. 48 P13 Form Edit Expense

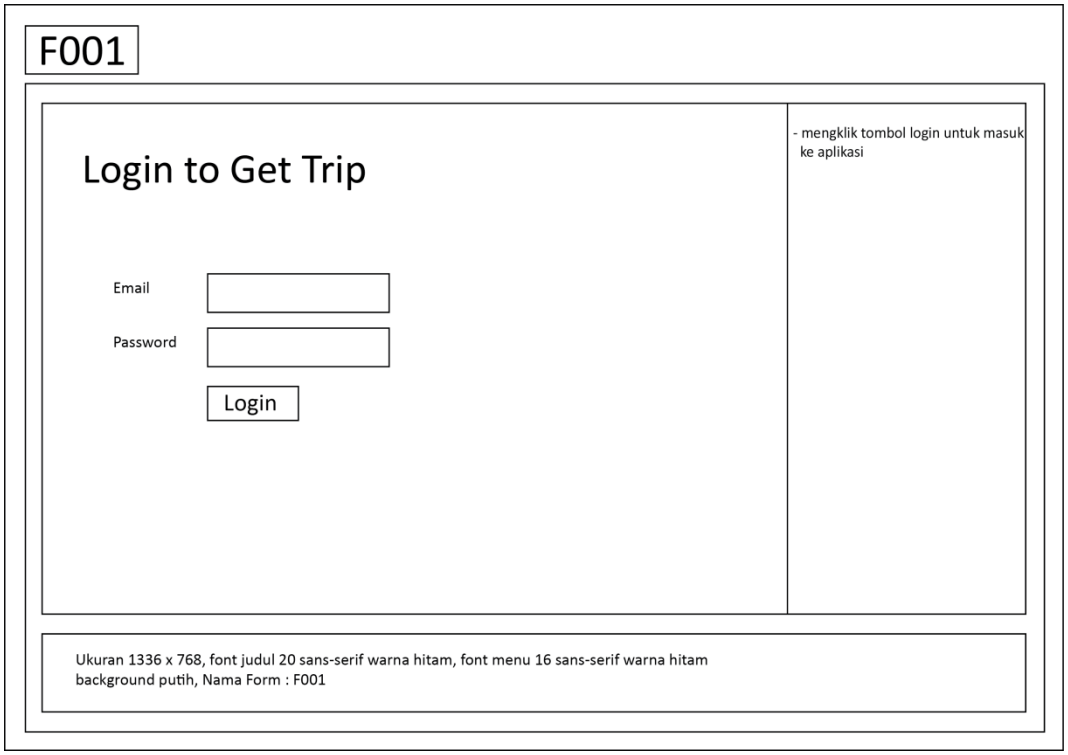
3.2.3.2 Perancangan Form Platform Website

Perancangan antarmuka form platform website pada perangkat lunak get trip dapat dilihat pada gambar-gambar dibawah ini.

Perancangan antarmuka form platform website pada perangkat lunak get trip dapat dilihat pada gambar-gambar dibawah ini.

1. Form Login

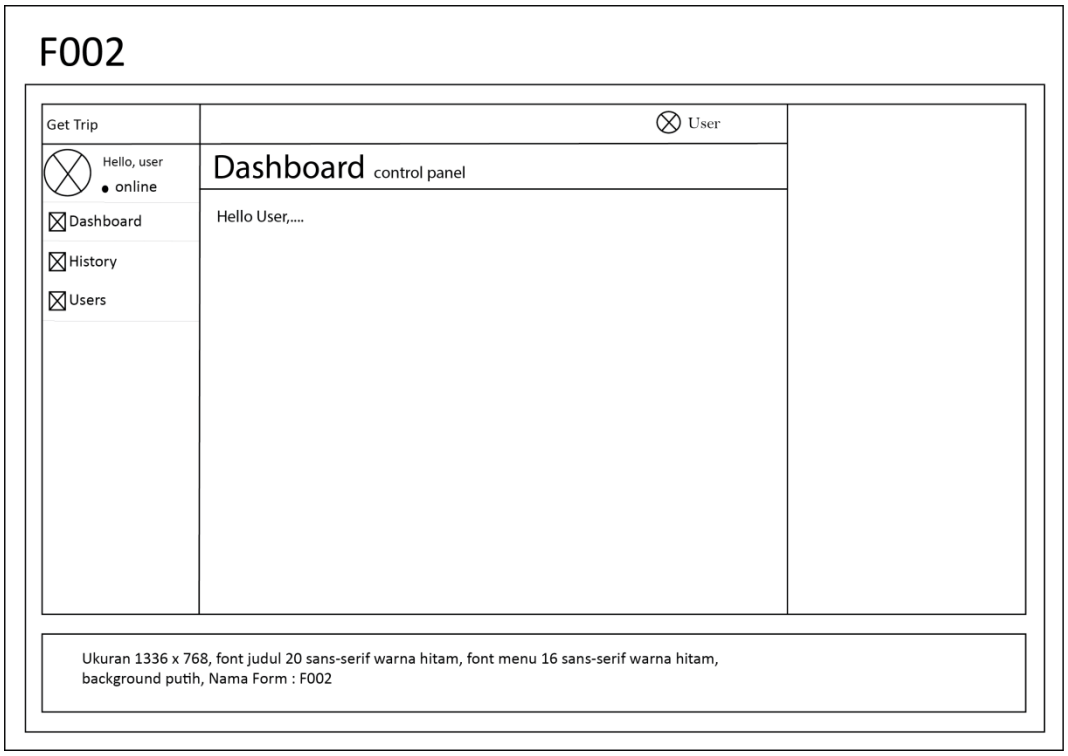
Berikut ini merupakan tampilan form login pada perangkat lunak get trip yang dapat dilihat pada Gambar 3.49 Form Login.



Gambar 3. 49 Form Login

2. Form Dashboard

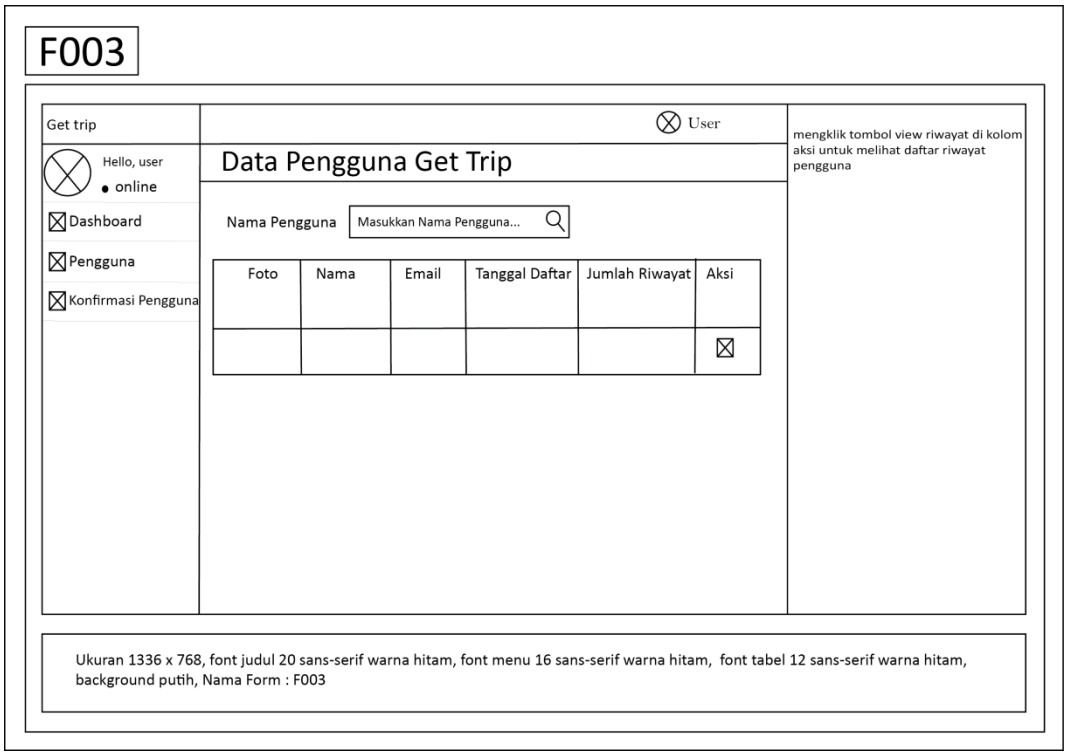
Berikut ini merupakan tampilan form Dashboard pada perangkat lunak get trip yang dapat dilihat pada Gambar 3.50 Form Dashboard



Gambar 3. 50 Form Dashboard

3. Form Pengguna

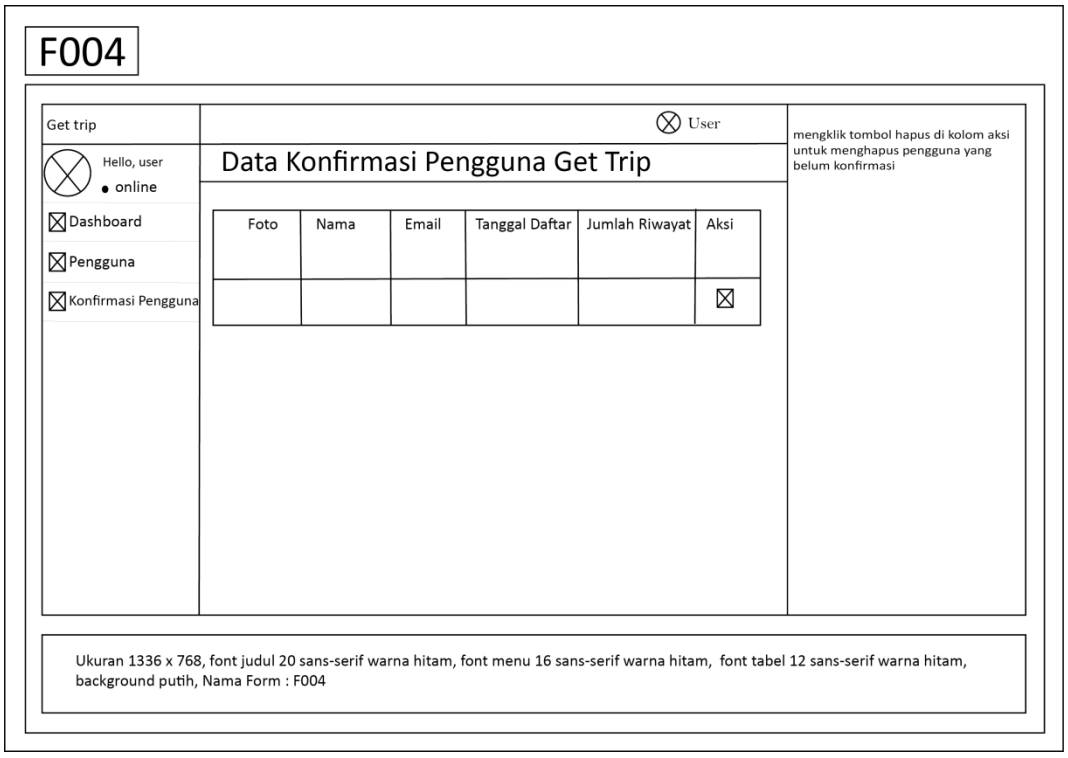
Berikut ini merupakan tampilan form Pengguna pada perangkat lunak get trip yang dapat dilihat pada Gambar 3.51 Form Pengguna.



Gambar 3. 51 Form Pengguna

4. Form Konfirmasi Pengguna

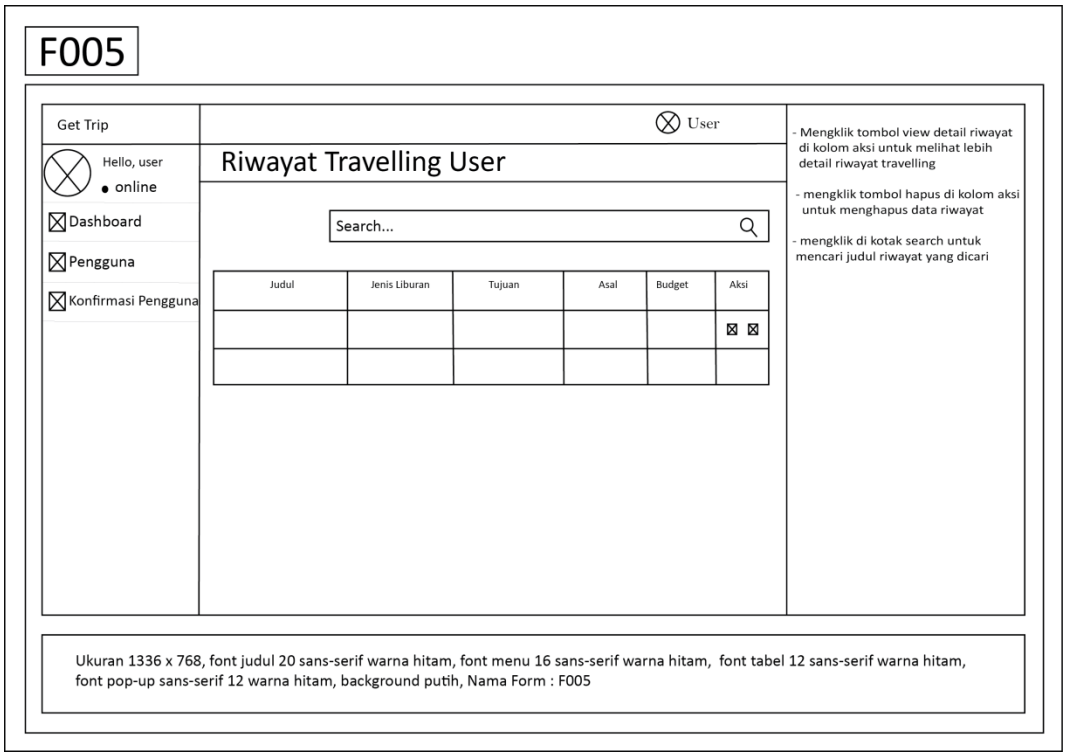
Berikut ini merupakan tampilan form konfirmasi pengguna pada perangkat lunak get trip yang dapat dilihat pada Gambar 3.52 Form Konfirmasi Pengguna.



Gambar 3. 52 Form Konfirmasi Pengguna

5. Form View History

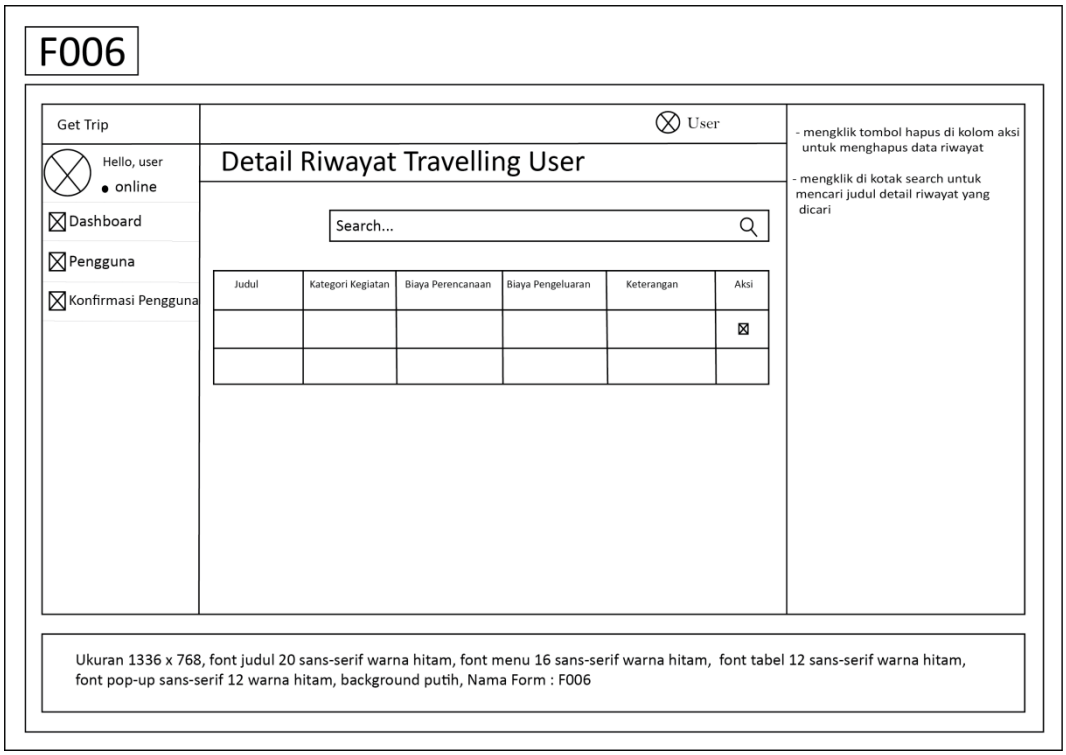
Berikut ini merupakan tampilan form view history pada perangkat lunak get trip yang dapat dilihat pada Gambar 3.53 Form View History.



Gambar 3. 53 Form View History

6. Form View Detail History

Berikut ini merupakan tampilan form view detail history pada perangkat lunak get trip yang dapat dilihat pada Gambar 3.54 Form View Detail History



Gambar 3. 54 Form View Detail History

3.2.4 Perancangan Pesan

Perancangan pesan pada perangkat lunak get trip dapat dilihat sebagai berikut.

1. Pesan teks

Pesan berikut ini merupakan pesan yang terdapat pada perangkat lunak get trip yang berbentuk teks, dapat dilihat pada tabel 3.5 Pesan yang terdapat pada aplikasi.

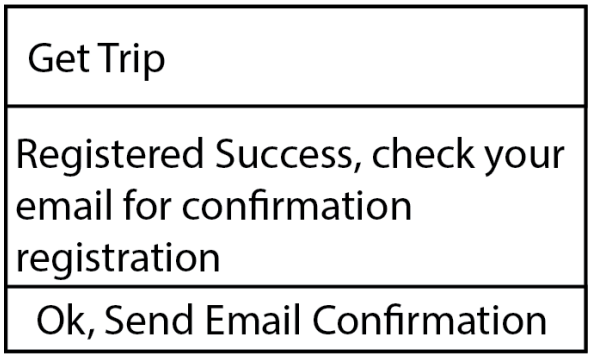
Tabel 3. 35 Pesan yang terdapat pada aplikasi

No.	Pesan	Kode	Jenis
1	Incorect email or password	M01	Teks
2	The email you entered already exist	M02	Teks
3	Your password too short, password minimum 8 character	M03	Teks
4	New password and Re-Type password is not match	M04	Teks
5	The email you entered is invalid	M05	Teks
6	History Travelling not found	M06	Teks
7	Please don’t left field empty	M07	Teks
8	Please don’t left field empty, maps and photo expense required	M08	Teks
9	Create plan success	M09	Teks
10	Create plan failed	M10	Teks
11	Add expense success	M11	Teks
12	Add expense failed	M12	Teks
13	Add history success	M13	Teks

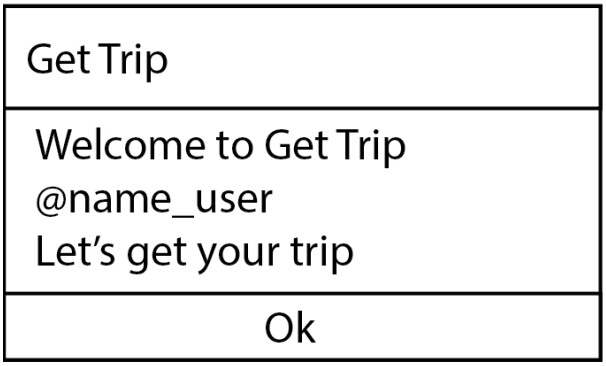
14	Add history failed	M14	Teks
15	Edit plan success	M15	Teks
16	Edit plan failed	M16	Teks
17	Edit expense success	M17	Teks
18	Edit expense failed	M18	Teks
19	Delete plan success	M19	Teks
20	Delete plan failed	M20	Teks
21	Delete expense success	M21	Teks
22	Delete expense failed	M22	Teks
23	Delete History Success	M23	Teks
24	Delete History Failed	M24	Teks
25	Sory history not found	M25	Teks
26	Your expenses greater than the cost of plannig	M26	Teks
27	Result of recommendation	M27	Teks

2. Pesan dialog

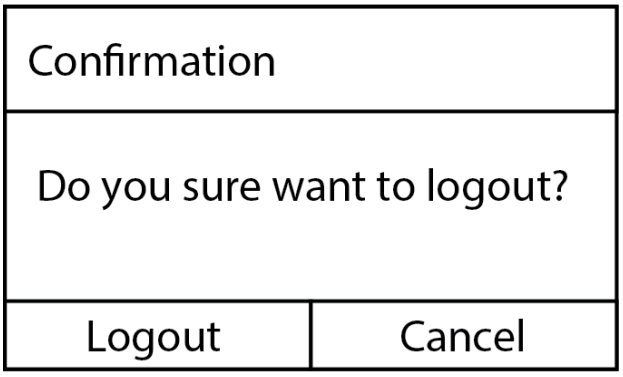
Pesan dibawah ini merupakan pesan yang berbentuk dialog yang ada pada perangkat lunak get trip, dapat dilihat pada gambar berikut ini.



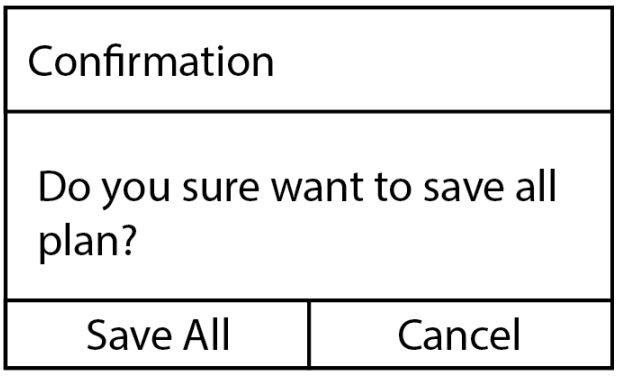
Gambar 3. 55 M28 Pesan dialog registrasi berhasil



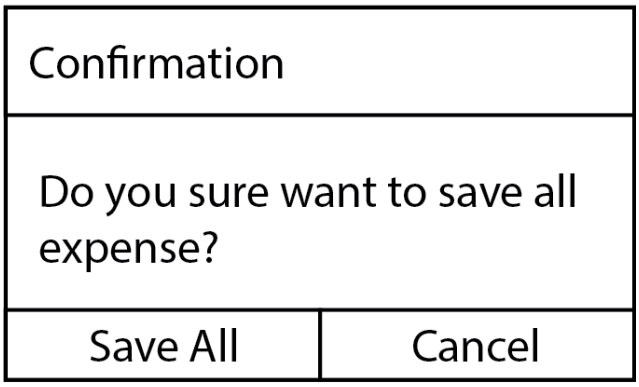
Gambar 3. 56 M29 Pesan dialog Login berhasil



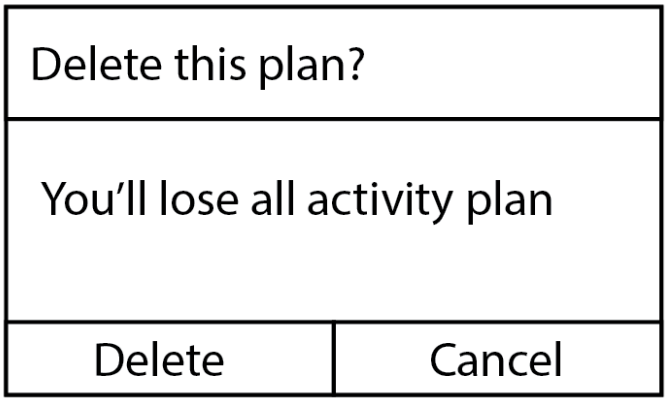
Gambar 3. 57 M30 Pesan dialog konfirmasi logout



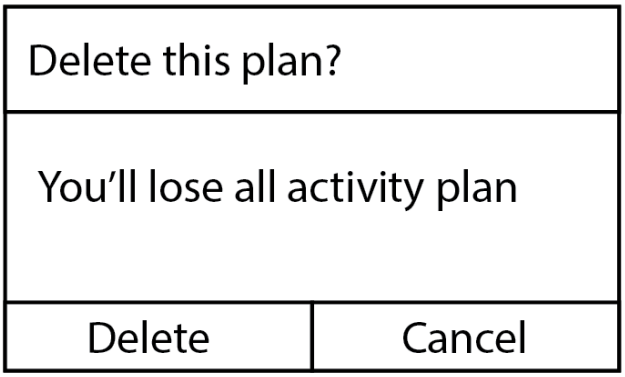
Gambar 3. 58 M31 Pesan dialog konfirmasi simpan perencanaan



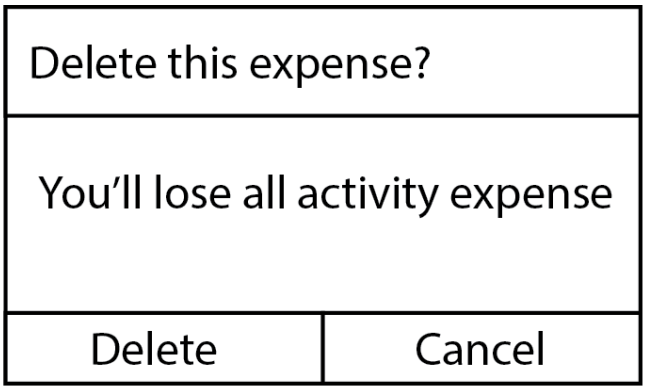
Gambar 3. 59 M32 Pesan dialog konfirmasi simpan pengeluaran



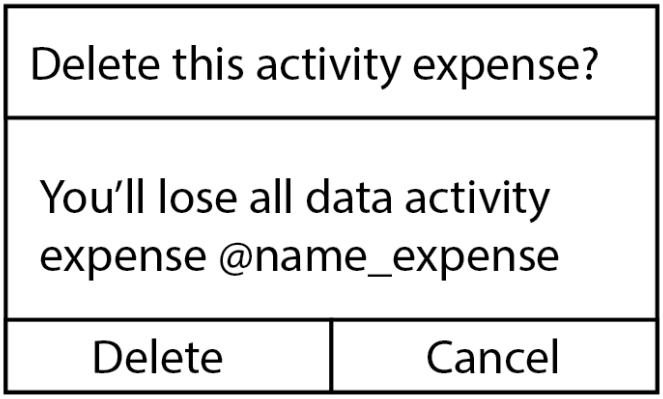
Gambar 3. 60 M33 Pesan dialog konfirmasi hapus perencanaan



Gambar 3. 61 M34 Pesan dialog konfirmasi hapus rencana kegiatan



Gambar 3. 62 M35 Pesan dialog konfirmasi hapus pengeluaran



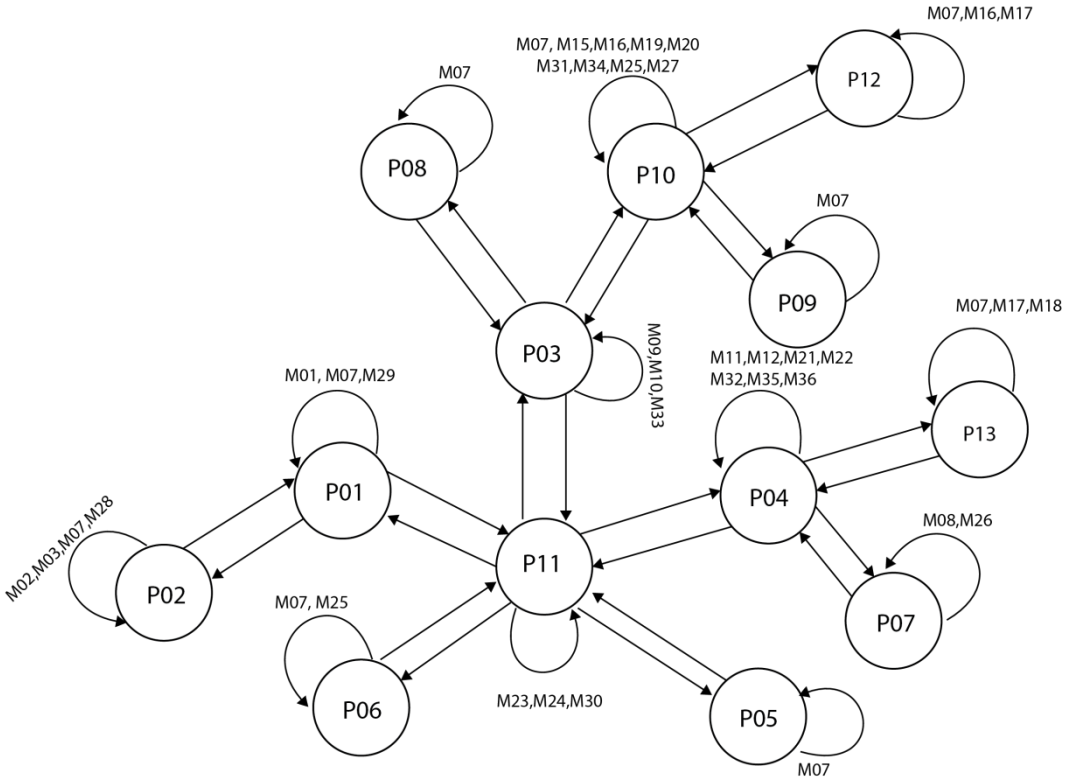
Gambar 3. 63 M36 Pesan dialog konfirmasi hapus pengeluaran kegiatan

3.2.5 Jaringan Semantik

Jaringan semantik pada perangkat lunak get trip dapat dilihat pada Gambar berikut.

1. Platform Android

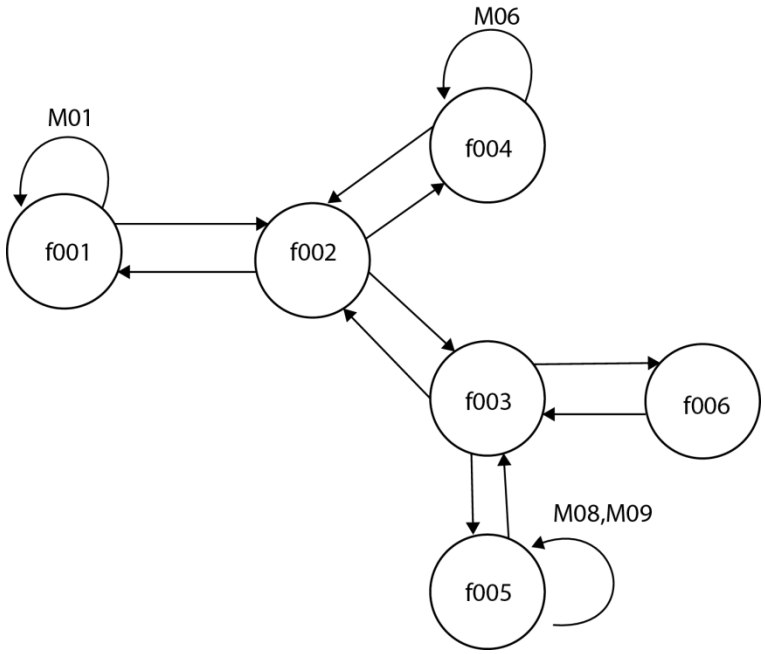
   Jaringan semantik pada platform mobile dapat dilihat pada Gambar 3.64 Jaringan Semantik Platform Mobile Perangkat Lunak Get Trip.



Gambar 3. 64 Jaringan Semantik Platform Mobile Perangkat Lunak Get Trip

2. Platform Website

Jaringan semantik pada platform website dapat dilihat pada Gambar 3. 65 Jaringan Semantik Platform Website Perangkat Lunak Get Trip.



Gambar 3. 65 Jaringan Semantik Platform Website Perangkat Lunak Get Trip