Version <1.0>

Presented by :

Project A2

Date

14 Januari 2018

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| <dd/mmm/yy> | <x.x> | <details> | <name> |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. Introduction 4

1.1 Purpose 4

1.2 Scope 4

1.3 Definitions, Acronyms and Abbreviations 4

1.4 References 4

1.5 Technologies to be used 5

1.6 Overview 5

2. Overall Description 5

2.1 Use Case Model Survey 5

2.1.1 Introduction 5

2.1.2 Actors 6

2.2 Architecture diagram & database design 7

2.2.1 Architecture diagram 7

2.2.2 Database Design 7

3. Specific Requirements 8

3.1 Use-Case Reports 8

3.2 Activity Diagram 10

3.2.1 Buat Akun 10

3.2.2 Edit Akun 11

3.2.3 Tambah Event 12

3.2.4 Hapus Event 13

3.2.5 View Event 14

3.2.6 Mencari Event 14

# Introduction

## Purpose

Tujuan dibuat dokumen ini untuk mempresentasikan descripsi detail aplikai web TravlendarA2.

Dokumen ini akan menjelaskan tentang tujuan dan fitur dari sistem, antarmuka dari sistem*,* apa yang

sistem lakukan, di mana sistem harus beroperasi dan bagaimana sistem akan bereaksi terhadap rangsangan eksternal. Dokumen ini dimaksudkan untuk *stakeholder* dam *developer* dari sistem.

## Scope

Sistem aplikasi ini berbasis web merupakan sistem aplikasi untuk membuat *schedule* kegiatan.

Sistem aplikasi dirancangan untuk membantu *schedule* kegiatan dan perjalanan secara efektif. Nama pengguna untuk aplikasi ini adalah *Traveller. Traveller* Akan menambahkan event, waktu, dan lokasi untuk memulai membuat *schedule* kegiatan maka sistem otomatis akan membuat rekomendasi keberangkatan agar kegiatan dan perjalanan tidak terganggu meskipun padatnya kegiatan yang dimilkii oleh seseorang. Sistem aplikasi ini menggunakan *framework google maps API* untuk membantu dalam berjalannya sistem aplikasi ini. *Traveller* juga bisa melakukan hapus, edit atau *update* *event* dari *Traveller* itu sendiri. Dibantu dengan *google calendar API* , untuk membantu dalam *reminder* tentang *schedule* kegiatan.

### Fasilitas Login dan Registrasi untuk siapapun untuk menjadi Traveller

### Fasilitas edit account oleh Traveller dengan mengubah data Traveller

### Fasilitas menambah event dengan menginput nama event, memilih lokasi event, lokasi Traveller,

### Fasilitas mengunduh list event ke versi pdf

### Fasililtas menghapus dan mengubah data event yang sudah diinputkan Traveller

## Definitions, Acronyms and Abbreviations

|  |  |
| --- | --- |
| **Term** | **Definition** |
| *Traveller* | Sebutan untuk pengguna aplikasi TravlendarA2 |
| *Database* | Koleksi dari semua data informasi yang di simpan di dalam sistem |
| Software Requirements Specification | Sebuah dokumen yang mendeskripsikan semua tentang fungsi dari yang sudah disepakati , dan batasaan tentang mengoprekasikan sistem |
| Stakeholder | Seseorang yang ikut andil pada pembuatan apliaksi tapi bukan developer |
| Developer | Developer adalah seorang programmer yang sudah terlatih, mereka tidak hanya menyelesaikan masalah, namun juga membangun suatu produk. |

## References

?

## Technologies to be used

* Bahasa Pemrogaman Java
* Apache tomcat 8.5 :  open source web server yang di kembangkan oleh Apache Sofware Fondation (ASF). mengimplementasikan Java Servlet dan juga Java Server Page (JSP) spesifikasi dari Oracle Corporation, dan menyediakan java HTTP web server yang bisa dijalankan dengan kode java.
* Full Calendar JQUERY : kalendar *event online*
* Google Maps API : layanan pemetaan web yang dikembangkan oleh Google

## Overview

Pada bab berikutnya adalah bagian uraian keseluruhan, dari dokumen ini memberikan gambaran tentang fungsi produk. Menggambarkan tentang *requirement informal* dan dipakai untuk konteks teknis *requirement specification* ada di bab selanjutnya.

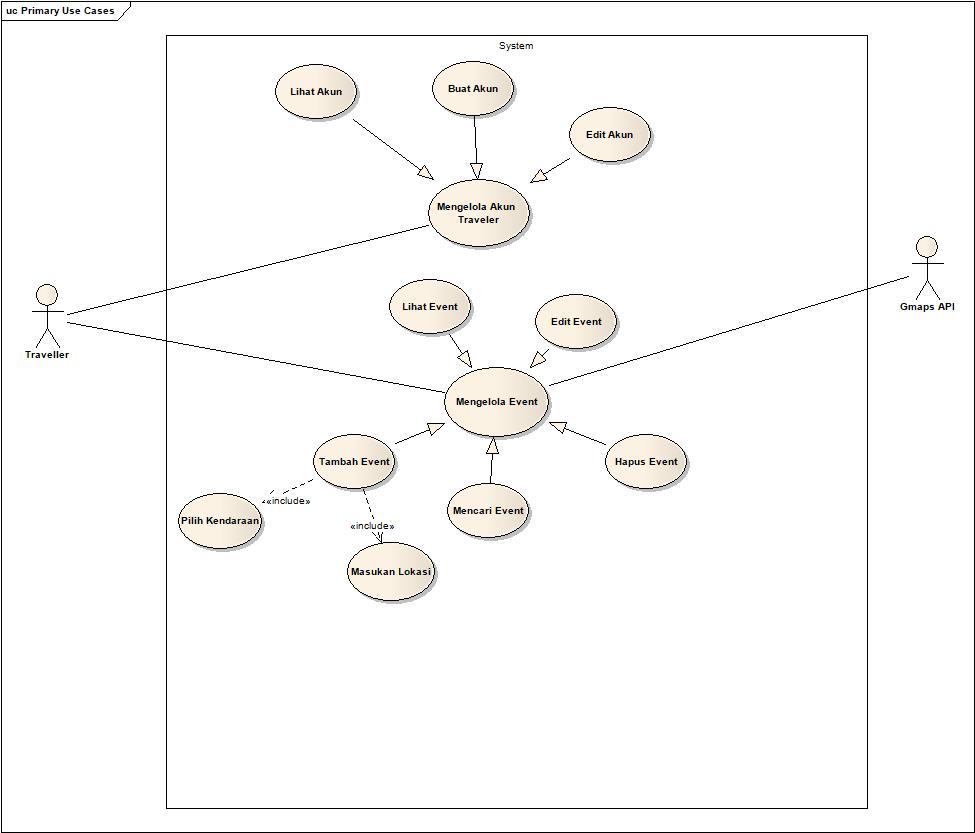
Bab 3 membahas tentang, menjeleskan tentang *requirement specification,* dokumen ini mendeskripsikan teknis dari detail fungsi aplikasi.

# Overall Description

## Use Case Model Survey

### Introduction

< Use Case Model Package Documentation>



Gambar 2.1 Use Case Model TravlendarA2

### Actors

|  |  |
| --- | --- |
| Nama Perangkat | Deskripsi |
| Nama | Traveller |
| Deskripsi singkat | Sebutan pengguna Aplikasi TravlendarA2 |
| Karakteristik | Traveller merupakah actor manusia harus memiliki pengetahuan tentang internet untuk bisa mengakses website Travlendar, aplikasi yang bisa digunakan oleh actor untuk mengakses adalah web browser dan jaringan internet. Jenis kelamin perempuan atau laki-laki, siapapun bisa memakainya. |
| Hubungan | Hubungan dari actor Traveller ke system dan interface gmaps API untuk mengambil fungsi fungsi yang dibutuhkan di aplikasi |

Table 2.1 deskripsi *Actor*

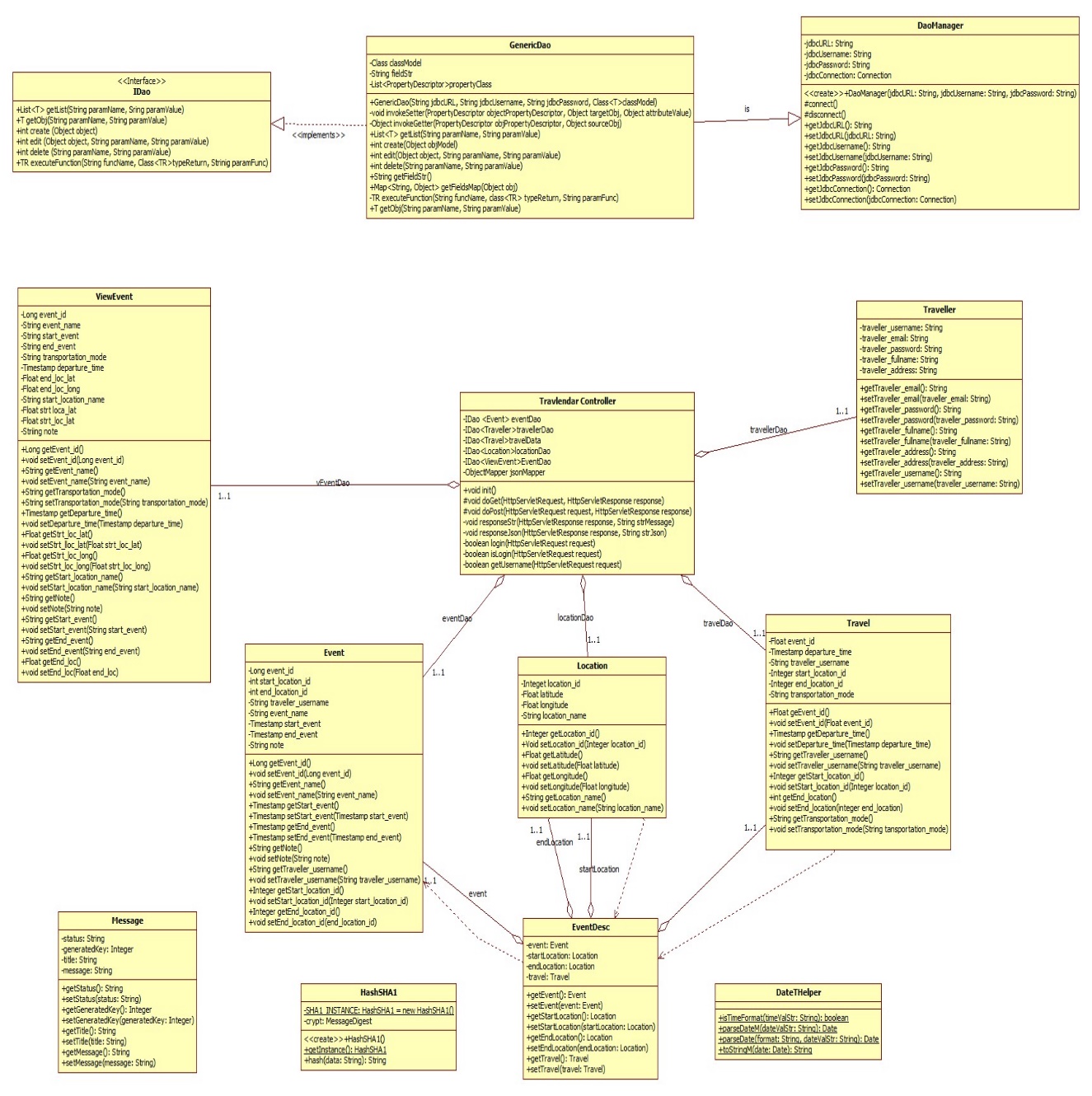
### 

## Architecture diagram & database design

### Architecture diagram

?

### Database Design

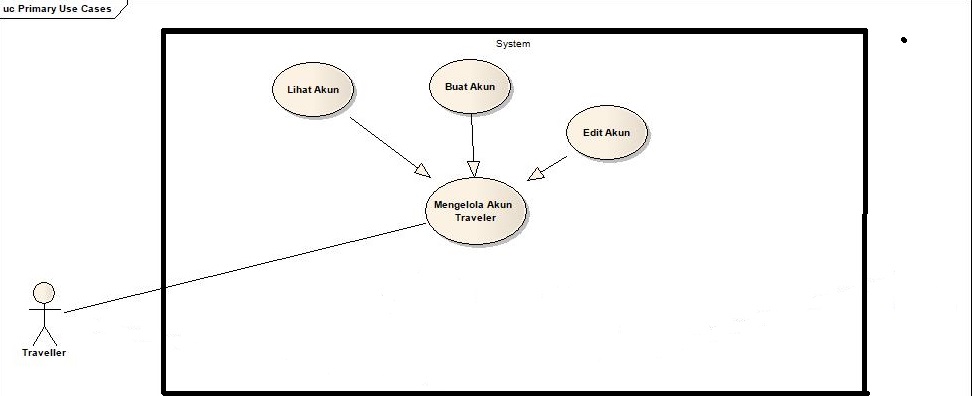


[Mention your application architecture diagram and database design schema here]

Gambar 2.3 *Class diagram*

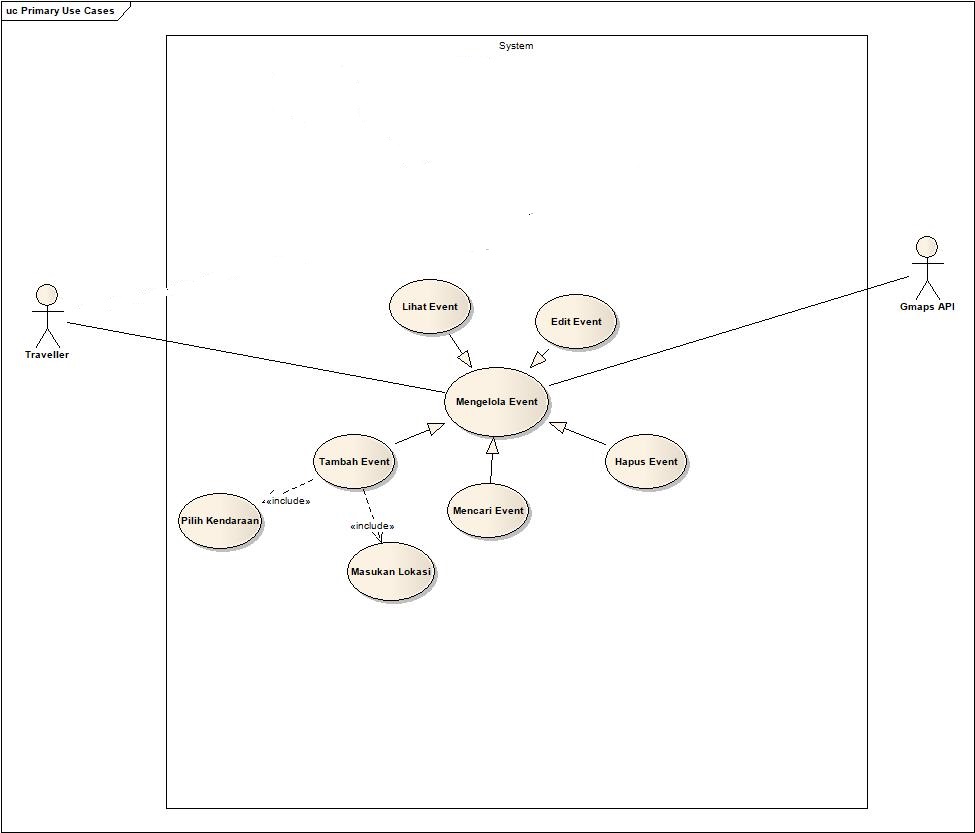
# Specific Requirements

## Use-Case Reports



Gambar 3.1 Use Case diagram Mengelola akun Traveller

|  |  |
| --- | --- |
| Use Case | Deskrpisi |
| Lihat Akun | Traveller bisa melihat informasi akun dari data yang sebelumnya diinput oleh Traveller |
| Buat Akun | Traveller membuat akun dengan menginput informasi nama lengkap, *username*, *password*, *email,* dan *home address* agar bisa menggunakan sistem aplikasi |
| Edit Akun | Traveller bisa mengedit informasi akun yang sebelumnya sudah diinput |



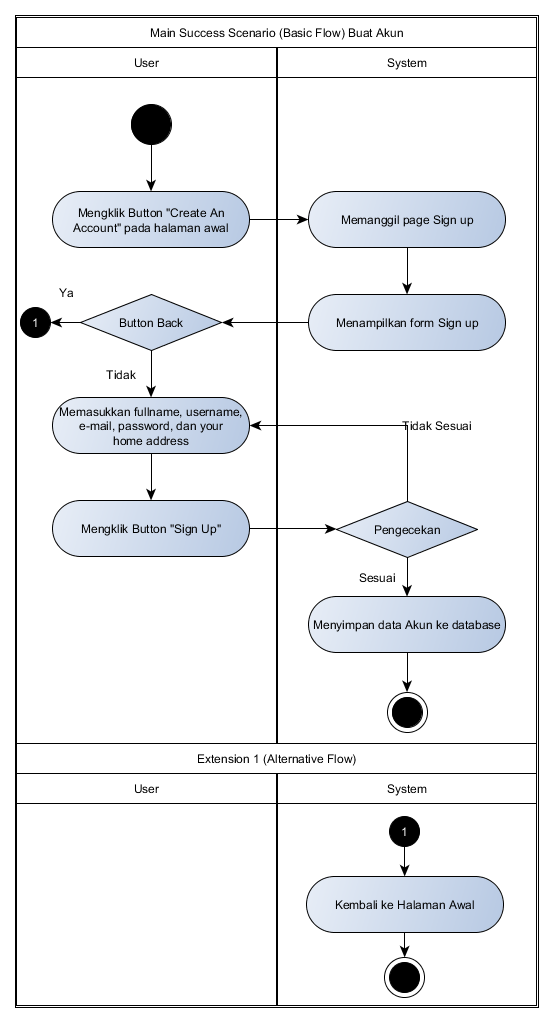
Gambar 3.2 Use Case Diagram Manage Event

|  |  |
| --- | --- |
| Use Case | Deskrpisi |
| Tambah *Event* | Traveller menambah *event* baru dengan meningput informasi nama event , lokasi, waktu *event*,waktu keberangkatan dan pilih kendaraan |
| Masukan Lokasi | Traveller memasukan lokasi *event* dan lokasi traveller |
| Pilih Kendaraan | Traveller setelah meningput informasi *event*, lokasi , dan waktu maka tampil pilih kendaraan yang akan di gunakan |
| Lihat *Event* | Traveller bisa melihat *event* yang sudah diinputkan sebelumnya |
| Mencari *Event* | Traveller bisa mencari data *event* yang sebelumnya sudah diinputkan |
| Hapus *Event* | Traveller bisa mengahapus *event* yang sudah diinputkan |
| Edit *Event* | Traveller bisa mengedit *event* yang sudah diinputkan sebelumnya |

## Activity Diagram

### Buat Akun

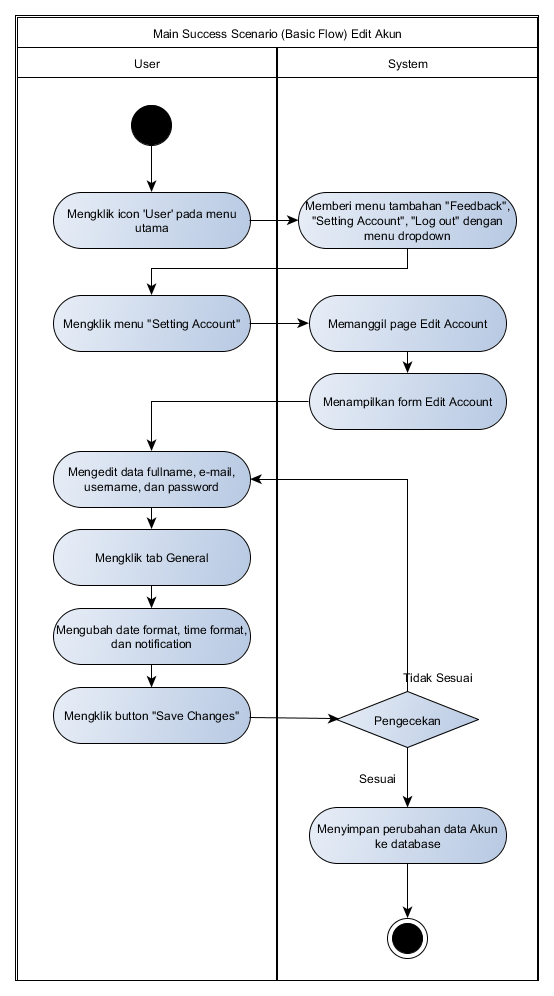
Traveller yang belum memliki akun melakukan registrasi atau buat akun, Traveller akan menginput data di form, data yang harus diinput Traveller adalah *fullname, email, password,* dan *your home address,* setelah itu akan dilakukan pengecekan kesesuaian data, setelah itu klik tombol *signup*.



Gambar 3.3 Activity Diagram Buat Akun

### Edit Akun

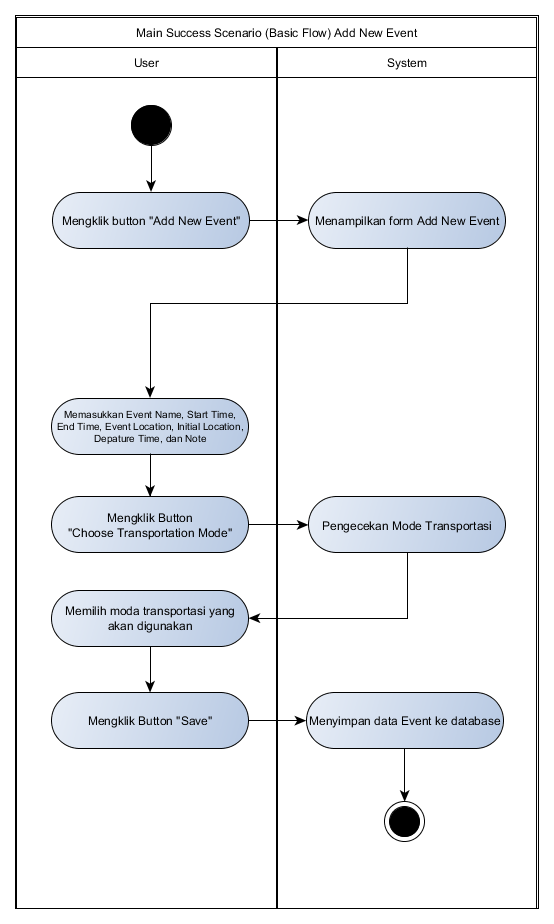
Traveller yang sudah memiliki akun mengubah data informasinya dari mulai mengedit nama lengkap, *email*, *username* dan *password.* Traveller akan mengubah data nya di form yang sudah terisi data yang sudah diinputkan. Setelah diubah maka dan mengklik *save changes.*



Gambar 3.4 Activity Diagram Buat Akun

### Tambah Event

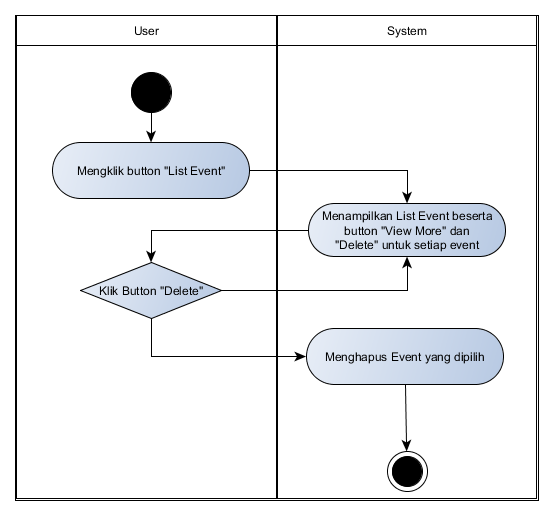
Traveller yang sudah memiliki akun, bisa menambah *event*  dengan mengisi form *add new event* dan mengisi data nama *event*, lokasi, waktu, dan pemilihan moda transportasi. Pada saat pemilihan lokasi dan moda transportasi menggunakan *gooogle maps API* sebagai framework nya.



Gambar 3.5 *Activity Diagram Add New Event*

### Hapus Event

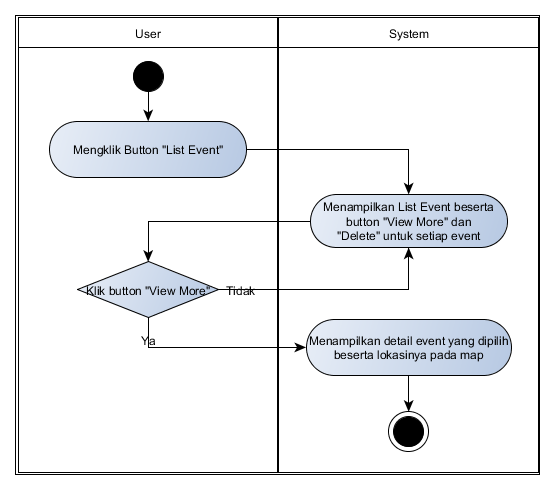
Traveller yang sudah menginput data *event ,* bisa menghapus *event* yang sebelumnya sudah diinputkan.



Gambar 3.6 *Activity Diagram* hapus *Event*

### View Event

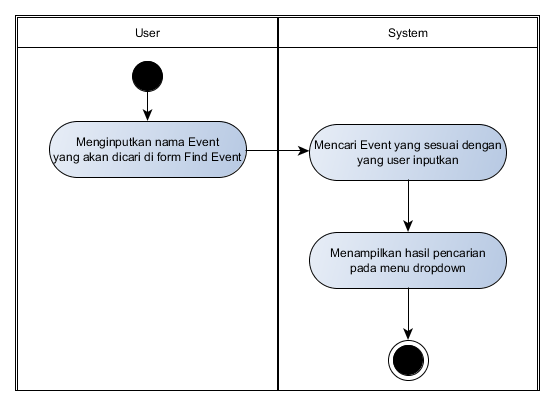
Traveller bisa melihat *event* yang sudah dimasukan sebelumnya berupa list *event*  dan untuk melihat detail informasi *event* klik *view more.*



Gambar 3.6 *Activity diagram view event*

### Mencari Event

Traveller sudah menginput data event, bisa mencari event yang pernah diinputkan sebelumnya dengan mengisi nama event di form *find event* lalu muncul hasil pencarian pada menu *dropdown.*



Gambar 3.6 *Activity Diagram search event*