|  |  |
| --- | --- |
| Name | CWID |
| Fatema Alteneiji | A20362529 |
| Moosa Hana | A20298732 |
| Saurabh Tiwari | A20356308 |
| Ernesto Garcia | A20364089 |
| Chethan Appaji | A20355296 |

Tourist Assist Application

CS487: Software Engineering

Table of contents

User Analysis 3

Requirements Specification 5

High-Level Designs 7

Test Plans 10

# User Analysis

A research has been done on what possible users might want, in a tourist application, a questionnaire has been given to a number of people in defferent age groups and different languages, this questionnaire helped the team member gain more understanding of what a tourist application might need to provide to its users, in order for it to be useful to them.

The purpose of that questionnaire was to answer these questions:

* **Who are the users?**

The users of this application are people who are visiting a new place, or planning to visit a new one.

* **What do the users want?**

The tourists mostly want an application that has a simple and efficient interface, one that they can understand and easily use. They want the application to be fast, in case they wanted to used while in a hurry of wanting find a new place to visit at that time, and they would also want the application to support different languages other than English such as French and Spanish, for not all users are fluent or even good with English.

* **What are the users’ goals?**

The tourists’ goals are to have an application that can help them with finding hotels, restaurants, the theatres, and the nightlife in the city. One that also shows them the famous attraction in the city, and the hospitals in case they needed one. They want to be able to read reviews on these places to see what other people think about them.

* **What are individual characteristics that may affect behavior with software or information designed?**

One of the main characteristics that could effect the software is the language of the tourists, the application may be built to accommodate those different languages, another one is the fact that many of them tend to want something to be always available and offers data fast, so the application will be built to accommodate those needs.

* **What do they know that helps them perform their tasks?**

From what the members noticed most of the people these days have at least basic knowledge at using web applications and their mobile phones, and as long as the user interface is built to be simple, most of the tourists should be able to use it.

* **Do they want a user interface that is fun, not boring?**

Many people have answered this question with a yes, but with a request of it being simple and not over cluttered

# Requirements Specification

From the user analysis that has been done by the team, a set of requirments has been deduced. The Tourist Assist application will be built on a set of functional and non-functional requirements:

**Functional requirements:**

* This application shall be able to find the location of:
  + Hotels
  + Famous Restaurants
  + Museums
  + Shopping malls
  + Hospitals
  + Bars Pubs or Clubs
  + Movies and Theatrical Shows
* This application shall behave in a way such that it can find the location and pin point the coordinates accordingly.
* This application shall have a search functionality that allows to filter the places by price, category, rating and popularity.

This application shall allow the user to view, write reviews and rate the places that have been visited.

* This application should be able to list current events in current city or the one to be visited.
* This application should show transit points
* This application should offer social network connection that the user may use to share his experience.
* This application should support multiple languages: English, French and Spanish.
* This application should memorize the places that have been already visited by the user.

**Non-Functional requirements:**

* The applications’ user interface should be designed by either bootstrap or any other available sources that allows user interface design.
* The overall architecture that the team will be following for example: Host server APIs, Database, UI server and design.

# High-Level Designs

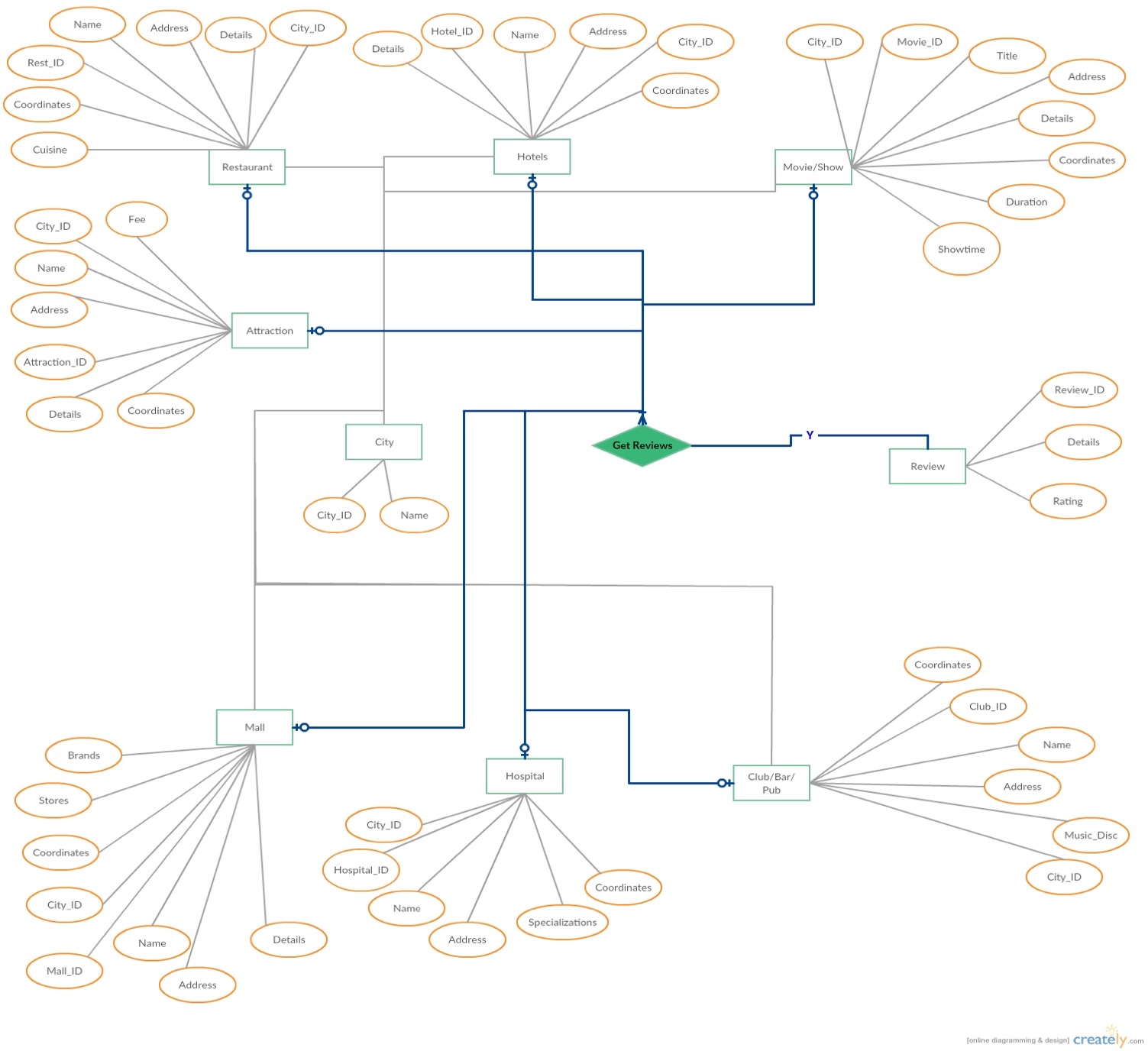
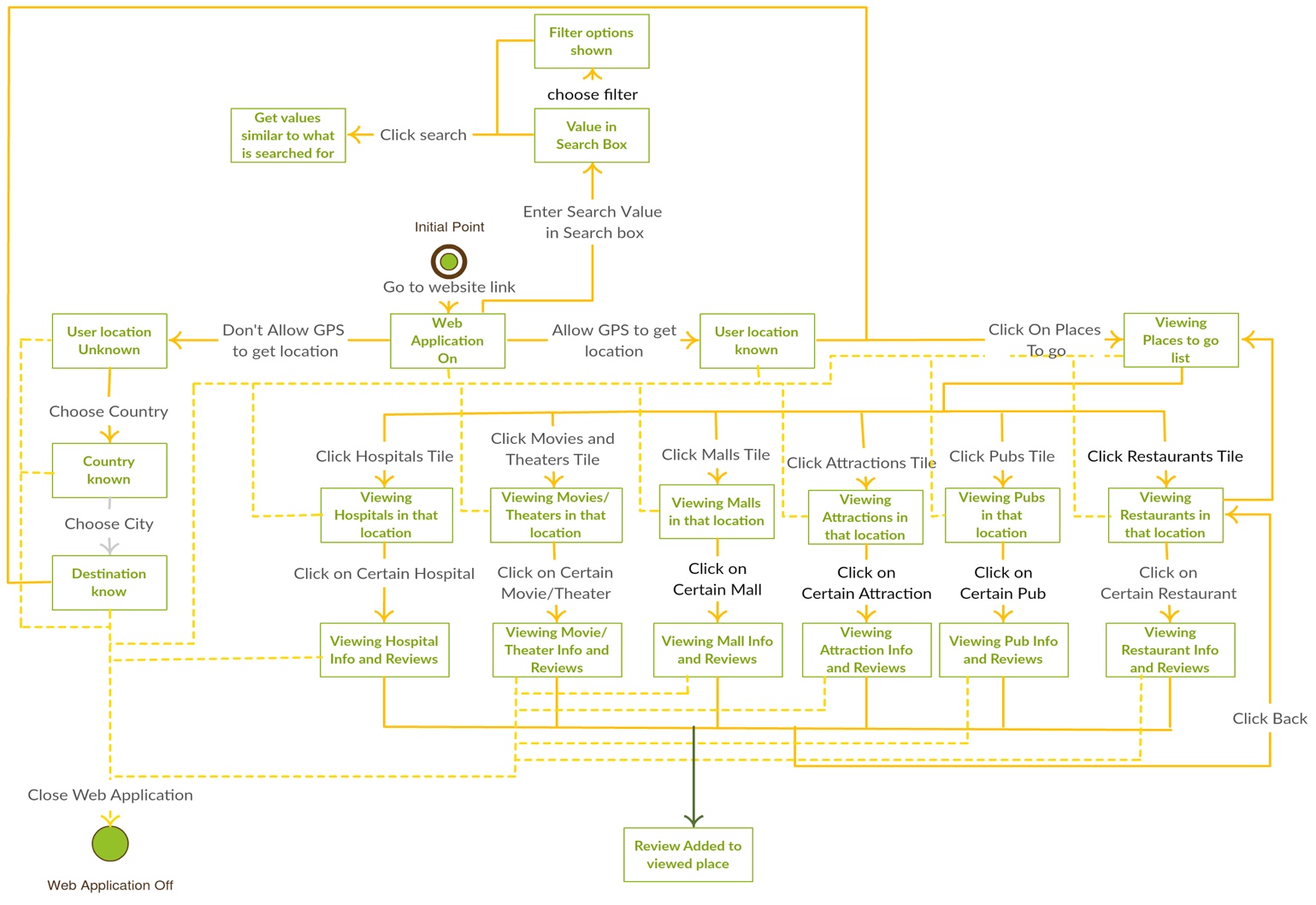
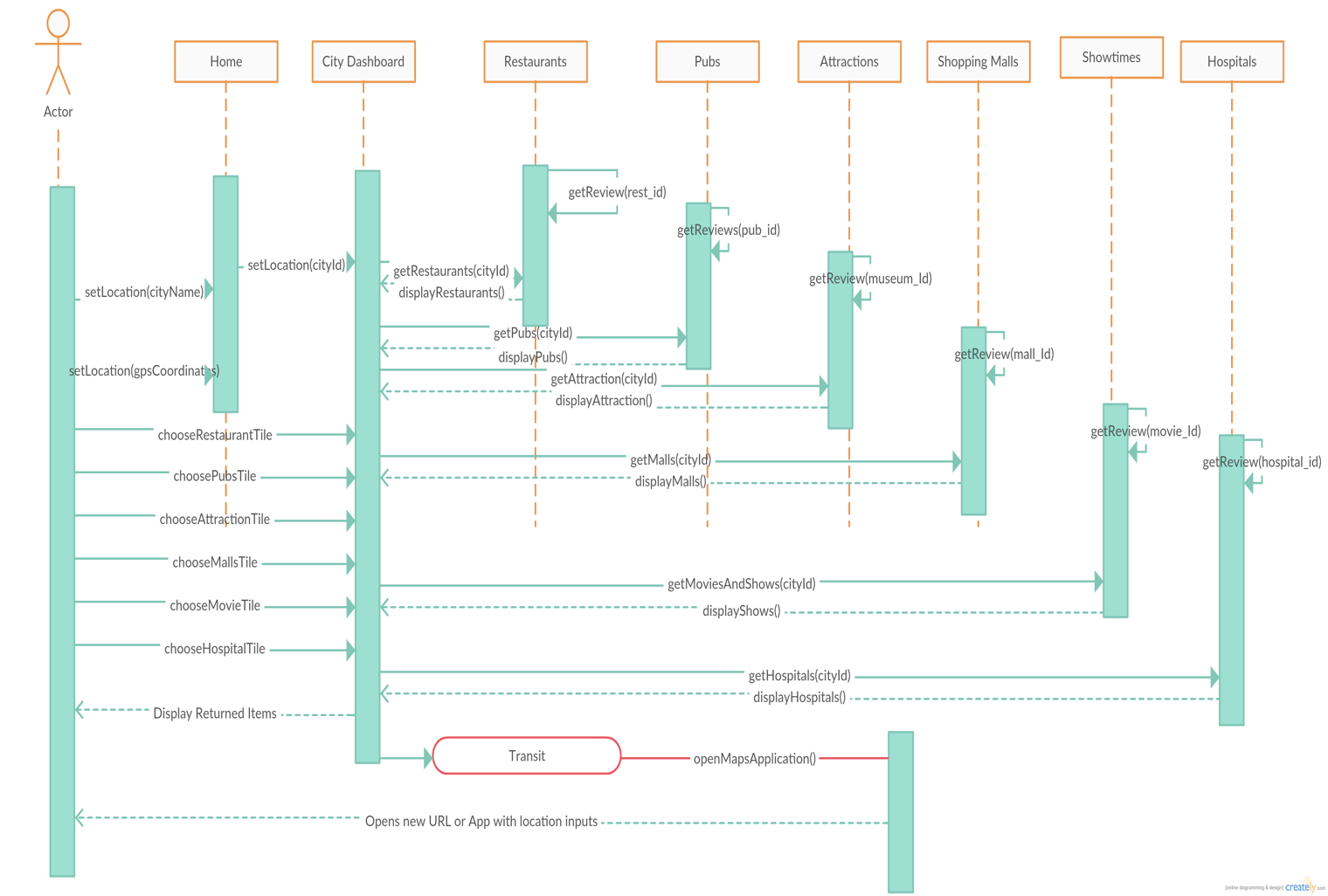
Once the decision on what the requirements of this application should be has been made, the team started to build the databases that the application will need to meet the users wants in a tourist application. This Entity Relation Diagram, shows the main tables in the database, the fields they contain and their connections to the other tables. After setting up the database and the required tables a state transition diagram has been built to clearly state how the application will act, to meet the stated requirements.

Figure 1: Entity Relationship Diagram of the databases the application will be using

Figure : State Transition Diagram for the Tourist Assist Application

Once the state transition diagram was drawn, a sequence diagram depicting the user interaction with the application and how the data flows through the system upon the user’s interaction with it.



# Test Plans

Software Testing is an investigation conducted to provide stakeholders with information about the quality of a particular product or service under test. In other words, software testing is a process of verification and validation. The success of testing depends on a well written test plan document. This document describes the objectives, scope, approach and focus of the software testing effort. A test plan serves as the means of communication with other members of the project team, testers, peers, managers and other stakeholders. It helps us to manage change effectively.

A test plan document contains information about the following:

* Scope of testing (Test scenarios/test cases to be validated).
* Schedule for Test scenario preparation, test documentation, test execution, start and end dates of test cycles etc.
* Roles and responsibilities of team members.
* Deliverables at each time frame (Item pass/fail criteria, Risks and contingencies).
* Test environment and tools to be used.
* Defect management and exit criteria.

**Traceability:**

It is used to track the requirements and to check if the project requirements are met by the prepared test cases. We have to relate the test cases to requirements as a many-to-many relationship and check the completeness of the relationship.

Requirement Traceability Matrix (RTM) captures all requirements proposed by the client or development team and their traceability in a single document delivered at the conclusion of the life-cycle. In other words, it is a document that maps and traces user requirement with test cases. The main purpose of RTM is to see that all test cases are covered so that no functionality should miss while testing.

Requirement Traceability Matrix – Parameters include

* Requirement ID
* Risks
* Requirement Type and Description
* Trace to design specification
* Unit test cases
* Integration test cases
* System test cases
* User acceptance test cases
* Trace to test script

Types of Traceability Matrix

* **Forward traceability**: This matrix is used to check whether the project progresses in the desired direction and for the right product. It makes sure that each requirement is applied to the product and that each requirement is tested thoroughly. **It maps requirements to test cases**.
* **Backward or reverse traceability:** It is used to ensure whether the current product remains on the right track. The purpose behind this type of traceability is to verify that we are not expanding the scope of the project by adding code, design elements, test or other work that is not specified in the requirements. **It maps test cases to requirements**
* **Bi-directional traceability (Forward+Backward):**This traceability metrics ensures that all requirements are covered by test cases. It analyzes the impact of a change in requirements affected by the defect in a work product and vice versa.

Advantage of Requirement Traceability Matrix

* It confirms 100% test coverage.
* It highlights any requirements missing or document inconsistencies.
* It shows the overall defects or execution status with a focus on business requirements.
* It helps in analyzing or estimating the impact on the QA team's work with respect to revisiting or re-working on the test cases.

**Scenarios and Use Cases:**

**Test Scenario**: A Scenario is any functionality that can be tested**.**It is also called**Test Condition or Test Possibility**. It is a high level description of test requirement group based on the functionality of the module. Test scenarios are derived from use cases. A test scenario can have multiple test cases associated with it. A test case is a detailed description of the test requirement.

A test scenario includes the following:

* Scenario ID
* Use case/Requirement ID
* Scenario Description
* No. of test cases

A test case consists of the following:

* Scenario ID
* Test case ID
* Test case description
* Test step no.
* Test step description
* Expected result
* Actual result

**Use Case:** A use case can be thought of as a collection of possible scenarios related to a particular goal. Use cases can be employed during several stages of software development, such as planning system requirements, validating design, testing software, and creating an outline for online help and user manuals.

A use case (or set of use cases) has these characteristics:

* Organizes functional requirements.
* Models the goals of system/actor (user) interactions.
* Records paths (called *scenarios*) from trigger events to goals.
* Describes one main flow of events (also called a basic course of action), and possibly other ones, called *exceptional* flows of events (also called alternate courses of action).
* Is multi-level, so that one use case can use the functionality of another one.

The followings are some of the scenarios and use cases for different modules of Tourist Assist Application:

Use case ID: 1

Use case Name: Look up for nearby Restaurants and check reviews

Actors: User (tourist)

Description: Test whether user is able to look up for a restaurant nearby current location and get reviews on the selected restaurant.

Precondition: User should have enabled the location services and set the city name for location

Post condition: User should select/click on restaurants and get review option

Normal flow:

1. Open the application.
2. Enable the location services and set location.
3. Look up for restaurants option.
4. Select the restaurant.
5. Click on get review.
6. Close the application.

Exceptions:

1. User is unable to set the location.
2. User is unable to get the nearby restaurants.
3. User is unable to get review.

Use case ID: 2

Use case Name: Look up for nearby Pubs and check reviews

Actors: User (tourist)

Description: Test whether user is able to look up for a pub nearby current location and get reviews on the selected pub.

Precondition: User should have enabled the location services and set the city name for location

Post condition: User should select/click on pubs and get review option

Normal flow:

1. Open the application.
2. Enable the location services and set location.
3. Look up for pubs option.
4. Select the pub.
5. Click on get review.
6. Close the application.

Exceptions:

1. User not able to set the location.
2. User not able to get the nearby pubs.
3. User not able to get review.

Use case ID: 3

Use case Name: Look up for nearby Attractions and check reviews

Actors: User (tourist)

Description: Test whether user is able to look up for attractions nearby current location and get reviews on the selected attraction/museum.

Precondition: User should have enabled the location services and set the city name for location

Post condition: User should select/click on attractions and get review option

Normal flow:

1. Open the application.
2. Enable the location services and set location.
3. Look up for attractions option.
4. Select the attraction/museum.
5. Click on get review.
6. Close the application.

Exceptions:

1. User not able to set the location.
2. User not able to get the nearby attractions.
3. User not able to get review.

Use case ID: 4

Use case Name: Look up for nearby Shopping Malls and check reviews

Actors: User (tourist)

Description: Test whether user is able to look up for shopping malls nearby current location and get reviews on the selected mall.

Precondition: User should have enabled the location services and set the city name for location

Post condition: User should select/click on shopping malls and get review option

Normal flow:

1. Open the application.
2. Enable the location services and set location.
3. Look up for shopping malls option.
4. Select the mall.
5. Click on get review.
6. Close the application.

Exceptions:

1. User not able to set the location.
2. User not able to get the nearby shopping malls.
3. User not able to get review.

Use case ID: 5

Use case Name: Look up for Showtime of a movie nearby and check review of the movie.

Actors: User (tourist)

Description: Test whether user is able to look up for a showtime nearby current location and get reviews on the selected movie.

Precondition: User should have enabled the location services and set the city name for location

Post condition: User should select/click on showtime and get review option

Normal flow:

1. Open the application.
2. Enable the location services and set location.
3. Look up for showtime option.
4. Select the movie.
5. Click on get review.
6. Close the application.

Exceptions:

1. User not able to set the location.
2. User not able to get the nearby showtime.
3. User not able to get review.

Use case ID: 6

Use case Name: Look up for nearby Hospitals and check reviews

Actors: User (tourist)

Description: Test whether user is able to look up for a hospital nearby current location and get reviews on the selected hospital.

Precondition: User should have enabled the location services and set the city name for location

Post condition: User should select/click on hospitals and get review option

Normal flow:

1. Open the application.
2. Enable the location services and set location.
3. Look up for hospitals option.
4. Select the hospital.
5. Click on get review.
6. Close the application.

Exceptions:

1. User not able to set the location.
2. User not able to get the nearby hospitals.
3. User not able to get review.

Use case ID: 7

Use case Name: Look up for Transit availability

Actors: User (tourist)

Description: Test whether user is able to look up for a transit nearby current location.

Precondition: User should have enabled the location services and set the city name for location.

Post condition: User should select/click on transit and input the name of the destination.

Normal flow:

1. Open the application.
2. Enable the location services and set location.
3. Look up for transit option and click on it.
4. Opens Maps Application (opens new URL or app with location inputs).
5. Enter the destination.
6. Close the application.

Exceptions:

1. User not able to set the location.
2. Maps application is not loaded.
3. User unable to enter location inputs.

**Requirements Use-case diagrams:**

* This application shall be able to find the location of:
  + Hotels
  + Famous Restaurants
  + Museums
  + Shopping malls
  + Hospitals
  + Bars Pubs or Clubs
  + Movies and Theatrical Shows

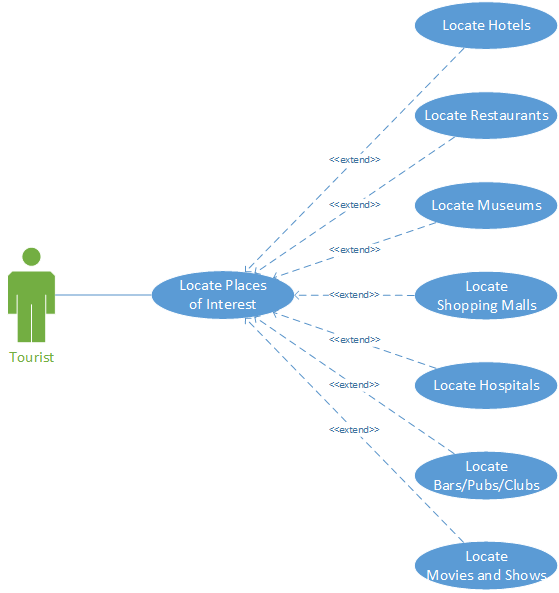


Figure : Use-case Diagram showing how the application would normally work when the user interacts with it

* This application shall behave in a way such that it can find the location and pin point the coordinates accordingly.

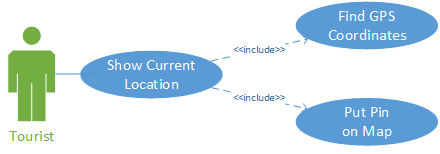


Figure : Use-case Diagram show how the application would normally work with the user's location

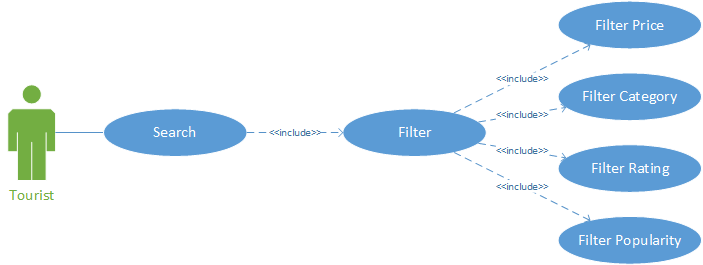
* This application shall have a search functionality that allows to filter the places by price, category, rating and popularity.

Figure : Use-case Diagram showing how the user would normally interact with the search function

* This application shall allow the user to view, write reviews and rate the places that have been visited.

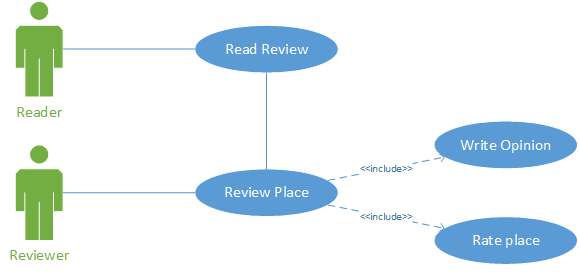


Figure 6: Use: Case Diagram showing how the application the user can normally interact with reviews

* This application should be able to list current events in current city or the one to be visited.

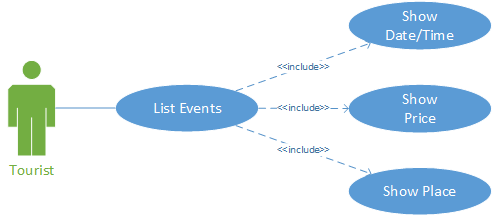


Figure : Use-case Diagram showing how the user interaction with the application would normally be to view event

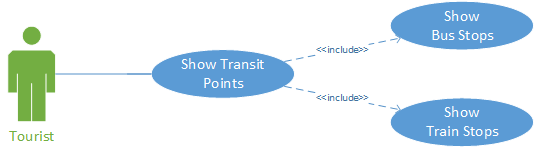
* This application should show transit points

Figure :Use case Diagram showing how the user would normally view Transit points

* This application should offer social network connection that the user may use to share his experience.

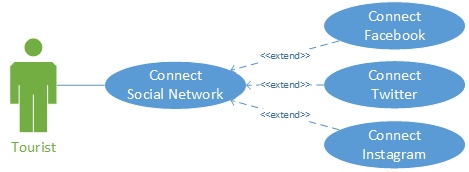


Figure : This application showing how the connection to the social network would be

**Usability:**

Usability is the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use. Usability assesses how easy the user interface is to use. Usability is performed by all different user groups. It is used to identify any usability problems from the user’s point of view.

In the tourist assist app, user groups are

* Tourist – of different age group, and different native languages

Usability testing will give you a better understanding of how your users think and how easy it is for the users to perform the task after encountering the design for the first time. A satisfied user achieves their goal, enjoys their experience, tell others and comes back again. Usability testing will tell you where your application has opportunities for improving all of above aspects, and in the process achieve a high return. Usability testing of the application has to be performed by users from all the user groups.