

Requirement Definitions

**By Benson Gao, Nick McDaniel,
Michael Ly, and Julian Tran**

Introduction

Our mobile application will let users around the world interact with each other. Users will mark locations they have been to or are familiar with. Other users can see these pinpointed locations and a list of users who has been there/familiar with. They may contact these people to learn more about the place.

What sets this application apart from other platforms like Yelp or Tripadvisor is the ability for users to receive more personal recommendations based on interests. For example, a user may want to visit a city that is not very well known or not considered a typical tourist location. The user may want to know about local events or relaxing/sightseeing places, but other platforms may not have information on this, especially if the locations the user are interested in are less known. This application can also benefit users interested in tourist locations with the ability to learn about different routes and shortcuts to take in the area at peak times, safe and dangerous areas, hidden gems catered to the user's interest (such as a restaurant that may not have as much exposure), and many more depending on how the users intend to use it.

System Model

The system needs to have an internet connection in order to properly communicate with the server and other users. The system also needs to utilize the device's GPS capabilities to locate them. The system should be able to interface with Google Maps and Yelp applications on the device to provide relevant additional information. In the case that Yelp is not installed, the device's web browser should be used instead.

System Evolution

The environment will change during the lifetime of the product as locations will change. The application will need to be updated with new information regarding the closing/opening of locations, stores, restaurants and landmarks. Additionally, as the user-base expands, the quality of the service will improve with a greater number and quality of experts.

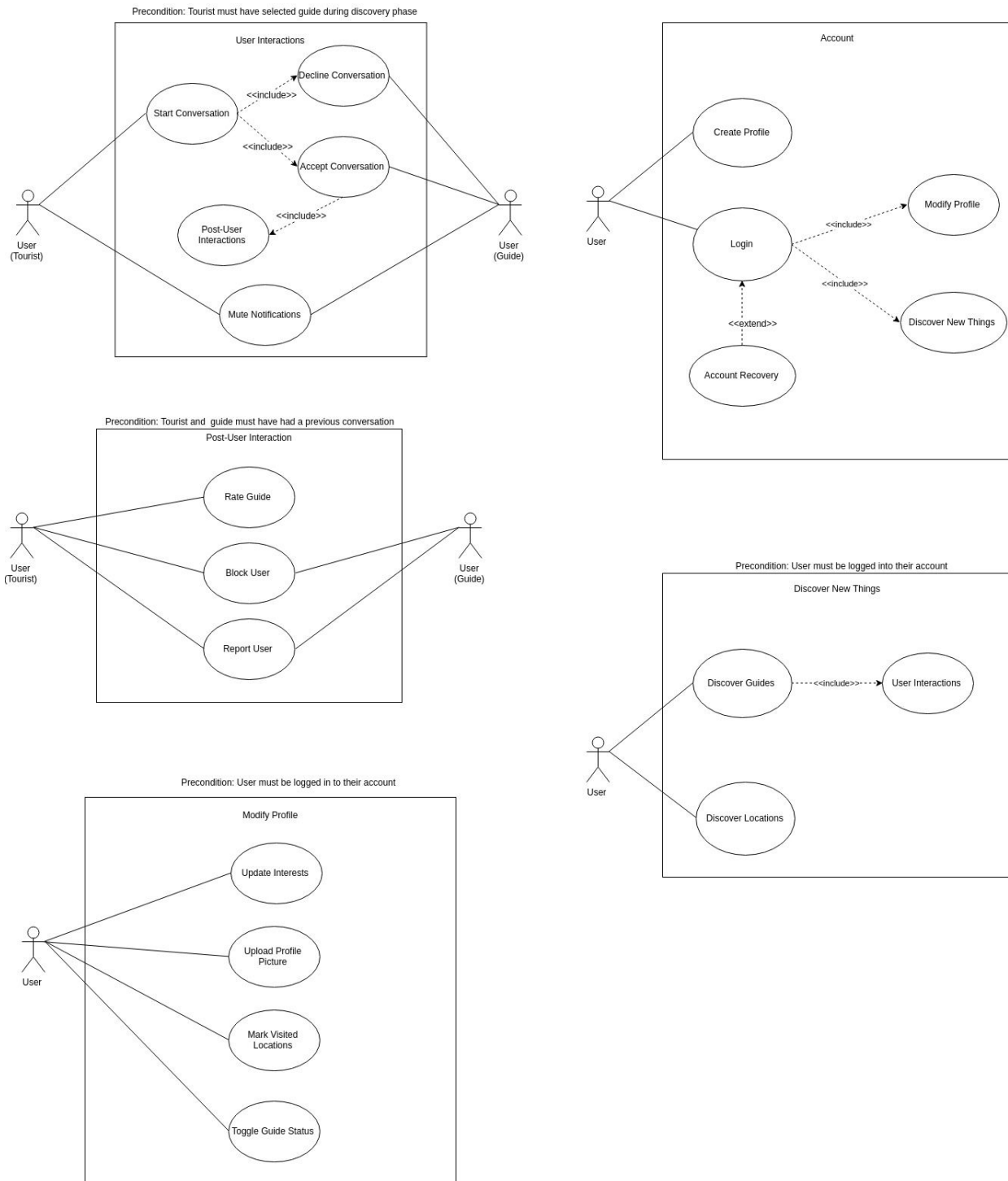
Functional Requirements

- Allow users to communicate with other users.
- Allow users to register / create profiles.
- Generate a map of recommended locations based upon personal interests.
- Allow users to find users with similar interests.
- Allow users to rate others that they have interacted with based on the helpfulness of their conversation.
- Connect with other services such as Google Maps or Yelp for relevant information.

Non Functional Requirements

- The user must have a data connection
- The user must have an Android device
- Scalability of the service as more users join
- Security for each users information

UML Use Cases Diagrams



Use Case Scenarios

- **Use Case: 1 Account**
- **CHARACTERISTIC INFORMATION**
 - Goal in Context: A system that allows users to log in to their existing profile.
 - Scope: Login Services
 - Level: Primary Task
 - Preconditions: The user already has an existing profile.
 - Success End Condition: The user is now logged in and can access all of the necessary functions within the application.
 - Failed End Condition: The user is given an error message and does not log into an account.
 - Primary Actor: User
 - Trigger: On opening of the application.
- **MAIN SUCCESS SCENARIO**
 - 1. The application is opened.
 - 2. The user is prompted to log into their account.
 - 3. Once successful login information is provided, the user is connected to the service and can perform any allowed functions.
- **EXTENSIONS**
 - 2a. User does not have an existing account.
 - Prompt the user to register for the service.
 - 2b. Invalid username/password is entered
 - Display an error and prompt the user to enter a valid username/password
- **SUB-VARIATIONS**
 - User may use
 - Username
 - Email address
 - Login with Facebook
 - Login with Google
- **SCHEDULE**
 - Due Date: 2/18/2019
- **Use Case: 2 Modify Profile**
- **CHARACTERISTIC INFORMATION**
 - Goal in Context: A system that allows users to log in to their existing profile.
 - Scope: Profile Management
 - Level: Primary Task
 - Preconditions:
 - The user already has an existing profile.

- The user's input is deemed valid.
 - Success End Condition: The user's profile is modified and saved.
 - Failed End Condition: The user's profile remains unmodified.
 - Primary Actor: User
 - Trigger: User selects the task.
- MAIN SUCCESS SCENARIO
 - 1. Modification is selected.
 - 2. User input is verified.
 - 3. User profile is changed and saved.
- EXTENSIONS
 - 2a. Invalid input is entered
 - Display an error and prompt the user to enter a valid input.
- SUB-VARIATIONS
 - N/A
- SCHEDULE
 - Due Date: 2/18/2019
- **Use Case: 3 Discover New Things**
- CHARACTERISTIC INFORMATION
 - Goal in Context: A system that allows the user to find new activities.
 - Scope: Suggestion System
 - Level: Secondary Task
 - Preconditions: The user has provided enough information to develop suggestions.
 - Success End Condition: A list of suggestions will be provided to the user.
 - Failed End Condition: Prompt the user to add additional information.
 - Primary Actor: User
 - Trigger: The user adds a interest.
- MAIN SUCCESS SCENARIO
 - 1. The user selects the option to discover.
 - 2. The system returns a list of locations based on user's interests and location.
 - 3. The user selects an option.
- EXTENSIONS
 - 1a. The user selects the option to connect with a guide.
 - The system returns a list of guides.
 - The user selects a guide (use case 4).
- SUB-VARIATIONS
 - The user may select a location based on:
 - GPS current location
 - Selecting on a map

- Zip Code / City Name
- SCHEDULE
 - Due Date: 3/25/2019
- Use Case: 4 User Interactions
- CHARACTERISTIC INFORMATION
 - Goal in Context: Connect tourists with guides and allow for a conversation between the two users.
 - Scope: Chat Systems
 - Level: Primary Task
 - Preconditions: The user is logged in and has a stable connection to the service
 - Success End Condition: Two users are able to have a private message thread established between them
 - Failed End Condition: Dialogue is not able to be established between users.
 - Primary Actor: Users (tourists/guides)
 - Trigger: User A (tourist) prompts User B (guide) to start a dialogue and User B accepts.
- MAIN SUCCESS SCENARIO
 - 1. The tourist selects a guide from a list, initiating a conversation.
 - 2. The guide accepts the conversation.
 - 3. The two users send message back and forth.
 - 4. The conversation ends.
- EXTENSIONS
 - 4a. The tourist rates the guide
 - Record and update the guide's rating (use case 5)
 - 4b. A user reports the other user
 - Record the conversation for review (use case 5)
- SUB-VARIATIONS
 - N/A
- SCHEDULE
 - Due Date: 3/11/2019
- Use Case: 5 Post-User Interactions
- CHARACTERISTIC INFORMATION
 - Goal in Context: Give users options to improve their experience on the application
 - Scope: Interaction Management System
 - Level: SubFunction
 - Preconditions: The user has completed a conversation with a guide.

- Success End Condition: The user is able to successfully leave a rating for the guide and both users are able to report the other if it is necessary.
- Failed End Condition: The user does not rate the guide.
- Primary Actor: User
- Trigger: At end of conversation
- MAIN SUCCESS SCENARIO
 - 1. The conversation between users is ended.
 - 2. The user is prompted to rate the guide.
 - 3. The guide and user are both also prompted if they would like to report the other user for unacceptable behavior.
 - 4. The rating is stored into the database and chat logs are stored if a report was issued.
- EXTENSIONS
 - 2a. The user does not rate their experience with the guide.
 - The guide rating is unchanged.
 - 2b. The user does rate their experience with the guide.
 - The score is averaged into the guide's already existing rating.
 - 3a. If a user reports another, then the chat logs between them are saved to a database to be reviewed for further action.
- SUB-VARIATIONS
 - 3. Users may also report another user before the conversation has ended.
 - The chat logs up to that point will be saved.
- SCHEDULE
 - Due Date: 3/11/2019

Hardware

- Android
- Firebase

Data Storage

- Locations
- User Information
- User Preferences
- Reviews
- Photographs of users/locations
- Types of activities

Budget

- 4 Computers
- 4 Android Devices
- Google Play Registration Fee (\$25)
- Firebase Monthly Fees (when user-base expands to over 100 simultaneous users - \$25/month or pay as you go)
- Man-hours - 360hrs total
 - Requirements Definition - 12hrs
 - Descriptions (Introduction, System Model, System Evolution, etc) - 4hrs
 - UML Use Cases - 3hrs
 - Use Case Scenarios - 3hrs
 - Gantt Chart - 2hrs
 - Requirements Specification - 12hrs
 - Architecture
 - Hardware
 - Project Design - 16hrs
 - Prototyping - 32hrs
 - User Manual - 32hrs
 - Basic Layout and UI Elements - 32hrs
 - Setup Firebase - 48hrs
 - Chat Component - 72hrs
 - Integrations Component - 32hrs
 - Debugging - 72hrs

Glossary

- Android - A mobile operating system developed by Google.
- Firebase - A mobile development platform provided by Google that allows for functionalities such as databases, logins, and analytics.
- Guide - A user that provides suggestions about locations, food, etc that another user should visit.
- Tourist - A user that is looking for suggestions about locations, food, etc.

Index

- Android - 3, 9, 10
- Firebase - 8, 9, 10
- GPS - 2, 6
- Google Maps - 2, 3
- Rating - 3, 7, 8
- Yelp - 2, 3