

**Software Requirements Specification (SRS) Document**

**Class**

CIS 453

**Contributors**

Eldin Sahbaz

Brian Heckman

Raymond Hu

Vincent Perez

Brice Buccolo

Jacob Connors

**Preface**

Document Description

This document is the Software Requirements Specification (SRS) for the mobile tour guide application, Halos.

**Version History**

Version 1.0 10/14/2016

* User Requirements Draft
* System Requirements Draft

Version 1.1 10/30/2016

* Finalized Introduction
* Finalized User Requirements
* Finalized System Requirements

**Introduction**

Purpose

Halos is an application that helps tourists in new cities by connecting them with people from all over the world or with the services that these people provide. This application, like Uber and Airbnb, is - at its core - a marketplace. Uber’s commodity is transportation. Airbnb’s commodity is real estate. Halos’ commodity is tourism. Through Halos, tourists are connected to curated tours and local guides (people who live in the user’s proximity and guide tours). However, Halos is more inclusive, in that it also allows tourists to take auto-generated tours or to make tours themselves.

Target Audience

We envision there to be two main categories of active users on this application. The first category consists of tourists and the second category consists of curators/guides. Tourists, curators, and guides are not necessarily mutually exclusive. A tourist may also be a curator or guide for his or her home location or places he or she has traveled. Likewise, curators may create tours for others as well as guiding tours (and vice-versa).

Project Overview

Halos is a tour guide application, and, to maximize our customer base, it will be available on both Android and iOS Platforms. This application aims to reduce the complexity of tourism by either auto-generating tours, allowing tourists to easily create their own tours, connecting tourists with tours others have created, or connecting tourists with locals.

There are going to be four essential ways of taking a tour via Halos. The first way a user may take a tour is by auto-generating one based on his or her current location, method of transportation, and tour duration. Using location services, Halos will find the points of interest nearest to the user and create a tour. This tour will fit within the specified time constraint and use the specified method of transportation. The second way a user may take a tour is creating one himself or herself. This can be done by drawing a path through any or all of the nearby landmarks generated by Halos. After the user draws the path, he or she is asked if this is the desired route. If the user is satisfied, all he or she has to do is continue onto the tour, otherwise, the user may erase the current path and restart. The third way a user may take a tour is to either find a curated tour in the store or to connect with a guide in Halos’ store. In the store, tours and guides (locals) alike can be found by searching for keywords, going to our featured page, top guide page, etc. By default, tours and tour guides in the user’s area will be shown to him or her (unless the user explicitly searches for something out of the area). In the store, the user may go search for and select the desired tour. Once selected, he or she will see a description page detailing information about that specific tour. The user then has the option of downloading the tour after purchasing it (if it’s not free). Likewise, users may go search for and select the desired guide(s) in their area. Users who put custom/curated tours in the store and tour guides may choose to charge a fee for their tours or services. Once the user purchases their desired tour, the tour is saved to the user’s playlist. All data in the playlist will be saved locally on the user’s phone so that the users may have access to this information while they are offline. This point brings us to the fourth way that a user may take a tour in Halos. As previously mentioned, the user will always have access their playlist. Therefore, he or she may retake tours that are saved in his or her playlist. The only caveat is that; the users will need internet access to actually take the tour.

During the auto-generated tours, Halos dictates important information about the landmarks once the user is within a certain radius of it. During curated tours, downloaded from the store, the curator will be responsible for both creating the text to be dictated for each landmark and for verify the validity of this information.

A neat feature that we include in all of our tours is the ability to add or remove points. Say the user auto-generates a tour in Halos, he or she then has the option to remove a point from the path (and we will recalculate the new path). When a user creates a custom tour, he or she has the option adding a landmark not shown by Halos (temporarily) to his or her path. If the same custom landmark is added by 100 different users, then Halos will begin including it as a regular landmark.

We also provide a group tour feature to our users. If the user chooses to turn on the group tour feature, he or she will send a notification to all other users nearby, requesting that they join the tour. The group as a whole has a limited amount of time to agree upon which points get removed/added from/to a tour, and may make any necessary adjustments. Changes are only made if the majority of the group agrees the change should be made, and only votes casted will be counted (i.e. no vote, no voice).

User profiles will feature ratings and comments on tours that the user has taken, created, or guided. Guides may leave ratings and comments on how the given user is as a tourist (i.e. is he or she respectful, etc.). Ratings will be on a scale of zero to five stars and the user’s profile will display an average rating that is an aggregation of all groups (i.e. tourist, guide, curator) that have available data. The user’s profile will have links that show all the tours a user has taken, created, or guided. Users on a tour will have the opportunity to provide feedback and rate their guide(s)/tours/landmarks.

Scope

The planning and designing phase for Halos will take place during the Fall of 2016. The actual implementation of Halos will take place in the Spring of 2017. We expect the Go-Live date to be sometime in the late Spring of 2017 - the exact date has not yet been determined.

**Glossary**

Guides –  guides are locals who live in the user’s proximity radius and take tourists on tours.

Curator – A person who creates tour routes for others to take.

Proximity Radius – The circle generated around your current location with an adjustable radius. Tours and guides are searched for within this radius.

**Functional Requirements**

**CREDENTIALS- LOGIN, FORGOT INFO, FINGERPRINT, CREATE ACCOUNT**

1. There shall be user and guest accounts.
   1. The application shall prompt users to sign in upon application launch.
   2. If the user does not have an account, the user shall have the option to either create a new account or continue as guest.
   3. There shall be an option for forgot password or forgot username
   4. If none of these options are chosen, remain on the login page.
   5. Halos shall offer the option to login via a fingerprint scan (on supporting devices)
2. Too many login attempts shall lock a user’s account
   1. A user shall have at most five contiguous failed attempts before his or her accounts is locked
   2. If a user’s account is locked, he or she is sent an email with a password reset link
   3. The user shall not be able to access the account until his or her password is reset
3. If the user logins from a new location, he or she shall receive an email notifying them of the account access
   1. Halos shall use location services to check if this is a familiar location for the user
   2. If the user has never been to current location, Halos shall send the user an email letting them know the account has been accessed in a new location
   3. Users shall have the option to report their account as compromised and reset their password
4. The user shall have the option to continue as a guest
   1. The user shall have the option to click button on login page saying ‘continue as guest’
   2. Personal information (e.g. tours taken, location, etc.) shall not be stored for guests after logout.
   3. Guests shall have the ability to contribute to the community by rating sites.
   4. System shall only display options to custom or auto-generated tours.
   5. System shall deny access to the store to guests.
   6. If a guest attempts to access the store, a message shall be displayed asking the user to create an account to access the requested features.
5. Halos shall verify the user’s credentials if he or she selects forgot password or username.
   1. Halos shall give option to scan fingerprint and shall email the username and a reset link to the user’s email if the fingerprint is valid.
   2. Halos shall give option to input an email address
   3. Halos shall scan the database to verify that the input email exits for a current user.
   4. Halos shall send an email containing temporary password or username to that email, with a link to create a new password.
6. The user shall create a new password upon password reset
   1. Users shall not use a previously used password
   2. Users shall be allowed at most 3 password resets per month before his or her account is locked for a week
   3. Upon account lock, the user shall be notified that his or her account is disabled for a week and to reset it once the allotted period of time is over

**LOG OUT**

1. The users shall be able to logout from their accounts
   1. Only registered users shall have database-stored data.
   2. Guest gets deleted upon logout.
   3. Logging out shall store all the of the user's data from their last session to the database.
   4. Old session data shall be overwritten by new session data

**USER PROFILE PAGE**

1. Personal information for registered users shall be saved to the database (e.g. tours taken, duration of tour, location, preferences, ratings, etc.).
   1. Registered users shall have profile pages.
   2. A user’s profile page shall include information such as username, favorite places, places they’ve been recently, interests, and a profile picture.
   3. Registered users shall have user levels.
   4. As users take and rate more tours, they shall gain levels
   5. Level-ups shall reward users with new perks.
2. Halos shall display map with generated route when user selects a tour from their user profile.
   1. The map shall display colored tour route containing the points of interests from the downloaded tour.
   2. Users shall have the option to change time allocated for tour route and system will adjust route accordingly.
   3. Users shall have option to adjust default proximity and system will adjust route accordingly.
   4. Users shall have the ability to remove any points of interest from the tour, and the route will readjust accordingly.
   5. User shall have the ability to add points of interest to the tour, and the route will adjust accordingly.
3. Halos shall load user’s list page when user list option is selected.
   1. System shall load user’s personal list page from database.
   2. This list shall contain information such as tour duration, ratings, and tour stops.
4. Users shall have a profile page.
   1. Users’ profile pages shall consist of a profile picture, name (username or real name), all ratings a user has given/received, and an overall user rating.
   2. Ratings shown on a user profile shall be separated into three categories, Traveled (ratings for tours a user has taken), Curated (ratings for tours a user has created), and Guided (ratings for yours a user has guided).
   3. Most recent ratings shall be shown first for each category in a user’s profile page.
   4. A user’s Traveled rating shall mostly be determined by the “rating of ratings” system (see req. 39.d.).
   5. Users shall be able to edit certain aspects of their user profile, such as name displayed and their profile picture.

**MAP**

1. Halos shall display a map of the user’s current location upon login.
   1. Map shall only show points of interest in a default search radius.
   2. Registered Users shall have the option to expand or retract their search radius.
   3. Guests shall not have the option to expand or retract their search radius.
   4. Users shall have the ability to remove any points of interest from their map (local to user account).
   5. Users shall have the ability to add any points of interest to their map (local to user account).

**MENU BUTTON**

1. Halos shall have a menu button in the top right hand corner of every page.
   1. Users shall be able to use the menu button to access any page within the app from their current page.
   2. The system shall redirect the user to his or her home page when the home page is selected.
   3. The system shall redirect the user to his or her playlist when the playlist is selected.
   4. The system shall redirect registered-user to the Halos store when the Halos store is selected.
   5. The system shall redirect user to the options page when options page is selected.
   6. The system shall log users out when the logout option is selected.

**SIGHTSEE**

1. Halos shall display a map with an auto-generated tour route when user selects map option.
   1. There shall be a default proximity that fits the user-set time constraint.
   2. Users shall have the option to change time allocated for their tour.
   3. Users shall have a default proximity radius.
   4. Users shall have the ability to adjust their proximity radius to their preference.
   5. Users shall have a default time limitation.
   6. Users shall have the ability to adjust the time constraint to their preference.
   7. Users shall have the ability to remove any points of interest from the tour, and the route will readjust accordingly.
   8. Users shall see a colored route on the map containing points of interest.
   9. Users shall have the ability to rate each individual point of interest as well as the overall tour on a 5-star rating system.
   10. User shall have the option to save the auto-generated tour to their local playlist.

**GROUP TOUR**

1. User shall have the option to take a ‘Group Tour’ before generating the suggested tour.
   1. There shall be an option on the auto-generate page to make the current tour, a group tour.
   2. If the group tour option is selected, a message shall be sent to other users who are in the current user’s proximity
   3. Users shall be matched with another person(s) who is/are interested in touring the same area.
   4. The user shall select their desired group size.
   5. Users will be added to the group chronologically until either time runs out or the group is full (whichever comes first).
   6. If the group is still empty, the user shall be able to request a group tour again or continue as a single tour.
   7. The user shall select a time limit to wait for other users to enter the group.
   8. Users shall be matched according to similar interests based on what they have liked in the past.
   9. Users shall give a suggested proximity radius.
   10. Users shall give a suggested time frame for the tour.
   11. User shall have to select a ‘Meet to Tour’ option in order to accept the suggested partner and tour.
   12. Users shall meet at the first tour stop at the time suggested in the previous step.
   13. Users shall be able remove stops from the tour by selecting the remove button.
   14. Upon selecting the remove button, an anonymous notification shall be sent to the rest of the group asking if they agree to remove the stop.
   15. If greater than 75% of the group agrees to remove the stop, the point of interest shall be removed.
   16. Members of the group who do not vote shall not have an impact on the overall tally.
       1. If there are 10 members in a group, 1 does not vote, and 7/9 vote to remove the point of interest, the stop will be removed (7/9 > 75%)
   17. Users shall have 15 minutes to make changes to the tour before the users must meet at the start location.

**CUSTOM TOUR**

1. Users shall have the option to create a custom tour.
   1. The user shall have the option to adjust their search proximity.
   2. Halos shall generate local landmarks for the user.
   3. The user shall be able to select a landmark and swipe their finger across the map, connecting the landmarks they wish to visit.
   4. Landmarks shall be added to the tour chronologically, in the order the user selected them.
   5. The last landmark in the selection shall be the ending location.
   6. There shall be only one starting location.
   7. There shall be only one ending location.
   8. Once the user finishes creating his or her route, Halos shall ask the him or her if he or she would like to proceed to the tour.
   9. If the user chooses to proceed, he or she shall be brought to a preview page.
   10. The review page shall give the user an overview of the tour.
   11. The user shall have the option to go back and edit the tour once on the preview page.
   12. The route shall be saved to the user’s local playlist.
2. Users shall be able to create a custom landmark
   1. The page shall be able to tap anywhere on the map to open a new address window.
   2. The user shall be able to input the address of a landmark not currently displayed on the map.
   3. Adding this address shall add a custom landmark to the map.
   4. Custom landmarks shall be a different color than the landmarks generated by Halos.
   5. The system shall add the custom landmark to the database.
   6. Halos shall add the custom landmark to the user’s map (local to user).
   7. If the custom landmark is tagged by at least 100 unique users, it shall be displayed on the map of all users in that area.
   8. If the custom landmark is not tagged by any other unique user, the custom landmark shall be removed from the database.
   9. If a custom landmark is removed from the database, it shall remain on the user’s map (local to user).

**LANDMARK DESCRIPTION**

1. Users shall be able to view a description of the landmark.
   1. The user shall have the ability to tap on a landmark to view details about it.
   2. The detail box shall be a pop-up box on the top of the map.
   3. The detail box shall include the name of the landmark.
   4. The detail box shall include the distance from the user to the landmark.
   5. The detail box shall include the estimated time of arrival to the landmark.
   6. The detail box shall include a brief summary of what the landmark is (if available).
   7. The detail box shall include and a picture of the landmark (if available).
   8. The detail box shall include the average time users spend at the location (if available).
   9. The detail box shall include the average rating the landmark has received (if available).
   10. If any of these fields are not available, “N/A” shall be displayed for it
2. Users shall have the ability to view the progress of their tour as they are taking it.
   1. At any time during the tour, the user shall be able to access the tour overview.
   2. In the tour overview, the user shall be able to view the current stop on the tour.
   3. The current stop shall be displayed at the bottom of the map.
   4. The user shall be able to click on the current stop to view the description of it.
   5. The overview screen shall be accessible through a button in the bottom right corner.
   6. The review screen (mentioned in 11.f) and the overview screen shall be the same.

**PLAYLIST**

1. Halos shall have a local playlist that contains all of the tours for the user, split into three distinct categories.
   1. The playlist shall have a “My Tours” tab, which contains all of the custom and auto-generated tours that the user has curated and/or guided.
   2. The playlist shall have a “Downloaded” tab, which contains all of the tours that the user has downloaded from the Halos store.
   3. The playlist shall also have a “Travelled” tab, which shows the user a comprehensive list of all the tours they have completed.
   4. The playlist shall allow the users to tap on any given tour to view the information page of the tour with the list of the landmarks the tour visits.
   5. Users shall then be able to tap on a specific landmark within a tour to view the details for the landmark.
2. Users shall be able to access the playlist through the dropdown menu.
   1. Users shall also be able to access the playlist through their profile page.
3. Users shall have the ability to upload a custom tour from the playlist to the Halos store.
   1. Each individual tour that the user has custom created shall contain a button that allows them to upload it to the Halos store.
   2. The button shall give the option for the user to upload their tour for free or for money.

**DICTATION**

1. Users shall have the option to choose if their tours dictate to them
   1. Users shall have to option to disable dictation in auto-generated tours.
   2. Users shall have to option to enable dictation in auto-generated tours.
   3. Users shall have to option to disable dictation in curated tours.
   4. Users shall have to option to enable dictation in curated tours.
2. Users of all backgrounds shall be able to make use of dictation
   1. Dictation shall dictate text from any language in the user’s language of choice.
   2. If there is no text available for dictation, there shall be no dictation.
3. Curators shall have the option to include dictation in their tours
   1. Curators shall manually input text to be dictated to tourists.
   2. Curators shall leave text fields blank if nothing is to be dictated.
   3. If there is no text available for dictation, there shall be no dictation.

**SUGGESTION ENGINE**

1. Halos shall make tour suggestions
   1. Halos shall suggest tours based on tour and rating history.
   2. Halos shall suggest both paid and free tours.
   3. Halos shall suggest only tours in the user’s proximity.
   4. Halos shall suggest only tours that can be completed in a user given time frame.
   5. If there are no tours available, no tours shall be suggested.
2. Halos shall make guide suggestions
   1. Halos shall suggest guides based on tour and rating history.
   2. Halos shall suggest both paid and free guides.
   3. Halos shall suggest only guides in the user’s proximity.
   4. If there are no guides available, no guides shall be suggested.

**NOTIFICATION SYSTEM**

1. Halos shall feature a notification system for alerts.
   1. Halos shall issue a notification and refund money if your tour is cancelled.
   2. Halos shall issue a notification when a transaction is completed.
   3. Halos shall notify groups/local guides when a tour is starting.
   4. Halos shall notify a user when taking off points.
   5. Halos shall send a notification if another group member has requested to remove a tour stop, asking for their vote.

**STORE**

1. Halos shall load store featured page when store button is selected or when featured button in store is selected.
   1. Store shall always have 4 buttons at the bottom consisting featured, top guide, search, My Location, and update regardless of currently loaded page.
   2. Store shall have 1 checkout button at the top right, regardless of currently loaded page.
   3. Store shall provide drop-down menu to leave store for other functions within the Halos.
   4. Featured page shall have horizontal list of 20 selectable squares with creators listed under it where each represent the newest tours which is all viewable by scrolling right or left.
   5. Featured page shall have a 2nd horizontal list of 20 selectable squares with creators listed under it where each represent the most recent highly rated tours.
   6. Featured page shall have a 3rd horizontal list of 20 selectable circles with pictures listed under it where each represent a tour chaperone currently hosting a tour.
   7. Lists of tours shall refresh every hour.
2. Halos shall display search page with a search bar when search button is selected.
   1. Search bar shall immediately print user input as user types.
   2. User will have the option to search by usernames or by location and enter to start search.
   3. System will bring up the first 10 closest matches and have a “see more button” at the bottom if more than 10 are found.
   4. User shall have the option to click on these tours and be redirected to their information page.
   5. System shall bring up a no-results found page with some suggestion for searching if no results are found with the search bar at the top of the page.
3. Halos shall display top tours beginning on the paid list when top chart button is selected.
   1. System shall display 4 tabs consisting of paid, free, top grossing, top local guides at the top.
   2. System shall display the top 10 paid tours with price when paid is selected with a “see more button” at the bottom.
   3. System shall display the top 10 free tours when free is selected with a “see more button” at the bottom.
   4. System shall display the top 10 most positive rated tours when most rated is selected with a “see more button” at the bottom.
   5. System shall display the top 10 best rated local guides (actual tours personnel) when top local guides is selected.
   6. The “see more button” shall load top 100 tours or local guides of that tab on a new page with another “see more button” at the bottom for the next top 100.
   7. System shall allow users to swipe down or scroll down to see the entire list.
   8. User shall have the option to click on these tours and be redirected to their information page.
4. Halos shall display the top 10 tours in your area when My Location is selected.
   1. System shall display a list containing the top 10 tours with their ratings from highest to lowest that begin in your area with a “see more button” at the bottom.
   2. User shall have the option to “see more button” which will bring up the top 25 guides in your area.
   3. User shall have the option to adjust proximity circle to expand or decrease search results from minimum 1 mile to maximum 5 miles.
   4. User shall have the option to click on these tours and be redirected to their information page.
5. Halos shall display update page when update button is selected.
   1. Update page shall display 2 option tab consisting of update and manage while starting on the update tab.
   2. Selecting update tab shall display a list of all the user’s downloaded tours that currently have an update status with an update button next to it or tours they are currently joined on.
   3. Users shall have the option to click on these tours and be redirected to their information page.
   4. Halos shall update the tour you select the update button for by deleting it from your local guide directory and downloading the newly updated tour.
   5. Selecting manage tab shall display a list of all tours created by the user with the option to delete or + (improve) next to each tour.
   6. System shall delete the tour from the store when delete is selected.
   7. System shall display two lists next to each other, called current points and area points with picture and description displayed above. when + button is selected
   8. Current point list shall allow users to see the list of points from beginning to end of the tour with the option to delete any points from the tour.
   9. Area point list shall allow user to drag and drop from a list of currently un-added points from closest to farthest from the end point of the tour within the proximity circle of that point.
   10. User shall have the option to adjust proximity circle from 1 block to 5 miles.
   11. Tour creators shall have option to change square picture and description by clicking on it and uploading their own picture or words.
6. Halos shall display checkout page for payment method and information when checkout button is selected.
   1. System shall display checkout page with all the tours in checkout cart with pay button at the bottom.
   2. System shall increment checkout cart total by 1 each time user clicks to download a tour and store that tour in the checkout cart.
   3. Free tours shall have a listed price of zero listed next to them.
   4. Paid tours shall have their price listed next to them.
   5. User shall have the option to remove tours from cart or click on one of the 6 store buttons (besides home) to add more tours which saves the current state of the cart before loading the button selected.
   6. Halos shall ask for credit card information and user information to verify authenticity of user with a submit button at the bottom when pay button is selected.
   7. User shall be re-directed to a pay page with empty fields again with message in red that the card is denied or user information is incorrect at the top.
   8. System shall take payment once payment is verified and submitted and being downloads of all the tours selected.
   9. System shall provide cancel button on each checkout page to go back to store homepage and all information inputted during checkout will be cleared.
   10. User shall see a receipt page of tours bought and prices with a home button at the bottom.
7. Halos shall display the tour information when its square is selected.
   1. System shall display a top area with tour icon with name on the top left with creator, rating, and price next to it.
   2. System shall display status button of tour, whether it is installed, download, or update in the top right of top area.
   3. System shall update app when user presses the status button with these statuses.
   4. System shall add tour to checkout cart when users selects download button.
   5. System shall display a list of all the points containing their names and times in the order from initial start point to the end which is viewable by scrolling down.
   6. System shall display average amount of time spent by other users on this tour.
   7. System shall display average amount of time spent by users at each location point.
8. Halos shall display tour chaperone (local/guide) information when circle is selected.
   1. System shall display a top area containing the local’s/guide’s name and picture with contact information, price and description.
   2. System shall display price next to a status button containing join, cancel (if user has already selected join.) within the at the bottom area.
   3. System shall display start and end time of tour in the top area.
   4. System shall display minimum and maximum of people allowed on the tour, and show current number of people that already joined the tour at the bottom.
   5. System shall increment current number of people already joining the tour by 1 every time after a user pays to join the tour in checkout.
   6. System shall decrement number of people every time a user cancels the tour.
   7. User shall receive notification once tour is cancelled by user or local/guide.
   8. System shall display list of points, each next to its own short description and name under the top area.

**SAVE INFORMATION**

1. Progress on a tour will be saved if Halos closes.
   1. If Halos was closed (voluntarily or involuntarily) and the user is still within a close proximity of the tour route, Halos will prompt the user if he/she would like to continue the tour. If yes, continue the tour, if no, stop the tour.
   2. Checkout cart in store shall clear if Halos is closed.

**RATING SYSTEM**

1. Halos shall generate a rating system for tours, points of interest, and tour guides (if applicable).
   1. A 5-star rating system shall be generated when user concludes tour or points of interest.
   2. System shall pop up a 5-star rating about point of interest when after user moves away from point of interest after spending time there.
   3. System shall pop up a 5-star rating about overall tour when user moves away from last point of interest or concludes tour (voluntarily).
   4. For ratings of tours, users shall have the option to rate other users’ ratings (similar to Amazon’s “x out of x users found this review helpful”). These ratings shall be used to filter out illegitimate reviews, among other reasons.

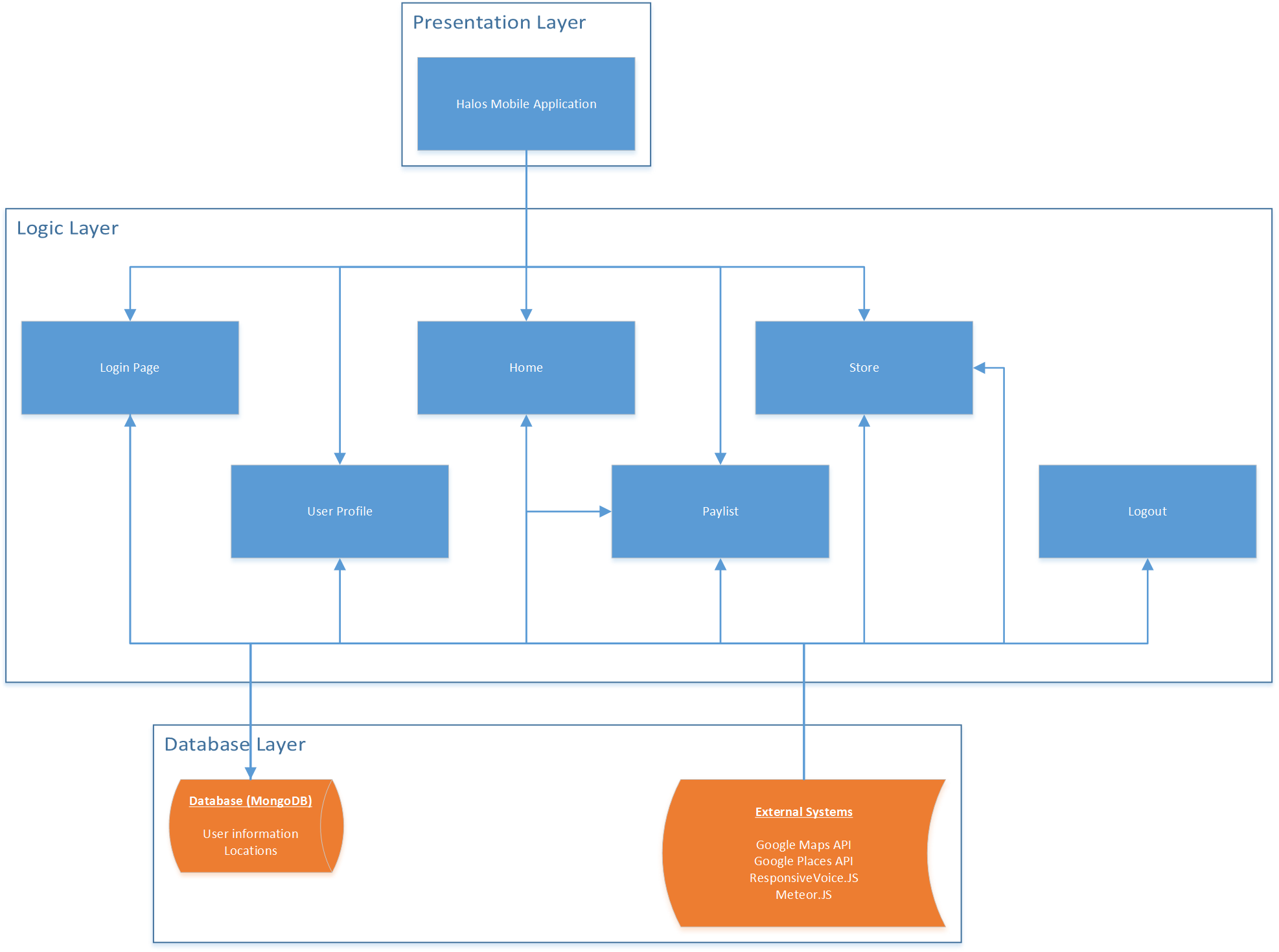
**ACCURACY STANDARDS**

1. The application shall be accurate.
   1. The user’s location shown in the application shall be the user’s actual location.
   2. Suggestions displayed to users shall accurately reflect their interests.
   3. Tours shall be recommended based on the user’s history of tours and reviews.
   4. The landmarks in the user’s proximity shall be actual landmarks
   5. The landmarks in the user’s proximity shall actually be in the user’s proximity
   6. The expected tour time shall be at most the time the user specified
   7. The expected transportation shall be the transportation the user specified
   8. The text to speech feature shall dictate text correctly.
   9. If auto-generated, text-to-speech shall be factually correct.
   10. If curated, the curator shall be responsible ensuring the information is factually correct.

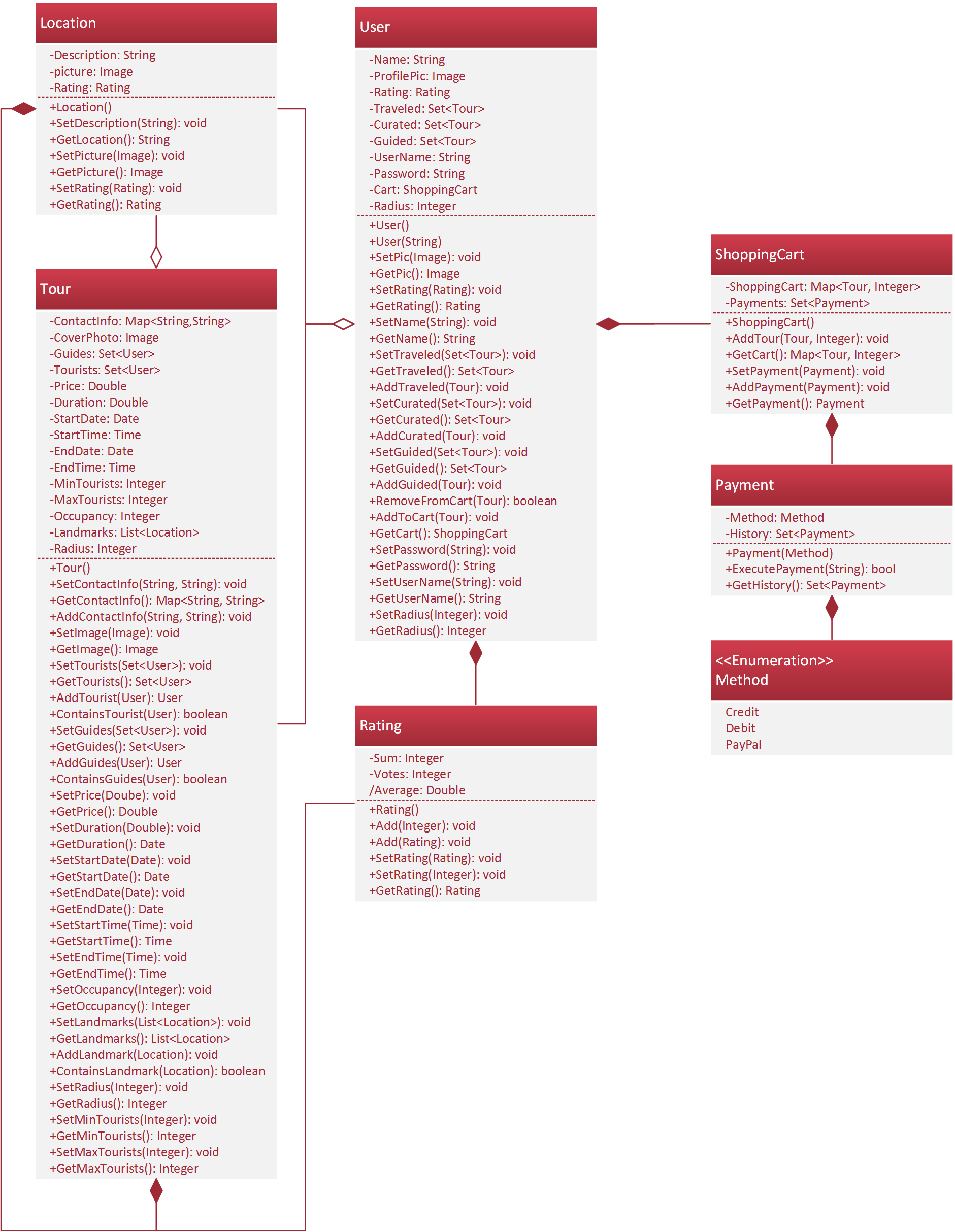
**Non-Functional Requirements**

1. User information shall be secure.
   1. Credit card information, name, birth date, password, username, and email are all encrypted.
   2. Data stored to local devices shall not corrupt the device.
   3. User information shall not be affected by any sort of application malfunction
      1. application crash
      2. phone shutdown
2. The application shall be time efficient.
   1. The application shall start up within 10 seconds.
   2. Users shall get their auto-generated tours within a twenty-second timeframe.
   3. Users shall be able to login (after entering valid credentials) within 2 seconds
   4. Users shall be able to log out within 2 seconds.
   5. Users shall be prompted to re-enter credentials (if invalid) within 2 seconds of submission.
   6. Upon clicking the reset password button, users shall receive an email within 5 minutes.
   7. Users shall connect to the store within a 5 seconds.
   8. Users shall be able to load content from the store within 2 seconds.
   9. Users shall be able to toggle between offline pages (within the app), in at-most 1 seconds.
   10. The transaction time between the server and database shall not exceed 1 seconds.
   11. Search shall return accurate results within 1 second of the query.
   12. Application shall immediately display user input in search bar as user types.
   13. Text-to-speech shall begin dictating (if applicable) information about landmarks to the user within 2 seconds.
3. The application shall be memory efficient.
   1. Only memory usage shall be to store a local copy of tours.
   2. The application shall not take up more than 200mb on the user’s phone
4. The application shall be maintainable
   1. The code shall be written in such a fashion to make sure developers can read and understand it.
   2. The code shall be easy to change (to adjust to changing requirements).
   3. The code shall be maintained in part by using a standard version control system such as GitHub or GitLab.
   4. There shall be documentation to explain each major piece of functionality.
5. The application shall be usable
   1. Users shall know how to operate the application after a day of usage.
   2. The system shall be interoperable between iOS and Android.
   3. If the application goes down, it shall not be down for more than 12 hours.
   4. Users shall update application whenever an update is out to be available at all times.
   5. User shall have access to application any day of the week

**System Architecture**

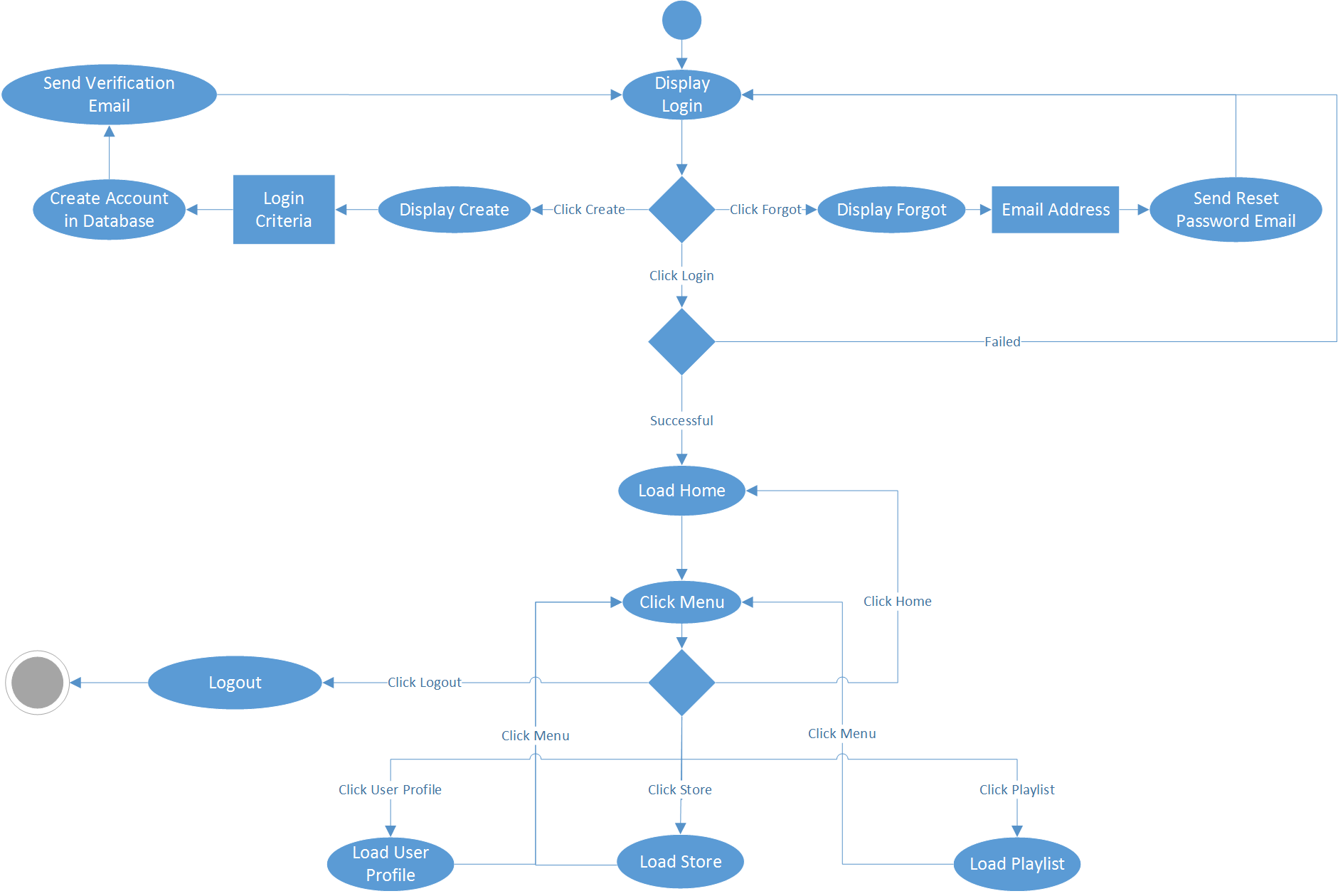


**Class Diagram**

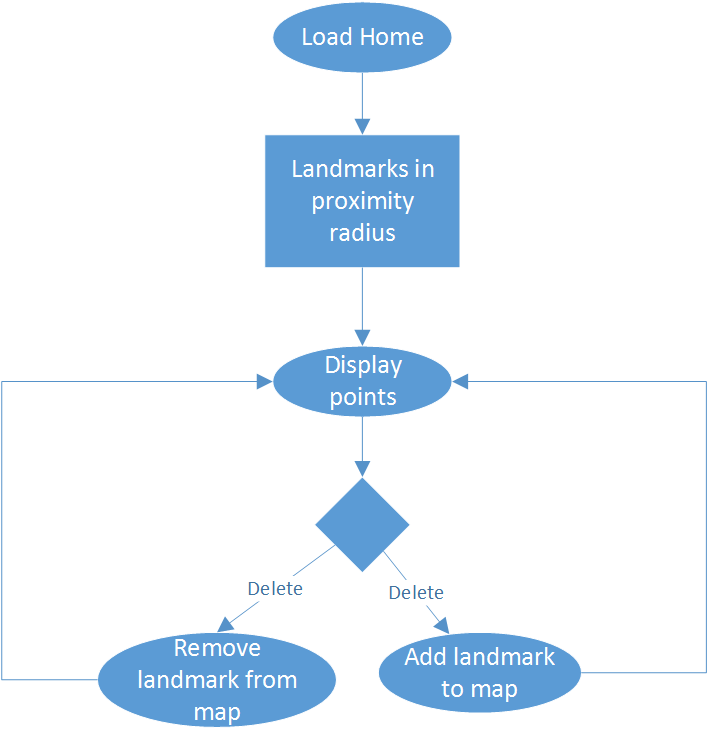
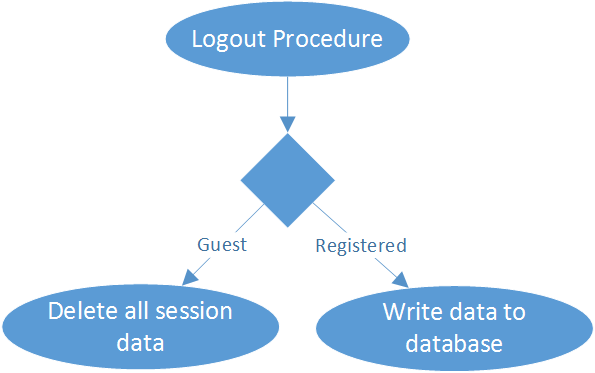


**Activity Diagram**

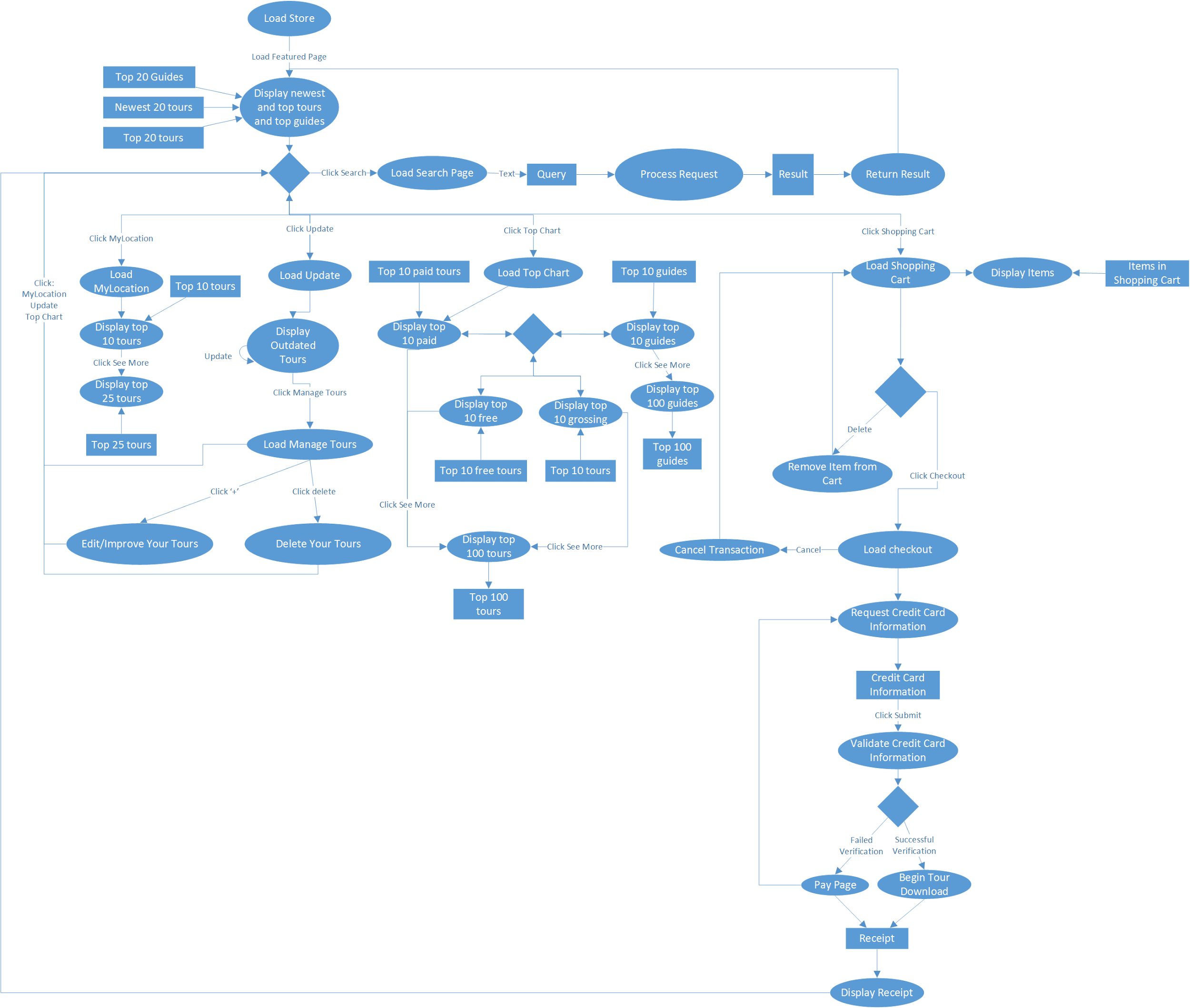
**(General View)**



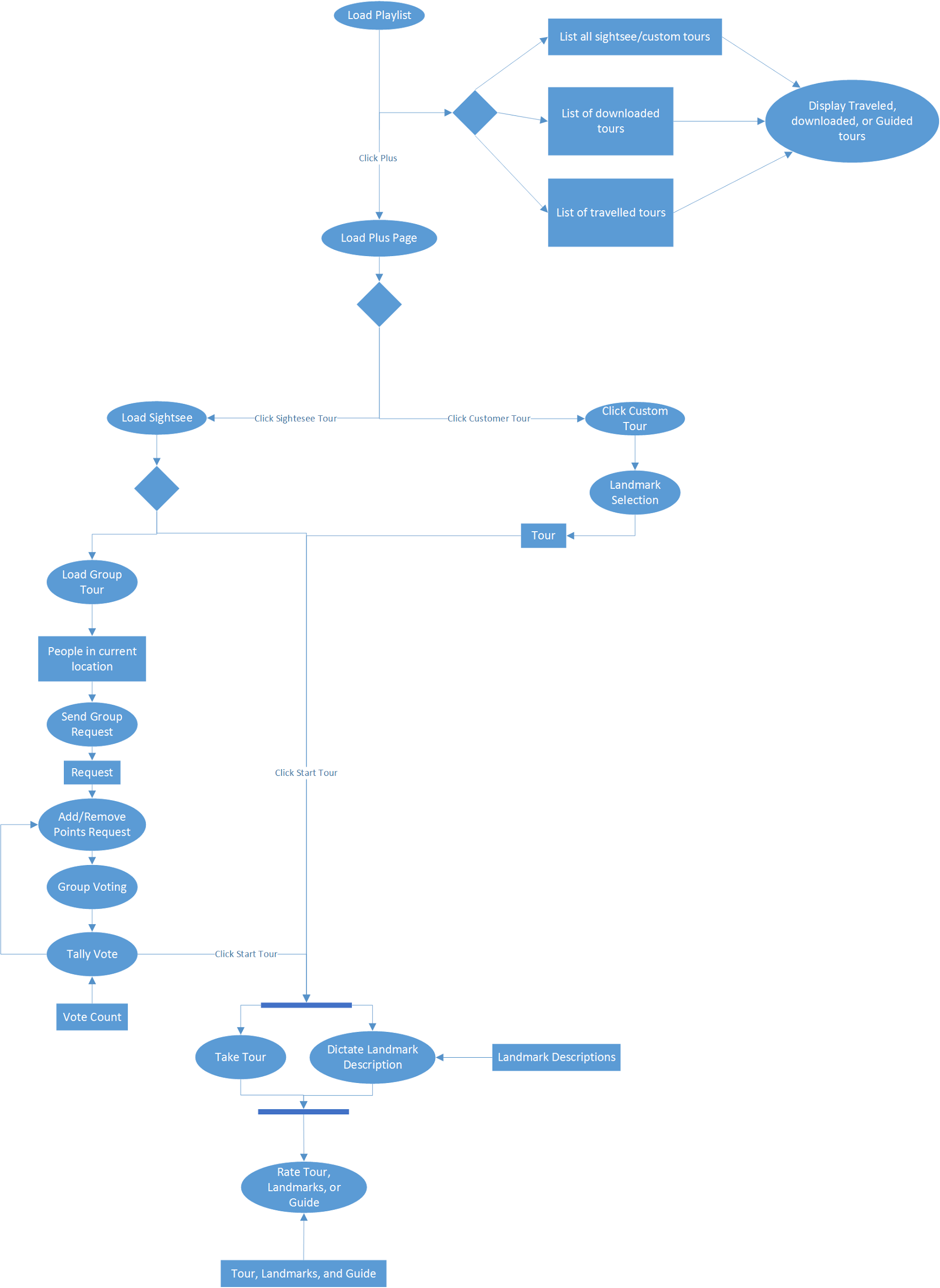
**(Logout View) (Home View)**



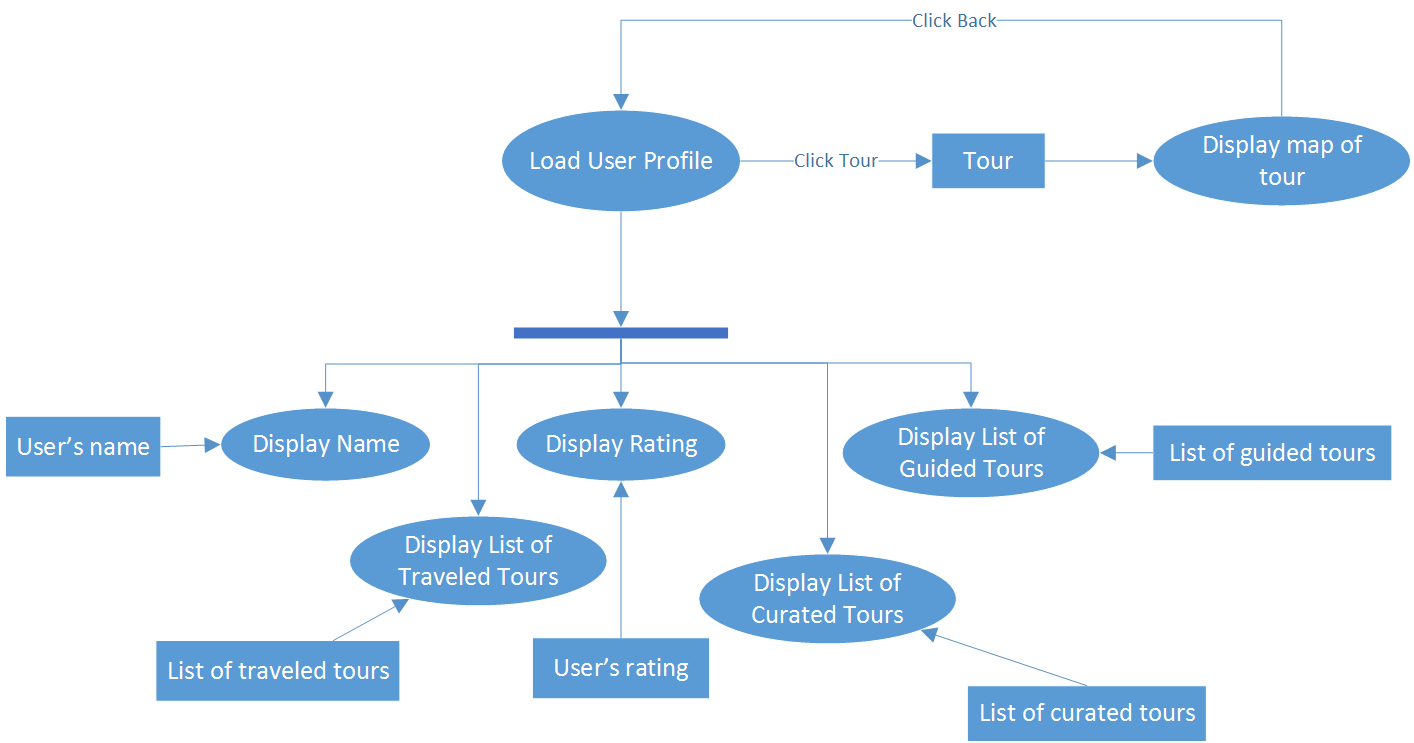
**(Store View)**

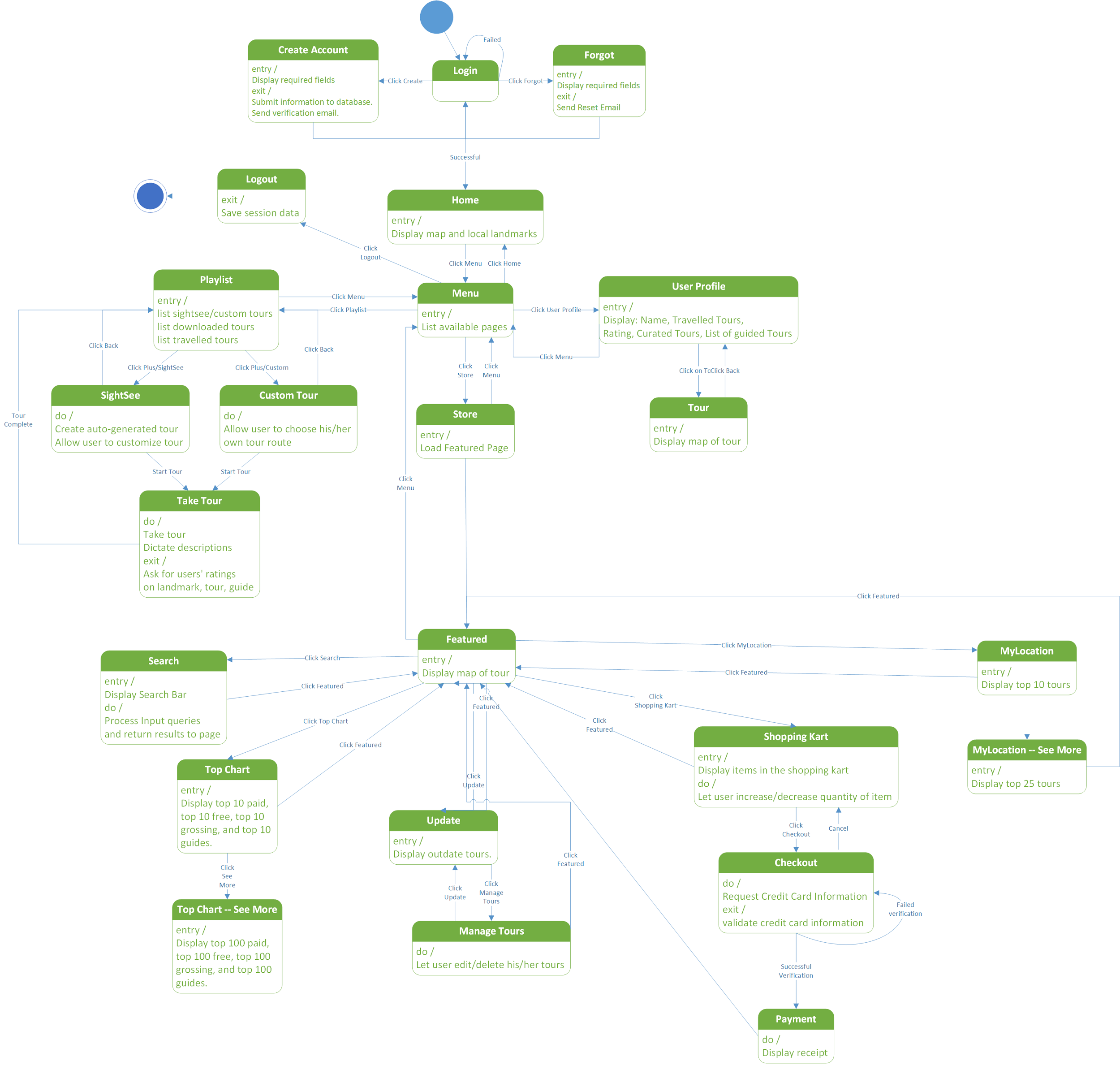


**(Playlist View)**

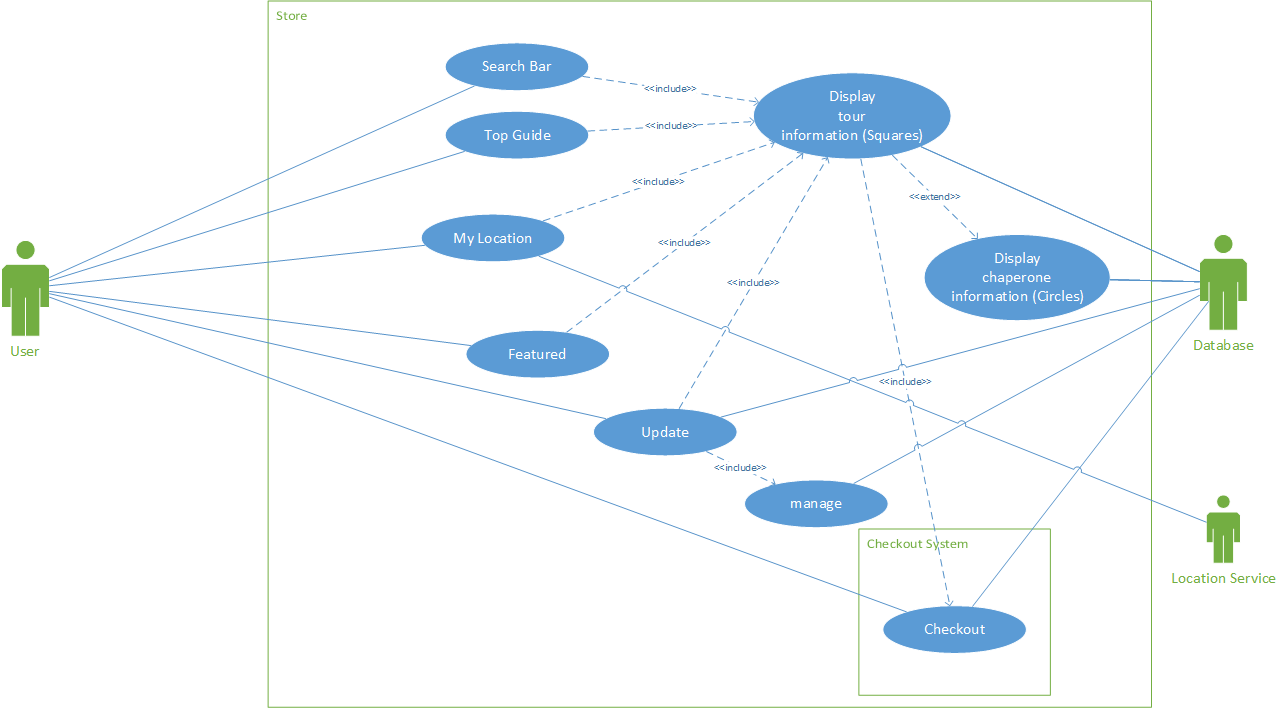


**(User Profile View)**

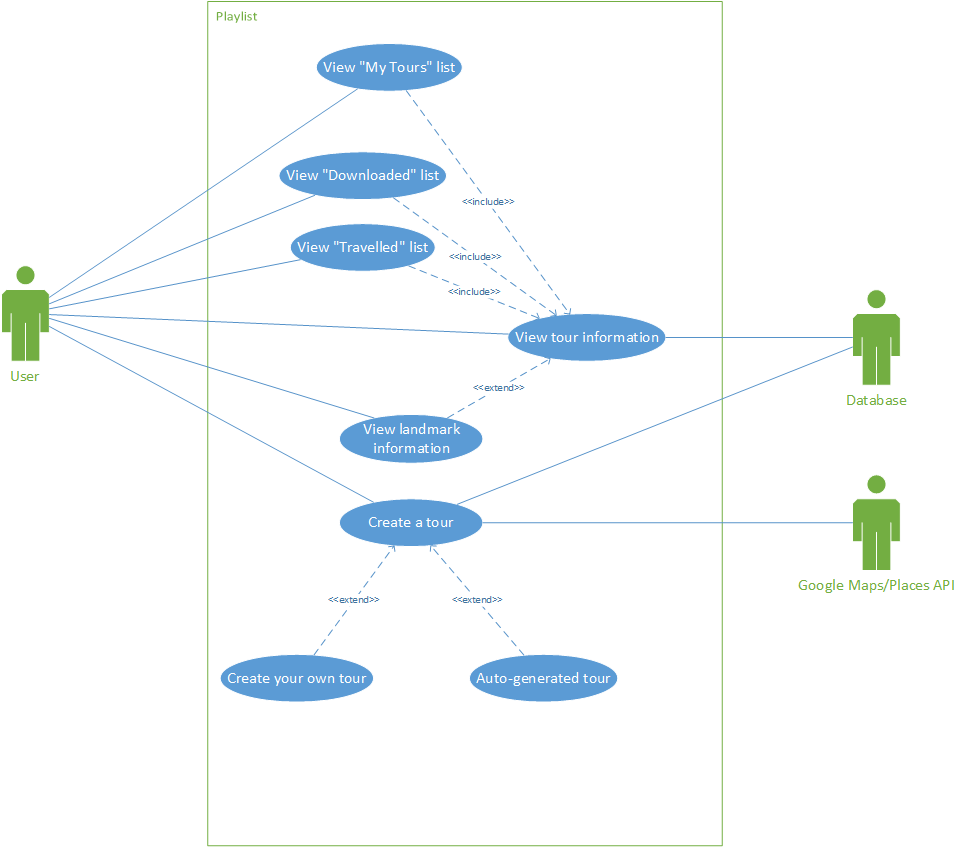


**State Diagram**

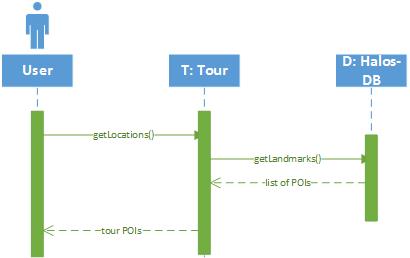
**Store Use-Case Diagram**



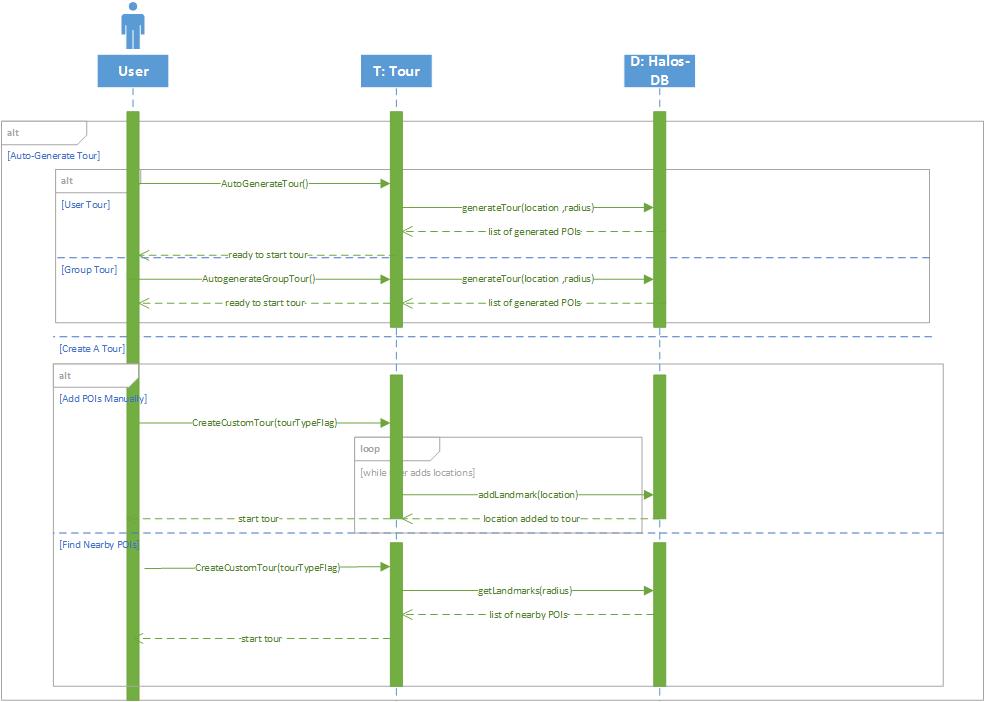
**Playlist Use Case Diagram**



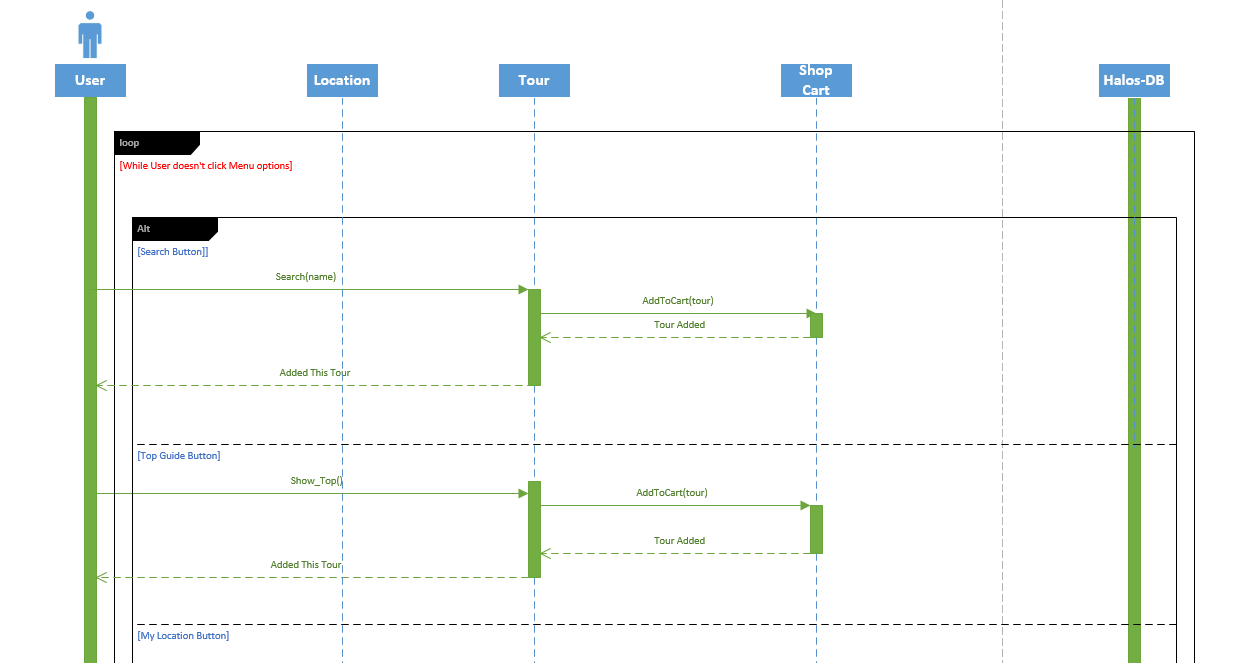
**View Tour Sequence Diagram**



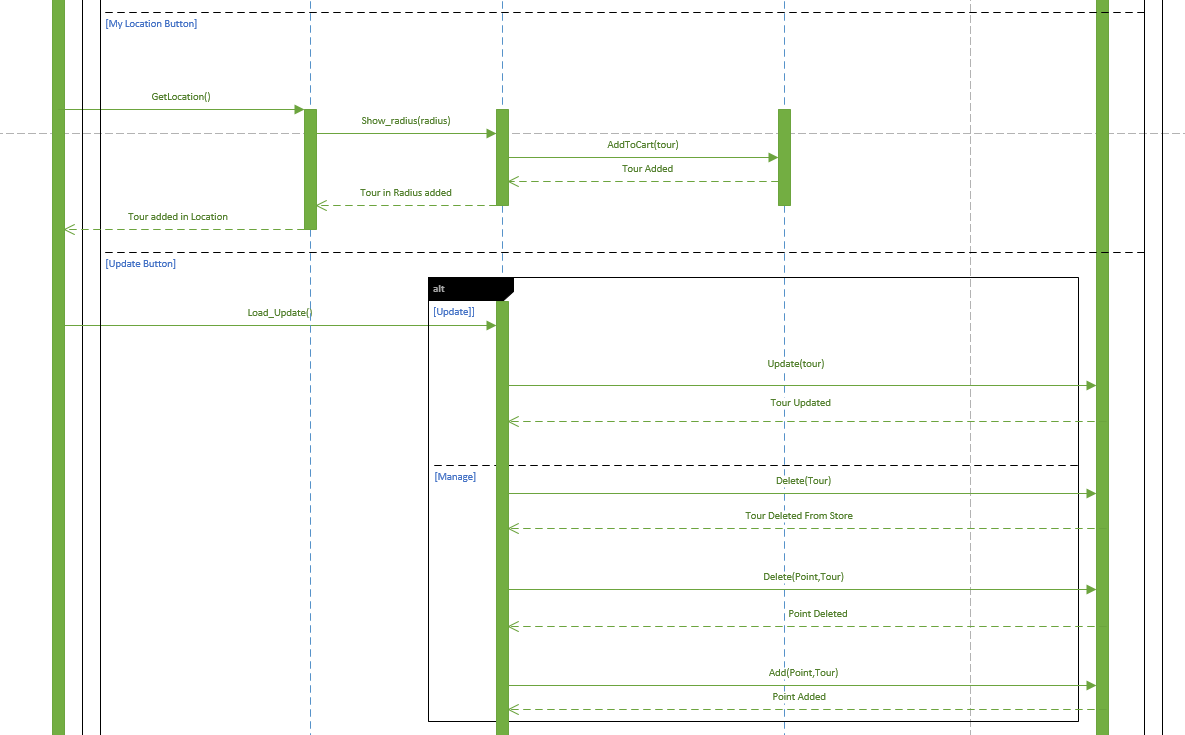
**Create Tour Sequence Diagram**



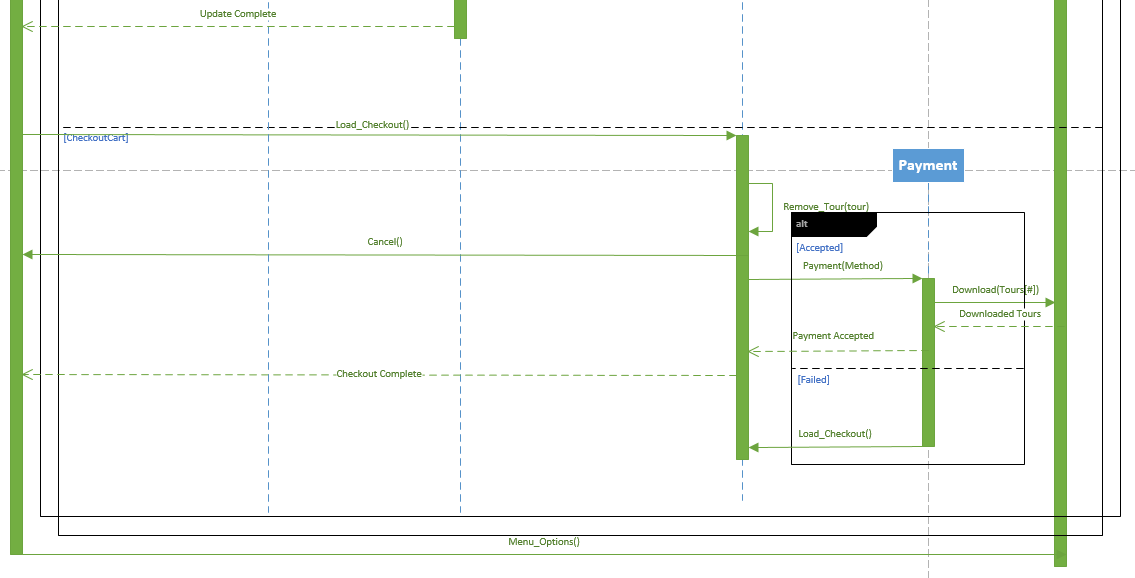
**Store Sequence Diagram**



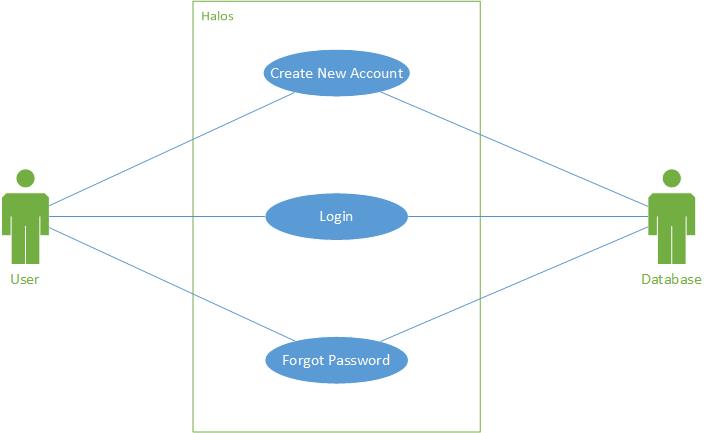
**Middle Part**



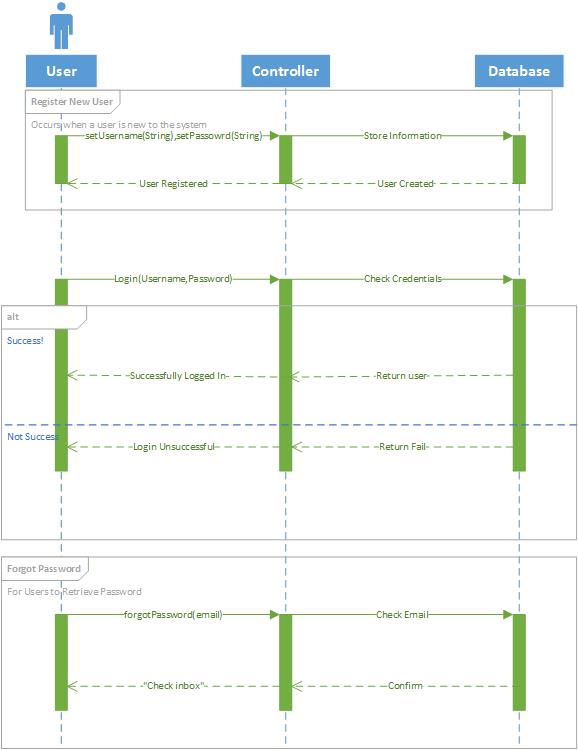
**Bottom Part**



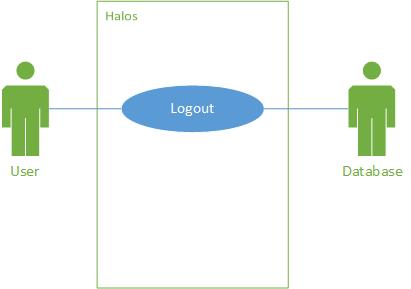
**Login Use Case Diagram**



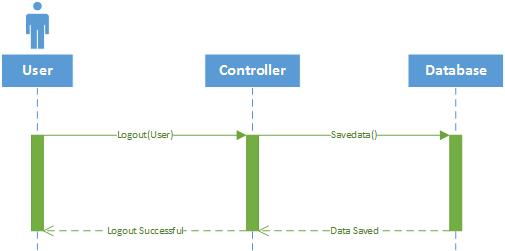
**Login Sequence Diagram**

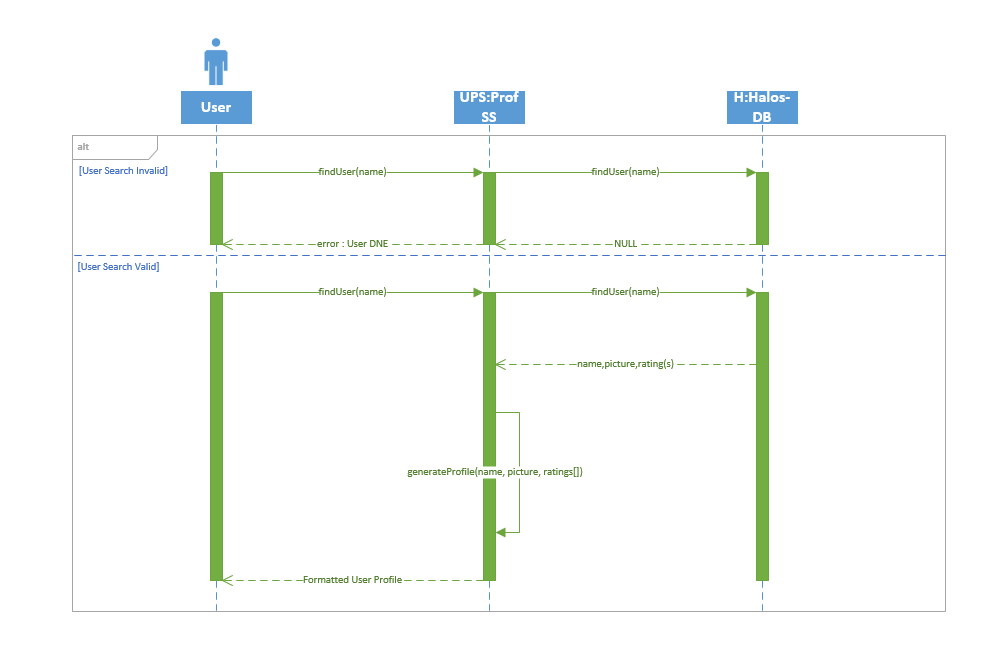


**Logout Use Case**



**Logout Sequence Diagram**



**User Search Sequence Diagram**

**User Profile Use Case Diagram**