

Hold It

"The Line Holding App"

Requirements Analysis Document

Team RKJ

Written By: Kegan Bako-Listy

Joseph Jindrich

Ryan Moore

Table of Contents

1. Introduction
 - 1.1. Purpose of the System
 - 1.2. Scope of the System
 - 1.3. Objectives and Success Criteria of the Project
 - 1.4. Definitions, Acronyms, and Abbreviations
2. Proposed system
 - 2.1. Overview
 - 2.2. System models
 - 2.2.1. Use Case Diagram
 - 2.2.2. Use case models
 - 2.2.2.1. Login
 - 2.2.2.2. Logout
 - 2.2.2.3. Account Creation
 - 2.2.2.4. Password Forgotten
 - 2.2.2.5. Delete Account
 - 2.2.2.6. Become Holdee
 - 2.2.2.7. Clients Chatting
 - 2.2.2.8. Edit Customer Account
 - 2.2.2.9. Edit Holdee Account
 - 2.2.2.10. Payment
 - 2.2.2.11. Event Creation
 - 2.2.2.12. Holdee Posts
 - 2.2.2.13. Customer Requests
 - 2.2.2.14. Buy Spot
 - 2.2.2.15. Dispute from Holdee
 - 2.2.2.16. Dispute from Customer
 - 2.2.2.17. Customer Files Complaint During Transaction
 - 2.2.2.18. Search by Tag
 - 2.2.2.19. Search by Address
 - 2.2.2.20. Search by Proximity
 - 2.2.2.21. Search for Parking
 - 2.2.2.22. Parking Event Creation
 - 2.2.2.23. Parking Purchase
 - 2.2.2.24. Posting Parking
 - 2.2.2.25. Holdee Loses GPS Location
 - 2.2.2.26. Holdee Cancels Transaction
 - 2.2.2.27. Holdee Cancellation Protocol
 - 2.2.3. Class Diagram
 - 2.2.4. Sequence Diagrams
 - 2.2.4.1. Event Creation
 - 2.2.4.2. Pay
 - 2.2.4.3. Account Creation
 - 2.2.4.4. Search
 - 2.2.5. State Diagrams

- 2.2.5.1. Delete Account
- 2.2.5.2. Chat
- 2.2.5.3. Report
- 2.2.5.4. Login/Logout
- 2.2.5.5. Forgot Password
- 2.2.5.6. Update Account

1 Introduction

1.1 Purpose of the System

The Purpose of the Hold it application is to create an automated system in which spots in lines can be held in exchange for monetary compensation. Customers will be able to either purchase a spot in a line as well as wait in line at the behest of someone else. This application will help to eliminate what many consider to be the worst part of events like purchasing tickets, getting a new phone, or even getting a spot for black friday.

1.2 Scope of the System

This project will consist of making a usable application for line holding. The Holdit application will be finished by June, 2017. The application will be usable for both clients who want to be paid to wait in line as well as ones who want to pay for spots in a line.

1.3 Objectives and Success Criteria of the Project

For the project to be successful this application will: Allow clients to purchase spots in line, allow clients to hold spots in line, allow clients to pay each other, have a functioning search by map, and allow clients to securely interact with each other.

1.4 Definitions, Acronyms, and Abbreviations

Holdee: A client who is holding a spot in line.

Customer: A client who is purchasing a spot in line

Tag: Label attached to an event to help with identification.

Spot: An offer to hold a spot at an event.

Tags: Identifiers to find an event, such as “Phone” or “Amiibo”.

2.2.1 Use Case Diagram:



2.2.2 Use Case Models

2.2.2.1 Login

Actors	<ul style="list-style-type: none">• Customer
--------	--

Flow of Events	<ol style="list-style-type: none"> 1. Customer opens the Hold It app on their phone. 2. System responds by displaying login screen to the Customer. 3. Customer enters their email address into the email address field and their password into the password field. 4. Customer pushes the login button.
Entry Condition	<ul style="list-style-type: none"> • Customer wants to use Hold in app
Exit Conditions	<ul style="list-style-type: none"> • Customer is logged into Hold It • Email address and password combination not recognized.
Quality Requirements	<ul style="list-style-type: none"> • Login confirmation or deny should take less than two seconds.

2.2.2.2 Logout

Actors	<ul style="list-style-type: none"> • Customer
Flow of Events	<ol style="list-style-type: none"> 1. Customer opens the Hold It app and pushes the profile button. 2. System displays the profile page. 3. From the Profile page the Customer pushes the settings gear. 4. System displays the settings page 5. From the settings page the Customer pushes the logout button 6. System displays logout confirmation. 7. Customer pushes yes on the logout confirmation. 8. System deactivates the Customer's account.
Entry Condition	<ul style="list-style-type: none"> • Customer no longer wants to use Hold It
Exit Conditions	<ul style="list-style-type: none"> • Customer logout complete. • Hold It application is closed.
Quality Requirements	<ul style="list-style-type: none"> • Logout confirmation takes no longer than 10 seconds

2.2.2.3 Account Creation

Actors	<ul style="list-style-type: none"> • Customer
Flow of Events	<ol style="list-style-type: none"> 1. Customer activates create account option on their phone. 2. System responds by displaying a form to the Customer. 3. Customer fills out email, username, password of minimum 8 characters and 1 symbol, password confirmation. 4. System receives the form and checks that the email/username is not already being used as well as checks the password meets the requirements.

	<ol style="list-style-type: none"> 5. System displays payment form to the customer 6. Customer fills out payment address and card information. 7. System confirms valid payment type. 8. System presents customer with terms and services. 9. Customer reads and accepts terms. 10. System sends confirmation of account creation email.
Entry Condition	<ul style="list-style-type: none"> • The customer wants to create an account
Exit Conditions	<ul style="list-style-type: none"> • Customer receives confirmation email saying account was successfully created. • Customer's application is terminated. • Customer's email is already in use.
Quality Requirements	<ul style="list-style-type: none"> • System takes no longer than 10 seconds to confirm email and username are unique and password meets requirements. • System takes no longer than 10 seconds to confirm payment information. • System takes no longer than 10 seconds to send confirmation email.

2.2.2.4 Password Forgotten

Actors	<ul style="list-style-type: none"> • Customer • Holdee
Flow of Events	<ol style="list-style-type: none"> 1. Customer/Holdee chooses forgot password from start screen. 2. System prompts customer with email address. 3. System confirms email address and matches with account. 4. System resets password to a randomly generated phrase. 5. System sends new password in email to Customer/Holdees email address. 6. System displays message to Customer/Holdee "Your new password has been sent to your email address."
Entry Condition	<ul style="list-style-type: none"> • Customer/Holdee Attempts to login but can no longer remember their password • Customer attempts to create an account but their email is already in use.
Exit Conditions	<ul style="list-style-type: none"> • New password is sent to customers/holdees email. • Customer/holdee terminates application.
Quality Requirements	<ul style="list-style-type: none"> • System takes no longer than 10 seconds to confirm email, generate new password, and send email.

2.2.2.5 Delete Account

Actors	<ul style="list-style-type: none">• Customer
Flow of Events	<ol style="list-style-type: none">1. Customer opens System.2. Customer taps Profile3. Customer taps Setting4. Customer taps delete account5. System prompts if Customer is sure6. Customer taps yes7. System shows Customer delete specifications8. Customer accepts9. Customer types in password twice10. Customer taps delete account11. If Customer is Holdee it stores the Social Security of the Holdee, and employment records.12. System deletes all information tied to account.
Entry Condition	<ul style="list-style-type: none">• Customer wants to delete account
Exit Conditions	<ul style="list-style-type: none">• Customer taps No at any time• Customer terminates System• Customer deletes account
Quality Requirements	<ul style="list-style-type: none">• Takes no longer than 10 seconds to send email confirmation• Takes no longer than 10 seconds per tap

2.2.2.6 Become Holdee

Actors	<ul style="list-style-type: none">• Customer• Admin• Holdee
Flow of Events	<ol style="list-style-type: none">1. A Customer selects become Holdee button in the profile page.2. System responds by presenting a form to the Customer.3. The Customer completes the form by entering their social security number, legal name, DOB, address, phone number, and banking information (routing number and account number?)4. System confirms the given information is valid and that the social security number is not already in use.5. System sends contact information, name, and age to Admin6. Admin receives information, looks it over, contacts Customer via email and decides to allow Customer to become a Holdee.7. Admin chooses Accept button on System application.8. System sends confirmation to email to Holdee.

Entry Condition	<ul style="list-style-type: none"> • Customer decides they would like to become a Holdee.
Exit Conditions	<ul style="list-style-type: none"> • Admin confirms the request and System send confirmation email to Holdee. • Admin denies the Customer request. • System application is terminated
Quality Requirements	<ul style="list-style-type: none"> • System takes no longer than 5 seconds to confirm confirmation email.

2.2.2.7 Chat

Actors	<ul style="list-style-type: none"> • Customer • Holdee
Flow of Events	<ol style="list-style-type: none"> 1. Customer or Holdee pushes the profile button. 2. System displays the profile screen. 3. Customer or Holdee pushes the open chat button. 4. System opens up the chat display. 5. Customer or Holdee types messages into textbox 6. System posts the message and notifies the recipient.
Entry Condition	<ul style="list-style-type: none"> • Customer/Holdee want to chat
Exit Conditions	<ul style="list-style-type: none"> • Hold It application is closed. • Customer or Holdee close chat
Quality Requirements	<ul style="list-style-type: none"> • Messages are posted two seconds after sending. • Customers or Holdees are notified within 10 seconds that a new messages is posted.

2.2.2.8 Edit Customer Account

Actors	<ul style="list-style-type: none"> • Customer
Flow of Events	<ol style="list-style-type: none"> 1. Customer pushes profile button. 2. System displays profile screen. 3. Customer pushes settings gear. 4. System displays the settings page. 5. Customer pushes change password. 6. System sends change password to the email of the Customer. 7. Customer pushes midnight skin. 8. System changes the color of the app to a darker theme. 9. Customer pushes payment information. 10. System displays payment information. 11. Customer changes payment information. 12. Customer pushes logout

	13. System runs the logout use case
Entry Condition	<ul style="list-style-type: none"> Customer wants to change account settings, including login information, payment information, and skin.
Exit Conditions	<ul style="list-style-type: none"> Customer exits the settings page. Hold It application is terminated.
Quality Requirements	<ul style="list-style-type: none"> Password change form is sent less than a minute after the request is put in.

2.2.2.9 Edit Holdee Account

Actors	<ul style="list-style-type: none"> Holdee
Flow of Events	<ol style="list-style-type: none"> Holdee opens app Holdee taps profile Holdee taps change to Holdee Holdee taps Settings gear Holdee taps change address <ol style="list-style-type: none"> System makes address field open to changes Holdee enters new address information Holdee taps save System saves new info Holdee taps change Banking information <ol style="list-style-type: none"> System displays current bank information Holdee taps change account info System tells Customer current info will be deleted, and that they cannot act as a Holdee until payment information Holdee taps change legal name <ol style="list-style-type: none"> System asks Holdee if they'd like to confirm Holdee taps yes. System emails Holdee information and documents to change name. Holdee taps change phone number <ol style="list-style-type: none"> System opens entry field to change phone number Holdee enters numberwhat Holdee taps confirm
Entry Condition	<ul style="list-style-type: none"> Holdee wishes to change account information, including login information, payment information, and skin.
Exit Conditions	<ul style="list-style-type: none"> Holdee has finished changing their account information. Hold It application is closed.
Quality Requirements	<ul style="list-style-type: none"> Changes saved within 2 seconds of clicking save

2.2.2.10 Payment

Actors	<ul style="list-style-type: none">● Holdee● Customer
Flow of Events	<ol style="list-style-type: none">1. Customer agrees with fee from Holdee and selects purchase.2. System prompts client with last 4 digits of card and asks to confirm payment.3. System confirms payment.4. System holds payment until notified otherwise.5. System sends notification to Holdee announcing they have been selected and prompts Holdee with form.6. System sends confirmation code to Customer.7. Customer approaches Holdee and presents them with code.8. Holdee fills out form with provided code.9. System confirms code and processes payment.10. Holdee receives payment.
Entry Condition	<ul style="list-style-type: none">● Customer wants to buy the Holdee's spot
Exit Conditions	<ul style="list-style-type: none">● Holdee receives payment● Payment is cancelled.● Time expires.
Quality Requirements	<ul style="list-style-type: none">● Confirmation of code takes no longer than 2 seconds.● Holdee receives purchase confirmation within 5 seconds.

2.2.2.11 Event Creation

Actors	<ul style="list-style-type: none">● Customer● Holdee● Admin
Flow of Events	<ol style="list-style-type: none">1. Customer pushes the post button.2. System displays the post screen.3. Customer enters in the name, date, time, address, and any additional comments into the post screen and pushes "Ask for Holdee" or "Create Event" if an Admin or Holdee is creating the event.4. System checks to see if an event is already created at that address around the same time.5. If the event is a duplicate System tells the Customer that a similar event already exists and asks if they would like to be taken to that event page.6. If the event is new System creates a new event page for it and displays the event on the Customers page.

Entry Condition	<ul style="list-style-type: none"> • Customer wishes to create an event.
Exit Conditions	<ul style="list-style-type: none"> • The new event is created. • The event already exists. • The app is terminated.
Quality Requirements	<ul style="list-style-type: none"> • New events are posted less than two minutes after creation.

2.2.2.12 Holdee Posts

Actors	<ul style="list-style-type: none"> • Holdee
Flow of Events	<ol style="list-style-type: none"> 1. Holdee opens Hold It. 2. System displays the home screen. 3. Holdee finds an event they want to wait in line for and goes to the event. 4. Holdee gets in the line at the event and checks in to the event. 5. System confirms via GPS that they are at the event location and displays the line display. 6. Holdee enters in their position in line and the asking price for their spot in line. 7. System adds the Holdee to the pool of Holdees at the event.
Entry Condition	<ul style="list-style-type: none"> • Holdee wants to find an event to wait in line for.
Exit Conditions	<ul style="list-style-type: none"> • Holdee enters the line of the event they chose.
Quality Requirements	<ul style="list-style-type: none"> • GPS needs to confirm the Holdee's location. • Holdee needs to be added to the pool of Holdee's at the event less than a minute after they check in.

2.2.2.13 Customer Requests

Actors	<ul style="list-style-type: none"> • Customer • Holdee
Flow of Events	<ol style="list-style-type: none"> 1. Customer opens the Hold It app. 2. System displays the home screen. 3. Customer finds an event they want to buy a spot for but there currently are no Holdees at the event. 4. Customer pushes the "Request Holdee" button to be added to the group of customers for that event. 5. When a Holdee checks in to the event System notifies the Customer that there is a spot they can buy.

Entry Condition	<ul style="list-style-type: none"> Customer wants to request a spot at an event.
Exit Conditions	<ul style="list-style-type: none"> Customer requests a spot at an event. Customer buys a spot in line.
Quality Requirements	<ul style="list-style-type: none"> Customers are notified of a Holdee at a first come first serve basis, with the notification moving on a minute after notifying the customer.

p

2.2.2.14 Buy Spot

Actors	<ul style="list-style-type: none"> Holdee Customer
Flow of Events	<ol style="list-style-type: none"> 1. Customer opens the Hold It app. 2. System displays the home screen. 3. Customer finds an event they want to buy a spot for and pushes it. 4. System displays the page for the event. 5. Customer taps on "Available Holdees". 6. System displays the Holdees and their positions in line that are not already purchased. 7. Customer selects the Holdee they want to buy their spot from. 8. System displays the information on the spot, and its price. 9. Customer pushes "Buy Spot". 10. System displays a confirmation page. 11. Customer confirms their purchase. 12. System notifies the Holdee that their spot has been purchased, generates a code for the Customer to give to the Holdee, and removes the Holdee's spot from the event page.
Entry Condition	<ul style="list-style-type: none"> Customer wants to buy a spot in line for an event.
Exit Conditions	<ul style="list-style-type: none"> Customer buys a spot in line.
Quality Requirements	<ul style="list-style-type: none"> Holdee is notified less than 30 seconds after their spot it bought.

2.2.2.15 Dispute from Holdee

Actors	<ul style="list-style-type: none"> Admin Holdee Customer
Flow of Events	<ol style="list-style-type: none"> 1. Holdee presses report button from transaction page. 2. System responds by presenting the Holdee with a form.

	<ol style="list-style-type: none"> 3. Holdee finishes the form by giving a description of the dispute and uploading any photos she has taken. 4. System sends the completed form, the Holdees and Customers GPS history since the purchase was made, and their entire conversation history to an Admin 5. The Admin reviews all of the provided information and decides to add a strike to the Customers strike count.
Entry Condition	<ul style="list-style-type: none"> • Holdee has an issue with a customer.
Exit Conditions	<ul style="list-style-type: none"> • The Customer is banned from System by having their account deleted. • The Customer has a strike added to their account by the Admin. • The Admin decides to give no strike or ban to the Customer. • The Holdees application is terminated.
Quality Requirements	<ul style="list-style-type: none"> • The form, gps history, and conversation history take no longer than 15 seconds to be sent from System to the admin.

2.2.2.16 Dispute from Customer

Actors	<ul style="list-style-type: none"> • Holdee • Customer • Admin
Flow of Events	<ol style="list-style-type: none"> 1. Customer presses report button from transaction page. 2. System responds by presenting the Customer with a form. 3. Customer finishes the form by giving a description of the dispute and uploading any photos he has taken. 4. System sends the completed form, the Holdees and Customers GPS history since the purchase was made, and their entire conversation history to an Admin 5. The Admin reviews all of the provided information and decides to ban the Holdee by deleting their account and marking their SSN with a flag.
Entry Condition	<ul style="list-style-type: none"> • Customer has an issue with the Holdee
Exit Conditions	<ul style="list-style-type: none"> • The Holdee is banned from System by the Admin. • The Holdee has a strike added to their account by the Admin. • The Admin decides to give no strike or ban to the Holdee. • The Customer's application is terminated.
Quality Requirements	<ul style="list-style-type: none"> • The form, gps history, and conversation history take no longer than 15 seconds to be sent from System to the admin.

2.2.2.17 Customer Files Complaint During Transaction

Actors	<ul style="list-style-type: none">• Admin• Customer• Holdee
Flow of Events	<ol style="list-style-type: none">1. Customer currently has an open transaction2. Customer opens System3. Customer taps Profile4. Customer taps contact us5. Customer taps report problem with transaction6. System asks the Customer if they are sure and this will cause the current transaction to be frozen.7. Customer taps yes8. System tells Customer that it is closing communication with Holdee and to expect to be contacted by an admin shortly9. System prompts Customer to give a short account of what happened10. System tells Holdee there is a complaint being filed with transaction and they are encouraged to take pictures of their surroundings and prepare for a dispute, and contact from admin11. System Holdee to give a short account of what happened12. System checks for available admins, and joins que for System transaction problems if full13. When Problem reaches an available admin, a chat is opened with the customer, and the Holdee separately.14. System sends GPS information of the Holdee to the Admin15. System sends transaction chat information to the Admin16. System sends both accounts of the situation to the Admin17. Admin works towards a resolution based on guidelines set by the company to reach a conclusion.
Entry Condition	<ul style="list-style-type: none">• Customer decides to report transaction
Exit Conditions	<ul style="list-style-type: none">• Conclusion is reached by Admin• Customer stops communicating
Quality Requirements	<ul style="list-style-type: none">• All information needs to be sent in under 10 seconds• All chat must be sent within 3 seconds

2.2.2.18 Search by Tag

Actors	<ul style="list-style-type: none">• Customer
Flow of Events	<ol style="list-style-type: none">1. Customer selects search by tag on System's search screen

	<ol style="list-style-type: none"> 2. System responds by displaying a singular text search screen. 3. The Customer begins to type 'Apple' in the search bar. 4. System displays tags beginning with 'A' sorted by popularity as soon as the Customer types their first letter. 5. The Customer sees their tag at the top of the list and selects it. 6. System responds by displaying all events that have the tag 'Apple' in it. 7. The Customer selects their event.
Entry Condition	<ul style="list-style-type: none"> • Customer decides search for a specific event on the System application.
Exit Conditions	<ul style="list-style-type: none"> • Customer's application is terminated. • Customer selects an event. • Customer chooses to explore a different part of System.
Quality Requirements	<ul style="list-style-type: none"> • System begins to show all tags that begin with same letters within 1 second of the Customer typing.

2.2.2.19 Search by Address

Actors	<ul style="list-style-type: none"> • Customer
Flow of Events	<ol style="list-style-type: none"> 1. Customer selects search by tag on System's search screen 2. System responds by displaying a singular text search screen. 3. Customer enters in a complete address. 4. System displays a map center around the inputted address extending ¼ mile in every direction. 5. The Customer selects the event on their inputted address.
Entry Condition	<ul style="list-style-type: none"> • Customer decides search for a specific address on the System application.
Exit Conditions	<ul style="list-style-type: none"> • Customer's application is terminated. • Customer selects an event. • Customer chooses to explore a different part of System. • Customer enters invalid address.
Quality Requirements	<ul style="list-style-type: none"> • System begins to show all tags that begin with same letters within 1 second of the Customer typing.

2.2.2.20 Search by Proximity

Actors	<ul style="list-style-type: none"> • Customer
--------	--

Flow of Events	<ol style="list-style-type: none"> 1. Customer selects search by proximity on the search page of System. 2. System requests the GPS on the Customer's phone for the Customers location. 3. System displays a map showing approximately ¼ mile in each direction from the Customer's phone 4. The Customer scrolls around the map and finds an events at the apple store nearby them and selects it.
Entry Condition	<ul style="list-style-type: none"> • Customer decides to look for spots being held near by her location.
Exit Conditions	<ul style="list-style-type: none"> • Customer's application is terminated. • Customer selects an event. • Customer chooses to explore a different part of System.
Quality Requirements	<ul style="list-style-type: none"> • System takes no longer than 5 seconds to display the map around the Customer.

2.2.2.21 Search for Parking

Actors	<ul style="list-style-type: none"> • Customer
Flow of Events	<ol style="list-style-type: none"> 1. The Customer selects search for parking on System's search screen. 2. System requests the GPS on the Customer's phone for the Customers location. 3. System displays a map showing approximately ¼ mile in each direction from the Customer's phone 4. The Customer browses the map and selects a spot being held 2 blocks from them.
Entry Condition	<ul style="list-style-type: none"> • Customer decides to search for parking spaces.
Exit Conditions	<ul style="list-style-type: none"> • Customer selects a spot. • Customer's application is terminated. • Customer chooses to explore a different part of System.
Quality Requirements	<ul style="list-style-type: none"> • The map is displayed by System within 5 seconds of the customer selecting search for parking.

2.2.2.22 Parking Event Creation

Actors	<ul style="list-style-type: none"> • Holdee • Customer • Admin
--------	---

Flow of Events	<ol style="list-style-type: none"> 1. Holdee/Customer opens System 2. Holdee/Customer taps Post 3. Holdee/Customer selects parking 4. Holdee/Customer Types in address of parking lot 5. New parking lot is sent to Admin notifications. 6. Admin Checks address to verify Parking lot validity. 7. Admin taps Parking lot verified 8. Parking lot attached to address, and event created with 10 slots open.
Entry Condition	<ul style="list-style-type: none"> • Holdee posts new parking lot address • Customer posts new parking lot address
Exit Conditions	<ul style="list-style-type: none"> • Admin verifies that the address is not a parking lot.
Quality Requirements	<ul style="list-style-type: none"> • Admin should be notified within 20 seconds of posting

2.2.2.23 Parking Purchase

Actors	<ul style="list-style-type: none"> • Customer • Holdee
Flow of Events	<ol style="list-style-type: none"> 1. Customer Opens System 2. Customer taps on search 3. Customer puts address of parking lot into Search bar 4. System Parking lot event opens and displays number of spots 5. Customer selects parking spot 6. Customer selects purchase 7. Customer charged for Spot and 10 minute timer activated 8. Holdee notified 9. Customer receives picture of car in spot with an accept button. 10. Customer arrives at spot 11. Customer pushes “arrived” on app 12. Holdee notified and pop up on app appears. 13. Holdee taps “leaving spot”. 14. System transfers money to Holdee 15. System emails customer survey.
Entry Condition	<ul style="list-style-type: none"> • Customer wants to buy a parking spot
Exit Conditions	<ul style="list-style-type: none"> • Customer completes transaction • 10 minutes passes and customer does not claim spot
Quality Requirements	<ul style="list-style-type: none"> • Verification of payment happens within 10 seconds of payment

	<ul style="list-style-type: none"> • Less than 5 seconds between Customer pressing arrived and Holdee receiving arrived status window
--	--

2.2.2.24 Posting Parking

Actors	<ul style="list-style-type: none"> • Holdee
Flow of Events	<ol style="list-style-type: none"> 1. Holdee opens the app 2. Holdee selects profile 3. Holdee activates switch to Holdee 4. Holdee taps post 5. Holdee puts in parking lot address 6. Parking lot event pops up 7. Holdee posts parking spot there are less than 10 parking spots taken. 8. System verifies Holdee is in the correct GPS location 9. System posts spot in parking event
Entry Condition	<ul style="list-style-type: none"> • Holdee parking at parking spot.
Exit Conditions	<ul style="list-style-type: none"> • More than 10 spots currently taken • Parking lot not verified • Holdee terminates app
Quality Requirements	<ul style="list-style-type: none"> • Search of parking lot takes less than 5 seconds • Transition between pages takes less than 2 seconds

2.2.2.25 Holdee Loses Location

Actors	<ul style="list-style-type: none"> • Holdee
Flow of Events	<ol style="list-style-type: none"> 1. Holdee waiting for customer 2. GPS pings Holdee every 30 seconds 3. 2 GPS pings in a row are out of the 100 foot radius of the address 4. System messages Customer that they have 30 seconds to return to spot or fix phone before they lose their sale, and receive strike 5. Holdee not pinged in location 6. System puts strike on Holdee account 7. System submits email attached to strike number 8. System does Holdee cancellation protocol with Customer
Entry Condition	<ul style="list-style-type: none"> • Holdee pinged outside of acceptable area twice in a row
Exit Conditions	<ul style="list-style-type: none"> • Holdee pinged 3rd time

	<ul style="list-style-type: none"> • Holdee informed that they have been finished
Quality Requirements	<ul style="list-style-type: none"> • Pings happening every 30 seconds • Pings must be accurate within 10 feet of location

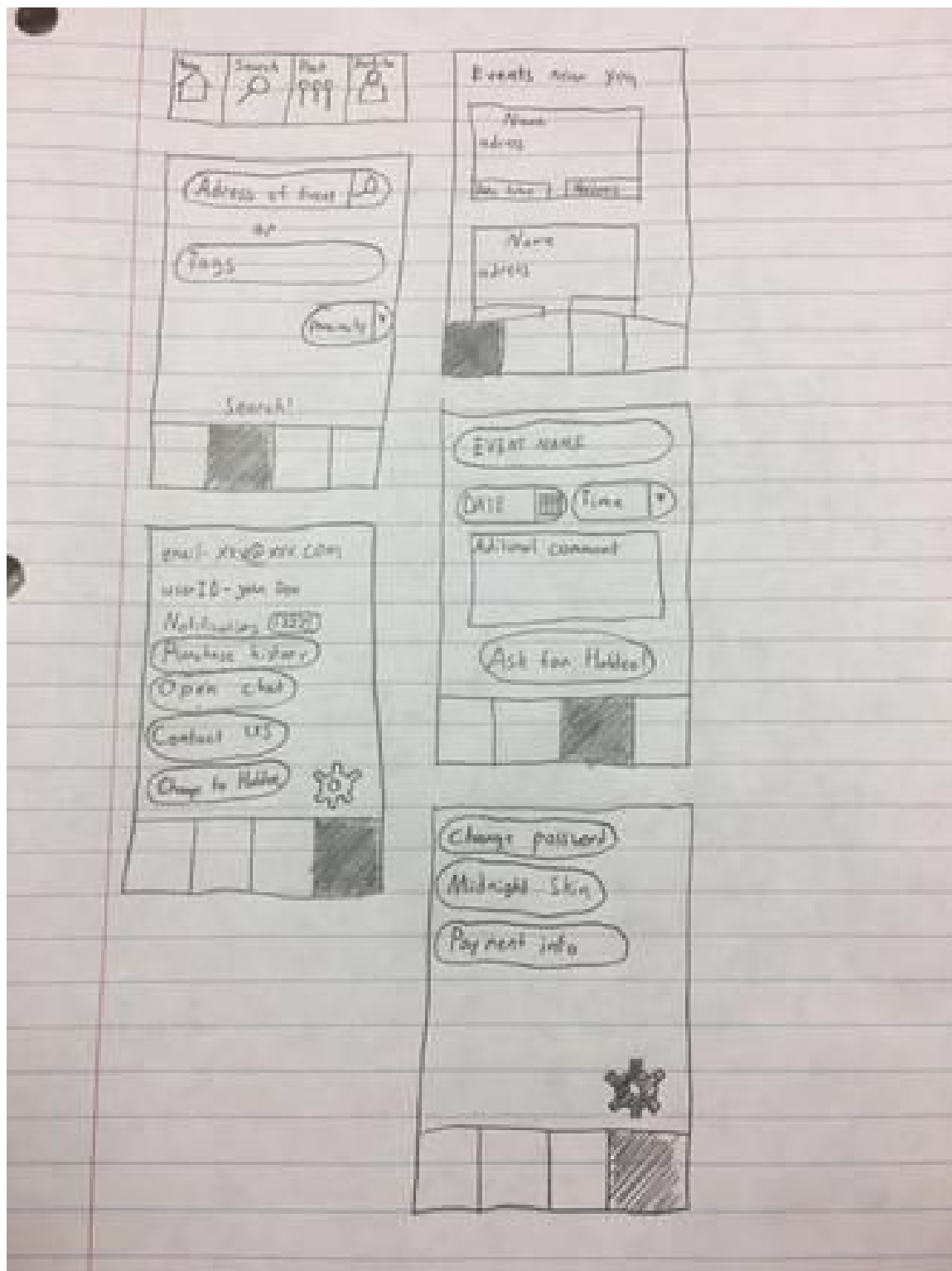
2.2.2.26 Holdee Cancels Transaction

Actors	<ul style="list-style-type: none"> • Holdee
Flow of Events	<ol style="list-style-type: none"> 1. Holdee taps cancel transaction 2. Holdee informed without ample reasoning they will receive a strike on their account. 3. Holdee taps I understand 4. Holdee Cancellation Protocol activates
Entry Condition	<ul style="list-style-type: none"> • In transaction Holdee needs to cancel Transaction
Exit Conditions	<ul style="list-style-type: none"> • Holdee states they do not want to cancel transaction
Quality Requirements	<ul style="list-style-type: none"> • Taps need to be registered in under a second

2.2.2.27 Holdee Cancellation Protocol

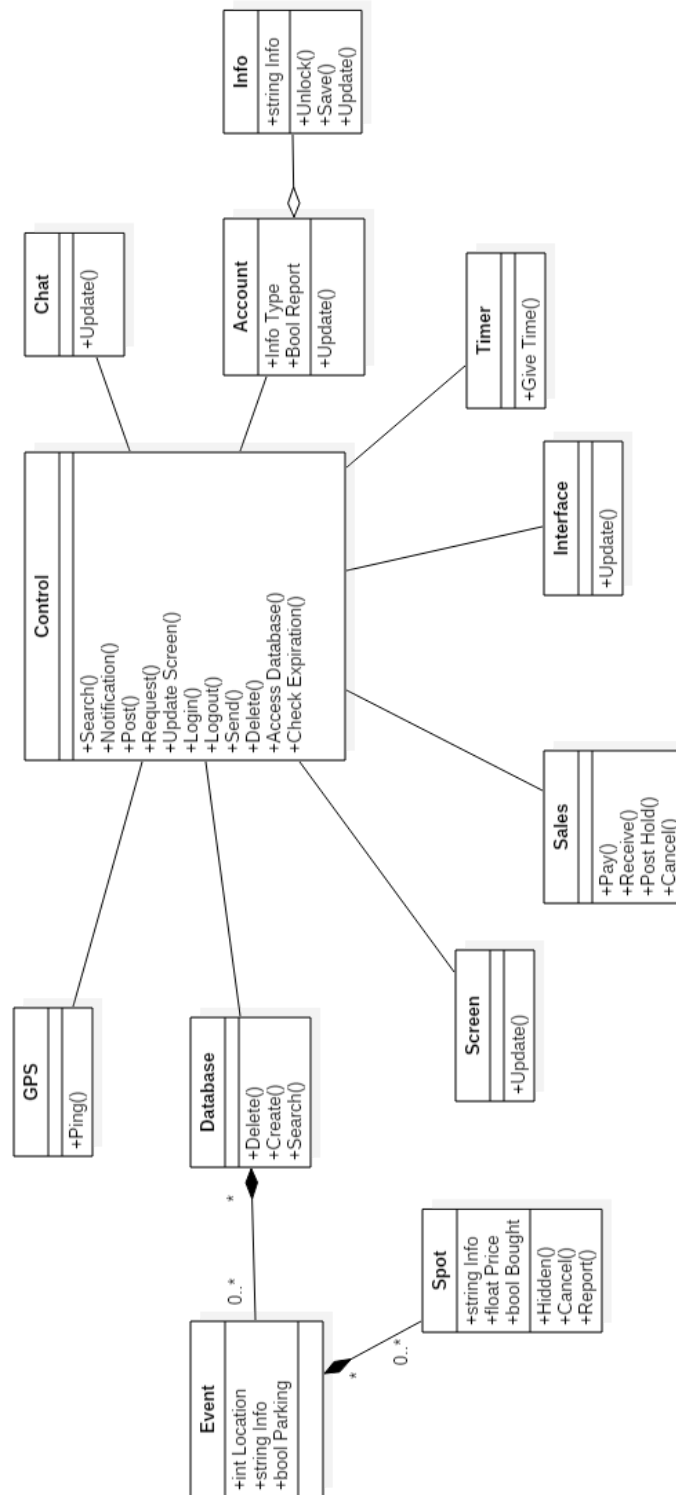
Actors	<ul style="list-style-type: none"> • Holdee • Customer
Flow of Events	<ol style="list-style-type: none"> 1. Holdee Cancels Parking after purchase 2. System checks if there is another Holdee in parking lot <ol style="list-style-type: none"> 2.1. There is another Holdee in parking lot <ol style="list-style-type: none"> 2.1.1. System sends Customer Notification that Holdee canceled purchase and asks if they'd like another spot and a refund of half the money 2.1.2. Holdee takes the spot and transaction sent to another Holdee, timer reset, and transaction continues 2.2. Customer doesn't take the spot, or not spots available and System sends a refund, and apology and a half off parking code. 3. Holdee Cancels Line holding after purchase and Holdee ahead of the canceled spot less than twice the original spot, or under 75\$ <ol style="list-style-type: none"> 3.1. Customer notified of cancellation by System 3.2. Customer informed of upgrade to spot upgrade for no charge.

	<p>3.3. New Holdee informed of purchase.</p> <p>3.4. Interaction continues as normal</p> <p>4. Holdee Cancels Line holding after purchase and no Holdee ahead of the canceled or all spots more than twice the original spot.</p> <p>4.1. Customer notified of cancellation by System</p> <p>4.2. Customer informed of possible downgrade and refund of difference plus 10% decrease in Downgrades price.</p> <p>4.3. System Prompts Customer if they'd like to accept transaction or receive refund.</p> <p>4.3.1. If Customer accepts</p> <p>4.3.2. Small refund is given</p> <p>4.3.3. System informs new Holdee.</p> <p>4.3.4. Transaction continues</p> <p>4.3.5. If Customer says No</p> <p>4.3.6. Full refund provided and apology sent with 10% discount code emailed to Customer</p> <p>5. No spots available in the line</p> <p>5.1. Customer informed of Cancellation</p> <p>5.2. System refunds Customer Money</p> <p>5.3. System sends apology email with free code for any spot up to the same price as the spot canceled.</p> <p>6. Customer doesn't respond to any prompts within the close time of the event</p> <p>6.1. Customer refunded</p> <p>6.2. System emails them 10% coupon and apology to Customer</p>
Entry Condition	<ul style="list-style-type: none"> • Holdee cancels a purchased spot.
Exit Conditions	<ul style="list-style-type: none"> • Customer finishes interaction • Customer interaction times out
Quality Requirements	<ul style="list-style-type: none"> • Search for new Holdee must take less than 2 seconds • Notification must happen within 2 seconds of cancellation • Refund must happen within 10 minutes of refund acceptance



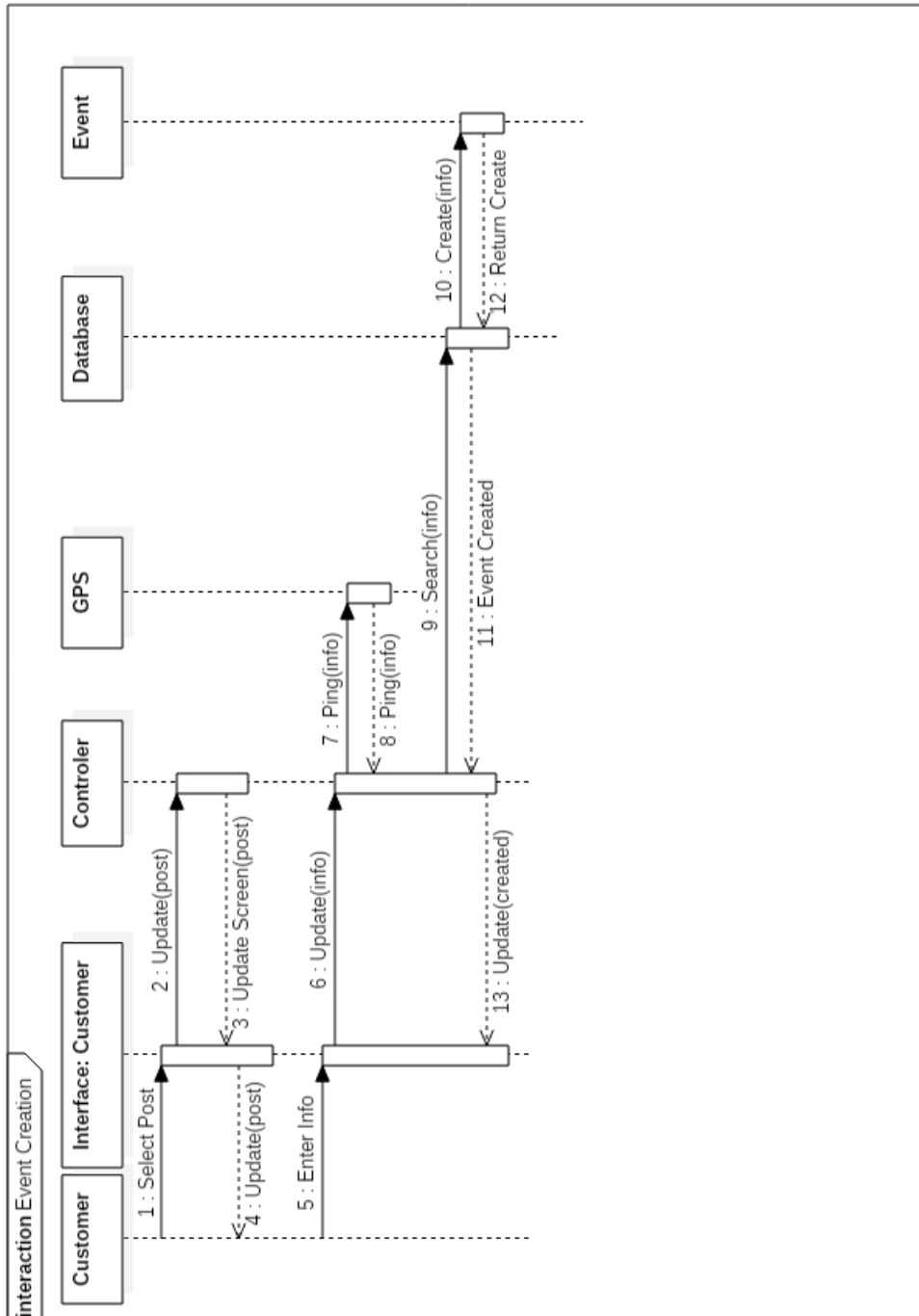
Storyboard

2.2.3 Class Diagram

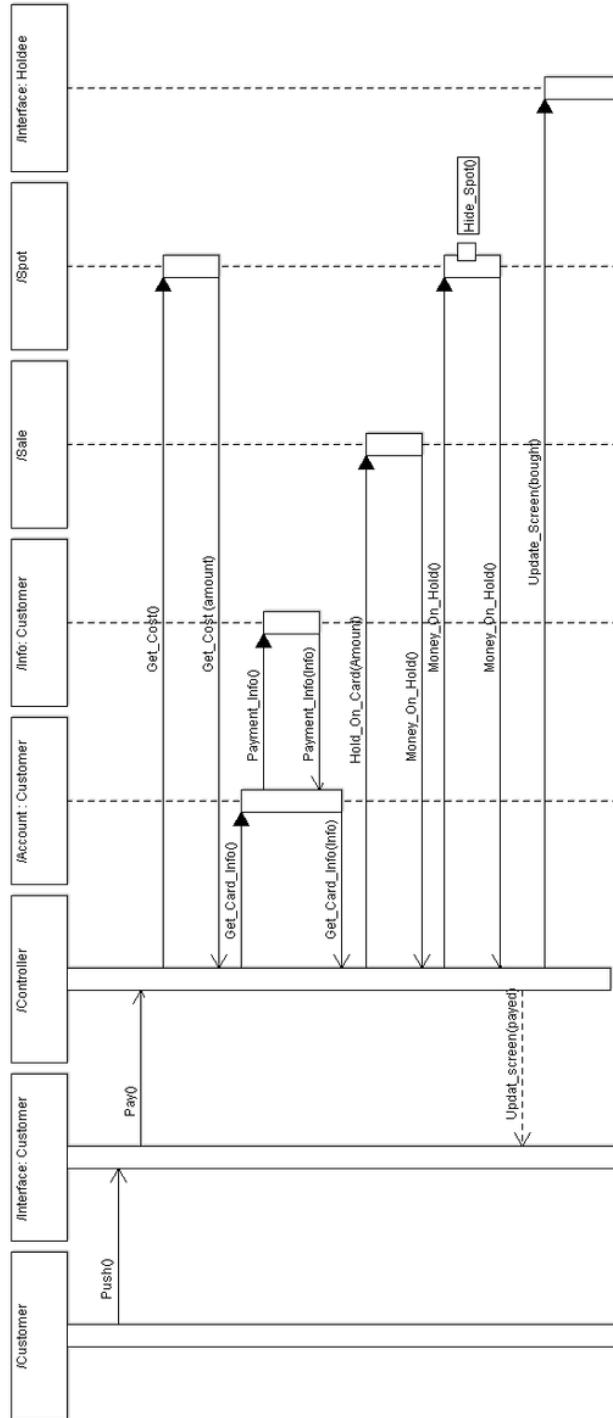


2.2.4 Sequence Diagrams

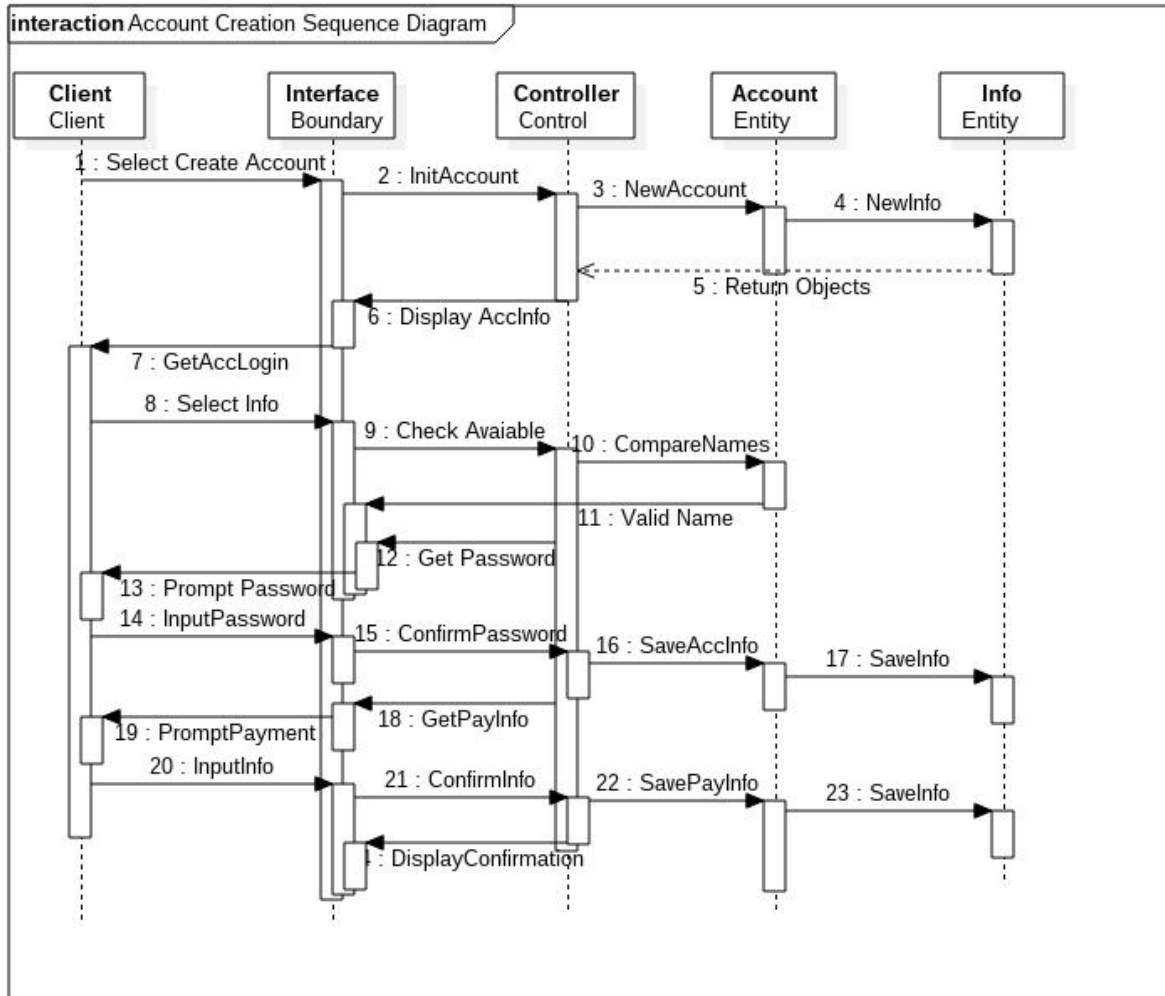
2.2.4.1 Event Creation



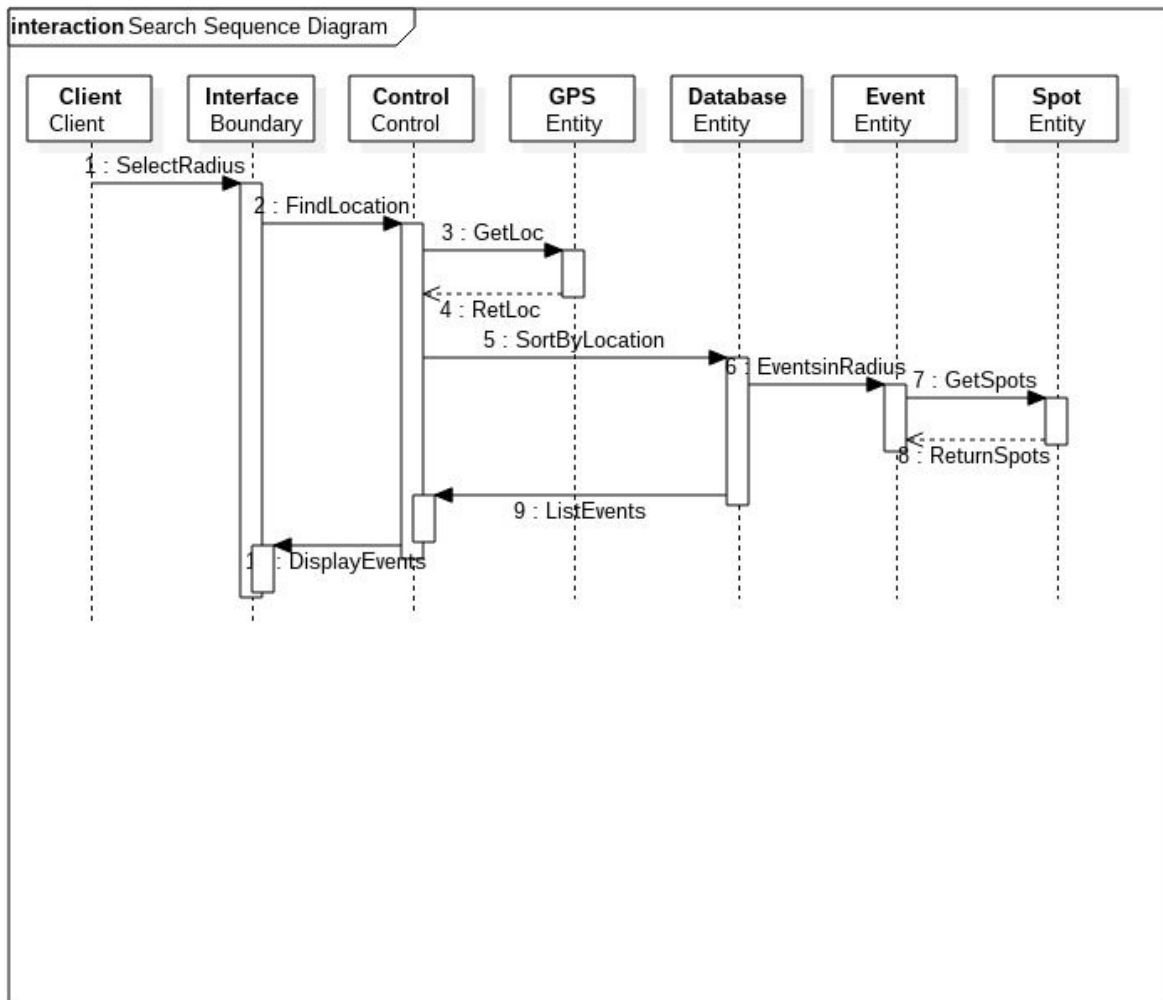
2.2.4.2 Pay



2.2.4.3 Account Creation

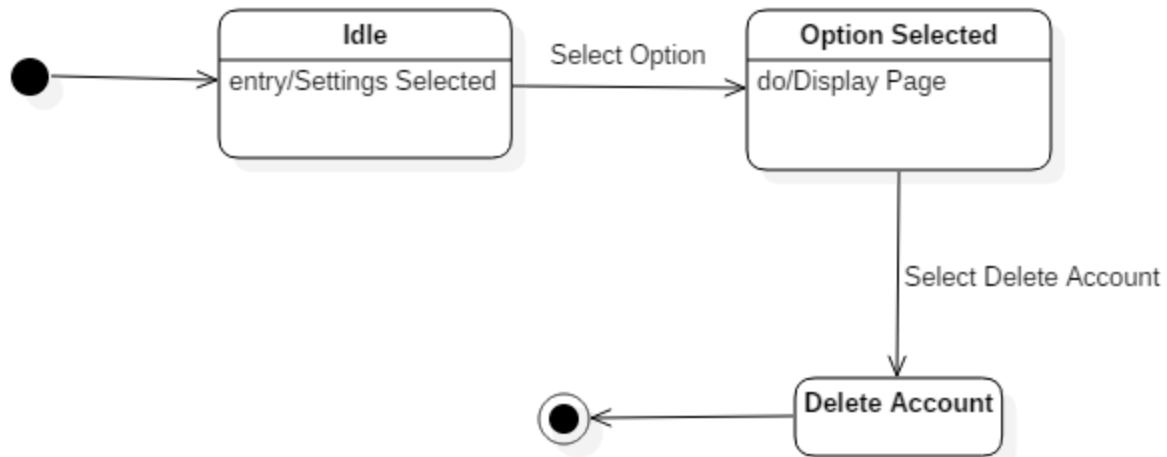


2.2.4.4 Search

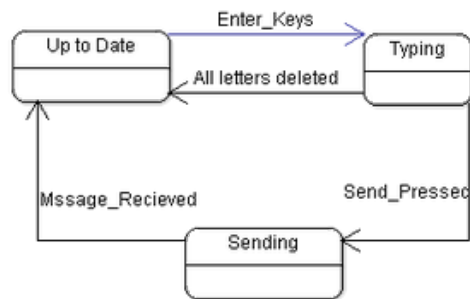


2.2.5 State Diagrams

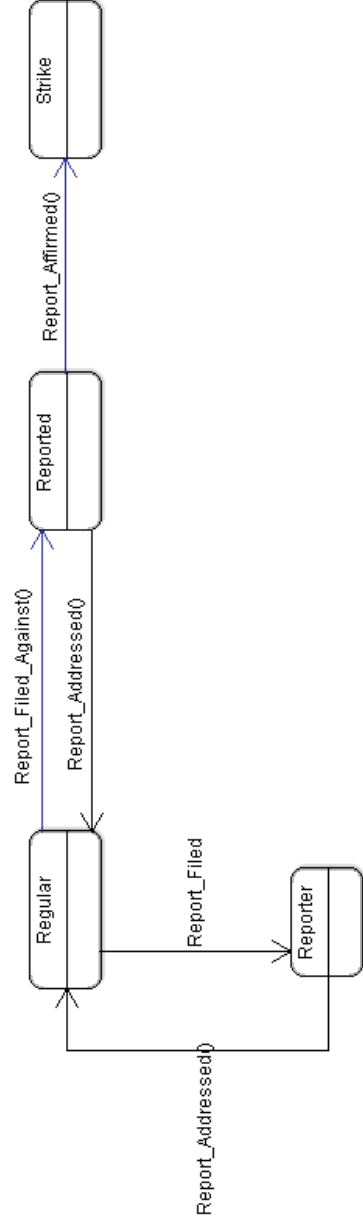
2.2.5.1 Delete Account



2.2.5.2 Chat

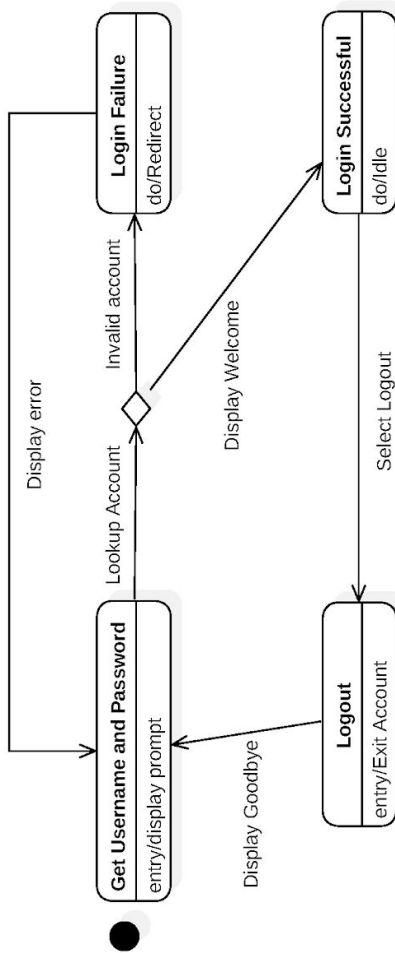


2.2.5.3 Report



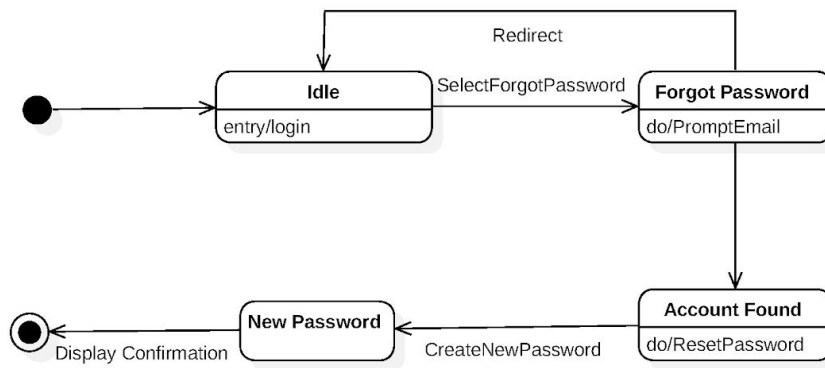
2.2.5.4 Account Verification

StateMachine1::Login/Logout State Diagram



2.2.5.5 Forgot Password

StateMachine1::Forgot Password State Diagram



2.2.5.6 Update Account

StateMachine1::Update Account State Diagram

