*Florida International University*

*School of Computing and Information Sciences*

CIS 4911 - Senior Capstone Project

Software Engineering Focus

Final Deliverable

Go Local Staff App

Team # 7

**Team Members**

Luis Castillo

Wilfredo Gomez

**Product Owner**: Eduardo Garcia

**Instructor**: Masoud Sadjadi

Copyright © by Go Local promotions Inc. and Florida International University.

All rights reserved. No part of this publication and material may be reproduced, distributed, or transmitted in any form or by any means, including photocopying, recording, or other electronic or mechanical methods, without the prior written permission of the owners.

***Abstract***

*The Deliverable of the Go Local App project is meant to provide the reader with all the necessary information regarding the system. The Go Local App is an iPhone application where individuals and organizations can find and post job offers respectively. The reader will be able to have a deeper understanding of the project in the introduction and some details of the plan taken to develop the aforementioned application.*

*The System provides employers, which need to hire talent in a relatively short notice, a way to post jobs that are visible to registered users in a predetermined amount of miles radius. Conversely, individuals are going to be able to see job post around them.*

Table of Contents

[Introduction 6](#_Toc437544205)

[Current System 6](#_Toc437544206)

[Purpose of New System 6](#_Toc437544207)

[User Stories 7](#_Toc437544208)

[Implemented User Stories 7](#_Toc437544209)

[Pending User Stories 11](#_Toc437544210)

[Project Plan 12](#_Toc437544211)

[Hardware and Software Resources 12](#_Toc437544212)

[Sprints Plan 14](#_Toc437544213)

[Sprint 1 14](#_Toc437544214)

[Sprint 2 16](#_Toc437544215)

[Sprint 3 18](#_Toc437544216)

[Sprint 4 18](#_Toc437544217)

[Sprint 5 20](#_Toc437544218)

[Sprint 6 20](#_Toc437544219)

[Sprint 7 21](#_Toc437544220)

[System Design 22](#_Toc437544221)

[Architectural Patterns 23](#_Toc437544222)

[System and Subsystem Decomposition 23](#_Toc437544223)

[Deployment Diagram 24](#_Toc437544224)

[Design Patterns 24](#_Toc437544225)

[System Validation 25](#_Toc437544226)

[Glossary 30](#_Toc437544227)

[Appendix 31](#_Toc437544228)

[Appendix A - UML Diagrams 31](#_Toc437544229)

[Static UML Diagrams 31](#_Toc437544230)

[Dynamic UML Diagrams 35](#_Toc437544231)

[Appendix B - User Interface Design 42](#_Toc437544232)

[Appendix C - Sprint Review Reports 63](#_Toc437544233)

[Appendix D - Sprint Retrospective Reports 65](#_Toc437544234)

[References 67](#_Toc437544235)

# Introduction

The rapid growth of demand for promotional talent imposes the need for robust tools not only to connect talent with employers but also connect employers with talent that are in that needed in order to execute events.

The following sections provide a description of the characteristics and limitations of the current system as well as a description of a proposed system that will address some of these limitations.

## Current System

The current system consists of staffing agencies. These staffing agencies act as an intermediate between the employers that are in need talent for their events and the talents that are need of jobs. In exchange for this service, agencies charge a substantial fee to the both parties, employers and talents. The agencies achieve this operation by communicating with employers and talent over the phone and email, input all relevant information on computers or written in paper, and lastly create all documents in paper or computer which are later printed signed by employers/talents and sent back to the agency thru email or fax.

Some of the current limitations of the system are:

* The User’s communication is limited to the hours of the operation of the staff agency
* Employers have to hire extra talent to anticipate unfortunate events in advance due to response time and availability of communication of the staffing agencies, which
* Talent booking and communication is subject to human error by staffing agencies employees

## Purpose of New System

The application will be able to allow users to register. There will be two different type of users. Namely company and staff. This is a relatively new approach to job seeking where a mobile application will allow you to either post job offers or look for ones depending if you are a company or staff, respectively.

The service will be location based, so it is tailored to the area where the acting user is.

# User Stories

User stories are an important part of agile software development because they are the basis for defining the functions of a system and simplify requirement management. User stories capture the what, who and why of a requirement in a simple way. The following sections provide a summary of the user stories implemented for this project as well as the pending user stories to be considered for future development.

## Implemented User Stories

**User Story # 672 - Create Web Login authentication**

As a user, I need to be able to authenticate with a secure username (email) and password to be used with web form.

Acceptance Criteria:

1. Create authentication system
2. Verify user email address
3. Authentication system (tables) must take into account future authentication techniques (e.g., Facebook, Gmail) but not implemented at this moment.

**User Story # 674 - Create Staff registration on IPhone**

As a user, I need to be able to register as a staff user using using the Go Local App so that I may able to use the Go Local App as staff user.

Acceptance Criteria:

1. Create a registration form for staff user
2. Allow users to register only if they didn't previously register on the web form and save their information on the database.
3. Test registration form data was properly inserted and updated using a pseudo authentication from database

**User Story # 676 - Create user account displays for App**

As a user, I need to be able to toggle between employer or staff user on the Go Local App so that I may be able to respectfully navigate on my staff or employer account.

Acceptance Criteria:

1. Create different Views for Staff and employer
2. Validate that users are registered for the account they are toggling/switching to.
3. Test that the validation process of the user account.

**User Story # 663 - Login on iPhone**

As a user, I would like to be able to login in to my existing account, so that I can take advantage of the services provided by the app.

Acceptance Criteria:

1. Implement login feature in the app.
2. User should be able to log in with an existing account.
3. Test login to validate user’s information.

**User Story # 679 - Forgot Password**

As a user, I need to be able to retrieve the password in case I forget it, so that I can use the app once I change it.

Acceptance Criteria:

1. User should be able to reset their password upon sms verification.
2. Test the sms verification for password reset.
3. Create documentation.

**User Story # 680 - Create a series of search filters for the employer home page**

As a user, I need to have a series of filters so I can narrow my search for talent.

Acceptance Criteria:

1. Create documentation.
2. Implement filters.
3. Test filters to ensure quality.

**User Story # 682 - Navigate Through Talent Vie and Get Information from talent**

As a user, I need to be able to navigate through a list of talents and upon selecting a talent get a detail view of information about the talent selected for the golocalapp iPhone application.

Acceptance Criteria:

1. Create a Talent View that list the talents from the data results of ‘search talent’
2. Create a Subview that details info about the current talent selected.
3. Create documentation on views and controllers
4. Test the views and controllers

**User Story # 699 - Create Employer Registration on iPhone**

As a user, I need to be able to register as an employer using the GoLocalApp iPhone application.

Acceptance Criteria:

1. Creation a registration form for employer user
2. Test registration form data was properly inserted and updated using a pseudo authentication form database
3. Add sms authentication

**User Story # 723 – Detailed profile View**

As a user, I need to be able to select a staff and see a detailed page of his or her profile using the GoLocalApp iPhone application.

Acceptance Criteria:

1. Create Documentation
2. Research usage of tableviews
3. Implement the detailed profile view

**User Story # 719 – Map View**

As a user, I need to be able to switch from the list of results of staff to a map showing the same amount of staff, so I can where they are in the map using the GoLocalApp iPhone application.

Acceptance Criteria:

1. Create Documentation on mapview
2. Do research on how to integrate maps using objective-c
3. Implement Map view
4. Update document with PIN information
5. Create PIN on Map
6. Research PIN on IOS Map

**User Story # 731 – Create Management Website**

As a user, I need to be able to able to log in to management web page and see a detailed page registered staff and employee using the GoLocalApp iPhone application.

Acceptance Criteria:

1. Create Management web page
2. Create Documentation
3. Update database schema

## Pending User Stories

**User Story # 671 - Create Web Staff Registration Form**

As a user, I need to have ability to store personal information on a database using a web form so that the information can be used later with the Go Local App.

Acceptance Criteria:

1. Create the web form with input fields, images, and documents
2. Create Database schema
3. Test data insertion, update and delete

# Project Plan

This project will be developed using agile development approach which insures that the project is constantly being reviewed and available to deploy at any time during its life cycle. By reviewing the project, the project can be re evaluated for the update of existing features and foreseen the addition of new features.

## Hardware and Software Resources

The following resources were used for this project:

**Digital Ocean**

Digital Ocean will be used the cloud provider to host server containing the database that will be used in the application. The following reasons led to the selection of this cloud provider among other cloud providers.

1. Options of pre-installed Linux distributions
2. Simple and reliable control panel to manage server
3. Its affordable pricing model, hourly or monthly
4. Storage type of server, SSD (Solid State Drives)

**Github**

Github will be used as the software source control tool.

**Gmail**

Gmail will be used as a communication tool.

**Google Drive**

Google Drive will be used as storage multi sharing development:

**Mac OS**

Mac OS will be use as the Operating system because it is the native environment for iPhone applications. moreover, A Mac OS is required in order to use other resources like Xcode.

**Mingle**

Mingle will be used as software management tool.

**MySQL**

MySQL will be used as the relational database. The following reasons led to the selection of MySQL amongst other relational databases:

1. Open-source relational database and its open source LAMP software stack
2. cross-platform support

**PHP**

PHP will be used as the server-side scripting language for the web development. The following reasons led to the selection of PHP:

1. The flexible of mixing with HTML code.
2. The flexibility of code style, object oriented and procedural

**Phpmyadmin**

Phpmyadmin will be used for managing MySQL databases:

**Sublime**

Sublime will be used as text editor to develop code.

**Visio 2013**

Visio will be used as the software modeling tool.

**Xcode**

Xcode will be used as the IDE to develop the IPhone application. The following reasons led to the selection of Xcode among other IDE’s:

1. User interface design, coding, testing and debugging features
2. contains iOS SDK’s, iOS simulator

## Sprints Plan

For each sprint, list the user stories selected for implementation in descending order of priority.

### Sprint 1

(08/31/2015 - 09/11/2015)

**User Story # 676 - Create User account displays for App**

***Tasks***

* Create toggle feature between different views (688)
* Create different views for staff and employer (687)
* Create Documentation (684)
* Read Objective-c Book (666)

***Acceptance Criteria***

* Create different Views for Staff and employer
* Validate that users are registered for the account they are toggling/switching to.
* Test that the validation process of the user account.

***Modeling***

Refer to UML diagrams in Appendix A that were created or modified to model the functionality that will be implemented in this sprint.

**User Story # 671 - Create Web form**

***Tasks***

* Create store process (692)
* Create Database Schema (691)
* Create Documentation (690)
* Read Objective-c Book (667)

***Acceptance Criteria***

* Create the web form with input fields, images, and documents
* Create Database schema
* Test data insertion, update and delete

***Modeling***

Refer to UML diagrams in Appendix A that were created or modified to model the functionality that will be implemented in this sprint.

**User Story # 672 - Create web login authentication**

***Tasks***

* Create Documentation (689)
* Integrate web form authentication (673)
* create email address authentication (670)

***Acceptance Criteria***

* Create authentication system
* Verify user email address
* Authentication system (tables) must take into account future authentication techniques (e.g., Facebook, Gmail) but not implemented at this moment

***Modeling***

Refer to UML diagrams in Appendix A that were created or modified to model the functionality that will be implemented in this sprint.

**User Story # 674 -Create Staff registration on IPhone**

***Tasks***

* Save registration form onto database (686)
* Create Staff Registration form on iPhone (685)
* Create documentation (677)
* Research on phone authentication (675)

***Acceptance Criteria***

* Create a registration form for staff user
* Allow users to register only if they didn't previously register on the web form and save their information on the database.
* Test registration form data was properly inserted and updated using a pseudo authentication from database

***Modeling***

Refer to UML diagrams in Appendix A that were created or modified to model the functionality that will be implemented in this sprint.

### Sprint 2

(09/14/2015 -09/25/2015)

**User Story # 674 -Create Staff registration on IPhone**

***Tasks***

* Update Financial Views (696)
* Update work experience views (695)
* Add Core Data (693)
* Save registration form onto database (686)
* Create Staff Registration form on iPhone (685)
* Create IPhone Authentication (678)
* Create documentation (677)
* Research on phone authentication (675)

***Acceptance Criteria***

* Create a registration form for staff user
* Allow users to register only if they didn't previously register on the web form and save their information on the database.
* Test registration form data was properly inserted and updated using a pseudo authentication from database

***Modeling***

Refer to UML diagrams in Appendix A that were created or modified to model the functionality that will be implemented in this sprint.

**User Story # 671 - Create Web form**

***Tasks***

* Create store process (692)
* Create Database Schema (691)
* Create Documentation (690)
* Read Objective-c Book (667)

***Acceptance Criteria***

* Create the web form with input fields, images, and documents
* Create Database schema
* Test data insertion, update and delete

***Modeling***

Refer to UML diagrams in Appendix A that were created or modified to model the functionality that will be implemented in this sprint.

**User Story # 672 - Create web login authentication**

***Tasks***

* Create Documentation (689)
* Integrate web form authentication (673)
* create email address authentication (670)

***Acceptance Criteria***

* Create authentication system
* Verify user email address
* Authentication system (tables) must take into account future authentication techniques (e.g., Facebook, Gmail) but not implemented at this moment

***Modeling***

Refer to UML diagrams in Appendix A that were created or modified to model the functionality that will be implemented in this sprint.

### Sprint 3

(09/28/2015 -10/09/2015)

**User Story # 699 -Create Employer Registration on IPhone**

***Tasks***

* Create Employer Core Data (705)
* Create Employer Model (704)
* Create Views for Employer (703)
* Create JSON Receivers (702)
* Create Documentation(Wilfredo) (701)
* Create Documentation(Luis) (700)

***Acceptance Criteria***

* Creation a registration form for employer user
* Test registration form data was properly inserted and updated using a pseudo authentication form database
* add sms authentication

***Modeling***

Refer to UML diagrams in Appendix A that were created or modified to model the functionality that will be implemented in this sprint.

### Sprint 4

(10/12/2015 -10/23/2015)

**User Story # 663 - Login on iPhone**

***Tasks***

* Login implementation (706)
* Research login techniques (665)
* Create documentation (664)

***Acceptance Criteria:***

1. Implement login feature in the app.
2. User should be able to log in with an existing account.
3. Test login to validate user’s information.

***Modeling***

Refer to UML diagrams in Appendix A that were created or modified to model the functionality that will be implemented in this sprint.

**User Story # 679 - Forgot Password**

***Tasks***

* Research forgot password techniques (708)
* Test forgot password feature (709)
* Create documentation (707)

***Acceptance Criteria:***

1. User should be able to reset their password upon sms verification.
2. Test the sms verification for password reset.
3. Create documentation.

***Modeling***

Refer to UML diagrams in Appendix A that were created or modified to model the functionality that will be implemented in this sprint.

**User Story # 680 – Create a series of search**

***Tasks***

* Implement search filters (713)
* Test filters (712)
* Create Documentation (710)

***Acceptance Criteria:***

1. Implement search filters
2. test filters
3. Create documentation.

***Modeling***

Refer to UML diagrams in Appendix A that were created or modified to model the functionality that will be implemented in this sprint.

### Sprint 5

(10/26/2015 -11/06/2015)

**User Story # 682 - Navigate through Talent View and get information on Talent**

***Tasks***

* Create JSON receivers that accept request of talent info. (718)
* Create Talent Info View and controller. (716, 717)
* Create documentation. (683, 714)
* Create Talent list view and controller (715)

***Acceptance Criteria***

* Create a Talent View that list the talents from the data results of “search talent”
* Create a Subview that details info about the current talent selected.
* Create documentation on views and controllers.
* Test the views and controllers.

***Modeling***

Refer to UML diagrams in Appendix A that were created or modified to model the functionality that will be implemented in this sprint.

### Sprint 6

(11/09/2015 -11/20/2015)

**User Story # 719 - Map View**

***Tasks***

* Create documentation for the map view. (720)
* Do research regarding map views on iOS. (721)
* Implement the map view. (722)

***Acceptance Criteria***

* Create documentation.
* Do research on how to integrate maps using Objective-C.
* Implementation.

***Modeling***

Refer to UML diagrams in Appendix A that were created or modified to model the functionality that will be implemented in this sprint.

**User Story # 723 - Detailed Profile View**

***Tasks***

* Create documentation for the Detailed View (724)
* Do research regarding Table Views on iOS. (725)
* Implement the Profile Detailed View (726)

***Acceptance Criteria***

* Documentation completed.
* Research about Table Views on iOS done.
* Implementation of the Profile Detailed View.

***Modeling***

Refer to UML diagrams in Appendix A that were created or modified to model the functionality that will be implemented in this sprint.

### Sprint 7

(11/23/2015 -12/04/2015)

**User Story # 719 - Map View**

***Tasks***

* Create JSON handlers from server to send Pin info (730)
* Research Pin on iOS Map (729)
* Update Documentation with Pin info (728)
* Implement Staff Pins to Map (727)
* Implement map view (722)
* Do research on iOS map view (721)
* Create documentation on map view (720)

***Acceptance Criteria***

* Create documentation on mapview
* Do research on how to integrate maps using Objective-C
* Implement document with pin information
* Create pin on map
* Research pin on iOS Map

***Modeling***

Refer to UML diagrams in Appendix A that were created or modified to model the functionality that will be implemented in this sprint.

**User Story # 731 - Create Management Website**

***Tasks***

* Update database schema for management website
* Create management webpage
* Create documentation on management webpage

***Acceptance Criteria***

* Update database schema
* Create documentation
* Create management webpage

***Modeling***

Refer to UML diagrams in Appendix A that were created or modified to model the functionality that will be implemented in this sprint.

# System Design

This chapter gives a high level description of the system design for the GoLocalStaffApp. It first provides an overview of the system by elaborating the architectural patterns being used for the system. Then explains how the system is broken into different components, their purpose and how they all work together.

## Architectural Patterns

After careful analysis and deep understanding of our system, the Model View Controller architectural pattern was used in this project. We made this decision taking into account the agile development techniques, which we are also using to develop the system. It helps us divide the iPhone application into three connected parts, so we can separate the internal code representing the information from the ways that information is shown to or received from the user. That is, it keeps the logic and view components separated.

## System and Subsystem Decomposition

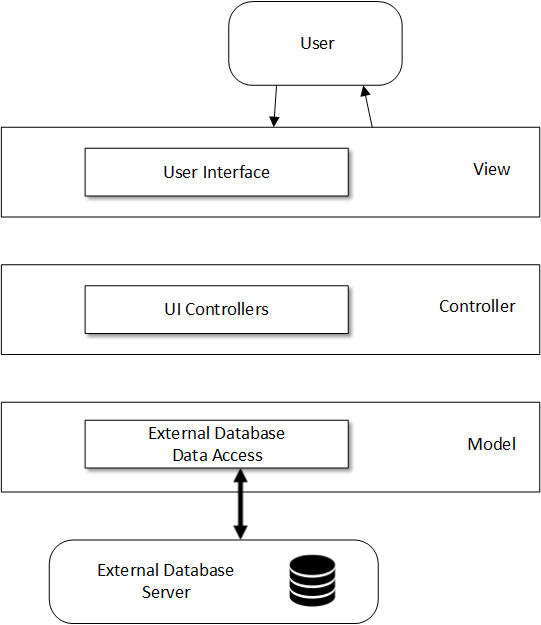


Figure 1 – System Decomposition

**Model:** Permit the access to the database like MySQL Database. This is in charge of connecting the external database to the application in a secure manner.

**View:** The purpose of this component is to separate other components from the user interface.

**Login:** This is the first view that the user sees when he or she loads up the application.

**Registration:** This component is divided between the view and the model, all handled by the controller.

**MySQL Database:** This is an external Database that connects to the model component.

## Deployment Diagram

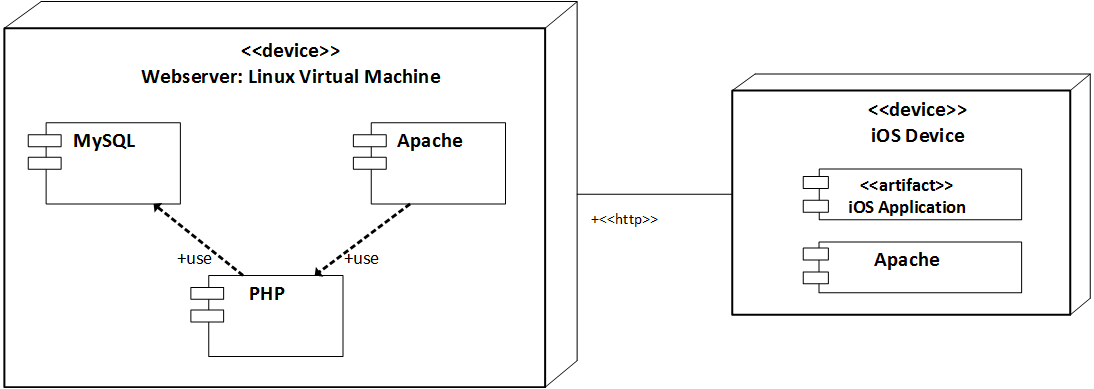


Figure #3 – Deployment Diagram

## Design Patterns

**Singleton**

In the model component of our architectural pattern, we use the singleton design pattern in a class that represents the connection to the external database. This allows us to maintain only one single object that handles these connections, which gives adds security to the system, since it avoids the possibility of another component changing the state of the database.

# System Validation

The system validation for the GoLocalStaffApp was done through a series of manual testing.

**User Story # 676 - Create User Account Display for App**

System Tests

|  |  |
| --- | --- |
| Identifier | TC1\_loadUserAccountDisplay |
| Purpose | To ensure that the app displays the user account layout |
| Pre-conditions | Open the app in the iPhone |
| Input | None |
| Expected Output | The app opens and displays the user account layout. |

**User Story # 674 - Create Staff Registration on iPhone**

System Tests

|  |  |
| --- | --- |
| Identifier | TC2\_registerNewUser |
| Purpose | To ensure that the user that the information provided by the user gets saved in the database |
| Pre-conditions | Open the app and tap sign up as staff |
| Input | User inputs personal data |
| Expected Output | The user's information gets saved to the database. |

**User Story # 672 - Create Web Login authentication**

System Tests

|  |  |
| --- | --- |
| Identifier | TC3\_webLogInAuthentication |
| Purpose | To ensure that a user get his or her email address authenticated |
| Pre-conditions | The user is registering |
| Input | The user provides a valid email address |
| Expected Output | The user will receive an email with a link to authenticate his or her email address |

**User Story # 663 - Login on iPhone**

System Tests

|  |  |
| --- | --- |
| Identifier | TC4\_iPhoneLogin |
| Purpose | validate user login in IPhone |
| Pre-conditions | the user needs not to be login |
| Input | enter username , enter password, click login |
| Expected Output | Successfully logs into their respective home page |

**User Story # 679 - Forgot Password**

System Tests

|  |  |
| --- | --- |
| Identifier | TC5\_forgotPassword |
| Purpose | To ensure that the user has a way to reset his or her password |
| Pre-conditions | The user has to be registered with a valid phone number |
| Input | The user inputs the same phone number that's on the account |
| Expected Output | The user will receive a text message with a valid unique code, which will let him or her reset the password |

**User Story # 680 - Create a series of search filters for the employer home page**

System Tests

|  |  |
| --- | --- |
| Identifier | TC6\_searchFilters |
| Purpose | To ensure that the employer user can select multiple fields to narrow down the search |
| Pre-conditions | The user has to be logged in as an employer user |
| Input | The user selects different criteria fields in the application |
| Expected Output | The application goes to the next view showing a list of staff matching the criteria previously selected |

**User Story # 682 - Navigate Through Talent View and Get Information from talent**

System Tests

|  |  |
| --- | --- |
| Identifier | TC7\_navigateFromTalentViewToDetailedView |
| Purpose | To ensure that once the user sees the list of staff, he or she can select one and get a detailed view of that user |
| Pre-conditions | The users is logged in and has done a search |
| Input | The user selects a specific staff |
| Expected Output | The application goes to a view where it shows a more detailed view of the selected staff |

**User Story # 699 - Create Employer Registration on iPhone**

System Tests

|  |  |
| --- | --- |
| Identifier | TC8\_employerRegistration |
| Purpose | register an employer user |
| Pre-conditions | user clicks on employer registration |
| Input | enter employer necessary field |
| Expected Output | display successful register message |

**User Story # 723 – Detailed profile View**

System Tests

|  |  |
| --- | --- |
| Identifier | TC9\_detailProfileView |
| Purpose | detail profile view of the current staff user touched from the staffSearchResultsTableview |
| Pre-conditions | employer user has requested to search for staff |
| Input | user clicks on a user from the staffsearchtableviewcell |
| Expected Output | display a new profile view with staff info |

**User Story # 719 – Map View**

System Tests

|  |  |
| --- | --- |
| Identifier | TC10\_mapView |
| Purpose | display mapview of search talent results |
| Pre-conditions | employer user needs to be signed in |
| Input | user clicks on map mode |
| Expected Output | mapview with pins reflecting the search staff results |

**User Story # 731 – Create Management Website**

System Tests

|  |  |
| --- | --- |
| Identifier | TC11\_managementSiteLogin |
| Purpose | To ensure that an admin user can log into the site |
| Pre-conditions | The computer is connected to the Internet |
| Input | The user inputs a valid set of credentials |
| Expected Output | The site authenticates the user and redirects him or her to the main page |

# Glossary

Define any domain-specific terms that the audience of this document may be unfamiliar with. You can assume the audience of this document to be technical savvy. Examples of terms that do not need to be defined are: HTML, CSS, Web Server, DB, and the like.

* **JSON (**JavaScript Object Notation**)** - File format that user’s data is sent from application to server.
* **NIB –** In Xcode, describes the elements of the user interface.
* **ViewController -** In Xcode, represents a view to a controller from a storyboard or NIB.
* **Storyboard –** In Xcode, represents the canvas of UI (User Interface)elements consisting of NIB’s.

# Appendix

## Appendix A - UML Diagrams

### Static UML Diagrams

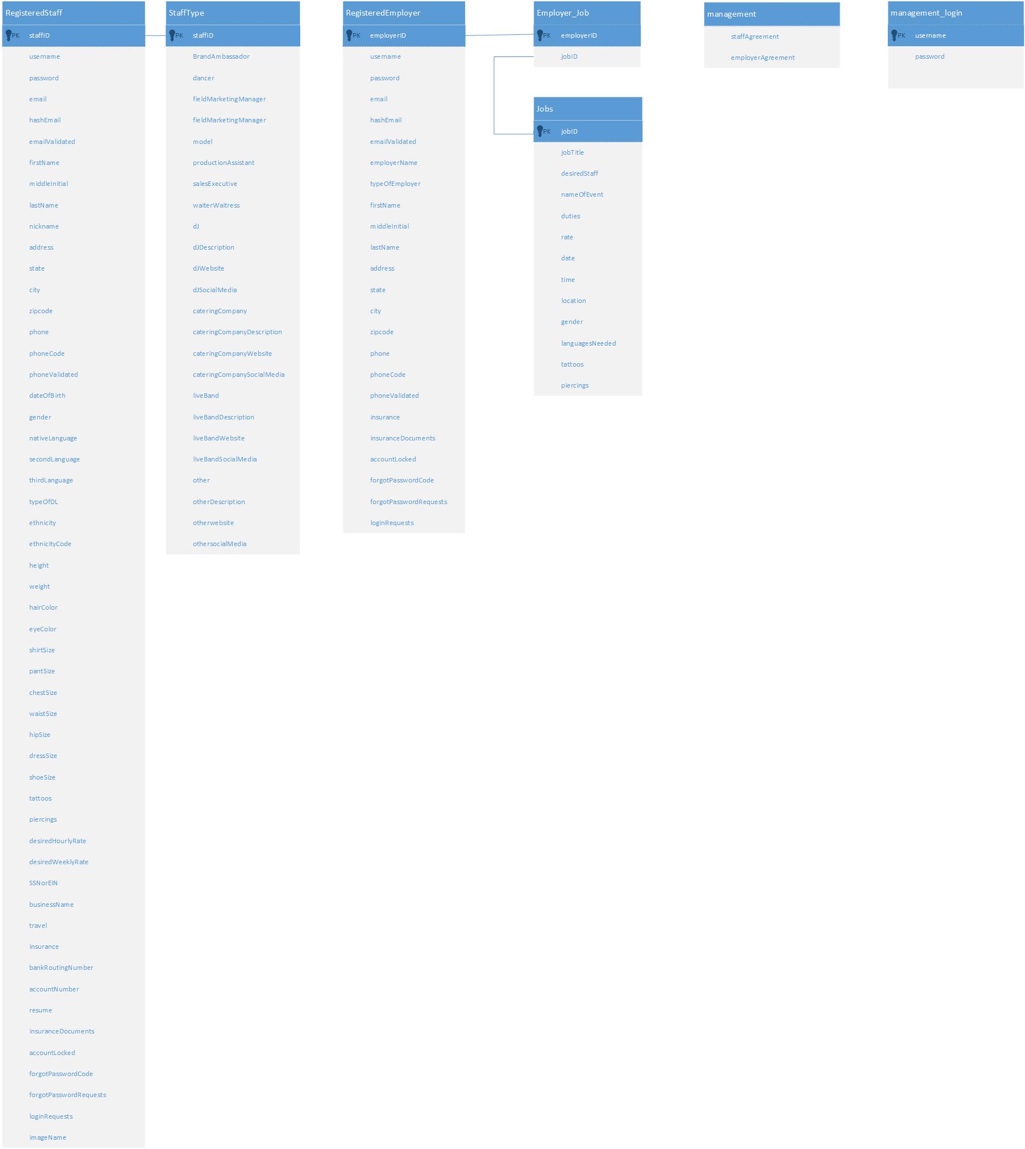


Figure #A.1 - MySQL DB Schema

…

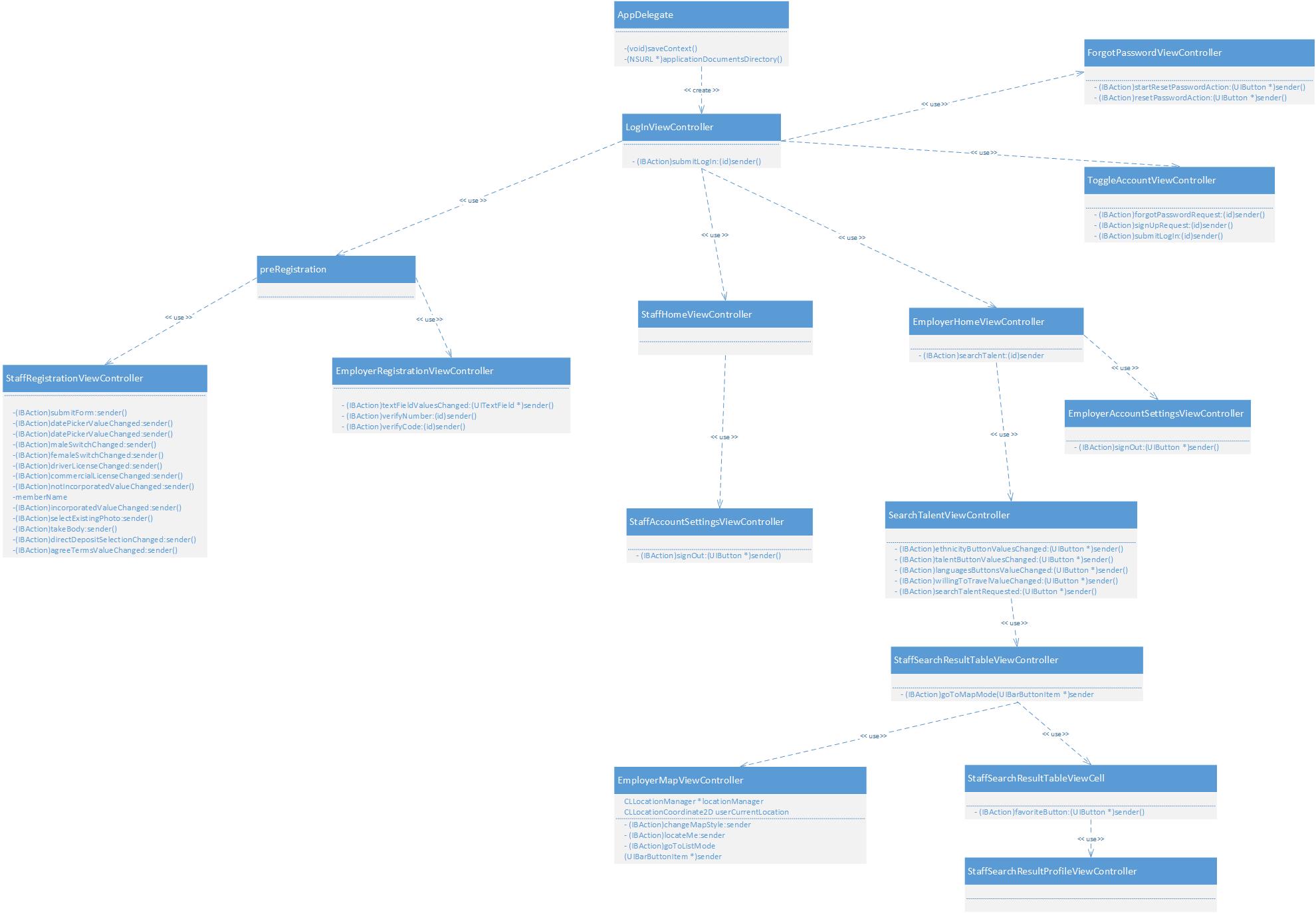


Figure #A.2 - Class Diagram

### Dynamic UML Diagrams

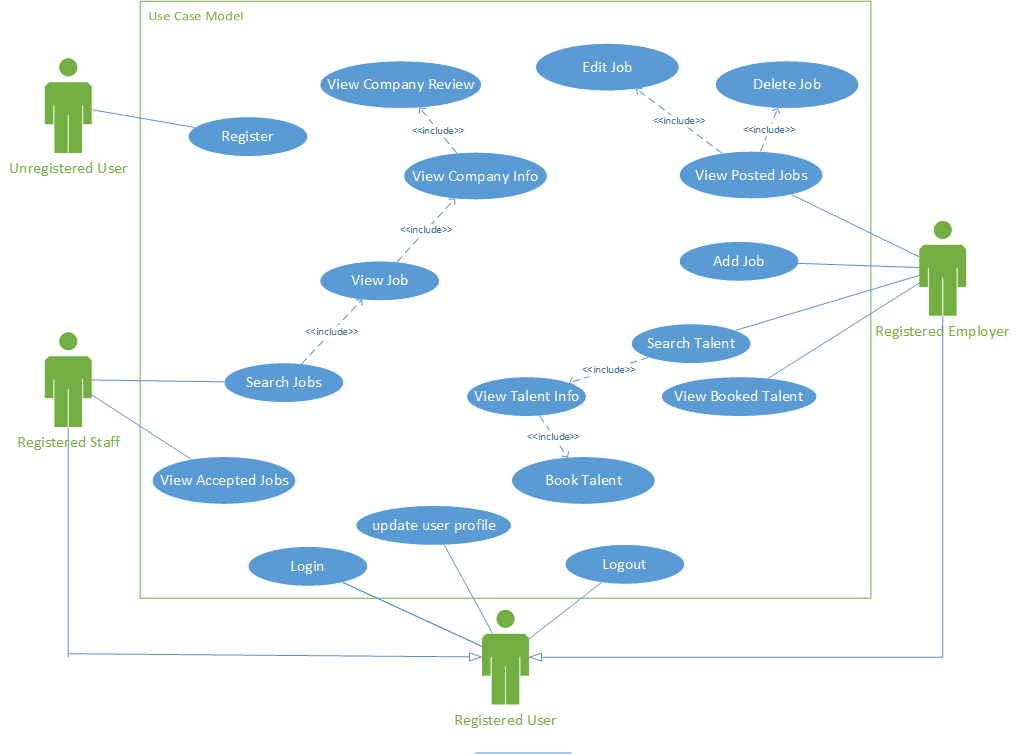


Figure #A.3 - Use Case Model

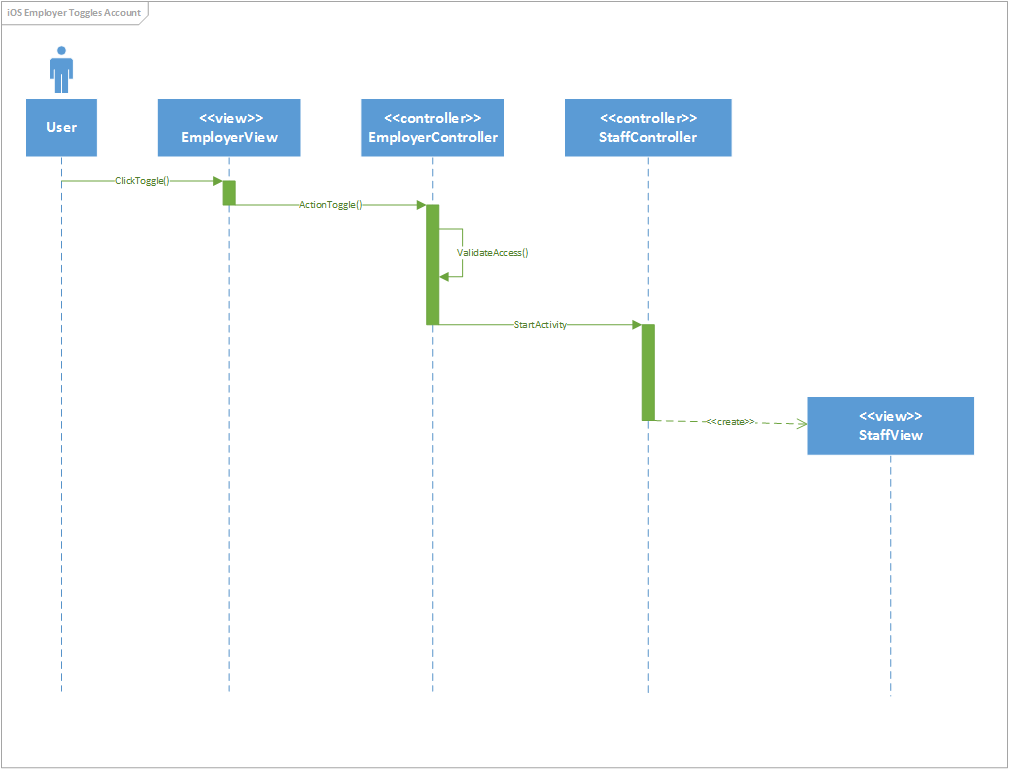


Figure #A.4 - iOS Employer Toggle Account

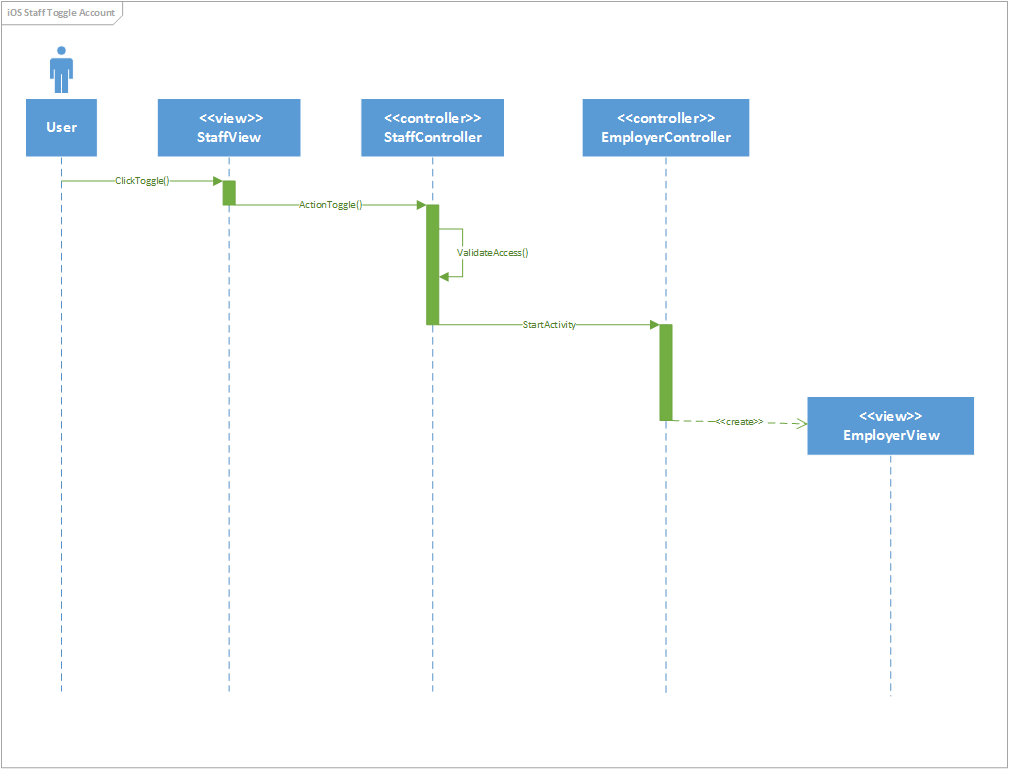


Figure #A.5 - iOS Staff Toggle Account

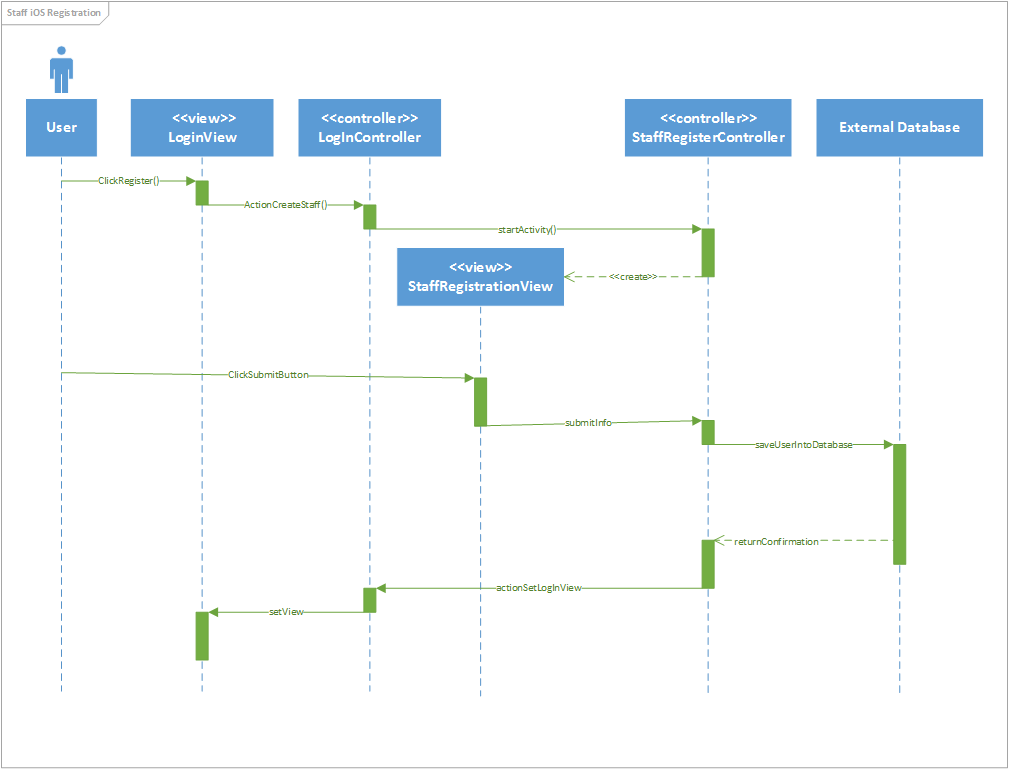


Figure #A.6 - iOS Staff Registration

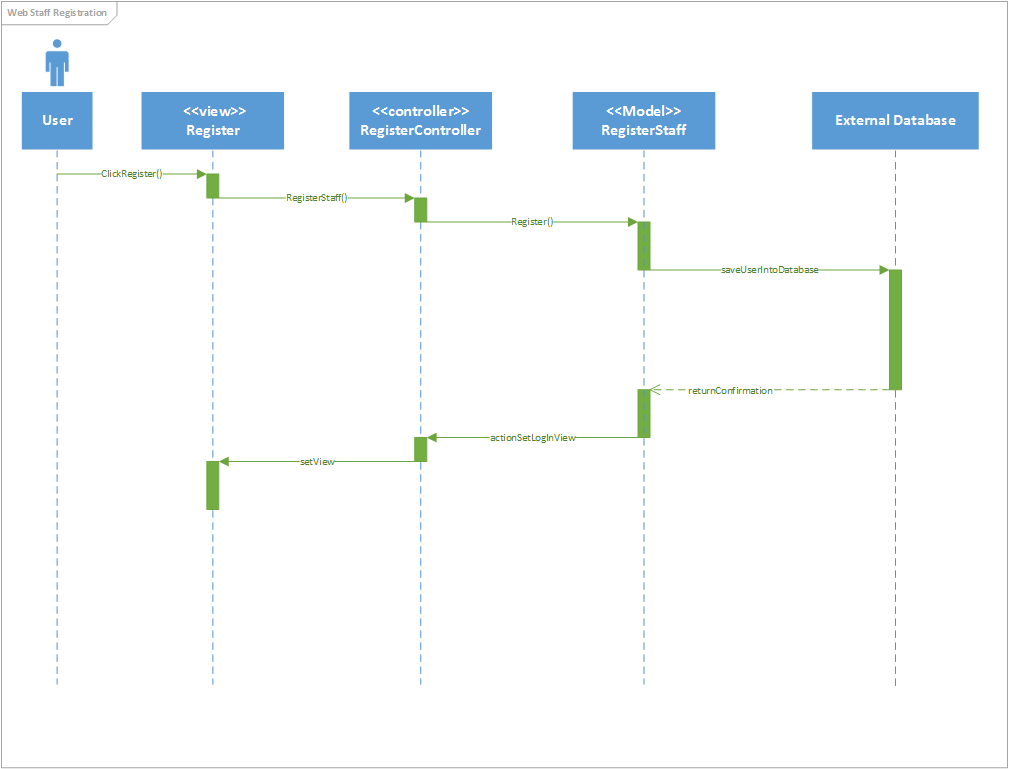


Figure #A.7 - Web Staff Registration

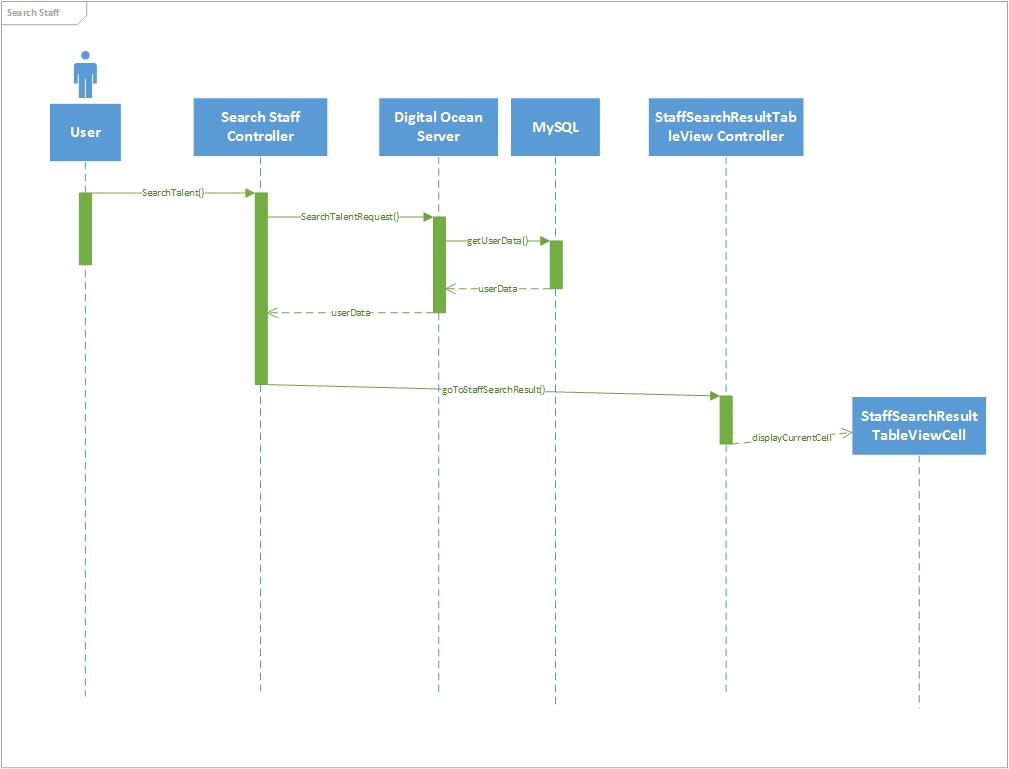


Figure #A.8 -Search Staff

## Appendix B - User Interface Design

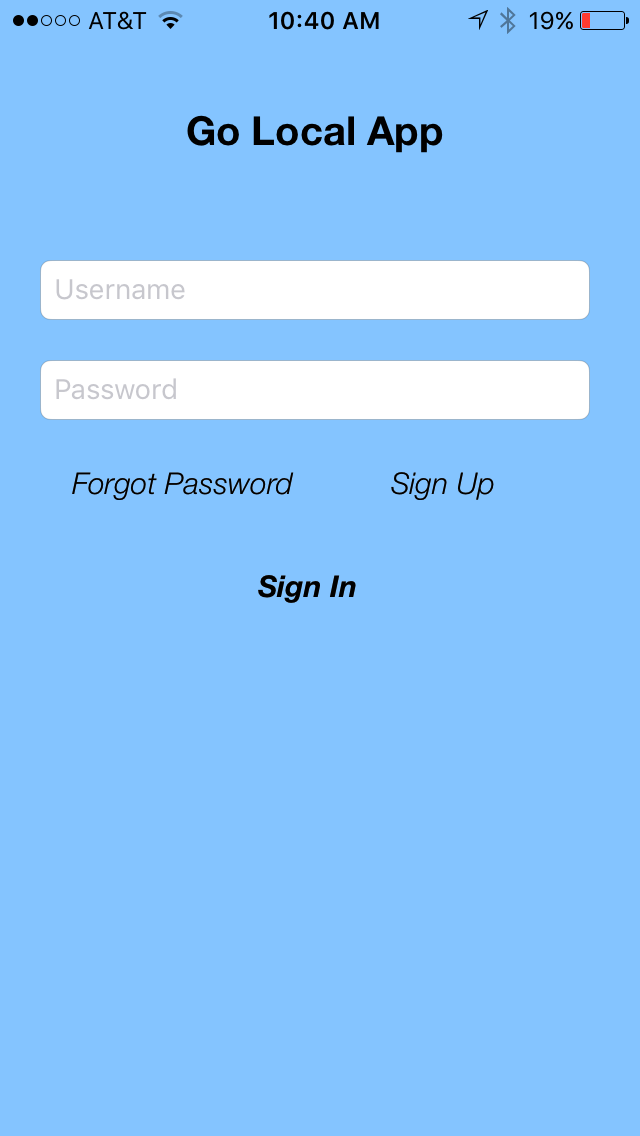


Figure #B.1 – Login Home

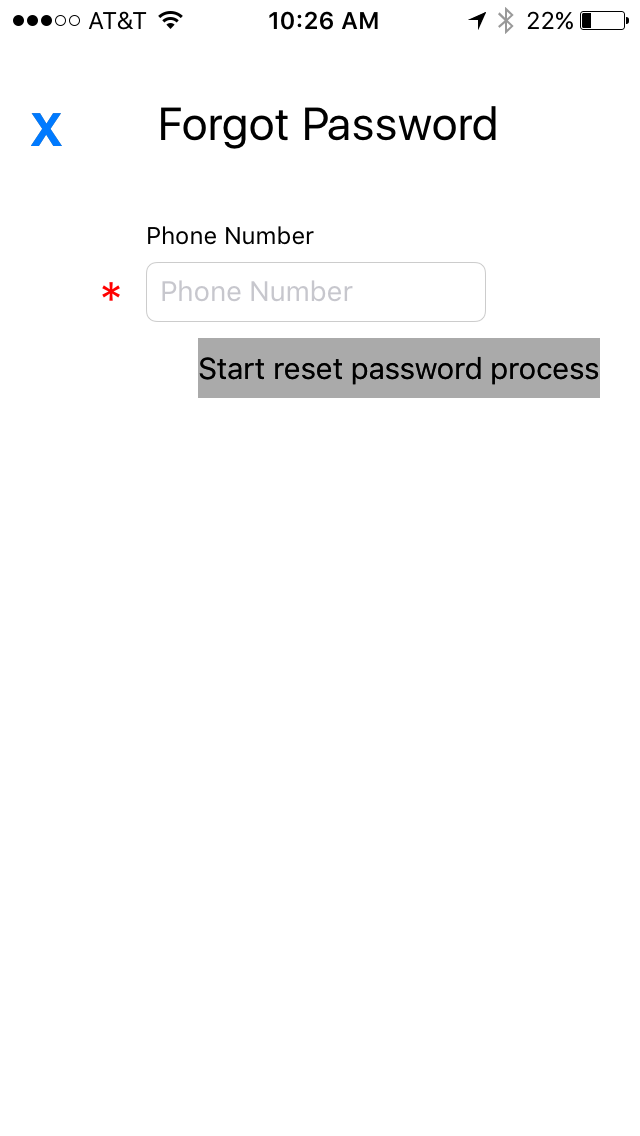


Figure #B.2 – Forgot Password

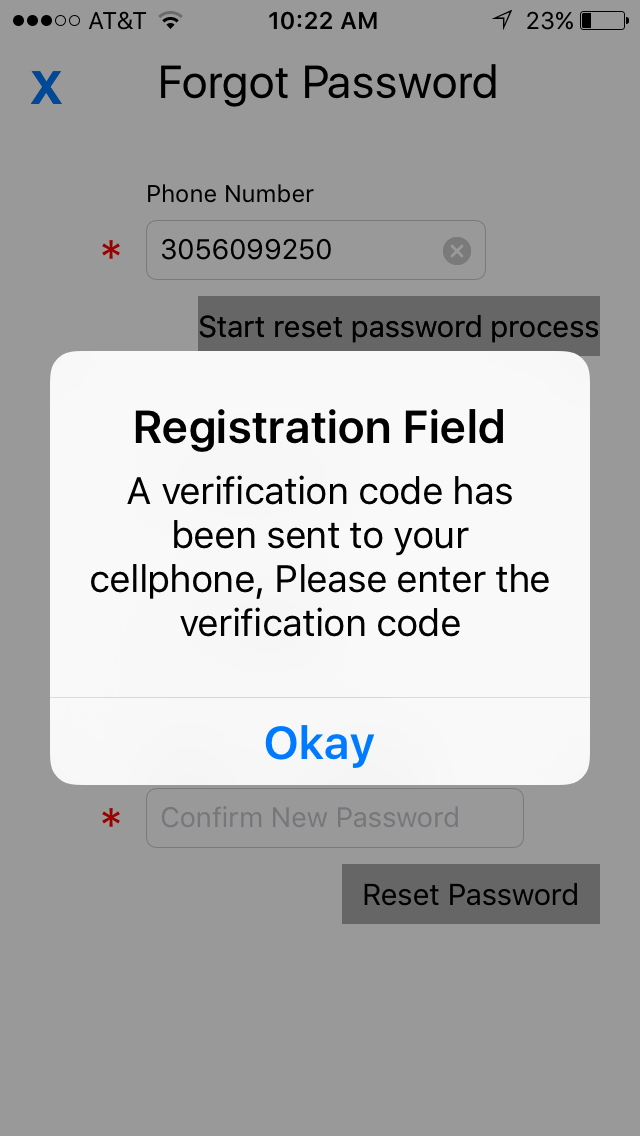


Figure #B.3 – Forgot Password Part 1

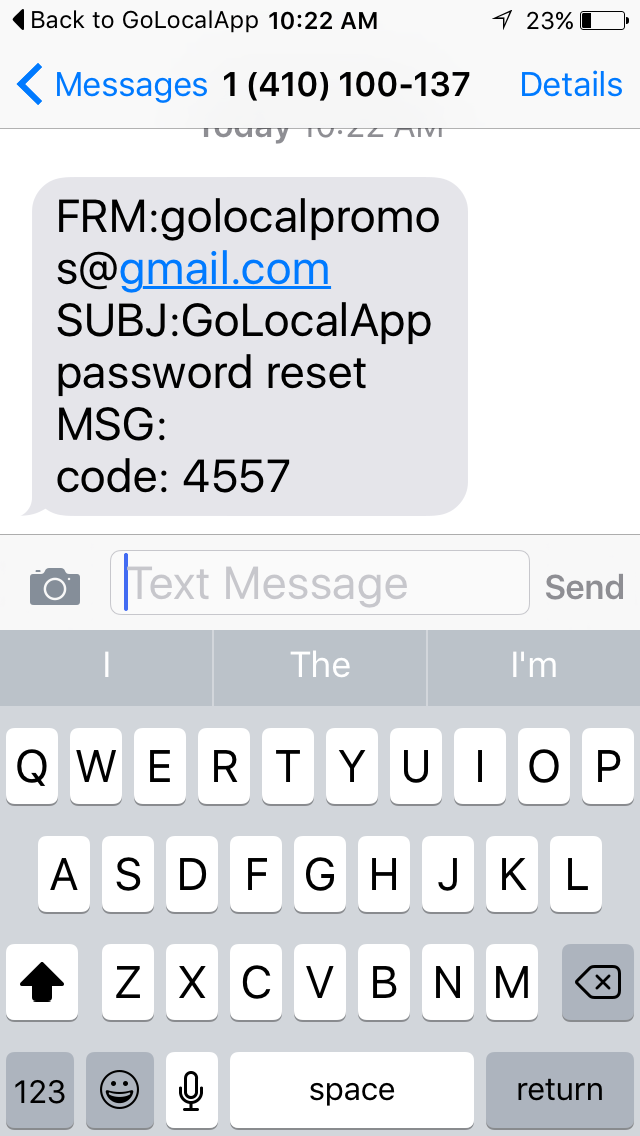


Figure #B.4 – Forgot Password SMS Code Received

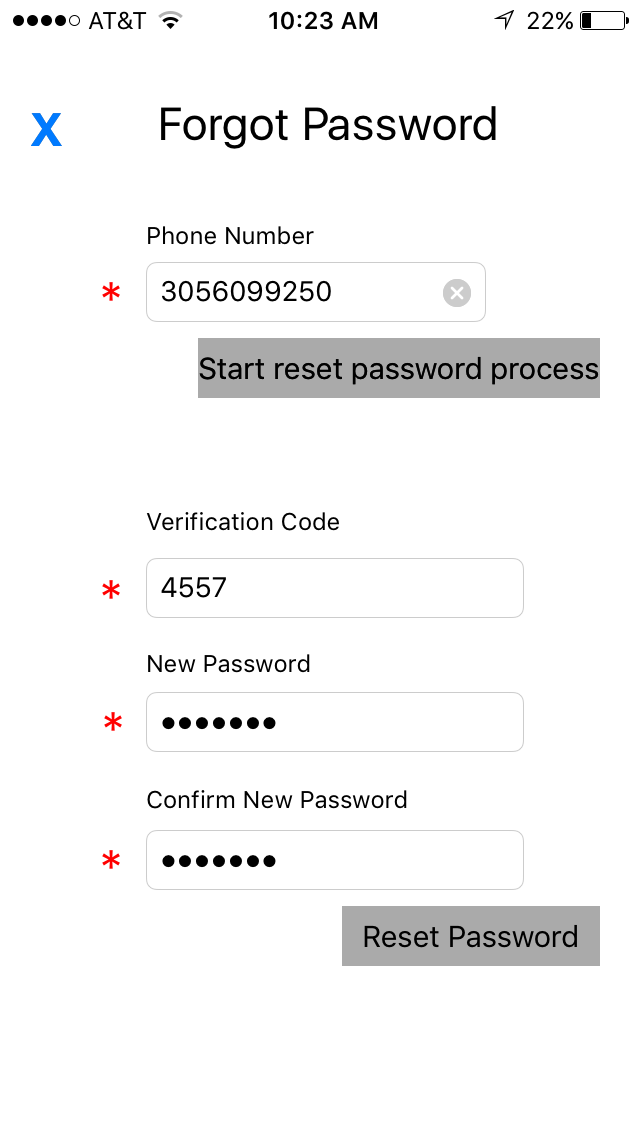


Figure #B.5 – Forgot Password Part 2

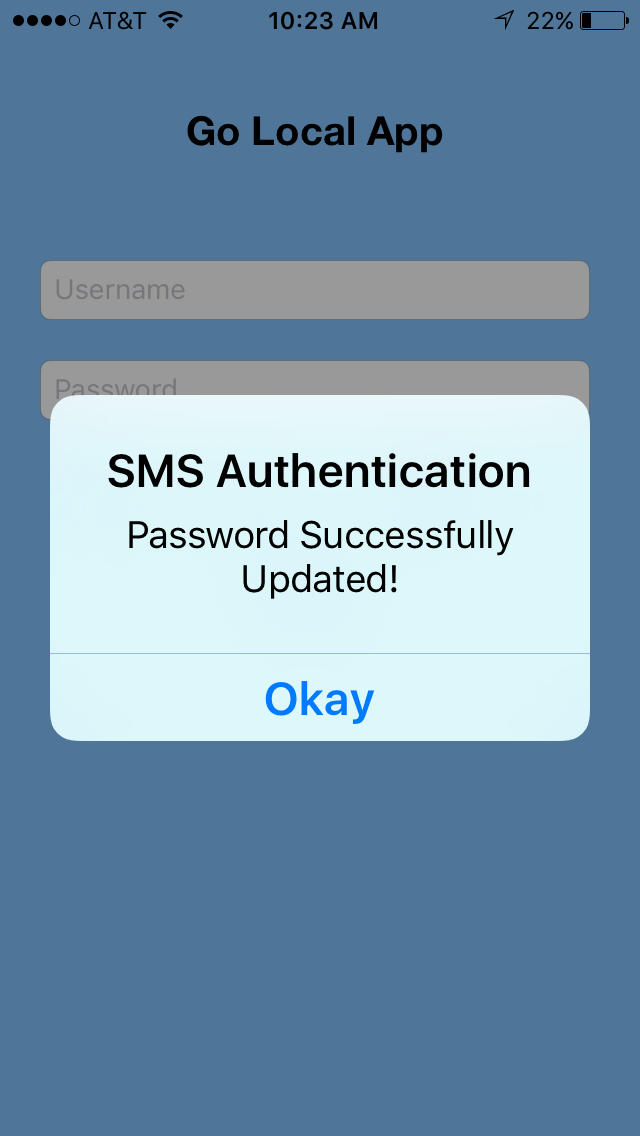


Figure #B.6 – Forgot Password Updated

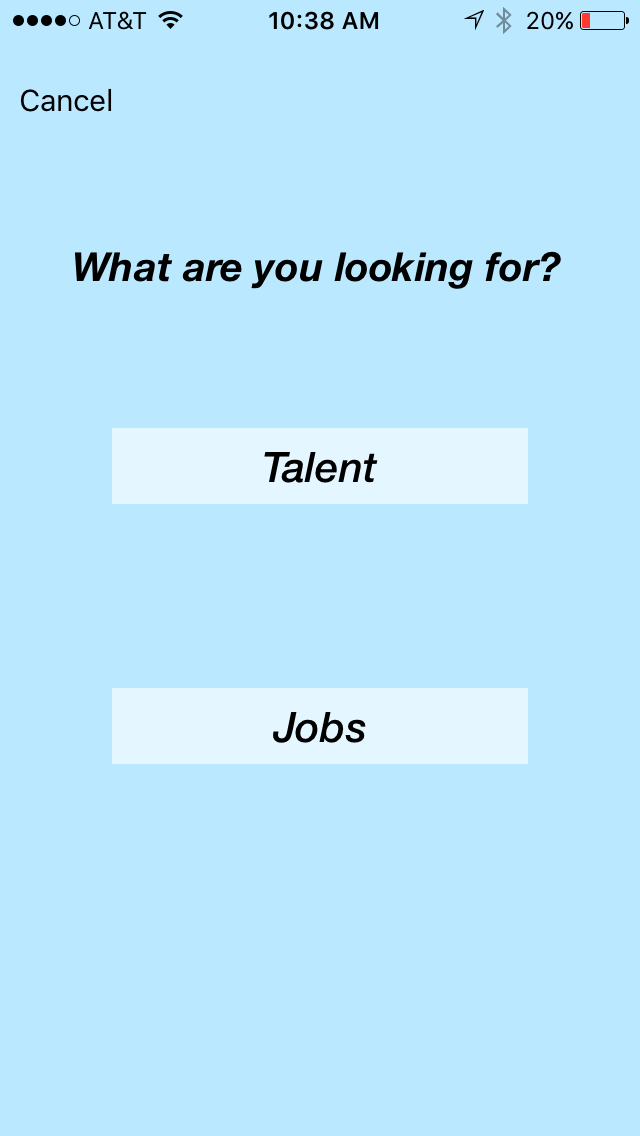


Figure #B.7 - Registration

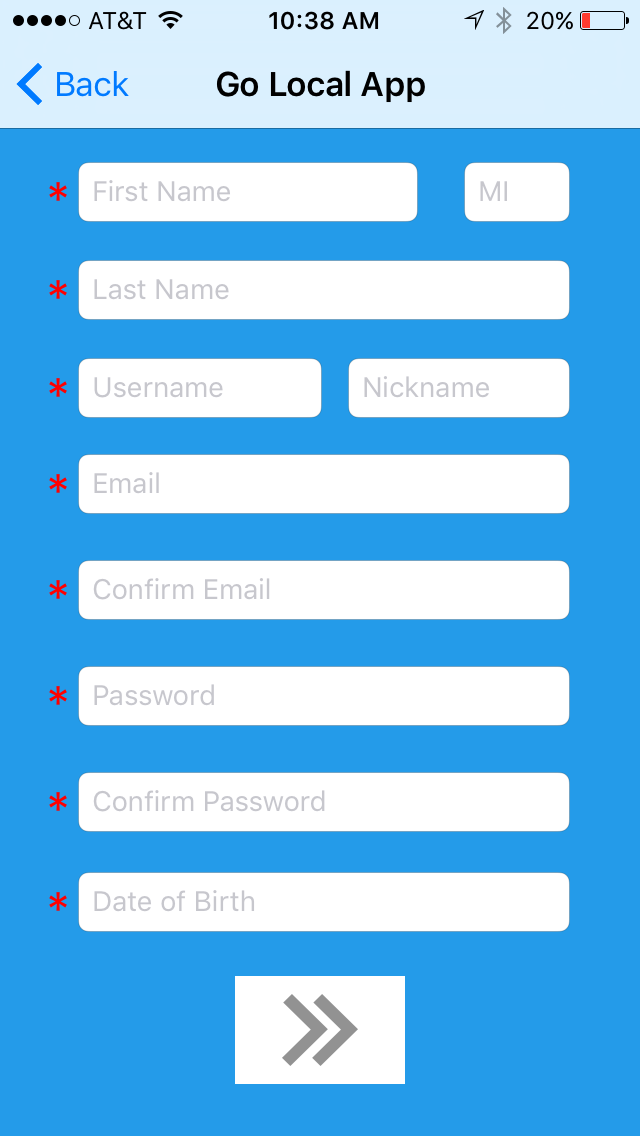


Figure #B.8 – Staff Registration

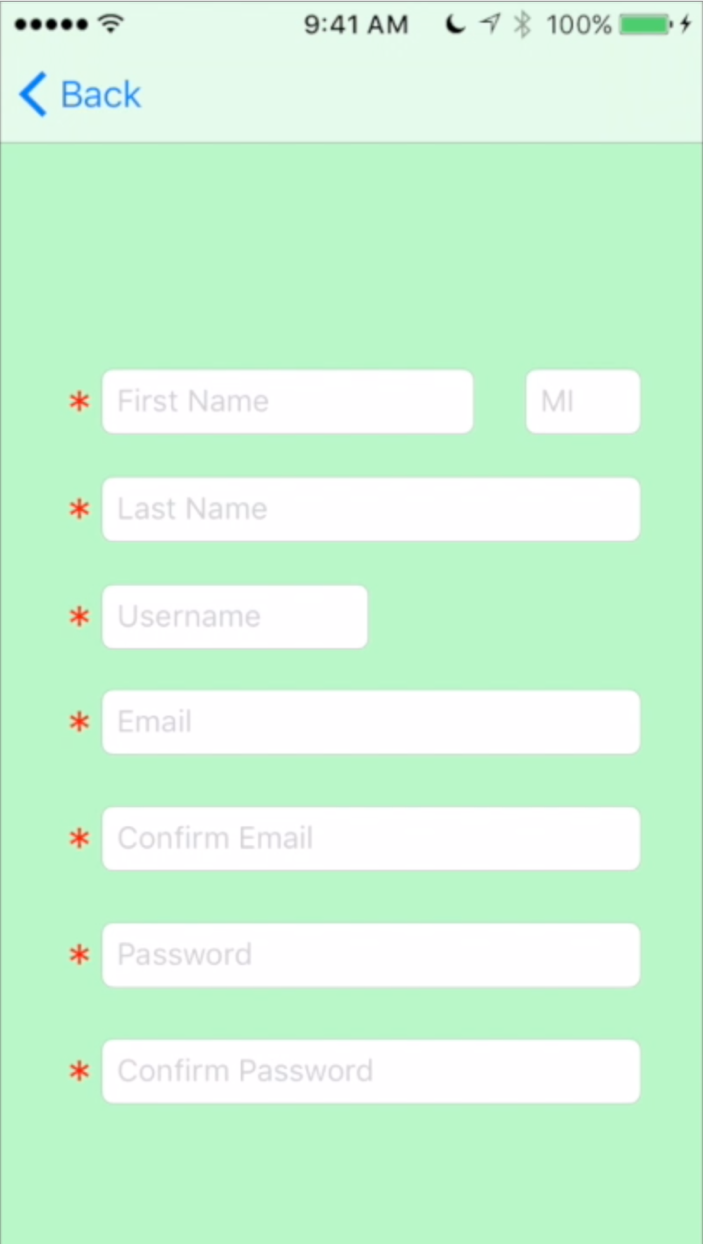


Figure #B.9 – Employer Registration

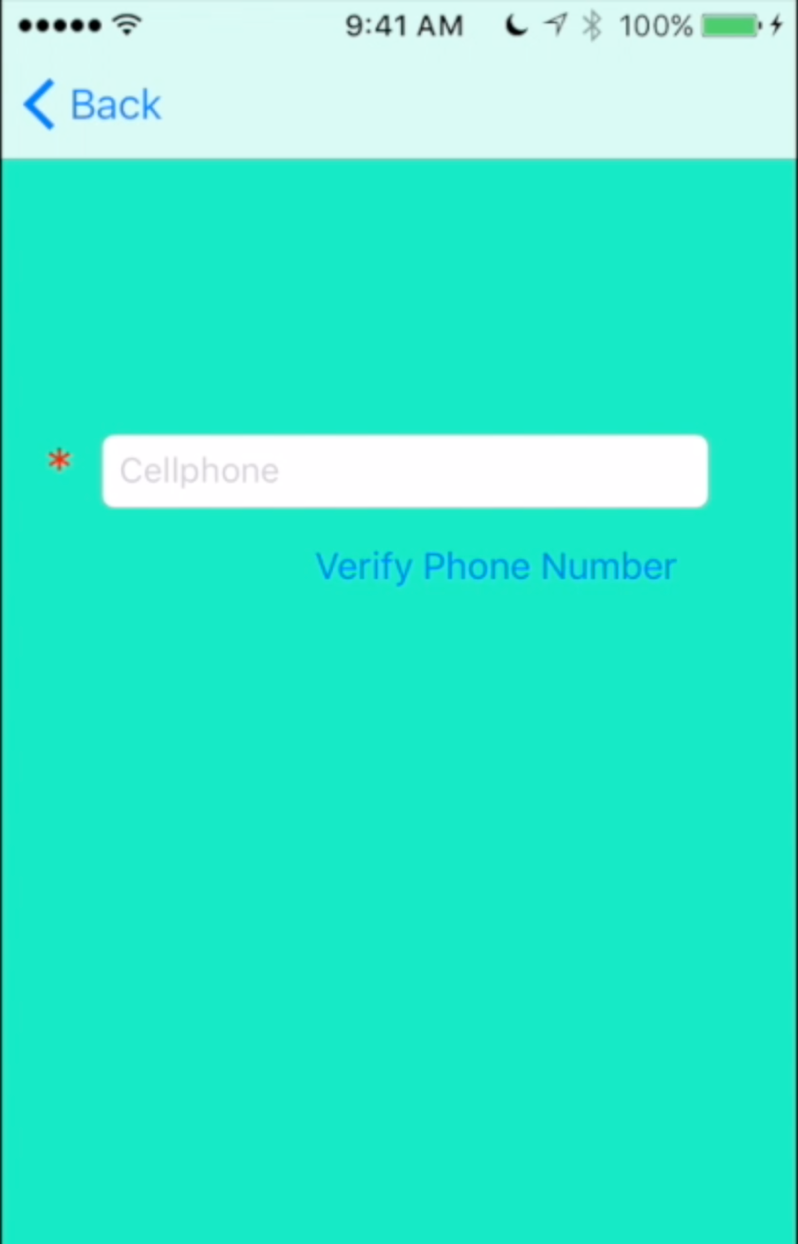


Figure #B.10 – Employer Registration SMS Part 1

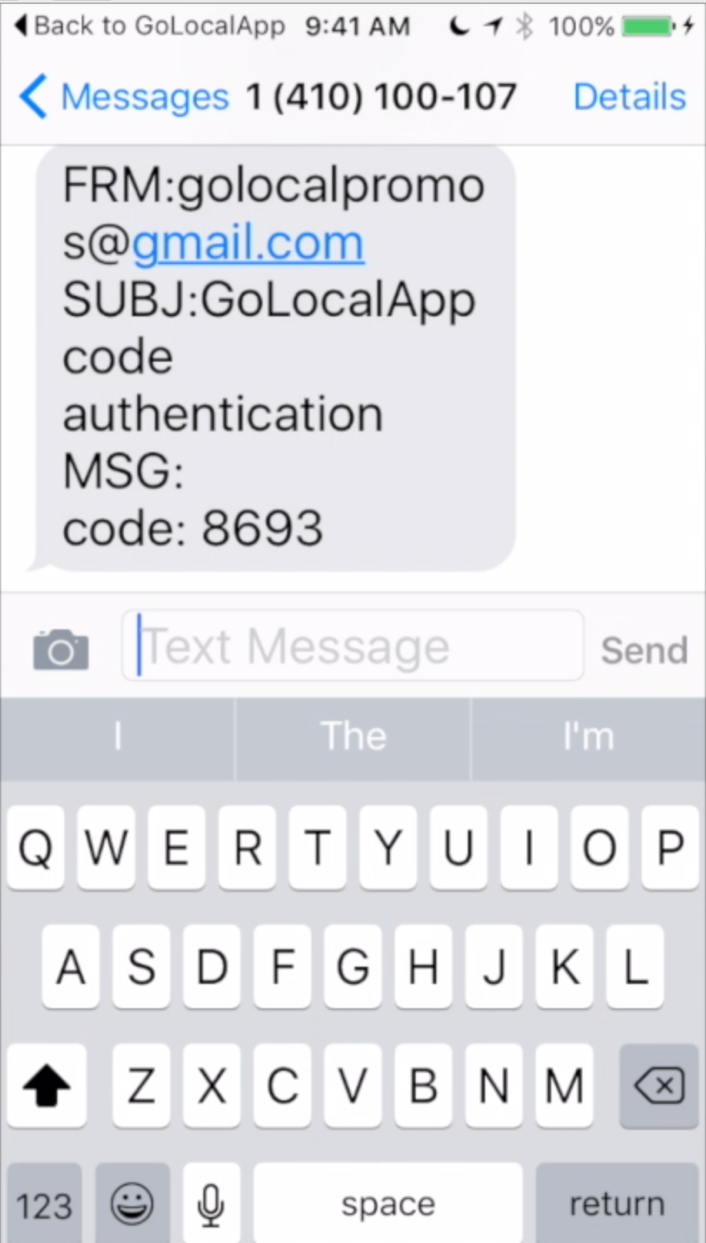


Figure #B.11 – Employer Registration SMS Code Received

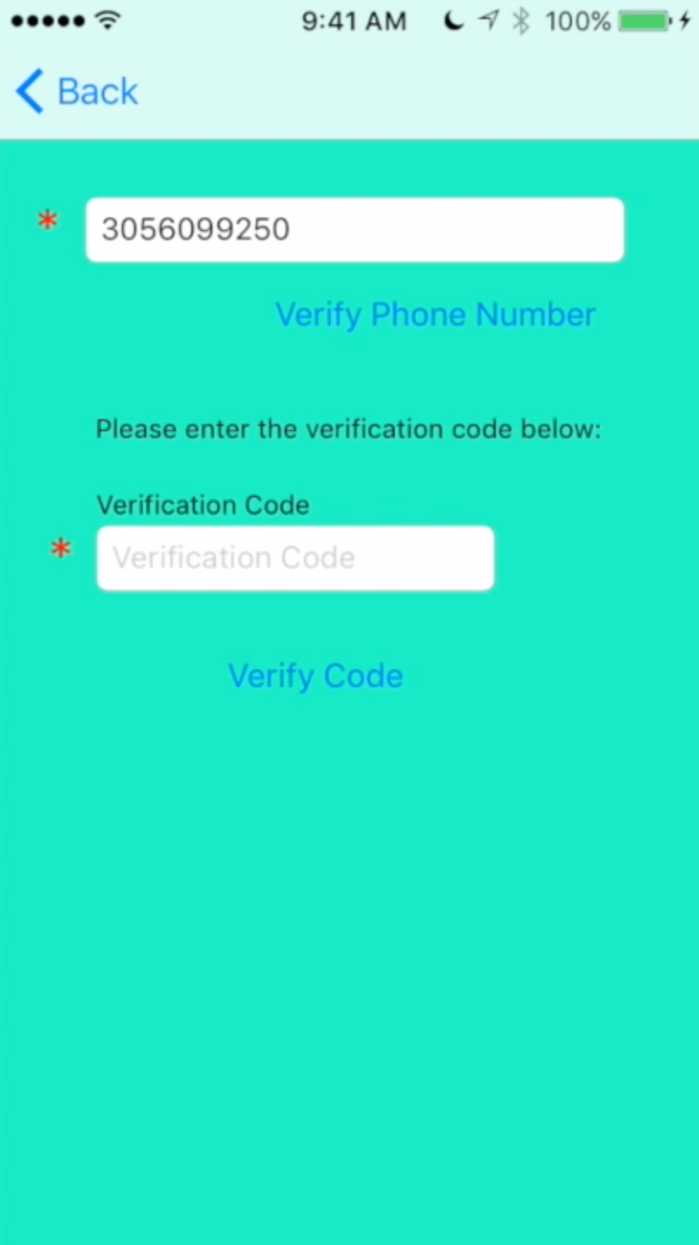


Figure #B.12 – Employer Registration SMS Part 2

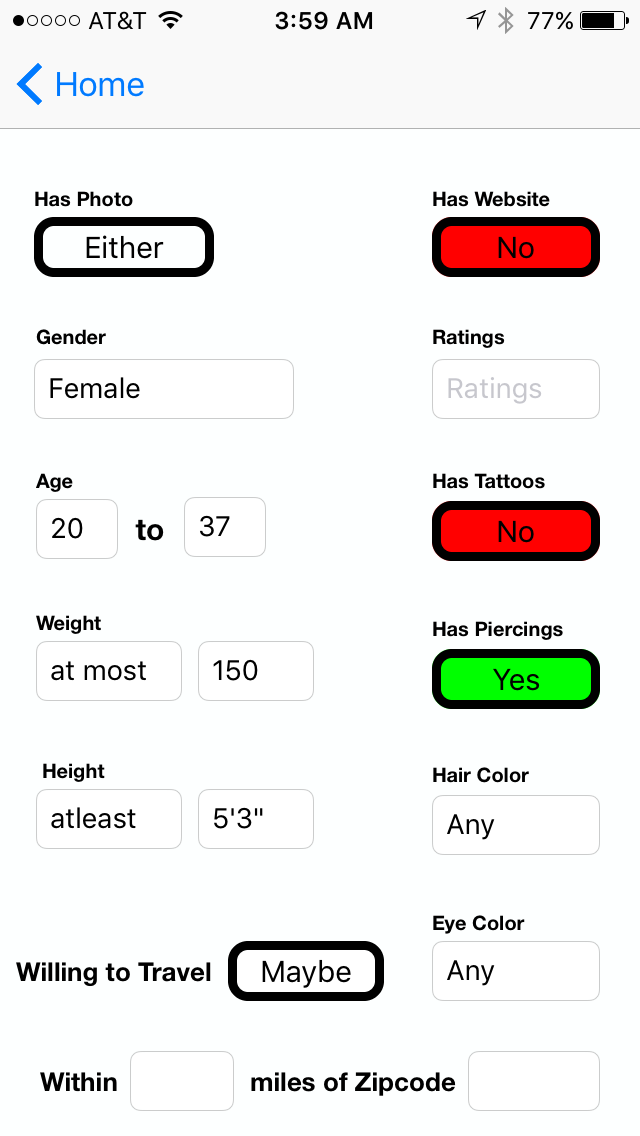


Figure #B.13 –Employer Staff Search 1



Figure #B.14 –Employer Staff Search 2



Figure #B.15 –Employer Staff Search 3

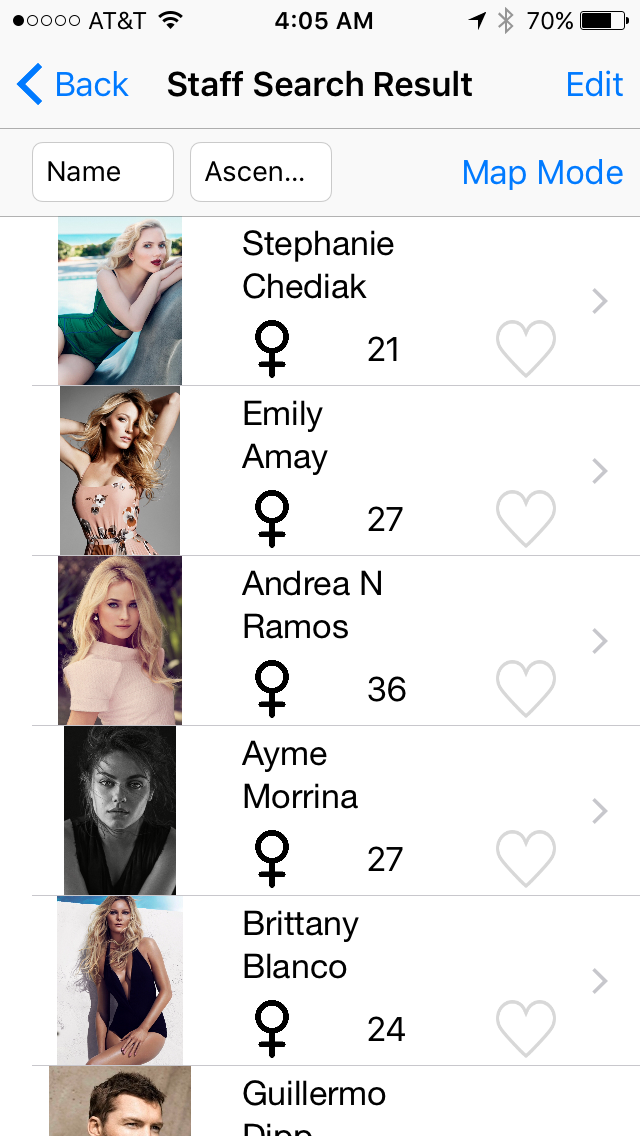


Figure # B.16 - List of Staff

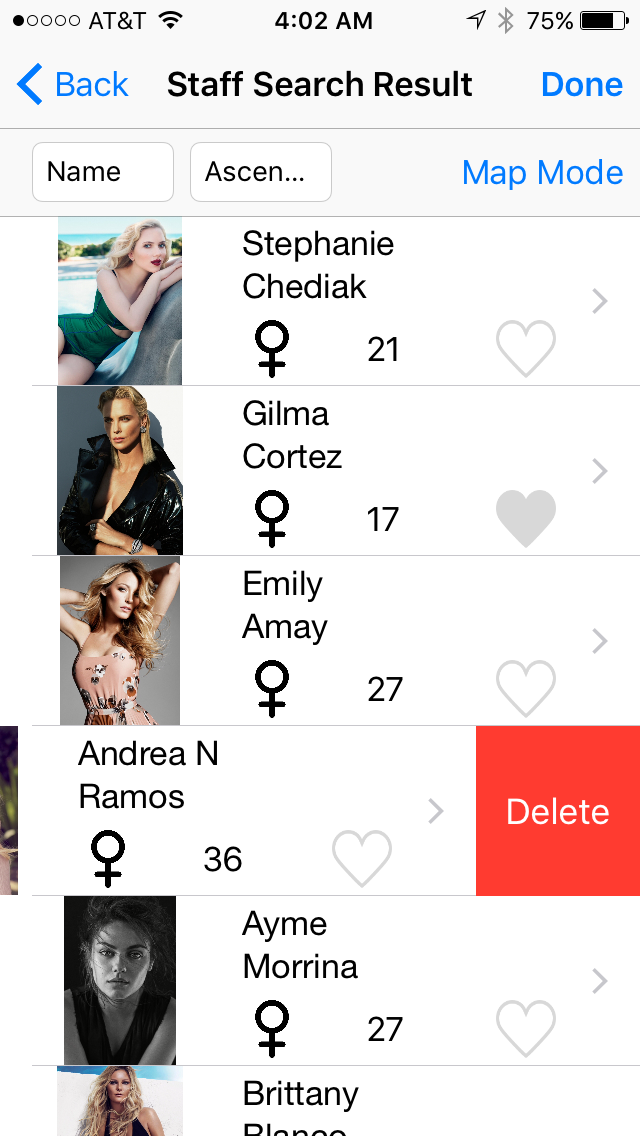


Figure # B.17 - List of Staff Delete option

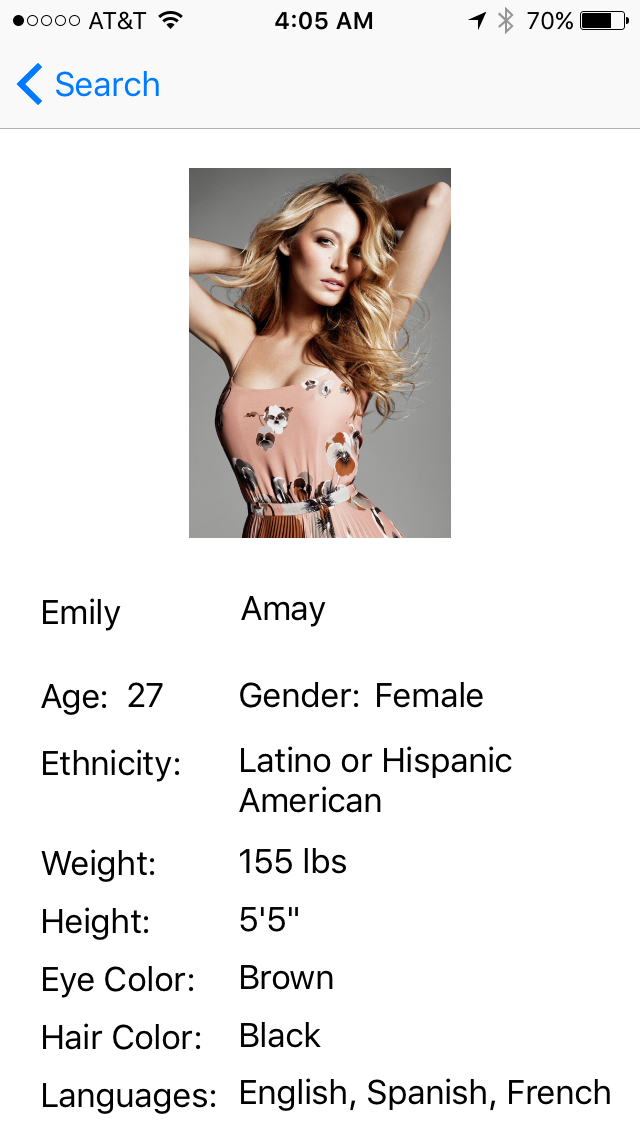


Figure #B.18 – Employer Staff Search Profile View

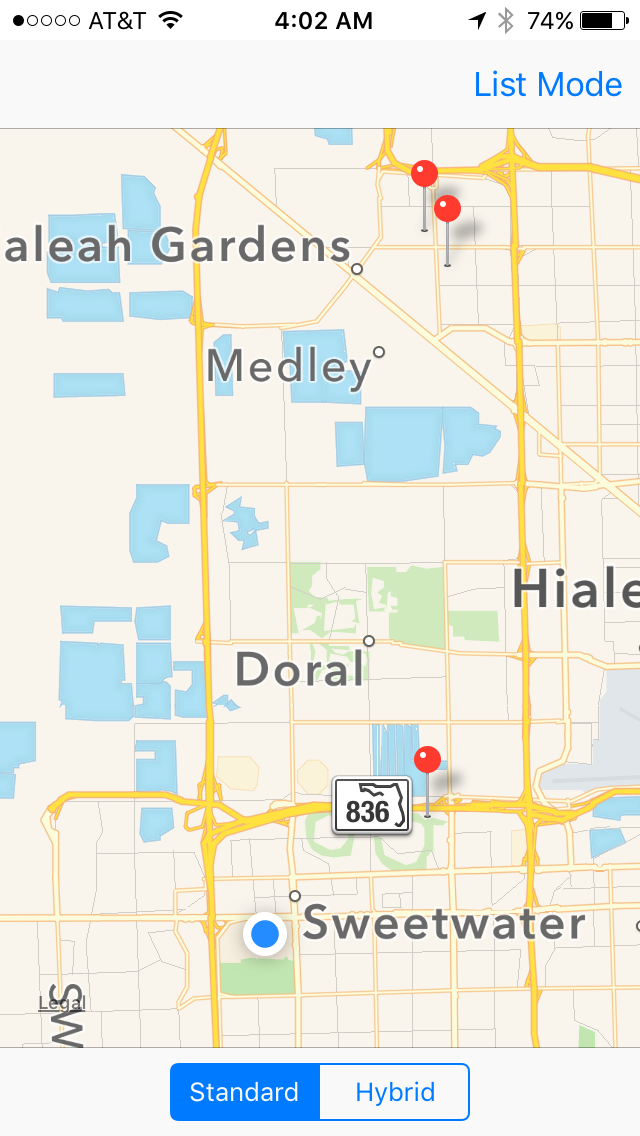


Figure # B.19 - Map View

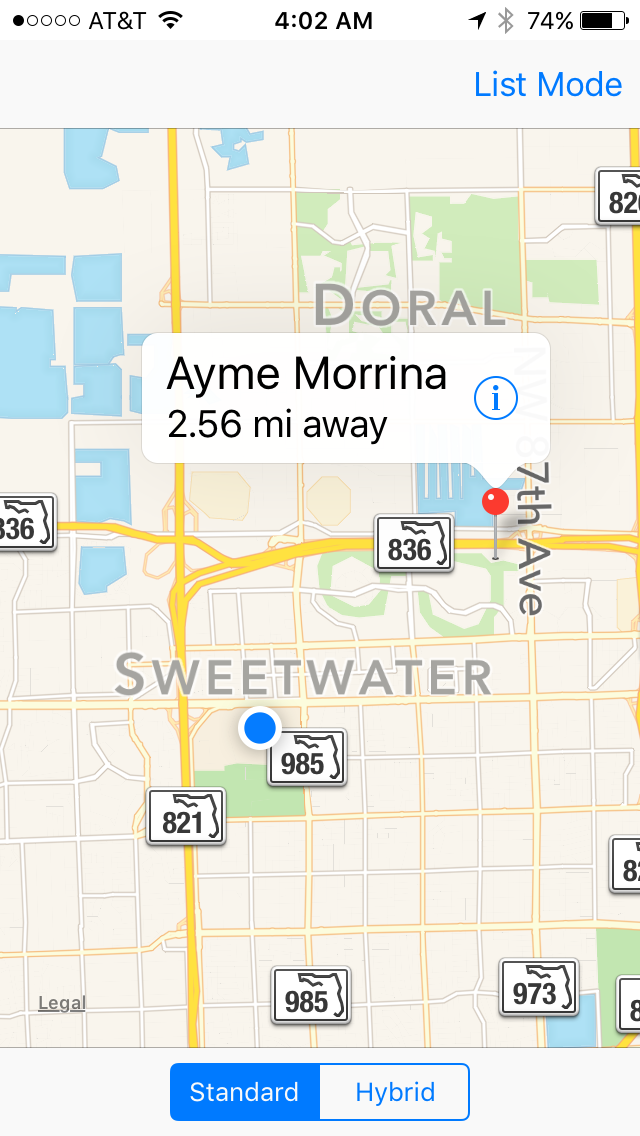


Figure # B.20 - Map View info

## Appendix C - Sprint Review Reports

**Sprint 1 Report**

**Date:** September 11, 2015

**Attendees:** Luis Castillo, Wilfredo Gomez, Eduardo Garcia, Masoud Sadjadi

**Discussed Topics:**

Setup server and created web registration form for staff talent. Add email authentication for registration process. Created iOS staff registration views and toggle feature between staff account and employer account. Part of the registration was not able to stored on the database because a sufficient amount was spent understanding JSON POST handling on the server, setting up the database on the server and configuring the server to implement the email authentication.

**Sprint 2 Report**

**Date:** September 25, 2015

**Attendees:** Luis Castillo, Wilfredo Gomez, Eduardo Garcia, Masoud Sadjadi

**Discussed Topics:**

Updated staff registration financial views and controllers along with the JSON handlers. Core data model on iOS project was added Updated staff registration experience view and controller along with the JSON handlers. IPhone sms authentication implemented. Web registration for staff user created with login authentication.

**Sprint 3 Report**

**Date:** October 9, 2015

**Attendees:** Luis Castillo, Wilfredo Gomez, Eduardo Garcia, Masoud Sadjadi

**Discussed Topics:**

Core data and Model was implemented for the employer user. Json receivers were created to handle the employer registration. IPhone controllers and views were created for the employer registration with sms authentication.

**Sprint 4 Report**

**Date:** October 23, 2015

**Attendees:** Luis Castillo, Wilfredo Gomez, Eduardo Garcia, Masoud Sadjadi

**Discussed Topics:**

Forgot password was created for the IPhone. This feature was done with sms authentication. A search feature was created so that employer may be able to search for talent based on several filters that will narrow the employer search.

**Sprint 5 Report**

**Date:** November 6, 2015

**Attendees:** Luis Castillo, Wilfredo Gomez, Eduardo Garcia, Masoud Sadjadi

**Discussed Topics:**

Created talent view and controllers that receives data being passed from the employer staff search talent. JSON receiver needed to be created. Talent list view and controller was also created.

**Sprint 6 Report**

**Date:** November 20, 2015

**Attendees:** Luis Castillo, Wilfredo Gomez, Eduardo Garcia, Masoud Sadjadi

**Discussed Topics:**

Created detailed profile view and controllers. Implement layout and foundation of map view for iOS application. After researching another sprint will be needed in order for the map pins to be added.

**Sprint 7 Report**

**Date:** December 4, 2015

**Attendees:** Luis Castillo, Wilfredo Gomez, Eduardo Garcia, Masoud Sadjadi

**Discussed Topics:**

The Map Pins were added to the map view. JSON handlers were created in order to pass information about the staff. The Map pins were created with actions in order to take the user from the map view pin to the user detail profile view. A management website was created so that manager users can be able to view at a high level the amount of registered staff and employer.

## Appendix D - Sprint Retrospective Reports

**Sprint 1 Retrospective**

**Date:** September 11, 2015

**Attendees:** Luis Castillo, Wilfredo Gomez, Eduardo Garcia, Masoud Sadjadi

**Discussed Topics:**

The initial server setup went well but the configuration of the server to handle emails did not go well. No issues to fix so far.

**Sprint 2 Retrospective**

**Date:** September 25, 2015

**Attendees:** Luis Castillo, Wilfredo Gomez, Eduardo Garcia, Masoud Sadjadi

**Discussed Topics:**

After this sprint meeting, the team and product owner were satisfied with the initial goals, user stories description, and acceptance criteria. In order to improve in future sprints, we will take more time for documentation and testing.

**Sprint 3 Retrospective**

**Date:** October 9, 2015

**Attendees:** Luis Castillo, Wilfredo Gomez, Eduardo Garcia, Masoud Sadjadi

**Discussed Topics:**

After this sprint meeting, the team and product owner were satisfied with the initial goals, user stories description, and acceptance criteria. In order to improve in future sprints, we will take more time for documentation and testing.

**Sprint 4 Retrospective**

**Date:** October 23, 2015

**Attendees:** Luis Castillo, Wilfredo Gomez, Eduardo Garcia, Masoud Sadjadi

**Discussed Topics:**

After this sprint meeting, the team and product owner were satisfied with the initial goals, user stories description, and acceptance criteria. In order to improve in future sprints, we will take more time for documentation and testing.

**Sprint 5 Retrospective**

**Date:** November 6, 2015

**Attendees:** Luis Castillo, Wilfredo Gomez, Eduardo Garcia, Masoud Sadjadi

**Discussed Topics:**

After this sprint meeting, the team and product owner were satisfied with the initial goals, user stories description, and acceptance criteria. In order to improve in future sprints, we will take more time for documentation and testing.

**Sprint 6 Retrospective**

**Date:** November 20, 2015

**Attendees:** Luis Castillo, Wilfredo Gomez, Eduardo Garcia, Masoud Sadjadi

**Discussed Topics:**

After this sprint meeting, the team and product owner were satisfied with the initial goals, user stories description, and acceptance criteria. In order to improve in future sprints, we will take more time for documentation and testing.

**Sprint 7 Retrospective**

**Date:** December 4, 2015

**Attendees:** Luis Castillo, Wilfredo Gomez, Eduardo Garcia, Masoud Sadjadi

**Discussed Topics:**

After this sprint meeting, the team and product owner were satisfied with the initial goals, user stories description, and acceptance criteria. In order to improve in future sprints, we will take more time for documentation and testing.

# References

You must reference any work that is not your own.