

EAGLE SOCIAL

Project Code Documentation 4/17/2018

Lacy Simpson, Jody Bailey, Michael Jason Pearson, Jeremiah Carr

To The Top Apps | CSC 317 – Object Oriented Programming | Group Project

Table of Contents

Group Collaboration - Overview	4
Technologies	4
1.1 Project Code	4
1.2 Project GUI Screen Shots.....	4
Jody Bailey.....	5
2.1 Newsfeed	5
2.2 Friends List.....	5
2.3 Login	5
2.4 Sign-Up	5
2.5 Screen Shots	6
Lacy Simpson	7
3.1 Various Tasks	7
3.2 Profile View Controller	7
3.3 Edit Profile View Controller	7
3.4 Screen Shots	7
3.4.1 New Eagle Social Logo	7
3.4.2 Profile View Controller	8
3.4.3 Edit Profile View Controller(About Me Label and Text Box)	8
Jeremiah Carr	9
4.1 Account Settings	9
4.2 Screen Shots	10
Jason Pearson.....	11
5.1 Message List View Controller	11
5.2 Class Message	13
5.3 Chat View Controller	14
5.4 Class Conversation	14
Various Tasks 4/10/18	14

Group Collaboration - Overview

Technologies

Integrated Development Environment: Xcode

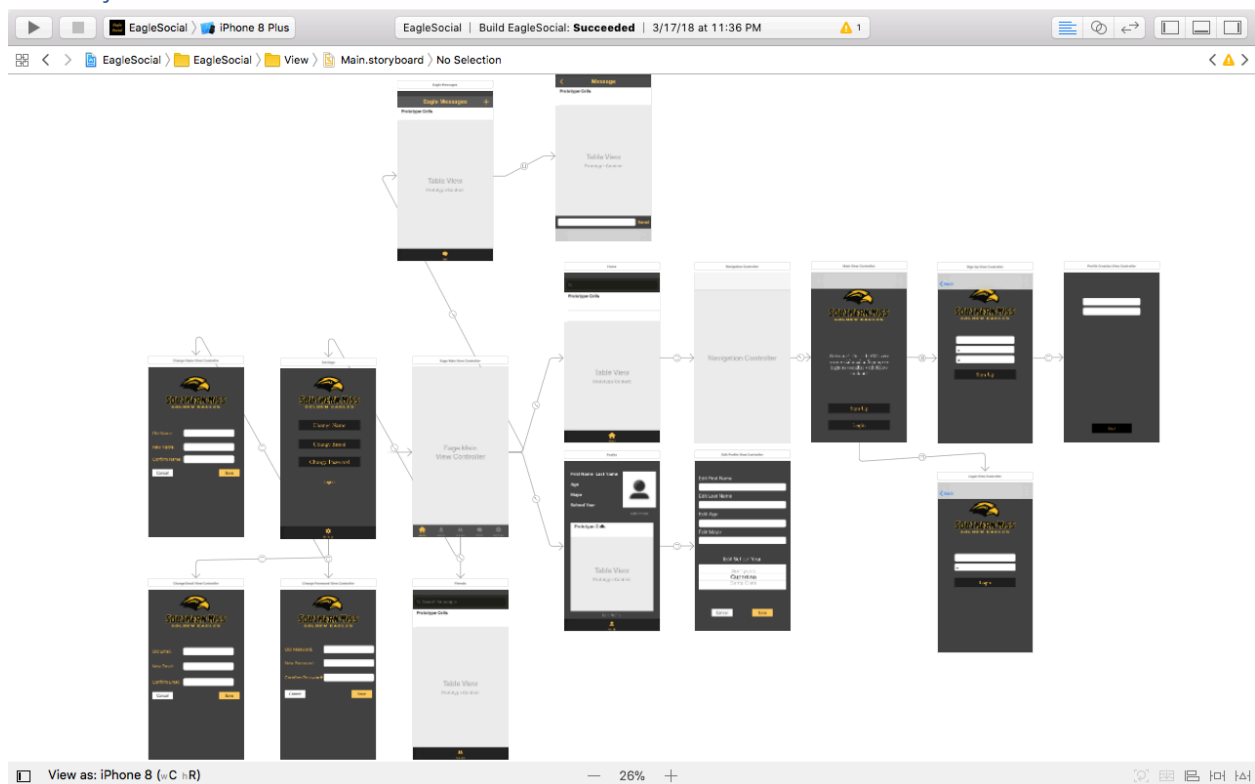
Programming Language: Swift

1.1 Project Code

The project code can be found at the following link.

<https://github.com/jody-bailey/EagleSocial>

1.2 Project GUI Screen Shots



Jody Bailey

2.1 Newsfeed

Went through all of the code and deleted all of the unused code as well as all of the old commented out code that was not being used. I still need to look into a rare exception that is being thrown when the user logs out and another user is logged in. I also still need to modify the news feed to only display posts from the user's friends list. This should be done this week.

2.2 Friends List

Deleted all of the unused code and removed all of the old commented out code that we do not need. I also made it so that when a user accepts a friend request then both users friends list will update accordingly. I still need to make sure that there is no longer a duplication issue that we were having previously but I think that is resolved.

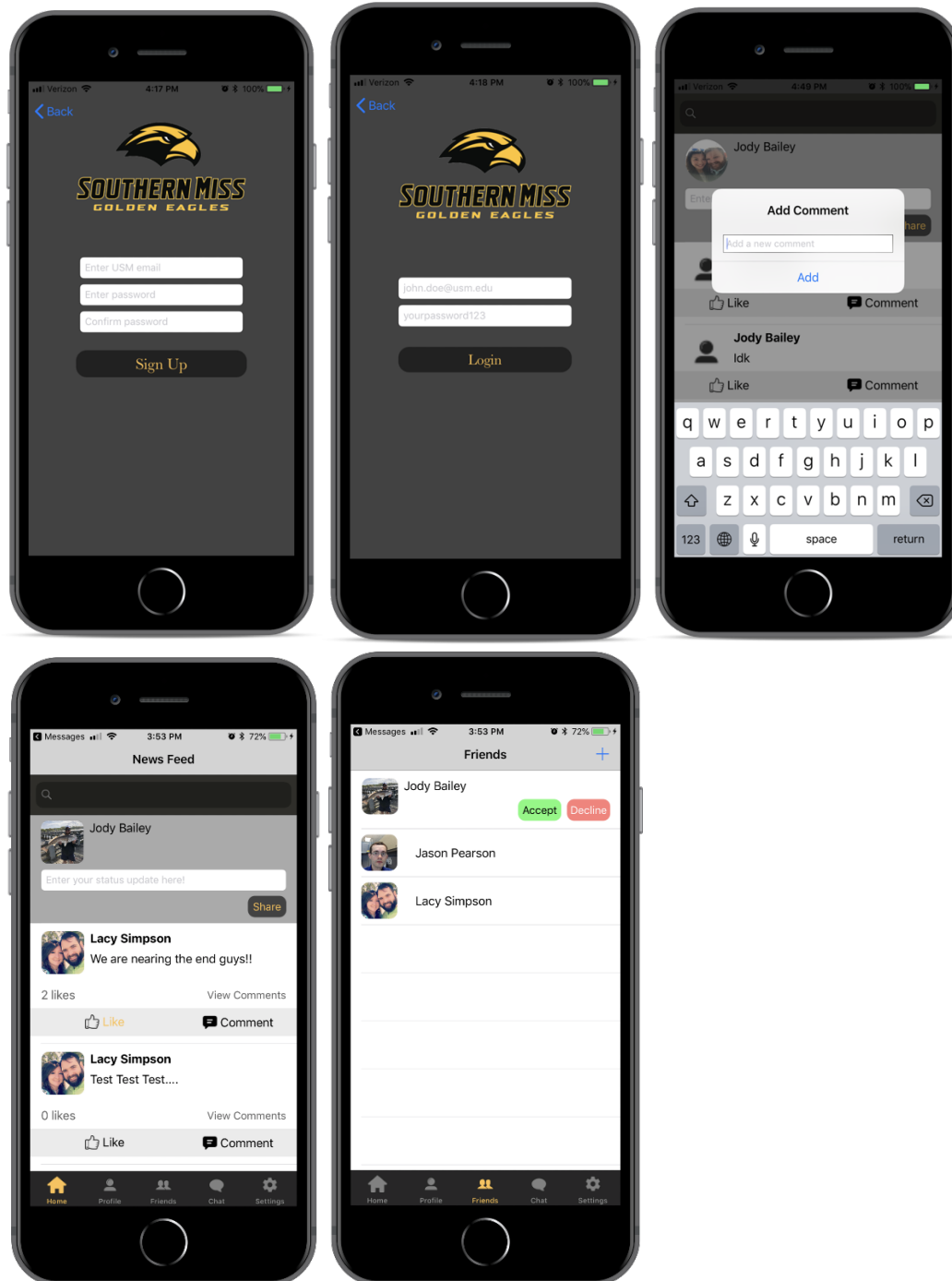
2.3 Login

Need to add a checkbox to the login screen that the user can check to stay logged in. I also need to add a forgot password button so that the user will be allowed to change their password from the login screen.

2.4 Sign-Up

Need to add security questions for the user to answer when they create an account so that they can reset their password from the login screen.

2.5 Screen Shots



Lacy Simpson

3.1 Various Tasks

My various tasks for the week involved creating a new logo for our home screen and app icon. I used new software (Affinity Designer). After coming to an agreement with the group, we decided on a design, and I updated the application to display the new image.

3.2 Profile View Controller

My main focus the week was getting the table view to display the newsfeed on the profile vc. To begin with, I was displaying all users posts in the news feed, but after some guidance from Jody and a few YouTube tutorials, I got the table view to display only the current users posts. I have a minor issue to fix which is causing the users posts to replicate and I still need to enable the comment segue to allow the user to view comments for each posts. I also created an about me custom cell to display as the first cell of the newfeed table view. I had some constraints issues, but Jason and Jody helped me resolve that issue.

3.3 Edit Profile View Controller

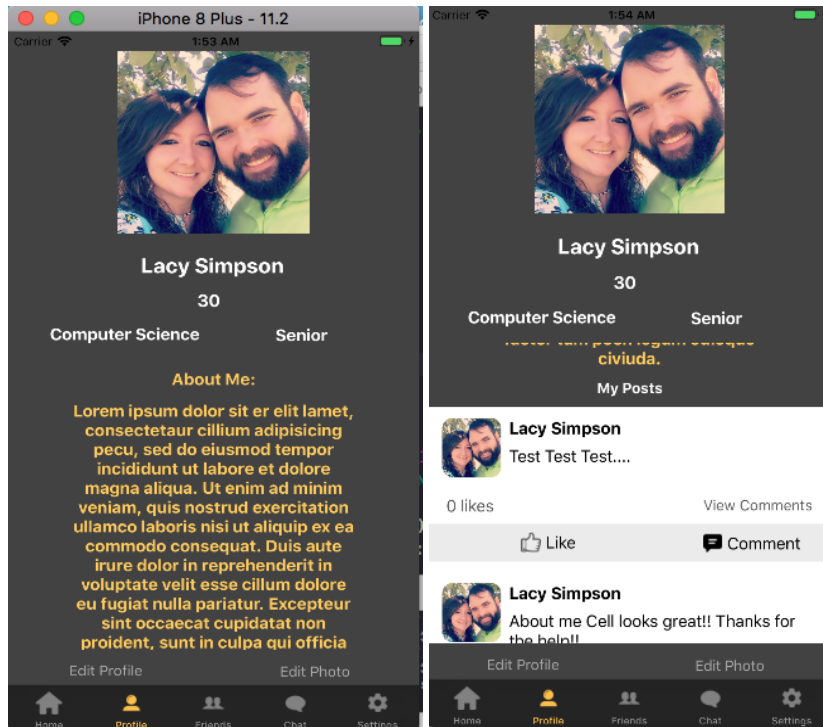
I added a text box and label to the edit profile vc to allow the user to add their about me text to their profile page. Currently the about me is not connected to the database, but hopefully I will have that functionality working this coming week.

3.4 Screen Shots

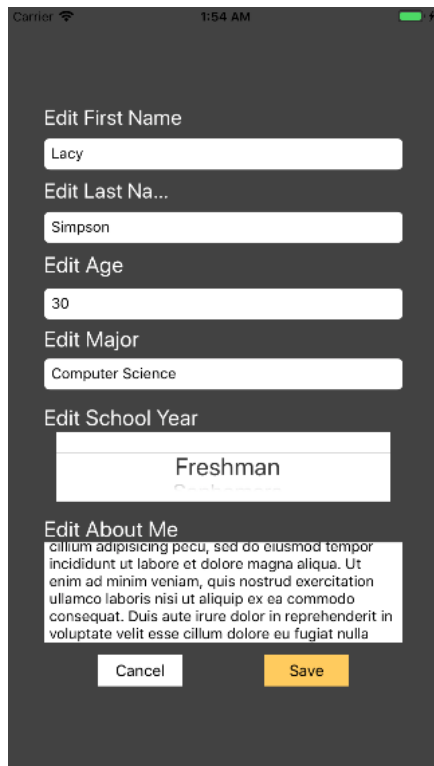
3.4.1 New Eagle Social Logo



3.4.2 Profile View Controller



3.4.3 Edit Profile View Controller(About Me Label and Text Box)



Jeremiah Carr

4.1 Account Settings

This week, I have been trying multiple things for the settings page. The code that I used compiled for the first time without any problems. And I ran into some computer problems involving xcode, but I have them resolved. My idea is when the user goes to the settings page, the user will have to type in the old password, then the new and retype the new password again, when they click save the information will update in firebase.

```
import UIKit
import Firebase
import FirebaseAuth

class SettingsViewController: UIViewController {

    @IBOutlet weak var logoutButton: UIButton!
    @IBOutlet weak var changeNameButton: UIButton!
    @IBOutlet weak var changeEmailButton: UIButton!
    @IBOutlet weak var changePasswordButton: UIButton!
    override func viewDidLoad() {
        super.viewDidLoad()

        self.changeNameButton.layer.cornerRadius = 15
        self.changeEmailButton.layer.cornerRadius = 15
        self.changePasswordButton.layer.cornerRadius = 15

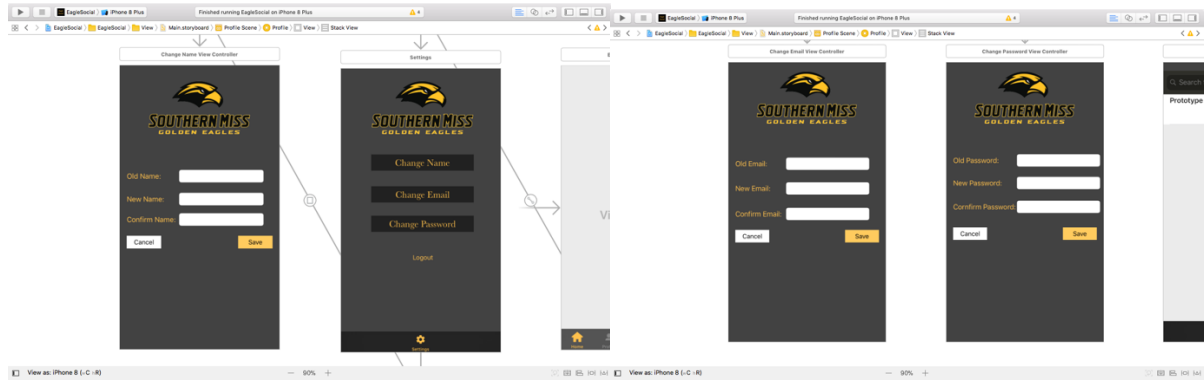
        // Do any additional setup after loading the view.
    }

    override func didReceiveMemoryWarning() {
        super.didReceiveMemoryWarning()
        // Dispose of any resources that can be recreated.
    }

    @IBAction func logoutPressed(_ sender: UIButton) {
        do{
            try Auth.auth().signOut()
        }
        catch {
            print("Error signing out!")
        }

        UserDefaults.standard.set(false, forKey: "isUserLoggedIn")
        UserDefaults.standard.synchronize()
    }
}
```

4.2 Screen Shots



Jason Pearson

5.1 Message List View Controller

The main part of the application that I worked on was the Message List and the Message screens. I setup the Graphical User Interface for both screens and added auto layout constraints so that they would adapt to any screen size. I added some code to the MessageViewController so that the message textbox and send button would rise to the middle of the screen when the keyboard comes into view. In addition to that, I made sure to dynamically get the height of the keyboard depending on the size of the device so that it will accommodate any device.

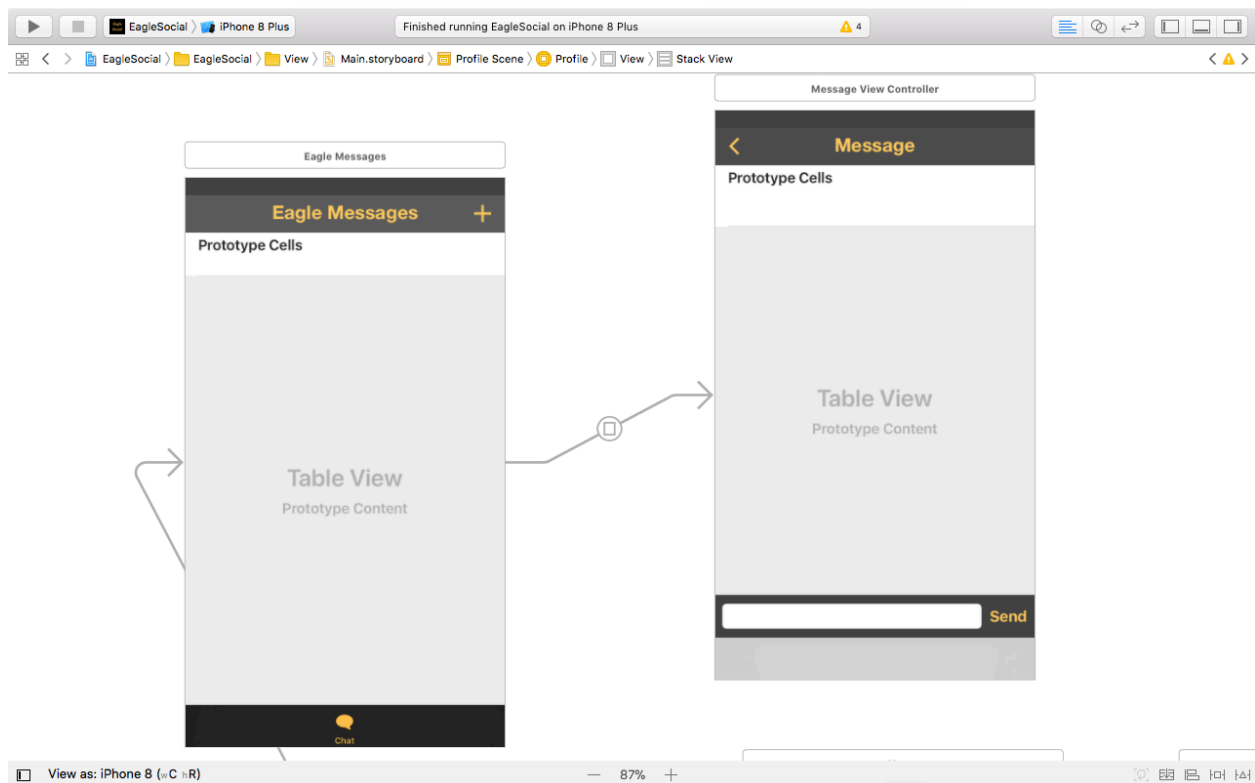
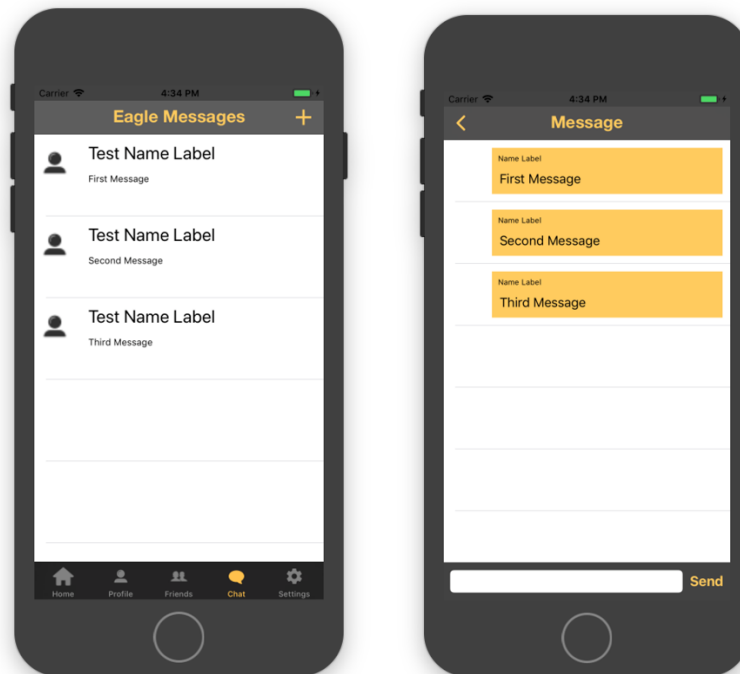
I began creating the Message Class; however, it is only partially complete. I did populate the MessageList and Message screens with some data so that I could test the appearance of the table views; however, it is not connected to the Firebase database yet. That is the next step.

3/27/2018 Jason Pearson:

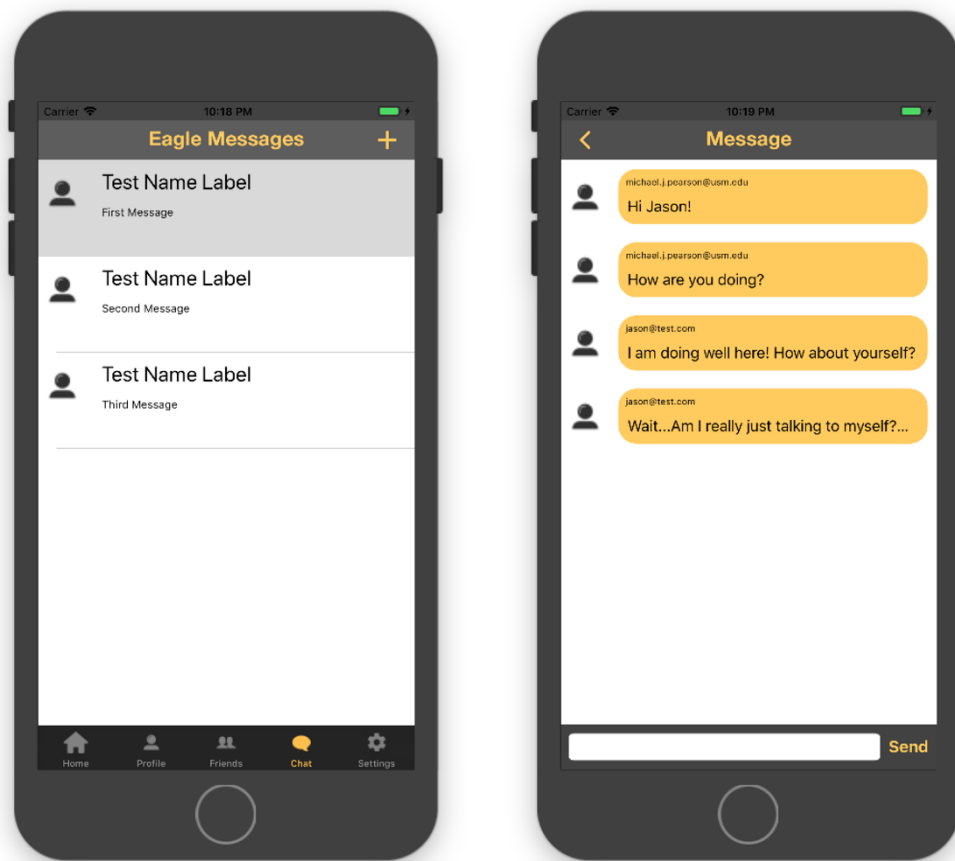
- Researching to figure out the firebase code integration took up a great deal of time this week.
 - Not only did it take up a lot of time, but once I somewhat understood how the code syntax worked I then had to plan out the data structure.
 - Firebase is a NoSQL JSON database and supports nesting data. Traditional relational table based format is not supported by firebase. I did try to keep the number nested levels to a minimum though.
 - I went with the following structure:
 - Conversation
 - Unique ID
 - Members
 - User ID : bool
 - Messages
 - Unique ID
 - Message Body
 - Sender Id
 - NOTE: I am not a fan of NoSQL database anymore. Their querying power is very limited.
 - Fixed an issue with the UITapGestureRecognizer so the function to dismiss the keyboard would work.
 - Rounded the corners on the message bubbles so the message list looks more like a messaging screen.
 - Added a generic profile image to each message retrieved at least until the profiles are built out complete with profile picture that I can pull from.
 - Removed the separator so there will be no lines in between the messages.
 - Migrated and refactored some of the firebase code for sending new messages away from the view controller and put it in the model.
 - Added more comments and spacing for improved code readability.

5.2.1 Screen Shots

3/20/2018 Jason Pearson



3/27/2018 Jason Pearson



5.2 Class Message

5/27/2018 Jason Pearson

- Made all the attributes private
- Added 3 new attributes: conversationID, messageDictionary, and members
- Added initializer function and overloaded initializer function.
- Added sendMethod method.
- Added getters for SenderId, messageBody, conversationID, messageDictionary, and members.
- Added setters for SenderId, messageBody, and messageDictionary.

TODO 4/3/2018

- Code up the ChatViewController (Eagle Message screen) to connect to the database.
- Add a tapGestureRecognizer to the ChatViewController to perform segue to the MessageViewController (Message Screen).
- Add a way to select a user to chat with if the + button is clicked on the ChatViewController (Eagle Message Screen).
- Figure out how to default the conversationTableView on the MessageViewController to show the table scrolled all the way to the last message.

5.3 Chat View Controller

4/3/2018

- Coded the ChatViewController to pull a list of conversations down to the Eagle Messages screen.
 - TODO: Need a method to grab the “Other User’s” display name. I have tried several things but have not been successful at reliably pulling the display name of users other than the current user down.
- Added a way for the user to select a row on the Eagle Messages screen and be taken the messages screen where all the messages in the conversation are displayed.
- Still a good way to go to make the messaging part of the application fully functional.

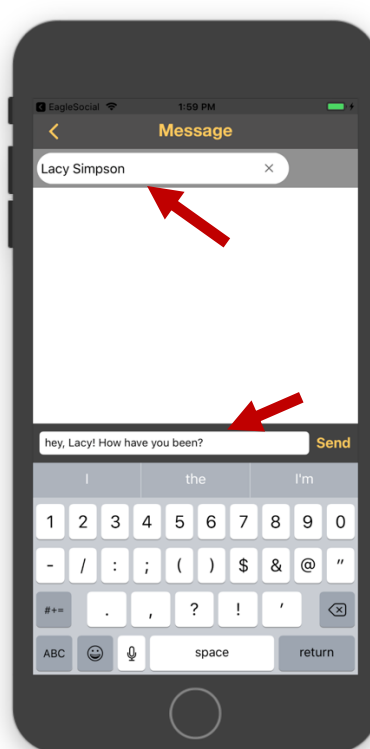
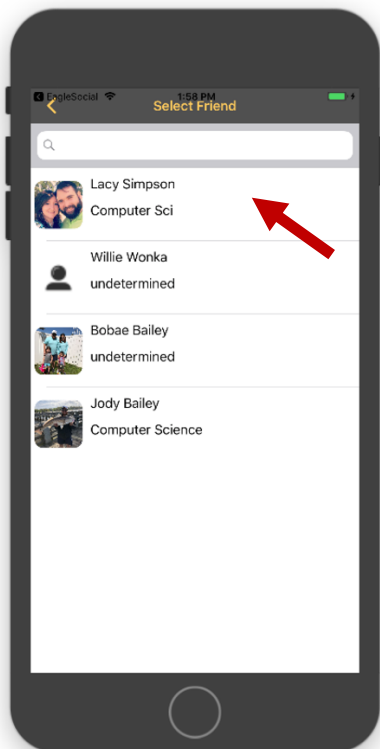
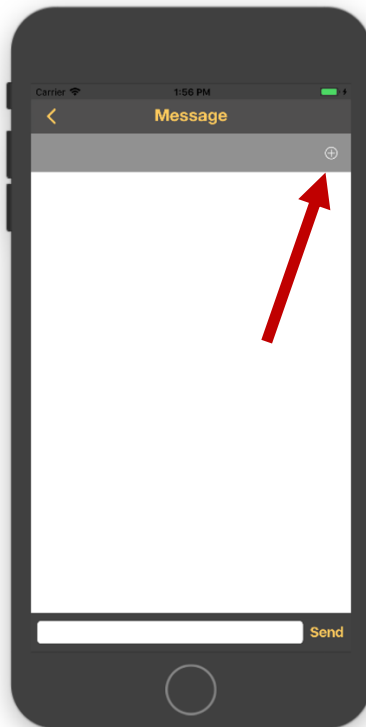
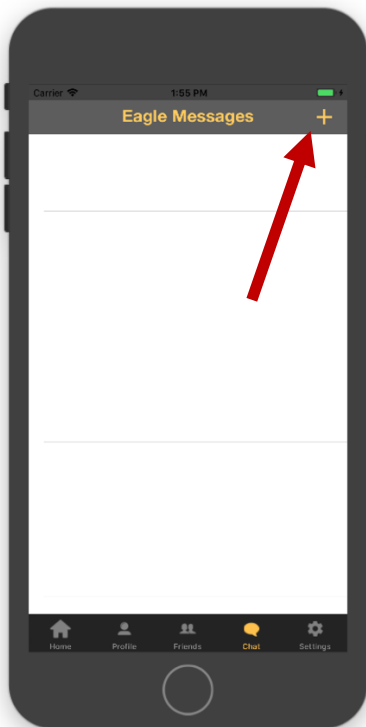
5.4 Class Conversation

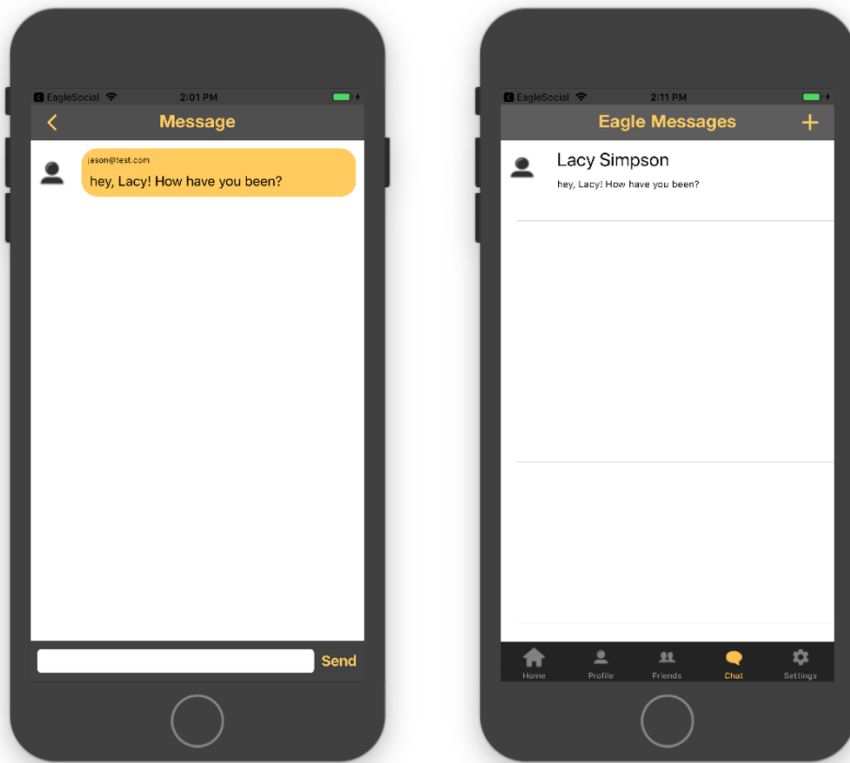
4/3/2018

- Added the conversation class. This aides the ChatViewController in pulling down a list of conversations.

Various Tasks 4/10/18

- Added the selectFriendViewController and used the Friend model Jody created. This view controller is used to select a friend to chat with.
- Updated the Message view controller to include some new UI elements to aid in selecting a friend to chat with. Also, updated it to use the selected friend passed back from the selectFriendViewController.
- Updated the Message model to set the display name on the conversation/members tree.
- Updated the Conversation model get the user’s display name.
- Resolved some merge conflicts I was having on the Main.storyboard file.





4/17/18

- Conversation screen refresh issue #80 - Done
- Messages Screen: When there are a lot of messages on the message screen you have to scroll down to see the latest one. Issue #81 - Done
- Eagle Messages: Pull the users' picture into the conversation screen (Eagle Messages) Issue #106 – Worked on but not completely finished with this issue.
- Eagle Messages: If user attempts to create a new conversation with a person they already have a conversation with, use the existing conversation instead of creating a new one. Issue #105 - Worked on but not completely finished with this issue.

