

Contributions:

Manuel Ponce

- Setup Firebase Cloud Firestore and created a 'User' model that will help store our users information.
- Added to Sean's register screen
 - Added backend code for when a user successfully registers, we save that user to the Firestore database.
- Worked on Display profile page
 - Made sure it segues correctly to view profile when profile icon is clicked.
 - Setup a default image to show future user profiles when they upload one.
 - Displaying user information being pulled from the Firestore database.
- Edit profile
 - Made sure it segues correctly to edit profile from view profile.
 - Set up a dummy page. No functionality is implemented yet
- Delete account page
 - Made sure it segues correctly to delete account from edit profile.
 - Set up a dummy page. No functionality is implemented yet.

Sean Chen

- Set up Firebase, Firebase Auth, Pods
 - Had similar errors to the homework due to new firebase update, followed the guide [here](#) to fix it
 - Added everyone to the Firebase project as owners
- Implemented Login Screen and Register Screen
 - Uses Firebase email + password authentication
 - Added Firebase Display Name to regular email + password authentication
 - Might add email confirmation, sms confirmation in the future, but for now it allows for fake emails during testing
 - Might restrict to only UT emails in the future as well?
- Implemented Create a Game and Selected Game views
 - Was originally assigned to Huy, but there were some Xcode issues late on Thursday
 - Create a game fills out the necessary information to create a game.
 - Locations are limited to a few selections, may expand to use Apple Maps, CoreLocation in the future
 - Types of sports are limited to a few selections for now.
 - For some reason UIDatePicker seems to be a little buggy graphically, it's stuck on the lower edge in the simulator.
 - Clicking on a game in the table view will open a selected game view
 - Displays game information

- Will implement chat and invite functionality in the future
- For now, it does not save to a cloud, Firestore to be implemented next stage
 - Firestore and accounts on Firebase will allow for participants and chat to be implemented next stage
 - RSVP/Join games will be implemented w/ Firebase as well (when you can see other people's games)

Reagan Lasswell

- Worked on main page for upcoming games
 - Started functionality to display games with table view
 - Set up layout of page and segues from main page for storyboard
 - Set up functionality of segues
 - Created game.swift file to set up properties and be able to communicate with database
- Selected Game page
 - Set up a dummy page. No functionality implemented
- Created a games collection on Firebase Cloud Firestore that will help store our game information.

Huy Nguyen

- Didn't contribute to the project, was supposed to work on the pages for creating games.
- Xcode was not up to the latest version where the project was built on so I had to reinstall but I wasn't able to install the latest version on my MacBookPro9,2.

Deviations:

- Overall graphics and GUI is a work in progress, more graphics and design work will be done next stage.
- We need to research a clean tool bar for the bottom buttons on the main home page. Along with other page layouts, we need to research sleek design techniques and strategies in order to recreate our design document illustrations.
- Currently the edit profile page is being shown as a modal view because we are unsure why it does not present in a full screen even after changing to segue to "show".
- Current text font used throughout the app is also not the one we used in our design. We have to find another similar one and import it to be used.
- Current height of the modal views is not reflecting correctly to what the design is showing. Currently they go almost to the top of the screen because we have not looked into a way to restrict the height given for the view.

- Original project proposal didn't explicitly assign cloud/database functionality to Alpha, but regardless it's not currently available and will be implemented next stage