

Cart Squad Beta README

Team #2

- Jose Alonso Rodriguez
- Alejandro Cisneros
- Justin Lee
- Angel Nevarez

Contributions:

- Alejandro Cisneros (Release 40%, Overall 40%)
 - Main Cart Screen
 - Add cart items screen
 - Add custom items to cart screen
 - Create new custom item for cart screen
 - Firebase firestore - store custom cart items, observe real-time database changes and update UI accordingly
 - Firebase cloud storage - store custom item images in firebase cloud
 - Helped merge cart screens into alpha branch
 - Created CartSquad Web API backend (<http://3.138.34.17/>)
 - Wrote a python script to get real product data from Target RedCircle API (stored in our own database)
 - Used an AWS EC2 instance for hosting
 - Created postgresSQL database on instance
 - Created flask app as web API
 - Used Gunicorn to handle incoming requests with flask app
 - Used nginx as a reverse proxy to forward incoming requests to custom Gunicorn service
 - Defined /api/products/search endpoint for searching by products by string matching with product name (ex: <http://3.138.34.17/api/products/search?searchTerm=orange>)
 - Handled deleting carts with Firestore synchronization
 - Edge case handling: cart deleted while others are in app
 - Ensured cart image was deleted on Firebase Storage upon deletion
 - Fixed CartScreenVC UI reload every frame to eliminate lag
 - Implemented items search for current store on CartAddItemsScreenVC (utilizing backend)
 - Handled updating UI for adding users to cart while in cart
 - Handles updating cart related UI while in cart when cart settings changed
 - Implemented editing cart settings after cart creation and synchronization with Firestore

- Implemented getting accurate cart info from Firestore for user on main menu on app startup (and syncing state if cart info was edited)
- Justin Lee (Release 20%, Overall 20%)
 - Set up project and git repo
 - Add cart screen
 - Add store screen
 - Search store screen
 - Cart and store objects
 - Incorporate main menu with new carts
 - Friend screens
 - Incorporated a friends screen that shows all available users to add and current friends
 - This functionality is implemented to allow for searching through all users in the future
 - Added friends container in firebase and worked out pathing to reuse users data
 - Bug that required the use of a completion handler and then another escaping closure to load users and friends sequentially
 - Adding friends
 - Added listeners to provide real time updates to the screen depending on added or modified friends
 - Removing friends:
 - Handled removing friends including updating firebase
- Angel Nevarez (Release 30%, Overall 20%)
 - Splash loading screen
 - Login Screen
 - Create account screen
 - Firebase authentication - store the username and password of different users created and be able to use the authentication to login to the app.
 - Helped with merging the branches into a main alpha branch
 - Segued the login screen and create account screen into the Main Menu Screen
 - Created a personal info and add profile screen that would be further developed in beta.
 - Added personal details view controller that stores the user information such as the name, address, state, and zip code into a firestore user collection.
 - Changed the "phone number" text field from the alpha build into the name text field.
 - Created a tab controller that has 3 segues into 3 different view controllers:

- A Navigation controller that segues into the add cart view controller as it's root controller
 - A second navigation controller that segues into the edit profile view controller.
 - A 3rd navigation controller which segues to an empty view controller which is a placeholder for a create Items view controller and functionality.
 - Added assets and text into the tab bar and modified it such that the image assets fit inside the tab bar as well as changing the color of the text and image when selected.
 - Created a profile view Controller.
 - The text field contains the stored user information from the firestore, if the information is not stored, then the text field would be left blank and the name would simply be "name"
 - If the user pressed the button, then the stored user information in the firestore database would also be updated with the new information.
 - Modified the create account view controller such that there would not be duplicate usernames stored in the firestore database.
- Jose Alonso Rodriguez (Release 10%, Overall 20%)
- Main Menu Screen
 - Cart Member Settings Screen (Present in storyboard but not yet merged)
 - Setting up XCode merging workflow (we did not use traditional git pull requests)
 - Merged login screens with main menu screen
 - Adding friends to cart
 - Modal segue VC
 - Friend class with static functions that retrieve information from firestore in order to create friend data table
 - Merging debugging
 - Race conditions debugging
 - Dealt with bug for the longest time without realizing closure being passed onto Firestore function was running in a separate thread leading to race conditions
 - Solution: pass another closure to execute inside of the closure!
 - Ensures select code is only run after thread finishes work
 - Helped debug similar issues

Differences:

- No implementation for member settings, removing cart members, privileges or budgets.
 - Assigned member had personal issues for a week. Began implementation but did not finish.

- Other non-owner/admin cart users can edit cart settings at the moment.
- Friend Requests/Searching for users not yet implemented
 - Several bugs kept us from reaching this point
- Cart deadline does not yet do anything and history demoted to stretch goals
- Notifications not implemented yet; moved to final build