**Spending Limit App**

1. **Definition Statement:** Put a self-imposed check on your spending habits. This app is for the people who want to limit their spending but go crazy when they are in a supermarket or a shopping mall.
2. **List of features you implemented**
   1. Fetching the data dynamically from Walmart API.
   2. Parsing the returned the data using SwiftyJSON and creating a Product model.
   3. Ability to scan the QR code and search the Walmart API using that query.
   4. App takes in the amount set as limit by the user and compares it with the total price in the cart and alerts if the total price exceeds the limit.
   5. User limit set as NSUserDefaults so that data persists if app is closed.
   6. Items in cart set as NSUserDefaults so that data persists if app is closed.
   7. Implemented different map than Apple’s default Maps - Google Maps for iOS to show the store location.
3. **Wish list of features you could put in the next version.**
   1. Make the app more dynamic in terms of fetching the data based on different categories sent by user.
   2. Choosing the store (using different API) based on the geographical location of the user.
   3. Ability to stop user from adding more items in the cart if the limit has been reached.
4. **Self evaluation and documentation**
   1. **Grade:** I would grade myself at A- in terms of the effort I have put in to research the stuff I could have used for my project out of which I ended up using a few things. I would rate my app at B or B+ on how it currently works The app is currently working in separate individual pieces and is able to somewhat achieve what I tried but it is not a finished product. It needs integration between different view controllers.
   2. **Describe** what each class does and how the pieces fit together. A UML diagram is optional.
      1. Product – Model for the items to be used across the app.
      2. Products – Model for the list of items.
      3. jsonProducts – Model to store data when fetched from API.
      4. GMapViewController – View controller to show google map.
      5. SetLimitVC – The initial view controller. Used by user to set the limit.
      6. QRCodeVC – The second view controller for scanning QR code or manually entering search query.
      7. ProductTableVC – The tableview to show the data in a table for the products fetched from API.
      8. ProductDetailedTableVC – The detailed tableview for showing detailed product info.
      9. CartTableVC – The actual shopping cart which collects items added by the user. Originally intended to be the shopping cart but I wasn’t able to send in the item price with item name and loop through the cells to get a total.
   3. **Above and beyond** – I think the above and beyond was that I ended up doing a lot of research on using other methods than NSUserDefaults to persist the data. I tried using Realm for Swift and Core Data but I guess due to time constraints I couldn’t fully utilize them. I researched and ended up using google maps for ios. Using data from an API and presenting in a table format.
   4. **Third-party frameworks, code snippets and tutorials:** 
      1. Google Map for iOS - <https://developers.google.com/maps/documentation/ios-sdk/>
      2. QRCode Reader Using Swift - <http://www.theappguruz.com/blog/qrcode-reader-using-swift>
      3. SwiftyJSON - <https://github.com/SwiftyJSON/SwiftyJSON>
      4. Walmart API - https://developer.walmartlabs.com/
      5. Cocoa Pods for Google Maps - <https://cocoapods.org/>