Swagafied

IOS Application

User Guidelines

V 1.

**System Requirement:**

OS version: iOS 9+

Xcode Version: Xcode 8+

Swift Version: Swift 3+

**Application Overview:**

Swagafied iOS Application consists of 3 main sections. They are

1. Search Tool
2. User Profile Tool
3. Account Tool

Search Tools gives you the facility to search your favorite kicks over thousands of available kicks. You can perform color wise, brand wise, Model wise search on this page.

User profile tools show your swagafied Portfolio, Watch list section along with Global timeline page.

Account Details consists for few sub sections like Notification page, Edit user profile page, Edit Password page, Terms Of Use page, Privacy Policy page and Login/Log out Page.

**Search Tool :**

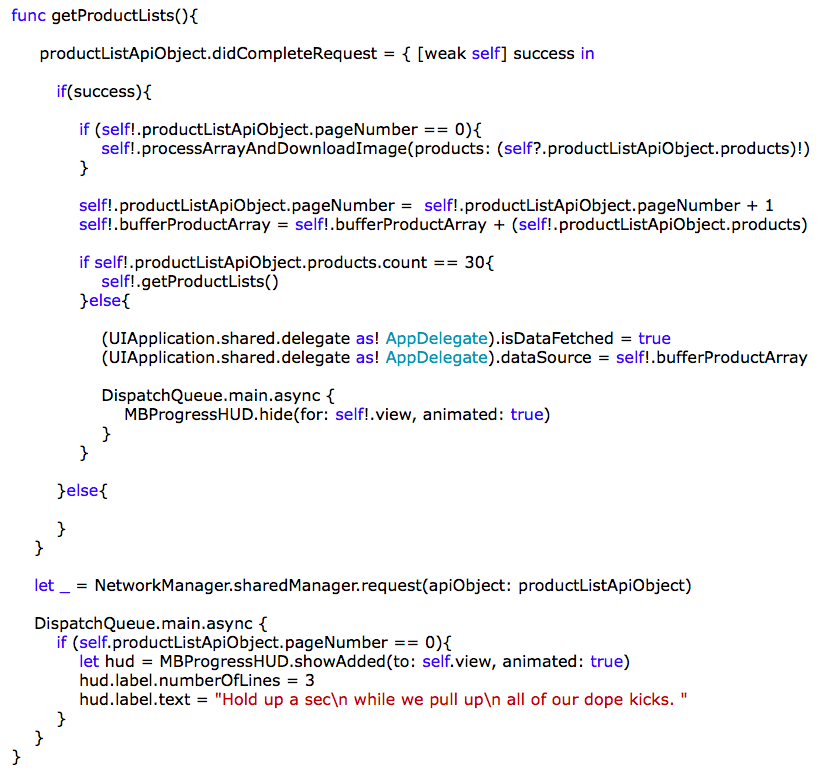
Search page lists you all the products currently available in the remote server. Application enables you to like, unlike, share, add to your watchlist, remove from your watchlist, add to your portfolio and remove from your portfolio function for each of the products. You can move to product detail page by clicking the product image of info icon.

Here you can also perform a search functionality like using product Brand, Product Model name, Product Color.

Filter functionality provides you to select and search over collection currently available. Like User can select Adidas and KD and hit “MAKE IT HAPPEN”, Application will show currently available Adidas and KD kicks.

Filter like wise button filters all the kicks that user has liked currently.

Function for receiving data from server:

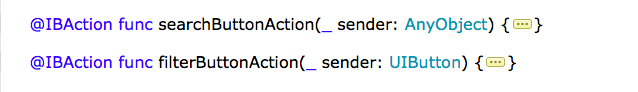


Network class responsible: ProductListAPI.Swift

API Endpoint:

"http://webolation.com/sneakers/web\_service/ product\_list "

Funtions are responsible for search and Filter:



**User Profile Tool:**

Timeline:

Timeline Page shows you all recent activities that has been performed by all the Swagafied users. Activities can be liking a Sneaker, Posting a Review, Add Sneaker to portfolio, Adding a Sneaker to Watchlist. From Timeline page user can also like a sneaker, Move to Sneaker detail page, can add to has/her watchlist. Comment button along with sneakers enables user to post a comment for a particular timeline object.

View all comments shows all the comments currently available for that particular sneaker. User can post his/her comments in All comments section too.

The share button enables user to share the Kick over all currently available options.

Like button can perform like or unlike the sneaker depending upon the current like state. If user haven’t liked it yet then he can perform Like functionality else User can perform dislike functionality.

Functions for receiving Timeline data:

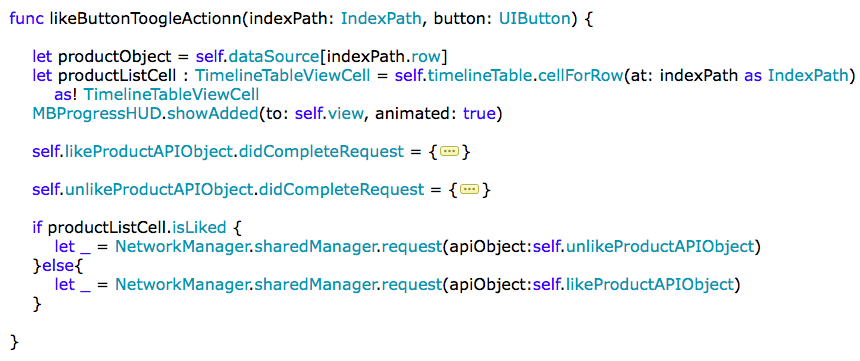


Network class responsible: TimelineAPI.Swift

API Endpoint:

"http://webolation.com/sneakers/web\_service/ timeline"

Function for liking the product:

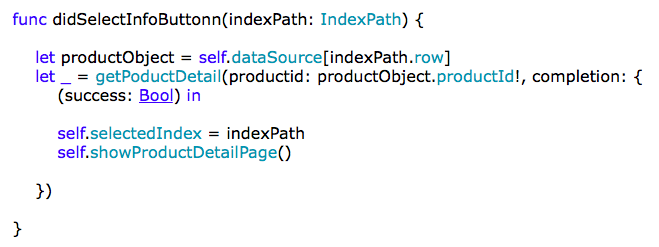


Network class responsible: LikeProductAPI.Swift

API Endpoint:

"http://webolation.com/sneakers/web\_service/ like\_product"

Function for fetching details about the product:



Network class responsible: ProductDetail.Swift

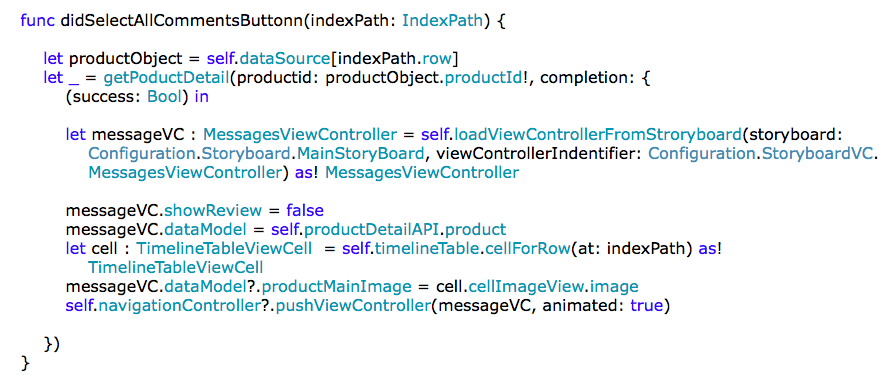
API Endpoint:

"http://webolation.com/sneakers/web\_service/ product\_details"

Share button Function:



Comment Button Function:

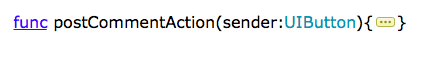


Network class responsible: AllCommentList.Swift

API Endpoint:

"http://webolation.com/sneakers/web\_service/ comment\_list"

Posting a comment:



API Endpoint:

"http://webolation.com/sneakers/web\_service/ post\_comment"

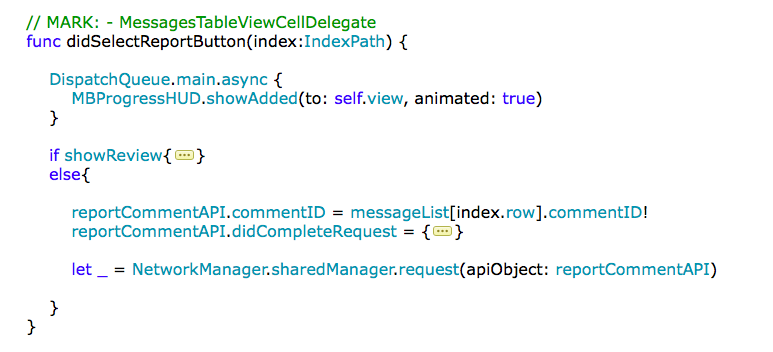
Liking a comment:



API Endpoint:

"http://webolation.com/sneakers/web\_service/ like\_comment"

Report a comment:



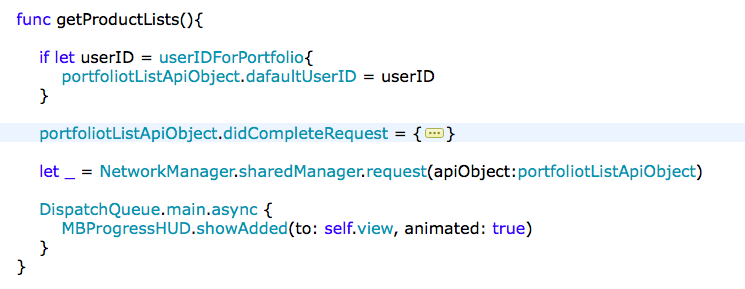
API Endpoint:

"http://webolation.com/sneakers/web\_service/ report\_comment"

Portfolio:

Portfolio Page shows all the product list that user has added to his/her portfolio. Clicking on each product leads to portfolio details page. Here user can modify the size and price of the sneaker. Along with modifying user can share, add to watchlist, like the sneaker.

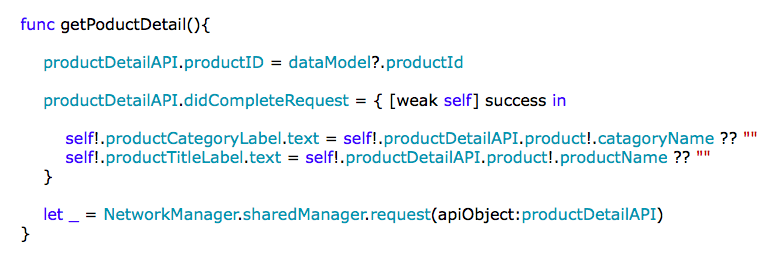
Fetching portfolio product list:



API Endpoint:

"http://webolation.com/sneakers/web\_service/ report\_comment"

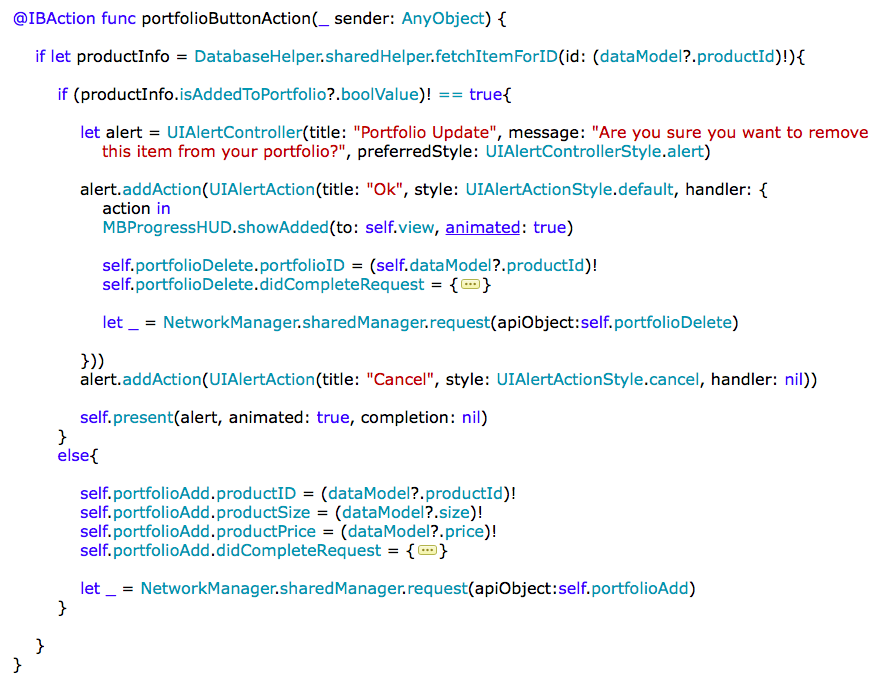
Fetching details for a particular product:



API Endpoint:

"http://webolation.com/sneakers/web\_service/ product\_details"

Adding/ Removing Product from User Portfolio:

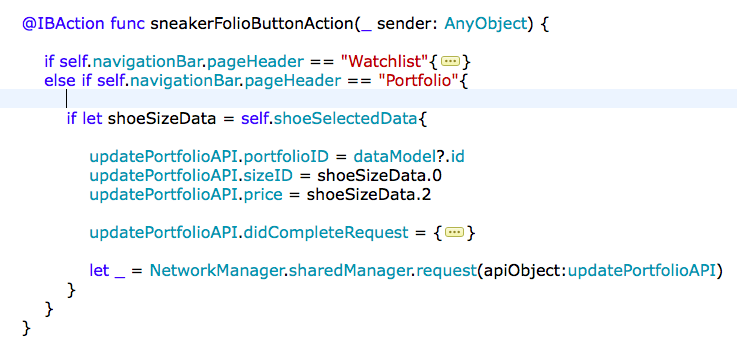


API Endpoints:

"http://webolation.com/sneakers/web\_service/ remove\_portfolio"

"http://webolation.com/sneakers/web\_service/ add\_portfolio"

Updating Senaker portfolio data:



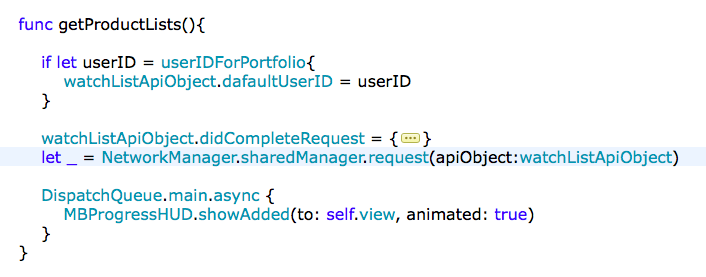
API Endpoint:

"http://webolation.com/sneakers/web\_service/edit\_portfolio"

Watchlist:

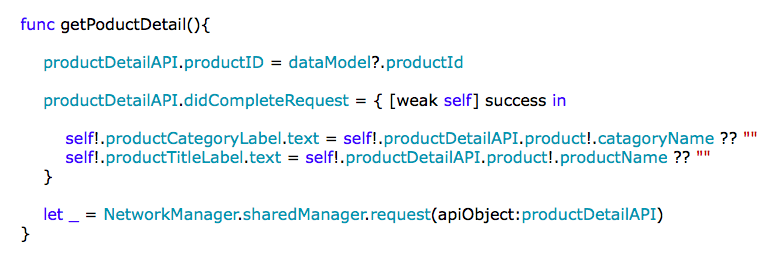
Watchlist Page shows the entire product list that user has added to his/her watchlist. Clicking on each product leads to watchlist details page. Here user can modify the size and price of the sneaker. Along with modifying user can share, add to portfolio, like the sneaker.

Fetching watchlist product list:

API Endpoint: 

"http://webolation.com/sneakers/web\_service/watchlist"

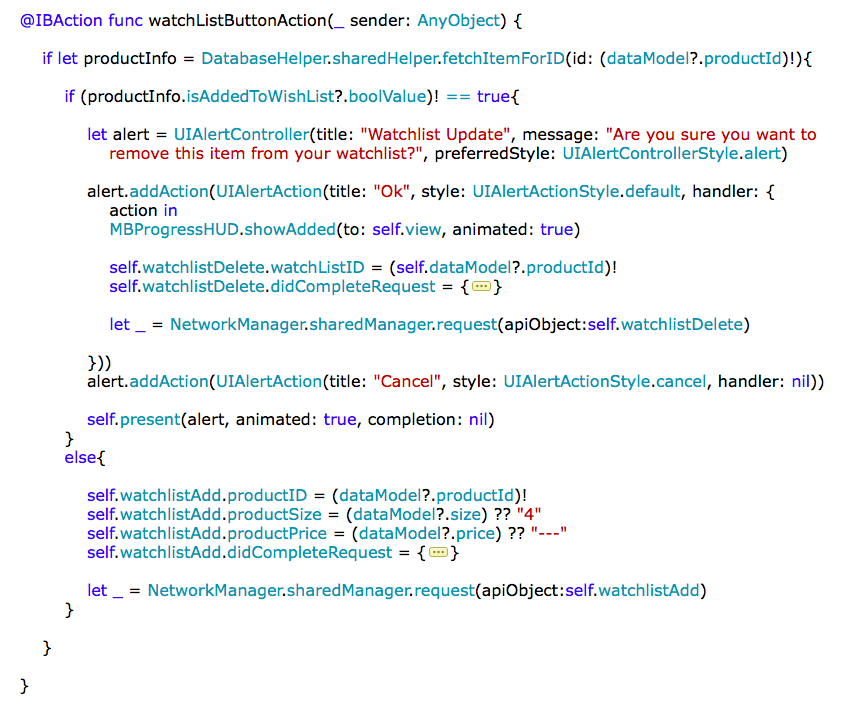
Fetching details for a particular product:



API Endpoint:

[http://webolation.com/sneakers/web\_service/ product\_details](http://webolation.com/sneakers/web_service/%20product_details)

Adding/ Removing Product from User Watchlist:

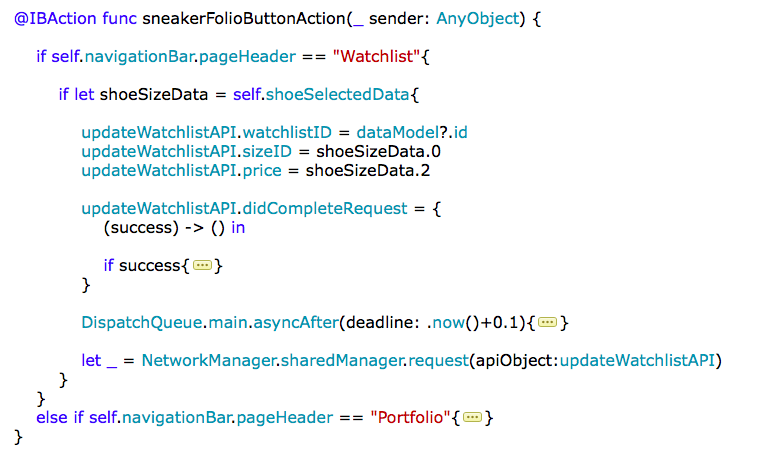


API Endpoints:

"http://webolation.com/sneakers/web\_service/remove\_watchlist"

"http://webolation.com/sneakers/web\_service/add\_watchlist"

Updating Sneaker watchlist data:



API Endpoint:

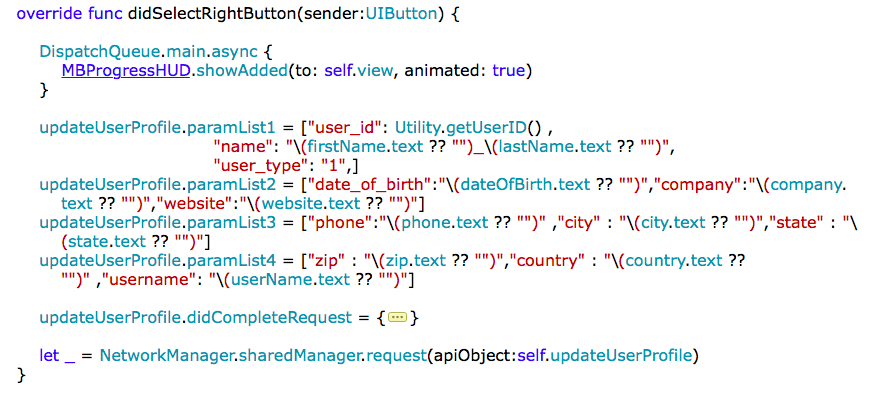
http://webolation.com/sneakers/web\_service/edit\_watchlist

**Account Tools :**

Edit Profile:

Edit profile page allows user to modify all user related information like Updating Name, User Name, Date of Birth, Company info, website address, Phone number and Address. Along with all this information user can modify his/her profile picture by simply tapping on the image.

Updating User Data:



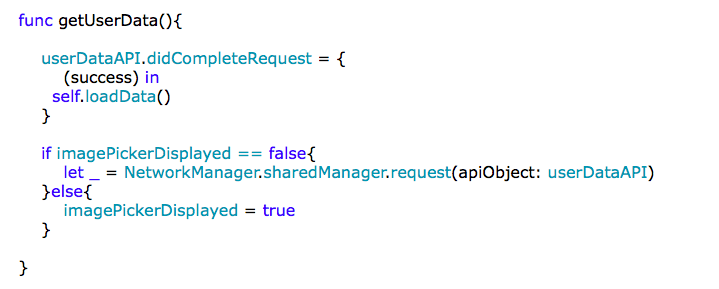
API Endpoint:

http://webolation.com/sneakers/web\_service/update\_user

Upload User Avatar Image: 

API Endpoint: http://webolation.com/sneakers/web\_service/ update\_profile\_picture

Fetching User saved Data:

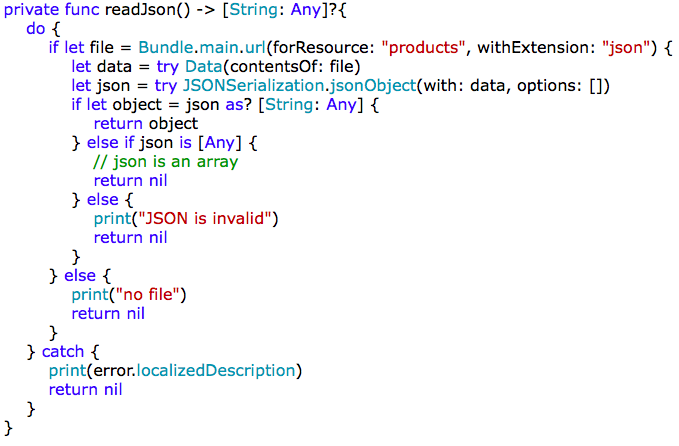


API Endpoint:

<http://webolation.com/sneakers/web_service/edit_watchlist>

InApp DataBase:

Read data From DB:



Update In App DB file:

Find out “products.json” file in file list and update any item. All the items are listed down here.

Terms of Use & Privacy Policy:

This section of the Application informs you about the legal terms, how user should use the application.

Login & LogOut :

One of the most important feature of the application, this enables application to identify the user by its credentials. Logout removes all your personal data by simply wiping off the database.

**Application Level configurations:**

***Updating one signal ID:***

Update “oneSignalAppID” variable in configuration.Swift file.

***Updating Google Ad Configuration:***

Step 1: Update “googleAdAppID” variable in configuration.Swift file.

Step 2: Update “adUnitID” variable in configuration.Swift file.

***Updating Base API URL:***

Update “baseAPIUrlStating” variable in configuration.Swift file.

***Updating API endpoints:***

Modify “path” structure variable inside “API” structure in configuration.Swift file.

***Integrate your own Facebook app:***

1. update “FacebookAppID” in info.plist file.
2. Update Url schemes with your own app id.

***Updating Onesignal, Facebook, Google Ad library:***

Just run “Pod Update” command in Terminal. It will fetch all latest libraries for you.

***Updating Crashlytics:***

1. Select your target > Build Phases > Run script. Add your own script provided by Crashlytics.
2. Open Info.Plist and update the below section by your own API key

