Homework 9: FB Search

1. Objectives

- Become familiar with Xcode, IOS App development and Facebook SDK for IOS.
- Build a good-looking IOS app.
- Build mobile user experience for Facebook Search using the Facebook Graph API
- Add social networking features using the Facebook SDK for IOS

2. Background

2.1 Xcode

Xcode is an integrated development environment (IDE) containing a suite of software development tools developed by Apple for developing software for OS X and iOS. First released in 2003, the latest stable release is version 7.3 and is available via the Mac App Store free of charge for OS X El Capitan users.

Features:

- Swift 2/3 support
- Playgrounds
- Interface Builder
- Testing
- User Interface Testing
- Code Coverage

The Official homepage of the Xcode is located at:

https://developer.apple.com/xcode/

2.2 IOS

iOS (originally iPhone OS) is a mobile operating system created and developed by Apple Inc. and distributed exclusively for Apple hardware. It is the operating system that presently powers many of the company's mobile devices, including the iPhone, iPad, and iPod touch. It is the second most popular mobile operating system in the world by sales, after Android.

The Official IOS home page is located at:

http://www.apple.com/ios/

The Official IOS Developer homepage is located at:

https://developer.apple.com/ios/

2.3 Swift

Swift is a general-purpose, multi-paradigm, compiled programming language created for iOS, OS X, watchOS, tvOS and Linux development by Apple Inc. Swift is designed to work with Apple's Cocoa and Cocoa Touch frameworks and the large body of existing Objective-C code written for Apple products. Swift is intended to be more resilient to erroneous code ("safer") than Objective-C and also more concise. It is built with the LLVM compiler framework included in Xcode 6 and later and uses the Objective-C runtime, which allows C, Objective-C, C++ and Swift code to run within a single program.

The Official Swift homepage is located at:

https://developer.apple.com/swift/

2.4 Facebook Graph API

The Graph API is the primary way to get data out of, and put date into Facebook's platform. It's a lowlevel HTTP-based API that you can use to programmatically query data, post new stories, manage ads, upload photos, and perform a variety of other tasks that an app might implement. To learn more about the Facebook Graph API visit the

2.5 Amazon Web Services (AWS)

AWS is Amazon's implementation of cloud computing. Included in AWS is Amazon Elastic Compute Cloud (EC2), which delivers scalable, pay-as-you-go compute capacity in the cloud, and AWS Elastic Beanstalk, an even easier way to quickly deploy and manage applications in the AWS cloud. You simply upload your application, and Elastic Beanstalk automatically handles the deployment details of capacity provisioning, load balancing, auto-scaling, and application health monitoring. Elastic Beanstalk is built using familiar software stacks such as the Apache HTTP Server, PHP, and Python, Passenger for Ruby, IIS 7.5 for .NET, and Apache Tomcat for Java.

The Amazon Web Services homepage is available at: http://aws.amazon.com/

2.6 Google App Engine (GAE)

Google App Engine applications are easy to create, easy to maintain, and easy to scale as your traffic and data storage needs change. With App Engine, there are no servers to maintain. You simply upload your application and it's ready to go. App Engine applications automatically scale based on incoming traffic. Load balancing, micro services, authorization, SQL and noSQL databases, memcache, traffic splitting, logging, search, versioning, roll out and roll backs, and security scanning are all supported natively and are highly customizable.

To learn more about GAE support for PHP visit the page:

https://cloud.google.com/appengine/docs/php/

3. Prerequisites

This homework requires the use of the following components:

3.1 Download and install Xcode

To develop iOS apps using the latest technologies described in these lessons, you need a Mac computer (OS X 10.10 or later) running the latest version of Xcode. Xcode includes all the features you need to design, develop, and debug an app. Xcode also contains the iOS SDK, which extends Xcode to include the tools, compilers, and frameworks you need specifically for iOS development.

Download the latest version of Xcode on your Mac free from the App Store.

To download the latest version of Xcode

- Open the App Store app on your Mac (by default it's in the Dock).
- In the search field in the top-right corner, type Xcode and press the Return key.
- The Xcode app shows up as the first search result.
- Click Get and then click Install App.
- Enter your Apple ID and password when prompted.
- Xcode is downloaded into your /Applications directory.

You may use any other IDE other than Xcode, but you will be on your own if problems spring up.

3.2 Add your account to Xcode

When you add your Apple ID to the Xcode Accounts preferences, Xcode displays all the teams you belong to. Xcode also shows your role on the team and details about your signing identities and provisioning profiles that you'll create later in this document. If you don't belong to the Apple Developer Program, a personal team appears.

Here is detailed documentation:

https://developer.apple.com/library/ios/documentation/IDEs/Conceptual/AppStoreDistributionTutorial/AddingYourAccounttoXcode/AddingYourAccounttoXcode.html

3.3 Install CocoaPods

CocoaPods is a dependency manager for Swift and Objective-C Cocoa projects. It has over ten thousand libraries and can help you scale your projects elegantly. You can install dependencies using it, we will need to install many third party modules and frameworks using it.

CocoaPods is built with Ruby and is installable with the default Ruby available on OS X. We

recommend you use the default ruby. Using the default Ruby install can require you to use sudo when installing gems.

Run the command below in your Mac terminal:

```
$ sudo gem install cocoapods
```

Once you have created you Xcode project, you can start to integrate cocoapods into your project.

Further guides on how to integrate cocoapods are available at: https://cocoapods.org/

4. High Level Design

In this exercise, you will develop an iOS Mobile application, which will have following functionality:

There will be a slide out-menu which will provide access to the different screens such as Home, Favorites and About Me. The content displayed in each of these sections will be similar to Homework 8 but we will go into details later on.

The initial screen would default to showing the home screen as shown below.

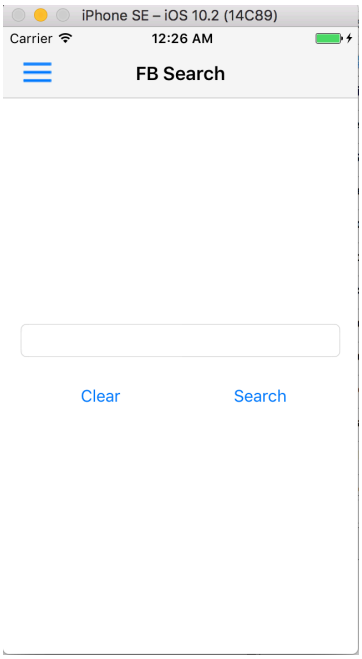


Figure 1. Initial Screen

The slide-out menu should open upon tapping the hamburger icon. Please refer the below screenshot for the menu design. The menu would close on tapping on the menu button.

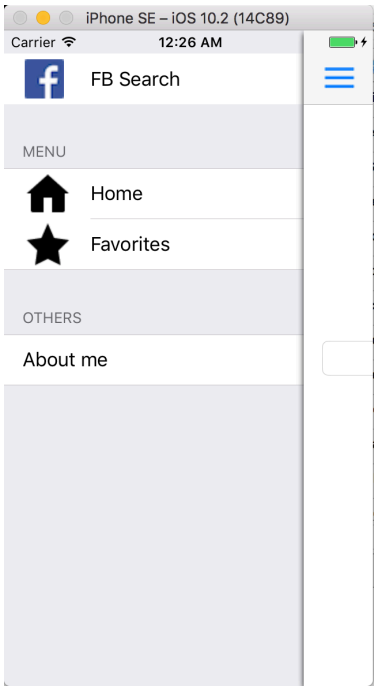


Figure 2. Slide-out Menu

5. Implementation

5.1 Home screen

You must replicate the Home Screen, as shown in Figure 1.

The interface consists of the following:

- A 'UITextField' to allow the user to enter the search query.
- 2 'UIButton' for clearing and submitting the query.
- A UI component to show the navigation bar that has the hamburger menu icon to show the slide-out menu.

The 'Search' button should submit the search query and move to the next screen showing the search results. Please note that it should display the message in case of an empty text field and avoid going to the next screen.

The 'Clear' button should clear the text field's content.

The hamburger menu icon should show the slide-out menu, which should be dismissed upon clicking the menu icon again.

5.2 Search Results

On submitting the query, the screen should move to display the results in 5 different tabs as shown below:

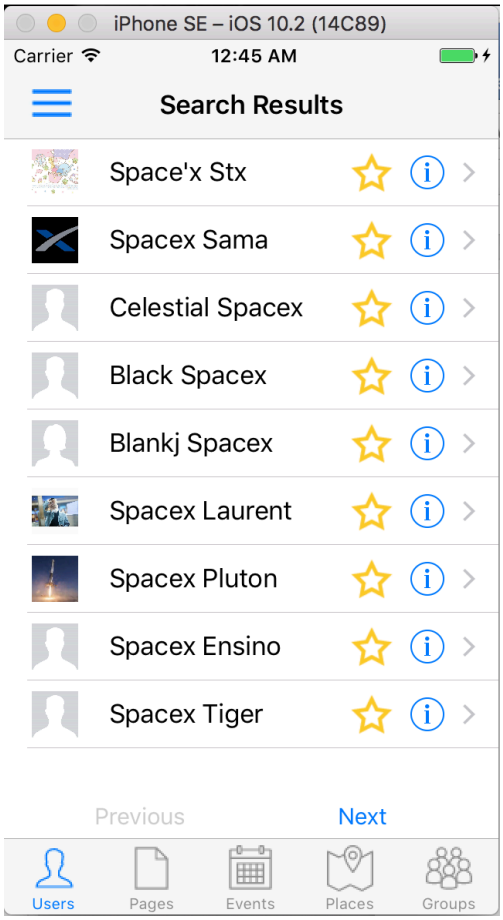


Figure 3: Search Results

The search results would display the results in 5 tabs for the following types, like in Homework 8:

- User
- Page
- Event
- Place
- Group

Please note that the search result would show the user tab, by default.

5.2.1 User Tab

The user tab would display the search result for the type – 'user'. The screen consists of the following:

1. 'UITableView' to show the users corresponding to the search query.
2. 2 'UIButton' for pagination
3. The navigation bar to display the hamburger icon to display the slide-out menu as in Figure 2.

Each row in the table would contain the following:

1. Icon – A 'UIImage' component for the icon of the user
2. Name – A 'UILabel' component for the name of the user
3. Favorite – 'UIButton' component indicating whether the user has been marked as favorite
4. Detail Disclosure – Use the UItableViewCell's detail disclosure to show the user's detail described in the user detail screen next.

Note: Please show a 'loading' screen, while you are fetching data from the AWS/GAE servers to avoid showing a blank screen to the user. Please show the 'loading screen' before any network activity and hide it after rendering the screen with the data. Please refer the video link for more details on the same.

Pagination

There are 2 buttons at the bottom of the screen to navigate among the search results. The 'Previous' button and 'Next' button should only be enabled, if applicable.

Within the screen, you need to display 10 rows to show the first 10 users. The user can then use the next and previous button to view more users.

5.2.2 Pages Tab

The page tab would display the search result for the type – 'page'. The screen consists of the following:

1. 'UITableView' to show the pages corresponding to the search query.
2. 2 'UIButton' for pagination
3. The navigation bar to display the hamburger icon to display the slide-out menu as in Figure 2.

Each row in the table would contain the following:

1. Icon – A 'UIImage' component for the icon of the page
2. Name – A 'UILabel' component for the name of the page
3. Favorite – 'UIButton' component indicating whether the page has been marked as favorite
4. Detail Disclosure – Use the UItableViewCell's detail disclosure to show the page's detail described in the page detail screen next.

Note: Please show a 'loading' screen, while you are fetching data from the AWS/GAE servers to avoid showing a blank screen to the user. Please show the 'loading screen' before any

network activity and hide it after rendering the screen with the data. Please refer the video link for more details on the same.

Pagination
There are 2 buttons at the bottom of the screen to navigate among the search results. The 'Previous' button and 'Next' button should only be enabled, if applicable.

Within the screen, you need to display 10 rows to show the first 10 pages. The user can then use the next and previous button to view more pages.

5.2.3 Events Tab

The event tab would display the search result for the type – 'event'. The screen consists of the following:

1. 'UITableView' to show the events corresponding to the search query.
2. 2 'UIButton' for pagination
3. The navigation bar to display the hamburger icon to display the slide-out menu as in Figure 2.

Each row in the table would contain the following:

1. Icon – A 'UIImage' component for the icon of the event
2. Name – A 'UILabel' component for the name of the event
3. Favorite – 'UIButton' component indicating whether the event has been marked as favorite
4. Detail Disclosure – Use the UItableViewCell's detail disclosure to show the event's detail described in the event detail screen next.

Note: Please show a 'loading' screen, while you are fetching data from the AWS/GAE servers to avoid showing a blank screen to the user. Please show the 'loading screen' before any network activity and hide it after rendering the screen with the data. Please refer the video link for more details on the same.

Pagination
There are 2 buttons at the bottom of the screen to navigate among the search results. The 'Previous' button and 'Next' button should only be enabled, if applicable.

Within the screen, you need to display 10 rows to show the first 10 events. The user can then use the next and previous button to view more events.

5.2.4 Place Tab

The place tab would display the search result for the type – 'place'. The screen consists of the following:

1. 'UITableView' to show the places corresponding to the search query.
2. 2 'UIButton' for pagination
3. The navigation bar to display the hamburger icon to display the slide-out menu as in Figure 2.

Each row in the table would contain the following:

1. Icon – A 'UIImage' component for the icon of the place
2. Name – A 'UILabel' component for the name of the place
3. Favorite – 'UIButton' component indicating whether the place has been marked as favorite
4. Detail Disclosure – Use the UITableViewCell's detail disclosure to show the place's detail described in the place detail screen next.

Note: Please show a 'loading' screen, while you are fetching data from the AWS/GAE servers to avoid showing a blank screen to the user. Please show the 'loading screen' before any network activity and hide it after rendering the screen with the data. Please refer the video link for more details on the same.

Pagination

There are 2 buttons at the bottom of the screen to navigate among the search results. The 'Previous' button and 'Next' button should only be enabled, if applicable.

Within the screen, you need to display 10 rows to show the first 10 places. The user can then use the next and previous button to view more places.

Please note that you get the user's current location to passed to the API for the parameter center.

5.2.5 Group Tab

The group tab would display the search result for the type – 'group. The screen consists of the following:

4. 'UITableView' to show the groups corresponding to the search query.
5. 2 'UIButton' for pagination
6. The navigation bar to display the hamburger icon to display the slide-out menu as in Figure 2.

Each row in the table would contain the following:

5. Icon – A 'UIImage' component for the icon of the group
6. Name – A 'UILabel' component for the name of the group
7. Favorite – 'UIButton' component indicating whether the group has been marked as favorite
8. Detail Disclosure – Use the UITableViewCell's detail disclosure to show the group's detail described in the group detail screen next.

Note: Please show a 'loading' screen, while you are fetching data from the AWS/GAE servers to avoid showing a blank screen to the user. Please show the 'loading screen' before any network activity and hide it after rendering the screen with the data. Please refer the video link for more details on the same.

Pagination

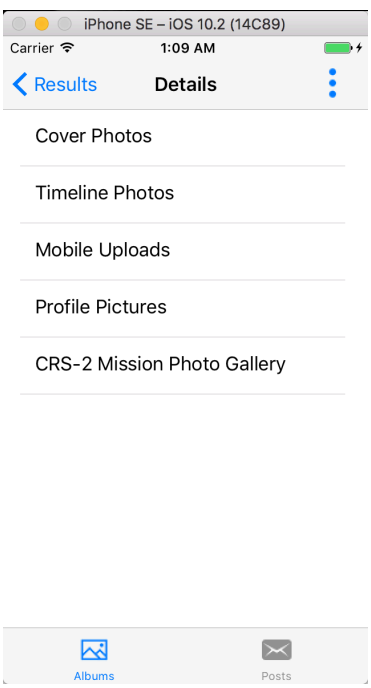
There are 2 buttons at the bottom of the screen to navigate among the search results. The 'Previous' button and 'Next' button should only be enabled, if applicable.

Within the screen, you need to display 10 rows to show the first 10 groups. The user can then use the next and previous button to view more groups.

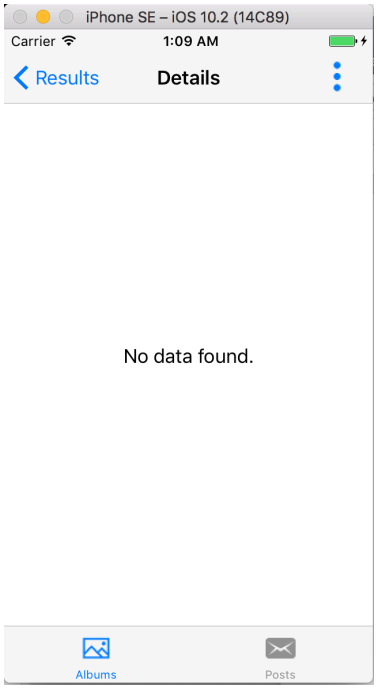
5.3.1 User Detail

Tapping any of the table row, should move to the user details to show the user's albums and posts, if any. The user detail screen contains two tabs for 'Albums' and 'Posts'. By default, the albums tab should be shown.

Please consider the following screenshot for reference.



Album data available

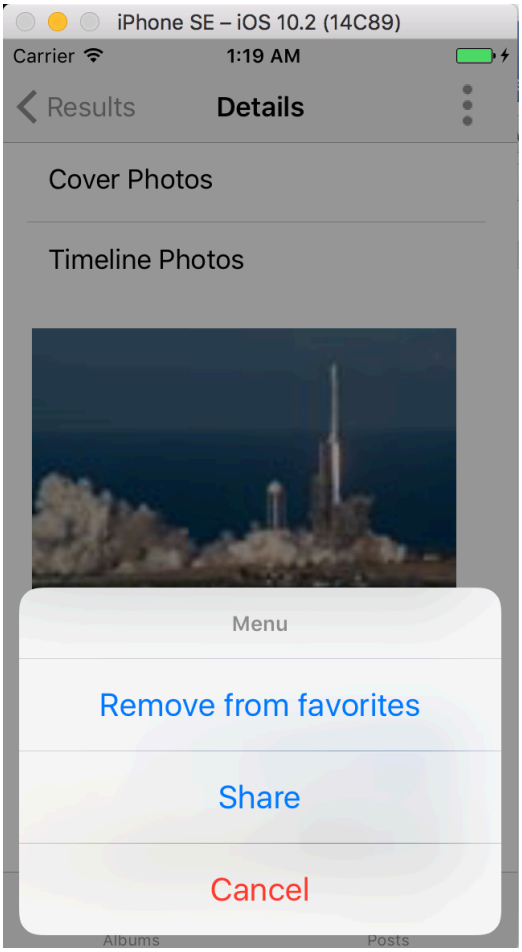


Album data unavailable

The albums tab should display the 5 albums, if available within the 'UITableView' component. Also note that the cell should be collapsed by default. However, if no album data is found, just show a 'UILabel' component showing an appropriate message.

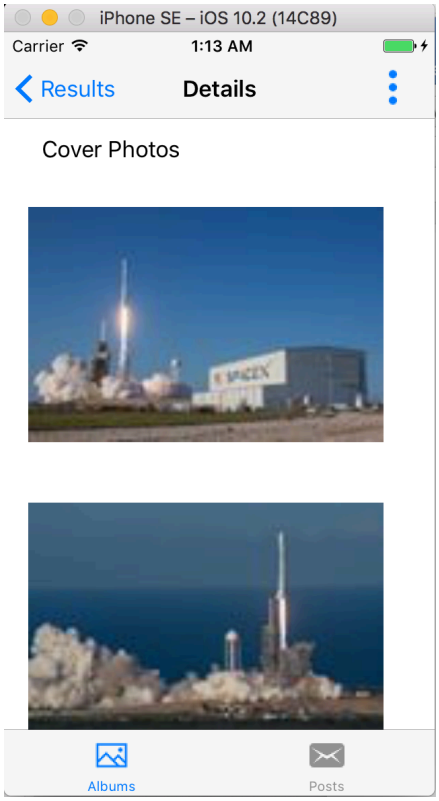
The user detail screen also contains the option menu option in the navigation bar. It would allow to mark the user as a favorite as well as share the user on Facebook. It would bring up the UIAlertController to show the two options – Favorite and Share, along with the Cancel option. Please refer the below screenshot for reference.

Please note that the text can be 'Add to favorites' or 'Remove from favorite' depending whether the user has already been added to favorite or not. Also note that marking the user as a favorite or removing the user from favorite should display an appropriate message.



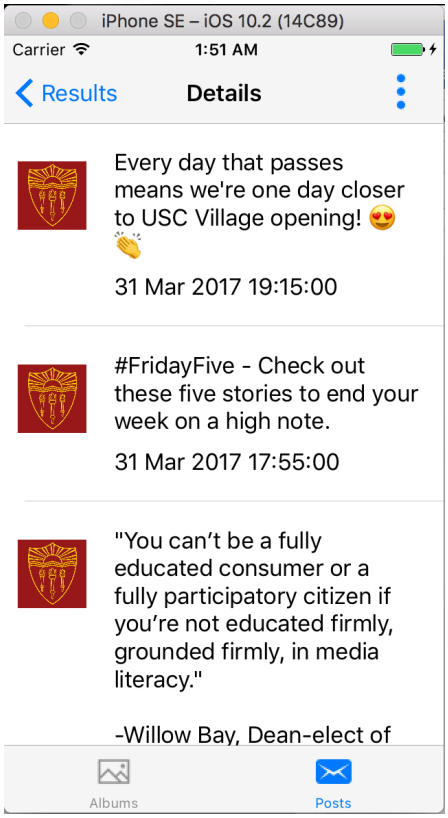
Thereon, the user can also share the user on Facebook. Please refer the Facebook Share section for reference.

On tapping an any of the 5 albums, the row should expand to show the 2 images within the concerned albums, as shown below. Furthermore, note that the upon tapping any other albums, the previously expanded album should collapse, if any. Also, upon tapping the expanded cell, the album should again collapse.



The posts tab should display the 5 posts, if available, within the 'UITableView' component.

Please note that each row of the table, should display the user's icon, the content of the post along with the formatted date.



Post data available

Post data unavailable

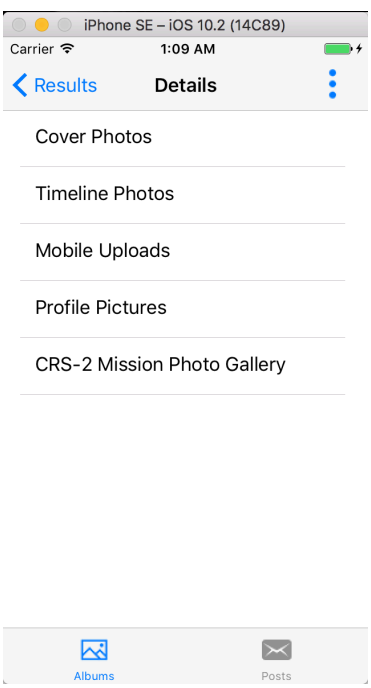
Please note that the back button should take you back to the search results screen on the 'User' tab.

Note: Please show a 'loading' screen, while you are fetching data from the AWS/GAE servers to avoid showing a blank screen to the user. Please show the 'loading screen' before any network activity and hide it after rendering the screen with the data. Please refer the video link for more details on the same.

5.3.2 Page Detail

Tapping any of the table row, should move to the page details to show the page's albums and posts, if any. The page detail screen contains two tabs for 'Albums' and 'Posts'. By default, the albums tab should be shown.

Please consider the following screenshot for reference.



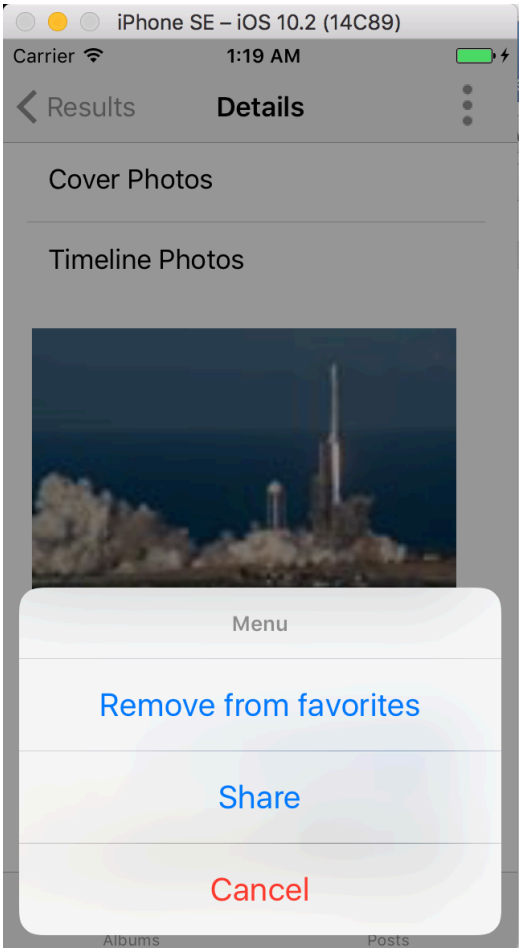
Album data available

Album data unavailable

The albums tab should display the 5 albums, if available within the 'UITableView' component. Also note that the cell should be collapsed by default. However, if no album data is found, just show a 'UILabel' component showing an appropriate message.

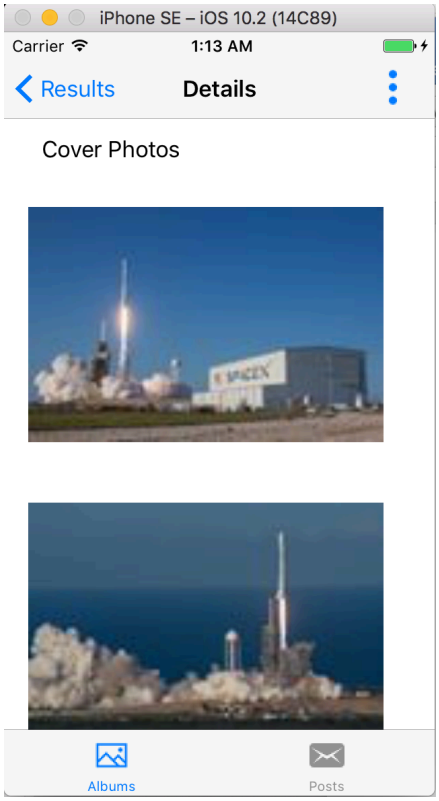
The page detail screen also contains the option menu option in the navigation bar. It would allow to mark the page as a favorite as well as share the page on Facebook. It would bring up the UIAlertController to show the two options – Favorite and Share, along with the Cancel option. Please refer the below screenshot for reference.

Please note that the text can be 'Add to favorites' or 'Remove from favorite' depending whether the page has already been added to favorite or not. Also note that marking the page as a favorite or removing the page from favorite should display an appropriate message.



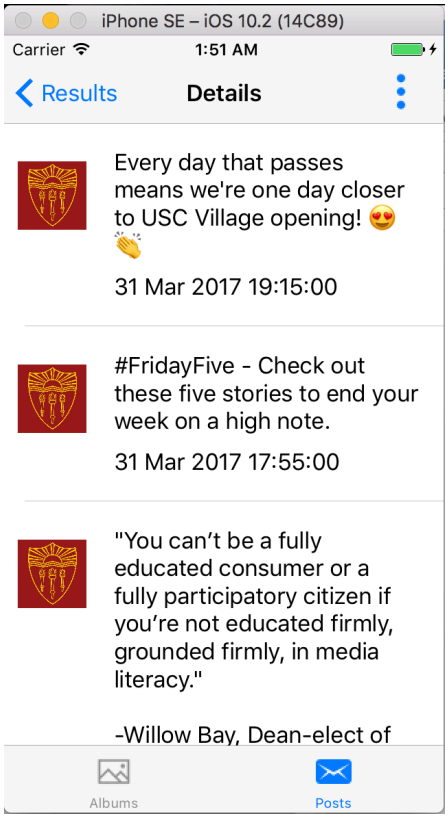
Thereon, the page can also share the page on Facebook. Please refer the Facebook Share section for reference.

On tapping an any of the 5 albums, the row should expand to show the 2 images within the concerned albums, as shown below. Furthermore, note that the upon tapping any other albums, the previously expanded album should collapse, if any. Also, upon tapping the expanded cell, the album should again collapse.



The posts tab should display the 5 posts, if available, within the 'UITableView' component.

Please note that each row of the table, should display the page's icon, the content of the post along with the formatted date.



Post data available

Post data unavailable

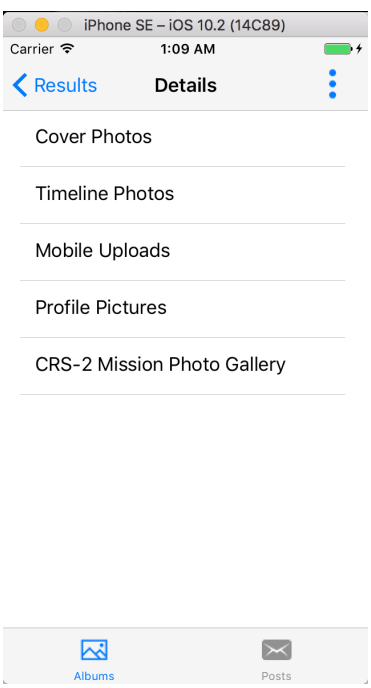
Please note that the back button should take you back to the search results screen on the 'Pages' tab.

Note: Please show a 'loading' screen, while you are fetching data from the AWS/GAE servers to avoid showing a blank screen to the user. Please show the 'loading screen' before any network activity and hide it after rendering the screen with the data. Please refer the video link for more details on the same.

5.3.3 Event Detail

Tapping any of the table row, should move to the event details to show the event's albums and posts, if any. The event detail screen contains two tabs for 'Albums' and 'Posts'. By default, the albums tab should be shown.

Please consider the following screenshot for reference.



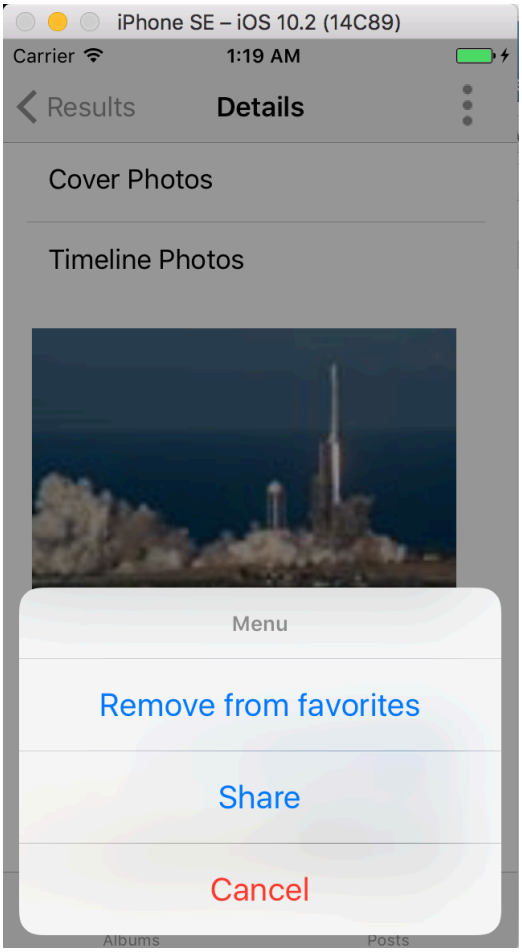
Album data available

Album data unavailable

The albums tab should display the 5 albums, if available within the 'UITableView' component. Also note that the cell should be collapsed by default. However, if no album data is found, just show a 'UILabel' component showing an appropriate message.

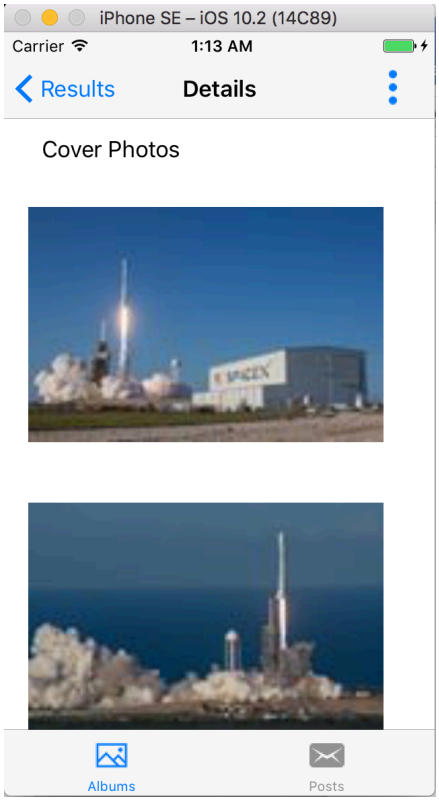
The event detail screen also contains the option menu option in the navigation bar. It would allow to mark the event as a favorite as well as share the event on Facebook. It would bring up the UIAlertController to show the two options – Favorite and Share, along with the Cancel option. Please refer the below screenshot for reference.

Please note that the text can be 'Add to favorites' or 'Remove from favorite' depending whether the event has already been added to favorite or not. Also note that marking the event as a favorite or removing the event from favorite should display an appropriate message.



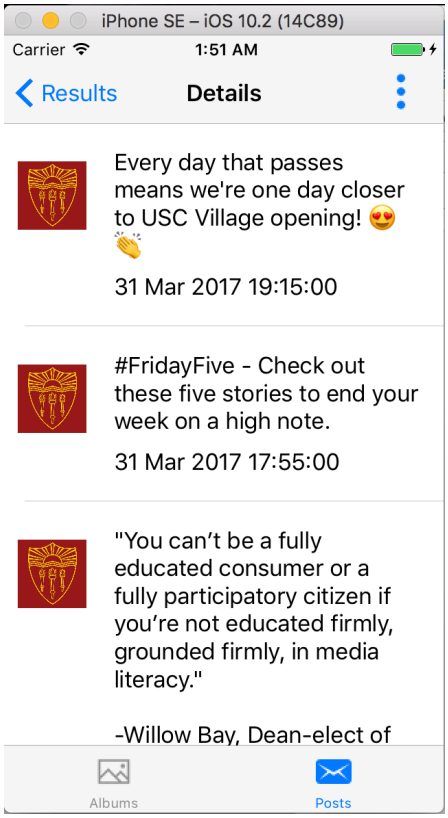
Thereon, the event can also share the event on Facebook. Please refer the Facebook Share section for reference.

On tapping an any of the 5 albums, the row should expand to show the 2 images within the concerned albums, as shown below. Furthermore, note that the upon tapping any other albums, the previously expanded album should collapse, if any. Also, upon tapping the expanded cell, the album should again collapse.



The posts tab should display the 5 posts, if available, within the 'UITableView' component.

Please note that each row of the table, should display the event's icon, the content of the post along with the formatted date.



Post data available

Post data unavailable

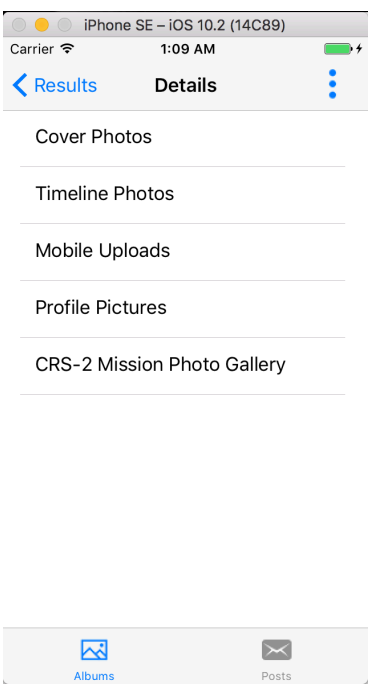
Please note that the back button should take you back to the search results screen on the 'Events' tab.

Note: Please show a 'loading' screen, while you are fetching data from the AWS/GAE servers to avoid showing a blank screen to the user. Please show the 'loading screen' before any network activity and hide it after rendering the screen with the data. Please refer the video link for more details on the same.

5.3.4 Page Detail

Tapping any of the table row, should move to the page details to show the page's albums and posts, if any. The page detail screen contains two tabs for 'Albums' and 'Posts'. By default, the albums tab should be shown.

Please consider the following screenshot for reference.



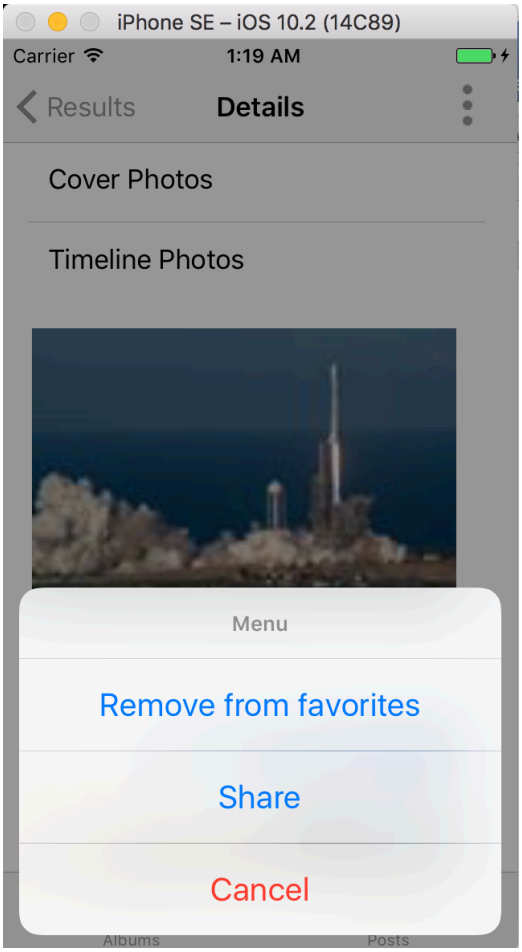
Album data available

Album data unavailable

The albums tab should display the 5 albums, if available within the 'UITableView' component. Also note that the cell should be collapsed by default. However, if no album data is found, just show a 'UILabel' component showing an appropriate message.

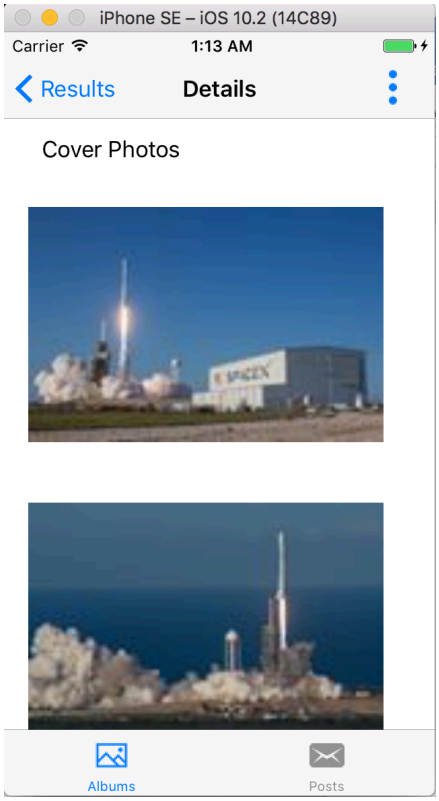
The page detail screen also contains the option menu option in the navigation bar. It would allow to mark the page as a favorite as well as share the page on Facebook. It would bring up the UIAlertController to show the two options – Favorite and Share, along with the Cancel option. Please refer the below screenshot for reference.

Please note that the text can be 'Add to favorites' or 'Remove from favorite' depending whether the page has already been added to favorite or not. Also note that marking the page as a favorite or removing the page from favorite should display an appropriate message.



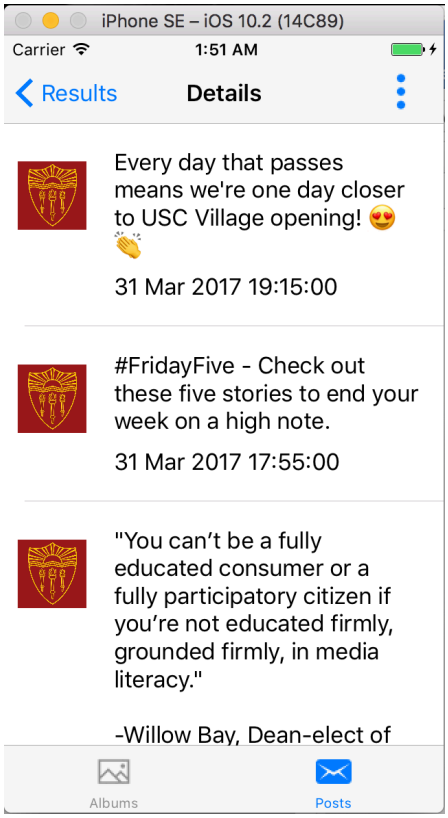
Thereon, the page can also share the page on Facebook. Please refer the Facebook Share section for reference.

On tapping an any of the 5 albums, the row should expand to show the 2 images within the concerned albums, as shown below. Furthermore, note that the upon tapping any other albums, the previously expanded album should collapse, if any. Also, upon tapping the expanded cell, the album should again collapse.



The posts tab should display the 5 posts, if available, within the 'UITableView' component.

Please note that each row of the table, should display the page's icon, the content of the post along with the formatted date.



Post data available

Post data unavailable

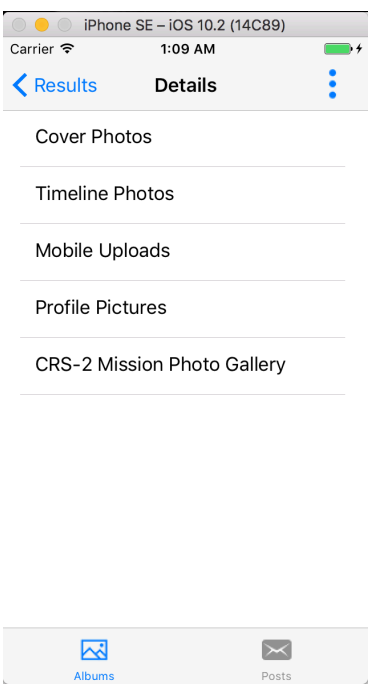
Please note that the back button should take you back to the search results screen on the 'Events' tab.

Note: Please show a 'loading' screen, while you are fetching data from the AWS/GAE servers to avoid showing a blank screen to the user. Please show the 'loading screen' before any network activity and hide it after rendering the screen with the data. Please refer the video link for more details on the same.

5.3.5 Group Detail

Tapping any of the table row, should move to the group details to show the group's albums and posts, if any. The group detail screen contains two tabs for 'Albums' and 'Posts'. By default, the albums tab should be shown.

Please consider the following screenshot for reference.



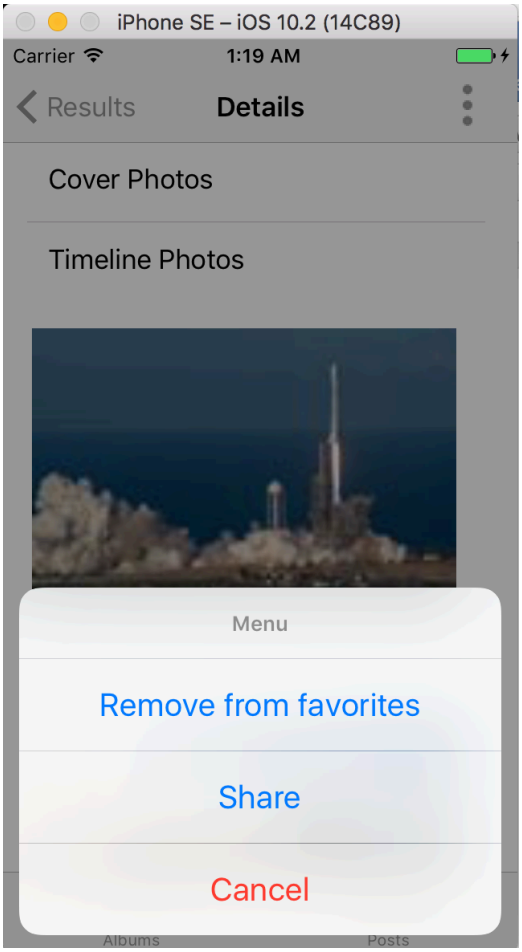
Album data available

Album data unavailable

The albums tab should display the 5 albums, if available within the 'UITableView' component. Also note that the cell should be collapsed by default. However, if no album data is found, just show a 'UILabel' component showing an appropriate message.

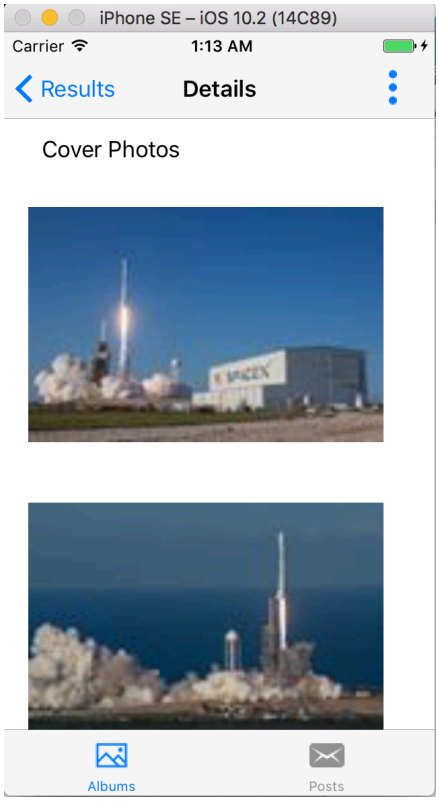
The group detail screen also contains the option menu option in the navigation bar. It would allow to mark the group as a favorite as well as share the group on Facebook. It would bring up the UIAlertController to show the two options – Favorite and Share, along with the Cancel option. Please refer the below screenshot for reference.

Please note that the text can be 'Add to favorites' or 'Remove from favorite' depending whether the group has already been added to favorite or not. Also note that marking the group as a favorite or removing the group from favorite should display an appropriate message.



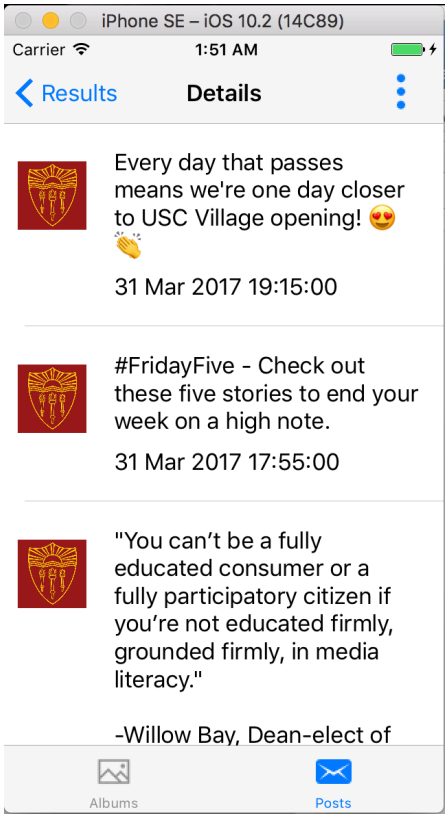
Thereon, the group can also share the group on Facebook. Please refer the Facebook Share section for reference.

On tapping an any of the 5 albums, the row should expand to show the 2 images within the concerned albums, as shown below. Furthermore, note that the upon tapping any other albums, the previously expanded album should collapse, if any. Also, upon tapping the expanded cell, the album should again collapse.



The posts tab should display the 5 posts, if available, within the 'UITableView' component.

Please note that each row of the table, should display the group's icon, the content of the post along with the formatted date.



Post data available

Post data unavailable

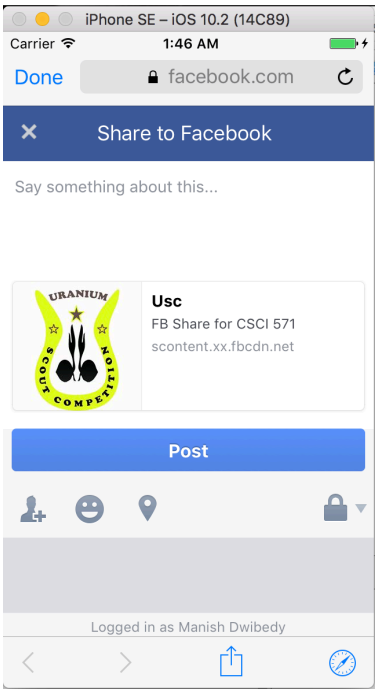
Please note that the back button should take you back to the search results screen on the 'Events' tab.

Note: Please show a 'loading' screen, while you are fetching data from the AWS/GAE servers to avoid showing a blank screen to the user. Please show the 'loading screen' before any network activity and hide it after rendering the screen with the data. Please refer the video link for more details on the same.

5.4 Facebook Share

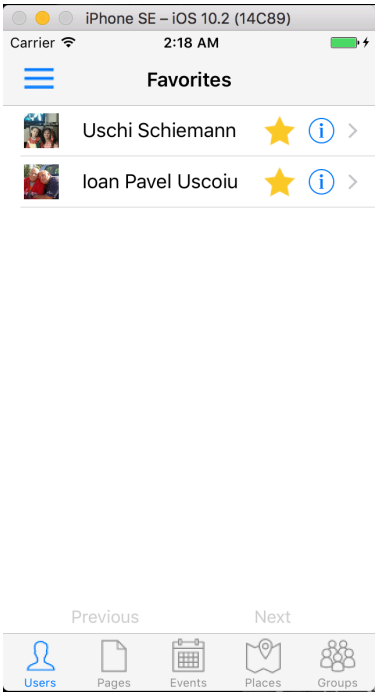
The user can share any of the searched results (user/page/event/place/group) on Facebook using the option menu displayed in the detail screen. Please refer the below screenshot for reference.

Please note the post needs to display the icon and name of the result that is being shared.

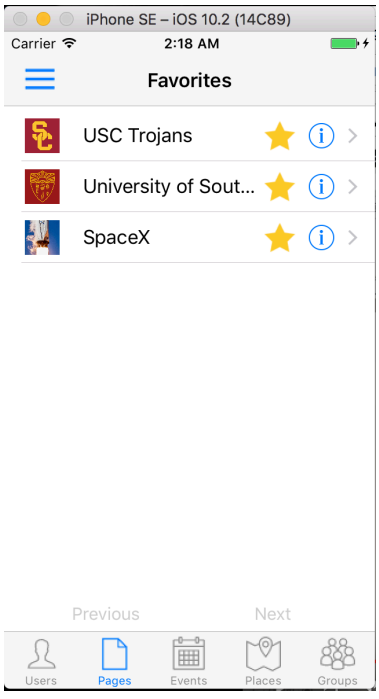


5.6 Favorite

The favorite screen would display all the favorited users/pages/events/places/group that have been marked as favorite. Please refer the below screenshot as reference.



Favorited users



Favorite pages

The design of the individual tabs remains exactly the same as the earlier search result's tab. The only difference being these display the favorited results instead of the search results. The order of the entry within a table would be the order the search results were marked as favorites.

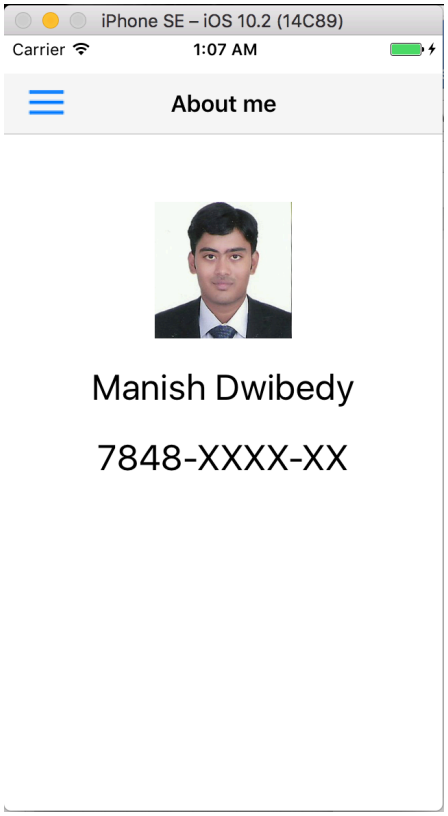
Also note that the selection of any row would also lead to the corresponding details screen. Please ensure that the table data is kept in sync with the favorites, especially when any of the result has been removed from the favorites.

Please note that the favorite section retains the pagination control like the search results.

5.7 About

You will also have to implement an additional menu item in the slide out menu that links to a screen which displays information about you.

The about page should look as below:



6. Resources

Please find the icons to be utilized in the app at:

http://cs-server.usc.edu:45678/hw/hw9/images/ios/album.png

http://cs-server.usc.edu:45678/hw/hw9/images/ios/empty.png

http://cs-server.usc.edu:45678/hw/hw9/images/ios/event.png

http://cs-server.usc.edu:45678/hw/hw9/images/ios/fb.png

http://cs-server.usc.edu:45678/hw/hw9/images/ios/filled.png

http://cs-server.usc.edu:45678/hw/hw9/images/ios/groups.png

http://cs-server.usc.edu:45678/hw/hw9/images/ios/home.png

http://cs-server.usc.edu:45678/hw/hw9/images/ios/menu.png

http://cs-server.usc.edu:45678/hw/hw9/images/ios/options.png

http://cs-server.usc.edu:45678/hw/hw9/images/ios/page.png

http://cs-server.usc.edu:45678/hw/hw9/images/ios/place.png

http://cs-server.usc.edu:45678/hw/hw9/images/ios/posts.png

http://cs-server.usc.edu:45678/hw/hw9/images/ios/users.png

http://cs-server.usc.edu:45678/hw/hw9/images/ios/favorite.png

7. Implementation Hints

See the HW9 IOS Clues file.

8. Material You Need to Submit

Unlike other exercises, you will have to "demo" your submission "in person" during a special grading session. Details and logistics for the demo will be provided in class, in the Announcement page and in Piazza.

You should also ZIP your project source directory and SUBMIT the resulting ZIP file. Make sure that the source path does not include the .app file or image files in the product folder.