**Mobile Computing – iOS Fall’23**

**Assignment05**

**30 Points**

**Please follow the following instructions to complete this assignment.**

1. Open Xcode from the launchpad of your Mac. Clone your private GitHub repository.
2. In that repo, create a new Xcode project. Select the iOS platform and click on the App template.
3. Click on next which will prompt you to choose options for the project.
4. Provide product name as **LastnameRecipeFinderApp**, “**edu.nwmissouri.fall23.cs44643**” for organization identifier, “**Storyboard**” as interface and “**Swift**” as the language.
5. Click on next and select an appropriate location to save your app and click on Create. A project directory will be loaded.
6. From the project navigator click on “Main.storyboard” file, a blank mobile screen will be loaded, where the required fields for an app need to be added.

**The View**

1. At first, remove the single view you see in the Main.storyboard file.
   1. Ctrl + Shift + L > Split View Controller
2. Drag and drop the split view controller to your Main.storyboard file. This will be the initial controller of your project.
3. Split View Controller has two controllers:
   1. Master View Controller: This view is a Table View Controller which is embedded in a Navigation Controller.
   2. Detail View Controller: A single view controller.
4. Embed the detail view controller in a navigation controller.
   1. Editor > Embed > Navigation Controller
5. Check Prefers Large Title property for the navigation bar (for both the controllers)
6. Create a Cocoa Touch Class “DashboardTVC” that is a sub class of UITableViewController and assign it as class to the Master View Controller that is present in the Main.storyboard file.
7. Add more view controllers to the storyboard and embed them in the Navigation Controllers as shown in Figure 1, respectively.

Table 1 The View Controllers

|  |  |
| --- | --- |
| **Screen** | **View Controller** |
| 1 | DashboardTVC |
| 2 | HomeVC |
| 3 | CategoriesTVC |
| 4 | MealsCVC |
| 5 | RecipeDetailsVC |
| 6 | PlayVideoVC |
| 7 | FavoritesTVC |

1. Refer to Table 1 to create required classes and assign them to proper view controllers in the storyboard.

6

5

4

3

A screenshot of a cell phone

Description automatically generated

Segue6

Segue5

Segue4

Segue2

1

7

2

Figure 1 The View

Segue1

Segue3

Table 2 Segue Details

|  |  |  |
| --- | --- | --- |
| **Segue** | **Name** | **Type** |
| Segue1 | homeSegue | Show Detail |
| Segue2 | categoriesSegue | Show Detail |
| Segue3 | favoritesSegue | Show Detail |
| Segue4 | showMeals | Show |
| Segue5 | showMealDetails | Show |
| Segue 6 | playVideo | Show |

**Note:** Segue between the Navigation Controllers and View Controllers is “Root View Controller”.

A screenshot of a phone

Description automatically generated

Figure 2 Auto Layout for DashboardTVC

A screen shot of a phone

Description automatically generated

Figure 3 Auto Layout for HomeVC

A screenshot of a phone

Description automatically generated

Figure 4 Auto Layout for CategoryTVC

Table 3 CategoryTVC’s properties for stack views

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Stack View** | **Axis** | **Alignment** | **Distribution** | **Spacing** |
| BigContainerSV | Horizontal | Fill | Fill | Standard |

A screenshot of a phone

Description automatically generated

Figure 5 Auto Layout for FavoritesTVC

Table 4 FavoritesTVC’s properties for stack views

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Stack View** | **Axis** | **Alignment** | **Distribution** | **Spacing** |
| BigContainerSV | Horizontal | Fill | Fill | Standard |

A screenshot of a phone

Description automatically generated

Figure 6 Auto Layout for MealsCVC

A screenshot of a phone

Description automatically generated

Figure 7 Auto Layout for PlayVideoVC

A screenshot of a phone

Description automatically generated

Figure 8 Auto Layout for RecipeDetailsVC

Table 5 RecipeDetailsVC’s properties for stack views

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Stack View** | **Axis** | **Alignment** | **Distribution** | **Spacing** |
| BigContainerSV, detailsSV, ArtistsSV | Vertical | Fill | Fill | Standard |
| CategorySV, AreaSV | Horizontal | Fill | Fill Equally | Standard |

1. Design the views as per the Figures 1, 2, 3, 4, 5, 6 and 7.

Table 6: UI elements configuration for RecipeDetailsVC

|  |  |  |
| --- | --- | --- |
| **UI element** | **Purpose** | **Outlet/action name** |
| 3 UI Labels | To display name of the meal | mealNameLBL |
| To display the area of the meal | areaLBL |
| To display the category of the meal | categoryLBL |
| 1 UI Text View | To display the instructions for preparing the meal | recipeTV |
| 1 UI BarButtonItem | To add the meal to favorites | addFavorite |
| 1 Image View | To display the image of meal | mealIV |

Table 7: UI elements configuration for PlayVideoVC

|  |  |  |
| --- | --- | --- |
| **UI element** | **Purpose** | **Outlet/action name** |
| 1 YouTube Player View | To play the video in YouTube | playerView |

*Note: Define appropriate methods wherever required to achieve the required functionality via @IBActions and @Objc functions.*

**The Model**

1. Download the Model.swift file provided to you in the assignment.
2. Study the file to understand the format of the data and other details.

**The Controller**

1. Add Lottie package to the Project using the GitHub link <https://github.com/airbnb/lottie-ios.git>. The animation used in HomeVC are provided in the assignment zip folder.
2. Add SDWebImage package to the Project using the GitHub link <https://github.com/SDWebImage/SDWebImageSwiftUI.git>. This package is used to display the images using the hyperlinks. All the images provided in the Model file are hyperlinks.
3. Add YouTube-IOS-Player-Helper package to the Project using the GitHub link <https://github.com/youtube/youtube-ios-player-helper.git>. This package is used to play the YouTube videos.
4. Add Alamofire package to the Project using the GitHub link <https://github.com/Alamofire/Alamofire.git>. This package is an HTTP networking library.
5. ***DashboardTVC***
   1. Sub class of UITableViewController.
   2. This is the Master View Controller of the Split View embedded in a Navigation Controller.
   3. Set the navigation title to “Delicious Bites”.
   4. This controller has 3 cells. Each cell has is associated with a label and an image. On tapping each cell, segue must be performed to the respective controllers.

Table 8: Table View Cells and their Segues in DashboardTVC

|  |  |  |
| --- | --- | --- |
| **Title** | **Image** | **Segue** |
| Home | homekit | Segue to HomeVC with identifier “homeSegue” |
| Categories | list.dash | Segue to CategoryTVC with identifier “categoriesSegue” |
| Favorites | heart | Segue to FavoritesTVC with identifier “favoritesSegue” |

1. ***HomeVC***
   1. This is the first Detail View Controller of the Split View, Sub class of UIViewController and embedded in a Navigation Controller.
   2. Set the navigation title as “Home”.
   3. Play the Lottie animation animation.json in a loop whenever the HomeVC is appeared.
2. ***CategoryTVC***
   1. This is the second Detail View Controller of the Split View.
   2. Sub class of UITableViewController, embedded in the Navigation Controller.
   3. Set the navigation title as “Categories”.
   4. Display the list of all categories in each cell of the table view.
   5. Each cell of the table view must contain a label “Name of the category” and an image “fork.knife.circle”.
   6. Height of each cell must be “60”.
   7. On tapping on any category, perform segue to “MealsCVC”.
3. ***MealsCVC***
   1. Sub class of UICollectionViewController.
   2. Set the navigation title as “Menu Items”.
   3. Display the images of the meals in the collection view. Use the same layout of the collection view provided in the output video.
   4. On tapping on any meal, perform segue to “RecipeDetailsVC”.
4. ***RecipeDetailsVC***
   1. Sub class of UIViewController.
   2. Set the navigation title as “Item Details”.
   3. Display the details of meals that is selected from the previous screen MealsCVC. (Details include name, category, area of origin, instructions to prepare the meal).
   4. A struct for favorites i.e., FavoriteMealDetails and to store the favorites locally favoriteMeals are already given in the Model.swift file. Use them to perform operations related to the favorites.
   5. On clicking on the “♡” bar button, the meal must be added to the favorites, and change the title of the button to “♥️” (filled heart). Play the sound 1152 while adding to favorites.
      1. On clicking the “♥️” (filled heart) must remove the meal from favorites, and change the title of the button to “♡”. Play the sound 1109 while removing from the favorites.
   6. Add a single tap gesture to the image “mealIV” and perform a segue to PlayVideoVC. (Make sure the image is user interactable).
5. ***PlayVideoVC***
   1. Sub class of UIViewController.
   2. Set the navigation title as “Preparation Video”.
   3. Extract the video ID from the “strYoutube” key.
      1. For example, <https://www.youtube.com/watch?v=4aZr5hZXP_s> is the URL, then the video ID will the set of characters after “v=” i.e., “4aZr5hZXP\_s”.
   4. Use the extracted video ID to play the video using YouTubeiOSPlayerHelper.
6. ***FavoritesTVC***
   1. This is the second Detail View Controller of the Split View.
   2. Sub class of UITableViewController, embedded in the Navigation Controller.
   3. Set the navigation title as “Favorites”.
   4. Display the meals added to the favorites based on their category. Each category of the meals must be a section in the table view.
   5. Height of each cell must be “60”.
   6. Add a leading swipe action for the cells in the table view, to remove the meal from the favorites.
      1. Set the title of the swipe action as “Delete”.

**Submission:** Push your entire Xcode project to your private GitHub repo and submit your private repository link on the Canvas via Text Entry.