**Exam 03**

**Points - 50**

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

1. Open Xcode from the launchpad of your Mac.
2. Create a new Xcode project. Select the iOS template and click on the App application. Provide product name as **Lastname\_****MovieDisplayApp**, “**nwmsu**” for organization identifier, “**storyboard**” as interface and swift as language.
3. Click on next and select an appropriate location to save your app and click on create. A project directory will be loaded.
4. Choose Project Format to “Xcode 13.0-compatible” in your Project Document settings and Deployment Info settings to “iOS 15.5”
5. From the project navigator click on “Main” file, a blank mobile screen will be loaded where the required fields for an app need to be added.
6. Rename the ViewController file to **MovieSectionViewController.**
7. Embed the **MovieSectionViewController** into a navigation controller.
8. For the data, please download the **Movie.swift** file from the Canvas assignment page and create a new Swift file with above name and copy paste all the provided data into created Swift file.
9. Download zip file of movie images provide in Canvas and paste them in Assets folder in XCode.
10. Insert a Table View in **MovieSectionViewController** from the library**.** Set the constraints to top = 0, bottom = 0, left = 0 and right = 0.
11. In the attribute inspector of Table View change the Prototype Cells to 1.
12. Name the Table View cell to **“movieSectionCell”** from Attribute Inspector.
13. Insert a title as “Movie Section” at the top of Table View in **MovieSectionViewController**.
14. Refer Figure 1 for design of **MovieSectionViewController.**

Graphical user interface, application

Description automatically generated with medium confidence

***Figure 1***

1. Now Create another view Controller and create a new Cocoa Touch Class and give the name as **MovieInfoViewController**. In the identity inspector of the view controller give the class name as **MovieInfoViewController**.
2. Now give a connection in the **MovieSectionViewController** for the table view and name the connection of your choice.
3. Create a segue from the table view cell of **MovieSectionViewController** to **MovieInfoViewController** and name the identifier of the segue as “**movieInfoSegue**”.
4. Add an image view and four labels in **MovieInfoViewController** from the library. Give the connection names accordingly for above image view and four labels in **MovieInfoViewController**.
5. Image View holds the image of movie, and four labels hold movie name, movie cast, movie release year and movie collection respectively.
6. Refer Figure 2 for design of **MovieInfoViewController.**

Graphical user interface, application

Description automatically generated

***Figure 2***

1. First view controller i.e., **MovieSectionViewController** contains all movie names from Movie.swift file as Table View. Populate movie names into Table View. Refer Figure 3 for reference.

Graphical user interface, application

Description automatically generated

***Figure 3***

1. Whenever you click on any movie name in table view of **MovieSectionViewController** you should be navigating to **MovieInfoViewController** where you will be populating the image view and labels with movie data. Write a prepare method to navigate between two view controllers and send the movie data of respective movie.
2. Populate the movie name as title for **MovieInfoViewController.**
3. Refer below images for output of **MovieInfoViewController.**

Graphical user interface, application

Description automatically generated Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated Graphical user interface, application

Description automatically generated

**Please submit your app as compressed file, your compressed files should contain Lastname\_ MovieDisplayApp folder and Lastname\_ MovieDisplayApp.xcodeproj file. Please check your submission by downloading the submitted file and rechecking in Xcode.**