Mobile Computing Class Project Final Report

CSCI 4500/6970 Mobile Computing Spring 2020

**IMDB IOS APP APPLICATION USING THE MVC ARCHITECTURE**

A PROJECT REPORT

Presented to the Department of Computer Science

Auburn University at Montgomery

In Partial Fulfillment

of the Requirements for the Degree

Master of Science in Computer Science

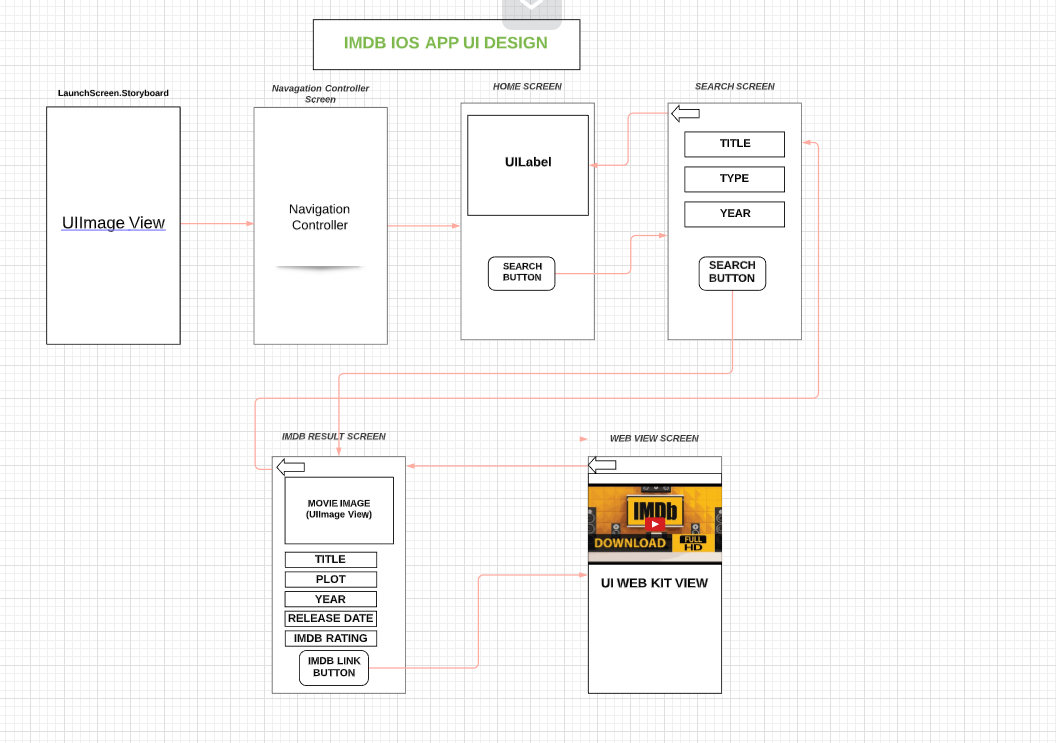
By

Ajay Saradhi Reddy Chilukuri

Jhansi Alugubelly

Krishna Chaitanya Kandimalla

# Introduce your UI Design and describe your UI elements (e.g. the functions of every buttons, etc.)



We have designed our IOS application by using 5 Scenes in Main.Storyboard and 1 Launch Screen Scene in LaunchScreen.Storyboard.

* The 5 Scenes in Main.Storyboard are as below

1. Navigation Controller Scene
2. IMDb Search Scene (Home Screen)
3. Search Scene (To enter the movie name, type and year)
4. Search Result Scene (This scene includes result of Search Scene like Movie Title, image, year and IMDB link)
5. Db Website View Controller Scene (It is browser scene which shows the IMDB page of the movie.)

* The LaunchScreen.Storyboard as one View Controller Scene which as UIImage View Element to show an IMDB image while launching the application.

**UI ELEMENTS:**

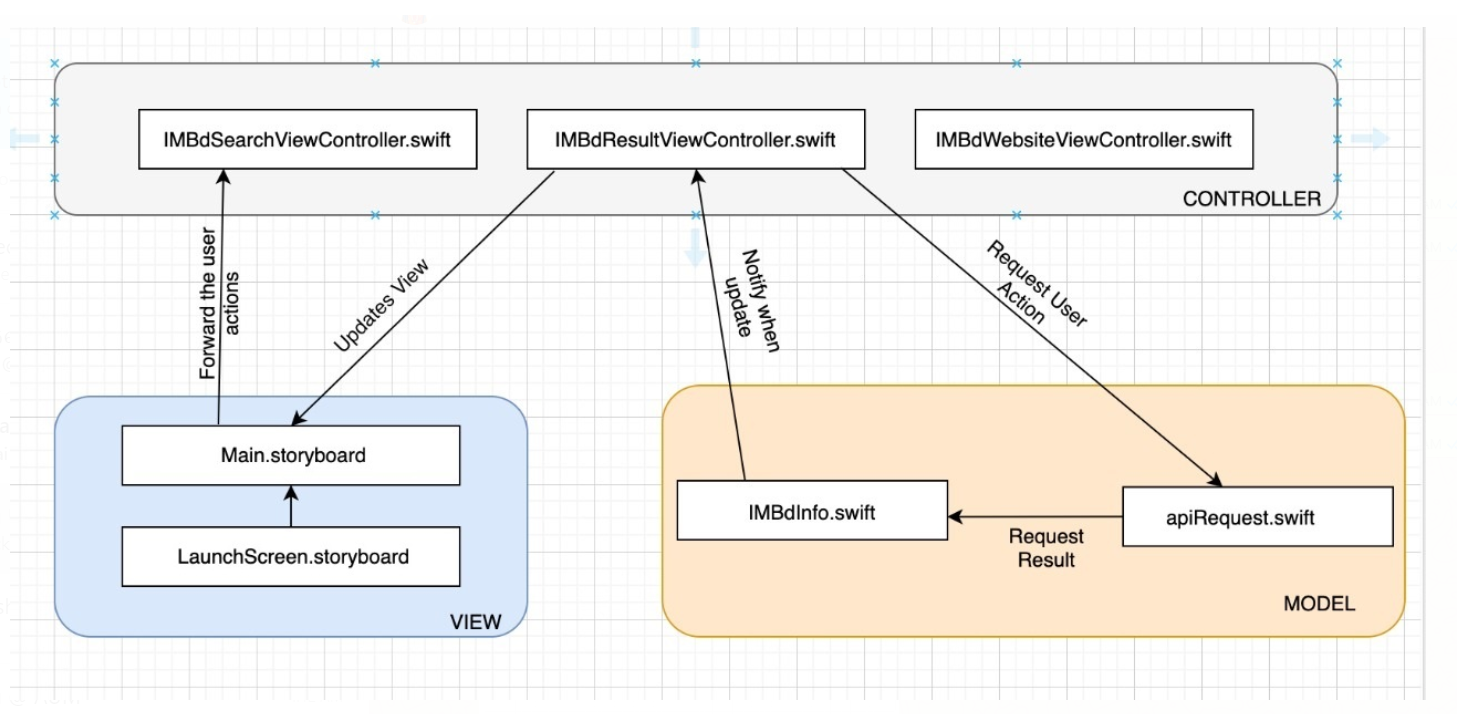
User interface (UI) elements are the parts we use to build apps or websites. They add interactivity to a user interface, providing touchpoints for the user as they navigate their way around; think buttons, scrollbars, menu items and checkboxes.

* We used below UI Elements in our Application
  1. **UILabel**: Used for displaying static content.
  2. **UIText Field:** UI element that enables the app to get user input.
  3. **UIView**: UIView provides a structure for drawing, laying out subviews, and handling events within a rectangular region of its parent in the view hierarchy.
  4. **UINavigationBar**: Contains the navigation buttons
  5. **UINavigationItem**: To be represented on the navigation bar, a navigation item must have a title.
  6. **UIButton**: Used for handling user actions.
  7. **UIImageView**: Used to display a simple image or sequence of images.
  8. **UIScrollView**: Used to display content that is more than the area of screen.
  9. **WKWebView**: Displays embedded web content and enables content navigation.
  10. **UIPickerView**: It can be used to make a selection from multiple choices (similar to what a dropdown does for a webpage).
  11. **Activity Indicator View**: Used to indicate processing for a task with unknown completion percentage.

# Introduce your design of mobile application, e.g., MVC models, UML diagrams, etc.

A screenshot of a cell phone

Description automatically generated



# List any technology that is not mentioned in the project description for getting bonus.

In our Project we used below extra features.

* + CocoaPods
  + UIPickerView
  + SDWebImage
  + IQKeyboardManagerSwift

**CocoaPods:** CocoaPods manages library dependencies for your XCode projects.

The dependencies for your projects are specified in a single text file called a Podfile. CocoaPods will resolve dependencies between libraries, fetch the resulting source code, then link it together in an XCode workspace to build your project.

**UIPickerView:** UIPickerView which is a UI element that can be used to select from multiple choices (like what a dropdown does for a webpage). A view that uses a spinning-wheel or slot-machine metaphor to show one or more sets of values.

**SDWebImage:** SDWebImage is a simple image library for iOS which provides caching and an easy way to load remote images asynchronously into UIImageViews.

This library provides a category for UIImageVIew with support for remote images coming from the web.

There are three ways to use SDWebImage in your project:

* using CocoaPods
* using Carthage
* by cloning the project into your repository

In our project we used it by CocoaPods by adding below content in the PODFILE

*use\_frameworks!*

*pod 'SDWebImage', '~> 5.0'*

**IQKeyboardManagerSwift:** Often while developing an app, we ran into an issues where the iPhone keyboard slides up and cover the *UITextField/UITextView*. IQKeyboardManager allows you to prevent issues of the keyboard sliding up and cover *UITextField/UITextView* without needing you to enter any code and no additional setup required. To use *IQKeyboardManager* you simply need to add source files to your project.

IQKeyboardManagerSwift is available through CocoaPods, to install it simply add the following line to your Podfile:

*pod 'IQKeyboardManagerSwift'*

In *AppDelegate.swift,* just import IQKeyboardManagerSwift framework and enable IQKeyboardManager.

# Please attach your codes (application) or a GitHub link to your project along with this report.

<https://github.com/chaitu1704/CSCI-6970-Sp.-Tp.-Mobile-Computing-Class-Project-Groups-5>

# Individual tasks

|  |  |
| --- | --- |
| **Student Name** | **Major tasks** |
| AJAY SARADHI REDDY CHILUKURI | Dividing the tasks to the team Members.  Done coding part i.e. MVC Model Design and Helping other team members in the coding part.  Been a part in the documentation for the final report.  Mostly Phase 3 task was taken. |
| JHANSI ALUGUBELLY | Whole UI Design (Phase 1) was taken care.  Prepared few slides in the Final Demo PPT.  Helped on the Phase 3 Coding part.  Been a part in the final report documentation for the UI Design. |
| KRISHNA CHAITANAYA KANDAMURI | Whole Phase 2 (Playground) code part was taken care.  Prepared few slides in the Final Demo PPT.  Helped on the Phase 3 Coding part.  Been a part in the final report documentation for the UML Diagram. |

# Input/Output (take screenshots from your apps)

* 1. **iOS Scene View 1:** **Home Screen**

Screenshot of input and output: Just and Home screen and when we clicked Search button it will take to the next Scene i.e. **Search Scene**

**Input Screen Output Screen**

A picture containing many, sitting, street, computer

Description automatically generatedA picture containing sitting, street

Description automatically generated

* 1. **iOS Scene View 2: Search Scene**

**Screenshot of input and output:** When we entered the movie title, type and year as input and click on search button it will take next Scene i.e. **Search Result Scene**

**Input Screen Output Screen**

A picture containing bus

Description automatically generatedA screenshot of a computer

Description automatically generated

* 1. **iOS Scene View 2:** **Search Result Scene**

**Screenshot of input and output:** In this screen we will see the content of the movie and other details like year, Release Date, Rating and IMDB button. When we clicked IMDB button it will take to main imdb link for the particular movie which is web content view.

**Input Screen Output Screen**

A screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generated

‘

# Screenshot of output from other additional functions that you think can get bonus points.

**IQKeyboardManagerSwift:**

A screenshot of a computer

Description automatically generated

**UIPickerView**

A screenshot of a computer

Description automatically generated