Info 6350: Smartphone programming

short line

# What will you get out of this class and Class Syllabus:

1. **Week 1** : Getting started with iOS and Swift
2. **Week 2:** Create a simple app and walk through the app building process. How to deploy app on iOS Device

**Homework**: Create a simple app and run it on a simulator

1. **Week 3**: Create a Dice Game using Random numbers and introduction to little bit of swift.

**Homework**: Create a card game

1. **Week 4**: Intro to playground and Swift programming language.

**Homework**: Programming assignment on Swift programming language

1. **Week 5**: Introduction to API integration and create an app which uses API (Weather API).

**Homework** : Make app using some other public API using similar principles.

1. **Week 6**: MVC design Pattern and Create an app using MVC format

**Homework**: Create a MVC app

1. **Week 7**: Auto Layout and making sure apps work correctly in all the devices.

**Homework** : Skeleton App using Autolayout which will be used in other apps .

1. **Week 8:** Cocoa pods REST API's + Swift Intermediate level.

**Homework**: Swift programming assignment

1. **Mid Term**: App using REST API and Auto layout
2. **Week 9**: More of REST API, Swift and cocoa pods , create an app which will track Stock market.
3. **Week 10**: Cloud Database, REST API, advanced swift.

**Homework**: Create an App like Instagram or WhatsApp

1. **Week 11**: GitHub, and version control In App purchases and apple store kit, Advances Swift and Intro to Machine learning.

**Homework**: Create a Git repo and add apps in their GitHub repo.

1. **Week 12**: Create ML app which does image recognition.

**Homework**: Start working on Final Project

1. **Week 13**: Introduction to ARKit Augmented reality app and create an ARKit app.

**Homework**: Start working on Final Project

1. **Week 14**: Create an app which uses REST API, Machine learning, Login, Databases, and ARKit.

**Homework**: Start working on Final Project

1. **Week 15**: Final, Publish App on App store. Implementation

## How does an App work:

There are three main components of a smartphone app:

1. Screen
2. Code
3. Data

These three things roughly account for (M)odel, (V)iew, and (C)ontroller or MVC.

Example of when a user clicks on a button on the screen then following things happen:

* Sensor detects the tap and sends message to Operating system.
* OS determines where the pressure was applied, how much etc. and lot of other things.
* OS figures out the information of which app was pressed and where was the button pressed and passes this info to the app.
* The app has code which determines what to do in case a button was pressed.
* Let’s say when a button was pressed we had to show weather data on the screen, so in this case the app will get data from some database and send the appropriate data to the code.
* Code will then let the screen know what to change and display the data on the screen.

OS also acts as a manager which figures out which app to start, which app to turn off and how to save the data in case a phone is coming etc. We need to plan accordingly while we are writing our app.

# App Lifecycle:

1. Why?
2. Idea
3. Designing and wireframing
4. Development of the app
5. Testing
6. Publishing
7. Marketing
8. Update

Hackintosh, VMWare are other solutions but you cannot deploy app on device.

# Universal Development:

* Xamrin (C#)
* Flutter
* Phone Gap

# XCode Walkthrough:

1. Single View App
2. Options for new Project
3. Project Location

# Panes:

* Status Pane
* Navigation Pane
* Interface Builder Pane
* Utilities Pane

# Simulator:

# Hello World App:

# My Name is… App: