

VERSION

2

**TESBO COURSE**

ELYSIUM  
ACADEMY  
TESBO  
MOBILE APP  
DEVELOPER  
**ELYSIUM  
ACADEMY  
TESBO  
MOBILE APP  
DEVELOPER**

**SR. CODE**

EAPL/TESBO/TSTC06

**COURSE CODE**

EATMA

**SUB CATEGORY**

MOBILE APP DEVELOPMENT

 TOTAL DURATION <b>180</b> HOURS	 THEORY TAKEN <b>135</b> HOURS	 PRACTICAL TAKEN <b>45</b> HOURS
--	--	--

ELYSIUM  
ACADEMY  
TESBO  
MOBILE APP  
DEVELOPER

## COURSE DESCRIPTION



This course begins with the most fundamental matters, introduces you to the iOS platform and the Swift programming language, and then gradually begins to deal with more advanced topics such as cloud servers, concurrency, device hardware, networking, or debugging.

## COURSE GOALS



One of the biggest advantages to getting started with iOS development is the ease of learning Swift, the official programming language for iOS apps. It was introduced by Apple in 2014 to replace Objective-C, which had been the standard up until then, and was quickly adopted by most iOS developers.

## FUTURE SCOPE



Mobile app development has a bright future and a lot of possibilities. Developers now have the resources necessary to produce really new and transformational mobile apps that have the potential to alter the way we live and work thanks to the rise of AI and ML, cross-platform development, progressive web apps, the Internet of Things, and 5G. iOS is the most feature-rich mobile operating system after Android. With the popularity of iPhones constantly growing, its market share will reach 27.73% in 2022.

# 01

## CHAPTER

## CORE JAVA

### O1. Java Basics

- a. Java Introduction
- b. Features of Java
- c. Keywords, Literals, Comments
- d. Data Types
- e. Operators

### O2. Object Oriented Programming Language

- a. Introduction to object Oriented Programming Language
- b. Creating the object using Constructor
- c. this Keyword
- d. Methods
- e. Scanner class
- f. Constructor
- g. Overloading Method
- h. Method overriding
- i. Final keyword
- j. Super keyword
- k. Runtime Polymorphism
- l. Abstract classes and methods
- m. Interface
- n. Packages
- o. access modifications

### O3. Arrays

- a. Single Dimensional Array
- b. Multi-dimensional Array

  
**2.5**  
HRS

  
**6.5**  
HRS

## O4. Strings

- a. Immutable String
- b. Concatenation
- c. Numbers and String
- d. Special Characters

## O5. Exception Handling

- a. What are Exception?
- b. Types of Exception
- c. Try catch-Block
- d. Multiple catch Block
- e. Nested try
- f. Finally Block
- g. Throw keyword

# 02

CHAPTER

## INTRODUCTION TO ANDROID AND RECYCLER VIEW

### O1. Introduction to Android

- a. What is Android?
- b. Setting up development environment
- c. Android Architecture
- d. Android Component
- e. Dalvik virtual machine & .apk file extension
- f. First Android App

### O2. Basic Building Block

- a. Activities, Services, Broadcast Receivers & content provider



1.5  
HRS



4.5  
HRS

- b. UI Components – views & notification
- c. Intent and Bundle passing

### **O3. Application structure**

- a. AndroidManifest.xml
- b. uses-permission & uses –SDK
- c. Resources & R.java
- d. Assets
- e. Layout & Drawable Resources
- f. Activities and activity lifecycle

# 03

## CHAPTER

## **EMULATOR ANDROID VIRTUAL DEVICE & BASIC UI DESIGN**

### **O1. Emulator Android Virtual device**

- a. Launching Emulator
- b. Editing Emulator Setting
- c. Emulator shortcuts
- d. Adding New Emulator
- e. Logcat usage
- f. Introduction to DDMS
- g. Android Device Monitor

### **O2. Basic UI Design**

- a. Form widget
- b. Text Fields
- c. UI screen components
  - Main Action bar
  - View Control



- Content Area
- Split Action Bar
- d. Types of Layout
  - Linear Layout
  - Absolute Layout
  - Table Layout
  - Frame Layout
  - Relative Layout
- e. Unit of Measurements - [dip, dp, sip, sp] versus px

# 04

CHAPTER

## PREFERENCES, MENUS, INTENTS AND ACTIVITY

### O1. Preferences

- a. Shared Preferences
- b. preferences from xml
- c. APIs – accessing preference
- d. Methods in shared preference

### O2. Menu

- a. Option menu and app bar
- b. Context menu and contextual action mode
- c. Popup menu
- d. Sub menu
- e. Menu from xml
- f. Menu via code

### O3. Intents

- a. Types of intent
- b. Explicit intents
- c. Intent Filter



1.5  
HRS



5.5  
HRS

d. Implicits intents

## O4. Basic UI Design

- a. Activity Life Cycle
- b. Configuring the Manifest
- c. Managing the Activity Life cycle
- d. Start Activity for Result
- e. Share App data

# 05

CHAPTER

## STYLES & THEMES, CONTENT PROVIDERS

### O1. Styles and Themes

- a. Styles.xml
- b. Structure of a style
- c. Style hierarchy
- d. Drawable resources for shapes, gradients (selectors)
- e. Style attribute in layout file
- f. Applying themes via code and manifest file

### O2. Content Provider

- a. SQLite Programming
- b. SQLiteOpenHelper
- c. SQLite Database
- d. Working of content provider
- e. Method of content provider
- d. Cursor
- e. Reading and Updating contacts
- f. Reading bookmarks



# 06

## CHAPTER

### **LINKIFY , ADAPTER & WIDGETS**

#### **O1. Linkify**

- a. Web URLs, Email address, text, Map Address
- b. Phone numbers
- c. Match filter & Transform filter

#### **O2. Adapter and Widget**

- a. Adapters
  - ArrayAdapter
  - Baseadapter
- b. Adapter views
- c. Types of Adapter
- d. ListView and Listactivity
- e. Custom listview
- f. Expandable listview
- g. GridView using adapter
- h. Gallery using Adapter

  
**01**  
HRS

  
**06**  
HRS

# 07

## CHAPTER

### **NOTIFICATIONS , CUSTOM COMPONENTS & MULTITHREADING**

#### **O1. Notifications**

- a. Creating and sending notification
- b. pending intent notification
- c. Notification with action button
- d. Broadcast Receiver
- e. Services and Notification

  
**01**  
HRS

  
**05**  
HRS



- f. Performance and Memory Management
- g. Android Notification and Alarms

## **O2. Custom Components**

- a. Custom Tabs
- b. Custom Animated popup panels

## **O3. MultiThreads**

- a. Thread
- b. Running on UI thread
- c. Handler & Runnable
- d. Multithreading  
(My splashscreen, stopwatch, ThreadHandler)
- e. Examples

# 08

## CHAPTER

## **ADVANCED ANDROID FEATURES & SERVICES**

### **O1. Advanced Android Features**

- a. Live Folders
- b. Using cards
- c. XML and JSON Parsing
- d. Enable device networking info reporting
- e. Manually updating user location
- f. Accessing Phone services (Call, SMS, MMS)
- g. Network connectivity services and Set offline
- h. Referral Tracking

### **O2. Services**

- a. Android Service Overview
- b. How to implement a service
- c. Life Cycle of Services



- d. Types of Services
- e. Communication : Inter - Process

# 09

## CHAPTER

### **MULTIMEDIA , LOCATION-BASED SERVICES & GOOGLE MAP**

#### **O1. Multimedia**

- a. Audio File Supported by Multimedia
- b. Playback Media-Simple
- c. Video Formats
- d. Mobile Agent and Peer to Peer Architecture

#### **O2. Local – Based Services & Google Map**

- a. Geo Coding and Reverse Geo Coding
- b. Intent Services
- c. To Get Location in an Android Apps
- d. To Track user's Location (Latitude, Longitude and Address)
- e. Google API Console
- f. To create Google Map API Key
- g. Google places API Key

  
**01**  
HRS

  
**4.5**  
HRS

# 10

## CHAPTER

### **SENSORS, WIFI & TELEPHONIC SERVICE**

#### **O1. Sensors**

- a. How Sensors work
- b. Using Orientation & Accelerometer sensors
- c. Motion Sensor
- d. Environmental Sensor
- e. Position Sensor

  
**01**  
HRS

  
**5.5**  
HRS

## **O2. Wifi**

- a. Monitoring and managing Internet Connectivity
- b. Managing active connection
- c. Managing WiFi networks

## **O3. Telephony Service**

- a. Making calls
- b. Monitoring data connectivity and activity
- c. Accessing phone properties and status
- d. Controlling the phone
- e. Sending messages
- f. Wireless Connectivity and Mobile Apps

# 11

**CHAPTER**

## **GRAPHICS , CAMERA & BLUETOOTH**

### **O1. Graphics**

- a. 2D graphics
- b. 3D graphics
- c. Performance and Multithreading
- d. Graphics and UI Performance

### **O2. Camera**

- a. Taking Pictures
- b. Media Recorder
- c. Using Existing Android Camera Application
- d. Directly using Camera Application

### **O3. Bluetooth**

- a. Controlling local Bluetooth device
- b. Discovering and bonding with Bluetooth device



- c. Scan for Other Bluetooth Device
- d. Connect to other devices through service discovery

# 12

## CHAPTER

## TESTING & DEBUGGING ANDROID APPLICATION

### O1. Fundamental of testing in Android App

- a. Fundamental of Testing
- b. Testing Tools in Android
- c. Types of test in Android
- d. Security with HTTPs and SSL
- e. Dependency Injection

### O2. AndroidX Test Libraries

- a. Set up project for AndroidX Test
- b. Unit4 rules with AndroidX Test
- c. Android JUnit Runner
- e. Dalvik Debug Monitor server – It's Role and use(DDMs)
- f. Android Application and Debugging
- g. The use of Filters, Breakpoints, Suspend and Resume

### O3. Espresso

- a. To Setup Espresso for functional Testing
- b. Espresso cheat seat
- c. To Create custome matcher in Espresso
- e. Testing Recycler view with Espresso
- f. Espresso Resource Idling
- g. Multiprocess Espresso
- h. Espresso Web



# 13

## CHAPTER

## **BASICS OF ANDROID SECURE CODING**

### **O1. Introduction**

- a. Securing tips
- b. Storing data

### **O2. Using Networking**

- a. Security with Network Protocols
- b. Types of Connectivity
- c. Network security configuration
- d. Networking Libraries (volley, OKHttp, etc.,)
- c. Update your security provider to protect against SSL exploits
- d. SafetyNet safe browsing API
- e. Media Loaders (Picasso, glide ,etc.,)

### **O3. Android permissions**

- a. Workflow for using permission
- b. Types of permission
- c. Broadcast receiver

### **O4. Protect data at Rest**

- a. Work with data more securely
- b. Cryptography
- c. Android keystore system

### **O5. Security**

- a. APK Attack Surface
- b. Application Obfuscation
- c. Mobile Application Security Scanner



**02**  
HRS



**6.5**  
HRS

# 14

## CHAPTER

## GETTING STARTED

### O1. About Tools

- a. XCode
- b. iOS
- c. Swift
- d. Installation setup

### O2. About Swift

- a. Comments
- b. The println() function
- c. Variables
- d. Constants
- e. Data types
- f. Optional
- g. Type inference
- h. Hands on

### O3. Control Flow

- a. For
- b. If else if
- c. Do While
- d. While
- e. Switch
- f. Break and continue statement
- g. Hands on control flow

  
**04**  
HRS

  
**05**  
HRS

## O4. Operators

- a. Comparison operators
- b. Arithmetic operators
- c. Logical operators
- d. Hands on operators

## O5. Strings

- a. Literals
- b. Mutable strings
- c. Comparing strings
- d. Concatenating strings
- e. Hands on strings

# 15

CHAPTER

## **FUNCTIONS, OOPS**

### O1. Functions

- a. What is Swift Functions?
- b. Swift Nested Function
- c. Parameter & Return Value
- d. Swift Recursion
- e. Function Overloading
- f. Hands on functions

### O2. Swift collections

- a. Swift Arrays
- b. Swift Dictionary
- c. Swift Sets
- d. Swift Closures



**02**  
HRS



**08**  
HRS

- e. Swift Typealias
- f. Hands on collections

### **O3. Loops**

- a. For loops
- b. While loops
- c. For in loops
- d. Iterating over arrays
- e. Hands on loops

### **O4. OOPS**

- a. Swift Enum
- b. Structures
- c. Classes and objects
- d. Inheritance
- e. Methods
- f. Class Methods
- g. Type check
- h. Any object
- i. Protocols
- j. Hands on Swift OOPS

### **O5. Closures, Protocol & Delegates**

- a. Defining a closure
- b. Closures with parameters
- c. Closures with returned values
- d. Closures as callbacks
- e. Defining & implementing protocol
- f. Delegate design pattern



- g. Implementing & using delegates
- h. Hands on closures, protocol, delegates

# 16

## CHAPTER

### **ADVANCED OOPS**

#### **O1. ENUM and Type Casting**

- a. Creating & using enumerations
- b. Type checking
- c. Type casting
- d. Down casting
- e. Hands on enum and type casting

#### **O2. Tuples, ARC**

- a. Creating & using tuples
- b. Creating & using type aliases
- c. Strong & weak references
- d. Avoiding strong reference cycles



**02**  
HRS



**05**  
HRS

## O4. Memory Management

- a. Reference Counting Basics
- b. Automatic Reference Count
- c. Retain Cycles

## O5. Error Handling

- a. Create enum of Errors
- b. Create a Throwing Function
- c. Function Call Using try Keyword
- d. Handling Errors Using do-catch Statement
- e. Hands on

# 17

CHAPTER

## EXTENSION, ACCESS CONTROL

### O1. Swift Generic

- a. Swift Generic Function
- b. Swift Generic Class
- c. Type Constraints in Swift Generics
- d. Hands on

### O2. Extension

- a. What is extension in swift?
- b. Computed Property In Extension
- c. Protocol Extension
- d. Hands on



### **O3. Swift Access Control**

- a. What is Swift Access Control?
- b. Types of Swift Access Controls
- c. Public Access Control
- d. Private Access Control
- e. file private Access Control
- f. Internal Access Control

### **O4. Type alias and Hashable**

- a. What is Swift Typealias?
- b. How to create a typealias?
- c. Typealias for built-in types
- d. Typealias for user defined types
- e. Typealias for complex types
- f. Swift hashable
- g. Hands on

### **O5. Swift Equatable**

- a. What is equatable?
- b. Strong Reference in Swift
- c. Swift Weak Reference
- d. Hands on

# 18

## CHAPTER

### **XCODE**

#### **O1. XCode and Interface builder**

- a. Purpose of XCode
- b. IOS Simulator
- c. Interface builder
- d. IOS application architecture
- e. Installation setup

#### **O2. COCOA Design Patterns**

- a. What is MVC?
- b. Model, View and Controller Classes
- c. Delegate and Data source
- d. Singleton Pattern
- e. Observer pattern
- f. Target-Action
- g. Cocoa coding standards

#### **O3. Controls in IOS**

- a. What is controls?
- b. Views
- c. Views Hierarchy

#### **O4. Common Interactions with UI**

- a. Button, label, Text fields
- b. switch, slider, progress bar
- c. Alerts, Action sheet
- d. Table views
- e. Scroll view, Web view, maps



**03**  
HRS



**07**  
HRS

- f. Search Bar, popovers
- g. Picker, Date picker, Image view, Image picker controller
- h. Gestures, Mouse events
- i. Mail, Message, Phone call
- j. Creating outlets and actions
- k. Hands on Basic UI Interactions

## O5. Auto Layout

- a. Stacks
- b. Nested Stacks
- c. Down casting
- d. Constraints
- e. Content Hugging Priority
- f. Compression Resistance Priority
- g. Hands on Auto layout

# 19

CHAPTER

## VIEWS, ANIMATIONS

### O1. Table View

- a. UITableViewController
- b. Working with multiple TableViews
- c. UITableViewCell
- d. Table View practices
- e. Custom Cell creation.
- f. Multi-View Applications
- g. view to view  
(Present model view controller )



**02**  
HRS



**10**  
HRS

- h. Navigation controller
- i. Tab bar controller
- j. Page view controller
- k. Split view controller
- l. Hands on Table View

## **O2. Create TO DO APP**

- a. Design UI
- b. Item Class
- c. Table View Items
- d. Adding Items Statically
- e. Table Header View
- f. Text Field Delegate Method
- g. Select an Item
- h. Break
- i. Deleting an Item
- j. No Duplicates
- k. Refactoring the Code
- l. Sections in Table View
- m. Animating the Transfer

## **O3. Touches**

- a. Touches Began and Ended
- b. Touches Moved
- c. Gestures
- d. Hands on Gestures
- e. Transformations
- f. Gestures and Transform
- g. Hands on touches

## O4. Animations

- a. Animation Blocks
- b. View Effects
- c. Hands on View Effects
- d. View Transitions
- e. Hands on animations

## O5. Maps and locations

- a. Maps
- b. Annotations
- c. Custom Annotations
- d. Location
- e. Location on the map
- f. Hands on Maps and location

# 20

CHAPTER

## **STORYBOARDS, CONCURRENCY**

### O1. Storyboards

- a. Storyboard File
- b. View Controller and Scene
- c. Segue
- d. Invoking a Segue
- e. XIB and Storyboards
- f. Table View Cell Prototype

  
**02**  
HRS

  
**08**  
HRS

## **O2. Testing and Debugging**

- a. Debugging with Print
- b. Breakpoints
- c. Exception Breakpoints
- d. Advanced Breakpoints
- e. Debug Navigator
- f. Unit Testing
- g. UI Testing
- h. Performance Testing
- i. Hands on testing and debugging

## **O3. Media and Devices**

- a. Camera
- b. Play an Audio
- c. Audio Recorder
- d. Play a Video
- e. AV Player Stand Alone
- f. Orientation
- g. Motion
- h. XCode Assets
- i. Hands on

## **O4. Concurrency**

- a. What is GCD?
- b. What is KVO?
- c. GCD Singleton
- d. Operation Queue
- e. Simple Operation



- f. Queued Operations - State and Value
- g. Queued Operations - Table View
- h. Queued Operations - Pending Operations
- i. Queued Operations - Table View Interactions
- j. Hands on

# 21

## CHAPTER

### **WEB SERVICES**

#### **O1. Web Services**

- a. What are web services?
- b. Why do we use web-services
- c. How apps function with the help of web-services
- d. Introduction to different web services
- e. JSON parsing, XML parsing
- f. Get and POST methods

#### **O2. Push Notifications**

- a. Notification
- b. NSNotification
- c. NSNotificationCenter
- d. UILocalNotifications
- e. Push Notification services



# 22

## CHAPTER

## **FIREBASE**

### **O1. About Firebase**

- a. What is firebase?
- b. Purpose of firebase
- c. Features of firebase
- d. Adding firebase to App
- e. Installation setup

### **O2. Firebase Integration and User Authentication**

- a. Setup Firebase
- b. Setup Authentication View Model
- c. Register Test User With Firebase
- d. Authentication View Model Bug Fix
- e. Keeping User Logged In
- f. Logging In / Signing Out

### **O3. Store the Data To Firestore**

- a. Setup Firebase Firestore
- b. Setup Firebase Storage
- c. Upload User Data To Cloud Firestore
- d. Upload Profile Photo UI
- e. Upload Profile Photo Presentation Logic
- f. Upload Profile Photo To Firebase Storage



**03**  
HRS



**10**  
HRS

## **O4. Fetching and Display User Data from Firestore**

- a. Fetch User Data From Firestore
- b. User Data Model
- c. Populating Settings View With User
- d. Load Profile Image & Authentication Bug Fix
- e. Fetch & Display Users In New Message View

Placement Assistance

**100%**

**135+** Professional Courses

Practical Sessions

**90%**

**67+** Global Pacts

Corporate Placements

**65%**

**170+** IT Companies Tie-Up

ELYSIUM  
GROUP OF  
COMPANIES

**ELYSIUM  
ACADEMY**

**PRIVATE  
LIMITED**

**AUTHORIZED INTERNATIONAL**

Partners

