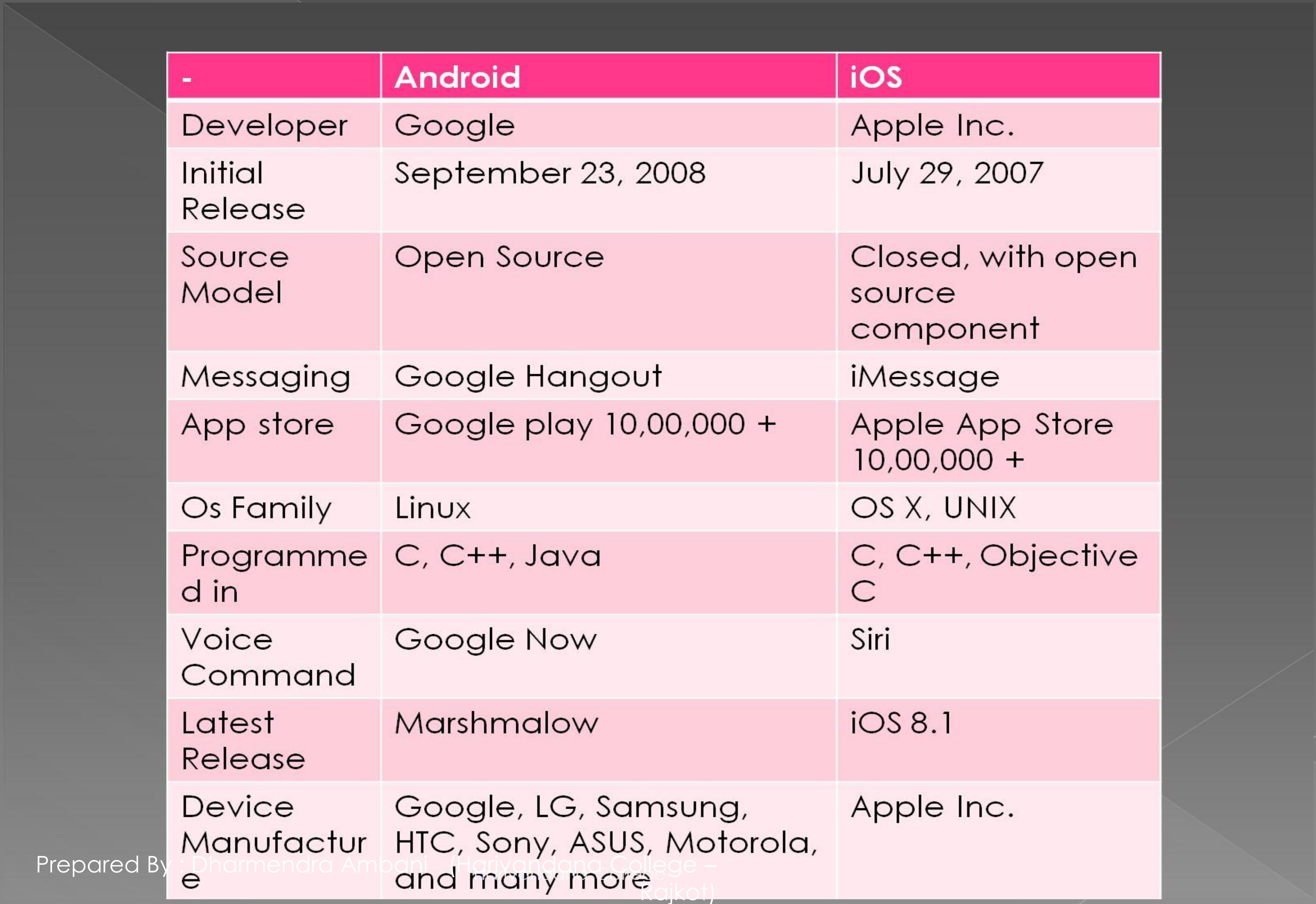
Introduction to iPhone

Chapter - 10

Prepared By : Dharmendra Ambani (Harivandana College – Rajkot)

Comparison between android & iOS



-	Android	iOS
Developer	Google	Apple Inc.
Initial Release	September 23, 2008	July 29, 2007
Source Model	Open Source	Closed, with open source component
Messaging	Google Hangout	iMessage
App store	Google play 10,00,000 +	Apple App Store 10,00,000 +
Os Family	Linux	OS X, UNIX
Programmed in	C, C++, Java	C, C++, Objective C
Voice Command	Google Now	Siri
Latest Release	Marshmalow	iOS 8.1
Device Manufacture	Google, LG, Samsung, HTC, Sony, ASUS, Motorola, and many more	Apple Inc.

Introduction to X-code(IDE)

- Xcode is an integrated development environment(IDE) containing a suite of software development tools developed by Apple for developing software for OS x and iOS First release in 2003.

Editor Looks

- 1) Toolbar
- In which option will display like run, stop, editor, view etc

- 2) Bundle
- In which option will display like .h, .m file like as android project directory.

- 3) Interface Builder
- In which you get find your required UI Component like Button, TextFiled, View etc.

- 4) TextEditor
- In which you can edit or create your custom text.

Framework

- A Framework is a hierarchical directory that encapsulates shared resources, such as a dynamic shared library, nib files, image files, localized strings, header files, and references documentation in a single package.

- A framework is also a bundle and its contents can be accessed using core foundation bundle services or the cocoa NSBundle class.

- Framework group realated resources together.
- Framework s can include a wider variety of resource types than libraries.
- Sharing resources reduces the memory of system and helps to improve performance.

- Controls
- A control communication tool between a user and an app
- Controls are simple, straightforward, and familiar to users because they appear throughout many iOS apps.

Design user interface for Button

- Button let a user initiate behavior with a tap.
- You communicate a button’s function through a textual label or with an image.
- Your app changes button appearance based upon user touch interactions, using highlighting, changes in the label or image, color, and state to indicate the button action dynamically.

- Purpose:
- Button allows users to
- -> Initiate behavior with a tap.
- > Initiate an action in the app with a single simple gesture.

- Content of Buttons:

- Type (RoundRect)
- Title (Plain)
- Font(Select Font)
- TextColor
- Shadow Color
- Image
- etc

TextField

- Text fields allow the user to input a single line of text into an app.

- You typically use text fields to gather small amount of text from the user and perform some immediate action, such as search operation, based on that text.

- Purpose :
- Text fields allows users to
- Enter text as input to an app.

- Content of Text Fields:

- Text
- Background
- Border style
- Font size
- Keyboard
- etc

- Elements similar to a text fields:
- 1) Text View :
- A text view accepts and displays multiple lines of text.
- Text Views support scrolling and text editing.

- Label :
- A label displays static text.

- Labels are often used in conjunction with controls to describe their intended purpose.

Creating And Building Simple Application

Cocoa Touch

- Cocoa touch is a user interface framework provided by Apple for building software applications for products like iPhone, iPad and iPod Touch.
- It is primarily written in Objective C language and is based on mac OS X.

- The high level application programming interfaces available in cocoa touch help to make animation, networking and adding appearace and behavior of the native platform to the develped applications possible with less code development.

- Main Features of cocoa touch include:

- 1) Audio And Video
- 2) Data Management
- 3) Graphics and Animation
- 4) Networking and Internet
- 5) User application like location based

MVC Inroducation

- MVC Software design pattern refers to three separate roles:

- Model
- View
- Controller

- The idea is that the objects in your application will take on these roles and work together to create and manage the user interface.

- The Model
- The model represents the data in your application.
- It's responsible for sorting, validating and organizing your data.

- The View

- The view is the user interface or what the user sees and interact with.

- You can create views programmatically through code using apple class such as UIView or you can create a XIB file to represent the view and visually layout your elements through Interface Builder.

- The Controller
- The controller manages the communication between the view and the model.
- It takes the data from the model and communicates it to the view for display.

- END