GOVERNMENT ARTS COLLEGE CBE

MOBILE APPLICATION DEVELOPMENT

UNIT-2

MSc COMPUTER SCIENCE

CONTENT

- MOBILE APPLICATION USERS
- SOCIAL ASPECT OF MOBILE INTERFACES
- ACCESSIBILITY
- DESIGN PATTERNS
- DESIGNING FOR THE PLATFORMS

MOBILE APPLICATION USERS

MOBILE APPLICATION USERS

- The Gestalt principles have had a considerable influence of design, describing how the human mind perceives and organize human data.
- Gestalt principles refers to the theories of visual perception developed German psychologists in the year 1920s.
- According to this principles, every cognitive stimulus is perceived by users in its simplest form.
- Key principles include proximity, closure, continuity, figure and ground, and similarity.

MOBILE APPLICATION USERS

Proximity:

Users tend to group objects together. Elements placed near each other are perceived in groups.

Closure:

If enough of a shape is available the missing pieces are completed by the human mind.

MOBILE APPLICATION USERS

Continuity:

The user's eye will follow a continuously perceived object. When continuity occurs, users are compelled to follow one object to another because their focus will travel in the direction they are already looking.

Figure and Ground:

A figure such as a letter on a page, is surrounded by white space or the ground

MOBILE APPLICATION USERS

Similarity:

Similar elements are grouped in a semi automated manner, according to the strong visual perception of color, form, size and other attributes.

SOCIAL ASPECT OF MOBILE INTERFACES

MOBILE INTERFACE

Determining and Reaching the Target Audience

- Holding an iPad with two hands in a meeting or thumbing through menus on the bottom of an Android phone screen.

MOBILE INTERFACE

The Social Aspect of mobile

- Users want to be connected and to share something within the world.

Usability

- In a perfect world, usability consideration would be a regular, ongoing part of the whole process, checking how pixels and fingers interact in the real world.

MOBILE USER INTERFACE DESIGN

- Design falls into the category of craftsmanship.
- Mobile devices are portable enough to carry at all times, connected to voice and data networks, and contextually aware by using sensors and networks to preemptively complete tasks.

Current mobile limitations:

- Band width
- Times when users cannot access wireless
- Lack of technical capabilities like flash.

INFORMATION ARCHITECTURE

- A strong foundation by organizing, labeling, and identifying all content that user will perceive.
- A growing segment of interactive content is faceted, tagging chunks of content with attributes.

INFORMATION ARCHITECTURE

.

- Classifying information by:
  - Name
  - Sequence
  - Fixed relationship between values.
  - The distance between values
  - Proximity

INFORMATION ARCHITECTURE

- The visual display of information is how people connect with loved ones, colleagues and friends.

Information Display

- A microwave has a simple display.
- People identify signals
- Interpret the meaning of these signals.
- Determine the goal accordingly.

MOBILE USER INTERFACE DESIGN

- Developers can concentrate on the precise features on the devices.
- Application can also use exciting new features like Zooming, Swiping, Tapping, Turning, and Shaking.

UNDERSTANDING MOBILE PLATFORMS

- Developers can take advantage of native functionality across mobile devices.
- More than a smaller, web like interface, an Android, BlackBerry,WP7,or iOS device can make phone calls as well as record and transmit contextual information like geo-location.

ACCESSIBILITY

ACCESSIBILITY

- An application that is easier for people to use with poor or diminished vision, limited dexterity , or a cognitive will be as a better way to reach existing user.

FIVE POINTS OF MOBILE ACCESSIBILITY

- **OVERALL BEHAVIOR:** Independent of specific device features , the guidelines for a general mobile experience.
- **NAVIGATION AND LINKS:** The ease of interaction with hyperlinks on the mobile device interfaces.
- **PAGE LAYOUT AND CONTENT:** How content is designed on each page, and how chunks of content are laid out for accessible consumption.

FIVE POINTS OF MOBILE ACCESSIBILITY

- **PAGE DEFINITION:** Defining content areas for mobile device interpretation.
- **USER INPUT:** Considerations of available mobile device input methods.

APPLICATION FEATURES

- Hearing
- Vision
- Speech
- Dexterity
- Cognition

HEARING

- Moderate to profound hearing loss can make it difficult for may people to communicate with mobile standard telephone, but many mobile devices offer features that make promising progress.
- For moderate to profound hearing loss , adjustable volume control offers a simple way to connect with mobile content.

VISION

- Many user depend on tactile, audio, or other sensory alerts to access resources using mobile devices.
- People with low vision through complete blindness may benefit from sliding or flipping a phone to answer and end calls and are likely to consider the hardware first.
- Popular and industry-standard devices without a flip or slide may be modified to meet the needs of low-vision users.

SPEECH

- Users with speech-related accessibility issues with out-related functionality using text features.
- Text messaging , email , and predictive text are popular solutions.
- Consider allowing users to save text inputs to reuse personalized outputs.”I am leaving the office now” could be recycled from a personalized dashboard within a social mobile application.

DETERITY

- Many people have difficulty for various reasons with the fine controls on a mobile device.
- A hands-free mode can limit how much the phone must be held to properly navigate, which benefits low-dexterity user as busy cooks, lost driver, and distracted parents.

COGNITION

- From birth to trauma to age-onset impairments, a large number of mobile devices are used by people with cognitive accessibility issues.
- Any feature to reduce the cognitive load-the amount of information users must hold in their memory-is helpful.
- Associating images or photographs with long lists of information such as memory-helpful.

DESIGN PATTERNS

INTRODUCTION…

- Hardware and operating system become irrelevant far quicker than design that reaches users.
- A design pattern recycles and repurposes components , reusing the best ideas.
- More than times efficiency, patterns have been refined by use.

NAVIGATION…

- Good design makes it clear how users can move through and use application features

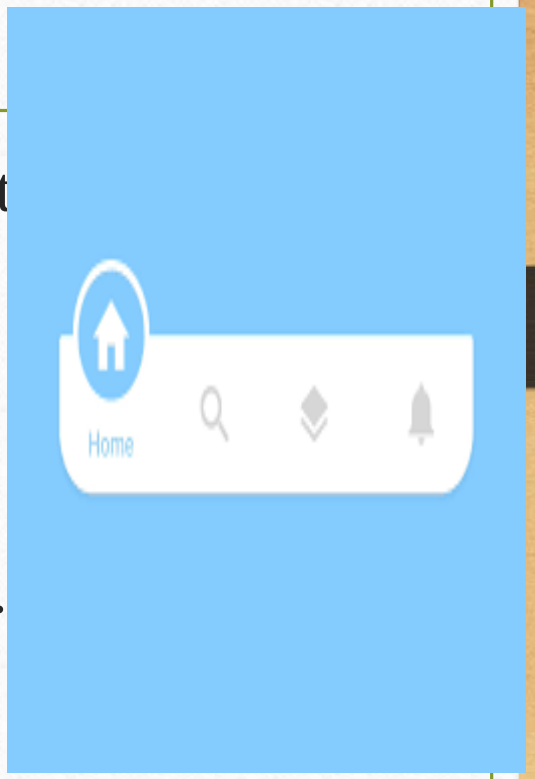
ANNUNCIATOR PANEL…

- Annunciator panel seen at the top gives information on the state of the mobile device.
- Though each mobile device will provide a slightly different panel, developers can modify or suppress the annunciator panel
  - which lists the hardware features such as network connection and battery power.
- Because the information in this area is only notifications , users will not usually have any direct interaction with the annunciator panel.

FIXED MENU…

- A menu that remain fixed to the view port as users roam cont in many situation.
  - ❑When users need immediate access to frequently selected functionality on multiple screens.
  - ❑When a revealable menu is already used on the screen.

➢Because fixed menus are always accessible(at the bottom and top) users can interact with page contents as needed



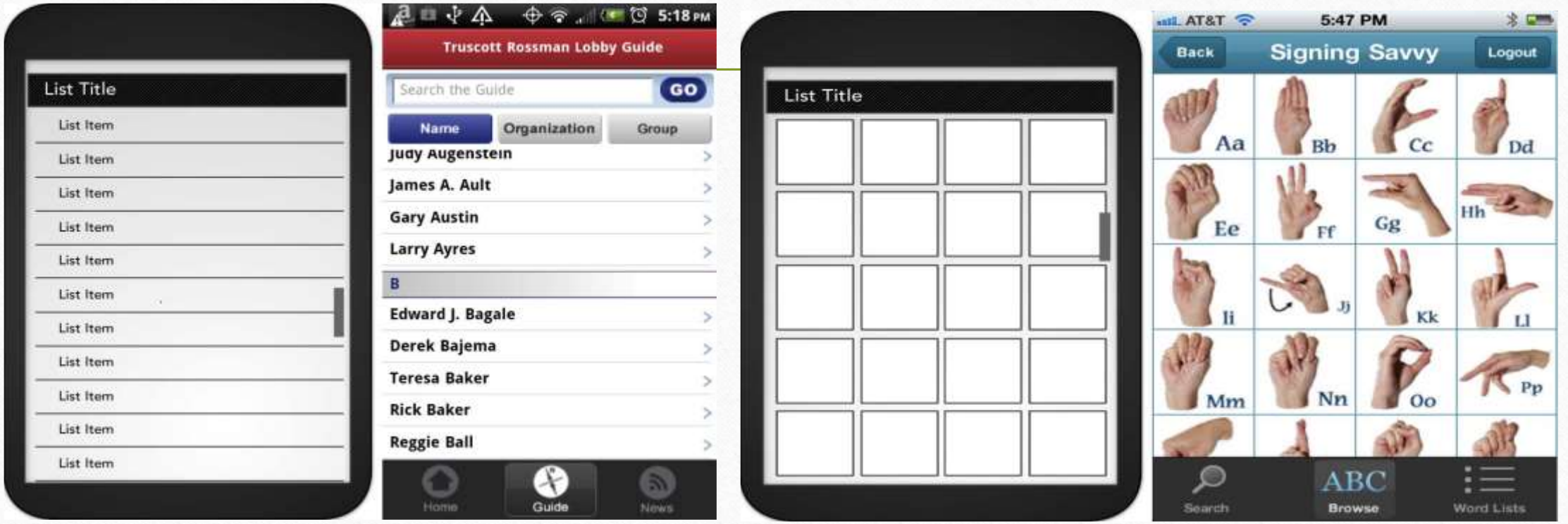
EXPANDABLE MENU…

- When all function options for one page cannot fit on the viewport, an expanding menu can provide selectively available options.
- A full menu similar to shown in the image will reveal once prompted.
- A gesture like a swipe or double tap, may prompt the reveal as well as selecting an on-screen icon.
- Users may exit a menu using a back button, a close button that is part of the revealed menu list.

SCROLL…

- It is best to limit scrolling, limiting application screens to the size of the viewport whenever possible.
- When scrolling must occur, check that the design communicates the area that can be scrolled, the current context of a user, and the ratio of the current view to the total content available.
- If there is an error generating information with predictive retrieval, the application will use explicit retrieval, with users selecting a “refresh” button to load more content.

SCROLL…


List Title
List Item
List Item
List Item
List Item
List Item
List Item
List Item
List Item
List Item
List Item
List Item
5:18 PM
Truscott Rossman Lobby Guide
Search the Guide
GO
Name
Organization
Group
Judy Augenstein
James A. Ault
Gary Austin
Larry Ayres
B
Edward J. Bagale
Derek Bajema
Teresa Baker
Rick Baker
Reggie Ball
Home
Guide
News
List Title
AT&T
5:47 PM
Back
Signing Savvy
Logout
Aa
Bb
Cc
Dd
Ee
Ff
Gg
Hh
Ii
Jj
Kk
Ll
Mm
Nn
Oo
Pp
Search
ABC
Browse
Word Lists

NOTIFICATION AND FEEDBACK…

- If the user needs new information, application creators must use notifications.
- Feedback is the user-perceived result of an interaction, providing immediate confirmation like a color change, message, or being led to a new page.
- Notifications, like those depicted in catch user attention to indicate that further action may be required, or that an action (such as a download or an update) has been completed.

CONTROL STRUCTURE

- Input mechanism, that controls users to manipulate to enter or modify the data.
- Use standard controls to create positive mind set to use app.
- Mobile platform differ on the minimal size of any touch point it should be large enough to touch using finger.

CONTROL STRUCTURE

- Controls should be on the bottom of the phone, Information should be on the top.

INTERVENING SCREEN

- Between delivering personalized functionality and life changing brilliance to users there will be times that content must load,or a device becomes locked while the users looks up from their device.

LOADING SCREEN

- It the progress of "loading" screen signals when data is loading, it can't show previous screen.
- It accumulate the information of level of loading.

ADVERTISEMENT

- The common mistakes in application is advertisement.
- Obvious that Advertising will confuse the application content.

CONTENT STRUCTURE AND USAGE

- Mobile application users are there to consume,produce,and share content,it does not matter how pretty or useful the application may seem to stakeholders if content is worthless to users.
- Users need to quickly locate and effectively use information.

CONTENT STRUCTURE AND USAGE

- Give every application a strong foundation by organizing, labeling, identifying all content user perceived.

TYPOGRAPHY

- Every application will often be textual content.
- Size, shape, contrast, color & position all matter.

PLAIN LANGUAGE

- "Plain Language" is the idea that content producers should speak in the language of their users in a way that is clear and understandable to the audience.
- Content usability is one of the most important factors for task success.
- Plain Language is usable language

.

TIPS FOR MOBILE APPLICATION CREATERS

- Omit unnecessary words .
- Use simplest form of verbs.
- Use simple short words.
- Avoid jargon and abbreviations.
- Use pronouns only.

DESIGNING FOR THE PLATFORMS

MOBILE PLATFORMS

Mobile operating system

- Android
- iOS
- BlackBerry OS
- Windows Phone 7

ANDROID

- Android is a mobile operating system based on a modified version of the Linux kernel and other open source software, designed primarily for touchscreen mobile devices such as smartphones and tablets.
- Initial release date: 23 September 2008
- Developer: Google and the Open Handset Alliance

ANDROID

- Android uses the Linux kernel. Because Linux is open-source, Google's Android developers could modify the Linux kernel to fit their needs. Linux gives the Android developers a pre-built, already maintained operating system kernel to start with so they don't have to write their own kernel.

ANDROID - ARCHITECTURE

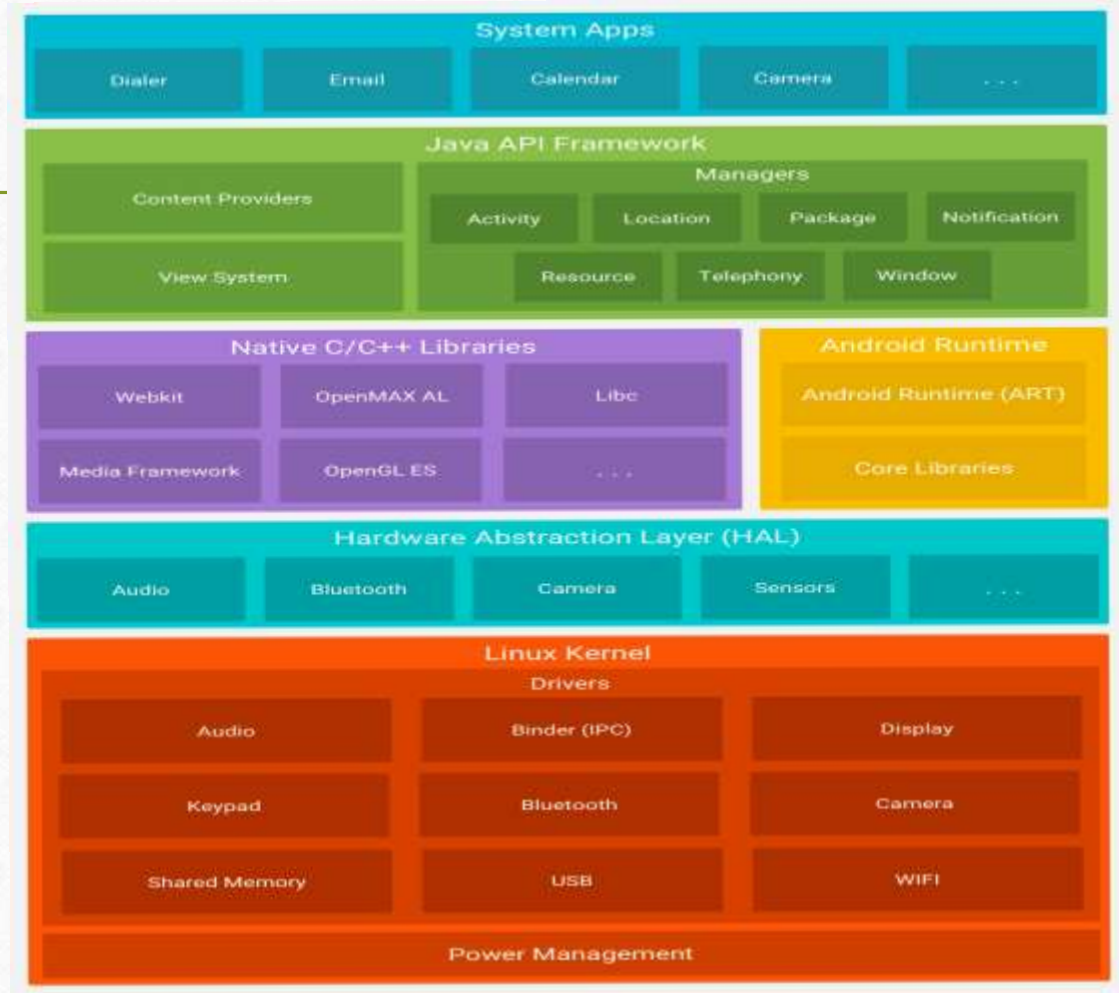


TABLE OF CONTENTS

1.THE LINUX KERNEL

- ✓ The foundation of the Android platform is the Linux kernel. For example, the Android Runtime (ART) relies on the Linux kernel for underlying functionalities such as threading and low-level memory management.
- ✓ Using a Linux kernel allows Android to take advantage of key security features and allows device manufacturers to develop hardware drivers for a well-known kernel.

2.HARDWARE ABSTRACTION LAYER (HAL)

- ✓ The hardware abstraction layer (HAL) provides standard interfaces that expose device hardware capabilities to the higher-level Java API framework.
- ✓ The HAL consists of multiple library modules, each of which implements an interface for a specific type of hardware component, such as the camera or Bluetooth module.

2.HARDWARE ABSTRACTION LAYER (HAL)

- ✓ When a framework API makes a call to access device hardware, the Android system loads the library module for that hardware component.

3.ANDROID RUNTIME

Some of the major features of ART include the following:

- ✓ Ahead-of-time (AOT) and just-in-time (JIT) compilation
- ✓ Optimized garbage collection (GC)
- ✓ Better debugging support, including a dedicated sampling profiler, detailed diagnostic exceptions and crash reporting, and the ability to set watch points to monitor specific fields.

3.ANDROID RUNTIME

- ✓ Android also includes a set of core runtime libraries that provide most of the functionality of the Java programming language, including some Java 8 language features, that the Java API framework uses.

4.NATIVE C/C++ LIBRARIES

- ✓ Many core Android system components and services, such as ART and HAL, are built from native code that require native libraries written in C and C++.
- ✓ The Android platform provides Java framework APIs to expose the functionality of some of these native libraries to apps.
- ✓ For example, user can access OpenGL ES through the Android framework's Java OpenGL API to add support for drawing and manipulating 2D and 3D graphics in app.

4.NATIVE C/C++ LIBRARIES

- ✓ If user is developing an app that requires C or C++ code, he can use the Android NDK to access some of these native platform libraries directly from native code.

5.JAVA API FRAMEWORK

The entire feature-set of the Android OS is available to user through APIs written in the Java language.

These APIs form the building blocks user need to create in Android apps by simplifying the reuse of core, modular system components and services, which include the following:

5.JAVA API FRAMEWORK

- ✓ A rich and extensible View System that can be used by the user to build an app's UI, including lists, grids, text boxes, buttons, including an embeddable web browser.
- ✓ A Resource Manager provides access to non-code resources such as localized strings, graphics, and layout files.
- ✓ A Notification Manager that enables all apps to display custom alerts in the status bar.

5.JAVA API FRAMEWORK

- ✓ An Activity Manager that manages the lifecycle of apps and provides a common navigation back stack.
- ✓ Content Providers that enable apps to access data from other apps, such as the Contacts app, or to share their own data.
- ✓ Developers have full access to the same framework APIs that Android system apps use.

6.SYSTEM APPS

- Android comes with a set of core apps for email, SMS messaging, calendars, internet browsing, contacts, and more. Apps included with the platform have no special status among the apps that the user chooses to install. So a third-party app can become the user's default web browser, SMS messenger, or even the default keyboard .

6.SYSTEM APPS

- The system apps function both as apps for users and to provide key capabilities that developers can access from their own app. For example, if user would like to deliver an SMS message, he need not build that functionality himself—he can invoke whichever SMS app is already installed to deliver a message to the recipient he specify.

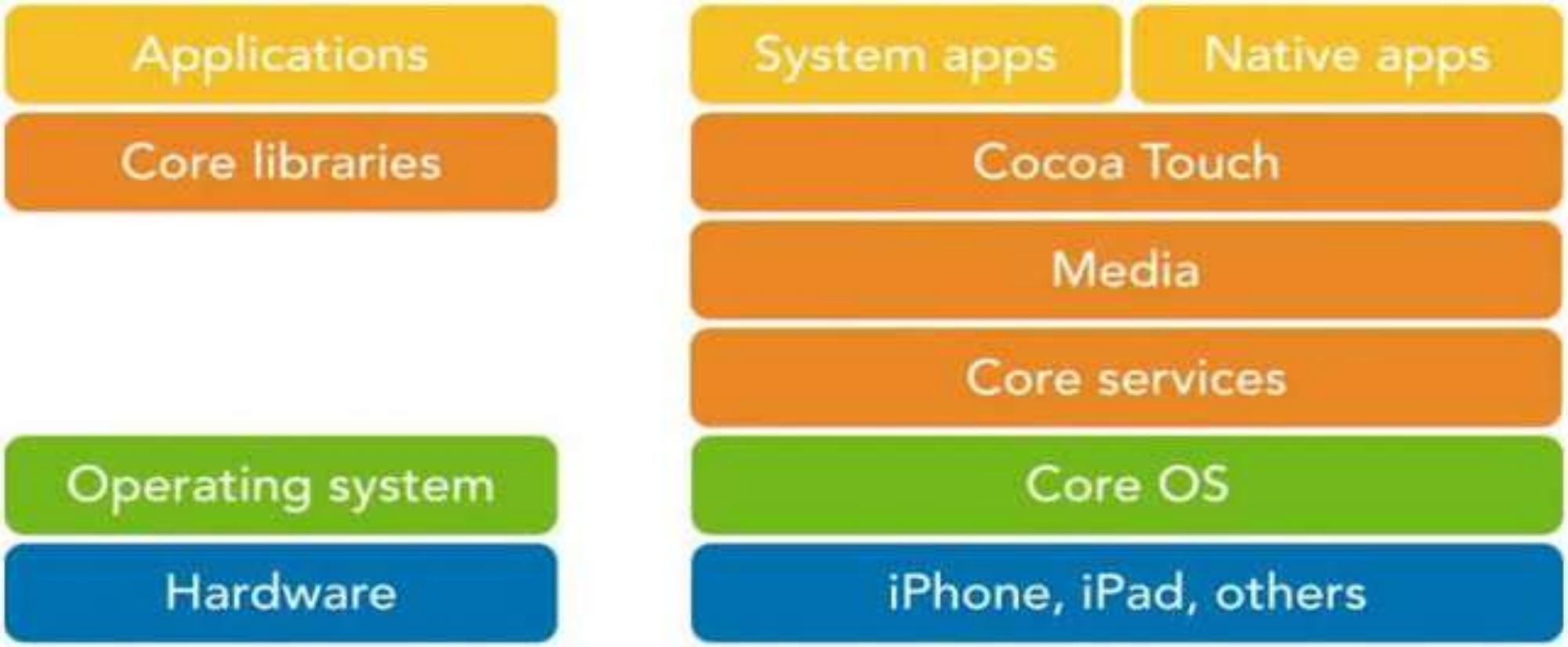
iOS

- The iOS is the operating system created by Apple Inc. for mobile devices.
- The iOS is used in many of the mobile devices for apple such as iPhone, iPod, iPad etc.
- The iOS is used a lot and only lags behind Android in terms of popularity.
- Developer: Apple Inc

iOS

- OS family: Unix-like, based on Darwin (BSD), iOS
- Default user interface: Cocoa Touch (multi-touch, GUI) iPod, iPad etc.
- The iOS is used a lot and only lags behind Android in terms of popularity.
- Initial release date:29 June 2007

iOS Architecture



iOS - ARCHITECTURE

- The iOS architecture is layered.
- It contains an intermediate layer between the applications and the hardware so they do not communicate directly.
- The lower layers in iOS provide the basic services and the higher layers provide the user interface and sophisticated graphics.

LAYERS IN IOS ARCHITECTURE

① Core OS
② Core Services
③ Media
④ Cocoa Touch

2.CORE SERVICES

There are many frameworks available in the core services layer.

- ✓ Cloudkit Framework
- ✓ Core Foundation Framework
- ✓ Core Data Framework
- ✓ Address Book Framework
- ✓ Core Motion Framework
- ✓ Healthkit Framework
- ✓ Core Location Framework

2.CORE SERVICES

All the iOS technologies are build on the low level features provided by the Core OS layer. These technologies include Core Bluetooth Framework, External Accessory Framework, Accelerate Framework, Security Services Framework, Local Authorization Framework etc.

2.CORE SERVICES

Cloudkit Framework

The data can be moved between the app the iCloud using the Cloudkit Framework.

Core Foundation Framework

This provides the data management and service features for the iOS apps.

2.CORE SERVICES

Core Data Framework

The data model of the model view controller app is handled using the Core Data Framework.

Address Book Framework

The address book framework provides access to the contacts database of the user.

2.CORE SERVICES

Core Motion Framework

All the motion based data on the device is accessed using core motion framework.

Healthkit Framework

The health related information of the user can be handled by this new framework.

.

2.CORE SERVICES

Core Location Framework

This framework provides the location and heading information to the various apps.

3.MEDIA

The Media layer enables all the graphics, audio and video technology of the system. The different frameworks are:

Core Graphics Framework

This provides support for 2-D vector and image based rendering and is the native drawing engine for iOS apps.

3.MEDIA

Core Animation

The Core Animation technology optimizes the animation experience of the apps.

Media Player Framework

This framework provides support for playing playlists and enables the user to use their

iTunes library.

4.COCOA TOUCH

The cocoa touch layer provides the following frameworks

EventKit Framework

This shows the standard system interfaces using view controllers for viewing and changing calendar related events.

.

4.COCOA TOUCH

GameKit Framework

This provides support for users to share their game related data online using Game center.

MapKit Framework

This provides a scrollable map which can be included into the app user interface.

REFERENCES

- Jeff McWherter and Scott Gowell , “Professional Mobile Application Development”, John Wiley & Sons, 2012.
- https://www.slideshare.net/mobile/shahidmasih1/6-mobile-user-interface-design
- https://www.slideshare.net/mobile/maxdesign/accessibility-interface-design
- https://www.slideshare.net/mobile/SanaeBEKKAR/introductiontodesignpatterns
- https://www.seamgen.com/blog/mobile-app-development-process