

Becoming a Pro

IN Mobile Applications Testing



Overview: Mobile APPS

Categories

Types

Distribution/Installation/Logs

Mobile Test Industry Standards

Remote Device Access (RDA)

Emulators

Simulators

Troubleshooting Guide

App Risk Analysis

MOBILE APPS: Categories



Utilities



Entertainment



Games



News



Productivity



Lifestyle



Social Networking

MOBILE APPS: Utilities



Calculators

Note-pads



Communi-
cation.
apps

Weather
apps



MOBILE APPS: Entertainment



Face
Juggler



Ice Effex



Duolingo



DubSmash

MOBILE APPS: Games



Angry
Birds



Sudoku

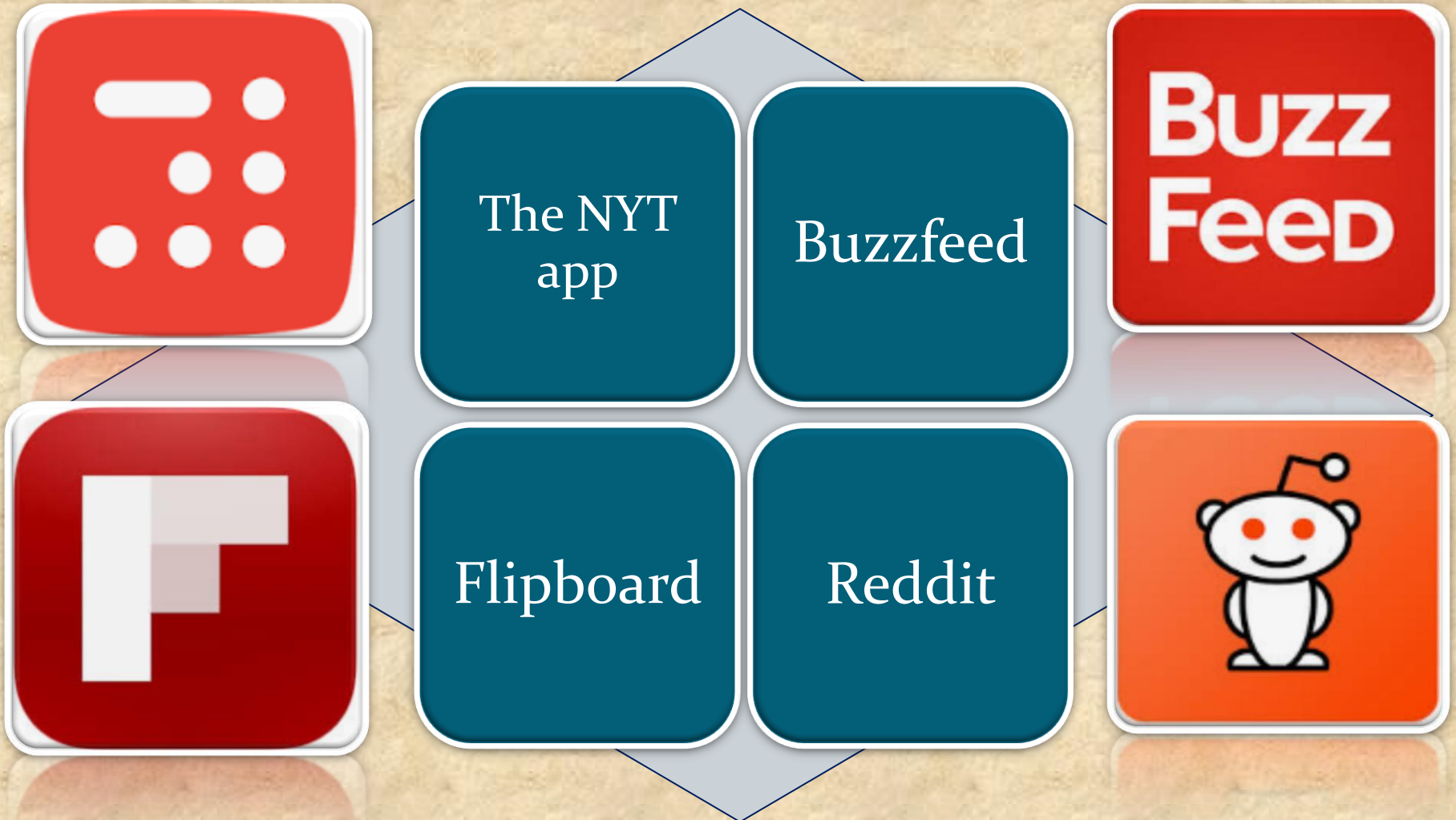


Trivia
Crack



Candy
Crash Saga

MOBILE APPS: NEWS



MOBILE APPS: Productivity



Finance
apps

Calendars



Translators

Grocery list
makers



MOBILE APPS: Lifestyle



Music apps

Travel
Apps



Food &
Drink apps

Dating
apps



MOBILE APPS: Social Networking



Overview: Mobile APPS

➤ Categories

➤ **Types**

➤ Distribution/Installation/Logs

➤ Mobile Test Industry Standards

➤ Remote Device Access (RDA)

➤ Emulators

➤ Simulators

➤ Troubleshooting Guide

➤ App Risk Analysis

MOBILE APPS: Types



Tree basic types of "app"



Native

Built specifically to the needs of the various operating systems such as Apple's iOS or Android



Web

Websites built using HTML that are designed specifically for smaller screens



Hybrid

Native app shell with feeds from the website

MOBILE APPS: Native APP

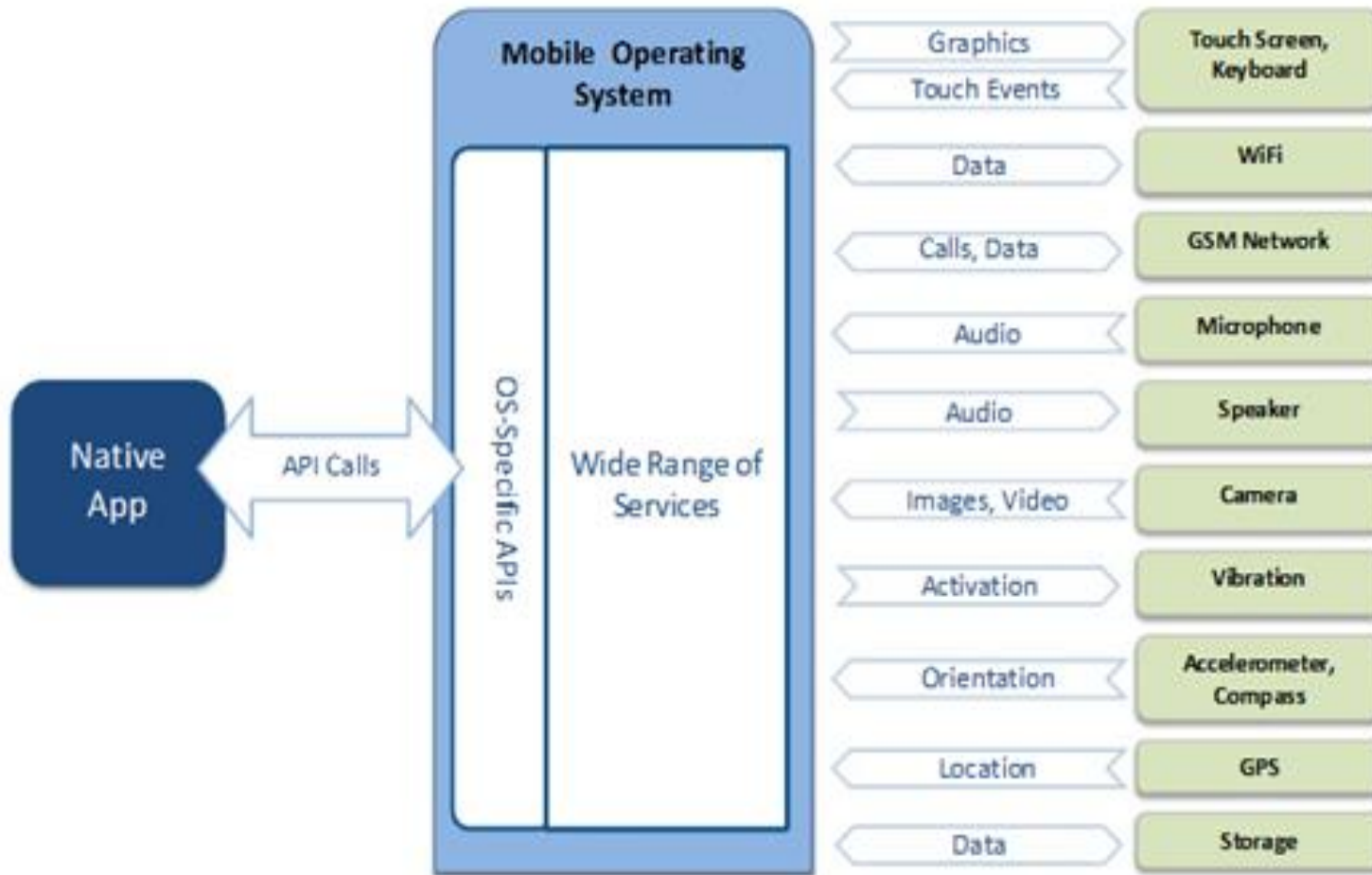


Written using the default language for the mobile platform, which is Objective C or Swift for iOS and Java for Android.

Compiled and executed directly on the device.

Using the platform SDK (API), the app can communicate with the platform to access device data or load data from an external website using http requests.

MOBILE APPS: Native APP



MOBILE APPS: Native APP



PROS

Native APIs

Performance

Same environment

CONS

Language requirements

Not cross platform

High level of effort

MOBILE APPS: WEB APP

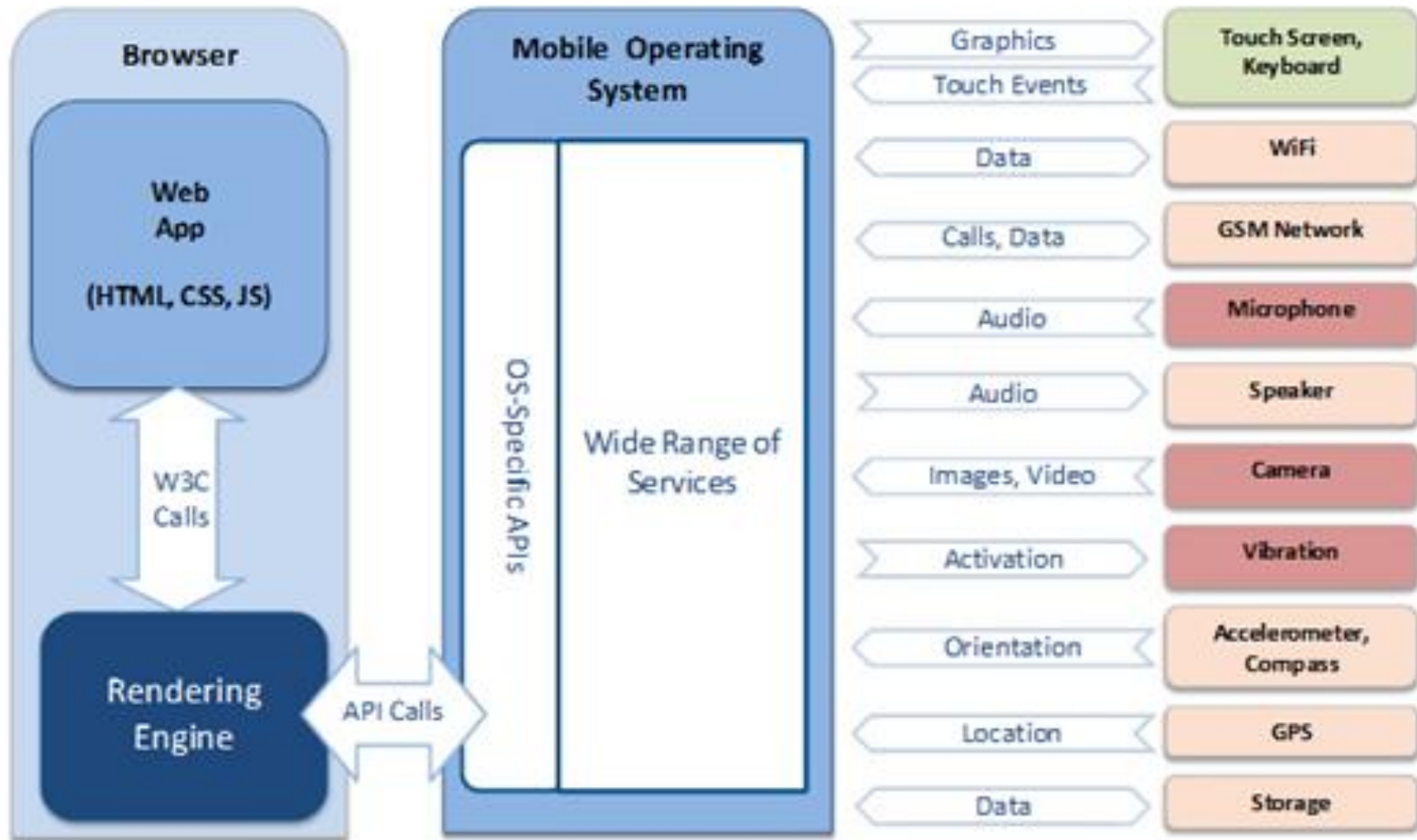


Mobile websites are applications that work well on a mobile device, but are accessed through the mobile browser.

These websites viewed on a mobile device in a mobile browser, with the exception of being designed to fit a mobile device screen size.

Web applications commonly use a combination of server-side script (ASP, PHP, etc) and client-side script (HTML, Javascript, etc.) to develop the application..

MOBILE APPS: WEB APP



MOBILE APPS: WEB APP



PROS

Maintainability

No
installation.

Cross platform.

CONS

No native access

Requires
keyboard to load

Limited user
interface.

MOBILE APPS: HYBRID APP

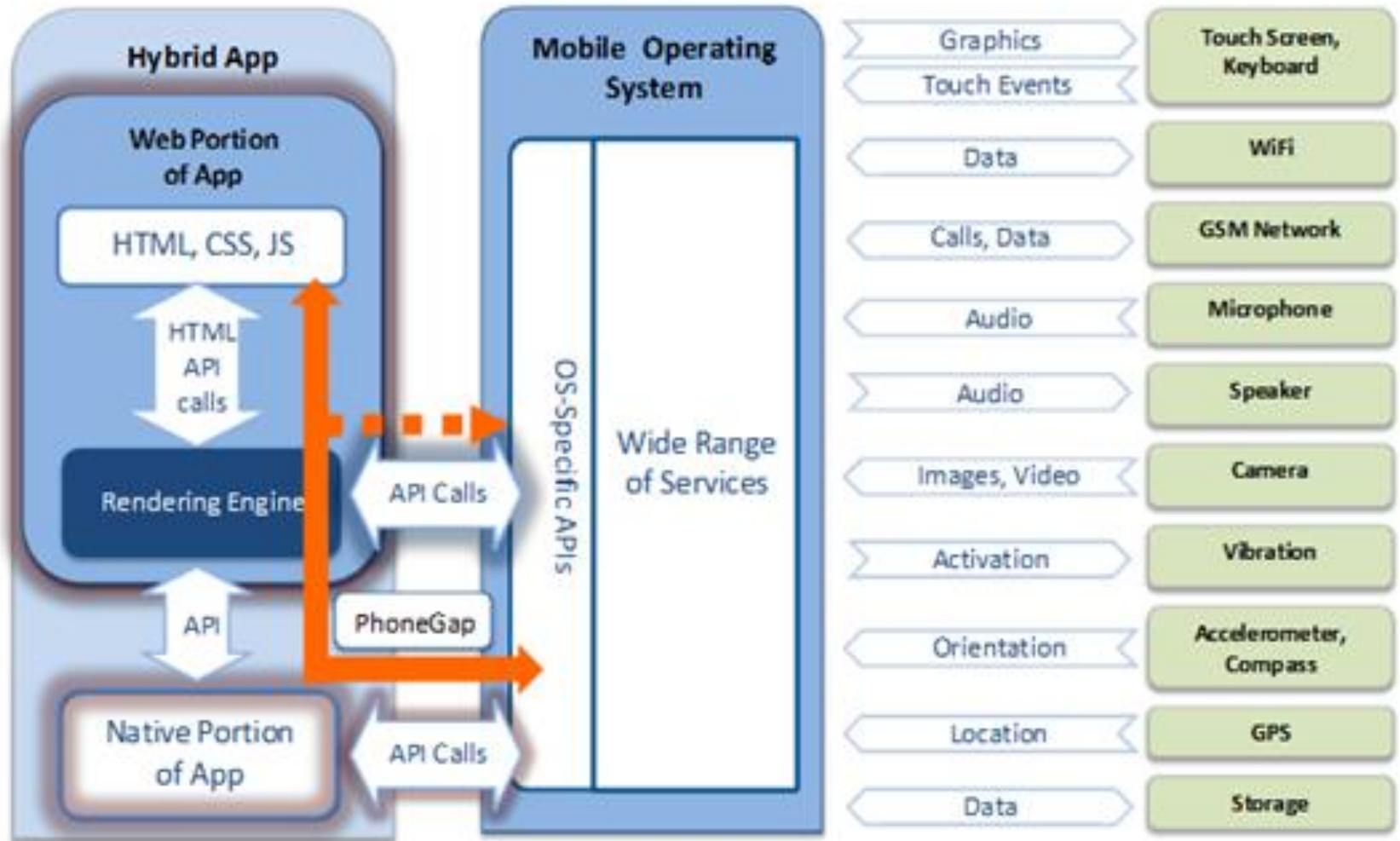


A hybrid app is one that combines elements of both native and Web applications

Hybrid apps are often mentioned in the context of mobile computing

Native source code is written and compiled into an executable program and a web based component written with HTML, JavaScript, and CSS

MOBILE APPS: HYBRID APP



MOBILE APPS: HYBRID APP



PROS

Cross platform

Same skills as web development

Access to device

Ease of development

CONS

Web view limitations

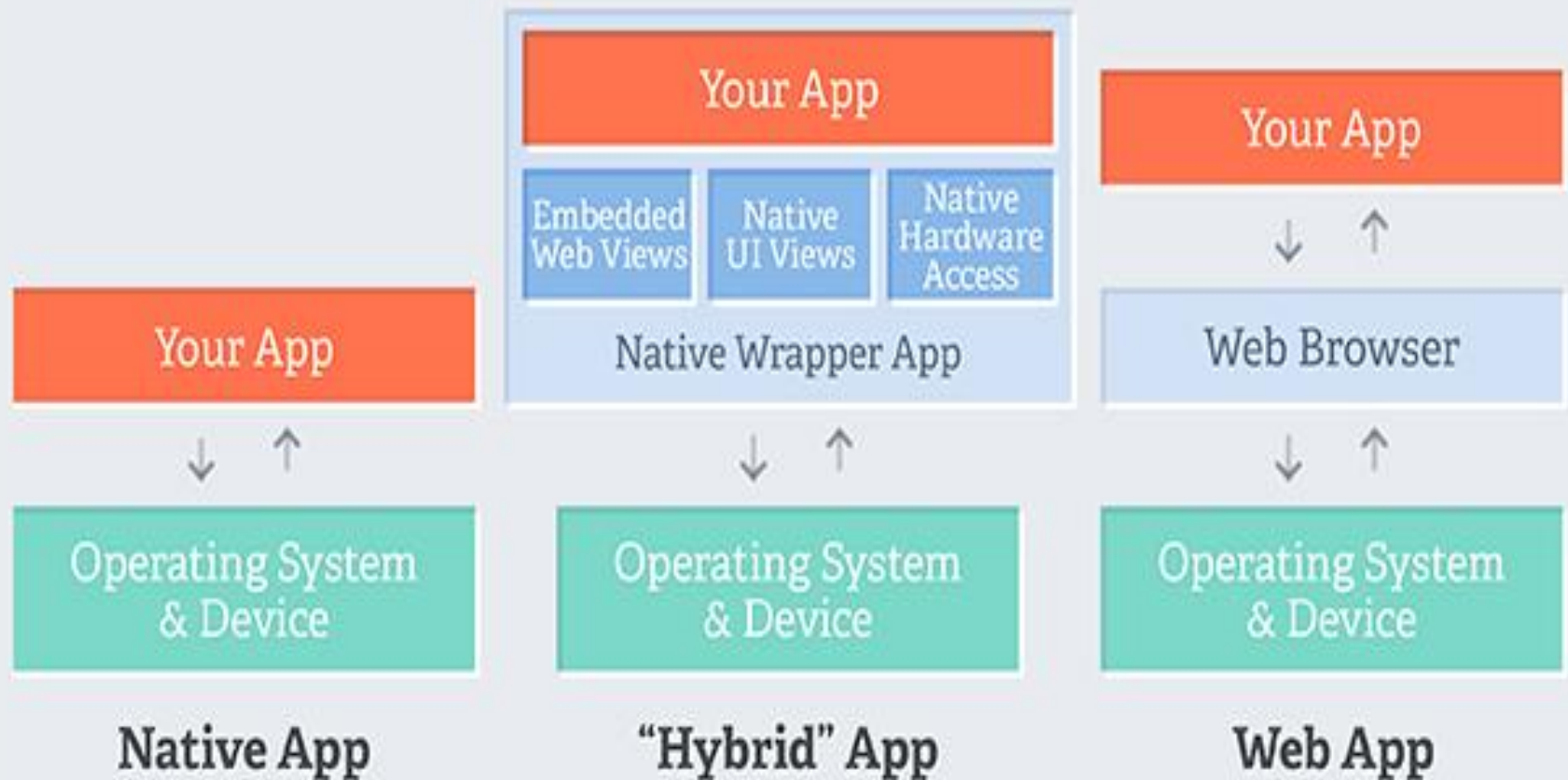
Native via plugins

No native user interface controls

Experienced developers

MOBILE APPS: SUMMARY

Mobile App Technology Stacks



MOBILE APPS: SUMMARY

Native Mobile App

- IOS - Developed using Objective-c
- Android - Developed using JAVA
- Need to Install from APP Store.
- Available as an Application on Device.

Mobile Web App

- Developed using typical web development technology - HTML, CSS, Java Script.
- View size of the Web page fit to the real-estate of the device.
- Accessed through the browsers on the device

Hybrid Mobile App

- Wrapping the HTML and creating Native like look and feel (HTML within the app itself). Framework like Phone Gap support this development.
- Native Mobile App with Web view control and render the HTML directly on the web view (HTML Rendered from enterprise server).
- View size of the Web page fit to the real-estate of the device.
- Accessed through the browsers on the device

Mobile APPS : Conclusion

LIST	Native	HTML5	Hybrid
App Features			
Graphics	<i>Native APIs</i>	<i>HTML, Canvas, SVG</i>	<i>HTML, Canvas, SVG</i>
Performance	<i>Fast</i>	<i>Slow</i>	<i>Slow</i>
Native look and feel	<i>Native</i>	<i>Emulated</i>	<i>Emulated</i>
Distribution	<i>Appstore</i>	<i>Web</i>	<i>Appstore</i>
Device Access			
Camera	<i>Yes</i>	<i>No</i>	<i>Yes</i>
Notifications	<i>Yes</i>	<i>No</i>	<i>Yes</i>
Contacts, calendar	<i>Yes</i>	<i>No</i>	<i>Yes</i>
Offline storage	<i>Secure file storage</i>	<i>Shared SQL</i>	<i>Secure file system, shared SQL</i>
Geolocation	<i>Yes</i>	<i>Yes</i>	<i>Yes</i>
Gestures			
Swipe	<i>Yes</i>	<i>Yes</i>	<i>Yes</i>
Pinch, spread	<i>Yes</i>	<i>No</i>	<i>Yes</i>
Connectivity	<i>Online and offline</i>	<i>Mostly online</i>	<i>Online and offline</i>