IST346: Mobile Services

Mobile Operating Systems

Apple iOS

- iPhone and iPad
- AppStore
- Code written in Objective-C or Swift

Google Android

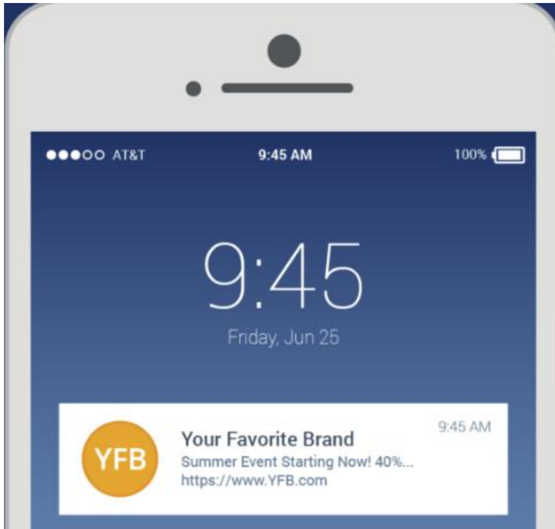
- Google Phones, Samsung Phones/Tablets and Many More
- Google Play
- Code written in Java

How is mobile different from web?

- Phones have a browser, but most of our experiences use native apps downloaded from application stores.
- Native offers advantages:
  - Push Notifications
  - Offline data
  - Access to hardware features, camera, microphone, GPS, accelerometer
  - Better performance!!!
- That is true today, but the gap is closing.

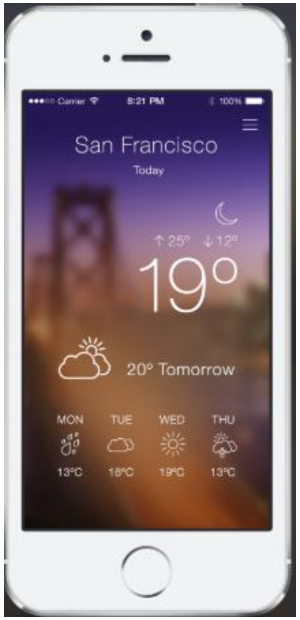
Push Notifications

- Messages that pop up on a mobile device.
- Each mobile operating system, including iOS, Android, Fire OS, Windows, and BlackBerry, has its own OS push notification service (OSPNS).
- **App publisher**. The app publisher enables their app with an OSPNS.
- **Client app**. In installed on your device and receives incoming notifications.
- Not the same as SMS/TXT

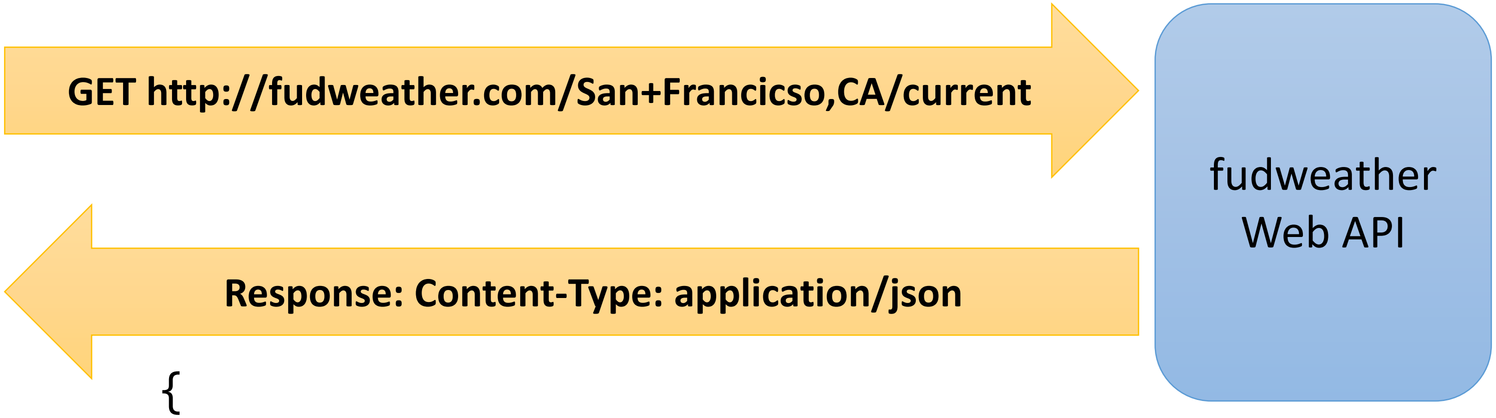


Your Favorite Brand
Summer Event Starting Now! 40%...
https://www.YFB.com

Mobile Apps and Web API’s



Weather App
On Phone



GET http://fudweather.com/San+Francicso,CA/current

fudweather
Web API

Response: Content-Type: application/json

```
{
    “temperature” : 19,
    “conditions” : “partly-cloudy”,
    “tomorrow” : 20
}
```

The Phone App is responsible for calling the API (requesting content) and drawing the API output on the screen

Mobile Development

Developing Mobile Apps

- There are a variety of methods one can choose to make a mobile app.
- **Native** – Write the app in the native language of the mobile operating system
- **Hybrid** – Write the app in an intermediate language which then either compiles to native or
- **Progressive Web App** – Write the app in HTML and JavasScript as you would a website.

Pros and Cons

Approach	Pros	Cons
Native	- Best performance - Access to all hardware features	- Different code for each platform iOS/Android - Challenging to maintain - Different deployment to each app store
Hybrid	- One programming language for all platforms - Access to most hardware features - Simplifies deployment to app stores	- Not suitable performance for some applications - Access to “most” hardware features
Progressive Web App	- Leverage Web development skillsets HTML/ JavaScript - No need to deploy to app stores	- Not suitable performance for some applications - Not supported by all platforms - Least access to hardware

Native App Development

- Written in Java on Android and Objective-C / Swift on iOS
- You need a Mac to develop for iOS
- Developer tools like Xcode and Android Studio are used to design the layouts for the application.

Hybrid-Native App Development

- Hybrid Native allow you to use a single programming language which is transpiled (cross-compiled) into Java on Android and Objective-C on iOS.
- Examples:
  - Xamarian,
  - Unity Gaming Engine
  - Titanium Mobile

Hybrid App Development

- Hybrid App development uses a single language to target both iOS and Android. The application runs in its original language (typically JavaScript) but is wrapped in a native code shell for deployment in the app stores.
- Layouts are designed in HTML and CSS
- Examples:
  - Cordova / Phonegap
  - Ionic
  - Google Flutter
  - React Native
  - Nativescript

Progressive Web Apps

- Progressive web apps are written in HTML, CSS, and JavaScript.
- The look like native apps but run in a web browser.
- They have access to most native features like the camera, GPS, and accelerometer.
- They are not deployed to app stores (yet)

Demos: https://pwa.rocks/

Mobile Device Management

Mobile App Deployment

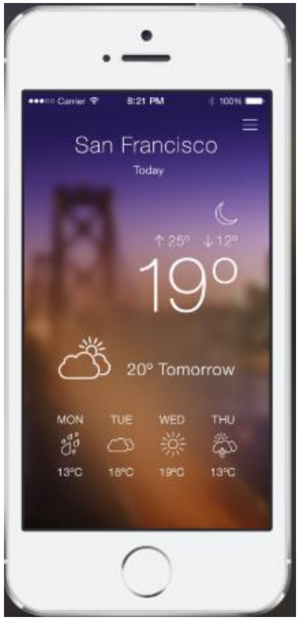
- Getting apps into the app stores
- Then search the app store for what you want
- And install it
- You know this and it works for consumers.
- How is this done in the enterprise?

MDM Systems

- Mobile Device Management Systems
- Remotely install apps on devices
- Run separate internal “App Stores”
- Control which apps can be launched / accessed.
- Locate lost devices / Remote Wipe (Erase)
- Examples:
  - IBM Maas 360
  - MDM for Office 365 (Intune)
  - VMWare AirWatch

Mobile Back-End as a Service (MBaaS)

Mobile Back Ends



Weather App
On Phone

GET http://fudweather.com/San+Francicso,CA/current

Response: Content-Type: application/json

```
{
    “temperature” : 19,
    “conditions” : “partly-cloudy”,
    “tomorrow” : 20
}
```

fudweather
Web API
(Mobile Back
End)

The Mobile Back End contains the Web APIs required by the application and other services like push notifications.

MBaaS Features

MBaaS can save you significant development and administrative resources in your mobile app initiatives.

- Data storage with Web API’s
- Push Notifications
- Auto-Scalability – grows with demand
- Client device analytics and usage statistics
- Handles Identity Management and Social Authentication

MBaaS Providers

- Microsoft Azure Mobile Services
- Google Firebase
- AWS Mobile
- Apple Cloudkit
- Kinvey