

Mobile App Engineering

SS 2016

Paul Spiesberger



Who is Paul?



INSO & DECO



RISE Experience



Airport Vienna



Austrian Health
Insurance Card



Austrian Railways



Mobile Debit Card



Austrian Post AG



EVVA mobile Lock
System



Austrian Automobile
Club

**and many
more**

RISE Experience



Airport Vienna



Austrian Health
Insurance Card



Austrian Railways



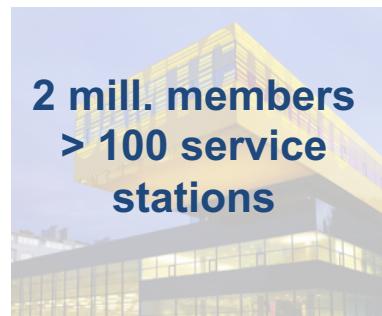
Mobile Debit Card



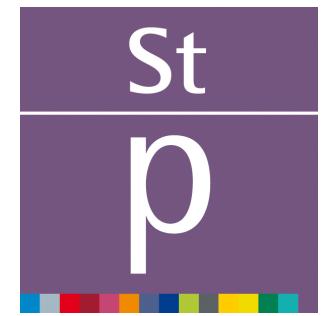
Austrian Post AG



EVVA mobile Lock
System

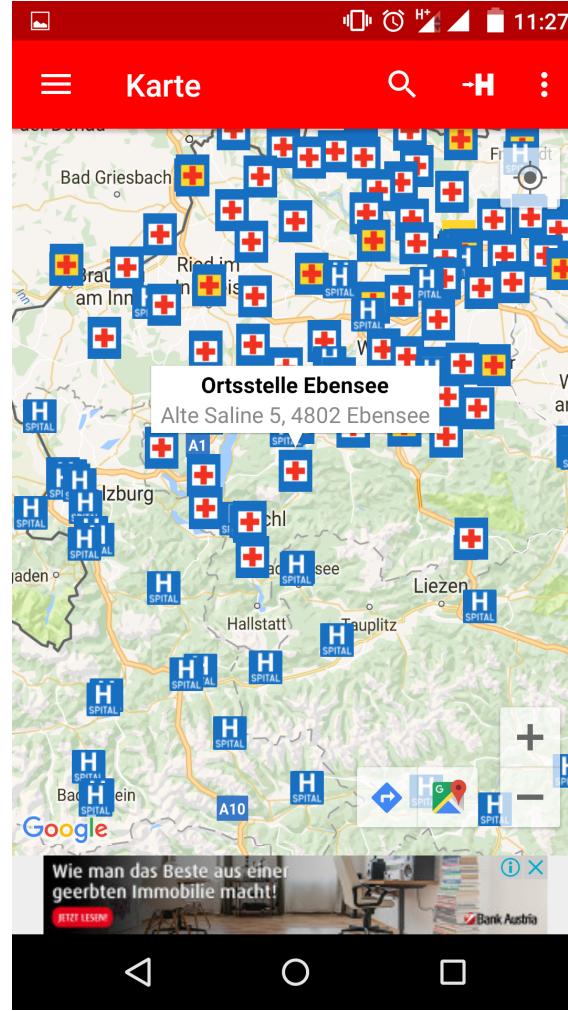
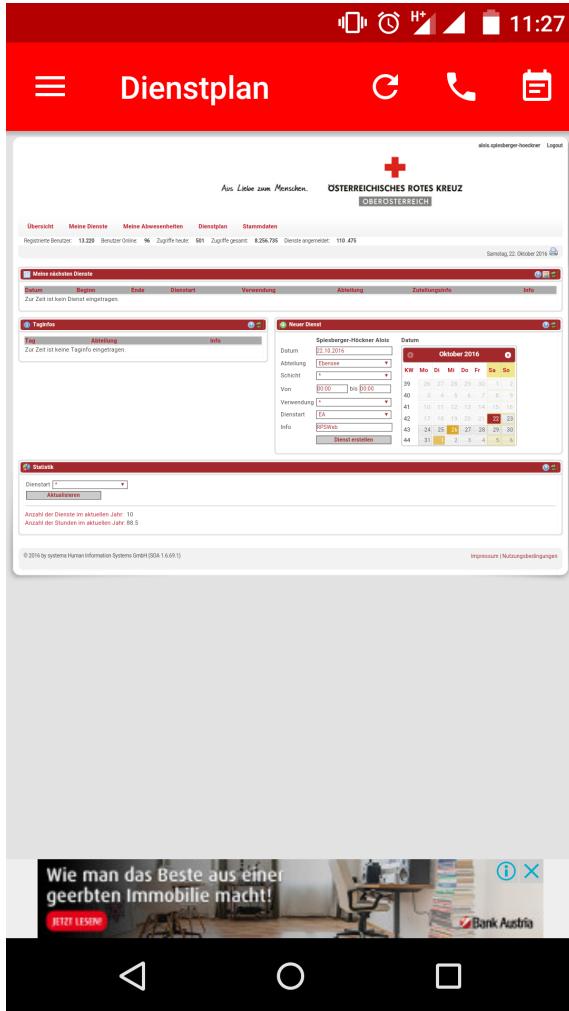


Austrian Automobile
Club

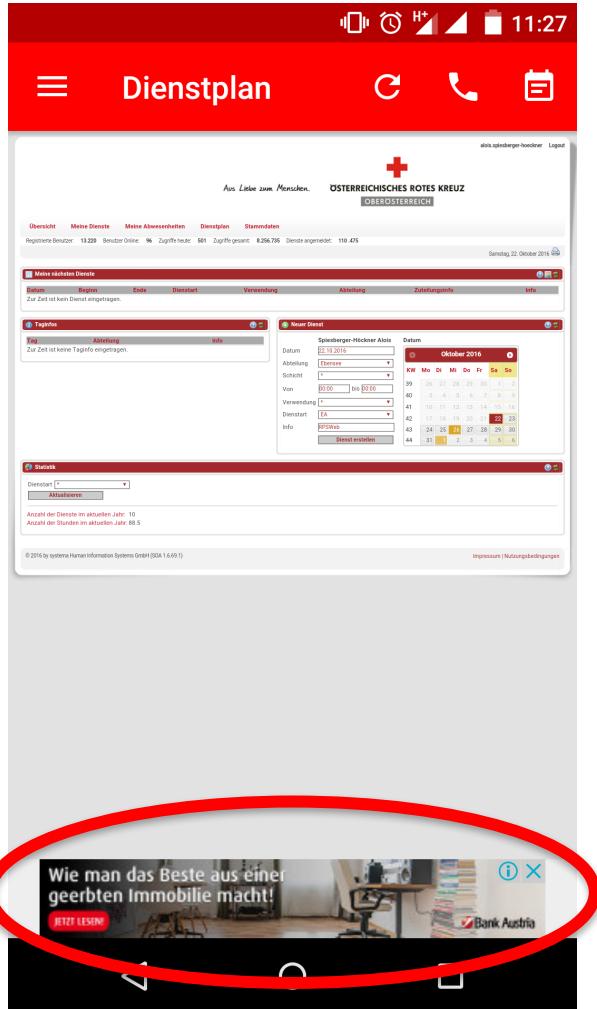


National Austrian
Award for
Consulting

2012 – App for the Red Cross



2012 – App for the Red Cross



~10000 downloads
~3000 active users
~800 daily launches

Monthly 50€ ~ INR 3600

o s



Os

- collective of independent app developers
- give people an understanding in social issues
- big data inspires us to think outside the box









Content

- Mobile Apps & Types
- Pattern and Building Blocks
- Automated Testing & Build Management
- Engineering in the Wild

Facts & Figures

at the end of 2014

- ~ 3 billion Internet users
- ~ 2.3 billion mobile-broadband subscriptions globally
- mobile-cellular subscriptions worldwide approaching the number of people on earth ~7 billion

(Brahima, 2014)

India

- India is the world's second largest mobile market.
- 412 million people are subscribing to mobile internet, a figure that is expected to grow to 668 million over the next four years.
- 320 million smartphone connections by the end of 2016
- More phones than toilet in India

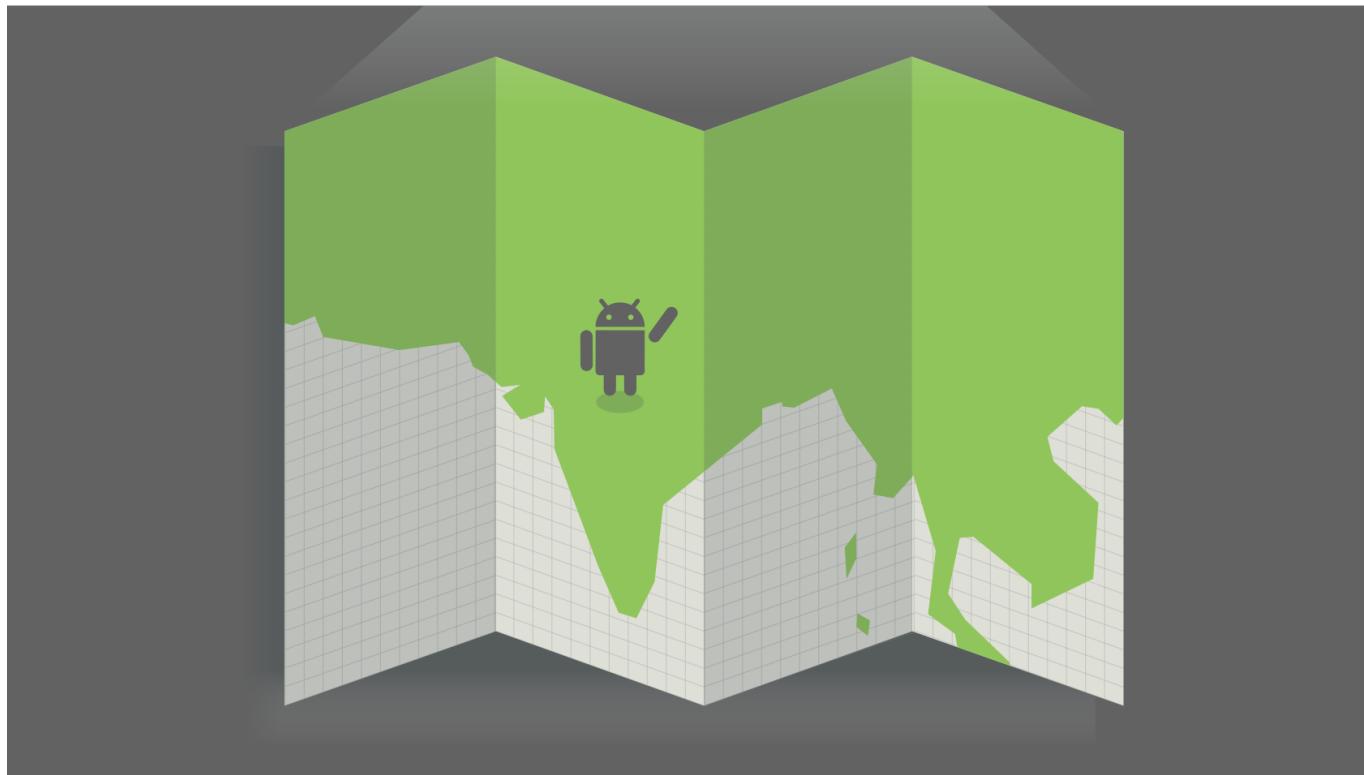
<http://www.un.org/apps/news/story.asp?NewsID=34369>

Connected Society – Telenor's mobile internet training projects in India: raising awareness of the benefits from getting online; GSMA

India & Android

- 97 % of the Indian smartphone market are Android smartphones
- The rest are iOS (~2%) and Windows Phone & others (~1%)

India & Google

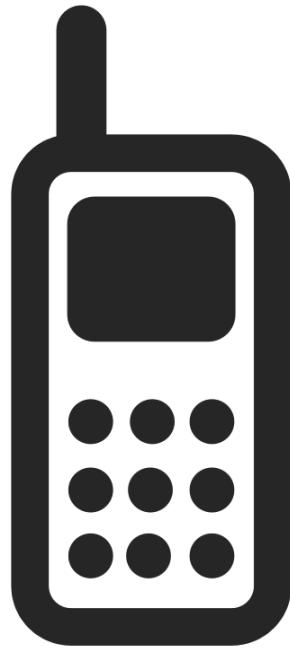


With its vision to help make India a global hub of high-quality mobile developers, Google is working on a Mobile Skilling program in India that aims to train **two million mobile developers**.

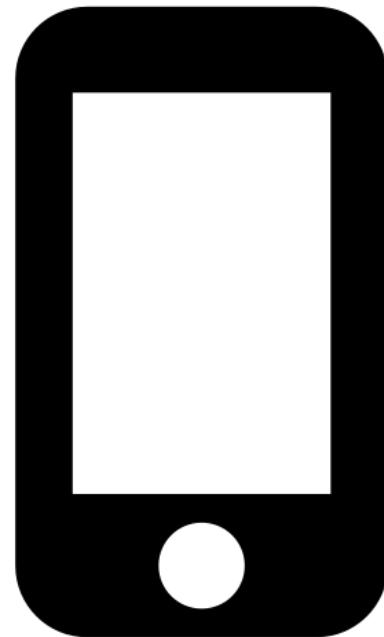
Content for Desktop & Mobile



Feature VS Smart

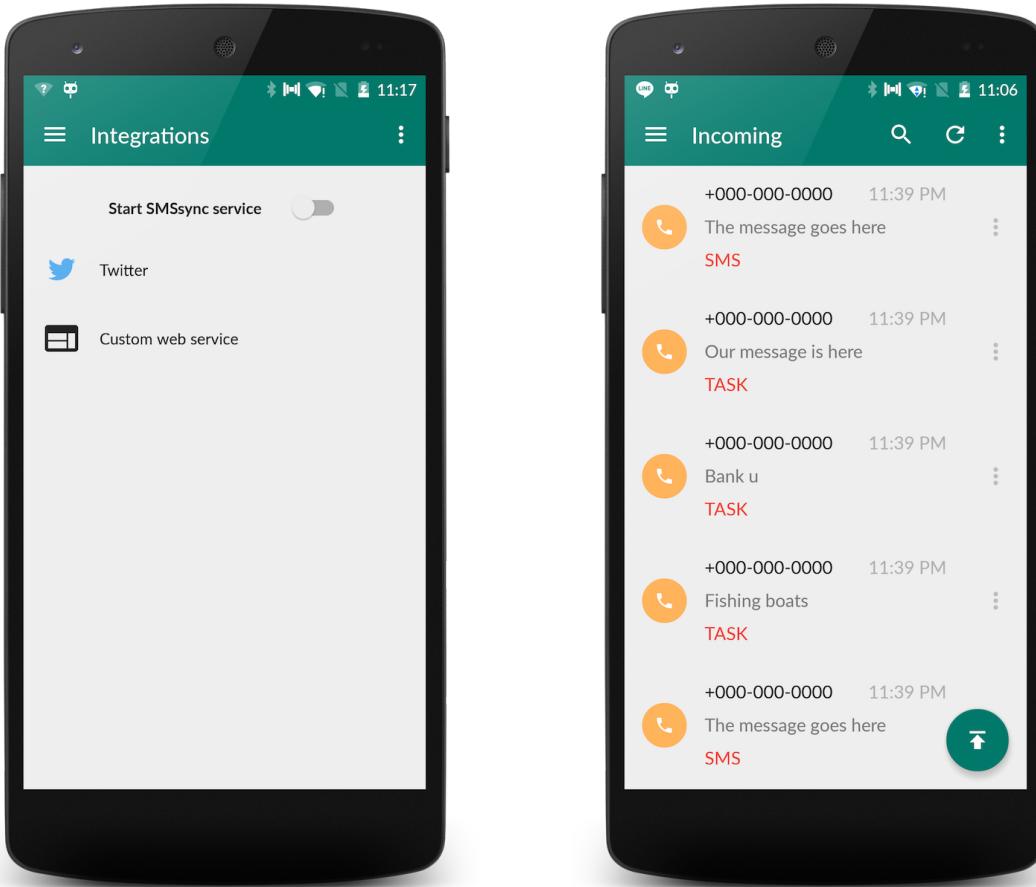


Calls & SMS
Limited Internet (WAP)
Limited processing power
Limited memory



Calls & SMS
HTML5 Browser + Internet
Good processing power
High memory
Capable of installing **Apps**

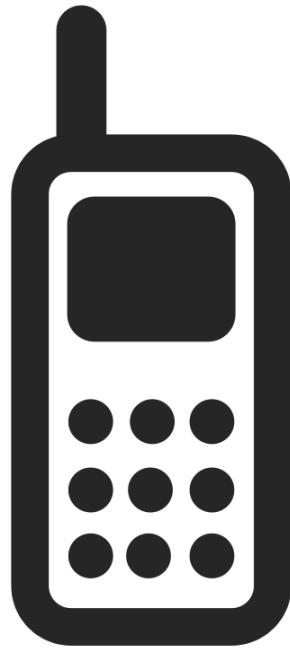
SMSSync



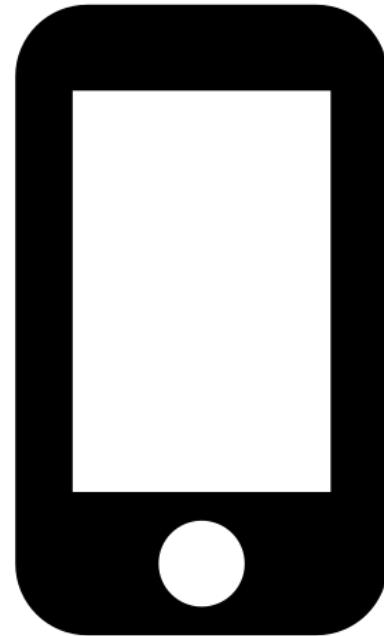
Receives SMS from Feature Phones, connects the request to the Internet and sends back a SMS to the feature phone.

<http://smssync.usahadi.com>

Feature VS Smart



Calls & SMS
Limited Internet (WAP)
Limited processing power
Limited memory



Calls & SMS
HTML5 Browser + Internet
Good processing power
High memory
Capable of installing **Apps**

2007 and the first iPhone



Challenges

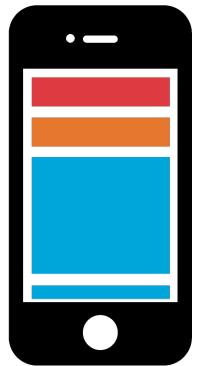
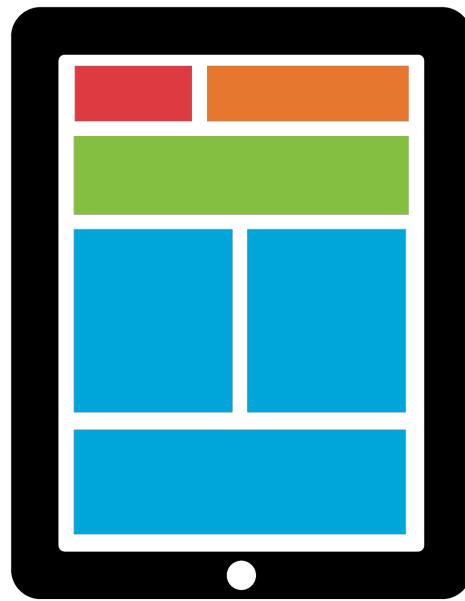
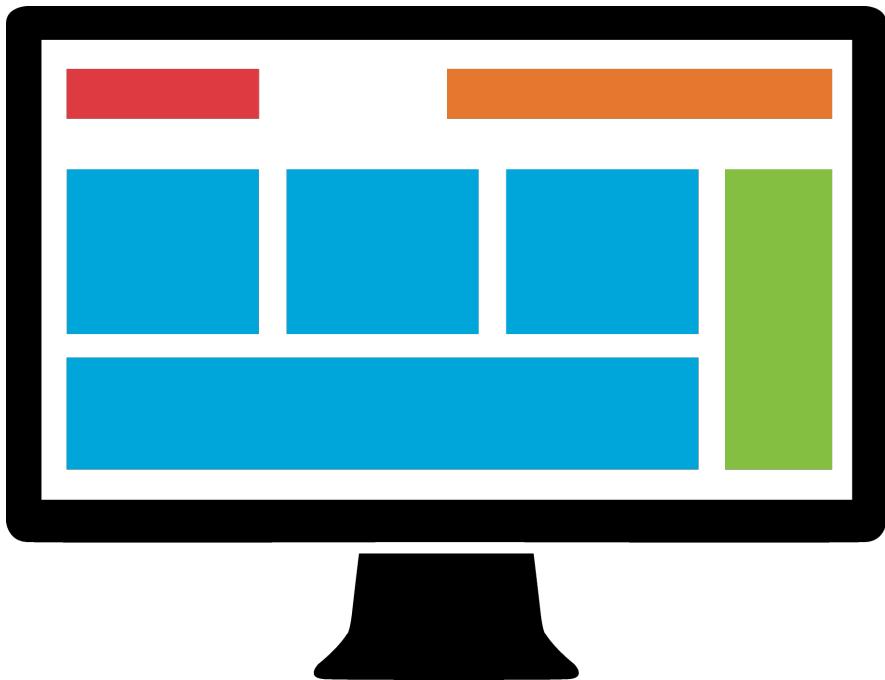
- Diverse technologies and platforms
- Various Design Guidelines/Approaches
- Different forms of website/software navigation
- Network communication and offline functionality
- Hardware, Sensors, APIs (Accelerator, GPS, NFC, Bluetooth, Push, Notifications, Wifi, SMS, ...)
- Connectivity (Internet of Things, Wearables, ...)
- Testing, Deployment & Monetization
- Security and Privacy

Survey - Mobile App Challenges

- Moving toward Fragmentation rather than Unification
- Monitoring, Analysis and Testing Support
- Open/Closed Development Platforms
- Data-Heavy Apps
- Keeping Up with Frequent Changes

(Joorabchi, Mesbah, & Kruchten, 2013)

From Desktop to Mobile



Mobile Operating Systems



Windows Phone



ubuntu



BlackBerry



Firefox OS

Mobile Operating Systems



Windows Phone



ubuntu



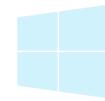
BlackBerry



Firefox OS



Mobile Operating Systems



Windows Phone



ubuntu



BlackBerry



Firefox OS



Mobile Operating Systems

		 iOS	
Market Share 16Q2	87.6%	11.7%	0.4%
Company	Google	Apple	Windows
OS Technology	Linux	Mac OS	Windows 10
IDE	Android Studio	Xcode	Visual Studio
Programming L.	Java	Swift, Objective C	C#, JS/HTML, C++
HTML5 Support	v5.0 - 452/555	v9.0 – 409/555	v10 – 401/555
Most used Version	5.0/5.1 – 34.1%	10 - 60%	8.1 - 81%

<https://www.idc.com/prodserv/smartphone-os-market-share.jsp>

<https://html5test.com/results/mobile.html>

<http://blog.adduplex.com/2016/03/adduplex-windows-phone-device-statistics-report-for-march-2016.html>

<https://developer.apple.com/support/app-store/>

<https://developer.android.com/about/dashboards/index.html>

Native / Web / Hybrid Apps

Native Platform Development

- One environment
- One design
- One official SDK

Advantages

- Access to all APIs and device specific features
- Performance
- User experience
- Official Support/Documentation
- Tools (IDE, Testing, ...)
- Debugging
- App store monetization

Disadvantages

- Costs
- Platform depended
- Slow development for multiple platforms
- Different programming languages & technologies

Native Platform Development

SDK & Design Guidelines & IDE



<https://developer.android.com/sdk/index.html>
<https://developer.android.com/design/index.html>



<https://developer.apple.com/xcode/downloads/>
<https://developer.apple.com/library/ios/documentation/UserExperience/Conceptual/MobileHIG/index.html>

Deployment Native

- Publish to specific app stores for every platform
- Advertising (Images, Videos, ...)
- Possibilities for Beta/Alpha testing
- Crash-Reports
- Monetization (Free/Paid/Subscription/In-App)
- Ratings and communication with users
- Updates are not mandatory
- Deployment for the user in a familiar environment

Web Development

- One technology
- Open web
- Independence

Advantages

- Instant updates
- Always up-to-date
- Independent from platforms
- “One code” for all
- Cover mobile and desktop
- Cost effective development
- No download/install required

Disadvantages

- Distribution
- Monetization
- Performance
- Permanent internet connection
- Many different Browsers
- APIs
- MouseOver vs. Selectors
- Data persistence

Web Development

Mobile Design Guidelines

<http://www.w3.org/standards/webdesign/mobilweb>

<https://developers.google.com/webmasters/mobile-sites/>

<https://dev.opera.com/articles/mobile-web-optimization-guide/>

<http://www.mobile-patterns.com/>

<https://developer.mozilla.org/en-US/Apps/Design>

Tools

<http://getbootstrap.com/>

<http://jquerymobile.com/>

<https://angularjs.org/>

<http://emberjs.com/>

<http://backbonejs.org/>

...

Deployment Web-App

- Publish web app to a server
- Instant updates (Thin Client)
- “Install” via Add to Homescreen

<https://w3c-webmob.github.io/installable-webapps/>



<https://developer.chrome.com/multidevice/android/installtohomescreen>

<https://support.mozilla.org/en-US/kb/how-add-shortcut-website-android>



<https://developer.apple.com/library/ios/documentation/AppleApplications/Reference/SafariWebContent/ConfiguringWebApplications/ConfiguringWebApplications.html>



<http://www.buildmypinnedsite.com/en>

Hybrid Platform Development

{ HTML5 }

NATIVE

{ 0010 }

MOBILE

<HTML5 />

{ }
HTML5
}

HYBRID

{ HTML5 }

Hybrid Platform Development

- Multiple platforms
- Native environment
- Advantages from native and web

Advantages

- One technology for many different platforms
- Distribution
- Monetization
- “One code” for all
- APIs
- Time & costs
- Advanced offline capabilities

Disadvantages

- Performance
- Testing
- Debugging
- No official SDK support
- Outdated WebViews
- User experience
- Design challenge

Hybrid Platform Development

- Cordova/PhoneGap **HTML5**
<https://cordova.apache.org/>
- Appcelerator **HTML5**
<http://www.appcelerator.com/>
- Xamarin **C#**
<http://xamarin.com/>
- Sencha Touch **HTML5**
<http://www.sencha.com/products/touch/>

Games

- Unity <http://unity3d.com/> **JS / C#**
- Marmalade <https://www.madewithmarmalade.com/> **C++**

Deployment Hybrid

- See Deployment Native

Monetization

<HTML5 />

- Keep 100% of your earnings with Web-Apps
- Handle money transactions & subscriptions on your own (with all opportunities and risks attached...)

{ 0010 } { HTML5 }

- Stores will keep ~30% of your profit
- Stores provide infrastructure for money transactions & subscriptions

Distinctions

NATIVE

{ 0010 }

One technology
for **one platform**

Native
environment

Predefined
Design Guidelines

Expensive and
slow for multi-
platform

Performance and
user experience

HYBRID

{ HTML5 }

Multiple
technologies **for all**

Semi-native
environment

Design Challenge

Cost effective and
fast

MOBILE

<HTML5 />

One technology
for all

Open web

Design Challenge

Cost effective
and fast

Open, hardware
restrictions

Native, Web or Hybrid?

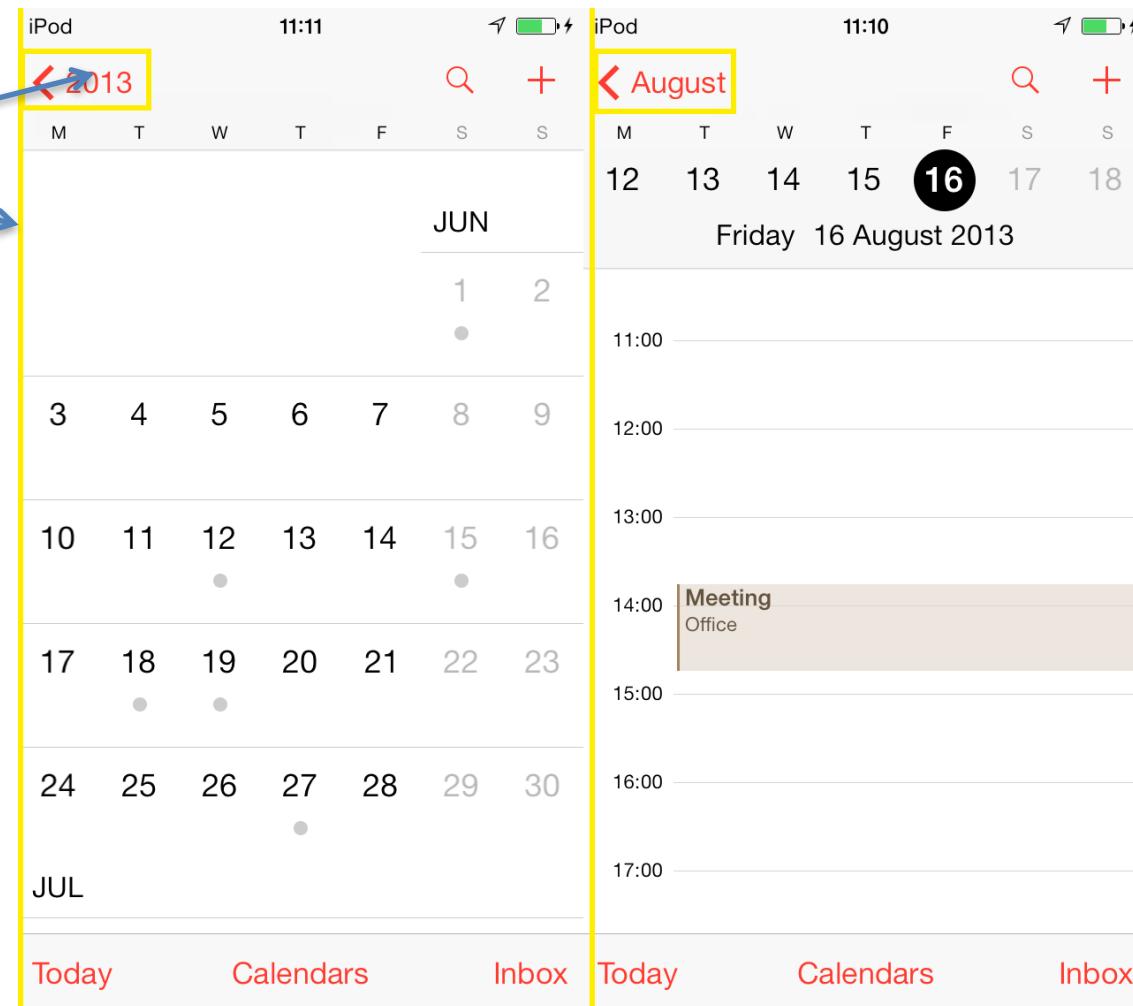
- There is no general answer
- Depends on
 - **Resources** - time, man power, know-how, costs, ...
 - **Requirements** - supported OSs, functionality, design
 - **Deployment & Monetization** – Possibilities and desired solution
 - ...

Design Challenge

Design Problem Example - Back



BACK



https://developer.apple.com/library/ios/documentation/UserExperience/Conceptual/MobileHIG/Navigation.html#/apple_ref/doc/uid/TP40006556-CH53-SW1

Design Problem Example - Back

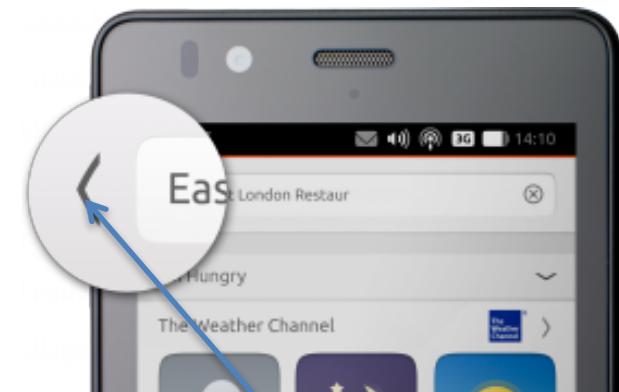
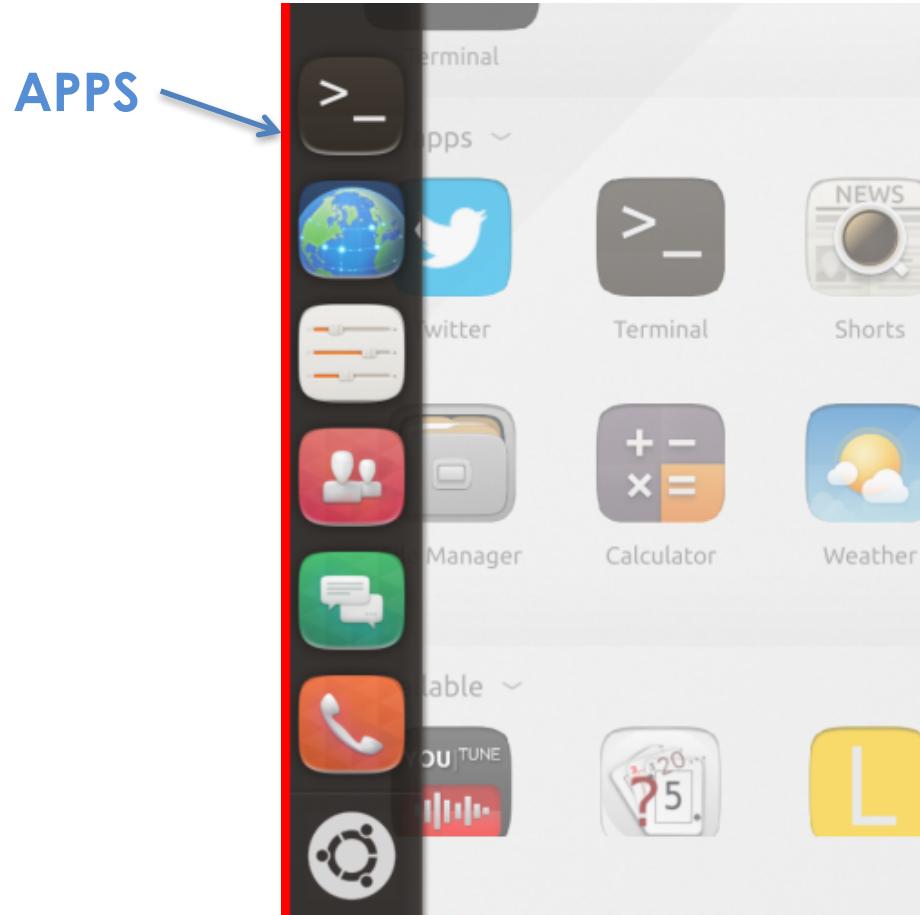


BACK



https://developer.blackberry.com/design/bb10/application_structure.html

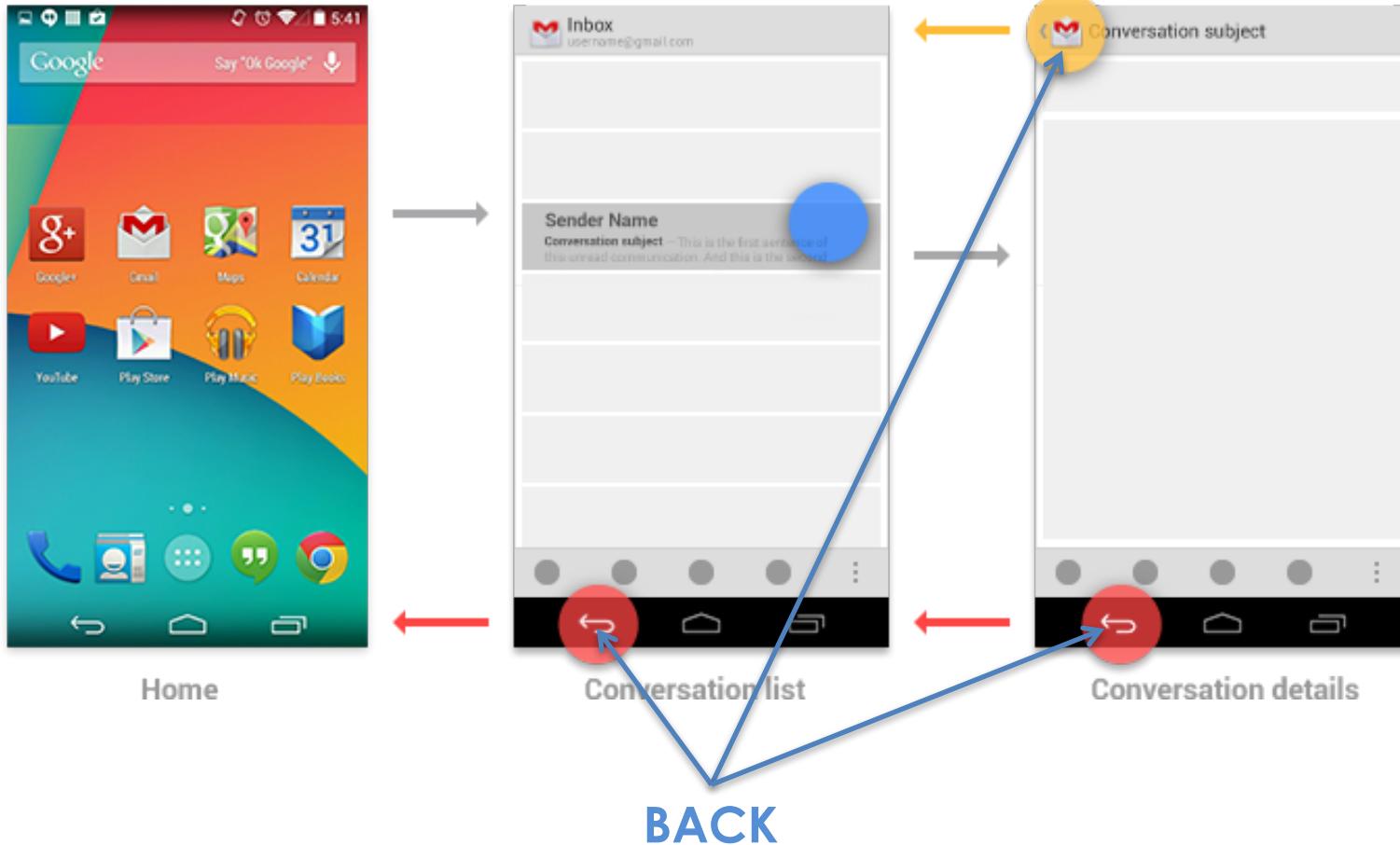
Design Problem Example - Back



BACK

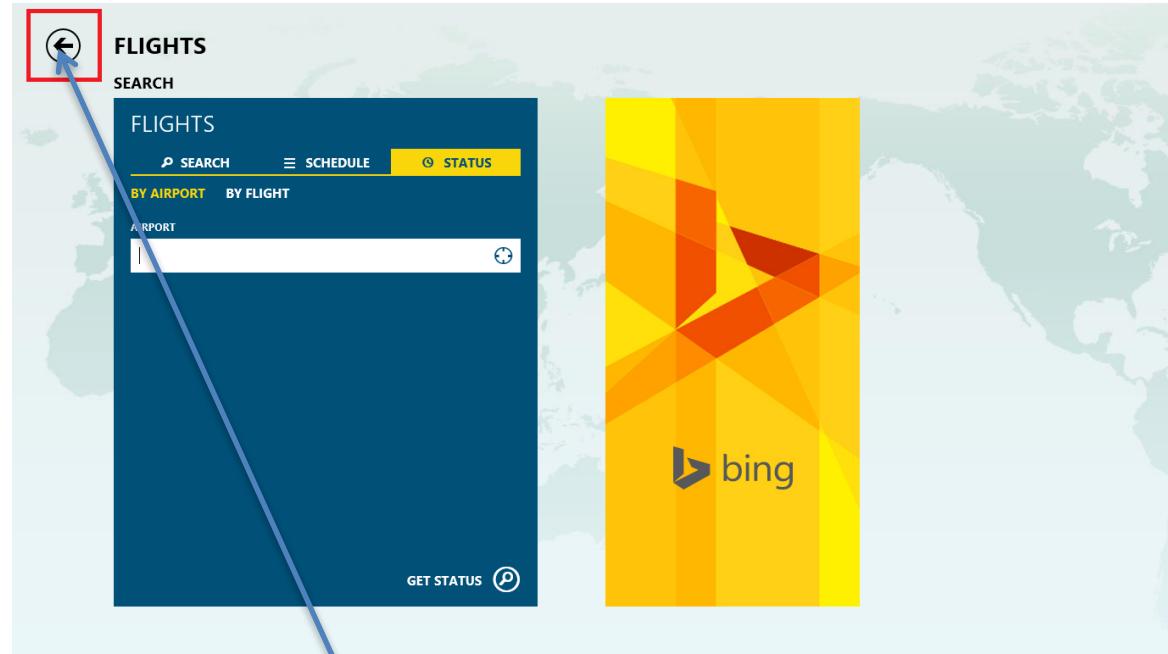
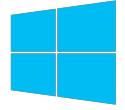
<https://design.ubuntu.com/apps/building-blocks/header>

Design Problem Example - Back



<http://developer.android.com/design/patterns/navigation.html>

Design Problem Example - Back



<https://msdn.microsoft.com/en-us/library/windows/apps/dn596119.aspx>

OS Navigation Links

- Android:
<http://developer.android.com/design/patterns/navigation.html>
- iOS:
https://developer.apple.com/library/ios/documentation/UserExperience/Conceptual/MobileHIG/Navigation.html#/apple_ref/doc/uid/TP40006556-CH53-SW1
- Windows Phone:
<https://msdn.microsoft.com/en-us/library/windows/apps/dn958438.aspx>
- Blackberry:
https://developer.blackberry.com/devzone/design/bb10/application_structure.html
- Firefox OS:
https://developer.mozilla.org/en-US/Apps/Design/UI_layout_basics
- Ubuntu Touch:
<https://design.ubuntu.com/apps/patterns/navigation>
- Sailfish OS:
<https://sailfishos.org/design/gestures/>

... and there is more

Dialogs



Permissions

This app determines your phone's location and shares it with Google in order to serve personalized alerts to you. This allows for a better overall app experience.

DECLINE ACCEPT



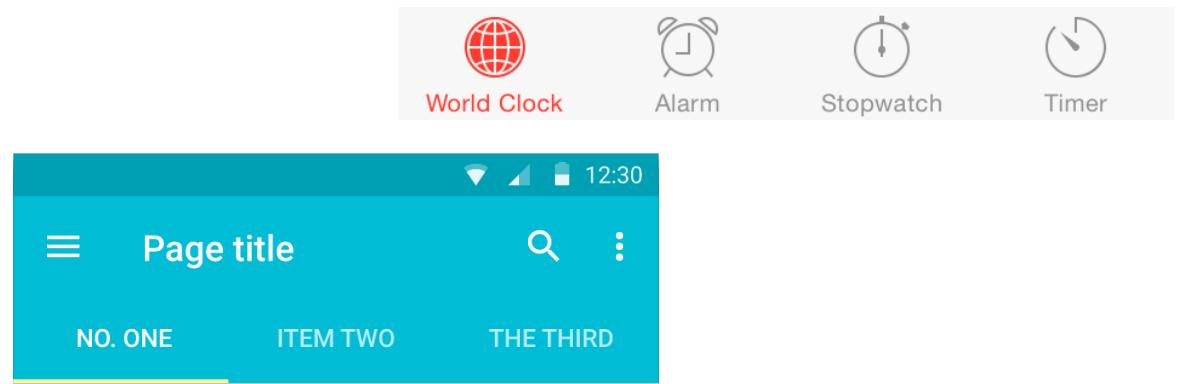
Touch ID for “Goal Length”

Authenticate to access Goal Length

Enter Passcode

Cancel

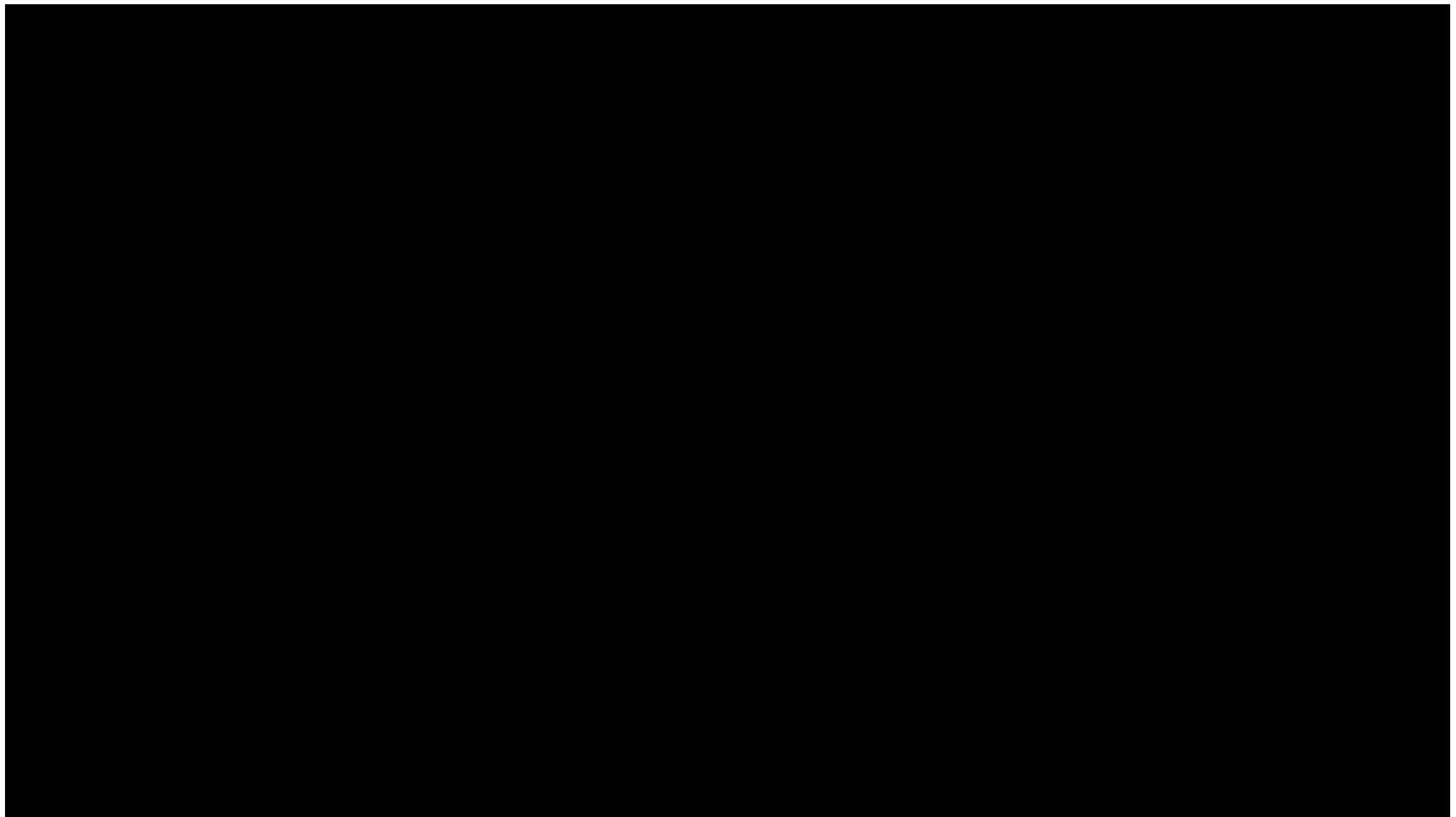
Tabs



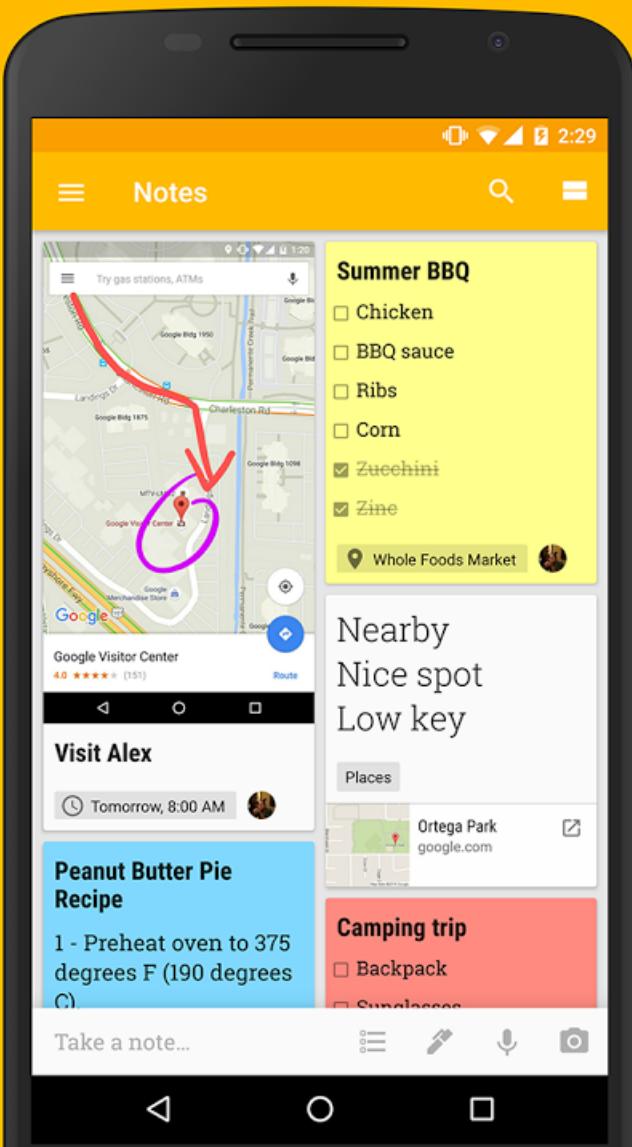
How to design a User Interface that fits all needs and works on all platforms?

Material Design

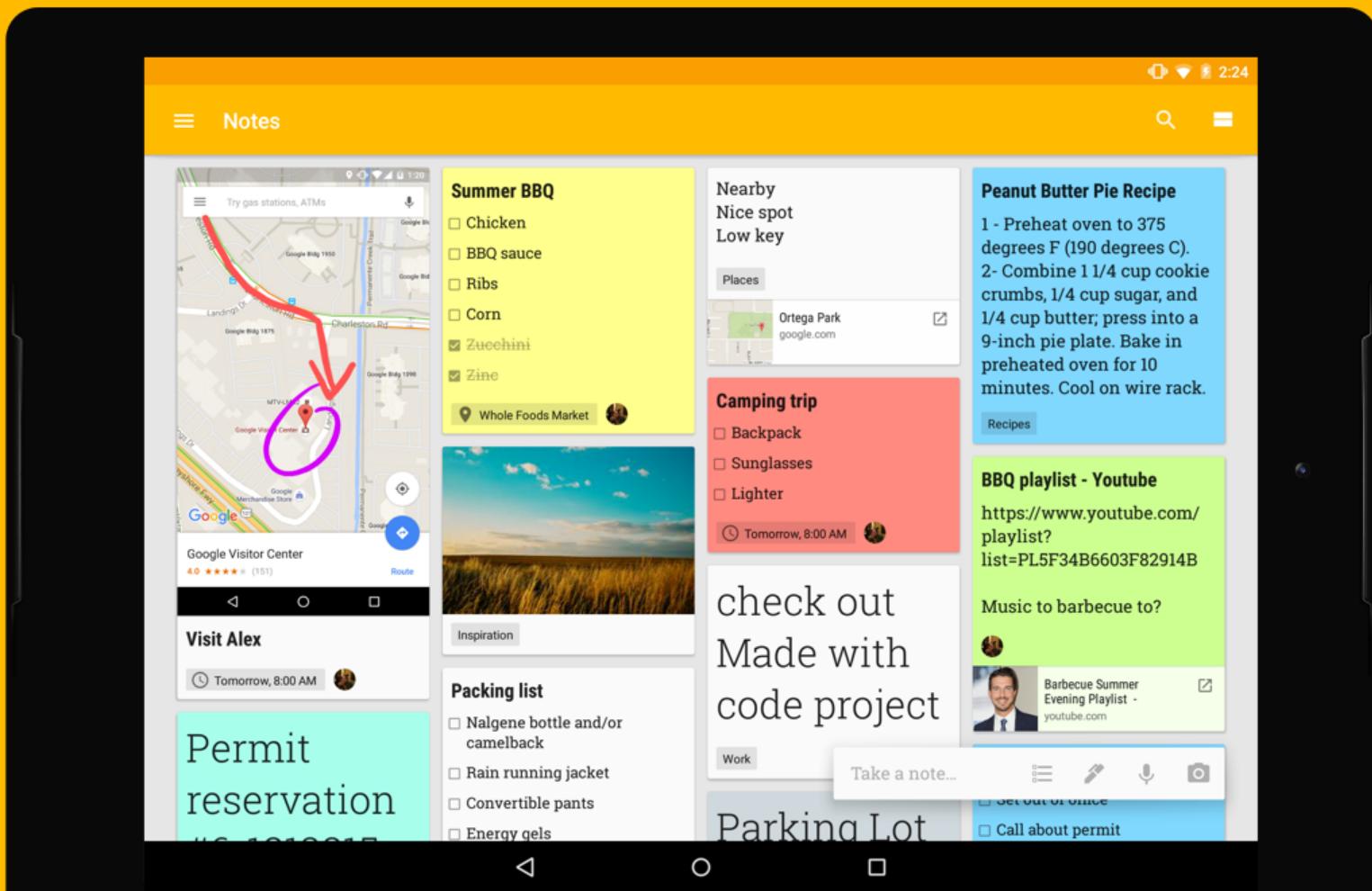
Intro To Material Design



Capture your thoughts in any format



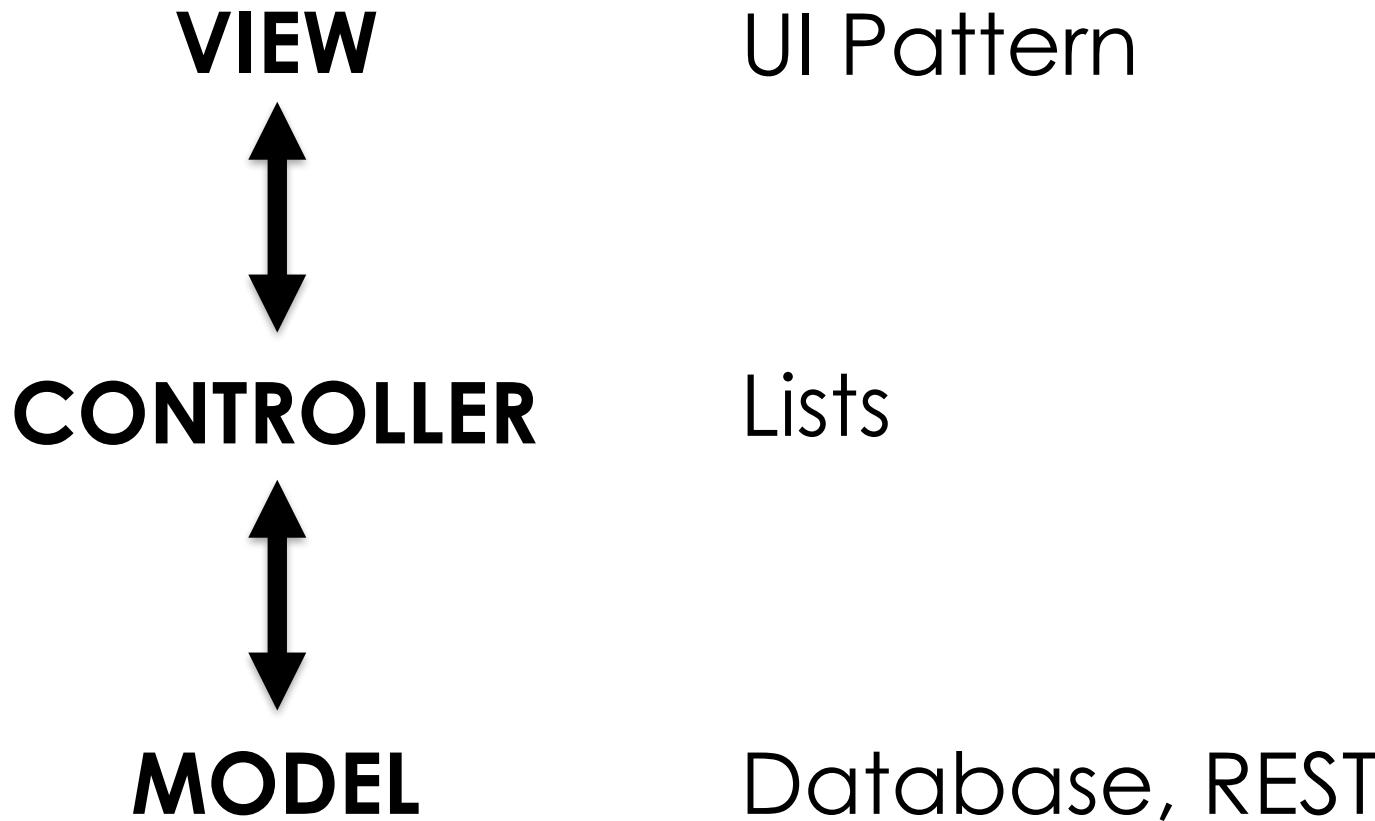
Capture your thoughts in any format



Showcase

Pattern and Building Blocks

MVC



UI Pattern

How do I present information
on a mobile screen?

With



<HTML5 />

{ HTML5 }

**Mobile
VIEW**

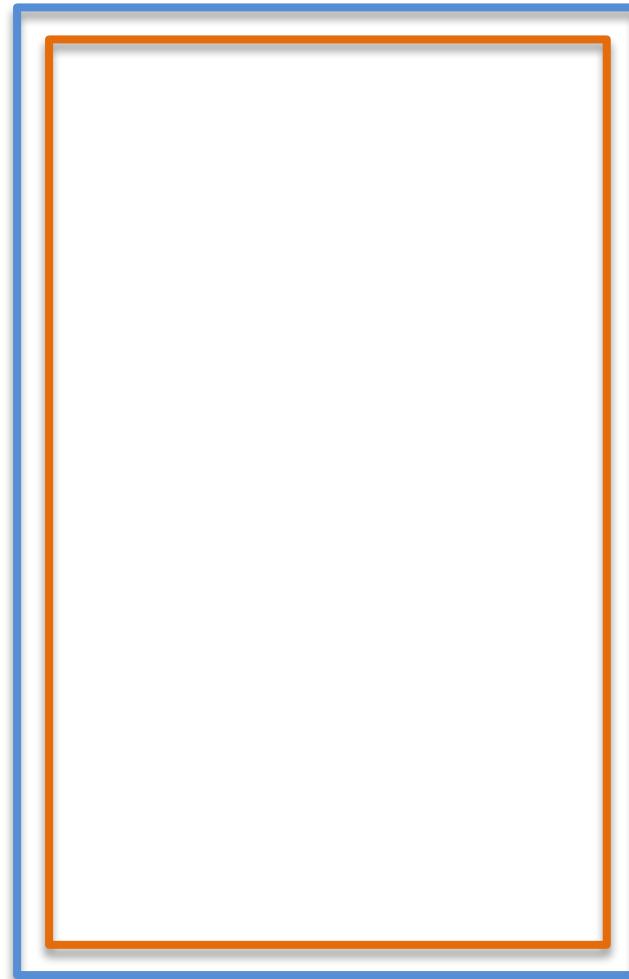


Activities and Fragments



Activity

can be seen as one Screen.
One Activity can have several
Fragments or views.

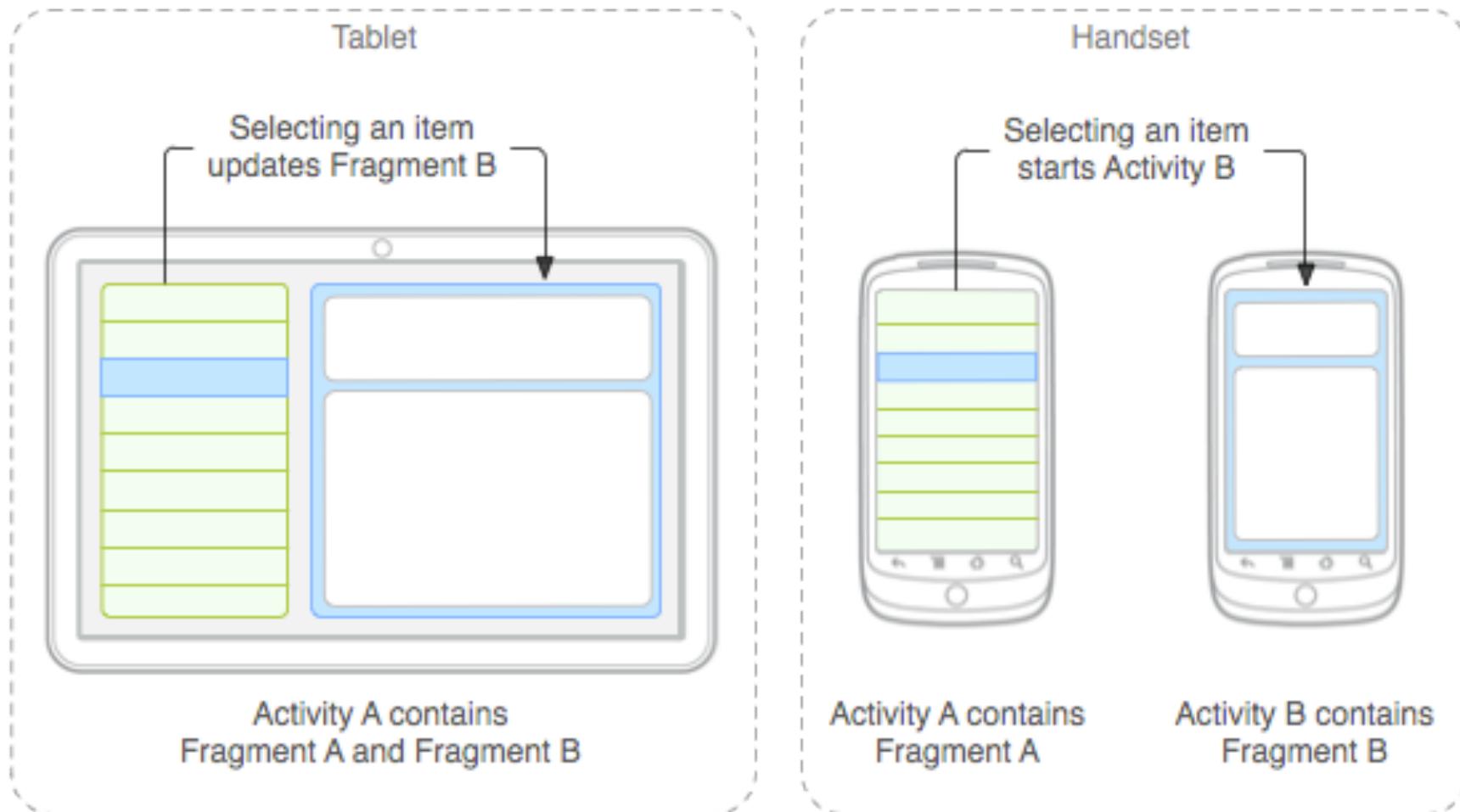


Fragment

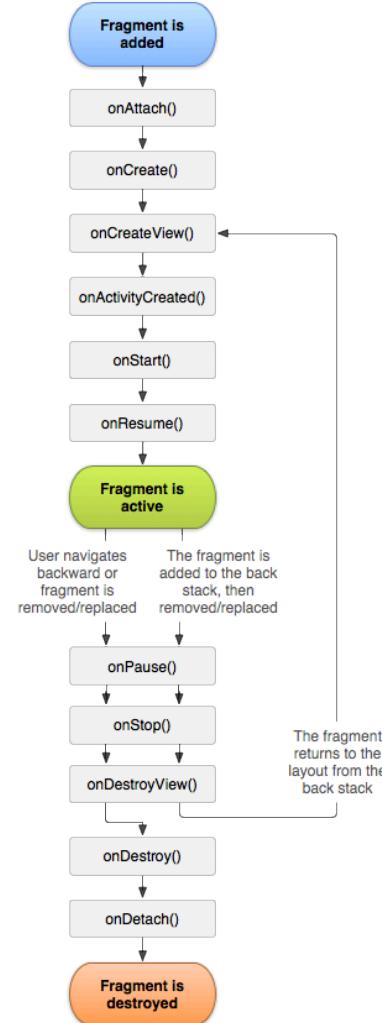
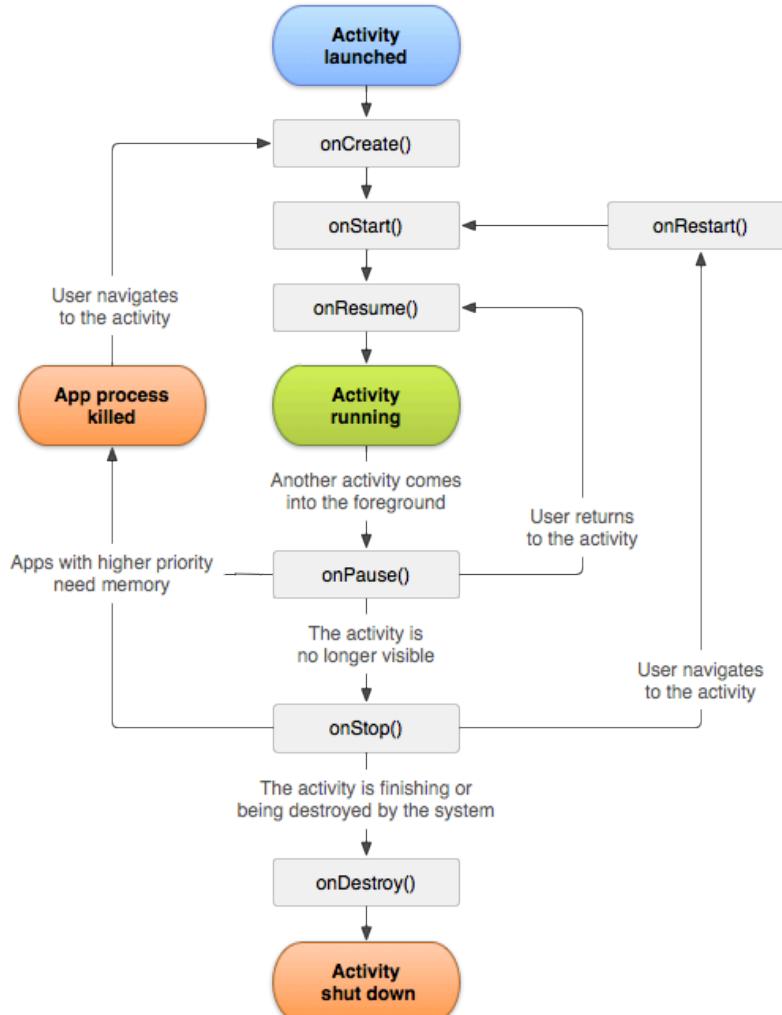
is a dynamic view container
and always has one Activity.

Event-driven programming

Activities and Fragments



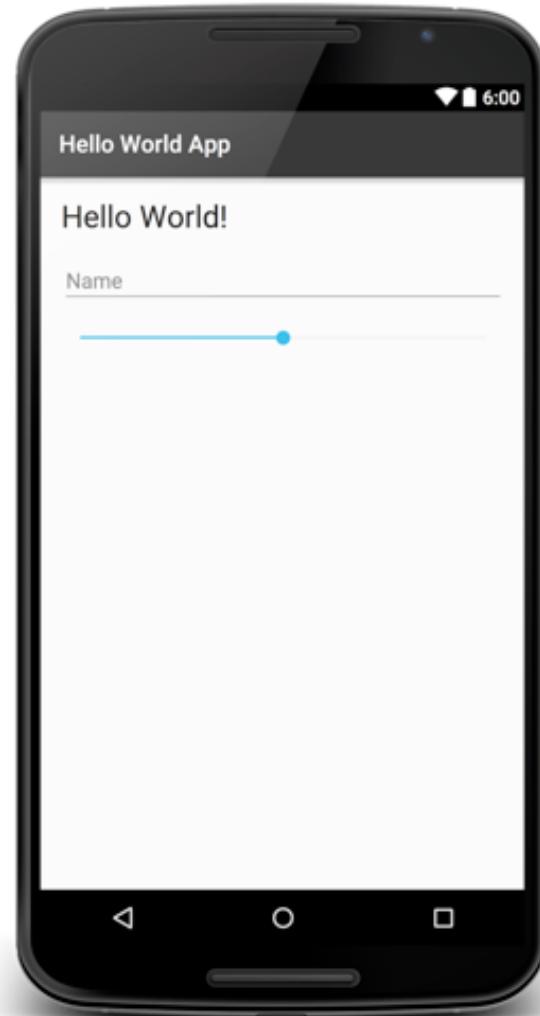
Lifecycles



Create Views



```
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout
3     xmlns:android="http://schemas.android.com/apk/res/android"
4     android:layout_width="match_parent"
5     android:layout_height="match_parent"
6     android:padding="18dp"
7     android:orientation="vertical">
8
9     <TextView
10        android:layout_width="match_parent"
11        android:layout_height="wrap_content"
12        android:text="Hello World!"
13        android:textSize="26sp"/>
14
15    <EditText
16        android:id="@+id/edit_text_id"
17        android:layout_width="match_parent"
18        android:layout_height="wrap_content"
19        android:hint="Name"
20        android:layout_marginTop="18dp"/>
21
22    <SeekBar
23        android:layout_width="match_parent"
24        android:layout_height="wrap_content"
25        android:layout_marginTop="18dp"
26        android:progress="50"/>
27
28 </LinearLayout>
```



Working with views



XML is **inflated** into an **Activity or Fragment**.

There is a Java class pendant to every XML UI element. Views are addressed and manipulated via an ID.

```
1 <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout
3      xmlns:android="http://schemas.android.com/apk/res/android"
4      android:layout_width="match_parent"
5      android:layout_height="match_parent"
6      android:padding="18dp"
7      android:orientation="vertical">
8
9      <TextView
10         android:layout_width="match_parent"
11         android:layout_height="wrap_content"
12         android:text="Hello World!"
13         android:textSize="26sp"/>
14
15      <EditText
16          android:id="@+id/edit_text_id"
17          android:layout_width="match_parent"
18          android:layout_height="wrap_content"
19          android:hint="Name"
20          android:layout_marginTop="18dp"/>
21
22      <SeekBar
23          android:layout_width="match_parent"
24          android:layout_height="wrap_content"
25          android:layout_marginTop="18dp"
26          android:progress="50"/>
27
28  </LinearLayout>
```

Data Binding

<https://developer.android.com/tools/data-binding/guide.html>



View Controller



View Controllers

can be seen as one screen and handle UI elements. The controller uses a Storyboard/XIB layout and handles the state changes of an iOS application.

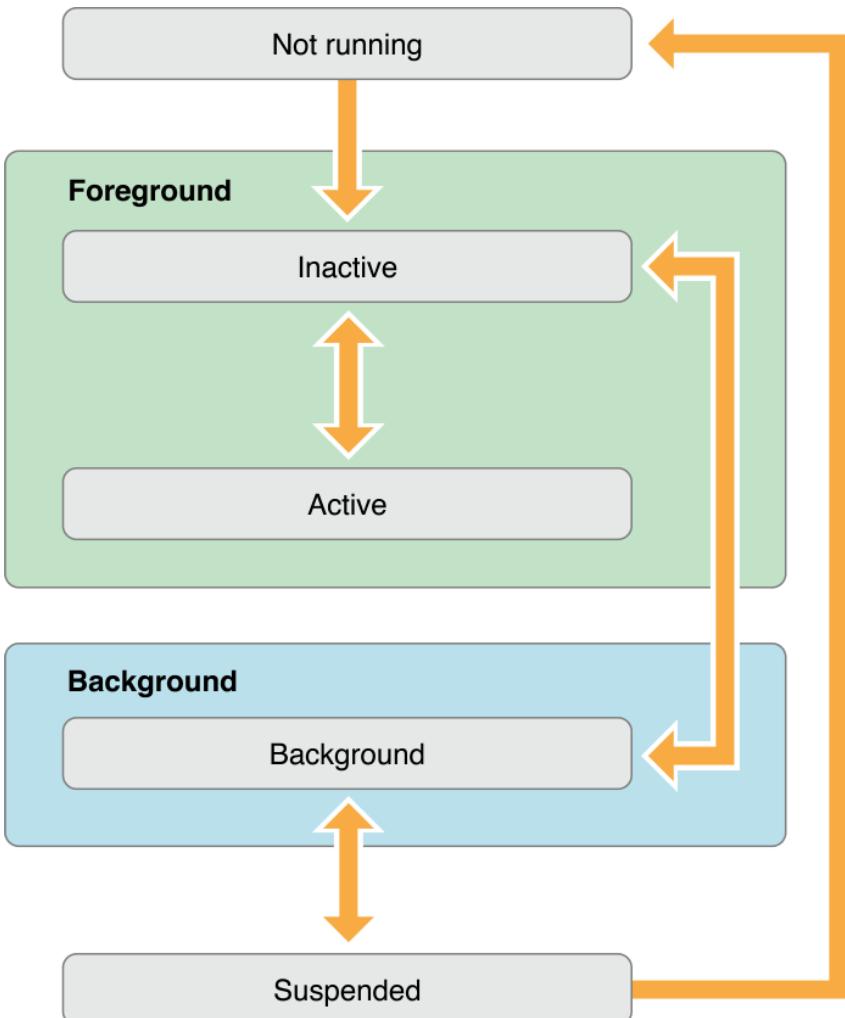


Event-driven programming

State Changes



Every iOS application runs through the same states like **viewDidLoad()** or **viewDidDisappear()**.

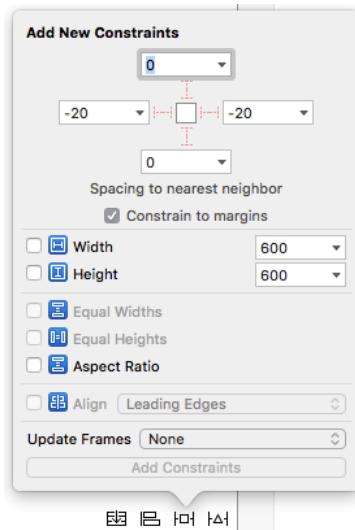


Create UI Elements



WYSIWYG Editor in two forms - **Storyboard** or **XIB-File**.

Drag and Drop with absolute Pixel or **Auto Layout**

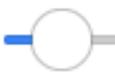


Label **Label** - A variably sized amount of static text.

Button **Button** - Intercepts touch events and sends an action message to a target object when it's tapped.

 **Segmented Control** - Displays multiple segments, each of which functions as a discrete button.

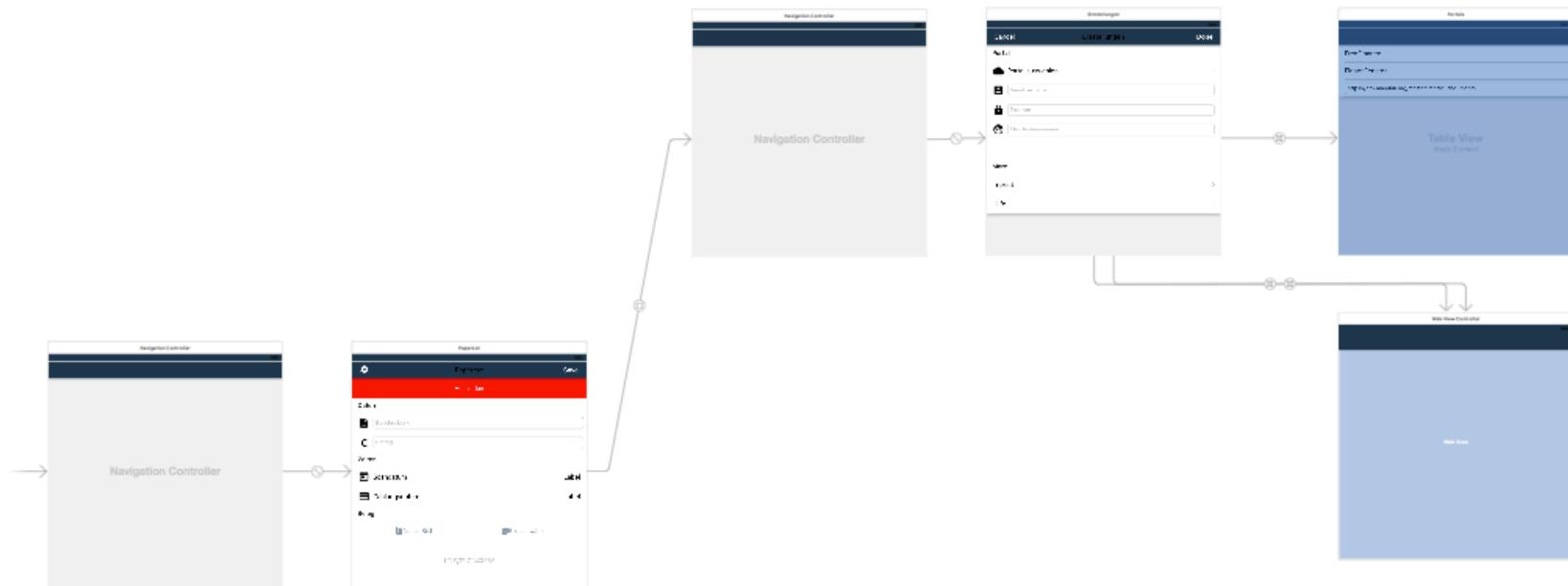
 **Text Field** - Displays editable text and sends an action message to a target object when Return is tapped.

 **Slider** - Displays a continuous range of values and allows the selection of a single value.

 **Switch** - Displays an element showing the boolean state of a value. Allows tapping the control to toggle...

 **Activity Indicator View** - Provides feedback on the progress of a task or process of unknown duration.

Storyboard



Working with UI Elements



```
//  
import UIKit  
import Foundation  
  
class ViewController: UIViewController {  
    @IBOutlet var navigationBar: UINavigationItem!  
    @IBOutlet var homeView: UIView!  
  
    override func viewDidLoad() {  
        super.viewDidLoad()  
        // Do any additional setup after loading the view, typically from  
        // a nib.  
    }  
  
    override func didReceiveMemoryWarning() {  
        super.didReceiveMemoryWarning()  
        // Dispose of any resources that can be recreated.  
    }  
}
```

<HTML5 />

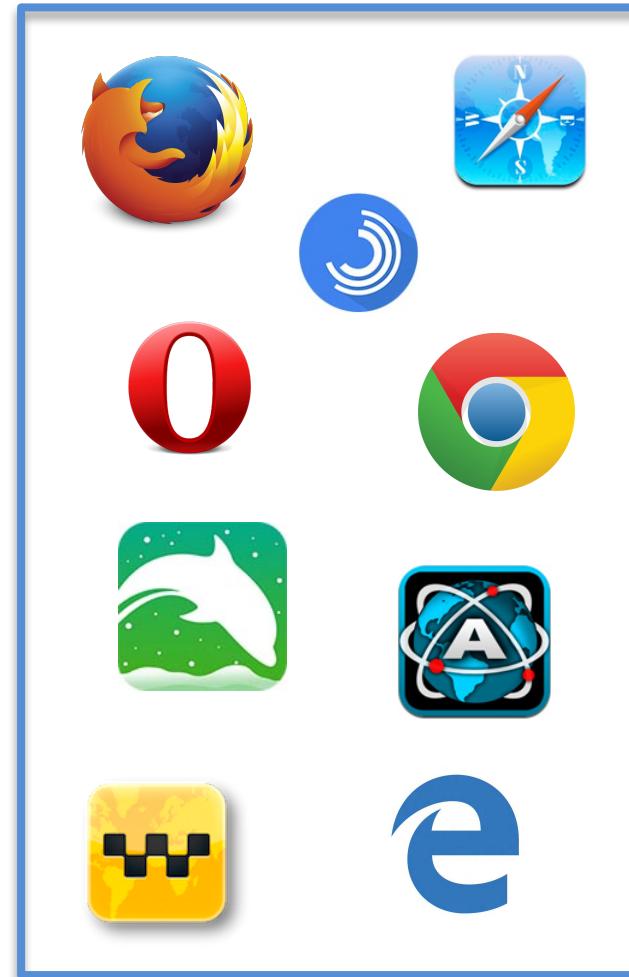
Mobile Browser

Great variety of **mobile browsers** which use different technologies to render HTML.

HTML Structure

CSS Styling

JS Functionality



HTML5 Components

HTML5 Input - not supported yet by all browser:

button, checkbox, colour, date, datetime, datetime-local, email, file, hidden, image, month, number, password, radio, range, reset, search, submit, tel, text, time, url, week

Also third party UI Elements and Frameworks **like**:

JQuery Mobile - <https://jquerymobile.com/>

Bootstrap - <https://getbootstrap.com/>

...

Web App

No standard for installing web apps, but there is:

- **Standalone switch** – “Do not start app in Browser”
 - iOS - <meta name="apple-mobile-web-app-capable" content="yes">
 - Chrome - <meta name="mobile-web-app-capable" content="yes">
- **Name** – The name of the application
 - HTML5 - <meta name="application-name" content="Awesome App!">
 - iOS - <meta name="apple-mobile-web-app-title" content="Awesome App!">
- **Icons** – Defines the Icon
 - HTML5 - <link rel="icon" sizes="196x196" href="icon">
 - iOS - <link rel="apple-touch-icon" href="apple-touch-icon.png">
- And more – Permissions, Start Page, Orientation and Navigation

Offline Content in HTML5

<HTML5 />

Why?

- Always-On is just a dream
- There is no perfect signal coverage
- Speed and resources play a big role

Use device Cache:

- Check for Browser support
- Include a manifest
`<html manifest="example.appcache">`
- Define files to cache, fallbacks and online resources

```
CACHE MANIFEST
# 2010-06-18:v2

# Explicitly cached 'master entries'.
CACHE:
/favicon.ico
index.html
stylesheet.css
images/logo.png
scripts/main.js

# Resources that require the user to be online.
NETWORK:
*

# static.html will be served if main.py is inaccessible
# offline.jpg will be served in place of all images in images/large/
# offline.html will be served in place of all other .html files
FALLBACK:
/main.py /static.html
images/large/ images/offline.jpg
```

HTML5 Offline Support

IE	Edge *	Firefox	Chrome	Safari	Opera	iOS Safari *	Opera Mini *	Android Browser	Chrome for Android
8			45					4.3	
9		43	46					4.4	
10		44	47					4.4.4	
11	13	45	48	9				47	49
	14		49	9.1	35	9.2	8		
			46	50		36	9.3		
			47	51		37			
			48	52					

<http://caniuse.com/#search=cache>

{ HTML5 }

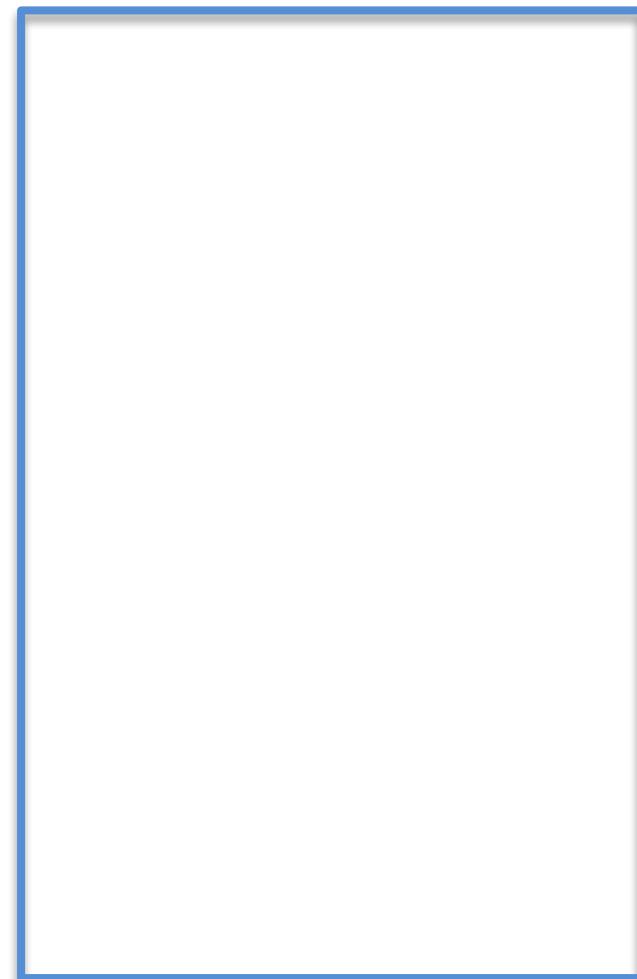
Webview

OS [Webview](#) is used to render local saved HTML content.

HTML Structure

CSS Styling

JS Functionality



Hybrid UI Elements

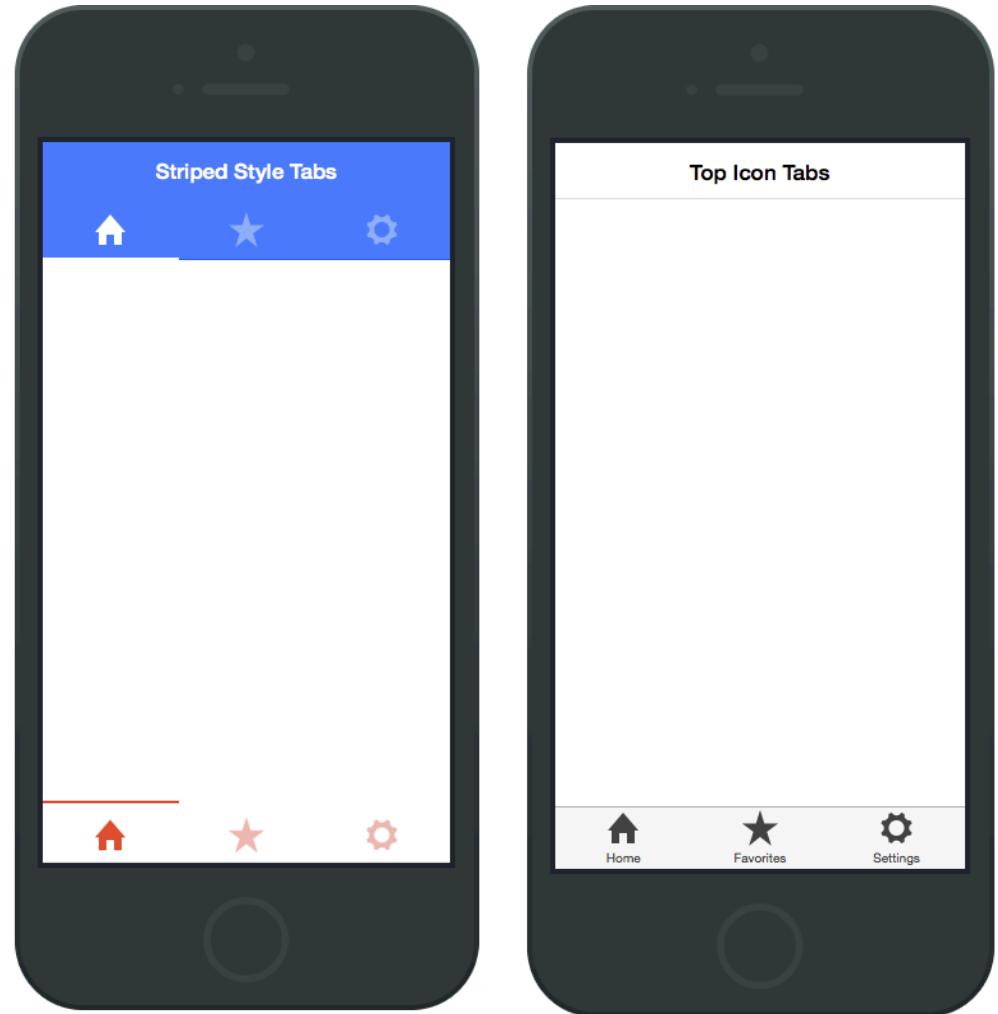
- Same as Web Apps and
- Hybrid Frameworks that try to emulate, expand or mimic native OS designs
 - Ionic - <http://ionicframework.com/>
 - Onsen - <http://onsen.io>
- Partly dynamically implement Android or iOS Design Guidelines

Ionic

„In many ways iOS and Ionic inadvertently share the same basic principles: simplicity, reduction of clutter, use of white and solid colors, zero gradients, and hardly any box-shadows or border-radii altercations. Anything more than that, we leave up to you.“

<http://blog.ionic.io/rise-above-the-platform/>

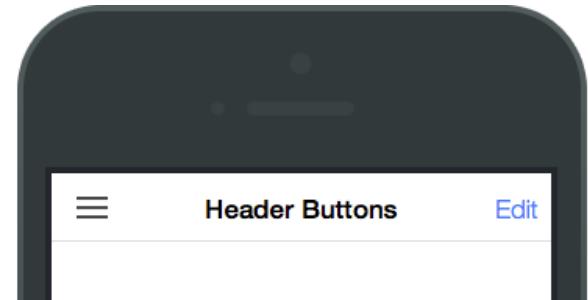
Ionic 2 is coming!



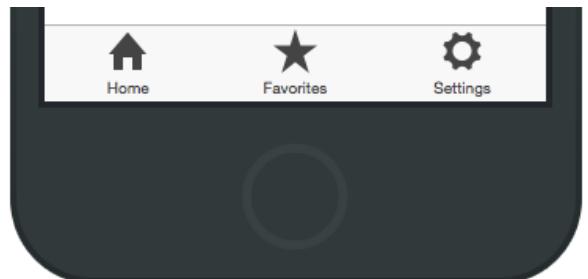
Ionic

<https://creator.ionic.io/>

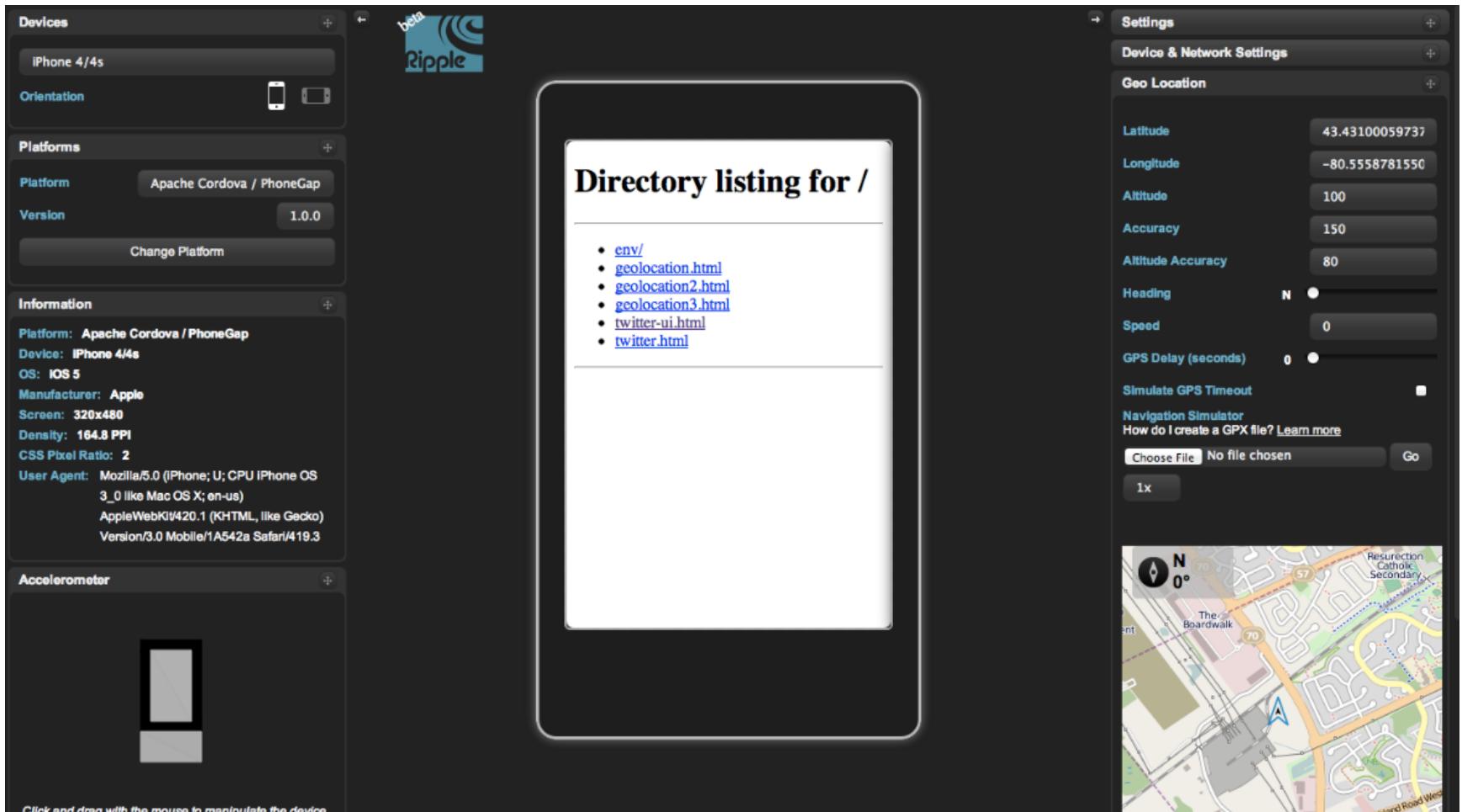
```
<div class="bar bar-header">
  <button class="button button-icon icon ion-navicon"></button>
  <div class="h1 title">Header Buttons</div>
  <button class="button button-clear button-positive">Edit</button>
</div>
```



```
<div class="tabs tabs-icon-top">
  <a class="tab-item">
    <i class="icon ion-home"></i>
    Home
  </a>
  <a class="tab-item">
    <i class="icon ion-star"></i>
    Favorites
  </a>
  <a class="tab-item">
    <i class="icon ion-gear-a"></i>
    Settings
  </a>
</div>
```



Ripple Project



Lists

How do I present a big amount of data on a mobile device?

With

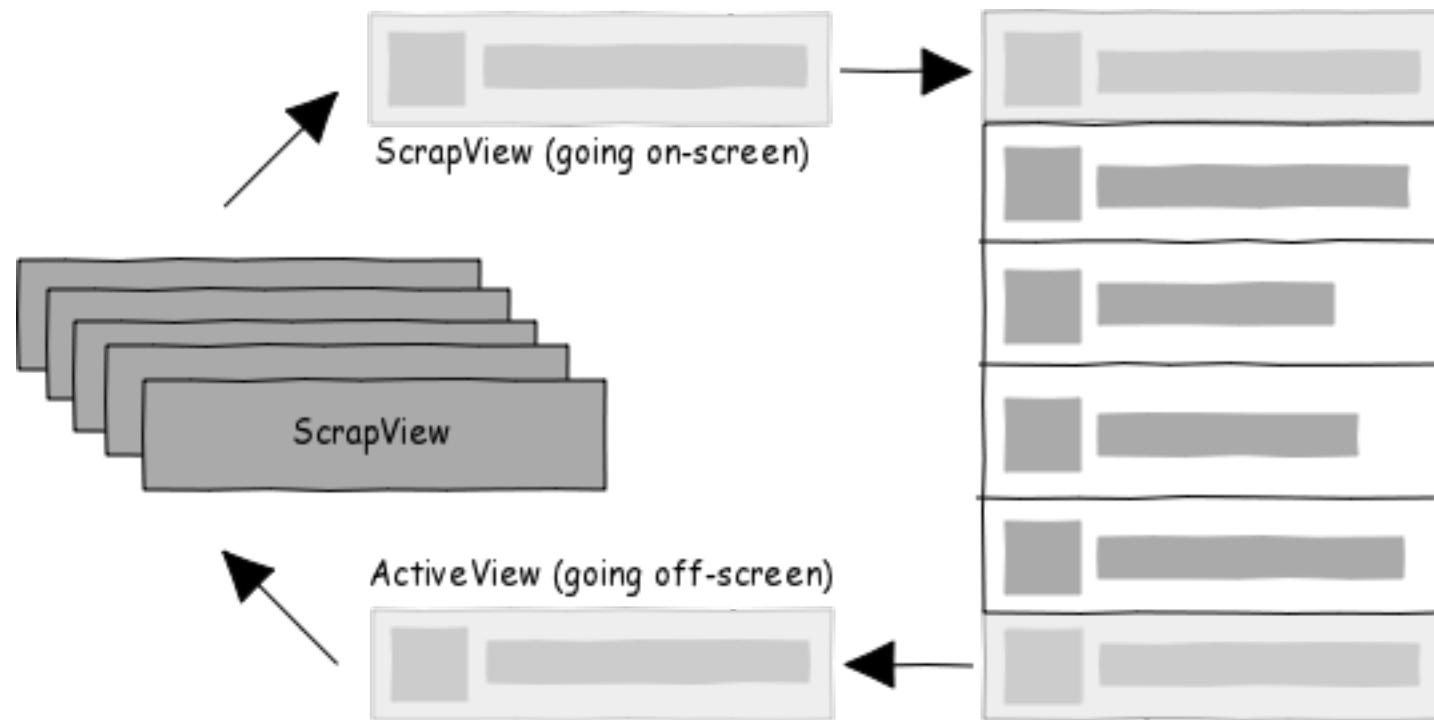


<HTML5 />

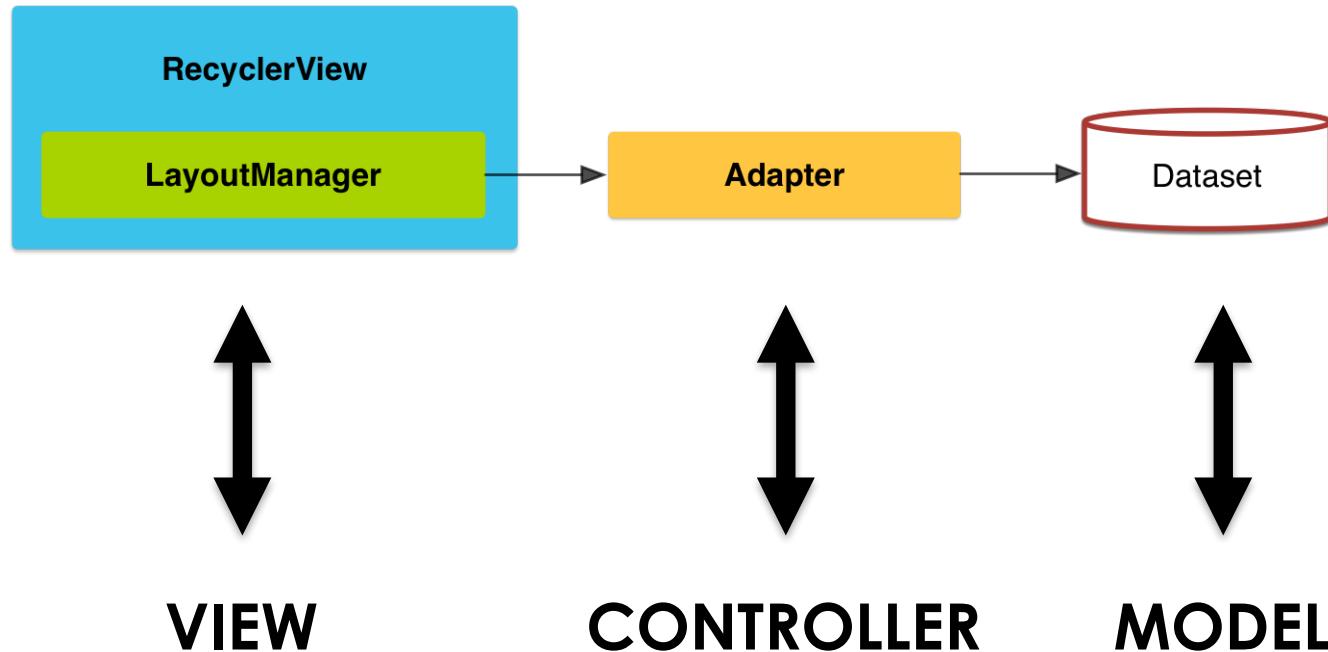
{ HTML5 }



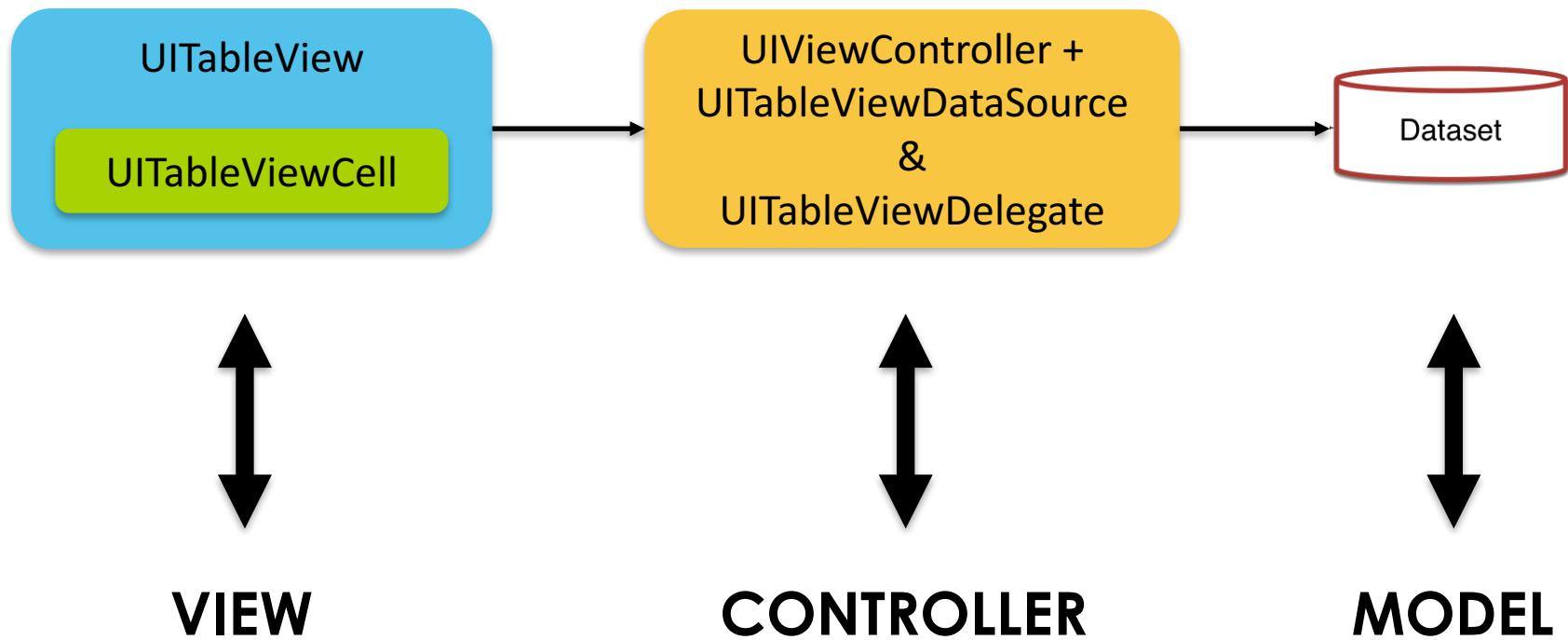
View Recycle Pattern



Recycler View



UI Table View



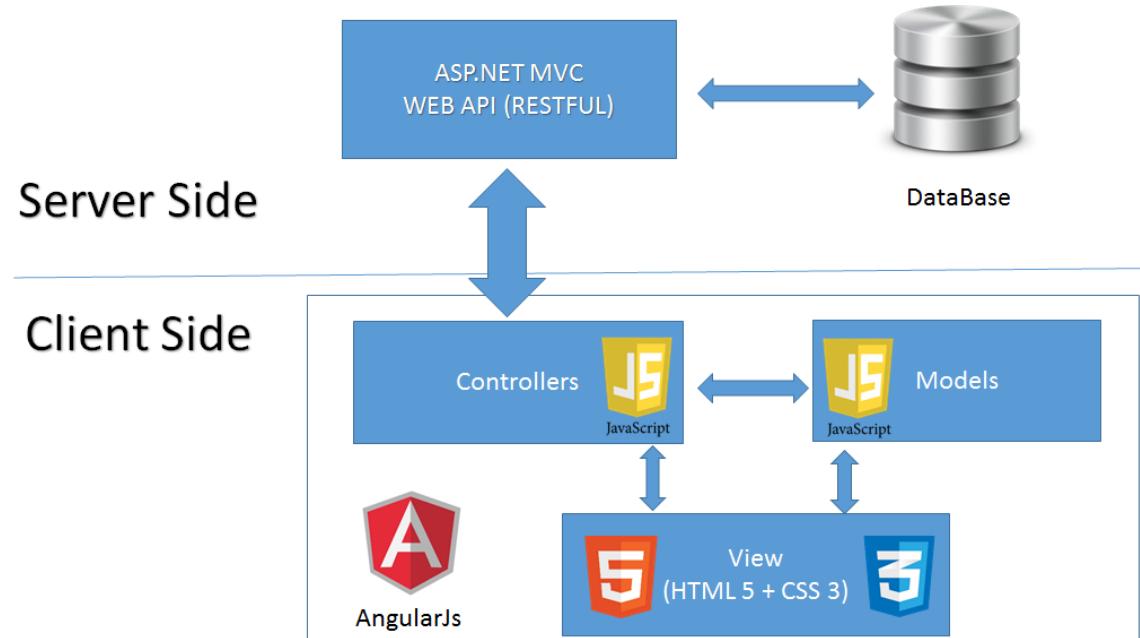
Web

Per default web render engines do not have view recycle pattern, but

- Lazy loading
- Scroll to load more
- Categories
- Various scrolling libs
- AngularJS - slow above 2500 two-way data bindings.

MVC Pattern for the Web

- Several different frameworks
 - emberjs
 - backbone.js
 - ...
 - **AngularJS**



Ionic

Uses **AngularJS** and it's MVC pattern in the core.

Lists can be improved with **collection-repeat** allows an app to show huge lists of items much more performant than **ng-repeat**. It renders into the DOM only as many items as are currently visible.

```
<ion-content>
  <ion-list>
    <ion-item collection-repeat="item in main.items">
      {{item}}
    </ion-item>
  </ion-list>
</ion-content>
```

Persist Data

How do I persist data on a mobile device?

With



<HTML5 />

{ HTML5 }



SQLite & Shared Prefs

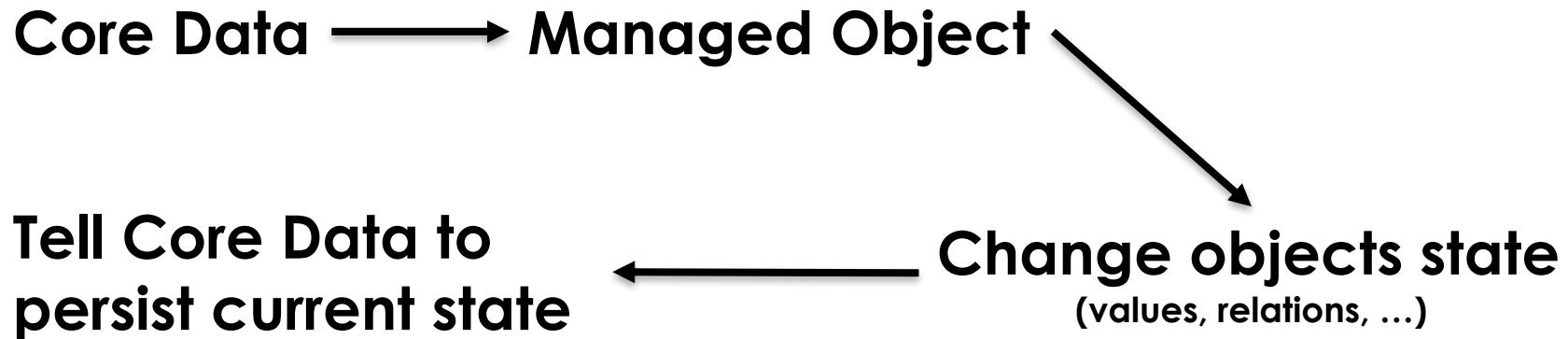


- Store structured data via SQL - but don't write SQL, use
 - ORMLite
 - GreenDao
- Use Shared Preferences for Key-Value data
 - `editor.putBoolean("silentMode", mSilentMode);`

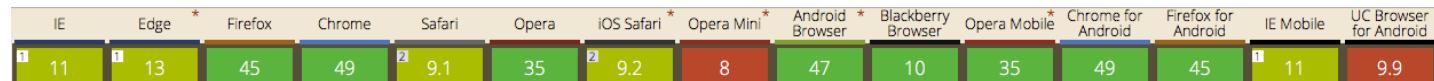
Core Data



Core Data is a framework that you use to manage the model layer objects in your application. It provides generalized and automated solutions to common tasks associated with object life cycle and **object graph management**, including persistence.

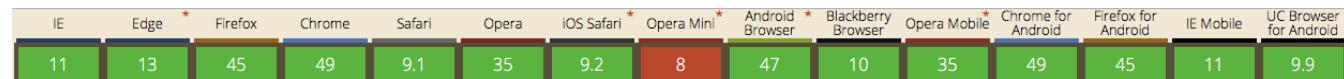


IndexedDB



lets you store and retrieve objects that are indexed with a "key." All changes that you make to the database happen within transactions.

LocalStorage



name/value pairs - `localStorage.setItem("lastname", "Smith");`

3rd Party Solutions like

- <https://www.javascripting.com/search?q=database>

Cordova

■ Local Storage

- Supported by all Cordova platforms.
- Limited in performance and complex data structures

■ WebSQL

- Good performance, SQL support
- Not supported by all platforms and deprecated

■ IndexedDB

- Good performance and simple to understand
- No iOS support

■ Plugin-Based Options

- Access file system to store data (e.g. via JSON)
- Several SQLite Plugins

REST

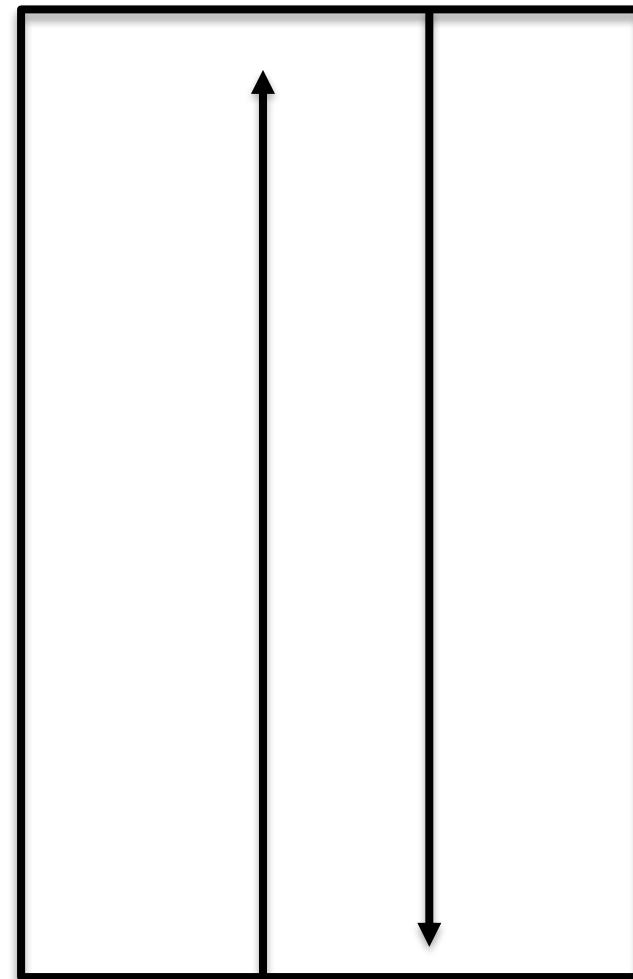
How do I send and receive data on a mobile device?

With



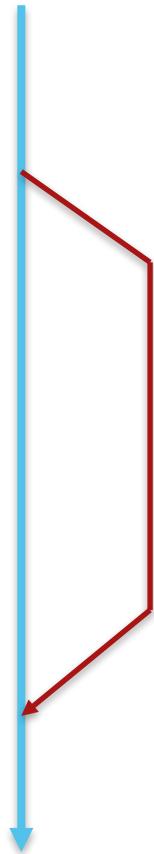
<HTML5 />

{ HTML5 }



Asynchronous Workload

UI Thread



REST calls

Download image
Intensive database query
Perform long task
...

REST



Android provides `AsyncTask`, `Threads`, ... and **iOS** `dispatch_async()` functions to execute code asynchronously.

But **REST** is so common that frameworks handle most of the workload for you:



Retrofit, Volley, Picasso



AFNetworking, Alamofire, SDWebImage

Hybrid and Web applications are build upon a multi-threaded technology – the browser.

Several ways of execution:

- XMLHttpRequest()
- jQuery.ajax()
- \$http()
- ...

CORS – enable Cross Origin Resource Sharing

JWT

JSON Web Token is a simple way to send information in the clear (usually in a URL) whose contents can be verified to be trusted. JWT **does not encrypt the payload**, it only signs it!

Advantages

- Easy Single Sign-On implementation
- Sends information that is not secret but needs to be verified
- Independent from client (works with all platforms)
- Enables a stateless server, good scalability and one token for all

Web App Tips

- Android Developers

<https://developer.android.com/guide/webapps/best-practices.html>

- Yahoo Developers

<https://developer.yahoo.com/performance/rules.html>

- Apple Developers

<https://developer.apple.com/library/ios/documentation/AppleApplications/Reference/SafariWebContent/CreatingContentforSafarioniPhone/CreatingContentforSafarioniPhone.html>

Automated Testing and Build Management

Why?

- Growing number of devices
- Many different OS versions and flavours
- Rapid Browser/OS development and release circles
- Quality management
- Frequent app updates are easier
- Faster development on the long run
- Continuous Integration

Challenges

- Testing with emulator or real device?
- Diversity of devices
- Different OS versions on different/same devices
- Testing from first build to production
- Maintenance of tests, team roles?

Android



- **Robotium**

is an open-source test framework for writing automatic gray box testing cases for Android and Hybrid applications.

<https://code.google.com/p/robotium/>

- **The Android testing framework**

provides an architecture and powerful tools that help you test every aspect of your application at every level from unit to framework.

http://developer.android.com/tools/testing/testing_android.html

- **UI/Application Exerciser Monkey**

is a stress-test application in a random yet repeatable manner.

<http://developer.android.com/tools/help/monkey.html>

Monkey



emulator64-arm Window

5554:rhuangab

FirstApp

3G 5:28

Basic Controls

Hardware Buttons

DPAD not enabled in AVD

Hardware Keyboard

Use your physical keyboard to provide input

Jerry

Candice

Austin

Micheal

Call Jerry

```
:Sending Trackball (ACTION_MOVE): 0:(-1.0,-2.0)
:Sending Trackball (ACTION_MOVE): 0:(0.0,-1.0)
Events injected: 1000
:Sending rotation degree=0, persist=false
:Dropped: keys=26 pointers=48 trackballs=0 flips=0 rotations=0
## Network stats: elapsed time=15217ms (15217ms mobile, 0ms wifi, 0ms not connected)
// Monkey finished
wf125-002:platform-tools Hushuiyi$ ./adb shell monkey -p com.example.firstapp -c android.intent.category.LAUNCHER -c android.intent.category.MONKEY -v 1000
:Monkey: seed=1373309690221 count=1000
:AllowPackage: com.example.firstapp
:IncludeCategory: android.intent.category.LAUNCHER
:IncludeCategory: android.intent.category.MONKEY
// Event percentages:
// 0: 15.0%
// 1: 10.0%
// 2: 2.0%
// 3: 15.0%
// 4: -0.0%
// 5: 25.0%
// 6: 15.0%
// 7: 2.0%
// 8: 2.0%
// 9: 1.0%
// 10: 13.0%
:Switch: #Intent;action=android.intent.action.MAIN;category=android.intent.category.LAUNCHER;launchFlags=0x10200000;component=com.example.firstapp/com.firstapp.MainActivity;end
    // Allowing start of Intent { act=android.intent.action.MAIN cat=[android.intent.category.LAUNCHER] cmp=com.example.firstapp/com.firstapp.MainActivity } in package com.example.firstapp
:Sending Touch (ACTION_DOWN): 0:(138.0,134.0)
:Sending Touch (ACTION_UP): 0:(156.4686,139.1827)
    // Allowing start of Intent { cmp=com.example.firstapp/com.firstapp.MainActivity } in package com.example.firstapp
:Sending Trackball (ACTION_MOVE): 0:(4.0,-1.0)
:Sending Trackball (ACTION_MOVE): 0:(-5.0,4.0)

exes. Prefix with pid:, app:, tag: or t verbose
YCODE_DPAD_LEFT, sec eventTime=1689651,
92 com.example.first... Choreographer Skipped 51 frames!
ts main thread.
```

- KIF – Keep it Functional

Automation of UI testing by leveraging the operating system's accessibility features and imitates actual user input.

<https://github.com/kif-framework/KIF>

- Xcode Test Cases

provides you with capabilities for extensive software testing.

https://developer.apple.com/library/prerelease/ios/documentation/DeveloperTools/Conceptual/testing_with_xcode/Introduction/Introduction.html#/apple_ref/doc/uid/TP40014132-CH1-SW1

KIF (Keep It Functional)



KIF | Build KIF: **Succeeded** | Today at 13:39 ⚠ 2

KIF Tests
104 tests

- AccessibilityIdentifierTests
 - testWaitingForViewWithAcc... ✓
 - testTappingViewWithAccess... ✓
 - testWaitingForAbscenceOfV... ✓
 - testLongPressingViewWithA... ✓
 - testEnteringTextIntoViewWit... ✓
 - testEnteringTextIntoViewWit... ✓
 - testClearingAndEnteringTex... ✓
- CascadingFailureTests
 - testCascadingFailure ✓
- CollectionViewTests
 - testTappingItems ✓
 - testTappingLastItemAndSec... ✓
 - testOutOfBounds ✓
 - testUnknownCollectionView ✓
 - testTappingItemsByLabel ✓
- CompositionTests
 - testTappingViewWithHint ✓
 - testTappingOnlyIfNotSelected ✓
- ExistTests
 - testExistsViewWithAccessib... ✓
- GestureTests
 - testSwipingLeft ✓
 - testSwipingRight ✓
 - testSwipingUp ✓
 - testSwipingDown ✓
 - testMissingSwipeableElement ✓
 - testSwipingLeftWithTraits ✓
 - testSwipingRightWithTraits ✓

```
1 //  
2 // SystemAlertTests.m  
3 // KIF  
4 //  
5 // Created by Joe Masiolli on 12/1/14.  
6 //  
7 //  
8 #import <KIF/KIF.h>  
9  
10 @interface SystemAlertTests : KIFTestCase  
11  
12 @end  
13  
14 @implementation SystemAlertTests  
15  
16 - (void)beforeEach  
17 {  
18     [tester tapViewWithAccessibilityLabel:@"System Alerts"];  
19 }  
20  
21 - (void)afterEach  
22 {  
23     [tester tapViewWithAccessibilityLabel:@"Test Suite" traits:UIAccessibilityTraitButton];  
24 }  
25  
26 - (void)testAuthorizingLocationServices {  
27     [tester tapViewWithAccessibilityLabel:@"Location Services"];  
28     [tester acknowledgeSystemAlert];  
29 }  
30  
31 - (void)testAuthorizingPhotosAccess {  
32     [tester tapViewWithAccessibilityLabel:@"Photos"];  
33     [tester acknowledgeSystemAlert];  
34     [tester tapViewWithAccessibilityLabel:@"Cancel"];  
35 }  
36  
37 - (void)testNotificationScheduling {  
38     [tester tapViewWithAccessibilityLabel:@"Notifications"];  
39     [tester acknowledgeSystemAlert];  
40 }  
41  
42 @end
```

Native/Hybrid/Web

- Selendroid

Test automation for native or hybrid Android apps and the mobile web with Selendroid.

<http://selendroid.io/>

- Calaba.sh

enables you to write and execute automated acceptance tests of mobile apps. It's cross-platform, supporting Android and iOS native apps.

<http://calaba.sh/>

Web App

- W3C Validator

checks the markup validity of Web documents in HTML, XHTML, SMIL, MathML, etc.

<https://validator.w3.org/>

- Browserstack

Live, Web-Based Browser Testing. Instant access to all desktop and mobile browsers.

<https://www.browserstack.com/>

- http://en.wikipedia.org/wiki/List_of_web_testing_tools

Build Tools

handles libraries, application flavours & multitenancy, additional scripts, resources and many more.

- **Gradle** for Android
- **CocoaPods** for iOS
- **NPM, Grunt, Bower, ...** for HTML5 and Hybrid

Engineering in the Wild

Engineering in the wild

You want to develop an app in a distributed setting (development, test, stakeholders) with minimal friction and overhead, yet thorough release planning and controlling.

How do you ensure:

- Traceability / Controlling
- Code quality
- Quality control and testing
- Deployment of test and release builds

Engineering in the wild

- Issue tracking incl. time tracking
- Requirements specification
- Coding guidelines
- Source control and code reviews
- **Working** continuous integration!
- Library handling
- Quality control and acceptance criteria
- Automated testing and deployment

Issue tracking

My project

Search: Jump to a project...

Overview Activity Roadmap Issues News Documents Wiki Files Repository Settings

Issues

Filters

Status Add filter:

#	Tracker	Status	Priority	Subject	Assigned to	Updated
127	Bug	New	Normal	Ticket with attachments		12/22/2007 12:11 PM
116	Bug	New	Low	Keep playing audio when rw/ff and preserve pitch.	John Smith	12/17/2007 09:56 PM
88	Feature	Assigned	Low	HTTP Challenge-MD5 authentication		12/22/2007 04:33 PM
83	Feature	Assigned	Low	Export the parameters of an input	John Smith	12/17/2007 09:56 PM
82	Feature	New	Low	Formatted text rendering support	Dave Loper	12/17/2007 06:58 PM
81	Feature	New	Normal	DVTS support		12/17/2007 06:58 PM
79	Feature	New	Low	QuickTime capturing		17/2007 06:58 PM
78	Feature	New	Low	Full H323 videoconferencing		17/2007 06:58 PM
77	Feature	Assigned	Low	Closed captions / Teletext support		17/2007 06:58 PM
74	Feature	New	Low	Progressive download playing		
73	Feature	New	Low	Dshow tuning support		
72	Feature	New	Low	V4L tuning support		
70	Feature	New	Low	Electric Program Guide		
69	Bug	New	Low	SDL vout cleaning		
65	Feature	New	Low	Protocol rollover support		
64	Feature	New	Normal	Improve ZLM functionality		12/22/2007 04:33 PM
63	Feature	New	Low	Gstreamer and Helix integration		12/17/2007 06:58 PM
62	Feature	New	Low	Gnutella servlet		12/17/2007 06:58 PM
59	Feature	New	Low	Finalization of Pocket PC port		12/17/2007 06:58 PM
58	Bug	Assigned	Low	Re-write of the AppleScript bindings		12/22/2007 04:33 PM
57	Feature	New	Low	MacOS X SVCD support	Dave Loper	12/17/2007 06:58 PM

New issue

Tracker:

Issues

[View all issues](#) [Summary](#) [Change log](#)

Custom queries

[Assigned to me](#) [Due this week](#) [Late features](#)

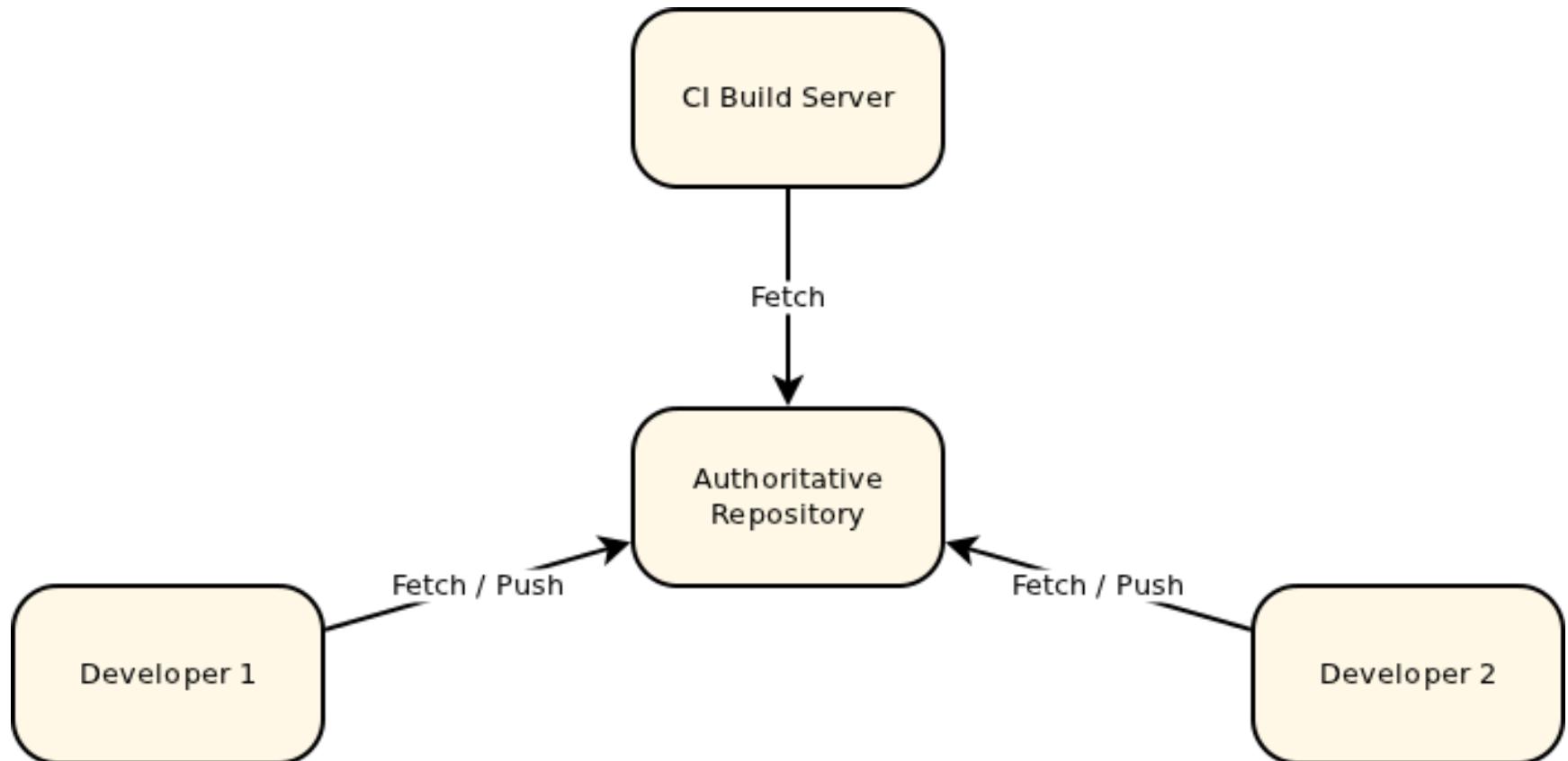
Issue tracking - traceability

No commit without ticket!

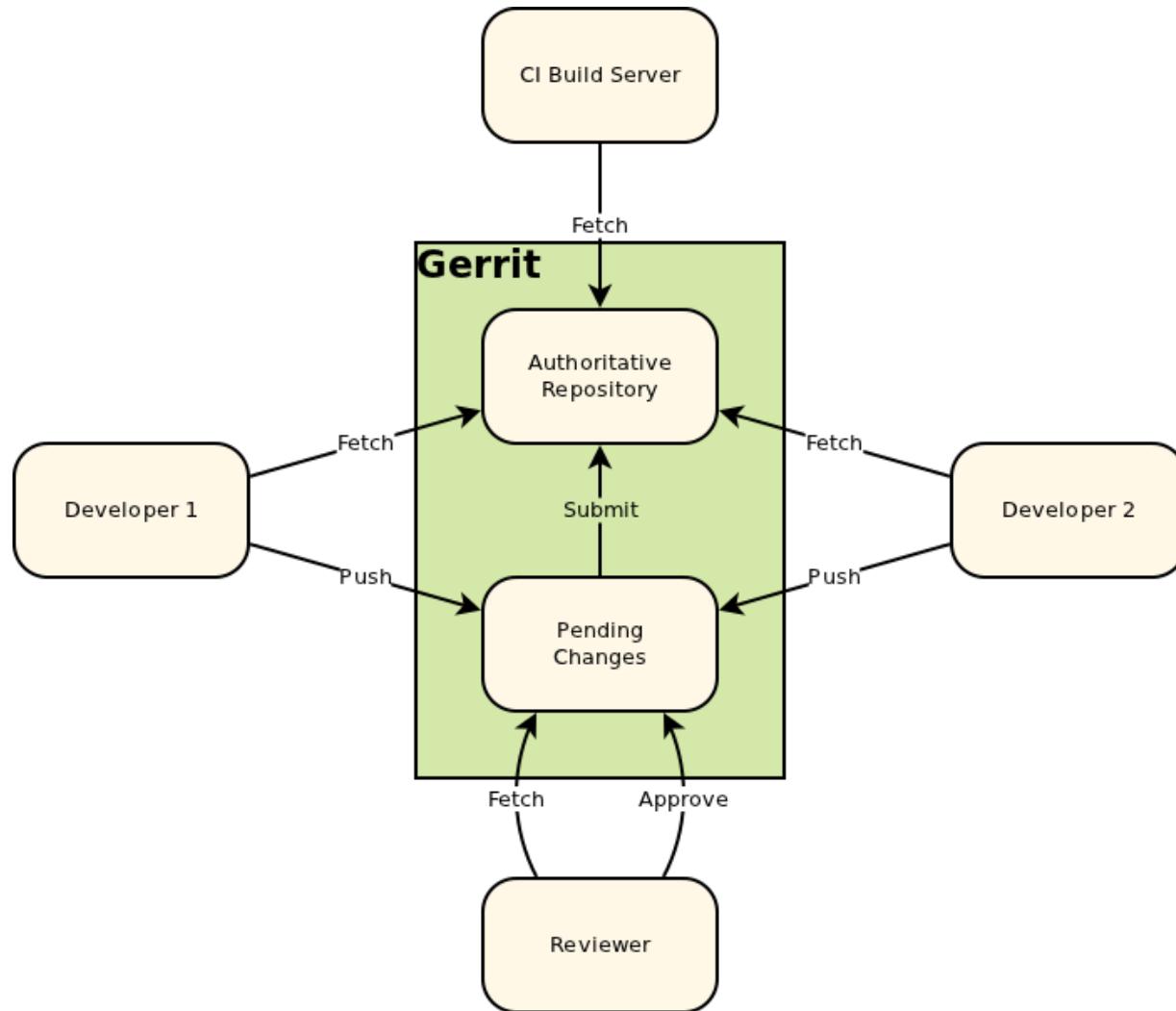
Add hooks to your commit messages:

```
fixes #123 @2.0 – adds new viewcontroller  
for contact details.
```

Source control



Code reviews (gerrit)



Code reviews (gerrit)

The screenshot shows a Gerrit code review interface comparing two versions of a file named `PizzaDough.txt`. The interface has tabs for `All`, `My`, `Differences`, and `Admin`. The `Differences` tab is selected, showing a `Side-by-Side` view.

Commit Message: [Up to change](#)

Old Version (Download)

```
1 Ingredients
2 -----
3 4 cups Flour
4 4 tsp Baking Soda
5 1 cup Water
6 15 ml Olive Oil
7
8 Method
9 -----
10
11 Combine Dry Ingredients and form a well.
12
13 Add Olive Oil and Water and mix to combine.
14
15 Knead into a rough ball, the roll out to form Pizza bases.
```

New Version (Download)

```
1 Ingredients
2 -----
3 4 cups Flour
4 1 tbsp Sugar
5 1 cup Yeast
(Draft)
1 cup yeast? That seems like a lot. Are you sure you don't mean 1 tbsp?
6 1 cup Tepid Water
7 15 ml Olive Oil
8
9 Method
10 -----
11
12 Add yeast and sugar to Tepid water and allow to
13 Create a well in the Flour.
14
15
16 Add Olive Oil and Water / Yeast mixture and mi
17
18 Knead into a rough ball, then turn out on to a
```

Annotations:

- Line 4: `4 tsp Baking Soda` is highlighted in red.
- Line 5: `1 cup Water` is highlighted in yellow.
- Line 6: `15 ml Olive Oil` is highlighted in pink.
- Line 11: `Combine Dry Ingredients and form a well.` is highlighted in red.
- Line 13: `Add Olive Oil and Water and mix to combine.` is highlighted in pink.
- Line 15: `Knead into a rough ball, the roll out to form Pizza bases.` is highlighted in red.
- Line 4: `1 tbsp Sugar` is highlighted in green.
- Line 5: `1 cup Yeast` is highlighted in green.
- Line 12: `Add yeast and sugar to Tepid water and allow to` is highlighted in green.
- Line 13: `Create a well in the Flour.` is highlighted in green.
- Line 16: `Add Olive Oil and Water / Yeast mixture and mi` is highlighted in green.
- Line 18: `Knead into a rough ball, then turn out on to a` is highlighted in green.

Code reviews (gerrit)

The screenshot shows a Gerrit web interface for a code review. At the top, the title bar reads "Change Ifb5296eb - Patch Set 1...". Below it, the URL is "http://localhost:8080/#change,publish,68,1". The status bar indicates "Updated Jul 30, 2011 1:30 PM" and "Status Review in Progress". A "Permalink" link is also present.

Code Review:

- +2 Looks good to me, approved
- +1 Looks good to me, but someone else must approve
- 0 No score
- 1 I would prefer that you didn't submit this
- 2 Do not submit

Cover Message:

Switching to a Yeast based dough is a good idea, but please recheck the quantities.

Patch Comments:

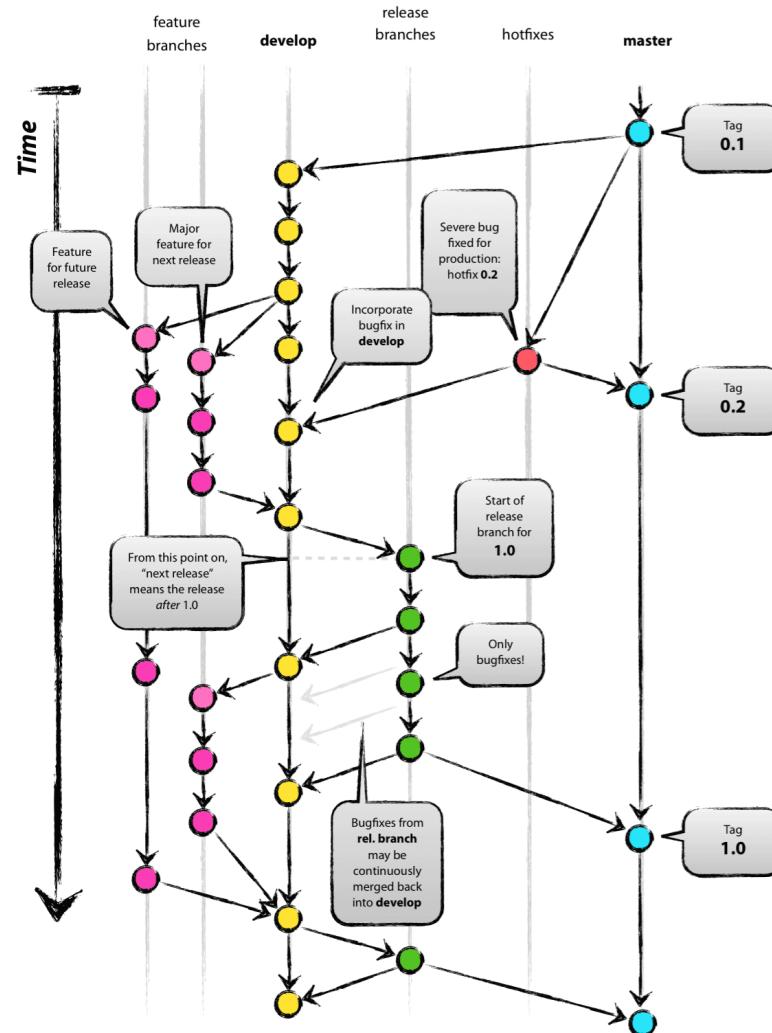
[PizzaDough.txt](#)

Line 5:

1 cup yeast? That seems like a lot. Are you sure you don't mean 1 tbsp?

[Edit](#) [Publish Comments](#) [Cancel](#)

GIT branching model



<http://nvie.com/posts/a-successful-git-branching-model/>

Continuous integration

Jenkins

Jenkins > All >

Suchen anmelden AUTO-AKTUALISIERUNG EINSCHALTEN

Benutzer

- Build-Verlauf
- Projektbeziehungen
- Fingerabdruck überprüfen
- Disk usage
- We Need Beer

Build Warteschlange (1)

- infra_update_center_v3

Build-Prozessor-Status

- master**
 - 1 infra_update_center_v3 #2356
 - 2 Ruhend
- cabbage**
 - 1 Ruhend
 - 2 Ruhend
- celery**
 - 1 Ruhend
 - 2 Ruhend
- remote-slave-3 (offline)**
- remote-slave-6**
 - 1 Ruhend
 - 2 Ruhend
- remote-slave-7 (offline)**
- remote-slave-8 (offline)**

All	All Disabled	All Failed	All Unstable	Infrastructure	Jenkins core	Libraries	Other Projects
S	W	Name ↓		Letzter Erfolg	Letzter Fehlschlag	Letzte Dauer	
●	☀️	config-provider-model		1 Jahr 7 Monate - #1367	3 Jahre 0 Monate - #1353	1 Minute 9 Sekunden	
●	☀️	danielbeck		Nicht anwendbar	Unbekannt	Nicht anwendbar	
●	☀️	fix-git-configuration-on-remote-slave-8		4 Jahre 2 Monate - #1	Unbekannt	2.8 Sekunden	
●	☀️	infra_accountapp		1 Jahr 7 Monate - #1	Unbekannt	5 Minuten 56 Sekunden	
●	☁️	infra_atlassian-base		8 Monate 13 Tage - #3	8 Monate 13 Tage - #1	1 Minute 15 Sekunden	
●	☀️	infra_backend-confluence-spam-remover		3 Jahre 0 Monate - #2	Unbekannt	29 Sekunden	
●	🌧️	infra_backend-merge-all-repo		2 Tage 8 Stunden - #195	16 Tage - #193	13 Stunden	
●	☀️	infra_backend-plugin-report-card		6 Stunden 24 Minuten - #1478	Unbekannt	8 Minuten 18 Sekunden	
●	🔴	infra_backend-war-size-tracker		20 Tage - #1126	7 Stunden 9 Minuten - #1146	1 Minute 47 Sekunden	
●	🔴	infra_backend_crawler		2 Tage 18 Stunden - #185	18 Stunden - #187	7 Minuten 50 Sekunden	
●	🔴	infra_backend_jenkins_ci_cloubess_com_filler		19 Tage - #3079	3 Stunden 57 Minuten - #3157	4 Minuten 38 Sekunden	
●	🌧️	infra_backend_pull_request_greeter		19 Tage - #3392	5 Stunden 52 Minuten - #3470	3 Minuten 44 Sekunden	
●	☀️	infra_bind		1 Jahr 7 Monate - #4	Unbekannt	5 Minuten 12 Sekunden	
●	☀️	infra_census_push_to_celery		16 Stunden - #418	1 Jahr 1 Monat - #3	3.8 Sekunden	
●	☀️	infra_changelog_refresh		6 Stunden 8 Minuten - #28096	Unbekannt	24 Sekunden	
●	☀️	infra_checkout_stats		23 Stunden - #1405	1 Jahr 7 Monate - #769	47 Sekunden	
●	🔴	infra_commit_history_generation		1 Jahr 5 Monate - #421	17 Stunden - #935	4 Minuten 8 Sekunden	
●	☀️	infra_confluence		8 Monate 0 Tage - #7	8 Monate 13 Tage - #2	15 Minuten	
●	🌧️	infra_confluence-cache		8 Monate 13 Tage - #21	8 Monate 13 Tage - #20	2 Minuten 19 Sekunden	
●	☀️	infra_deprecated-usage-in-plugins		37 Minuten - #396	19 Tage - #1	32 Sekunden	

References

- Brahima, S. (2014). The World in 2014: ICT Facts and Figures. Retrieved March 10, 2015, from <http://www.itu.int/en/ITU-D/Statistics/Pages/facts/default.aspx>
- Joorabchi, M.E., A. Mesbah, and P. Kruchten. "Real Challenges in Mobile App Development." In *2013 ACM / IEEE International Symposium on Empirical Software Engineering and Measurement*, 15–24, 2013. doi:10.1109/ESEM.2013.9.
- Felt, A. P., Ha, E., Egelman, S., Haney, A., Chin, E., & Wagner, D. (2012). Android Permissions: User Attention, Comprehension, and Behavior. In *Proceedings of the Eighth Symposium on Usable Privacy and Security* (pp. 3:1–3:14). New York, NY, USA: ACM. <http://doi.org/10.1145/2335356.2335360>



deco.inso.tuwien.ac.at

deco@inso.tuwien.ac.at

Thank you!

Material

github.com/spipau

App

Search for “**RISE Weather**” in the
Play Store

