DA: Course Presentation

Applications for mobile devices - Theory - Unit 0

Didac Florensa Cazorla

Any: 2021-2022
Curs: 102386
Institut: University of Lleida (Campus Igualada)
Titulació: Bachelor's degree in Digital Interaction and Computing Techniques (GTIDIC)

Agenda

Introduction

Mobile Platforms

Course

Homework

Introduction

Warm-up

- Dr. Jordi Mateo Fornés
- **Office**:
  - Office A.12 (Campus Igualada)
  - Office 3.08 (EPS Lleida)
- **Email**: jordi.mateo@udl.cat
- **Twitter**: https://twitter.com/MatForJordi
- **Github**: https://github/JordiMateoUdL

- Ph.D Dídac Florensa Cazorla
- **Office**:
  - Office A.12 (Campus Igualada)
  - Office 3.08 (EPS Lleida)
- **Email**: didac.florensa@udl.cat

Presentation and Expectatives

Who are you?

⇒ Answer briefly to this questions:

- Which is your dream job?
- What are your coding skills?
- What are you doing in this class?
- What do you expect to learn?

What should you learn?

- Learn to code and launch mobile apps using JAVA.

Presentation and Expectatives

Who are you?

⇒ Answer briefly to this questions:

- Which is your dream job?
- What are your coding skills?
- What are you doing in this class?
- What do you expect to learn?

What should you learn?

- Learn to code and launch mobile apps using JAVA.
- Learn to code and develop backends.

Presentation and Expectatives

Who are you?

⇒ Answer briefly to this questions:

- Which is your dream job?
- What are your coding skills?
- What are you doing in this class?
- What do you expect to learn?

What should you learn?

- Learn to code and launch mobile apps using JAVA.
- Learn to code and develop backends.
- Integrate DEV-OPS.

Presentation and Expectatives

Who are you?

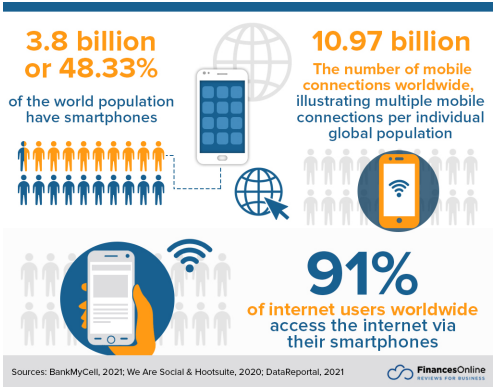
⇒ Answer briefly to this questions:

- Which is your dream job?
- What are your coding skills?
- What are you doing in this class?
- What do you expect to learn?

What should you learn?

- Learn to code and launch mobile apps using JAVA.
- Learn to code and develop backends.
- Integrate DEV-OPS.
- Work in a real coding project.

Motivation



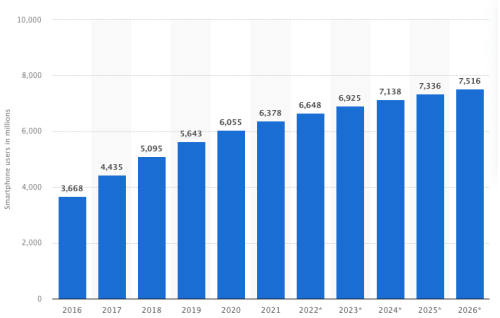
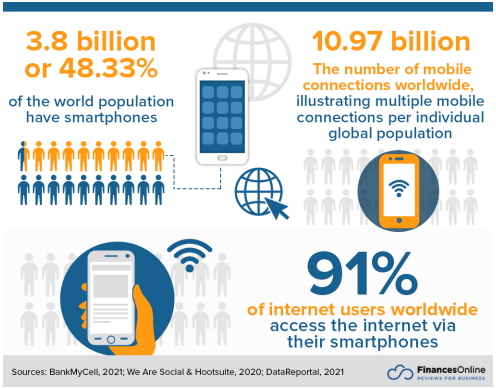


Figure 1: Number of smartphone users from 2016 to 2021 (in billions) from: statista

- How many use a **smart-X** (smart-phone, smart-watch, ...)?

Motivation



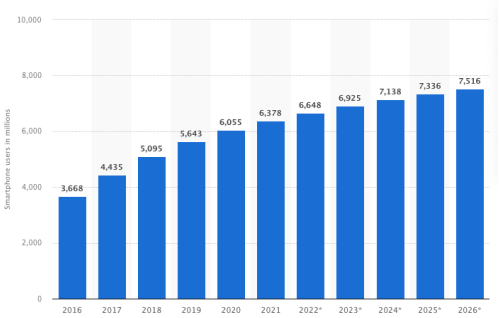
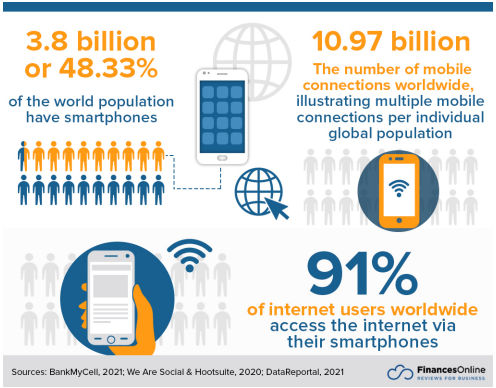


Figure 1: Number of smartphone users from 2016 to 2021 (in billions) from: statista

- How many use a **smart-X** (smart-phone, smart-watch, ...)?
- How many hours do you expand on the mobile phone?

Motivation



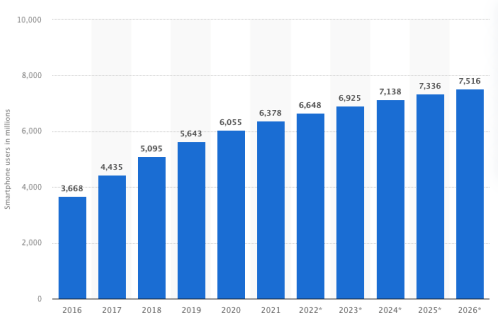


Figure 1: Number of smartphone users from 2016 to 2021 (in billions) from: statista

- How many use a **smart-X** (smart-phone, smart-watch, ...)?
- How many hours do you expand on the mobile phone?
- What do you do (social media, web navigation, multimedia,...)?

What is mobile computing?

Mobile Computing is a technical field that covers the *design, development and evaluation* of **mobile applications** using appropriate solutions that meet *user requirements*. Portable devices include Smartphones, Tablets, Laptops, wearable devices, vehicles, and more

In summary:

- Ability to use technology while moving.

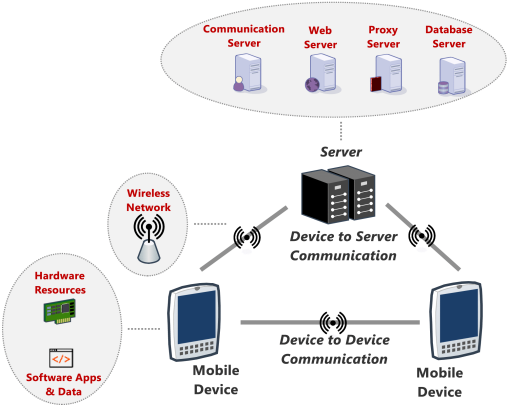


Figure 2: Mobile computing from IEEE Computer Society

What is mobile computing?

Mobile Computing is a technical field that covers the *design, development and evaluation* of **mobile applications** using appropriate solutions that meet *user requirements*. Portable devices include Smartphones, Tablets, Laptops, wearable devices, vehicles, and more

In summary:

- Ability to use technology while moving.
- Ability to interact in any context.

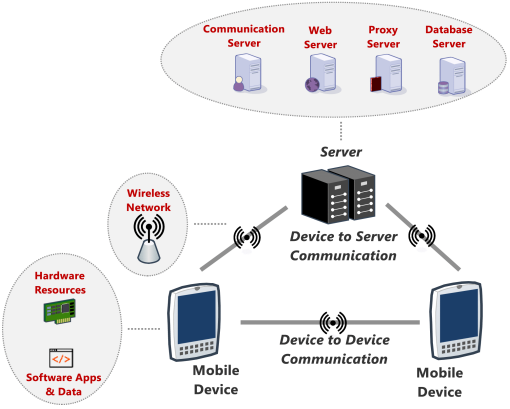


Figure 2: Mobile computing from IEEE Computer Society

What is mobile computing?

Mobile Computing is a technical field that covers the *design, development and evaluation* of **mobile applications** using appropriate solutions that meet *user requirements*. Portable devices include Smartphones, Tablets, Laptops, wearable devices, vehicles, and more

In summary:

- Ability to use technology while moving.
- Ability to interact in any context.
- Computing devices not associated with a location.

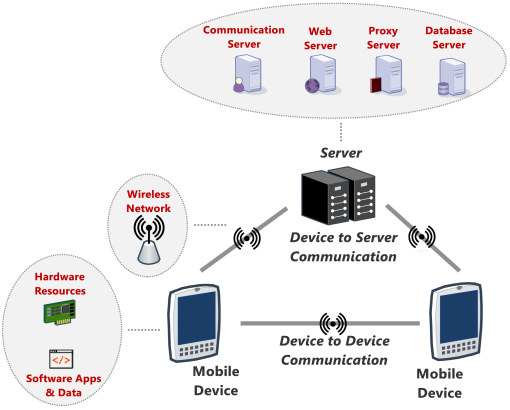


Figure 2: Mobile computing from IEEE Computer Society

What is mobile computing?

Mobile Computing is a technical field that covers the *design, development and evaluation* of **mobile applications** using appropriate solutions that meet *user requirements*. Portable devices include Smartphones, Tablets, Laptops, wearable devices, vehicles, and more

In summary:

- Ability to use technology while moving.
- Ability to interact in any context.
- Computing devices not associated with a location.
- Battery powered devices.

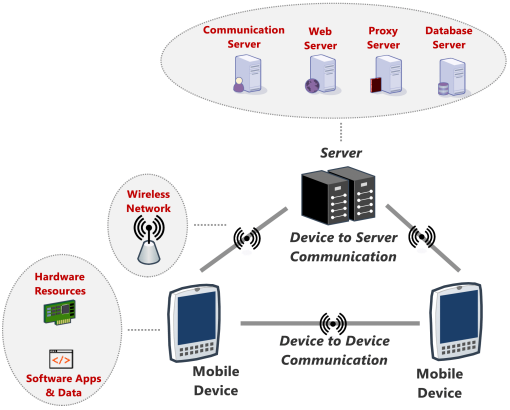


Figure 2: Mobile computing from IEEE Computer Society

What is mobile computing?

Mobile Computing is a technical field that covers the *design, development and evaluation* of **mobile applications** using appropriate solutions that meet *user requirements*. Portable devices include Smartphones, Tablets, Laptops, wearable devices, vehicles, and more

In summary:

- Ability to use technology while moving.
- Ability to interact in any context.
- Computing devices not associated with a location.
- Battery powered devices.
- High accessibility with and without internet.

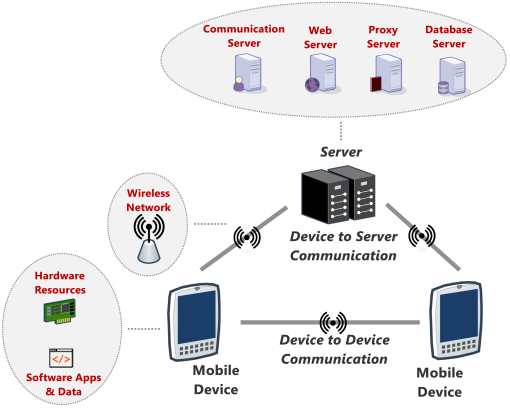


Figure 2: Mobile computing from IEEE Computer Society

Challenges in mobile computing

Mobile devices are often small and have limited processing capabilities.

- Security, secrecy, and privacy

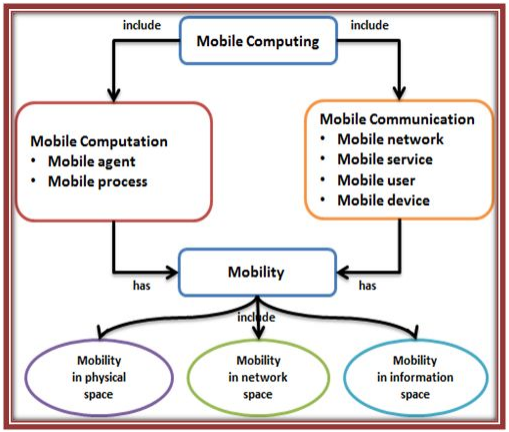


Figure 3: Mobile Computing Challenges from ns3simulation

Challenges in mobile computing

Mobile devices are often small and have limited processing capabilities.

- Security, secrecy, and privacy
- Reliability in presence of disconnections.

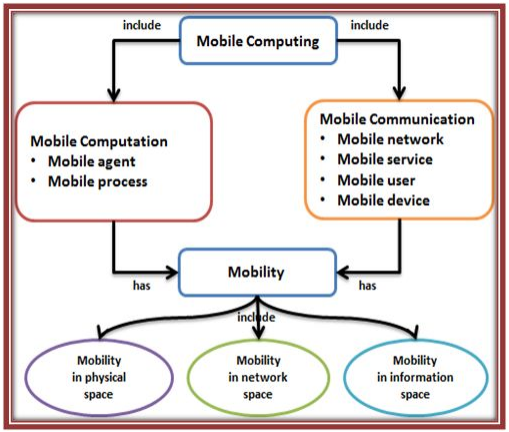


Figure 3: Mobile Computing Challenges from ns3simulation

Challenges in mobile computing

Mobile devices are often small and have limited processing capabilities.

- Security, secrecy, and privacy
- Reliability in presence of disconnections.
- Mobility/context aware applications.

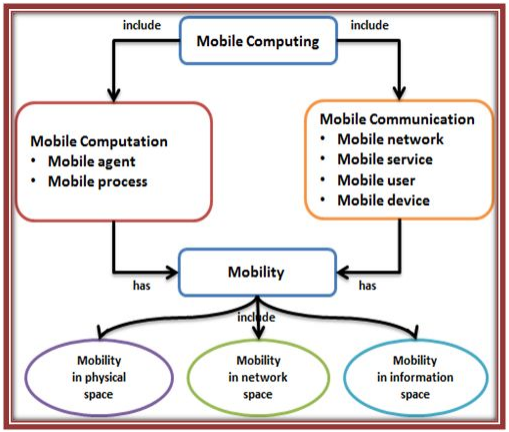


Figure 3: Mobile Computing Challenges from ns3simulation

Challenges in mobile computing

Mobile devices are often small and have limited processing capabilities.

- Security, secrecy, and privacy
- Reliability in presence of disconnections.
- Mobility/context aware applications.
- Resource management: Memory and cpu performance.

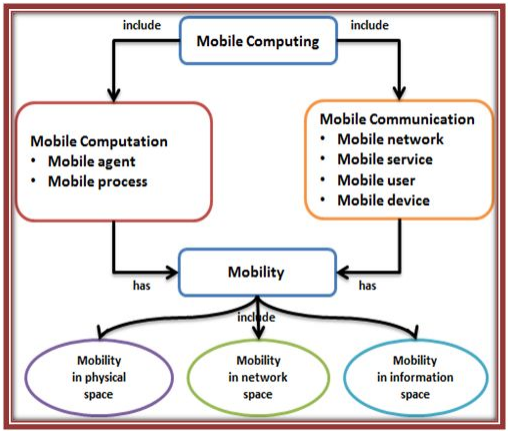


Figure 3: Mobile Computing Challenges from ns3simulation

Challenges in mobile computing

Mobile devices are often small and have limited processing capabilities.

- Security, secrecy, and privacy
- Reliability in presence of disconnections.
- Mobility/context aware applications.
- Resource management: Memory and cpu performance.
- Communication infrastructure

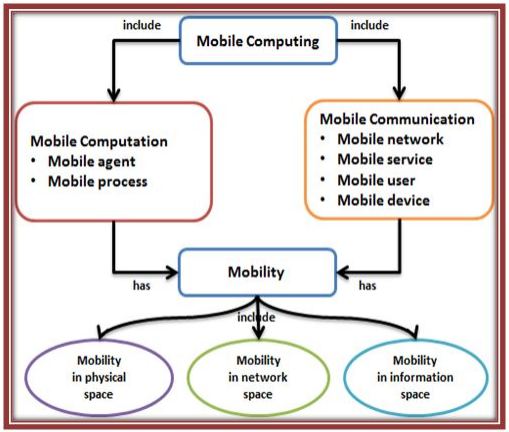


Figure 3: Mobile Computing Challenges from ns3simulation

Challenges in mobile computing

Mobile devices are often small and have limited processing capabilities.

- Security, secrecy, and privacy
- Reliability in presence of disconnections.
- Mobility/context aware applications.
- Resource management: Memory and cpu performance.
- Communication infrastructure
- Need to keep power consumption down

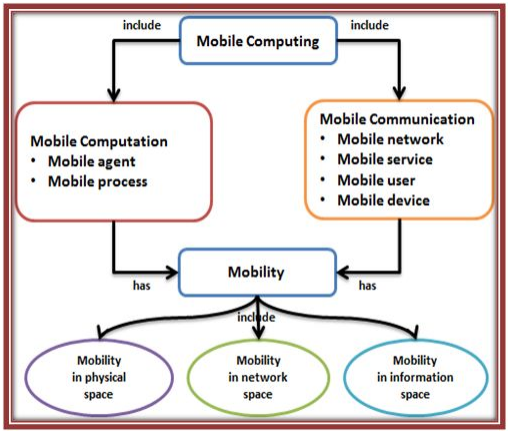


Figure 3: Mobile Computing Challenges from ns3simulation

Challenges in mobile computing

Mobile devices are often small and have limited processing capabilities.

- Security, secrecy, and privacy
- Reliability in presence of disconnections.
- Mobility/context aware applications.
- Resource management: Memory and cpu performance.
- Communication infrastructure
- Need to keep power consumption down
- Complex environment

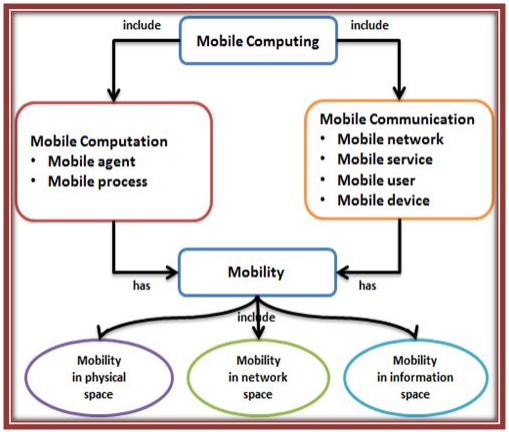


Figure 3: Mobile Computing Challenges from ns3simulation

Challenges in mobile computing

Mobile devices are often small and have limited processing capabilities.

- Security, secrecy, and privacy
- Reliability in presence of disconnections.
- Mobility/context aware applications.
- Resource management: Memory and cpu performance.
- Communication infrastructure
- Need to keep power consumption down
- Complex environment
- Limited bandwidth

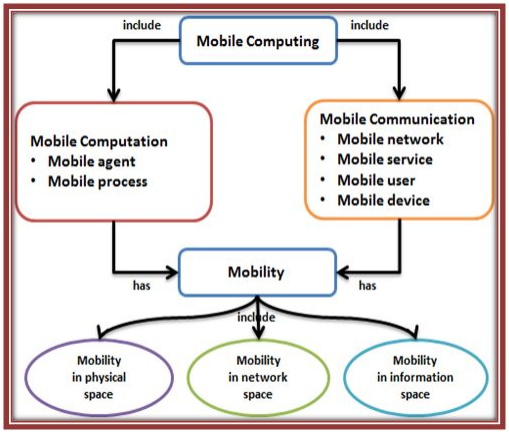


Figure 3: Mobile Computing Challenges from ns3simulation

Challenges in mobile computing

Mobile devices are often small and have limited processing capabilities.

- Security, secrecy, and privacy
- Reliability in presence of disconnections.
- Mobility/context aware applications.
- Resource management: Memory and cpu performance.
- Communication infrastructure
- Need to keep power consumption down
- Complex environment
- Limited bandwidth
- High range of diferent devices

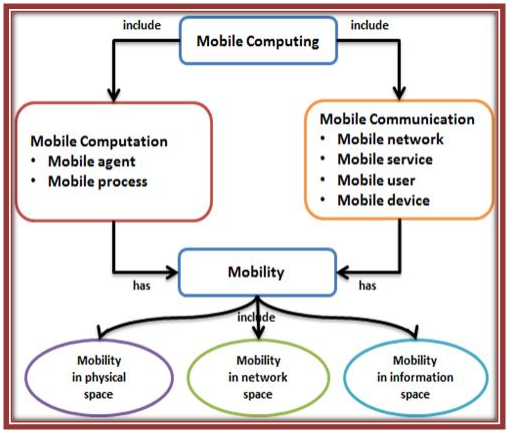


Figure 3: Mobile Computing Challenges from ns3simulation

Mobile App Development

- Programming is easy, but software engineering is hard.
- Developing apps is multidisciplinary
  - Write code.
  - Develop or use third-party APIs.
  - Design and scalable and maintainable architecture.
  - Creative and Usable design.

Useful workflow

- Agile methods
- Git flow
- Review code
- Testing
- Integration

Useful Tools

- IDE
- Sketch, JustinMind
- GIT
- Issue trackers
- Slack

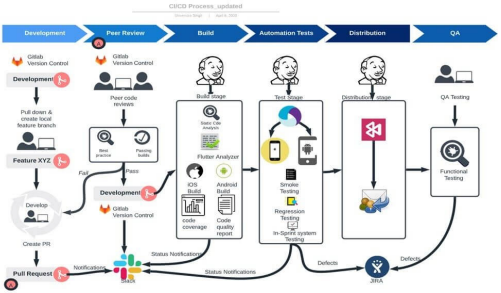


Figure 4: Mobile App Development CI/CD from DevOps Institute

Mobile Platforms

Warm-Up

- What mobile platforms do you know?
- What is your experience?

Which are the main mobile platforms?

Native tools

- Android
- iOs

Advantages

- Allows users to quickly learn.
- Easy to discover (Play Store or Apple Store).
- Easy to use the device hardware.
- High Performance and Great UX.

Cross-platform tools:

- PhoneGap
- React Native
- Xamarin
- Flutter
- Kotlin Native

Advantatges

- Portability.
- Faster development.
- Cheaper.
- Easy to support and maintain.

What about responsive web apps?

Main platforms

Android

- Advantages
  - Java and Kotlin as programming languages.
  - Mature architecture.
  - ConstraintLayout 2.0.
  - Google is behind
- Disadvantages
  - Fragmentation
  - Android X

iOS

- Advantages
  - Swift as a programming language.
  - Swift is open source.
  - Fast adoption of latest OS.
- Disadvantages
  - Cost
  - You need a mac to develop.
  - Strict App review.

React Native

- React library and JavaScript to deliver a native experience on iOS and Android.
- Fast development.
- Used by: Instragram, Facebook, Airbnb, Walmart, Tesla ...

Ionic

- Makes the best use of web tech HTML, CSS, javascript.
- Strong community support.
- Highly interactive apps.
- Easy learning curve.

Flutter

- Dart is a modern, multi-paradigm and objected-oriented programming.
- High performance.
- Used by: Alibaba, Hamilton Musical, Google Ads,...

What is the best option?

It depend. Everyone needs to choose the one that suits their needs in a better way. We need to consider: *One or multiple platforms*, *users*, *cost*, *maintenance*, *technology*...

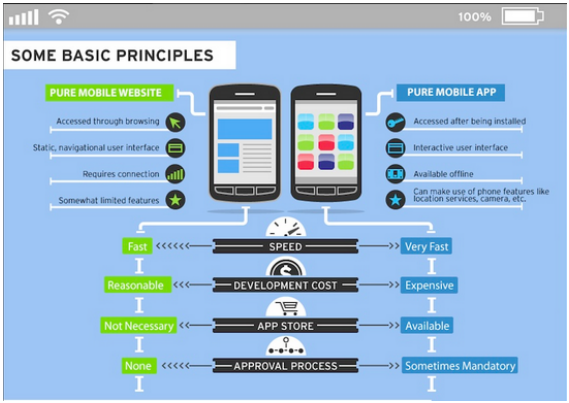


Figure 5: Mobile Apps vs Web Apps from bevisible

Course

Objectives

- Understand the Android platform and the elements that make it up.
- Understand and use coding patterns.
- Develop applications for the Android operating system.
- Establish the bases for the implementation of additional functionalities (access to the database, access to resources and features of the mobile, etc.).
- Develop and use API's as a backend.
- Get to know the step of publishing Android applications.

Android Programming Languages

Java

- Advantages:
  - Easy to learn, understand and flexible.
  - A good choice for cross-platform apps.
  - Java has a large open-source ecosystem.
  - More compact and light apps.
  - Fast build compared with Kotlin.
- Disadvantages:
  - Limitations that causes problems in android design.
  - You need to write more.
  - Requires a lot of memory

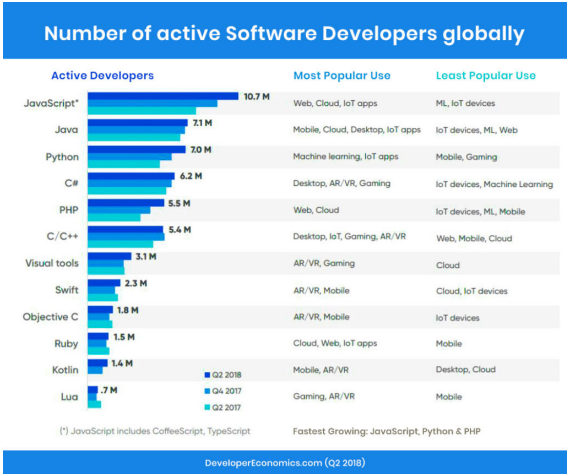
Kotlin

- Advantages:
  - Easy to switch from Java.
  - Smart extensions to build.
  - More concise.
  - Compatible with all Java libraries, frameworks and JVM.
  - Compatible with Gradle or Maven.
  - Fast build compared with Kotlin.
- Disadvantages:
  - Slower compilation.
  - Less community.
  - Not as mature as JAVA.

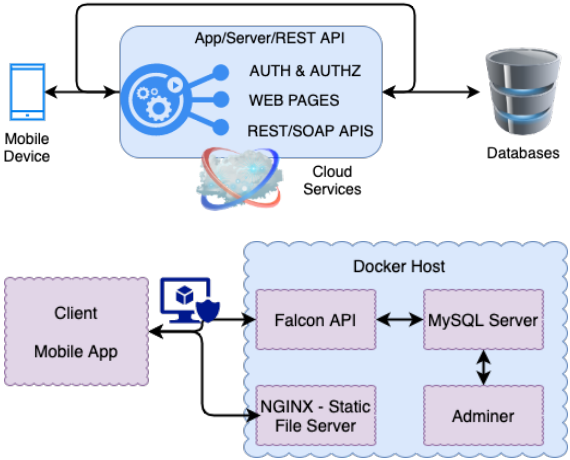
Java vs Kotlin

Attributes	Java	Kotlin
App Performance	High	Super High
Android Studio 3.0 Support	Partial	Excellent
Code Quality	Not-Optimized	Excellent
Market Presence	Excellent	Good
Adoption Cost	High	Low
App Security	Good	Excellent
Support for Complex Architecture	Excellent	Not Good

Why JAVA?


Number of active Software Developers globally
Active Developers
Most Popular Use
Least Popular Use
JavaScript*
10.7 M
Web, Cloud, IoT apps
ML, IoT devices
Java
7.1 M
Mobile, Cloud, Desktop, IoT apps
IoT devices, ML, Web
Python
7.0 M
Machine learning, IoT apps
Mobile, Gaming
C#
6.2 M
Desktop, AR/VR, Gaming
IoT devices, Machine Learning
PHP
5.5 M
Web, Cloud
IoT devices, ML, Mobile
C/C++
5.4 M
Desktop, IoT, Gaming, AR/VR
Web, Mobile, Cloud
Visual tools
3.1 M
AR/VR, Gaming
Cloud
Swift
2.3 M
AR/VR, Mobile
Cloud, IoT devices
Objective C
1.8 M
AR/VR, Mobile
IoT devices
Ruby
1.5 M
Cloud, Web, IoT apps
Mobile
Kotlin
1.4 M
Mobile, AR/VR
Desktop, Cloud
Lua
.7 M
Gaming, AR/VR
Mobile
Q2 2018
Q4 2017
Q2 2017
(*) JavaScript includes CoffeeScript, TypeScript
Fastest Growing: JavaScript, Python & PHP
DeveloperEconomics.com (Q2 2018)

Architecture to learn during the course



Evaluation

- **Project**: Development of your own app. **50%**.
  - DA1. Scaffolding and domain model.
  - DA2. Logic and tests.
  - DA3. Data persistence and data model(API).
  - DA4. API communication and User sessions.
  - DA5. Custom I.
  - DA6. Custom II.
  - DA7. Custom III.
- **Exams 30%**
- **Integrated Project**: **20%**.

Project Rules

- Use JAVA as programming Language.
- Develop a backend using python falcon framework.
- Use a relational database (mysql-alchemy).
- Use docker to create the required services.
- Everything must be public in GitHub.

Exan Rules

- Coding based exams using computer.
- You must solve the task and present the results using a pull request.
- Students are allowed to check internet.

Homework

Task A: Install Java Development Kit (JDK)

- Open terminal.
- Check if you have the Java Developer Kit

```
java -version
```

- If not, download the Java Development Kit.

Task B: Install Android Studio

- Download Android Studio.
- Accept terms and conditions.
- Install Android Studio.
- Standard Install Type.
- Select the desirable theme.

That's all

QUESTIONS?

About me

www — jordimateofornes.com
github — github.com/JordiMateo
twitter — @MatForJordi
gdc — Distributed computation group