Mobile Applications

Lab1. Project kick-off

Boni García

boni.garcia@uc3m.es

Telematic Engineering Department
School of Engineering

2023/2024

uc3m | Universidad Carlos III de Madrid

Table of contents

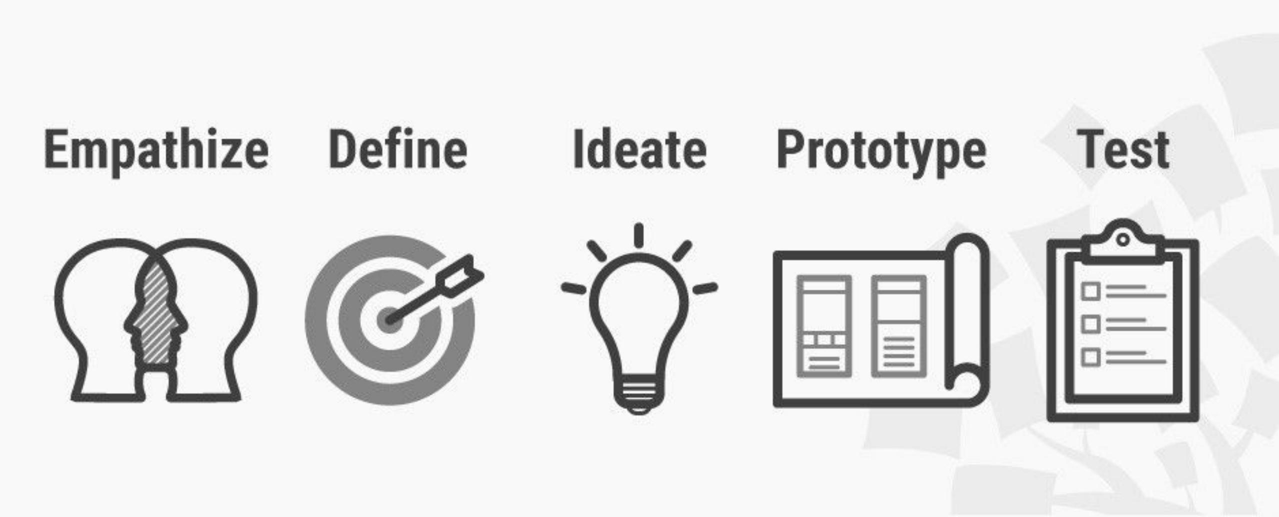
1. Introduction
2. Methodology
3. Requirements
4. Milestones
5. Evaluation
6. Takeaways

1. Introduction

- The practical part of this course is focused on developing a mobile application (app) for Android
  - We refer as this practical part as the **lab project**
  - You will work in the development of this lab project in groups of three people (preferably)
  - The first step is that you define the members of each group
- You can find the **Statement of Work** (SOW) for the lab project in Aula Global
- This lab session is aimed to present the lab project, the methodology for working in the lab, the requirements for developing the app, the project milestones, and the evaluation rubrics

2. Methodology

- We will use a simplified version of the "design thinking" methodology
- **Design thinking** is a popular way for designing software applications
  - It is a iterative process that teams use to understand users, challenge assumptions, redefine problems and create innovative solutions
  - It is usually defined in five phases:



© Teo Yu Siang and Interaction Design Foundation, CC BY-NC-SA 3.0

2. Methodology

1. Empathize: Research your users’ needs
2. Define: State your users’ needs and problems
3. Ideate: Challenge assumptions and create ideas
4. Prototype: Start to create solutions
5. Test: Try your solutions out

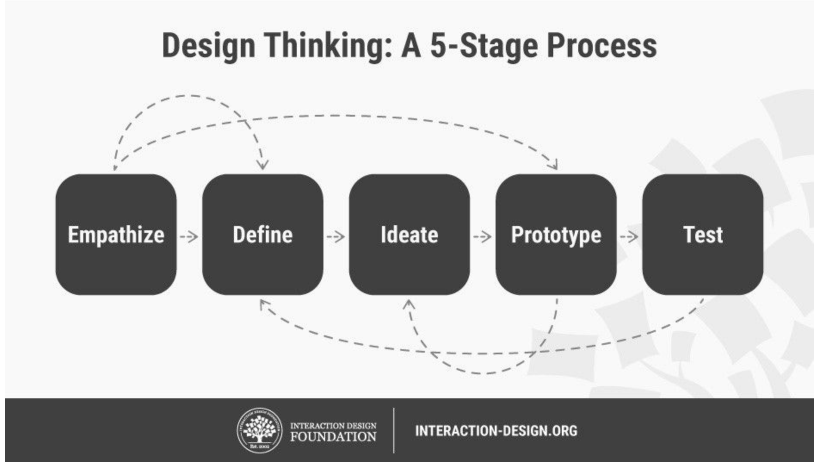
Learn about audience

Construct point of view based on user needs

Come up with some solutions

Build representation of your ideas

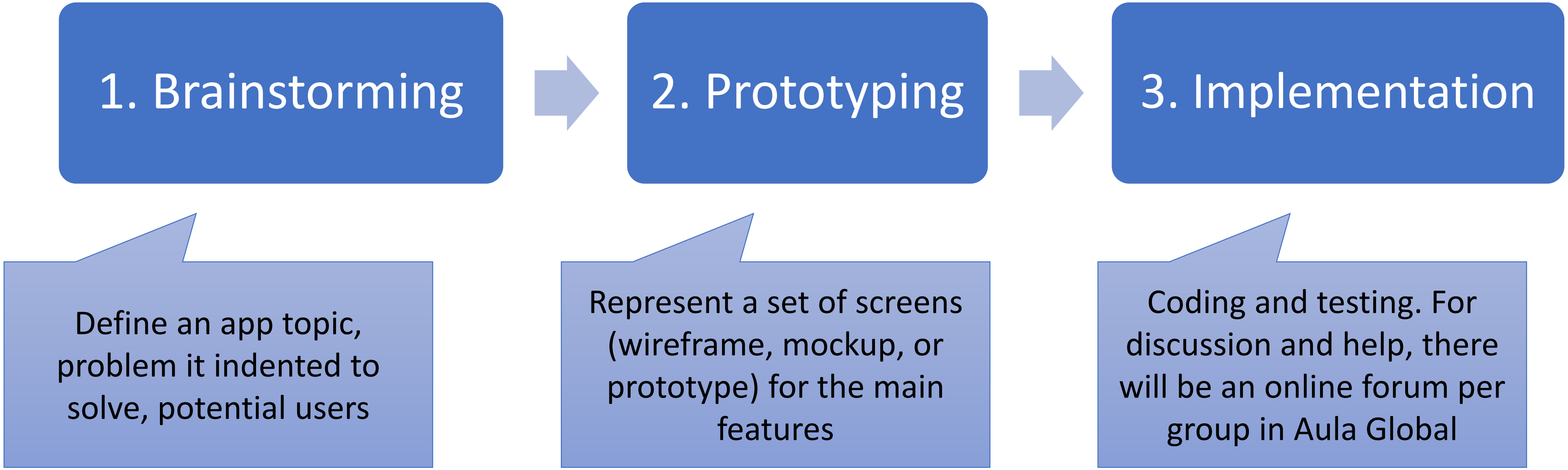
Implement and get feedback


Design Thinking: A 5-Stage Process
Empathize
Define
Ideate
Prototype
Test
INTERACTION DESIGN FOUNDATION
INTERACTION-DESIGN.ORG

© Teo Yu Siang and Interaction Design Foundation, CC BY-NC-SA 3.0

2. Methodology

- In this course, the proposed methodology is as follows:



1. Brainstorming

2. Prototyping

3. Implementation

Define an app topic, problem it indented to solve, potential users

Represent a set of screens (wireframe, mockup, or prototype) for the main features

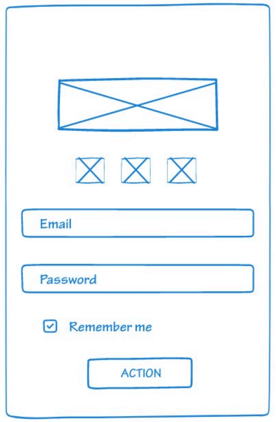
Coding and testing. For discussion and help, there will be an online forum per group in Aula Global

2. Methodology - Brainstorming

- The group members should discuss and agree on the app to be developed
- The app topic is unrestricted (e.g., health, sports, education, etc.)
- To define the idea, each group should discuss and agree the app objective, potential users, main features, etc.
- Optionally, you can visit a mobile application store (e.g., Google Play) and check if there is already a similar app. This way, you can see related functionalities to define your idea
- When an app topic is agreed, some initial prototype should be done

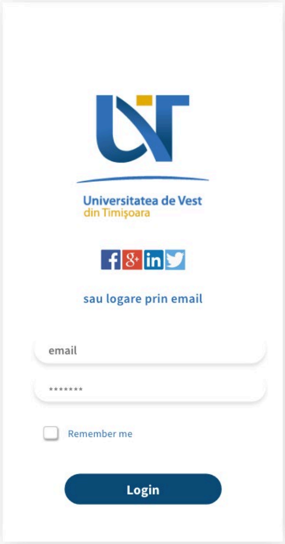
2. Methodology - Prototyping

- An initial design (i.e., a set of screens) should be created before starting the implementation part
- Depending on the level of details the prototyping, we distinguish between:



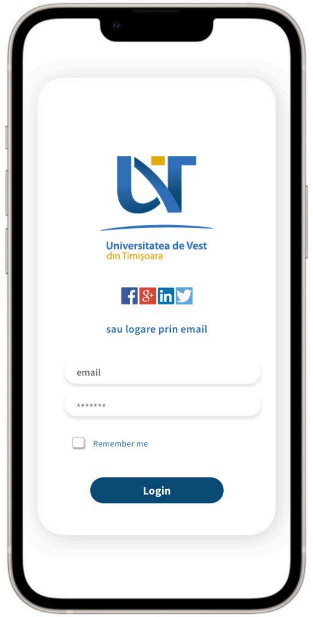
wireframe

quick sketch



mockup

realistic visual design



prototype

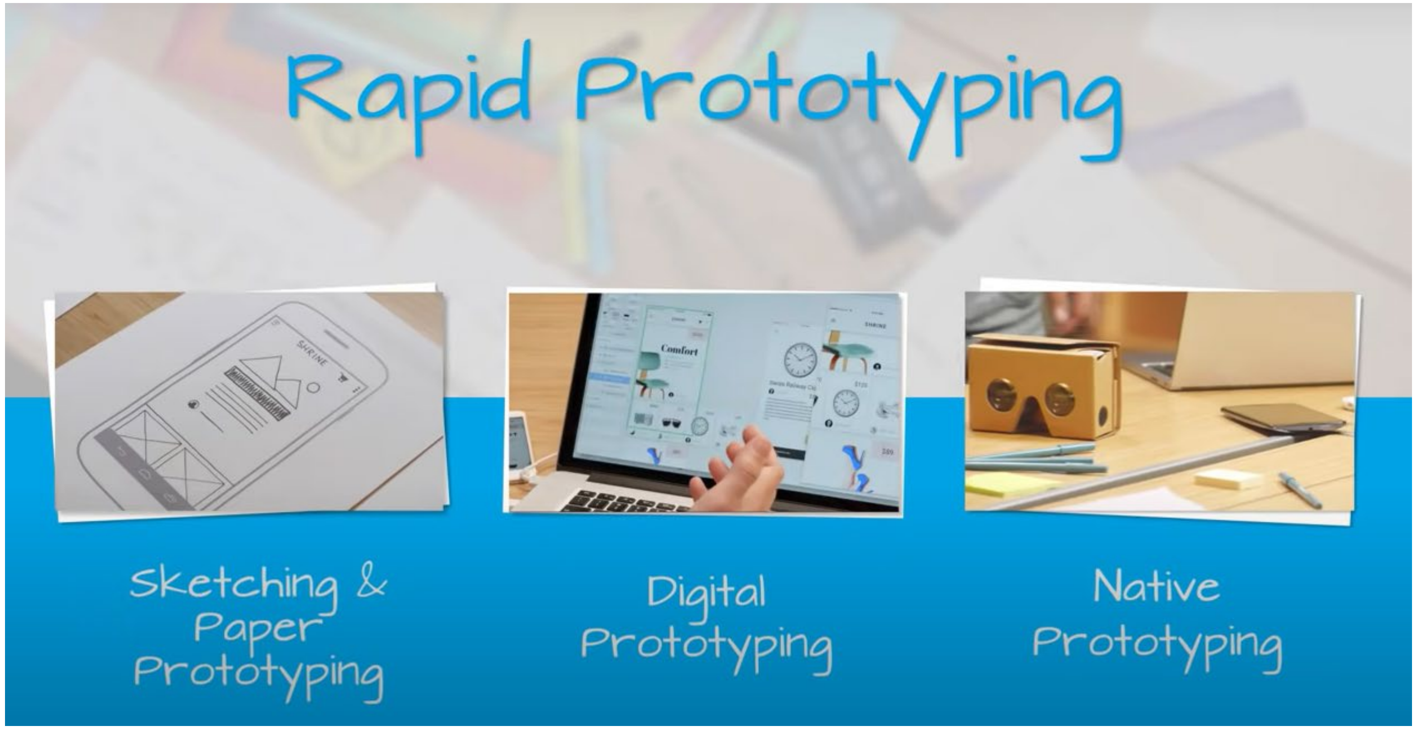
interactive simulation

2. Methodology - Prototyping

- There are plenty of tools for wireframing, mocking, and prototyping, e.g.:
  - Pencil Project: https://pencil.evolus.vn/
  - Quant-UX: https://www.quant-ux.com/
  - Draw.io: https://drawio-app.com/
  - PenPot: https://penpot.app/
  - Wondershare Mockitt: https://mockitt.wondershare.com/
  - Wireframe.cc: https://wireframe.cc/
  - FluidUI: https://www.fluidui.com/
  - Wireflow: https://wireflow.co/
  - Cacoo: https://cacoo.com/
  - Figma: https://figma.com/
  - Adobe XD (eXperience Design): https://helpx.adobe.com/support/xd.html

2. Methodology - Prototyping

- Additional resources about prototyping:



https://youtu.be/JMjozqJS44M https://youtu.be/lusOgox4xMI

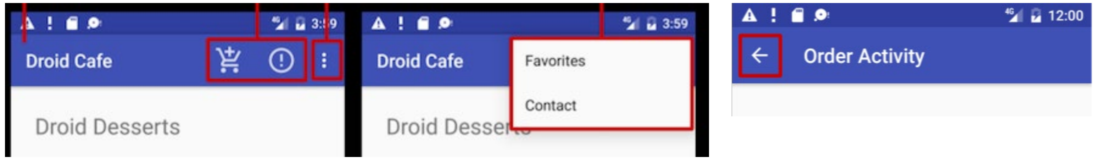
https://youtu.be/KWGBGTGryFk

3. Requirements

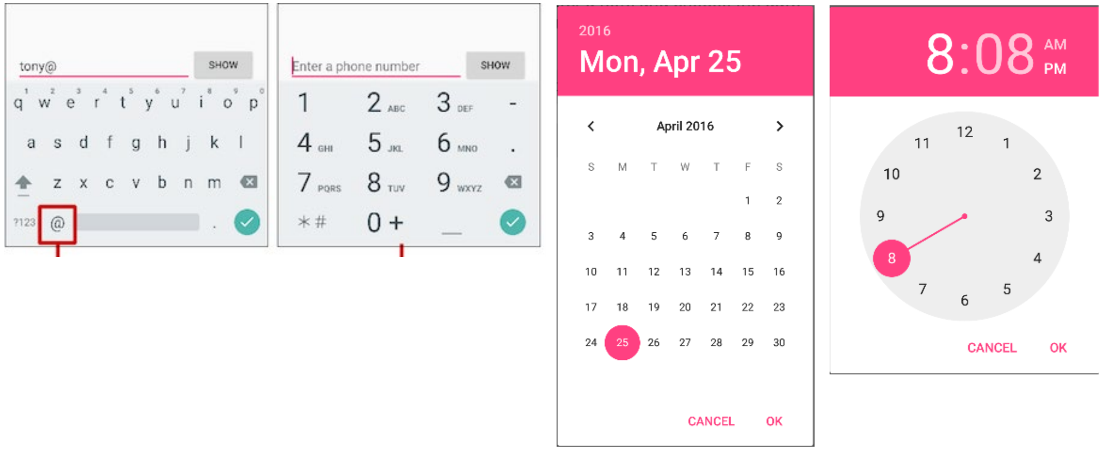
- The app must have a **User Interface** (UI)
- It must be possible to deploy and execute the app in an **Android Virtual Device** (AVD) we can do it in a real device but it must run correctly in a virtual one
- The app must meet some **main features** of at least two of the following units seen during the course:
  - Persistent storage
  - Maps or location services
  - Interaction with external services
  - Background services, notifications, or alarms
- The app might meet some **secondary features**, e.g. using third-party libraries, deployment in physical devices, automated tests, or other

3. Requirements - User interface

- For proposing your idea, you can think of using basic elements, e.g.:
  - Menus, bar, buttons, …

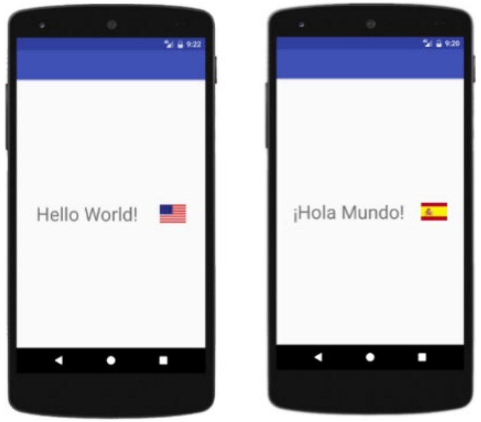


  - Controls for text input, date pickers, …

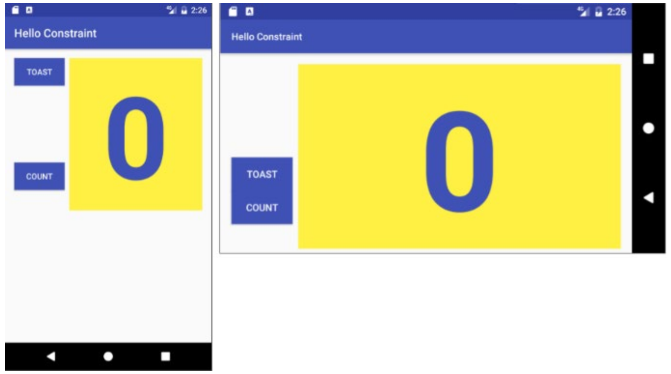


3. Requirements - User interface

- Also, you can think in more advance features such as:
  - Multilanguage

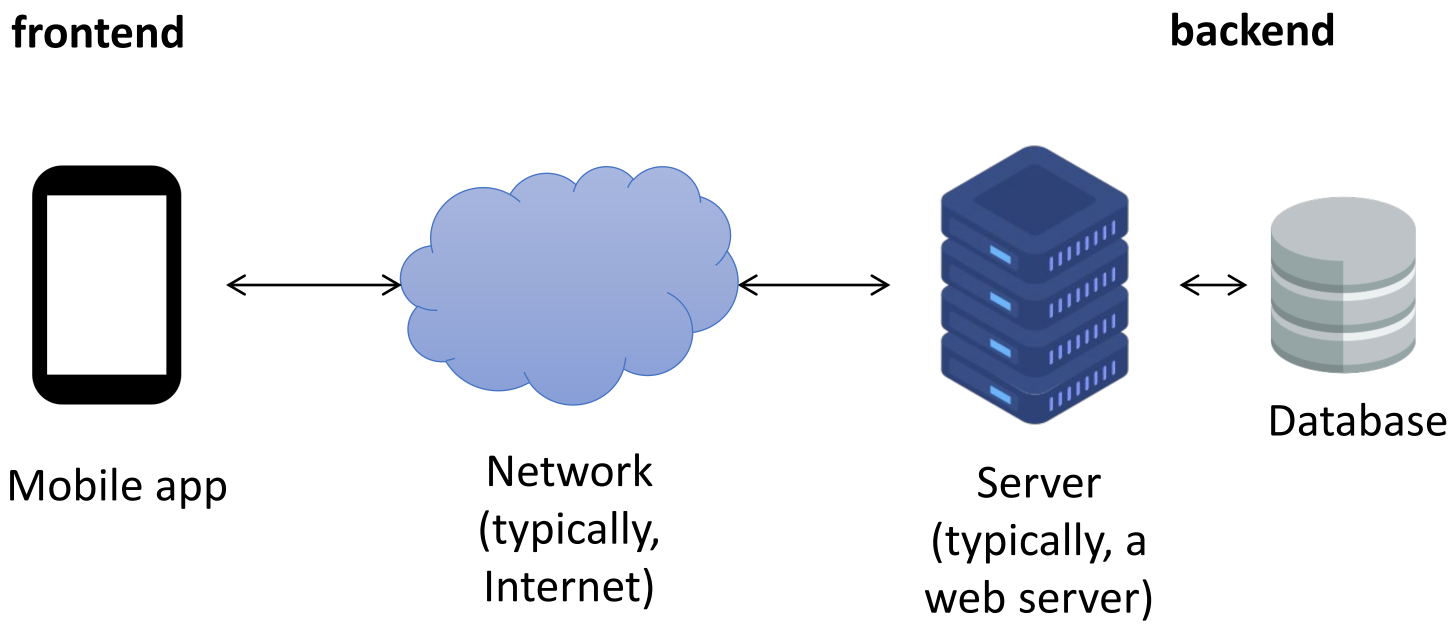


  - Responsive/adaptable



3. Requirements - Persistence storage

- Like web applications, the typical architecture or a mobile app also has a server-side (backend)
  - Most of our work should be on the frontend (client-side)
  - Although this year, we also see Firebase for remote storage


frontend
backend
Mobile app
Network (typically, Internet)
Server (typically, a web server)
Database

3. Requirements - Persistence storage

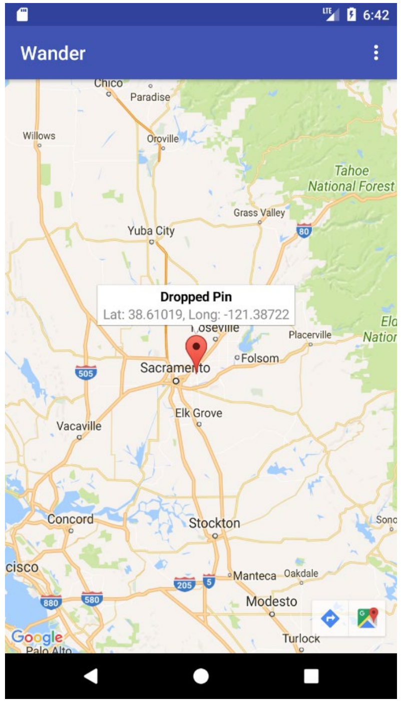
- We will learn how to make persistence storage in our apps using:
  - Preferences
    - Profile, settings
  - Files
    - Regular files stored in the internal or external memory
  - Local database
    - Structured data (SQLite) stored locally
  - Remote database
    - Backend (Firebase) stored in a remote server (in the “cloud”)

Local storage

Remote storage

3. Requirements - Maps and localization

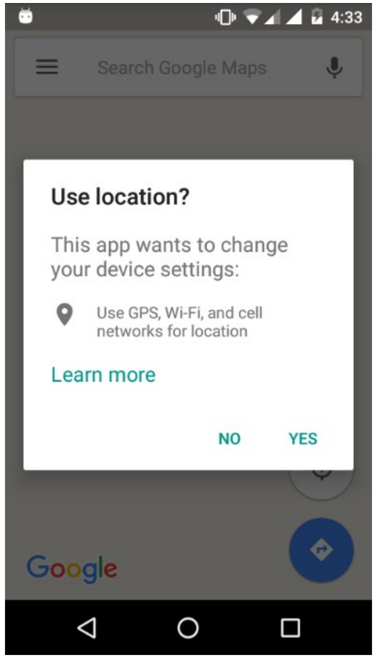
- The Google Maps API (Application Programming Interface) can be used to:
  - Rendering a map
  - Move location to some coordinates
  - Put markers into the map



Dropped Pin
Lat: 38.61019, Long: -121.38722

3. Requirements - Maps and localization

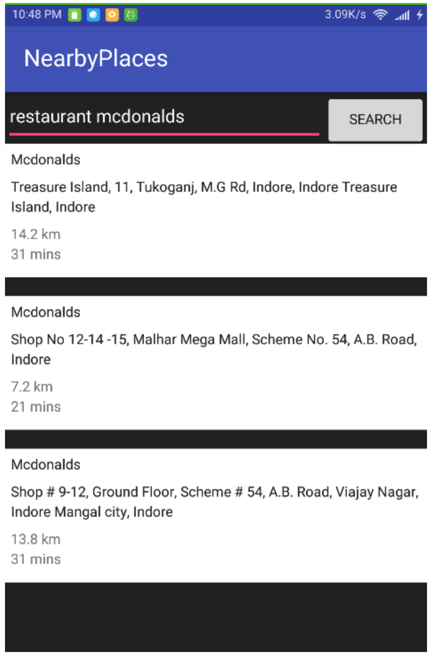
- Localization services involve:
  - Get current device coordinates
  - Geocoding (convert address/name to coordinates)
  - Find nearby places



Use location?

This app wants to change your device settings:

Use GPS, Wi-Fi, and cell networks for location



Mcdonalds

Treasure Island, 11, Tukoganj, M.G Rd, Indore, Indore Treasure Island, Indore

14.2 km
31 mins

Mcdonalds

Shop No 12-14 -15, Malhar Mega Mall, Scheme No. 54, A.B. Road, Indore

7.2 km
21 mins

Mcdonalds

Shop # 9-12, Ground Floor, Scheme # 54, A.B. Road, Viajay Nagar, Indore Mangal city, Indore

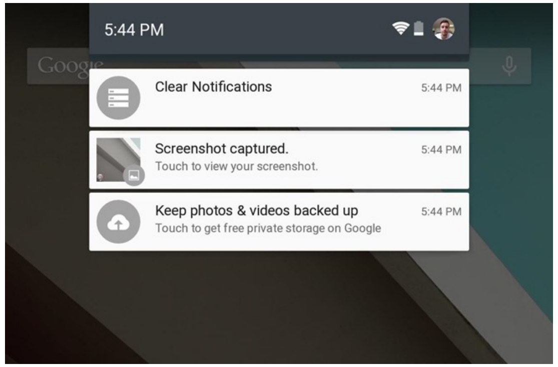
13.8 km
31 mins

3. Requirements - External services

- Apps usually require to connect to an external service to get some data (e.g., news, weather forecast, cryptocurrency market, …)
  - These services are typically supported on top of HTTP (web services)
  - The exchanged data use data-interchange format such as JSON, YAML, or XML
- Some examples of external services:
  - Wikipedia: https://www.mediawiki.org/wiki/API:Main_page
  - Google Books: https://developers.google.com/books/docs/overview
  - Google Places: https://developers.google.com/places/web-service/intro
  - AEMET: https://opendata.aemet.es/centrodedescargas/inicio
  - Open Data European Union: https://data.europa.eu/euodp/en/home

3. Requirements - Other features

- Also, we will about
  - Background services
  - Notifications
  - Alarms



5:44 PM

Clear Notifications 5:44 PM

Screenshot captured. 5:44 PM
Touch to view your screenshot.

Keep photos & videos backed up 5:44 PM
Touch to get free private storage on Google

- If you need it, you can also use other Android services, such as:
  - User media access (microphone, camera)
  - Sensors (accelerometer, gyroscope, etc.)
  - Connectivity (SMS, WiFi, etc.)
  - Other services

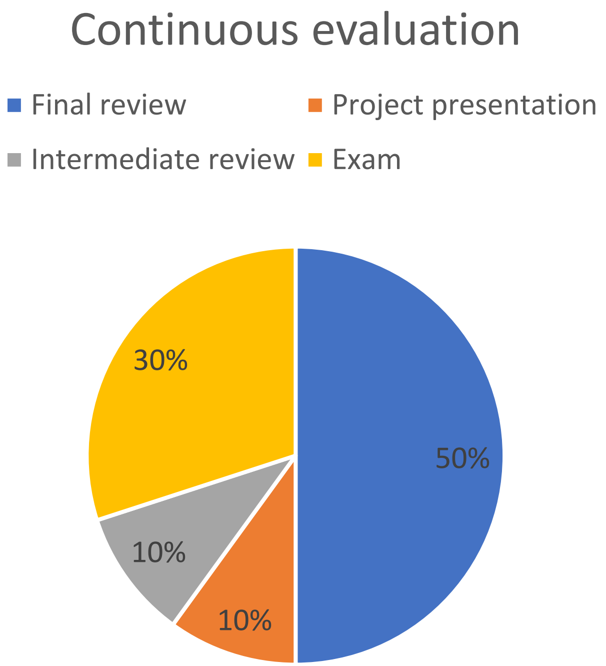
3. Requirements - Code reutilization

- Code reutilization from external sources is allowed
- An external source can be:
  - Code from the Web (e.g., StackOverflow)
  - Code from public repositories (e.g., GitHub)
  - Code from an GenAI (e.g., ChatGPT, GitHub Copilot)
  - Code from private projects (e.g., from a friend)
- The requirement to reuse code from the external source are:
  - Acknowledge the authorship of the original creator (i.e., name the source, URL, etc.) as comments in the source code
  - If acknowledgement is not given, it will be considered as plagiarism
    - Plagiarism is evaluated with an score of 0

4. Milestones

1. Project presentation (week 3)
   - Each group makes a presentation of 5 minutes to explain the idea of the application, focused on its functionality
2. Intermediate review (week 8)
   - Each group shows the development status in Android Studio and explains the roadmap for the rest of the course
3. Final review (week 15)
   - Each group shows the final app explaining the main and secondary features

See details in SOW



5. Evaluation - Milestone 1

- Milestone 1 (project presentation) will be evaluated as follows (max 10 points):

Category	**Item**	**Max**
Presentation	Speech clarity and distribution (each member should present a part)	1
	Timing (max 5 minutes) 5 mins en total	1
Submission	Slides (clarity, layout)	2
	Proposal (app idea, main features, secondary features)	2.5
	Prototype (set of screens for the main features)	1.5
	Initial roadmap (milestones, schedule, distribution of work)	1.5
	Strategy for collaborative development (e.g., using GitHub or other mechanism)	0.5

Each one of these items will be scored as: excellent (max*1), good (max*0.66), fair (max*0.33), or poor (0)

If a student does not participate in the presentation, his/her milestone will be scored as 0

5. Evaluation - Milestone 2

- Milestone 2 (intermediate review) will be evaluated as follows (max 10 points):

Category	Item	Max
Interview	Explanation of the app status	1
	App execution during the interview	1
Submission	Report (max 8 pages, structure, clarity, conciseness)	1.5
	A portion of the User Interface (layout, usability)	2.5
	Detailed roadmap (main and secondary features, work distribution, way to work collaboratively) for the rest of course	2
	Source code (readability, code format, CamelCase, don't repeat yourself, avoid hardcoded strings)	1.5
	App deployment (import and execute in AVD)	0.5

Each one of these items will be scored as: excellent (max*1), good (max*0.66), fair (max*0.33), or poor (0)

If a student does not participate in the interview, his/her milestone will be scored as 0

If the code does not compile or a copy is detected, the milestone of all members will be scored as 0

5. Evaluation - Milestone 3

- Milestone 3 (final review) will be evaluated as follows (max 10 points):

Category	Item	Max
Interview	Explanation of the app status	1
	App execution during the interview	1
Submission	Report (structure, clarity, conciseness)	1
	Promotional video (5 minutes max, clarity)	1
	User Interface (layout, usability)	1
	Main features (persistence, maps, external services, etc.)	2
	Secondary features (automated tests, third-party libraries, etc.)	1
	Source code (readability, code format, CamelCase, don't repeat yourself, avoid hardcoded strings)	1
	Distribution of work (task leaders, balanced workload)	0.5
	App deployment (import and execute in AVD)	0.5

Each one of these items will be scored as: excellent (max*1), good (max*0.66), fair (max*0.33), or poor (0)

If a student does not participate in the interview, his/her milestone will be scored as 0

If the code does not compile or a copy is detected, the milestone of all members will be scored as 0

6. Takeaways

- Mobile Applications is a very practical course, in which the lab project (i.e., the development of and Android app in groups) is the 70% of the continuous evaluation score
- There will be three milestones for the lab project. Each milestone is composed by a face-to-face activity (to be done in the lab) and a submission to be done in Aula Global
  - 1st milestone. Each group will present their app to the rest of the class. The submission will be the slides used for the presentation
  - 2nd milestone. Each group will show the app status (at least, some UI and a roadmap for the rest of the course). The submission will be a report
  - 3rd milestone. Each group will show the final status. The submission will be a report (containing a link to the source code and a promotional video)