University of Padova
Department of Information Engineering

Biomedical Wearable Technologies
for Healthcare and Wellbeing

Setup the environment

A.Y. 2022-2023

Giacomo Cappon

We need some tools in our belt

- Developing mobile apps requires some tools

- As programmers, we need to setup our **development environment** in order to be able to write code, compile it, test its behaviour, and deploy it to the final user machine (in this case, a phone).

- To do so, we need to:
  - Use some software to write the actual code
  - Choose a framework and the respective programming language
  - Have specific libraries in place to support the phone operating system
  - Have some tools to be able to work as a team

- This document will give an overview of the development environment we are going to use during this course and will tell what to do to prepare it.

The environment: Overview

IDE

(To write code, compile, and test)

Flutter + Dart

(The framework and its programming language)

Android Studio

(For Android support)

XCode

(For iOS support)

VCS

(For version control and to enable teamwork)

The environment: IDE

- The first component of the environment is the IDE (Integrated Development Environment).

- The IDE is where we actually will write the code: it is a text editor with some flavour (high-level functionalities).

- The IDE of choice in this course is Visual Studio Code (VS Code)

The environment: Framework and compiler

- The second component of the environment is, of course, the framework (and the programming language) we are going to use to develop mobile apps.

- We will use Flutter: a brand-new framework by Google based on the Dart programming language.

- Why Flutter? Because it allows us to **write a single code base and compile to either iOS or Android**. This means that:
  - We will build one app that will look the same in both iOS and Android
  - We will not be constrained by the operating system (OS)
  - Developing time is halven

The environment: OS support

- The third component of the environment are the OS-specific (iOS or Android) libraries to install in order to let Flutter do its magic and compile.

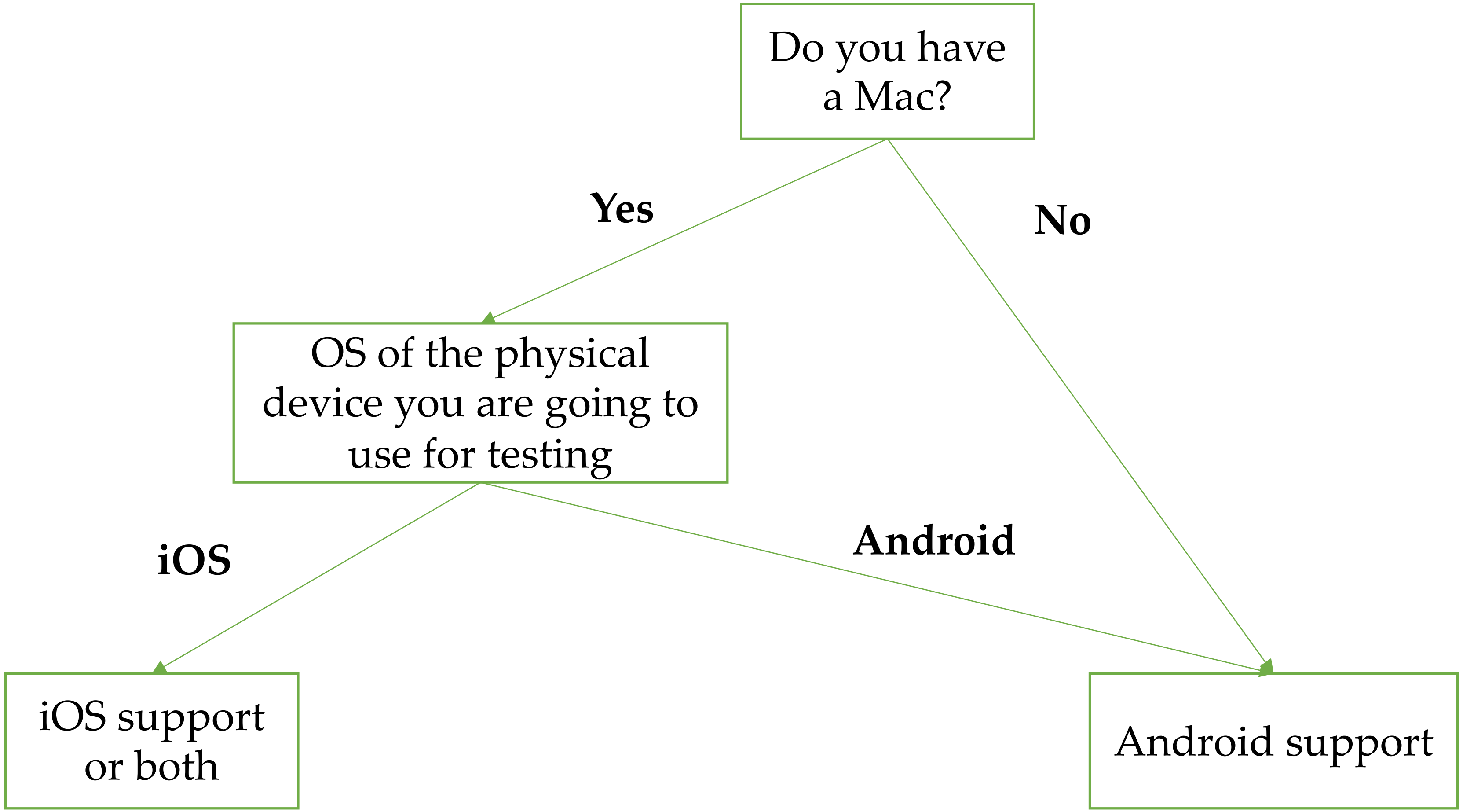
- Both the iOS and the Android libraries will install the compiler for Flutter and a virtual phone simulator to allow you to test the mobile app without actually having a physical device.

- Since Flutter is OS-agnostic, you have a choice here:
  - Install iOS support
  - Install Android support
  - Install both

- A good way to chose the best option is: ok, I will have a virtual device, but at some point I would like to deploy my app to an actual phone. So, which physical device I am going to use?

- Note that if you have an iPhone but you do not have a Mac, you need to go for Android (XCode is not available for PC)

iOS or Android support?



The environment: VCS

- The final component of the environment is the Version Control System (VCS)

- As you will learn in the first lab lesson, the VCS is a software that allows to maintain and manage the various version of the code your are going to write and it will be fundamental to work as a team.

- In this course, we will use GIT, the most famous VCS.

The environment: Git-Bash (for Windows users)

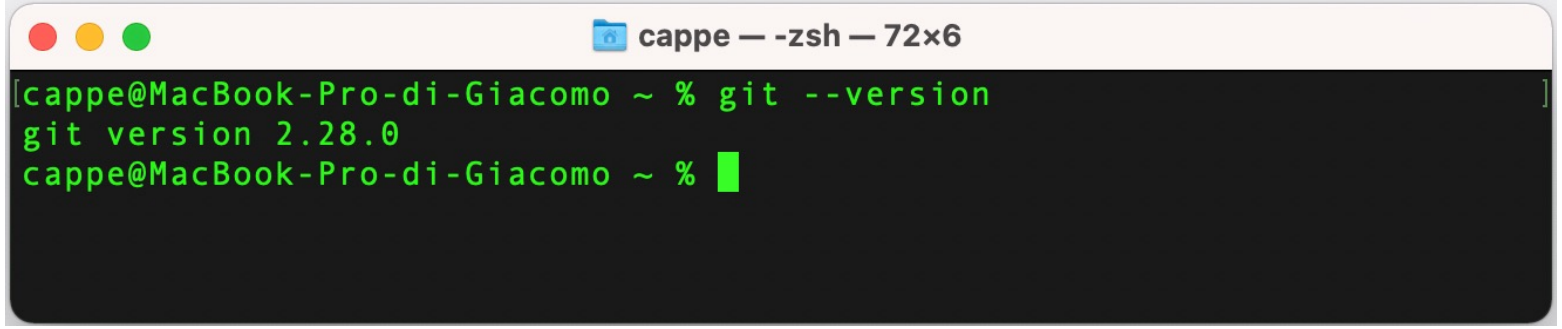
- You will work with GIT from the command line:
  - ″cmd″ in Windows
  - “Terminal” in UNIX systems (Mac and Linux)

- Some commands differs between cmd and Terminal. So to “unify” this set of commands, Windows users should install and use **git-bash** instead of cmd.

- Using git-bash you will be able to use the same commands of Terminal.

The environment: VCS

- First check if it is already installed (~99% probability)
  - Open the Terminal > Type “`git --version`”



```
[cappe@MacBook-Pro-di-Giacomo ~ % git --version                    ]
git version 2.28.0
cappe@MacBook-Pro-di-Giacomo ~ % 
```

- If an error appears you will need to install it (step 4 of slide 11). Otherwise, if you see the git version printed out, you can skip step 4 of slide 11.

Install everything (Part 1)

- Do the following steps, in this order:
  - **Step 1**: Install Flutter and Dart
    - Go to > https://flutter.dev/docs/get-started/install
    - Follow the instructions…
  - **Step 2**: Install OS support
    - Same link as Step 1, just go ahead with the instructions until the end
  - **Step 3**: Install VS Code and integrate it with Flutter and Dart
    - Go to > https://docs.flutter.dev/get-started/editor?tab=vscode
    - Follow the instructions…

Install everything (Part 2)

- Do the following steps, in this order:

  - **Step 4**: Install GIT
    - Go to > https://git-scm.com/book/en/v2/Getting-Started-Installing-Git
    - Follow the instructions…
    - **Step 4b (for Windows users only)**: Install GIT Bash
      - Go to > https://gitforwindows.org/
      - Click the Download button
      - Download and install Git-2.35.1.2-64-bit.exe
    - After the installation, open the terminal and run the two following commands:
      `git config --global user.name ”FirstName LastName”` (where FirstName and LastName are your actual first name and last name, e.g., Giacomo Cappon)
      `git config --global user.email ”email@domain.com”` (where `email@domain.com` is the email you want to use as identifier. For simplicity, use the same email you will use to create the GitHub account)

  - **Step 5**: Test that everything is working
    - Go to > https://docs.flutter.dev/get-started/test-drive?tab=vscode
    - Follow the instructions…