**WEEK 1:**

**MODULE 1: INTRODUCTION TO MOBILE DEVELOPMENT**

**OVERVIEW:**

* Hybrid and Native apps
* Software installation
* Version control

**Hybrid and Native Applications**

## **What is a Hybrid app?** A hybrid app is a regular mobile site, “wrapped” in its native built-in mobile OS browser, and working as an application.

Inside such an application is a website adapted for a mobile device. Through native operating system components, it can access push notifications and geolocation, and it can be distributed through app stores.

A hybrid application is more suitable for those who already have ready-made sites adapted for mobile devices, and who want to quickly check whether a full-fledged mobile app will be in demand among customers.

Advantages:

1. Lower cost compared to native apps;
2. Fast development;
3. Native APIs;
4. One codebase;
5. Maintenance.

### Disadvantages:

1. Connection limitations;
2. Restriction on the use of the device function;
3. Platform differences entail additional improvements;
4. User Experience (UX) is worse than in native apps;
5. Lower performance.

## What is a Native app?

A native app is a type of software that is developed for a specific operating system (OS) installed on the gadget such as Android, iOS, etc. Native applications use the hardware functionality of a specific OS: camera, geolocation, voice recorder, etc. In addition, native mobile apps have access to device services: audio and video files, notifications, calendar, etc. If desired, the user includes push notifications, which are also supported.

To build a native iOS app, Swift or Objective-C will be used. For native Android apps, Kotlin or Java are suitable. For crossplatform , Flutter , React Native are used.

### Advanages:

1. Easy to deploy;
2. The application works more efficiently;
3. Higher performance;
4. Easy user interface customization;
5. Safe and reliable;
6. Quality functionality and user experience.

### Disadvantages:

1. Expensive development cost;
2. Unique source code prohibits reuse;
3. The higher cost of service;
4. It takes a lot of time to develop compared to cross-platform.

**Software Installation**

## **System requirements for windows**

To install and run Flutter, your development environment must meet these minimum requirements:

* Operating Systems: Windows 7 SP1 or later (64-bit), x86-64 based.
* Disk Space: 1.64 GB (does not include disk space for IDE/tools).
* Tools: Flutter depends on these tools being available in your environment.
  + [Windows PowerShell 5.0](https://docs.microsoft.com/en-us/powershell/scripting/install/installing-windows-powershell) or newer (this is pre-installed with Windows 10)
  + [Git for Windows](https://git-scm.com/download/win) 2.x, with the Use Git from the Windows Command Prompt option.  
    If Git for Windows is already installed, make sure you can run git commands from the command prompt or PowerShell.

## **System requirements for macOS**

To install and run Flutter, your development environment must meet these minimum requirements:

* Operating Systems: macOS
* Disk Space: 2.8 GB (does not include disk space for IDE/tools).
* Tools: Flutter uses git for installation and upgrade. We recommend installing [Xcode](https://developer.apple.com/xcode/), which includes git, but you can also [install git separately](https://git-scm.com/download/mac).

For installation steps: <https://docs.flutter.dev/get-started/install/>

**Version Control**

Version control is a system that records changes to a file or set of files over time so that you can recall specific versions later.

Types

### **Local Version Control Systems:** A local version control system is a local database located on your local computer, in which every file change is stored as a patch. Every patch set contains only the changes made to the file since its last version**.**

### 

### 

### 

### **Centralized Version Control Systems:** Centralized version control systems are based on the idea that there is a single “central” copy of your project somewhere (probably on a server), and programmers will “commit” their changes to this central copy. “Committing” a change simply means recording the change in the central system.



### **Distributed Version Control Systems:**A distributed version control system (DVCS) is a type of version control where the complete codebase — including its full version history — is mirrored on every developer's computer. It's abbreviated DVCS. Changes to files are tracked between computers. For example, my workstation and yours.

Read more on version c0ntrol:<https://git-scm.com/book/en/v2>