UNIVERSITÉ DES
MASCAREIGNES

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

DEPARTMENT OF SOFTWARE ENGINEERING

BSc (Hons) Software Engineering

Academic Year 2022/2023

Year 3 Semester 1

Mobile Application Development

Setting up your Development Environment

1 Setting up your development environment

In this module we are going to use React Native to create mobile applications. React Native is an open source mobile application framework created by Facebook.

1.1 Node.js LTS Release

Node is a platform that allows developers to write JavaScript outside of a browser environment. Node.js enables us to write server-side applications in JavaScript. Node.js includes a tool called `npm`, which stands for Node Package Manager. `npm` enables us to install thousands of libraries and JavaScript tools within our projects.

1. Download **Node.js** for your platform from https://nodejs.org/en/. The LTS version at the time of writing is 16.17.1.
2. Double-click the installer, follow the prompts, and accept the defaults. By default, Node.js is installed at:

   C:\Program Files\nodejs

   NPM is installed at:

   C:\Program Files\nodejs\npm

The installer should have added the above Node directory to your path. To test the installation, open a command window (make sure you open a **new** command window to get the updated path settings made by the Node.js installation), and type:

```
node --version OR   node -v
```

If the version number is displayed (v16.17.1), Node.js is installed and working!

You can test the `npm` version as well:

```
npm --version  OR   npm -v
```

1.2 Install Oracle JDK 17 LTS (the latest version at the time of writing)

To develop Android apps, you need to install Java and the Android SDK.

To check if Java is installed on your machine type the following at a command prompt:

```
java --version
```

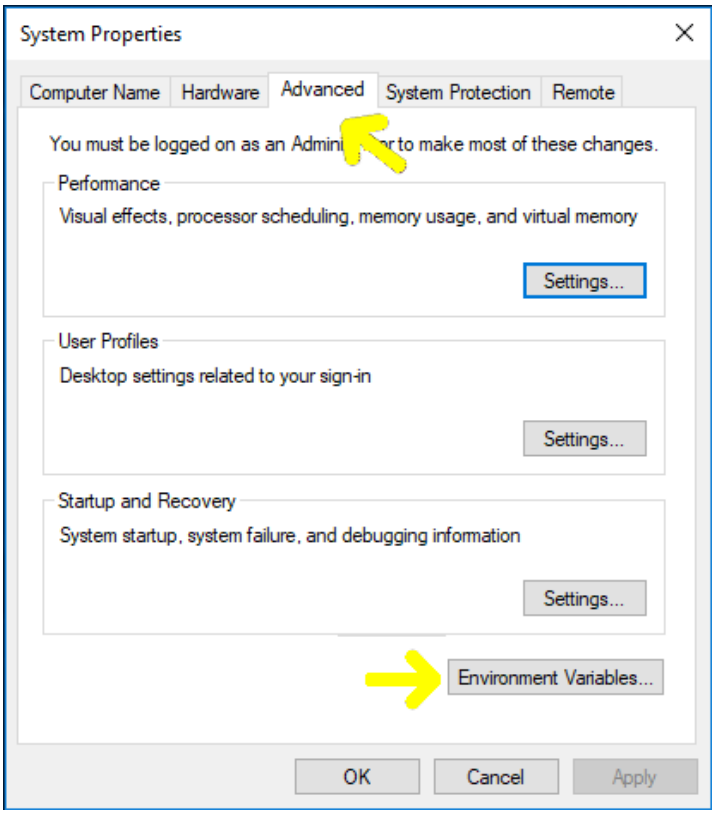
If Java is not installed, go to the Java SE Development Kit 17 Downloads page:

https://www.oracle.com/java/technologies/downloads/

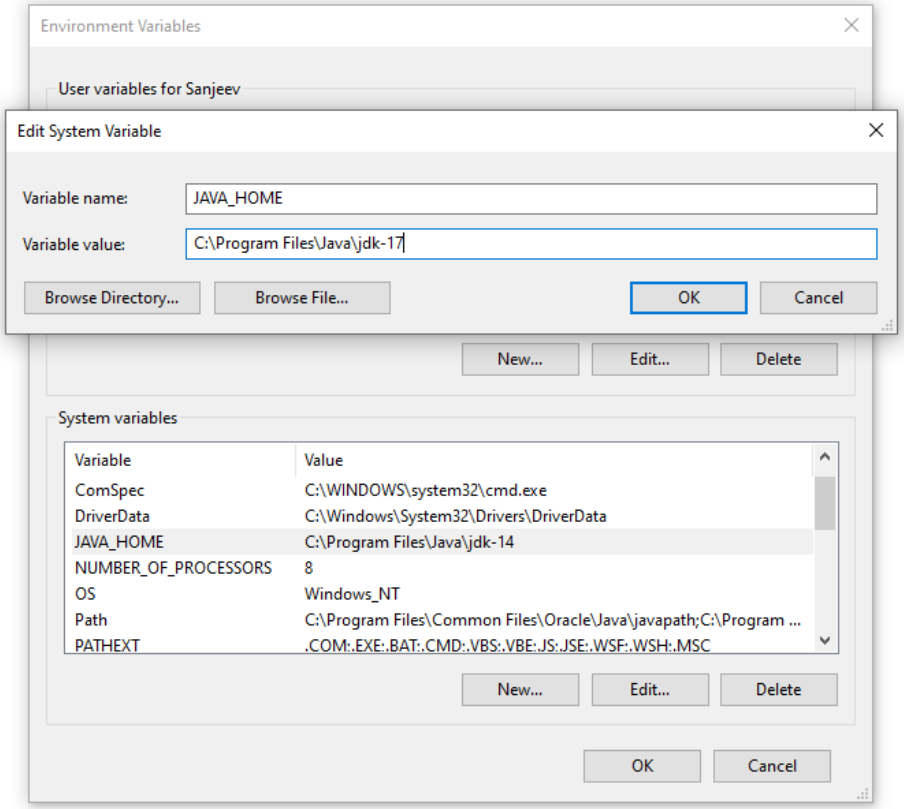
Download the `.exe` file for your version of Windows. Start the JDK 17 installer by double-clicking the installer's icon. Follow the instructions provided by the Installation wizard. The default Java location is **C:\Program Files\Java** and the version will be installed in **C:\Program Files\Java\ jdk-17**.

Setting environment variables:

(a) Go to **Control Panel > System > Advanced > Environment Variables**.



(b) In **System Variables**, click **New...** and set the **Variable Name** to **JAVA_HOME** and the **Variable value** to the location of the installed Java JDK, i.e. `C:\Program Files\Java\jdk-17`.



(c) Next, we add Java's path to the system's **Path** variable. Find **Path** under **System variables**. Click **Edit...** Then click **New**. Write **%JAVA_HOME%\bin.** Click **OK** to accept the changes and then **OK** again to save.

You can verify installation by entering the following command at a command prompt:

```
java --version
```

1.3 Install Android Studio

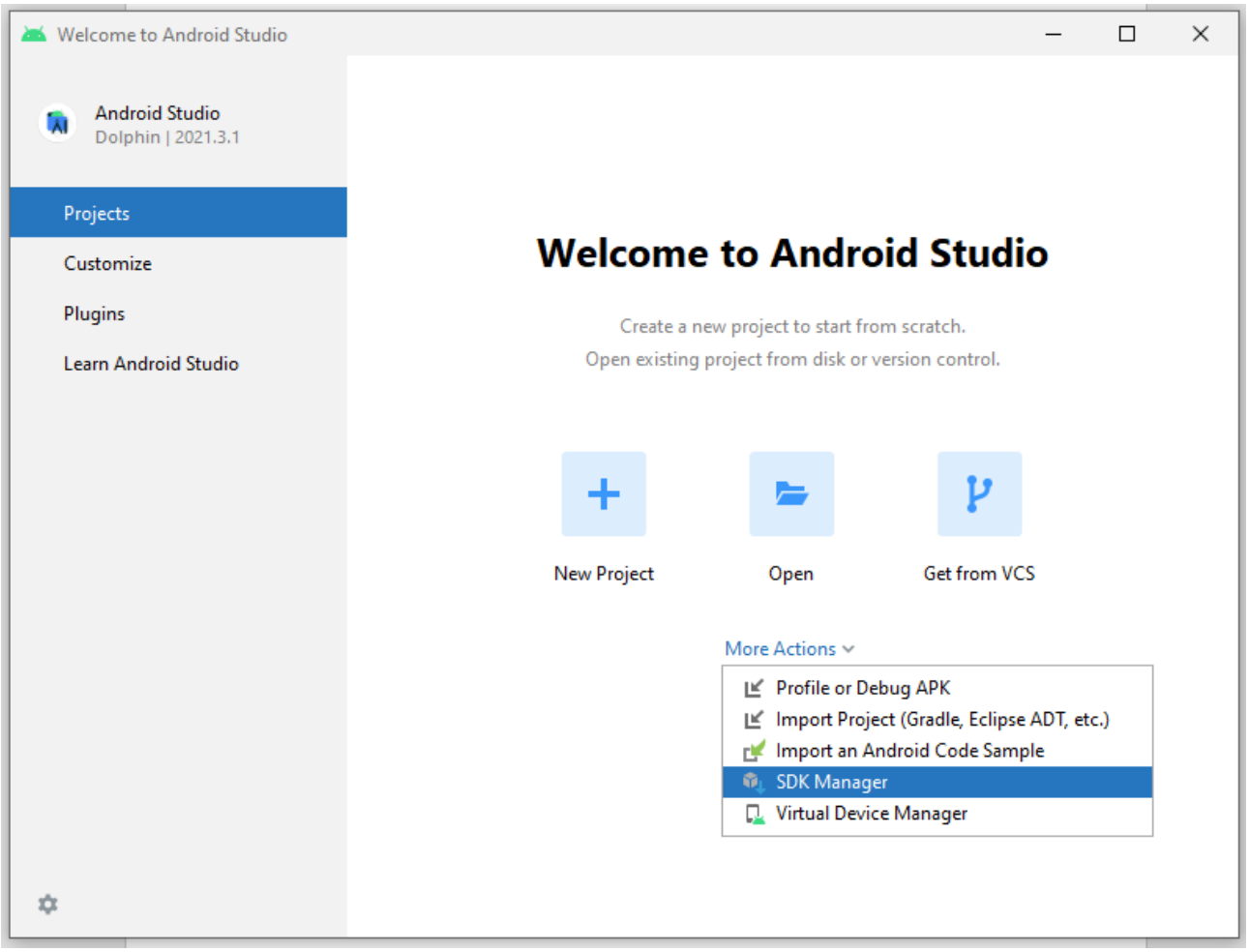
If you want to develop applications for the Android platform, you must install the Android Software Development Kit (SDK) and platform tools. You can accomplish this task by installing Android Studio from

https://developer.android.com/studio/install.

The latest version at the time of writing is version **2021.3.1**. By default, the latest versions of the required packages are installed when you install Android Studio.

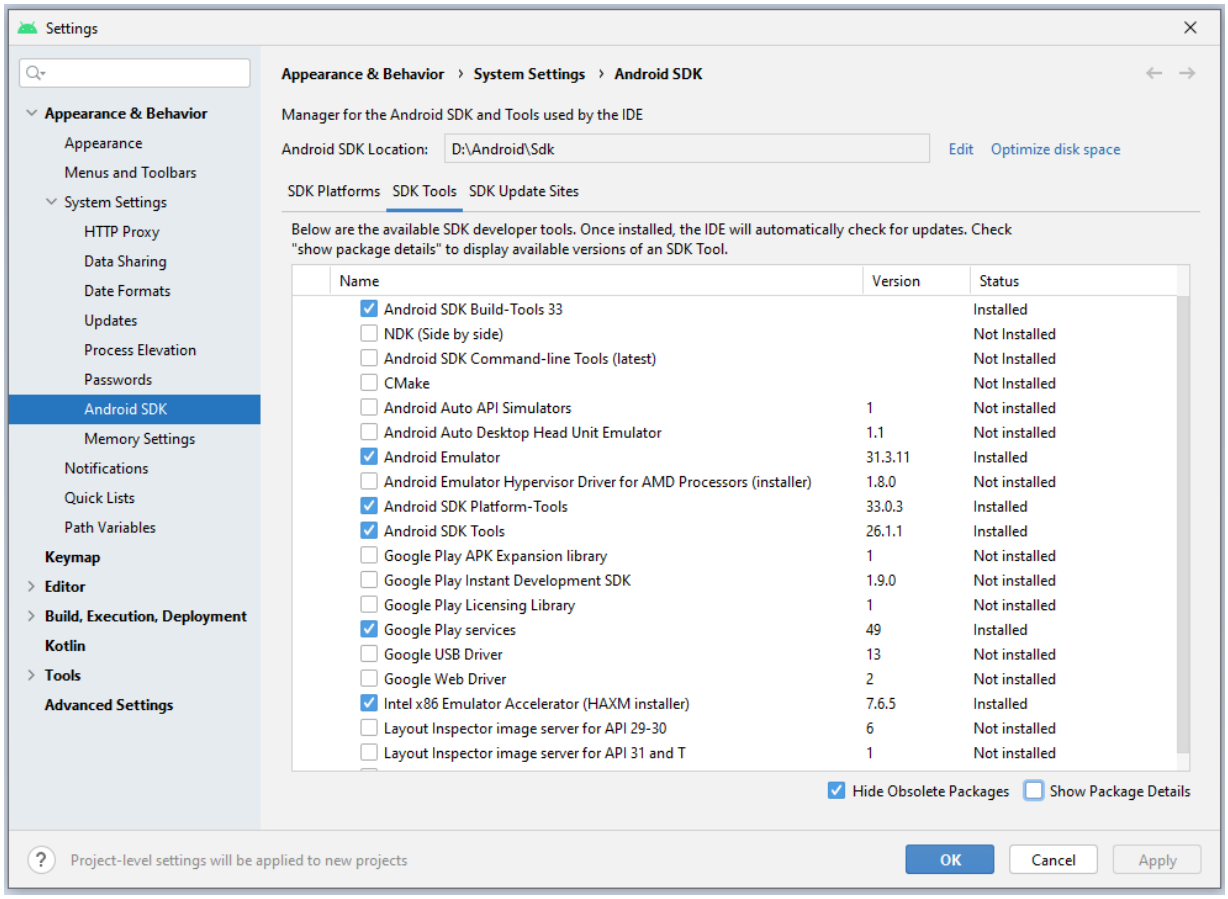
After installing Android Studio, you must also install the packages for whatever API level you wish to target. It is recommended that you install the latest SDK version (The latest stable version at the time of writing is version 13).

Use the Android SDK Manager to install the SDK packages. Open the Android Studio, and click on **More Actions -> SDK Manager**.



Open the **SDK Tools** tab and install the following

- Android SDK Build-Tools 33 (the latest version at the time of writing)
- Android Emulator
- Android SDK Platform-Tools
- Android SDK Tools
- Google Play Services
- Intel x86 Emulator (HAXM Installer)



1.3.1 Set environment variables

Finally, set the ANDROID_HOME environment variable to the location of your Android SDK installation. To do this, click **New…** and set the **Variable Name** to **ANDROID_HOME** and the **Variable value** to the location of the installed Android SDK.

In **System variables**, add the path of the SDK Tools to the **PATH** variable:

%ANDROID_HOME%\tools
%ANDROID_HOME%\platforms
%ANDROID_HOME%\platform-tools

1.4 Creating an Android Virtual Device

To test Android applications on an emulator, you must configure one or more Android virtual devices by using the Virtual Device Manager. For more information about using the Virtual Device Manager, go to

https://developer.android.com/studio/run/managing-avds.html.

In Android Studio, select **More Actions > Virtual Device Manager**. The Device Manager screen appears. Create a new Virtual Device by following the steps that are documented here:

https://developer.android.com/studio/run/managing-avds.html#createavd.

IMPORTANT: The API level of the target device is important, because your app will not be able to run on a system image with an API level which is less than that required by your app.

To learn more about Android API levels, please consult the following web pages:

https://docs.microsoft.com/en-gb/xamarin/android/app-fundamentals/android-api-levels?tabs=vswin

https://support.google.com/googleplay/android-developer/answer/11926878?hl=en

1.5 Expo

Expo is a set of tools, libraries, and services which let you build native iOS and Android apps using JavaScript.

Expo apps are React Native apps which contain the Expo SDK. The SDK is a library which provides access to the device's native features such as the camera, contacts, local storage, push notifications, social authentication, etc. This means that you do not need to write any native code, and it also makes your project very portable because it can run in any native environment containing the Expo SDK.

More information can be obtained from **expo.io** website.

There are two tools which you need to develop apps with Expo: a command line tool (Expo CLI) and a mobile client (Expo Go) to open your app on iOS and Android platforms.

We need to install both the Expo command line tool as well as the Expo app for iOS or Android.

Expo CLI

To install the command line tools, type the following into your terminal:

```
npm install -g expo-cli
```

Type `expo --version` at a command prompt to verify installation.

You can use Expo CLI without installation by leveraging `npx` – a Node package runner which executes Node packages.

Expo Go

Expo Go helps view your projects while you are developing them. Download and install the Expo Go client app on your iOS or Android device. It is available on both the iOS App Store and Android Play Store.

To preview your app on an Android device, use the Expo Go app to scan the QR code from your terminal. On iOS, use the camera app.

You can also preview your app using an iOS or Android simulator on your computer.

1.6 Git

Git is the most popular version control software, allowing you to do things like copy code repositories, merge code with others, and create branches of your own code that do not impact one another. Download from https://git-scm.com/downloads and run the installer.

If required, add the path to the Git **bin** directory to the **PATH** system variable.

To make sure that Git has been installed, type

```
git --version
```

After you install Git, set your user name and email address:

```
git config --global user.name "James Bond"
git config --global user.email bond007@udm.ac.mu
```

1.7 Create a GitHub Account

Create a GitHub account at www.github.com.

1.8 Installing an IDE

Download and install Visual Studio Code from https://code.visualstudio.com/. We will install a few useful VS Code extensions such as Prettier – Code Formatter, Expo Tools, Highlight Matching Tag.

1.9 Developing iOS apps

To develop iOS apps, you need to do the following:

1. Sign up for an Apple account from https://developer.apple.com/

2. To deploy apps on devices (iPhone and iPad) and to publish apps on the AppStore, you should join Apple's iOS Developer Program. This costs 99 USD per year.

3. Install Xcode from: https://developer.apple.com/xcode/downloads.

4. When the install is complete, launch Xcode and enable the command line tools for Xcode:

    - From the **Xcode** menu, select **Preferences...**
    - Select the **Downloads** tab.
    - Under **Components**, locate **Command Line Tools** and click the download icon.