Open Source Engine Integration Installation Guide

Harris SmartWorks
Open Source Engine Integration
Computer Science Dept
COSC 470/471
Okanagan College

Aubrey Nickerson
Bao Mai
Derek Manchee
Joseph Egely
Keaton Canuel

April 29, 2021

Revision Sheet			
Revision	Date	Summary of Changes	Name
Version 1.1	04-22-2021	Baseline Draft	Aubrey Nickerson
Version 1.2	04-23-2021	Install Android Studio section	Aubrey Nickerson
Version 1.3	04-24-2021	Install Cordova section	Aubrey Nickerson

Contents

1 Introduction

The installation guide describes a step by step tutorial on how to install the Open Source Integration app on the emulator and users personal phone. Additional items that require to be assembled are included. How to install Android Studio, Apache Cordova, Git Bash, and install the android app on the users personal device. The user must follow the directions using a windows computer and an android phone.

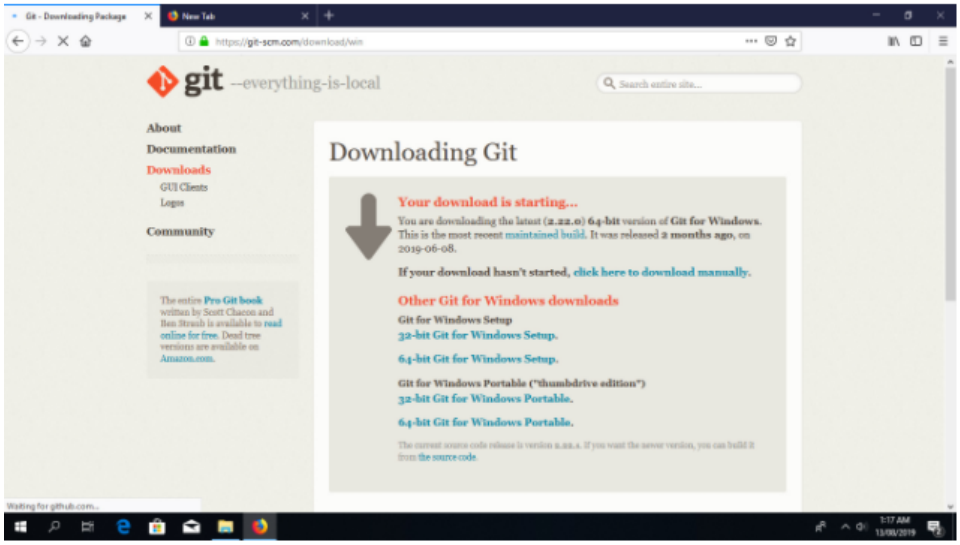
2 Install Git Bash

Git is an open-source version control system for tracking source code changes when developing software. Bash is a Unix command-line shell. It comes with useful Unix commands like ssh, SCP, and so on, which are not usually found on Windows.

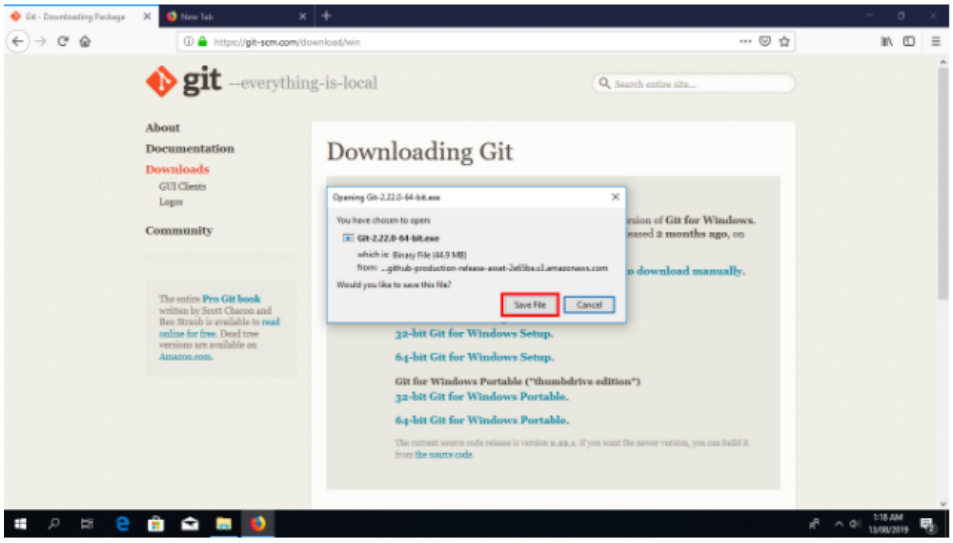
Step 1: Visit the Official Git Bash Website:
Download the latest version of Git Bash from their official website: https://git-scm.com/

Click the “Download for windows” button.

Step 2: Start Git Bash Download:
Next, the user will be redirected to a page that lets the user know that he/she is about to start downloading.



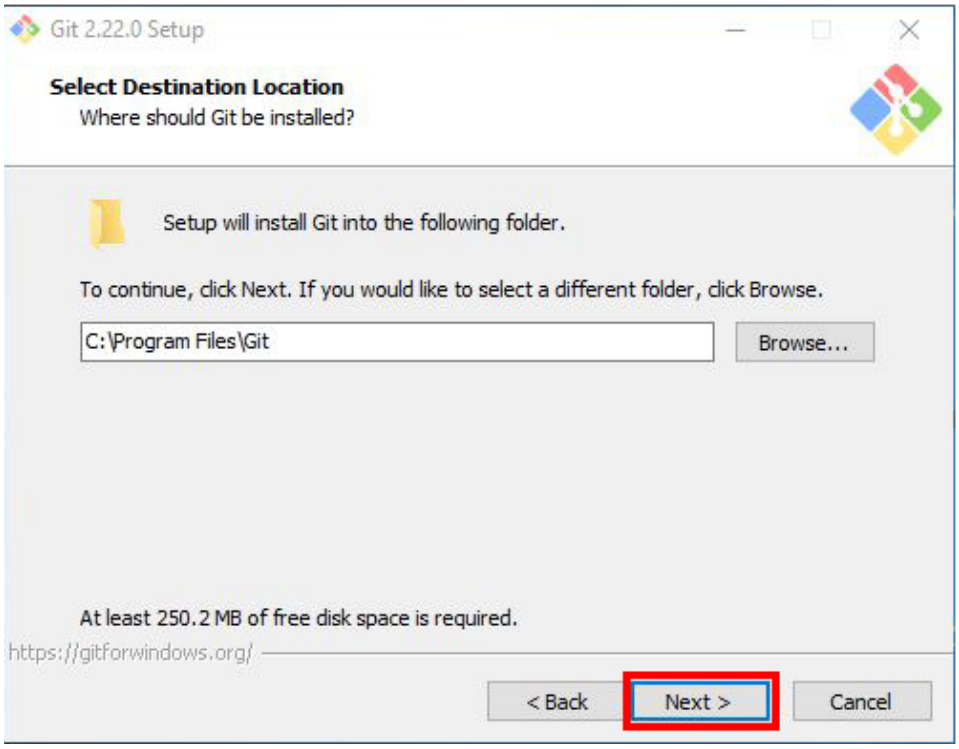
If all goes well, the download should start automatically.



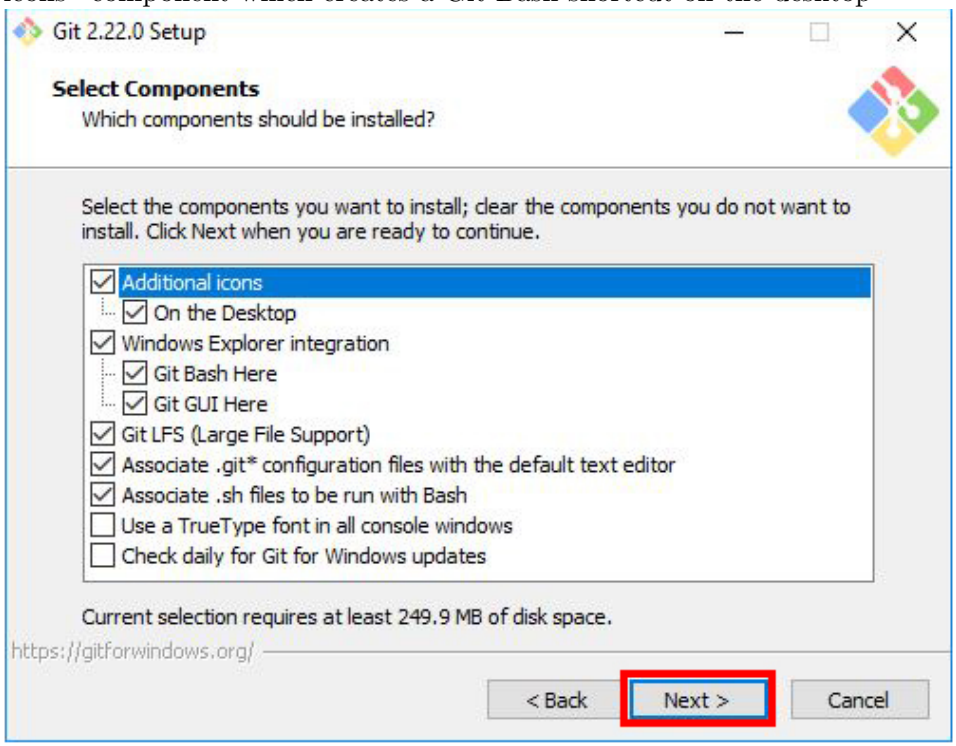
Click on "Save File" to start downloading the executable

Step 3: Run the Installer:
Once the user has downloaded the Git Bash executable, click it to run the installer.

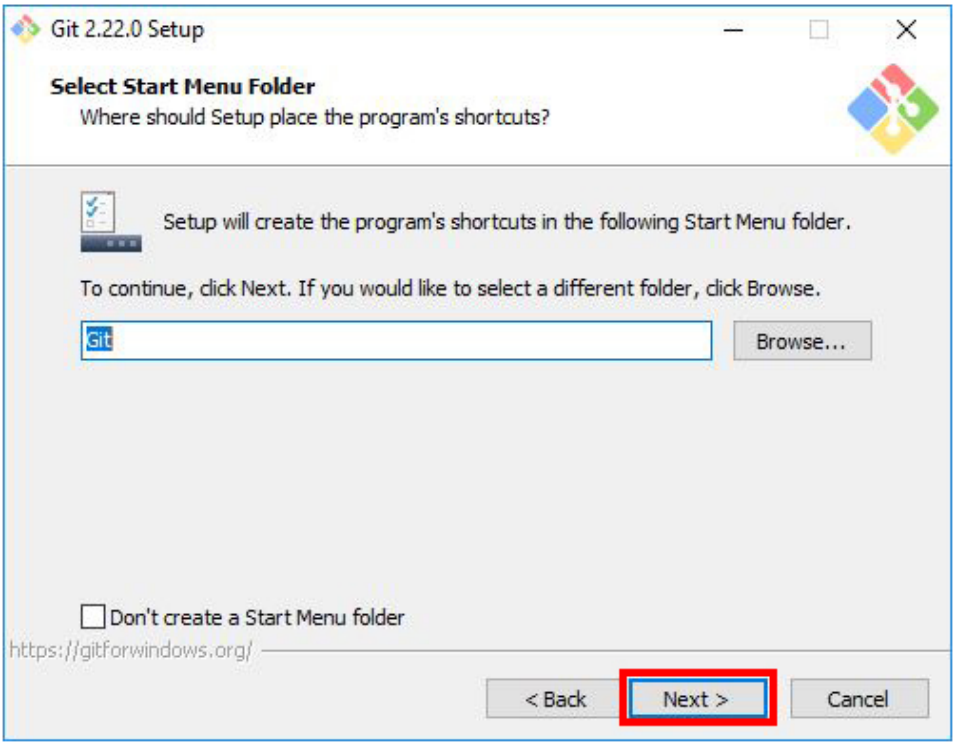
Step 4: Select Destination Location:
Next, select the location that the user wants to install Git Bash. It is recommended to just leave the default option as it is, and click “Next”.



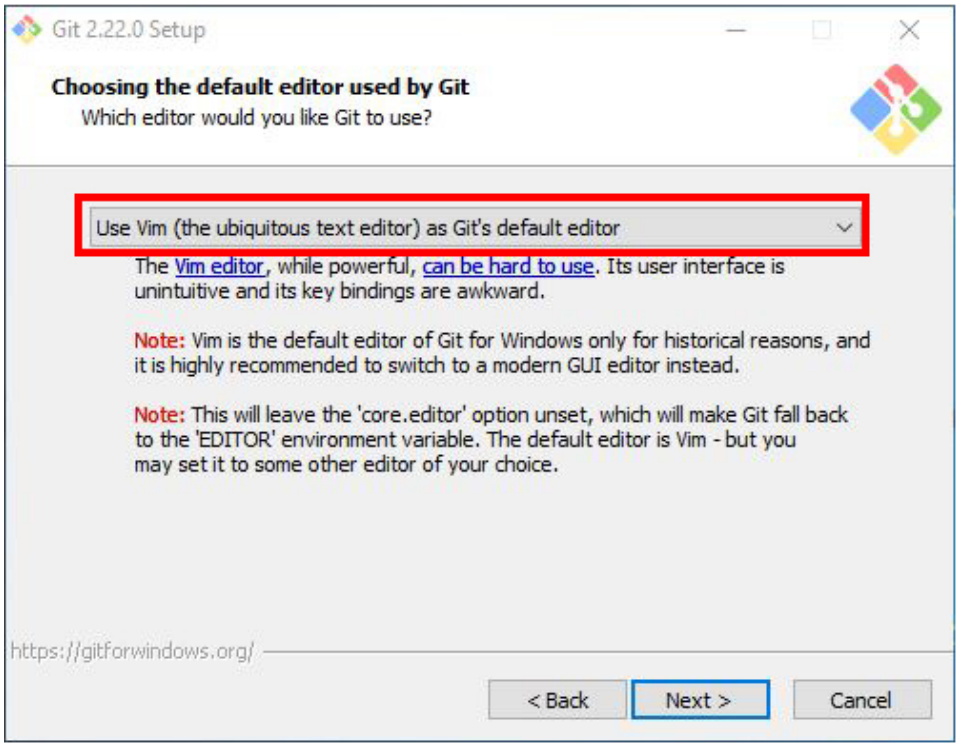
Step 5: Select Components:
Choose the components the user wants to install, or just proceed with the default options and click “Next”. It is preferred to select the “Additional icons” component which creates a Git Bash shortcut on the desktop

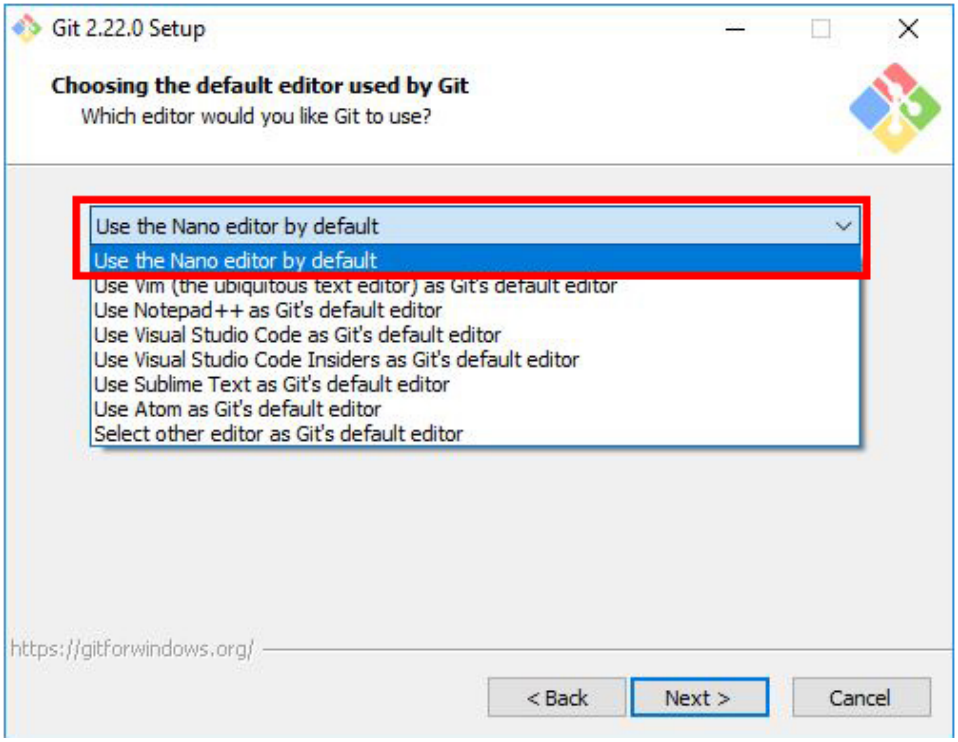


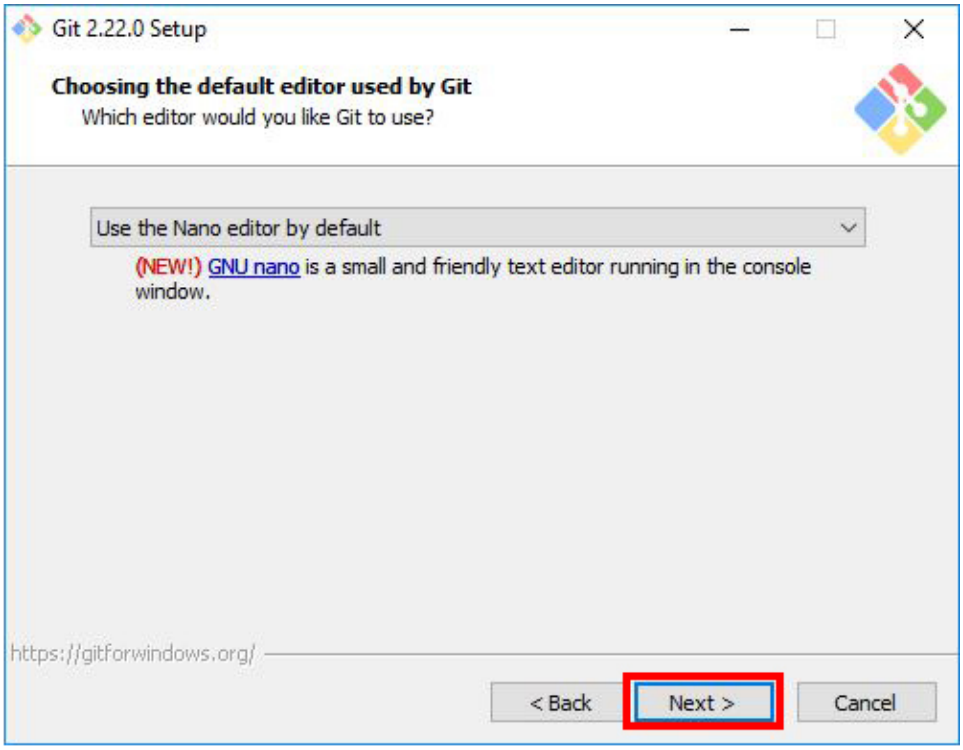
Step 6: Select Start Menu Folder
The user can change the name of start menu folder here if he/she want, or just leave the default name and click “Next”.



Step 7: Choose the Default Editor used by Git
Next, select the default editor for Git to use. Choose the one that the user likes and click "Next". It is recommended to proceed with Nano or Notepad++. Don't proceed with the default option "Vim" as it has a steep learning curve.

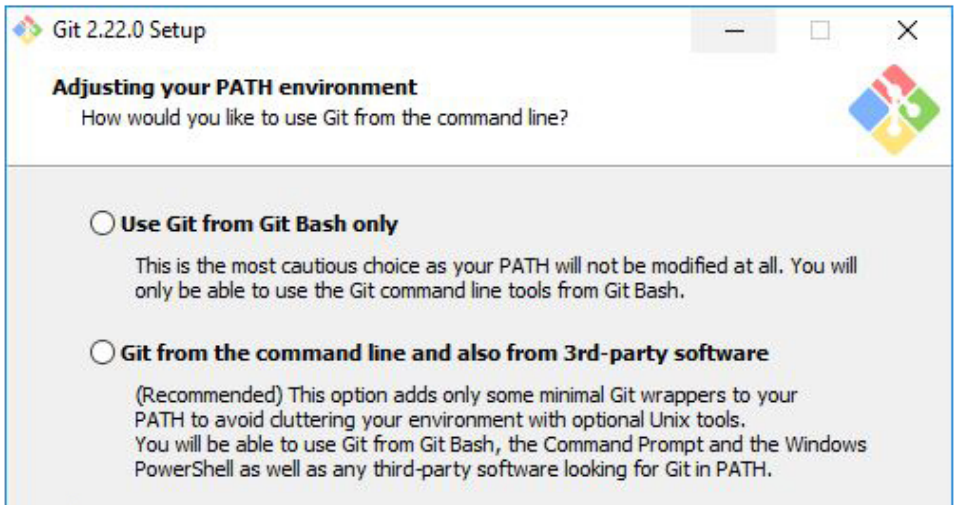

Git 2.22.0 Setup
Choosing the default editor used by Git
Which editor would you like Git to use?
Use Vim (the ubiquitous text editor) as Git's default editor
The Vim editor, while powerful, can be hard to use. Its user interface is unintuitive and its key bindings are awkward.
Note: Vim is the default editor of Git for Windows only for historical reasons, and it is highly recommended to switch to a modern GUI editor instead.
Note: This will leave the 'core.editor' option unset, which will make Git fall back to the 'EDITOR' environment variable. The default editor is Vim - but you may set it to some other editor of your choice.
https://gitforwindows.org/
< Back
Next >
Cancel


Git 2.22.0 Setup
Choosing the default editor used by Git
Which editor would you like Git to use?
Use the Nano editor by default
Use the Nano editor by default
Use Vim (the ubiquitous text editor) as Git's default editor
Use Notepad++ as Git's default editor
Use Visual Studio Code as Git's default editor
Use Visual Studio Code Insiders as Git's default editor
Use Sublime Text as Git's default editor
Use Atom as Git's default editor
Select other editor as Git's default editor
https://gitforwindows.org/
< Back
Next >
Cancel

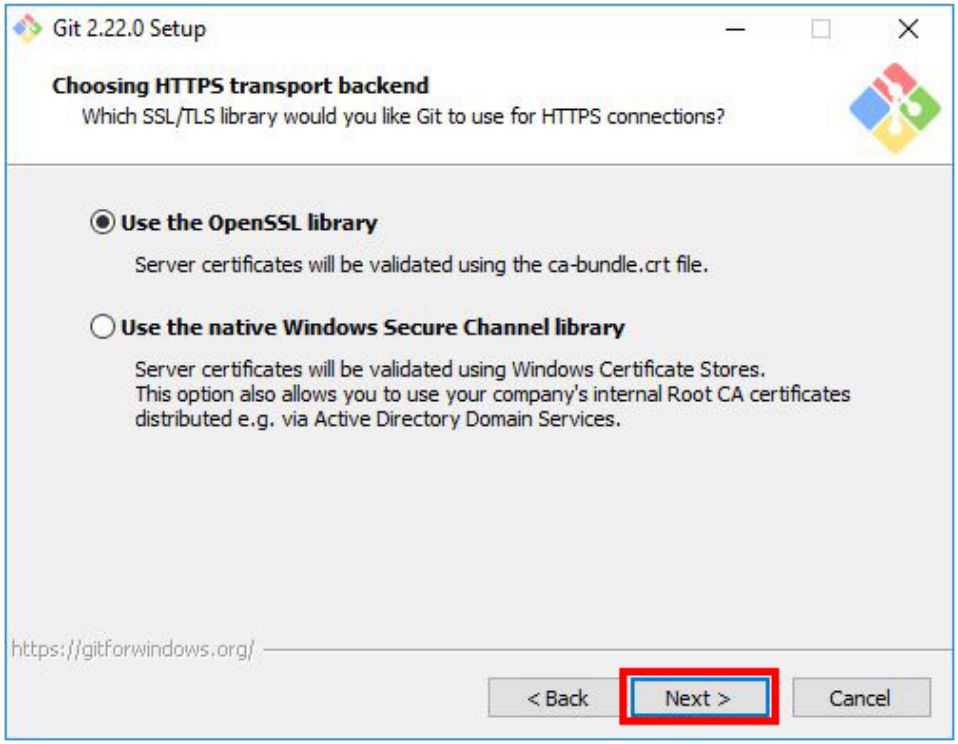


Step 8: Adjust the PATH Environment
Choose the option the user wants depending on where the user wants use Git and click "Next".

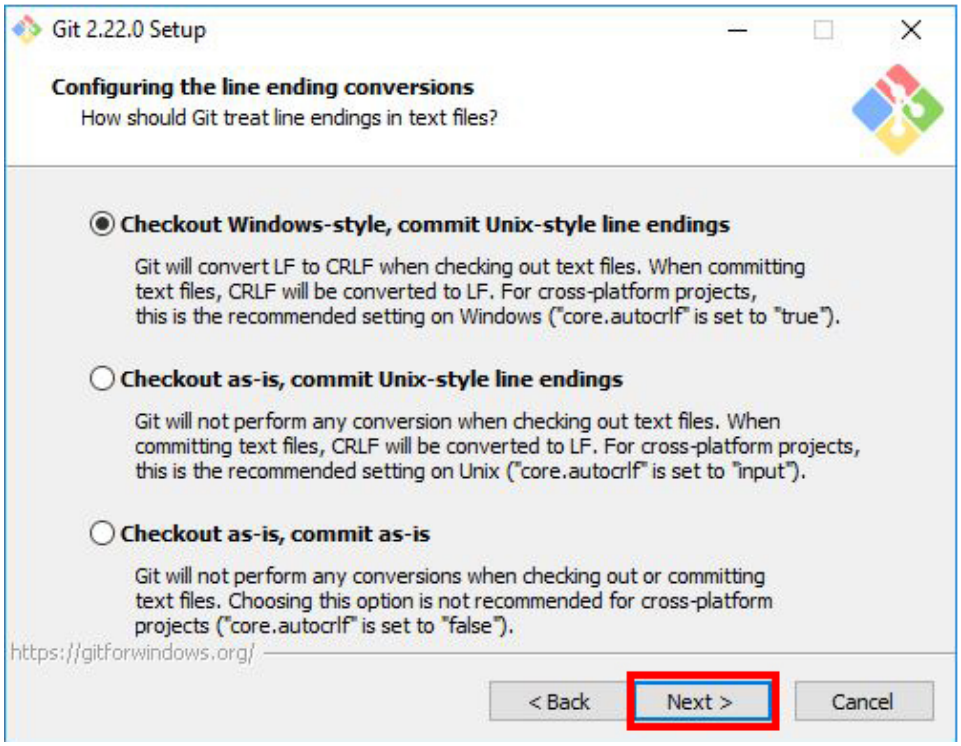
Select "Use Git from Git Bash only" option to run Git and Bash commands from Git Bash only. This means that the user will not be able to run Git commands such as git status on Windows Command Prompt or Powershell. They will only be found on Git Bash.



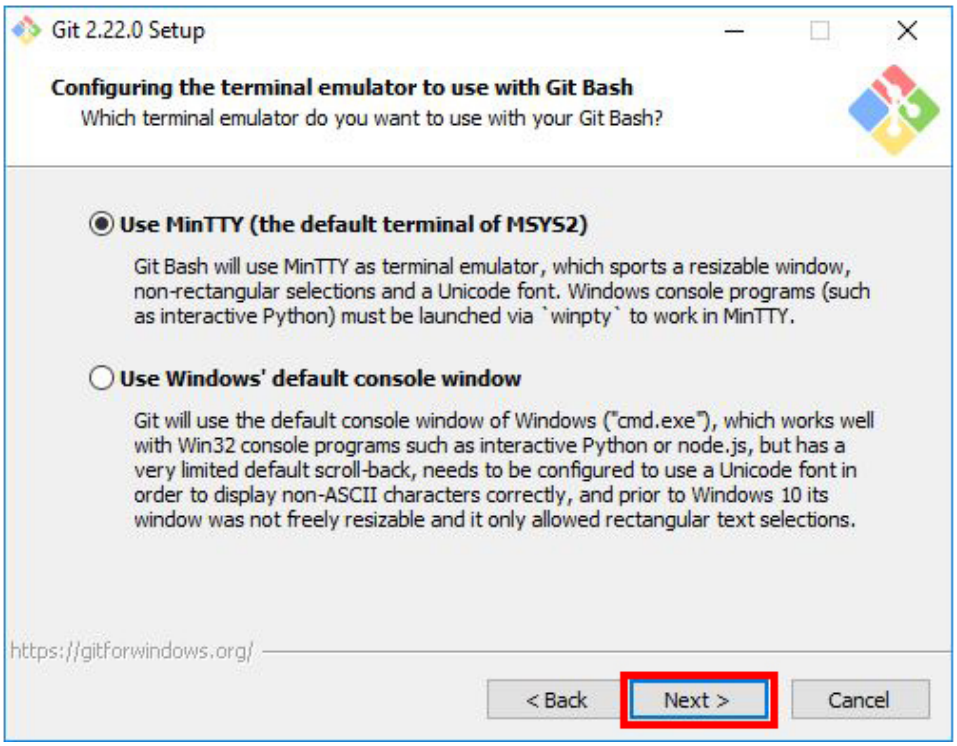
Step 9: Choose HTTPS Transport Backend
Next, select "Use the OpenSSL library" and click "Next".



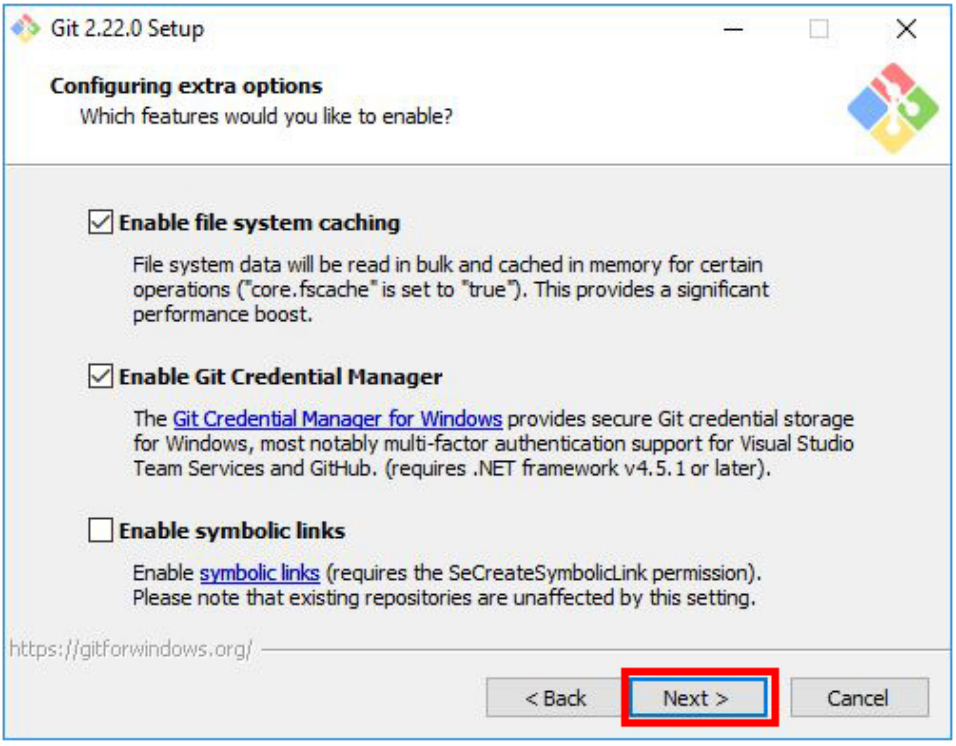
Step 10: Configure the Line Ending Conversions
Select how Git should treat line endings in text files. It's safe to go with the default option "Checkout Windows-Style, commit Unix-style line endings". Click "Next" to proceed.



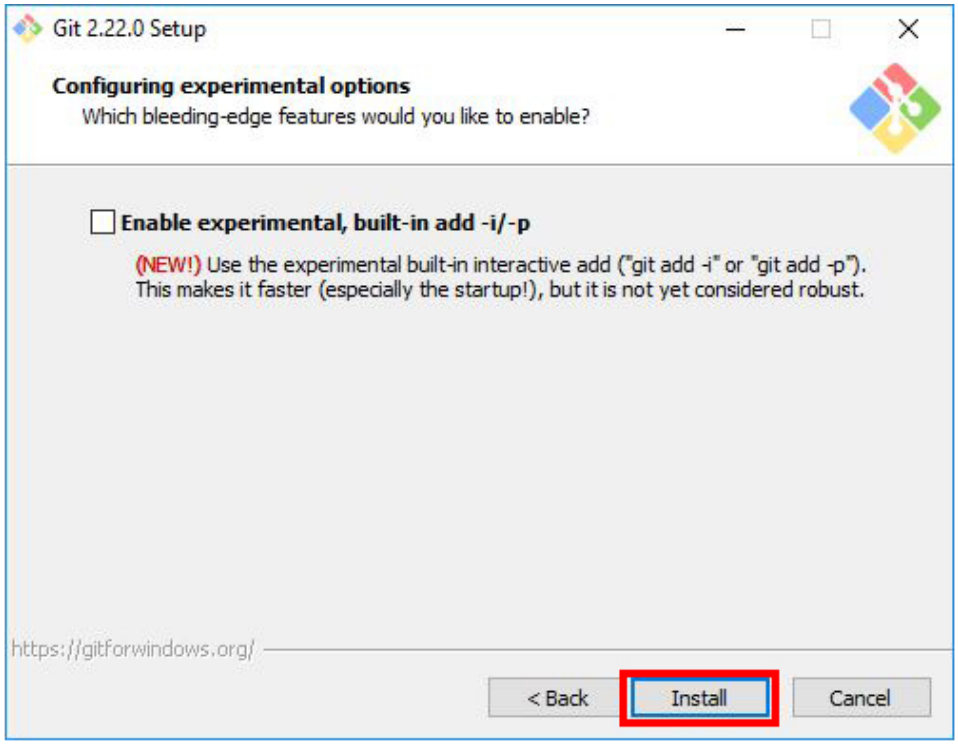
Step 11: Configure the Terminal Emulator to use with Git Bash
Next, select the terminal emulator the user wants Git Bash to use. Proceed with the default option "Use MinTTY(the default terminal of MSYS2) and click "Next".



Step 12: Configuring Extra Options
Select the features the user wants (the default options are fine) and click "Next".



Step 13: Configuring Extra Options
Enable experimental options if the user want. Enabling them allows the user to try out newer features that are still in development. It is not recommended, just proceed by clicking "Install" to start the installation process.

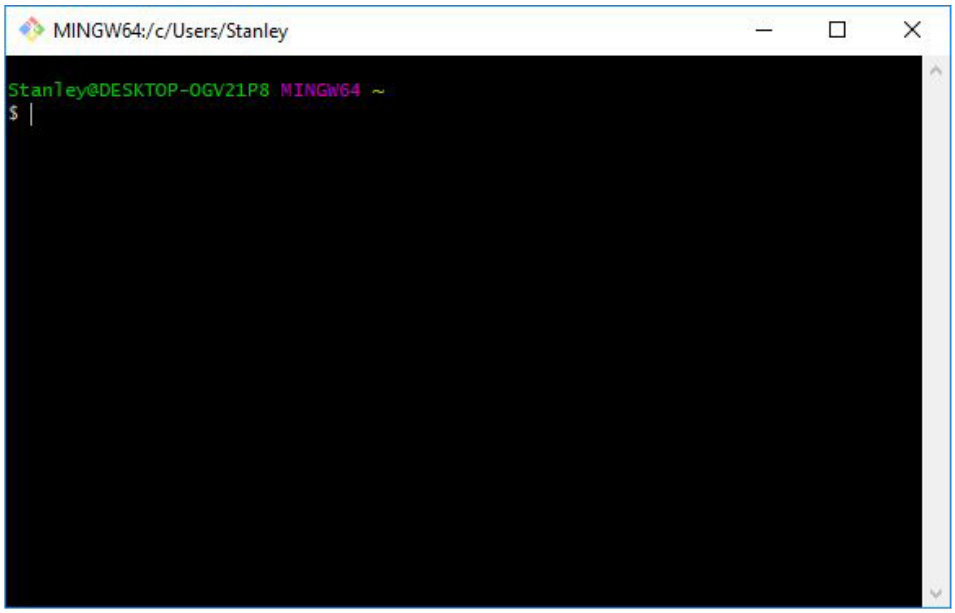


Step 14: Wait for Installation

Step 15: Complete the Git Setup Wizard

After the installation has finished, check the "Launch Git Bash" and click "Finish" to launch Git Bash.

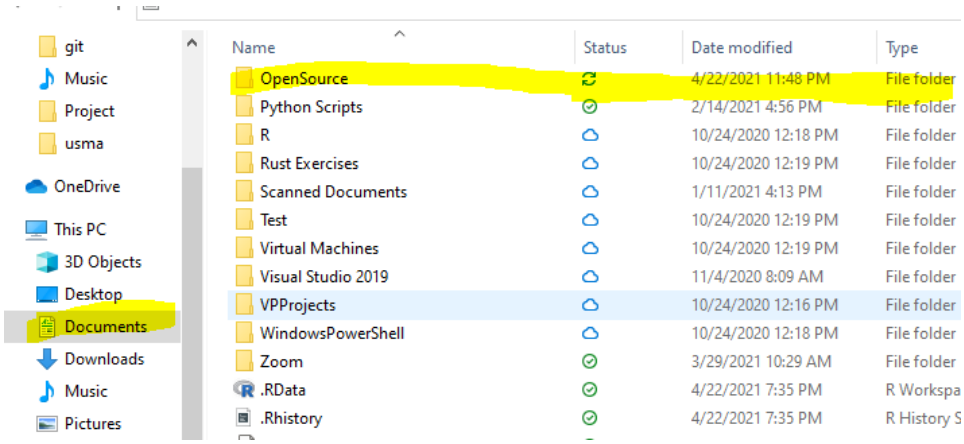
The Git Bash terminal will now open and the user will be able to enter Git and Bash commands.



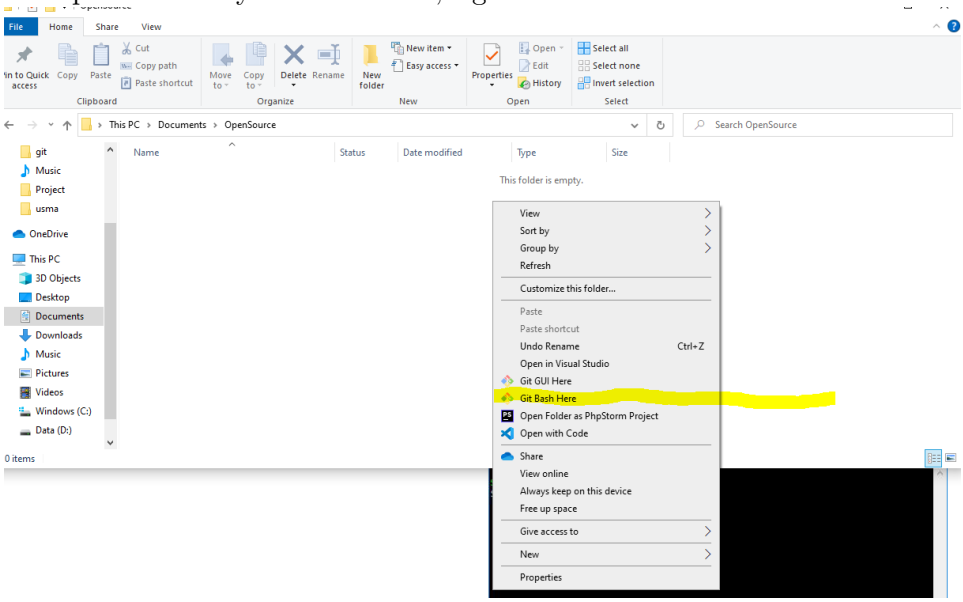
2.1 Clone Repository

To obtain the source code and architecture of the app, the user must clone the repository. When the user clones the repository, they can view the source code of the front end and back end that makes the app function. The user can also run the app on an android emulator or personal device but there will need to be additional installations which will be explained furthermore in the later sections.

Step 1: Create a folder where the user can clone the repository. For example create a folder in Documents called OpenSource.



Open the newly created folder, right click and select "Git Bash Here"

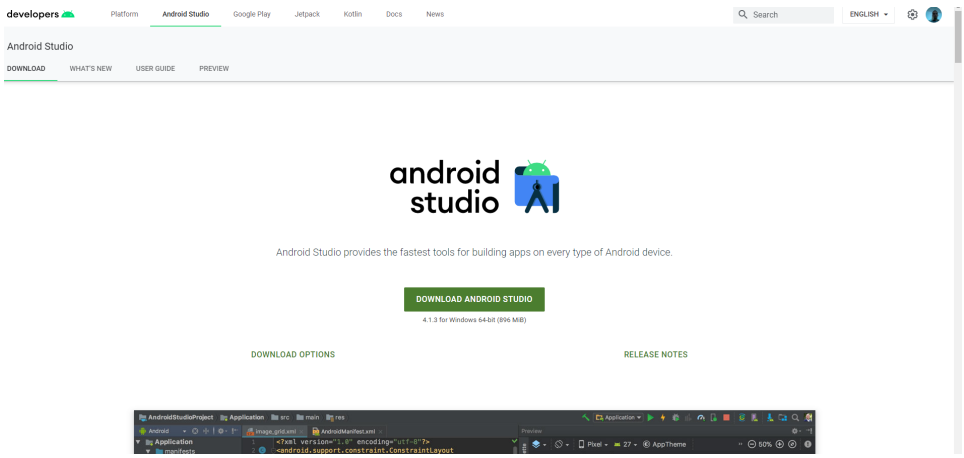


Step 2: In the terminal enter "git clone https://bitbucket2020.okanagan.bc.ca/scm/oie/engine.git"

Please wait a few seconds. The user should now have the source code that makes the app function.

3 Install Android Studio

To run android studio on an emulator, the user must download android studio for windows on the official website: https://developer.android.com/studio

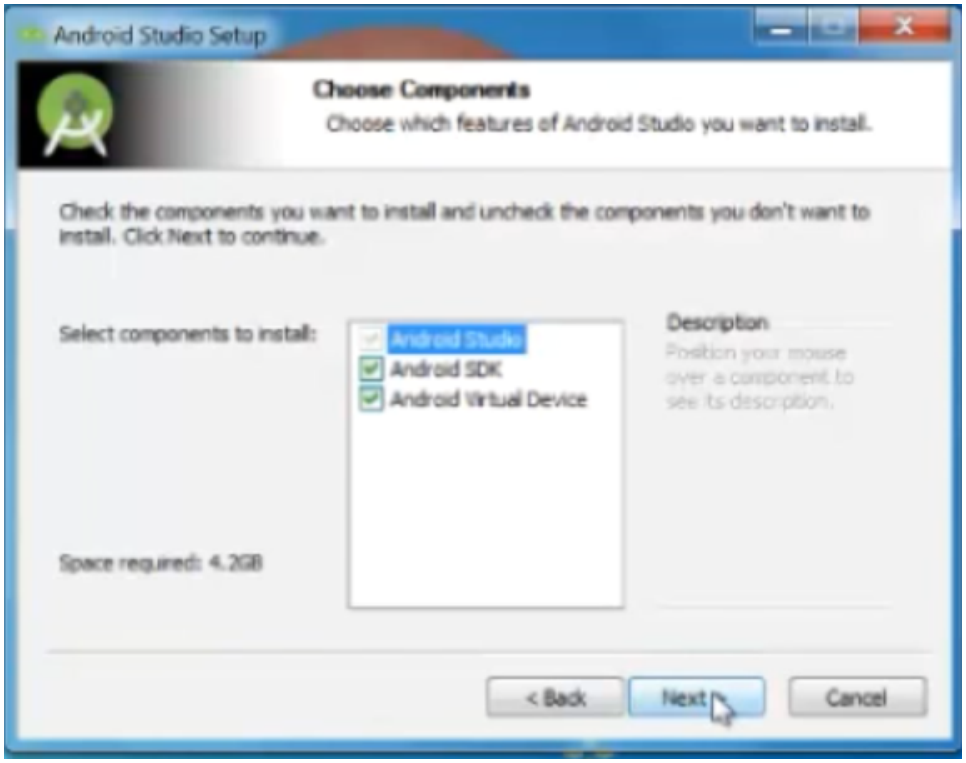


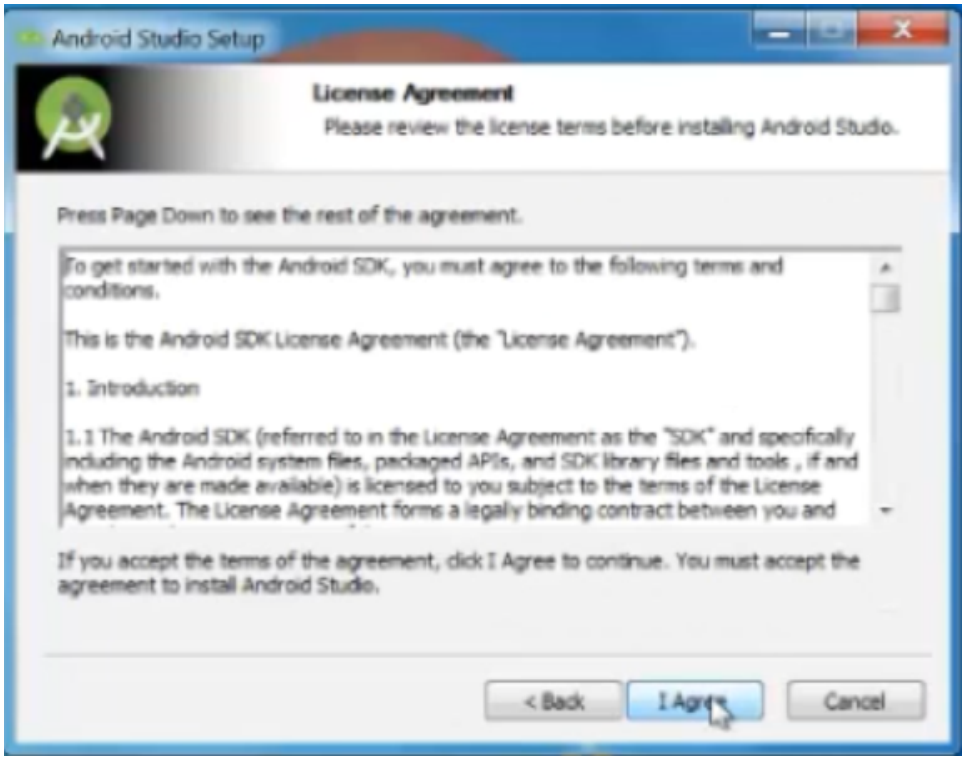
1. If the user downloaded an .exe file (recommended), double-click to launch it.

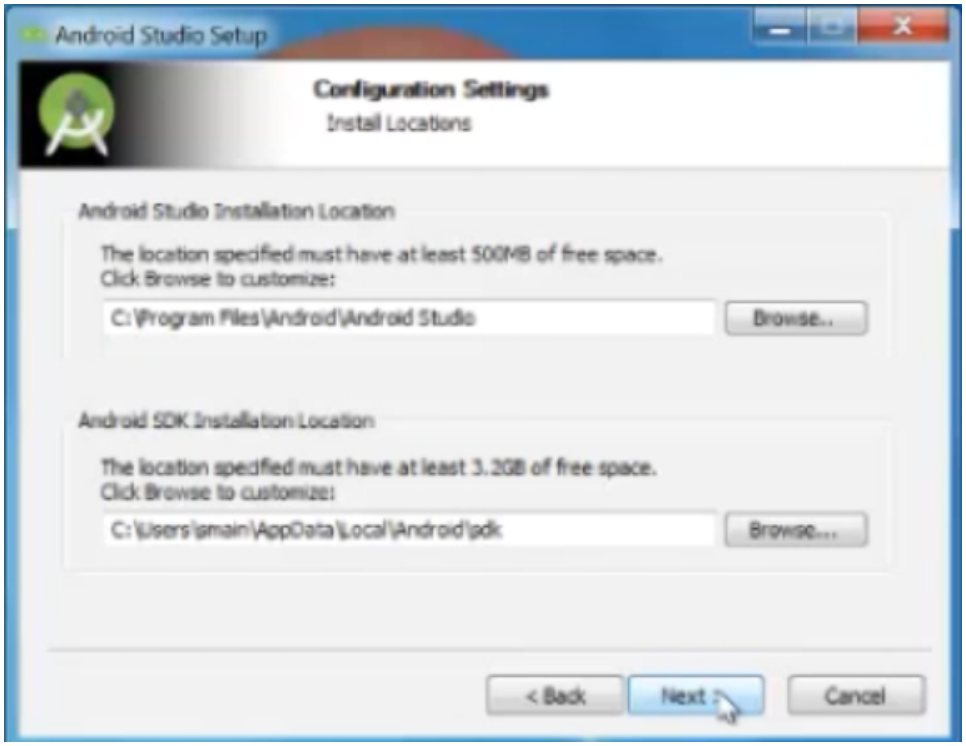
If the user downloaded a .zip file, unpack the ZIP, copy the android-studio folder into the Program Files folder, and then open the android-studio bin folder and launch studio64.exe (for 64-bit machines) or studio.exe (for 32-bit machines).

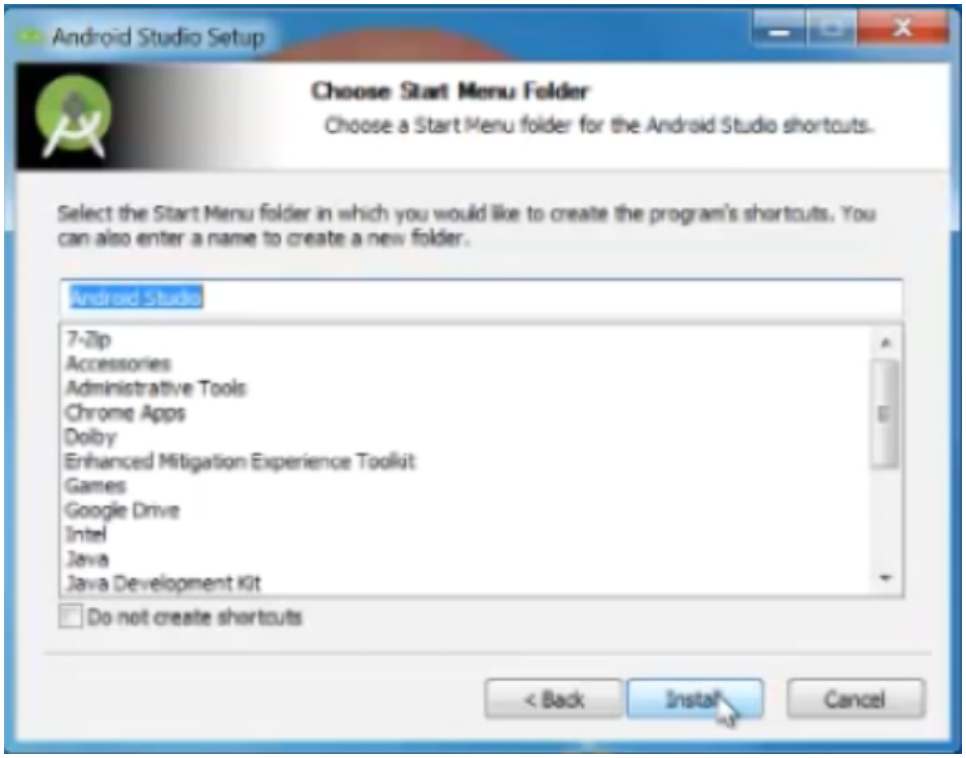
2. Follow the setup wizard in Android Studio and install any SDK packages that it recommends.

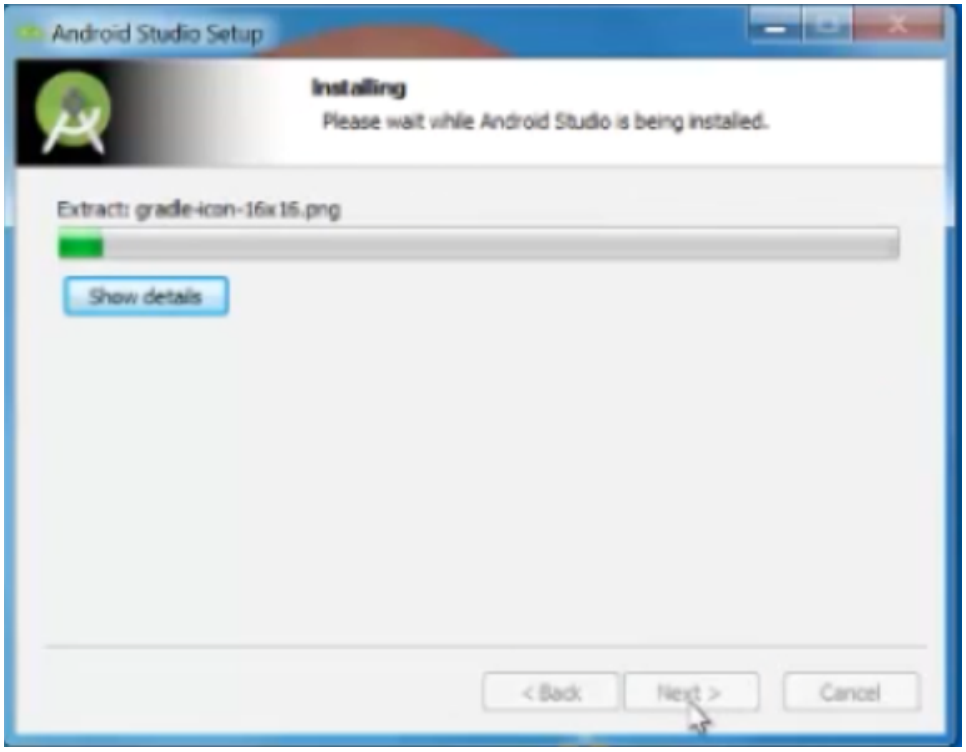

Android Studio Setup
Welcome to Android Studio Setup
Setup will guide you through the installation of Android Studio.
It is recommended that you close all other applications before starting Setup. This will make it possible to update relevant system files without having to reboot your computer.
Click Next to continue.
Android Studio
< Back
Next >
Cancel


Android Studio Setup
Choose Components
Choose which features of Android Studio you want to install.
Check the components you want to install and uncheck the components you don't want to install. Click Next to continue.
Select components to install:
Android Studio
Android SDK
Android Virtual Device
Description
Position your mouse over a component to see its description.
Space required: 4.2GB
< Back
Next
Cancel

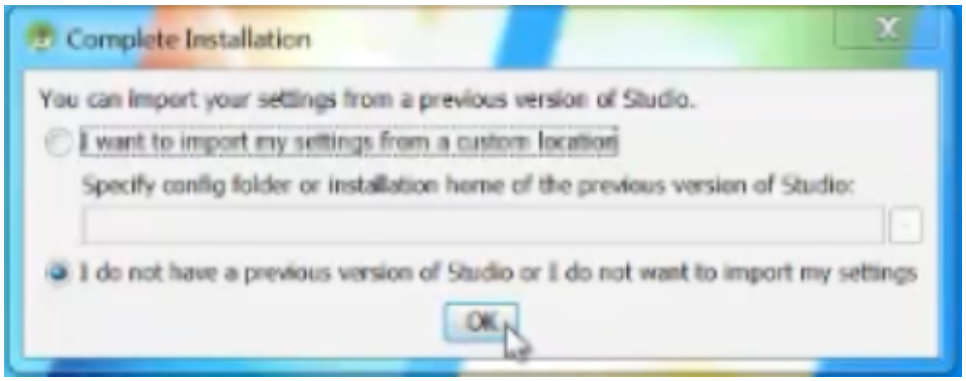

Android Studio Setup
License Agreement
Please review the license terms before installing Android Studio.
Press Page Down to see the rest of the agreement.
To get started with the Android SDK, you must agree to the following terms and conditions.
This is the Android SDK License Agreement (the "License Agreement").
1. Introduction
1.1 The Android SDK (referred to in the License Agreement as the "SDK" and specifically including the Android system files, packaged APIs, and SDK library files and tools , if and when they are made available) is licensed to you subject to the terms of the License Agreement. The License Agreement forms a legally binding contract between you and
If you accept the terms of the agreement, click I Agree to continue. You must accept the agreement to install Android Studio.
< Back
I Agree
Cancel

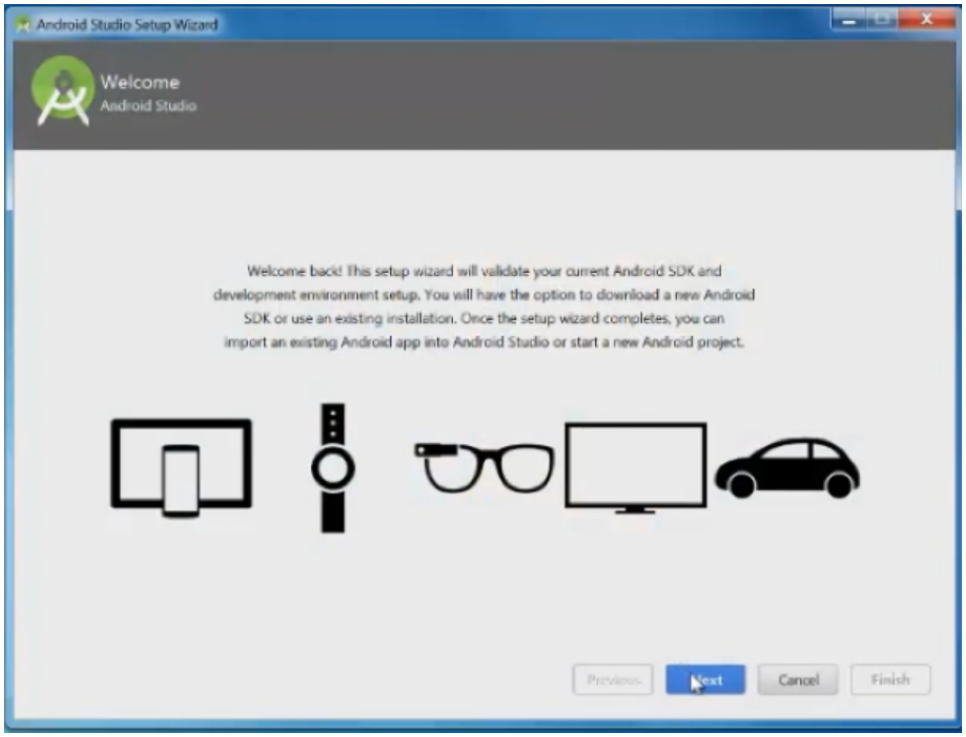

Android Studio Setup
Configuration Settings
Install Locations
Android Studio Installation Location
The location specified must have at least 500MB of free space.
Click Browse to customize:
C:\Program Files\Android\Android Studio
Browse...
Android SDK Installation Location
The location specified must have at least 3.2GB of free space.
Click Browse to customize:
C:\Users\smain\AppData\Local\Android\sdk
Browse...
< Back
Next >
Cancel

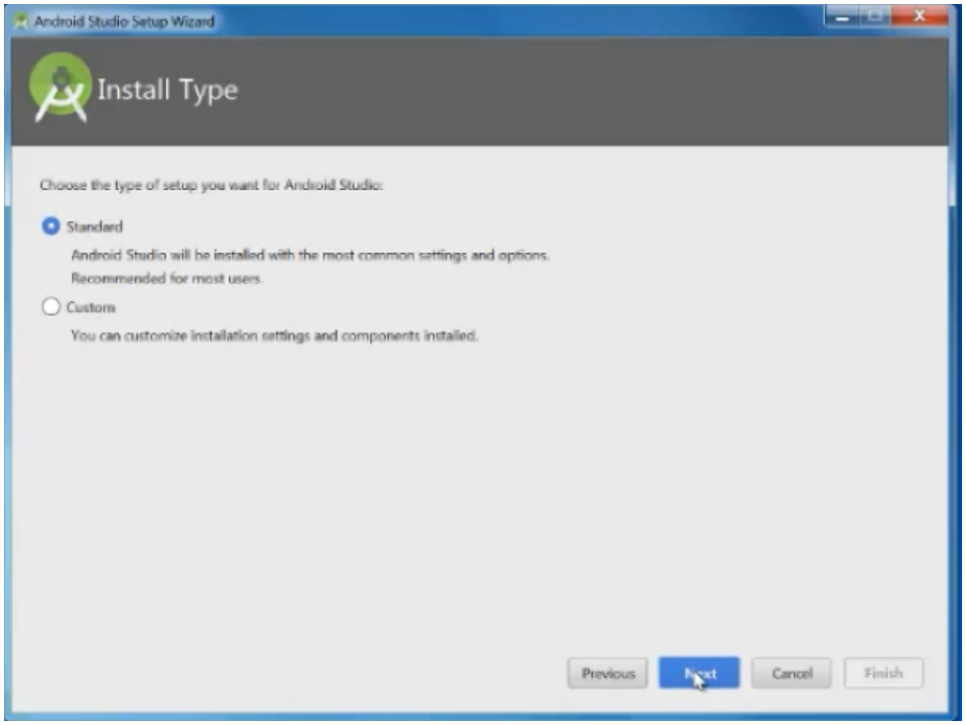

Android Studio Setup
Choose Start Menu Folder
Choose a Start Menu folder for the Android Studio shortcuts.
Select the Start Menu folder in which you would like to create the program's shortcuts. You can also enter a name to create a new folder.
Android Studio
7-Zip
Accessories
Administrative Tools
Chrome Apps
Dolby
Enhanced Mitigation Experience Toolkit
Games
Google Drive
Intel
Java
Java Development Kit
Do not create shortcuts
< Back
Install
Cancel

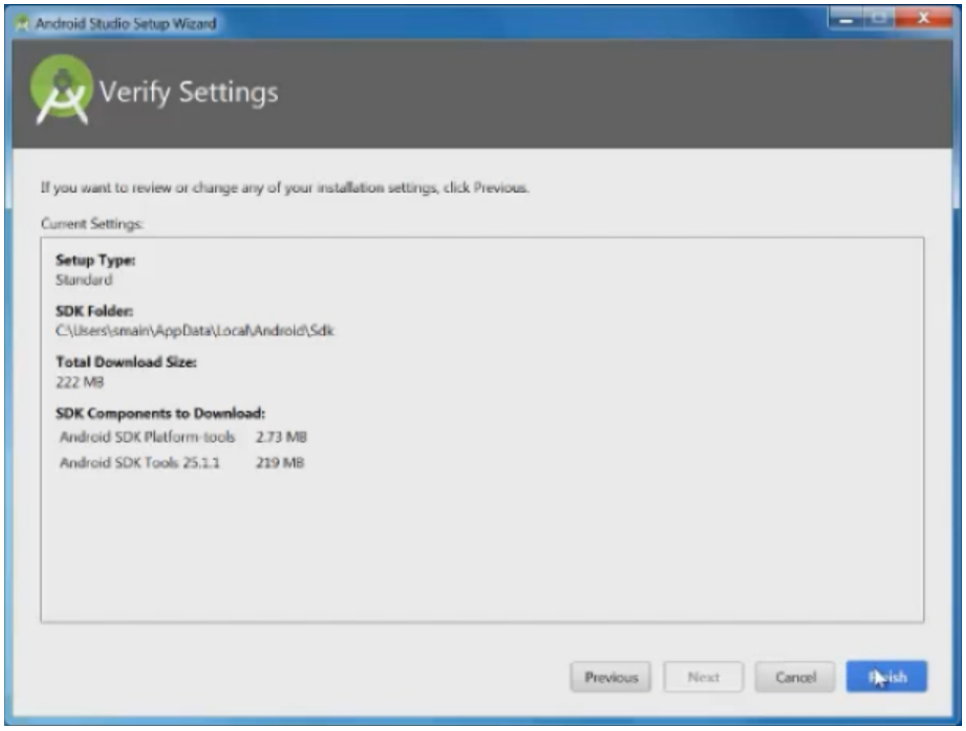

Android Studio Setup
Installing
Please wait while Android Studio is being installed.
Extract: gradle-icon-16x16.png
Show details
< Back
Next >
Cancel


Android Studio Setup
Completing Android Studio Setup
Android Studio has been installed on your computer.
Click Finish to close Setup.
Start Android Studio
Android
Studio
< Back
Finish
Cancel


Complete Installation
You can import your settings from a previous version of Studio.
I want to import my settings from a custom location
Specify config folder or installation home of the previous version of Studio:
I do not have a previous version of Studio or I do not want to import my settings
OK


Android Studio Setup Wizard
Welcome
Android Studio
Welcome back! This setup wizard will validate your current Android SDK and development environment setup. You will have the option to download a new Android SDK or use an existing installation. Once the setup wizard completes, you can import an existing Android app into Android Studio or start a new Android project.
Previous
Next
Cancel
Finish


Android Studio Setup Wizard
Install Type
Choose the type of setup you want for Android Studio:
Standard
Android Studio will be installed with the most common settings and options.
Recommended for most users.
Custom
You can customize installation settings and components installed.
Previous
Next
Cancel
Finish


Android Studio Setup Wizard
Verify Settings
If you want to review or change any of your installation settings, click Previous.
Current Settings:
Setup Type:
Standard
SDK Folder:
C:\Users\smain\AppData\Local\Android\Sdk
Total Download Size:
222 MB
SDK Components to Download:
Android SDK Platform-tools 2.73 MB
Android SDK Tools 25.1.1 219 MB
Previous
Next
Cancel
Finish

4 Install Apache Cordova/PhoneGap

Apache Cordova, also known as PhoneGap, is a great tool for developing mobile applications in HTML, CSS, and JavaScript. To install it on the users personal computer, the user must have Node.js and npm installed. NPM stands for Node Package Manager, which is an application and repository for developing and sharing JavaScript code.

4.1 How to Install Node.js and NPM on Windows

Step 1: Download Node.js Installer

In a web browser, navigate to https://nodejs.org/en/download/. Click the Windows Installer button to download the latest default version. At the time this was documented, version 14.16.1-x64 was the latest version. The Node.js installer includes the NPM package manager.

Step 2: Install Node.js and NPM from Browser

1. Once the installer finishes downloading, launch it. Open the downloads link in the browser and click the file. Or, browse to the location where the user saved the file and double-click it to launch.

2. The system will ask if the user wants to run the software – click Run.

3. The user will be welcomed to the Node.js Setup Wizard – click Next.

4. On the next screen, review the license agreement. Click Next if the user agrees to the terms and install the software.

5. The installer will prompt the user for the installation location. Leave the default location, unless the user has a specific need to install it somewhere else – then click Next.

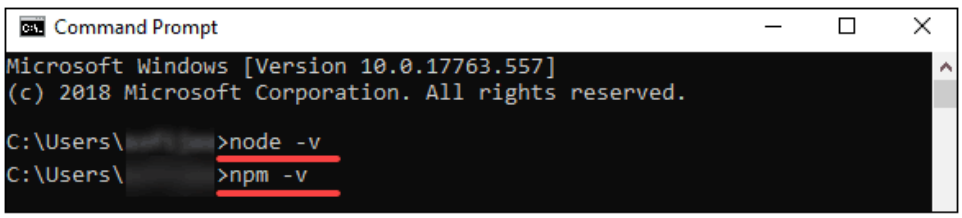
6. The wizard will let the user select components to include or remove from the installation. Again, unless the user has a specific need, accept the defaults by clicking Next.

7. Finally, click the Install button to run the installer. When it finishes, click Finish.

Step 3: Verify Installation

Open a command prompt (or GitBash), and enter the following:
node -v

The system should display the Node.js version installed on the users system. The user can do the same for NPM:
npm -v



4.2 Installing Cordova on Windows

This section is quite a complex process, which may take some time depending on the users previous experience with installing SDKs and build tools.

Step 1: Install Cordova:

Cordova is installed using the Node Package Manager (npm). Type the following in Git Bash to install:

npm install -g cordova

Step 2: Test the Cordova install by typing:

cordova -v

If the user sees the version number, they have successfully installed Apache Cordova.

Step 3: Check Requirements

Open the OpenSource folder that was created in page 18 of this document and open the engine folder. Go to CordovaApp folder, then opensourceintegration folder, then right-click, and "Git Bash Here". In the terminal type cordova requirements.

```
$ cordova requirements
Requirements check results for android:
Java JDK: installed .
Android SDK: installed
Android target: installed android-19,android-21,android-22,android-23,Google Inc.:Google APIs:19,Google Inc.:Google API
Gradle: installed

Requirements check results for ios:
Apple OS X: not installed
Cordova tooling for iOS requires Apple OS X
Error: Some of requirements check failed
```

The goal for this section is to make sure that all of the requirements are installed like the image shown above. If none of the requirements are installed or some are missing then we must install all of them by starting with the Java JDK.

Step 4: Install Java JDK

The Android SDK needs the Java Development Kit (JDK) to be installed, version 1.7 or later. Note that the Java Runtime Environment (JRE) is not sufficient, the user will need the JDK. To check if the user has the JDK installed already, type this on Git Bash:

javac -version

If the user does not have the JDK installed, proceed as follows:

Step 5: Download the recent version of Java SE JDK (SE = Standard Edition) from Oracle: www.oracle.com/technetwork/java/javase/downloads/. Click the Java SE Download to see the list of downloads. Get the "Windows x86" download if the user has 32-bit Windows, and "Windows x64" if the user has 64-bit Windows. If the user does not know which version, find out using the Control Panel by selecting "System and Security" and then "System", where the user will find the "System type" saying if Windows version is 32-bit or 64-bit.

Step 6: Go along and run the downloaded installer file. Using the default selections should be fine, but take a note of the directory in which the user installs the JDK. The user will need to add this to the PATH in a later step below.

Step 7: Next, update the path to include the JDK. Open the Control Panel, click System and Security, click System, click Advanced system settings on the right, then click the Environment Variables button.

Step 8: In the User variables select PATH and click the Edit button. (If there is no PATH entry in the list, click the New button to create one.)

Step 9: At the end of the field Variable value, add a semi colon followed by the path to the bin directory of the JDK install. Here is an example (note that this must be the actual path used for the install on the users machine):

```
;C:\Program Files\Java\jdk1.8.0_11\bin
```

An easy way to do this is to prepare the path to add in a text editor, then paste it at the end of the input field. When done click the OK button.

Step 10: Next add the JAVA_HOME variable if it is not present (and if it is in the list, the user may need to update its value using the Edit button). Click the New button. In the field Variable name type:

JAVA_HOME

In the field Variable value enter the path to the directory where the JDK is installed, without the semicolon and the /bin subdirectory, for example:

```
C:\Program Files\Java\jdk1.8.0_11
```

Click the OK button.

Step 11: Click the OK button again to close the Environment Variables window.

Step 12: Now the user is ready to test the install. Close any open command windows, and open a new command window and type:

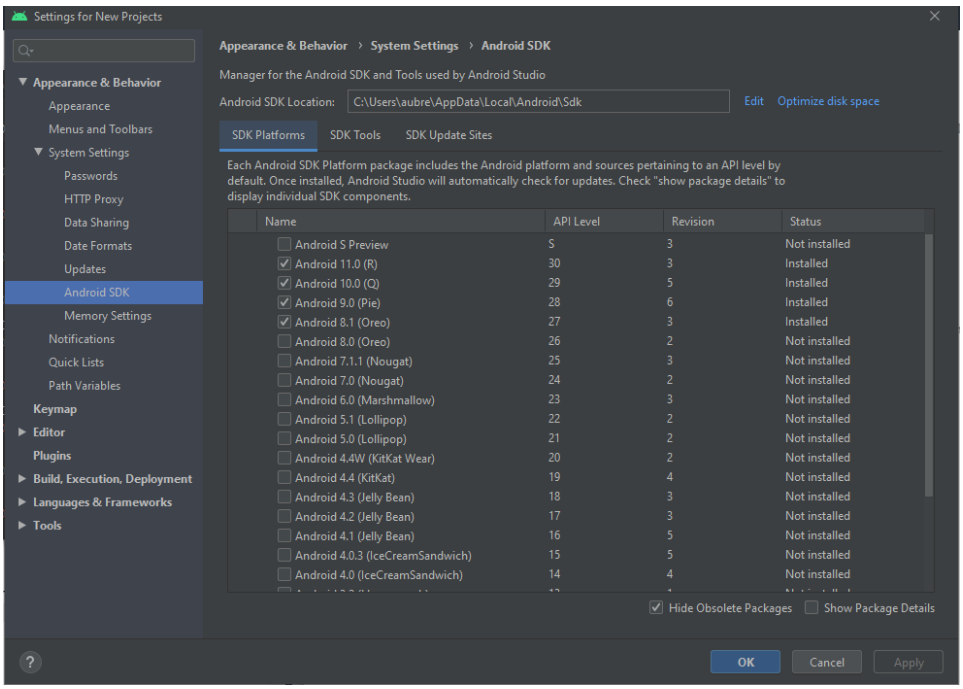
javac -version

If the user sees a version number then they are done with the JDK install.

The user has already opened Android Studio for the first time and was guided through the process of installing the Android SDK in page 21 of this document. After installing the Android SDK, the user must also install the packages for whatever API level they wish to target. It is recommended that the user installs the highest SDK version.

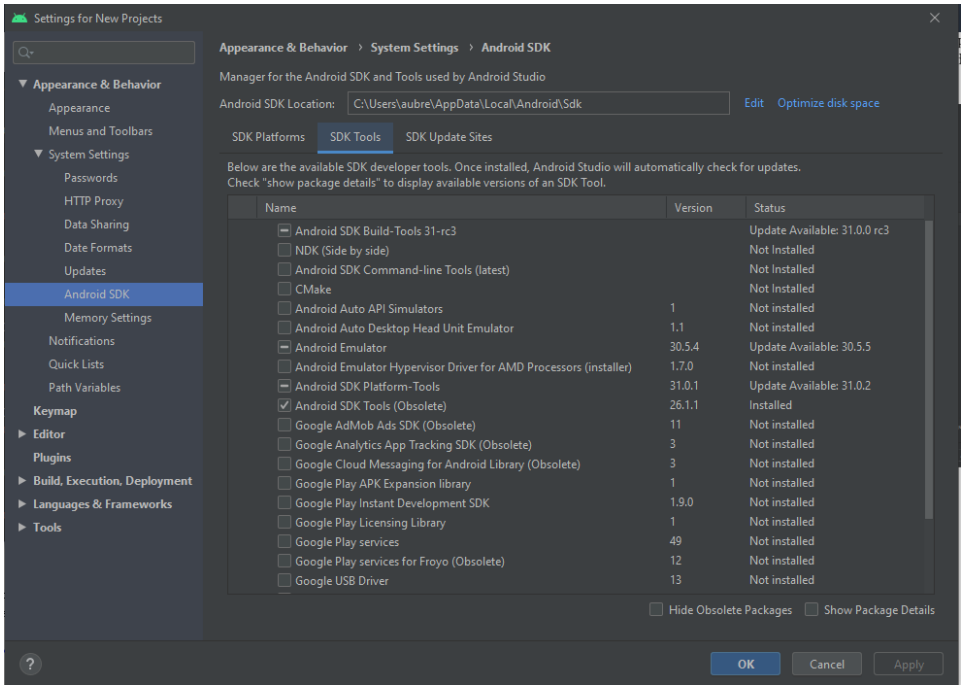
Step 13: Install Android SDK Platforms
Open Android Studio if it is not already opened and select the configure button in the bottom corner and open the SDK Manager

Be sure to select the four check boxes as shown below to run the app on the highest API level. Download these four APIs and wait until the download is finished.



Step 14: Install SDK Tools

Next go to SDK Tools and make sure that the four options below are downloaded. Android SDK Build-Tools 31-rc3, Android Emulator, Android SDK Platform-Tools, and Android SDK Tools (Obsolete). Uncheck the "Hide Obsolete Packages" on the bottom right corner to download the Android SDK Tools (Obsolete) package.



After this, the user should now have the JDK, Android SDK and Android Target installed as shown in the cordova requirements in page 32 of this document. The last requirement to install is the Gradle.

Step 15: Installing Gradle Manually
The current version for Gradle is 7.0, released on April 9th, 2021 when this document was created.

Go to this link: https://gradle.org/next-steps/?version=7.0&format=bin and the download for Gradle should begin right away

Step 16: Unpack Zip File
Create a new directory C:/Gradle with File Explorer.
Open a second File Explorer window and go to the directory where the Gradle distribution was downloaded. Unzip the ZIP archive to expose the content. Drag the content folder gradle-7.0 to the newly created C:/Gradle

folder.
Alternatively you can unpack the Gradle distribution ZIP into C:/Gradle using an archiver tool of your choice.

Step 17: Configure System Environment:
In File Explorer right-click on the This PC (or Computer) icon, then click Properties, Advanced System Settings, Environmental Variables. Under System Variables select Path, then click Edit. Add an entry for

`C:\Gradle\gradle-7.0\bin.`

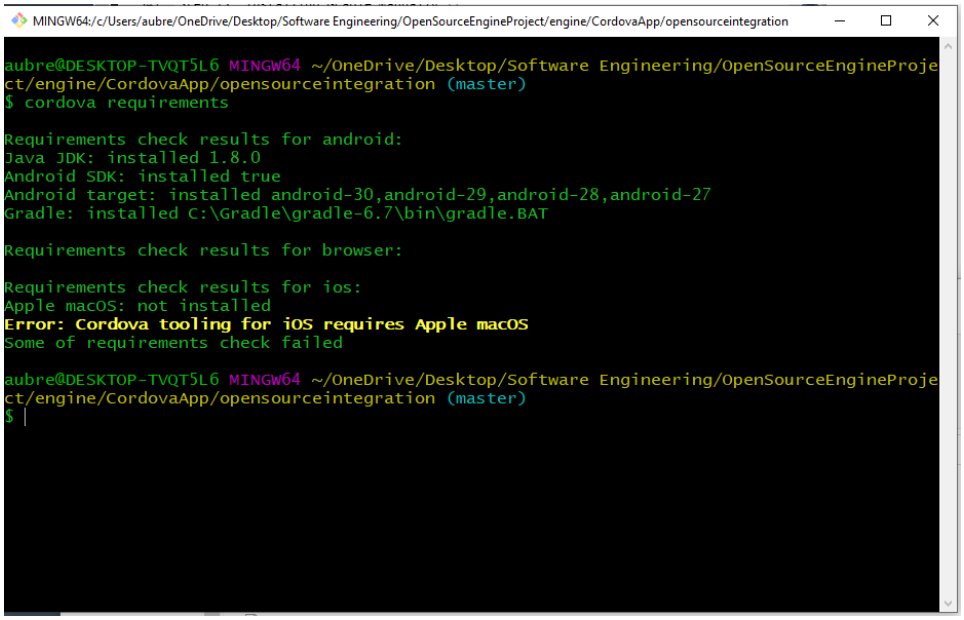
Click OK to save.

Step 18: Verify the Gradle installation
Open Git Bash (or a Windows command prompt) and run gradle -v to run gradle and display the version, e.g.:

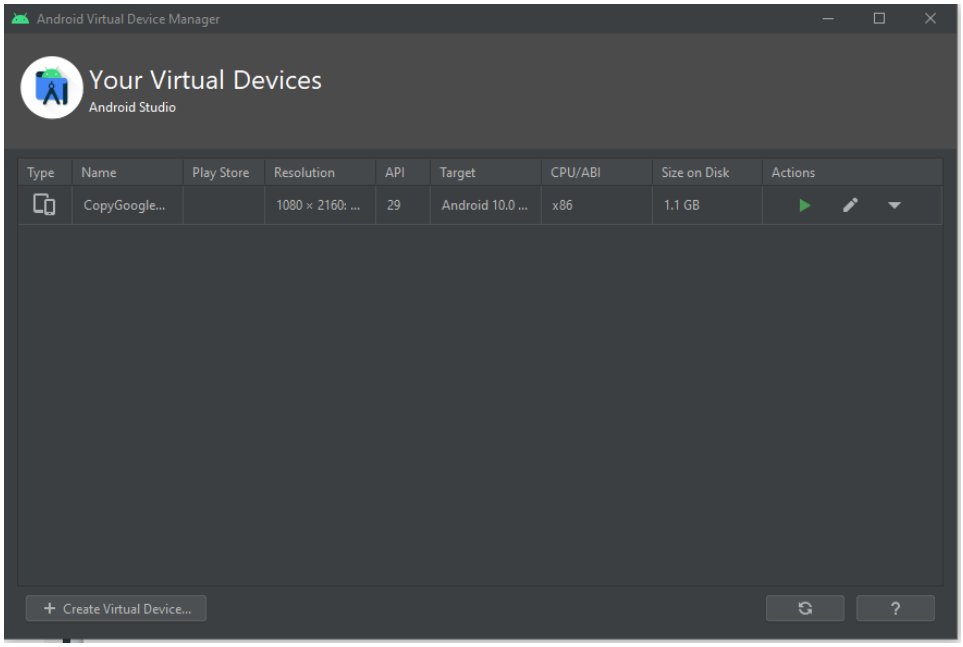
```
$ gradle -v

------------------------------------------------------------
Gradle 7.0
------------------------------------------------------------
```

In file explorer Go back to the OpenSource project folder that was created on page 18 of this document and open engine folder, then CordovaApp, then opensourceintegration folder, then right-click and "Git Bash Here"

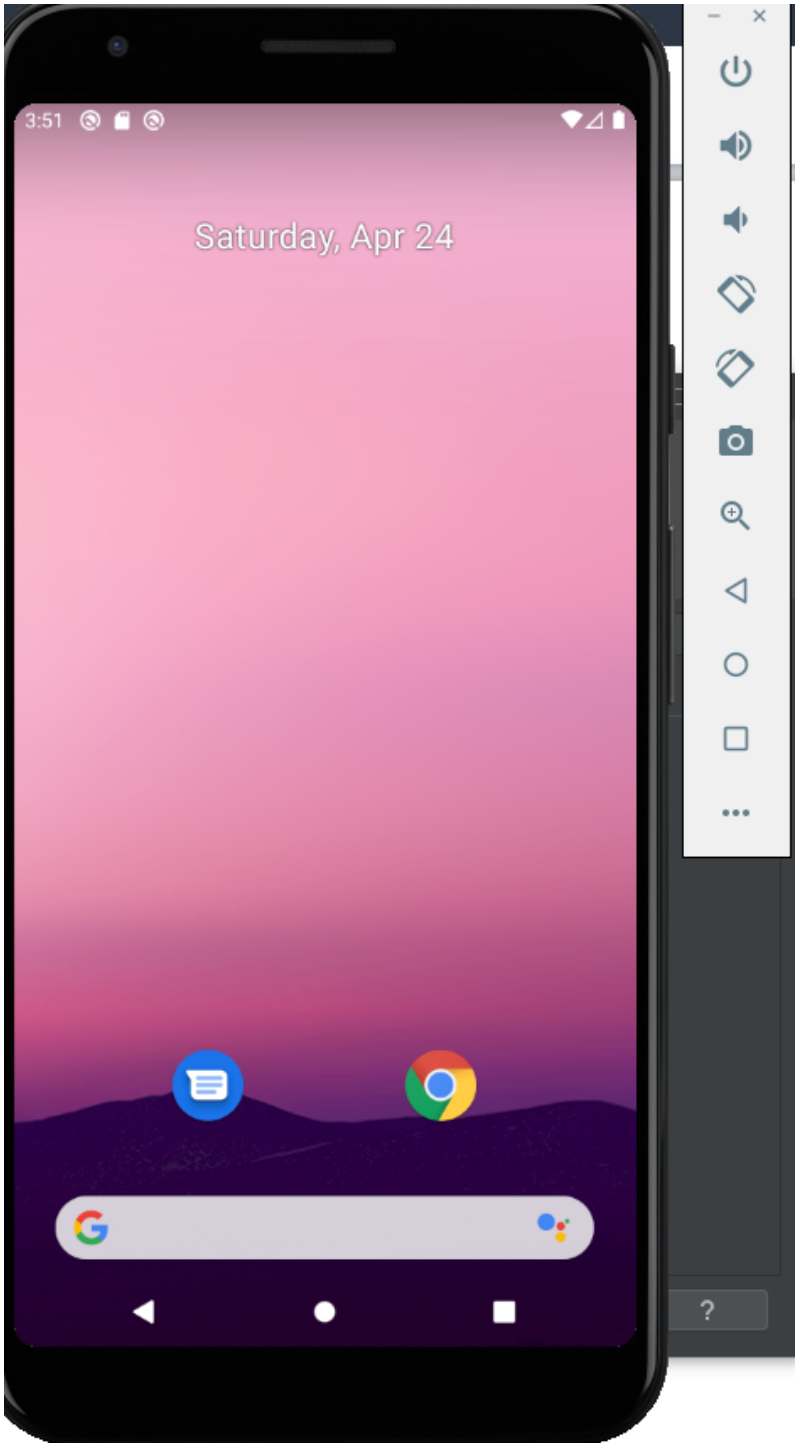
Type cordova requirements and the user should have the four requirements installed shown below. Java JDK, Android SDK, Android Target, Gradle installed. Do not worry about the Error. We are not using an Apple OS or iPhone in this project.



Step 19: Open Android Emulator
Open Android Studio, go to Configure on the bottom right, open AVD manager, and the user should see the screen below. Be sure that the API is 29.



Click the play button on the right and the emulator should load.

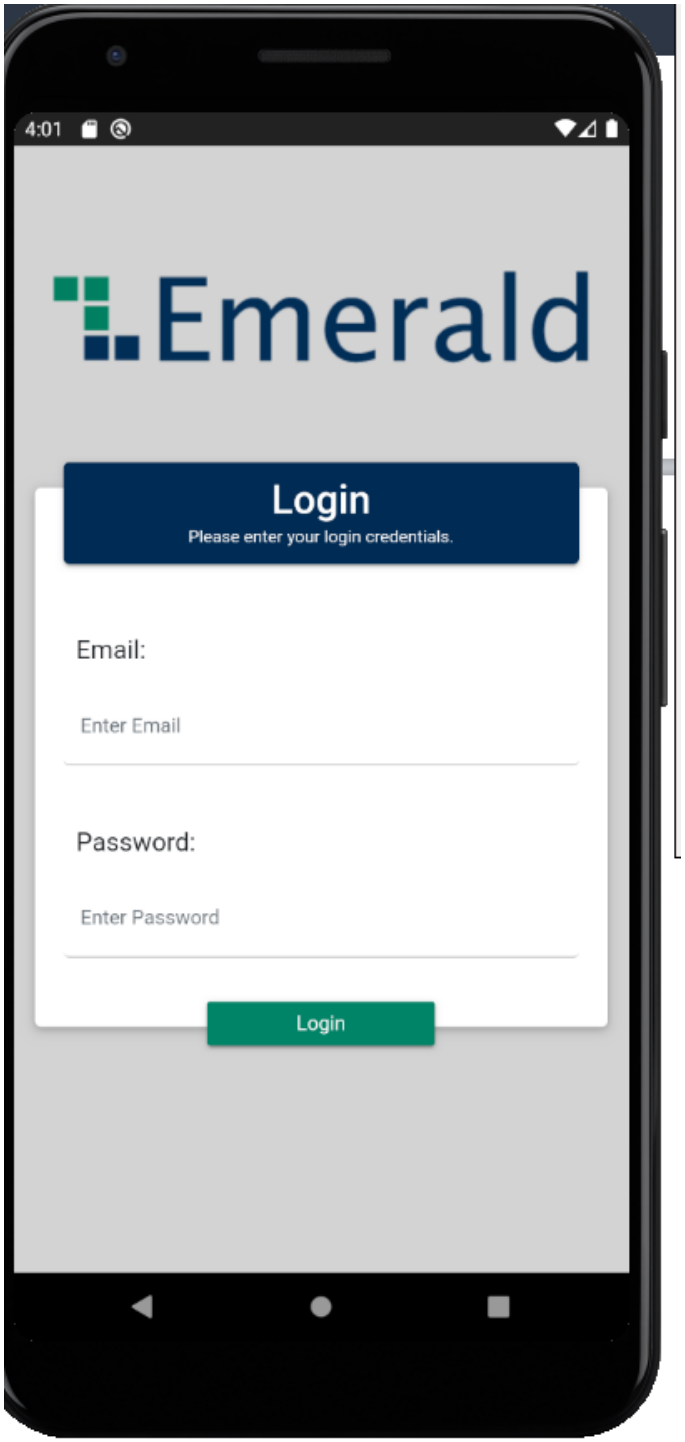

3:51
Saturday, Apr 24

Step 20: Run Cordova App

Go back to Git Bash and make sure you are in the folder opensourceintegration folder from page 38 of this document. Enter the following command in Git Bash.

```
cordova emulate android
```

The user should be able to see the app on the emulator.


4:01
Emerald
Login
Please enter your login credentials.
Email:
Enter Email
Password:
Enter Password
Login

Please carefully read the User Manual document to login with the respected roles such as patient, doctor or administrator. More details are provided to login and access the additional features of the app. Please carefully read the Software Configuration document which goes in more detail of how to set up the remote server that handles the database and back end logic of the app. Flask (Python web framework) and MySQL are used for the back end of the app.

5 Install App on Mobile Device

The user can run the app on a physical mobile device by following the step by step instructions below.

Step 1: Open Project Folder
Go to the OpenSource project folder created on page 18 of this document. Open the engine folder, then the CordovaApp folder, then the opensourceintegration folder, then right-click and "Git Bash Here". Leave this Git Bash terminal open until later.

Step 2: Configure On-Device Developer Options
The Settings app on Android includes a screen called Developer options that lets you configure system behaviors that help you profile and debug your app performance.

Step 3: Enable Developer Options and USB Debugging
On Android 4.1 and lower, the Developer options screen is available by default. On Android 4.2 and higher, you must enable this screen. To enable developer options, tap the Build Number option 7 times. You can find this option in one of the following locations, depending on your Android version:

- Android 9 (API level 28) and higher: Settings, About Phone, Build Number
- Android 8.0.0 (API level 26) and Android 8.1.0 (API level 26): Settings, System, About Phone, Build Number
- Android 7.1 (API level 25) and lower: Settings, About Phone, Build Number

At the top of the Developer options screen, the user can toggle the options on and off. The user probably wants to keep this on. When off, most options are disabled except those that don't require communication between the device and the users development computer.

Step 4: Before the user can use the debugger and other tools, the user must have a USB cable connected from the computer to the phone. The user needs to enable USB debugging on their phone, which allows Android Studio and other SDK tools to recognize the physical device when connected via USB. To enable USB debugging, toggle the USB debugging option in the Developer Options menu. The user can find this option in one of the following locations, depending on the users Android version:

- Android 9 (API level 28) and higher: Settings, System, Advanced, Developer Options, USB debugging
- Android 8.0.0 (API level 26) and Android 8.1.0 (API level 26): Settings, System, Developer Options, USB debugging
- Android 7.1 (API level 25) and lower: Settings, Developer Options, USB debugging

Step 5: Go back to the Git Bash terminal from Step 1 and type:

```
cordova run android --device
```

Be sure that the USB is connected to the phone from the computer and that USB debugging is turned on on the personal device. If all went well then the user should see the app on their personal android phone.

6 Conclusion

This document describes how to install the Open Source Integration app on the android emulator from Android Studio and the users personal device. Please read the User Manual on how to use the app and please read the Configuration Management on how the database and back end are set up on the remote server.