**Setting Up the Spectrum Toaster App Working Environment**

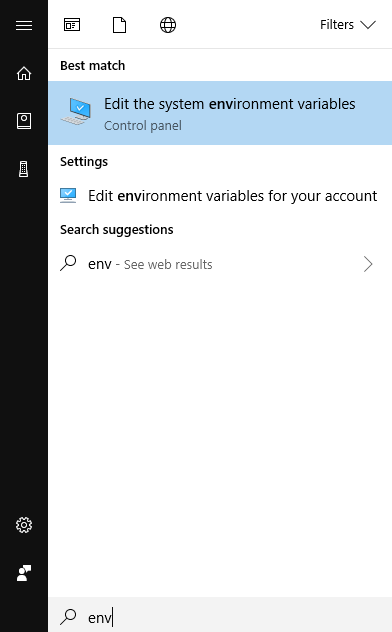
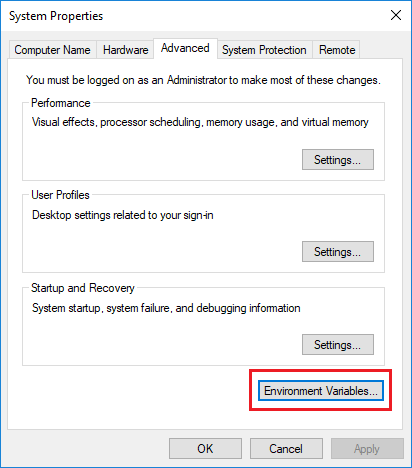
**Installing Flutter**:

Install the Flutter SDK version 2.5.3 (be sure to select the correct operating system): [*Here*](https://docs.flutter.dev/development/tools/sdk/releases)

Extract the downloaded zip file and place the Flutter folder into your desired install location.

NOTE: Avoid installing into a directory that may require elevated permissions, such as C:\ProgramFiles, C:\ProgramFilesX86.

Next up is adding the environment variable so that Flutter commands can be used through the command prompt.

To update the environment variables on your machine, open the start menu and begin a search for “env”. Select ‘Edit environment variables for your account’. Then, select ‘Environment Variables…’

Next, under ‘User Variables for [Account Name]’, look for a variable named ‘Path’ or ‘PATH’. If one does not exist, create a new environment variable using the ‘New…’ button. Then, append the full path to your Flutter bin inside the installation folder (shown in second image, mine just happened to be installed in my F drive.)

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, application, email

Description automatically generated

Now that your command prompt is set up for Flutter commands, open up command prompt and navigate to your flutter installation. Once there, run the command ‘flutter doctor’, it’ll look like this if Flutter is successfully installed:

Text

Description automatically generated

If you have any issues at this step, flutter doctor will be able to show you have to get what you need. Android Studio and the Android Toolchain (SDK) will need to be installed in the next section.

**Android Studio and Android SDK:**

First, download and install Android Studio: [*Here*](https://developer.android.com/studio#downloads).

After running the installer (or extracting and running the application), select ‘More Actions…’ and select SDK Manager. Inside the SDK Manager window, navigate to Android SDK and select the same android versions as selected below and apply:

Graphical user interface, text

Description automatically generated

Next, return to Android Studio and select ‘More Actions…’ again, but this time, select AVD Manager. From the AVD Manager window, select ‘Create Virtual Device…’, as well as the device to emulate. On the System Image page, select the release that is compatible with Android 30 or 31, the target Android version will be shown on the list. Lastly, name the device and select ‘Finish’. Finally, to run the emulator phone, return to the AVD Manager Window and run your created device.

Now, accept the Android Licensing to be able to test and use their emulator by running the following command in command prompt (or any elevated console):

flutter doctor –android-licenses

After accepting each license, run ‘flutter doctor’ again to make sure all the licenses were accepted.

**IDE Recommendations:**

Personally, I have used VS Code without any issues, there are a couple of setup steps needed to build and have type ahead for the Dart programming language.

The plugins I use for VS Code are [*Flutter*](https://marketplace.visualstudio.com/items?itemName=Dart-Code.flutter) and [*Dart*](https://marketplace.visualstudio.com/items?itemName=Dart-Code.dart-code).

Android Studio works as a relatively good IDE as the virtual Android phone we will be emulating is built into Android Studio. In addition, Android Studio is necessary to install anyway, it may just be easier to get set up rather than multiple applications.

Just use the software you are more used to, and if you are used to neither of these, they are not hard to set up.

**Running the App for Testing:**

Within Android Studio, open the AVD Manager window and create a virtual device if you haven’t already. Once your virtual device has booted up on your machine, open your IDE.

If you’re using Android Studio, the emulator will already be linked to your environment, you will just need to create a configuration to run the project.

If you’re using VSCode, the emulator may be linked with your environment from startup. If not, make sure you have the necessary plugins/extensions for running Flutter projects. Also make sure your VSCode is up to date.

Other IDEs work with the Android Studio emulators, but may require a process different than what is listed here.