**User Manual**

**For**

**GESTURE AUTOMATION SYSTEM FOR VARIOUS APPLICATIONS**

**Developed By**

**AKANKSHA RAUT**

**NIDHI DONTULWAR**

**HIMANSHU BHOLE**

**HRISHIKESH DHOLE**

* **Introduction**

There has been a great emphasis lately on Human Computer-Interaction (HCI) research to create easy-to-use interfaces by directly employing natural communication and manipulation skills of humans. Among different human body parts, the hand is the most effective general-purpose interaction tool, due to its dexterity. The word gesture is used for many different phenomena involving human movement, especially of the hands and arms only some of these are interactive or communicative.

* **Purpose**

The main purpose of gesture recognition is to identify a particular human gesture and convey information to the computer. The gesture recognition technique is an important technology for friendly human-computer interaction and has received a lot of attention in recent years. This kind of applications requires restricted background, set of gesture commands and a camera for image capturing. We have considered single handed gestures and their directional motion defines a gesture for the application. In this application image acquisition is done using a web cam.

1. **Installation Of Python**

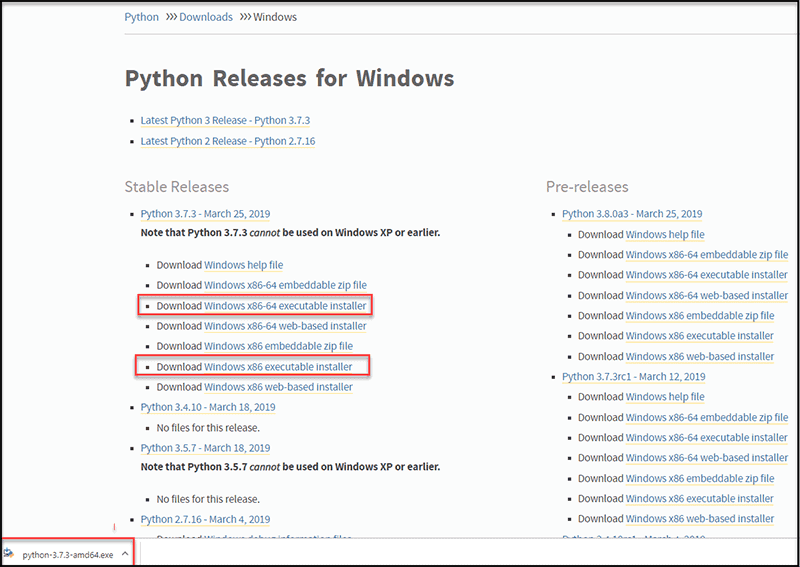
Python is a widely used high-level programming language first launched in 1991. Since then, Python has been gaining popularity and is considered as one of the most popular and flexible server-side programming languages.

Unlike most Linux distributions, Windows does not come with the Python programming language by default. However, you can install Python on your Windows server or local machine in just a few easy steps.

**Step 1: Select Version of Python to Install**

**Step 2: Download Python Executable Installer**

1. Open your web browser and navigate to the [Downloads for Windows section](https://www.python.org/downloads/windows/) of the [official Python website](https://www.python.org/).
2. Search for your desired version of Python. At the time of publishing this article, the latest Python 3 release is version 3.7.3, while the latest Python 2 release is version 2.7.16.
3. Select a link to download either the **Windows x86-64 executable installer** or **Windows x86 executable installer**. The download is approximately 25MB.

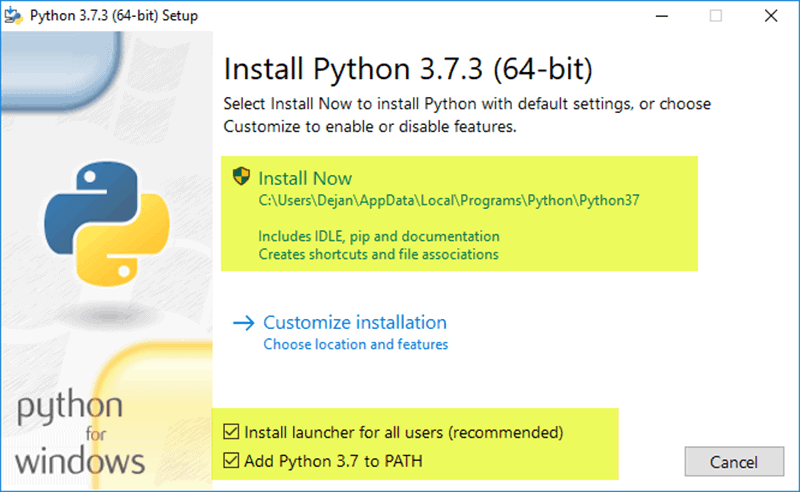


**Step 3: Run Executable Installer**

1. Run the **Python Installer** once downloaded. (In this example, we have downloaded Python 3.7.3.)

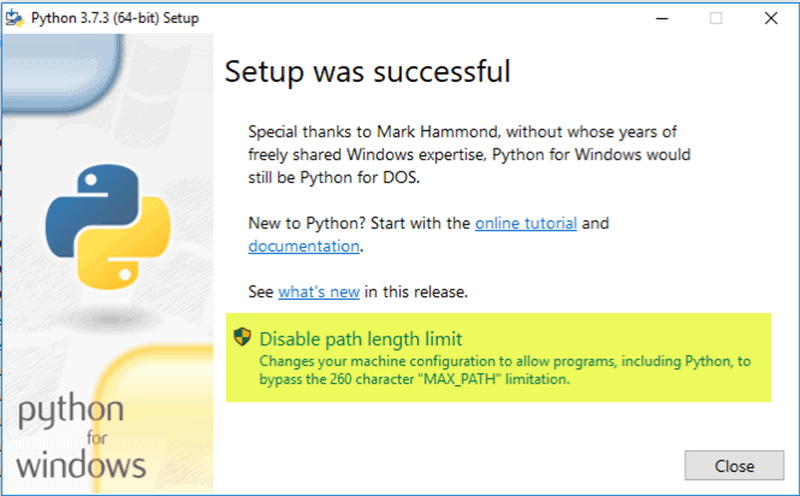
2. Make sure you select the **Install launcher for all users** and **Add Python 3.7 to PATH** checkboxes. The latter places the interpreter in the execution path. For older versions of Python that do not support the **Add Python to Path** checkbox.

3. Select **Install Now** – the recommended installation options.



For all recent versions of Python, the recommended installation options include **Pip** and **IDLE**. Older versions might not include such additional features.

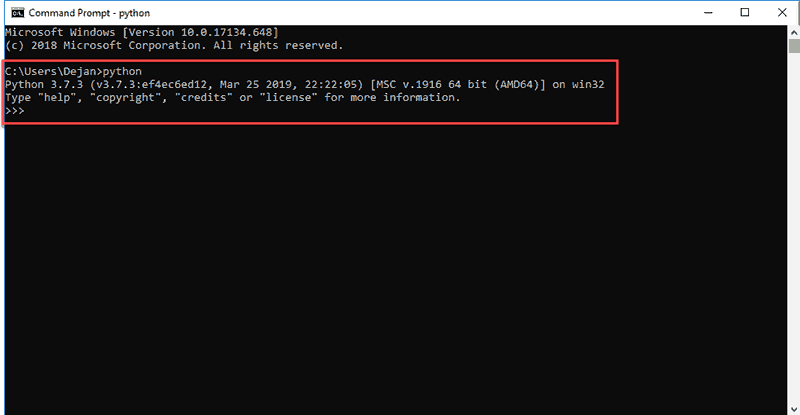
4. The next dialog will prompt you to select whether to **Disable path length limit**. Choosing this option will allow Python to bypass the 260-character MAX\_PATH limit. Effectively, it will enable Python to use long path names.



The **Disable path length** **limit** option will not affect any other system settings. Turning it on will resolve potential name length issues that may arise with Python projects developed in Linux.

**Step 4: Verify Python Was Installed on Windows**

1. Navigate to the directory in which Python was installed on the system. In our case, it is **C:\Users\*Username*\AppData\Local\Programs\Python\Python37**since we have installed the latest version.
2. Double-click **python.exe**.
3. The output should be similar to what you can see below:



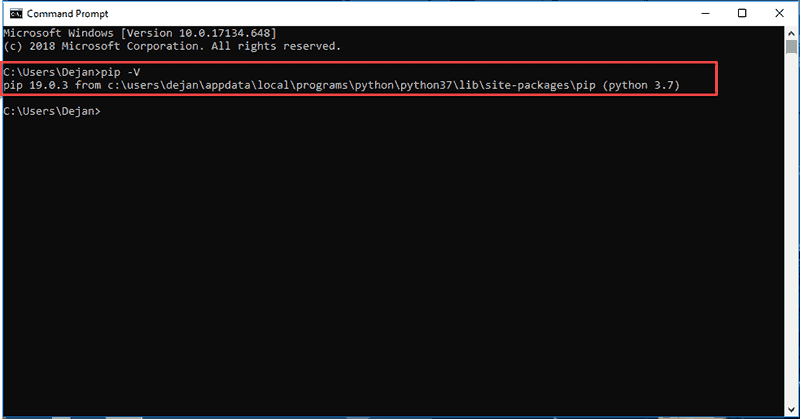
**Step 5: Verify Pip Was Installed**

If you opted to install an older version of Python, it is possible that it did not come with Pip preinstalled. Pip is a powerful package management system for Python software packages. Thus, make sure that you have it installed.

We recommend using Pip for most Python packages, especially when working in virtual environments.

To verify whether Pip was installed:

1. Open the **Start** menu and type "**cmd.**"
2. Select the **Command Prompt** application.
3. Enter **pip -V** in the console. If Pip was installed successfully, you should see the following output:



Pip has not been installed yet if you get the following output:

'pip' is not recognized as an internal or external command,

Operable program or batch file.

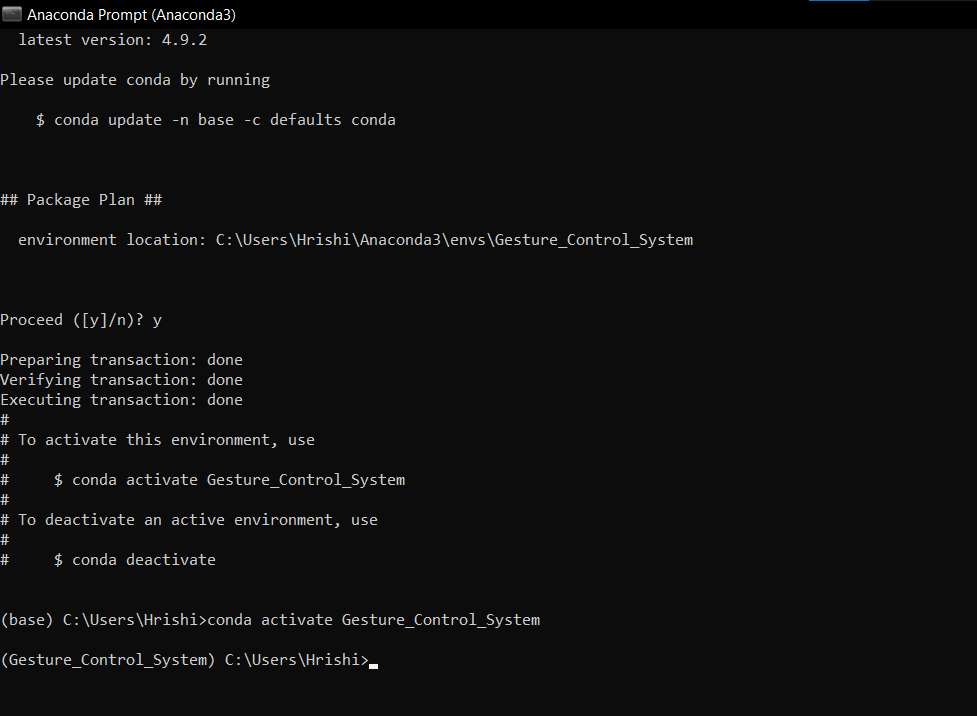
If your version of Python is missing Pip, see our article [How to Install Pip to Manage Python Packages on Windows](https://phoenixnap.com/kb/install-pip-windows).

1. **Installation Of Anaconda Navigator**

Follow the steps below to install the Anaconda distribution of Python on Windows.

**Steps:**

1. Visit [Anaconda.com/downloads](https://www.anaconda.com/download/)
2. Select Windows
3. Download the ***.exe*** installer
4. Open and run the ***.exe*** installer
5. Open the **Anaconda Prompt** and run some Python code
6. **Creation and Activation Of conda environment for the project**



1. **Installation Of libraries using pip used in the project**

Libraries used in our project are

* Keras package
* Pandas
* NumPy
* TensorFlow
* Matplotlib

**Step 1**

Open the Anaconda Prompt and create a new environment. I have named my environment new\_env—type in the command in my Anaconda Prompt terminal:

conda create --name new\_env

Make sure you run Anaconda Prompt as Administrator.

**Step 2**

Activate the environment, type:

activate new\_env

**Step 3**

Now, it’s the time to install Keras. Type the following command:

install -c anaconda keras

This will install keras and many other libraries, including NumPy, TensorFlow, etc. The command will take some time to download and install all the relevant packages.

Make sure you press y–(Yes) when asked to continue.

**Step 4**

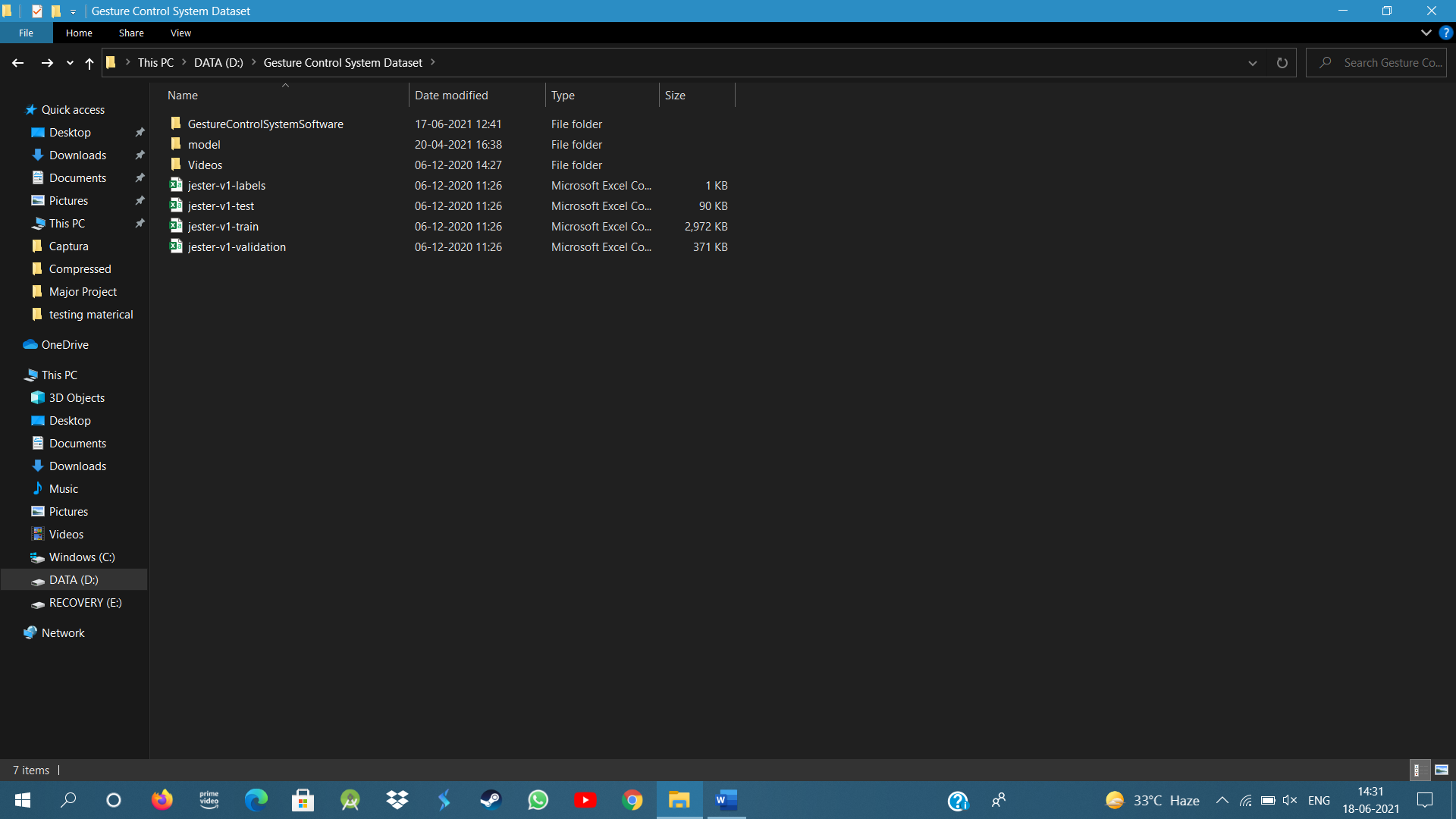
Since this is a new environment, a few more installations are required; otherwise, a ModuleNotFoundError could be generated. Many users using Anaconda either work with Spyder or Jupyter Notebook. Therefore, a user is required to run the following commands, one by one:

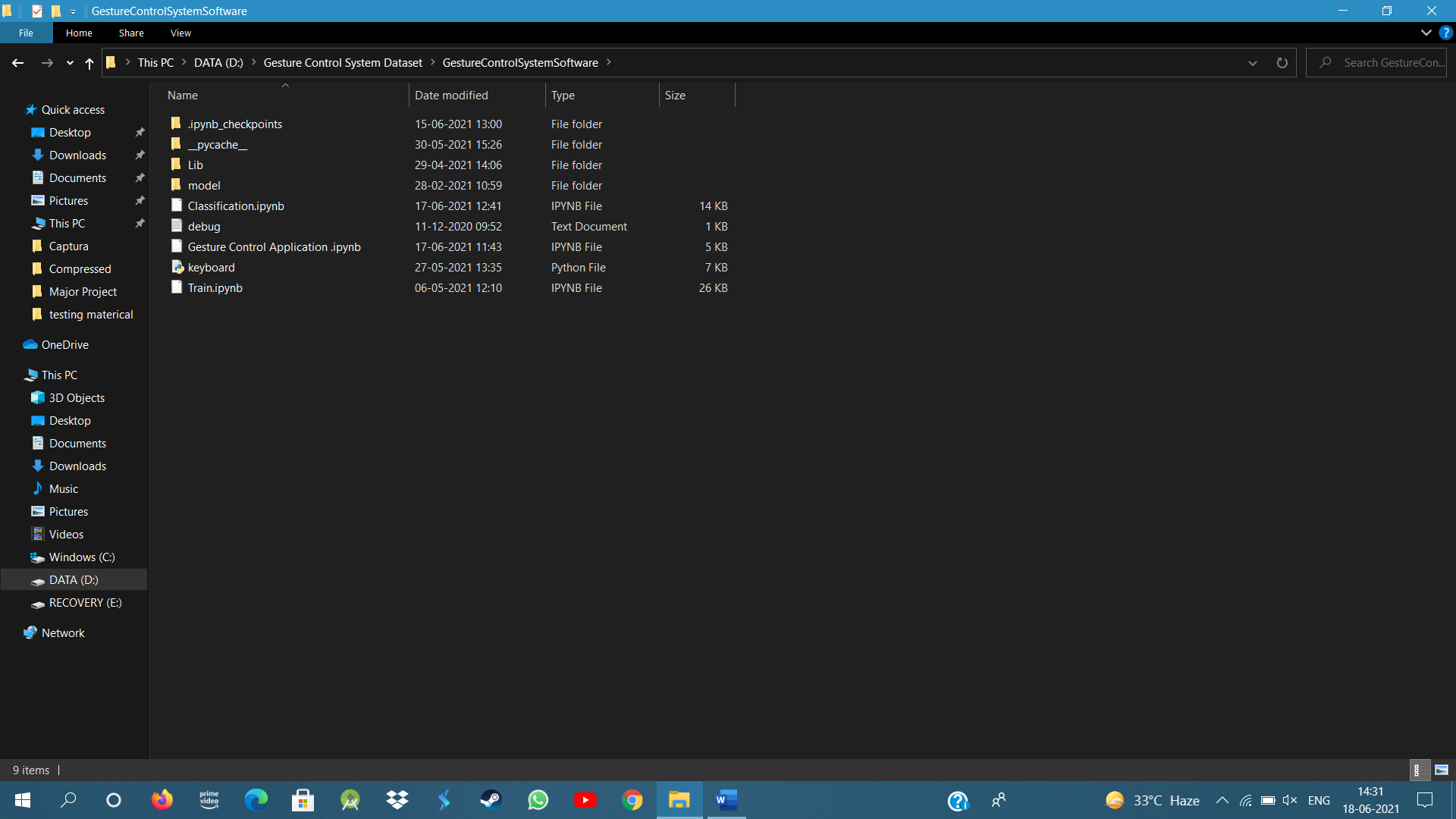
conda install jupyter

If any of the packages are still missing, install the library using the following command:

conda install <package>  
// package could be matplotlib, pandas, etc.

After the installation of the above requirements go to the project file in your system

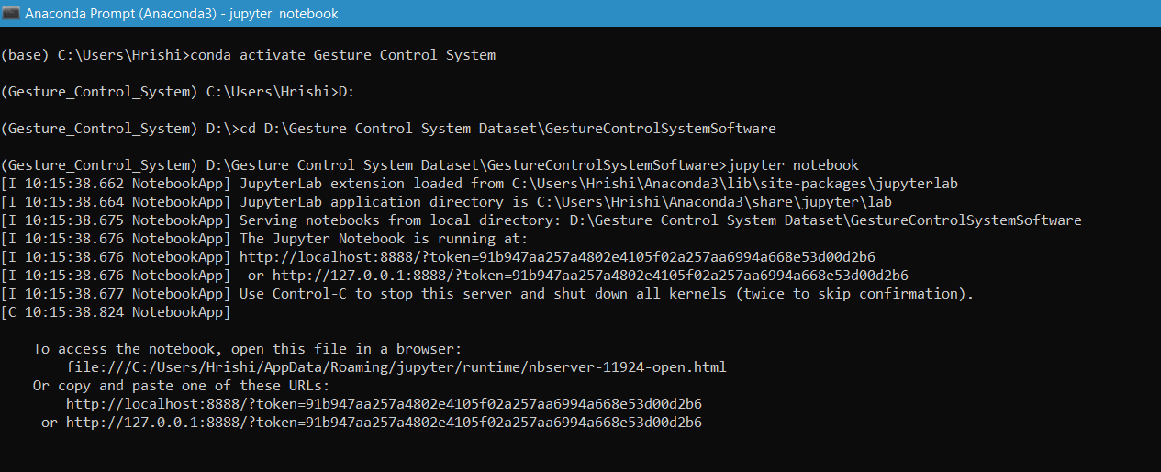




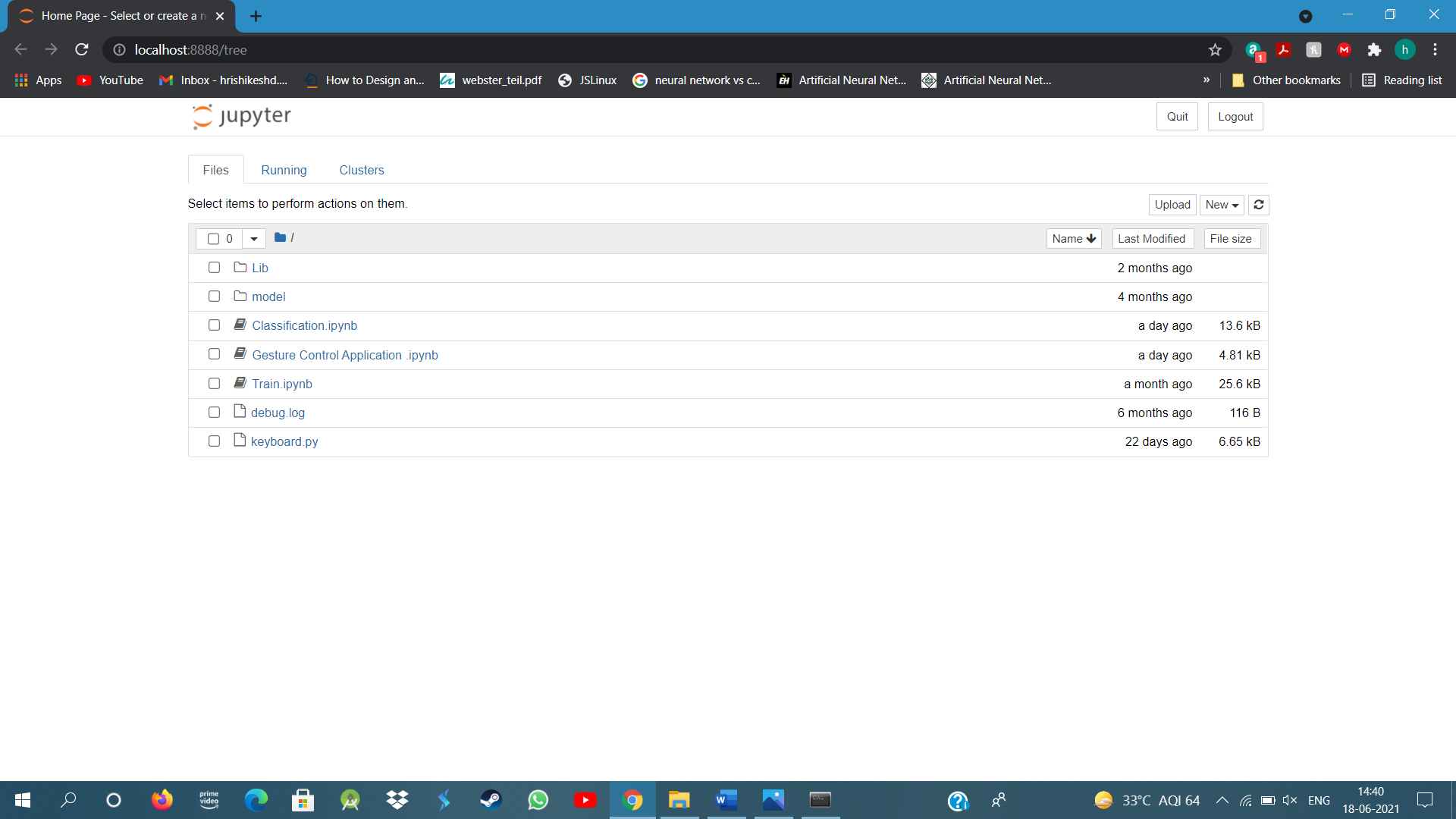
Copy the path for the root directory to access the files in jupyter notebook

****

After this activate Jupyter notebook through conda environment created for your project



After this open “Gesture Control Application.ipynb” in jupyter notebook



After open the file run all the cells given in the file and open the windows media player to access the gestures to automate the Windows media player.

**Gestures implemented to control the Windows media player**

1] The Drumming Fingers Gesture is used to play the video in Windows Media Player.



2] The Stop Sign is used to pause the video in Windows Media Player.



3] The Swiping Up gesture is used to increase the volume of the Windows Media Player.



4] The Swiping Down gesture is used to decrease the volume of the Windows Media Player.



5] The Thumbs Up gesture is used to mute the volume of the Windows Media Player



6] The Swiping left gesture is used to skip the video forward in the Windows Media Player.



7] The Swiping Right gesture is used to skip the video reverse in the Windows Media Player.

