

Facial Recognition for Attendance Checking and Eye-Tracking
to Determine the Student's Attention During Online Class
Using Mask Region-Based Convolutional Neural Network
Algorithm

USER'S MANUAL

Breva, Ian Robin
Lumawag, James Bryan
Macawili, Edcel Jeremy
Parco, Ian

I have carefully evaluated the User's Manual for Facial Recognition for Attendance Checking and Eye-Tracking to Determine the Student's Attention During Online Class Using Mask Region-Based Convolutional Neural Network Algorithm.

This document has been completed in accordance with the requirements of the School of Computer Studies and Technology System Development Methodology.

Certification

_____ The User's Manual is accepted.

_____ The User's Manual is accepted pending the changes noted.

Research Adviser's Signature Over Printed Name

Moderator's Signature Over Printed Name

TABLE OF CONTENTS

Description	Page
1.0 General Information	1
1.1 System Overview	2
1.2 Help Desk	3
2.0 System Summary	5
2.1 Hardware and Software Requirements	6
2.2 User Types	7
3.0 Installation	9
3.1 Microsoft Build Tools	10
3.2 Git	13
3.3 Python	16
4.0 Setting Up the System	20
4.1 Python Dependencies	21
4.2 Agora SDK	26

5.0	Running the System	29
5.1	Video Conference App	30
6.0	Using the System	31
6.1	Video Conferencing	32
6.2	Facial Recognition	34
6.3	Eye-Tracking	35

1.0 GENERAL INFORMATION

SYSTEM OVERVIEW

The system that we have developed is a web-based application that can capture and analyze video and audio data to enable real-time video conferencing, with the following additional functionalities or features:

User Interface:

- A web application accessible through a web browser, allowing users to interact with the video conferencing application.

Video and Audio Capture:

- These modules will access the camera and microphone on the user's device to capture a live feed of the video and audio of the user's face and voice.

Facial Recognition:

- This module will take a screenshot of the current state of the video conference and analyze the faces captured on the screenshot to identify and provide information such as name and time of capture.

Eye-Tracking:

- This module will track the user's eye contours and analyze it to provide insights such as the user's engagement or attention during the conference call.

HELP DESK

For technical support, please contact the following:

Name: Ian Robin Breva
Email: ianbrevva12@gmail.com
Contact: 09668437480



Name: James Bryan Lumawag
Email: lumawagjamesbryan@gmail.com
Contact: 09563168026



Name: Edcel Jeremy Macawili
Email: edceljmacawili@gmail.com
Contact: 09278031712



Name: Ian Parco
Email: parcoian0929@gmail.com
Contact: 09472542683



2.0 SYSTEM SUMMARY

Hardware Requirements

Hardware	Minimum Requirement
Processor	Intel i3 (7th Gen)/AMD Ryzen 3 (2nd Gen)
Graphics Card	Nvidia 1050/AMD RX460
RAM	8GB
Disk Space	128GB
Camera	720p (1280x720)

Software Requirements

Installation:

- Git
- Python
- Visual C++

Usage:

- Web Browser (Google Chrome, Mozilla Firefox)

User Types

Faculty:

- Can access the web application and can interact with the user interface and functionalities
- Has the facial recognition functionality

Student:

- Can access the web application and interact with the user interface
- Cannot use the facial recognition functionality
- Has the eye-tracking functionality

3.0 INSTALLATION

Microsoft C++ Build Tools

The Microsoft C++ Build Tools provides MSVC toolsets via a scriptable, standalone installer without Visual Studio. It includes compilers, linkers and other development tools for building applications in various programming languages.

1. To start, first go to this link: <https://visualstudio.microsoft.com/visual-cpp-build-tools/> and download the installer.

Microsoft C++ Build Tools

Standalone MSVC compiler, libraries, and scripts

Download Build Tools

2. After downloading the installer, run the executable and click '**Continue**'.



Visual Studio Installer

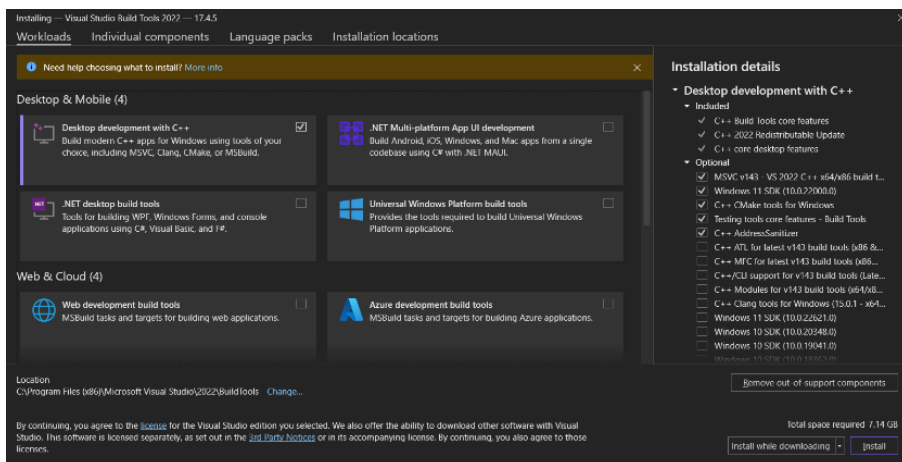
Before you get started, we need to set up a few things so that you can configure your installation.

To learn more about privacy, see the [Microsoft Privacy Statement](#).

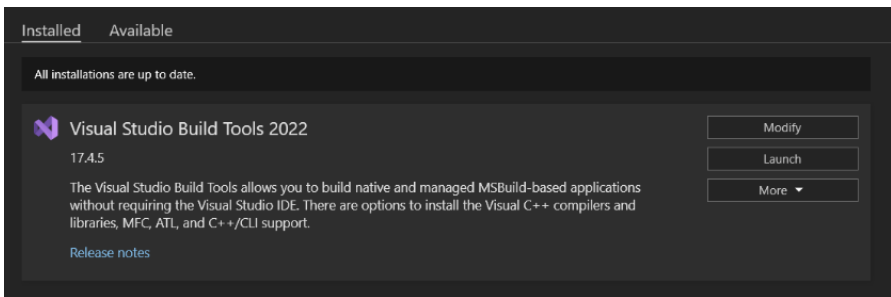
By continuing, you agree to the [Microsoft Software License Terms](#).

Continue

3. Choose ‘Desktop development with C++’ and then click ‘Install’.



4. The download and installation may take a while as the file size is large, but after the installation process, it should look like the image below to indicate that the installation is successful.



Git

Git is a popular version control system used for managing and tracking changes to code or other types of files. It allows developers to collaborate on projects with ease, as they can work on different parts of the codebase independently and merge their changes together seamlessly. Git also provides a complete history of all changes made to the code, making it easy to revert to previous versions if necessary.

1. To install git, download the installer from the website: <https://git-scm.com/download/win> and choose the appropriate version for your system.

Download for Windows

[Click here to download](#) the latest (2.39.2) 64-bit version of Git for Windows. This is the most recent [maintained build](#). It was released 5 days ago, on 2023-02-14.

Other Git for Windows downloads

Standalone Installer

[32-bit Git for Windows Setup.](#)

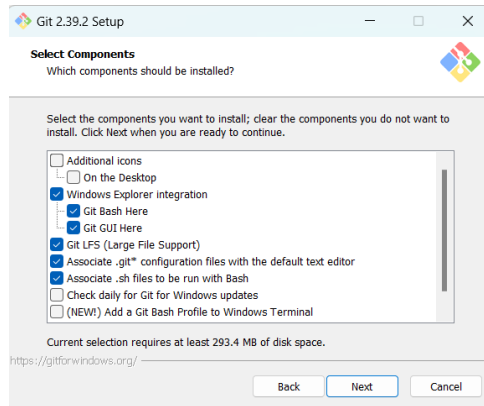
[64-bit Git for Windows Setup.](#)

Portable ("thumbdrive edition")

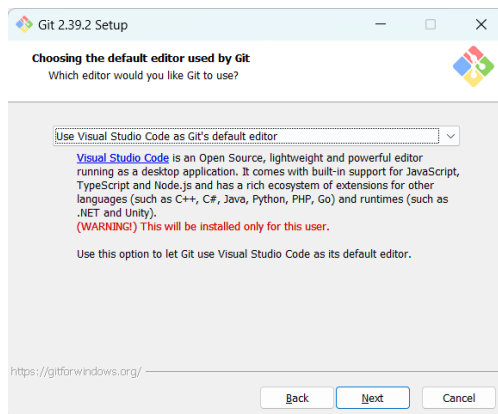
[32-bit Git for Windows Portable.](#)

[64-bit Git for Windows Portable.](#)

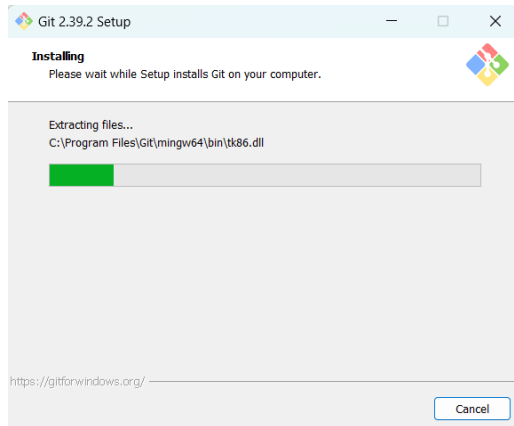
2. Run the downloaded installer and proceed with the steps.



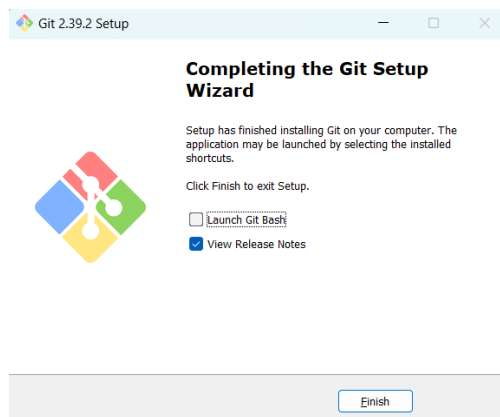
3. In one of the steps, choose your default code editor for the installation process. In our case, we are using Visual Studio Code.



4. For the next few steps, it is entirely dependent on the user's preference, but we recommend just using the default settings for installation.



5. If the installation is successful, it should show an image similar to the one below.



Python

Python is a high-level, interpreted programming language known for its simplicity, readability, and versatility. It is widely used in a variety of applications such as web development, data analysis, artificial intelligence, and scientific computing.

1. Go to the official Python website at <https://www.python.org/downloads/> and look for the latest version of Python 3.10.

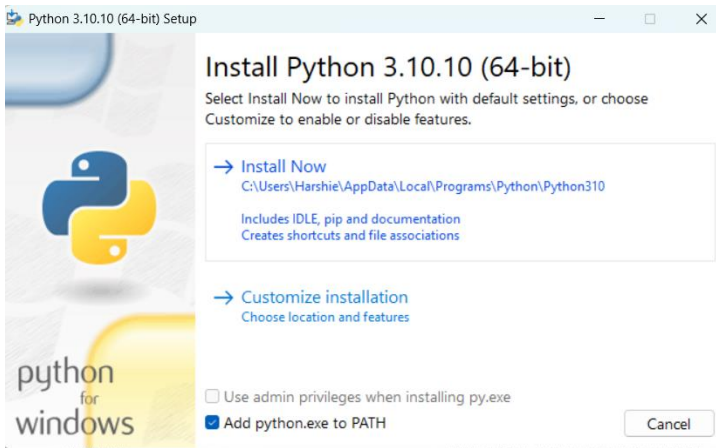
Looking for a specific release?

Python releases by version number:

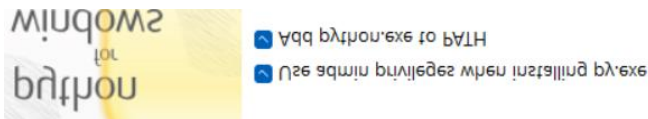
Release version	Release date	Click for more	
Python 3.10.10	Feb. 8, 2023	Download	Release Notes
Python 3.11.2	Feb. 8, 2023	Download	Release Notes
Python 3.11.1	Dec. 6, 2022	Download	Release Notes
Python 3.10.9	Dec. 6, 2022	Download	Release Notes
Python 3.9.16	Dec. 6, 2022	Download	Release Notes
Python 3.8.16	Dec. 6, 2022	Download	Release Notes
Python 3.7.16	Dec. 6, 2022	Download	Release Notes

[View older releases](#)

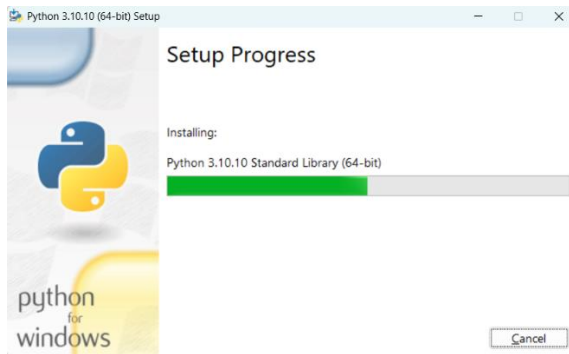
2. Once the download is complete, locate the installer file and double-click on it to begin the installation process.



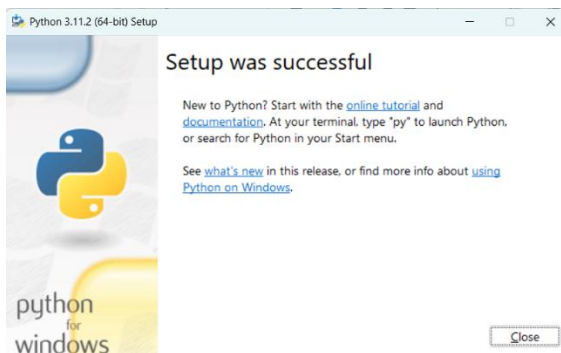
3. In the Python Setup window, select the **'Install Now'** option and ensure that the **'Add python.exe to PATH'** option is checked.



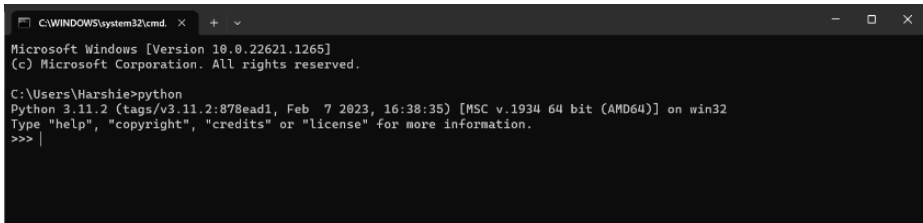
4. Click on the **'Customize installation'** option if you want to change any installation settings. Otherwise, click on the **'Install Now'** button to start the installation. Wait for the installation process to complete. This may take several minutes, depending on your system and the version of Python you are installing.



5. Once the installation is complete, click on the **'Close'** button to close the installer.



6. To verify that Python has been installed correctly, open a command prompt and type **'python'** without the quotes. If Python has been installed correctly, you should see the Python interpreter prompt, which looks like **'>>>'**.



```
C:\WINDOWS\system32\cmd. x + v
Microsoft Windows [Version 10.0.22621.1265]
(c) Microsoft Corporation. All rights reserved.

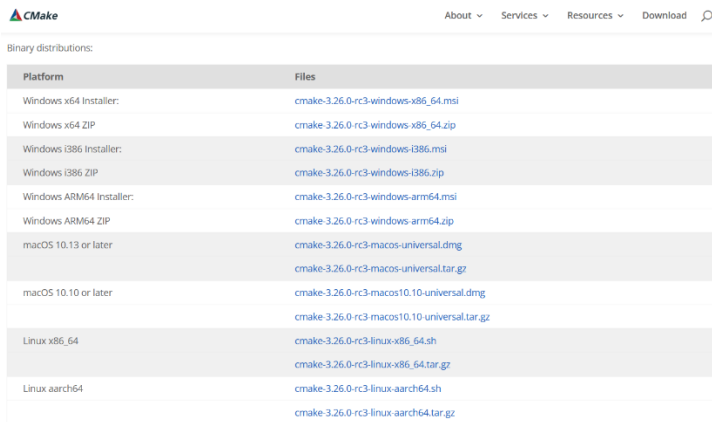
C:\Users\Harshie>python
Python 3.11.2 (tags/v3.11.2:878ead1, Feb  7 2023, 16:38:35) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> |
```


4.0 SETTING UP THE SYSTEM

Python Dependencies

For this system, there are multiple Python packages needed to make sure that the application is running well and smoothly.

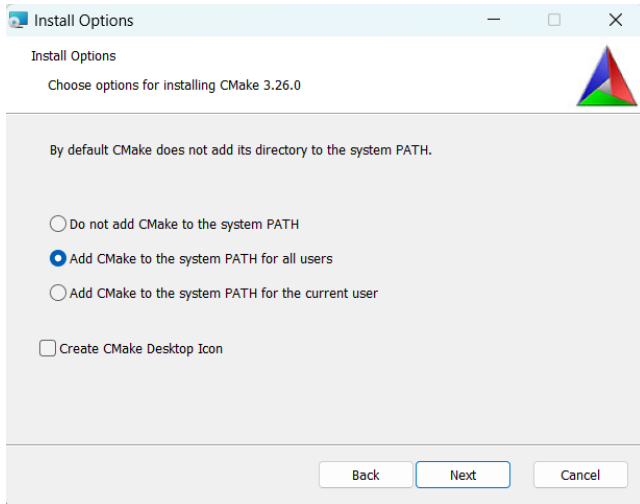
1. The first dependency we must install is CMake. From the website <https://cmake.org/download/>, choose the appropriate version for your system and download it.



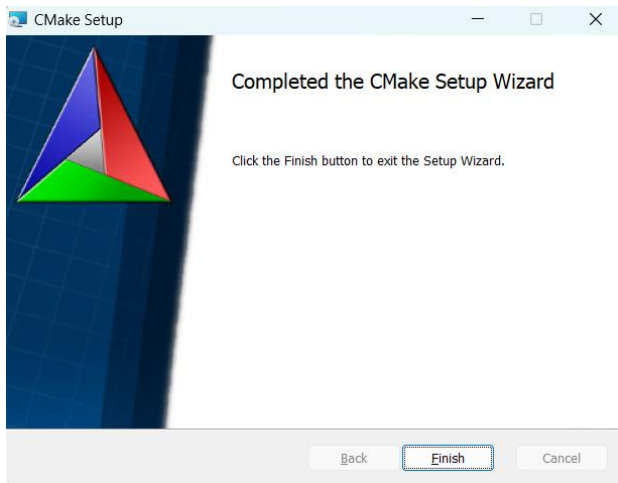
The screenshot shows the CMake website's 'Binary distributions' page. It features a table with two columns: 'Platform' and 'Files'. The table lists download links for various operating systems and architectures, including Windows (x64, i386, ARM64), macOS (10.10 or later, 10.13 or later), and Linux (x86_64, aarch64). The CMake logo is in the top left, and navigation links (About, Services, Resources, Download) are in the top right.

Platform	Files
Windows x64 Installer:	cmake-3.26.0-rc3-windows-x86_64.msi
Windows x64 ZIP	cmake-3.26.0-rc3-windows-x86_64.zip
Windows i386 Installer:	cmake-3.26.0-rc3-windows-i386.msi
Windows i386 ZIP	cmake-3.26.0-rc3-windows-i386.zip
Windows ARM64 Installer:	cmake-3.26.0-rc3-windows-arm64.msi
Windows ARM64 ZIP	cmake-3.26.0-rc3-windows-arm64.zip
macOS 10.13 or later	cmake-3.26.0-rc3-macos-universal.dmg cmake-3.26.0-rc3-macos-universal.tar.gz
macOS 10.10 or later	cmake-3.26.0-rc3-macos10.10-universal.dmg cmake-3.26.0-rc3-macos10.10-universal.tar.gz
Linux x86_64	cmake-3.26.0-rc3-linux-x86_64.sh cmake-3.26.0-rc3-linux-x86_64.tar.gz
Linux aarch64	cmake-3.26.0-rc3-linux-aarch64.sh cmake-3.26.0-rc3-linux-aarch64.tar.gz

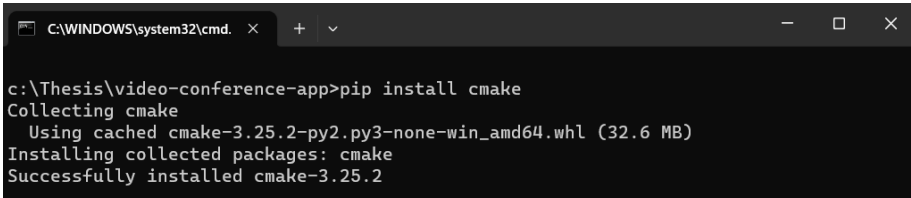
2. Run the downloaded installer and proceed with the usual steps until you reach the **'Install Options'** part of the setup. Choose the setting **'Add CMake to the system PATH for all users'** during the process of installation, which can also be seen in the image below.



3. After the installation process, click **‘Finish’** to complete the setup.

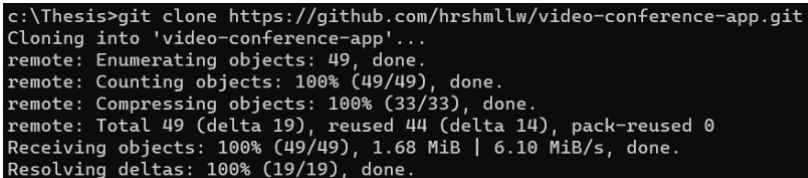


4. To finalize installing the CMake dependency, open a command prompt, and type in **'pip install cmake'**.



```
C:\WINDOWS\system32\cmd. x + v
c:\Thesis\video-conference-app>pip install cmake
Collecting cmake
  Using cached cmake-3.25.2-py2.py3-none-win_amd64.whl (32.6 MB)
Installing collected packages: cmake
Successfully installed cmake-3.25.2
```

5. To install the other necessary dependencies, we must first clone the project repository from <https://github.com/hrshmlw/video-conference-app> to our device. To do this, in a command prompt, type in **'git clone https://github.com/hrshmlw/video-conference-app.git'**. It should show something similar to the image below.



```
c:\Thesis>git clone https://github.com/hrshmlw/video-conference-app.git
Cloning into 'video-conference-app'...
remote: Enumerating objects: 49, done.
remote: Counting objects: 100% (49/49), done.
remote: Compressing objects: 100% (33/33), done.
remote: Total 49 (delta 19), reused 44 (delta 14), pack-reused 0
Receiving objects: 100% (49/49), 1.68 MiB | 6.10 MiB/s, done.
Resolving deltas: 100% (19/19), done.
```

6. Using the command prompt, navigate to the folder containing the project files, and type in **'pip install -r requirements.txt'**. If successful, it should show something similar to the image below.

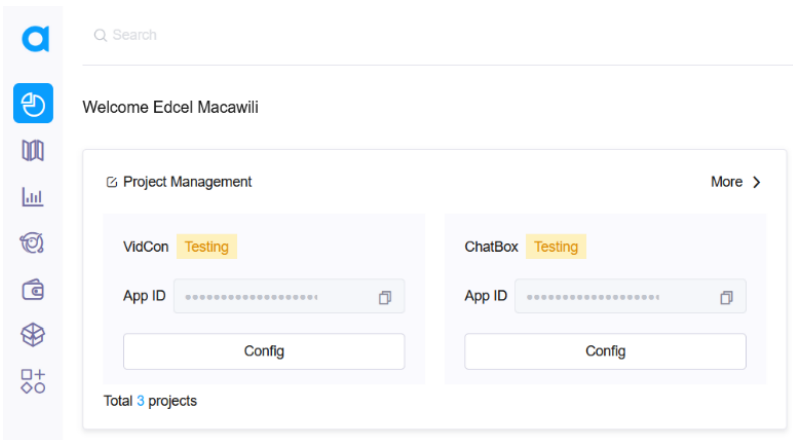
```
C:\WINDOWS\system32\cmd.exe
C:\Users\harshie\appdata\local\pip\cache> pip install -r requirements.txt
Collecting face_recognition==1.3.0
  Using cached face_recognition-1.3.0-py2.py3-none-any.whl (15 MB)
Collecting Flask==2.2.2
  Using cached Flask-2.2.2-py3-none-any.whl (101 kB)
Collecting numpy==1.23.0
  Using cached numpy-1.23.0-cp310-cp310-win_amd64.whl (34.6 MB)
Collecting Pillow==9.3.0
  Using cached Pillow-9.3.0-cp310-cp310-win_amd64.whl (2.5 MB)
Collecting python-dotenv==0.21.0
  Using cached python_dotenv-0.21.0-py3-none-any.whl (10 kB)
Collecting dlib==19.9
  Using cached dlib-19.24.0.tar.gz (3.2 MB)
  Preparing metadata (setup.py) ... done
Collecting click==8.1.3
  Using cached click-8.1.3-py3-none-any.whl (96 kB)
Collecting face_recognition_models==0.3.0
  Using cached face_recognition_models-0.3.0-py2.py3-none-any.whl
Collecting Jinja2==3.1.2
  Using cached Jinja2-3.1.2-py3-none-any.whl (133 kB)
Collecting itsdangerous==2.0
  Using cached itsdangerous-2.1.2-py3-none-any.whl (15 kB)
Collecting Werkzeug==2.2.3
  Using cached Werkzeug-2.2.3-py3-none-any.whl (235 kB)
Collecting colorama==0.4.6
  Using cached colorama-0.4.6-py2.py3-none-any.whl (25 kB)
Collecting MarkupSafe==2.1.2
  Downloading MarkupSafe-2.1.2-cp310-cp310-win_amd64.whl (16 kB)
Building wheels for collected packages: dlib
  Building wheel for dlib (setup.py) ... done
  Created wheel for dlib: filename=dlib-19.24.0-cp310-cp310-win_amd64.whl size=2956615 sha256=8b9f770e586e82862d6e92a744eb8780049d6dd8af6da818d4f15bd5a80ddac
  Stored in directory: c:\users\harshie\appdata\local\pip\cache\wheels\49\fd\12\8db371b40ce7b38c2b5a1f96f73bf7d9712db16b09c1ef13f
Successfully built dlib
Installing collected packages: face_recognition_models, dlib, python-dotenv, Pillow, numpy, MarkupSafe, itsdangerous, colorama, Werkzeug, Jinja2, Click, Fla
ak, face_recognition
Successfully installed Click-8.1.3 Flask-2.2.2 Jinja2-3.1.2 MarkupSafe-2.1.2 Pillow-9.3.0 Werkzeug-2.2.3 colorama-0.4.6 dlib-19.24.0 face_recognition
_models-0.3.0 face_recognition-1.3.0 itsdangerous-2.1.2 numpy-1.23.0 python-dotenv-0.21.0
```

Agora SDK

Agora SDK is a software development kit that enables real-time audio and video communication in web and mobile applications. It provides a set of APIs and tools for developers to integrate voice and video chat features into their applications.

To make use of the Agora SDK in the system, we must create a token to connect to the servers, which in turn gives us access to creating our own video conference rooms.

1. Access the Agora console and choose the project you are working on by clicking '**Config**'.



2. Scroll down and look for the **‘Generate temp RTC token’** link, as seen below.

Web demo	Generate link
Temp token for audio/video call	Generate temp RTC Token
Flexible Classroom	Enable
Media Push	Enable
Cloud Recording	Enable
Whiteboard	Enable
Media Pull	Enable
Chat	Enable/Configure
Cloud Proxy (Force UDP and	Enable

3. Next, input the channel name of the system you are trying to access. In this instance, we will use the **‘livestream’** channel, and click **‘Generate’**.

App ID: 0814c9b1c6bb4a7aae18007e82a33091

App certificate:



Channel Name:

livestream

Temp Token:

Generate

[Back](#)

[To learn how to deploy a token server, please click here](#)

- Copy the generated token and input it in the source code for the project.

```
1 // Agora.io web SDK config credentials
2 const APP_ID = "8014c9b1c6bb4a7aae18007e82a13091"
3 const TOKEN = "0b7ec2c9f10abxviedtpp/qplqPeo0kZlf15QCLYh/v3nThUzTSyWwGVCNjKwTJL5J3NE9fTMdMdwITPLUSJQZmABTT/9lhuMyMIgonuEHZEBAKFBl0oc3LIUapKL1HRCBgvA+o4Xa=/"
4 const CHANNEL = "livestream"
```

5.0 RUNNING THE SYSTEM

Video Conference App

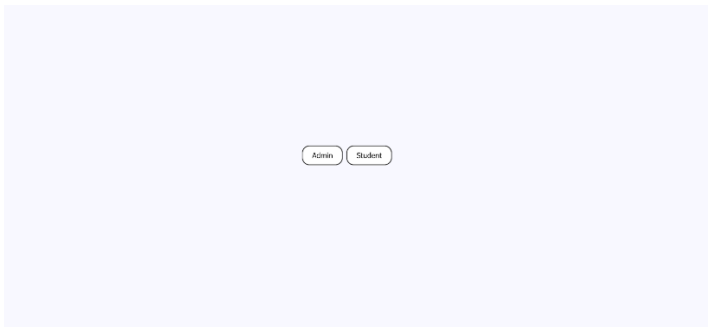
1. To run the Video Conference App, first open up a command prompt, and navigate to the directory where the project is located.
2. Type in **'flask run'**.
3. The process of running the server should look similar to the image shown below.

```
please C:\B+C fo dntf
* Running on http://127.0.0.1:2000
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Debug mode: off
* Restarting with f5bb ,setlax:by,
C:/Users/NT900-condaconda-9bb>f5bb Lnu
```

6.0 USING THE SYSTEM

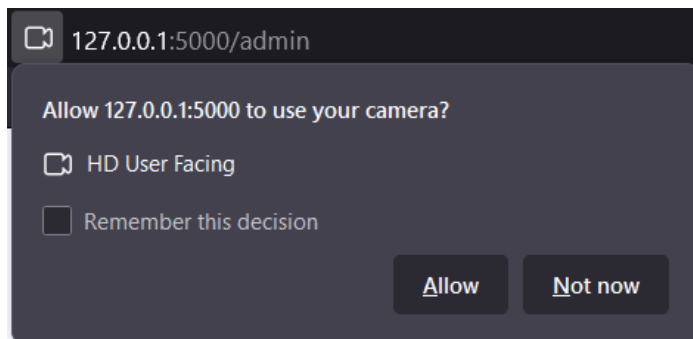
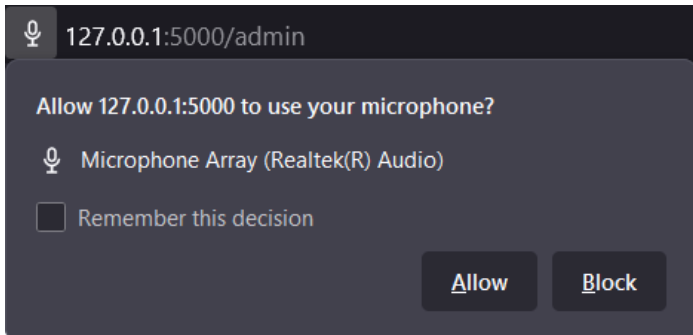
Video Conferencing

1. To access the video conferencing app using a web browser, go to the link provided by the command used in the previous section **'flask run'** which is **'http://127.0.0.1:5000'**. You can easily access it by pressing Ctrl and clicking the link in the command prompt, which will open your default web browser.



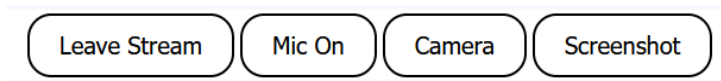
2. You can choose your role, whether you are a faculty or a student.
3. The default functionalities of both roles are the mic and camera toggles, and join and leave stream.

4. Upon joining the stream, it will request access for both your device's microphone and camera.



Facial Recognition

1. The facial recognition functionality is given to the Admin role. It can be accessed by clicking on the role when opening the video conference app and joining the stream.
2. To use the facial recognition function, the admin can simply click the **'Screenshot'** button which can be seen on the bottom part of the page.



3. Directly after, the application will take a screenshot of the video conference room and apply the facial recognition algorithm to detect and recognize all the faces within the room.

Eye-Tracking

1. The eye-tracking functionality is only given to the student role. It is turned on by default and cannot be turned off.
2. As soon as the user joins the stream under the Student role, the eye-tracking function will be activated.



3. The eye-tracking function will send out notifications to the whole room when a student's eyes are not detected in the frame.

