1. **INTRODUCTION**

Attendance being a very necessary side of administration may normally become an arduous, redundant activity, pushing itself to inaccuracies. The traditional approach of making roll calls proves itself to be a statute of limitations as it is very difficult to call names and maintain its record especially when the ratio of students is high. Some organizations use document-oriented Approach and others have implemented these digital methods such as biometric fingerprinting techniques and card swapping techniques. however, these methods prove to be a statute of limitations as it subjects students to wait in a time-consuming queue.

The system of intelligent attendance is generally implemented with biometrics help. Recognition of face is one of the Biometric ways of improving this system. Face recognition proved to be a productive method for taking attendance. It requires very few hardware resources hence it is a cost-friendly system.

**1.1 Scope**

This project can be used in schools, colleges and in organizations ,where ever the attendance is calculated or considered.

**1.2 Existing System**

The existing system makes the use of Haar classifiers, KNN, CNN, SVM, Generative

adversarial networks, and Gabor filters.

**The following are the drawbacks of the existing manual System:**

* Time Delay
* It is very complicated system

**1.3 Proposed System**

The proposed system makes use of face\_recognition library. The proposed system is very simple, effortless, and manageable with lucid operations. It embraces a database of student's faces and their details like name, enrolment number, course. Then we capture the image of students. Once the image acquisition is done when the teacher\admin triggers the system by making a click on the start button thereafter system will undergo face detection. after the faces are detected in an image taken by all cameras at all given instances then detected faces will be compared with stored images of the students in the database. Once the face is matched then present is marked in front of its corresponding enrollment number and name in excel format.

**MERITS-**

* It is very simple to use.
* It takes less time.

1. **SYSTEM ANALYSIS**

This System Analysis is closely related to [requirements analysis](http://en.wikipedia.org/wiki/Requirement_analysis). It is also "an explicit formal inquiry carried out to help someone (referred to as the decision maker) identify a better course of action and make a better decision than he might otherwise have made."This step involves [breaking down](http://en.wikipedia.org/wiki/Work_breakdown_structure) the system in different pieces to analyze the situation, analyzing project goals, breaking down what needs to be created and attempting to engage users so that definite requirements can be defined.

**2.1 Functional Requirement Specification**

The System after careful analysis has been identified to be present with the following modules.

* 1. **Faces Extraction Module:**

In the faces Extraction module we will be doing two functions , one is to capture the image and another is to extract faces of that image and returning the path of the folder.

* 1. **dbms Module:**

This module has all the functions related to database operations to be done like , updating attendance or updating student details.

* 1. **UI Module:**

In this module we have all the UI related code.

* 1. **Main Module:**

In this module we integrate all the previous three modules.

**2.2 Performance Requirements**

Performance is measured in terms of the output provided by the application. Requirement specification plays an important part in the analysis of a system. Only when the requirement specifications are properly given, it is possible to design a system, which will fit into required environment. It rests largely with the users of the existing system to give the requirement specifications because they are the people who finally use the system. This is because the requirements have to be known during the initial stages so that the system can be designed according to those requirements. It is very difficult to change the system once it has been designed and on the other hand designing a system, which does not cater to the requirements of the user, is of no use.

The requirement specification for any system can be broadly stated as given below:

* The system should be able to interface with the existing system
* The system should be accurate
* The system should be better than the existing system

The existing system is completely dependent on the user to perform all the duties.

**2.3 Software Requirements**:

* **Operating System** : Windows 10
* **IDE** : SPYDER
* **Language** : Python 3.8 and other libraries likes tkinter,opencv2,etc..

**2.4 Hardware Requirements**:

* **Processor** : i5
* **RAM** : 8 GB RAM
* **Hard disk** : 1TB

**3.SYSTEM DESIGN**

Systems design is the process of defining the architecture, components, modules, interfaces, and [data](http://en.wikipedia.org/wiki/Data) for a [system](http://en.wikipedia.org/wiki/System) to satisfy specified [requirements](http://en.wikipedia.org/wiki/Requirement). One could see it as the application of [systems theory](http://en.wikipedia.org/wiki/Systems_theory) to [product development](http://en.wikipedia.org/wiki/Product_development). [Object-oriented analysis and design](http://en.wikipedia.org/wiki/Object-oriented_analysis_and_design) methods are becoming the most widely used methods for computer systems design.

**3.1 Architectural Design**

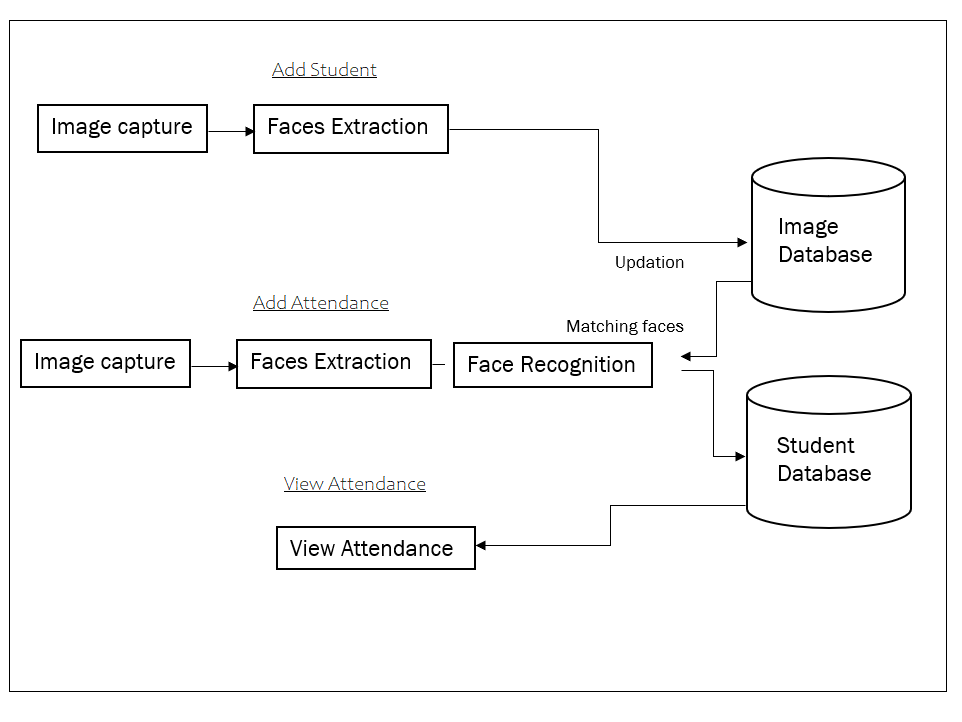


Fig 3.1 Architectural Design

The Architectural Design diagram above shows the whole architecture of the project, it illustrates the three main functions of the project :

* Add Student
* Add attendance
* View attendance

**3.2 Modules**

**1.Faces Extraction Module:**

In the faces Extraction module we will be doing two functions , one is to capture the image and another is to extract faces of that image and returning the path of the folder.

**2.dbms Module:**

This module has all the functions related to database operations to be done like , updating attendance or updating student details.

**3.UI Module:**

In this module we have all the UI related code.

* 1. **Main Module:**

In this module we integrate all the previous three modules.

**3.3 UML Diagrams**

The Unified Modeling Language (UML) is a general-purpose, developmental, modeling language in the field of  software engineering that is intended to provide a standard way to visualize the design of a system.It is analogous to the blueprints used in other fields, and consists of different types of diagrams.

UML Diagrams for our application are as follows:

**3.3.1 Use Case Diagrams**

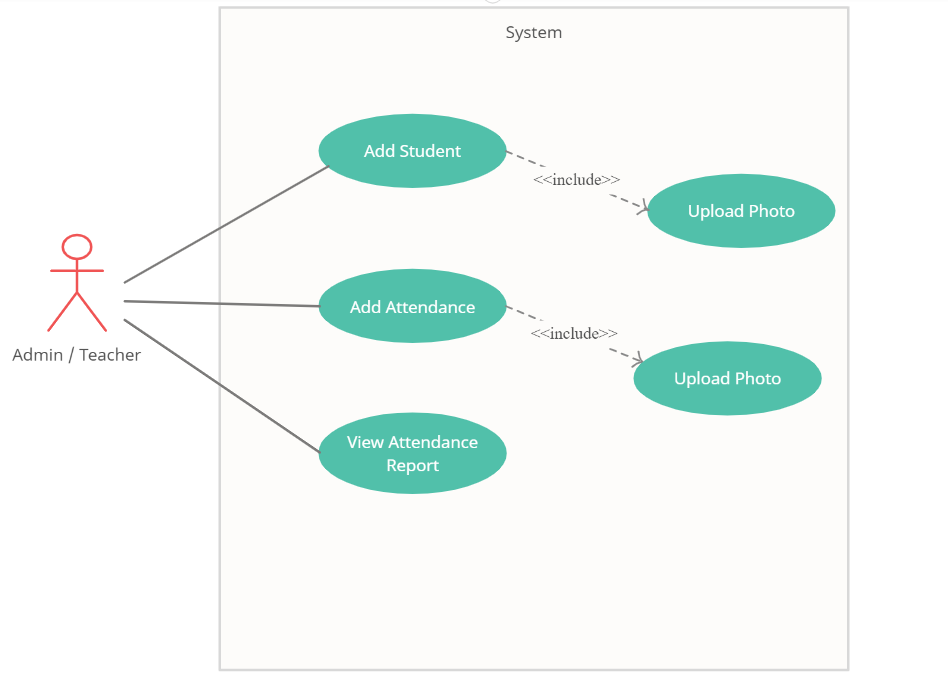


Fig 3.2 Usecase Diagram

This Use Case Diagram depicts the functionality of Teacher/Admin. Adminis has three functionalities namely Add student , Add Attendance and View attendance Reports .Here actor is Administrator and Add student , Add Attendance and View attendance Reports are his use cases .

**3.3.2 Class Diagrams**

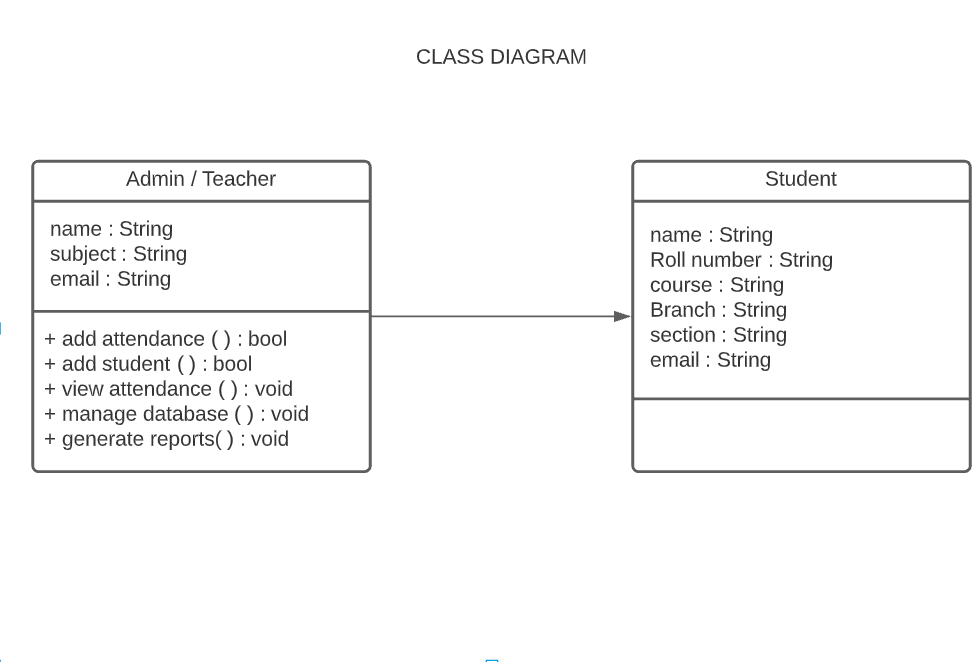


Fig 3.3 Class Diagram

In attendance management system we have identified mainly two classes they are , Teacher/Admin class and Student class . Their respective attributes and methods are shown in the above diagram.

**3.3.3 Sequence Diagrams**

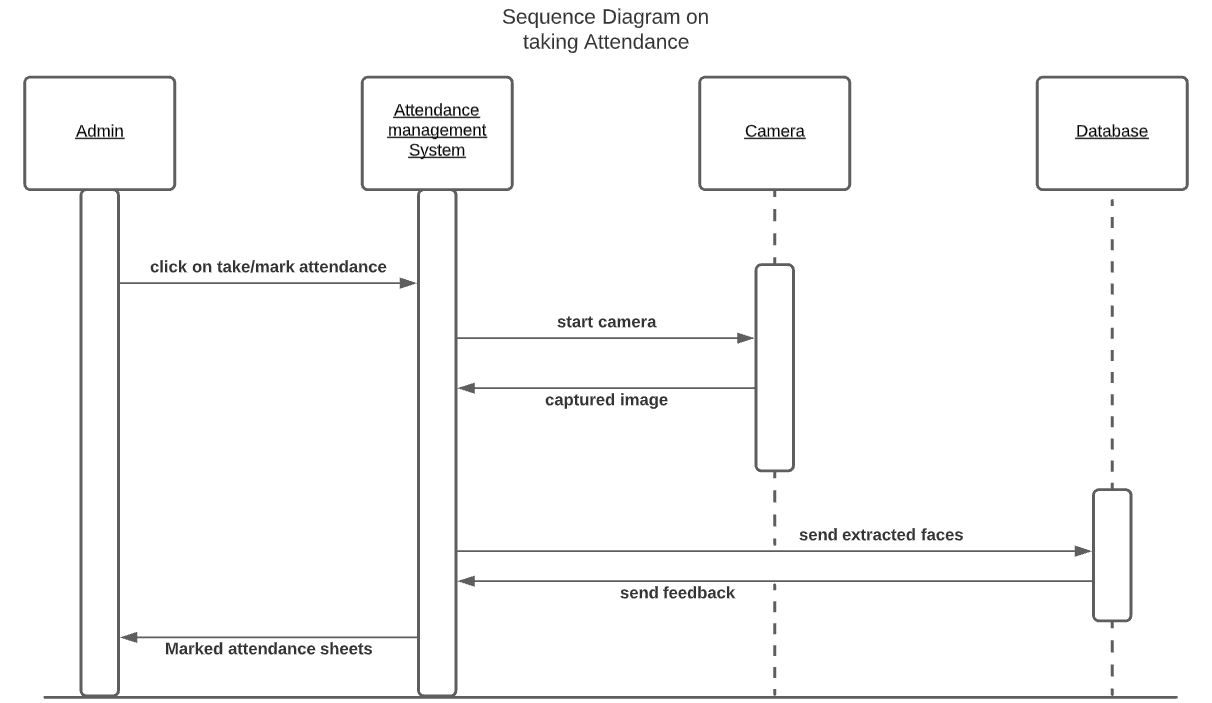


Fig 3.4 Sequence Diagram for taking attendance

Sequence Diagram for taking attendance is shown above,where first the admin have to click on Take Attendance button then the System starts camera and captures image then the captured image is saved and then we extarct faces in the system and we upload the extracted faces to our database , then the system marks the attendance sheets. , then the marked sheets were sent to the admin.

**3.3.4 Activity Diagrams**

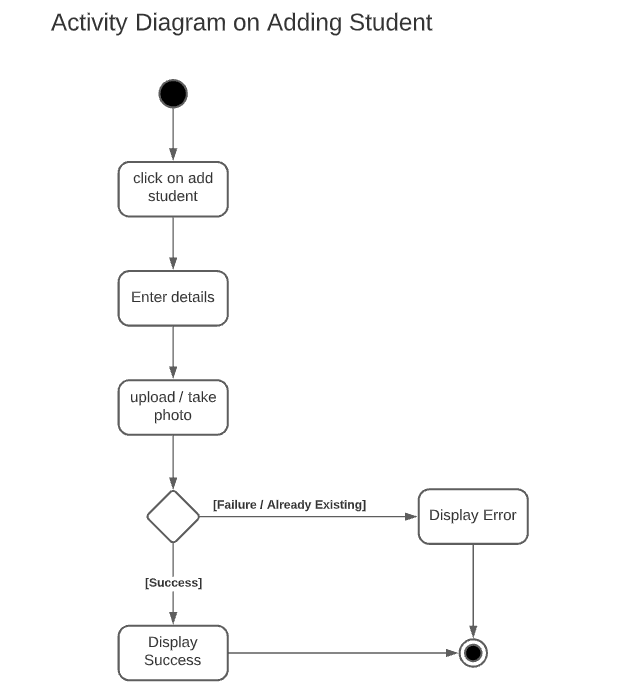


Fig 3.5 Activity Diagram for Adding Student

The above activity diagram for adding student shows the flow chart to add a student , first the admin clicks on Add Student button then the admin uploads the details and image of the student it displays the result.

**3.3.5 Package Diagrams**

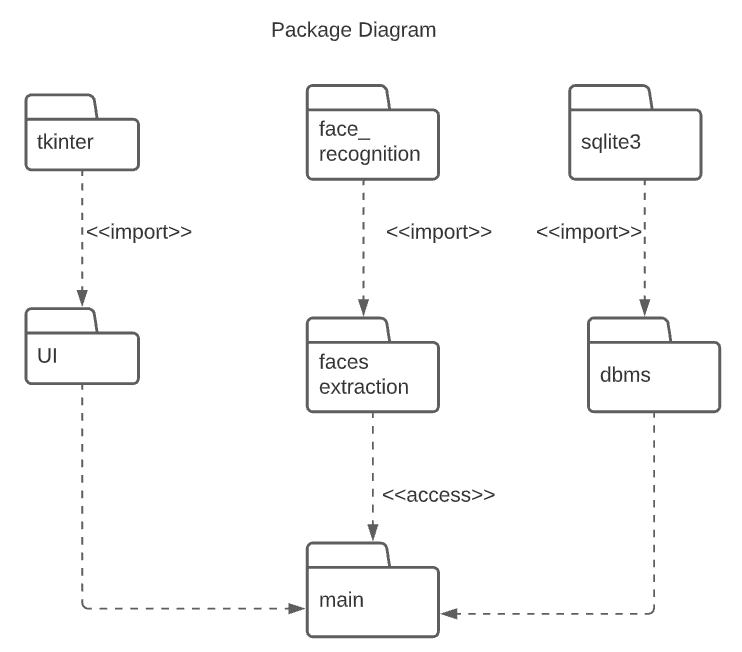


Fig 3.5 Package Diagram

The above diagram represents the package diagram of the system.

**4.SYSTEM IMPLEMENTATION**

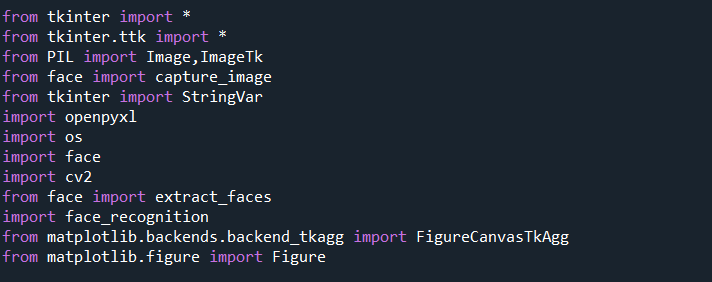
The implementation stage of any project is a true display of the defining moments that make a project a success or a failure. The implementation stage is defined as the system or system modifications being installed and made operational in a production environment. The phase is initiated after the system has been tested and accepted by the user. This phase continues until the system is operating in production in accordance with the defined user requirements.

**Implementation of code in Python**

**4.1 Importing Required libraries**

Required libraries are ..

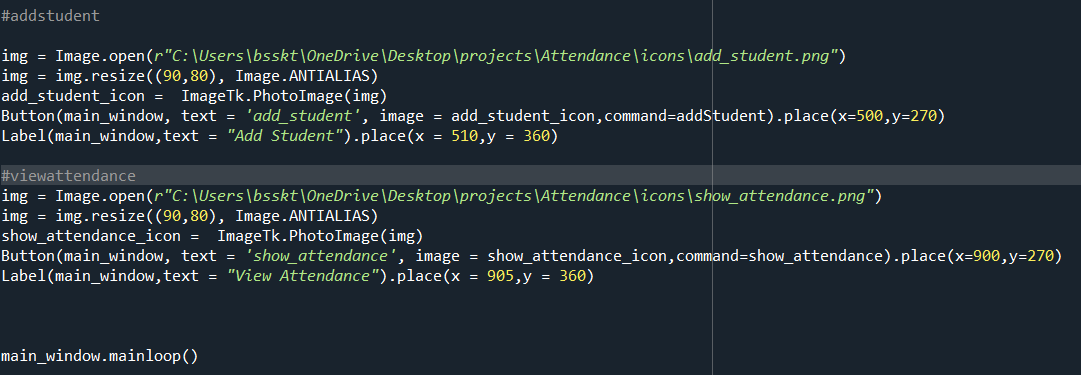
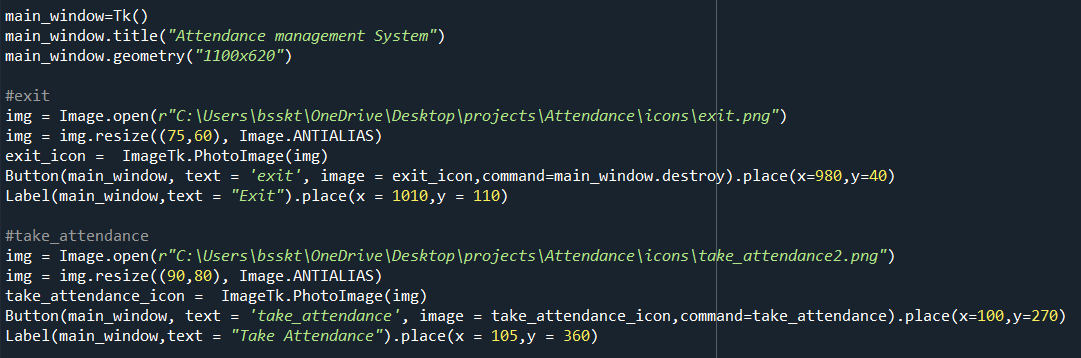
* face\_recognition
* tkinter
* openpyxl
* PIL
* Matplotlib etc..



**4.2 Main Window UI**

In the main window we have 4 buttons for functionality:

* Take Attendance – to take attendance
* Add student – to add student
* View Attendance – to view attendance of a student
* Exit – to close the window

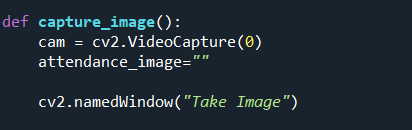


**4.3 Take Attendance**

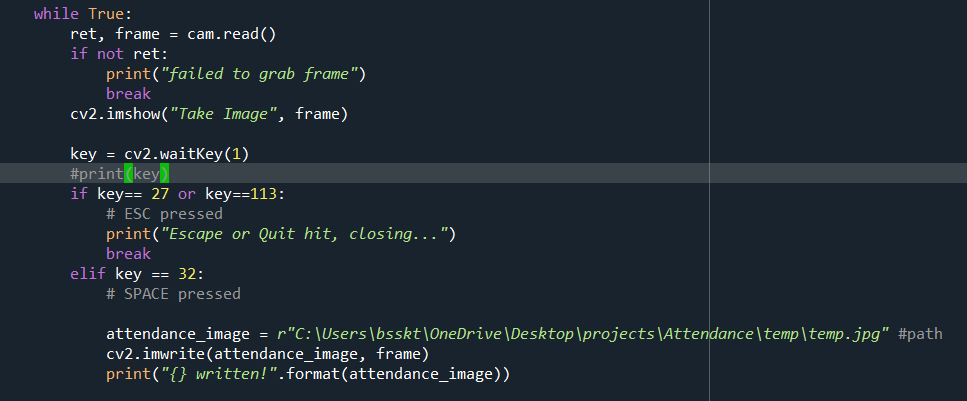
* After clicking the “Take Attendance” Button ,a window named “Take Image” pops out asking to capture a image (By pressing “space” we can capture image and by pressing “q” or “esc” we can close window).
* Here we implement two functions (capture\_image() and extract\_faces() )

**4.3.1 Capture Image**

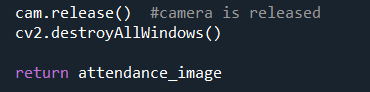
* We are capturing an image by using the cv2 module which takes access to the camera.
* OpenCV is a vast library that helps in providing various functions for image and video operations.
* With OpenCV, we can capture a video from the camera.
* It lets you create a video capture object which is helpful to capture videos/capture images through webcam and then we can perform desired operations on that video/image.



* cv2.VideoCapture(0) - is used to get a video capture object for the camera.
* Cv2.namedWindow() – is used to name the window.



* read() -  method to read the frames using the above created object.
* imshow() - method to show the frames in the video.
* If space is pressed then the image(frame) is saved in the given path
* If esc or q is pressed then we can come out of the infinite loop and the camera window closes



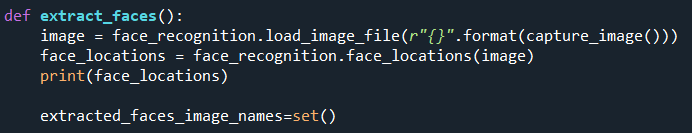
* Camera object is released and all the windows belong to cv2 were destroyed and

path of the image is returned.

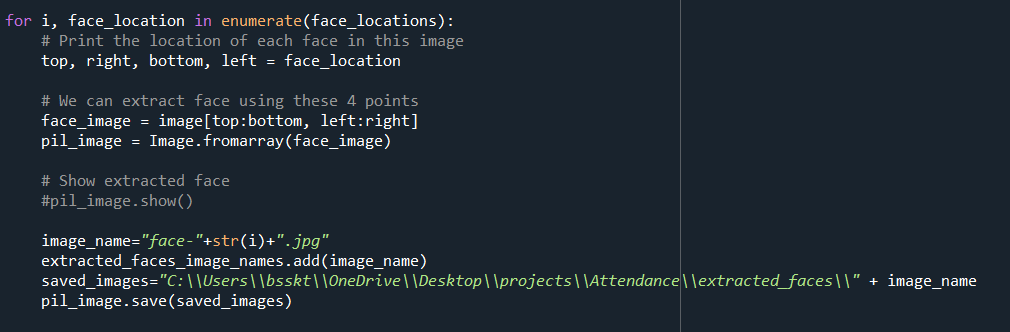
* After capturing a image extraction of faces comes next.

**4.3.2 Extract faces**

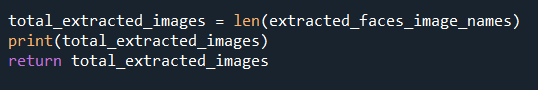
* Here we use face\_recognition module to extract faces.
* face\_recognition is a module in python used to locate or recognize faces.
* Here we implement extract\_faces function to extract the faces and store into a folder named extracted\_faces.



* load\_image\_file - is used to load the image by giving path to the image (we have returned the captured image path in capture\_image() function)
* face\_locations– is used to get all the face locations , it returns a list of tuples of face locations.

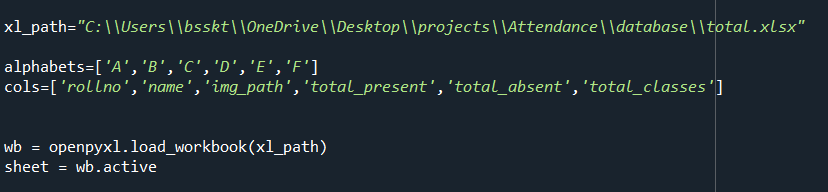


* The “Image” module provides a class with the same name which is used to represent a PIL image. The module also provides a number of factory functions, including functions to load images from files, and to create new images.
* Image.forarray() - Creates an image memory from an object exporting the array interface (using the buffer protocol).
* Then we are saving the extracted faces into extracted\_faces folder



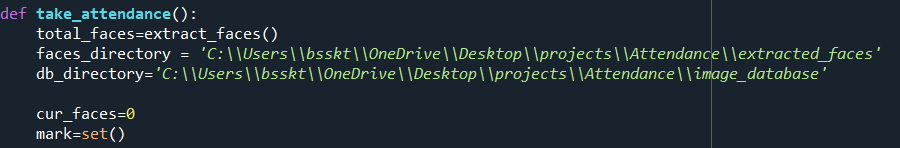
* We are returning no.of faces extracted in the image.

**4.3.2 Student Database**

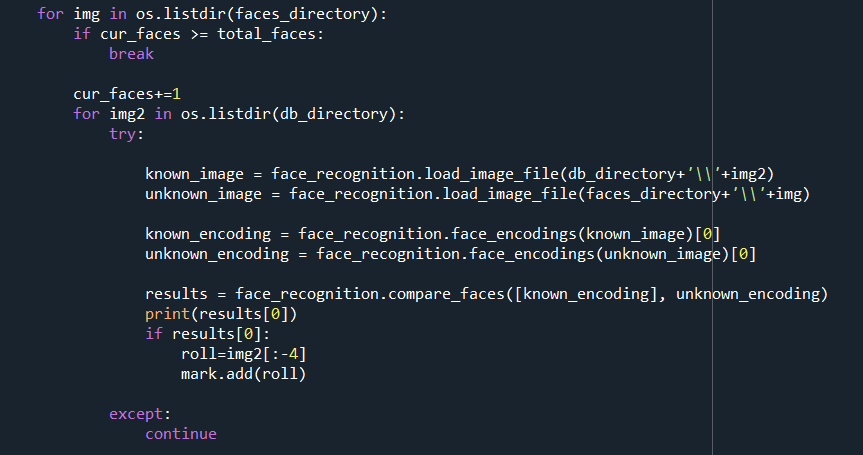


* openpyxl is a Python library for reading and writing Excel files.
* The openpyxl module allows Python program to read and modify Excel files.
* load\_workbook () – method loads the excel file.
* active – attribute grabs the active worksheet.

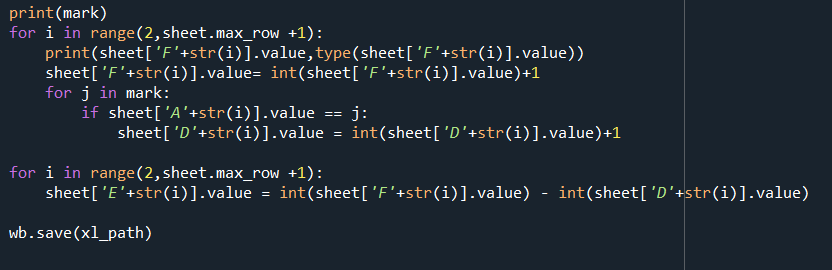
**4.3.3 Take Attendance – Method**



* In total\_faces variable we have the total faces that we extracted (extract\_faces() function call will be done here) using extact\_faces method.
* In faces\_directory we have extracted faces and in db\_directory we have the student image\_database.



* Here we are iterating through two directories (image\_database and extracted\_faces) and we are matching the faces in extracted\_faces and image\_database.
* We are comparing two faces using compare\_faces() function.If the two faces match then we insert the roll no into mark set.
* Before comparing two faces we do encoding of faces using face\_encoding() function.
* We maintain current faces (cur\_faces) variable to count the faces extracted in this particular instance.

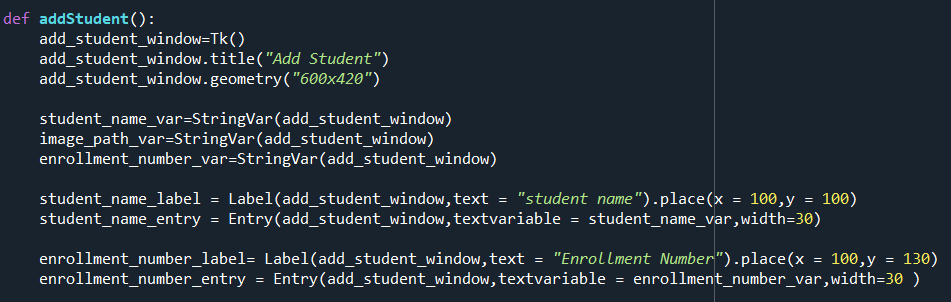


* We are updating the results in mark set into the excel sheet that we are maintaining (here column ‘F’ - for total classes , ‘E’ – for total absent, ‘D’ – for total present ).
* save() method is used to save the worksheet.

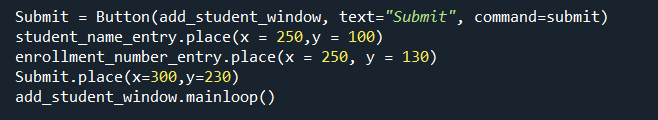
**4.4 Add Student**

**4.4.1 Add Student UI**

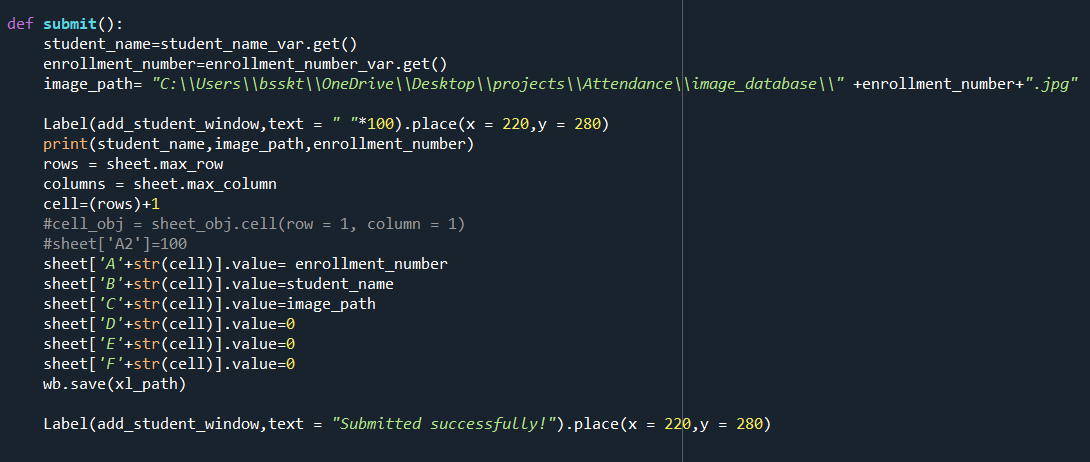
* In add student UI we have two labels and two fields to fill (name and enrollment number).
* Upload button is used to upload the image of the student.
* Submit button is used to submit the entered details.
* Upload button follows the same functionality of extract\_faces() function but here the extracted face is saved in image\_database folder.
* The Tkinter StringVar helps you manage the value of a widget such as a [Label](https://www.pythontutorial.net/tkinter/tkinter-label/) or [Entry](https://www.pythontutorial.net/tkinter/tkinter-entry/) more effectively.



* The Tkinter StringVar helps us to manage the value of a widget such as a Label or Entry more effectively.



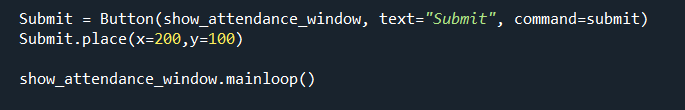
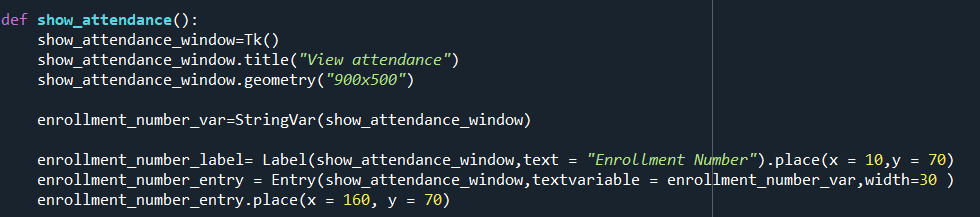
**4.4.2 Add Student – submit button**



* After clicking submit name, roll no and image path are added to the excel sheet.
* In the submit function ,if the details are submitted (added in the excel sheet) then it displays the message as submitted successfully.

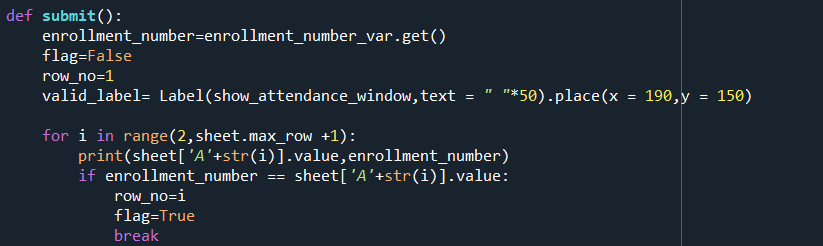
**4.5 View Attendance**

**4.5.1 View Attendance UI**

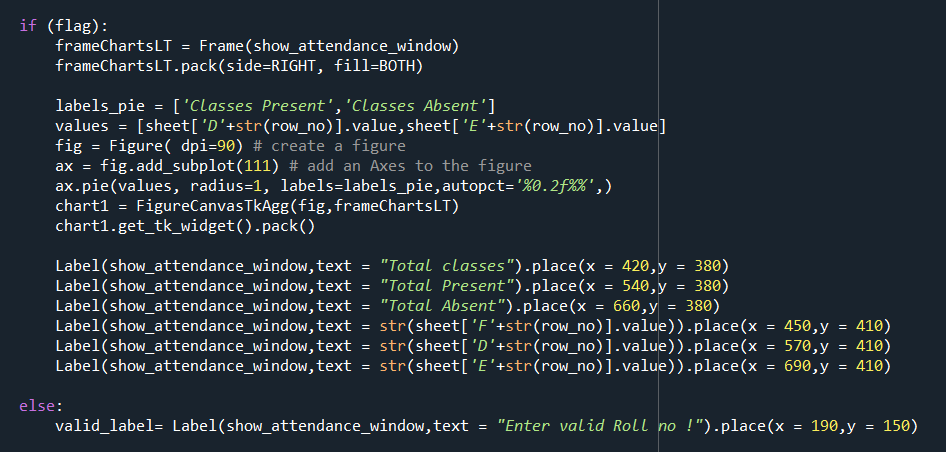


* View attendance Ui has one Label , one Entry and one Button.
* Enter belongs to the enrollment Number.
* After entering the enrollment number in the entry, after hitting submit it checks whether the enrollment number is present in the database or not.
* If it is in the database it gives us the output
* If not it displays a message to enter valid enrollment number.

**4.5.2 View Attendance – submit button**



* In the submit button we are verifying whether the entered roll no is present in the database or not
* If it is present we are setting flag as True.



* If the given roll no is present we are providing outputs else a message is displayed.
* Here we are placing a matplotlib chart into the tkinter window.
* Figure() – is used to create a object that contains the plot.
* add\_subplot() – is used to add axes to the figure.
* Pie() – to plot a pie chart.
* FigCanvasTkAgg() – it contains the matplotlib plot.

**5. OUTPUT SCREENS**

Output Screens of various functionalities in our application are shown over here along with the description.

**5.1 Main UI**

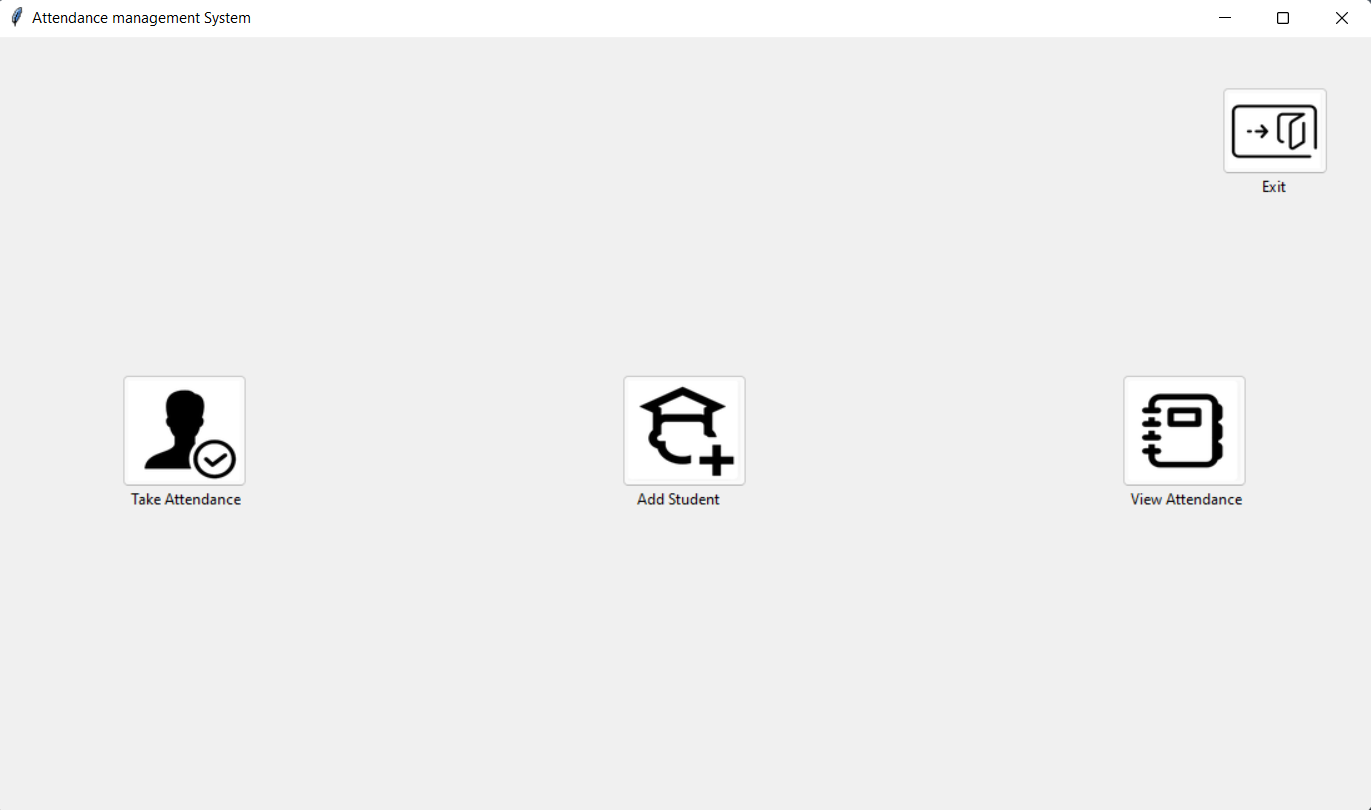


Fig 5.1 Main window UI

In the main window we have 4 buttons for functionality:

* Take Attendance – to take attendance
* Add student – to add student
* View Attendance – to view attendance of a student
* Exit – to close the window

**5.2 Take Attendance – UI**

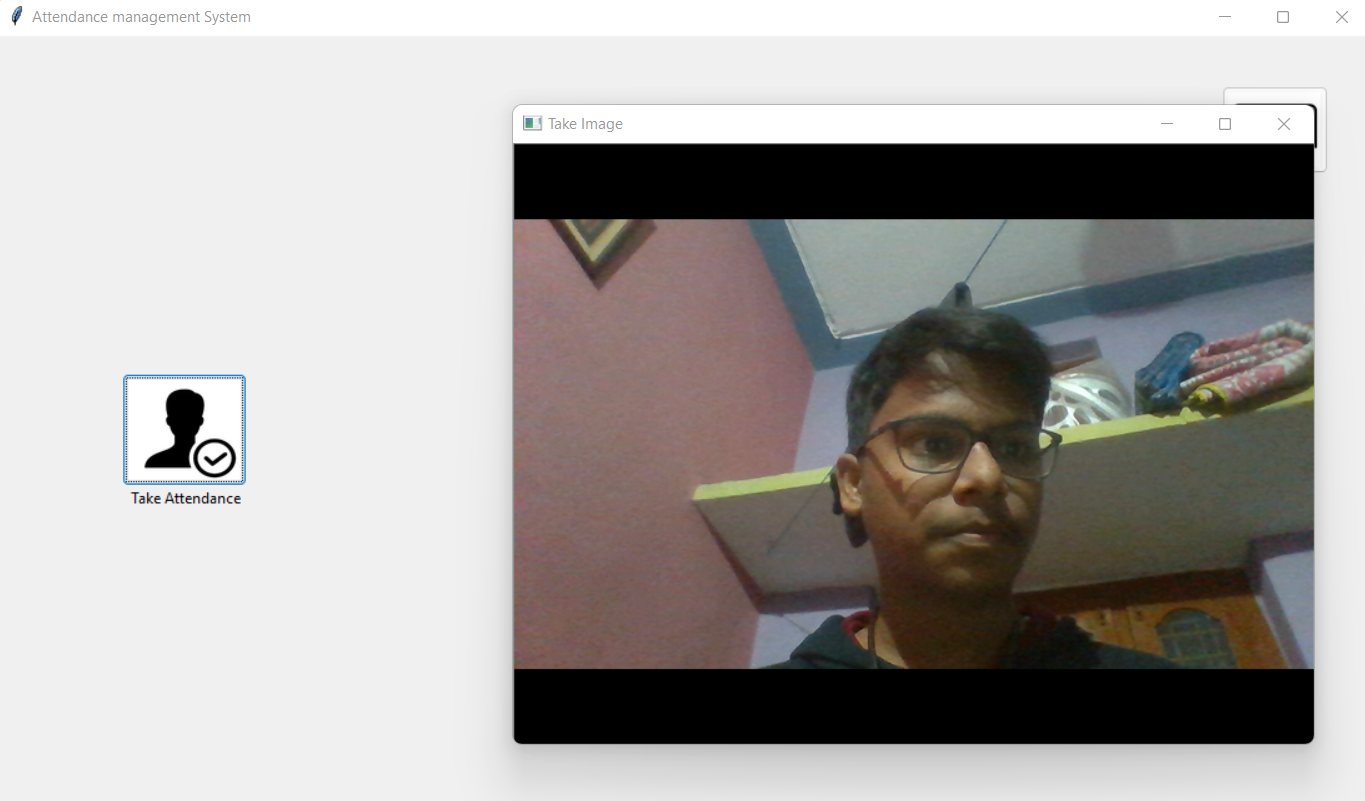


Fig 5.2 Take attendance UI

* After clicking the “Take Attendance” Button ,a window named “Take Image” pops out asking to capture a image (By pressing “space” we can capture image and by pressing “q” or “esc” we can close window).
* Here we implement two functions capturing image and extracting faces.

**5.3 Add Student – UI**

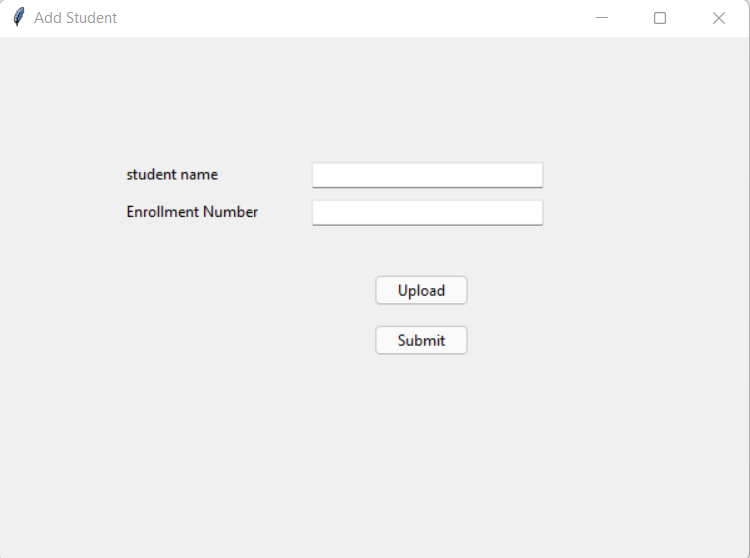


Fig 5.3 Add Student UI

* In add student UI we have two labels and two fields to fill (name and enrollment number).
* Upload button is used to upload the image of the student.
* Submit button is used to submit the entered details.
* Upload button follows the same functionality of extracting faces ,but here the extracted face is saved in image\_database folder.
* The Tkinter StringVar helps you manage the value of a widget such as a [Label](https://www.pythontutorial.net/tkinter/tkinter-label/) or [Entry](https://www.pythontutorial.net/tkinter/tkinter-entry/) more effectively.
* In the upload function if the image is uploaded , if it finds any faces it shows message as upload successful or if it doesn’t recognize any faces it shows a message to upload again.

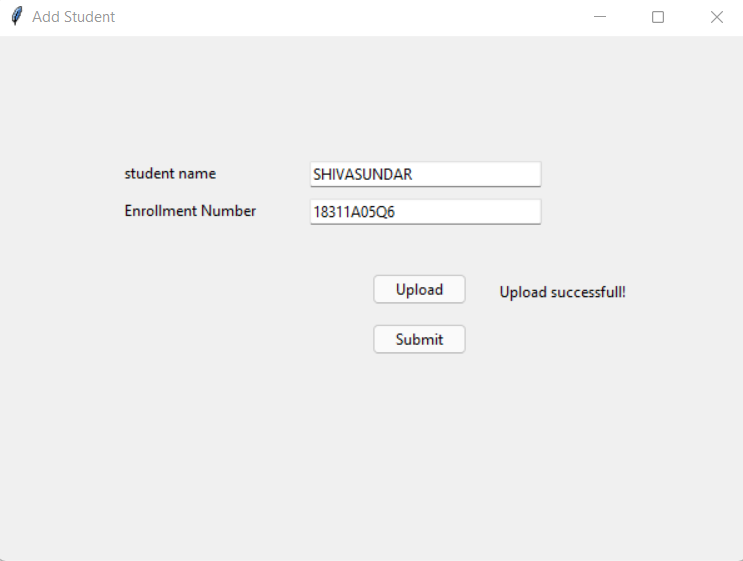


Fig 5.4.1 Add Student – Upload Successful

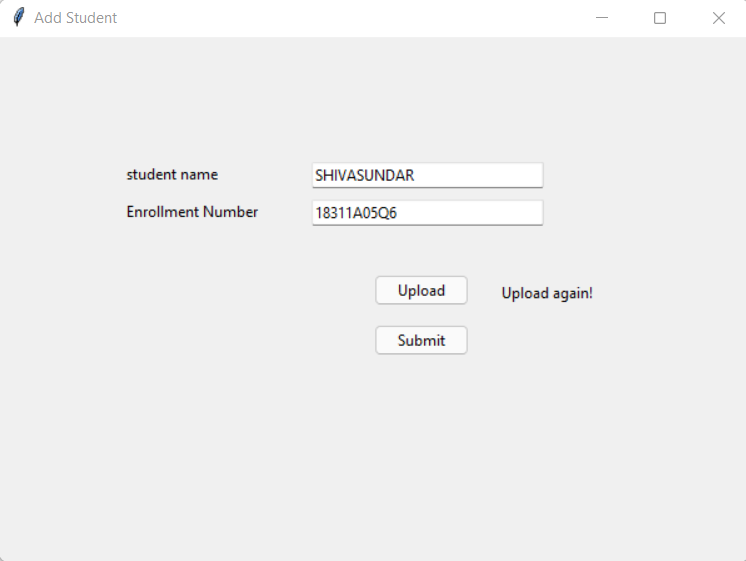


Fig 5.4.2 Add Student – Upload Unsuccessful

* In the submit function ,if the details are submitted (added in the excel sheet) then it displays the message as submitted successfully.

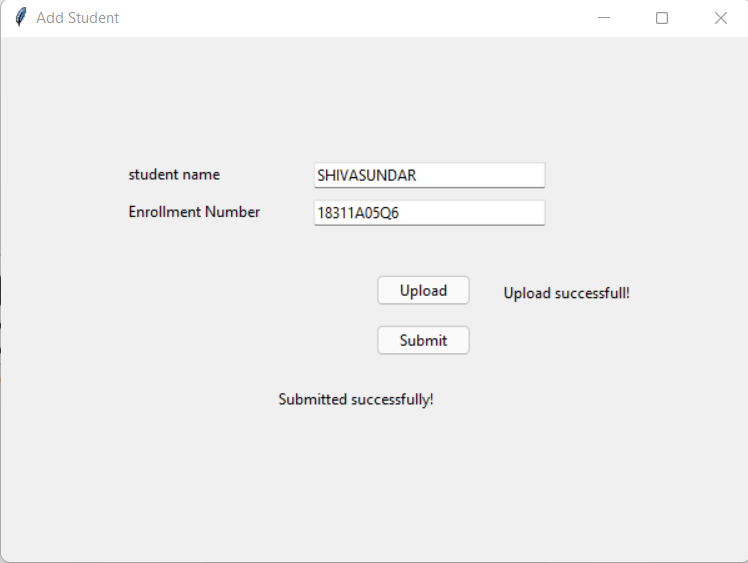


Fig 5.5 Add Student – Submit Successful

* This is how the excel sheet (student database) looks like after a few operations.

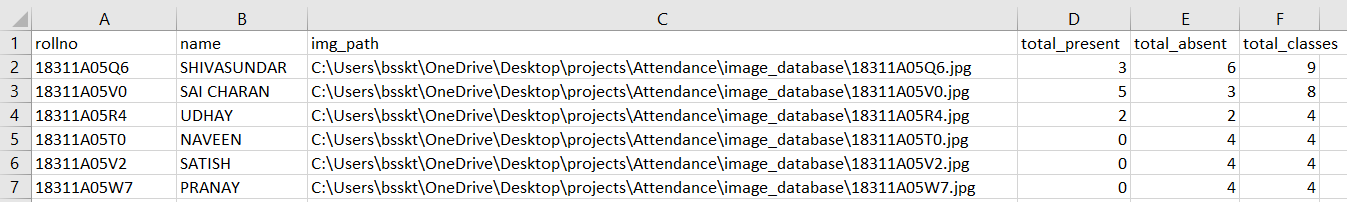


Fig 5.6 Database

**5.4 View Attendance– UI**

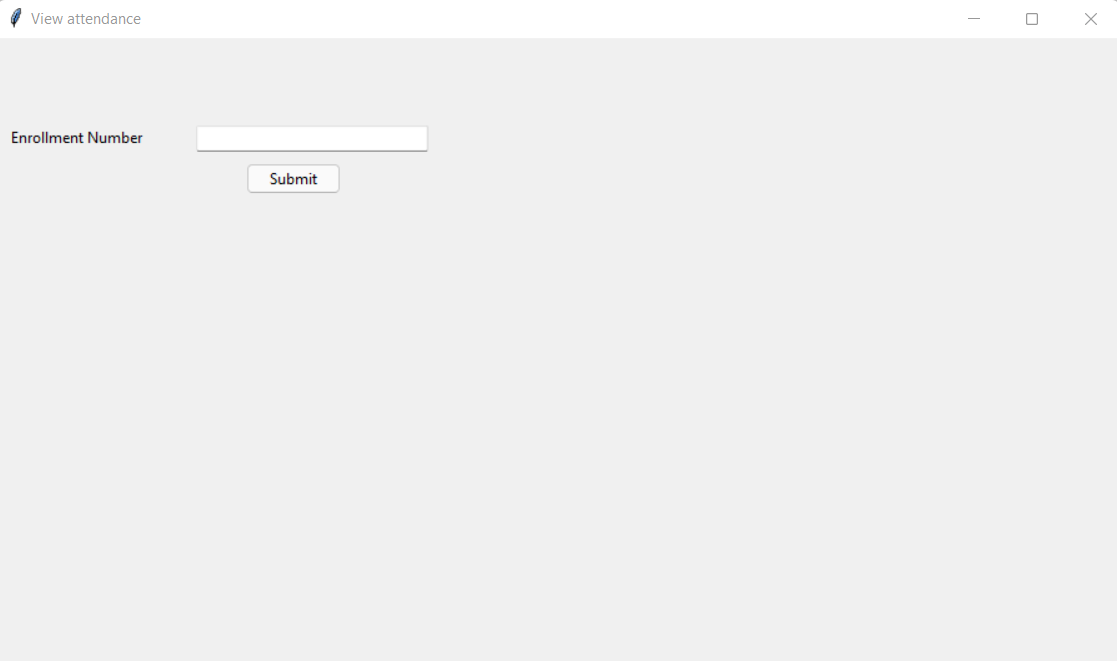


Fig 5.7 View Attendance UI

* After entering the enrollment number in the entry, after hitting submit it checks whether the enrollment number is present in the database or not.
* If it is in the database it gives us the output
* If not it displays a message to enter valid enrollment number.

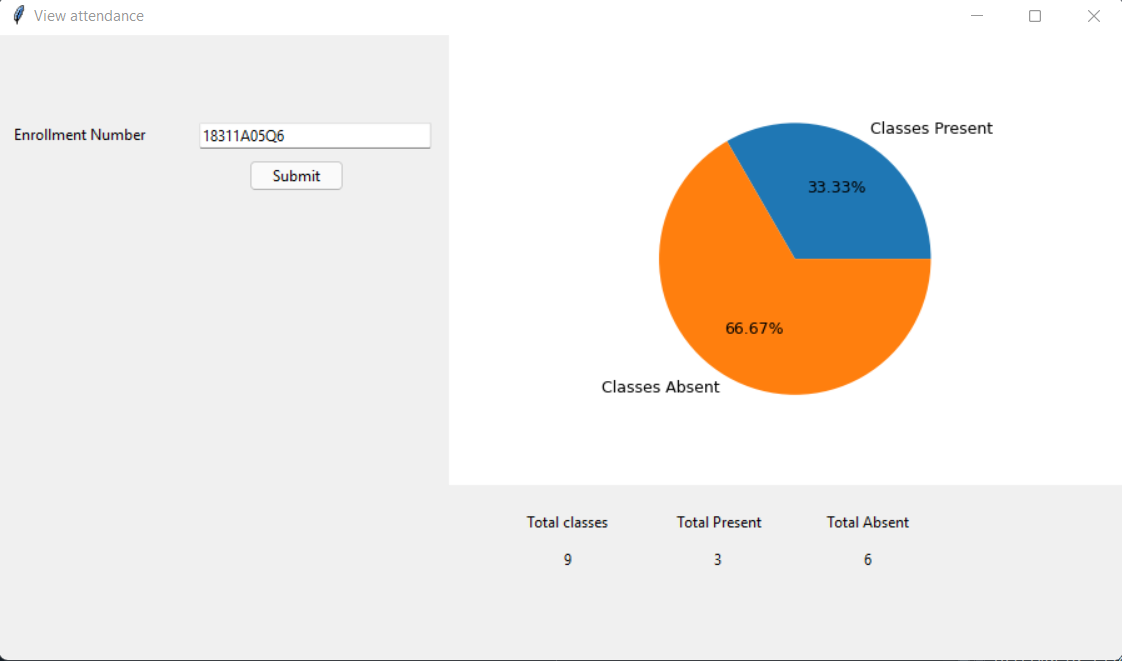


Fig 5.8.1 View Attendance – Details Successfully fetched

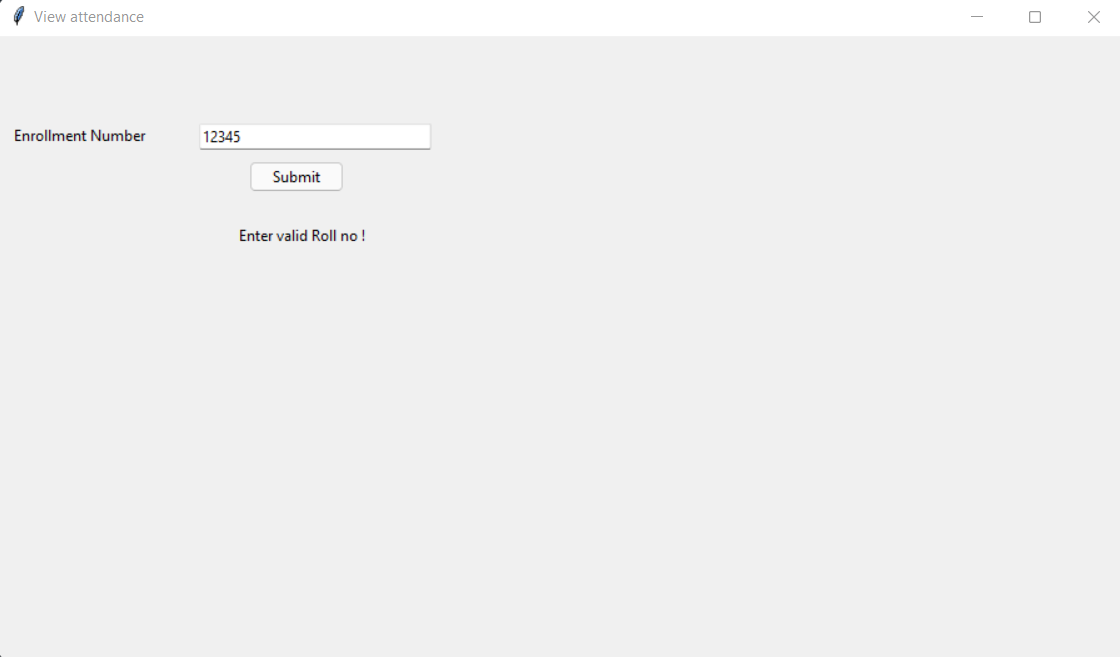


Fig 5.8.2 View Attendance – Details Not Found

**6.CONCLUSION AND FUTURE SCOPE**

**CONCLUSION**

* We have successfully implemented all the features of attendance management system using tkinter GUI.
* The overall accuracy depends of the project depends on the face\_recognition module accuracy, camera quality and lighting while taking a image

**FUTURE SCOPE**

* This project can be used in schools ,colleges and in organizations ,where ever the attendance is calculated or considered.
* Future scope of this project is , we can implement an web interface so that any mobile or laptop device can access , a login system and a real database updation using backend languages etc..

**7.BIBILIOGRAPHY**

* Student Attendance System using Face Recognition by Samridhi Dev, Tushar Patnaik . ( <https://ieeexplore.ieee.org/document/9215441> )
* Upgrad - <https://www.upgrad.com/blog/image-processing-projects-ideas-topics/>
* [Creatly - https://creately.com/blog/diagrams/uml-diagram-types-examples/](https://creately.com/blog/diagrams/uml-diagram-types-examples/)
* [String Var - https://www.pythontutorial.net/tkinter/tkinter-stringvar/](https://www.pythontutorial.net/tkinter/tkinter-stringvar/)
* PIL - <https://www.geeksforgeeks.org/python-pillow-a-fork-of-pil/>

<https://pillow.readthedocs.io/en/stable/reference/Image.html>

* Openpyxl - [https://www.geeksforgeeks.org/python-reading-excel-file-using- openpyxl-module/](https://www.geeksforgeeks.org/python-reading-excel-file-using-%20%20openpyxl-module/)
* Face recognition - <https://face-recognition.readthedocs.io/en/latest/readme.html>
* Image button - <https://www.geeksforgeeks.org/python-add-image-on-a-tkinter-button/>
* Opencv - <https://www.geeksforgeeks.org/python-opencv-capture-video-from-camera/>

**APPENDIX-A: PYTHON AND OTHER LIBRARIES**

**About Python**:

**Python** is an [interpreted](https://en.wikipedia.org/wiki/Interpreted_language) [high-level](https://en.wikipedia.org/wiki/High-level_programming_language) [general-purpose programming language](https://en.wikipedia.org/wiki/General-purpose_programming_language). Its design philosophy emphasizes [code readability](https://en.wikipedia.org/wiki/Code_readability) with its use of [significant indentation](https://en.wikipedia.org/wiki/Off-side_rule). Its [language constructs](https://en.wikipedia.org/wiki/Language_construct) as well as its [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming) approach aim to help [programmers](https://en.wikipedia.org/wiki/Programmers) write clear, logical code for small and large-scale projects**.**

Python is [dynamically-typed](https://en.wikipedia.org/wiki/Type_system#DYNAMIC) and [garbage-collected](https://en.wikipedia.org/wiki/Garbage_collection_(computer_science)). It supports multiple [programming paradigms](https://en.wikipedia.org/wiki/Programming_paradigm), including [structured](https://en.wikipedia.org/wiki/Structured_programming) (particularly, [procedural](https://en.wikipedia.org/wiki/Procedural_programming)), object-oriented and [functional programming](https://en.wikipedia.org/wiki/Functional_programming). It is often described as a "batteries included" language due to its comprehensive [standard library](https://en.wikipedia.org/wiki/Standard_library).

[Guido van Rossum](https://en.wikipedia.org/wiki/Guido_van_Rossum) began working on Python in the late 1980s, as a successor to the [ABC programming language](https://en.wikipedia.org/wiki/ABC_(programming_language)), and first released it in 1991 as Python 0.9.0.[[33]](https://en.wikipedia.org/wiki/Python_(programming_language)#cite_note-33) Python 2.0 was released in 2000 and introduced new features, such as [list comprehensions](https://en.wikipedia.org/wiki/List_comprehension) and a [cycle-detecting](https://en.wikipedia.org/wiki/Cycle_detection) garbage collection system (in addition to [reference counting](https://en.wikipedia.org/wiki/Reference_counting)). Python 3.0 was released in 2008 and was a major revision of the language that is not completely [backward-compatible](https://en.wikipedia.org/wiki/Backward_compatibility). Python 2 was discontinued with version 2.7.18 in 2020.

Python consistently ranks as one of the most popular programming languages.

It is used for:

* web development (server-side),
* software development,
* mathematics,
* system scripting.

**TKINTER**

Python offers multiple options for developing GUI (Graphical User Interface). Out of all the GUI methods, tkinter is the most commonly used method. It is a standard Python interface to the Tk GUI toolkit shipped with Python. Python with tkinter is the fastest and easiest way to create the GUI applications.

Tkinter is the standard GUI library for Python. Python when combined with Tkinter provides a fast and easy way to create GUI applications. Tkinter provides a powerful object-oriented interface to the Tk GUI toolkit.

Creating a GUI using tkinter is an easy task.

To create a tkinter app:

1.Importing the module – tkinter

2.Create the main window (container)

3.Add any number of widgets to the main window

4.Apply the event Trigger on the widgets.

**Tkinter Widgets**

Tkinter provides various controls, such as buttons, labels and text boxes used in a GUI application. These controls are commonly called widgets.

There are currently 15 types of widgets in Tkinter. We present some of widgets used in the project as well as a brief description in the following table –

|  |  |
| --- | --- |
| **SL.No.** | **Operator & Description** |
| 1 | [Button](https://www.tutorialspoint.com/python/tk_button.htm)  The Button widget is used to display buttons in your application. |
| 2 | [Canvas](https://www.tutorialspoint.com/python/tk_canvas.htm)  The Canvas widget is used to draw shapes, such as lines, ovals, polygons and rectangles, in your application. |
| 3 | [Checkbutton](https://www.tutorialspoint.com/python/tk_checkbutton.htm)  The Checkbutton widget is used to display a number of options as checkboxes. The user can select multiple options at a time. |
| 4 | [Entry](https://www.tutorialspoint.com/python/tk_entry.htm)  The Entry widget is used to display a single-line text field for accepting values from a user. |
| 5 | [Frame](https://www.tutorialspoint.com/python/tk_frame.htm)  The Frame widget is used as a container widget to organize other widgets. |
| 6 | [Label](https://www.tutorialspoint.com/python/tk_label.htm)  The Label widget is used to provide a single-line caption for other widgets. It can also contain images. |
| 7 | [Listbox](https://www.tutorialspoint.com/python/tk_listbox.htm)  The Listbox widget is used to provide a list of options to a user. |
| 8 | [Menubutton](https://www.tutorialspoint.com/python/tk_menubutton.htm)  The Menubutton widget is used to display menus in your application. |
| 9 | [Menu](https://www.tutorialspoint.com/python/tk_menu.htm)  The Menu widget is used to provide various commands to a user. These commands are contained inside Menubutton. |
| 10 | [Message](https://www.tutorialspoint.com/python/tk_message.htm)  The Message widget is used to display multiline text fields for accepting values from a user. |
| 11 | [Radiobutton](https://www.tutorialspoint.com/python/tk_radiobutton.htm)  The Radiobutton widget is used to display a number of options as radio buttons. The user can select only one option at a time. |
| 12 | [Scale](https://www.tutorialspoint.com/python/tk_scale.htm)  The Scale widget is used to provide a slider widget. |
| 13 | [Scrollbar](https://www.tutorialspoint.com/python/tk_scrollbar.htm)  The Scrollbar widget is used to add scrolling capability to various widgets, such as list boxes. |
| 14 | [Text](https://www.tutorialspoint.com/python/tk_text.htm)  The Text widget is used to display text in multiple lines. |
| 15 | [Toplevel](https://www.tutorialspoint.com/python/tk_toplevel.htm)  The Toplevel widget is used to provide a separate window container. |
| 16 | [Spinbox](https://www.tutorialspoint.com/python/tk_spinbox.htm)  The Spinbox widget is a variant of the standard Tkinter Entry widget, which can be used to select from a fixed number of values. |
| 17 | [PanedWindow](https://www.tutorialspoint.com/python/tk_panedwindow.htm)  A PanedWindow is a container widget that may contain any number of panes, arranged horizontally or vertically. |
| 18 | [LabelFrame](https://www.tutorialspoint.com/python/tk_labelframe.htm)  A labelframe is a simple container widget. Its primary purpose is to act as a spacer or container for complex window layouts. |
| 19 | [tkMessageBox](https://www.tutorialspoint.com/python/tk_messagebox.htm)  This module is used to display message boxes in your applications. |

Table A-1 tkinter widgets

**Openpyxl**

Python provides the Openpyxl module, which is used to deal with Excel files without involving third-party Microsoft application software.Openpyxl is a Python library for reading and writing Excel (with extension xlsx/xlsm/xltx/xltm) files. The openpyxl module allows Python program to read and modify Excel files.

For example, users might have to go through thousands of rows and pick out a few handful of information to make small changes based on some criteria. Using Openpyxl module, these tasks can be done very efficiently and easily.It is used to perform excel tasks such as read data from excel file, or write data to the excel file, draw some charts, accessing excel sheet, renaming sheet, modification (adding and deleting) in excel sheet, formatting, styling in the sheet, and any other task. Openpyxl is very efficient to perform these tasks for you.

Data scientists often use the Openpyxl to perform different operations such as data copying to data mining as well as data analysis.

The Openpyxl library is used to write or read the data in the excel file and many other tasks. An excel file that we use for operation is called Workbook that contains a minimum of one Sheet and a maximum of tens of sheets.

* Sheets consist of Rows (horizontal series) starting from 1 and Columns (vertical series) starting from A.
* Row and column together make a grid and form a cell that may store some data. Data can be of any type, such as numeric, string.
* Openpyxl provides flexibility to read data from the individual cell or write data to it.

**Opencv**

OpenCV is the huge open-source library for the computer vision, machine learning, and image processing and now it plays a major role in real-time operation which is very important in today’s systems. By using it, one can process images and videos to identify objects, faces, or even handwriting of a human. When it integrated with various libraries, such as NumPy, python is capable of processing the OpenCV array structure for analysis. To Identify image pattern and its various features we use vector space and perform mathematical operations on these features.

The first OpenCV version was 1.0. OpenCV is released under a BSD license and hence it’s free for both academic and commercial use. It has C++, C, Python and Java interfaces and supports Windows, Linux, Mac OS, iOS and Android. When OpenCV was designed the main focus was real-time applications for computational efficiency. All things are written in optimized C/C++ to take advantage of multi-core processing.

Applications of OpenCV:

There are lots of applications which are solved using OpenCV, some of them are listed below :

* face recognition
* Automated inspection and surveillance
* number of people – count (foot traffic in a mall, etc)
* Vehicle counting on highways along with their speeds
* Interactive art installations
* Anamoly (defect) detection in the manufacturing process (the odd defective products)
* Street view image stitching
* Video/image search and retrieval
* Robot and driver-less car navigation and control
* object recognition
* Medical image analysis
* Movies – 3D structure from motion
* TV Channels advertisement recognition

OpenCV Functionality **:**

* Image/video I/O, processing, display (core, imgproc, highgui)
* Object/feature detection (objdetect, features2d, nonfree)
* Geometry-based monocular or stereo computer vision (calib3d, stitching, videostab)
* Computational photography (photo, video, superres)
* Machine learning & clustering (ml, flann)
* CUDA acceleration (gpu)

**PIL**

Python Imaging Library (expansion of PIL) is the de facto image processing package for Python language. It incorporates lightweight image processing tools that aids in editing, creating and saving images. Support for Python Imaging Library got discontinued in 2011, but a project named pillow forked the original PIL project and added Python3.x support to it. Pillow was announced as a replacement for PIL for future usage. Pillow supports a large number of image file formats including BMP, PNG, JPEG, and TIFF. The library encourages adding support for newer formats in the library by creating new file decoders.

The Python Imaging Library adds image processing capabilities to your Python interpreter.This library provides extensive file format support, an efficient internal representation, and fairly powerful image processing capabilities.

The core image library is designed for fast access to data stored in a few basic pixel formats. It should provide a solid foundation for a general image processing tool.

**Matplotlib**

Matplotlib is an amazing visualization library in Python for 2D plots of arrays. Matplotlib is a multi-platform data visualization library built on NumPy arrays and designed to work with the broader SciPy stack. It was introduced by John Hunter in the year 2002.

One of the greatest benefits of visualization is that it allows us visual access to huge amounts of data in easily digestible visuals. Matplotlib consists of several plots like line, bar, scatter, histogram etc.

**face\_recognition**

face\_recognition is one the simplest library to recognize and manipulate faces in python.

It is Built using [dlib](http://dlib.net/)’s state-of-the-art face recognition built with deep learning. The model has an accuracy of 99.38% on the [Labeled Faces in the Wild](http://vis-www.cs.umass.edu/lfw/) benchmark.

This also provides a simple  “face\_recognition” command line tool that lets you do face recognition on a folder of images from the command line!

Its features are to :

* Find faces in pictures

### manipulate facial features in pictures

### Identify faces in pictures

**APPENDIX-B:UNIFIED MODELING LANGUAGE**

The Unified Modeling Language (UML) is a general-purpose visual modeling language that is used to specify, visualize, construct, and document the artifacts of a software system. It captures decisions and understanding about systems that must be constructed. It is used to understand, design, browse, configure, maintain, and control information about such systems. It is intended for use with all development methods, lifecycle stages, application domains, and media. The modeling language is intended to unify past experience about modeling techniques and to incorporate current software best practices into a standard approach. UML includes semantic concepts, notation, and guidelines. It has static, dynamic, environmental, and organizational parts. It is intended to be supported by interactive visual modeling tools that have code generators and report writers. The UML specification does not define a standard process but is intended to be useful with an iterative development process. It is intended to support most existing object oriented development processes.

The UML captures information about the static structure and dynamic behavior of a system. A system is modeled as a collection of discrete objects that interact to perform work that ultimately benefits an outside user. The static structure defines the kinds of objects important to a system and to its implementation, as well as the relationships among the objects. The dynamic behavior defines the history of objects over time and the communications among objects to accomplish goals.

Modeling a system from several separate but related viewpoints permits it to be understood for different purposes.

The UML also contains organizational constructs for arranging models into packages that permit software teams to partition large systems into workable pieces, to understand and control dependencies among the packages, and to manage the versioning of model units in a complex development environment. It contains constructs for representing implementation decisions and for organizing run-time elements into components.

UML is not a programming language. Tools can provide code generators from UML into a variety of programming languages, as well as construct reverseengineered models from existing programs. The UML is not a highly formal language intended for theorem proving. There are a number of such languages, but they are not easy to understand or to use for most purposes. The UML is a general-purpose modeling language. For specialized domains, such as GUI layout, VLSI circuit design, or rule-based artificial intelligence, a more specialized tool with a special language might be appropriate. UML is a discrete modeling language.

It is not intended to model continuous systems such as those found in engineering

and physics. UML is intended to be a universal general-purpose modeling

language for discrete systems such as those made of software, firmware, or digital

logic.

