A picture containing text, wheel

Description automatically generated

**AI-Facial Recognition Attendance System**

**Lam Tze Sun**

**A project report submitted in partial fulfilment**

**of the requirements for the award of**

**Bachelor in Computing (Hons)**

**School of Information Technology**

**SEGi University**

**DECLARATION**

I hereby declare that this project report is based on my original work except for citations and quotations which have been duly acknowledge. I also declare that it has not been previously and concurrently submitted for any other degree or award at SEGi UNIVERSITY or other institutions.

Name: Lam Tze Sun

Student ID: SCSJ1900870

Submission Date:

Signature:

Lecturer Name:

Submission Date:

Signature:

**ACKNOWLEDGEMENT**

“I find that the stranger life gets, the more it seems to make sense.”

First of all, I would like to thank my supervisor Mr.Foong for supervising me along project, giving me several recommendation and guide me through the project. He allowed me to work on this project and try out new things that I have been learned from internet sources by myself and the learning outcome delivered by SEGi College that collaborated with University of Greenwich.

I am very fortunate that I had the chances to be supervised by Mr.Foong, Ms. Nadiah, Ms. Hazlina and other lecturer during my degree study which I learned several programming languages under their guidance, even my most profound gratitude is not enough.

Along with that, I would also like to thank my school principal Mr. Calvin Chan Yee Yuen wholeheartedly for giving me the opportunity to study in SEGi College Subang Jaya so that I could have a wonderful study environment for learning hard and soft skills.

I would also want to thank my parents for raising me up and letting me study abroad in the college. So with due regard, I express my gratitude to them. Apart from that, thanks to Weng Yee Lai for enlighten my present and future. Last but not least, I wanted to express my gratitude toward my friend on the forum that helped me to get an idea for the project and provided the resource for me as reference.

**ABSTRACT**

Manual event for many educational institutions until now is the attendance marking. A lecturer's daily life is an obligatory, common and significant activity. Manual involvement is a little challenging procedure, taking time when analysing it or reporting generations on it. To address this complexity, few automated methods have been created. There are still so many inconveniences, such cost-effectiveness, false generation, precision. A sophisticated and automatic attendance system is needed to replace these initiatives. This study focused precisely on developing a safe attendance marking system, based on one of Face's human gestures. The technique is implemented in one step which is the face recognition that produced from trained faces, using LHBP classifiers. The suggested system would record the attendance of individuals who are present autonomously in a classroom setting, and this is the easiest method of analyzing and proof-oriented approaches.

**TABLE OF CONTENT**

Contents

[**LIST OF TABLE** 6](#_Toc78982296)

[**LIST OF FIGURE** 7](#_Toc78982297)

[**INTRODUCTION** 8](#_Toc78982298)

[1.1 PROJECT BACKGROUND 8](#_Toc78982299)

[1.2 AIM AND OBJECTIVE 8](#_Toc78982300)

[1.3 PROJECT SCOPE 9](#_Toc78982301)

[1.4 PROJECT RESOURCE 10](#_Toc78982302)

[1.5 PROJECT SCHEDULE 11](#_Toc78982303)

[**CHAPTER 2: LITERATURE REVIEW** 12](#_Toc78982304)

[2.1 How does the AI facial recognition system assist? 12](#_Toc78982305)

[2.2 How the system record the attendance of the student? 12](#_Toc78982306)

[2.3 How does the system work? 12](#_Toc78982307)

[2.4 Comparing with the existing system (RFID and Iris scanning) 13](#_Toc78982308)

[2.5 Technique that used by the system – Template matching 15](#_Toc78982309)

[2.6 LITERATURE REVIEW MATRIX 16](#_Toc78982310)

[2.7 COMPARATIVE STUDY 18](#_Toc78982311)

[2.8 CONCLUSION 19](#_Toc78982312)

[**CHAPTER 3: DEVELOPMENT METHODOLOGY** 20](#_Toc78982313)

[3.1 CHOSEN METHODOLOGY 20](#_Toc78982314)

[3.2 JUSTIFICATION OF METHODOLOGY 22](#_Toc78982315)

[**CHAPTER 4: SYSTEM DESIGN** 23](#_Toc78982316)

[4.1 SYSTEM FLOW CHART 23](#_Toc78982317)

[4.2 USE CASE DIAGRAM 24](#_Toc78982318)

[4.3 DATABASE DESIGN 24](#_Toc78982319)

[4.4 USER INTERFACE DESIGN 25](#_Toc78982320)

[**CHAPTER 5: SYSTEM DEVELOPMENT.** 34](#_Toc78982321)

[5.1 SYSTEM DESCRIPTION 34](#_Toc78982322)

[5.2 FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS 34](#_Toc78982323)

[5.2.1 FUNCTIONAL REQUIREMENTS 34](#_Toc78982324)

[5.2.2 NON-FUNCTIONAL REQUIREMENTS 35](#_Toc78982325)

[5.3 SYSTEM ARCHITECTURE 35](#_Toc78982326)

[5.4 HARDWARE AND SOFTWARE REQUIREMENT 36](#_Toc78982327)

[5.5 SYSTEM SCREENSHOT 37](#_Toc78982328)

[**CHAPTER 6: TESTING AND EVALUATION** 49](#_Toc78982329)

[6.1 TEST CASE 49](#_Toc78982330)

[6.2 EVALUATION PLAN 51](#_Toc78982331)

[6.3 RESULT AND DISCUSSION 54](#_Toc78982332)

[**CHAPTER 7: CONCLUSION AND RECOMMENDATIONS** 65](#_Toc78982333)

[7.1 INTRODUCTION 65](#_Toc78982334)

[7.2 PROBLEM ENCOUNTERED 66](#_Toc78982335)

[7.3 SYSTEM LIMITATION 67](#_Toc78982336)

[7.4 FUTURE WORK AND IMPROVEMENT 68](#_Toc78982337)

[**REFERENCES** 69](#_Toc78982338)

# **LIST OF TABLE**

[**Table 1.0 Literature review matrix** 17](#_Toc78982072)

[**Table 2.0 Comparative Study** 18](#_Toc78982073)

[**Table 3.1 Test case** 50](#_Toc78982074)

[**Table 4.1 Comparison of existing system and my developed system** 68](#_Toc78982075)

# **LIST OF FIGURE**

[**Figure 1.1 Gantt Chart** 10](#_Toc78982250)

[**Figure 2.1 RFID System** 13](#_Toc78982251)

[**Figure 3.1 Rapid Application Development** 21](#_Toc78982252)

[**Figure 4.1 System Flowchart** 22](#_Toc78982253)

[**Figure 5.1 Use Case Diagram** 23](#_Toc78982254)

[**Figure 6.1 Database Design** 23](#_Toc78982255)

[**Figure 7.1 Main Screen of the System** 24](#_Toc78982256)

[**Figure 8.1 Alert pop out after pressing the “add image to database” button** 25](#_Toc78982257)

[**Figure 9.1 Webcam start rolling and accepting live image input.** 26](#_Toc78982258)

[**Figure 10.1 Alert pop out when pressed “Start program with live camera”** 27](#_Toc78982259)

[**Figure 11.1 Camera started to accept live image as input and detecting face id.** 28](#_Toc78982260)

[**Figure 12.1 option window pop out to let user choose whether to import image or video.** 29](#_Toc78982261)

[**Figure 13.1 Face detector detect the face id from imported video.** 30](#_Toc78982262)

[**Figure 14.1 Face detector detect the face id from imported image.** 31](#_Toc78982263)

[**Figure 15.1 Option window pop out to let user choose the option.** 32](#_Toc78982264)

[**Figure 16.1 Database diagram** 33](#_Toc78982265)

[**Figure 17.1 System Architecture for Ai-FRAS** 34](#_Toc78982266)

[**Figure 18.1 Main page** 36](#_Toc78982267)

[**Figure 19.1 Capture image after enter name in entry block** 37](#_Toc78982268)

[**Figure 20.1 Image capture window pop-out** 38](#_Toc78982269)

[**Figure 21.1Alert bot pop out after pressing “start program with live camera”** 39](#_Toc78982270)

[**Figure 22.1 Detector started to get working** 40](#_Toc78982271)

[**Figure 23.1 Option for import video or image** 41](#_Toc78982272)

[**Figure 24.1 File explorer launched to choose image/video** 42](#_Toc78982273)

[**Figure 25.1 Detector use imported image as input for detecting face id.** 43](#_Toc78982274)

[**Figure 26.1 Detector use imported video from google meet as input for detecting face id.** 44](#_Toc78982275)

[**Figure 27.1 option for opening attendance report** 45](#_Toc78982276)

[**Figure 28.1 Attendance sheet open up in .csv format for review.** 46](#_Toc78982277)

[**Figure 29.1 opening readme in visual studio.** 47](#_Toc78982278)

[**Figure 30.1 Gender** 53](#_Toc78982279)

[**Figure 31.1 Age** 54](#_Toc78982280)

[**Figure 32.1 Roles** 55](#_Toc78982281)

[**Figure 33.1 System usability testing.** 55](#_Toc78982282)

[**Figure 34.1 User friendly evaluation** 56](#_Toc78982283)

[**Figure 35.1 Simplicity and feasibility evaluation** 56](#_Toc78982284)

[**Figure 36.1 System function evaluation** 57](#_Toc78982285)

[**Figure 37.1 Button function evaluation** 57](#_Toc78982286)

[**Figure 38.1 Input acceptance evaluation** 58](#_Toc78982287)

[**Figure 39.1 Importing image and video** 58](#_Toc78982288)

[**Figure 40.1 Attendance sheets review evaluation.** 59](#_Toc78982289)

[**Figure 41.1 Entry acceptance evaluation.** 59](#_Toc78982290)

[**Figure 42.1 Alert box evaluation** 60](#_Toc78982291)

[**Figure 43.1 Webcam activation evaluation** 60](#_Toc78982292)

[**Figure 44.1 Efficiency evaluation** 61](#_Toc78982293)

[**Figure 45.1 Ease of access and comfortable evaluation.** 62](#_Toc78982294)

[**Figure 46.1 Rating of the system.** 63](#_Toc78982295)

# **INTRODUCTION**

## 1.1 PROJECT BACKGROUND

This research is about the AI Facial Recognition Attendance System that use Python programming languages to create an interface for the user to record the student’s face that enter the classroom physically. Whereby the system is connected to the College student database, the system will automatically use the student id’s picture as the compared picture to match with the face of the student that walk in-front of the camera, thus recording their attendance in the database.

## 1.2 AIM AND OBJECTIVE

The AI Facial Recognition Attendance System is aim to assist both the lecturer and student for the attendance in the class. Thus, ensuring that the student does not missed the attendance of the class that will lead to the miscalculating of total mark for their final examination. After that, the objective that has been set to achieve for the system is:

1. 1. To promote the new technology of artificial intelligence and show the advantages of it in assisting our daily life.
2. Assisting lecturer in confirming the student’s attendance with just one click or touch and able to review the concurrent attendance.
3. To ensure that the student’s attendance is taken at a more precise way which can be up to 100% accurate.

## 1.3 PROJECT SCOPE

As we know, human motions are the best aspect to accomplish high-end safety. A gesture is the face, one of the greatest. The component is face recognition. The best elements are provided in our existing systems in terms of security progress. Due to the condition, there are two step to resolve the proposed system which is Face Detection and Face Recognition.

1. **Train Data**: We must train the data to accept the input before resolving facial recognition. The camera is deployed here to train and scan the data for the future picture data training. Haar functions will be retrieved from the picture and stored in a specific RGB folder/database, which may be processed by the system to collect each person's facial data.
2. **Face Detection**: We require an item and must identify it by camera before training or setting up a classifier. We need to work with hair classifiers to accept whether it is facial or not. We have collected classifiers for the location. The number of steps and the minimum size of the faces to be identified are different amongst them. The faces in various locations have been detected. After collecting all data, they are displayed in OpenCV file “haar casecade.xml” and this helps us recognise a face that is an object.
3. **Face Recognition**: OpenCV allows XML file generation to hold the features collected from the Face Recognizer class datasets. Images stored are imported, grayed-out and stored in two lists with the same indexes with identical IDs. Objects are generated with face recognizer class. Here for each image, LHBP classifier is produced. In bigger cells, histograms are utilised to speed up the occurrence frequency of values. The edges can be discovered when values change by evaluating cell results. Feature vectors can be produced by calculating the values of all cells and by combining histograms. Pictures may be categorised using an associated ID processing. Electronic pictures are categorised with the same technique and the data set is compared to the distance. If it is a known or unknown face, a threshold is established. Recognizer objects are built and pictures are imported, redimensioned, transformed and saved in a vector to NumPy Arrays. The picture ID is collected and saved in another vector by dividing the file name. They are trained with all three items.

## 1.4 PROJECT RESOURCE

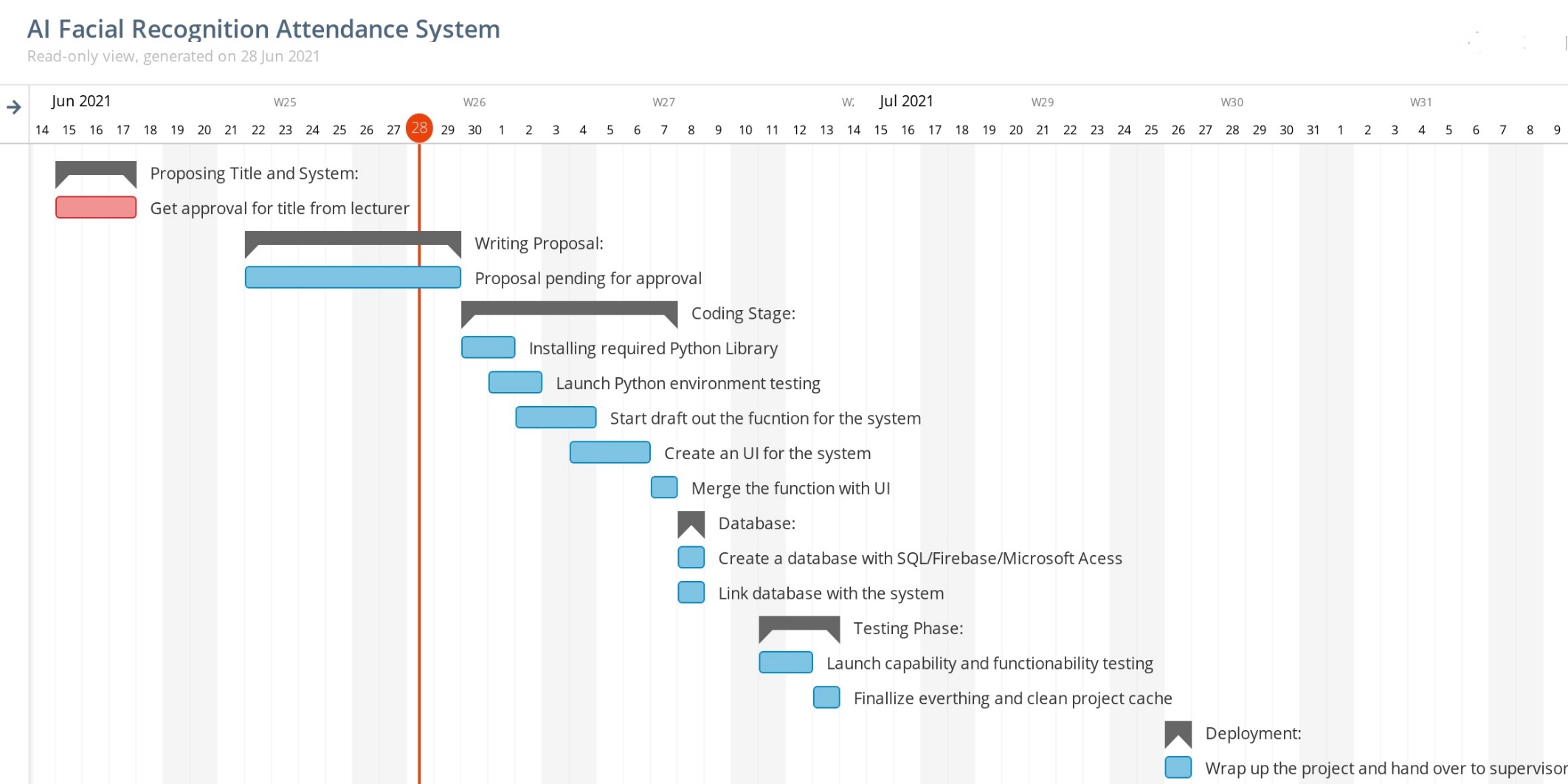
In this project, I decided to use Python as the programming languages and its technology to help me develop the facial recognition system. Due to the circumstances of its huge external library dependency packages, I will be able to download and import the module that has been downloaded. There are several packages that I have used when developing the system, first are Open-Cv. OpenCV is a package with hundreds of functions for image processing. The library is quite handy for the processing of images.

Even without creating a single code one may obtain the intended result. Under the open-source BSD licence, the library is cross-plan and free to use. The following is an example of certain available functions:

1. **Derivation**: Contour delimitation gradient/laplacian computation.
2. **Segmentation**: Thresholding, transformation of distance, first- and foreground detection, segmentation of the water shaft.
3. **Filtering**: Linear and non-linear filters, morphology operation.
4. **Interest point**: Detection and match.
5. **Video processing**: Flow optic, subtraction of background, object tracking.
6. **Photography**: Realisation of panoramas, HDR, painting of images.
7. **Cascade detector**: Face, eye and vehicle plates detection.
8. **Histogram**: Calculation, equalisation and location of objects with back screening algorithm.
9. **Hough transform**: Detection of lines, segments, circles and geometric forms.

Other than that, I also used the facial recognition model which help the Open-Cv to recognize the face id of user in a more precise way. For the GUI design, Tkinter are used to aid me in designing the graphical user interface. This module will help me to make the work more efficient by just adding several lines of code to generate all those labels, button and also the windows for the program. For database, I decided to use Microsoft access database which are linked with the .csv file that are generated by the system.

## 1.5 PROJECT SCHEDULE



**Figure 1.1 Gantt Chart**

# **CHAPTER 2: LITERATURE REVIEW**

## 2.1 How does the AI facial recognition system assist?

Each learning institution needs a solid and comprehensive method to track their students' attendance and has its own way to achieve this, some attend with a paper sheet by calling their names throughout lecture hours and some have implemented biometrics systems, such as fingerprint, RFID card reader and Iris system for taking part (R.S., 2020). The usual way of calling student names manually is time-consuming. If each student assigns an identification card on the RFID card system, but a risk of card loss or unauthorised individual can abuse the card for false participation. This is not possible. While they all have their own defects and are not 100% accurate in other biometrics such as fingerprint, iris or voice recognition. The clever approach to attendance management is the use of facial recognition for the purposes of attendance marking. Facial recognition among other approaches is more precise and quicker and decreases the likelihood of proxy participation. Face recognition provides a passive identity that is not necessary for the identity of a person to be identified

There are two phases that are included in face recognition, one step involves facial detection, and the second step involves identification with the current database of detected facial pictures. There are several ways for face detection and recognition. Face recognition either works in appearance that includes whole-face or geometric characteristics such as eye, nose, eyebrows and cheeks to identify the face (Ghalib Al-Muhaidhri, 2019).

## 2.2 How the system record the attendance of the student?

Once the picture in the JSON-file has been identified, python creates the actual student roll numbers and returns that the system constructs the attendance table, with the appropriate topic id, when data is returned. Then the data is sent to python to automatically save the table to an excel sheet. For example, if a system generates excel sheet by delivering a compiled sheet to a python array, the python initially checks if any excel sheet exits from that date, or if it creates a distinct worksheet by object ID so that the presence of various subjects may be separated.

## 2.3 How does the system work?

The suggested method is meant to automate attendance and minimise the faults in the present manual approach. The system calculates the attendance subjects wisely, that is to say, students and topics are manually entered by the manager and, if the time comes for the respective subject, the system begins snapshots and finds out whether or not the human face is in the picture concerned. The system utilised the Oriented Gradient Histogram in the calculation and comparison of 128-d facial characteristics for facial identification with deep learning techniques. When faces are identified, the system calculates student attendance with a particular topic ID in real time for recognised students and the system will automatically create and save an Excel sheet. The first half of our system consists of the front end of GUI, based on a Java-Script Electron JS stack that is used as a customer; the second part is the back end that consists of a logic and Python server. And we know that we have been using IPC (inter-personal communication) methods with zero library as a bridge for communicating those two languages, and that both the different languages cannot interact directly.

With the assistance of the Zero PC Library, the Electron JS will call Python functions and exchange data over TCP.

## 2.4 Comparing with the existing system (RFID and Iris scanning)

To compare with the RFID (Radio-frequency identification) and iris scanning, we need to know that how they are functioning. First of all, RFID is the most contemporary technology to minimise our complicated everyday lives. Radio frequency Identification is a technology. RFID is essentially a small electrical gadget composed of a small chip and an antenna. The chip can usually carry few or fewer bytes of data. The RFID gadget works with the same purpose as a credit card or ATM card's barcode or a magnetic strip; this enables a unique object identity and the RFID unit must be scanned to obtain the identifying information exactly like a bar code or magnetic strip needs to be scanned to obtain the information.



**Figure 2.1 RFID System**

Other than that, the iris is an externally visible yet protected organ that remains constant throughout adult life. These features make it highly desirable for the identification of individuals as a biometric. Imaging processors may be used to extract from a digital image of the eye the unique iris pattern and encode it into a biometric template which can be kept in a repository database. The biometric template provides the unique data contained in the iris with an objective mathematical expression and comparisons between modules may be done. When you want to identify a subject using a system of an iris recognition, your eyes are photographed first, followed by a template for your iris area. This template is evaluated to other templates in a databases until the object has either been recognised or no match has been located and the subject has been unidentified.

To make a comparison, RFID technique must have either the bar code or magnetic stripe that has been deployed into the student id card or their personal belonging and it is not efficient compared to the facial recognition which is already have on their body. Thus, the student id card that has the RFID tag may be also lost and therefore the student will not be able to mark down their attendance. In this case, facial recognition is more convenient and safety to ensure their attendance is marked even though they lost their student id card. Other than that, iris scanning is better than RFID but the system needs time and the student must queue up before they enter the classroom in order to let the system identify their iris and mark their attendance. This will affect the class time and facial recognition can solve this problem by just scan the facial ID for the student that enter the classroom without the need of queuing up and waiting for the scan.

## 2.5 Technique that used by the system – Template matching

A basic variant of the template match is that the test image is compared using an appropriate metric, such as the Euclidean distance, representing the complete surface by a single template. More advanced models match facial recognition are also available. More than one face template from various angles of view can be used to portray the face of a person. A face may alternatively be rendered from a particular point of view by a group of several unique smaller templates. Before matching, the facial picture of grey levels may also be processed properly. The four templates, eyes, nose, mouth and face, have been automatically picked for all possible faces in the section. They tested the efficiency of their matching geometric method with the matching template algorithm on the same faces dataset which comprises certain picture from the total picture in the trained dataset.

## 2.6 LITERATURE REVIEW MATRIX

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ID | Year | Author | Title | Focus area | Problem | Methodology | Findings |
| 1 | 2019 | M. Khan, S. Chakraborty, R. Astya and S. Khepra | Face Detection and Recognition Using OpenCV | reduce the large amount of data storage to the size of the feature space that is required to represent the data economically. The wide 1-D pixel vector made of the 2-D face picture in compact main elements of the space function is designed for facial recognition by the PCA. | Face recognition precision is not efficient | Use an algorithm by developing programming on OpenCV, Haar Cascade, Eigenface, Fisher Face, LBPH, and Python | Face recognition in a real-time setting has an exciting area and a rapidly growing challenge. |
| 2 | 2018 | Bhatti, K. L., Mughal, L., Khuhawar, F. Y., & Memon, S. A. | Smart Attendance Management System Using Face Recognition | Developing a smart attendance management system using face recognition | Traditional attendance marking took too much time and not precise | By using python and the component of open-cv and also haar-cascade algorithm | The total precision and attendance marking rose up. |
| 3 | 2019 | Jawed, S. | Face Recognition for Smart Classroom Attendance Management System using Computer Vision | Implementing computer vision in classroom attendance marking. | Make the attendance taking session more efficient and also convenient. | Implement computer vision to the system | Improved system usability and feasibility |
| 4 | 2019 | Sutabri, T., Pamungkur, P., Kurniawan, A. and Saragih, R. | Automatic Attendance System for University Student Using Face Recognition Based on Deep Learning | Deploying deep learning algorithm into the smart attendance marking system. | Attendance taking only using the computer vision is less efficient. | Deploy deep learning algorithm as a hot fix into the system | Improved precision of the system for facial recognition. |
| 5 | 2020 | Yadav, R., Chauhan, S., Meenu, M. and Gupta, S. | FACE RECOGNITION BASED ON ATTENDANCE MANAGEMENT SYSTEM | Use facial recognition technology to assist in taking attendance | Traditional method of taking attendance is not that efficient | Developing a system with python and its library package module. | Improved efficiency and also cost of time taking during attendance marking. |

**Table 1.0 Literature review matrix**

## 2.7 COMPARATIVE STUDY

|  |  |  |
| --- | --- | --- |
| Feature of system | RFID Attendance | Fingerprint Attendance |
| Contactless | Yes | No |
| Privacy | Yes | Yes |
| Work with glasses/goggles | Yes | Yes |
| Stability | Low | High |
| Speed | Excellent | Good |
| Accuracy | Moderate | High |
| Work with glove | Yes | No |
| User acceptance | Moderate | Moderate |
| Long term stability | Moderate | High |
| Security level | Low | High |
| Error incidence | Lost of access card | Dryness, dirt, age |

**Table 2.0 Comparative Study**

## 2.8 CONCLUSION

The AI Facial Recognition Attendance system is intended to overcome existing manual systems problems. We employ the notion of facial recognition to recognise the student's participation and improve the system. In many positions and modifications, the system functions satisfactorily. This system has to be improved in the future since occasionally students from some distance cannot identify this system, and there are some processing limitations, and work with a processing system that can make this system even more precise and efficient.

# **CHAPTER 3: DEVELOPMENT METHODOLOGY**

## 3.1 CHOSEN METHODOLOGY

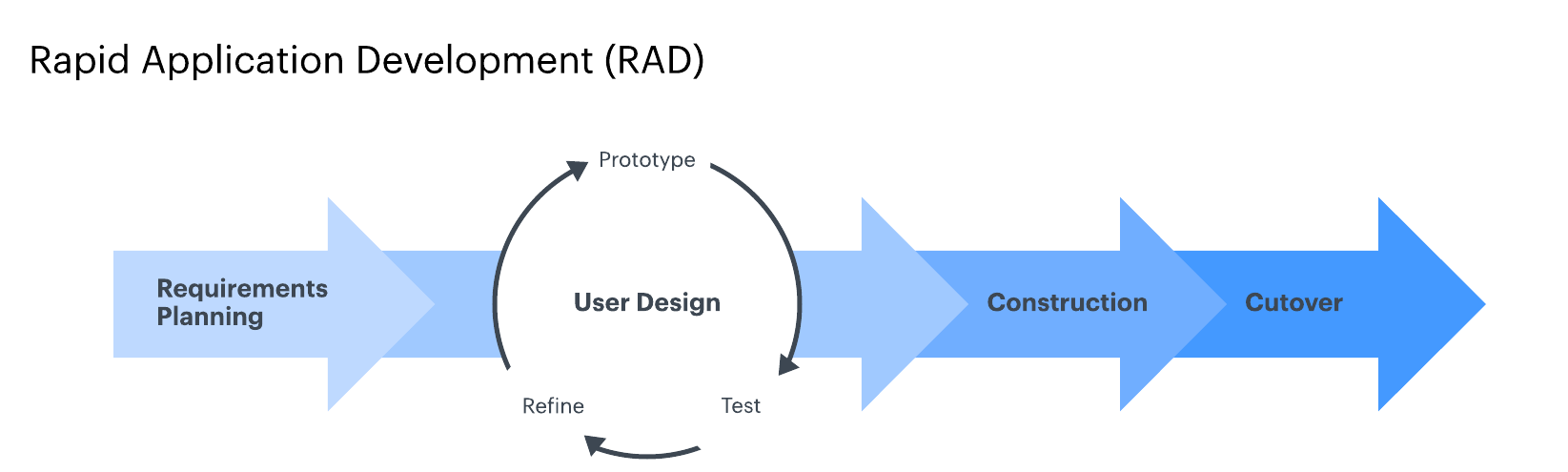
The methodology I have decided to use for this facial recognition project is the Rapid Application development model (RAD). The development model that used in developing the AI Facial Recognition Attendance System is the RAD model or as well-known as rapid application development model which prioritise to develop a software or system in a short time. For RAD model, there are 4 phases which is requirement planning, user design, construction and cutover. Due to the time frame that is short, this model will ensure that the system will be run just in time according to the project plan that has been planned. RAD means a life cycle meant to provide the traditional life cycle a much faster development and higher quality systems. It has been developed to use sophisticated software like CASE tools, tools for prototyping and the production of code. High speed, high quality and low cost are the key objectives of RAD. RAD is a development methodology that focuses on people and increments. It is essential for all parties to engage active users, as well as collaboration and cooperation. In the development life cycle, testing is incorporated so that both developers and users gradually test and assess the system. Rapid Application Development is a 'computer system construction strategy' that combines CASE tools, user-driven prototyping, and tight time-limits for project delivery into a powerful, proven, dependable recipe for high quality and productive performance. RAD increases dramatically the quality of final systems and reduces their time for construction."

There are 4 phases in the Rapid Application Development model which are:

1. Requirements planning: This step corresponds to a project scope meeting. While the planning phase is condensed in relation to other methods of project management, this is a vital step towards the project's eventual success. Throughout this phase, developer, customer (software user) and team members are communicating to identify project objectives and expectations, as well as present and future difficulties to be handled throughout construction.
2. User design: Once the concept is finalised, it is time for development, through several prototype iterations, to construct the user design. This is RAD's flesh and potato methods—and what distinguishes them from other techniques of project management. During this stage, customers cooperate with developers to guarantee that their demands are addressed at all stages of the design process. It is nearly like custom software development in which consumers may test each product prototype to fulfil their expectations at every step. In an iterative process all problems and kinks are created. The developer develops a prototype, the customer (user) tests it and then communicates what has worked and not. This approach allows developers to adjust the model until it reaches an acceptable design. Both software developers and customers learn from the experience to ensure that something is not possible to crack.
3. Rapid construction: Phase 3 starts with the design stage and transforms the prototypes and beta systems into the functioning model. Since many challenges and modifications have been handled in the comprehensive iterative design stage, developers may build the final model of work faster than they can by using a traditional method to project management. The phase divides into a few minor steps:
4. Speed building preparation
5. Development of the programme and application
6. Coding
7. System, integration and unit testing

The programmers, coders, testers and developers' team for software development works together to ensure all is worked properly and that the results fulfil the client's requirements and goals. This third step is crucial since the customer can always enter the process. You can provide adjustments, modifications or even new ideas that can fix problems as they emerge.

4. Cutover: This is the stage when the final product is launched. It involves the conversion of data into the new system, testing and transformation, and user training. All last modifications are done while the coders and customers are still looking for errors in the system.



**Figure 3.1 Rapid Application Development**

## 3.2 JUSTIFICATION OF METHODOLOGY

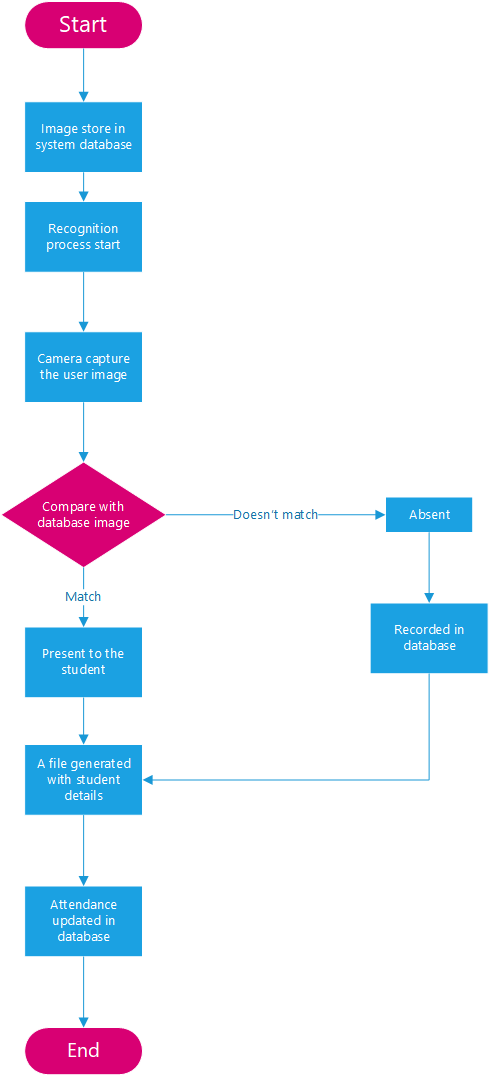
RAD is the SDLC model of one kind. It is a model ideal for modest projects that must be completed over a short period of time. The Ai-Facial Attendance Recognition System (Ai-FRAS) aims to build a facial recognition recording system. The complexity is not that great, it is not regarded as a big job. In addition, this project does not require any costly high-tech gadget that poses great danger for the project.

In comparison to the old SDLC paradigm, RAD is a superior model for development. The conventional approach of SDLC follows a rigorous process model which focuses more on the analysis and collection of requirements before the coding begins. The client must provide a confirmation requirement that cannot generally be changed afterwards. This prevents users from feeling the items, because for a long time there are no functional prototypes ("SDLC - RAD model"). RAD is more flexible in terms of this problem. A prototype will be created to allow the user to observe how the project progresses.

After a prototype view, sometimes users may feel their initial need is not acceptable. If so, they can provide the developer with comments. In the design phase, developers can revert to any requirements throughout the RAD design phase. This makes RAD adaptable since it can always be swapped from building to design to meet the user's requirements. In addition, RAD is developing quicker. This is because throughout the project development the prototype was produced. The whole prototype that is being constructed during the development of the project is maintained. If a part of the prior prototype is selected, and tested previously, the test time can be saved, as it does not need to be tested again. The reusability of the component can therefore aid to shorten the duration of development. Finally, all this is why RAD was chosen for the project as the SDLC model.

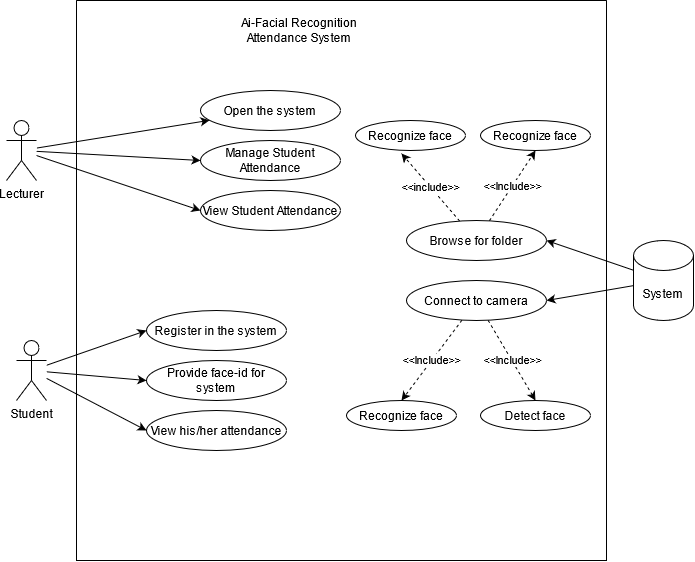
# **CHAPTER 4: SYSTEM DESIGN**

## 4.1 SYSTEM FLOW CHART



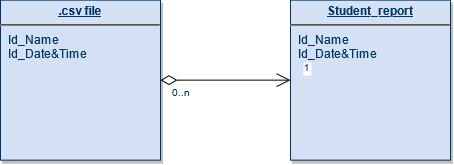
**Figure 4.1 System Flowchart**

## 4.2 USE CASE DIAGRAM



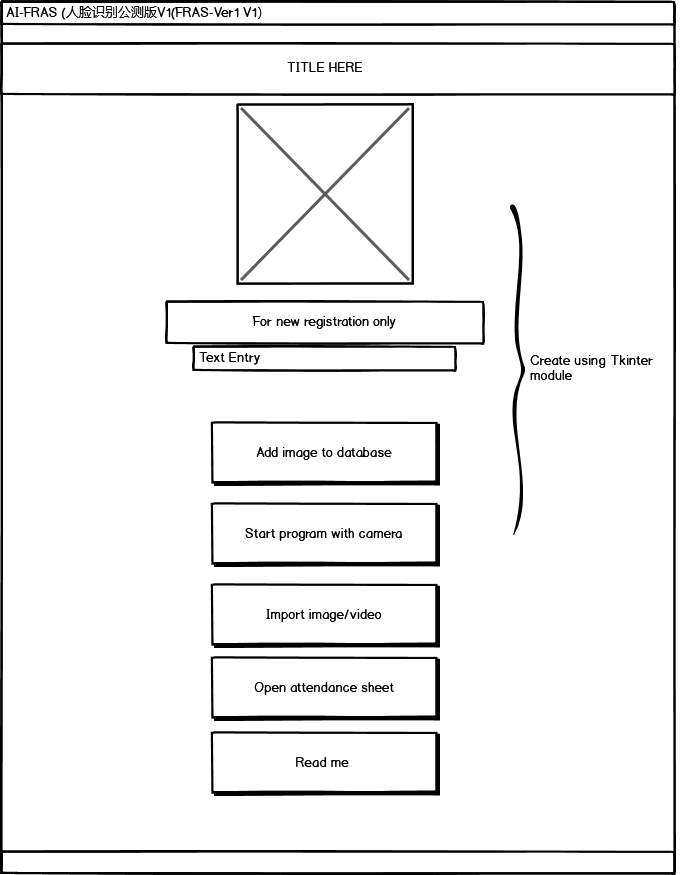
**Figure 5.1 Use Case Diagram**

## 4.3 DATABASE DESIGN

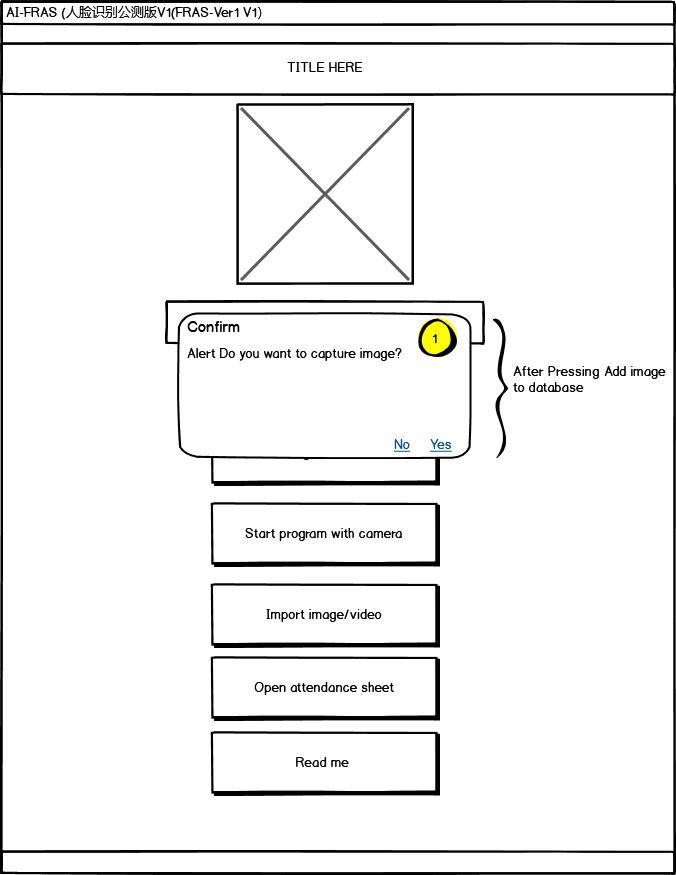


**Figure 6.1 Database Design**

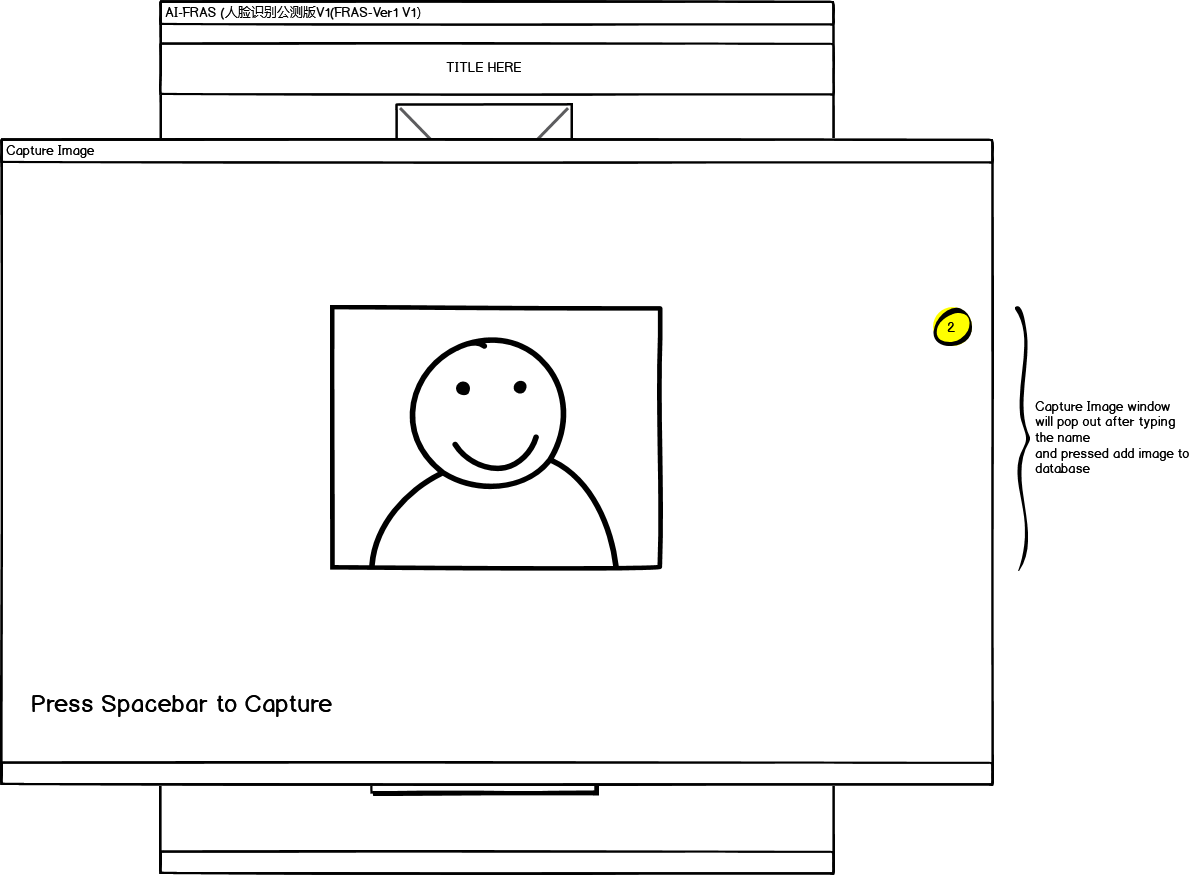
## 4.4 USER INTERFACE DESIGN



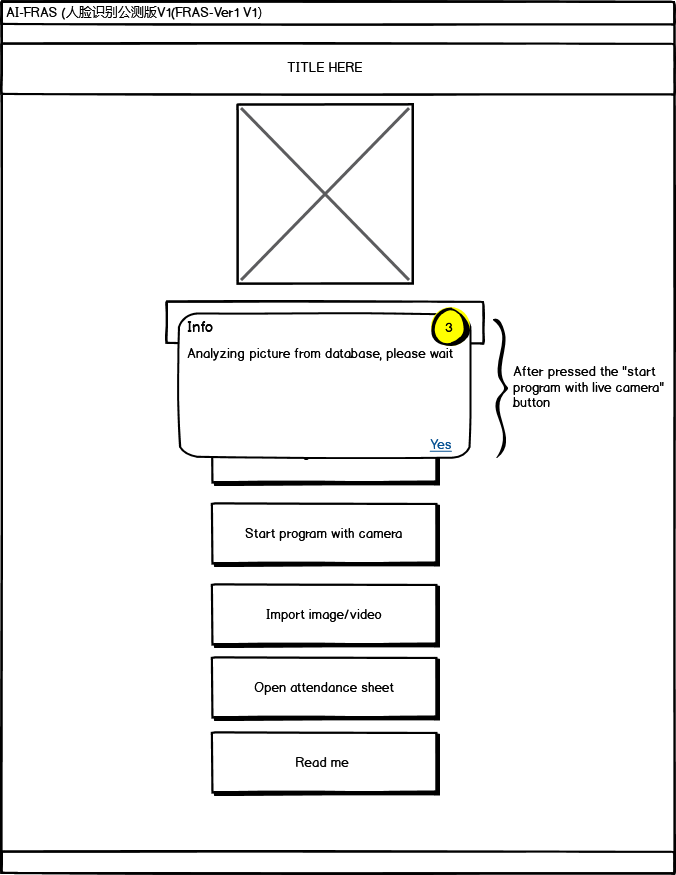
**Figure 7.1 Main Screen of the System**



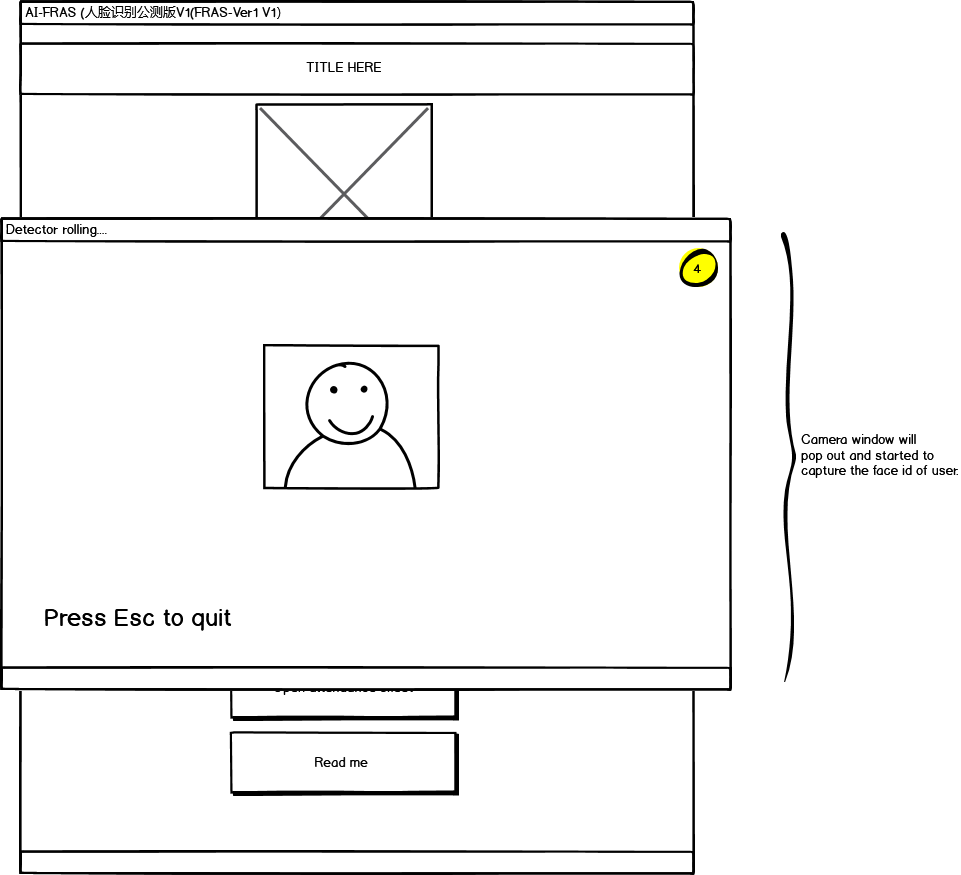
**Figure 8.1 Alert pop out after pressing the “add image to database” button**



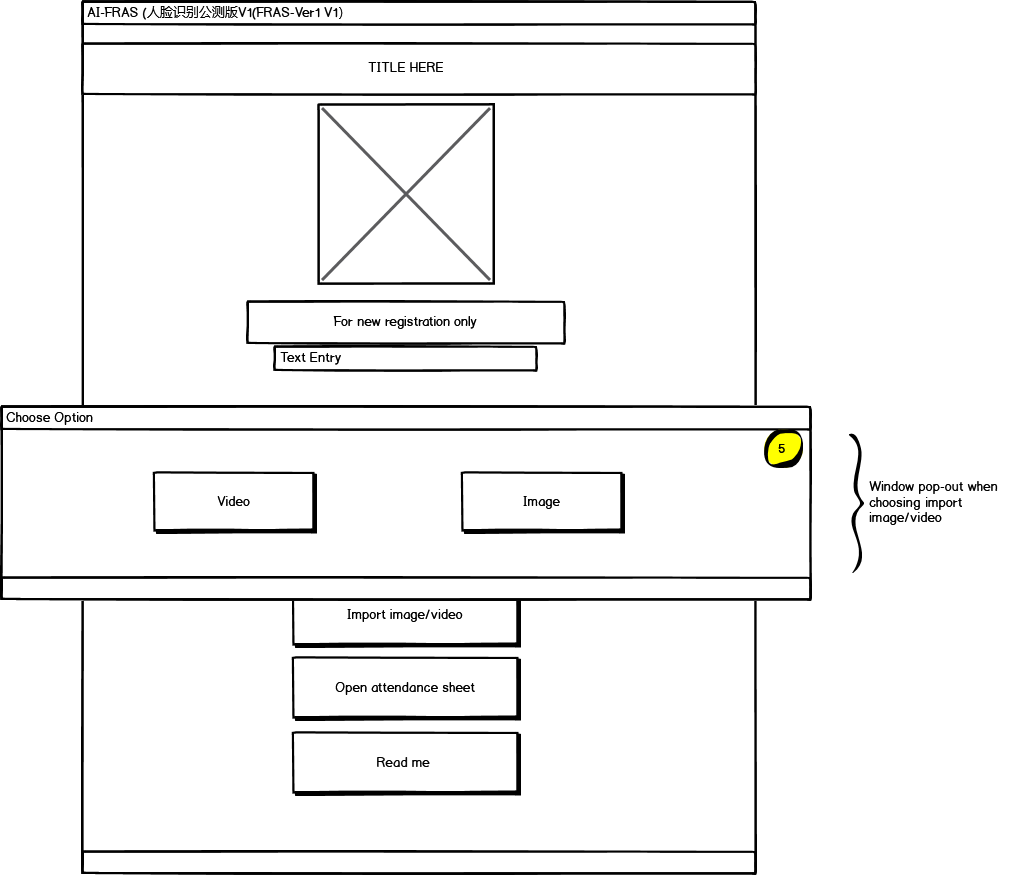
**Figure 9.1 Webcam start rolling and accepting live image input.**



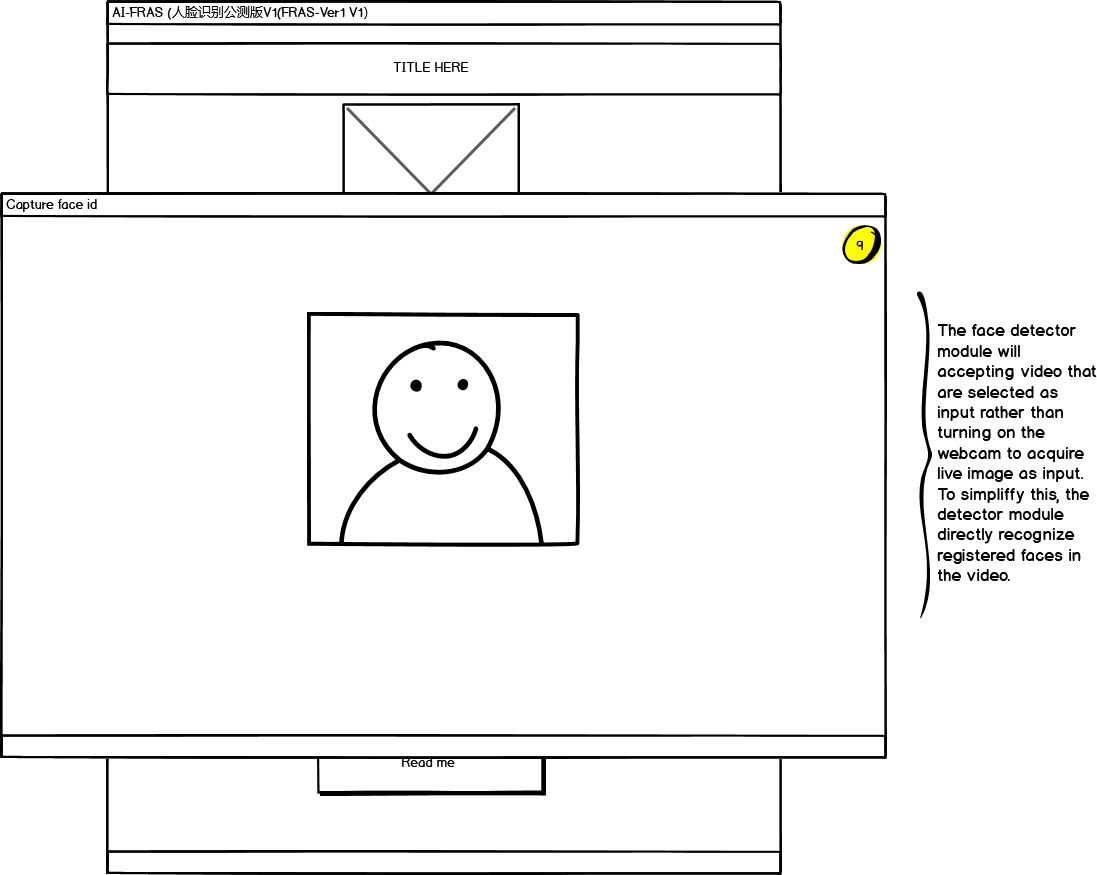
**Figure 10.1 Alert pop out when pressed “Start program with live camera”**



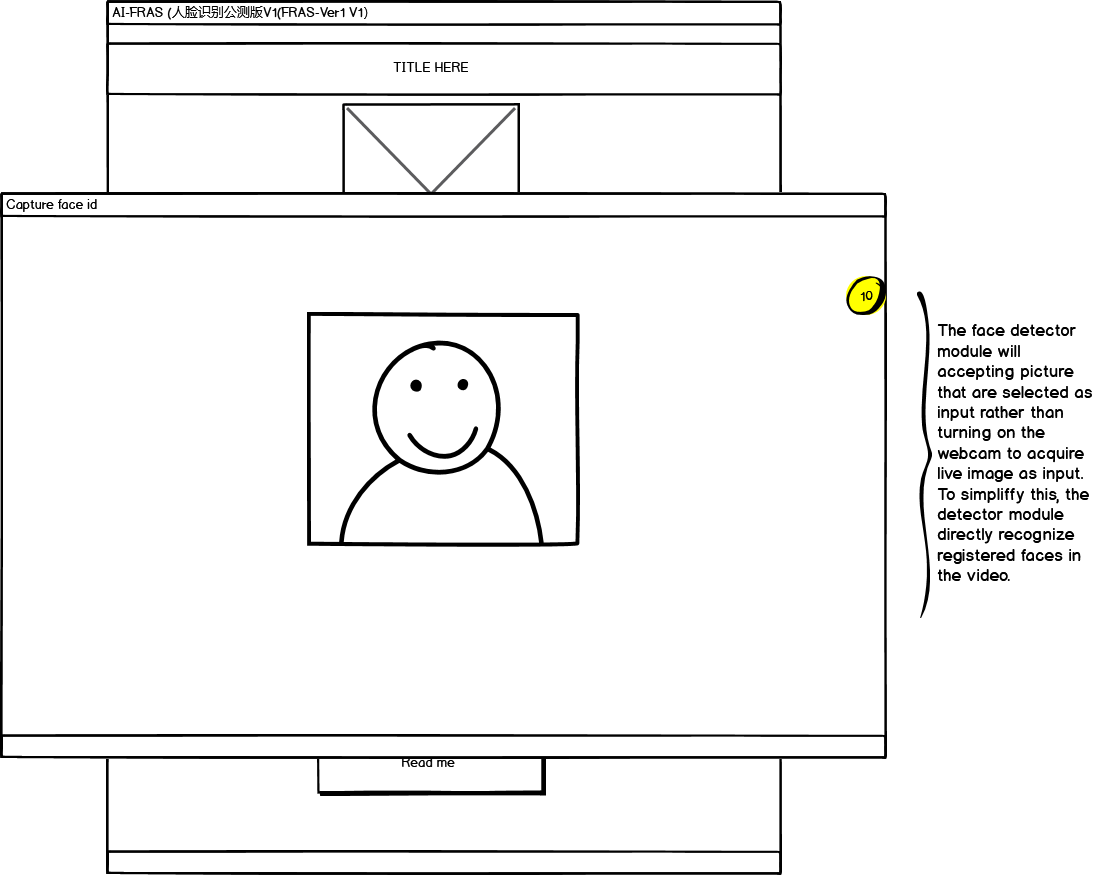
**Figure 11.1 Camera started to accept live image as input and detecting face id.**



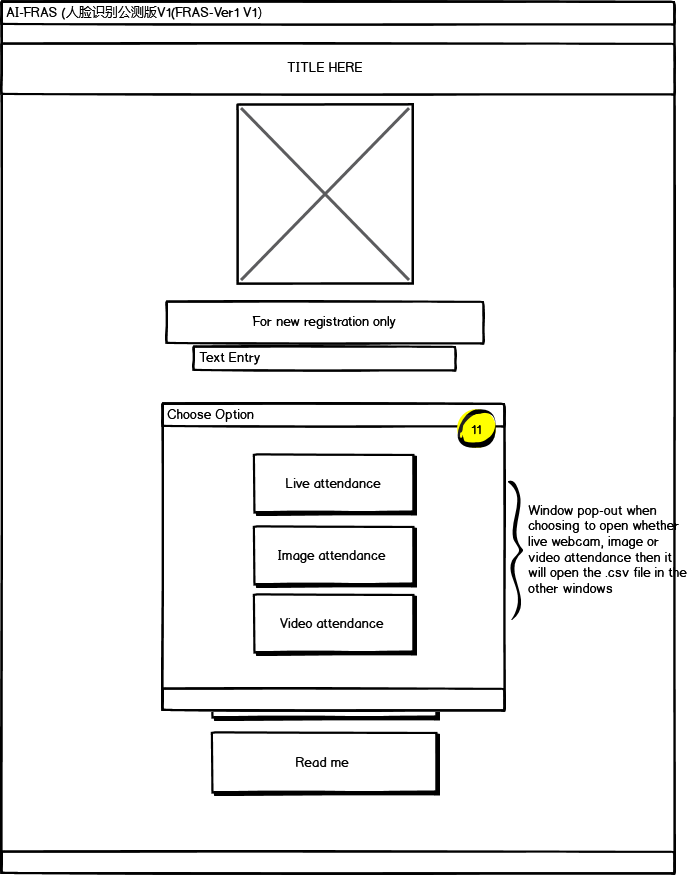
**Figure 12.1 option window pop out to let user choose whether to import image or video.**



**Figure 13.1 Face detector detect the face id from imported video.**



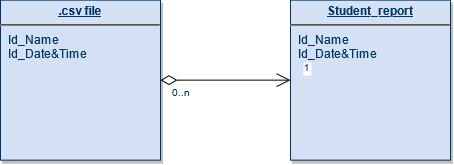
**Figure 14.1 Face detector detect the face id from imported image.**



**Figure 15.1 Option window pop out to let user choose the option.**

# **CHAPTER 5: SYSTEM DEVELOPMENT.**

## 5.1 SYSTEM DESCRIPTION



**Figure 16.1 Database diagram**

The flowing of data took place when the face id of a student is detected. Each face id has different face positioning that are processed by the Haar-Cascade and OpenCV algorithm. In this case, the detail of the student will be recorded inside the .csv file that are generated by the system into the Microsoft access database which linked to it.

## 5.2 FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS

## 5.2.1 FUNCTIONAL REQUIREMENTS

1. User are able to open the system

2. User are able to enter their name/id inside the entry field

3. User are able to add image into the database for image training

4. User able to start the webcam after pressing “start program with live camera”

5. The camera is functioning and accepting face id input

6. User able to choose the option whether to import image or video as the input.

7. User are able to import image as input

8. User are able to import video as input

9. User able to open the attendance sheets in 3 options.

## 5.2.2 NON-FUNCTIONAL REQUIREMENTS

**Maintainability**

Pycharm is used in the development process as a tool for easy maintenance.

**Portability**

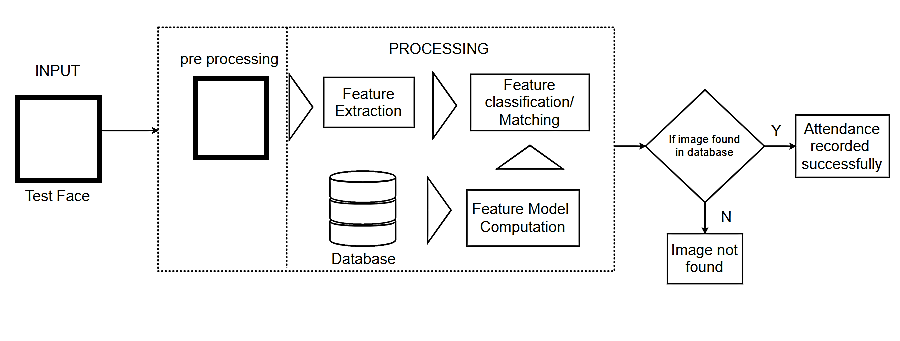
The programme is written in the Python language to ensure portability.

**Reliability and consistency**

Users should be able to monitor the attendance of their students using the system.

In addition, tendencies must be well managed. This means that the system should generate the proper output based on the inputs

## 5.3 SYSTEM ARCHITECTURE

****

**Figure 17.1 System Architecture for Ai-FRAS**

## 5.4 HARDWARE AND SOFTWARE REQUIREMENT

**5.4.1 HARDWARE REQUIREMENTS**

* Processor: Minimum 1Ghz CPU; Recommended 2Ghz or more.
* Hard Drive: Minimum 20GB; Recommended 50GB or more for larger image library
* RAM (Memory): Minimum 2GG; Recommended 4GB or more
* Webcam or camera attached

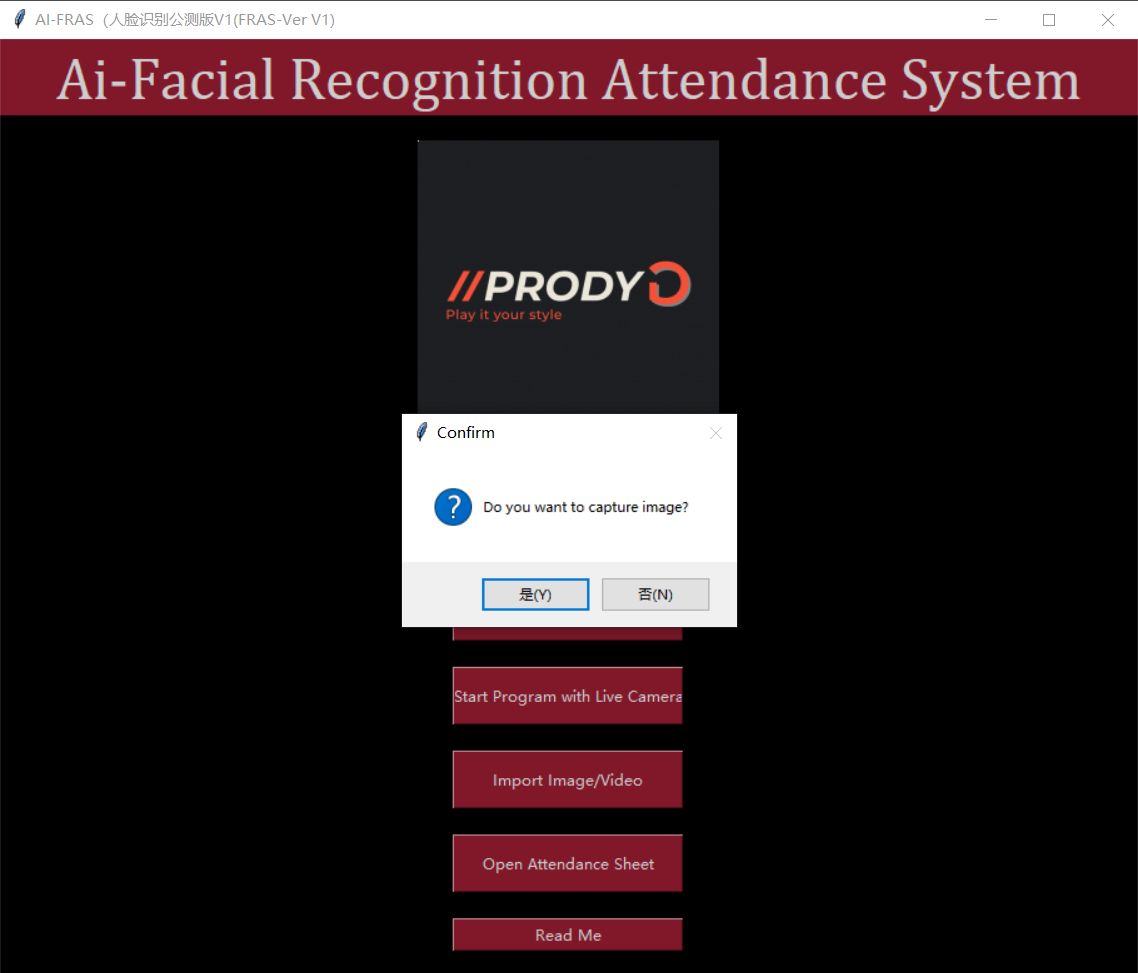
**5.4.2 SOFTWARE REQUIREMENTS**

* OS: Minimum Windows 7; Recommended Window 10 or Linux 5.13.7
* Python 3.9
* Microsoft Visual Studio (C++ Dependencies installed)
* Microsoft Access Database
* Python module in requirement.txt (For developer)

## 5.5 SYSTEM SCREENSHOT

****

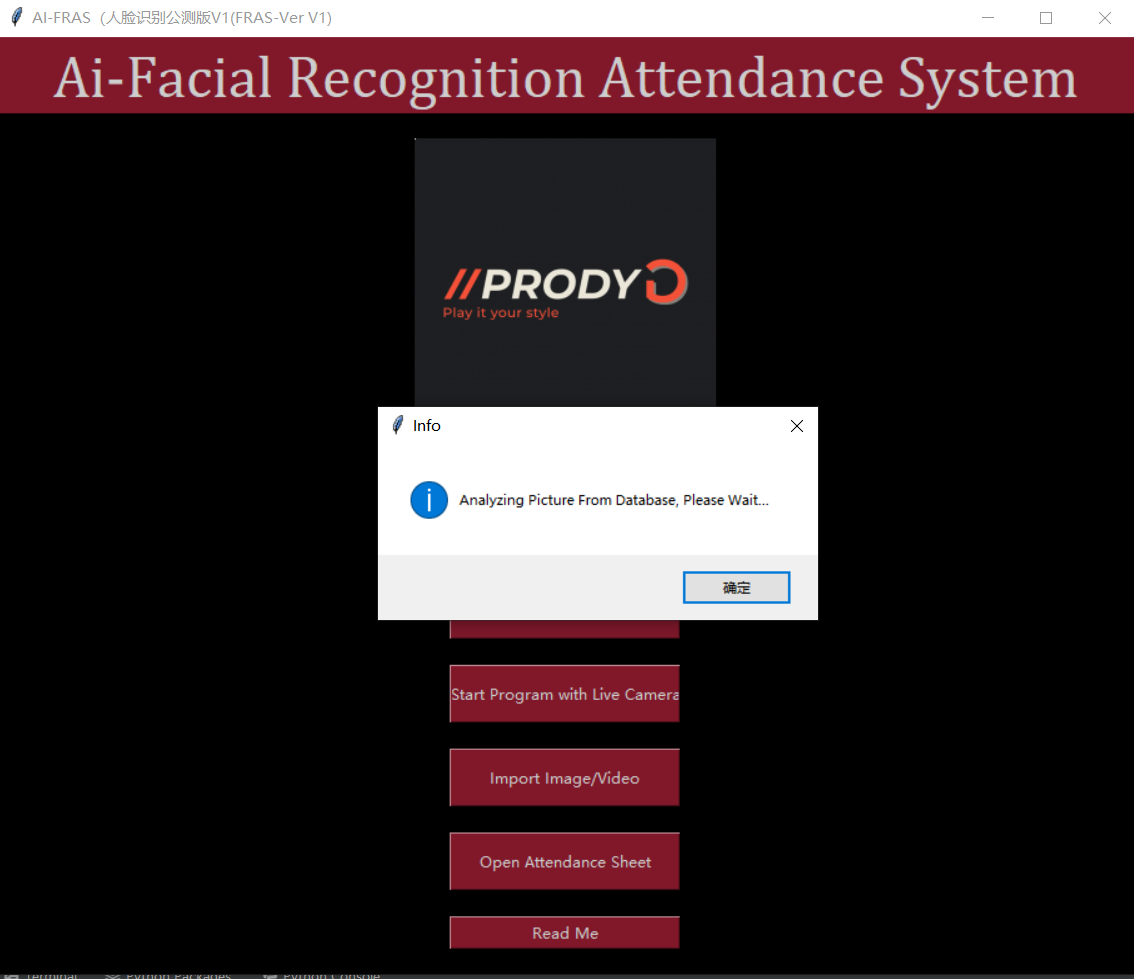
**Figure 18.1 Main page**



**Figure 19.1 Capture image after enter name in entry block**



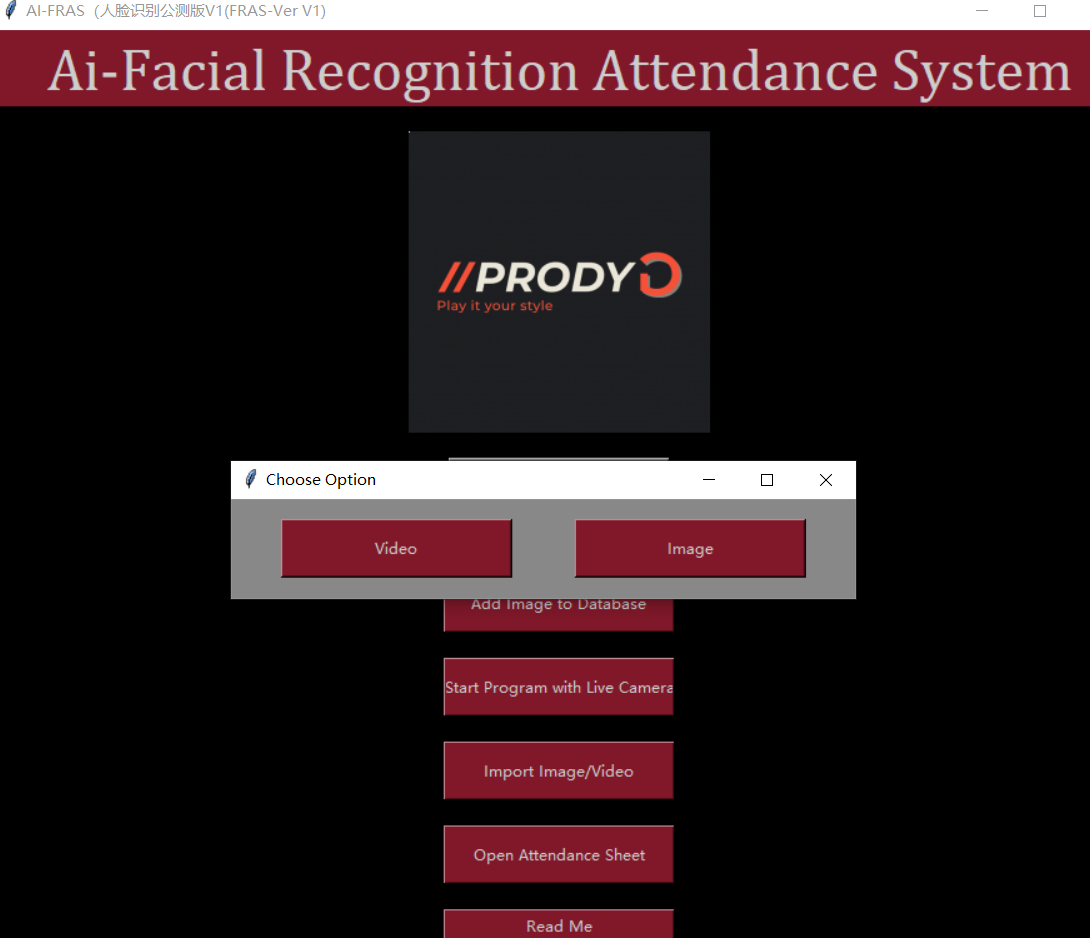
**Figure 20.1 Image capture window pop-out**



**Figure 21.1Alert bot pop out after pressing “start program with live camera”**



**Figure 22.1 Detector started to get working**



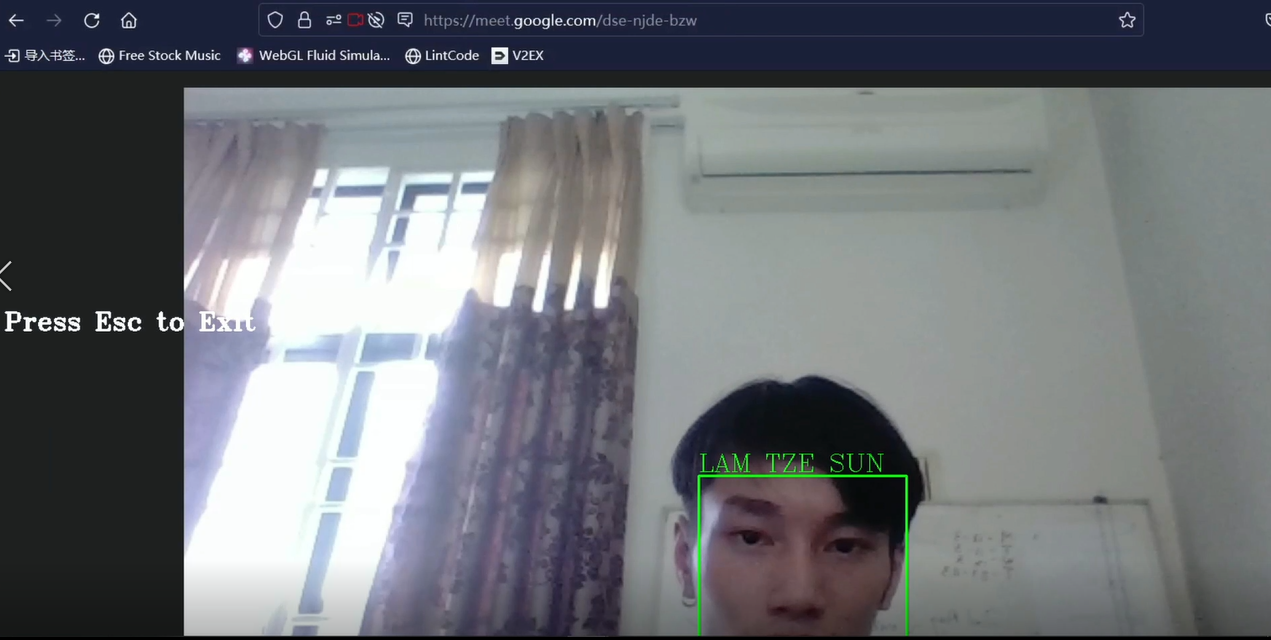
**Figure 23.1 Option for import video or image**



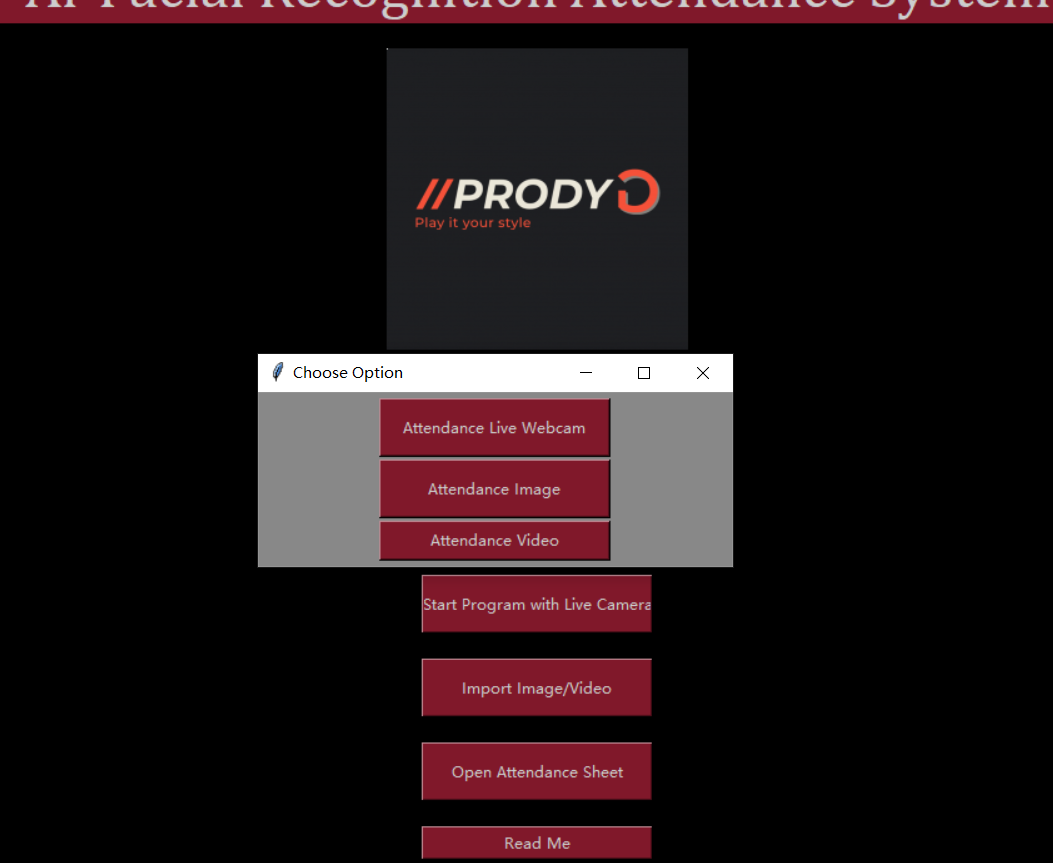
**Figure 24.1 File explorer launched to choose image/video**



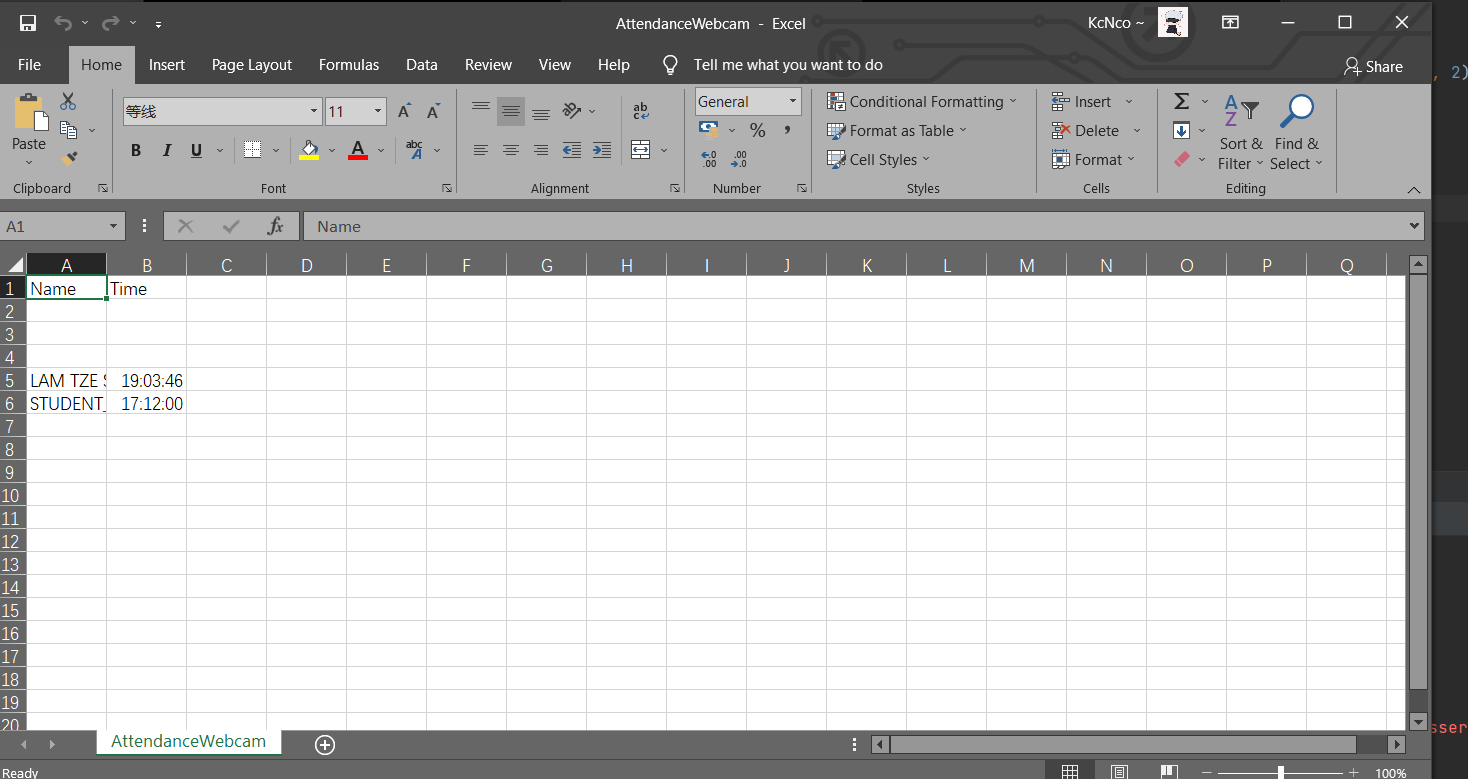
**Figure 25.1 Detector use imported image as input for detecting face id.**



**Figure 26.1 Detector use imported video from google meet as input for detecting face id.**



**Figure 27.1 option for opening attendance report**



**Figure 28.1 Attendance sheet open up in .csv format for review.**



**Figure 29.1 opening readme in visual studio.**

# **CHAPTER 6: TESTING AND EVALUATION**

## 6.1 TEST CASE

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Use case | Test Case Description | Step Detail | Test data | Expected result | Actual Result | Status (pass/fail) | Remark |
| User entry and acceptance | To ensure that the input could be accepted by the system | 1. Enter the name 2. Click add image to database | SCSJ1900870 | Can input and accepted | SCSJ1900870 | Pass |  |
| Storing of information | To identify that the system could store the information | 1. Start the Ai-Fras program 2. Register at entry block | SCSJ1900870 | Input are stored | SCSJ1900870 | Pass |  |
| Webcam and camera configuration | To ensure that the webcam or camera is correctly configured | 1. Start the Ai-Fras program 2. Fill in the entry 3. Student face registration |  | Image capture window should be initialized and camera should be connected. | Image capturing window initialized and camera are working correctly. | Pass |  |
| Face ID registration | Ensure that the system could detect faces, capture and exit after pressing Esc button | 1. Start the Ai-Fras program 2. Fill up entry block 3. Choose add image to database |  | The face should be detected and image are capture then press Esc to exit the image capture session. | Face from the user are recognized and detected. The detecting window will be close after pressing Esc button | Pass |  |
| Image dataset storing | Identify that the dataset folder is available | 1. Start the Ai-Fras program 2. Fill up entry block 3. Choose add image to database |  | Check if the image dataset folder is available or not. If not exist, create a new one. | The image dataset folder is available. | Pass |  |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Use case | Test Case Description | Step Detail | Test data | Expected result | Actual Result | Status (pass/fail) | Remark |
| Recognising live face | Identify that the system could recognising the live faces | 1. Start the Ai-Fras program 2. Start program with live camera | Face ID – SCSJ1900870 | Webcam should connect and started up to detect face input and shows up the registered entry name in a rectangular box around the face. | Webcam started up and the face is detected with rectangular box around the face that has the id of SCSJ1900870 | Pass |  |
| Importing image and video | Ensure that the system could import image and video | 1. Start the Ai-Fras program 2. Import image/video 3. Browse for image and video | Image/video | Image and video that are being imported should be work as input for the detector to detect face id from it. | Live camera is replaced by the image and video that are being imported and system is detecting face id from it. | Pass |  |
| Attendance entry in .csv file | Ensure that the attendance is stored in .csv file | 1. Start the Ai-Fras program 2. Start program with live camera |  | Check that the date and name that user enter are printed at the excel sheets. | Date and name of the user are printed at the .csv file sheet correctly | Pass |  |
| View attendance | Identify that the attendance could be view | 1. Start the Ai-Fras program 2. Open Attendance sheets |  | .Csv file will be open up by Excel for user to view the attendance. | The file can be open up and user can view the attendance. | Pass |  |

**Table 3.1 Test case**

## 6.2 EVALUATION PLAN

Part A – Personal Information

1. Your gender?

* Male
* Female
* Prefer not to say

1. Your age?

* Below 18
* 18 ~ 21 years old
* 22 ~ 30 years old
* 31 ~ 40 years old
* 41 and above

1. Your roles inside the testing phase?

* Student
* Lecturer
* Program tester

Part B – System Evaluation

1. Are you able to use the system?

* Yes
* No

1. Is the system user-friendly?

* Yes
* No

1. Simplicity of the system and feasibility?

* 1: Dull
* 2: Slightly dull
* 3: Neutral
* 4: Slightly Innovative
* 5: Innovative

1. Is the function of system working?

* Yes
* No

1. Are the button clickable?

* Yes
* No

1. Are image being captured and stored?

* Yes
* No

1. Could you import the image/video?

* Yes
* No

1. Are you able to open the attendance sheets generated by the system?

* Yes
* No

1. Are you able to input your name/student id in the entry form?

* Yes
* No

1. Are the message box popped out when you click the “add image to database” button?

* Yes
* No

1. Are you able to start the webcam when selecting “start program with live camera”?

* Yes
* No

Part C – Your feedback

1. If you are the user of the system (whether lecturer or student), do you think that the system will help you during attendance taking?

* Yes
* No
* Maybe

1. If you are the user of the system, are you feeling comfortable and ease when using the system?

* Yes
* No
* Maybe

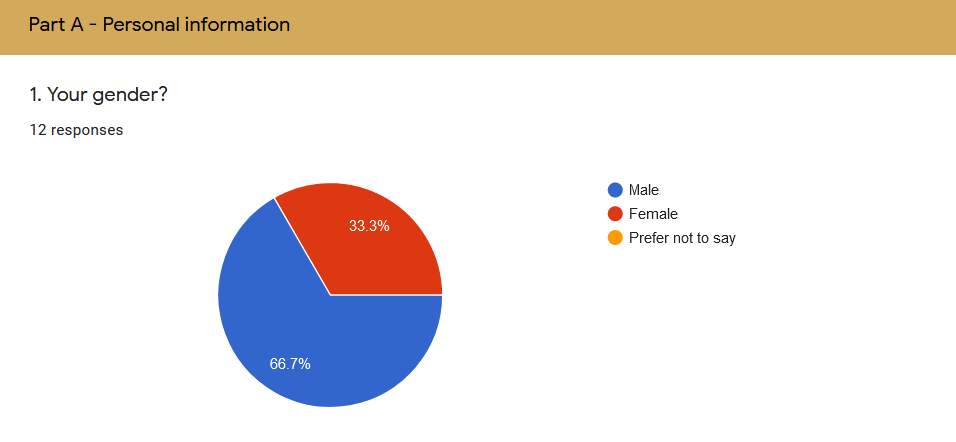
1. Rate the system in your own opinion

1: Not Satisfied

5: Very Satisfied

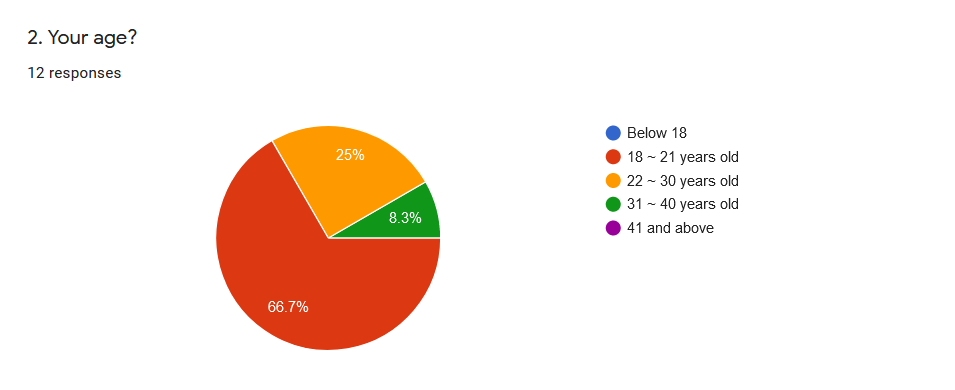
## 6.3 RESULT AND DISCUSSION

For the questionnaire which has been conducted, the participant has been randomly selected from the Discord server which consist of 138 people. There are 12 respondent who agreed to join the questionnaire session through the method of using Pycharm code with me function which have the function like other meeting software but it is an IDE for coding. The software would share the code to the person that has the link that are generated by host. In this case, there are 3 parts for the questionnaire which is 3 questions of Part A for the personal detail, 11 questions of Part B for the system evaluation and also 3 questions of Part C for the feedback.



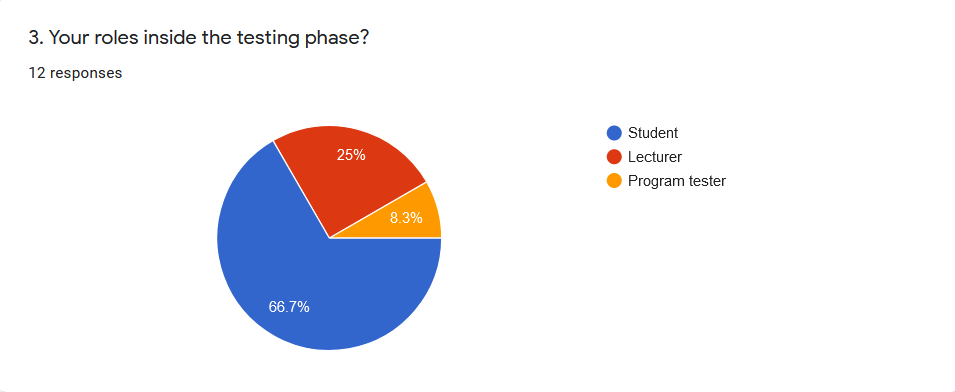
**Figure 30.1 Gender**

Throughout the system evaluation plan, I have found out the fact that there are 66.7% of the respondent are male and only 33.3% of respondent are female.



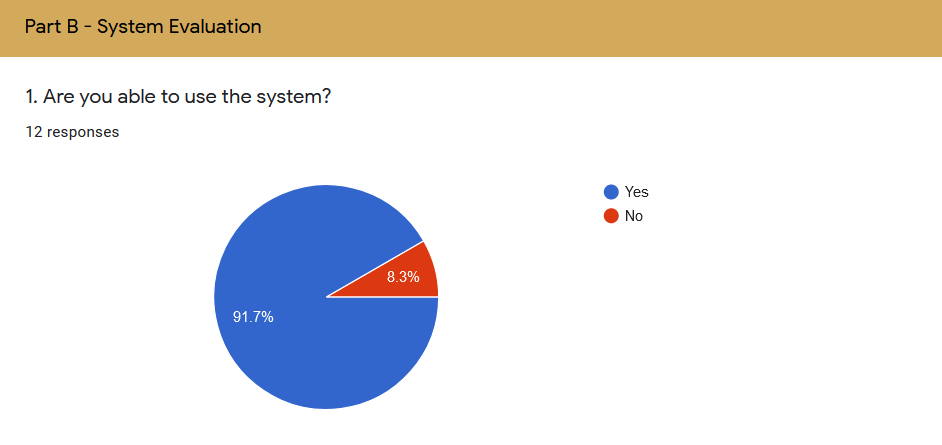
**Figure 31.1 Age**

From the figure 6.2 above, it shows that the majority of respondents is caped from 18 years old to 21 years old which in a total of 66.7%. The rest of the respondent consist of two age category which is 25% of respondent from 22 ~ 30 years old and also 8.3% of respondent from 31 ~ 40 years old.



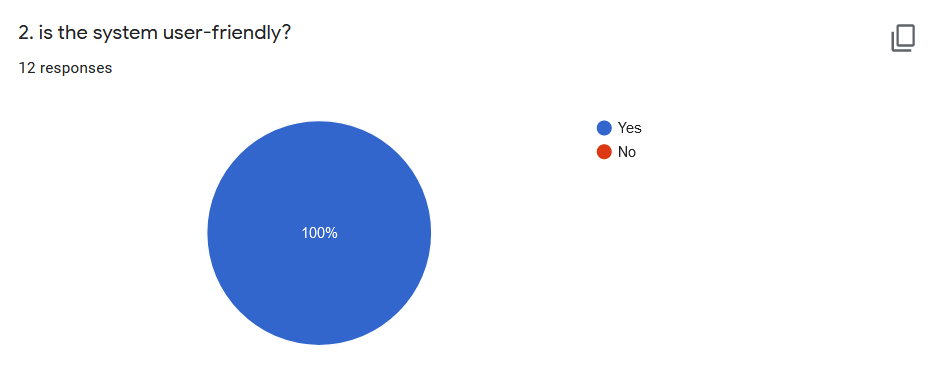
**Figure 32.1 Roles**

From the figure 6.3, the statistic shows that there are a majority of 66.7% of respondent is students. Furthermore, 25% of the respondent are lecturer and 8.3% of respondent is program tester.



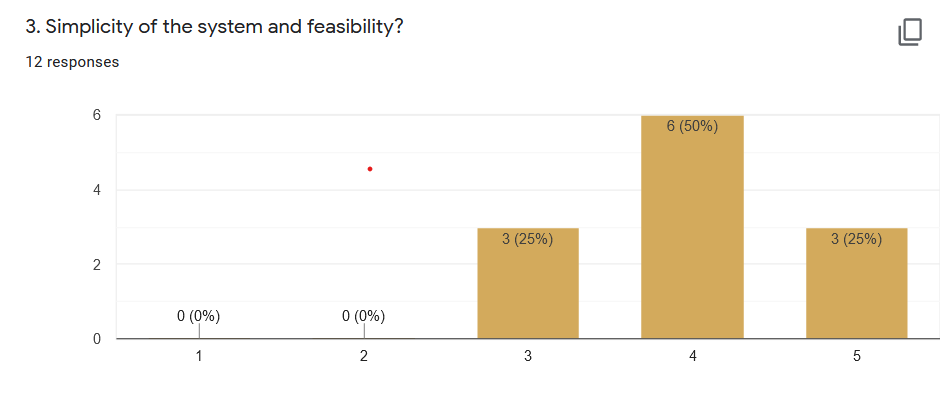
**Figure 33.1 System usability testing.**

From the figure above, we can see that 91.7% of the user could use the system without any error which means that the system is functioning and only 8.3% of the respondent claim that the system could not be use. This may cause by the uninstalled package module using the pip install command from the requirement text file.



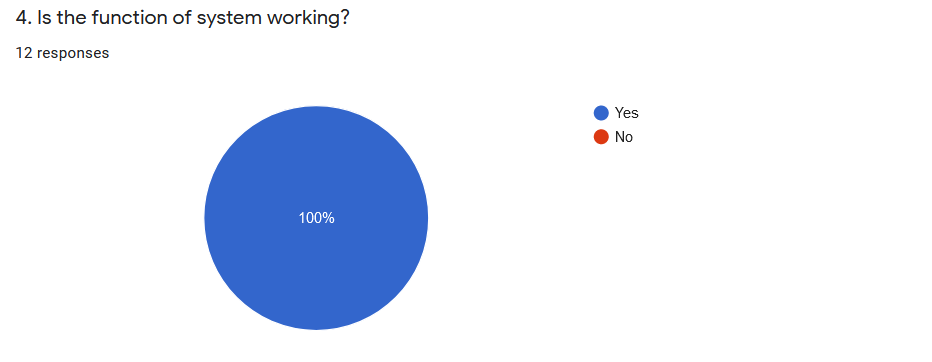
**Figure 34.1 User friendly evaluation**

The figure above shows that the user-friendly acceptance by the respondent. From the pie chart above we can conclude that all of the respondent agrees to said that the system is user-friendly.



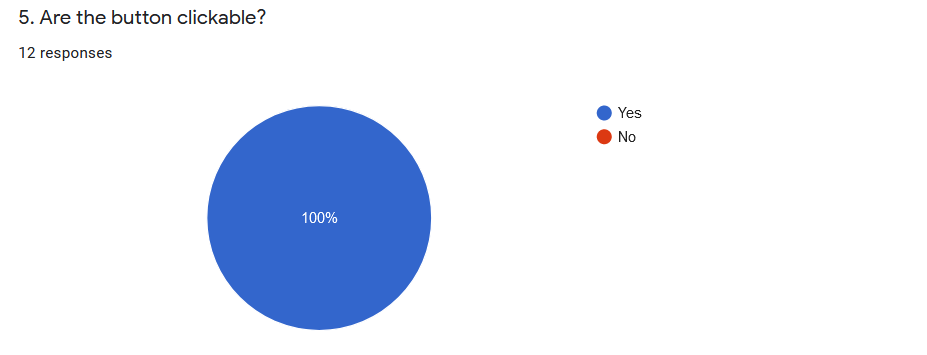
**Figure 35.1 Simplicity and feasibility evaluation**

From the bar chart above show the response of simplicity and system feasibility evaluation. 25% of the respondent remain neutral for the statement and 50% of the respondent agree that the system feasibility is innovative and simple. Besides that, 25% of the respondent think that the system is very innovative.



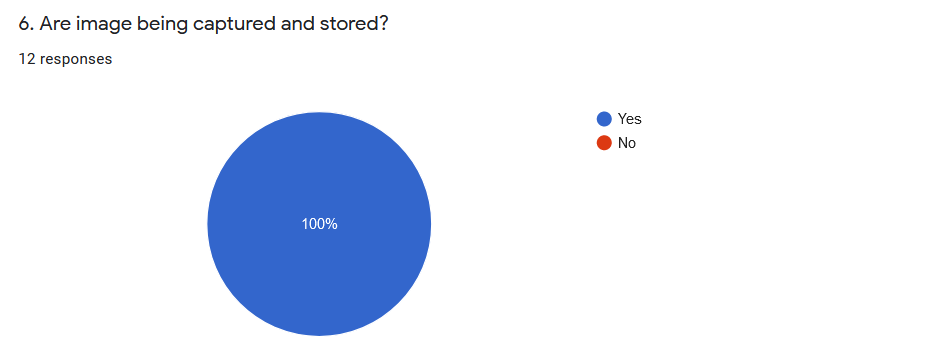
**Figure 36.1 System function evaluation**

From the figure 36.1 above shows that the statistic for the system functioning evaluation by the respondent. As you can see, every respondent claim that the function of the system is working fine on their computer without any error occur.



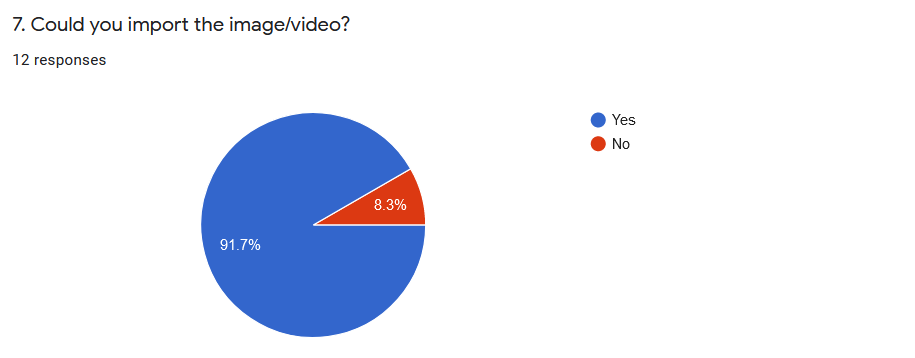
**Figure 37.1 Button function evaluation**

The figure above shows the response for the button function evaluation. Throughout the diagram we can conclude that the button in the Ai-Fras is working without any error.



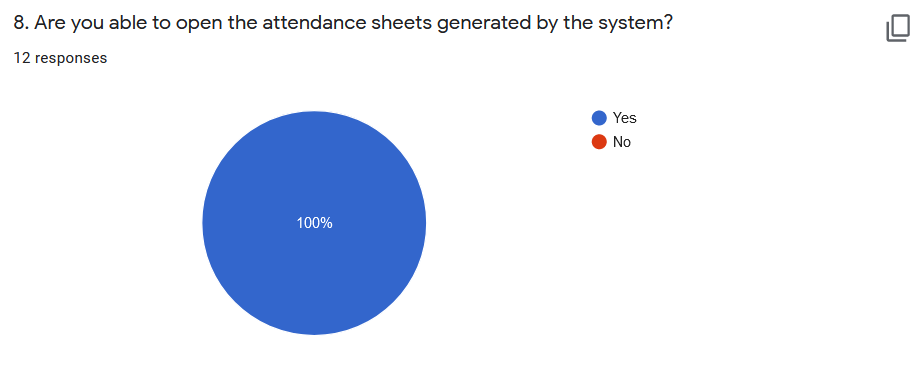
**Figure 38.1 Input acceptance evaluation**

Based on the figure that shows the response on input acceptance evaluation, every respondent claims that their image is captured and stored by the system.



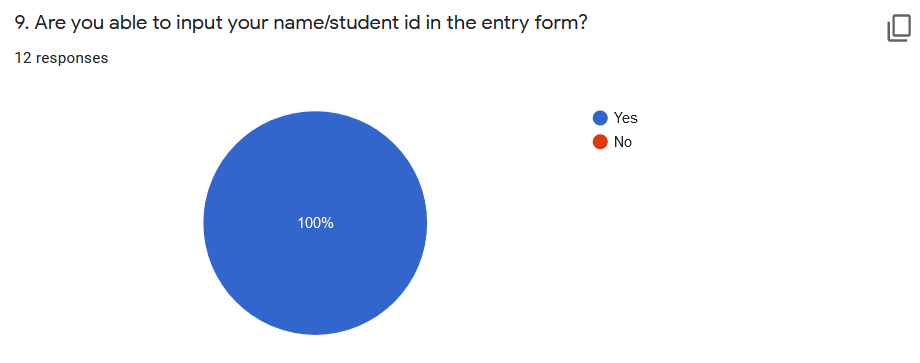
**Figure 39.1 Importing image and video**

From the figure above shows that the importing of image and video. With the statistic above, there are 91.7% of the respondent said that they are able to import the image and video from the file explorer window which will automatically open after choosing the import option and only 8.3% of them claim that they could not import the image and video as the input for the detector to work and detect the face id from the imported media.



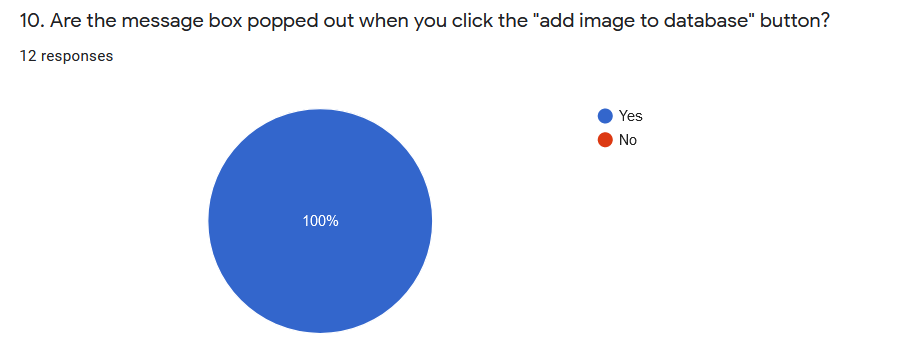
**Figure 40.1 Attendance sheets review evaluation.**

From the figure above shows that the response for the attendance sheets review evaluation which the user could open and review the attendance sheet that are generated by the system. Based on the pie chart, all of the respondent claims that they can open and review the attendance sheet that are generated by the system without any error.



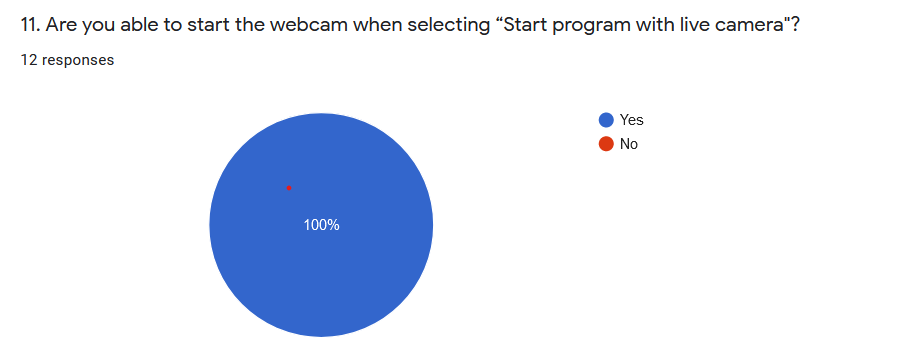
**Figure 41.1 Entry acceptance evaluation.**

From the figure above shows that the response for the entry acceptance evaluation which test that whether the user could input their name or id into the entry block in the system. Based on the pie chart above, all of the respondent agreed that they are able to input their name or id into the entry block without any error.



**Figure 42.1 Alert box evaluation**

According to the alert box evaluation, all of the respondent agreed to the statement that the message box popped out when they clicked the “add image to database” button.



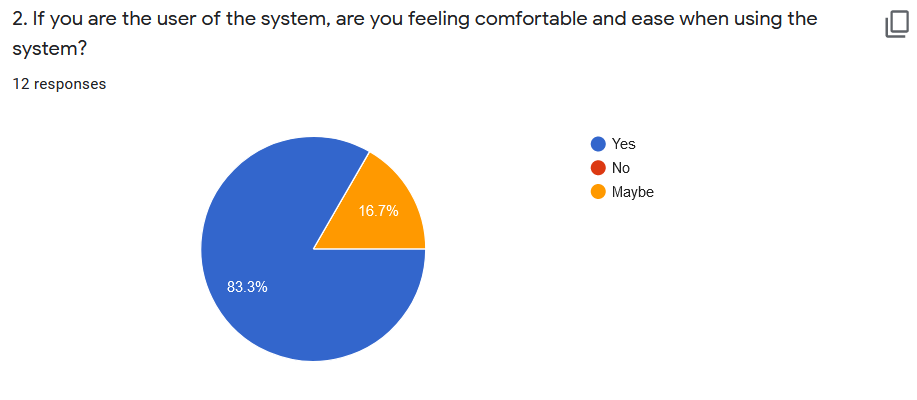
**Figure 43.1 Webcam activation evaluation**

In this evaluation, the user will need to identify that the webcam or camera that are attached to the computer will activate when the system opens the face detection windows. Based on the pie chart, 100% of the respondent are able to start the webcam when selecting the option of “Start program with live camera”.



**Figure 44.1 Efficiency evaluation**

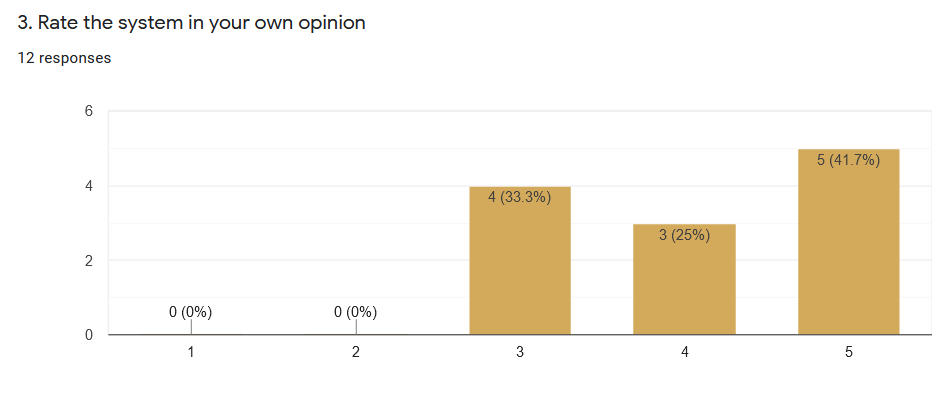
the part c which is the feedback from user, the first question is about the opinion of the user for the efficiency evaluation of the system. Based on the figure 6.15 regarding the efficiency evaluation, there are 75% of the respondent agreed to the statement that the system is efficient for them in assisting attendance taking while only 25% of them remain neutral for the time being.



**Figure 45.1 Ease of access and comfortable evaluation.**

For the ease of access and comfortable evaluation that are shown in the pie chart above, the statistic shows that there are 83.3% of the respondents agreed to the statement where they feel

comfortable and ease while using the system while 16.7% of the respondent remain neutral for the ease of access and comfortable evaluation.



**Figure 46.1 Rating of the system.**

The bar chart above shows the rating of the system from the respondents. Based on the graph, 33.3% of the respondents remain neutral and 25% of the respondent are satisfied with the system where as 41.7% responded as very satisfied.

# **CHAPTER 7: CONCLUSION AND RECOMMENDATIONS**

## 7.1 INTRODUCTION

In daily life, face recognition is important to identifying families, friends or someone we know. We may not understand that numerous procedures have been done to recognise people's faces. Human intelligence enables us to receive information in the process of recognition and to understand it. Through the picture displayed in our eyes, we get information, particularly in retina light. Light is a kind of electromagnetic waves emitted on an object, projected into the perception of man from a source. We really classify shape, size, contour and texture of the item for the purpose of analysing information after visual processing by the human visual system. The information analysed is compared with other object or facial representations that are recognisable in human memory. Indeed, building an automated system is a difficult challenge to have the same facial recognition power as human beings. However, we need a vast recollection of faces, at the universities for example, there are many students of different races and genders, without making errors it is hard to recall every person's face. Computers with nearly endless memory, great processing speed and power in facial recognition systems are utilised to overcome human limitations.

The major goal of this project is to build an automated student attendance system based on facial recognition. In order to improve the performance of this technique, the test picture and trained images are confined to the face recognition and face recognition model. In order to assure no quality variation, test photo and trained image must be taken by the same device and the student must also register for their face id in the same device’s database. Registration may be carried out locally using the user-friendly graphical user interface (GUI).

This facial recognition attendance system was created using the Python programming languages and its vast collection of dependency packages that can be downloaded and import into the IDE. The database was created using Microsoft Access database. After the development stage, testing phase took part where all the test case are made. In order to verify the system is functioning as intended, test cases and its result are well documented for the reference of future improvement. In addition of evaluation, the entire project was designed from the needs to a full system. The created system has fulfilled its objectives and outcomes. However, certain problems have been addressed and handled throughout implementation.

Throughout this final year project, it was an excellent opportunity to carry out the facial recognition technology software development. Other than that, Python is a powerful language of programming and to me it is the easiest language I've seen so far, well I don't have much language expertise, actually I have had contact just before with C++ and JS. I've been told JS is the simplest beginner language. After learning Python from the E-book that has been downloaded from the online source and also other learning institution, I can certainly say that Python is the most awesome programming languages. Back to the main track, I could said that the final year project was a great opportunity for me to actually try to develop a system with the new programming language that I learn and the algorithm knowledge that I gain when study so far in Diploma in Information Technology and turn theory that I have learn into reality. In the project lifecycle, I learned on how to manage the resource, allocating schedule for the project, focusing on the scope, trying hard to achieve the aim and objective that has been identify during the early stage of preparation.

## 7.2 PROBLEM ENCOUNTERED

Several difficulties have modest impediments to progress during the development of the project. Initially, a GUI has been built to help the user store his portrait for the development of face databases. To accomplish so, the GUI is being downloaded from an external library known as guizero. However, this library does not have many restrictions as it supports the display of other image file type except the image file type.gif. Images cannot thus be shown via the window. In addition, the layout of the GUI window contains numerous constraints that make the developed interface unwanted. In the later period, the use of guizero is therefore abandoned and Tkinter is subsequently substituted.

Apart from that, there are insufficient faces in the generated database before the face recognition model could be tested as only a limited number of volunteers are agree to help building the face database during the Covid-19 pandemic where government announced the movement control order. However, by using the pre-prepared picture from the online source face database which the facial data has been standardised in to greyscale colour scheme. In this case, it is extremely handy for the testing procedure to take place.

Also, the pre-processing of the recorded image is quite difficult. Thankfully, searching the Internet to find recommended remedies can address these difficulties. In conclusion, it may be quite straightforward to create a facial recognition system if there is enough prior knowledge of the workings of the process, as the majority of a difficult algorithm is supplied in the library itself that just needs to be understood to incorporate it into the development system.

## 7.3 SYSTEM LIMITATION

The system that has been developed has several limitations comparing with the existing system. Firstly, the main feature are aim to use by the admin panel which are lecturer and only admin panel was developed. Meaning that the student can only register themselves and not able to change their detail after the registration without the help of lecturer. After that, the assessment of the system's facial recognition portion yielded findings that were not as predicted. The assessment indicated that the face recognition model can obtain an identification rate of around 60% depending on the image resolution. There is a significant possibility of a system failure due to a poor image resolution. In addition, the user must ensure that the photograph is at the correct resolution described in this report in order to accomplish this outcome. Some individuals don't like the directions to follow. This might result in poor picture quality and poor performance of the system. Since the system is aim for the use of admin panel there are no security measurement technique implement on it. So that the data is not secured. For the final system limitation, the developed facial recognition system does not have the account creation function which would record a more detail information for the user when they registering into the system.

|  |  |  |  |
| --- | --- | --- | --- |
| Feature of system | RFID Attendance | Fingerprint Attendance | Ai-Fras |
| Contactless | Yes | No | Yes |
| Privacy | Yes | Yes | No |
| Work with glasses/goggles | Yes | Yes | Yes |
| Stability | Low | High | Yes |
| Speed | Excellent | Good | Yes |
| Accuracy | Moderate | High | Yes |
| Work with glove | Yes | No | Yes |
| User acceptance | Moderate | Moderate | High |
| Long term stability | Moderate | High | High |
| Security level | Low | High | High |
| Error incidence | Loss of access card | Dryness, dirt, age | Low resolution, size of images. |

**Table 4.1 Comparison of existing system and my developed system**

## 7.4 FUTURE WORK AND IMPROVEMENT

A further research on a project like such is necessary. The approaches used can be coupled with others in order to get excellent outcomes. According on the literature study, many approaches were applied in the past. There are several improvements that could be done in the future and make it as improvement. Some of it could be made into new features of the system:

1. Security Login: For security considerations, a login mechanism on the system would be developed.
2. Standalone Setup: In future improvement, the standalone version could be done using the Pyinstaller and also improve the GUI by using the PyQT designer that works like Visual Studio which uses drag and drop to create an interface.
3. Automated Absent Email Function: For convenient, the system could integrate the automated mailing function that will email the absent student that are registered into the system by the Yagmail module, a smtp/Gmail based client that could send main to the targeted audience.
4. Institutional database integration: integrating the database of institute could solve the problem of capturing image that is low resolution. Since the picture of student id from the institute database is clear and the resolution is excellent for the facial recognition model to detect the face id of student. It could be integrated with the system to ensure the precision of the detection.

# **REFERENCES**

ANSARI, A. H. (2021). FACIAL RECOGNITION SYSTEM.

Baggio, D. L. (2012). *Mastering OpenCV with practical computer vision projects*. Packt Publishing Ltd.

Bhatti, K. L., Mughal, L., Khuhawar, F. Y., & Memon, S. A. (2018). Smart Attendance Management System Using Face Recognition. *EAI Endorsed Transactions on Creative Technologies*, *5*(17).

Culjak, I., Abram, D., Pribanic, T., Dzapo, H., & Cifrek, M. (2012, May). A brief introduction to OpenCV. In *2012 proceedings of the 35th international convention MIPRO* (pp. 1725-1730). IEEE.

Dawson-Howe, K. (2014). *A practical introduction to computer vision with opencv*. John Wiley & Sons.

Dirin, A., Delbiaggio, N., & Kauttonen, J. (2020). Comparisons of Facial Recognition Algorithms Through a Case Study Application.

Fong, P. K., & Sien, V. Y. (2021). An Investigation on the Effectiveness of OpenCV and OpenFace Libraries for Facial Recognition Application. In *Advances in Computer, Communication and Computational Sciences* (pp. 919-927). Springer, Singapore.

Ghalib Al-Muhaidhri, 2019. Smart Attendance System using Face Recognition. *International Journal of Engineering Research and*, V8(12).

Gomes, C., Chanchal, S., Desai, T., & Jadhav, D. (2020). Class Attendance Management System using Facial Recognition. In *ITM Web of Conferences* (Vol. 32, p. 02001). EDP Sciences.

Howse, J. (2013). *OpenCV computer vision with python*. Birmingham: Packt Publishing.

*International Journal of Innovative Technology and Exploring Engineering*, 2020. Smart Attendance Notification System using SMTP with Face Recognition. 9(5), pp.337-342.

*International Journal of Innovative Technology and Exploring Engineering*, 2019. Attendance Management System using Face Recognition. 8(12), pp.1377-1381.

*International Journal of Recent Technology and Engineering*, 2020. Student Smart Attendance Through Face Recognition using Machine Learning Algorithm. 9(1), pp.2348-2352.

Jawed, S., 2019. Face Recognition for Smart Classroom Attendance Management System using Computer Vision. *INTERNATIONAL JOURNAL OF RECENT TRENDS IN ENGINEERING & RESEARCH*, 05(07), pp.51-55.

Joshi, P., Escrivá, D. M., & Godoy, V. (2016). *OpenCV By Example*. Packt Publishing Ltd.

Kanth, P. and Biswal, S., 2019. Attendance Marking System Using Face Recognition. *Indian Journal of Science and Technology*, 12(48), pp.1-3.

Khurana, L., Chauhan, A., & Singh, P. (2020, January). Comparative Analysis of OpenCV Recognisers for Face Recognition. In *2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence)* (pp. 485-490). IEEE.

Laganiere, R. (2014). *OpenCV Computer Vision Application Programming Cookbook Second Edition*. Packt Publishing Ltd.

Pulli, K., Baksheev, A., Kornyakov, K., & Eruhimov, V. (2012). Real-time computer vision with OpenCV. *Communications of the ACM*, *55*(6), 61-69.

R.S., D., 2020. Attendance Authentication System Using Face Recognition. *Journal of Advanced Research in Dynamical and Control Systems*, 12(SP4), pp.1235-1248.

Selvi, K. S., Chitrakala, P., & Jenitha, A. A. (2014). Face recognition based attendance marking system. *International Journal of Computer Science and Mobile Computing*, *3*(02), 337-34

Sutabri, T., Pamungkur, P., Kurniawan, A. and Saragih, R., 2019. Automatic Attendance System for University Student Using Face Recognition Based on Deep Learning. *International Journal of Machine Learning and Computing*, 9(5), pp.668-674.

Suwarno, S., & Kevin, K. (2020). Analysis of face recognition algorithm: Dlib and opencv. *JOURNAL OF INFORMATICS AND TELECOMMUNICATION ENGINEERING*, *4*(1), 173-184.

Villán, A. F. (2019). *Mastering OpenCV 4 with Python: A practical guide covering topics from image processing, augmented reality to deep learning with OpenCV 4 and Python 3.7*. Packt Publishing Ltd.

Wagner, P. (2012). Face recognition with python. *Tersedia dalam: www. bytefish. de [diakses pada 16 Februari 2015]*.

Yadav, R., Chauhan, S., Meenu, M. and Gupta, S., 2020. FACE RECOGNITION BASED ON ATTENDANCE MANAGEMENT SYSTEM. *International Journal of Innovative Research in Computer Science & Technology*, 8(3).

刘方义. (2018). 基于 Python 的人脸识别算法分析. *收藏*, *21*.

张枝令. (2018). Python 实现基于深度学习的人脸识别. *电子商务*, *5*, 47-48.

徐浩浩. (2019). 基于 Python 深度学习的人脸识别方法探究. *收藏*, *17*.

王欣然. (2020). 基于 Python 的人脸识别技术的应用. *电脑知识与技术*.