Middle East Technical University Northern Cyprus Campus
Computer Engineering Program

CNG491 Computer Engineering Design I

Alpsen Zeybek

Abdullah Küçük

Sabawun Afzal Khattak

Onur Can Güven

Student Attendance System using Facial Recognition

<u>Supervised by</u>

Meryem Erbilek & Enver Ever

First Iteration

1. Introduction

1.1. *Motivation*

Nowadays It has become difficult for people to have close contact with each other due to the Covid – 19 Pandemic. As the world is ravaged by the pandemic, the development of systems to reduce its spread is getting more and more important every day. One of the important places for virus to spread is unfortunately the universities and schools. As the closure of the universities as part of the quarantine makes learning inefficient and difficult, solutions must be found to reduce the risk of spread.

Classrooms are the main place where the virus can easily spread between professors and students. One of the measures taken was to shorten the lesson times, which led to the emergence of new problems such as the inability to finish course topics in time. One of the ways to increase the time efficiency is to automate the attendance recording system by utilizing a face recognition, which is motivation of this project.

1.2. *Aim & Objectives*

We aim to implement automated attendance system using face recognition.

1.2.1 - Recognize the faces from a live video feed

1.2.2 - Process the class and the section information using time and date.

1.2.3 - Recognize and identify the face of the students to take attendance.

1.3. Methodology

How to detect a face from a photograph or a video feed?

Face detection can be done in a number of ways. Currently we are researching “Multi-Task Cascaded Convolutional Neural Network”. This method was introduced in an IEEE paper titled “Joint Face Detection & Alignment Using Multitask Cascaded Convolutional Networks”. A brief description of what MTCCNN is can be understood from the following: Pipeline of the cascaded framework includes three-stage multi-task deep convolutional networks. Firstly, candidate windows are produced through a fast Proposal Network (P-Net). After that, we refine these candidates in the next stage through a Refinement Network (R-Net). In the third stage, The Output Network (O-Net) produces final bounding box and facial landmarks position.

What are the factors to consider when recognizing a face?

Face direction. Face recognition works best when the face is pointed at the camera and both eyes are visible, but the face can be recognized (sideways) in the profile. size. The system is less likely to detect smaller faces than larger ones, especially for low resolution images and images affected by compression and other artifacts. Media quality. The quality of the image or video sent to can have a significant impact on face detection performance. The face should not be sharp and blurry. Ideally, the image or video should be evenly illuminated, as dark shadows and bright highlights on the face reduce accuracy. The image or video should be immune to significant compression artifacts that may affect some formats (such as highly compressed JPEG images). Facial features. Face detectors can handle a relatively wide range of facial expressions, but neutral faces are always detected with the highest possible reliability. Especially abnormal facial expressions can cause face recognition to fail. In addition, glasses and large amounts of facial hair make it difficult to recognize the face. Glasses with dark lenses can fail to detect.

How to take attendance according to the recognized faces?

The attendance will be taken, after facial recognition has successfully occurred. Once the system recognizes the students in the image/video against the database, an excel file will be generated with the ones recognized marked as present and the others marked as absent. This excel file can then be linked with odtuclass attendance system, but that is out of the domain of this project.

According to our limited research KNN usually has an accuracy of 98%, while CNN has shown to hit the 96% mark, followed by SVMs at 95%. We will hopefully be targeting any accuracy above 98%.

2. Requirements

2.1. *Stakeholders*

2.1.1 - Students

2.1.2 - Professors

2.1.3 - IT Administrators

2.2. *Functional System Requirements*

2.2.1 - Takes the attendance and records in an excel file. The reason that excel file format is used is because that format is used in universities a lot, including ours. Also the fact that linking excel with a database is more efficient.

2.2.2 - Supports recognizing students face with glasses and accessories. This is because we won't have people dress as in they are part of the control groups, so there would be people with no accessories as well as excess accessories, the system should be ready for that.

2.2.3 - Supports printing the generated excel file. This makes the job of the user easier so they wouldn't have to open excel select print options each time they get a report.

2.2.4 - Tracks the time to find the correct lecture and the students registered to that section. This is easier than making professors select the current lecture, classroom and section. Automating this process would make the system more reliable.

2.3. *Non-functional Requirements*

2.3.1 - Camera angle and quality should be sufficient.

2.3.2 - Recording the attendance shouldn't take more than 10 seconds.

2.3.3 - It should store at least 5 different photos of each student.

2.3.4 - All the students' whom the system will be used on should be informed prior to the use of the software and should be asked for their consent as this is an ethical issue.

2.3.5 - Privacy is an issue so, all the users and people affected by the system should act accordingly as to keep the information secure.

2.3.6 - There should be no copies of the live video feeds that the system uses due to privacy issues.

2.4. *Domain Requirements*

2.4.1 - The live video feed will not be recorded, and cameras shut down after class hours.

2.4.2 - The area that the cameras will be placed should be secure enough as in physical security, we wouldn't want anything happening to them such as vandalism.

2.4.3 – The placement area of the cameras should also be protected from weather events such as wind, rain, lightnings as that would probably lead to camera getting broken.

2.4.4 – Camera lenses should be cleaned weekly, considering the weather in Cyprus is dusty.

2.4.5 – The cameras should be elevated from the floor by at least 2 meters to avoid probable pranks from students.

2.4.6 – The lecture and the students should be in a classroom in order for the system to work properly.

2.4.7 – The classroom should have good-enough lighting for good footage.

2.5. *Assumptions and Justifications*

2.5.1 - Assuming the dedicated class room is always used during the designated timeslots.

2.5.2 - Assuming the students' faces are facing the camera while the attendance recording is ongoing.

2.5.3 - Assuming the program always has enough storage for storing students' photos.

2.5.4 – Assuming the cameras are working and intact.

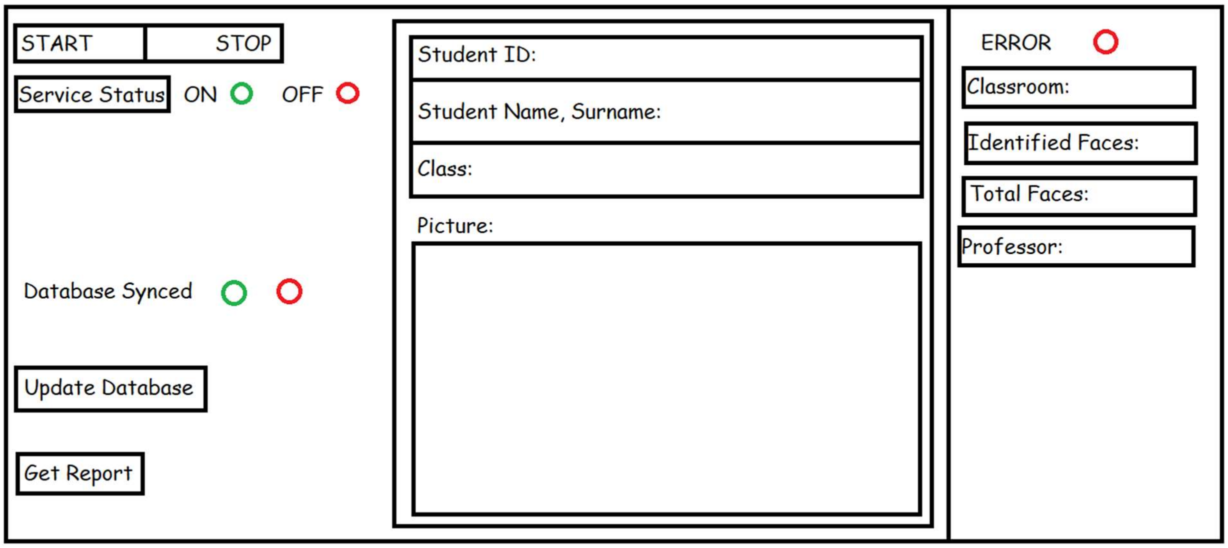
2.5.5 – Assuming the database is working properly and connected.

2.5.6 – Assuming the students are not out of the camera's view angle.

2.5.7 – Assuming that the students aren't wearing any masks to cover their faces.

3. Graphical User Interface

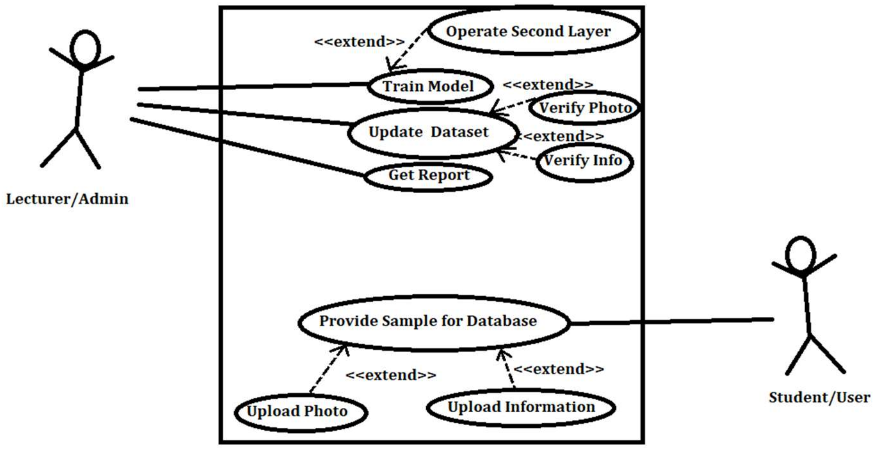
Since our project is not necessarily need a complicated GUI, we are just providing a simple example.



- "START/STOP" buttons will be responsible from the service's start/stop operations.
- The service status, whether the service is currently online or not, is shown below that.
- "Update Database" button will be used anytime we want a new person, a student in our case, to be recognized. The user will insert the new information via that button.
- "Get Report" button is used to take an excel file output.
- The middle section of the interface shows the student that is recognized alongside with their picture, the class they're in and their name, surname.
- The right-hand section of the interface is used for determining errors, that is if the identified faces are less than the number of faces in the class. Giving out necessary information afterwar

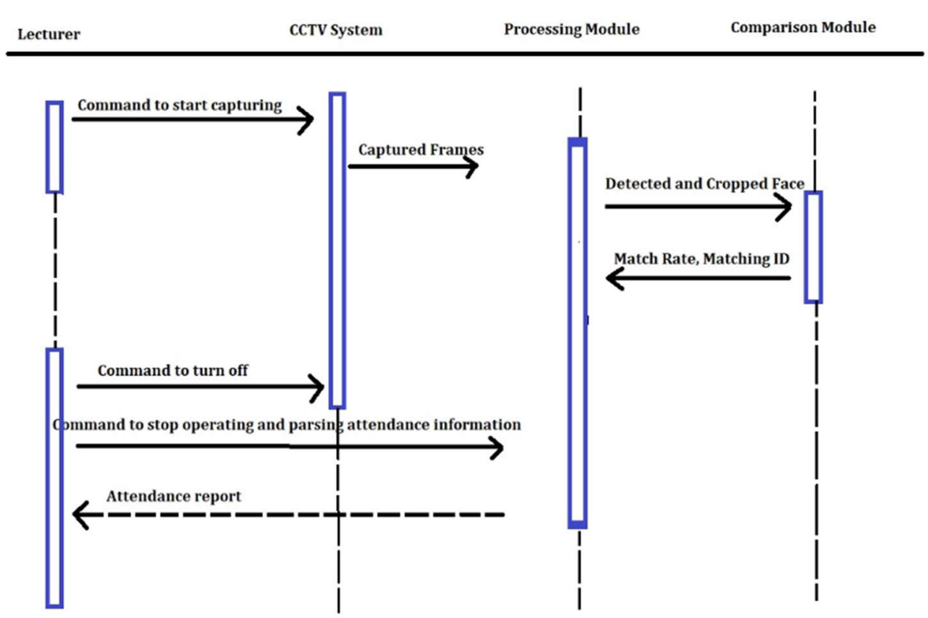
4. System Modelling

4.1. *Structured Use Case Diagram*



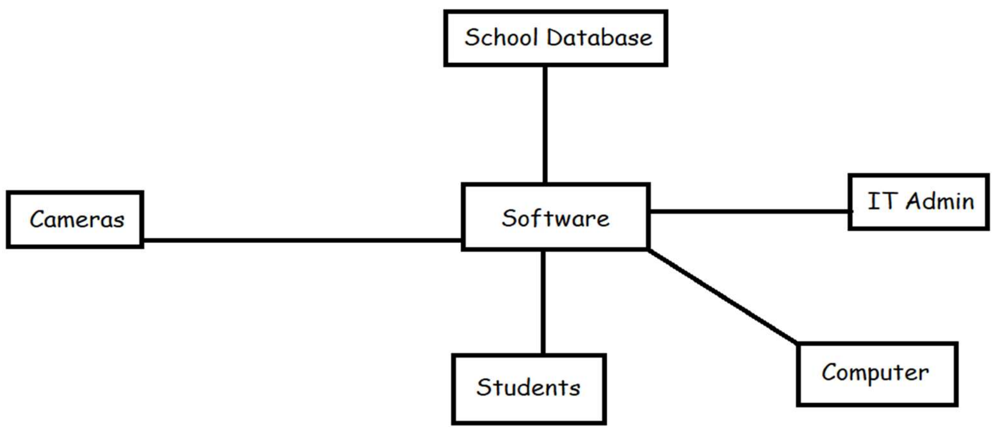
As you can see from the structure use case diagram, there is no direct interaction between the users. The student is only required for the database, while the admin is responsible for dealing with the software training and report.

4.2. *Sequence Diagrams of the Major Use Cases*



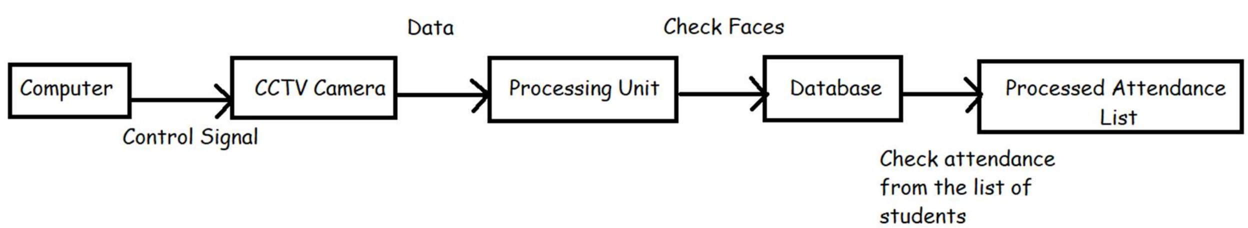
As seen in the chart, the Lecturer object starts the live CCTV System and this system sends captured images, video stream in our case, to the brain, Processing Module. In this module, we process the input frame by frame and detect faces, crop these faces, and send it to the pre-trained Comparison Module to see if there is an accurate match between this face and our dataset. Comparison Module warns the Processing Module should there is a match. The Processing Module processes this information and stores it temporarily to save it at the end of the process. To finalize the process, the Lecturer turns off the CCTV System and also sends a warning to Processing Module to stop processing and start saving the attendance information. Finally, the Processing Module sends the attendance report as an answer to the Lecturer.

4.3. *Context Model*



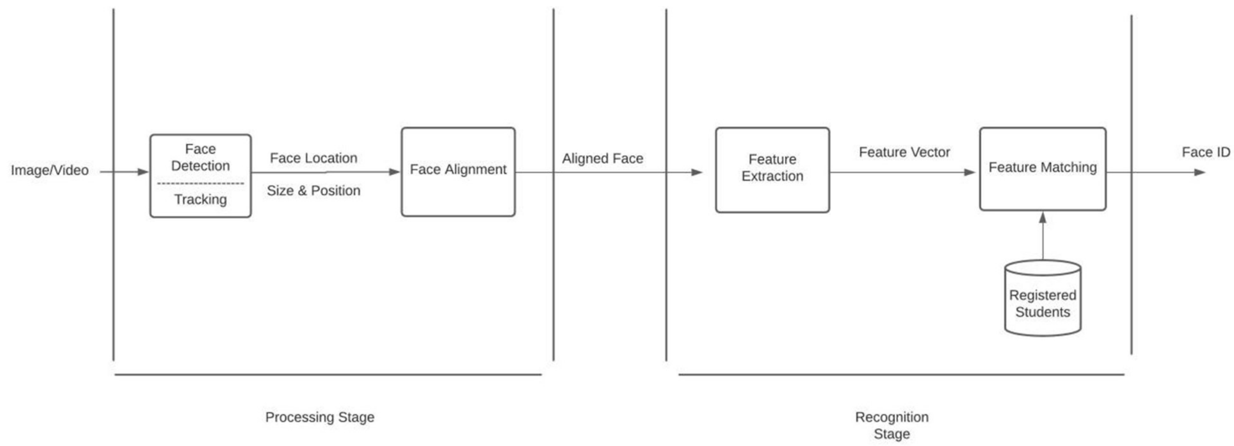
This is the context model for our system. As you can see, the software requires the use of cameras, the student data base, students, computer and administrator.

4.4. *Architectural Model*



The architectural model shows how the system will perform. The camera will record the students, send the data to the processing unit where the processing will happen, which will be then checked against the database. After this, an attendance list will be generated.

4.5. *Process Model*



4.5.1 Detection / Tracking:

This part of the preprocessing phase is responsible for identifying and tracking faces in specific image or video files. When this process is complete, you can be sure that the input you entered contains a face and is ready for further processing. The tracking phase also allows you to track specific parts, special features, or facial expressions as needed.

4.5.2 Alignment:

Face detection problems are exacerbated by the fact that certain photo or video faces do not follow the guidelines. Face recognition issues are even more difficult because people can zoom in and out, peek from behind obstacles, and more. The placement of the face is important here. This will give you an idea of where the facial lines are in a particular image / video and what contours the facial features have.

4.5.3 Feature extraction:

As the name implies, in this phase of the process individual facial features such as eyes, nose, chin, and lips are used by the next stage of the algorithm. Again, features can be geometric or more complicated. At this point, the computer is collecting enough hard data to clearly distinguish the face.

4.5.4 Feature matching:

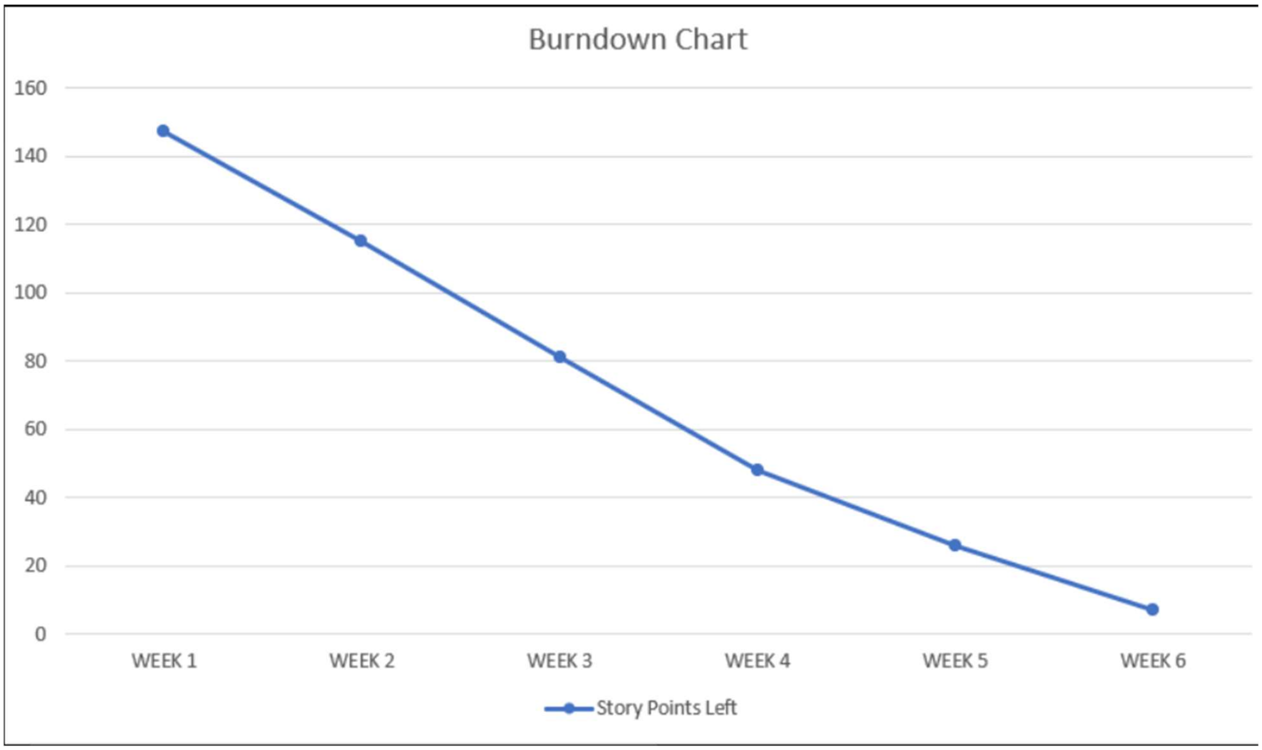
In this phase, the input received from the feature extraction is compared to a particular database to infer the identity of the person.

5. Agile Development with Scrum

5.1. *Sprint Backlog*

	Member	Estimated Hours					
		W1	W2	W3	W4	W5	W6
Reading Related Work	All	2	2	2	1	0	0
Image Processing Alg. Research	All	3	3	3	2	2	2
Datasets Research	Abdullah	3	3	2	2	2	0
Face Detection Met. Research	Alpsen	3	3	3	2	2	2
Face Recognition Met. Research	Sabawun	4	4	4	2	2	0
Introduction	Onur	2	2	2	2	1	1
Related Work	Onur	2	2	2	2	2	0
Requirements	Onur	0	2	2	2	2	0
System Modeling	Alpsen	5	5	5	2	2	2
Graphical User Interface	Sabawun	2	2	2	1	1	0
Agile Development with Scrum	Onur	2	2	2	2	1	0
Project Estimation	Abdullah	4	4	4	2	2	0

5.2. *Sprint Burndown Chart*



5.3. *Sprint Review*

We started by reading and researching related topics. The purpose of this is to understand the concept of face recognition and the algorithms that is related to it.

In week two, we started by talking about the First Iteration of Report. We started to talk about it before it is released to give us more time to research. Then we made work sharing between us. Everyone started searching and preparing their parts so that we can discuss it after First Iteration of Report is released.

In week four, we started to arrange our work for group discussion. Then we checked our work together and made necessary corrections and additions. After that, we prepared our report and submitted it for the feedback session on week 5.

To conclude, we didn't face any problems while doing our own researches and preparations. We encountered some schedule crashes, but other than that it was going as planned. We finished our first iteration before the time. However, we are not fully satisfied yet, so waiting feedback session to improve our report.

5.4. *Sprint Retrospective*

Because we were learning and executing the assignment at the same time, several of our tasks took longer than they should have. It would take less time if we know the subjects. In certain cases, we might want to start with an outline. Including the communication issues we had, we estimate a ten -hour loss in overall time.

We have to focus more on learning details about the face recognition algorithms to choose the most efficient one for this implementation. It is also necessary to train it well so that our implementation will become successful at taking the attendance effectively.

As the concept requires high-level technical details, we should have spared more time to do the research and digest the information. Moreover, every team member should have included in that part more in order to reduce the time wasted spreading the word and having a debate on it later on. Every group member also should have taken a major part in the reporting process since it is designed solely for a group of 4-5, it is almost impossible to be completed by a lesser number due to its workhour weight. Finally, we should have acted more proactively at the beginning of the project. Instead of giving reactions, we could have directed the project's aim and details more instead of expecting our instructor to direct us.

6. Project Estimation

6.1 *UTFP*

Inputs

(H) Video feed from camera for face detection & recognition

(H) Schedules of the lectures

Outputs

(H) Attendance Information

(H) Student Info

(L) Error Message

(L) Unwanted Student

Inquiries

(H) Face recognition operation

(H) Attendance marking operation

Logical Internal Files

(L) Student Information {Name, Surname, Student ID, Unique ID}

(H) Student Photos

(H) Trained model of the class

External Interface File

(H) .csv file

(H) KNN Algorithm

Inputs Total = 2 x 6(H) = 12

Outputs Total = 2 x 7(H) + 2 x 4(L) = 22

Inquiries Total = 2 x 6(H) = 12

Logical Internal Files = 2 x 15(H) + 1 x 7(L) = 37

External Interface Files = 2 x 10(H) = 20

Total UFTP = 103

6.2 VAF

General System Characteristics	Degree of Influence
Data communications	3
Distributed data Processing	3
Performance	2
Heavily used configuration	5
Transaction rates	1
On line data entry	3
End-user efficiency	4
Online updates	3
Complex processing	5
Reusability	5
Installation ease	0
Operational ease	3
Multiple sites	0
Facilitate change	4

Total TDI = 41

VAF = TDI * 0.01 + 0.65 = 1.06

6.3 *ATFP*

ATFP = UTFP * VAF = 103 * 1.06 = 109.18

6.4 *Estimated LOC*

LOC = ATFP * Language Unit Size = 109.18 x 20 = 2183.6 (Using Python)

6.5 *COCOMO*

Development Mode is Organic.

a = 2.4, b= 1.05, c = 0.38

KDSI = ATFP * Language Unit Size / 1000 = 2.1836 *We are using Python, which is 20

MM = a * (KDSI ^ b) = 5.44

TDEV = 2.5 * (5.44) ^ c = 4.76

6.6 *Jones's First Order Schedule Estimation*

Calculations for the average estimation:

Class Exponent = 0.45 (System Software)

Estimate Effort (Man-Month) = ((ATFP) ^ (3*Class Exponent)/27 = 20.90

Rough Schedule Estimate = (ATFP)^class exponent = 8.26

Schedule in months = 3*man-months^1/3 = 5.28 months

6.7 Shortest Possible Schedules

	Systems Products		Business Products		Shrink-Wrap Products	
System Size (lines of code)	Schedule (months)	Effort (man-months)	Schedule (months)	Effort (man-months)	Schedule (months)	Effort (man-months)
10,000	8	24	4.9	5	5.9	8

6.8 Estimated Team Size

Estimated Team Size = estimated schedule / estimated effort = MM / TDEV = 1.14

Estimated Team Size = 1.14

7. Concept

With the coronavirus pandemic's unexpected rise, priorities we need to consider have slightly changed in the classroom environment. We need to remove physical interactions and passings touched/used objects from one to another to prevent contamination via the item. Passing attendance-taking paper is also risky, considering the nature of it. To overcome this problem and eliminate a physical paper requirement for taking attendance, we wanted to develop a system that can take attendance over a CCTV camera system. We plan to use face detection and recognition algorithms besides multiple images of students to build this system and do a stand-alone project.