

Computer Vision Based Surveillance System

BY

Sarwar Maruf Jebon

ID: 171-15-8988

AND

Md. Rakibul Hasan

ID: 181-15-10535

This Report Presented in Partial Fulfillment of the Requirements for the
Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

Md. Tarek Habib

Assistant Professor

Department of CSE

Daffodil International University

Co-Supervised By

Md. Jueal Mia

Sr. Lecturer

Department of CSE

Daffodil International University



DAFFODIL INTERNATIONAL UNIVERSITY

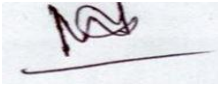
DHAKA, BANGLADESH

JANUARY 2022

APPROVAL

This Project titled “**Computer Vision Based Surveillance System**”, submitted by Sarwar Maruf Jebon, ID No: 171-15-8988, Md. Rakibul Hasan, ID No: 181-15-10535 to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 02/01/2022.

BOARD OF EXAMINERS

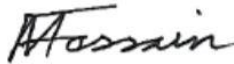


Chairman

Dr. Md. Ismail Jabiullah

Professor

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

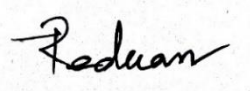


Internal Examiner

Dr. Md. Fokhray Hossain

Professor

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University



Internal Examiner

Md. Reduanul Haque

Assistant Professor

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University



External Examiner

Dr. Mohammad Shorif Uddin

Professor

Department of Computer Science and Engineering
Jahangirnagar University

DECLARATION

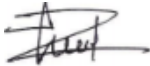
We hereby declare that, this project has been done by us under the supervision of **Md. Tarek Habib, Assistant Professor, Department of CSE** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:



Md. Tarek Habib
Assistant Professor
Department of CSE
Daffodil International University

Co-Supervised by:

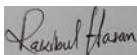


Md. Jueal Mia
Sr. Lecturer
Department of CSE
Daffodil International University

Submitted by:



Sarwar Maruf Jebon
ID: 171-15-8988
Department of CSE
Daffodil International University



Md. Rakibul Hasan
ID: 181-15-10535
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

First, we express our heartiest thanks and gratefulness to almighty God for His divine blessing makes us possible to complete the final year project/internship successfully.

We really grateful and wish our profound our indebtedness to **Md. Tarek Habib, Assistant Professor**, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of “*Artificial Intelligence, Image Processing and web application development*” to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior draft and correcting them at all stage have made it possible to complete this project.

We would like to express our heartiest gratitude to Professor Dr. Touhid Bhuiyan, Professor, and Head, Department of CSE, for his kind help to finish our project and also to other faculty member and the staff of CSE department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

ABSTRACT

Nowadays CCTV cameras are almost everywhere in Bangladesh and the number is increasing rapidly. This CCTV footage is still manually observed and it is too hard for a human to keep eyes on every single camera continuously which is boring and most of the time, because of human weaknesses, could be missed.

As a result, various types of incidents occur, which could have been prevented by taking necessary steps in time. During such various movements a large number of cars, shops, government, and private properties are destroyed, maybe by setting things on fire or breaking things that caused great loss to the country. Or people in danger do not get the help they need in time.

These days CCTV's and the internet are very cheap. These being used everywhere, CCTV's can be found in every corner of the road, school, college, university, and markets. But there is a lack of an efficient system to use this CCTV smartly. As a result, storage is wasted because they are recording all the time. But a smart system can be turned on recording automatically just whenever it needs to record. And regular CCTV systems do not send notifications or alarms in a proper way. So, most of the time crucial events are missed just because this footage is being checked manually. So, the main purpose of using CCTV being failed. People does not get help at right time and fail to secure their properties.

So, we developed a video surveillance system (web application) that will detect fire, hand gestures, and suspicious or unwanted human beings quickly and will inform (Email, SMS and Web notifications) the authorities about these unwanted incidents with proper evidence (photos and videos). It will be possible to save from the huge loss of life and property. And people can get the necessary help urgently.

TABLE OF CONTENTS

CONTENTS	PAGE
Board of examiners	i
Declaration	iii
Acknowledgements	iv
Abstract	v
CHAPTER 1: INTRODUCTION	01-03
1.1 Introduction	01
1.2 Motivation of Work	01
1.3 Objectives of the Software	02
1.4 Expected Outcome	02
1.5 Report Layout	03
CHAPTER 2: BACKGROUND	04-06
2.1 Introduction	04
2.2 Related Works	04
2.3 Comparative Studies	05
2.4 Scope of the Problem	05
2.5 Challenges	06
CHAPTER 3: REQUIREMENT SPECIFICATION	07-17
3.1 Business Process Modeling	07
3.2 Requirement Collection and Analysis	07
3.3 Use Case Modeling and Description	09
3.4 Logical Data Model	15
3.5 Design Requirements	17

CHAPTER 4: DESIGN SPECIFICATION	18-31
4.1 Front-end Design	18
4.2 Back-end Design	26
4.3 Interaction Design and UX	29
4.4 Implementation Requirements	31
CHAPTER 5: IMPLEMENTING AND TESTING	32-36
5.1 Implementation of Database	32
5.2 Implementation of Front-end Design	33
5.3 Implementation of Back-end Design	33
5.4 Testing Implementation	34
5.5 Test Results and Reports	36
CHAPTER 6: CONCLUSION AND FUTURE SCOPE	37-38
6.1 Discussion and Conclusion	37
6.2 Limitations	37
6.3 Scope for Further Developments	38
APPENDIX	39
REFERENCES	40
Plagiarism Report	41

LIST OF FIGURES

FIGURES	PAGE NO
Figure 3.3.1 Use Case Model of the system	9
Figure 3.4.1 E-R Diagram of the system	16
Figure 4.1.1 Login Page of 'Third Eye Application'	18
Figure 4.1.2 Signup Page	19
Figure 4.1.3 Signup Page with filled User Information	20
Figure 4.1.4 Home Page	21
Figure 4.1.5 Add Camera Page	21
Figure 4.1.6 Camera's Page	22
Figure 4.1.7 Camera Full View	22
Figure 4.1.8 Evidence Page	23
Figure 4.1.9 Evidence View Page	23
Figure 4.1.10 Monitoring Settings	24
Figure 4.1.11 Setting	24
Figure 4.1.12 Admin Home Page	25
Figure 4.1.13 Super User Interface	25
Figure 4.1.14 Client's Data	26

Figure 4.2.1 Roles of Backend	27
Figure 4.2.2 Tables in the Database	27
Figure 4.2.3 Full View of the Database	28
Figure 4.2.4 User Info table1	28
Figure 4.2.5 User Info table2	28
Figure 4.2.6 User Info table2	29
Figure 4.2.7 django_admin_log Table	29
Figure 5.1.1 models.py	26

LIST OF TABLES

TABLES	PAGE NO
Table 3.3.1: Use case description of create account	10
Table 3.3.2: Use case description of login	10
Table 3.3.3: Use case description of Add Cameras	11
Table 3.3.4: Use case description of Cameras	11
Table 3.3.5: Use case description of Monitoring settings	12
Table 3.3.6: Use case description of Control Room	12
Table 3.3.7: Use case description of Notification	13
Table 3.3.8: Use case description of settings	13
Table 3.3.9: Use case description of Logout	14
Table 3.3.10: Use case description of Delete Client Account	14
Table 5.4.1: Test case evaluation	34

CHAPTER 1

INTRODUCTION

1.1 Introduction

The widespread use of CCTV around the world began decades ago. With the continuous advancement of computer science and the increase in the processing power and accessibility of electronic devices, technologies based on artificial intelligence and machine learning have become suitable for the use of the common man. But even then, CCTV technology is still used in the old-fashioned way, especially in Bangladesh, people still manually look at CCTV monitors and observe events, many important events are missed due to human weakness. Less use of new technologies in CCTV and a lot of storage is lost due to various reasons. With that in mind, we created this web application using machine learning and artificial intelligence. Our web application solves many of the problems mentioned and provides some new benefits. For example, CCTV can be added to it and monitoring can be done in different modes on that CCTV. We initially worked on three features: it will start recording when it sees a specific object, it will send a notification for help with a specific hand gesture, and will notify the fire service immediately if it detects a fire. In all three modes, we have adopted three methods to inform the authority about the incident: email, SMS, and notifications in the web application. In response, you will receive email and web notifications with pictures and videos of that incidents.

1.2 Motivation

The continuous development of various technologies and some of the tragic events that have taken place in Bangladesh have inspired us to build this web application. Observing at the right time would have saved you from those dangers.

The technical advantages:

1. CCTV cameras are cheap
2. Huge development of Machine Learning & Deep Learning
3. Huge development in Hardware
4. Storage devices are cheap
5. Uses of CCTV Surveillance is growing rapidly

Because of over population, managing emergency services becoming tough day by day. No smart video surveillance system that can manage emergency services in Bangladesh. After observing above situations in Bangladesh, we feel the needs to develop a video surveillance system that will detect unwanted incidents automatically and quickly inform the authorities about these unwanted incidents to provide emergency services.

1.3 Objectives

The main purpose of creating this application is to make people's lives safer. Make the best use of CCTV for human safety. Making the best use of it as well as saving its storage and making it cost less. Saving people's time by telling them when to look at which CCTV footage. Sending immediate help or taking immediate action by recognizing the hand gestures of people in danger. In heavy-duty mode, ensure the safety of any important place. In such a situation, if people are not allowed to enter a very important place after a certain period of time, CCTV will immediately inform the authority if it can detect people. In this way, our application will enhance human safety and will save security costs, time, and effort.

1.4 Expected Outcome

This web application will definitely help increase security. None of the things that will be taken care of will be left out. When using it, people do not have to stare at the monitor for hours on end. Events will be notified automatically by notification, phone message and email in the specified situation. This application will notify you in case of fire, need of help or presence of thief in the middle of the night. It will make life easier.

1.5 Report Layout

In chapter 1: Here is a brief overview of the whole project. Here is a brief discussion of what the project is, why, what the purpose is and what motivated us to work on this project.

In chapter 2: This chapter discusses the background study that has to be done while doing this project, the development of such applications in Bangladesh and the world situation.

In chapter 3: In this chapter, we present the business process model and discuss the requirement collection, analysis, purpose and logical data model.

In chapter 4: This chapter is all about the design of the web application. We discuss front-end to Back-end and implementation requirements.

In chapter 5: In this chapter, we discuss how we implement the database, the front-end design, testing implemented features and Test Results.

In chapter 6: We talked about the limitations of this project and the scope of further developments.

CHAPTER 2

BACKGROUND

2.1 Introduction

An early mechanical CCTV system was developed in June 1927 by Russian physicist Leon Theremin. Mainly asked by the Soviet of Labor and Defense, the system comprised of a manually-operated scanning-transmitting camera and remote shortwave transmitter and collector, with a determination of a hundred lines. Another one was introduced by Siemens AG at Test Stand VII in Peenemunde (Nazi Germany in 1942) for watching the dispatch of V-2 rockets. Before 1970 video surveillance systems needed to be in constant monitoring mode because there was no way to record and store information. some years later, the development of reel-to-reel media removed those limitations and enabled the recording of surveillance footage. [1] The development of digital multiplexing (during the 1990s) brought a huge change in CCTV surveillance's world. The result was that several cameras could record at once, as well as time-lapse and motion-only recording. These saved a lot of time and money, as a result, the use of CCTV had been increased rapidly. And now, in this modern era, rapid development in both hardware and software technology brought a dramatic change in the video surveillance world. Huge development in machine learning automates a lot of things in video surveillance system.

2.2 Related Works

We developed a computer vision-based web application that can be used as an automated video surveillance system. There are many software's available that can be used as a CCTV or video management system. Here we are going to inform you about five such applications that provide this kind of facility. The first one is 'Cisco Meraki'. It's a well-known brand that provides a wide range of video-related solutions for security. It offers AI-powered video analytics. It provides motion detection and motion search features. The next one is 'Rhombus'. Similar to Cisco's Meraki, it offers some features like face detection, face

recognition, and license plate recognition. Another one is 'EyeLine Video Management Software'. The main feature of this software is it can monitor and record over 100 camera sources simultaneously. The next one is 'ContraCam'. It offers motion detection and license plate recognition. After that, 'Luxriot Evo' is another popular free CCTV management software that provides CCTV footage monitoring features.[2]

2.3 Comparative Studies

Most of these software's are paid and doesn't provide services like emergency help, the unauthorized person on the specific locations. These software do notify you through app notifications but do not send SMS's or email to the client with proper evidence instantly. Our developed applications notify the authority about the incidence in three ways: Email, phone SMS's and web notifications. 'ContraCam' doesn't work well with Ip cameras. But our application performs well with Ip cam. Most of the software we mentioned above is mainly developed for pc. But our application is web-based. So, this can be operated remotely.

2.4 Scope of the Problem

Our application can work on three different modes. At this moment, in emergency help mode, it can detect hand gesture (In our case, showing 10 fingers). In heavy duty mode, it can detect unauthorized person and In Fire detection mode, it can detect fire. Though this web application work perfectly with Ip cameras (also camera's those provide Ip address to access), it doesn't support wired camera. Since our application connect with cameras through internet, it's mandatory to have internet connections all the time. Another problem is low bandwidths slows down the system.

2.5 Challenges

To develop this application our first challenge was learning and implementing the computer vision related things like how to fetching footage frame from Ip cameras to analyze to identify what is happening there. We had to learn OpenCV, some machine learning algorithms, MVC architecture to develop the platform from where all the function will work and user can interact. Other challenges were building the backend that can send emails with evidence and phone SMS's immediately after detecting specific events. And we also had to learn the basic concepts of video cameras, video frames, frame rates etc. After that, we faced difficulties when select all three features at the same time to process or analyze simultaneously. To handle this, we had to learn concepts like multiprocessing, multithreading. Multiprocessing allows us to run more than one process or piece of code at the same time. So, those were the challenges.

CHAPTER 3

REQUIREMENT SPECIFICATION

3.1 Business Process Modeling

Client, Admin, Authority and victim are the entities. The client can log in, add cameras, can monitor all the added camera's and can select monitoring mode by login his account. A client can only access his/her account only when the admin approves the account. On the other hand, Admin can log in, add cameras, monitoring cameras, delete cameras and also have the access to get into the admin panel or the control panel. From there the admin can do a lot of tasks. He/she can see when a client joins in the application when the admin was active on the site. He/she can add or remove clients and can approve or reject any joining request. The main task that this application does is automatically filter video footage in various modes and notify corresponding authorities immediately. Last entity is victim. In the scope of our application, sometimes people need emergency help. But he/she cannot get emergency help at the right time because of the lack of rapid commutation system. Of course, a random people do not know the police phone number where they are needing help at that moment. This application is able to detect that and will send an email, phone messages, and web notifications to the authorities. Same goes for fire and motion detection. So, another entity, Authority will all the notifications and through an account, he/she can get access to the web application and can monitor all the evidence of an incidence.

3.2 Requirement Collection & Analysis

Client: In our application client can be anyone. From big businessmen to grocers, from teachers to government officials anyone can get proper service from our web application. Anyone who has some CCTV cameras and want to surveilling automatically and with smart record system, only recording just whenever needed, get saves his/her storage cost. Maybe client have some sensitive place where do not want anyone to enter. Client wants

such a application that will save his time and money. Most of client do not want to monitor his/her CCTV cameras manually. They want to automated those process. And want to get notification whenever any wrong or unwanted situation things happened on there area.

Admin: For a CCTV surveillance admin panel, the main challenges are to detect the incidence at that time. Because of human limitations most of the time, all the important events are missed. Whenever they get that, they find that it is already too late to take action. And monitoring manually is very boring. And if we think about the financial side, the admin wants a system that will save his time and money. And want to get notified whenever he/she need to take action to avoid unwanted events. Our developed application solves those problem and provide some extra features.

Authority: Here authority will be the service provider peoples. This community provides service as soon as they get a notice about a specific incidence. So, they just want real information very quickly to help. Our application can solve this problem. It will send notifications very quickly with all the evidence and location where the incidence takes place.

Victim: Here victim can be anybody who needs emergency help or anyone who enters others' property without permission or any location where there is an unwanted fire. A victim wants quick help even if he/she does not have an emergency help number of that location. Through our application a victim does not need to know the emergency help number. On that CCTV-controlled area, they can get help just by a simple hand gesture. Our application solves the victim's problems too. At this moment, it works on three conditions: emergency help, unauthorized person on a area and fire detection to provide emergency help.

3.3 Use Case Modeling & Description

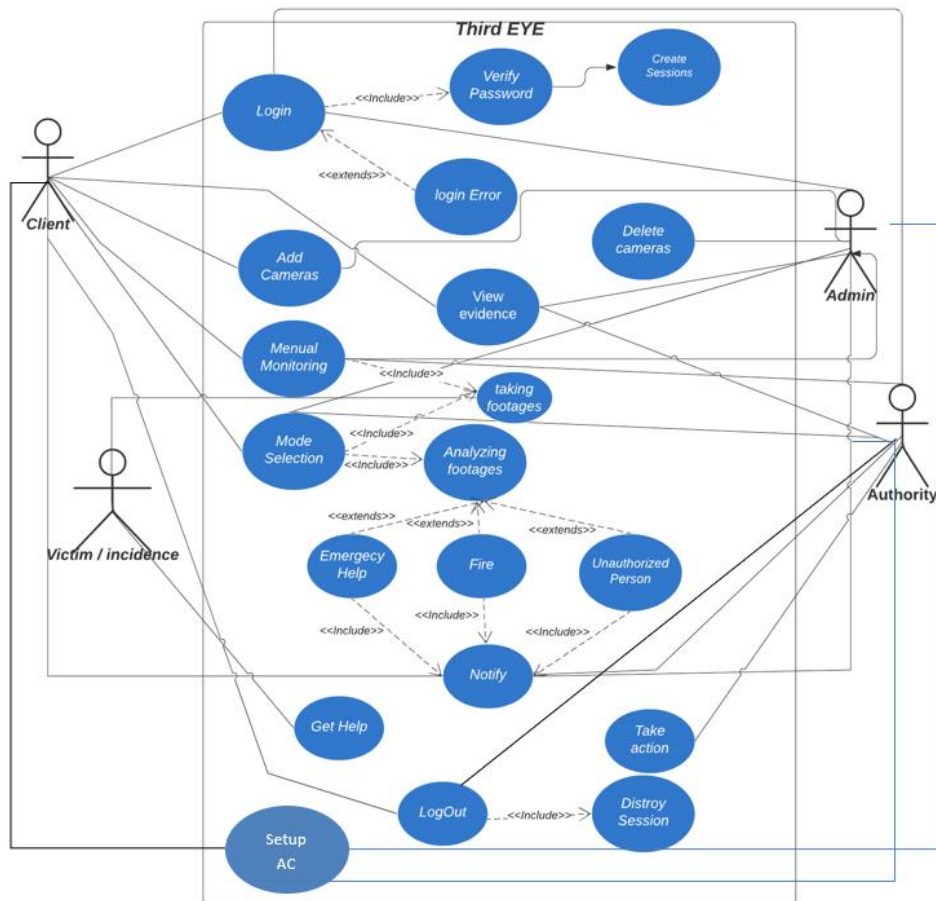


Figure 3.3.1: Use Case Model of The System

Use Case Description: In software and systems engineering, use case is something like a collection of actions or workflow of system's entities that show the clear interactions between an actor (role) and a system to achieve a specific goal. In this application there are four actors who plays roles in the system to achieve a specific goal. There are four kinds of actor in this application: client, victim or incidence, admin and authority. The system provides functionalities and services according to their activities and actions [Figure 3.3.1]

Table 3.3.1: Use case description of create account.

Use Case 01	Create Account
Primary Actor	Client/Authority/Admin
Secondary Actor	Admin
Pre-Condition	Need to have an email account and active internet connection.
Main Success Scenario	Client/Authority/admin will give their email and provide required information correctly and an account opening request will send to the admin panel for approval.
Exception Scenario	Invalid Email, Internet connection problem, Database doesn't response.

Table 3.3.2: Use case description of login.

Use Case 01	Login
Primary Actor	Client/Admin/Authority
Secondary Actor	Null
Pre-Condition	Need to have registered email account, registered user name, password and internet connection.
Main Success Scenario	Approved account will login and redirect to the main application and presenting the homepage.
Exception Scenario	Username or password error, not internet connection, server down.

Table 3.3.3: Use case description of Add Cameras.

Use Case 01	Add Cameras
Primary Actor	Client/Admin/Authority
Secondary Actor	Null
Pre-Condition	Need to be logged in, IP camera, have Ip address of that camera, Location, phone number and email address of emergency service provider and authority.
Main Success Scenario	Camera will successfully connect with the application.
Exception Scenario	Wrong camera Ip address, failed to connect or internet problem

Table 3.3.4: Use case description of Cameras.

Use Case 01	Cameras
Primary Actor	Client/Admin/Authority
Secondary Actor	Null
Pre-Condition	One or more cameras are already added to the application, user logged in and have internet connection.
Main Success Scenario	Client/Admin/Authority will see all the added camera in this section.
Exception Scenario	No camera added/available, camera is offline and no internet connection.

Table 3.3.5: Use case description of Monitoring settings.

Use Case 01	Monitoring settings
Primary Actor	Client/Admin/Authority
Secondary Actor	Null
Pre-Condition	Need to be logged in, have one or more added and active Ip cameras and internet connection.
Main Success Scenario	Will see three modes/filters to active automatic surveilling and able to select modes.
Exception Scenario	Camera is offline, no active internet connection, server down.

Table 3.3.6: Use case description of Control Room.

Use Case 01	Control Room
Primary Actor	Admin
Secondary Actor	Null
Pre-Condition	Needs to be an admin, logged in as admin and having active internet connection.
Main Success Scenario	Admin will redirect to the admin panel/control panel and with superuser power he/she can add, update, remove, delete client account, evidence
Exception Scenario	Not a superuser/admin, wrong credentials, No internet connection, server down.

Table 3.3.7: Use case description of Notification.

Use Case 01	Notification
Primary Actor	Client/Admin/Authority
Secondary Actor	Null
Pre-Condition	Need to be logged in, have one or more added, active Ip cameras, at least one mode is active, internet connection.
Main Success Scenario	Will get email, phone message and web notifications with evidence, location and times whenever incidence happen.
Exception Scenario	No mode in active, cameras offline, no internet.

Table 3.3.8: Use case description of settings.

Use Case 01	settings
Primary Actor	Client/Admin/Authority
Secondary Actor	Null
Pre-Condition	User Logged In, internet connection
Main Success Scenario	Will redirect to the settings page where he/she can change existing password providing old password.
Exception Scenario	Old password is wrong, must content 8 characters, No Internet.

Table 3.3.9: Use case description of Logout.

Use Case 01	Logout
Primary Actor	Client/Admin/Authority
Secondary Actor	Null
Pre-Condition	Need to be logged in, internet connection
Main Success Scenario	Will see the login/signup page and session will be destroyed.
Exception Scenario	No internet, server down.

Table 3.3.10: Use case description of Delete Client Account.

Use Case 01	Delete Account
Primary Actor	Admin
Secondary Actor	Null
Pre-Condition	User Logged In, need to be superuser/admin, Internet connection.
Main Success Scenario	Can delete any client's account from admin panel.
Exception Scenario	No internet, Server Down.

3.4 Logical Data Model

A logical data model is something that describes all the data elements it has, in detail. And besides that, it presents the visual understandings of data entities, attributes, keys, and relationships among them. In another word, it (LDM) can be thought of as a blueprint that represents the meaning or overview and characteristics of data elements. A logical data model of a project describes and presents the database's data characteristics. It shows all the relationships and shows that how they behave in the project and how the application identifies them and uses them.

Some of the characteristics of a logical data model are given bellow:

- Features autonomous of particular database and data capacity structures.
- Mentioning and showing the entities and attributes to be implemented.
- Identification of the business rules and connections between those entities and attributes.
- Definitions of the essential keys, outside keys, alternate keys, and reversal entities.

The logical model is behaved like a bridge from the application designer's side to see to the database plan and the developer's specifications. This demonstration ought to be utilized to approve whether the coming about applications that are built fulfill commerce and information necessities.

Our application uses a database to store data and using those data application performs as expected. Entity Relationship Diagram (ERD) is a graphical representation of the data's definition that is stored in our database and how they are related to each other. We draw our ER Diagram following Bachman's notation. In the figure below, the ER diagram of our application is given [Figure 3.4].

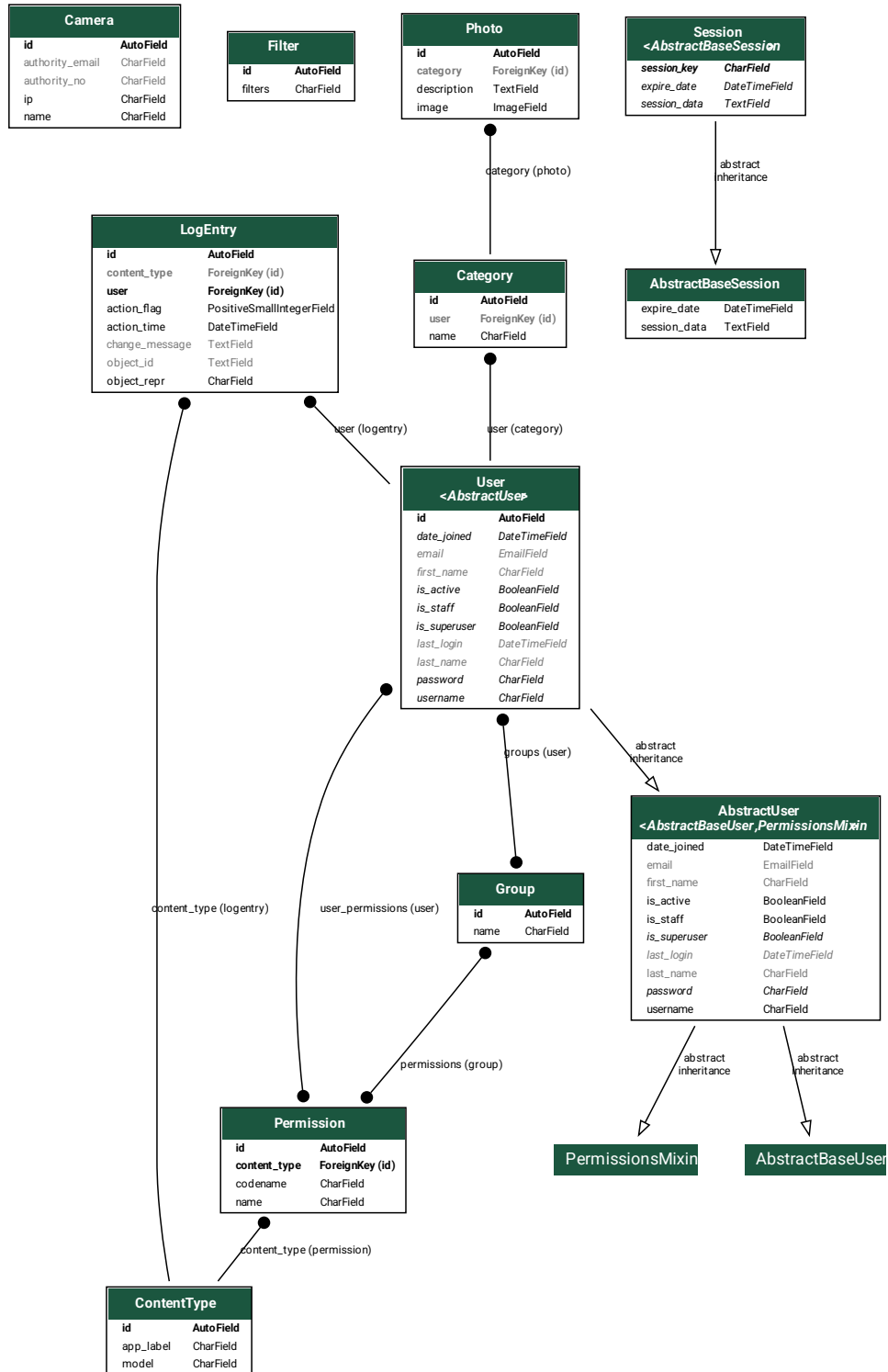


Figure 3.4.1: E-R diagram of the System

3.5 Design Requirements

System design plays a crucial role in the software industry. A good designed software turns a complex idea into reality, gives user smooth user experience and works perfectly. On the other hand, without it, system becomes very complicated and sometimes its use puts people in unwanted situation. Design system make teams work smarter, not harder. We studied about system design and found some very import directions. We looked at these aspects while designing our system.

User Friendly: User-friendly design attracts the user most. a good design is simple, easy to use, and elegant. So, we build a simple but very intuitive user interface. The user will understand the interface at the first glance.

Compatible to various Device: This is the world of modern technology. You will find a huge number of different types of monitors of different sizes. Mobile phone become very popular these days because it is compact, easy to carry, and wireless. So, we design our system following “mobile-first design”. Our application is responsible for a different type of display.

Efficient: To make it smooth and fast, we always tried to make it more litten. Here are some mentionable features:

- There are only four types of agents, who will play roles: Client, admin, authority and victim.
- Only admin can have the access of the control panel.
- Admin have the full power in this application. He/she can do a lot of operations. For example: Can delete any client, can delete any evidence, can see the clients all the data (join date, last log in, active or not)
- Client only can login if he/she have approval.
- Application can detect fire, unauthorized person and people who need emergency help.
- Client, admin and authority will get notification (Phone message, email, web notification) with evidence immediately.
- It is possible to get notification even if user is using feature phone.

CHAPTER 4

DESIGN SPECIFICATION

A design specification is something like a document that provides or specifies the design requirement of a specific project. In this section, we are going to cover everything about our project's design specifications. We are going to show the client user interface, the admin interface, the admin control panel and inform you about the logical reason why we choose those.

4.1 Front-end Design

The front end of a web application is that in where users directly interact with. So, we were very choosy when designing the front end. We try heart and soul to make it as simple as possible but at the same time, we're trying to make it nice looking, interactive, and very functional.

Registration & Login page for Client/Admin/Authority: Journey starts from here. If you have an account and the account has approval to login. Shown in figure 4.1.5.

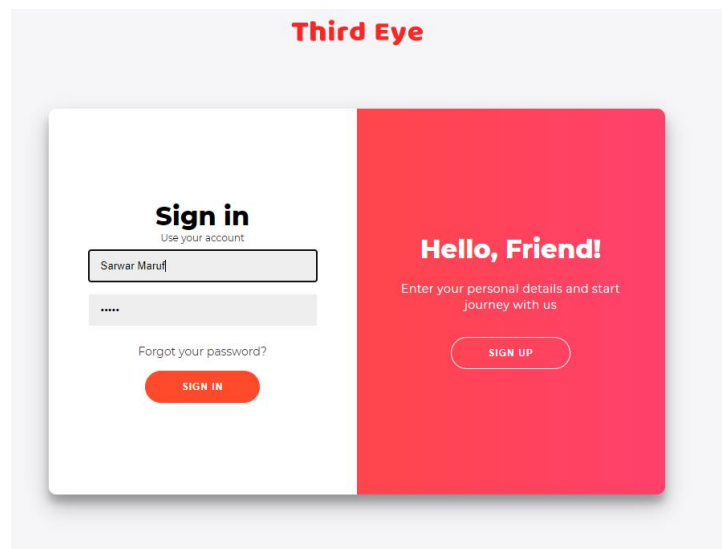


Figure 4.1.1: Login Page

This login page asks you for your user's name and password. After clicking on the login button, it sends you information to the backend for verification.

If you do not have an account, then you can apply for creating a client account from signup page. It will ask you for some basic information, for example: name, email, password.

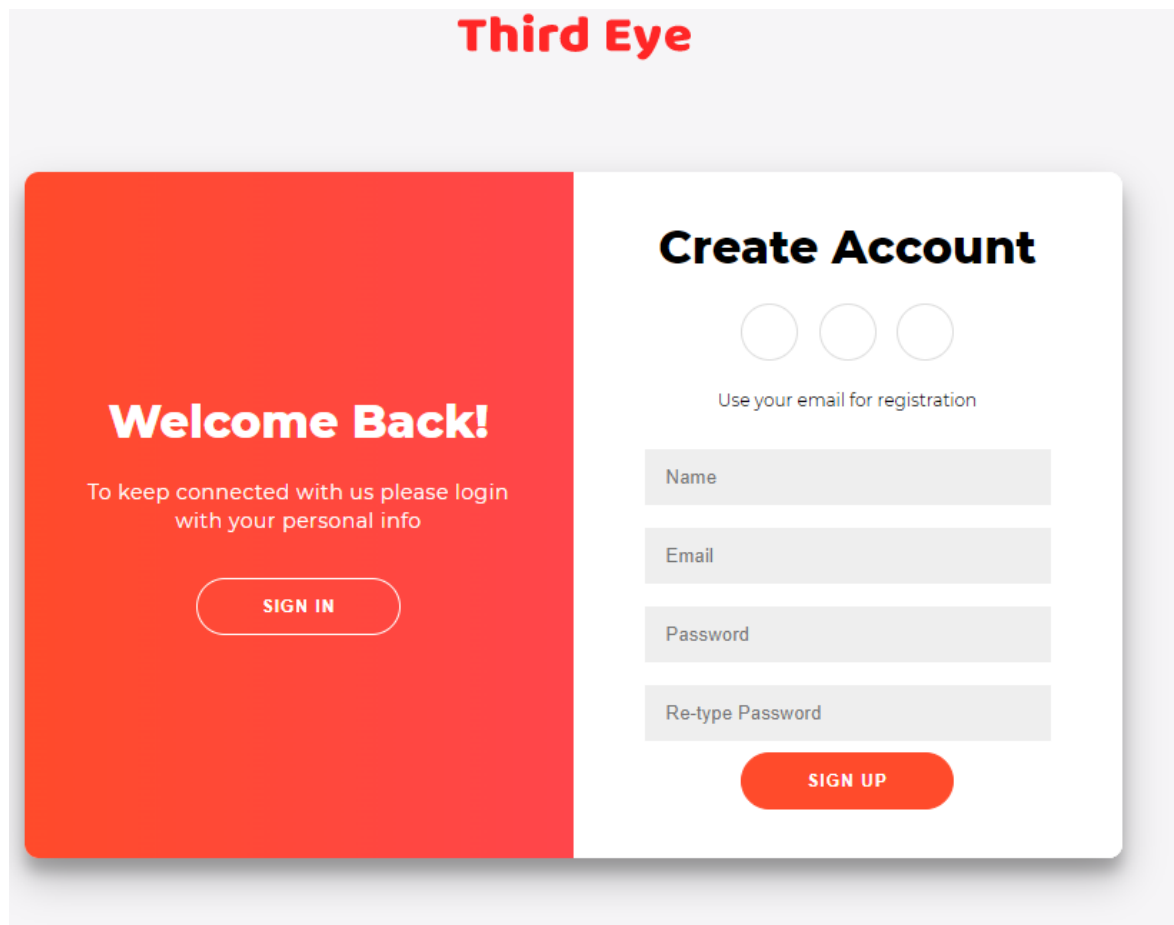


Figure 4.1.2: signup Page

After successfully filling up the form a account opening request will send to the backend to get approval. And primarily this page will validate if the email address is written in right format and check if the 'Password' and 'Re-Type Password' are same. If it doesn't match it will show you error immediately. We validate this thing using JavaScript.

Signup page with info filled shown in figure 4.1.3

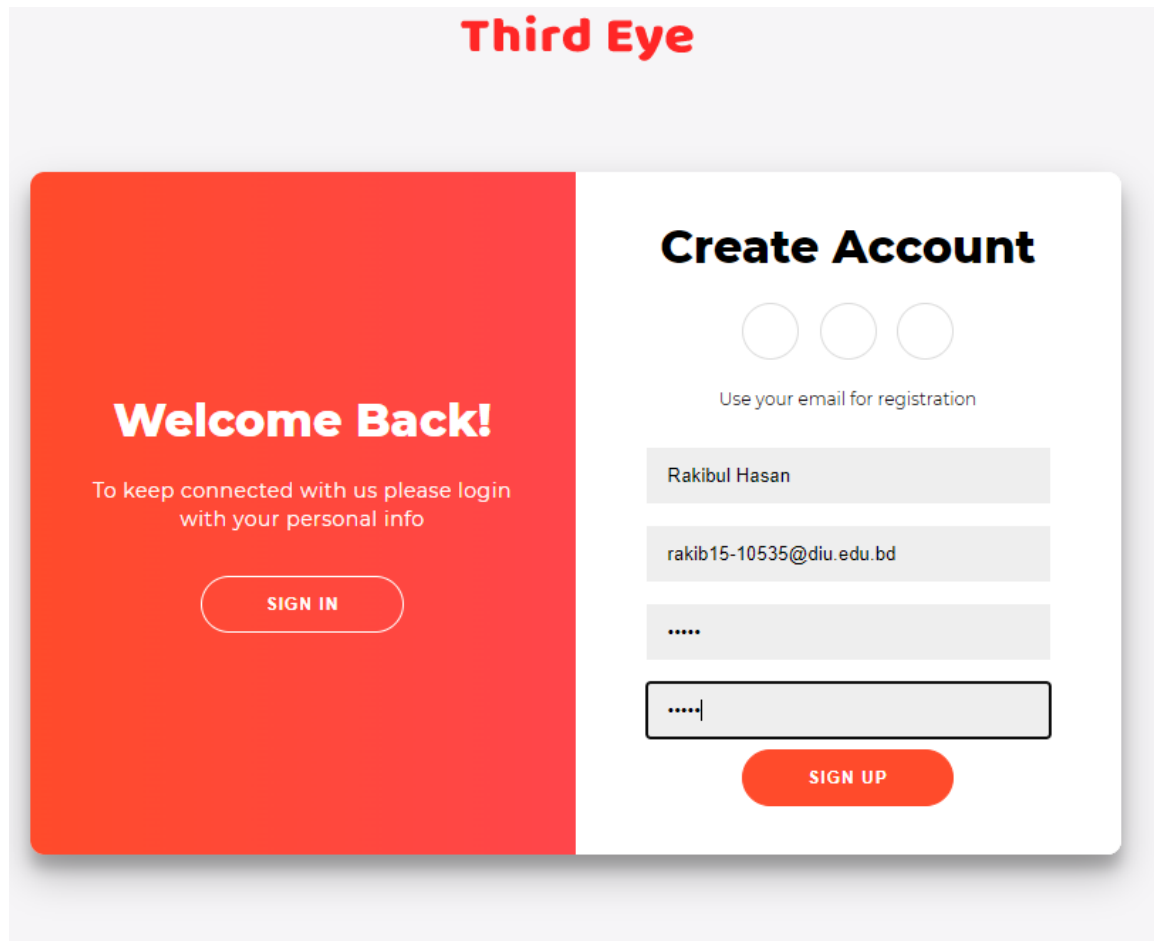


Figure 4.1.3: signup Page with filled user information

Home Page: After getting approval from admin (superuser) in the response of new account opening request, you can login to the application. After successfully login you will redirect to the main application, now you can access the ‘home page’. In ‘home page’ your view or your interface will be setting up according to your role in the web application. If you have a client, you can add cameras, set monitoring mode and view all the added cameras. But you cannot delete the added cameras. You can change your password providing your old password. Home page shown in figure 4.1.

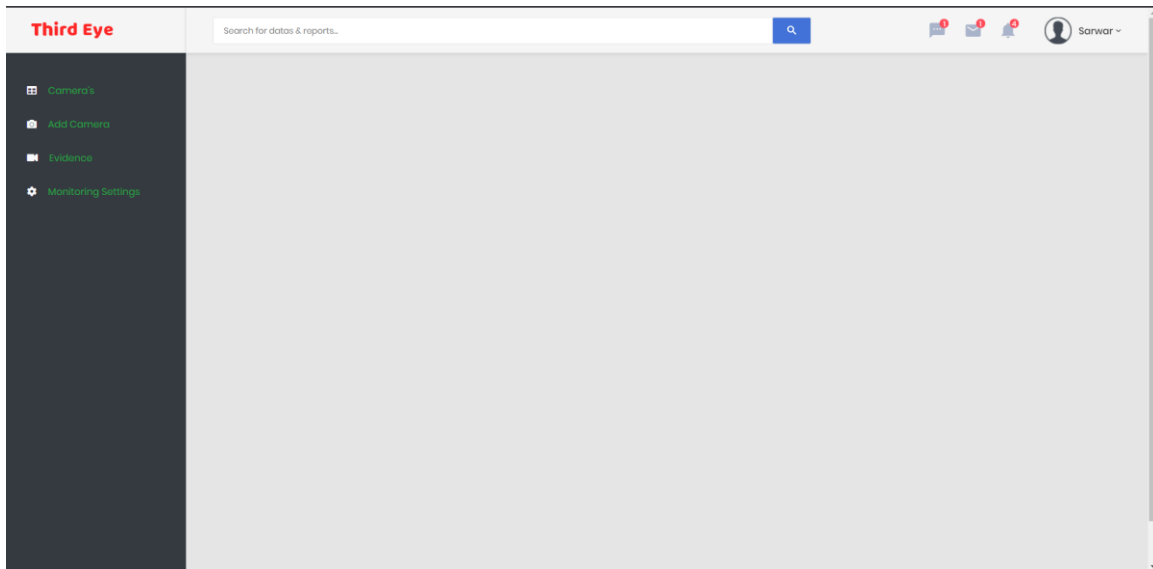


Figure 4.1.4: Home Page

Add Camera Page: Now you need to setup your IP cameras here. After clicking ‘Add Camera’ you will be asked for some information about the camera. ‘Add camera’ page shown in figure 4.1.5

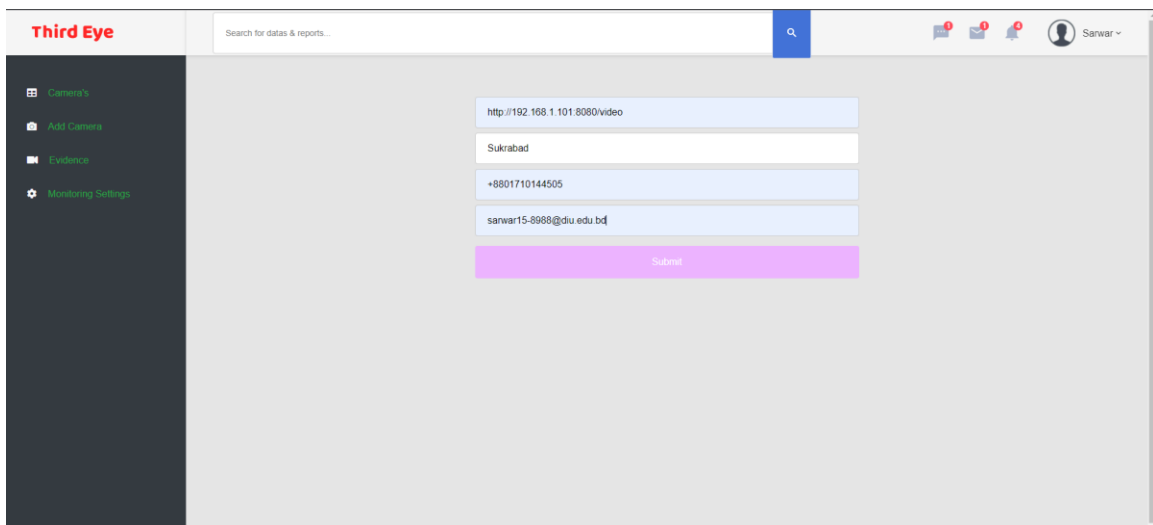


Figure 4.1.5: Add Camera Page

Camera’s Page: After successfully adding camera, you can now monitor all the added cameras from ‘Cameras’ option. From here you can manually monitor all the cameras. And the location will be shown below every camera. ‘Cameras’ page shown in figure 4.1.6

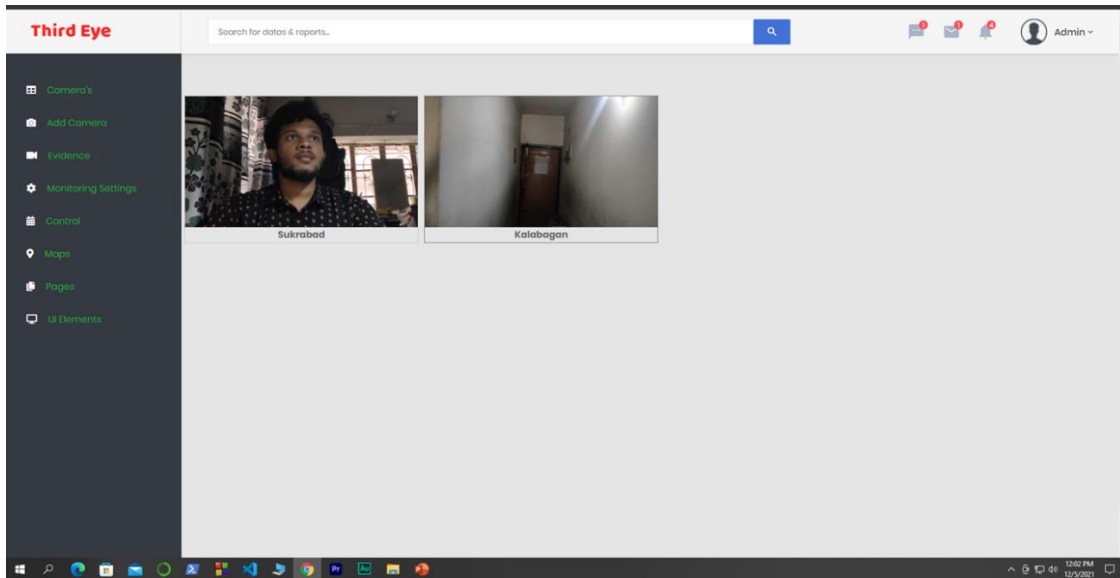


Figure 4.1.6: Camera's Page

Full View Page: You can see the full display view of any camera just clicking on it. 'Cameras' page shown in figure 4.1.7

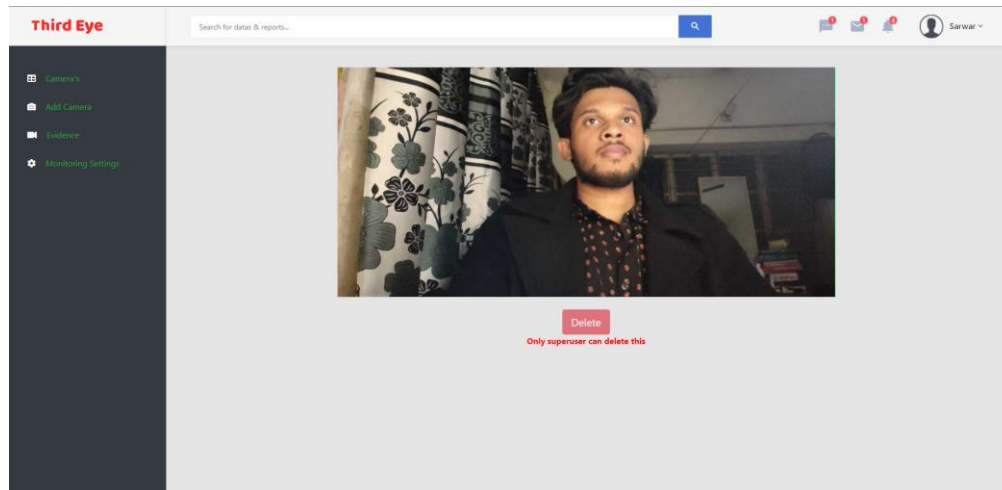


Figure 4.1.7: Camera Full View

Evidence Page: In 'Evidence' page you will find all the taken footage from CCTV camera in specific mode. You will find those in categorical order. 'Evidence' page shown in figure 4.1.8

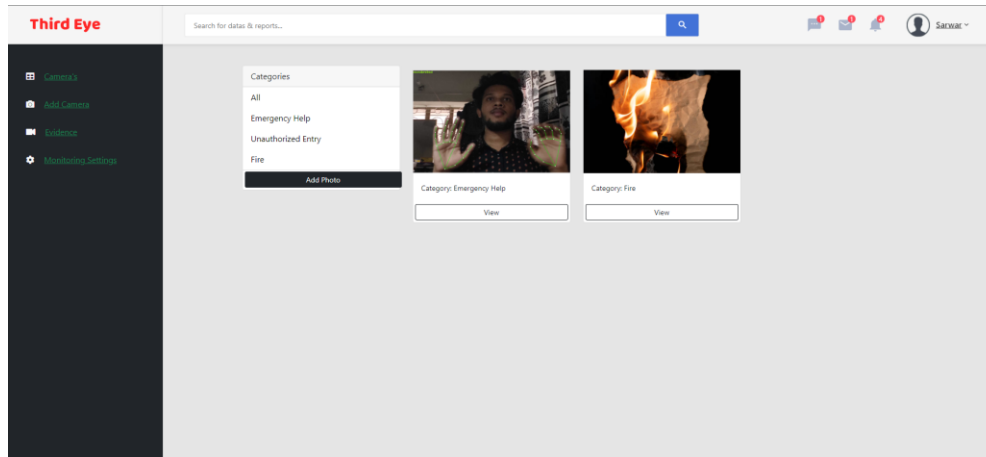


Figure 4.1.8: Evidence Page

Evidence View Page: If you want to see those evidence in full screen mode, just click on view button and you will see the full footage with description. Evidence ‘view’ page shown in figure 4.1.9

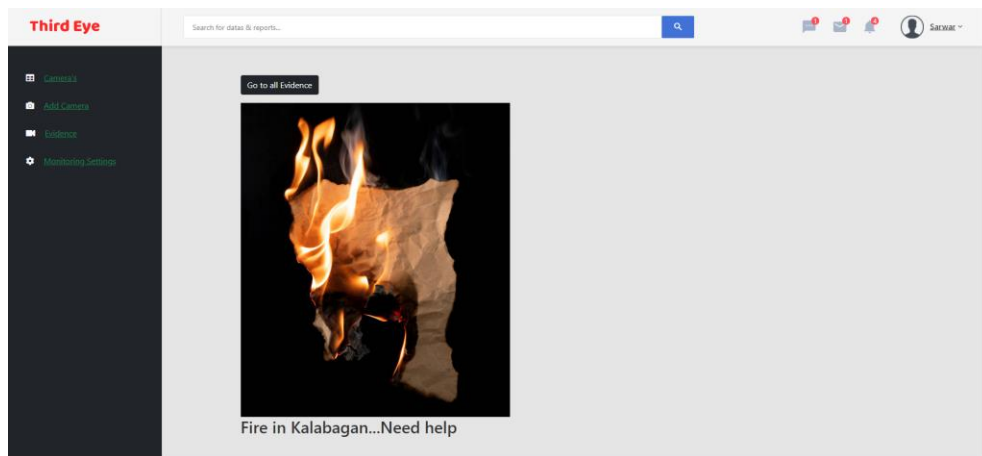


Figure 4.1.9: Evidence View Page

Monitoring Setting Page: In ‘Monitoring Settings’, you can select monitoring mode and this mode will active when you press the ‘Start Surveilling’ button. It’s an automatic process. Whenever it detects that specific thing, it will notify you with evidence (images/videos). ‘Monitoring Settings’ page shown in figure 4.1.10

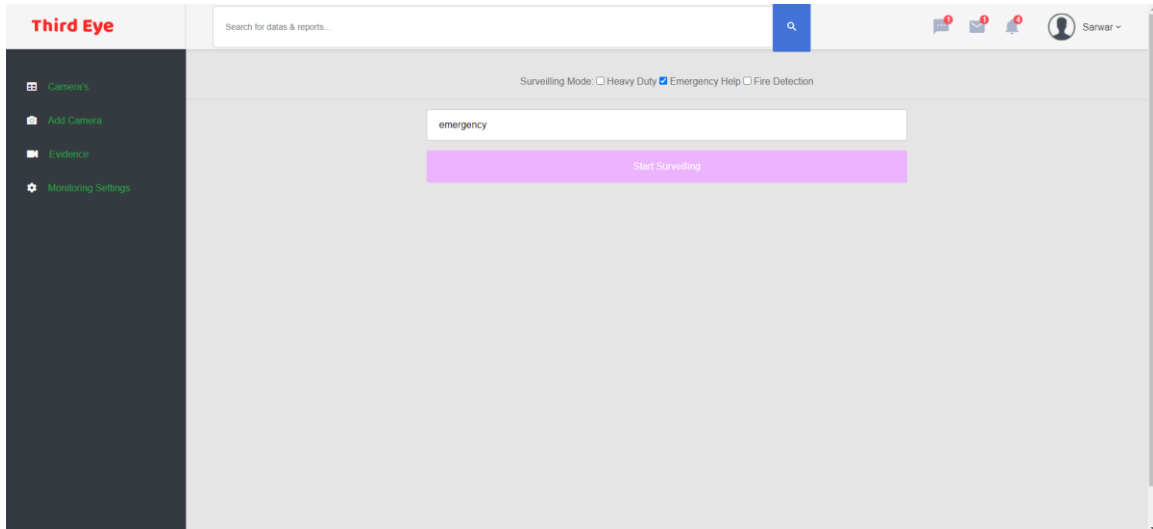


Figure 4.1.10: Monitoring Settings

Setting Page: Client can change his/her password from setting option. ‘Setting’ page shown in figure 4.1.11

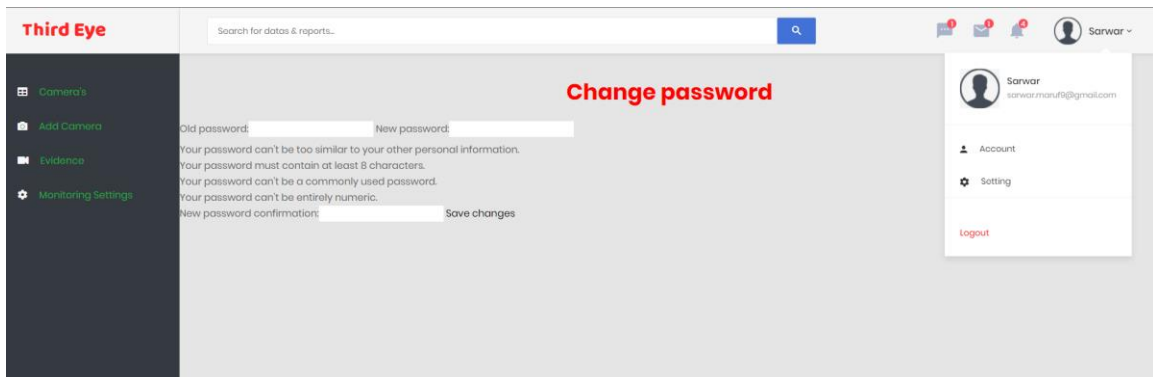


Figure 4.1.11: Setting

Control Room: Only admin can access this from his/her user interface. He/she will see some advance option in his/her user interface. ‘Admin Home Page’ page shown in figure 4.1.12



Figure 4.1.12: Admin Home Page

After clicking 'Control Room' option, he will be redirected to the 'Super User panel'. He/she can do a lot of things here. He/she get the full access of the total application. 'Super user interface' shown in figure 4.1.13

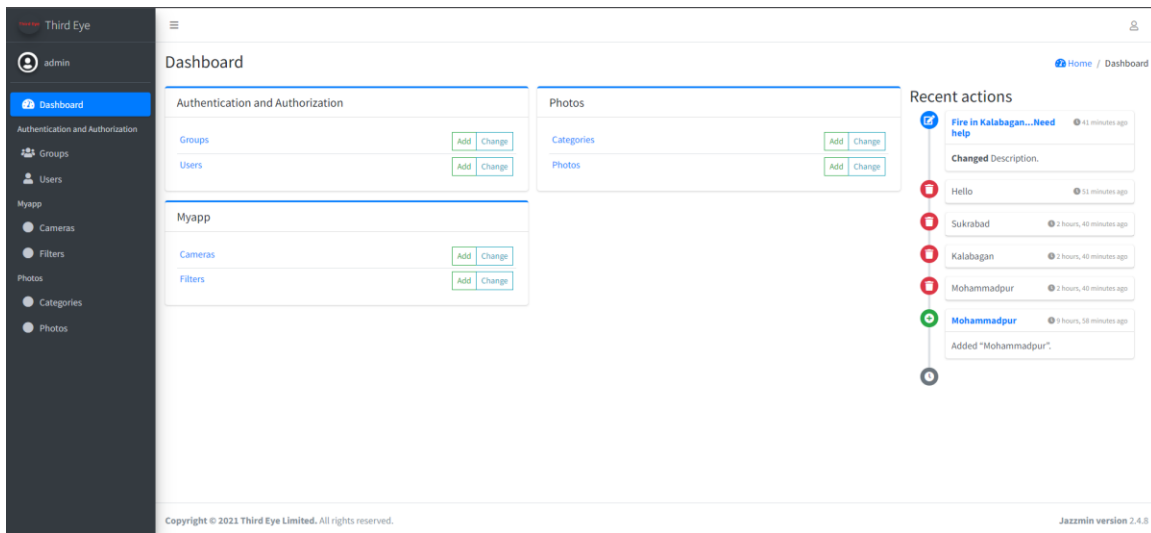


Figure 4.1.13: Super User Interface

Admin with superuser interface can add, update, delete almost all kind of data from database. Admin can see full information about client including join time, last login time etc. 'Clients Data' shown in figure 4.1.14

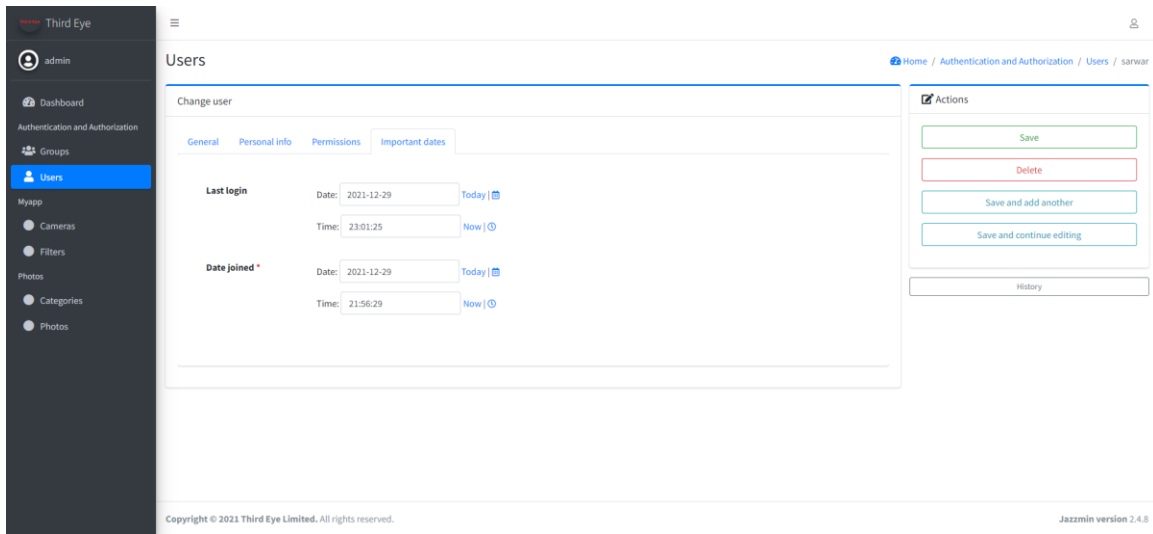


Figure 4.1.14: Client's Data

4.2 Back-end Design

Back-end refers to the part of an application that is responsible for the main logical side. A user of an application cannot view or see the backend part. Whenever anything happened in the frontend, most of the time it sends data to the backend to get a response or to get enough amount of data to manage the next view of the page for the client-side. The backend mainly handles the logical part. It processes the input data and according to that data it fetches, adds, or removes data from database. Database is must for almost all kind of application. Backend of an application lying between frontend and the backend. It behaves like a medium. There are many concepts when we are talking about backend design. For example: you have to choose a server-side programming language to build the backend, you have to have the concept of database management, authentication, authorization, data parsing, data validation, data backup technique and so on.

For this application (Third Eye), We use python as our primary language to build the backend, PostgreSQL as database and Django as the framework. 'Roles of Backend in full application' is shown in figure 4.2.1

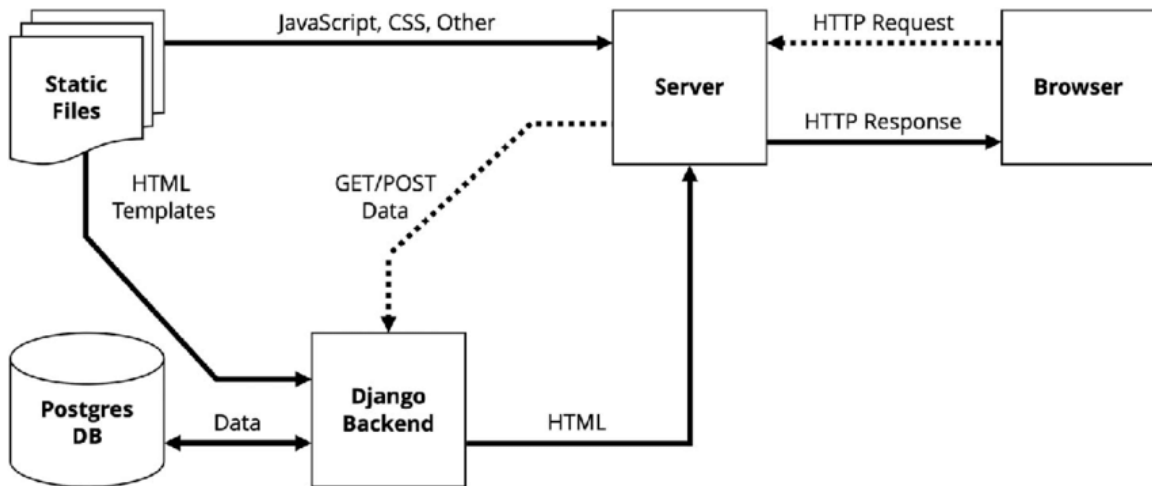


Figure 4.2.1: Roles of Backend

Using these technologies, we build a backend system that can analyze and access the database to add, remove and retrieve data. Some snapshot of our database is given below those are generated from the backend side after analyzing the data send by frontend.

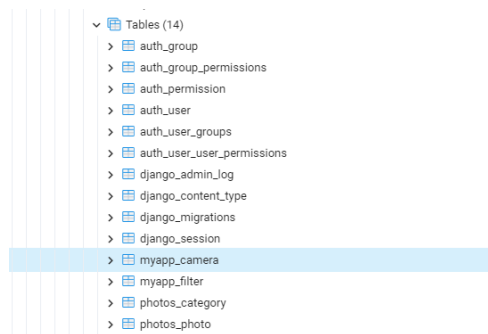


Figure 4.2.2: Tables in the Database

Currently, our application needs 14 different tables to perform fully. These 14 tables store everything we need to operate the 'Third Eye' application. You can see the full view of the PostgreSQL database of our application. 'Full view of our database' shown in figure 4.2.3

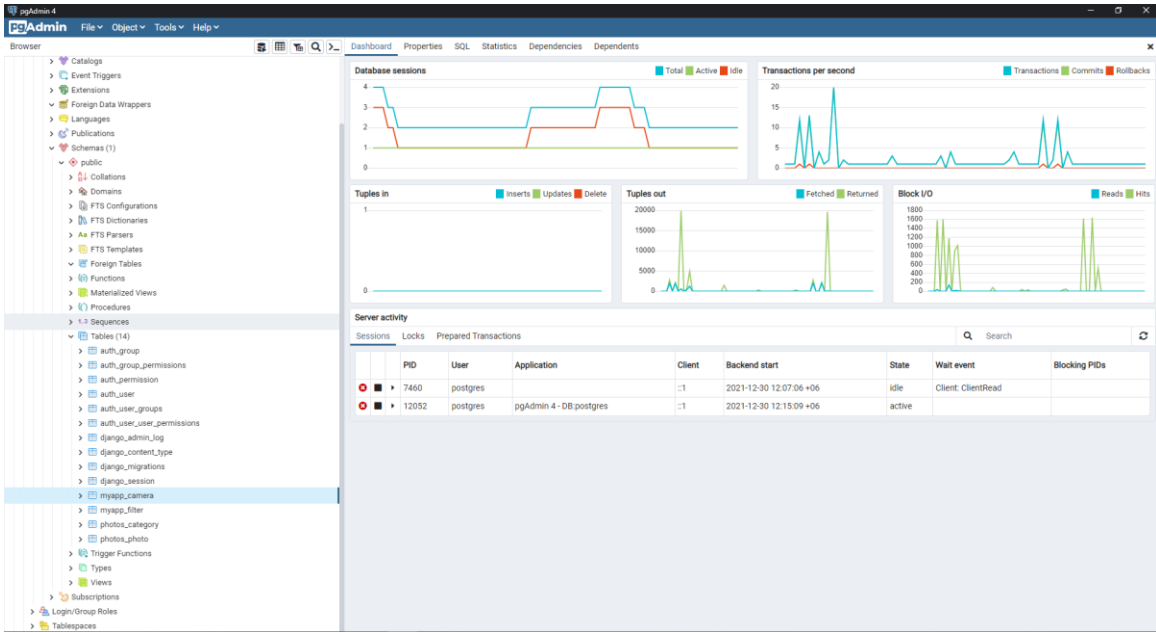


Figure 4.2.3: Full View of the Database

id	password	last_login	is_superuser	username	first_name	last_name	email
1	pbkdf2_sha256\$216000\$80pa0ev3cu0n5ChjRzxr6B0DUHFrMRTTjPunx6USRQDB0Ea0psenSHo+	2021-12-30 12:33:55.983218+06	true	admin			sarwar.maruf9@gmail
2	pbkdf2_sha256\$216000\$80eyD0loc9E0U\$Rokr2N/0FCXJ/V9N/KRJuL7vkuYyvjgFWokX/mZKq=	2021-12-29 02:26:25+06	false	Sarwar	Sarwar	Maruf	sarwar15-8988@diu.e
3	pbkdf2_sha256\$216000\$8XUHSw3Jir0P9MMaxDBUHLULUD3ubZBaoJ85qjHGAMKkJKnz669S=	2021-12-29 23:01:25.075664+06	false	sarwar			sarwar.maruf9@gmail
4	pbkdf2_sha256\$216000\$8oJaU0h7CHMSE/SELXYy1AIN3TbX1T0m+/6h183HnuK63oNmEchyQ=	2021-12-30 12:33:43.408736+06	false	Rakibul Hasan			rakibul15-10535@diu.
5	pbkdf2_sha256\$216000\$8DbAaC0dUense5YVKMuQSkWuSurnsFY7acIVORdQynVwMdyP1sAH94+	2021-12-30 12:38:32.097764+06	false	kabbo			kabbo40@gmail.com
6	pbkdf2_sha256\$216000\$8MTweIDASy8ST5aV6+2zq02efA9IVIOG/soHNBA+2NP8bIAKkInHQ=	2021-12-30 12:39:06.232496+06	false	Alif Babu			alif15@hotmail.com
7	pbkdf2_sha256\$216000\$8e8pJLbcDU0G0p3p3q5SK7M0Qext19DRedAnjyBTStzjMo+/Xj004Lzq=	2021-12-30 12:39:32.536165+06	false	Herok Kabiraz			herok005@gmail.com
8	pbkdf2_sha256\$216000\$8yFWRI4OUyE3V5CtHarDGDhNwR0PFQZPpVgpTFFEXaXPUF/YsEYJJo+	2021-12-30 12:40:24.946031+06	false	Remon Ansary			remonansary@gmail.c
9	pbkdf2_sha256\$216000\$8BkTBlo3MRH0S32N6y5QbdFVU/063hR1177InA-k/AFxi0+HPz373vYE+	2021-12-30 12:41:06.068079+06	false	Sabbir Shihab			sabbir12@gmail.com

Figure 4.2.4: User Info table1

username	first_name	last_name	email	is_staff	is_active	date_joined
admin			sarwar.maruf9@gmail.com	true	true	2021-12-29 01:58:24.979375+06
Sarwar	Sarwar	Maruf	sarwar15-8988@diu.edu.bd	false	true	2021-12-29 02:26:24+06
sarwar			sarwar.maruf9@gmail.com	false	true	2021-12-29 21:56:29.059961+06
Rakibul Hasan			rakibul15-10535@diu.edu.bd	false	true	2021-12-30 12:33:43.270107+06
kabbo			kabbo40@gmail.com	false	true	2021-12-30 12:38:31.964123+06
Alif Babu			alif15@hotmail.com	false	true	2021-12-30 12:39:06.09985+06
Herok Kabiraz			herok005@gmail.com	false	true	2021-12-30 12:39:32.403528+06
Remon Ansary			remonansary@gmail.com	false	true	2021-12-30 12:40:24.812388+06
Sabbir Shihab			sabbir12@gmail.com	false	true	2021-12-30 12:41:05.934436+06

Figure 4.2.5: User Info table2

This table stores all the user's data. It encrypts the user's password for security

Here you can see the ‘myapp_camera’ table that stores all the information about added cameras. ‘myapp_camera’ table is shown in figure 4.2.6

Data Output		Explain	Messages	Notifications	
id	[PK] integer	name	ip	authority_email	authority_no
		character varying (50)	character varying (200)	character varying (200)	character varying (200)
1	4	Sukrabad	http://192.168.1.101:8080/video	sarwar15-8988@diu.edu.bd	+8801710144505
2	5	Mohammadpur	http://192.168.1.106:8080/video	rakibul15-10535@diu.edu.bd	+8801628475767
3	6	Kalabagan	rtsp://admin:12345scw@192.168.1.210/media/video2	ferdoussshahalam@gmail.com	+8801819953018
4	7	Puran Dhaka	rtsp://admin:12345scw@192.168.1.210:554/cam/realmonitor?channel=1&subtype=1	sarwar.maruf9@gmail.com	+8801624144505

Figure 4.2.6: User Info table2

This is the admin log data table. It stores all the object data with corresponding date and time. ‘django_admin_log’ table is shown in figure 4.2.7

Data Output		Explain	Messages	Notifications				
id	[PK] integer	action_time	object_id	object_repr	action_flag	change_message	content_type_id	user_id
		timestamp with time zone	text	character varying (200)	smallint	text	integer	integer
1	1	2021-12-29 02:48:56.735495+06	1	Emergency Help	3			3
2	2	2021-12-29 02:49:06.795603+06	1	Image of Sukrabad	3			4
3	3	2021-12-29 02:59:40.995295+06	2	Sarwar	2	["changed": {"fields": ["First name", "Last name"]}]		8
4	4	2021-12-29 13:43:50.782443+06	1	Sukrabad	1	["added": {}]		1
5	5	2021-12-29 13:44:09.270029+06	2	Kalabagan	1	["added": {}]		1
6	6	2021-12-29 13:44:25.4807+06	3	Mohammadpur	1	["added": {}]		1
7	7	2021-12-29 21:02:06.813969+06	3	Mohammadpur	3			1
8	8	2021-12-29 21:02:06.833917+06	2	Kalabagan	3			1
9	9	2021-12-29 21:02:06.834916+06	1	Sukrabad	3			1
10	10	2021-12-29 22:51:40.843726+06	6	Hello	3			4
11	11	2021-12-29 23:00:58.905054+06	7	Fire in Kalabagan...Need help	2	["changed": {"fields": ["Description"]}]		4
12	12	2021-12-30 12:58:33.017603+06	5	Mohammadpur	1	["added": {}]		1
13	13	2021-12-30 13:01:22.406322+06	6	Kalabagan	1	["added": {}]		1
14	14	2021-12-30 13:02:36.349859+06	7	Puran Dhaka	1	["added": {}]		1

Figure 4.2.7: django_admin_log Table

There is total 14 tables. All the table stores important data and using backend logic, we access those data and use as per our need.

4.3 Interaction Design and UX

Interaction design is more like a concept that talks about the relationship between users and the product and how they respond to each other according to their actions. [3] In our case, the product is our web application. Interaction design involves various design elements, for

example, sound design, interactive button, the color of the elements, even spaces between browsing elements matters here.[4]

Specialist says, there are 5 dimensions of interaction design. We follow those dimensions when we were developing our web applications. Those are:

Interaction design is more like a concept that talks about the relationship between users and the product and how they respond to each other according to their actions. In our case, the product is our web application. Interaction design involves various design elements, for example, sound design, interactive button, the color of the elements, even spaces between browsing elements matters here.

Specialist says, there are 5 dimensions of interaction design. We follow those dimensions when we were developing our web applications. Those are:

1. Words: Words for button labels, directional labels, warning message should be meaningful. We were very choosy when we named our various labels.
2. Visual representations: Images, icons, button design should be meaningful and intuitive.
3. Physical objects or space: This is the medium or devices by which the users interact with our application. So, we made our website responsible for different types of displays. Cameras should be set in the right position.
4. Time: This is very important. No one wants to waste his/her time. So, we tried to make our application as smooth as possible. And the time related to animation appearing and vanishing, page transition, etc were set up smartly. So that, User will not feel boring.
5. Behavior: We made our application as interactive as possible. The interface is very simple, user-friendly, and welcoming.

On the other hand, User experience (UX) refers to the overall performance and interaction of the web application. In our application client, the admin, authority can log in, log out, perform their task, see customized homepage according to their role.

4.4 Implementation of Requirements

To implement this project, we need a text editor for writing HTML, CSS, Bootstrap4, JavaScript, jQuery, and python code. We also need a database, here we prefer PostgreSQL. Because it's free and we can use it at our production level. We need photoshop to design the UI/UX. We also need at least one IP camera to implement this project. It is possible to use a smartphone as an IP camera. Later we need to host our web application to a cloud platform. So, that anybody can access the application

CHAPTER 5

IMPLEMENTATION AND TESTING

5.1 Implementation of Database

In this application, we have used the PostgreSQL database to store all the data. We have used this database because it is free for both commercial and noncommercial purposes. It works very smoothly with the Django framework. There is a concept named object-relational mapping layer (ORM). Django framework comes with this feature. We can use ORM in Django. It can be used to interact with application data from various relational databases such as PostgreSQL, MySQL, and SQLite; in our case it is PostgreSQL. It's very easy to make a table using ORM in Django. [5] Developers do not need to write the actual SQL queries, just write the model component and it will be automatically converted into a database table using Django ORM. Here is an example how easy it is to create table in database using Django ORM. 'models.py' is shown in figure 5.1.1

```
photos > models.py > ...
1 from django.db import models
2 from django.contrib.auth.models import User
3
4 # Create your models here.
5
6
7 class Category(models.Model):
8     class Meta:
9         verbose_name = 'Category'
10        verbose_name_plural = 'Categories'
11
12    user = models.ForeignKey(
13        User, on_delete=models.SET_NULL, null=True, blank=True)
14    name = models.CharField(max_length=100, null=False, blank=False)
15
16    def __str__(self):
17        return self.name
18
19
20 class Photo(models.Model):
21     class Meta:
22         verbose_name = 'Photo'
23         verbose_name_plural = 'Photos'
24
25    category = models.ForeignKey(Category, on_delete=models.SET_NULL, null=True, blank=True)
26    image = models.ImageField(null=False, blank=False)
27    description = models.TextField()
28
29    def __str__(self):
30        return self.description
31
```

Figure 5.1.1: models.py

5.2 Implementation of Front-end Design

We always tried to make our web application simple looking and attractive for user. We have used some core technologies and some new technologies for building this application.

We have used:

- HTML5: To build the basic structures.
- CSS3: To design the structures.
- JavaScript: To make the interface interactive.
- jQuery: To make things animated.
- Bootstrap4: To build the responsive pages.

5.3 Implementation of Back-End:

You have used a python-based framework to build our application. We have also used some APIs to enable some functions. All the technologies and algorithm we used to develop our application is given below:

- OpenCV: It's a computer vision library. It allows to perform image processing related tasks and also support model execution for Machine Learning (ML) and Deep Learning. In our application most of the time we need to take footage from connected IP cameras to analyze those footage and OpenCV allows us to take very frame from streaming videos.
- Django: This is the python-based framework where we write our whole backend program.
- MediaPipe: It is an open-source cross-platform, customizable machine learning solutions for live video (coming from Ip camera or other source) and streaming media. We have used it to detect and making specific hand gesture. [6]
- Haar Cascade Classifier: We have used it for detecting object in a frame that are coming from cameras.
- Gmail: We have used Gmail to send email from our web application as notifications. To do this we had to enable less secure apps from Gmail.

- Twilio: This is a communication APIs. We have used in our backend to send mobile messages.

5.4 Testing Implementation

Software testing is a procedure or method to check whether the software product's behavior matches with the expected requirements or not and also to ensure that the software product is defect-free. [7] It is very important to test any developed software before publishing it or hosting it. We tested all the functions of our application and the test case evaluation table is given below.

Table 5.4.1: Test case evaluation

Test case	Test input	Expected outcome	Obtained outcome	Pass/Fail	Tested on
1. Login	Login using various browsers	Login successfully	Successful login	pass	18-10-20
2. Signup	Need to input information email, user name and password	Have a new account	Have a new account	pass	18-10-20
3. Add Camera	Need to provide cameras information	A new Ip camera is added	A new Ip camera is added successfully	Pass	20-10-20
4. Monitor cameras	Need to click on the camera option	All the added cameras should show here	All the added cameras are monitorable	Pass	20-10-20

Test case	Test input	Expected outcome	Obtained outcome	Pass/ Fail	Tested on
5. Monitoring Mood Selection	Need to select monitoring mode	Camera start monitoring on that mode	Camera start monitoring on that particular mode	Pass	27-10-20
6. Get Notification	One monitoring mode need to be turned on	Application will send email and messages at selected situations	Application sends email and messages at a particular situation	Pass	30-10-20
7. Evidence viewing	Need to go to evidence page	Can view all the added photos and videos	Viewing photos and videos successfully	Pass	12-11-20
8. Changing Password	Need to be logged in and enter the old password	Password is changed	Password changed successfully	Pass	27-11-20
9. Delete Cameras	Have an admin account and press delete button	Camera is deleted	Camera deleted successfully	Pass	06-12-20

Test case	Test input	Expected outcome	Obtained outcome	Pass/ Fail	Tested on
10. CRUD Operations	Need to logged in as admin, go to control room option and create, delete update client information	Account deletes, create and update possible	Successfully able to add client account, delete account and update account	pass	21-12-20

5.5 Test Results and Reports

Test report is very important to evaluate the application, it's more like a formal way to prove that reflects testing results. It's a documented proof that everything is working as expected and besides that it also shows that which functionality doesn't work properly.

In this chapter, we have shown all the test cases, test input, expected output and our actual obtained output. And according to this information we labeled our test status; is it Pass the test or not.

So, at the end of the development, all the features are working as expected (shown in Table: 5.4.1)

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

6.1 Discussion and Conclusion

Finally, our project has been completed. Although no project is ever completed, there is always room for improvement. With that in mind, we are pausing our project's work here. We are confident about our 'Third Eye' web application. We had started this journey thinking that our developed application will add some value to social security and will speed up the emergency services. At this stage, our application is functioning as expected. We tested it in different situations and almost all of the time it performs perfectly. Hopefully, the wise use of the features that have been given in our project so far will prevent many social crimes in the society, on the other hand, will make the lives of the people a little more comfortable.

6.2 Limitations

We have tried our best to eliminate your errors, but some errors remain due to limited knowledge, resources, and time. Some of the limitations are also mentioned below:

- Adding too many IP cameras to the application adds complexity due to a shortage of resources or processing power.
- It is totally internet-based, cameras must have to be internet-connected to synchronize with the application. A slow internet connection might hamper the services.
- Some sort of disabled people maybe faces difficulties while asking for emergency help because it requires you to show your ten fingers.
- Since the whole project relies on camera footage, the environment can have some effect on decision-making.

6.3 Scope for Future Developments

This project can be labeled as a completed project if you look at our project requirements. But there is a lot of scopes to work with this type of project. And we also have some plans to work on a larger scale with this project. Some of them are given bellow:

- We will add more advanced features like finding individuals and tracking.
- We will make it faster, smoother, and enables it to work with a huge number of cameras.
- We will build an android app which can be connected with this application.
- We will develop our analytical model so that it can predict human behaviors earlier to prevent unwanted situations.

APPENDIX

Appendices A

Projection Reflection: We started this project in the fall 2019 semester as the final project of the last year. While thinking about the project idea, we have tried to find something that will have a positive impact on society, on people's life. Now, after completing this project we are happy. Hope that it will be a useful application.

Appendices B

Abbreviations and Acronyms

HTML: Hypertext Markup Language.

CSS: Cascading Style Sheets.

DOM: Document Object Model.

IP: Internet Protocol.

CCTV: Closed-circuit television.

DBMS: Database management system.

ORM: Object-relational mapping layer.

UI: User Interface.

UX: User experience.

REFERENCES

- [1] Closed-circuit television, available at << https://en.wikipedia.org/wiki/Closed-circuit_television>>, last accessed on 10-10-2021 at 10:30 AM
- [2] The 10 Best Video Management Software (VMS), available at << <https://www.getkisi.com/blog/best-video-surveillance-software> >>, last accessed on 10-10-2021 at 11:00 AM
- [2] Django backend, available at << https://www.researchgate.net/figure/Web-application-with-Django-backend-Adapted-from-Mozilla-2019_fig3_340444969 >>, last accessed on 15-11-2020 at 12:00 PM.
- [3] Interaction Design for Usability, available at << <https://www.interaction-design.org/courses/interaction-design-for-usability> >>, last accessed on 12-12-2021 at 1:00 PM.
- [4] What is Interaction Design? available at << <https://www.interaction-design.org/literature/article/what-is-interaction-design> >>, last accessed on 12-12-2021 at 1:30 PM.
- [5] Django ORM, available at << <https://www.fullstackpython.com/django-orm.html>>>, last accessed on 18-12-2021 at 8:00 AM.
- [6] Live ML anywhere, available at <<<https://mediapipe.dev/#:~:text=MediaPipe%20offers%20open%20source%20cross,Android>>>, last accessed on 20-12-2021 at 10:00 PM.
- [7] What is software testing, available at <<<https://www.guru99.com/software-testing-introduction-importance.html>>>

Plagiarism Report

Submission date: 01-Jan-2022 11:40AM (UTC+0600)
Submission ID: 1736719417
File name: Project_Report_Sarwar_Maruf_Jebon_171-15-8988_Pruned.docx (3.26M)
Word count: 6414
Character count: 34103

Plagiarism Report

ORIGINALITY REPORT

14%	12%	1%	9%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Daffodil International University Student Paper	5%
2	dspace.daffodilvarsity.edu.bd:8080 Internet Source	5%
3	en.wikipedia.org Internet Source	1%
4	Submitted to University of Wales Swansea Student Paper	<1%
5	Submitted to University of Greenwich Student Paper	<1%
6	Submitted to Colorado Technical University Student Paper	<1%
7	Submitted to Middlesex University Student Paper	<1%
8	A. Oliveira, A.B. Fortunato, J. Rogeiro, J. Teixeira et al. "OPENCoastS: An open-access service for the automatic generation of coastal forecast systems", Environmental Modelling & Software, 2020 Publication	<1%