

**FIREEYE: A WEB BASED SYSTEM FOR SAVING LIVES FROM FIRE
INCIDENTS**

BY

Md. Shameem Alam
ID: 173-15-10373

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

SUPERVISED BY

Ms. Nazmun Nessa Moon
Assistant Professor
Department of CSE
Daffodil International University

CO-SUPERVISED BY

Dr. Sheak Rashed Haider Noori
Associate Professor and Associate Head
Department of CSE
Daffodil International University



DAFFODIL INTERNATIONAL UNIVERSITY
DHAKA, BANGLADESH
SEPTEMBER 2021

APPROVAL

This Project/internship titled **FireEye: A Web Based System For Saving Lives From Fire Incidents**, submitted by **Md. Shameem Alam** ID No: **173-15-10373** 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 September 09, 2021.

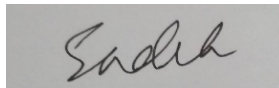
BOARD OF EXAMINERS



Dr. Touhid Bhuiyan
Professor and Head

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

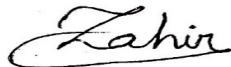
Chairman



Md. Sadekur Rahman
Assistant Professor

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

Internal Examiner



Gazi Zahirul Islam
Assistant Professor

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

Internal Examiner



Shah Md. Imran
Industry Promotion Expert
LICT Project, ICT Division, Bangladesh

External Examiner

DECLARATION

We hereby declare that this project has been done by us under the supervision of **Ms. Nazmun Nessa Moon, 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:



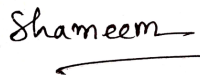
Ms. Nazmun Nessa Moon
Assistant Professor
Department of CSE
Daffodil International University

Co-Supervised by:



Dr. Sheak Rashed Haider Noori
Associate Professor and Associate Head
Department of CSE
Daffodil International University

Submitted by:



Md. Shameem Alam
ID: 173-15-10373
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

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

We are really grateful and wish our profound indebtedness to **Ms. Nazmun Nessa Moon, Assistant Professor**, Department of CSE Daffodil International University. Deep Knowledge & keen interest of our supervisor in the field of **FireEye: A Web Based System For Saving Lives From Fire Incidents** to carry out this project. Her endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stages have made it possible to complete this project.

We would like to express our heartiest gratitude to Prof. Dr. Touhid Bhuiyan and Head, Department of CSE, for his kind help to finish our project and also to other faculty members 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 discussion while completing the course work.

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

ABSTRACT

“FireEye: A Web Based System For Saving Lives From Fire Incidents” is a web-based project management system where any kind of fire is detected by cameras and sends notifications on the web to take necessary actions against the fire. Nowadays fire incidents are becoming a curse for our country as well as worldwide. Whenever a fire accident occurs it takes too much time to inform the fire service unit to take the necessary steps. As the fire spread very quickly, many people died which also caused huge destruction of people’s property. But if it becomes possible to detect the fire as early as possible then it will have a lower cost of lives and property. That is the target of our project. Our project's aim is to especially act as a firefighter which will help the fire service unit to detect every possible fire and show them in live camera through a website. As soon as any fire is found in any area, it will send the notification to the fire service unit with a live camera. So that it’ll be easier for fire service units to take necessary steps against the rising fire incidents. Fire service units can send drones to the incident area and watch the progress of fire accidents. And thus they can take necessary steps easily. The project is built with Python as a server-side scripting language using the Django Framework for the back-end development. The web application is designed with HTML, CSS, and TailWind CSS Framework.

TABLE OF CONTENTS

CONTENTS	PAGE
Board of examiners	ii
Declaration	iii
Acknowledgements	iv
Abstract	v
CHAPTER	
CHAPTER 1: INTRODUCTION	1-3
1.1 Introduction	1
1.2 Motivation	1
1.3 Objectives	2
1.4 Expected Outcome	2
1.5 Report Layout	2
CHAPTER 2: BACKGROUND	4-6
1.1 Introduction	4
1.2 Related Work	4
1.3 Comparative Studies	5
1.4 Scope of the Problem	5
1.5 Challenges	5
CHAPTER 3: REQUIREMENT SPECIFICATION	7-11

3.1 Business Process Modeling	7
3.2 Requirement Collection and Analysis	8
3.3 Use Case Modeling and Description	8
3.4 Logical Data Model	10
3.5 Design Requirement	11
CHAPTER 4: DESIGN SPECIFICATION	12-15
4.1 Front-end Design	12
4.2 Back-end Design	12
4.3 Interaction Design and UX	13
4.4 Implementation Requirements	15
CHAPTER 5: IMPLEMENTATION AND TESTING	16-19
5.1 Implementation of Database	16
5.2 Implementation of Front-end Design	17
5.3 Implementation of Interactions	18
5.4 Testing Implementation	18
5.5 Test Result and Report	18
CHAPTER 6: CONCLUSION AND FUTURE SCOPE	20
6.1 Discussion and Conclusion	20
6.2 Scope for Further Development	20
REFERENCES	21

LIST OF FIGURES

FIGURES	Page No
Figure 2.1: Different sub-tasks of fire detection	5
Figure 3.1: Data Flow Diagram of FireEye	7
Figure 3.2: Use Case Diagram of FireEye	9
Figure 3.3: ER Diagram of FireEye	10
Figure 4.1: Login Page Design of FireEye	12
Figure 4.2: Home Page view of FireEye Web	12
Figure 4.3: Blog Page of FireEye Web	13
Figure 4.4: Fire Detection From Live Web Camera	14
Figure 4.5: Fire Detection Notification through email	14
Figure 5.1: Database table for Blog Posts	17
Figure 5.2: Django Unit Test Report	19
Figure 5.3: Chrome Performance Result	19
Figure: Plagiarism Report	22

CHAPTER 1

INTRODUCTION

1.1 Introduction

According to the previous and recent history, every year many people die and huge property destruction occurs due to fire accidents. There is not any web-based system that will help the fire service unit to reduce the cost of lives and property destruction. Our project will help the fire service unit to detect any kind of fire and send an alarm quickly. The fire service unit can watch the whole scenario of a fire-affected area live through cameras. Also, the fire service unit sends drones and watches the live scenario. Thus our project will help to reduce the death and loss of property destruction.

1.2 Motivation

Fire is something that is a very useful thing consisting of bright light, heat, smoke and flames. But, inconsistent fire sometimes can be dangerous and disastrous which can cause loss of people and property. Nowadays, the lives of people and animals are always at high risk of fire incidents. And Thus the risk of people's lives and destruction of property is becoming very high.

Detecting fire and informing about it to the fire rescuers or police station is kind of challenging and time-consuming. People also get panicked during the time of any fire events and it makes them indecisive about what to do. It's hard for humans to recognize the fire in the initial stage and until it gets burning high enough. In that particular situation, fire spreads rapidly and widely if any flammable materials are available nearby.

In the case of such fires, there is a need to detect it at the right time and act immediately. Our project would be able to handle that task. It will enable quick action against fire by sending fire events information to the rescuers' team at the initial stage with photo and location information.

1.3 Objectives

As like other applications we also have some estimated targets that need to be fulfilled by these projects. One of them is to be aware as much as possible of the deaths and destruction of property for the fire-detected area using a software system. We will have a live dashboard of residential and commercial areas for fire detection if it occurs. Some of the most important objectives of our application are listed below:

- To enable early detection of fire in any kind of area we want to cover.
- To notify the rescue crew and police station in real-time with location information.
- To ensure the shortest way to reaching the desired fire detected location.
- To send the drones for observing the situation of fire detected areas.
- To enable a centralized dashboard for monitoring the overall status of fire with real-time information updates for all areas where the systems are deployed.

1.4 Expected Outcome

- Detect fire in the initial stage to be able to take action quickly.
- Monitor statuses of the areas from a centralized dashboard.
- Live video stream of the affected location from the dashboard to be able better understand the situation.
- Reduce death and property destruction caused by fire.
- Separation of industrial and residential areas according to the location of the event.

1.5 Report Layout

Chapter 1: We have already discussed the actual targets in this chapter and why we want to build this project. Thus the normal introduction with project motivation, the main objectives and what we expected from this project are discussed.

Chapter 2: As the next chapter of introduction we have discussed the project background, about some related works with FireEye, the actual points that we can resolve and finally we discussed some probable challenges we may face in this project.

Chapter 3: In this chapter we discussed the requirement specification of our application. Then make the business model, finalize the data model and use-case diagram. We have briefly described them here.

Chapter 4: In this chapter we discussed the front-end and back-end technology and how they interact with UX with the final needed tools for the implementation of this project.

Chapter 5: In this chapter we discussed how we implemented our database and connection with the front-end. We also give an idea regarding testing and the final outcome.

Chapter 6: Lastly we enclosed our report with a conclusion and how we can update our project with higher future purpose.

CHAPTER 2

BACKGROUND

2.1 Introduction

Bangladesh has been doing well in the IT sector for the last couple of years. Something like “FireEye” is needed badly nowadays for fire incidents. “FireEye” will help the fire service unit as a new era and diminish many sufferings. For the developing purpose of this application, we did a lot of case study on the internet and research papers. At first, we check everywhere for things that work like our project. But, we didn't find anything that matches our demand. Though there are several systems to detect fire, there are no website systems so far. Thinking about all of them, we build this project to solve all these limitations and make an effective system.

2.2 Related Work

After a lot of research, we didn't find anything that gives the required web services such as “FireEye”. But we found some Internet of Things work and research papers which can detect fire using Embedded Systems. Though we didn't find any complete projects, the following are similar kinds of projects.

1. Fire Detection using Embedded Systems[1].
2. Fire and smoke detection with Keras and Deep Learning[2].
3. Fire detection with video alert system[3].

Meanwhile we found some data which is very useful to build our project shows in figure 2.1

Get complete, integrated fire and life safety solutions from Johnson Controls.

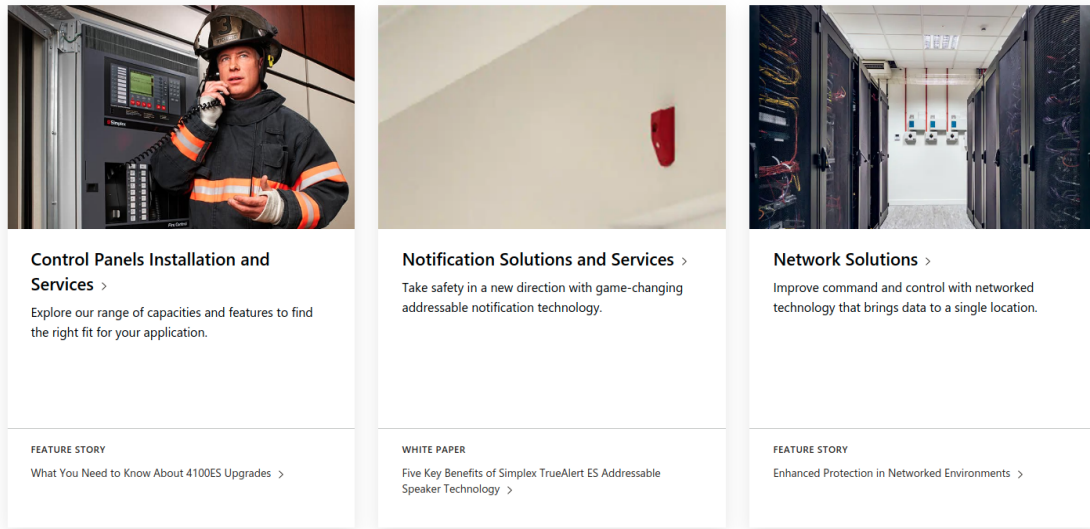


Figure 2.1: Different subtasks of fire detection

2.3 Comparative Studies

The goal of this project is to find any web-based best solution which can be very useful for the firefighters. Though, we find many different kinds of sub tasks but we need to merge them and make a complete website. And, we should take any decision using this website after being notified about any fire incidents.

2.4 Scope of the Problem

As this project help us om detecting fire and send notifications on live, there can be some problems like below mentioned:

- We need a server which will monitor the webcam and store video.
- The website “live camera” section needs to be monitored by any firefighter as we don’t know when fire incidents happen.
- Most of the time it’s not possible to invest more money and effort to update the situation etc.

2.5 Challenges

To develop such a project is very timeworthy and costly. And we faced a lot of difficulties to develop this project. Above all of them, we have tried to develop a user

friendly project. The challenges we faced while implementing this project are listed below:

- We can't test this project for a large number of area
- It wasn't possible to test our system for larger fire accident cases.
- We weren't familiar with fire detection using image processing before. We had to learn this stuff to make our project possible.

CHAPTER 3

REQUIREMENT SPECIFICATION

3.1 Business Process Modeling

To build any application we need to follow a business process model. Before implementing the project we have cleared the business process model of our application. Business process modeling can help us to determine the project structure and the interconnections between front-end and back-end. Figure 3.1 shows the data flow diagram of the project.

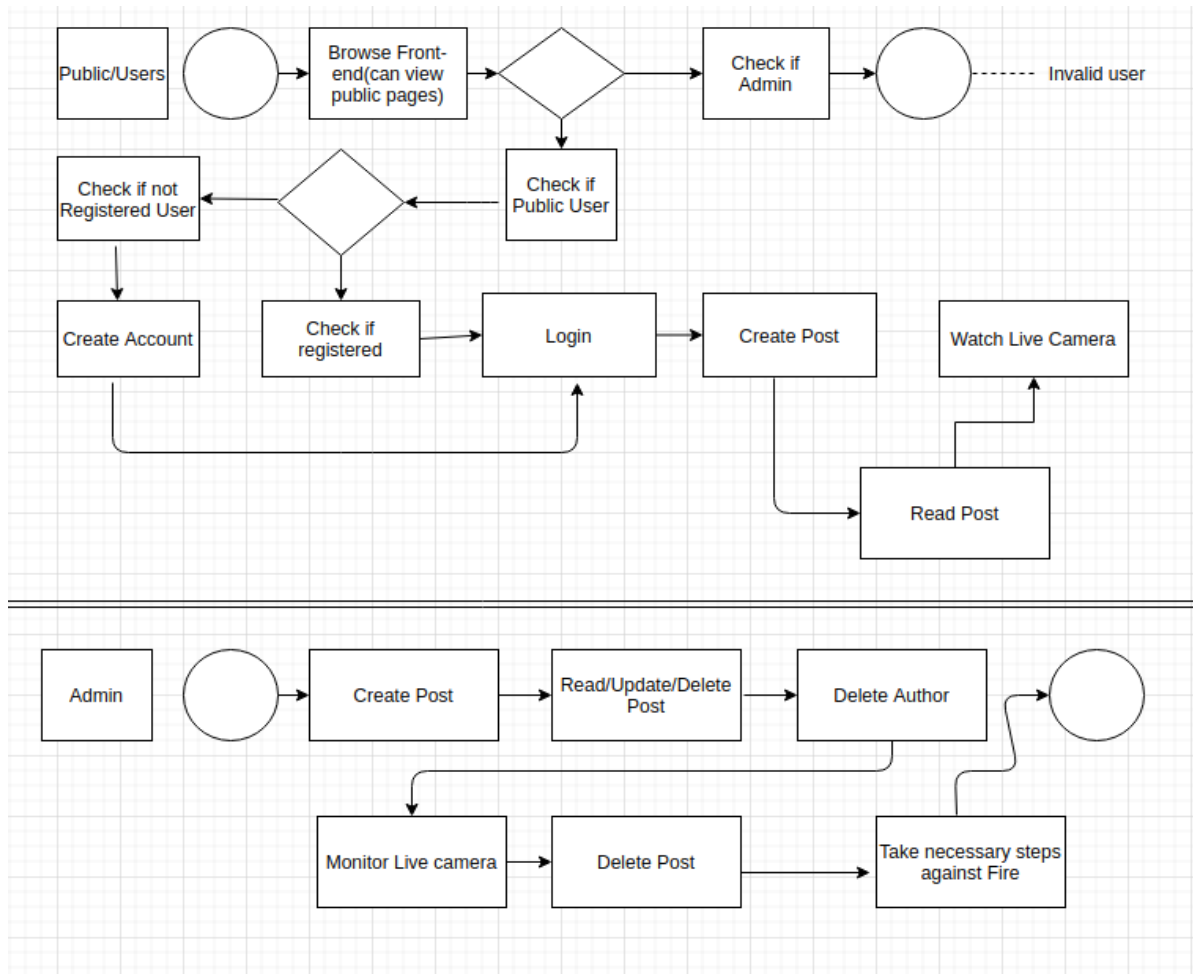


Figure 3.1: Data Flow Diagram of FireEye

3.2 Requirement Collection and Analysis

The requirement analysis is also known as functional analysis in the software industry. The functional analysis needs to be cleared, relevant and executable in any application. Before implementing any project, the project team must need to have a clear understanding regarding which functionality has been used. This requirement is one kind of description of the properties and attributes of any application.

The collected FireEye requirements based on our analysis are listed below.

- Admin as a superuser.
- Admin rules and regulations.
- Database Management.
- Create/Read/Update/Delete operation for database.
- Static File(audio/video)

3.3 Use Case Modeling and Description

FireEye has multiple individual actors respectively Admin and User. These actors can get access to the project as they are privileged. The admin panel can access all pages with the power to delete users, create user profiles, watch web camera live, add/delete database models as required. On the other hand, users can read the blog post, prerequisites page, and watch the live cameras by logging in. After the visit they can also log out.

The required Use Case Diagram of our application is shown in figure 3.2.

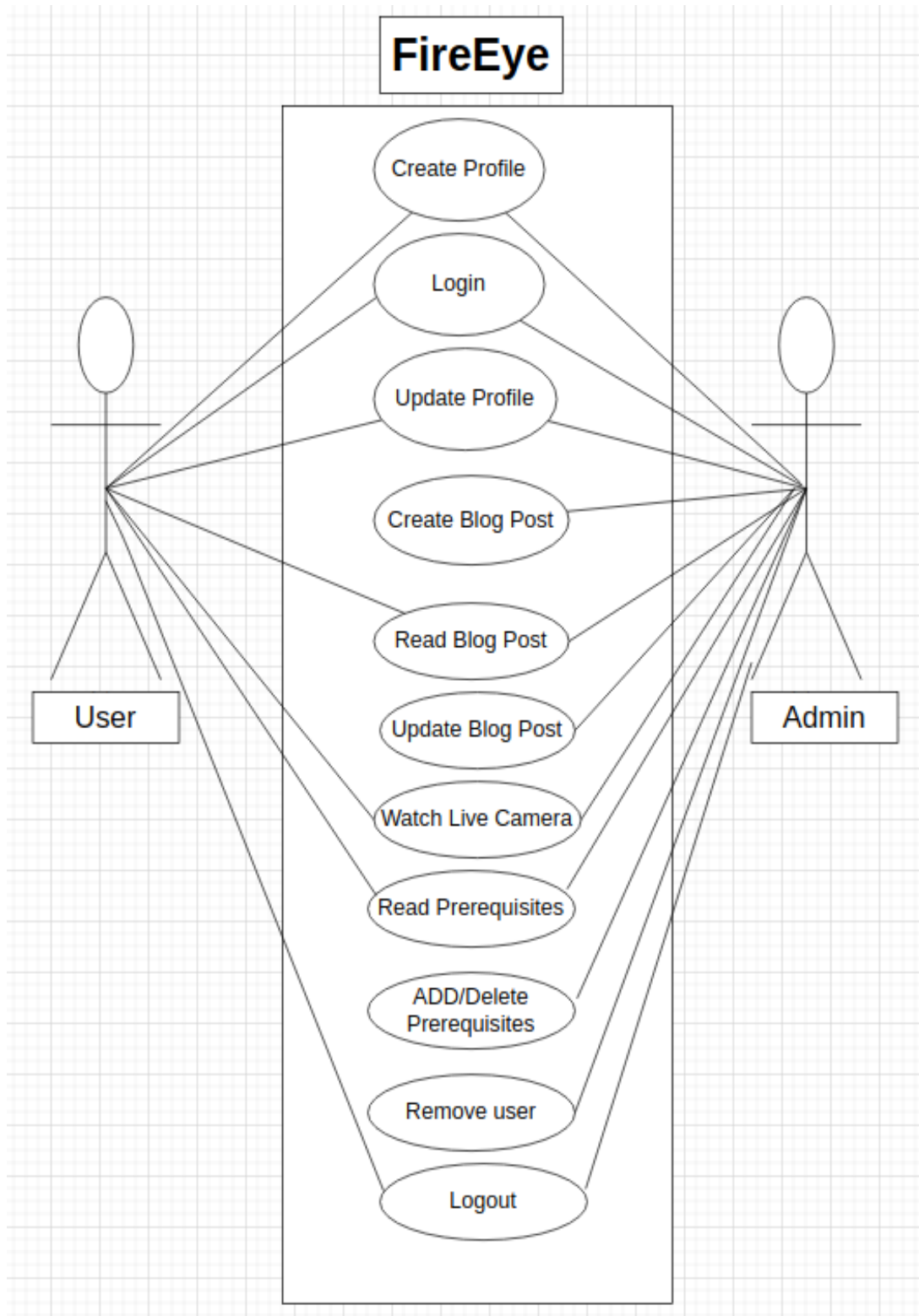


Figure 3.2: Use Case Diagram of FireEye

3.4 Logical Data Model

We need the logical data model for saving the previous-current data and demonstration of them. FireEye didn't require a huge database collectivity. We only use the database system for the static files and audio/video and blog post models. Figure 3.3 diagram shows the logical data model of FireEye.

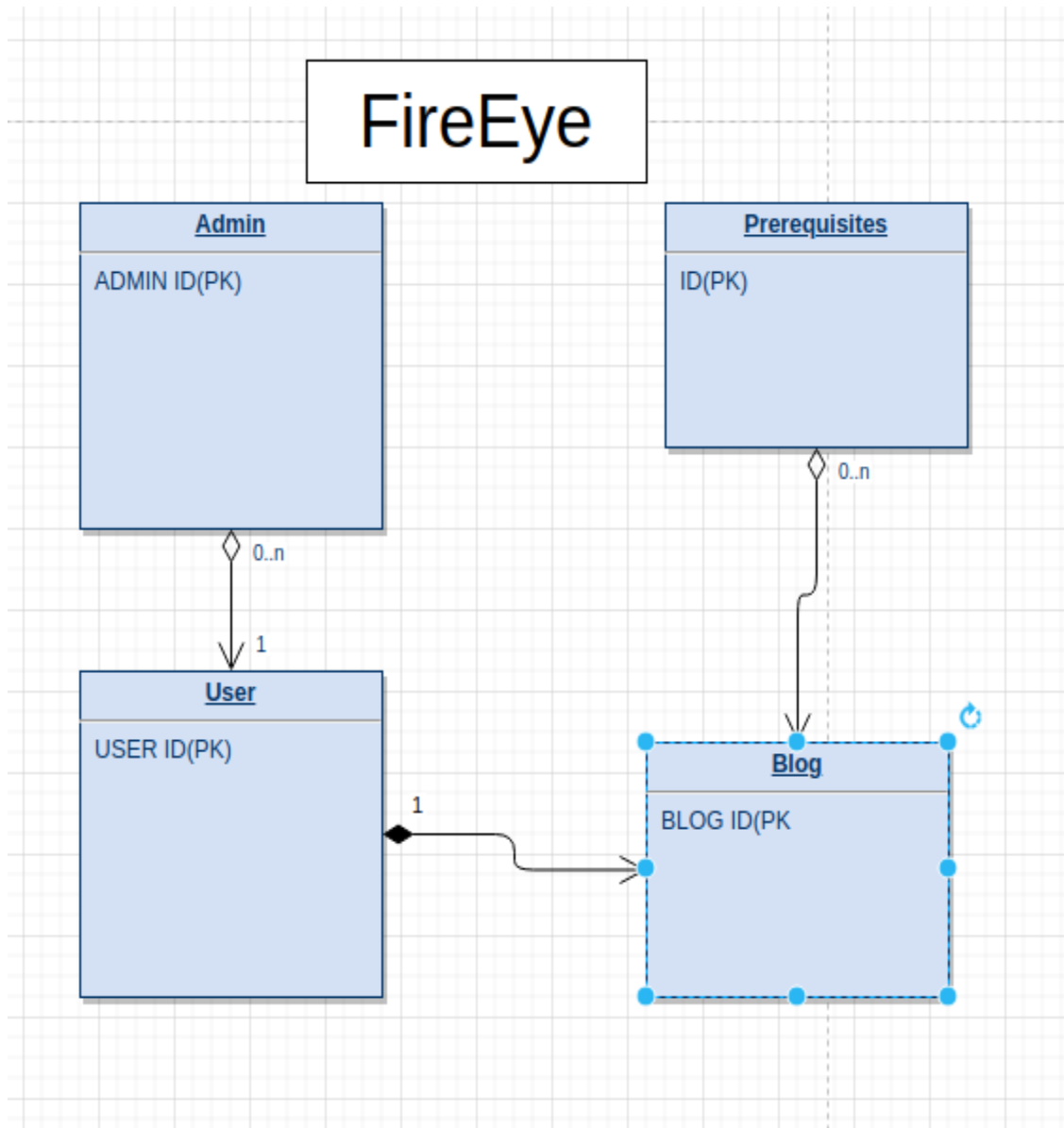


Figure 3.3: ER Diagram for FireEye

3.5 Design Requirement

As our project is a web based application, the design requirements can vary with others. According to the database model and others, we are using Django with the Tailwind CSS framework. We may also use JavaScript or jQuery as per needed. The combination of the design requirements and technical requirements mentioned above will make our application better in terms of performance and user experience.

CHAPTER 4

DESIGN SPECIFICATION

4.1 Front-end Design

Front-end design is one of the most integral things of any web-based project. To implement front-end design we can structure the application using HTML. And for styling we have applied CSS. For making our project more sensible and usable we have used the trending CSS framework Tailwind CSS and Bootstrap. Figure 4.1 demonstrates the front-end design of the login page.



Username: fire@eye.com

Password: ●●●●●●

Login

Figure 4.1: Login page of FireEye

Figure 4.2 demonstrates the home page view of FireEye Web.

FireEye Web

[About](#)

[LiveCamera](#)

[Blog Page](#)

[New Post](#)[Drafts](#)[Logout](#)Welcome: Admin

Figure 4.2: Home page view of FireEye Web

4.2 Back-end Design

Back-end design is another most integral thing of web-based projects. The back-end code is run on the server, takes the request from the user and sends the required response to the client. The back-end always has the capability to control the whole application. Only the admins can access the back-end. To access the back-end, the admins need different login credentials to manage and look after the web application. Figure 4.3 demonstrates the blog page design of our project.

Fire Eye Project

Publish

The back-end design is the one of the most integral parts of web application. The back-end code is run on the server, takes the request from the user and sends the required response to the client. The back-end always has the capability to control the whole application. Only the admins can access the back-end. To access the back-end, the admins need different login credentials to manage and look after the web application. Following are the overview of back-end in our web application. Figure 4.3 and 4.4 demonstrate the admin panel after the login as admin.



No Comments

Figure 4.3: Blog Page of FireEye Web

Figure 4.4 shows the detection of fire from live video.



Figure 4.4: Fire Detection from Live Web Camera

Figure 4.5 shows the email notifications of detecting fire.

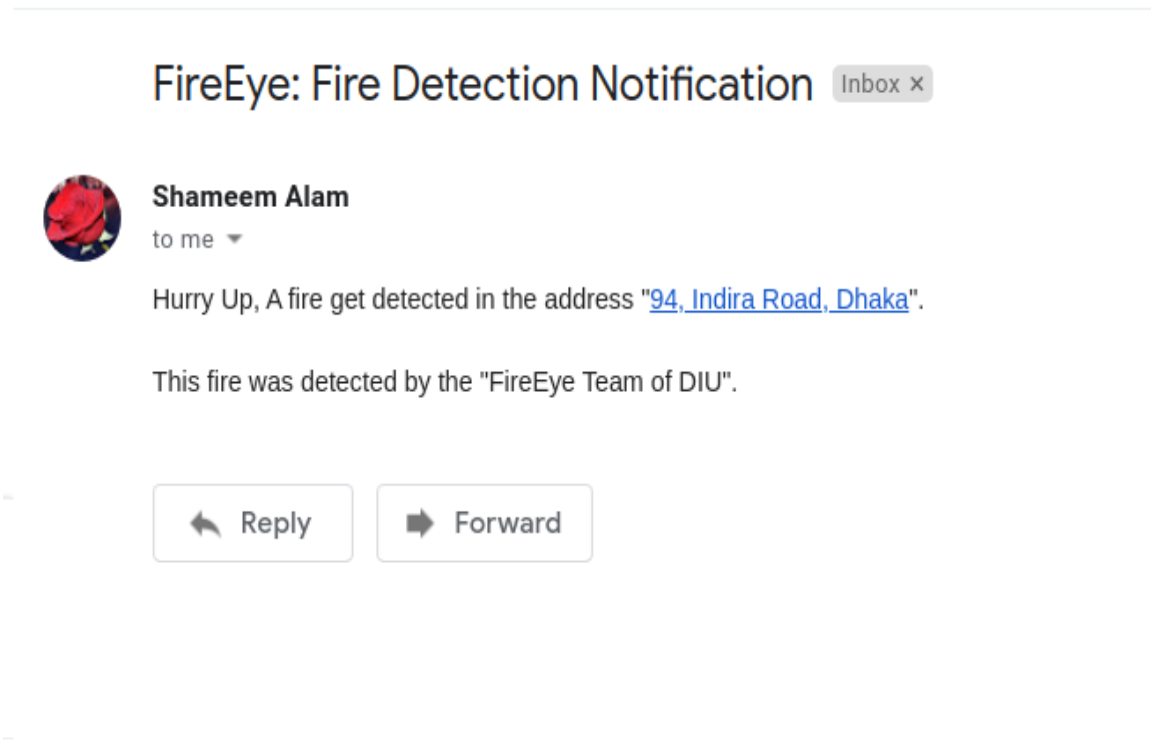


Figure 4.5: Fire Detection Notification through email

4.3 Interaction Design and UX

The interaction Design is one of key parts within the giant UX implementation. Interaction Design has 5 important dimensions. All these dimensions help for the great user experience(UX). Better interactions are very necessary so that the user gets a quick response which leads to a better user experience. For each web application we need to make communication between client and server and finally the response requested by the user. But this process may be more time consuming. To make our web application less time consuming, we used an AJAX server which will help us to use modern features of interaction. Thus get the responses and reply to the user with expected outcome quickly. All of the above things make our web application more lively and interactive.

4.4 Implementation Requirements

The main language we use to implement this project is Python with Django(v3.8). For front-end development we have one of the popular CSS Framework Tailwind CSS. The Django Framework makes the collaboration of front-end and back-end systems easier. Django needs the latest version of Python with some python extension. For the fire detection we have used the opencv extension of python. A Relational Database Management also needs to store all kinds of data and to ensure the safety of data. All the implementation requirements mentioned below in a detailed description.

Server Requirements

- Python >= 3.8
- Apache and/or Nginx

Python Packages

- XML Python
- GD Library
- JSON Python Extension
- Ctype Python Extension
- PyOpenSSL Package

Database:

- PostgreSQL or MySQL as a Relational Database Management System

CHAPTER 5

IMPLEMENTATION AND TESTING

5.1 Implementation of Database



For a web based project database implementation is one of the most important required things for data storage and further usage. In case of purposeful data storage DBMS(Database Management Systems) helps FireEye systems. FireEye needs to store, manipulate and update data involved with the fire incidents, blog posts and the whole data regarding previous fire incidents reports and visualizations.

DBMS(Database Management Systems) have four main functionalities which can help with data efficiently. And these functionalities are the core things of Database Management Systems. The functionalities are:

- Store data
- Update data
- Delete data and
- Fetch data from the database.

We implemented FireEye using a RDBMS called PostgreSQL. has one of the most advanced features of databases[4]. For the sake of database connectivity switching in any other SQL would be required which might be possible by changing a simple configuration.



As already mentioned, we have used Python's Django framework to implement this project. Django framework supports database systems out of the box. Django makes database connectivity pretty easy to implement and use in the way we want to. We use a virtual environment to configure files, it just requires some command line programs with proper database credentials. We can find the proper database credentials in django documentation page[5]. Figure 5.1 shows the database table of our blog posts.

Author: Admin  

Title:

Text:

The back-end design is one of the most integral parts of the web application. The back-end code is run on the server, takes the request from the user, and sends the required response to the client. The back-end always has the capability to control the whole application. Only the admins can access the back-end. To access the back-end, the admins need different login credentials to manage and look after the web application. Following is the overview of the back-end in our web application. Figures 4.3 and 4.4 demonstrate the admin panel after the login as admin.



Created date: Date: Today 
Time: Now 
Note: You are 6 hours ahead of server time.

Figure 5.1: Database table for Blog Posts

5.2 Implementation of Front-end Design

We have used HTML with CSS to implement the Front-end design of this project. Some open-source packages also make the workflow look better and to make the whole process with better performance.

Tailwind CSS in now-a-days become a popular UI framework that can be used for structural and more beautiful features of any web project. For the front-end implementation we have focused more on this framework. Tailwind CSS has some open-source packages which has helped us to make our front-end design more interactive.

The libraries and tools we have used are listed below:

- HTML
- CSS
- Tailwind-CSS
- Vue.js
- Bootstrap

5.3 Implementation of Interactions

For the great interaction design we have used Python and a few popular Python libraries into FireEye. We also used Django's built-in Admin panel to monitor the whole thing.

Libraries and Tools:

- Python
- Tailwind CSS
- JavaScript
- jQuery
- Bootstrap

5.4 Testing Implementation

We know, in this project we have used Python's Django framework to implement the whole project. Surprisingly, Django has an excellent manual testing system functionality. We have written as many test cases as we need to test whether everything is okay with expected outcome without any issues. If the test didn't pass that means we need to update our implementation as required. In addition, we have tried to write enough test cases to test all the things on the project.

Finally we tested the required output and performances using Google chrome's developer tools. Google Chrome's developer tool can help us to estimate the overall performance of our implementation project. It also gives us the analysis regarding project performance and time complexity respectively.

Libraries and Tools:

- Django Unit Testing
- Chrome DevTool

5.5 Test Result and Report

We showed Django Unit Testing Result on Figure 5.1

```
1. khorshed@Khorsheds-MacBook-Pro: /Volumes/Macintosh HD/Sites/blackboard (zsh)
→ blackboard git:(master) ✕ phunit
PHPUnit 6.5.7 by Sebastian Bergmann and contributors.

..
2 / 2 (100%)

Time: 455 ms, Memory: 20.00MB
OK (2 tests, 2 assertions)
→ blackboard git:(master) ✕
```

Figure: 5.1 Django Unit Test Report

Figure 5.2 shows The Google Chrome DevTool performance result.

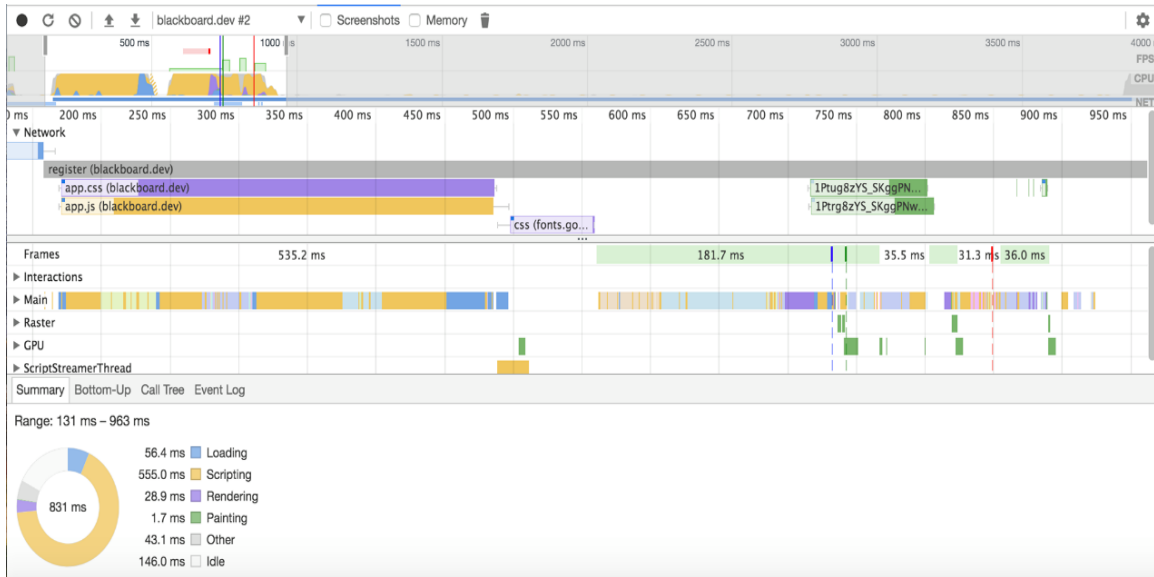


Figure: 5.2 Chrome Performance Result

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

6.1 Discussion and Conclusion

At present time and according to the previous history, every year many people die and huge property destruction occurs due to fire accidents. There is not any web-based system that will help the fire service unit to reduce the cost of lives and property destruction. Our project will help the fire service unit to detect any kind of fire and send an alarm quickly. The fire service unit can watch the whole scenario of a fire-affected area live through cameras. Also, the fire service unit sends drones and watches the live scenario. Thus our project will help to reduce the death and loss of property destruction.

6.2 Scope for Further Development

Fire is something that is a very useful thing consisting of bright light, heat, smoke and flames. But, inconsistent fire sometimes can be dangerous and disastrous which can cause loss of people and property. Nowadays, the lives of people and animals are always at high risk of fire incidents. And Thus the risk of people's lives and destruction of property is becoming very high.

Detecting fire and informing about it to the fire rescuers or police station is kind of challenging and time-consuming. People also get panicked during the time of any fire events and it makes them indecisive about what to do. It's hard for humans to recognize the fire in the initial stage and until it gets burning high enough. In that particular situation, fire spreads rapidly and widely if any flammable materials are available nearby.

In the case of such fires, there is a need to detect it at the right time and act immediately. Our project would be able to handle that task. It will enable quick action against fire by sending fire events information to the rescuers' team at the initial stage with photo and location information.

REFERENCES

- [1] Tinkercad website, <https://www.tinkercad.com/things/8twPR1uNwzK-simple-fire-alarm-system/>, last accessed on 01 January, 2021
- [2] Pyimagesearch website, <https://www.pyimagesearch.com/2019/11/18/fire-and-smoke-detection-with-keras-and-deep-learning/>, last accessed on 03 January, 2021
- [3] Researchgate website, https://www.researchgate.net/publication/325228863_Computer_Vision_based_-fire_detection_with_a_video_alert_system/, last accessed on 30 March, 2021
- [4] Amazon website, <https://aws.amazon.com/rds/postgresql/what-is-postgresql/>, last accessed on 15 February, 2021
- [5] Django Tables website, <https://django-tables2.readthedocs.io/en/latest/pages/tutorial.html/>, last accessed on 23 December, 2021

Fire-eye

ORIGINALITY REPORT

10%	9%	0%	6%
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	4%
3	Submitted to St. Petersburg High School Student Paper	1%
4	dspace.uiu.ac.bd Internet Source	<1%
5	trepo.tuni.fi Internet Source	<1%

Figure: Plagiarism Report