Prototype II and Customer Feedback

E Cubed Group (P2)

Deliverable G	Rares Constantin Serban	Emma Belal
November 10th	Louis-Philippe Keith	

Abstract

The development of prototype two is detailed here. The primary focus is on ensuring there are better notifications and means of tracking all access events. The prototype has been analyzed using a numerical model to be able to accurately project the estimated response time of 5.479 seconds for the Enterprises. Following client feedback there is a list of definitions to add clarity to our project. The prototype test plan for prototype three has been developed and will be implemented for the following internation of the project.

Table of Contents

Introduction 4
Prototype Two 4
Model 5
Prototype Test 6
  Prototype Test Analysis 6
  Prototype Test Results 6
Feedback and Comments 6
Definitions 6
Prototype Three Test Plan 7
Conclusion 7

Introduction

The development of prototype two follows from the feedback on prototype one we have received from client meeting two and three, as well as feedback from prospective users. In this Deliverable we will detail prototype two. We will justify the choices and modifications made to prototype one to create prototype two. There will be an explanation of test results using a numerical model for the newest prototype.

We have also gathered feedback and comments from prospective users on this internation of the prototype and how we will make developments for prototype three. Using all the feedback gathered we will develop a prototype test plan for the final prototype.

To note there are no updates to our bill of materials or our target specifications from the previous Deliverables.

Prototype Two

Our second prototype focuses on improving the quality of information from the notification, as well as a log system to keep track of unauthorized and authorized access attempts.

To improve the quality of information from the notification we have chosen to send a 3-second video to the administration. This is a direct update from prototype one. Prototype one was only able to send a still image of who caused the unauthorized event. The image that was sent to them was often blurred since the person at the door is not perfectly still. Making this change lets us send the administration more information on who caused the restricted access attempt.

Another way we are improving our system is to add a log system to keep track of all attempts made at door access. Since it is important for Enterprise’s to keep track of all door access attempts. It is not enough that they receive information when the event occurs, they need to be able to provide that information to the appropriate people after the event has occurred.

These updates to the prototype are essential to have a product that is closer to target specifications and to having a viable product to demonstrate to the client.

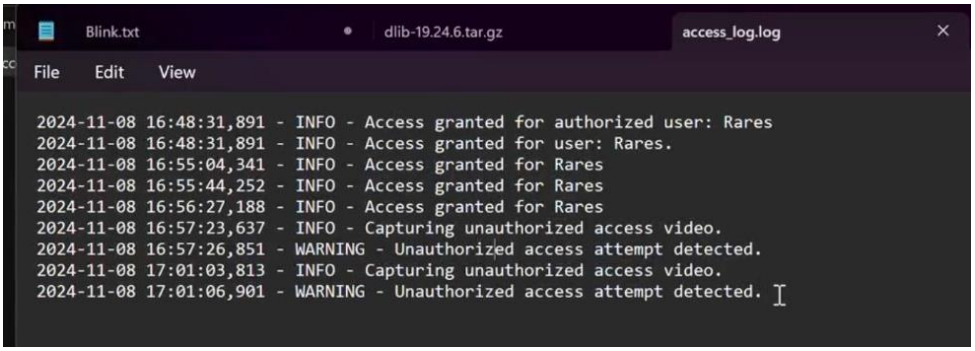
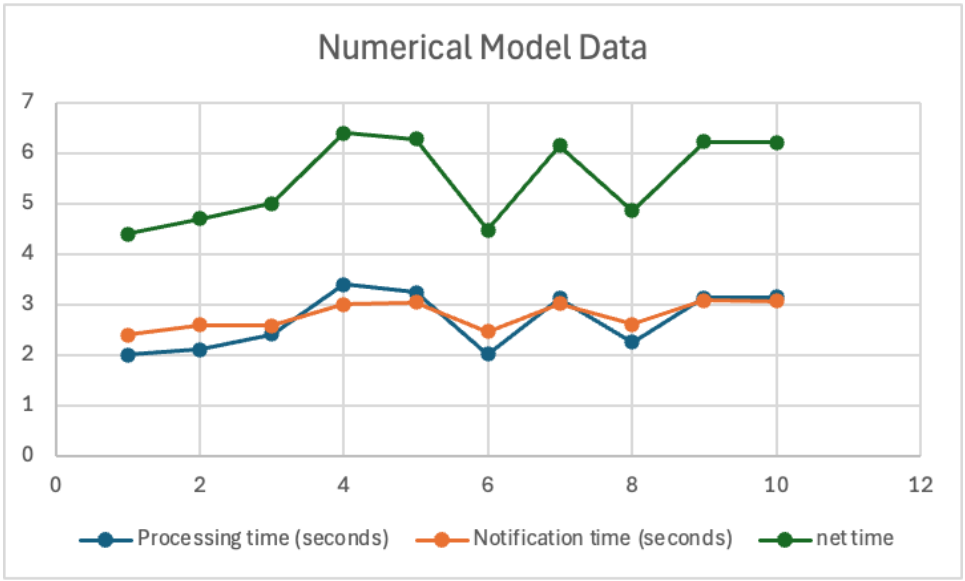


Figure 1: Image of Access Log after Consecutive Log in Attempts

Model

Using a numerical model, we could test what our final product could do without Shabodi's API's. Our model consists of a program coded in Python and a stopwatch application. We ran the trial ten times to gather a small sample of the amount of time it takes for the notification to be received once the information has been processed. We made two assumptions to simply our model. The first is that there was no delay in starting and stopping the stopwatch application. The second is that the system correctly processes the information every time. Below we have provided a graph of the information we gathered.

Graph 1: Response Time to Unauthorized and Authorized Access Attempts



From the data gathered we got an average time of 5.479 seconds. The model also revealed that it performs faster for authorized access events. This data can be used to estimate future time responses of the system.

Prototype Test

When testing the code, it worked as intended because when it recognized the user's face it would authorize access; whereas when it's done recognized the face of the user it would send a video of the unrecognize person to administration.

Prototype Test Analysis

The prototype works very consistently even though the frame rate can be a bit slow at times it does not affect the functionality of the product. We tried the system 5 times with an authorized user which the system recognized the face every time. We also tried to

Prototype Test Results

	Authorized user test	Recognized face
Trying the system	5	5

	Un-authorized user test	Recognized face
Trying the system	5	0

Feedback and Comments

From the third client meeting we received feedback that was considered for the overall design and in particular prototype two. The client has reiterated that our product needs to fulfill a specific scenario. Keeping this in mind we will define exactly the type of Enterprise we believe benefit from this system, and what constitutes a 'good notification'. Feedback on this prototype two from perspective users was that they enjoyed only receiving one notification rather than two when granted access. They expressed concern about not being able to get information about the location of the event. An idea we will implement in prototype three.

Definitions

Following the feedback from the client we have decided to define some terms, ideas and stopping points for our project.

- A Good Notification is a notification that contains relevant information in a timely manner.
- Relevant information consists of the following:
    - What is the event
    - Who caused the event
    - Location of the event
    - Time of the event
- Enterprise: A company that uses door pins as their primary way of restricting access to its employees.
- Unauthorized Access Attempt: is when someone tries to enter a room with a door code either using someone else's information or without a code at all.
- Minimal Viable Product: a product that is python code that can notify the administration of an unauthorized access attempt in an Enterprise that has a good notification.

Prototype Three Test Plan

Week of November 11th	Week of November 18th	Week of November 25th
Tracking case must be roughly coded. With at least 1-2 API's. Security camera should incorporate at least 2-3 API from Shabodi	Door access should be roughly coded.	All systems must work perfectly for design day.

Conclusion

To conclude in prototype two, we implemented a short video recording when someone is not recognized by the facial recognition software. Our prototype test was successful because we were able to access and deny access reliably. The feedback from our third client meeting helped us realize that we should have a specific use case which we know do.