Final Report

Thermal Scanning App

Colter Roche, Jose Bastardo

Senior Design 1
COP4934C.01
December 2, 2020

Contents

List of Tables

List of Figures

1 Introduction

Corserva is a managed IT service provider that develops and sells custom software and hardware solutions. Corserva's customers include hospitality and other in-person focused related businesses. Official CDC guidelines to businesses encourage taking steps to prevent the spread of Covid-19 among employees and customers, including temperature checks. Corserva has sponsored this project to produce a thermal screening solution capable of processing people quickly and without requiring user interaction to minimize additional contact.

The scope of this project is to produce an application and companion mobile application to measure and report high temperatures of people passing through the system. The thermal camera will use an auto calibration system to increase accuracy of readings. Mobile application to smooth the onboarding process and provide reports to users.

The business scope is to provide business with a kiosk and mobile app system that will make it easier for them to maintain safety precautions during the current Covid 19 pandemic while also increasing the speed in which staff and customers can enter their place of business.

2 Functional Decomposition

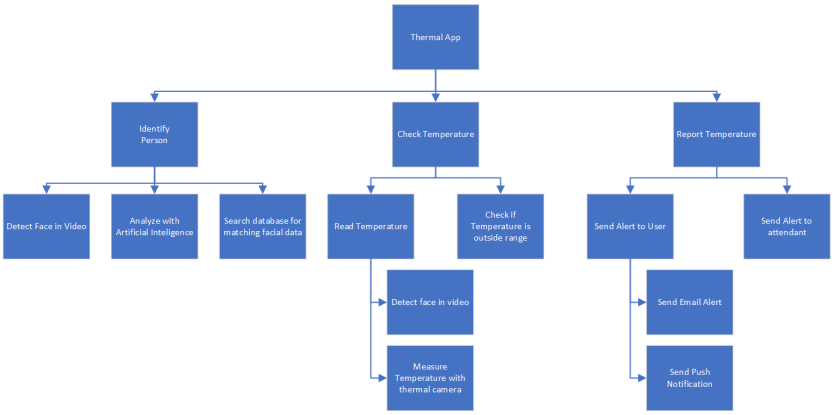

Figure 1: Functional Decomposition Diagram - Thermal scanner

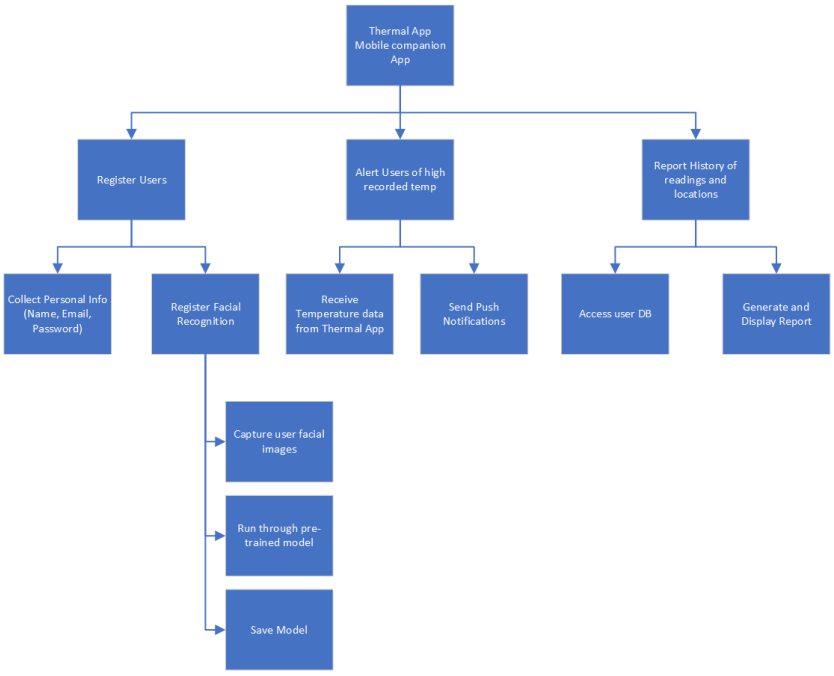

Figure 2: Functional Decomposition Diagram - Comapanion App

3 Data Flow and Application Structure

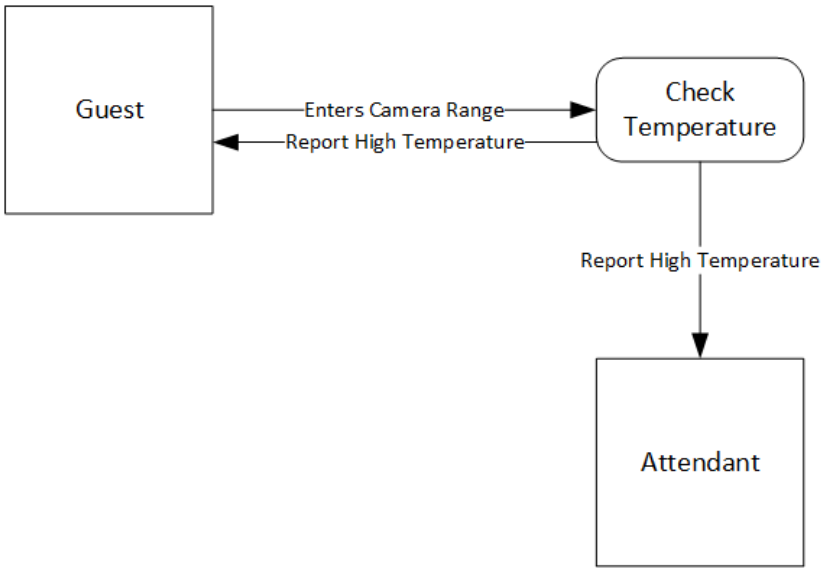

Figure 3: Top Level Data Flow Diagram

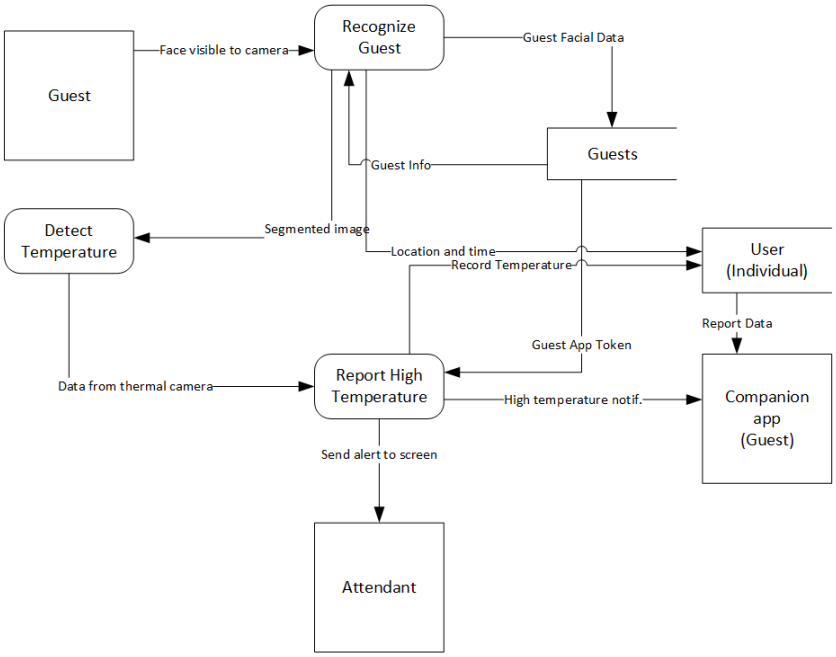

Figure 4: Level 1 Data Flow Diagram

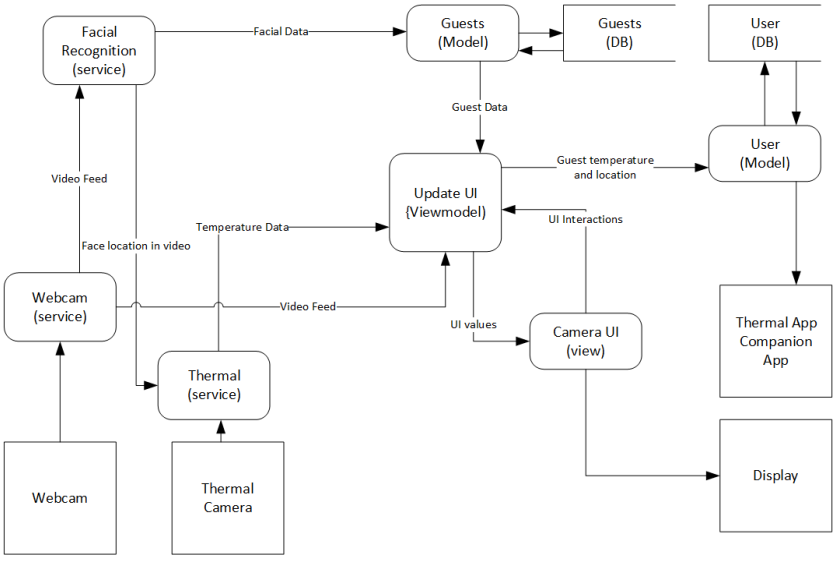

Figure 5: Level 2 Data Flow Diagram

4 User Stories and Tasks

Epic	User Story
Provide facial recognition for user identification and measure user temperature	As a user, I want the system to recognize me in under 2 seconds so I can save time
	As a kiosk attendant, I want to see a confirmation that the user is recognized, for security purposes
	As a kiosk attendant, I want temp measurements to be accurate withing a degree, so I do not have to check extra people/miss people
	As a kiosk attendant, I want the thermal camera to self-calibrate, to avoid the need for time consuming troubleshooting
Provide notifications and reports to users and attendants	As a kiosk attendant, I should be alerted if a registered user is detected with a high temperature to perform a manual temperature check
	As a kiosk attendant, I should be alerted if a person is not recognized as a registered user
	As a user, I should be alerted if my temperature is too high
Provide a user onboarding system	As a user, I want to register through a mobile app, for easier remote registration
	As a user, I want to register facial data through the app, so I can skip in person registration
	As a system admin, I want to have control over what information is collected from users and how long the data is stored, to increase security

Table 1: Epics and User Stories

5 Derived tasks and Pseudocode

The following tasks and Pseudocode were derived from the user stories listed in Table 1.

5.1 Detect faces in video

Face detection is used to determine whether a user has entered the view of the camera in order to begin facial recognition and measuring their temperature. This is done using the opencv library in Python.

```
1        Read camera stream
2        Use OpenCV to detect faces
3        Store face pixel array in var
4        Apply smoothing and post processing
5        Return section array
```

5.2 Calculate temperature discrepancy and adjust the camera

The thermal camera is known to lose accuracy depending on the surrounding temperature and weather conditions. A hot plate will be placed in the vision of the thermal camera and will be used to calibrate it. This will provide more consistent and accurate measurements.

Algorithm 5.2: Thermal cam calibration.

```
1        timer: every 3 minutes
2        on timer firing
3            select section of video containing calibration device
4            measure temperature
5            if measured temperature does not equal set value for temperature
6                subtract set from measured temp
7                adjust calibration by result
```

5.3 Measure temperature within the segment of the thermal image that corresponds to the user's face

Once the user's face has been segmented from the video stream, the rgb and thermal images need to be compared and the correct pixels measured for temperature. Because the resolution is different between the images, additional calculations are required. The data from the thermal camera is then returned.

Algorithm 5.3: Temperatur reading.

```
1        Receive array of pixel data containing face
2        Calculate corresponing pixel locations in thermal image
3        return thermal readings from those locations
```

5.4 Check if user is registered

Once a face is detected within the cameras vision, the application will use facial detection to store the users facial data and compare it to the facial data already stored in the database to check whether the has been registered through the system.

Algorithm 5.4: Facial info search.

```
1        Connect to database
2        Receive tuple with facial data files
3        Loop for length in tuple
4            If new data is equivalent to registered data
5                User is registered
6                Break loop
```

5.5 Send notifications to user and alerts to screen

If a high temperature is detected by the system, the application will send an alert through the screen on the tablet and to a kiosk attendee and the user. The system will also be able to send an alert to the attendee if the user is not registered.

Algorithm 5.5: Temperature notification

```
1        If high temperature detected
2            Display high temperature alert on screen
3            Connect to database
4            Search for email using user id
5            Send message to user email
6            Send alert to attendee email
```

5.6 Compare user temperature to past temps and check for too high temp

The application will consider the average temperature of the user by searching through the user's past scans and taking the average of their temperature

history. A limit will be set based on their average temperature. If the user's temperature passes that limit it will be considered a high temperature.

Algorithm 5.6: Check user temperature

```
Search database for user temperature history using user id
Receive array with past temperatures
Initialize counter at 0
Initialize sum at 0
Loop for size of array
    Sum is equal to sum plus tuple[counter]
    Average is equal to sum divided by size of array
    Limit is equal to average plus 3 degrees Fahrenheit
    If limit is greater than current temp
        User temp passes
    Else
        User temp fail
```

5.7 Register facial data through app

To reduce the amount of time the user needs to spend registering in person, and to improve registration times for large userbases, users will be able to register for the facial recognition from a companion app. Several images will be captured, the algorithm will be trained, and then the images will be deleted and the model stored in the database.

Algorithm 5.7: Register facial data

```
App receives access to phone camera
Use facial detection to detect face
If face detected
    Take pictures of user
    Convert pictures into facial recognition data files
    Send files to database
```

5.8 Register users through app

Users will be required to provide some identifying piece of information to create an account and eventually register facial recognition data

Algorithm 5.8: Register user data

```
User enters identifier
User enters password
Form entries are verified for format with regex
If info matches format
    Create new entry in database
    Store user info
```

```
7               Send user confirmation email
8           if user confirms
9               finalize user registration
```

5.9 Allow admin to control data requested and storage duration

User data collected for registration will vary based on use case. Most use cases will minimize data collected to an Employee ID number, but the admin may want to collect more data. To account for these variations, the application will allow the admin to set what information will be collected and how long the data will be stored.

Algorithm 5.9: Manage User data settings

```
1           Display list of options with on/off toggles
2           Display data storage duration with default value of 1 month
3           if at least one option is selected
4               enable save button
5           on save button clicked
6               set user data requirements
7               set data expiration time
```

5.10 Produce reports for organization and users

Companies and users will be able to view reports of temperature measured and time/location the data was collected. Company reports will list all users for a period of time, including times, location and temperature. Users will have access to similar reports, but only applying to them.

```
1           Request time period for reports (Default is 1 month)
2           if Company
3               query db for users recorded during that period
4               display sortable table listing users, locations and temperature
5           if User
6               query db for users recorded during that period
7               display sortable table listing locations and temperature for that user
```

6 Reflection

6.1 Lessons Learned

So far, the team has learned how important it is to communicate and organize tasks, especially between cross disciplinary teams. During the beginning of the project we were a bit confused and overwhelmed because roles and tasks were not clearly defined. Once the team started to document the project and got a better idea of what we had to do it became easier to assign tasks.

One of the applications that helped us assign and organize tasks was Trello. With Trello we were able to keep track of what everyone is working on and helped us share documentation for the project.

6.2 Areas to Improve

The team could have done a better job of communicating early in the project and discussing initial requirements such as what programming language we should use. It was not much of an issue because we were still in the early development phase of the project, but it did delay progress somewhat.

6.3 Concrete Steps

Our team will perform better in the future by communicating more with the Computer Engineering team. We need to meet regularly to know how much progress we have made on the project and discuss implementation and communication of certain functions. This will help to make sure the team is on the same page and the sponsor gets the application they want.

During the project we have not had a structured development cycle and it has slowed down our progress. We will adopt more of an agile development approach to increase productivity. This will make sure everyone has a task to work on. The team as well as the sponsor will know that progress is being made.

6.4 Personal Thoughts

6.4.1 Jose Bastardo

A personal weakness of mine that I discovered was being able to communicate with the sponsor and teammates. There were points where I could have asked more questions and discussed things more thoroughly. This possibly comes from some social anxiety, but it has improved as we have moved further into the project.

A strength that was revealed during this project was my ability to take lead in scheduling and organizing presentations and assignments. I feel like this semester I was able to work on my leadership skills while working on the project and off the project in my own life.

6.4.2 Colter Roche

Some personal strengths and weaknesses highlighted by this project involved personal struggles with time management and difficulties communicating effectively with the other half of the team. The change from .NET to Python was not clearly stated in a timely way, so a lot of time was wasted on our end planning the .Net version. We also did not have weekly meetings with the other half of the team, so the status of the calibration system is mostly unknown. Going forward, much more rigorous adherence to meeting scheduling and task completion will be necessary. The trello has only been lightly used, this will need to change for the implementation phase next semester.

One positive from the experience has been working closely with Jose to complete reports and presentations, and the process of writing and formatting all the documents. The move from using word to using LaTeXfor report and documentation writing has been a personal project, and this class was a great opportunity to utilize that skill.

7 Conclusion

The goals and requirements for this project were fairly straightforward, especially for the CS team: design and implement a UI, implement or otherwise handle facial recognition, and interface with the thermal camera system the CE team is working on. Unfortunately, communication broke down with the CE team, due to a lack of effort on our part and a (perceived on Colter's part) lack of professionalism and openness to different solutions.

Regardless, we thoroughly planned out the structure and functions of the system and have successfully implemented some of the basic requirements for storing user information. In addition, the programming language for the application has been finalized and work has begun on the UI and the facial recognition. The progress on the thermal calibration is somewhat in doubt, but at the very least, they have the thermal camera working and are learning the skills they need for image processing with OpenCV.

Next steps are to continue to investigate and test different options for the facial recognition system. The code provided from a previous project is essentially two other programs copied and pasted to work together, and the overall process is too slow. We will attempt to reduce the time by using one of the faster models included with OpenCV. UI implementation should be straightforward, but we have not started any of the designs for the mobile onboarding app.

One of the main focuses for next semester's work will need to be designing the UI and functionality for the app. The interfacing with the temperature and calibration system will hopefully be straightforward, but the CE team's status and class structure is unknown since the switch to Python. Regular inter-team meetings will also need to be scheduled.

8 Team Member Participation

Participation was split as follows:

- Colter Roche: 50%
- Jose Bastardo: 50%