Machine Learning for Human Biometrics

DESIGN DOCUMENT

SDDEC22-14

Client: JR Spidell

Advisor: Akhilesh Tyagi

Nathanael Morris/ Machine Learning Manager

Ritvik Maripally/ Security Manager

Ron Mei Hang Teon/ Database Manager

Yee Shen Teoh/ Hardware Manager

Zi-Jan Wong/ Machine Learning Manager

sddec22-14@iastate.edu

https://sddec22-14.sd.ece.iastate.edu/

Executive Summary

Development Standards & Practices Used

- IEEE 1012-2016/Cor 1-2017: Draft Standard for System, Software and Hardware Verification and Validation - Corrigendum 1
- IEEE 3652.1-2020: IEEE Guide for Architectural Framework and Application of Federated Machine Learning
- P3167: Standard for Secure Biometrics Device Interface

Summary of Requirements

- Analysis shall be performed in real-time in less than 1 second.
- An appropriate prompt shall be straightforward and clear and provided to the user to take a break.
- The camera shall capture images at a rate of 10 ms.
- The pupil detection algorithm shall consume 10-bit grayscale images, process images at a rate greater than 60 fps, analyze the frames under any lighting conditions and with various degrees of noise, and achieve an accuracy of 96%.
- Data shall be securely transferred to and from the device and the database.
- The database shall accept a stream of data from REMoDNaV, perform constant queries, provide data visualizations and have access to relevant parts of the system.
- A survey shall be used to analyze the accuracy of our model.
- Our project shall not violate any laws.

Applicable Courses from Iowa State University Curriculum

- COM S 363: Introduction to Database Management Systems
- CPR E 288: Embedded Systems I: Introduction
- COM S 309: Software Development Practices
- SE 329: Software Project Management
- SE 339: Software Architecture and Design
- CybE 231: Cybersecurity Concept and Tools

New Skills/Knowledge Acquired that was not Taught in Courses

- Convolution Neural Network (CNN) machine learning model
- Robust Eye-Movement Detection for Natural Viewing (REMoDNaV) algorithm
- InfluxDB Database
- Jupyter Notebook

Table of Contents

List of Figures

List of Tables

List of Definitions

- Convolution Neural Network (CNN): An eye or pupil-movement detection machine learning algorithm that detects the location of a pupil on an x-y plane at each frame using neural networks.
- Robust Eye Movement Detection for Natural Viewing (REMoDNaV): An eye-movement classification algorithm that takes in the output of the CNN model and identifies the type of eye-movement classification, i.e., saccades, fixation, error, frame by frame.
- Saccades: Quick, simultaneous movement of both eyes between two or more phases of fixation in the same direction.
- Fixations: When our eyes stop scanning a scene, holding the central visitation in place so the visual system can take in detailed information about what is being looked at.

1 Team

1.1 Team Members

- Nathanael Morris
- Ritvik Maripally
- Ron Mei Hang Teoh
- Yee Shen Teoh
- Zi-Jan Wong

1.2 Required Skill Sets for your Project

- Python
- Machine Learning
- Image Processing
- HW Design
- FPGA Design
- Linux/Unix
- Database

1.3 Skill Sets Covered by the Team

- C: Nathanael Morris, Ritvik Maripally
- C++: Nathanael Morris
- Python: Zi-Jan Wong, Ritvik Maripally
- Machine Learning: Nathanael Morris, Zi-Jan Wong
- Image Processing: Nathanael Morris, Zi-Jan Wong
- HW Design: Yee Shen Teoh
- FPGA Design: Yee Shen Teoh
- Linux/Unix: Nathanael Morris, Zi-Jan Wong, Ritvik Maripally
- Database: Ron Teoh

1.4 Project Management Style Adopted by the Team

We all take turns to be team leader by adopting the Democratic style as well as using an agile
project management methodology

1.5 Initial Project Management Roles

- Nathanael: Machine learning and image processing
- Ron: Databae management
- Ritvik: Leader and ssecurity
- Yee Shen: FPGA and HW design
- Zi-Jan: Machine learning and image processing

2 Introduction

2.1 Problem Statement

Our project is trying to solve the problem of truck drivers continuing to perform deliveries while under stress and fatigue that will negatively impact their performance. Through this project, we are aiming to reduce the chances that an error will occur during a delivery as a result of a trucker's stress levels or work fatigue.

2.2 Requirements & Constraints

2.2.1 Requirements

- The model shall perform analysis in less than 1 second (close to real-time).
- An appropriate prompt shall be provided for the surgeon to take a break or incorporate stress reduction behaviors.
- The Prompt provided shall be clear and straightforward to not serve as a distraction for the surgeon.
- The pupil detection algorithm shall analyze the eye movement data of a surgeon in any room condition and with various degrees of noise.
- The pupil detection algorithm shall consume 10-bit grayscale images.
- The pupil detection algorithm shall achieve an accuracy of 96%.
- The pupil detection algorithm shall process images at a rate of greater than 60 fps.
- The video camera shall capture images at a rate of 10 ms.
- The database shall accept a stream of data from REMoDNaV.
- The database shall perform constant queries.
- The database shall provide data visualizations.
- The database shall have access to relevant parts of the system.
- Data shall be securely transferred to and from the device to the database.
- A survey shall be provided after the surgery to verify the accuracy of our project.
- Our project shall not violate any laws.

2.2.2 Constraints

- If the video camera is stationary or moving which could lead to background noise
- Lighting condition of the room
- Glares on the pupil from light or from wearing glasses

2.3 Engineering Standards

- IEEE 1012-2016/Cor 1-2017: Draft Standard for System, Software and Hardware Verification and Validation - Corrigendum 1
- IEEE 3652.1-2020: IEEE Guide for Architectural Framework and Application of Federated Machine Learning
- P3167: Standard for Secure Biometrics Device Interface

2.4 Intended Users and Uses

The users who benefit from the results of the project are:

- Truck Drivers
- All Drivers
- Surgeons
- Patients
- Family/friends of patients and drivers
- Insurance companies

The project will be used as such:

- Predicting the condition of truck drivers
- Providing prompts for breaks
- Decreasing human error due to fatigue and cognitive overload / stress
- Providing data for analysis to continue improving the performance of the project
- Training new truck drivers

3 Project Plan

3.1 Project Management/Tracking Procedures

Our team is adopting a waterfall+agile project management style because we have to iteratively develop our project. For instance, we need to have the database up and running in order to store the output of the machine learning algorithm. In addition to that, we need to get the Ultra96 board setup before we can connect a camera and get real-life input. However, those tasks can be overlapped to a certain degree, so we think that a waterfall+agile project management style best suits our project. We will be mainly using Discord to communicate with each other, and GitLab issues to track progress of our code repository.

3.2 Task Decomposition

We broke down the project into different tasks, and decomposed them into multiple subtasks and tied them with the specific requirements to be satisfied.

Task	Subtask	Requirements
Hardware	- Connect the microcontroller with a camera. - Check the connection between the board and a monitor. - Work on FPGA for visual processing.	- Camera shall capture images at a rate of 10 ms. - Visual processing speed shall be at least 1.5 times.
Machine learning	- Set up the environment and install modules required. - Optimize the model and reduce error.	- Shall achieve at least 96% accuracy. - Shall process images at a rate greater than 60 fps.
Database	- Set up a database environment. - Integrate database into the rest of the system, - Perform testing on the database.	- Shall accept data from REMoDNaV and perform queries. - Shall provide data visualization.
Security	- Research the legalities of working with human biometrics. - Determine the authentication method and frequency to access the database.	- Shall grant database access to relevant parts of the system. - Shall verify key and grant/reject permission. - Shall not violate any laws.

Table 1: Task Decomposition

3.3 Project Proposed Milestones, Metrics, and Evaluation Criteria

We broke down the tasks and defined the milestones of each task and their respective metrics, as well as the evaluation criteria on each milestone.

Task	Milestones	Metrics	Evaluation Criteria
Hardware	Display image from the camera on a monitor with less than 10 ms delay.	Delay	We can check that using timecode view.
	Visual processing speed increased 1.5 times or more.	Speed	Check by processing a set of visuals without the FPGA, then again with the FPGA.
Database	Set up the database environment.	Database	Database is up and able to perform simple queries
	Receive and store constant input data streams of more than 10 kB/s.	kB/s	Run the system and make sure there's no disruption of data flow
	Visualize the data generated by REMoDNaV in less than 5 seconds.	Seconds	Generate various graphs using a visualization tool and get accurate representation of the collected data
Machine learning	Achieve accuracy of 96% and above.	Accuracy	Can cross-check based on a fixed dataset
	Process frames at a rate greater than 60 fps	fps	Processing a 5 second video of 30 to 60 fps and check if the output generates 30 to 60 rows of data respectively for each frame
Security	Verify key and grant/reject permission	Key	Test by providing different keys and making sure the system grants permission to only the correct key and rejects the wrong key
	Project does not violate any laws	Law	Thorough checking the laws to ensure no violation

Table 2: Project Proposed Milestones, Metrics, and Evaluation Criteria

3.4 Project Timeline/Schedule

Our project will be carried out throughout two academic semesters, so we break down the different tasks and goals we hope to accomplish each month. However, the project schedule might change due to delays from previous tasks.

Machine learning for Human Biometrics	Jan	Feb	Mar	Apr	May	Aug	Sept	Oct	Nov	Dec
Research regarding projects.	░	░								
Obtain data and code from client, develop project.		░	░							
Run machine learning algorithm through database			░							
Set up security			░							
Train ML algorithm				░						
Integrate database with hardware				░						
Setup connection between ML algorithm and database.				░						
Optimize ML algorithm and security					░					
Catching up regarding project							░			
Polish product							░	░		
Marketing design, advertisement								░		
More testing								░		
optimize system									░	
final presentation and documentation										░

Figure 1: Project Timeline/Schedule Chart

Timeline	Tasks
January	Researching databases, security, and machine learning for the project, purchasing microcontroller for the project, develop problem statements, users, and proposed solutions (done)
February	Obtain and understand machine learning algorithm and datasets from the client, develop use cases and bounce diagram for databases (done)
March	Run machine learning algorithm fully through the datasets without errors, set up security for database
April	Train the machine learning algorithm with larger datasets, integrate the database with the microcontroller, has machine learning algorithm send data to database
May	Optimize machine learning algorithm and test security of database
August	Continuation of the project from May, catching up on any backlog work
September	Polish existing features, fix bugs
October	Marketing design, create advertisement
November	Additional features, optimize system
December	Final Documentation, Final Presentation

Table 3: Project Timeline/Schedule Chart

3.5 Risks and Risk Management/Mitigation

There are multiple risks associated with our project, so we analyzed and broke them down into how different risks are tied to each task, the probability of the risk happening, the significance of the risk, and mitigation strategies to each risk.

Task	Risk	Probability	Impact	Mitigation Strategies
Database	InfluxDB crashing	0.2	High	Have a backup database to create redundancy
Machine learning	CNN model crashing	0.1	High	
	CNN model inaccuracy	0.5	Moderate	Intensive training and testing with larger datasets
Hardware	Ultra96 overheating	0.1	High	Attach a cooling system
Security	Data leak	0.1	High	Use encryption for our data

Table 4: Risks and Risk Management/Mitigation

With Security being a very important aspect of application, we must anticipate potential risks and mitigations in a physical and cyber sense of this project.

For the physical aspect of application we have some example scenarios:

Threat 1: Ultra96 is stolen
Mitigation 1: Implementation of physical tokens such as keyfobs or MFA is needed. There will be a tracker as well which informs admin's if the portal is being accessed from somewhere else.

Threat 2: Unauthorized usb or disk is inserted without permission
Mitigation 2: All ports within the ultra96 will have a whitelist and would need perms to have access to the data. If the client wants their data on a disk then it will notify and re-enter credentials in order to move data within the drive.

For the cyber aspect of our project we have some more example scenarios:

Threat 3: Malware attempts to enter the ultra96
Mitigation 3: There will be anti-malware executable software that is server grade which will let administrators know that this system will be quarantined till further notice.

Threat 4: Hacker gains access to ultra-96 remotely
Mitigation 4: All data will be encrypted and would need more authentication. The whitelist should be enough to stop unwanted users from accessing Ultra 96. The wifi will be disabled till further notice and the firewall will be updated once fixed.

Overall there will be physical security around the project where secure locations to conduct the access of data and correct practices such as having the whitelist of access to the wifi and making sure unwanted devices aren't connected to the ultra96 ports. For the cybersecurity side of the project, there will be an adequate firewall and anti malware executable to stop any threats within the devices. There will also be top of the line encryption where it is nearly impossible to decrypt without a key if the hacker gains access from the transmission lines.

3.6 Personnel Effort Requirements

According to the different tasks and subtasks we broke down in Section 3.2: Task Decomposition, we came up with explanations on the tasks to perform, amount of people needed, and hours required to complete them.

Task	Explanation	Person(s)	Hours
Database	Research and select database for project	1	8
	Set up the database environment	1	2
	Integrate the database to the rest of the system	3	6
	Write code to perform required functionalities	2	6
	Testing the database	5	8
Ultra96 microcontroller	Select and purchase microcontroller	1	2
	Do smoke test on microcontroller	1	1
	Select and purchase video camera	1	0
	Integrate database	2	4
	Set up microcontroller for project	1	8
	Integrate machine learning algorithm	3	6
	Test and polish	5	8
Machine learning	Set up the environment, install modules, and import datasets to run the algorithm	2	4
	Understand machine learning algorithm	2	4
	Train the machine learning algorithm with larger dataset	2	8
	Optimize machine learning algorithm	5	8
Security	Research ways to secure information transfer	1	2
	Integrate security in database	2	4
	Test integrated security	5	6
	Fine tune security	5	4

Table 5: Personnel Effort Requirements

3.7 Other Resource Requirements

Based on our targeted user, we might need to conduct tests on real truck drivers, then collect the data from truckers to analyze and improve the machine learning algorithm. We will conduct a survey to aid in analyzing the accuracy of our project.

4 Design

4.1 Design Context

4.1.1 Broader Context

Area	Description	Examples
Public health, safety, and welfare	Increase the well-being of truck drivers. Reduce the chance of accidents or mistakes during deliveries. Reduce mortality rate of truck drivers and the drivers around them. Could also help in the quality of new truck driver's training.	● This device aims to identify signs of stress or fatigue in a truck driver during deliveries and provide appropriate warning/advice to reduce the chance of human error.
Global, cultural, and social	It would increase other driver's faith in the truck driver, as the device aims to reduce human error on the road.	● Since there will be something to monitor the truck drivers' performance on the road, people can rest assured that one more risk has been eliminated - human error.
Environ-mental	The acquisition of raw materials used to manufacture the circuit board like the Ultra96 could potentially harm the environment in several ways.	● For the purpose of this project, the Ultra96 would ideally be purchased once. Therefore, the negative effects on the environment would be minimal. ● In terms of selling this device as a product on the market, the environmental impact would be more of an issue, since manufacturing the product would require energy, as well as raw materials.
Economic	Our project requires an Ultra96 circuit board, and it costs around $250, which can be somewhat expensive. If we include the camera modules and the cost of purchasing software, the cost of the project could be more than $400. However, most of the purchases are just one-time purchases, therefore allowing for maintenance and testing without needing to purchase additional materials.	● Due to the cost of this project, it is safe to assume that it is not affordable to the general public. ● However, since our targeted users are truck drivers working for a company, the cost of our device is very reasonably priced, and it is worth investing in since it can be a life-saving device.

Table 6: Broader Context

4.1.2 User Needs

Truck Drivers: They need a way to warn them of signs of stress/fatigue because driving while feeling stressed/fatigued greatly increases the risk of human error.

Trainers/Instructors: Instructors need a way to visualize the biometrics or emotions of a truck driver to use it as an example to train new truckers.

Truck Company Management: Some truck drivers tend to get fatigued or perform poorly under pressure, and that is not ideal. Management needs a way to evaluate a truck driver's capability in performing deliveries with high-value items or driving long hours.

4.1.3 Prior Work/Solutions

Machine learning algorithms for pupil detection and tracking exist in the market, and models such as Convolution Neural Network (CNN), Support Vector Machine (SVM), and Starburst are used widely. Our project is similar in terms of detecting the location of the pupil and providing classification of the eye-movement.

According to a research article titled "Using machine learning to detect events in eye-tracking data" by Zemblys et al., the shortcomings of most machine learning algorithms for eye detection is that hyperparameters, a number of settings, must be manually set and tweaked until the model is performing optimally. It is possible for the model to suffer from overfitting, where a model describes random error or noise instead of the underlying relationship due to having too few samples to train on, resulting in a model with poor generalization performance. (Feb 23, 2017)

The CNN model is significantly faster than other models such as SVM because we can run the model on multiple threads, and it is easily accessible on any device. A research by Yardimci and Ayyildiz on "Comparison of SVM and CNN classification methods for infrared target recognition" stated that the CNN model is about 7.7% faster compared to the SVM model. (Apr 30, 2018)

According to an article by Dertat titled "Applied Deep Learning - Part 4: Convolutional Neural Networks," CNN model would start to memorize the training data, and would fail to generalize new instances, causing overfitting and the validation performance becomes worse. (Nov 8, 2017)

According to the same article by Zemblys et al., the solution to overcome overfitting is data augmentation. Data augmentation is a method where we generate more data for training based on our current available data sets. It augments the original data sets by generating new and different examples by random transformation on existing data sets and increasing the amount of data we can train on the model, therefore reducing overfitting. Some common augmentation

include readjusting the size, rotation, brightness, exposure, contrast, and so on to the original data to generate new and realistic images so that the augmented data are learnable for the model.

As we develop our model further, we would be making a lot of tweaks to the hyperparameters so that our model has increased accuracy and would have reduced overfitting using data augmentation. Besides the training model, we would also be incorporating a database to store data collected from the output of REModNav for analysis and evaluation.

4.1.4 Technical Complexity

Our design consists of four major components which each use different engineering principles. The first component is a video input from a camera which goes into an Ultra96 board using FPGA. The next one is CNN model which is implemented as a machine learning algorithm that computes the location of the pupil in each frame of the video stream for pupil detection. Then, REMoDNaV determines if a certain frame from the output of the pupil detection algorithm is a saccade, fixation, smooth pursuit, or unknown. The final component is a database which stores the eye movement classifications at each frame. The stored data is then used to determine the user's condition.

There are many challenging requirements for each of the components of the system and the interfaces between components which are specific and attainable. We can assume that these requirements match current industry standards because our client has made recommendations for most of the requirements and approved them. Some examples of these requirements are:

- The pupil detection algorithm shall consume 10-bit grayscale images.
- The pupil detection algorithm shall achieve an accuracy of 96%.
- The pupil detection algorithm shall process images at a rate of greater than 60 fps.

4.2 Design Exploration

4.2.1 Design Decisions

- We decided to use an Ultra96 board because it is targeted for machine learning and connectivity for add-on sensors.
- We are using the CNN model as our pupil detection algorithm because it will be faster since we can run in on multiple threads to perform matrix calculations with tensorflow.
- We are using the REMoDNaV algorithm for eye-movement classification because it is one of the efficient algorithms for this classification and it effectively classifies eye-movements even in environments with lots of noise (e.g. a surgery room).
- We are going to use influxDB for our database, since it is perfect for custom monitoring, real-time analytics, and it is optimized for time-series data.

4.2.2 Ideation

For the fourth design decision above, where we chose to use InfluxDB, we also considered using two other databases. The other options we considered were MongoDB and PostgreSQL. We identified the possible options by searching for databases that would be appropriate for our project, and then compared all of them. By searching for a broad list of options and then narrowing it down to which one we wanted, allowed us to choose the best option for our project.

4.2.3 Decision-Making and Trade-Off

The process we used to identify the pros and cons between our options was to prepare a presentation which would be used to inform the rest of the team about the potential technologies we could use. Then at the end of the presentation, a list of pros were written out to summarize each of the options, which made it easier to visualize the differences between technologies. When deciding what database to use, our database manager, Ron Mei Hang Teoh, prepared and delivered a presentation reviewing each of these databases and what they are good for, and then collectively as a team we decided which one would be best for our project.

4.3 Proposed Design

- From a machine learning perspective, we have tried some basic training and understand how the CNN model identifies and guesses the image.
- From an embedded/hardware perspective, we have received the Ultra96 board from our client and began experimenting with some simple functionalities.
- From a database perspective, we've compared several different databases and decided to use InfluxDB.

4.3.1 Design Visual and Description

In this first visual, it gives an overview of each of our main components that were talked about in *Section 3.2: Task Decomposition*. The video input is stored in DDR memory, and then the pupil detection algorithm calculates the x and y axis of the pupil from the video at each frame, the REMoDNaV movement classification component determines what type of eye movement occurred at each frame, and then the results are stored in the InfluxDB.
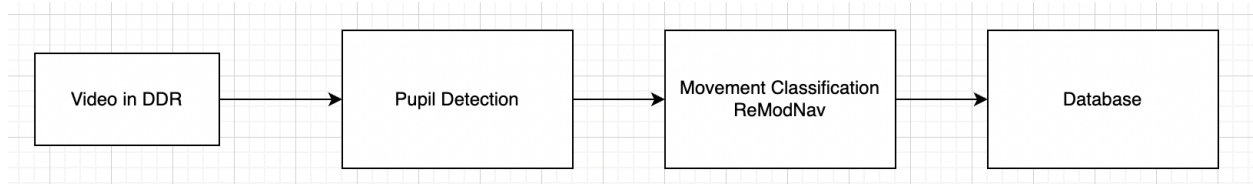

Figure 2: Main Components

In figure 3: Detailed Program Flow, we get an overview of the system. This figure shows each component of the system: the video camera, pupil detection algorithm, REMoDNaV, condition classification algorithm, and the influxDB. The whole system is within the Ultra96 device.
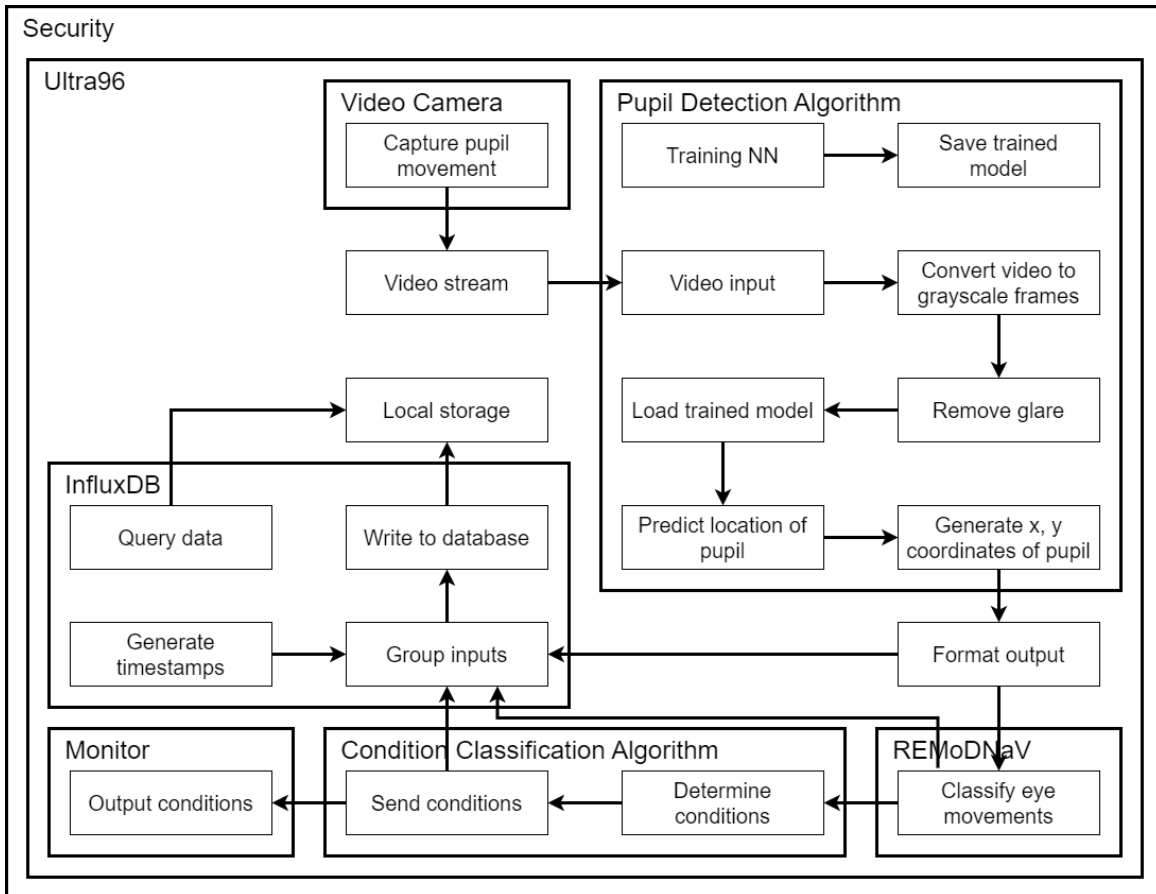

Figure 3: Detailed Program Flow

The third visual shows what the Ultra96 board will look like. The camera sends a video stream into the FPGA on the Ultra96, which is then stored in DDR Memory. The Ultra96 board contains four Application Processing Unit (APU) processors and two Real-time Processing Unit (RPU) processors. The APU processors send reset signals as well as communicate with the camera using an I2C serial communication bus.
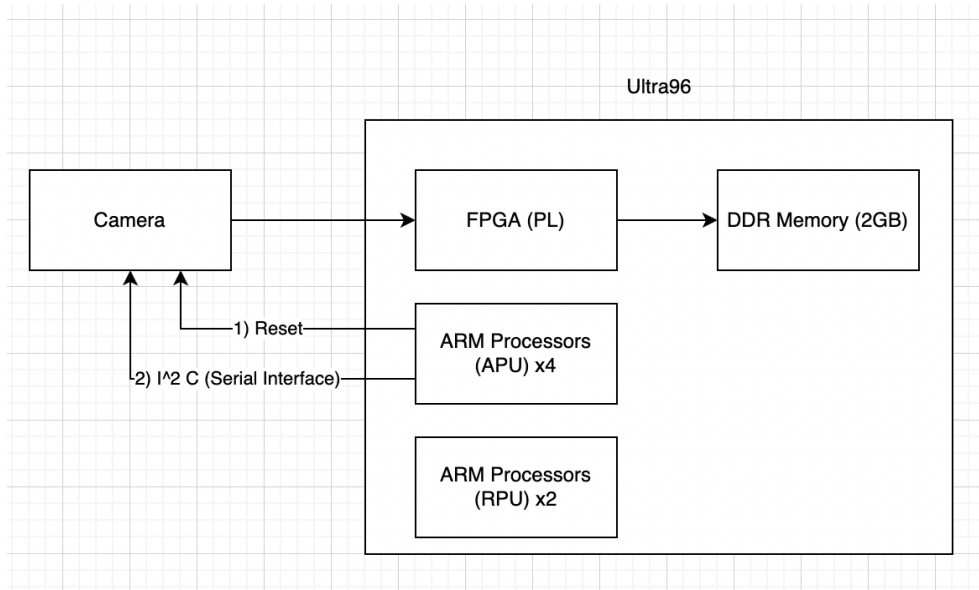
Figure 4: Components on the Ultra96 Board

The fourth visual representation shows how different components interface with the Ultra96 board. The camera uses the Mipi display serial interface to send video streams to the board. We are using the Mipi Display Serial Interface (DSI), a high-speed interface for displays, which is important for a real-time system. The PC uses a USB interface and programming interface to interact with the camera's datastream.

Figure 5: Various Components Interface with Ultra96

4.3.2 Functionality

This design is intended to operate in the cabin of a semi truck while a driver is performing a delivery or operating the vehicle. The design will take input from some camera, either a mounted camera in the truck or one that is on the truck driver's glasses. The camera will then send its data stream to the Ultra96 board through a wire, WiFi, or Bluetooth.
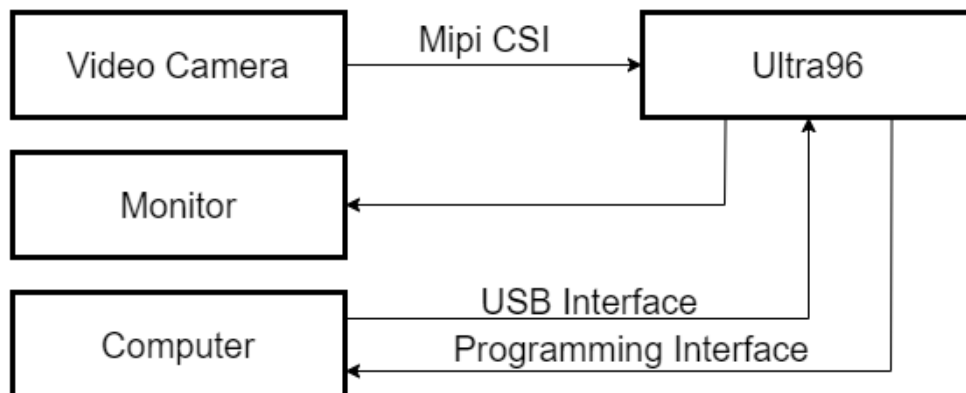
Next, the computations will be performed on the Ultra96 and the output will display if the driver is stressed or under work-overload in as close to real-time as possible, and then give recommendations on how to lower stress and fatigue levels. The current design satisfies functional and non-functional requirements because it will determine if the trucker is stressed within an environment which could have lots of noise or inadequate lighting in real-time.

4.3.3 Areas of Concern and Development

Given our current design, some concerns are: what camera will be used to collect the data in the truck cabin? How are we going to make the system run in real-time? How do we ensure that the predictions about the trucker being stressed and fatigued are accurate?

Our immediate solution to addressing the concern about which camera will be used is that we will use a camera that can have a wired input into the Ultra96 board. This camera will be used for testing our product and ensuring that it operates properly. Then, when the project is used, a different camera can be used based on what the trucker or the trucking company wants.

The system must run in real time to meet the requirement of receiving 60 fps from the camera. This would mean that each frame must be processed in < 16.7ms. We can address this problem by ensuring that we choose the quickest algorithms and interfaces to make sure data is processed quickly. Over time we will continue to optimize the machine learning classifications to increase accuracy and reduce runtime.

The last concern is how do we ensure that the trucker is actually stressed/fatigued or not? Our immediate solution to this problem is to talk to the truck driver after the delivery and review times where our system determined that they were stressed/fatigued. We can ask the driver if they were actually stressed/fatigued or not, and then use this data to help the machine learning algorithm learn.

4.4 Technology

For our hardware, we have considered two different kinds of boards. They are Arduino and Ultra96. Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino are flexible enough to be used in any kind of projects, and are widely used for fast prototyping. On the other hand, Ultra96 is targeted at a few specific applications, which includes machine learning. It is also widely used for the purpose of image processing. With Ultra96, we could also run our machine learning algorithm though the board itself. But with Arduino, we would need to connect it to an external PC or computing device since Arduino itself does not have enough processing power. Ultimately, we have decided to use Ultra96 since it is targeted for machine learning and can easily be used for image processing. Most importantly, they have enough processing power to run a machine learning algorithm on its own, making it easier to use and more compact for our project.

In terms of databases, we were considering two options: InfluxDB and MongoDB. One of the strengths of InfluxDB is that it is a time-series database, meaning that each value in the database is associated with a timestamp. MongoDB on the other hand is suited to store arrays and objects. Both databases have free versions, however, we decided to proceed with InfluxDB as it would be very useful if all data in our database has a timestamp associated with it.

We are using CNN for our pupil detection algorithm as it is widely used in the market as described in *Section 4.1.3: Prior Work/Solutions*. We can run CNN on multiple threads to perform fast matrix calculations to increase the processing speed. However, some weakness of the CNN model are training the model so much that it starts overfitting and negatively impacts the performance and accuracy on the new data; or not enough training that it underfits and does not capture the input and output of the variables used on the new data accurately.

For the classification of eye movements, we decided to use REMoDNaV, because it is very effective when coming to evaluating dynamic stimuli. Current algorithms are lackluster when it comes to viewing dynamic stimuli, and REMoDNaV was created with a velocity-based algorithm, so it is suitable for both dynamic and static stimuli. The algorithm also yields eye-movement event data even in suboptimal conditions, which is extremely important in a surgery room setting, since there can be a lot of noise and various light levels. The eye-movement classifications also include the timestamps in which they take place, which is needed to figure out when a user is stressed or overloaded.

For the cybersecurity aspect of our project we used multiple safety measures for our project. Our first major security consideration was adding encryption, mainly the PGP encryption which allows the data that's being sent from machine to database to be harder to decrypt and read in the wrong hands. Another major security measure that our team considered was how to add isolation methods onto the database so when it's not being used so cybercriminals don't have a way to access the machine at all virtually or off site. Last but not least is authentication measures,

when an employee needs to check into the database to access a patient's record, the database that is authenticating the employee should have the necessary steps to prove that the employee is who they say they are.

4.5 Design Analysis

The initial proposed design mentioned in *Section 3.3: Project Milestones, Metrics, and Evaluation Criteria* is not working as well as we would like. We are able to hit some of the milestones, but not all of them. Several unexpected errors were faced during the implementation of each part, resulting in additional time required for each individual part of the project. We observed that it would take time to learn all these new software and integrate them together last semester, and it was apparent that we underestimated the time needed. By reflecting on our overall progress so far, we are able to create a slide deck or notes, if needed, detailing the challenges we faced and the process we went through for the next group that would be working on the continuation of our project.

4.6 Design Plan

Our design plan is to consider the many different use cases for the truck driver using this device and incorporate it into our hardware device and our software. We have four main components within our project, and each use case will be more prevalent in certain modules. The four components are video in DDR, pupil detection, eye movement classification in REMoDNaV, and the InfluxDB database.

The truck driver will be using this device in a high-pressure environment, so they must receive a response within 0.1 second after the classification has been made that they are experiencing a negative response, e.g., stressed, overloaded, or fatigued. Truckers in training could also use this device to be aware of when they are experiencing negative responses so that they can know what events are causing distress, and how to reduce the distress. Drivers in training could also use this device to reflect on their emotions or behavior during deliveries so that they can identify areas of improvement, and how to react during certain situations.This will ensure that a surgeon takes action to reduce these side effects sooner rather than later. We will ensure that this requirement is met by streamlining the interfaces and modules so that they run efficiently and accurately.

In addition, our team brainstormed various scenarios where our finished device is compromised, and our mitigation strategies for the different scenarios. For example, the device was stolen from the truck, WiFi connection of the device was intercepted, The database that is stored on USB drive is stolen, a hacker gains WiFi access, a hacker gains access to the PC where the database is stored, a hacker installs data monitoring software onto the machine. The threats

are never ending and it's our job to remedy and prevent these threats when they come intact with our machine.

5 Testing

Testing is an extremely important component of our project, so we came up with testing plans to make sure that our project is working as intended and satisfy the requirements provided. Our test plans are divided into unit, interface, integration, system, regression, acceptance, and security testing, as well as the results of our test plans.

5.1 Unit Testing

Our pupil detection algorithm, mainly using Convolution Neural Network (CNN) for training and prediction, will be tested by comparing the actual x-y coordinates against the predicted x-y coordinates. Then, we can apply the Pythagoras Theorem to calculate the difference in distance. Basically, the larger the distance, the less accurate our CNN model is.

The eye-movement classification algorithm, REMoDNaV, will also be tested by comparing the actual classifications against the generated classifications. We were able to test this by breaking up the REMoDNaV movement predictions by each frame, and then breaking up the actual classifications by frame. Next, we compared each of the frames to see if they matched up or not.

As for Ultra96, We will run a smoke test to check if the hardware is good enough for the simplest software. After we have got most of the software working, run a stress test to make sure the hardware is able to handle heavier loads. The tools needed are a working software program and the Ultra96 hardware. Once we have those, we can start the Ultra96 testing.

For the database storage on the physical system, we will perform a penetration test to figure out bugs and safety measures of the system. Lastly, we will be doing a penetration test on the security unit as well as a risk assessment to determine the parts to be secured are protected.

5.2 Interface Testing

We have three interfaces in our design: the pupil detection interface, the eye movement classifier interface, and the database interface. The database interface can be tested by using unit tests, so we can ensure that the data being sent to the database is received in the correct format and order and to make sure that none of the data got corrupted.

The eye movement classifier interface can be tested by comparing it to data we already have. After we get the pupil locations from the pupil detection component, when we send it to the REMoDNaV movement classification component, we can ensure that the locations are

accurate by comparing it to correct data that was manually gathered. After testing these interfaces, we can make adjustments to the algorithms to lower the error percentage.

5.3 Integration Testing

The few integration paths we have in our design are the machine learning algorithm, the Ultra96 board, and the database. The integration of the machine learning algorithm can be tested by seeing if the algorithm has no problems taking in the provided input, and generates a correct output format. In terms of the integration of the Ultra96 board, we need to test if the board is able to communicate with the camera to generate input and store it in the database. For database integration, we need to test if the database is able to receive input from the camera and is able to perform read/write operations.

5.4 System Testing

We will be manually providing video footage of eye movements and confirming that the system acts appropriately. All the interface and integration tests will be applicable to the system test as well.

5.5 Regression Testing

We would first create a duplicate and make sure we can roll back to our previous working model, then add new functionalities and make sure old ones do not break, and last we combine them. Besides, more testing will be required when new additions are added in to ensure that old functionalities do not break after the addition. The main implemented critical features that we need to ensure do not break are the CNN model, ReModNaV classification algorithm, tables in InfluxDB, Ultra96, as well as the security of our model.

5.6 Acceptance Testing

We will be running a few of the tests using some sample video footage that is provided by the client. After that, the output data, run time, process time, and any other observation will be recorded. Then, we will verify with the client that the design requirements are satisfied.

5.7 Security Testing

For security purposes in this project, we will be doing a risk assessment on exactly what needs to be protected and potential risks that can be used against us or future clients that may use our service/product. When the risk assessment is done, the initial phase of tweaking to increase safety measures will be implemented then will proceed with penetration testing acting as the

perpetrator in order to breach our system. This process will repeat till the majority or if not all safety measures are implemented.

5.8 Result

We have not done much testing yet as we are still in the process of developing our project. However, we are testing each component before they are integrated into the main components to ensure compliance with the requirements by observing and obtaining data from each test. From those data, we would be able to determine if the requirements are satisfied. Furthermore, we can start building up more testing for our devices once all components are finalized, since we will be more familiar with our design and understand what kinds of testing are important for our implementation.

6 Implementation

Our implementation plan for this semester was to improve the accuracy and to process live video by the pupil detection algorithm. Next, we tried to increase the accuracy of the REMoDNaV algorithm and lower the error between the real and predicted eye movement classifications.

We also worked on streaming data through all the components. We will have to take live data as an input into a camera, and then run it through the classification algorithms and then into the database. So far, we have accomplished streaming a video file of the eye into the pupil detection algorithm and eye-movement classification algorithm to get an output.

With testing, we will have a survey that contains a list of questions that can help us determine the accuracy of our model after the truck driver has a delivery. Questions would be "Are you feeling stressed at [timestamp] time?" or "Do you think the prompt given [prompt] was helpful?" are useful for us so that we can improve our model.

Steps that should be implemented in future groups would be to create a model using the eye movement classifications to predict if the user is fatigued or stressed. We have research that shows how to determine the user's condition using eye movement classifications, so the next group can use that information to assist them with the next iteration.

7 Professionalism

This discussion is with respect to the paper titled "Contextualizing Professionalism in Capstone Projects Using the IDEALS Professional Responsibility Assessment", International Journal of Engineering Education Vol. 28, No. 2, pp. 416–424, 2012

7.1 Areas of Responsibility

We decided to look into the comparisons between the area of responsibility of NPSE and IEEE code of ethics as well as how they differ from each other.

Area of responsibility	Definition	NSPE Canon	IEEE Canon	Comparison between NPSE and IEEE
Work Competence	Perform work of high quality, integrity, timeliness, and professional competence	Perform services only in areas of their competence; Avoid deceptive acts.	To maintain and improve our technical competence and to undertake technological tasks for others only if qualified by training or experience, or after full disclosure of pertinent limitations.	NSPE has a similar definition to IEE Canon. NSPE defines work competence as only working in areas that you're competent in, while IEEE Canon defines it as only working in areas that you are qualified or have experience in.
Financial Responsibility	Deliver products and services of realizable value and reasonable cost	Act for each employer or client as faithful agents or trustees.	to reject bribery in all its forms	NSPE defines financial responsibility as being faithful and delivering products at a reasonable cost while IEEE defines it as rejecting all forms of bribery.

Table 6: Comparing NPSE and IEEE Code of Ethics

Area of responsibility	Definition	NSPE Canon	IEEE Canon	Comparison between NPSE and IEEE
Communication Honesty	Report work truthfully, without deception, and understandable to stakeholders	Issue public statements only in an objective and truthful manner; Avoid deceptive acts.	To be honest and realistic in stating claims or estimates based on available data.	NPSE and IEEE differ a bit, where NPSE requires public statements to be truthful while IEEE must be honest and realistic in stating claims and should be backed up by data to prove.
Health, Safety, Well-being	Minimize risks to safety, health and, well-being of stakeholders	Hold paramount the safety, health, and welfare of the public.	To accept responsibility in making decisions consistent with the safety, health, and welfare of the public, and to promptly disclose factors that might endanger the public or the environment.	The NSPE and IEEE have similar triads for this area. But the IEEE has additional clauses that specify disclosing factors that might endanger the public.
Property Ownership	Respect property, ideas, and information of clients and others	Act for each employer or client as faithful against or trustees.	To seek, accept, and offer honest criticism of technical work, to acknowledge and correct errors, and to properly credit the contributions of others.	NSPE defines property ownership as respecting the owners of any properties, ideas, or information, while IEEE defines this as to credit anything that doesn't belong to yourself.
Sustainability	Protect environment and natural resources locally and globally	N/A	N/A	There is no code of ethics that corresponds to the area of sustainability for both NSPE and IEEE.
Social Responsibility	Produce products and services that benefit society and communities	Conduct themselves honorably, responsibly, ethically, and lawfully so as to enhance the honey, reputation, and usefulness of the profession.	To be honest and realistic in stating claims or estimates based on available data.	The NSPE version is more about public images, but the IEEE version focuses more on being honest and informative.

Table 7: Continuation of Comparing NPSE and IEEE Code of Ethics

7.2 Project Specific Professional Responsibility Areas

We rank (High, Medium, Low, and N/A) each of the professional responsibility areas in *Table 6: Comparing NPSE and IEEE Code of Ethics* according to how relevant they are towards our project:

- Work Competence: This responsibility area does apply to our project's professional context because our project will need to be of high quality, or the device may not be as accurate as needed, leading to incorrect predictions. Timeliness is also important with our project because our whole team is relying on other team members to get work done. If some members don't work competently, it could delay the project. [High]

- Financial Responsibility: Our team is responsible for sticking to the budget given to us by our client. We will be delivering our product at a reasonable cost since we will only be using a few hardware components that are provided to us by our client. [High]

- Communication Honesty: This responsibility area is VERY important to our group as this project requires a lot of time and communication. If information is not being communicated properly, it could lead to problems and distrust by the client which could end very badly for this project and class. If information is also not truthfully said, it could lead to imbalances and misunderstandings in the group which also could be a major problem. That's why we need to keep honest and consistent communication throughout the project. [High]

- Health, Safety, Well-Being: This responsibility area is not super applicable to our project, since there are not many health and safety risks present. For the well-being portion of this area, we can ensure that members are being helped when working on a challenging part of the project. [N/A]

- Property Ownership: This responsibility is somewhat important for our project since we are being loaned hardware equipment from our advisor and it is our responsibility to take good care of it. We also must be respectful of the ideas and information that our team members and clients share with one another by having healthy conversations. [High]

- Sustainability: As a team, our project really doesn't affect the environment as much as other things but the one thing we could do is not waste or reuse resources within our project. [N/A]

- Social Responsibility: This is similar to communication honesty, the team has a responsibility to produce a quality product from the resources given by the client. We also have moral standards/responsibilities for this class so we must be honest and thoughtful for what we do. Furthermore, we are working on a project that would help the general public determine their health with eyeball movements, which would greatly benefit society now since most people are overworked and have jumbled-up sleep schedules. [High]

7.3   Most Applicable Professional Responsibility Area

One area of professional responsibility that is important to our project, and for which our team has demonstrated a high level of proficiency in the context of our project is the social responsibility where we produce products and services that benefit our society and communities.

Our project is to develop a human biometric system that tracks the eye movement of users and determine if they are stressed or fatigued and will determine if the user needs to rest or any action is required to improve the user's well-being. Because the human eye is very vulnerable and valuable, we should do everything we can to protect it.

Our team constantly thinks of ways to have more data to produce an accurate result, ensuring that our product is consistent and accurate so that it can reliably give advice to anyone who uses our product. Our project started out as something that would be focused on helping society, so it was already the main focus since the start of our project. In conclusion, our high social responsibilities would benefit society and communities.

8 Closing Material

8.1 Project Evolution

Last semester, we were mostly just exploring Ultra96 on the factory image, so not a lot of work is done in terms of integrating the whole project. This semester, we realized that there are issues with downloading libraries and dependencies, so we moved on to using pyng image for Ultra96. After successfully displaying images and videos on a display monitor, we moved on to integrating the algorithm and database into the board. Unfortunately, we are not successful in this regard as there are still issues with downloading some required libraries, such as Tensorflow, which requires more storage space than the Ultra96 had.

In the last semester, we spent most of our time researching and understanding how machine learning, especially CNN, work, and then spent even more time experimenting and understanding how to train the CNN model and to use it for predicting the x-y coordinates of pupils. Not much work was put in developing the model. In this semester however, we use the Pythagoras Theorem to calculate the difference between the distance of the actual x-y coordinates and the predicted x-y coordinates to determine the accuracy of our model. Then, we saved several CNN models to be used for experimenting and integrating into the Ultra96. However, we noticed that Ultra96 cannot install Tensorflow, so we opted for Tensorflow lite instead, but there are some errors installing Tensorflow lite onto Ultra96. In addition, we tried removing the glare in the pupil and iris by converting the frames to grayscale, then masking the white pixels, and removing the glare. In our surprise, the results made a huge improvement.

During SE 491, we mostly focused on which algorithms will be used in this project to figure out the eye classifications. Lots of research was done verifying that the REMoDNaV algorithm would work for our project, and how this algorithm could be implemented. We also learned about different eye movement classifications, and about the impact that fatigue and stress can have on eye movements. This section evolved this semester into actually using the algorithm and making it more accurate. We worked on reducing the error based on the test and real data provided, and dove into how different parameters affect the output of the movement classification predictions.

During 491, a lot of research was done to identify the potential database that we could use, and we narrowed it down to InfluxDB, PostgreSQL, and MongoDB. We decided to go with InfluxDB mainly because it is optimized for time-series or time stamped data, which is what we're working with in this project. We also looked into various use cases for the database, and besides storing the output from REMoDNaV and CNN, we decided to use the collected data to perform post-event analysis. For example, if a car accident happens while the system is active, we could tell if it is due to driver error or something that is purely an accident, based on the eye movements of the driver. This semester, we spent most of our time integrating the database into the Ultra96 board. A lot of time was spent prototyping on a different machine/environment since

we only have one Ultra96 Board. Eventually, we installed the database on the Ultra96 board, but we had trouble getting the server and client to work together.

For the security side, It was mainly just mock ups for the 491 semester and learning what scenarios would happen during the course of our project. This was more of a relearn of everything that was taught in college and how to apply it to a real world project. Due to the security portion of the project being a one man army, it was difficult to time manage with other stuff but for 492 we decided to focus more on following protocols and have set and stone methods that would take place but no real testing. If there was a third semester then the actual implementation would occur starting off in virtual machines when then would be applied into the Ultra 96.

8.2 Discussion

We have not implemented our product yet. So far, we are working on different parts of the project before putting it all together. Those parts consist of hardware, a machine learning algorithm, eye-movement classification algorithm, a database, and security. While we are working towards completing different parts of the project, we have been meeting the requirements in each area. We will continue to check if the requirements are met when working towards our final product to make sure that our overall product is as specified.

8.3 Conclusion

For the hardware portion of our project, we used Ultra96 as the computer for the project. First, we are using the factory image provided with Ultra96. Then, after finding several errors when trying to use the image with our project, we decided to move on to using the pynq images for the Ultra96 board. Most of the issues are solved when we switch to pynq, and we are able to display images and videos from the memory to a display monitor. However, during the end of the semester, there is still an error when we are trying to integrate the algorithm and database into the board. Currently, we are going to backup the microSD card into an image and send it to our client, so that the next group he works with can flash the images into their microSD card, and start from where we left off.

We used CNN as part of our pupil detection algorithm. A video stream is fed into the pupil detection algorithm, then converted into grayscale and removing glares. Next, the CNN is used to predict the x-y coordinates of the pupil, generating a tab-separated output file to be fed into REMoDNaV. We noticed that removing glares from the pupil and iris resulted in a more accurate model. Pythagoras theorem is used to determine the difference between actual x-y coordinates and predicted x-y coordinates, and the larger the difference, the lower the accuracy.

For the eye-movement classification algorithm, we used REMoDNaV to convert the x-y coordinates from the pupil detection algorithm into eye movement classifications. To improve

the accuracy of this algorithm, we had to find the error between the real data that was provided and the output of the algorithm. There were 17 parameters in REMoDNaV that could be adjusted to improve the accuracy. The parameters that had resulted in the lowest error when changed were sampling rate, px2deg (factor to convert pixel coordinates to visual degrees), sav-gol length, and min-saccade-duration. We did research to find out how these classifications can be used to determine fatigue and stress levels. Increases in blink duration/frequency and a decrease in peak velocity and saccade duration shows that individuals are becoming fatigued. When users have longer fixation periods on objects (less facades), it is an indication that they are using higher levels of focus which can lead to cognitive overload and stress.

In terms of the database, we used InfluxDB to store the pupil coordinates from the CNN's output, and the eye movement classifications from REMoDNaV. Initially, we had the database set up in a local computer that has a MacOS environment as a simple prototype. We eventually moved on to prototyping the database on a Linux Virtual Machine on a Windows Laptop. After that, we finally integrate the database into the Ultra96 Boad itself. The reason for prototyping is because we only have one Ultra96 board, and since everyone needs the board to do integration/testing, it is better if we make sure we have a semi-working database before integrating it into the board, as we want to prevent holding on to the board for too long. We've also successfully set up the server and client part of the database on the Ultra96 board. During the database installation process, we encountered several problems trying to install the libraries and files needed for InfluxDB to run on the board, since the Ultra96 board only comes with minimum libraries and utilities installed. We were unable to get the client and the server to work together due to issues with file path permissions. Overall, the database has been integrated, we just need to figure out how to get it to read and write data.

For the security portion of our project, we have come up with methods that will help secure data during transition and while kept in storage at a server. Due to time constraints and resource allocation, the team was unable to have any real testing on the security portion but as a team we decided that Public Key Infrastructure and OpenPGP were best suited methods for this project. OpenPGP is a good standard on delivering data from and to the server as transmission lines are susceptible to tapping so to counteract this we have a public and private key which is created when the client requests data with authentication with MFA on the ultra96. For the Public Key Infrastructure it is another alternative to OpenPGP but it helps with private key being hard to crack which is another counter to man in the middle attacks. Overall the security methods in place can change due to the nature of threats being more meticulous so this could change in reliability for the future but as of right now these two methods are overall the most reliable going forward with this project.

This project is going to be passed to another team to continue its development. Some suggestions we have for the new team are to add an algorithm that uses the eye-movement classifications to predict if the user is fatigued or stressed, include real-time graphing, streaming data through the system in real time instead of in blocks, and finding better hardware images or a more effective method to use Tensorflow on the Ultra96

8.3 References

A. H. Dar, A. S. Wagner, and M. Hanke, "REMoDNaV: Robust Eye-movement classification for dynamic stimulation," *Behavior Research Methods*, vol. 53, no. 1, pp. 399–414, 2020. Available: https://link.springer.com/article/10.3758/s13428-020-01428-x

Dertat, A. (8 Nov, 2017). "Applied Deep Learning - Part 4: Convolutional Neural Networks." *Towards Data Science*. Available: https://towardsdatascience.com/applied-deep-learning-part-4-convolutional-neural-networks-584bc134c1e2#:~:text=The%20main%20advantage%20of%20CNN,CNN%20is%20also%20computationally%20efficient

J. Chandran and A. Joseph, "2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)," in *Eye Gaze as an Indicator for Stress Level Analysis in Students*, 2018.

Mccormack, J., Beyerlein, S., Davis, D., et al. "Contextualizing Professionalism in Capstone Projects Using the IDEALS Professional Responsibility Assessment." *International Journal of Engineering Education*, vol. 28, no. 2, pp. 416 - 424, 2012.

"Ultra96-V2," *Avnet*. [Online]. Available: https://www.avnet.com/wps/portal/us/products/avnet-boards/avnet-board-families/ultra96-v2/ultra96-board-family/?CMP=AVNET-AMER-PPC-GG-USA-English-AVE14-DSA-1658060323-62203480245-122018&gclid=Cj0KCQjw6pOTBhCTARIsAHF23fLWYiI0vHfm8BuELrqo1gk494Z5A09KFl2vmPrdUV7Uo_pLC72iue0aAruIEALw_wcB. [Accessed: 24-Apr-2022].

Yardimci, O., & Ayyıldız, B. C. "Comparison of SVM and CNN classification methods for infrared target recognition." *SPIE Proceedings*, vol. 10648, no. 4, 30 Apr, 2018. Available: https://doi.org/10.1117/12.2303504

Zemblys, R., Niehorster, D. C., Komogortsev, O., et al. "Using machine learning to detect events in eye-tracking data." *Behavior Research Methods*, vol. 50, pp. 160 - 181, 2018. Available: https://doi.org/10.3758/s13428-017-0860-3

8.4 Appendices
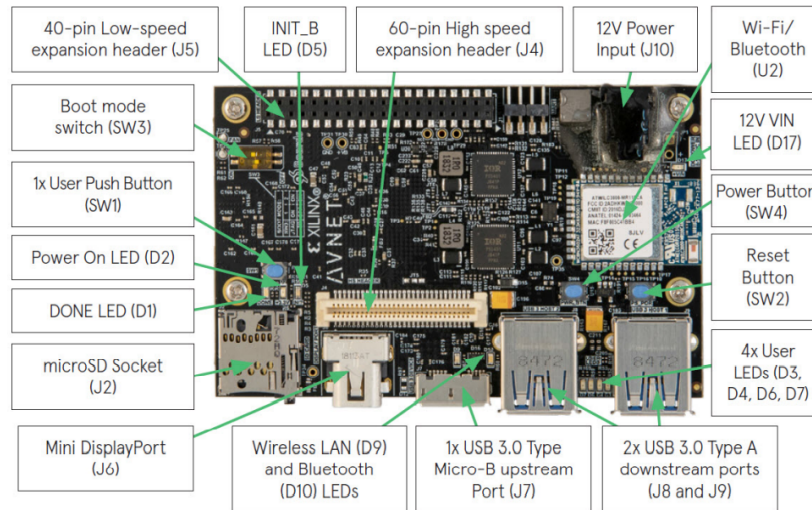
8.4.1 Operation Manual

Figure 6: Ultra96 Topology

The image above shows the expected out-of-box configuration along with various topology components shown.

To run the Ultra96:

1. Connect Ultra96 to 12v Power supplier in J10.
2. Connect a display monitor to the Mini DisplayPort, J6.
3. Connect a mouse and keyboard to the USB Type A ports, J8 and J9.
4. Press the Power Button, SW4.
5. Wait for the PYNQ environment to boot up. A Chromium browser should show up automatically. Browse to 192.168.3.1 on the search bar. If asked for a password, enter **xilinx**. A jupyter notebook tab should show up after that.
6. For WiFi connection, go to the 'common' folder on the jupyter notebook, then click on 'wifi.ipynb' and follow the instructions given.
7. The Ultra96 is now ready to download applications or run program as desired.

To run the pupil detection algorithm:

1. Install Python with version of at least 3.6.13.
2. Install Anaconda or Miniconda.
3. Install Tensorflow with version of at least 2.6.2.
4. Install Jupyter Notebook.
5. Activate a Tensorflow environment using Anaconda or Miniconda.
6. Install necessary libraries and dependencies such as tensorflow, cv2, pandas, numpy, and so on.

7. Open Jupyter Notebook to run algorithms.

To train and save a CNN model:
1. Open Jupyter Notebook.
2. Tweak parameters as needed.
3. Run algorithm to train CNN model.
4. Save the CNN model using *model.save()*.

To demo the pupil detection algorithm:
1. Open Jupyter Notebook
2. Load the CNN model using *tf.keras.models.load_model('<path>')*.
3. Can be cross-checked using *model.summary()*, which shows the model type, output shape, total parameters, trainable parameters, non-trainable parameters, and so on.
4. Run algorithm to open the camera and output predicted x-y coordinates.

To test accuracy of the CNN model:
1. Open Jupyter Notebook.
2. Run algorithm to plot scatter graph with all the x-y coordinates from the actual data and predicted results.
3. Run algorithm to calculate the difference between actual data and predicted results.

To run the REMoDNaV algorithm separately from the system:
1. You can install REMoDNaV using pip install remodnav
2. Remodnav has four required parameters, and many optional parameters that can be used
   a. Infile: x-y coordinates in a .tsv file with column 1 being the x coordinates of the pupil and column 2 being the y coordinates of the pupil
   b. Outfile: File that results will be printed to after algorithm is run on infile
   c. Px2deg: Factor to convert pixel coordinates to visual degrees. The following formula can be used for this factor: *degrees(atan2(.5 * screen_size, viewing_distance)) / (.5 * screen_resolution)* where the screen size is the size of the eye video file
   d. Sampling rate: Sampling rate of the data in hertz. Should be equal to the number of frames per second of the eye video use in the pupil detection algorithm

To run the REMoDNaV algorithm with the system:
1. You will have to ensure that all dependencies from REMoDNaV are installed on your python interpreter.
2. Open Jupyter Notebook
3. Change the file that the variable txt_file uses to be a .txt file with the x-y coordinates of the eye (leave this as 'output.txt' if you want to use the results from the pupil detection algorithm)

4. The results of remodnav will be printed to a file called remodnav_output.tsv and a graph called remodnav_output.png will be created to show the coordinates of the pupil at different timestamps.

Database
Security (Idk if this can be implemented somehow?)

8.4.2 Alternative/ other initial versions of the design

For all the different databases that are out there, we narrowed it down to three databases - InfluxDB, MongoDB, and PostgreSQL. After researching on the strength, weaknesses, and features of each database, we found that MongoDB is perfect for mobile app development and real-time analytics, PostgreSQL is most suitable for building resilient and secure software environments, and InfluxDB is optimized for metrics collection, real-time analytics, and time-series data. We ultimately decided on InfluxDB, since its strengths and features is the most suitable and can benefit our project the most.

Some alternative security measures were to have a standard one encryption type like AES only throughout the entire transmission and data storage but the security world is always changing so with that there is no complete right answer. There is also another possibility to add in a 3rd party software in order to combat malware and threats instead of making a complete program from scratch.

Some alternative hardware we thought about is using arduino. Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino are flexible enough to be used in any kind of projects, and are widely used for fast prototyping. But with arduino, we would need to connect it to an external PC or computing device since Arduino itself does not have enough processing power. With that considered, we ultimately decided to use Ultra96 since it is essentially a computer, and would have enough processing power for machine learning and can easily be used for image processing.

8.4.3 Other Considerations

Another way to calculate error is to only use relative errors in the pupil detection or eye movement. The eye movement can be classified such as a fixation, saccade, smooth pursuit, and so on. Fixation can also be the eye circling around some pixels in a circular motion, so, the relative error for fixation is to only calculate the coordinates within the circle. The outer coordinates that circulates the circle can be ignored as we already know the pupil is in a fixation, and it should concentrate around the few pixels within the circle.

Saccades on the other hand, are burst movement from one location to another, so the relative error for saccade can be calculated using only the starting and the ending frame of the

saccade. The movement in between starting and ending frames can be ignored as they are not that important. Similarly, smooth pursuit can have the same strategy for calculating relative error.

However, we gave this strategy a try and found out that there were some issues. First, how can we make sure that fixation always occurs in a circle? Second, the saccades almost only occurs in a frame to about ten. This would be hard to calculate because we can have a starting frame but not an ending frame for every single saccade frames.
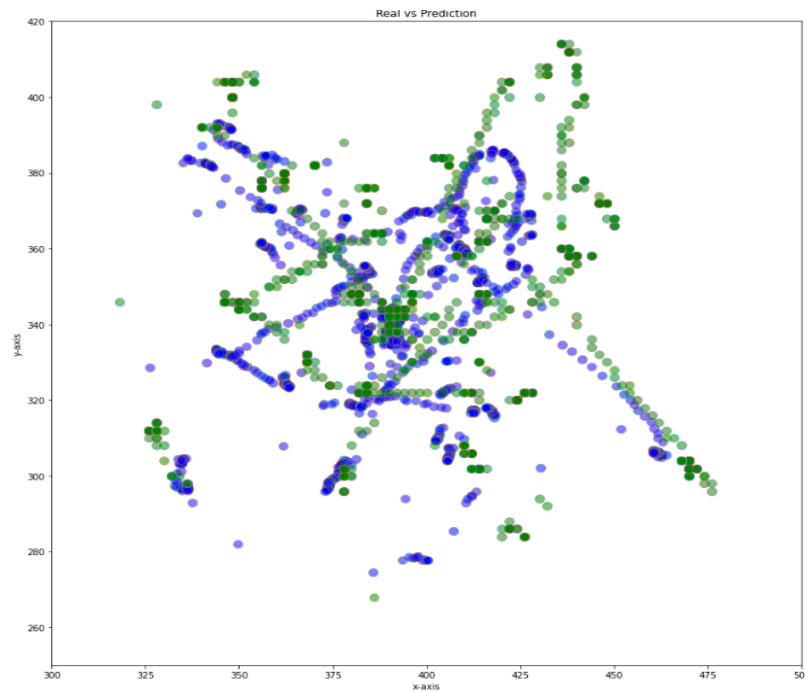
8.4.4 Images and Graphs

Figure 7: Scatter plot of real(green) vs predicted(blue) x-y coordinates with glare in iris and pupil
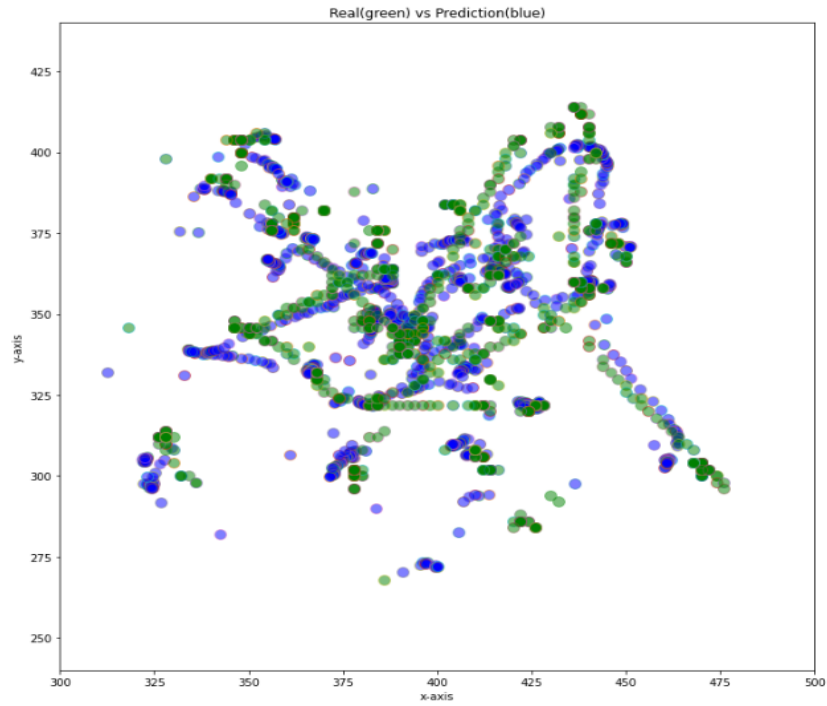
Figure 8: Scatter plot of real(green) vs predicted(blue) x-y coordinates without glare in iris and pupil

Graph of loss and mean squared error (mse) when training the pupil detection algorithm:
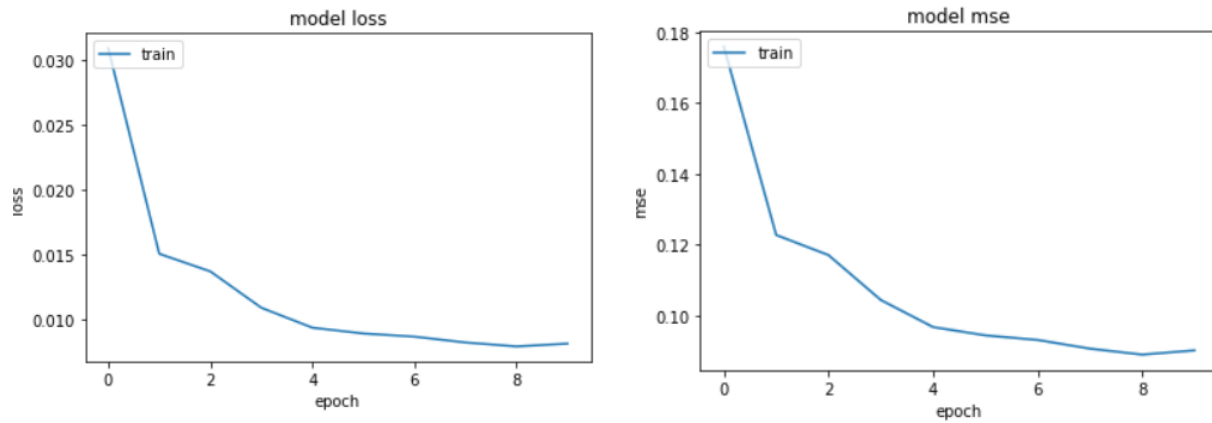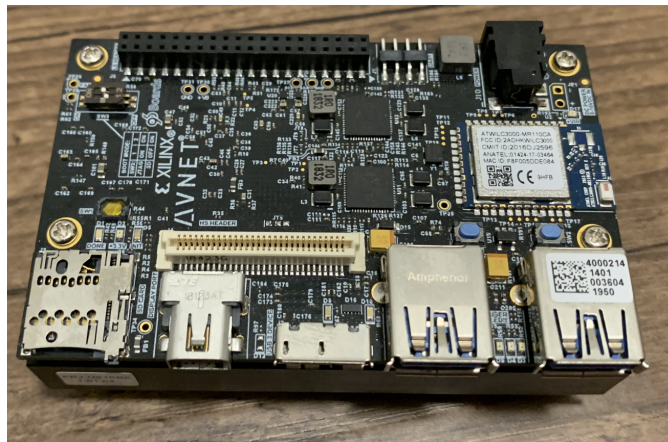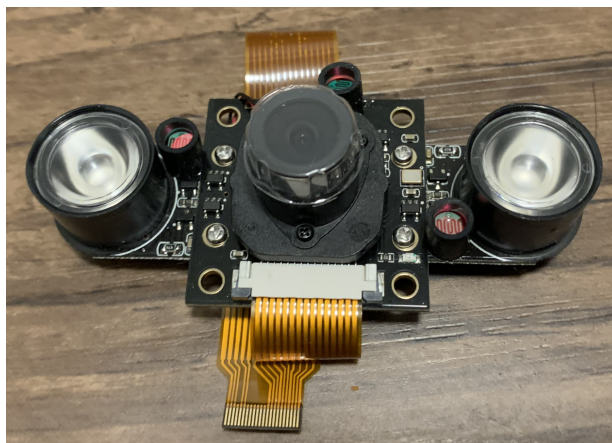

Image of the Ultra96-V2 Board:

Image of OV5647 Camera Module by ArduCam:

8.4.5 Team Contract

8.4.5.1 Team Members

- Nathanael Morris
- Ritvik Maripally
- Ron Mei Hang Teoh
- Yee Shen Teoh
- Zi-Jan Wong

8.4.5.2 Team Procedures

- Day, time, and location for regular team meetings:
  - Virtually on Tuesdays at 1:00pm with the advisor.
  - Virtually or Physically on Wednesdays at 6:00pm with the team.
  - Virtually on Thursdays at 6:00pm with the client.

- Preferred method of communication updates, reminders, issues, and scheduling:
  Email, Telegram and Discord.

- Decision-making policy:
  Consensus

- Procedures for record keeping:
  We will take turns to write the meeting minutes, and the minutes will be shared via Discord.

8.4.5.3 Participation Expectations

- Expected individual attendance, punctuality, and participation at all team meetings:
  - Full attendance and participation, and punctual to all meetings: team members need to be informed at least 2 days in advance if members are expected to be absent.

- Expected level of responsibility for fulfilling team assignments, timelines, and deadlines:
  - High level of responsibility: completing assignments in a timely manner and in good quality.

- Expected level of communication with other team members:
  - High level of communication: important information must be announced in the appropriate Discord channel.

- Expected level of commitment to team decisions and tasks:
  - High level of commitment: must have sufficient input during meetings and committed to completing tasks given.

8.4.5.4 Leadership

- Leadership roles for each team member:
  - Nathanael: Machine learning manager
  - Ron: Database manager
  - Ritvik: Team organization/communication and security manager
  - Yee Shen: Testing and hardware manager
  - Zi-Jan: Client and advisor interaction, and machine learning manager

- Strategies for supporting and guiding the work of all team members:
  - Each team member will present their work for the past week during the meeting so that everyone knows what is going on.
  - It is also essential that everyone can pick up where they left off.

- Strategies for recognizing the contributions of all team members:
  - A simple meeting minutes will be written by different members each meeting to record contributions and ideas given by each member.
  - The contributions of each member will be written in the weekly report that will be submitted and sent to the advisor and the client.

8.4.5.5 Collaboration and Inclusion

- Skills, expertise, and unique perspectives each team member brings to the team:
  - Nathanael: I love to learn new things from other people and I have good teamwork and conflict management skills. I also have good programming skills.
  - Ron: Willing to learn new things, have experience working long term with a group of people.
  - Ritvik: Programming skills related to cybersecurity and leadership qualities from previous clubs.
  - Yee Shen: Moderated programming skills. Embedded system knowledge. HW, FPGA, and VLSI knowledge and design skills.Bringing in the perspectives of an EE student.
  - Zi-Jan: Good programming skills, capable of working in a team, willing to learn new things and take on challenges.

- Strategies for encouraging and support contributions and ideas from all team members:
  - Listen to ideas from everyone and be respectful to each other's ideas.
  - Our team's brainstorming motto is: "*No idea is a dumb idea!*"

- ○ Wait for members to finish speaking before chiming in with opposing ideas or thoughts.

- Procedures for identifying and resolving collaboration or inclusion issues:
  - ○ Bring up the issue privately with the particular team member. If there is no improvement, we will bring it up during team weekly meetings and we will discuss as a team and compromise to reach a common ground.

8.4.5.6 Goal-Setting, Planning, and Execution

- Team goals for this semester:
  Plan and research our project so that all team members have an excellent understanding of how the project will work and how we will build it/execute our planning next semester.

- Strategies for planning and assigning individual and team work:
  Each week we will be having a meeting with our client and with our group, and during these meetings we will discuss what each team member will be working on for the following week.

- Strategies for keeping on task:
  We will have weekly meetings to make sure everyone is on the same page, and that all team members are putting in an adequate amount of work.

8.4.5.7 Consequences for Not Adhering to Team Contract

- If there are infractions of any of the obligations of this team contract, we will bring up the infractions with all team members and figure out a way to fix it.

- If the infractions continue, we will have an internal discussion to fix the issue. If the issue persists, bring up the issue to the professor/advisor.

****
a) *I participated in formulating the standards, roles, and procedures as stated in this contract.*
b) *I understand that I am obligated to abide by these terms and conditions.*
c) *I understand that if I do not abide by these terms and conditions, I will suffer the consequences as stated in this contract.*

1) RON MEI HANG TEOH   DATE 2/11/2022
2) ZI-JAN WONG   DATE 2/11/2022
3) NATHANAEL MORRIS   DATE 2/13/2022
4) YEE SHEN TEOH   DATE 2/13/2022

5) RITVIK MARIPALLY   DATE 2/13/2022