The Edward S. Rogers Sr. Department of Electrical and Computer Engineering **University of Toronto**
ECE496Y Design Project Course **Group Final Report**
Title: Capacitive Plethysmograph for human biometric recognition

Team #:	391	
Team members:	**Name:**	**Email:**
	Saminul Islam	s.islamsamin@mail.utoronto.ca
	Rahul Banerjee	rahul.banerjee@mail.utoronto.ca
	Pratyush Menon	pratyush.menon@mail.utoronto.ca
	Dhruv Patel	dhruvkumar.patel@mail.utoronto.ca
Supervisor(s):	Dimitrios Hatzinakos	
Section #:	8	
Administrator:	Milan Graovac	
Submission Date:	March 31, 2022	

Group Final Report Attribution Table

Section	Student Names			
	Saminul	Rahul	Pratyush	Dhruv
Executive Summary	MR	ET	MR	RD
Introduction	ET	MR	RD	ET
Final Design	RD	RD	MR	ET
Testing and Verification	RD	RD	RD	MR
Summary and Conclusion	RD	ET	MR	ET
Project Management	ET	MR	ET	RD,RS
Appendix	RD, MR	RD, MR	RD, MR	RD, MR
References	RD, MR	RD, MR	RD, MR	RD, MR
All	FP	FP	FP	FP,CM

RS – responsible for research of information
RD – wrote the first draft
MR – responsible for major revision
ET – edited for grammar, spelling, and expression
“All” row abbreviations:
FP – final read through of complete document for flow and consistency
CM – responsible for compiling the elements into the complete document

Signatures

By signing below, you verify that you have read the attribution table and agree that it accurately reflects your contribution to this document.

Name	Rahul Banerjee	**Signature**	Rahul Banjerjee	**Date:** 03/31/2022
Name	Saminul Islam	**Signature**	Saminul Islam	**Date:** 03/31/2022
Name	Pratyush Menon	**Signature**	Pratyush Menon	**Date:** 03/31/2022
Name	Dhruv Patel	**Signature**		**Date:** 03/31/2022

Voluntary Document Release Consent Form

Consent Statement

We verify that we have read the above letter and are giving permission for the ECE496 course coordinator to use our reports as outlined above.

Team #: ________
Project Title: ____

Supervisor(s): ______________________
Administrator: ______________________

Name	Rahul Banerjee	**Signature**	
Name	Saminul Islam	**Signature**	
Name	Pratyush Menon	**Signature**	
Name	Dhruv Patel	**Signature**	

Group Highlights

After the progress report, the team was using a split approach to work on developing capacitive plethysmography (CPG) signal acquisition hardware alongside developing a real-time authentication software system using existing photoplethysmography (PPG) data and hardware. The team faced an early setback as it was deemed that the CPG hardware development had been unsuccessful, so the team regrouped and focused on creating a real-time authentication system with PPG signals. The four main aspects of this system involved establishing communication with the hardware, cleaning and processing gathered data, building a real-time authentication model, and creating a consolidated interface for the functionality to be intuitively used.

On the hardware front, the group created a Python script to connect with the PPG hardware using Bluetooth, which allows data collection to be more streamlined. On the dataset side, the group was able to convert the signals from the database which were in a Matlab file format to a Numpy format which was then used to replicate the preprocessing and denoising steps mentioned in prior literature.

In terms of the actual model, the group successfully developed a machine learning model that can learn to identify and authenticate any individual given 90 seconds of processed PPG data corresponding to that individual. This was done by building a classifier model which learned to extract the important features in the PPG signal, and then feeding those into a small feed-forward network that is unique for each individual.

Finally, a web app based interface was implemented to consolidate the prior functionality into one cohesive application that allows for intuitive operation. This allows prospective users to avail themselves of the authentication functionality without requiring any technical knowledge.

Despite earlier setbacks, the team is now on track to complete the project on time with respect to our internal deadlines, as shown in our most recent Gantt chart (attached later in this document). The team successfully overcame the unforeseen circumstances (ie. the return of COVID) that proved to hinder us before our last update, and have almost completed the project.

Individual Contributions

N/A - Tasks started but aborted later due to insufficient results

Pratyush Menon

My technical contributions to this project can be split into two major categories: our original project plan, and then our modified plan as we pivoted our project from using CPG data to PPG data. Originally, I was responsible for researching and helping to build a hardware design over this reporting period, but we were unable to successfully build a CPG device due to a variety of reasons. As such, I focused on working on the software aspect of the project. I was responsible for using the Biosec2 dataset and creating a preprocessing pipeline to filter and clean the raw measurements. Next, I implemented the state-of-the-art model described in our supervisor's paper as a starting point for our future software development. Finally, I created the real-time authentication model which can be used to recognize and authenticate any specific user after being provided a 90 second snippet of their PPG measurements.

Task #	**Task Title**	**Completion Date**
1.5	Research a Capacitive Plethysmography Design	Oct 18, 2021
2.6	Build Measurement Circuit	N/A
3.4	CPG Individual Data Collection	N/A
5.1	Dataset Conversion	Jan 10, 2022
5.2	Data Preprocessing	Jan 10, 2022
5.3	Replicating Supervisor's Model with Convolutional and Recurrent Layers	Jan 23, 2022
5.6	Create Real-Time Authentication Model	March 5, 2022
6.4	Final Testing and Bug Fixing	In Progress

Table 1: Individual Contributions to Technical Work - Pratyush

Saminul Islam

Initially, I worked with my team members to research and implement designs for our CPG device. As the team divided the work, I continued to work on the CPG design but unfortunately that did not provide sufficient results at the end. So I shifted my focus on PPG and used the PPG device provided by our professor to gather new data and create a full end-to-end application with a frontend component which is able to authenticate a person in real time. Currently I am just working on the final testing and fixing some bugs we have with the application.

Task #	**Task Title**	**Completion Date**
1.5	Research Capacitive Plethysmography Design	Oct 18, 2021
2.1	Gather hardware requirements and purchase equipment	Nov 15, 2021
2.4	Build Sensor	N/A
2.5	Build Measurement Circuit	N/A
2.6	Setup Arduino for Measurement	Dec 13, 2021
4.1	CPG Individual Data Collection	N/A
3.1	PPG Device Setup	Jan 16, 2022
3.2	PPG Data Collection	Jan 23, 2022
6.1	Building End-to-End Workflow	March 21, 2022
6.2	Frontend Application	March 21, 2022
6.4	Final Testing and Bug Fixing	In-progress

Table 2: Individual Contributions to Technical Work - Saminul

Rahul Banerjee

My technical contributions were assisting in building the CPG Device and setting up the Arduino to collect the data. For the PPG device, I set it up and wrote a script in python to collect data and plot the PPG signal. Unfortunately the CPG did not provide sufficient results so I shifted my priorities towards creating the end-to-end workflow using the model created by my team members. After which I designed and implemented the frontend for the application. My current focus is on final testing and bug fixing.

Task #	Task Title	Completion Date
1.5	Research Capacitive Plethysmography Design	Oct 18, 2021
2.1	Gather hardware requirements and purchase equipment	Nov 15, 2021
4.1	CPG Individual Data Collection	N/A
4.3	CPG Data Cleaning	N/A
2.6	Setup Arduino for Measurement	N/A
3.2	PPG Data Collection	Jan 16, 2022
3.3	Script to read and plot data from PPG device	Jan 23, 2022
4.3	CPG Data Cleaning	N/A
6.1	Front End	March 21,2022
6.2	Building End-to-End Workflow	March 21,2022
6.4	Final Testing and Bug Fixing	In-progress

Table 3: Individual Contributions to Technical Work - Rahul

Dhruv Patel

My contribution to the project can be categorized into three areas: researching the design for the hardware, designing multiple authentication models to classify the signal and deploying the model to the cloud to train and classify the signal in real-time. At the beginning of the project, my major contribution was researching various hardware models. I researched a direct circular capacitance plethysmography device. The design wasn't selected. During the second phase, my major contribution became exploring different machine learning models for authentication and classification. I worked on autoencoders with the focus to understand the encoded signal and use that as a unique signature to classify individuals. The second model that I explored was attention LSTM. The unique thing about this model was the "attention" aspect, its ability to remember long sequences of patterns within the signal. Each model served a unique purpose, Attention LSTM to classify the signal, while the Autoencoder to identify the uniqueness within the signal. While both models were a novel approach to the problem they fall short in the accuray. During the last phase of the project, my contribution became deploying the final model to a cloud service to be used in real-time. This deployment pipeline built using Amazon SageMaker will be responsible for training with the provided signal and then producing the classification in real-time. Once built we can build upon it through an application.

Task #	**Task Title**	**Completion Date**
1.4	Research Hardware design	10/18/2021
2.6	Setup Arduino for Measurement	12/13/2021
4.4	CPG Individual Data Collection	N/A
5.4	Build an Attention Long Short-term Memory	02/21/2022
5.5	Build a Regularized Autoencoder	02/21/2022
6.3	Build cloud model pipeline using Amazon SageMaker	In Progress

Table 4: Individual Contributions to Technical Work - Dhruv

Acknowledgements

We would like to thank our supervisor, Professor Dimitrios Hatzinakos for his guidance and support. Additionally we would also like to thank Bilal Taha, who is currently a Graduate Student working under our supervisor. Finally, we would like to thank our administrator, Milan Graovac and our Communication Instructor, Benjamin Kinsella for giving us feedback on our document and helping us define our project's goals and requirements.

Executive Summary

The original goal of this project was to develop a method for using capacitive plethysmography (CPG) signals measured from the human touch to provide biometric recognition and authentication. The team faced some obstacles in achieving this goal, with the return of COVID restrictions being a major hindrance in terms of developing the hardware required. As such, the team was able to build a CPG design, but fell short of the set requirements. With input from our supervisor, the decision to pivot to a backup project goal was made.

The backup project goal was to develop a method for using photoplethysmography (PPG) signals to provide real-time biometric recognition and authentication. The project requirements remain nearly identical, with the single difference that we will be using PPG signals instead of CPG signals.

The team explored various authentication architectures and was able to successfully build an end-to-end workflow for real-time authentication on any arbitrary individual, using PPG signals. An accompanying web app was created to ensure the system is intuitive for the technologically inexperienced as well.

The model achieves an equal error rate (EER) of 33%, which is below state of the art models in the field. However, this is understandable as the model was evaluated on unseen PPG signals acquired from arbitrary individuals, to match real-world scenarios. As such, a slight decline in performance as compared to models evaluated solely on datasets consisting of a limited number of participants is not unexpected.

This work builds on existing literature as a proof-of-concept that PPG signals can be used for biometric authentication in real-world scenarios. PPG-based authentication is a promising field for the future, with devices such as smartwatches coming with optical sensors built in. As such, the developments in this project may have wide-ranging future applications in helping to popularize the use of PPG signals for authentication and recognition.

Table of Contents

1. Introduction

1.1 Background and Motivation

In a digitally dominated world, privacy is becoming more important than ever. In order to ensure the security of our data, we need strong authentication methods. Recently, biometric authentication methods have been gaining popularity [1] as they cannot be forgotten or stolen, and are difficult to reproduce or modify [2, 3, 4]. Currently, one of the biometrics on which research is being conducted are plethysmogram signals [1] as it reflects the pseudo-periodicity of one's blood pressure [5]. Specifically, state-of-the-art approaches use a photoplethysmogram signal (PPG) [1] - ie. a plethysmogram signal obtained with an optical sensor [6].

However, a few deficiencies regarding PPG-based authentication currently exist:

- PPG requires a light source and photodetector to measure the plethysmogram signals [6]. This hardware may prove challenging to acquire and use in some situations.
- PPG authentication systems have only been tested on pre-collected datasets consisting of a limited number of participants, and as such, results may not be as relevant towards real-world scenarios [1].

As such, two goals were initially explored: to investigate a new method of using the plethysmogram as a biometric source for authentication, and to devise a method of using plethysmogram signals for real-time authentication, which would allow authentication of any arbitrary individuals.

While the first goal was unsuccessful, the alternative method that was studied was capacitive plethysmography (CPG), which uses a capacitive sensor. An advantage of using CPG over PPG for authentication is that, if successful, it would be much easier for future research to implement this method in existing technology, as capacitors are important components of any touchscreen device.

The second goal was achieved to some success, with an authentication system being created that serves as a proof-of-concept that PPG signals are viable for real-time authentication in real world scenarios.

1.2 Project Goals and Requirements

Functions

ID	Project Functions	Criteria
1	Signal Acquisition	Acquire a time-varying CPG signal with a sampling frequency of at least 200hz
2	Store Signal	The sensor is able to store signals measured locally using onboard memory or additional memory provided using expansion (ex. SD cards) of at least 1 MB, equivalent to signals collected from 4 people.
3	Clean Signal	SNR of at least 10dB [7]
4	Authentication Algorithm	The Algorithm should achieve an EER (Equal Error Rate) of at most 25% to match the worst-performing PPG Authentication Algorithm [1].
5	Output	Binary output to determine whether user authentication was successful or not.

Table 5: Project Functions

Objectives

ID	Project Objectives	Description
1	Sampling Rate of 300Hz	The PPG signal database with the highest sampling rate is 300 Hz [1], so the CPG signal should have parity.
2	SNR of 20dB	The cleaned signal should have an SNR of 20dB or more
3	EER of 5%	The Authentication Algorithm should achieve an EER of 5% or lower to match PPG Authentication Algorithms [1].
4	Authentication Time of 3 seconds	The time for the CPG signal transmission and the model to authenticate users should be at most 3 seconds

Table 6: Project Objectives

Constraints

ID	Project Constraints	Description
1	Cost	Team should not spend more than $500

Table 7: Project Constraints

2. Final Design

2.1 System-Level Overview

The original goal of this project was to develop a method for using Capacitive Plethysmography (CPG) signals measured from the human touch to provide biometric recognition and authentication. However, after a lot of effort, the team was unable to build a usable CPG circuit. The team acquired a Photoplethysmography (PPG) device that was used to measure the signals we needed for biometric recognition and authentication.

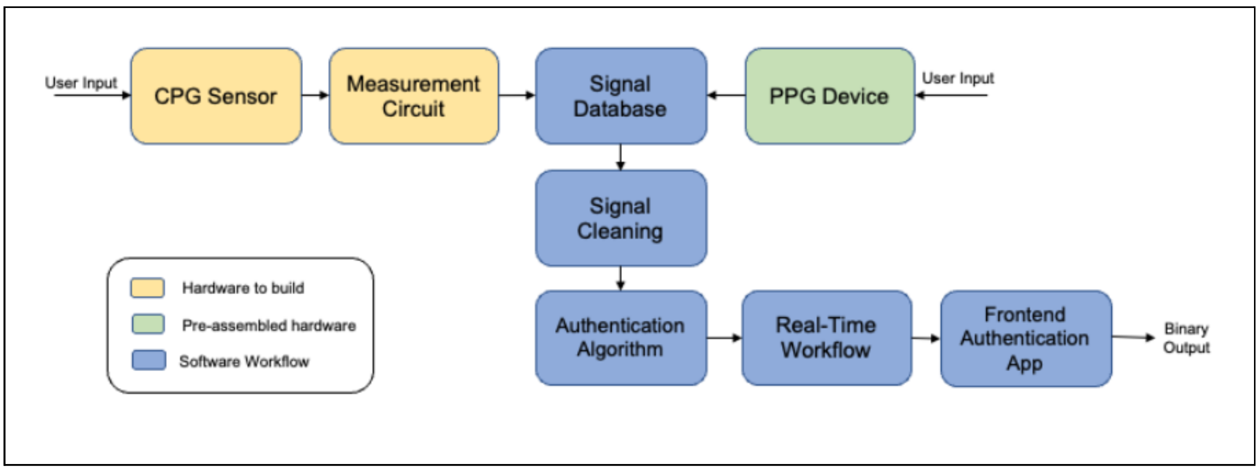


Figure 1: Project System-level overview diagram

The blocks in yellow represent the CPG measurement device and the block in green represents the PPG measurement device. The CPG device uses a Capacitive Sensor while the PPG device uses an optical Sensor. Although they are using different sensors, they are still measuring the same attribute, Plethysmograph Signals. Therefore, the yellow blocks and the green block are interchangeable and either one of them can be used as an input to the blue blocks. The blocks in blue are the parts which are common, i.e they are needed for both the CPG measurement device and the PPG measurement device.

During the final part of the project, the team decided to work solely with PPG signals.

2.2 System Block Diagram

2.2.1 CPG Sensor and Measurement Circuit

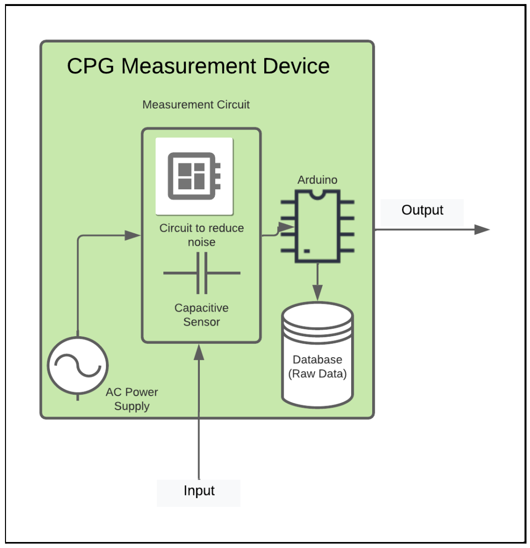


Figure 2: CPG Measurement Device block diagram

The measurement circuit consists of the capacitive sensor and some other electrical components such as resistors, capacitors, and operational amplifiers. The capacitive sensor is the surface the user touches. The rest of the circuit helps in reducing the noise in the measured voltage. The Arduino captures the signal as a time series of data points based on its sampling frequency. This time-series data is then stored in external storage such as an SD card.

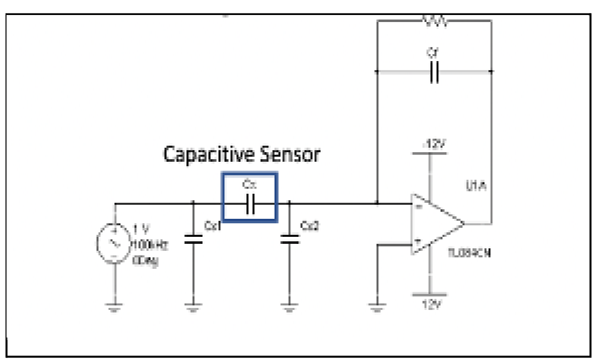


Figure 3: CPG Sensor Circuit Schematic

After thorough research, the circuit design shown above was chosen by the team. It was originally created by Phillips et al [8]. The output signal generated by the design was deemed acceptable by the team and the supervisor. For the Capacitive Sensor, we tried to use a sheet of foil paper and a store-bought Capacitive Touch Sensor [9] (Appendix D, Appendix E). The values for the various components in the circuit were not provided in the paper and as a result we had to try out various values. However, after a lot of trial and error, we were unable to build a circuit which was able to reduce the noise during measurement of the signal. The team then decided to acquire a pre-assembled PPG measurement device.

2.2.2 PPG Device

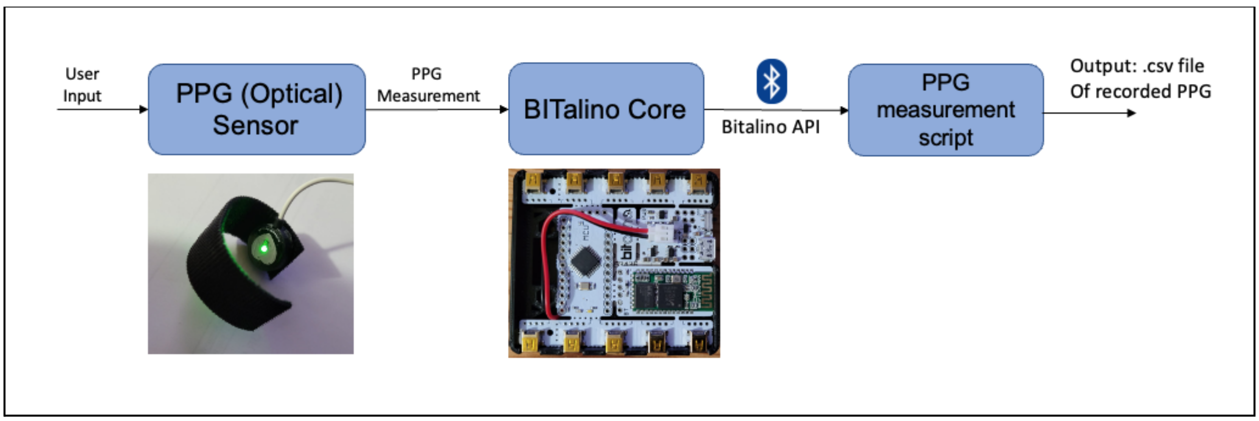


Figure 4: PPG Device Block Diagram

The PPG measurement device was acquired from our supervisor. It consists of a BITalino Core [10] attached to a PPG (Optical) Sensor [11]. The BITalino Core transmits the signal read by the PPG sensor via Bluetooth. The PPG measurement script uses BITalino's Python API [12] to receive PPG signals from BITalino Core and store it in a CSV file.

2.2.3 Signal Dataset

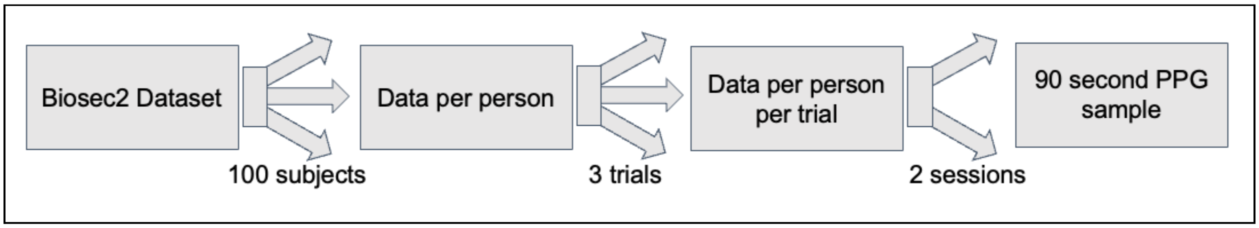


Figure 5: BioSec2 Database Representation

The BioSec2 PPG Dataset [13] was used for training the machine learning models used for classification and authentication. It consists of measured signals for 100 participants and was provided to us by our supervisor. As shown in the figure, each participant recorded their PPG signal for 90 seconds in 2 separate sessions in 3 different trials.

2.2.4 Signal Cleaning

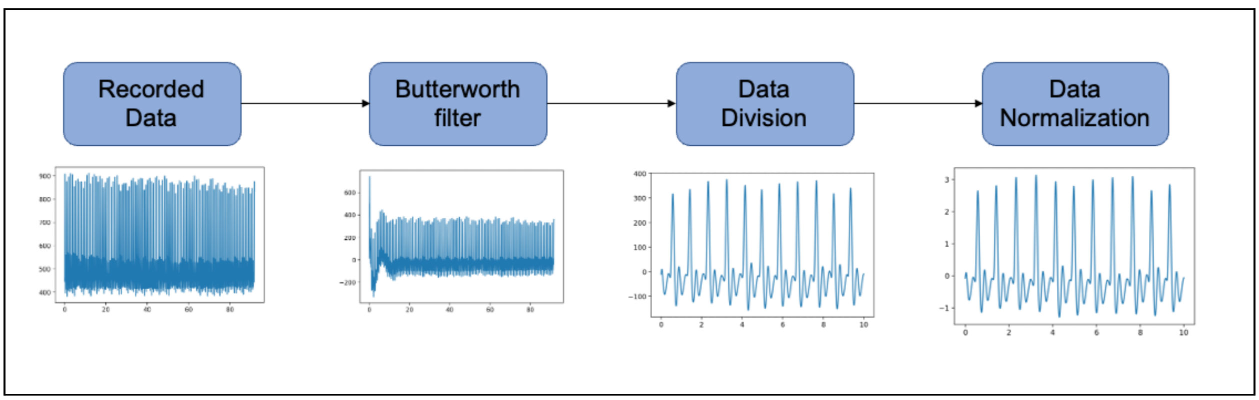


Figure 6: Signal Cleaning Block Diagram

The PPG data from each participant is passed through a series of functions to clean and denoise it, as shown above. A Butterworth band-pass filter is used to reduce the noise in the PPG signal. This is helpful during model training since when the signal is fed to the model, it only consists of the important characteristics. Each 90 second data is then divided into 10 equal intervals and each interval is mean-normalized.

2.2.5 Authentication Algorithm

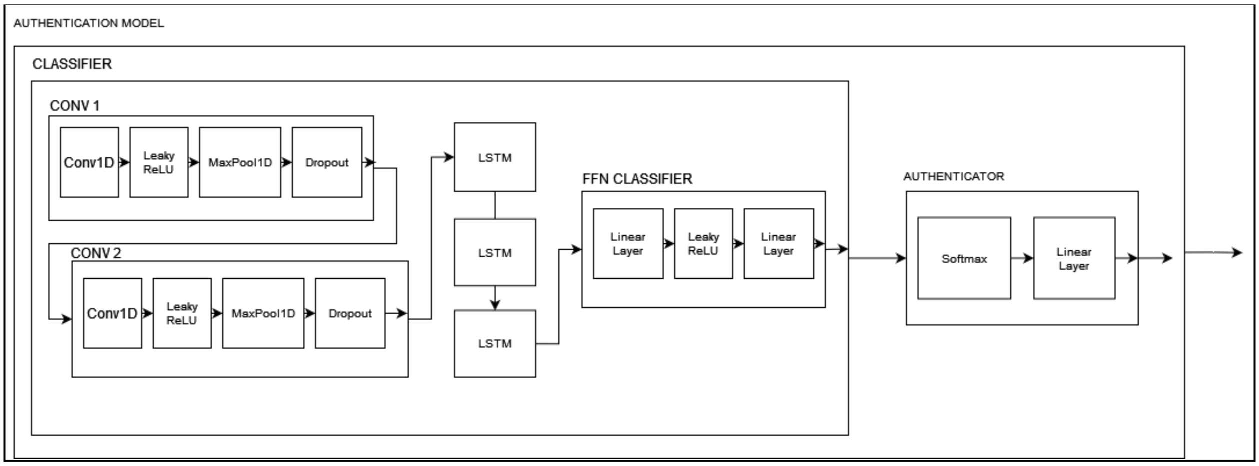


Figure 7: Authentication Algorithm block diagram

The overall authentication model is divided into 2 major sections: the classifier and the authenticator. The classifier is a neural network model with 1D convolutional and LSTM (Long Short-Term Memory) layers, with the goal of matching an input PPG signal to the person in the dataset to whom it belongs to. This architecture is based on state of the art models in the literature, developed by our supervisor [14].

To train the classifier, the majority of the preprocessed data is used; however, a few randomly chosen participants were excluded from the training set used for the classifier. The data corresponding to the excluded participants is set aside to train the authenticator model. Once the classifier model is trained to a high degree of performance, the parameters are then frozen and saved.

The next step is training real-time authentication models. In order to train a real-time authentication model for a specific person, preprocessed data corresponding to the individual that the model will identify is passed in, along with the data corresponding to the previously excluded participants. The real-time authentication model is built upon the saved classifier model, with a simple feedforward network using the classifier output as input features and transforming them into a binary output that indicates an authentication match or not. At this stage, only the simple feedforward network's weights are adjusted, while the saved classifier weights stay frozen. This allows models that can authenticate any desired individual to be created and trained extremely quickly, by using the important extracted features of the PPG signal that were learned by the classifier.

2.3 Module-Level Descriptions and Designs

PPG Device
Input: - User touch: the user wraps the PPG sensor around one of their fingers
Output: - Stream of integer values values representing the measured PPG of the user
Function: The function of the PPG device is to measure the user's PPG and transfer it over bluetooth to a computer that is connected to the device via bluetooth or wire. The signal can be measured and observed by using the OpenSignals software [15]

Table 8: PPG Device Module

PPG Measurement Script
Input: - Bluetooth connection with PPG device - Recorded Samples from the device
Output: - Record the measured PPG and store it as a CSV file to be used later.
Function: The function of this module is to connect directly to the device through bluetooth and record the user's PPG for a specified time and store it as a .csv file to be later loaded. This script allows us to directly record the data without having to use the OpenSignal software [15] by using the BITalino (r)evolution Python API [12].

Table 9: PPG Measurement Script Module

Signal Database
Input: - CSV files of recorded PPG data
Output: - Larger database with new recorded data
Function: The initial signal database is the Biosec2 database[13] which consists of 100 participants. As we use the PPG device to measure new individuals, we keep adding the stored CSV files to the provided Biosec2 database to create a larger database which will be used for training.

Table 10: Database Module

Signal Processing
Input: - Database of signals - New recorded PPG signal
Output: - Cleaned and denoised input signals
Function: The primary purpose of this module is to take raw PPG data and clean and process the signal. The signal is processed from the raw measurements using a Butterworth band-pass filter and then divided into 10 second segments. Each segment is then normalized and the preprocessed input is saved as a numpy file (.npy) which can be used as input for the models. This module is used to process the initial signal database as well as any measured signals used for real-time authentication.

Table 11: Signal Processing Module from Figure 6

PPG Signal Classifier
Input: - Preprocessed PPG dataset
Output: - Trained classifier model saved as a pickle file (.sav)
Function: This module trains a classifier model that learns to predict the person to whom the input signal belongs to, from the Biosec2 dataset. It does not use the entirety of the signal database as unseen participants are used to test the authenticator model in another module.

Table 12: Classifier Module

PPG Authenticator
Input: - Trained classifier model saved as a pickle file (.sav) - Recorded and preprocessed PPG signal of duration 90 seconds for new user - Preprocessed PPG dataset
Output: - Model used to authenticate on incoming PPG signals
Function: The classifier model is used as the basis of a real-time authentication model that learns how to authenticate a specific person after being provided a sample PPG signal belonging to that person. This sample and previously unused signals from the PPG dataset are applied as training data to produce a model that can authenticate the desired person.

Table 13: Authenticator Module

Real Time Workflow
Input: - Authenticator Model - New recorded PPG signal
Output: - Binary output of whether the user is authenticated
Function: The Real Time Workflow automates the process of data collection and model training. This prevents the user from having to deal with the underlying models and other functions.

Table 14: Real Time Workflow Module

Frontend Authentication App
Input: - User interactions - User's PPG Signal
Output: - Model/Data Collection Status Messages - Success/Error message depending on whether the user is authenticated or not
Function: The front end is built using Streamlit [16], a Python Framework. It provides the user the ability to interact with an User Interface to collect data, train the model and finally test the model. The UI also has steps directing the user on how to collect the data and train the model.

Table 15: Authentication Algorithm Module from Figure 1

2.4 Assessment of Final Design

The goal of our project was to use plethysmography signals for biometric authentication. Our final design made use of a preassembled PPG measurement device and techniques described in existing literature [14]. We were able to build a model that can be trained on an user's PPG signal in under 5 minutes and can be later used to authenticate the user (while rejecting anyone else). We were able to automate the entire process so the user wouldn't have to deal with any of the underlying models or functions. Additionally, we built a web-app to make it easier for users to use our final design.

The real-time authentication model was able to achieve a testing accuracy of 79.3% on unseen test data (Appendix F), while having an EER of 33.3%. Our requirement was to have an EER below 25%, however this threshold was based on metrics for models that were only trained on a single participant in a provided dataset. Our final design on the other hand is able to learn characteristics from an user's PPG data and train a model in real time; as such, a slightly higher EER is an acceptable trade-off.

We had set a requirement for our measured data to have a minimum SNR of 10dB. The SNR for the PPG dataset we trained our model on was 13.7 dB (Appendix C). Additionally our final design also met the requirement of having a minimum sampling frequency of 200Hz. Finally, the final design met the sole constraint of costing less than $500, which is under our planned budget ("Total Cost Required Funding" in Figure 15).

3. Testing and Verification

3.1 Verification Table

ID	Requirement	Verification Method	Result and Proof
1	Signal Acquisition: Plethysmogram signal with a sampling frequency of at least 200hz.	**REVIEW OF DESIGN:** Ensure the device used to record the signal can support the sampling rate.	PPG: Pass. See section 3.2.1 CPG: Fail. See section 3.2.2
2	Store Signal: Store data using onboard memory or additional memory provided using expansion (ex. SD cards) of at least 1 MB.	**REVIEW OF DESIGN:** Ensure that the database has a storage capacity of at least 1MB.	Pass. Database is stored on a laptop with storage capacity of 250+ GB.
3	Clean Signal: Minimum SNR (Signal to Noise Ratio) of 10dB.	**TEST:** Direct measurement/calculation of SNR of the cleaned signal.	PPG: Pass. See section 3.3.1 CPG: Fail. See section 3.3.2
4	Authentication Algorithm: Achieve an EER (Equal Error Rate) of 25% maximum on training data.	**TEST:** Measure EER based on algorithm outputs on previously unseen/unused training data.	Fail. See section 3.4.
5	Output: Indicate whether the user was authenticated or not.	**REVIEW OF DESIGN:** Ensure that there is an indicator of authentication success.	Pass. See section 3.5.

Table 16: Project requirements Verification Table

3.2 Signal Acquisition

3.2.1 Using PPG Device

The Bitalino software allows us to have a maximum sampling frequency of 1000Hz. We are also allowed to set our own sampling frequency in the PPG measurement script upto 1000 Hz.

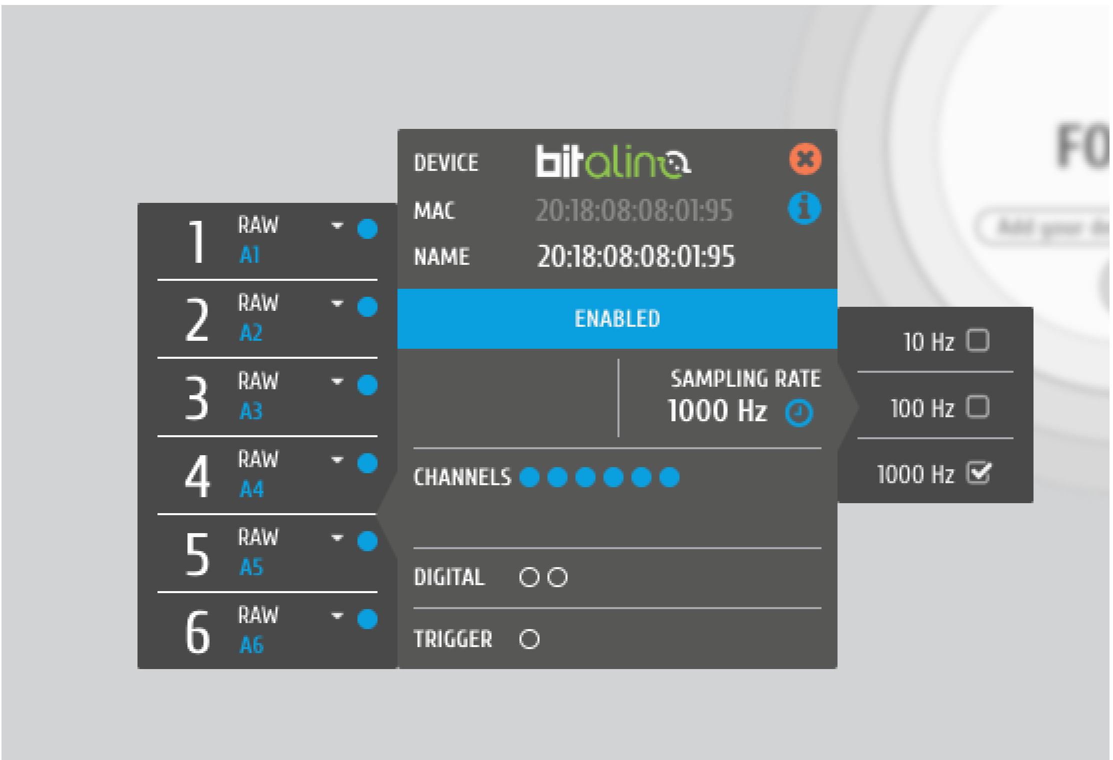


Figure 8: Sampling Rate used by PPG Measurement Device

3.2.2 Using CPG Device

When using the CPG device, we had to use an Arduino to record the measurements. We used a tool called ArduSpreadSheet (see Appendix A) to be able to store the data in a CSV. The tool gave us no control over the sampling frequency and after analyzing the stored csv, we calculated the sampling frequency to be around 40 Hz (Calculations in Appendix B).

3.3 Clean Signal

3.3.1 Using PPG Device

The average SNR of acquired PPG signals in the Biosec2 database was calculated to be around 13.7dB, according to a method commonly used for optical biosensors [17]. The code used to calculate this is attached in Appendix C.

This calculated SNR is higher than the required 10dB, and as such the design verifies the requirement. However, it does not pass the objective of having an SNR of 20dB or more.

3.3.2 Using CPG Device

Below is a sample CPG Signal collected using our CPG Measurement Circuit.

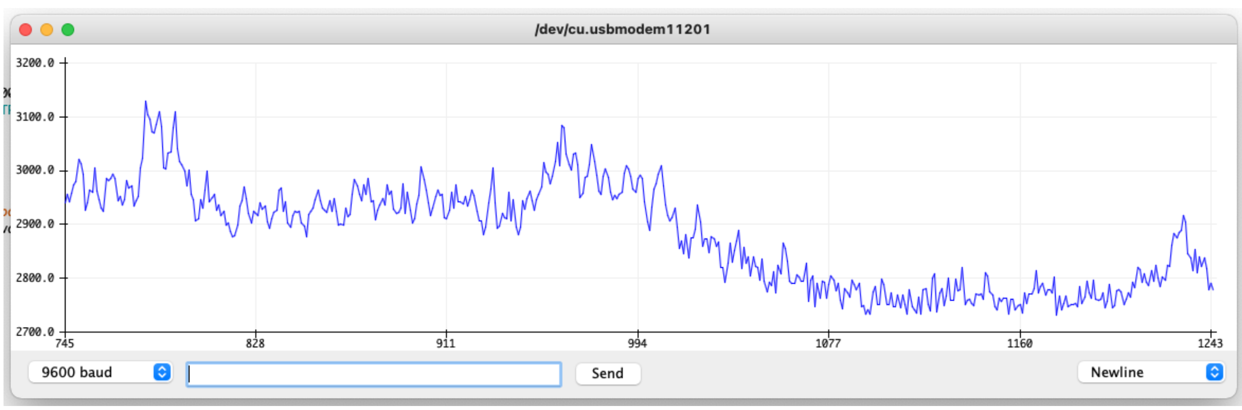


Figure 9: Sample CPG Signal

Based on existing research on CPG signals, we expected to see consistent time varying periodic signals. The CPG signal should look similar to the PPG signal. However in the above CPG data, the amplitude of the peaks are inconsistent and the data is not periodic. We believe the reason we were not able to get a periodic CPG signal was due to external noise while data measurement. Additionally, some of the sensors we used were designed as proximity sensors thus their performance when measuring CPG was low as they were too sensitive.

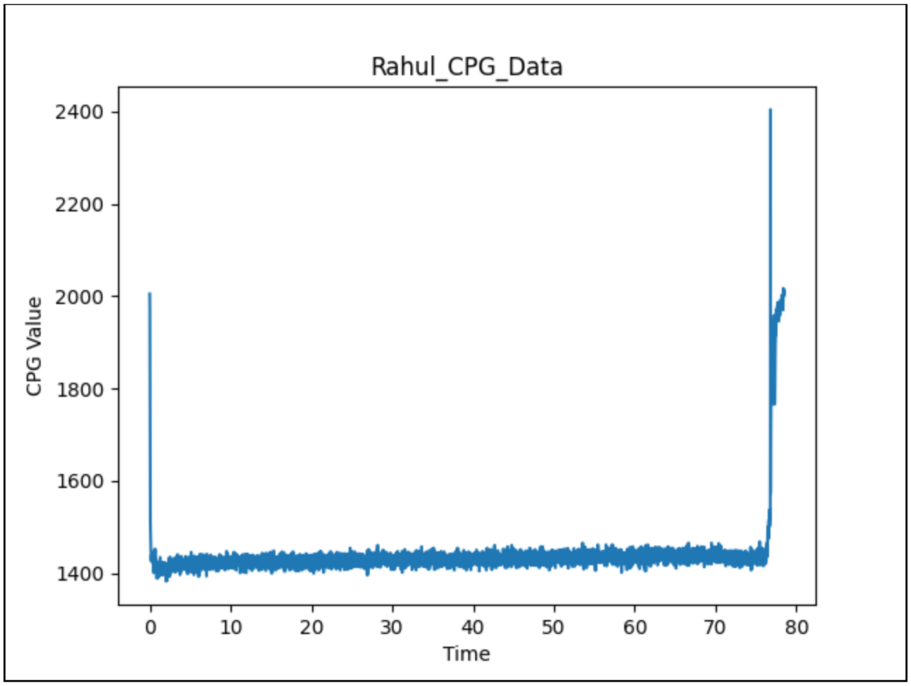


Figure 10: Sample CPG Signal

The graph shown above is the CPG data of a user. The two spikes were caused when another individual was present next to the user whose CPG signal was being measured.

3.4 Authentication Algorithm

In order to measure the effectiveness of the biometric authentication models, standard metrics such as EER are commonly used [1]. As such, EER was used to measure the effectiveness of the real-time authentication method devised here.

To measure the EER, real-time authentication results were recorded on a dataset of 25 previously unseen participants, with model sensitivity being varied to adjust the false positive and false negative rate. This was done at various sensitivity levels until the rate at which false positives and false negatives matched was found. The final FAR (False Acceptance Rate) vs FRR (False Rejection Rate) graph is shown below, with the intersection of both curves indicating the EER, which is approximately 33.3%.

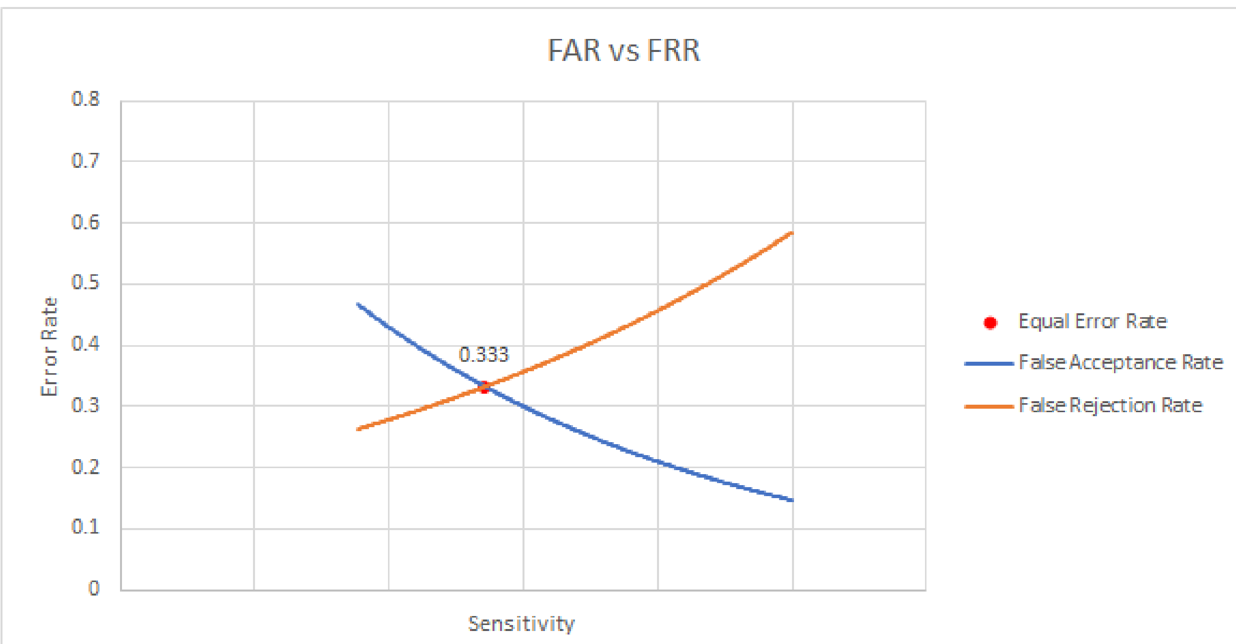


Figure 11: Equal Error Rate

An EER of 25% or below was required to verify this requirement, as other PPG authentication algorithms are able to achieve such results [1]. As such, the design fails this requirement. However, this is forgivable, as adding real-time functionality for any individual reduces this model's performance as compared to authentication systems that are trained on a single specific participant in a provided dataset.

3.5 Output

Our front end application is able to clearly distinguish between positive and negative authentication results. When authentication fails, the application displays an error message in red saying “Not Authentication” and when authentication is successful, the application displays a success message in green saying “Authenticated”.

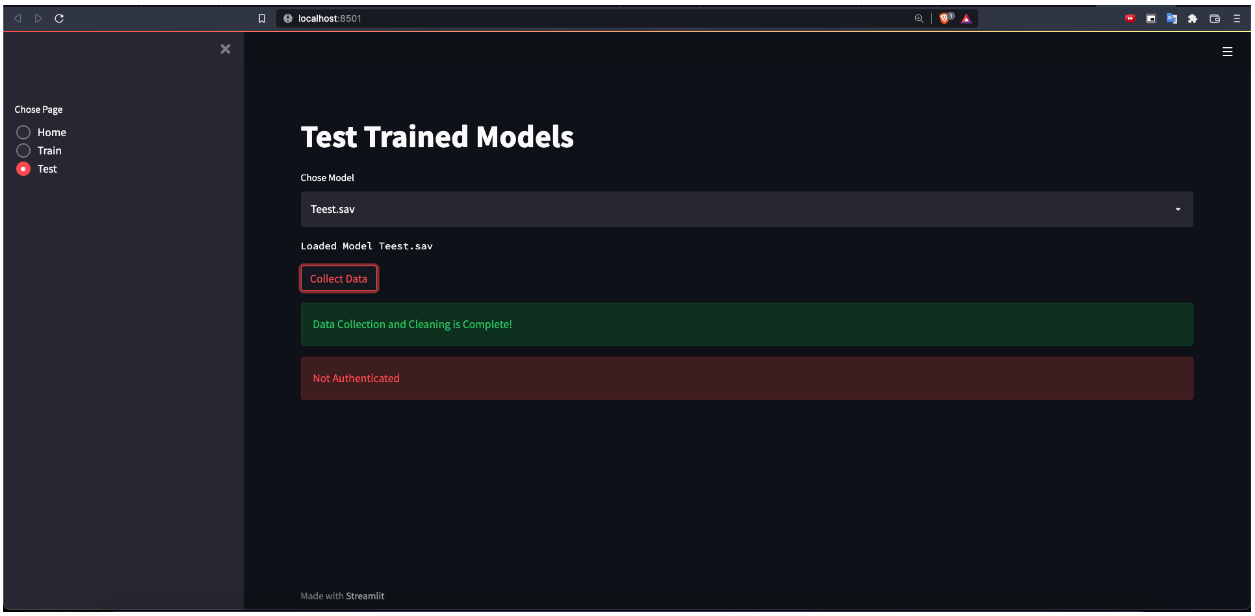


Figure 12: Output when authentication failed

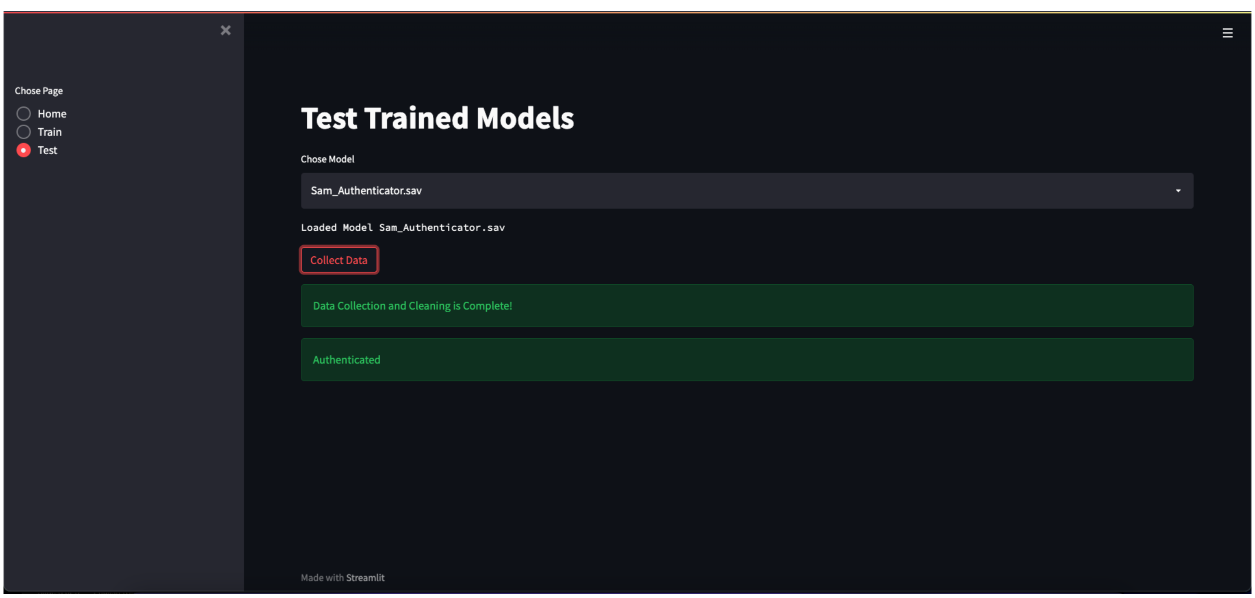


Figure 13: Output when authentication successful

4. Summary and Conclusions

The initial aim of this project was to build a capacitive sensor and use it to measure plethysmographs for use in authentication. But unfortunately the team was unable to build a sensor that met the project requirements needed to be used for authentication. Thus the team had to compromise and use a different type of sensor - an optical sensor for measuring the PPG signal. This was still gathering the same information, but in a different manner. Thus many of our project requirements stayed the same, but now focusing more on the authentication models and creating a real time application where the work can be demonstrated.

Using this sensor, we were able to create a workflow that met almost all of the project requirements we set out. Our end-to-end system allows us to test and demonstrate our work to any new individual and showcase the use cases of this project. The end-to-end real-time authentication system is a development above current literature in this field, as it works as a proof-of-concept that PPG signals can be used for biometric authentication in real-world scenarios.

In order to ensure the security of our data, we need strong authentication methods. In the future, our work suggests that a PPG-based authentication system would be a reliable source of authentication and can be used over other methods in multiple cases. For example, when wearing a smartwatch, it is able to record your PPG signal using the optical sensor and thus always keep you authenticated while you are wearing it. This authentication is also more accessible than others as theoretically the plethysmograph can be measured from any part of the body.

5. Project Management

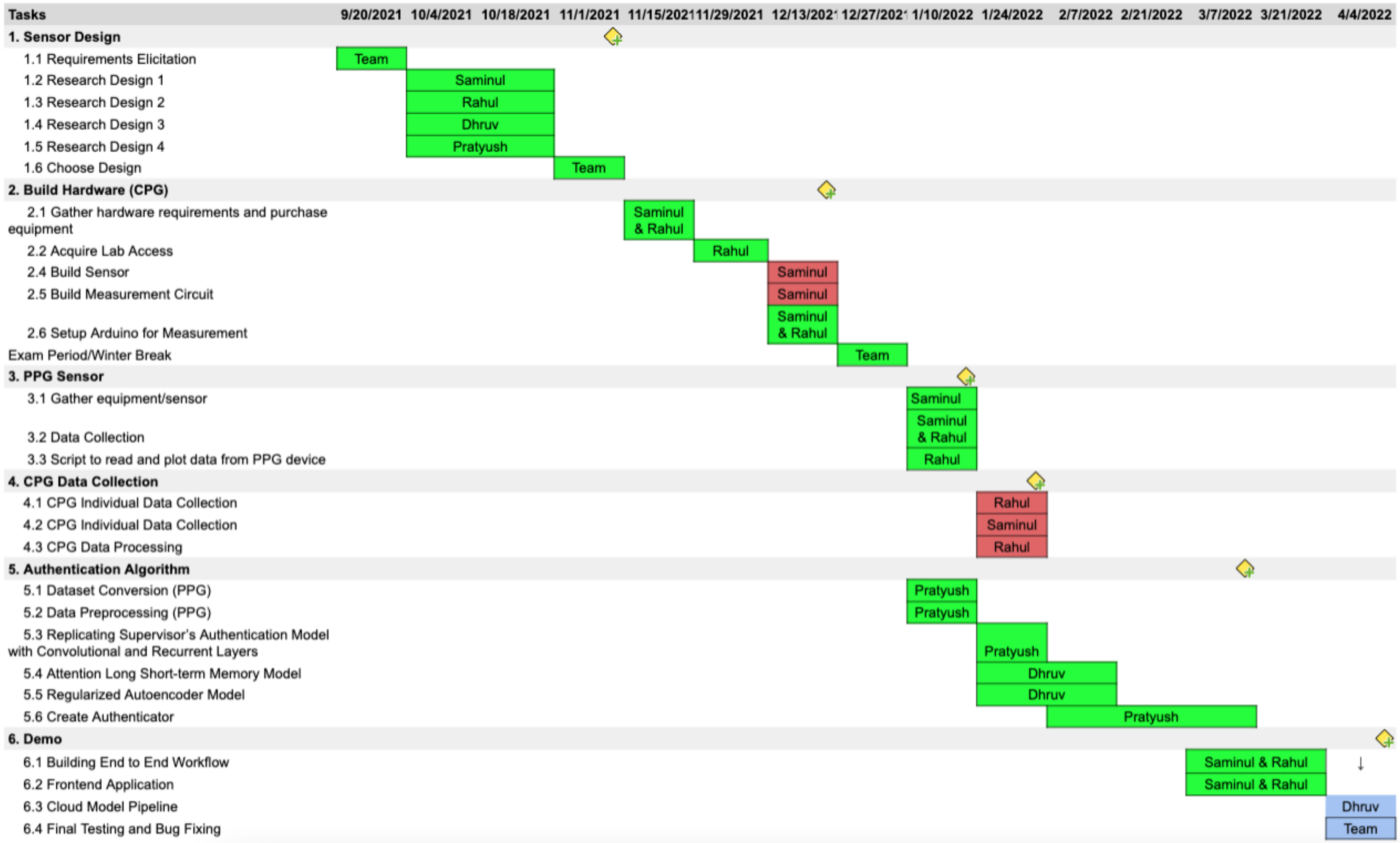


Figure 14: Gantt Chart

Consumables & Services								
Item	**Priority**	**Cost/Unit**	**Quantity (units or hours)**	**Total Cost**	**Required Funding**		**Funding**	
Arduino	1	$20.00	1	$20.00	Y		Students ($100 each)	$400.00
Arduino Components	1	$21.99	1	$21.99	Y		Supervisor	$0.00
Aluminium Foil	1	$5.00	1	$5.00	Y		Others	$0.00
Capacitive Sensor	1	$15.00	1	$15.00	Y		**Total Funding**	$400.00
Asana Basic	2	$0.00	4	$0.00	N			
Cloud Compute Credits	2	$13.99	4	$55.96	Y			
SD card and module	1	$25.00	1	$25.00	Y			
Total Required Funding				$142.95				
Capital Equipment								
DC Power Supply	1	$100.00	1	$100.00	N			
Oscilloscope	1	$650.00	1	$650.00	N			
AC Power Supply	1	$2,000.00	1	$2,000.00	N			
Function generator	1	$300.00	1	$300.00	N			
Total Cost				$3,050.00				
Total Required Funding				$0.00				
Student Labour								
Dhruv	-	$30.00	800	$24,000.00	N			
Pratyush	-	$30.00	800	$24,000.00	N			
Rahul	-	$30.00	800	$24,000.00	N			
Saminul	-	$30.00	800	$24,000.00	N			
Total Student Labour				$96,000.00				
Total Cost				$99,192.95				
Total Cost Required Funding				$142.95				

Figure 15: Financial Plan and Budget table

6. References

[1] J. Sancho, Á. Alesanco and J. García, "Biometric Authentication Using the PPG: A Long-Term Feasibility Study", Sensors, vol. 18, no. 5, p. 1525, 2018.

[2] A. K. Jain, S. Pankanti, S. Prabhakar, Lin Hong and A. Ross, "Biometrics: a grand challenge," Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004., 2004, pp. 935-942 Vol.2, doi: 10.1109/ICPR.2004.1334413.

[3] P. Tuyls, A. Akkermans, T. Kevenaar, G. Schrijen, A. Bazen and R. Veldhuis, "Practical Biometric Authentication with Template Protection", Lecture Notes in Computer Science, pp. 436-446, 2005.

[4] S. Pankanti, R. Bolle and A. Jain, "Biometrics: The future of identification [Guest Editors' Introduction]", Computer, vol. 33, no. 2, 2000.

[5] D. Hatzinakos, private communication, Oct. 2021.

[6] U. Yadav, S. N. Abbas and D. Hatzinakos, "Evaluation of PPG Biometrics for Authentication in Different States," 2018 International Conference on Biometrics (ICB), 2018, pp. 277-282, doi: 10.1109/ICB2018.2018.00049.

[7] "How to: Define Minimum SNR Values for Signal Coverage", Wireless-nets.com, 2021. [Online]. Available: http://www.wireless-nets.com/resources/tutorials/define_SNR_values.html. [Accessed: 13- Oct- 2021]

[8] Phillips, M. Hickey and P. Kyriacou, "Evaluation of Electrical and Optical Plethysmography Sensors for Noninvasive Monitoring of Hemoglobin Concentration", Sensors, vol. 12, no. 2, pp. 1816-1826, 2012.

[9] Amazon.ca, 2022. [Online]. Available: https://www.amazon.ca/gp/product/B07CQB7DYB/ref=ppx_yo_dt_b_asin_title_o03_s00?ie=UTF8&psc=1. [Accessed: 31- Mar- 2022].

[10] "BITalino Core (Dual Bluetooth BTH & BLE)", Plux Wireless Biosignals, 2022. [Online]. Available: https://www.pluxbiosignals.com/products/bitalino-core-mcu-ble-bt-power. [Accessed: 31- Mar- 2022].

[11] "Photoplethysmography (PPG) Sensor", Plux Wireless Biosignals, 2022. [Online]. Available: https://www.pluxbiosignals.com/products/photoplethysmography-ppg-sensor. [Accessed: 31- Mar- 2022].

[12] "GitHub - BITalinoWorld/revolution-python-api: Python API for BITalino (r)evolution", GitHub, 2022. [Online]. Available:

https://github.com/BITalinoWorld/revolution-python-api. [Accessed: 31- Mar- 2022].

[13] "BioSec.Lab PPG Dataset", Comm.utoronto.ca, 2022. [Online]. Available: https://www.comm.utoronto.ca/~biometrics/PPG_Dataset/data_desc.html. [Accessed: 31- Mar- 2022].

[14] D. Hwang and D. Hatzinakos, "PPG-based Personalized Verification System", *2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE)*, 2019. Available: 10.1109/ccece43985.2019.9052394.

[15] "Bitalino", Bitalino.com, 2022. [Online]. Available: https://bitalino.com/downloads/software. [Accessed: 25- Jan- 2022]

[16] "Streamlit • The fastest way to build and share data apps", Streamlit.io, 2022. [Online]. Available: https://streamlit.io/. [Accessed: 31- Mar- 2022].

[17] "Signal-to-Noise Ratio as a Quantitative M | Maxim Integrated", Maximintegrated.com, 2022. [Online]. Available: https://www.maximintegrated.com/en/design/technical-documents/app-notes/6/6410.html. [Accessed: 31- Mar- 2022].

Appendices

Appendix A: Using ArduSpreadSheet to measure and save data

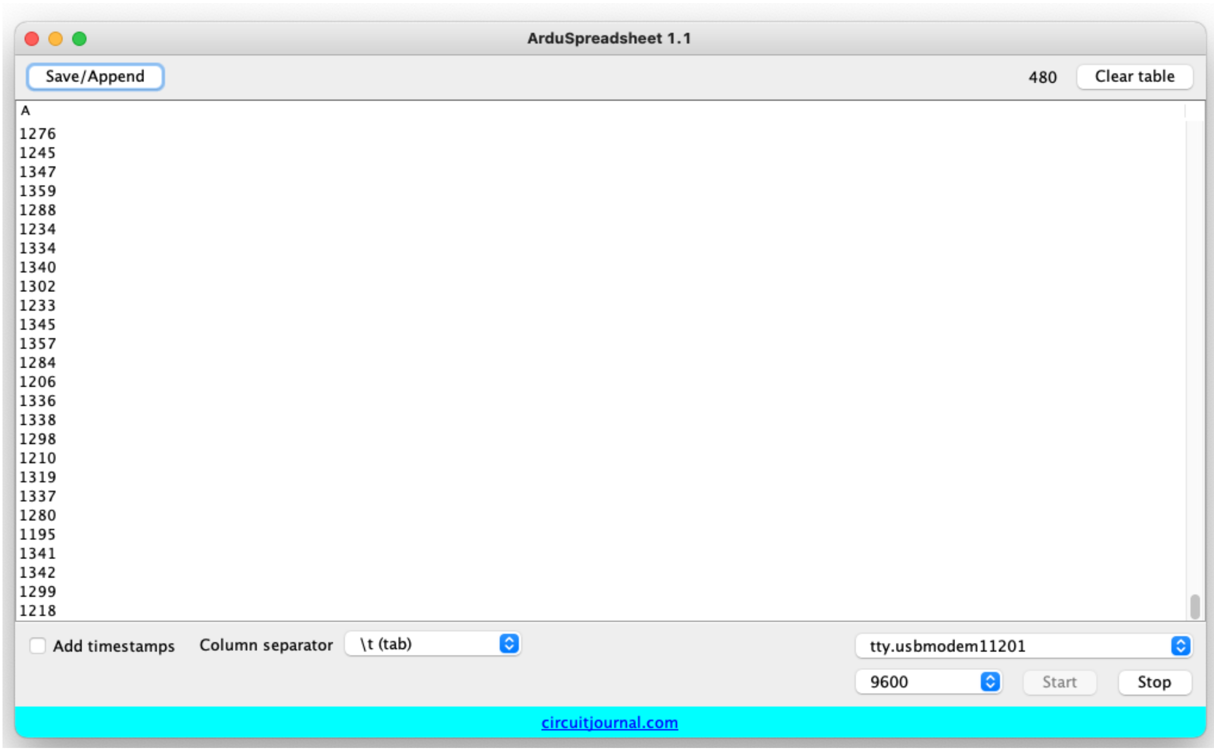


Figure A1: Using ArduSpreadSheet to measure and save data

Appendix B: Calculation of sampling frequency for CPG

	A	B
1	36:41.8	7204
2	36:41.8	7049
3	36:41.8	7166
4	36:41.8	7117
5	36:41.8	7134
6	36:41.9	7121
7	36:41.9	7107
8	36:41.9	7195
537	36:56.4	8236
538	36:56.4	8325
539	36:56.4	8390
540	36:56.5	8231
541	36:56.5	8349
542	36:56.5	8362
543	36:56.6	8317
544	36:56.6	8258

Snipped of saved data

$$Start\ time = 36{:}41.8$$
$$End\ time = 36{:}56.6$$
$$Time\ Taken = 14.8s$$

$$\text{Number of samples} = 544$$

$$Sampling\ freq = \frac{number\ of\ samples}{time\ taken} = \frac{544}{14.8} = 36.756\ Hz$$

$$\sim 40\ Hz$$

Figure B1: Calculation of sampling frequency for CPG device using ArduSpreadSheet

Appendix C: SNR Calculation for Biosec2 Database

```
data = np.load("/content/gdrive/MyDrive/ECE496 Capstone Project/Dataset/biosec2_raw_data.npy", allow_pickle=True)

def calculate_snr(timeseries):
  mean = np.mean(timeseries)
  stdev = np.std(timeseries)

  return 20*np.log10(mean/stdev)

SNRs = []

for x in range(len(data)):
  for y in range(len(data[x])):
    for z in range(len(data[x, y])):
      SNRs.append(np.apply_along_axis(calculate_snr, 1, data[x, y, z]))

print("Average SNR of raw data: {} dB".format(float(sum(SNRs)/len(SNRs))))
```

Figure C1: Code to calculate average SNR of the raw Biosec2 data

Appendix D: CPG Circuit

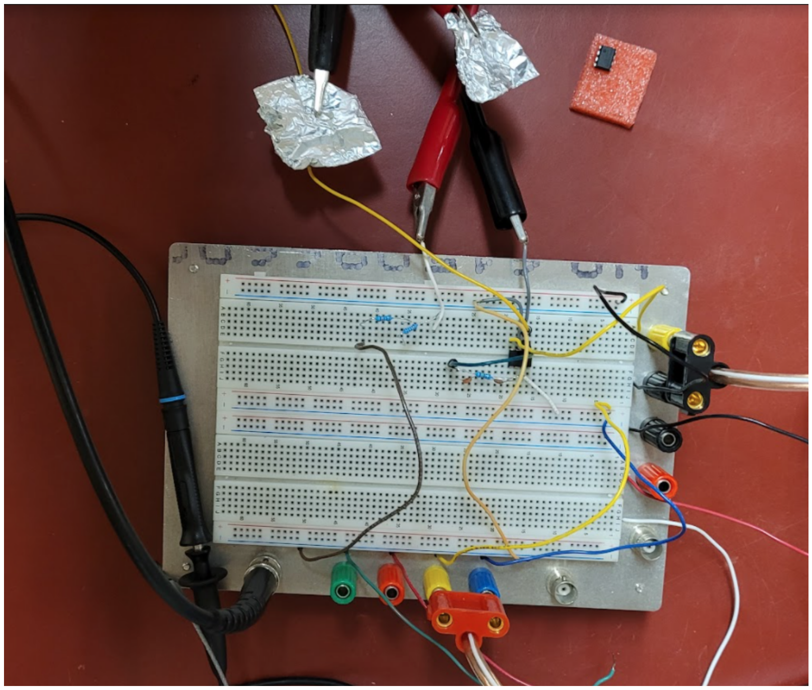


Figure D1: CPG Circuit

Appendix E: CPG Circuit with purchased Capacitive Touch Sensor

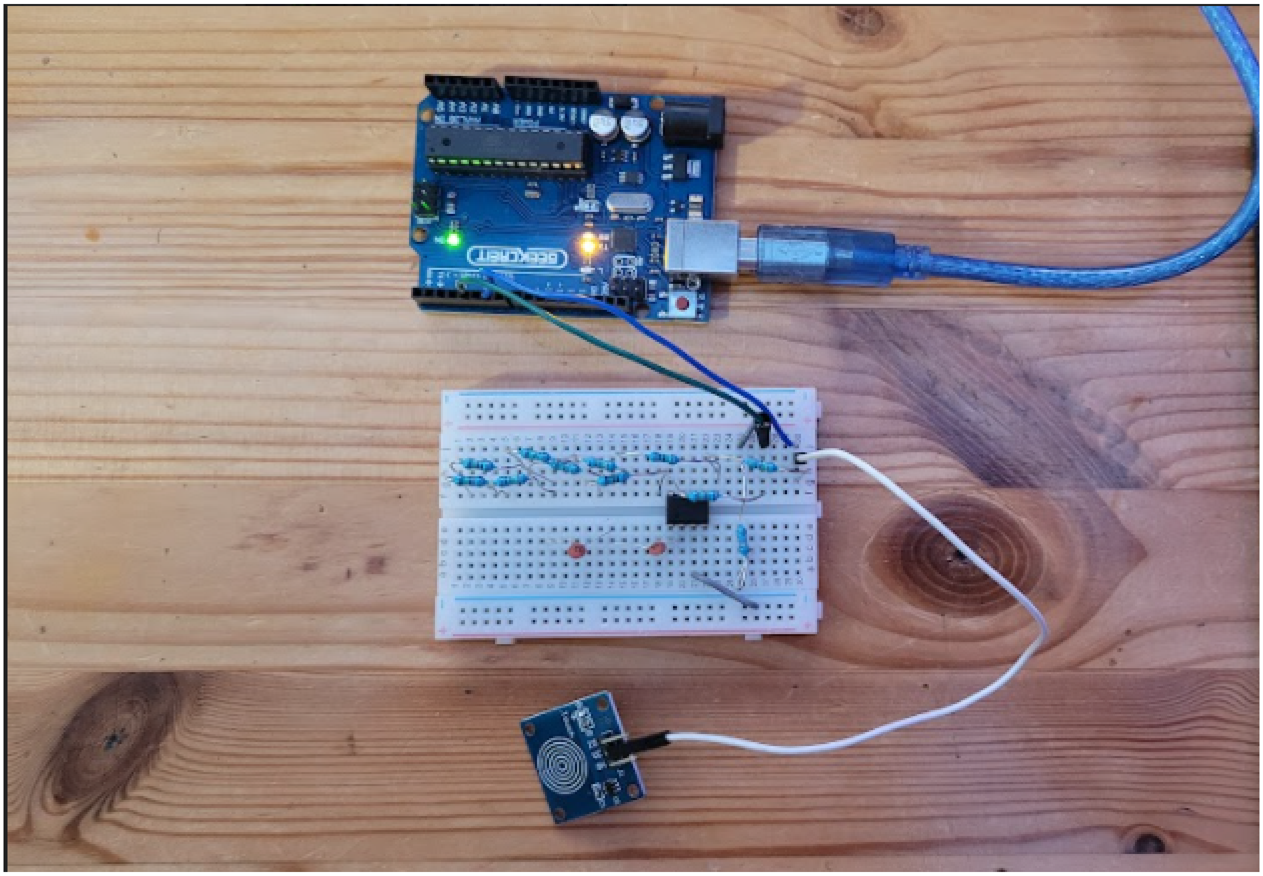
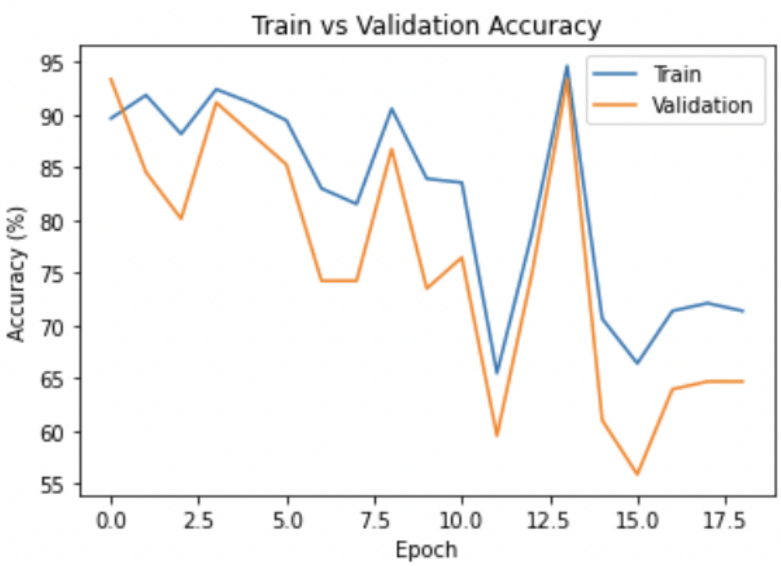


Figure E1: CPG Circuit with purchased Capacitive Touch Sensor

Appendix F: Real-Time Authenticator Performance

```
Class Weights: tensor([ 0.5199, 13.0385])
```




```
Best Validation Accuracy is  93.38235294117648 at Epoch 1.
[[399 267]
 [ 10  17]]
Average final train accuracy: 82.9963099630996
Average final validation accuracy: 80.76470588235293
Average test accuracy: 79.33044733044733
[[13448  3184]
 [  397   296]]
```

Figure F.1: Performance on an unseen participant, and average stats over 25 unseen participants