Palestine Polytechnic University

College of Information Technology and Computer Engineering
Department of Computer Engineering

Graduation Project

Smart Personalized Fitness Coach

Students

Mohammed Shanan, Omar Zatari

Supervisor

Dr. Ayman Wazwaz

2023 - 2024

Acknowledgment

In the name of "Allah", the most beneficent and merciful who gave us strength, knowledge and helped us to get through this project.

To the people that have inspired, supported, and molded us into the people that we are today, our families, friends and our supervisor. We would’ve never been able to reach this achievement without your support, care and encouragement.

We would like to express our sincere appreciation to our graduation project supervisor Dr. Ayman Wazwaz for his guidance, continuous encouragement, and support throughout the semester, for being there to answer our questions and provide us with valuable suggestions and tips which helped us in choosing some system components.

Furthermore, we have to show our gratitude to Eng. Wael Takrouri for being there to answer our questions and provide us with valuable suggestions and tips which helped us in choosing some system components.

Moreover, it is our duty to thank our families for their generous encouragement and continuous support throughout our life. We are truly grateful for all your support throughout the whole education stage. At last, we would like to thank all the people who helped, supported, and encouraged us to successfully finish the graduation project.

Abstract

The project "Smart Fitness Coaching, for Body Exercises" aims to create a fitness solution to meet the demand for personalized, effective and engaging body exercise guidance. This system utilizes methods and user centered design to improve the workout experience.

The main concept involves capturing real time images of individuals while they work out analyzing their movements to ensure form. This analysis helps determine the correctness of exercises. The findings, such as the exercise name and images are shown on a website to assist users.

To make the fitness experience even better the project features real time monitoring of oxygen levels and heart rate during workouts. Moreover, a gym access system using ID cards is included for individualized entry for each user. The information, including images and user records is stored securely. Managed efficiently offering a solution, for fitness training.

Abbreviations

SCL	Serial Clock
IoT	Internet of Things
GPIO	General Purpose Input/output
VCC	Voltage, Common Collector
GND	Ground
PC	Personal Computer
USB	Universal Serial Bus
SDA	Serial Data
MQTT	Message Queuing Telemetry Transport
PHP	Hypertext Preprocessor
PY	Python
Esp32	Espressif modules
DB	Database
IDE	Integrated Development Environment

Table 1:Abbreviations

Table of Contents

List of Figures

List of Tables:

CHAPTER ONE
Introduction

1.1Preface

The "Smart Fitness Training for Body Exercises" project aims to develop innovative fitness solutions to address the need for appropriate and effective physical exercise training This program includes advanced technology and system a it involves the user together to improve the body's exercise experience.

1.2 Project Aims and Objectives

Project Aims:

This project is aimed for an overall goal to create Smart Personalized Fitness Coach for body exercises next level. This creative system shall envisage an amalgamation of hardware elements, Internet-of-Things (IoT) technology, and sophisticated Artificial-Intelligence (AI) algorithms. Their main features will include real-time direction during exercises, form-specific feedback and individual routines.

In this way, the project will have an impact on the suitability and effects of exercises that involve the entire body. The project also seeks to encourage long-term compliance with fitness regimens by making workouts fun, customizable and accessible, in an effort to support users in overall health and wellness, regardless of fitness level.

Project Objectives:

- Automated real time images capturing system for individuals performing exercises and body movement analysis using sophisticated algorithms.
- analyzed data to a controller to check whether the exercises were performed in the right way.
- Show exercise names and images on a web-based application to give feedback on exercise performance.
- Efficient modes of data transmission suitable for unimpeded communication between all parts of the system.
- Implement a pulse oximeter with heart rate monitor that measures during exercise
- Create a consolidated user interface that provides an effortless method of interfacing with exercise tracking and feedback on the web platform.
- Gym login using a unique ID card(User)
- Store login details and the image that will be captured in a secure structure to have a safe record of your gym entry.

1.3 Problem Statement

Even though fitness training is gaining a lot of popularity, many people do not execute exercises in the right way and do not even follow the appropriate health metric tracking while they exercise. The problem this causes is a poor workout, possible injuries and a lack of progress towards our goals. However, most of the solutions in the market do not monitor physical activity in real-time and lack a comprehensive monitoring system that can be used effectively for users to fine-tune their exercise routine.

What is a big problem with most fitness monitoring systems today? How about: The Have no analysis of exercise performance, and they are not integrated with real health metrics tracking. We see the result of this gap with consumers struggling to accurately measure their efforts and design effective fitness training.

Our proprietary technology delivers custom body exercise plans designed to invigorate your wellness routine with body-first workouts for fitness aficionados. They have done so by marrying smart technology with.

1.4 Project Requirements (Functional and Non-Functional)

Functional Requirements:

- Capture images of people exercise while they do exercises.
- Process body movements from the images captured by the most advanced algorithms.
- Send Analyzed data to a controller to assess exercise performance is correct.
- Use a fast protocol for data transmission: From image capture system to web platform.
- Build a user-friendly web-interface for the ease of recording and reviewing exercises.
- Display names and pictures of exercises on a web platform; suggesting whether these exercises were well executed.)
- Integrate an oxygen saturation measurement sensor and heart rate during exercises.

Non-Functional Requirements:

- Performance: The system should return feedback to the user with very little lag ensuring a smooth user experience.
- Scalability: For any system figured out the number of users and exercises should be handily taken care off without bringing down the performance of the application.
- Security: To have data protection and confidentiality both in transmission and storage of user data while using the website

- Compatibility: The system must be accessible and compatible with a large number of devices and web browsers.
- Maintainability: Well documented and the codebase to be modular for easy future enhancements.
- Usability: the interface is very intuitive and simple for users at all levels of IT expertise

1.5 System Description

The proposed system comprises of a fitness monitoring system that will act as a feedback system to offer data feedback in time during exercises, workouts and any physical activity. It has various sub-systems such as image capturing mechanism, movement tracking system, data communication system, health monitoring system, health indicators measurement, and web-based user interface.

It commences with image acquisition, done by a camera that helps to acquire live images of people as they perform exercises. These images are then passed through an algorithm that compares different body postures to evaluate the accuracy of exercises which will be discussed in the next chapter, Chapter 2.

If the analysis is done, then next data goes to verification of exercise. The controller proposes the analyzed movements compared to the standard exercises and offers information related to exercise accuracy. During exercise sessions, participants' physical health data such as oxygen concentration and pulse rate are recorded. Some feedback systems master the use of an auditory mode to inform the users that movements are incorrect or deviate from a right exercise form.

The analysed exercise data and the chosen health metrics are then relayed to the web platform. This interface allows users to easily view information on exercise feedback, Health, and performance trends. Furthermore, the data accrued from the users is saved in a database which facilitates surveillance of the users' progress in the subsequent weeks.

The benefits of this system are relevant real-time feedback, exhaustive monitoring of both the exercise correctness and the state of health, engaging a user by an easily understandable web interface, while providing the user with useful and helpful information on their health and fitness. Also, data stored in a database means that users can monitor their goals and decide on their fitness objectives in case of installing this system.

Also, the system has included an identification system readed by a sensor where the users can log in the gym with a special identification card. Log-in information and images acquired are preserved through the use of safe technology hence adequately protecting records of access to the gym.

1.6 Project Limitations/Constraints

Project limitations include:

- Constraints on the depth of AI model training due to resource availability.
- Dependence on reliable internet connectivity for IoT functionality.
- Limited availability of specialized technical expertise.
- Time constraints for project completion.

1.7 Project Schedule

The project will span eight months, divided into key phases:

- **Introduction Phase (4 Months):**
  - Month 1: Project Selection and Proposal
  - Month 2: Research and Planning
  - Month 3: Find the hardware and parts needed for the project
  - Month 4: Software Development Planning

		الاسم	المدة	البداية	النهاية
1		Project Selection and Proposal	13 days	ص 08:00 15/10/23	م 05:00 01/11/23
2		Research and Planning	15 days	ص 08:00 02/11/23	م 05:00 22/11/23
3		Find the hardware and parts needed for the project	18 days	ص 08:00 23/11/23	م 05:00 18/12/23
4		Software Development Planning	20 days	ص 08:00 19/12/23	م 05:00 15/01/24
5		documentation	70 days	ص 08:00 15/10/23	م 05:00 19/01/24

Table 2: project GANNT(introduction)

- **Graduation Project Phase (4 Months):**
  - Month 5: Hardware Procurement and Setup
  - Month 6: IoT Integration, Database and Software Development
  - Month 7: AI Model Training and Testing
  - Month 8: Testing, Refinement, and Final Presentation

		الاسم	المدة	البداية	النهاية
1		Hardware Procurement and Setup	13 days	ص 08:00 15/02/24	م 05:00 04/03/24
2		IoT Integration, Database and Software Development	15 days	ص 08:00 05/03/24	م 05:00 25/03/24
3		AI Model Training and Testing	18 days	ص 08:00 26/03/24	م 05:00 18/04/24
4		Testing, Refinement, and Final Presentation	20 days	ص 08:00 19/04/24	م 05:00 16/05/24
5		documentation	67 days?	ص 08:00 15/02/24	م 05:00 17/05/24

Table 3: project GANNT(Graduation Project)

CHAPTER TWO

Theoretical Background

2.1 Preface

The subject matter of this chapter is fitness technology, a fusion of sophisticated sensors, artificial intelligence and machine learning that are aimed at enhancing physical training and monitoring health. It distinguishes itself from the rest by specifically focusing on creating a smart Personalized Fitness Coach for Body Exercises that is highly integrated with most advanced technologies. This review compares these two projects, as well as highlighting their goals, approaches and technicalities that make them stand out.

2.2 Theoretical Background

1. MediaPipe Algorithm:

A Google-developed open-source framework called MediaPipe was used to develop cross-platform applications that deal with real-time recognition of human body gestures, facial expressions, hand poses and objects among other things. It is backed by machine learning models that are trained on large volumes of data sets thereby making it possible to perform precise and efficient inferences on different tasks.

To this end, we utilize CNN-based pose estimation model in Media Pipe. These are networks which have been learnt to detect keypose points thus enabling us assess movements as well as the correctness of exercises taking place in real time.[18]

Google's MediaPipe Pose is a next-generation machine learning system for tracking body poses in real-time. This model has advanced technology to recognize joints and bones on the body part that help analyze movement and rate performance. See Figure 1 below:

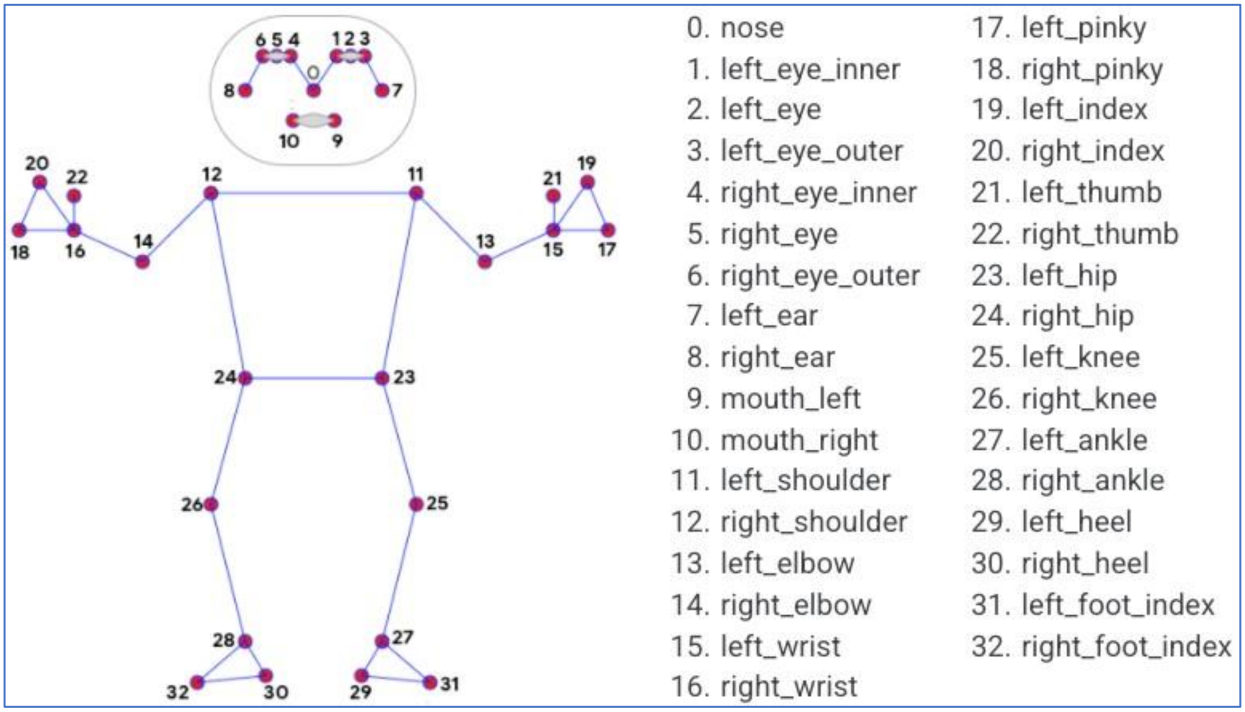

Figure 1: Media Pipe's32 pose [25]

2. Numpy & CV2 Algorithms:

Python Numpy is an excellent library for numerical computing, with support for the creation of large multidimensional arrays and matrices as well as a set of mathematical functions that make operations on such arrays faster.

CV2 (OpenCV) is an open-source computer vision software library that specializes in image processing and analysis. Some of its functions include: edge detection, contour detection, object recognition among others.

When working on our project, both Numpy and CV2 are used in tandem for image processing and analysis. Numpy arrays can be used to represent digital images efficiently while CV2 has functions like filtering, resizing or contour detection which are critical in preparing images before feeding them through the Mediapipe pose estimation model.

3. MQTT Protocol:

A lightweight messaging protocol, the MQTT (Message Queuing Telemetry Transport) is designed for constrained environments such as those with low bandwidth or high latency. In this messaging pattern, it has a publish-subscribe operation in which clients could either send messages to topics or subscribe to them and receive messages.

Our project uses MQTT for efficient data transmission between image capture system, RPI controller and web platform. It ensures that exercise data and health metrics are delivered reliably and in real-time thus facilitating smooth integration among various components of the system.

4. MySQL Database:

Reliability, scalability, and performance are some of the qualities that make MySQL a popular open-source RDBMS. It provides a strong foundation for storage and control of structured information which makes it best for applications needing to store, retrieve or analyze data.

In our project we have employed MySQL as a storage system for various kinds of information such as exercise feedbacks, health metrics among others. This enables the users to monitor their progress over time, identify patterns and devise plans in order to achieve their fitness targets. Furthermore, it guarantees the safety and integrity of stored data offering a reliable scalable solution for managing it.

5. Sensors:

The complete MAX30100 sensor is an integrated pulse oximeter and heart-rate monitor sensor. It uses red light and infrared rays to measure oxygen saturation (SpO2) and identify heart rate (HR). The sensor emits light onto the skin and takes measurements of the amount of it that is absorbed or reflected by blood vessels thus enabling it calculate SpO2 and HR.

The ESP32 microcontroller interfaces with the sensor, processes its data, and sends real-time reports to the Internet. Hence, people can monitor their oxygen saturation as well as heart rate during workouts providing essential knowledge about their health status as well as performance history.

Through incorporation of MAX30100 sensor into the fitness monitoring system, users are able to ensure that they exercise within limits that are safe for them hence optimizing their workouts for maximum effectiveness while minimizing the risk of overexertion or injury.

This technology involves using radio waves in finding an item around you simply by broadcasting a signal through a tiny chip which has been programmed with information specific to the object being searched. These chips may be regularly read using RFID readers or scanners which retrieve electronic information from stored chips on your tags. In addition, the RFID system has three main components: tags, readers and database systems. ESP32 Controller with RFID Sensor: It's used in gym entries where users have special identification cards.

6. Internet of Things (IoT) Principles:

IoT involves the interconnection of computing devices embedded in everyday objects, enabling them to send and receive data and enables interconnected devices to communicate and share data. IoT principles are used to facilitate communication between the fitness trackers, Raspberry Pi 4, and the cloud server.

As the IoT connectivity is central to the project, facilitated by the Raspberry Pi 4. This device is selected for its advanced processing and networking capabilities, distinguishing it from more basic options [2]. Where the Raspberry Pi 4's acts as an edge device, is vital for ensuring seamless communication between the wearable fitness trackers and the cloud-based system, thereby maintaining the integrity and efficiency of the data flow.

7. Voice Recognition Technology:

In our system, Voice Recognition is implemented as an exigent feature for voice-enabled applications using SpeechRecognition API to understand the commands given by voices. Another notable feature of this API is that it relies on the NLP techniques for assessing user intent and converting it into actionable input. Through the incorporation of this technology, our web application also provides users with voice control, which can assist in the overall functionality of the system.

A hands-free and easy-to-control application was also required as a primary need, which is why the SpeechRecognition API was added. Although touch interfaces are also provided for the purpose of the manual command input, the voice recognition functionality is noteworthy by its true convenience, as to compared to the touch input, the voice command is much more engaging and efficient for the user in the cases when direct touch is either impossible or not comfortable.

8. Cloud-Based Storage:

Firebase is an extensive platform developed by Google that presents developers with numerous services for creating web and mobile applications. It provides developers with tools, infrastructure and monetization options to help design, scale and make money from their apps.

Firebase technology: Utilized to safely stock captured images or RFID sensor information within a scalable cloud atmosphere that ensure accessibility as well as reliability of stored information.

9. Buzzer:

It's an electronic gadget that produces an auditory signal, typically a warning or alarm. In the instance of our project, audible feedback can be given by a buzzer when a user completes a legitimate movement from the prescribed exercise routine.

2.3 Literature Review

Below, we will introduce some of the previously proposed and projects that are similar to this project:

- **Custom IMU-Based Wearable System for Human Body Parts Orientation Tracking and 3D Movement Visualization. [4]**

Description:

This project presents a custom wearable system with IMUs for tracking human body parts orientation and visualizing 3D movements. It addresses the limitations of expensive high-end commercial solutions and less featured entry-level ones by offering an affordable and robust solution. The system uses a custom 2.4 GHz communication protocol to ensure reliable data acquisition in environments crowded with Bluetooth and Wi-Fi signals.

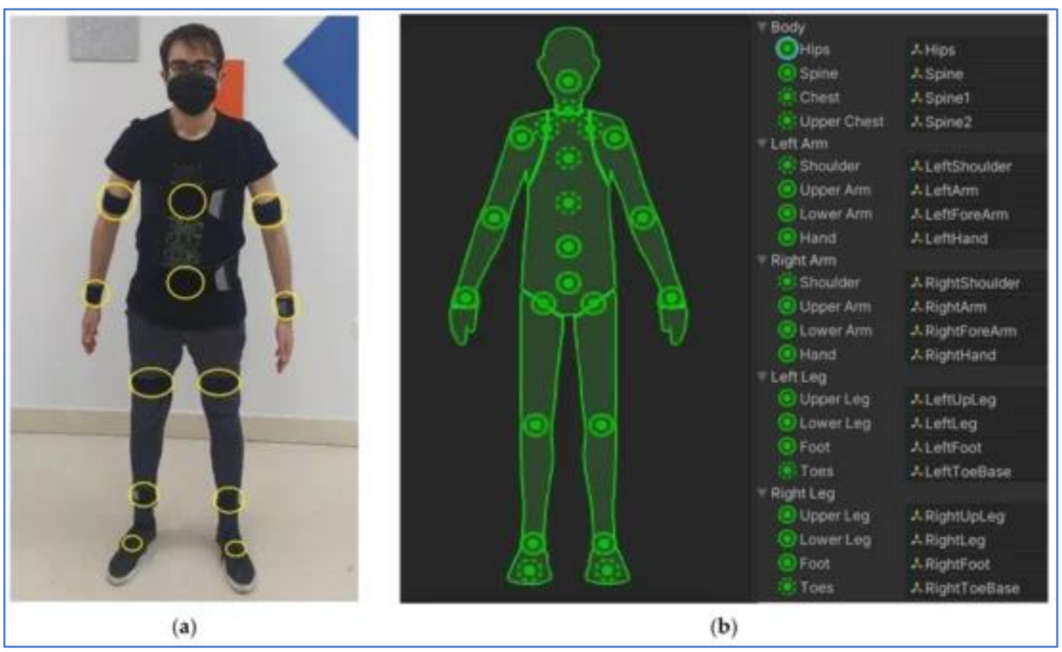

Figure 2: sensors placement on the human body [4]

Features:

1. **IMU-Based Sensors:** Incorporates BNO080 inertial sensors, offering high responsiveness to environmental disturbances.
2. **Wireless Communication:** Utilizes a custom 2.4 GHz protocol with channel hopping to minimize interference and enhance data transmission reliability.
3. **Real-Time Data Acquisition:** Supports up to 10 sensors operating at least at 50 Hz for comprehensive body movement tracking.
4. **3D Movement Visualization:** Employs Unity3D for representing movements on a virtual avatar, offering an immersive visual feedback experience.
5. **Sensor Data Representation:** Uses the ENU (East-North-Up) coordinate system for intuitive spatial understanding.

Limitations:

1. **Kinematic Model Constraints**: The system primarily estimates bone rotations, not their absolute positions in 3D space, limiting spatial movement analysis.
2. **Application-Specific Design:** More suitable for applications that do not require highly detailed kinematic analysis.
3. **Testing and Validation:** Needs extensive testing against standard optical systems for validation in clinical or sports applications.

◈ Portable Smart Fitness Suite for Real-Time Exercise Monitoring and Posture Correction.[5]

Description:

This project develops a portable smart fitness suite, facilitating at-home exercise routines with real-time monitoring and posture correction. Focused on two exercises (T-bar and bicep curl), it employs gyroscope and EMG sensors integrated into smart fitness clothing, coupled with an Android application that serves as a virtual gym trainer.

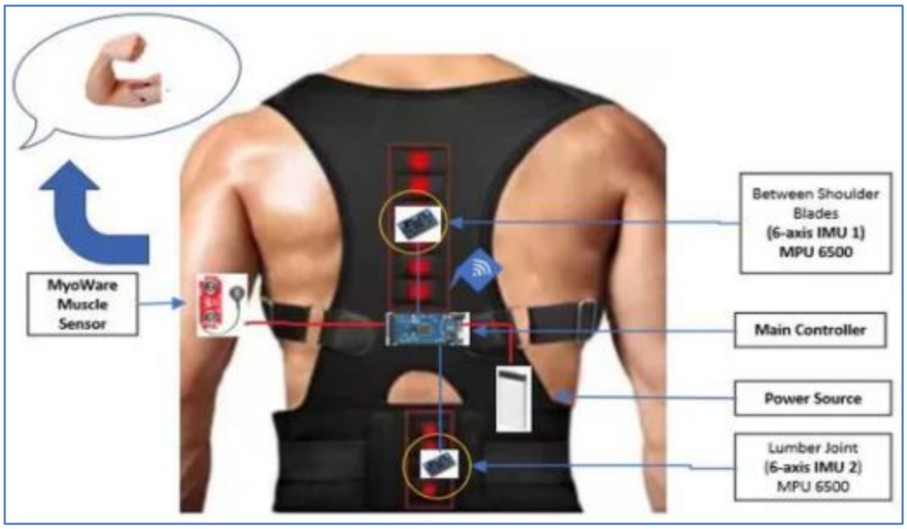

Figure 3: sensors orientation on [5]

Features:

1. **Integrated Sensor Technology:** Features MPU 6500 IMU sensors and MyoWare EMG sensors for precise muscle movement monitoring.
2. **Android Application Interface:** Provides real-time feedback and posture guidance, acting as a virtual gym trainer.
3. **Exercise-Specific Design:** Tailored for T-bar and bicep curl exercises, addressing common workout issues like improper posture.
4. **Machine Learning Classification:** Utilizes models like KNN for exercise classification and guidance, achieving up to 89% accuracy.
5. **Cloud Integration:** Processes and stores data on cloud platforms for enhanced data management and scalability.

Limitations:

1. **Limited Exercise Scope:** Initially designed for only two specific exercises, which may restrict its applicability for a broader range of workouts.
2. **Sensor Dependency:** The effectiveness of the system heavily relies on the precision and reliability of the embedded sensors.
3. **Internet Requirement:** Needs a stable internet connection for cloud-based functionalities.
4. **Model Accuracy:** The system's effectiveness is contingent on the accuracy and robustness of the machine learning algorithms used.

❖ AI Fitness Trainer - Build Using MediaPipe for Squat Analysis. [6]

Description:

The AI Fitness Trainer leverages MediaPipe's Pose pipeline, a deep learning-based human pose estimation algorithm, for high-accuracy body pose tracking. It infers 33 3D landmarks on the body from video frames, using technologies like BlazePose. This system aims to help users perform squats with proper form and technique.

Figure 4: BlazePose Landmarks [6]

Features:

1. **Pose Estimation:** Utilizes MediaPipe Pose, which tracks the body in 3D space using 33 landmarks.
2. **Two-Step Detection:** Includes a detection-tracking pipeline for identifying the person and predicting pose landmarks.
3. **Feedback Mechanism:** Provides feedback based on the analysis of various angles like hip-knee, knee-ankle, and shoulder-hip with the vertical.
4. **Modes of Operation:** Offers Beginner and Pro modes with tailored feedback.
5. **Feedback Messages:** Includes specific feedback like bending forward, lowering hips, and adjusting knee positions.
6. **Inactivity Detection:** Resets counters for correct and incorrect squats based on inactivity.

Limitations:

1. **Limited Exercise Range:** Initially focuses on squat analysis, which may limit its application to other exercises.
2. **View Dependency:** Relies on the side view of the person for accurate estimations, which might not capture all aspects of the exercise.
3. **Feedback Specificity:** The feedback is based on preset thresholds and may not account for individual variations in body types and fitness levels.

2.4 Summary Literature Review

In the development of our project, we scrutinized current technological solutions in fitness and motion analysis. the Portable Smart Fitness Suite that focused on fixing postures during home exercises in real-time and AI Fitness Trainer that was using MediaPipe to analyze squats.

We have therefore realized that there is a need for a system that is accurate and user-friendly across many different exercises. From these systems' features such as AI/ML algorithms for instant feedback, as well as advanced pose estimation techniques, our project concept has been generated. We are aiming at creating an intricate robust solution for body workouts to bridge the gap left in fitness technology market.

2.5 System components (software and hardware)

2.5.1 Hardware Components

1. Microcontroller: ESP32

A core that acts as an ESP32 microcontroller is the central component of this system. One of its features is the ability to process data from MAX30100 and other sensors, alongside display and audio outputs, wireless communications included. It has integrated Wi-Fi and Bluetooth which enable it to easily connect with mobile applications and cloud servers. The ESP32 also performs well in terms of converting raw sensory data into useful information and maintaining stable wireless connections for data transmission, firmware updating and remote-control operations. [7]

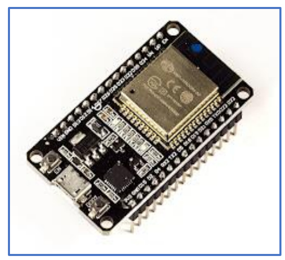

Figure 5 :ESP32

2. IoT Gateway: Raspberry Pi 4

As a crucial IoT gateway in the system. The Raspberry Pi 4 is the main pedestrian to complete data transfer. Its main function is to act as an intermediate device, gathering and processing data from the ESP32 and the camera, before sending it to the cloud storage and web browser. The Raspberry Pi 4 is well equipped to handle complex processing jobs, which might be beyond the capability of the ESP32. Its processing power not only enhances performance of the system as a whole but also ensures that local hardware and cloud services run together smoothly.[8]

Figure 6 :Raspberry Pi 4

3. Sensors: MAX30100

The MAX30100 is an integrated pulse oximetry and heart- rate monitor sensor solution. It combines two LEDs, a photodetector, optimized optics, and low-noise analog signal processing to detect pulse oximetry and heart-rate signals.

Figure 7 :MAX30100

The MAX30100 operates from 1.8V and 3.3V power sup- plies and can be powered down through software with negligible standby current, permitting the power supply toremain connected at all times. [9]

4. USB Camera

A USB camera, also called a webcam, is a camera device linking to computer or other compatible equipment through what is known as "the USB port". For the fitness monitoring platform, a USB camera can serve as an essential part of taking real-time footage showing people exercising.[10]

Figure 8 :USB Camera

5. Microphone

The microphone in the system is really necessary. Thus, making this one of those handy devices that just about anyone can use without even needing hands two ways have voices an active popular that features Improved Accessibility & Convenience This component is especially valuable to users in situations where they could not handle things manually, like a workout or going out for a walk and cannot lock onto anything themselves anymore. [11]

Figure 9: Microphone

6. Speaker

The system's speaker provides invaluable auditory feedback on workout instructions, exercises and motivational phrases. More than just outputting the audio signals, it also acts as an interactive trainer for routines that people are new to. The speaker's prompts and directions are key in guiding the practice of the exercises, while its reminders help to make user interaction more natural and engaging. This can produce a workout experience that is more pleasurable. [12]

Figure 10 :Speaker

7. Display Screen

The display is vital in the system, offering visual feedback and instructions for a more intuitive user experience. It displays exercise routines, real-time body positioning, and progress updates, making workouts clearer, more engaging, and effective. [13]

Figure 11: Display screen

8. Power Source: 9V Battery for ESP32

The ESP32 relies on a 9V battery, ensuring mobility and autonomy. This portable solution, with low power consumption, makes the fitness system ideal for outdoor workouts or scenarios without a power outlet, offering users a reliable and convenient power source for extended usage. [15]

Figure 12: 9V Battery

9. RFID (Radio Frequency Identification)

RFID (Radio Frequency Identification) is a technology that uses electromagnetic fields to identify and track tags attached to objects. These tags keep the digital storage of the components that can be studied alternately with the RFID reader or RFID scanner. The three main components of the RFID system are tags, readers, and a database. [23]

Figure 13: RFID

RFID technology has a number of advantages over the traditional barcode systems, however:

1. Efficient and automated: RFID system use automatic identifying and trace ability of objects by means of RFID tags without the need to change the line of sight or manual scanning.
2. Speed: RFID can read multiple tags simultaneously in comparison to barcode systems, at higher speed.
3. Durability: RFID tags can endure till the extreme climatic conditions like dirt, moisture, snow, and so on.

10. Buzzer

An A buzzer is an electronic device that issues a sound signal and is usually used to signal by alerting or warning. For our project, a buzzer can be implemented to alert the user with auditory feedback if inappropriate (deviant) exercise or movements are performed from what is expected.[22]

Figure 14: buzzer

2.5.2 Software Components

- **Raspbian:**

Although Raspberry Pi can utilize different types of Linux distributions for its default operating system (OS), Raspbian is the recommended one. It is a free operating system based on Debian, optimized for the Raspberry Pi hardware. It comes with over 35,000 packages; precompiled software bundled in a good format for easy installation on Raspberry Pi hardware, [16].

A free operating system based on Debian optimized for the Raspberry Pi hardware. Coming with over 35,000 packages, pre-compiled software. This operating system provides significantly faster performance for applications that make heavy use of floating-point arithmetic operations [17].

- **Arduino IDE**

Arduino code is written in C+ with an addition of special methods and functions. C++ is a human-readable programming language. The Arduino Integrated Development Environment (IDE) is the main text editing program used for Arduino programming. It is where you'll be typing up your code before uploading it to the board you want to program. Arduino code is referred to as sketches, It is processed and compiled to machine language, and Arduino sketch code is used to program wemos microcontrollers.[14]

Fritizing

It is an open-source initiative to support designers, artists, researchers and hobbyists to take the step from physical prototypes to actual product. We make this software in the spirit of processing and Arduino, developing a tool that allows users to document their Arduino and other electronic models, and create a PCB layout for manufacturing. The complementary website helps users share and discuss drafts and experiences as well as reduce manufacturing costs.

Fritizing is essentially electronic design automation software with a low barrier to entry, suitable for the needs of designers and artists. It uses the metaphor of a breadboard, so that it is easy to transfer your hardware drawing to the software. From there, it is possible to create PCB layouts to turn them into a robust PCB on your own or with the help of a manufacturer.[19]

XAMPP

It is the program used to create a virtual local server. The name is an abbreviation, where each letter represents one of the five main components. The software package contains the Apache web server, the MySQL (or MariaDB) database management system, and the Perl and PHP programming languages. The initial X indicates the operating systems it works with: Linux, Windows, and Mac OS.[20]

2.6 Software technologies

This section will provide some information about the main software technologies used in our project .

- ❖ **MQTT protocol**

The MQTT Dashboard utilizes the HiveMQ MQTT broker. You can use any MQTT client or library to publish to the broker. Testing and usage is for free but please do not use it for sensitive information because everybody is allowed to subscribe to every topic, including wildcard. Feel free to play with MQTT and the HiveMQ broker. Please consider to add a reconnect logic to your client because we may update the underlying HiveMQ instance at any time, so we cannot promise 100% uptime. With our free, fully managed MQTT Cloud Platform HiveMQ Cloud you can create reliable, scalable and secure MQTT cloud-broker clusters that are built for production. Sign up and you are ready to connect up to 100 IoT devices at no cost (no credit card required).[21]

MQTT broker /https://broker.hivemq.com

- ❖ **Firebase**

Firebase is a powerful web browser development platform developed by Google and provides a wide array of tools and services to develop high-quality apps. This is owned by google itself and is a complete suite for developers using which they can build, enhance and expand their applications. What is Firebase and A brief overview of its parts:

1. **Database :** A NoSQL cloud-hosted database, store and sync data in real-time It is especially adept for applications that demand real-time updates like chat or collaborative tools.
2. **Cloud Fire store:** Is another NoSQL database provided by Firebase. it is created to scale, and to keep the data synchronized in all devices. The query capabilities are much more powerful, and the data model is also more flexible.[24]

CHAPTER THREE

System Design

3.1 Preface

This chapter focuses on the intricate design aspects of the Smart Personalized Fitness Coach for body Exercises. It delves into the components, both hardware and software, that collectively form the backbone of the system. By exploring design alternatives and methodologies, the chapter aims to provide a comprehensive view of the project's technical foundation.

3.2 Block diagram

The figure is the architecture of an integrated system with these components:

- Local server A local server operated by PHP server with MySQL database is used to handle and save data effectively with getting data.
- Firebase is used to store historical data and images and has allowed remote access to data and scalability of the system through FireBase Cloud storage.
- Web Browser - JavaScript allows the user to interact with the system through a web browser, so that a user can get real-time feedback and control.
- Sensors: RFID RFID sensor and a MAX30100 pulse oximetry heart rate sensor that is connected to ESP32 to collect critical health monitoring and system interaction data
- Data transmission protocol - WRT MOQA protocol is used for reliable data transmission between sensors and other system components over WiFi network.
- Raspberry Pi Configuration: Uses raspbian OS for board control and_low-level programming, mediapipe for image and motion analysis, Open CV (CV2), numpy for image manipulation and verification of exercise Performance requires USB

imaging in images with automated detection of body movement in comparison to the prescribed exercise plan.

- Peripherals: The microphones and speakers interfaces facilitate native audio input as well as output and proves to be beneficial in communication as well as system performance.
- Warning system: The buzzer which is related to the raspberry provides sound alarms to invite the attentions of the user to important occasions or warnings while exercising taking into consideration analysis and integrity.

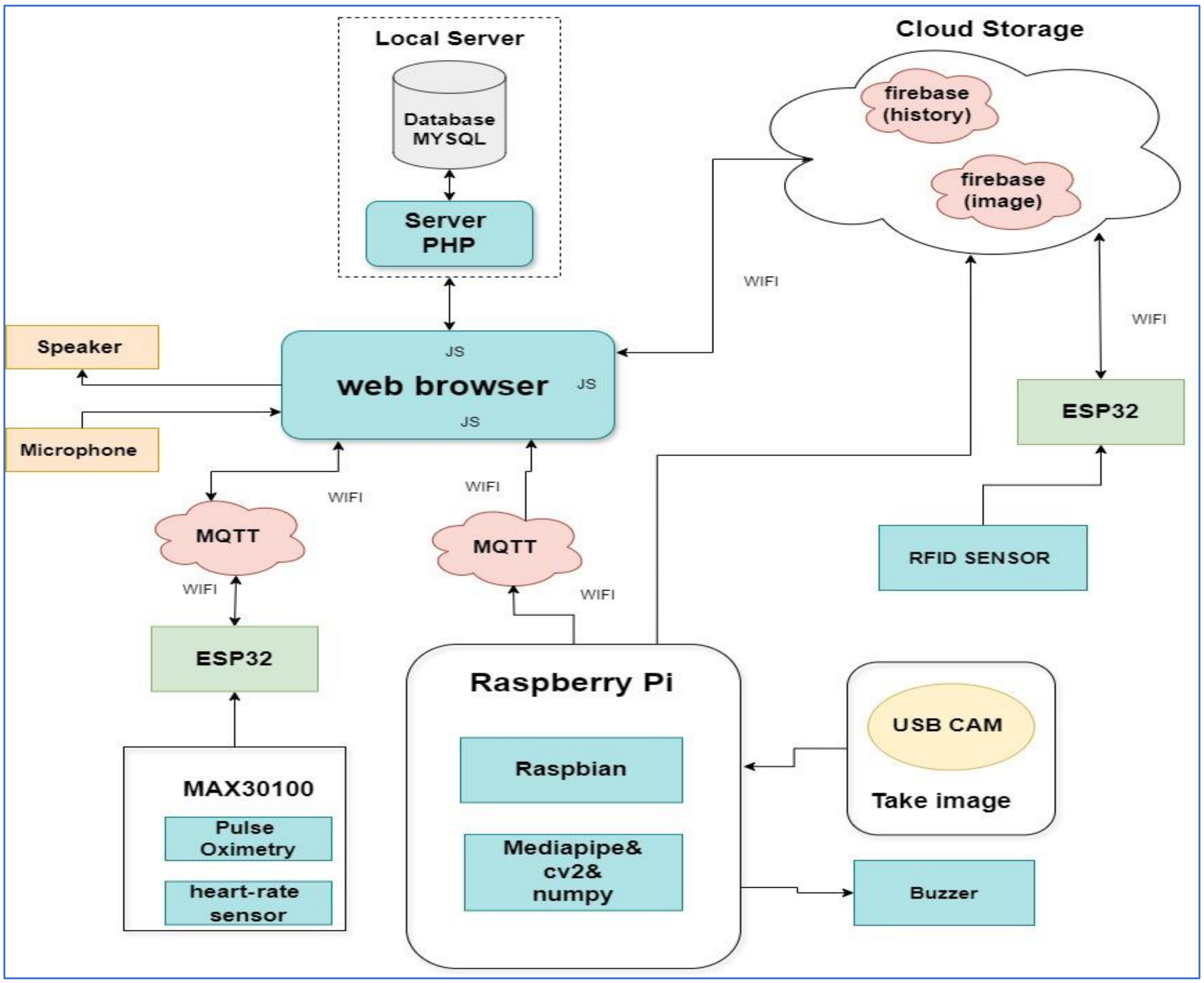

Figure 15: block diagrame

3.3 Software Algorithms (flowchart)

The proposed system begins by establishing all the apparatus needed to track the degree of fitness continually. The process begins at the image capture stage which involves the acquisition of images of people performing exercises using a camera. The identified images are then passed through algorithms that tend to recognize body movements and check the correctness of the exercises. It can be seen that the system carries on the image capture and processing operation until it may attain an adequate image.

In the last step, the system checks or validates the exercise after an effective processing step. If it correctly, the program gives a positive feedback whereas if wrong, it reveals the correct answer and rings a bell.

At the same time, a characteristics the heart rate and O2 aftar the exercise session. The data gathered, such as the feedback produced as a result of the individual exercises and health indicators, is then shared through MQTT to a web-based which provides a graphical representation of the users performance and overall health trends for ease of use.

All data delivered by the application is further preserved in a MySQL database. Furthermore, the system also has an RFID sensor that works with an ESP32 board and contains a login system and corresponding pictures saved in Firebase.

The detailed features of this program includes image analysis, health check up, real time data transfer as well as secure storage of data to improve the fitness and health check up of the user.

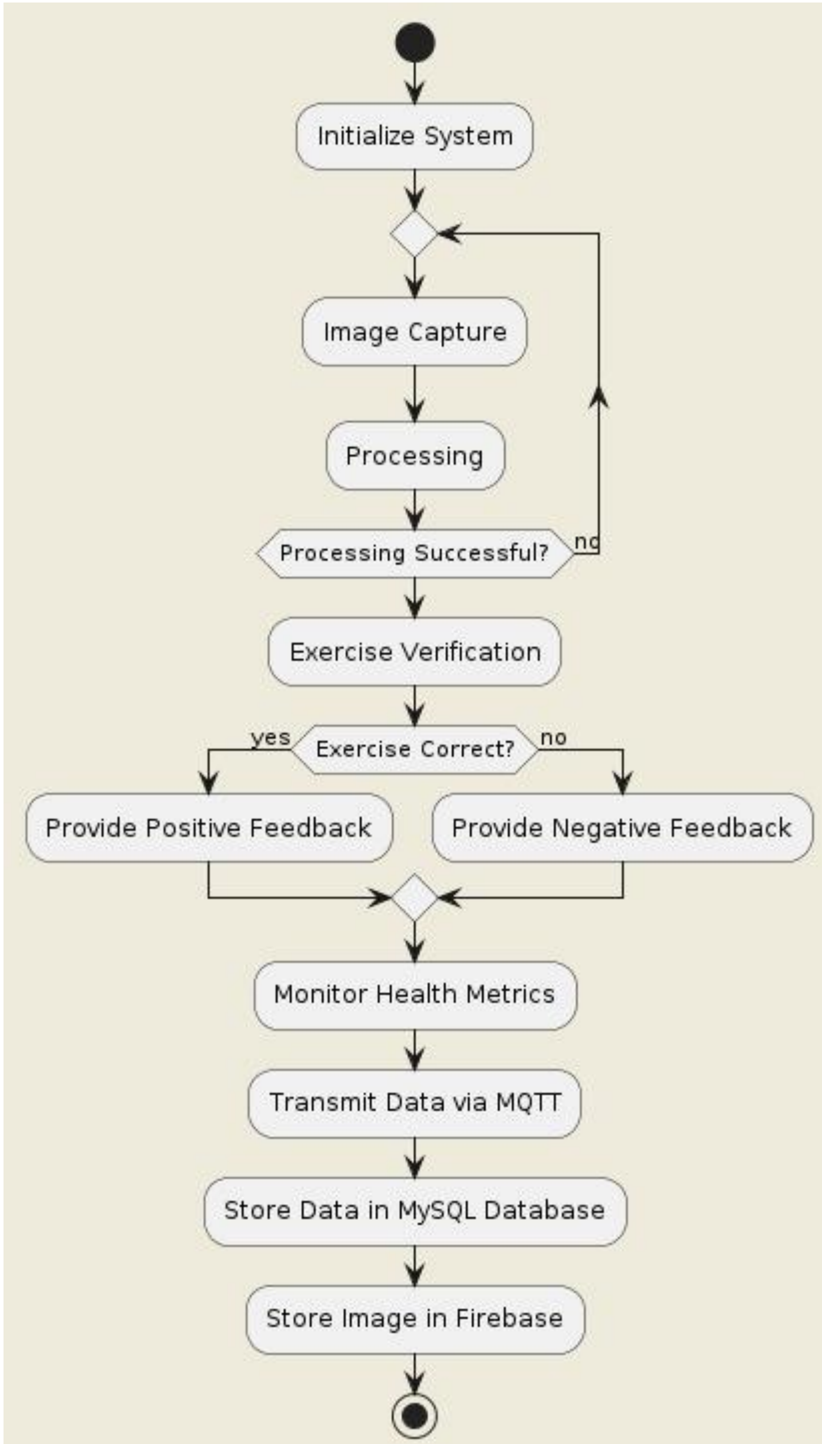

Figure 16: flowchart

3.4 System Schematic Diagrams

System Schematic Diagrams of this project for this project provide a clear physical representation of how different elements will connect together and work cohesively to support the functionality of the elaborate fitness monitoring system. These diagrams are valuable to know about how to assemble disassemble and maintain this device, so it gives good perspective about the interior.

In this section, some components are illustrated with the interfacing of the ESP32 microcontroller as indicated in the following figure; These equipment's will be physically connected by its pins as is depicted in the following figures. It makes sure that all in the system work and share data effectively, for instance, providing feedback on the performance of workouts and general health status throughout workouts.

- ESP32 connect with MAX30100 :

ESP32 microcontroller	MAX30100
VCC	VCC
GND	GND
GPIO23	SCL
GPIO21	SDA

Table4 :ESP32 connect with max30100

■ Schematic diagram between the ESP32 controller and MAX30100 sensor:

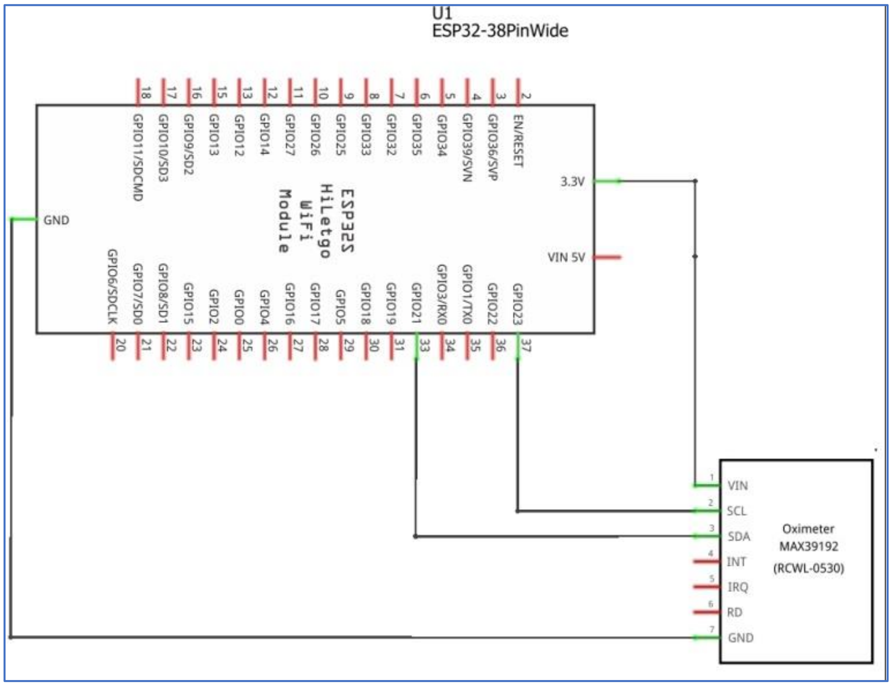

Figure 17:Schematic diagram between the ESP32 &MAX20100

■ ESP32 connect with RFID sensor :

ESP32 microcontroller	RFID
VCC	VCC
GND	GND
GPIO23	MOSI
GPIO19	MISO
GPIO5	REST
GPIO18	SCK
GPIO22	SDA

Table5 :ESP32 connect with RFID

Schematic diagram between the ESP32 controller and RFID sensor:

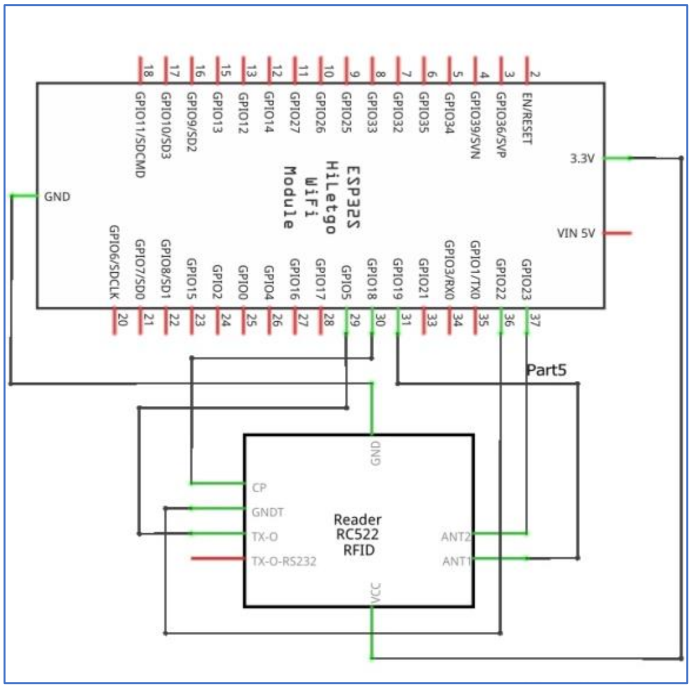

Figure 18:Schematic diagram between the ESP32 &RFID

Raspberry Pi connect with buzzer :

Raspberry Pi microcontroller	Buzzer
GPIO14	VCC
GND	GND

Table6 :   Raspberry Pi connect with buzzer

Schematic diagram between the Raspberry Pi controller and buzzer :

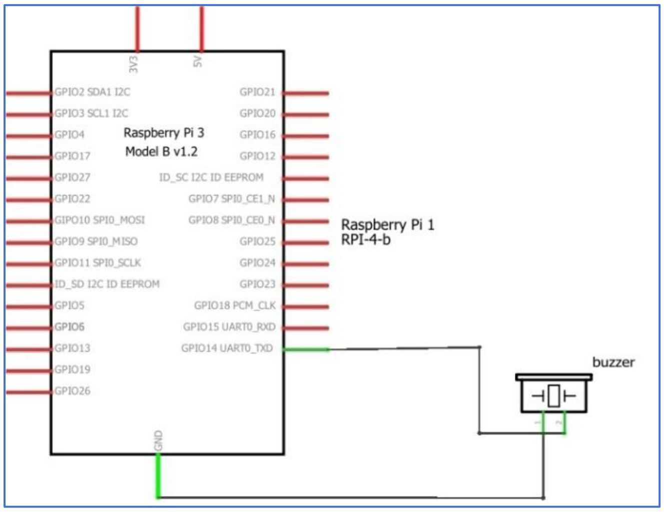

Figure 19:Schematic diagram between the Raspberry Pi & buzzer

CHAPTER FOUR
Software & Hardware Implementation

4.1 Preface

This chapter focuses on identifying the comprehensive fitness monitoring platform and evaluate the efficacy of implementing, as well as testing. In the following sub sections, we explain how to move from an initial system design to that of an operational prototype by describing the procedures, intended algorithms, and the protocols that facilitate the interconnection between devices. We then explain about the type of tests that were conducted to ensure that the system works correctly and specifically the correctness of exercises, the accuracy of the calculated health metrics, the reliability of data transmission, and the functionality and usability of the user interface.

4.2 Software Implementation

Is the process of turning a software design or plan into a functioning software product or system. It involves translating the specifications and requirements of a software project into a usable and executable software website .

WEBSITE

In this figure23 ,the login interface is the portion of the graphical user interface through which a user logs into the site possibly through use of user identification code and secret code. It is used as an anti-fraud tool that establishes the user's identity, and as a tool for limiting access to the website.

Here's a breakdown of the typical components and functionality you might find in a website login interface: Here's a breakdown of the typical components and functionality you might find in a website login interface:

1. **Username/Email Input:** This is where the user inputs the username or email address, they intend to use in the account creation process.
2. **Password Input:** Users made to type their password here. Passwords are generally masked due to obvious security steps taken while extracting data.
3. **Login Button:** Pressing this button sends the username/email and password for /authentication.
4. **Error Messages:** Visually shows messages to alert users about various problems involving their username and password, incorrect login name/password or account, and the like.
5. **Create Account/Signup Link:** Enable users create a new account if they have none since the specified account is already in use.

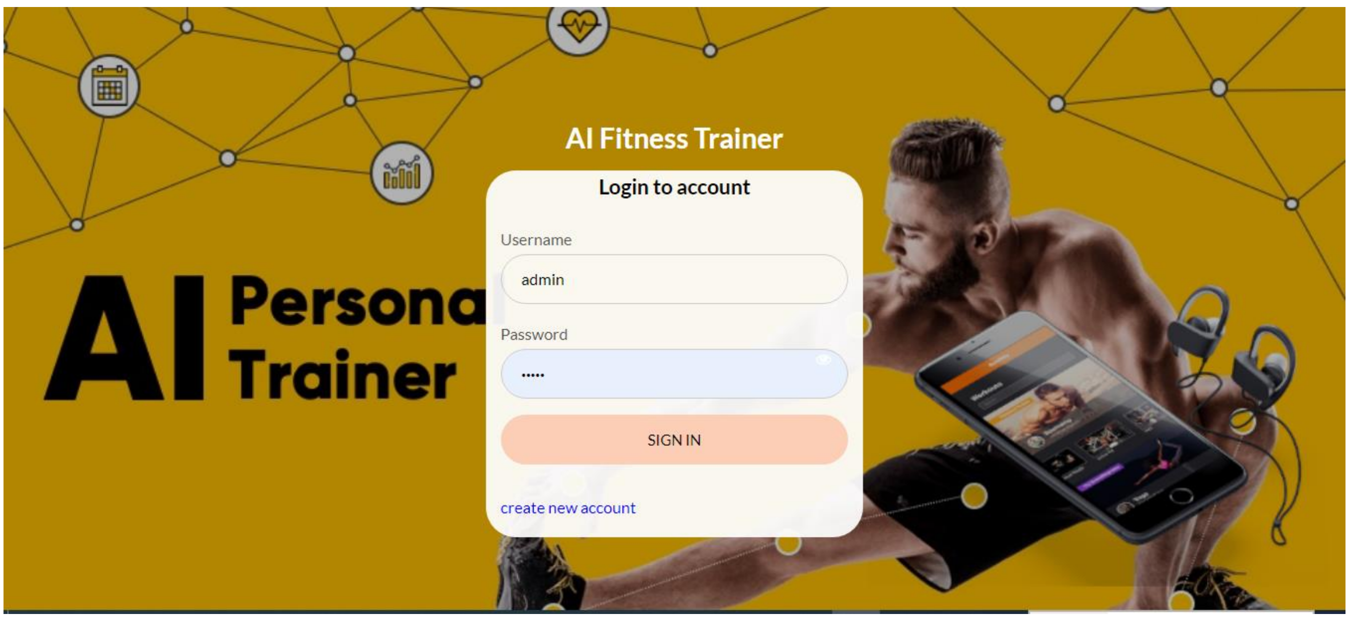

Figure 19:login page

Create Account

Here's a breakdown of the typical components and functionality you might find in a "Create Account" interface:

1. **Full Name Input:** Here, clients type in their full name into this field.
2. **Username Input:** Here, users type in the name they want to be identified by online. This is often validated for uniqueness, more so befitting its originality in a world that seeks to create distinct identities.
3. **Password Input:** This is a password entry where users input their preferred/proposed password. There could be policies that are put in place to ensure the higher degree of passwords safety.
4. **Repeat Password Input:** After one enters their password, they are prompted to confirm the entry in order to avoid inadvertent wrong input.
5. **Signup Button:** This submits the user's information in order to create the account, whenever this button is clicked.
6. **Error Messages:** Shows messages to let the users know about procedural problems concerning the inputs they have provided, for instance, invalid username or non-matching passwords.

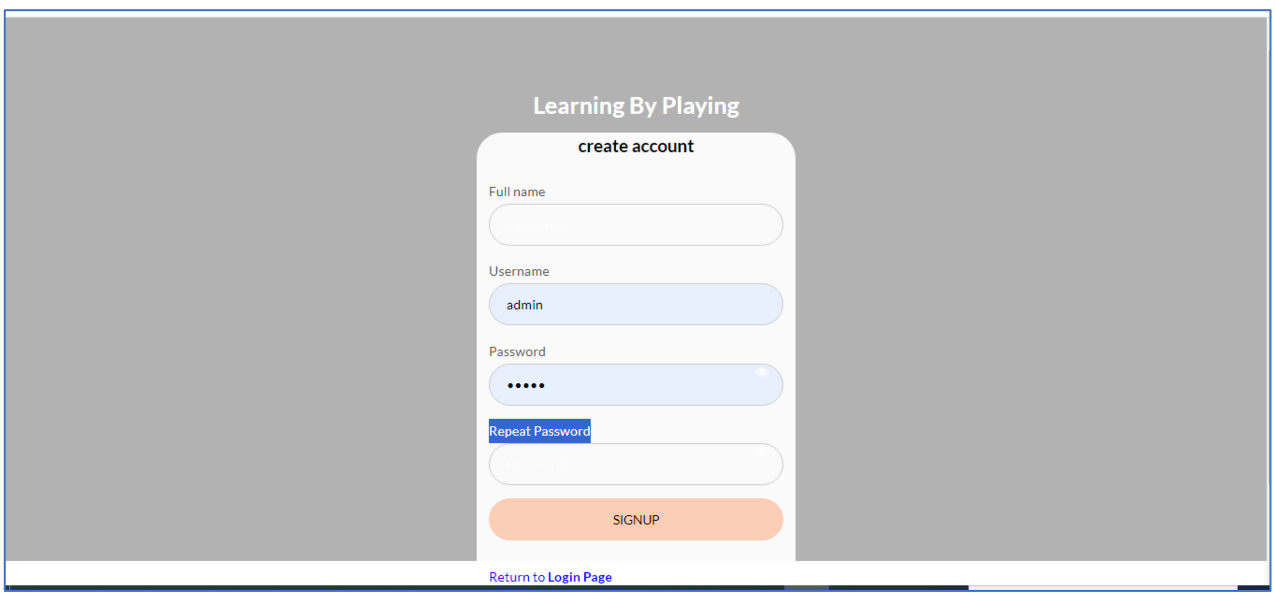

Figure 20:create account

Home page for admin

Here's a breakdown of the typical components and functionality you might find on the home page for an admin:

1. **Admins :** This enables the admin to set up a number of parameters concerning the admin profile including the password, the email id and other details.
2. **Students Accounts:** Given access for creation of student account, modifications of the account, or complete removal of the account as well as viewing of details on the account.
3. **Information:** a section on the HOMEPAGE for managers enables the admin to manage information regarding the AI training program. This section will help users get familiar with AI training through an introduction to a program present in this section.
4. **Exercise:** Offers the means for accessing and modifying exercises that exist in the database so, creating a new exercise, modifying specific exercise or completely deleting an exercise from the database.
5. **Logout:** This feature enables the admin to exit his account and, therefore, close his or her session at the Point of Sale.

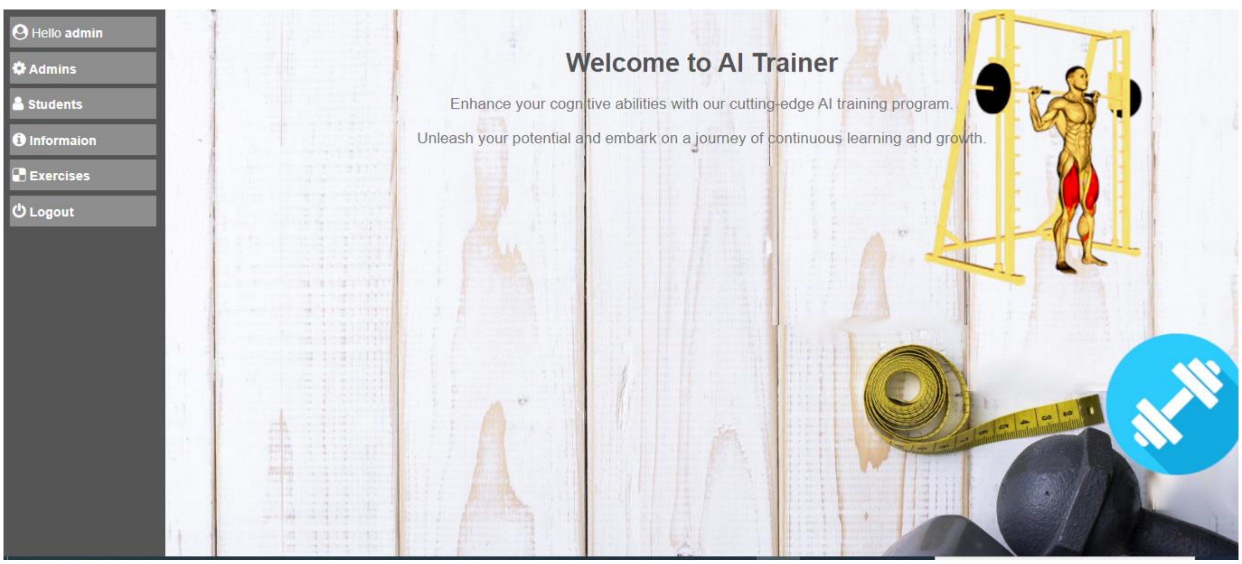

Figure 21: Home page for admin

Admins Settings page

The feature often referred to as the admin page gives the administrator capacity to manage account preferences including passwords, e-mail addresses, and profiles. Also, the one managing the administration also has a function in the system whereby he or she can put in a new account of an administrator or remove an existing one. This functionality provides flexibility and control over the administration.

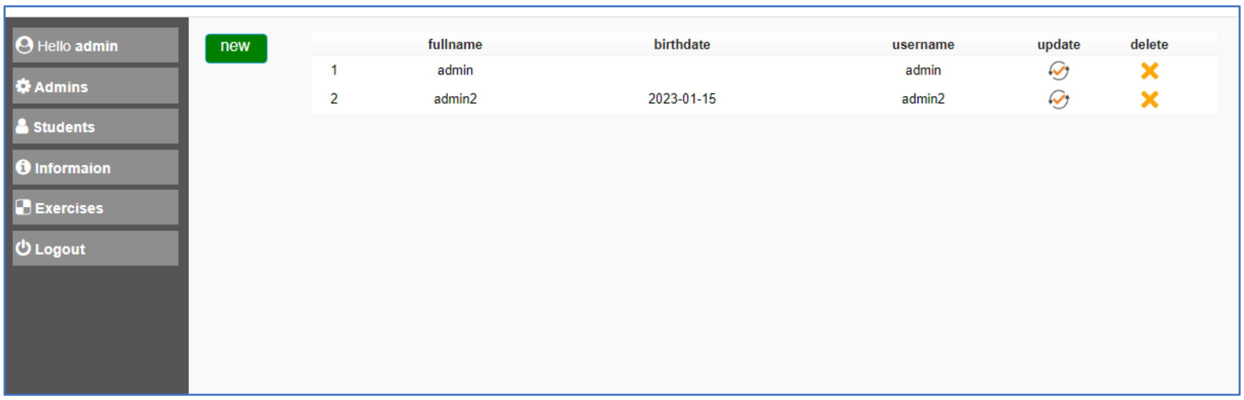

Figure 22: Admins Settings page

Students Accounts page

Provides access to manage student accounts, including creating, editing, or deleting accounts, as well as viewing account details.

Figure 23: Students Accounts page

Information page

The Information AI Training Program (Fitness): The information about the AI training program is provided in a special section on the admin home page, which gives an opportunity to manage it properly. Aimable, an admin, needs to be able to upload an image, put in a video, erase information that is already in the site, modify the content, and then write a small amount of text about the program telling people how it can benefit them. It is a brief description where the users can get all the information concerning the training in the program in the area of Artificial Intelligence.

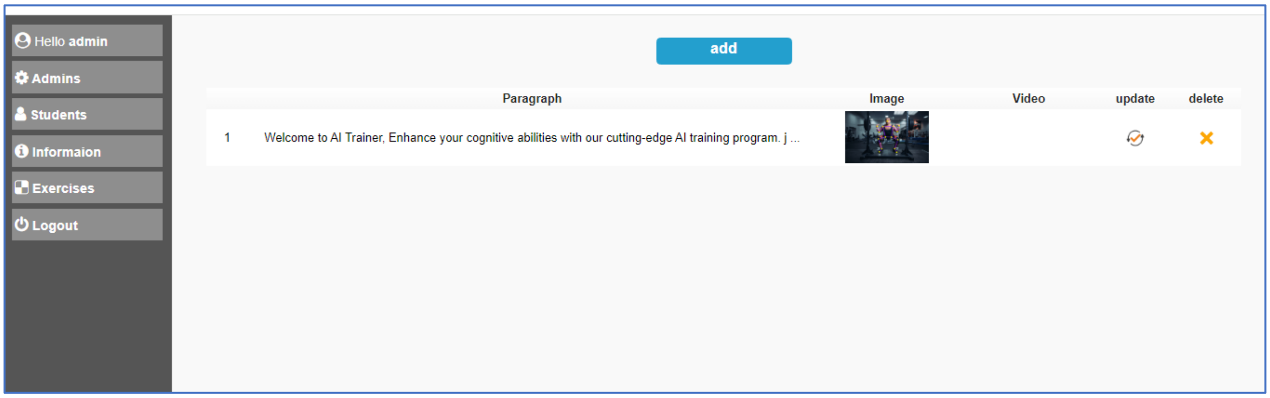

Figure 24: Information page

Exercise page

The exercise page is where administrators can manage exercises in the system. It includes the following features:

1. Exercise Name: It will allow showing the name of the exercise being performed, including its number and name.
2. Exercise Name ID: A new number corresponding to the exercise that allows for its identification.
3. Exercise Image: It enables the administrator to post an image for the exercise to support its identification.
4. Exercise Video: Offers the chance to have an image that links visitors to a video that shows how the exercise is done.

5. Repetition: Defines how many times the use of this exercise is going to be repeated.
6. Duration (Seconds): Sets the amount of seconds between each repetition of the next element of an array.
7. Set Group Day: Divides exercises into groups, or days depending on the workout schedule of a particular program.
8. Sort: It provides the possibility for administrators to organize exercises in the order they wish.
9. Update: Allows the administrator give more information to the exercise such as, name, picture or video and number of repeats, time etc.
10. Delete: Enables the administrator to delete a certain exercise within the system.

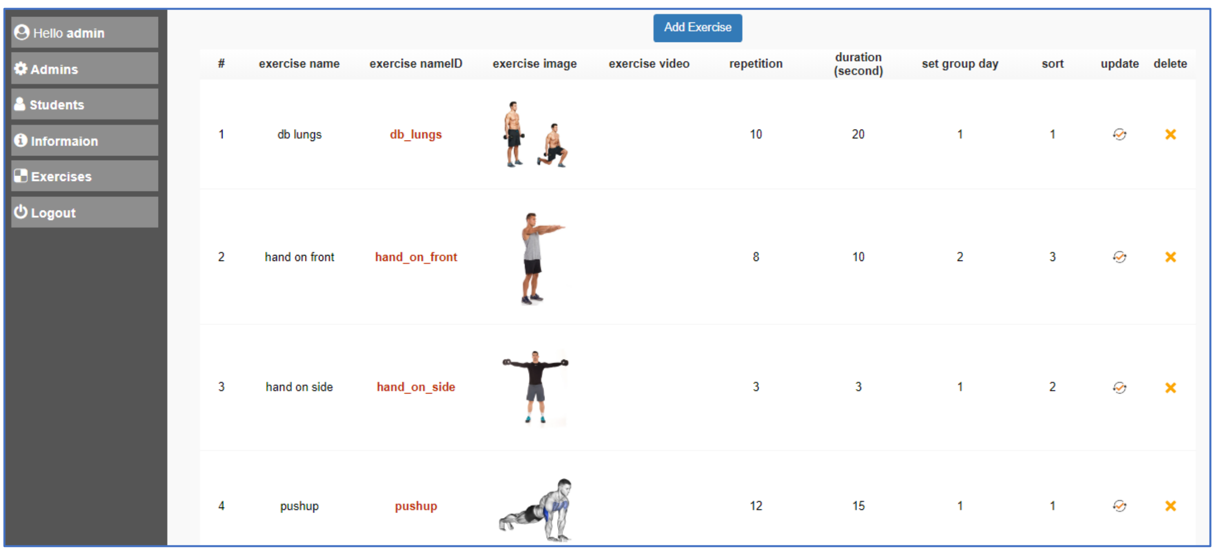

Figure 25: Exercise page

plan page

The plan page for the fitness monitoring platform organizes workout plans into specific days, such as "First Day" and "Second Day. " Each day's plan typically includes the following: The plan page for the fitness monitoring platform organizes workout plans into specific days, such as "First Day" and "Second Day. " Each day's plan typically includes the following:

Day Name: Includes specific terms for the day of the workout plan, for instance: "Day One" or "Day Two".

List of Exercises: Gives a list of exercises as practiced on that particular day and the repeat, sets and time take to undertake the exercise.

Exercise Details: Enables users to make general instructions, how the exercise is to be done, and the kind of equipment needed depending on the type of exercise.

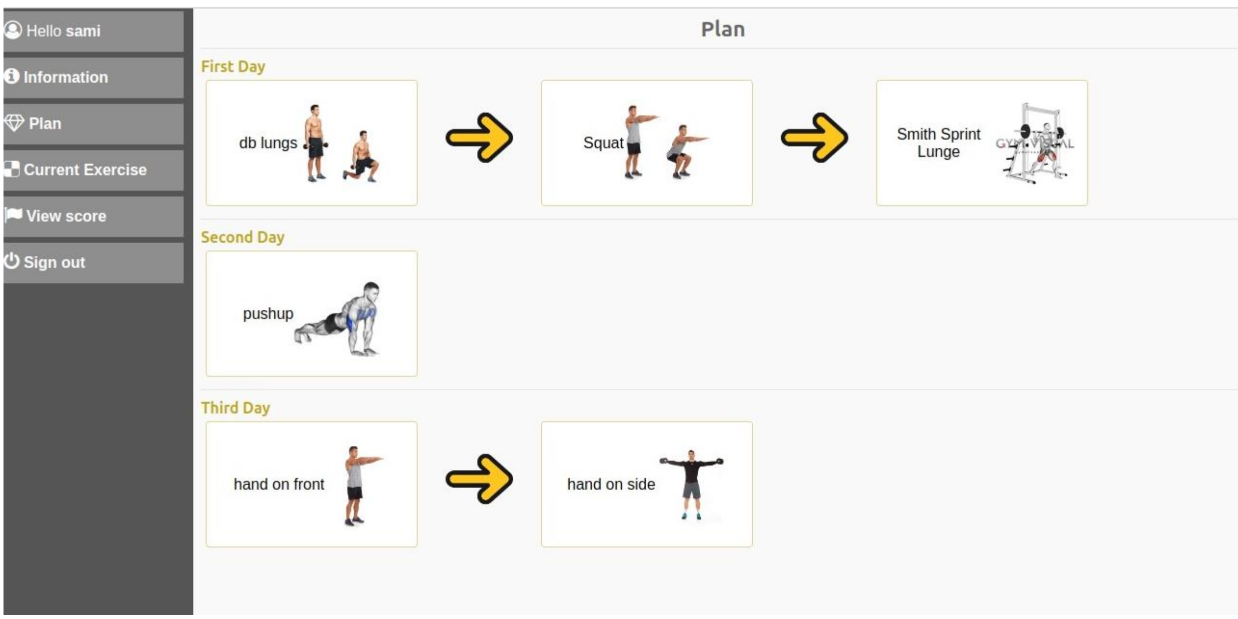

Figure 26: plan page

Current Exercise

The "Current Exercise" page presents data about the exercise being done at the moment. undefined

Exercise Name: It contains information about the name of the current exercise that is being performed.

Image: A sample of the exercise, which the user will use to guide him/her throughout the exercise.

Video: Optionally, a video that shows the correct form and technique when performing the particular exercise.

Repetition: The approximate number of repeats to be performed in the current exercise.

Duration (Second): The time it takes for each repetition in seconds.

Set Group Day: Indicates the group or day the exercise is taken in the workout plan.

Order: The position of the exercise in the workout plan of the current day.

Max30100 Value: The readings of the parameters measured by the MAX30100 sensor in the current state, including oxygen saturation and pulse rate.

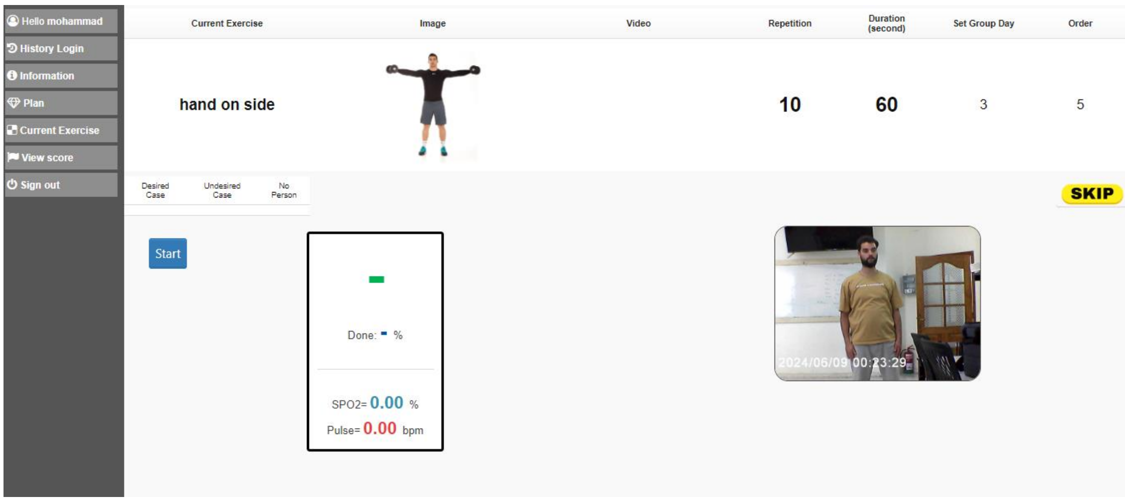

Figure 27: Current Exercise page

4.3 Hardware Implementation

When introducing the fitness monitoring platform's hardware implementation, this step primarily involved identifying the general hardware requirements as well as configuring the hardware to work with the system. Key steps and considerations included:

Camera System: Choosing the right camera system suitable enough to provide still shots in real time. The first consideration was to select the right type of camera that was able to provide the right resolution, frame rate and field of view so as to capture movement well.

Microcontrollers: Initializing microcontrollers for image processing and data transmission for the RPI controller as well as for the ESP32 controller, to monitor vital signs. This involved such tasks as configuration of GPIO pins, download of all essential libraries and integrating the microcontrollers to ensure that they could interconnect with other elements appropriately.

Sensors: Using the inclusion of such sensors like the MAX30100 sensor of tracking body vital such as oxygen levels and the pulse rate. This included wiring up of the sensors to the microcontrollers and making sure that they are properly interfaced and give the right readings while exercising.

Communication Modules: Connecting basic input/output ports like Wi-Fi/Bluetooth for data transfer between the developed device and the web platform. This included networks by setting up inside the ESP32 development boards, make connections, and gonna implement some protocol called MQTT to ensure the establishment of connection and in the same exercise, if making movement, it will use buzzer to warn.

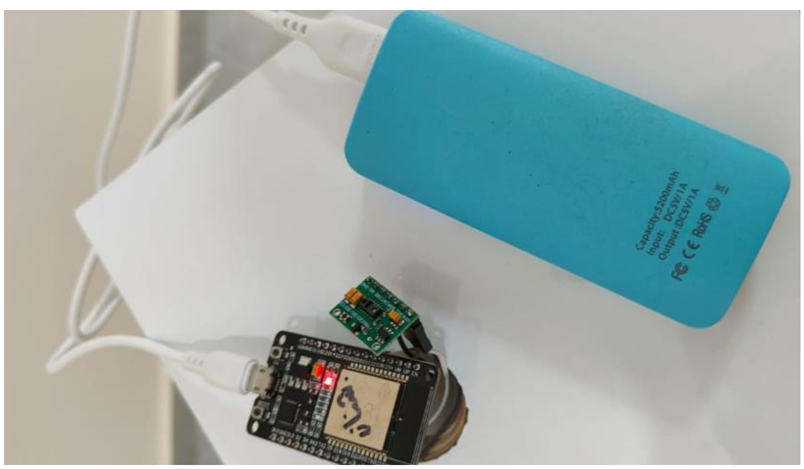

Figure 28: hardware (esp32 connect MAX30100)

Figure 29: hardware (RPI connect USB CAM)

The following figure represents how to connect the RFID sensor with the controller to pass each person's card to record their identification number when entering the exercise hall.

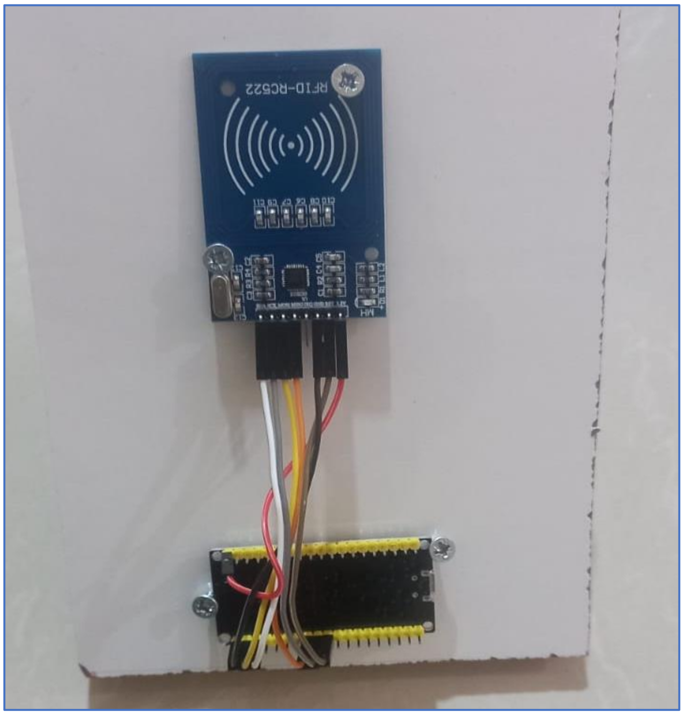

Figure 30: hardware (esp32 connect RFID)

4.4 Implementation challenges

During the implementation of the fitness monitoring platform, several challenges were encountered, including:

Hardware Compatibility: To maintain compatibility and future-proofing of a multitude of various hardware components including sensors, microcontrollers and communication modules, issues regarding incompatible specifications, data interfaces arose.

Algorithm Optimization: Real-time image processing as well as movement analysis necessitated the usage of efficient algorithms the development and optimisation of these proved cumbersome in that their effectiveness had to be tested rigorously on few supplied processing resources.

Data Transmission Reliability: The division that was most daunting when implementing the reliable communications between the wearable device and the browser-based web platform was the intermittent connectivity over wireless networks.

Testing Environment: Choosing the right testing environment to model actual exercising conditions and substantiate the perform- ance of a system when operating in real situations and circumstances was relatively easy, however, reconstructing actual exercising environment was not easy.

4.5 Validation and testing

Validation and testing of the fitness monitoring platform were conducted to ensure its reliability, accuracy, and effectiveness. This involved various tests, including:

1. Functional Testing: The exercise correctness detection, health metric variations, data transfer, and usability of all system components features were confirmed.
2. Performance Testing: Monitoring the functionality of the system and how it responds to different loads and conditions in terms of, respones time, robustness and speed of response.

3. Integration Testing: Exercising the hardware and software components of the total system to verify how the components worked jointly or whether when combined they passed the required standards.
4. Usability Testing: Taking a proactive approach to gain feedback from users and assess the actual usability, as well as possible overall satisfaction of users in regards to the system.
5. Security Testing: Evaluating the security features that have been installed on the system, the methods that are used to secure the data, the measures that are in place to secure the data from the vulnerabilities and the threats.

4.6 Summary

In Summary, it can be said that several issues concerning the integration of the hardware components, the optimization of the algorithm used, the data transmission, the power supply and consumption, the interface design and the tests all in all arise when implementing the fitness monitoring platform. Various validation and tests were conducted to check the reliability, accuracy as well functionality of the platform for monitoring and residing exercising performance and overall health stats. The last two steps consist of the actual placement of the application into the live environment, and periodically checking and making changes to the application for its continued effective running and high user satisfaction.

CHAPTER FIVE
Validation & Testing

5.1 Preface

In this chapter, the process followed in order to verify and evaluate the fitness monitoring platform developed in the previous chapters is described. Validation is the act of ensuring that the system to be developed is capable of satisfying the needs of the users that it is designed to serve. Testing is primarily about assuring that the implemented system operates flawlessly, is robust, and efficient. Here we will briefly outline the various procedures applied to the process of validation and testing in order to confirm that will work properly and is of high quality.

5.2 Validation

Finally, verification is the process of making sure that the fitness monitoring platform developed fits the user needs and requirements. This also involves ensuring that the process of exercising is being recorded properly by the system and this gives real time feedback on the records of health that are being kept. Key aspects of validation include:

1. Requirement verification: Affirm that the system has been implemented to the requirements and that the functionality achieved is satisfactory.
2. User acceptance: To assure the user that the system has been developed to meet his or her requirements, one has to get them to use the system in order to test it and give their opinion on it.
3. Performance evaluation: Evaluate the performance of the system in the aspects such as the rate at which it provides its services as well as it stability.

4. Compatibility testing: Ensure that the system can connect and operate in compatibility with other service applications or accessories it has to interface with.

Exercise Verification

The project contains body exercise movements which are; db lunges, Bicep, Hand on front, Hand on side, Hand on up, Pushing ups, Squats. These sets of exercises were provided in form of codes and were generated by the aid of the MediaPipe Algorithm with the assistance of the Numpy and CV2. These codes employ equations that determine the range of the exercise movements by considering the joint angles for that exercise. Here is an example of the "hand_on_sides" exercise and below this is how the exercising method was created:

```
import cv2
import mediapipe as mp
import numpy as np

mp_drawing = mp.solutions.drawing_utils
mp_pose = mp.solutions.pose

# Define function to check if in desired position
def detect_desired_position(left_shoulder, left_elbow, right_shoulder, right_elbow, left_wrist,
right_wrist):

    if abs(left_shoulder[1] - left_elbow[1])<=0.01 and abs(left_shoulder[1] - left_wrist[1])<=0.02 and \
       abs(right_shoulder[1] - right_elbow[1])<=0.01 and abs(right_elbow[1] - right_wrist[1])<=0.02 and
abs(left_wrist[0]-right_wrist[0])>=0.6:
        return "100%"
    elif abs(left_shoulder[1] - left_elbow[1])<=0.03 and abs(left_shoulder[1] - left_wrist[1])<=0.04 and
\
       abs(right_shoulder[1] - right_elbow[1])<=0.03 and abs(right_elbow[1] - right_wrist[1])<=0.04 and
abs(left_wrist[0]-right_wrist[0])>=0.6:
        return "80%"
    elif abs(left_shoulder[1] - left_elbow[1])<=0.05 and abs(left_shoulder[1] - left_wrist[1])<=0.06 and
\
       abs(right_shoulder[1] - right_elbow[1])<=0.05 and abs(right_elbow[1] - right_wrist[1])<=0.06 and
abs(left_wrist[0]-right_wrist[0])>=0.6:
        return "60%"
    else:
        return 0

def hand_on_sides(image_path):
  # Load image
  image = cv2.imread(image_path)
  image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)

  # Configure pose detection
  with mp_pose.Pose(static_image_mode=True) as pose:
```

```
      # Process the image
      results = pose.process(image)

  # Check if the desired position is detected
  if results.pose_landmarks is not None:
      landmarks = results.pose_landmarks.landmark
      kp = mp_pose.PoseLandmark

      left_shoulder = [landmarks[kp.LEFT_SHOULDER].x, landmarks[kp.LEFT_SHOULDER].y]
      left_elbow = [landmarks[kp.LEFT_ELBOW].x, landmarks[kp.LEFT_ELBOW].y]
      left_wrist = [landmarks[kp.LEFT_WRIST].x, landmarks[kp.LEFT_WRIST].y]
      right_shoulder = [landmarks[kp.RIGHT_SHOULDER].x, landmarks[kp.RIGHT_SHOULDER].y]
      right_elbow = [landmarks[kp.RIGHT_ELBOW].x, landmarks[kp.RIGHT_ELBOW].y]
      right_wrist = [landmarks[kp.RIGHT_WRIST].x, landmarks[kp.RIGHT_WRIST].y]

      # Check if in desired position
      in_desired_position = detect_desired_position(left_shoulder, left_elbow, right_shoulder,
right_elbow, left_wrist, right_wrist)

      if in_desired_position:

          print("===> The person is in the desired position."+str(in_desired_position))
          return True
      else:
          print("===> The person is not in the desired position.")
          return False


      # Draw pose on the image
      mp_drawing.draw_landmarks(image, results.pose_landmarks, mp_pose.POSE_CONNECTIONS,
                                mp_drawing.DrawingSpec(color=(0, 255, 0), thickness=2, circle_radius=2),
                                mp_drawing.DrawingSpec(color=(0, 255, 255), thickness=2,
circle_radius=2))

      print("No person detected in the image.")
      return "no-person"
```

This code is specifically meant to identify a particular manner of positioning of the body in an image, through the use of computer vision and pose estimation. Here’s a simple breakdown of what the code does:

1. Libraries and Setup: The code has implemented OpenCV library (`cv2`) for image processing and MediaPipe (`mp`) for the pose detector. It loads the drawing utilities and also begins by loading the pose detection model from the MediaPipe.

2. Function Definitions:

- **`detect_desired_position()`:** In this function, the coordinates for specific/key body landmarks (shoulders, elbows, and wrists) are passed into the function, to verify they are at specific positions that define the so called 'ideal pose'. The positions are assessed according to experts' assessment on the relative vertical distance (Y) and/or lateral distance (X) from these landmarks.
- **`hand_on_sides()`:** This function reads an image, adjusts the colors to the proper format and can use MediaPipe's pose model to detect persons landmarks. After that, it utilizes the landmarks of the shoulders, elbows, and wrists to establish whether or not the person is in the desired posture by invoking `detect_desired_position()`. Finally, it plays the message to let the user know whether the person is in the right position or not.

3. Processing and Output:

- In the current implementation pose detection is performed on a static image.
- If the landmarks are detected, it analyzes if the pose of the person is correct according to the key points. Depending on the adequacy that the person's pose corresponds to the criteria, the output displays a message stating the quality match (100 %, 80%, 60%, or not in position).
- If an empty picture is scanned and no person is observed, then the warning "No person detected in the image" is shown.

5.3 Software Testing(website)

Testing of the website element of the fitness monitoring platform requires various tests to determine whether the website is functional, easily navigable, and well-performing. This includes:

Ensuring that all elements of the website are functional and functional, including the tracking of exercises, recording of health-related features, the ability to register the users, and login and logout functions.

Evaluating the usability and course details specifically, given to the consumer of the website. This includes ease of navigation, clear instructions that have been uploaded, and the general experience that users get when interacting with the website.

Benchmarking the website and its performance considering various scenarios, including testing with high volumes of traffic and on different browsers. This is where the load times, responsiveness and overloading of server are put to test.

Valuable tasks such as the site displaying properly on various web browsers like Chrome, Firefox and on devices like Computer, Laptop, Tablet and Mobile. This means that there is testing for layout and look and feels across different platforms.

Auditing the security parameters that would be followed by the website to ensure confidential information of users are safe from fraudsters. This is specifically in areas of probe and assess for such conditions as SQL injection, cross-site scripting, and data leak.

Cross-browser Testing: To check if the work done is cross-browser compatible, the website has to be checked on, Chrome, Firefox, Safari, Edge and Internet Explorer.

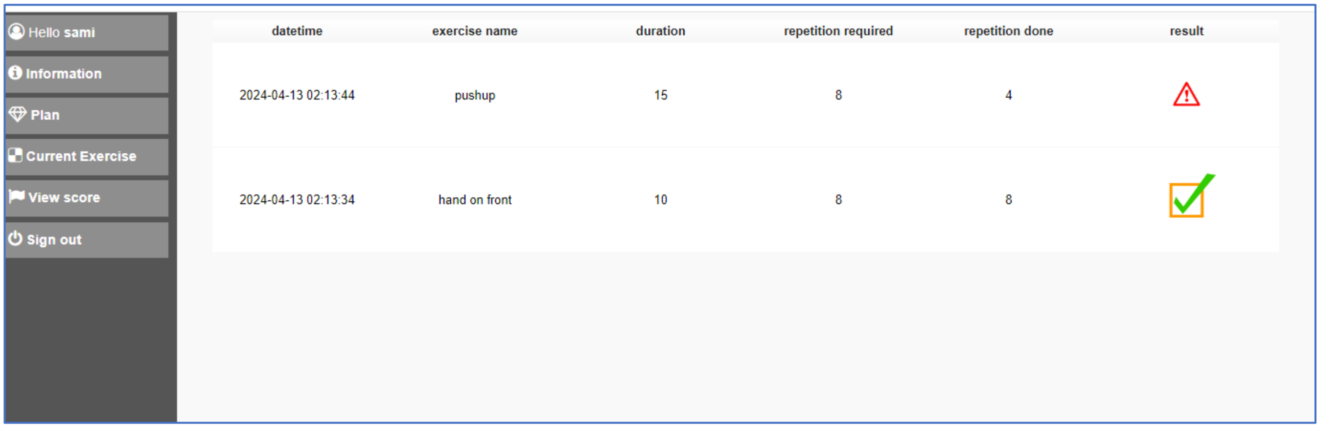

Figure 25: software testing for website

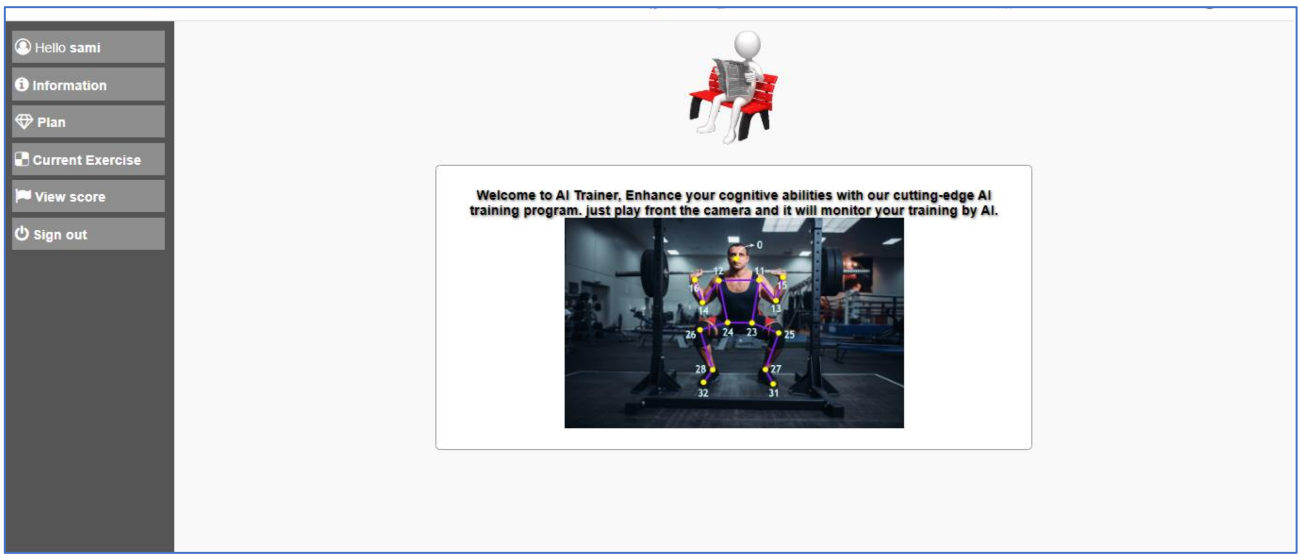

Figure 26: software testing

Figure 33: Cross-Browser Testing

This figure37 is a records webpage that is pulled from a website using a system that picks information from the Firebase online database including the person’s name, card number, and a log of the date and time.

Ensuring that the entries such as names, card numbers, and timestamps are logged properly also ensure that the information is easily accessible in a structural manner and well arranged on the website to enable users have the latest information.

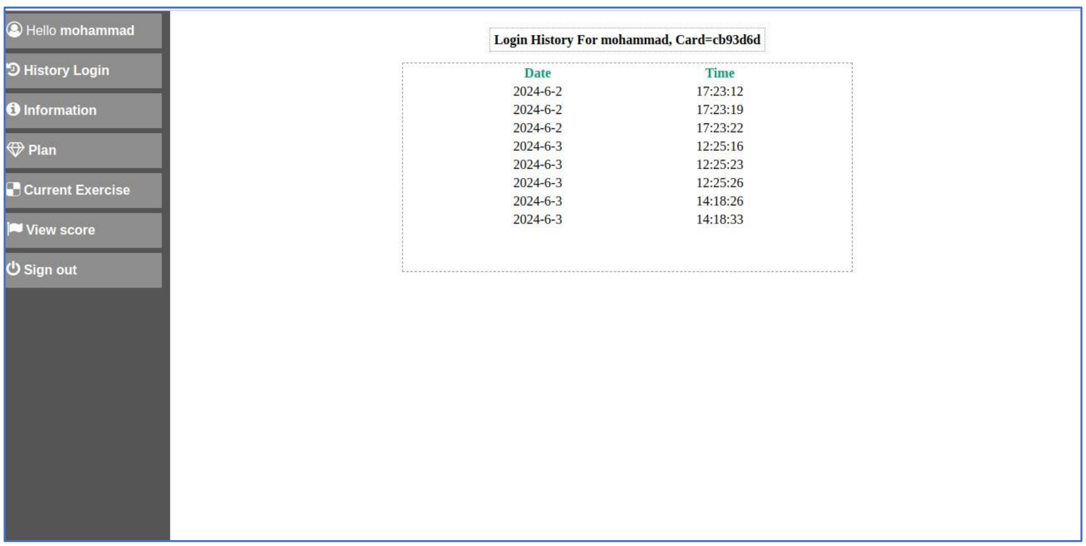

Figure 34: firebase history

5.4 Hardware Testing

Pre-acceptance testing involves the testing of all the physical hardware to guarantee that the system incorporates the right hardware and that it is in proper working condition. The following tests can be conducted:

1. **Sensor testing:** Make sure that some of the vital sensors like the MAX30100 sensors used to check the oxygen saturation levels, the heart rate and other health related features.
2. **RPI testing:** Shake the USB camera and Raspberry Pi a bit to check that they perform the expected action without fail.
3. **Connectivity testing:** Check the proper connectivity between the microcontroller and other peripherals to make sure that the loop is running coherently and the devices are communicating well.
4. **Integration testing:** Verify the compatibility of the hardware parts and the software environment where the application is to be implemented to avoid compromising the efficiency of the system.
5. **Buzzer Testing**: check the buzzer reaction as a way of ascertaining whether sounds signal to be used in facilitating correct exercise movements are produced immediately.

The validation and testing phase is the last step, and during this phase, the user must report all the problems that are observed and narrate the errors as well as bugs that are existent in the system and then have to debug the system and rectify the faults. Such as :

Lately, we have had some issues with acquiring a raw stable sampling rate from the MAX30100 sensor. Even though the trials were repeated with assurance, and it was set and reset with hand bearings, the readings remained off. We even attempted fixing the sensor to the project model for the longest time but the issue was not resolved. In the process it dawned on us that the sensor is faulty and is giving feedbacks that are different from what we want.

Figure 35: Test the buzzer & Raspberry Pi

Figure 36: hardware Testing

Project structure for Smart Personalized Fitness Coach.

Figure 37: Project structure

CHAPTER SIX

Conclusion & Future work

6.1 Preface

summarizing the steps of creating the fitness monitoring platform in this chapter, we want to highlight the primary achievements and positive experiences that we have acquired during the process. The last outcome and conclusion section will summarise the overall benefits and functionality of the system, also a general evaluation of the degree to which goals set in this project were achieved. Moreover, it will be examined what further developments could enhance the system more and/or improve the user system even more.

6.2 Final Result

The development and implementation of the fitness monitoring platform have yielded several significant results:

Real-Time Feedback: It effectively offers an instant reference of the form-input throughout the workout, enabling prompt correction of the exercise mannerisms.

Health Metrics Monitoring: Combination of the MAX30100 sensor and a controller in the form of the ESP32 ensures captured important health indicators including oximetry and pulse rate during workouts.

Movement Analysis: The utilization of the MediaPipe, Numpy, and CV2 algorithms enables the system to successfully analyse body movements and detect exercise accuracy.

Data Transmission and Storage: The analyzed information is sent through MQTT to a web site and preserved in the MySQL database to let users see the results after previous time interval.

User Interface: It offers a web-based application interface which enables users to monitor feedback on exercises, their health, and performance progress to boost user experiences and interaction.

buzzer: that it is feasible for a client to give a warning when carrying out movements that is in the same exercise.

6.3 Futures Works

While the current system provides a robust foundation for fitness monitoring, several areas for future improvement and expansion have been identified:

1. Engineering better AI Models with higher capability in estimating and identifying the user's movements more effectively and in a more sophisticated manner.
2. Incorporating other health parameters beside the ones reviewed above with the help of such extra sensors as EMG sensors to capture muscle activity, or other sensors for measuring the level of hydration.
3. To develop functionality that would be integrated into a separate mobile application allowing for more options and ease of use for the clients applying the web platform.
4. Including the recommendations of effective workout regimes that are based on the data provided by the user and the selected goals to achieve.
5. Incorporating mechanisms that contribute to social support like sharing the progress with friends or joining social competitions to enhance motivation amongst users.

6.4 Conclusion

The elaboration process of the fitness monitoring platform has been challenging but fruitful, and has resulted in the construction of a system that allows real-time metrics specific to the exercise being performed, along with overall health status tracking. From this project, one is able to appreciate how the application of image processing, machine learning, Internet of Things sensors has been incorporated to improve the fitness services. It is not only a solution to the current problem of people looking for an efficient way to get the guidance to work out, but it also has a solid ground in case of continuous improvement and enhancement.

This project proves how effective a collaboration between separate specializations such as software development, hardware incorporation, and user interface design is necessary and contributes to the overall efficient production of a final product. Incorporating practical issues in monitoring fitness and offering concrete solutions, the discussed platform testifies to the fact that technology is capable of positing changes for the better when the focus is shifted to enhancing the quality of people's lives.

To sum up, the outline of the fitness monitoring platform presents the further evolution of fitness technology products. The ability of technology to enhance exercise can be used to turn fitness goals into something that could easily transform the experience of exercising for the better. Though the accomplishments of this project, the foundation is paved for future development, thereby continuing to drive the goal of better health and improved quality of life with the use of technology.

References

[1] Pulse Oximeter and Heart-Rate Sensor IC for Wearable Health. Retrieved July 2019, from https://www.analog.com/media/en/technical-documentation/data-sheets/max30100.pdf

[2] Wikipedia contributors. (2021, December 20). Raspberry Pi. Wikipedia. Retrieved October 2021, from https://www.raspberrypi.com/documentation/

[3] Real Time Operating System (RTOS) . Retrieved October 2023, from https://www.geeksforgeeks.org/real-time-operating-system-rtos/

[4] González-Alonso, J.; Oviedo-Pastor, D.; Aguado, H.J.; Díaz-Pernas, F.J.; González-Ortega, D.; Martínez Zarzuela, M. Custom IMU-Based Wearable System for Robust 2.4 GHz Wireless Human Body Parts Orientation Tracking and 3D Movement Visualization on an Avatar. *Sensors* 2021, *21*, 6642.https://doi.org/10.3390/s21196642

[5] Hannan, A.; Shafiq, M.Z.; Hussain, F.; Pires, I.M. A Portable Smart Fitness Suite for Real-Time Exercise Monitoring and Posture Correction. *Sensors* 2021, *21*, 6692. https://doi.org/10.3390/s21196692

[6] Mallick, S; Tim, M; AI Fitness Trainer - Build Using MediaPipe for Squat Analysis. *Sensors* 2022, *20*, 6748. https://learnopencv.com/ai-fitness-trainer-using-mediapipe/

[7] ESP32 Microcontroller. Retrieved July 2020, from https://www.amazon.com/HiLetgo-ESP-WROOM-32-Development-Microcontroller-Integrated/dp/B0718T232Z

[8] Raspberry Pi 4 4GB. (n.d.). CanaKit. Retrieved October 2021, from https://www.canakit.com/raspberry-pi-4-4gb.html

[9] MAX30100 Pulse Oximeter Heart Rate Sensor Module. Retrieved May 11, 2021, from https://free-electronic.com/product/max30100-pulse-oximeter-heart-rate-sensor-module/

[10] USB Camera Module with Sony IMX179 Sensor, 8 Megapixels: Amazon. (n.d.). Amazon. Retrieved May 2024, from https://www.amazon.com/Magepixel-USB-Camera-3264X2448-Industrial/dp/B07R598TRZ.

[11] Lavalier microphone. (2013, September 13). In Wikipedia. https://fifinemicrophone.com

[12] Speaker for Raspberry Pi Pico. Retrieved March 2020, from https://botland.store/raspberry-pi-pico

[13] " Display screen Backpack. Retrieved July 2013, from https://www.philips.com.ph

[14] Download and install Arduino IDE – Arduino Help Center. Retrieved May 2024, from https://support.arduino.cc/hc/en-us/articles/360019833020-Download-and-install-Arduino-IDE.

[15] DuracellC opperTop battery - 12 x 9V - alkaline. Retrieved February 2007, from https://www.cdw.com/product/duracell-coppertop-alkaline-9v-battery-12-pack/5499085

[16] Amazon.com: Raspberry Pi 4 Model B Official PSU. (n.d.). Amazon. Retrieved October 2021, from. https://www.amazon.com/Raspberry-Model

[17] Raspbian. (n.d.). Rasbian. Retrieved October 8, 2021, from https://www.raspbian.org/

[18] Mediapipe. (n.d.). Retrieved 11, 2021, from https://mediapipe.dev/

[19] Fritzing. (n.d.). Retrieved May 2024, from http://www.fritzing.org/

[20] Apache Friends. (n.d.). XAMPP Installers and Downloads for Apache Friends. Retrieved May 2024, from https://www.apachefriends.org/index.html.

[21] MQTT - The Standard for IoT Messaging. (n.d.). MQTT.org. Retrieved May 2024, from https://mqtt.org/.

[22] Buzzer Basics. (n.d.). Buzzer Working Principles. Retrieved March 15, 2022, from https://www.buzzerbasics.com/

[23] RFID Journal. (n.d.). RFID Technology Overview. Retrieved April 22, 2022, from https://www.rfidjournal.com/

[24] Firebase. (n.d.). Firebase Documentation. Retrieved May 5, 2022, from https://firebase.google.com/docs

[25] Figure 1: Media Pipe's32 pose, from https://www.researchgate.ne