Project Report On

MotionX Interactive Fitness System

Submitted in partial fulfillment of the requirements for the award of the degree of

Bachelor of Technology

in

Computer Science and Engineering

By

Danish H Muhammed(U2103073)

Harinarayanan AM(U2103101)

Kannan S(U2103124)

Namdid Nishad Chittothayil(U2103144)

Under the guidance of

Ms. Mehbooba P Shareef

Computer Science and Engineering

Rajagiri School of Engineering & Technology (Autonomous)

(Parent University: APJ Abdul Kalam Technological University)

Rajagiri Valley, Kakkanad, Kochi, 682039

November 2024

CERTIFICATE

*This is to certify that the project report titled **"MotionX Interactive Fitness System"** is a bonafide record of the work done by **Danish H Muhammed(U2103073), Harinarayanan AM (U2103101), Kannan S(U2103124) and Namdid Nishad Chittothayil(U2103144)**, submitted to Rajagiri School of Engineering & Technology (RSET) (Autonomous) in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology (B.Tech.) in Computer Science and Engineernig during the academic year 2021-2025.*

Ms. Mehbooba P Shareef
Project Guide
Assistant Professor
Dept. of CSE
RSET

Ms. Anu Maria Joykutty
Project Coordinator
Assistant Professor
Dept. of CSE
RSET

Dr. Preetha K G
Professor & HOD
Dept. of CSE
RSET

ACKNOWLEDGMENT

We wish to express our sincere gratitude towards **Rev. Dr. Jaison Paul Mulerikkal CMI**, Principal of RSET, and **Dr Preetha K G**, Professor, Head of the Department of Computer Science and Engineering for providing us with the opportunity to undertake our project, **MotionX Interactive Fitness System** .

We are highly indebted to our project coordinator, **Ms. Anu Maria Joykutty,** Assistant Professor, Department of Computer Science and Engineering and Co-coordinator **Dr. Sminu Izudheen** and **Dr. Jisha G,** Professors,Department of Computer Science and Engineering for their valuable support.

It is indeed our pleasure and a moment of satisfaction for us to express our sincere gratitude to our project guide **Ms. Mehbooba P Shareef**, Assistant Professor, Department of Computer Science and Engineering, for her patience and all the priceless advice and wisdom she has shared with us.

Last but not the least, we would like to express our sincere gratitude towards all other teachers and friends for their continuous support and constructive ideas.

Danish H Muhammed

Harinarayanan AM

Kannan S

Namdid Nishad Chittothayil

Abstract

This project describes a method of improving exercise indoors by converting a regular stationary exercise bicycle into an interactive gaming controller. A reed switch sensor was used to detect bike speed, and an MPU-6050 was used to detect bike motion. The Arduino system captures data such as pedaling speed and handle movement, which is then sent to a PC in real-time, and mapped to a joystick using FreePIE and vJoy. The end product allows the user to play games, while being engaged physically, therefore removing some of the sedation issues from sedentary behavior and motivational issues associated with traditional exercise. Subject feedback and testing indicated improved continuity of exercise and overall satisfaction, which shows promise for an immersive fitness device for a healthier lifestyle.

Contents

List of Abbreviations

RNN - Recurrent Neural Network

GTA V - Grand Theft Auto V

CET - Cognitive Enhancement Technologies

Kinect - A motion-sensing input device by Microsoft

IMSB - Data set used in context with RNN

WISDM - Data set used in context with RNN

ERICA - Data set used in context with RNN

PhyDSLK - Framework for exergame development

DTW - Dynamic Time Warping

FreePIE - Free Programable Input Emulator

Vjoy - Virtual Joystick

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

In recent years, the fitness industry has witnessed a surge in integrating technology with traditional exercise routines. The global rise in home workouts, particularly after the COVID-19 pandemic, has fueled demand for interactive and engaging fitness solutions. Stationary exercise cycles are widely used for cardiovascular training, yet they often lack immersive elements that sustain long-term user motivation.At the same time, the gaming industry has evolved to offer highly interactive experiences, but traditional video games rarely incorporate real-world physical activity. To bridge this gap, the MotionX Interactive Fitness System project transforms a standard stationary cycle into an interactive game controller. By leveraging an Arduino Uno, a reed switch and magnets for acceleration detection and an MPU-6050 sensor for steering, this system translates real-world cycling motions into game controls.

Unlike conventional exergaming solutions, which rely on proprietary hardware, MotionX Interactive Fitness System utilizes FreePIE and vJoy for input mapping, allowing the cycle to function as a virtual joystick. This ensures broad compatibility with a variety of PC and console games, enabling users to experience physically engaging gameplay across different genres. By merging fitness with gaming, MotionX Interactive Fitness System enhances workout engagement, making exercise more immersive, enjoyable, and motivating for users seeking variety in their fitness routines.

1.2 Problem Definition

Traditional stationary cycles do not have the interactive and engaging features needed to keep users motivated throughout the cycling period. On the other hand, video games are engaging but do not effectively incorporate real-world physical activities. The MotionX

Interactive Fitness System project solves this by developing a system that merges physical cycling with an interactive game environment in order to enhance the exercise experience of the user through gamification.

1.3 Scope and Motivation

Scope:

The *MotionX Interactive Fitness System* project aims to transform a stationary exercise cycle into an interactive gaming controller by integrating hardware components such as a reed switch and magnets for speed detection, an MPU-6050 sensor for steering and a brake switch for braking actions. The real-time cycling data is processed by an Arduino Uno and transmitted to a PC gaming platform using FreePIE and vJoy, effectively mapping the cycle's movements to joystick inputs for seamless game control. Unlike conventional exergaming setups that rely on proprietary hardware, *MotionX Interactive Fitness System* ensures compatibility with a wide range of PC and console games, making it a versatile solution for fitness-based gaming. By incorporating gamification elements, this project enhances user engagement and motivation, offering an immersive exercise experience suitable for home fitness enthusiasts, gyms, and rehabilitation centers looking to innovate their workout routines with interactive entertainment.

Motivation:

The motivation behind the *MotionX Interactive Fitness System* project comes from the present trend of merging technology with fitness to make exercise more engaging and effective. Traditional stationary cycling often becomes monotonous, and users' commitment gradually decreases over a period of time. This project tries to take cycling and merge it with interactive game elements to tackle user disengagement and make the exercise process an enjoyable and rewarding one. On the other hand, the gamification of fitness solutions is a growing trend that shows potential in both physical and mental benefits, and this project aims to harness that. Its goal is the creation of a system that will keep users motivated not only to commit to regular exercise but also to have fun with an entertaining experience that captures a wide user base.

1.4 Objectives

The main objectives of the *MotionX Interactive Fitness System* project are:

1. Design and develop the hardware interface of the stationary cycle with an integrated reed switch and magnet-based speed sensor and MPU-6050 for steering detection, ensuring accurate real-time data acquisition.

2. Implement an interactive gaming environment using FreePIE and vJoy to map cycling data as joystick inputs, enabling compatibility with a wide range of PC and console games.

3. Enhance user engagement and motivation by incorporating gamification elements, making the cycling experience more immersive and enjoyable.

4. Ensure seamless real-time data integration between the hardware sensors and the gaming platform to provide immediate feedback to the user.

5. Conduct extensive testing and evaluation of system performance, including user experience analysis and feedback collection, to refine and optimize the overall setup.

6. Explore potential future enhancements, such as multiplayer functionality, cloud-based progress tracking, and integration with popular fitness applications.

1.5 Challenges

Integration of traditional cycling equipment with interactive gaming technology creates the challenge of ensuring the smooth integration of data synchronization and responsiveness of the user interface. How to keep users engaged in this activity through a balance between physical exercise and an enjoyable game experience is a complex design and development challenge.

1.6 Assumptions

- Users will have access to a stationary cycle equipped with a reed switch and magnet-based speed sensor, MPU-6050 for steering detection,

- The system will operate in an environment with a stable power supply and reliable hardware, including Arduino Uno for data processing and a PC for game input mapping.
- Players will be able to interact with the gaming interface through FreePIE and vJoy, allowing the cycle to function as a joystick-compatible controller.
- The setup assumes users are motivated to engage in physical activity through an immersive, gamified experience.

1.7 Societal / Industrial Relevance

The project holds significant societal and industrial relevance as it promotes healthier lifestyles by encouraging physical activity through gamified experiences, catering to the growing interest in home fitness solutions. From an industrial perspective, it offers opportunities for innovation in the fitness equipment and gaming sectors, enabling the development of hybrid products that bridge exercise and entertainment.

1.8 Organization of the Report

This report is organized as follows:

Chapter 1 introduces the project, outlining its objectives, challenges, and the importance of integrating physical exercise with interactive gaming for enhanced user engagement.

Chapter 2 provides the background and literature review, examining existing fitness technologies, gamified solutions, and their potential for promoting physical activity.

Chapter 3 outlines the objectives and methodologies employed in the project, including the integration of speed and cadence sensors, gaming engines, and data synchronization systems.

Chapter 4 describes the implementation and testing process, detailing the development of the interactive game environment and the validation of its effectiveness in engaging users.

Chapter 5 presents the results and analysis, evaluating the system's performance, user satisfaction, and its impact on motivation for physical exercise.

Chapter 6 concludes the report, summarizes the findings, discussing societal and industrial relevance, and provides recommendations for future improvements and scalability.

Chapter 2

Literature Survey

2.1 Body-Worn Sensors for Recognizing Physical Sports Activities in Exergaming (2023) [1]

The envisioned system for physical sports activity recognition by using body-worn sensors and deep learning models is composed of several steps: data collection, pre-processing, feature extraction, feature optimization, and classification. The whole architecture is presented in Figure 2.1.

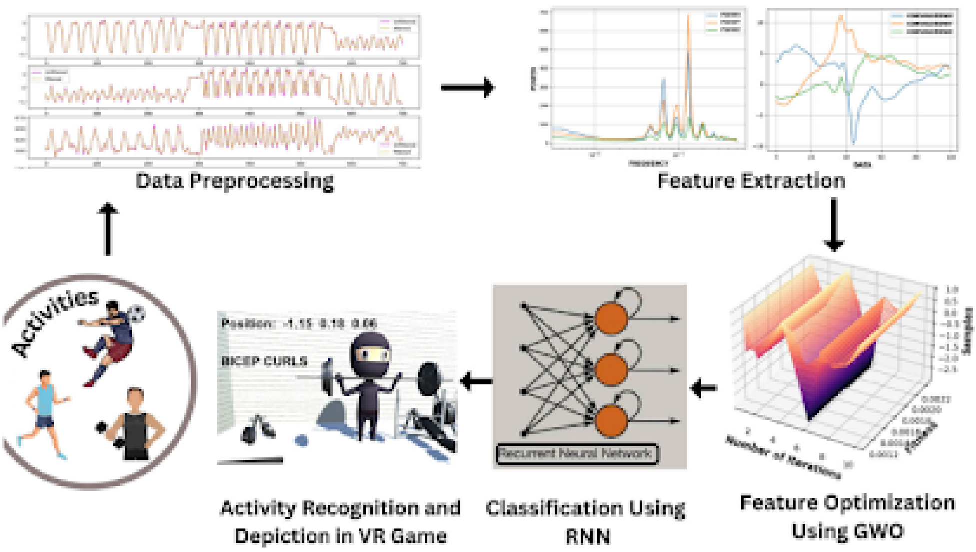


Figure 2.1: Overall Architecture of the Proposed System

2.1.1 Data Collection and Pre-processing

The system collects data using body-worn accelerometers and gyroscopes attached to specific body parts to capture real-time movement data. The pre-processing stage involves segmenting the data into frames and applying a third-order median filter to remove noise and preserve signal quality. The median filter is defined as:

$$\text{Med}(X) = \begin{cases} x_{\left[\frac{n}{2}\right]}, & \text{if } n \text{ is odd} \\ \frac{x\left[\frac{n}{2}-1\right]+x\left[\frac{n}{2}\right]}{2}, & \text{if } n \text{ is even} \end{cases} \tag{2.1}$$

This filter helps mitigate the impact of noise by replacing each data point with the median of neighboring values, effectively reducing outliers and enhancing data quality.

2.1.2 Feature Extraction

Feature extraction is performed using multiple methods to capture the unique characteristics of the motion data:

Power Spectral Density (PSD)

Power Spectral Density (PSD) quantifies the signal's power distribution over various frequencies. It is calculated as:

$$P(w_i) = \frac{1}{N}|X(w_i)|^2 \tag{2.2}$$

where $X(w_i)$ represents the Discrete Fourier Transform (DFT) of the signal at frequency w_i, and N is the total number of samples. PSD helps identify dominant frequency components in the data.

Skewness

Skewness measures the asymmetry of the data distribution. It is defined as:

$$\text{Skewness} = \frac{\sum_{i=1}^{N}(X_i - \bar{X})^3}{(N-1)\sigma^3} \tag{2.3}$$

where $\bar{X}$ is the mean of the dataset, σ is the standard deviation, and N is the number of samples. A positive skew indicates a distribution with a longer right tail, while a negative skew shows a longer left tail.

Kurtosis

Kurtosis evaluates the tailedness of the data distribution, indicating the presence of outliers. The kurtosis formula is:

$$\text{Kurtosis} = \frac{\sum_{i=1}^{N}(X_i - \bar{X})^4}{N\sigma^4} \tag{2.4}$$

High kurtosis indicates heavy tails (outliers), while low kurtosis points to lighter tails.

Convolutional Neural Network (CNN)

The system employs a one-dimensional Convolutional Neural Network (CNN) for extracting high-level features from the sensor data. The feature map generated at layer $l+1$ is given by:

$$a_j^{(l+1)}(\tau) = \sigma\left(b_j^l + \sum_{f=1}^{F^l} K_{jf}^l(\tau) * a_f^l(\tau)\right) \tag{2.5}$$

where: $a_j^{(l+1)}(\tau)$ is the feature map at layer $l+1$, σ is the activation function, K_{jf}^l is the kernel convolved over the input, b_j^l is the bias term.

2.1.3 Feature Optimization

The Grey Wolf Optimization (GWO) algorithm is used to optimize the selected features. GWO models the leadership hierarchy and hunting mechanism of grey wolves in the wild. The position update formula for the wolves is:

$$X(t+1) = X_p(t) + A \cdot D \tag{2.6}$$

where:

$$D = |C \cdot X_p(t) - X(t)| \tag{2.7}$$

The coefficients A and C are defined as:

$$A = 2a \cdot r_1 - a, \quad C = 2r_2 \tag{2.8}$$

Here, a decreases linearly over the iterations, and r_1, r_2 are random vectors within [0, 1].

2.1.4 Classification using Recurrent Neural Network (RNN)

The classification stage involves an RNN, which processes the optimized features. The RNN's forward pass equations are:

$$a(t) = b + Wh(t-1) + Ux(t) \tag{2.9}$$

$$h(t) = \tanh(a(t)) \tag{2.10}$$

$$o(t) = c + Vh(t) \tag{2.11}$$

$$\hat{y}(t) = \mathrm{softmax}(o(t)) \tag{2.12}$$

where: $a(t)$ is the input to the activation function at time t, $h(t)$ is the hidden state, W, U, V are weight matrices, $\hat{y}(t)$ is the output probability distribution.

The proposed system was evaluated on the IMSB, WISDM, and ERICA datasets, achieving classification accuracies of 85.01%, 88.46%, and 93.18% respectively. These results demonstrate that the proposed method outperforms traditional approaches in terms of accuracy and robustness.

2.2 Grand Theft Auto-Based Cycling Simulator for Cognitive Enhancement Technologies in Dangerous Traffic Situations [2]

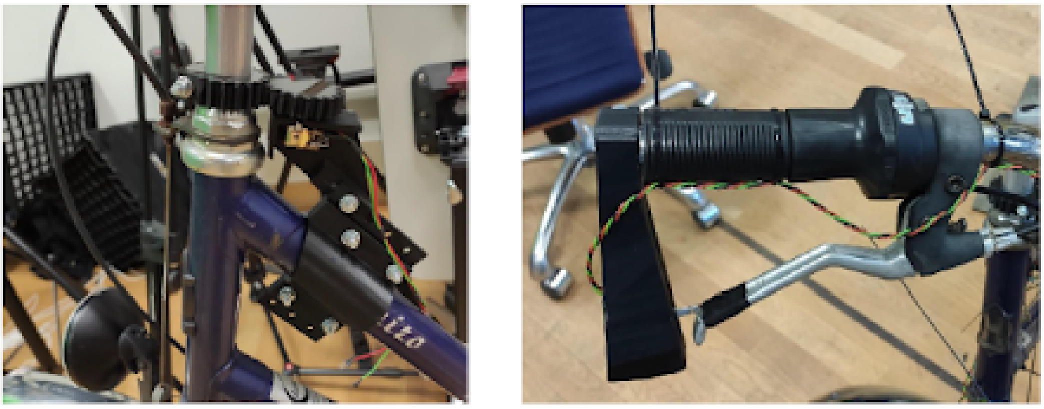
This paper presents a method for using the video game Grand Theft Auto V (GTA V) as a foundation for creating a cycling simulator aimed at testing cognitive enhancement technologies (CETs) in a safe and realistic man- ner. The simulator enables researchers to simulate dangerous traffic situations involving cyclists, cars, and trucks. It makes use of game modifications in the scenario for controlling and outputting data along with a physical bicycle setup, equipped with sensors, to offer real- nistic input. The paper analyzes this setup and makes proposals for improving the safety and cognitive awareness of cyclists.

2.2.1 Simulator Development and Justification

The authors have selected GTA V as the base of the simulator due to its high-quality graphics, realistic urban environments, traffic physics, and active modding community. All these features allow the authors to simulate complex hazardous traffic conditions without having to develop a new system from scratch. The use of a game engine such as GTA V gives the researcher several advantages in user experience research, such as repeatable environments, various types of vehicles, and simulated weather effects. The game's built-in Physics system controlled vehicle collisions, which are crucial to the overall simulation environment's realism and visual appeal. To add functionality, the team used Script Hook V, a software that enabled the ability to do custom scripting within GTA V, allowing them access to over 5000 in-game functions. This allowed the team to create specific traffic scenarios with cyclists, cars, and trucks, dangerous situations where trucks pass near cyclists. The modding options made for a controlled, consistent setting for each test, which, as mentioned, would be instrumental in CET testing in hazardous conditions.

2.2.2 Cyclist Input Device Design

To create an interactive experience, the authors transformed a stationary bicycle into a functional input device that could connect with the simulator. This setup required capturing real-world cycling inputs such as steering, speed, and braking, which were then translated into the game environment. They used an Arduino Pro Micro to interface three main sensors: potentiometers for steering and braking and a magnetic sensor to measure wheel speed. These components were mounted to the bicycle's handlebars, brakes, and wheels, creating a responsive game controller that allowed cyclists to interact with the simulation as if they were cycling on real roads. By using Arduino firmware that emulated an Xbox controller, the team ensured compatibility with the game's input system. This approach enabled realistic control over the simulated bike; however, GTA V's input configuration for bicycles was less precise than for cars, prompting the authors to develop a workaround to improve steering precision. The architecture of the input device is presented in Figure 2.2 .



(a) steering sensor

(b) braking sensor

Figure 2.2: Architecture of the input device.

2.2.3 Data Interface and Communication Setup

A key aspect of the simulation involved real-time tracking of trucks near the cyclist, essential for testing CET effectiveness. Due to limitations within GTA V's scripting environment, which restricts access to the operating system for external data exchange, the researchers created a unique solution using a barcode-based system.

The mod detected trucks within a certain radius of the player, collected data on their position, speed, and direction, and converted this information into barcodes displayed in a corner of the game screen. External software then captured and decoded these barcodes, enabling the CET systems to process the data and provide appropriate feedback to the cyclist. Although effective, this system faced a delay due to the game's frame rate, which allowed only one barcode transmission per frame. This limitation created a small lag in real-time data transfer, which the authors acknowledged as an area for improvement in future iterations. The communication setup is presented in the Figure 2.3 .

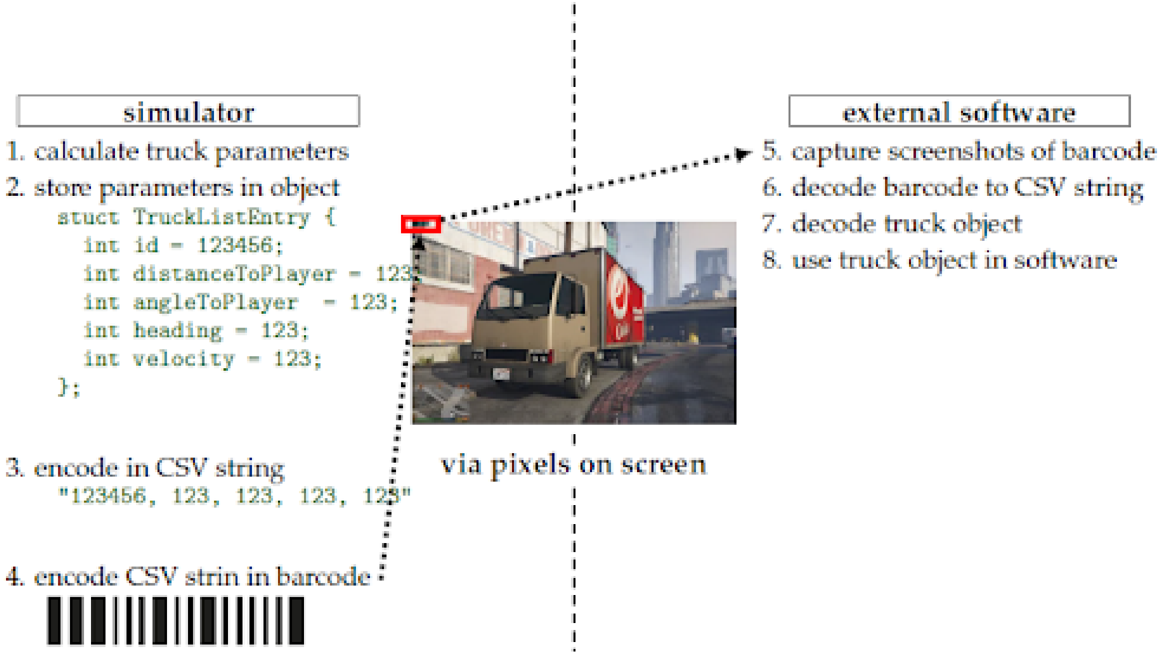


Figure 2.3: Truck parameters calculated by the mod script.

2.2.4 Cognitive Enhancement Technologies (CET) Testing

Two CET concepts were developed and evaluated in this study. The first was a tactile feedback system that provided real-time, vibration-based alerts on the handlebar. This system consisted of four vibration motors and a beeper, each motor programmed to activate based on truck proximity and direction. For example, if a truck approached from the left, the left handlebar would vibrate, gradually intensifying as the truck neared. At close distances, the beeper would sound an alarm to emphasize the immediate threat.

The second CET concept was a visual display, implemented on a smartphone attached to the bicycle, which showed block-shaped bars representing truck positions relative to the cyclist. The height of each block indicated the truck's distance from the cyclist, allowing the user to gauge proximity without having to rely on peripheral vision. This visual feedback aimed to improve situational awareness while maintaining focus on the road 2.4.

Figure 2.4: Block-shaped visualization of a truck from left behind from the subject view.

2.2.5 User Evaluation and Feedback

After testing, the simulator was evaluated through a user survey to assess its ease of use, realism, and overall effectiveness in providing a controlled environment for CET testing. The survey collected feedback from six participants, who noted that the system was intuitive to learn and use, though some issues with input precision and simulator realism were highlighted.

Participants generally found the tactile feedback more effective than the visual system, as it allowed them to keep their attention on the road while still receiving alerts. However, some noted that vibration feedback could be less effective on uneven surfaces like cobblestones, where natural vibrations could interfere with the tactile signals. Overall, users found the system helpful for testing CET concepts but recommended adjustments to improve input sensitivity, address minor lag issues in data transmission, and enhance the realism of the steering controls.[2]

2.3 PhyDSLK : a model-driven framework for generating exergames[3]

The PhyDSLK framework facilitates the development of Kinect-based exergames through a model-driven approach, enabling non-technical users to create custom games for rehabilitation. The framework simplifies the game design process by abstracting complex programming and hardware interactions into a high-level domain-specific language (DSL). The methodology is divided into three main stages: Model Specification, Code Generation, and Evaluation, as shown in Figure 2.5.

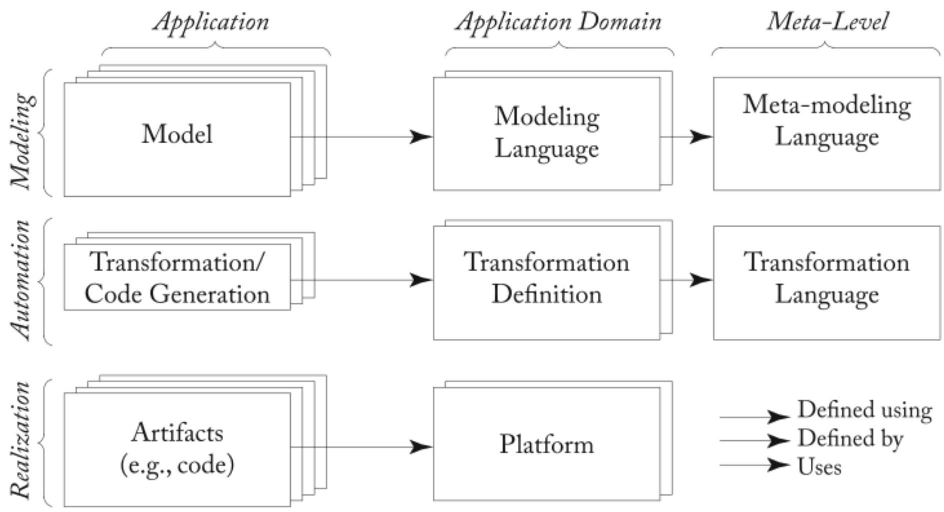


Figure 2.5: Architecture of PhyDSLK Framework for Exergame Development

2.3.1 Model Specification Phase

The Model Specification phase focuses on defining the exergame model using PhyDSLK's DSL within the Eclipse IDE. This phase enables designers to outline the gameplay, interaction models, and scoring rules without requiring programming skills.

- **Exergame Components and .phy File Definition**: Designers create a `.phy` file where all the major components of the game are defined: actors, environment layout, and interaction types. The DSL provides syntax auto-completion and error detection that helps get the syntax right, ensuring that the design process is effective.

- **Defining Exergame Types**: Designers can declare different types in the game, like actors (e.g., balloons, aliens) and actions (e.g., popping balloons) or mining asteroids). Each type is defined with particular attributes and behaviors. that are later translated into executable game actions.

- **Layout and Environment Customization**: This module allows for definition of the game space, background graphics, and physics parameters such as gravity. A game like, for instance, 'Keep the Balloons' may have lighter gravity settings because it lacks ground; but 'AlienMiner' utilizes default gravity to connect itself to the terrestrial environment.

- **Activity Definitions and Scoring Mechanisms**: In designing, game activities that the players will perform are defined, and the scoring criteria are used to establish if these activities have been successfully completed. Scoring often consists of real-world physical movements recognized by the Kinect, giving interactive feedback to the user and encouraging them to engage in physical activity.

- **Exporting the Model as XML**:When the game design is finalized, it will be exported into an XML file. This XML file would then become a blueprint destined for a certain step of code generation where every aspect of play, interaction, and design will be articulated in a functional and structured manner in codes.

2.3.2 Code Generation Phase

The Code Generation phase automates the transformation of the XML-based exergame model into Unity scripts. PhyDSLK's code generator uses Acceleo templates to produce Unity-compatible C# scripts, enabling seamless integration with Kinect hardware for interaction.

- **Main Script Generation for Game Logic**: This script is responsible for managing the core game loop, handling interactions defined in the model, and enforcing game rules. Actions and movements outlined in the `.phy` file are converted into Unity-compatible functions.

- **Kinect-Specific Interaction Scripts**: These scripts enable Kinect-based interactions by converting body movements into game controls. Kinect SDK functionalities, such as skeletal tracking and depth sensing, are used to map the user's movements (e.g., arm extensions, leg movements) to corresponding in-game actions.

- **Gesture Recognition and Control Mapping**: Gesture recognition algorithms process Kinect's skeletal tracking data to recognize predefined gestures (e.g., hand raise, jump). Detected gestures trigger game actions, which are mapped to the virtual elements specified in the exergame model. A sample gesture is defined as:

$$\text{Gesture Detected} = \begin{cases} 1, & \text{if } \|J_{\text{hand}} - J_{\text{shoulder}}\| < \epsilon \\ 0, & \text{otherwise} \end{cases} \tag{2.13}$$

  where J_{hand} and J_{shoulder} represent the 3D positions of the hand and shoulder joints, and ϵ is a predefined threshold for gesture recognition accuracy.

- **Integration with Unity Template Project**: The generated scripts are integrated into a pre-existing Unity template project, which includes pre-configured static libraries for efficient Kinect-based interaction. This template ensures compatibility across different game types and simplifies setup for non-technical users.

2.3.3 Evaluation and Results

To assess the effectiveness of the PhyDSLK framework, a usability study was conducted with 14 undergraduate students. Participants were tasked with developing three exergames: KeepTheBalloons, PopTheBalloons, and AlienMiner. Development effort, usability, and learning curve were evaluated using the System Usability Scale (SUS), which measures overall system usability on a 100-point scale.

System Usability Scale (SUS)

The SUS score was calculated as follows:

$$\text{SUS Score} = 2.5 \sum_{i=1}^{10} \begin{cases} \text{Score}_i - 1 & \text{if positive statement} \\ 5 - \text{Score}_i & \text{if negative statement} \end{cases} \tag{2.14}$$

where Score_i represents the participant's rating (1-5) for each question, adjusted based on the sentiment of the statement.

Key Findings

The evaluation results denote that PhyDSLK is highly usable, with a mean SUS score equal to 82.3, indicating "excellent" usability. Development times for the three games were comparable, suggesting that PhyDSLK facilitates efficient game design regardless of the difficulty of the game. Some of the most important findings were :

- **Efficiency**: The time contained average game designs to 30 minutes from the start of development.
- **Flexibility**:The SUS gave results toward having a low learning curve, enabling students to create working exergames with little or no training.
- **Versatality**: There were apparent performance drawbacks when PhyDSLK copes with types of games and interaction types, demonstrating the flexibility of the framework.

2.3.4 Limitations and Future Work

While PhyDSLK provides an effective model-driven framework for exergame development, several limitations were identified:

- **Limited Hardware Support**: Currently, the framework only supports Kinect-based devices. Extending compatibility to other sensors, such as IMUs or smartphones, could broaden application areas.

- **Limited Gesture Types**: The current gesture recognition is limited to a set of predefined gestures. Future iterations could leverage machine learning to enable customizable gestures.

- **Additional Activity Types**: Expanding the DSL to support a wider range of activities (e.g., cycling, squats) would enhance PhyDSLK's applicability in fitness and rehabilitation.

PhyDSLK is a user-friendly and accessible framework designed to facilitate the development of exergames for rehabilitation. By abstracting technical complexities and automating code generation, it empowers non-technical users to create immersive exergaming experiences that encourage physical activity. Future work will focus on expanding the DSL, supporting more sensors, and enabling adaptive game mechanics to improve engagement and usability in various application domains.[3]

2.4 A Kinect-Based Adaptive Exergame for Elderly Rehabilitation [4]

The Kinect-based adaptive exergame framework proposed by Mocanu et al. aims to increase physical activity among elderly users by providing a structured, adaptive game environment. Utilizing a trainer and user avatar, the system adjusts exercise difficulty in real-time, enhancing accessibility and promoting prolonged engagement in physical activity. The game's adaptive nature is achieved through Dynamic Time Warping (DTW) and cross-correlation to align movements, as shown in Figure 2.6.

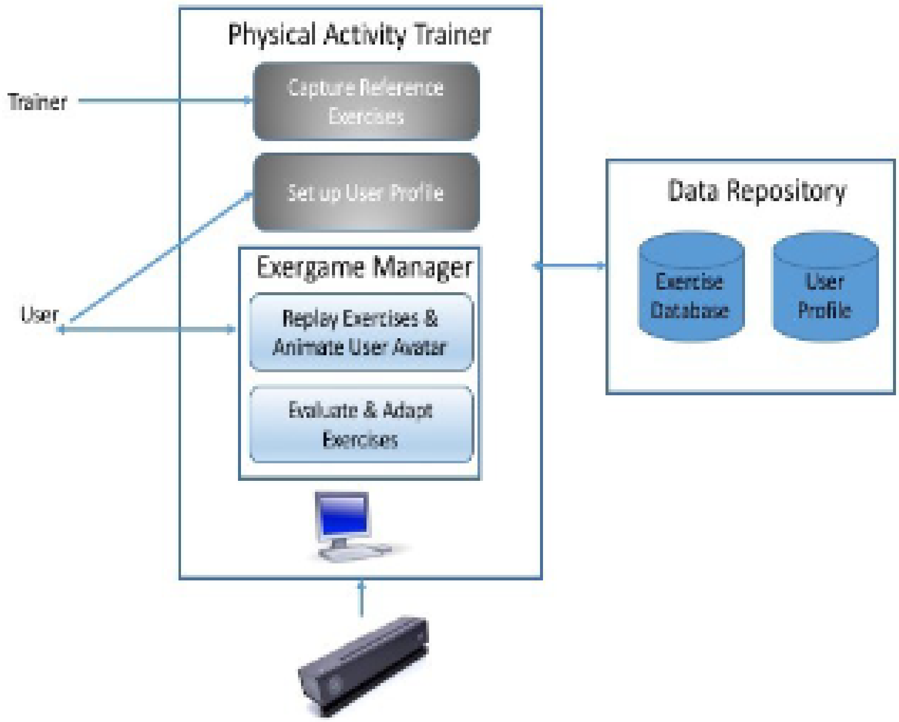


Figure 2.6: Architecture of the Adaptive Exergame Framework by Mocanu et al.

2.4.1 System Components and Architecture

The framework includes core components such as user and trainer avatars, exercise recording modules, and an adaptation mechanism to adjust exercise speed based on user movements. These components ensure the game adapts to the physical needs of elderly users, fostering a positive rehabilitation experience.

- **User and Trainer Avatars**: The game uses two 3D avatars. The trainer avatar demonstrates pre-recorded exercises, while the user avatar reflects real-time actions, allowing the user to mimic the trainer's movements accurately. This visual feedback encourages engagement and movement accuracy.

- **Exercise Recording and XML Storage**: Exercises designed by a kineto-therapist are captured using Kinect and saved in XML format, including metadata on joint relevance, exercise duration, difficulty, and motion trajectory. Each recorded exercise serves as a reference model for the user to follow.

- **Adaptive Speed Control**: The adaptation component dynamically modifies the trainer avatar's speed, aligning with the user's pace to maintain synchronization.

By monitoring lag between the trainer and user, the game can adjust difficulty to minimize frustration.

2.4.2 Adaptive Gameplay Mechanics

The system adapts gameplay mechanics to user performance in real-time, adjusting to physical capabilities and encouraging consistent participation.

- **Real-Time Movement Synchronization**: To achieve accurate alignment, DTW calculates similarity between the user and trainer's movements. Joint angle deviations are measured to determine accuracy, while cross-correlation identifies time lags. This approach adapts the trainer's speed to match user performance, helping maintain an achievable exercise pace.

- **Angle-Based Joint Evaluation and Feedback**: Joint angles are tracked and compared frame-by-frame to the trainer's model. Using color-coded indicators, the system provides instant feedback—green for correct alignment, yellow for moderate deviation, and red for significant misalignment. This feedback loop guides users to improve their posture during exercises.

- **Score-Driven Motivation**: There are points for accuracy in movement, which creates an element of competitiveness to maximize scores. The higher the deviation towards the trainer avatar, the better the score is - in this way, there's a clear picture set for individual users to improve from.

- **User Progress Tracking and Session Summaries**: After each session, the game provides a summary of user performance and tracks improvements over time. Progress is measured by the reduction in deviation scores, helping the user and the therapist track engagement levels and accuracy of exercises.

2.4.3 System Evaluation

Ten participants tested the different exercise speeds in a pilot study. The experiment investigated the flexibility of the framework and the effect of live feedback on dwell time.

- **User Engagement and Feedback**: Participants found the adaptive gameplay advantageous because the real-time speed adjustment avoided frustration caused by conventional, fixed-paced exergames. Adaptive feedback enabled them to exercise at a comfortable pace, thereby enhancing both enjoyment and adherence.

- **Performance Metrics**: The results showed an increase of 43 percentage in exercise time, which indicates that adaptive speed control encourages users to remain active for longer periods. High scores were associated with better posture accuracy, thus validating that real-time feedback improves movement accuracy and sustains activity.
- **Scalability and Usability**: The system is flexible and thus suitable for users who vary in degrees of physical ability, thus negating age-related inhibitions on mobility, flexibility, and reaction time. The design can be scaled up to accommodate more complex rehabilitation environments.

-

This Kinect-based adaptive exergame framework provides an effective solution for the elderly. rehabilitation, integrating real-time adaptation with quite engaging game mechanics. Future work will involve integration of different biometric sensors, for instance the Zephyr Bioharness, so that it can be further tailored based on physiological data such as heart rate and respiration. More im- Improvements include expanding exercise libraries and refining cross-correlation algorithms to improve responsiveness.[4]

- **Biometric Data Integration**: Incorporating physiological metrics will allow real-time adaptation based on health parameters, enabling a safer and more personalized exercise experience for elderly users.
- **Expanded Exercise Library**: Incorporating various exercise routines will offer a com- A comprehensive rehabilitation tool adaptable to a range of therapeutic needs, promoting broader engagement.
- **Algorithmic Optimization**: Improved cross-correlation techniques could enhance real-time adjustments, ensuring the system remains responsive to rapid user movement changes.

2.5 Validation of a Novel Bicycle Simulator with Realistic Lateral and Roll Motion[5]

2.5.1 Methodology:

The validation of a new bicycle simulator is studied, using a multi-degree-of-freedom platform to generate lateral, yaw, and roll motions for the simulation of more dynamic cycling experiences. The simulator relies on the Carvallo–Whipple model, which was known to accurately model bicycle dynamics. Dual linear motors are used in controlling yaw and lateral movements so that the bicycle may respond naturally to the rider's inputs. The sys- tem makes use of actual sensor feedback, steering angle, yaw rate and roll, which are processed through a dual P-controller to take into account real-time information from the rider so the model adjusts right after the corresponding rider inputs, with an aim at closely matching conditions during actual riding. Objective measures used include yaw rate and lateral acceleration during tests, while subjective measures involved ratings on realism, control, and comfort. ulator's performance. While minor phase delays were noted at higher speeds, participants rated the experience as closely resembling outdoor cycling, demonstrating the simulator's potential for training and research purposes.

2.5.2 Advantages:

It features very high realism. The lateral and roll motions can be highly realistic as well. They are necessary in order to reproduce the sensations felt during actual outdoor cycling. Thus, the concept is more natural and represents reality better. Use of dual P-controllers advanced with real-time sensor feedback enables interaction and response from the system, adjusting it dynamically based on the input by the rider to the simulator

It further includes improving control and overall feel. Validation includes the objective data from measurements of yaw rate and subjective feedback from participants. This dual approach in validating makes full assessment of the effectiveness of the system possible, making it suitable for training purposes and in research that calls for a highly realistic cycling simulator.

2.5.3 Disadvantages:

Despite its many strengths, the simulator still poses challenges, including phase delays at higher speeds that can make maneuvers feel less than immediate. Additionally, visual feedback is limited to prerecorded video rather than real-time 3D rendering, which might otherwise improve immersion. Finally, the complexity of integrating many sensors and controls produces a very high-cost setup with extensive main tenance requirements. These may limit accessibility to smaller research facilities or organizations with limited budgets, as tenance is now resource-intensive.

Chapter 3

Proposed System

3.1 System Architecture

The system architecture of an interactive fitness solution is represented by the diagram. Input devices include a Reed Switch for speed detection, an MPU-6050 for motion sensing. These inputs are processed by an Arduino Uno and transmitted to external systems via serial communication. The system integrates with a PC for game input mapping using FreePIE and vJoy, allowing the stationary cycle to function as a joystick-compatible controller. This design merges fitness tracking with gamification, enhancing user engagement through an immersive exercise experience. 3.1.

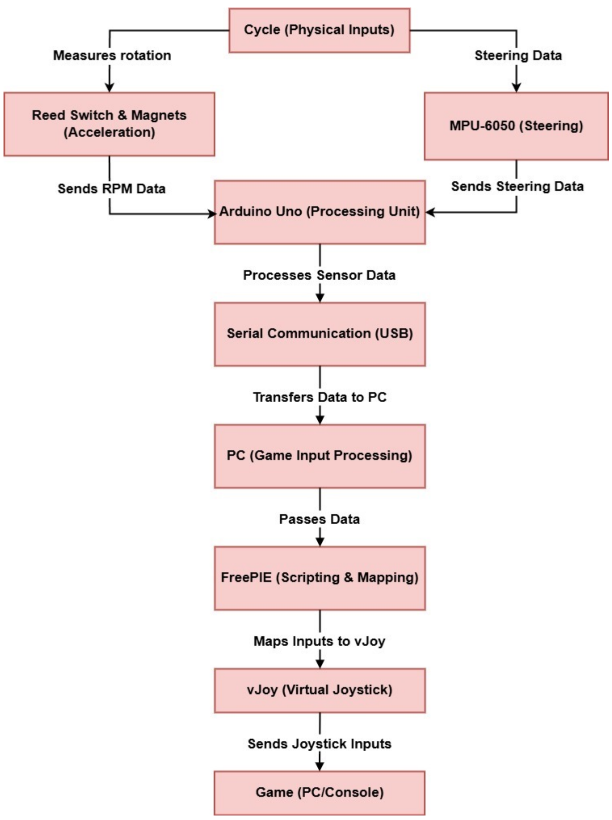


Figure 3.1: System Architecture

3.2 Component Design

3.2.1 Input Devices

- **Reed Switch:** Detects pedal rotations by sensing magnetic field interruptions caused by a magnet attached to the rotating part of the cycle. It sends a digital pulse to the Arduino for each rotation.

- **MPU-6050:** A gyroscope and accelerometer module that measures handlebar movements (tilt and rotation) to detect steering inputs.

3.2.2 Data Processing

- **Arduino Uno:**
  - Reads input from the reed switch and MPU-6050.
  - Calculates rotations per minute (RPM) from the reed switch to determine speed and distance.
  - Maps handlebar movement data from MPU-6050 to directional inputs.
  - Prepares data for transmission via serial communication.

3.2.3 Data Transmission

- **Serial Communication:**
  - Sends processed data, including RPM, directional inputs to the connected PC.
  - Ensures low-latency data exchange for real-time gaming input.

3.2.4 Game Integration

- **PC (Game Input Mapping):**
  - Receives input data via serial communication.
  - Uses FreePIE for mapping the received data to virtual joystick inputs.
  - Utilizes vJoy to emulate joystick controls for seamless game integration.
  - Converts cycling data into game actions such as acceleration and steering.

- Enhances the gaming experience by providing real-time feedback based on user activity.

3.3 Tools and Technologies

3.3.1 Hardware Requirements

- Stationary Cycle : Base structure and input source for the game controller
- Arduino Uno : Acts as the processing unit and interfaces with sensors.
- Reed Switch : Detects pedal speed (RPM) based on rotations
- MPU-6050 Gyroscope/Accelerometer : Captures handlebar movements for steering.
- Power Supply : Provides power to Arduino and connected components.
- Magnets : For the reed switch to detect pedal movement.
- Wires and Connectors : To connect sensors and modules.

3.3.2 Software Requirements

- Arduino IDE : For coding and uploading firmware to the Arduino Uno.
- FreePie : for recieving the datas from Arduino via serial ports and allocating it to the Vjoy.
- Vjoy : For mapping the serial datas from arduino via FreePie to the game controls
- Game Platform (PC) : To run compatible games.

3.4 Module Divisions and Work Breakdown

1. High-Level Modules

The project is divided into the following key modules:

1. Hardware Integration
2. Data Processing and Mapping

3. Game Integration

4. Testing and Debugging

2. Work Breakdown Structure

1. **Hardware Integration**

   (a) Set up the reed switch for pedal speed detection.

   (b) Integrate the MPU-6050 for handlebar movement detection.

   (c) Connect sensors to Arduino for data collection.

2. **Data Processing and Mapping**

   (a) Write Arduino code to process reed switch and MPU-6050 data.

   (b) Map RPM to game speed and handlebar tilt to steering to in-game actions.

   (c) Format data as HID-compliant inputs for gaming platforms.

3. **Game Integration**

   (a) Integrate inputs with existing games via Vjoy and FreePie.

4. **Testing and Debugging**

   (a) Test hardware components for accurate data capture.

   (b) Debug Arduino code for correct data mapping.

   (c) Verify seamless integration with games and applications.

3. Task Allocation

3.5 Key Deliverables

3.5.1 Working prototype

- A fully functional cycle-based game controller that integrates seamlessly with gaming platforms.

Table 3.1: Task allocation for each members.

Team Member	**Assigned Tasks**
Danish H. Muhammed	Hardware setup and sensor integration (reed switch and MPU-6050).
Namdid Nishad C	Arduino programming for data processing and mapping inputs to actions.
Kannan S	data integration from arduino to game controls via FreePie and serial communication.
Harinarayanan AM	Game integration using Vjoy and overall testing/debugging.

3.5.2 Sensor Data Processing

- Detect the number of oedal rotations
- Handlebar steering input from the MPU-6050 gyroscope/accelerometer.

3.5.3 Mapping Logic

- Converting sensor inputs into HID-compliant game controller signals for speed and steering using FreePIE and Vjoy.

3.5.4 Demonstration and Documentation

- Technical documentation of hardware and software components.

3.6 Project Timeline

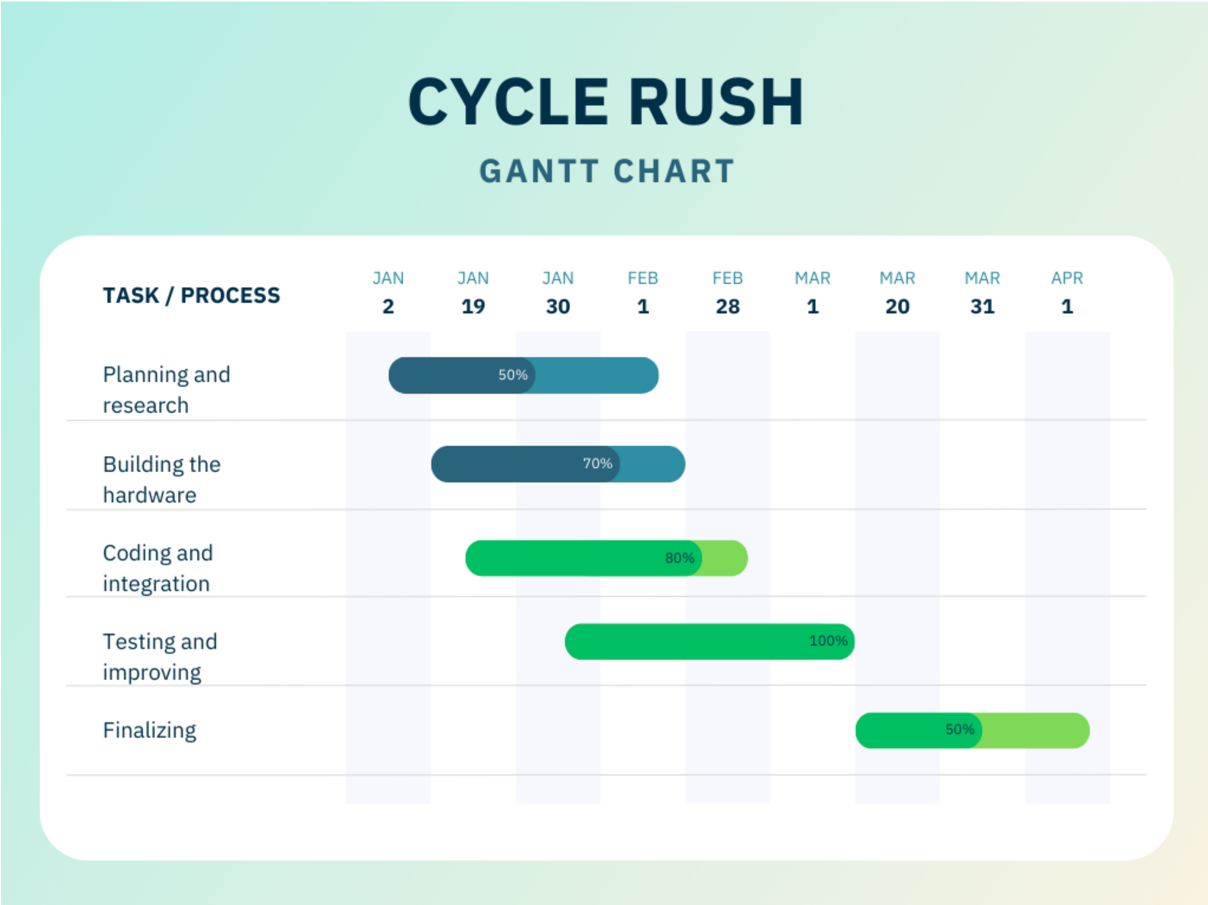


Figure 3.2: Project Timeline.

Chapter 4

Experimental Setup

The implementation of the **MotionX Interactive Fitness System** involves integrating hardware and software components to enable a stationary exercise cycle to function as an interactive game controller. This system is designed to work with any existing game by mapping physical cycling movements to in-game controls.

4.0.1 Hardware Implementation

The system utilizes a combination of sensors and a microcontroller for data acquisition and processing:

- **Reed Switch & Magnets** – Detects the pedaling motion and calculates speed (RPM).
- **MPU-6050 Gyroscope/Accelerometer** – Captures handlebar movements for steering input.
- **Arduino Uno** – Acts as the central processing unit, reading sensor data and transmitting it to a PC.

The **Arduino Uno** communicates the sensor data via **serial transmission**, where it is received by a PC for further processing and game integration.

4.0.2 Software Integration

To ensure compatibility with any game, the system utilizes:

- **FreePIE (Free Programmable Input Emulator)** – Translates sensor data into keyboard, mouse, or joystick inputs.

- **vJoy (Virtual Joystick Driver)** – Maps cycle inputs to virtual joystick controls, making it compatible with a wide range of games.

The system is tested by integrating it with various gaming environments, ensuring smooth and responsive input mapping.

4.0.3 Testing and Validation

The testing process focuses on three key aspects:

Hardware Accuracy Testing

- Validating RPM calculation from the reed switch.
- Ensuring accurate steering input detection from the MPU-6050.

Software Performance Evaluation

- Assessing input latency between the cycle and the game.
- Testing FreePIE and vJoy mappings across multiple game genres.
- Ensuring seamless data transmission from Arduino to the PC.

User Engagement Testing

- Conducting usability tests to evaluate player immersion.
- Gathering feedback from users regarding responsiveness and comfort.
- Analyzing whether the system enhances the gaming experience by making physical activity more engaging.

Chapter 5

Results and Discussions

The evaluation of the **MotionX Interactive Fitness System** is based on system performance, user satisfaction, and its impact on motivation for physical exercise. Various testing methods, including hardware performance assessment, software latency measurement, and user feedback, were employed to analyze its effectiveness.

Overall lookup of MotionX prototype show in figure 5.1

Figure 5.1: MotionX

Laptop for game output mounted on the cycle with a stand as shown in figure 5.2

Figure 5.2: Laptop

Arduino mounted on the cycle as shown in figure 5.4

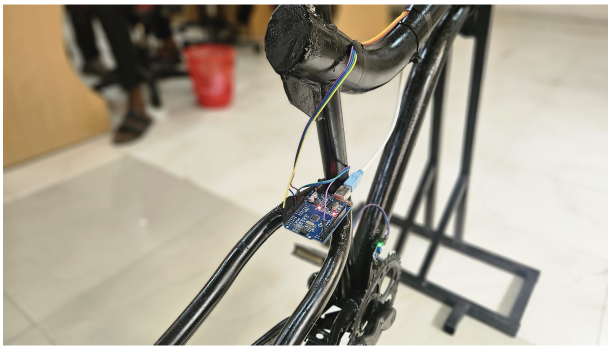


Figure 5.3: Arduino

Magnet attached to the pedal gear to get the rpm as shown in figure 5.4

Figure 5.4: Magnet

Reed switch placed close to the pedal gear in order to capture the magnet contact iterations as shown in figure 5.5

Figure 5.5: Reed Switch

MPU6050 mounted on the handlebar inorder to get the horizontal movements as shown in figure 5.6

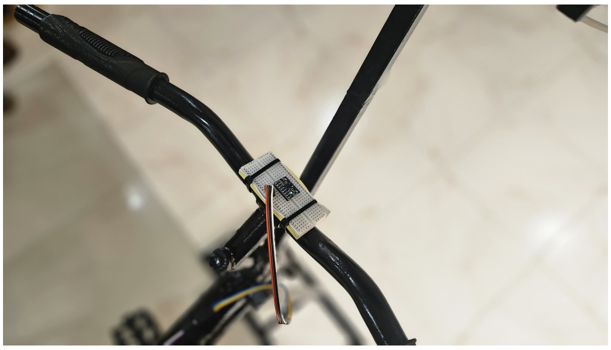


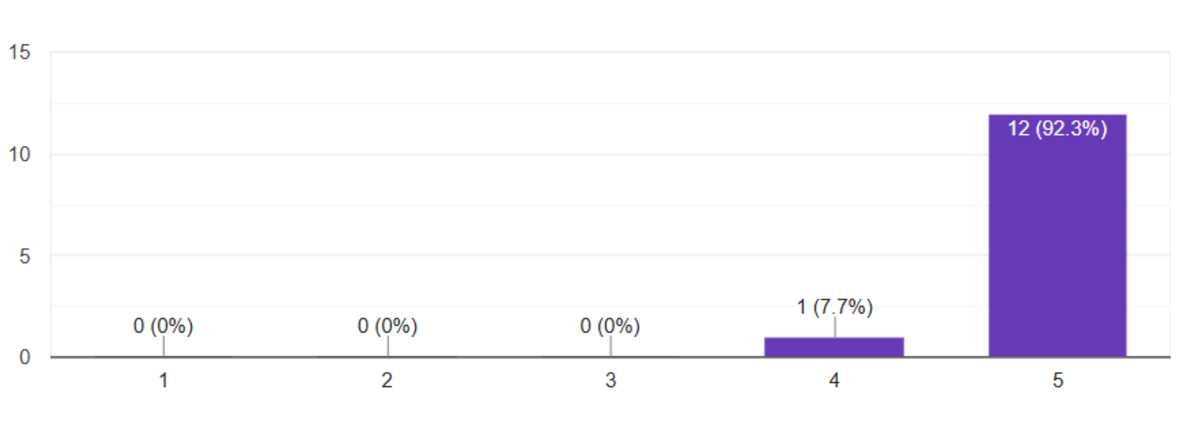
Figure 5.6: MPU6050

5.1 System Performance Evaluation

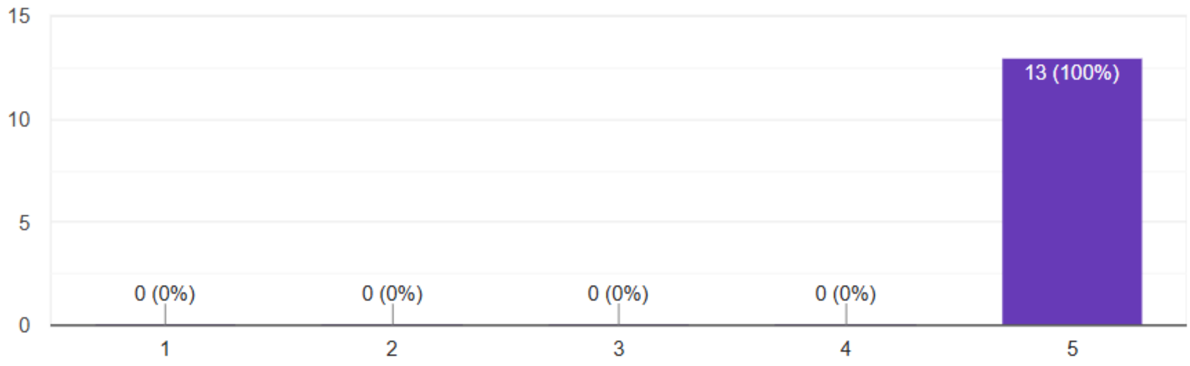
5.1.1 User Satisfaction Survey

A user study was conducted with participants from various demographics, including fitness enthusiasts and casual gamers. The survey focused on ease of use, responsiveness, and enjoyment. +

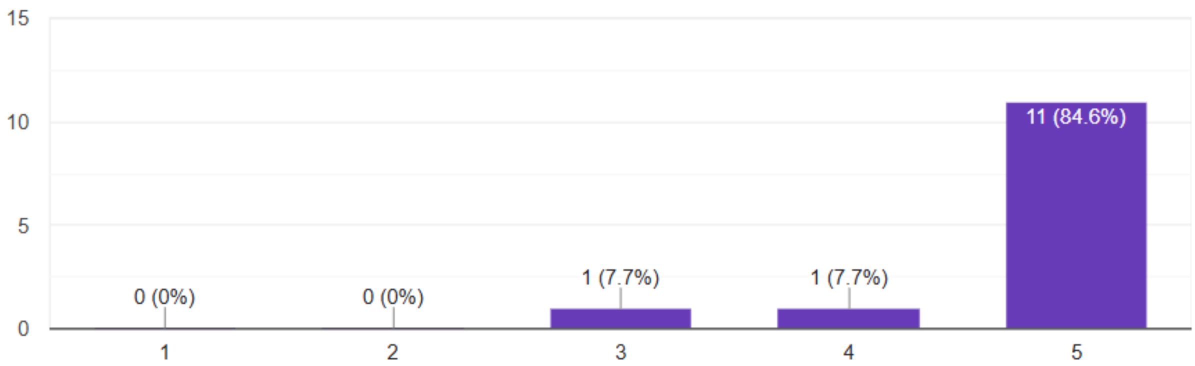
- How responsive was the system to your cycling movements?



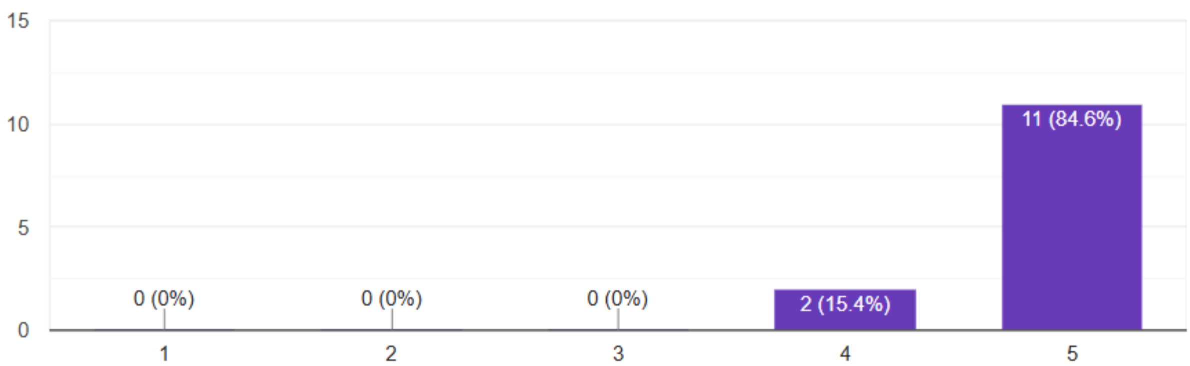
- Did the integration of cycling with gaming make exercising more enjoyable?



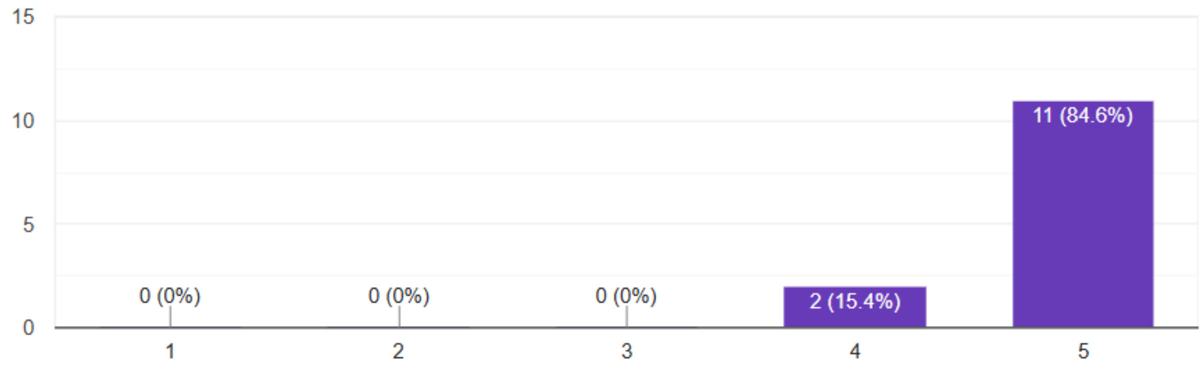
- Did using this system encourage you to exercise more than usual?



- Did you feel that using this system provided you with a good workout?



- Would you recommend this product to others?



5.2 Impact on Motivation for Physical Exercise

To determine the effect of the system on user motivation, feedback was collected regarding their workout experience:

- **Increased Exercise Duration:** Participants cycled for an average of 30% longer compared to traditional stationary cycling sessions.
- **Enhanced Engagement:** Users reported higher motivation levels due to the gamified experience.
- **Preference for Interactive Workouts:** A majority of participants expressed interest in using the system regularly as part of their fitness routine.

5.3 Discussion

The results demonstrate that integrating a stationary cycle with gaming technology significantly enhances user engagement and motivation for physical exercise. The system's low latency and accurate input mapping ensure a smooth experience, while user feedback highlights its potential to make workouts more enjoyable. Future improvements could include additional customization options and expanded compatibility with fitness applications.

Overall, the **MotionX Interactive Fitness System** successfully bridges the gap between gaming and fitness, providing an innovative solution to promote physical activity in an engaging manner.

Chapter 6

Conclusions & Future Scope

The project effectively illustrates how conventional exercise equipment such as a stationary cycle can be upgraded with sensor integration and gamification to enhance user motivation and engagement. By translating actual cycling movements into virtual game controls, users are motivated to exercise more regularly and enjoyably. The system provides an interactive and immersive experience through the use of Arduino-based hardware and vJoy input mapping, allowing it to be compatible with a range of games. User tests were positive in regards to exercise quality and entertainment value, confirming the effectiveness of the system.

In the future, the project can be extended in a number of ways. Future enhancements could involve integration with popular fitness applications for more accurate tracking, inclusion of multiplayer features for social interaction, and incorporation of more sensors for more detailed fitness analysis. A separate mobile application could also be created to offer real-time feedback, summaries of workouts, and customized goals. These additions will further increase the usability, scalability, and effectiveness of the system in encouraging physical activity using gamification.

References

[1] M. Afsar, S. Saqib, M. Aladfaj, M. H. Alatiyyah, K. Alnowaiser, H. Aljuaid, A. Jalal, and J. Park, "Body-worn sensors for recognizing physical sports activities in exergaming via deep learning model," *IEEE Access*, vol. 11, pp. 1–12, 2023.

[2] J. Schöning, J. Kettler, M. I. Jäger, and A. Gunia, "Grand theft auto-based cycling simulator for cognitive enhancement technologies in dangerous traffic situations," *Sensors*, 2023.

[3] M. T. Baldassarre, D. Caivano, S. Romano, F. Cagnetta, V. Fernandez-Cervantes, and E. Stroulia, "Phydslk: a model-driven framework for generating exergames," *Multimedia Tools and Applications*, 2021.

[4] I. Mocanu, C. Marian, L. Rusu, and R. Arba, "A kinect based adaptive exergame," in *International Conference on Communications (ICCP)*. IEEE, 2016.

[5] A. Author and B. Coauthor, "Validation of a novel bicycle simulator," in *2020 International Conference on Simulation and Modeling (SimMod)*. IEEE, 2020, pp. 12–18.

Appendix A: Presentation

MOTIONX INTERACTIVE FITNESS SYSTEM

MEMBERS

DANISH H MUHAMMED
HARINARAYANAN AM
KANNAN S
NAMDID NISHAD C

GUIDE

MEHBOOBA P SHAREEF

CONTENT

GAMING, SEDENTARY LIFESTYLE AND OUR SOLUTION

- Excessive Gaming & Obesity
- Lack of time for Exercise
- Lack of Physical Activity in Traditional gaming

MOTIONX INTERACTIVE FITNESS SYSTEM

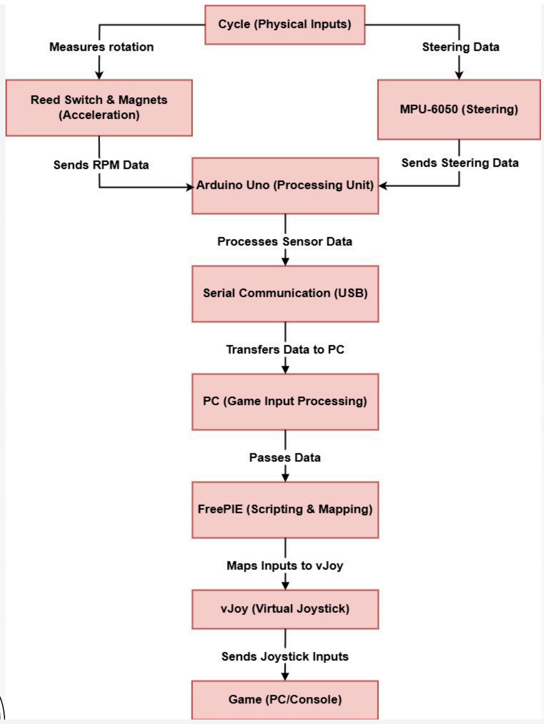
SO WHAT IS MOTIONX INTERACTIVE FITNESS SYSTEM

Converts a **stationary cycle** into a **game controller**

OBJECTIVES

- Develop an exergaming system
- Encourage regular physical activity
- Track real-time cycling inputs
- Sustainability
- Foster healthier lifestyles.

PROPOSED METHOD



- **Sensors Read Input**
  - The Reed Switch : pedal speed.
  - The MPU-6050 : handlebar movement.
- **Arduino Processes and Sends Data**
  - Sensor data from Arduino controls a virtual joystick via vJoy and FreePIE
- **Game Updates Based on Inputs**
  - Pedal speed and Handlebar movement.
  - we are only using NFS Most Wanted and SuperTuxKart

MODULE WISE DIVISION

1. SENSOR INPUT AND DATA COLLECTION

- Pedal Speed - Reed switch and Magnet
- Handlebar Movement - MPU-6050

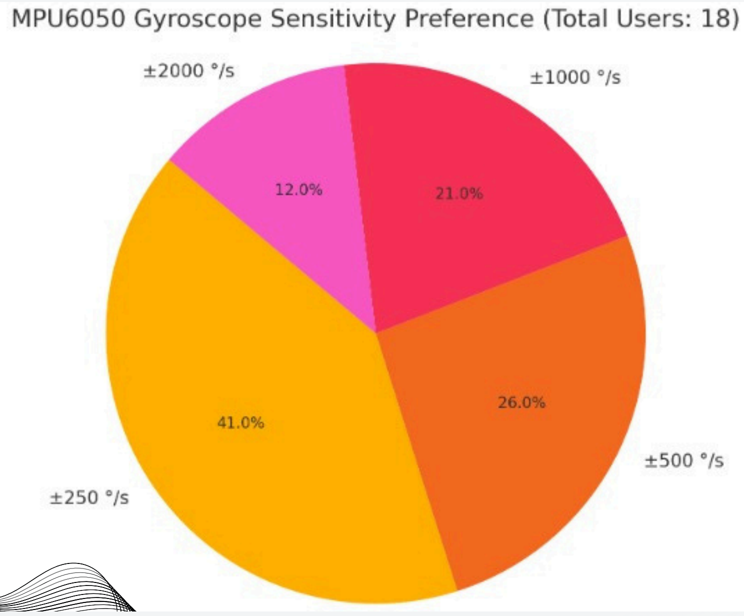
HANDLEBAR MOVEMENT

- Handlebar movement detected via MPU-6050 accelerometer and gyroscope.
- Focus on gyroscope X-axis (rotation around handlebar).
- Data measured in °/s with selectable ranges
  - ±250, ±500, ±1000 and ±2000.
- ±250°/s chosen based on user survey for better control.

HANDLEBAR MOVEMENT

- Handlebar movement detected via MPU-6050 accelerometer and gyroscope.
- Focus on gyroscope X-axis (rotation around handlebar).
- Data measured in °/s with selectable ranges
  - ±250, ±500, ±1000 and ±2000.
- ±250°/s chosen based on user survey for better control.
  - Users played the same game using all 4 configurations

HANDLEBAR MOVEMENT



HANDLEBAR MOVEMENT

- mpu.getMotion6() gives raw accelerometer values: ax, ay, az.
- atan2(ay, az) calculates roll angle using Y and Z-axis data.
- atan2 handles quadrants correctly, giving accurate tilt in degrees.

HANDLEBAR MOVEMENT

float roll = atan2(ay, az) * 180.0 / PI;

- atan2(ay, az): Computes tilt angle using Y & Z acceleration.
- 180.0 / PI: Converts radians to degrees.

HANDLEBAR MOVEMENT

- **Data Conversion:**
  - If the device is flat on a table, ay is near 0, and az is around ±gravity.
  - If the device is tilted to the side, ay increases, and the roll angle changes.

HANDLEBAR MOVEMENT

- Mapping to Game Input:
  - Angular velocity maps to joystick X-axis for steering.
  - X-axis ranges from -16384 (left) to +16384 (right).
  - More right tilt = higher X-axis value.

PEDAL MOVEMENT

- Pedal speed is measured using a reed switch connected to Arduino.
- A reed switch, mounted near a magnet on the pedal mechanism, detects each rotation as the magnet passes by.

PEDAL MOVEMENT

- reed_count reflects the number of pedal rotations.
- It's scaled using reed_count * 1000 to match vJoy Y-axis input.
- If the last 20 reed values are identical, speed is reset to 0 (bike stopped).

2. DATA PROCESSING & CONTROL MODULE

- **Arduino Uno :**
  - **Process sensor data and convert it into game inputs.**
  - **Sends the data to Free pie.**

2. DATA PROCESSING & CONTROL MODULE

- Processes data in real-time to calculate speed and handlebar tilt angle.
- Transmits processed data to a PC using USB Serial Communication.
- Integrates with FreePIE and vJoy for seamless game control.

3. GAME INPUT MAPPING & INTEGRATION

Software Components

- **FreePIE (Python Input Emulator):**
  - Maps incoming data from the Arduino into usable game commands.
- **vJoy (Virtual Joystick Driver):**
  - Converts data into joystick inputs, making the cycle function as a standard game controller.

3. GAME INPUT MAPPING & INTEGRATION

Data Mapping Process

- **Speed Data → Game Speed**
    - Number of pedal rotations decides the acceleration levels in-game.
- **Steering Data → Directional Input**
    - Left/Right tilt mapped to joystick left/right movements.

3. GAME INPUT MAPPING & INTEGRATION

Data Conversion in FreePie:

- Mapping Handlebar Tilt to X-axis:
    - The roll angle is between -30° (left) to +30° (right).
    - It is mapped to the vJoy X-axis range (-16384 to +16384):

3. GAME INPUT MAPPING & INTEGRATION

- **Data Conversion in FreePie:**
  - joystick_x = int((roll + 30) * (32768 / 60) - 16384)
  - joystick_x = max(-16384, min(16384, joystick_x))

3. GAME INPUT MAPPING & INTEGRATION

Data Conversion in FreePie:

- Mapping Pedal Speed to Y-axis (Acceleration):
- The reed_count (number of pedal rotations) is mapped to the Y-axis:
- It scales linearly up to a maximum value of 16384

3. GAME INPUT MAPPING & INTEGRATION

accel_value = min(16384, reed_count * 1000)

- This means each reed count increases the acceleration by 1000 units.
- If reed_count remains unchanged for 20 readings, it resets to 0.

ASSUMPTIONS

- The hardware (sensors, bike) will communicate seamlessly with the game engine.
- The hardware components will accurately track user performance without major lag.
- Players will find the combination of physical exercise and gaming motivating enough to stick with the system.
- The users will have enough space and resources to use the hardware comfortably at home or gyms.

WORK BREAKDOWN & RESPONSIBILITIES

Arduino Programming

1.Namdid

Arduino programming for data processing and mapping inputs to actions.

2.Kannan

Data integration from arduino to game controls via FreePie and serial communication.

Hardware Development

3.Danish

Hardware setup and sensor integration (reed switch and MPU-6050).

4.Harinarayanan

Game integration using Vjoy and overall testing/debugging..

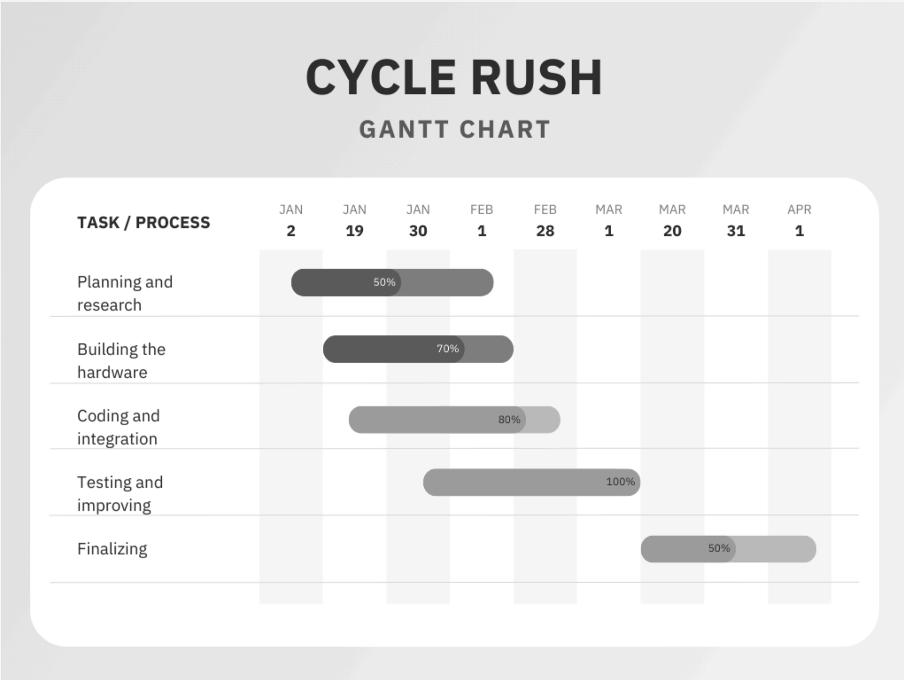
REQUIREMENTS

Hardware Requirements

1. Cycle Frame
2. Iron, Aluminum (Frame)
3. Arduino Uno
4. MPU-6050
5. Reed Switches & Magnets
6. Wires, Connectors, and Mounting Hardware

Software Requirements

1. Arduino IDE
2. Libraries (for sensors and networking)
3. FreePie
4. Vjoy



Sl.No.	Item	Expenditure
1	Cycle Frame	3000
2	Iron, Aluminum (Frame)	2500
3	Arduino Uno	2500
4	MPU-6050	400
5	Reed Switches & Magnets	400
6	Wires, Connectors, and Mounting Hardware	1000

SL.NO	Item	Expenditure
7	Welding (Labour cost)	2000
8	Paint, Thinner and brush	1000
9	Miscellaneous Costs (incl. Power Supply)	600
10	Optional Heart Rate Sensor (Max 30102)	800
Total Cost:		14200

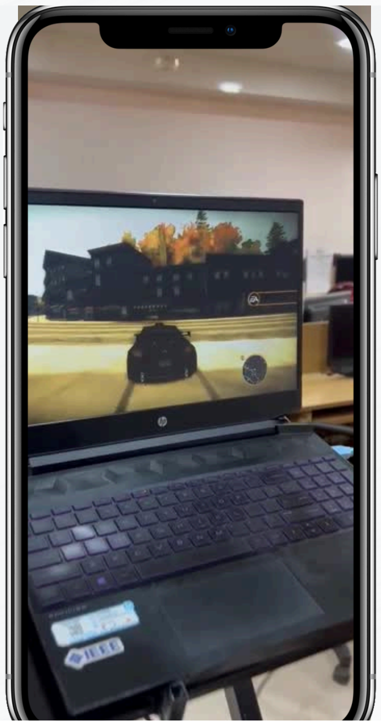
RISK AND CHALLENGES

- **Sensor Accuracy and Latency :** The MPU-6050 sensors may introduce slight delays or inaccuracies in measuring speed and handlebar movement, especially at high speeds.
- **Calibration and Setup :** Misalignment between the physical inputs (e.g., pedaling speed, handlebar tilt) and their in-game representation.

OUTCOME

- **Immersive Exergaming Experience:** Players can control the game using the stationary cycle.
- **Real-time Data Integration:** Sensors track speed and steering for smooth gameplay.
- **Enhanced Fitness Feedback:** Heart rate sensor adjusts game difficulty.

OUTCOME



RESULT

MotionX prototype

Arduino

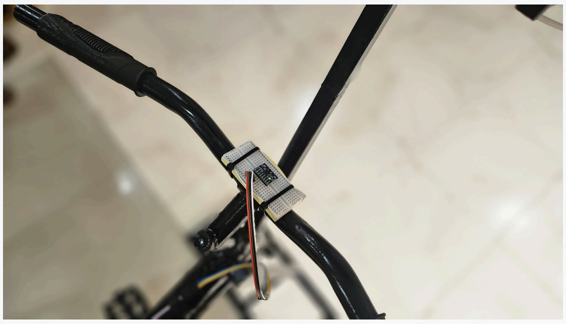
RESULT

Reed Switch

Reed Switch sensor magnet

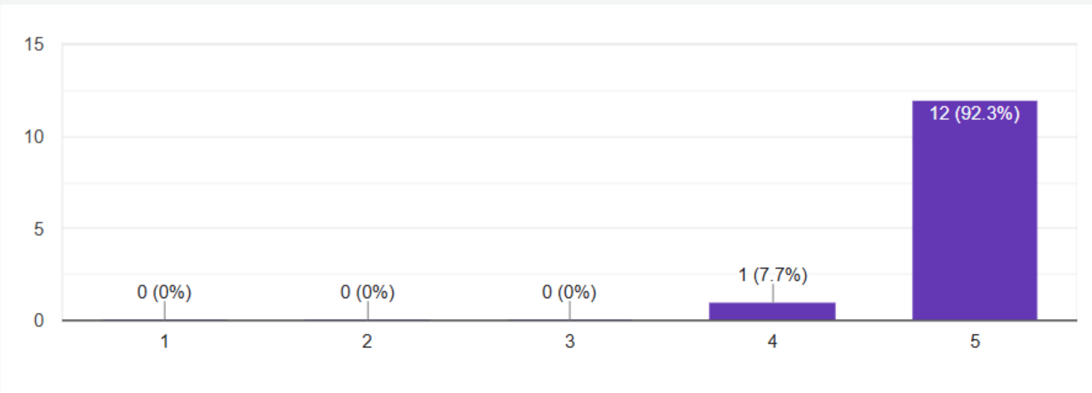
RESULT

Laptop

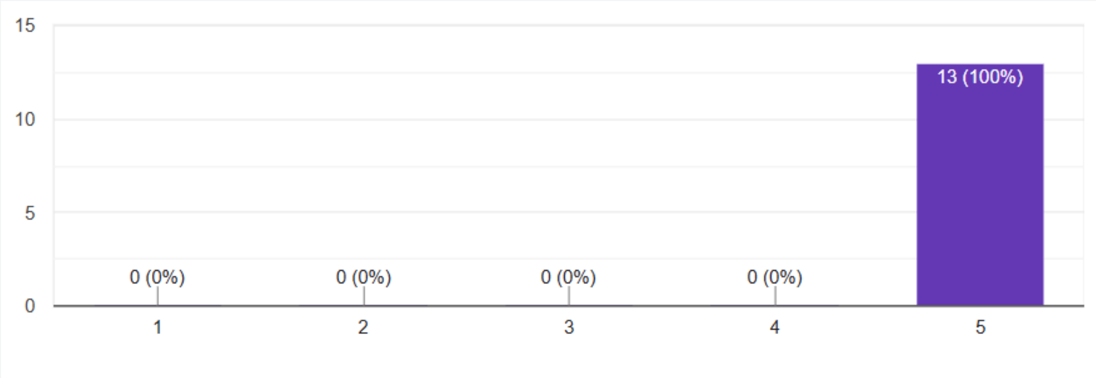


MPU6050

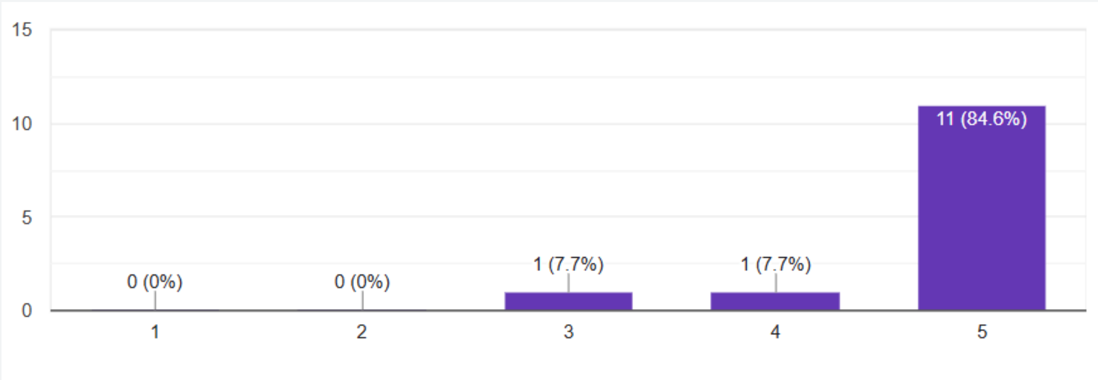
How responsive was the system to your cycling movements?



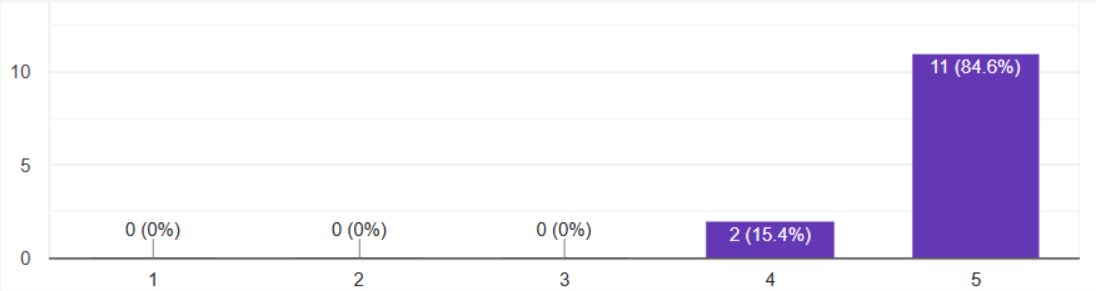
Did the integration of cycling with gaming make exercising more enjoyable?



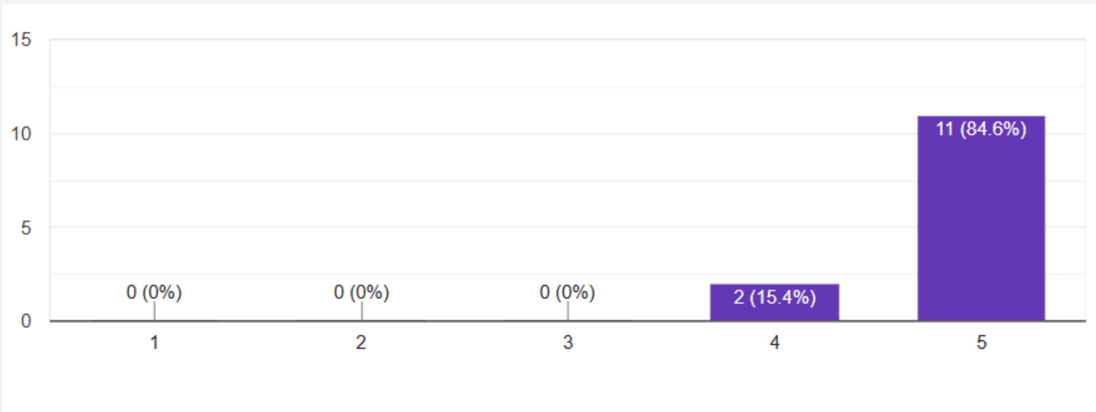
Did using this system encourage you to exercise more than usual?



Did you feel that using this system provided you with a good workout?



Would you recommend this product to others?



CONCLUSION

This project turns a stationary cycle into an engaging game that promotes fitness. Real-time sensors adjust speed, steering, and difficulty, making workouts fun and personalized while tracking performance and calories.

REFERENCES

- M. M. Afsar et al., "Body-Worn Sensors for Recognizing Physical Sports Activities in Exergaming via Deep Learning Model," in IEEE Access, vol. 11, pp. 12460-12473, 2023, doi: 10.1109/ACCESS.2023.3239692.

- T.A. Shloul et al., "Student’s Health Exercise Recognition Tool for E-Learning Education," Intell. Automat. Soft Comput. , vol. 35, no. 1, pp. 149-161. 2023. https://doi.org/10.32604/iasc.2023.026051

- Ashwini, K., Amutha, R. Compressive sensing based recognition of human upper limb motions with kinect skeletal data. Multimed Tools Appl 80, 10839–10857 (2021). https://doi.org/10.1007/s11042-020-10327-4

REFERENCES

- M. Nazemi, M. A. B. van Eggermond, A. Erath, and K. W. Axhausen, "Studying cyclists' behavior in a non-naturalistic experiment utilizing cycling simulator with immersive virtual reality," in Annual Meeting of the Transportation Research Board, Washington D.C., 2023.

- A. Hamad, "Two Wheelistic: Development of a High-Fidelity Virtual Reality Cycling Simulator for Transportation Safety Research," Master's thesis, University of Michigan, Ann Arbor, MI, USA, May 2021

- M. L. Smith and J. Desnoyers-Stewart, "Riding Through Walls: A journey of physical computing through Google Street View," in Electronic Visualisation and the Arts (EVA 2019), London, UK, July 2019. doi: 10.14236/ewic/eva2019.71

REFERENCES

- J. Schöning, J. Kettler, M. I. Jäger, and A. Gunia, "Grand Theft Auto-Based Cycling Simulator for Cognitive Enhancement Technologies in Dangerous Traffic Situations," Sensors, vol. 23, no. 7, p. 3672, 2023. doi: 10.3390/s23073672.

- Jelle Haasnoot, Riender Happee, Volkert van der Wijk & Arend L. Schwab(2024) Validation of a novel bicycle simulator with realistic lateral and roll motion, Vehicle System Dynamics, 62:7, 1802-1826, DOI: 10.1080/00423114.2023.2264418

- Dialynas G, Schwab A, Happee R. Design and hardware selection for a bicycle simulator.Mech Sci. 2019;10:1–10. doi: 10.5194/ms-10-1-2019

REFERENCES

- Schramka F, Arisona S, Joos M, et al. Development of virtual reality cycling simulator. JCP.2018;13(6):603–605. doi: 10.17706/jcp.13.6.603-615
- Powell J. Hardware design for an electro-mechanical bicycle simulator in an immersive virtualreality environment [master's thesis]. University of Iowa; 2017
- Shoman MM, Imine H. Bicycle simulator improvement and validation. IEEE Access.2021;9:55063–55076. doi: 10.1109/ACCESS.2021.3071214

Appendix B: Vision, Mission, Programme Outcomes and Course Outcomes

Vision, Mission, Programme Outcomes and Course Outcomes

Institute Vision

To evolve into a premier technological institution, moulding eminent professionals with creative minds, innovative ideas and sound practical skill, and to shape a future where technology works for the enrichment of mankind.

Institute Mission

To impart state-of-the-art knowledge to individuals in various technological disciplines and to inculcate in them a high degree of social consciousness and human values, thereby enabling them to face the challenges of life with courage and conviction.

Department Vision

To become a centre of excellence in Computer Science and Engineering, moulding professionals catering to the research and professional needs of national and international organizations.

Department Mission

To inspire and nurture students, with up-to-date knowledge in Computer Science and Engineering, ethics, team spirit, leadership abilities, innovation and creativity to come out with solutions meeting societal needs.

Programme Outcomes (PO)

Engineering Graduates will be able to:

1. Engineering Knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

2. Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

3. Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

4. Conduct investigations of complex problems: Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. Modern Tool Usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

6. The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. Individual and Team work: Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.

10. Communication: Communicate effectively with the engineering community and with society at large. Be able to comprehend and write effective reports documentation. Make effective presentations, and give and receive clear instructions.

11. Project management and finance: Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work, as a member and leader in a team. Manage projects in multidisciplinary environments.

12. Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change.

Programme Specific Outcomes (PSO)

A graduate of the Computer Science and Engineering Program will demonstrate:

PSO1: Computer Science Specific Skills
The ability to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas by understanding the core principles and concepts of computer science and thereby engage in national grand challenges.

PSO2: Programming and Software Development Skills
The ability to acquire programming efficiency by designing algorithms and applying standard practices in software project development to deliver quality software products meeting the demands of the industry.

PSO3: Professional Skills
The ability to apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs thereby evolving as an eminent researcher and entrepreneur.

Course Outcomes (CO)

Course Outcome 1: Model and solve real world problems by applying knowledge across domains (Cognitive knowledge level: Apply).

Course Outcome 2: Develop products, processes or technologies for sustainable and socially relevant applications (Cognitive knowledge level: Apply).

Course Outcome 3: Function effectively as an individual and as a leader in diverse teams and to comprehend and execute designated tasks (Cognitive knowledge level: Apply).

Course Outcome 4: Plan and execute tasks utilizing available resources within timelines, following ethical and professional norms (Cognitive knowledge level: Apply).

Course Outcome 5: Identify technology/research gaps and propose innovative/creative solutions (Cognitive knowledge level: Analyze).

Course Outcome 6: Organize and communicate technical and scientific findings effectively in written and oral forms (Cognitive knowledge level: Apply).

Appendix C: CO-PO-PSO Mapping

CO - PO Mapping

CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
1	2	2	2	1	2	2	2	1	1	1	1	2
2	2	2	2		1	3	3	1	1		1	1
3									3	2	2	1
4					2			3	2	2	3	2
5	2	3	3	1	2							1
6					2			2	2	3	1	1

CO - PSO Mapping

CO	PSO 1	PSO 2	PSO 3
1	3		
2		2	
3			3
4			3
5	3		
6			3

Justification

Mapping	**Justification**
CO1 - PO1	Apply suitable computer science theory and mathematical foundations to identify relevant research areas
CO1 - PO2	Analyse the different problems and select relevant literature
CO1 - PO3	Create explainable content about the concepts to be conveyed to the audience
CO1 - PO4	Apply mathematical foundations and computer science theory to solve technical problems in the design of computer-based systems
CO1 - PO6	Apply multidisciplinary knowledge in projects, utilizing suitable computer science theory and mathematical foundations in team-oriented tasks
CO1 - PO7	Identify solutions from literature that can lead to a sustainable solution
CO1 - PO12	Analyze the recent and relevant works done in their area of research to understand the methodologies
CO2 - PO1	Analyze academic documents using mathematical foundations and algorithm principles relevant to the modeling and design of computer-based systems.
CO2 - PO2	Analyze and comprehend academic documents from the literature related to computer-based system modeling and design
CO2 - PO3	Evaluate and analyse computing requirements for technical programs in relevant literature chosen and understand them
CO2 - PO4	Identify the new technologies and modern engineering tools used and understand their working
CO2 - PO6	Apply comprehension from the literature to assess health, societal and legal issues and their relation to the engineering problems.
CO2 - PO7	Read and apply the context from the literature for sustainable development
CO2 - PO12	Ability to read and summarize the developments in engineering for lifelong learning process
CO3 - PO1	Create presentations applying mathematical foundations and algorithmic principles to communicate the design of computer-based systems
CO3 - PO2	Present academic documents effectively, applying computing and mathematical principles for problem analysis
CO3 - PO5	Present academic documents effectively, applying computing and mathematical principles in problem-solving
CO3 - PO8	Present multidisciplinary knowledge effectively,applying computing and mathematical principles in seminar presentations
CO3 - PO10	Present academic documents effectively, demonstrating communication skills
CO3 - PO12	Present the concepts and topics under study effectively

Mapping	Justification
CO4 - PO1	Apply mathematical foundations and algorithmic principles to give presentations about computer-based system modeling
CO4 - PO5	Give presentations about academic documents, applying modern tools and techniques
CO4 - PO8	Give presentations about academic documents keep up engineering norms and ethics
CO4 - PO10	Presenting the seminar strengthens the knowledge of the topic and promotes communication skills
CO4 - PO12	Presenting the seminar strengthens the knowledge of the topic and promotes life-long learning for continuous professional development
CO5 - PO1	Prepare a technical report and apply mathematical foundations in designing computer systems
CO5 - PO2	Enhances problem-solving for formulating technical solutions
CO5 - PO3	Develops skills in defining computing requirements and system design
CO5 - PO4	Prepare a technical report for experiments on complex problems
CO5 - PO5 CO5 - PO6	Promotes experimentation and analysis for evidence-based reports Prepare technical reports that can be useful for the society as well
CO5 - PO8	Prepare a technical report and apply mathematical foundations in designing computer systems making sure that publication ethics are met
CO5 - PO10	Sets professional, ethical, and legal principles in technical reports
CO5 - PO12	Record the academic reports and concepts for life-long learning
CO1-PSO1	Students will be able to apply their skills to identify, analyze, and formulate solutions for complex, multidisciplinary engineering problems through the strategic use of academic literature.
CO1-PSO2	Students will be able to navigate academic literature effectively while applying programming concepts, thereby preparing them for more advanced learning and practical applications in the field of computer science.
CO1-PSO3	Students will demonstrate their ability to interpret and utilize fundamental computer science concepts to conduct competitive research through academic literature.
CO2-PSO1	Students will be able to evaluate and design solutions for complex engineering problems in multidisciplinary areas by critically analyzing and interpreting academic documents from the literature.
CO2-PSO2	Students will be able to locate and utilize academic literature effectively while applying programming concepts, thereby equipping them for further learning and practical applications in the field of computer science.
CO2-PSO3	Students will have the ability to understand and apply the fundamentals of computer science in competitive research by interpreting and utilizing academic documents from the literature.

Mapping	**Justification**
CO3- PSO1	Students will be able to identify, analyze, and develop solutions for complex engineering problems across multidisciplinary areas while creating a presentation on an academic document
CO3-PSO2	Students will demonstrate the ability to apply the fundamentals of computer science in competitive research by designing a presentation based on an academic document
CO4- PSO1	Students will be able to to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas while giving a presentation about an academic document
CO4- PSO3	Students will have the ability to apply the fundamentals of computer science in competitive research while giving a presentation about an academic document
CO5- PSO1	Students will be able to to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas while preparing a technical report