Arab Academy for Science, Technology and Maritime Transport

College of Engineering and Technology

Computer Engineering and Electronics & Communication Engineering

B. Sc. Final Year Project

CONTROL SYSTEM FOR SPECIAL NEEDS

A Brain Computer Interface Approach

Presented By:

Mazen El Sedfy, Mariam Reda, Omar El Leissy, Ahmed Tarek, Mohab Haytham, and Marwan Mahrous

Supervised By:

Prof. Mohamed Taher El Soni, Dr. Mohamed Hossam Zaky, and Dr. Noureldin El Medany

JULY– 2025

DECLARATION

I hereby certify that the material which I now submit for assessment in the programme of study leading to the award of the Bachelor of Engineering in Computer Engineering and Electronics and Communication Engineering is entirely our own work. I affirm that I have exercised reasonable care to ensure that the work is original, does not, to the best of my knowledge, breach any copyright laws, and has not been copied or taken from the work of others, except where such work has been properly cited and acknowledged within the text.

Furthermore, I declare that this work is the intellectual property of the Arab Academy for Science, Technology and Maritime Transport – AASTMT, Abu Qir Campus, specifically the Department of Computer Engineering and the Department of Electronics and Communication Engineering. This project and its contents are intended solely for academic purposes. No part of this work may be copied, distributed, reproduced, or used in any manner without the explicit written consent of the supervisors named in this document, the AASTMT Abu Qir Branch, and the project team members.

MAZEN SAMEH MOHAMED MOHMED EL SEDFY	20100527
MARIAM REDA MOHAMED EL SAED	20102684
AHMED TAREK AHMED MOHAMED ABD ALMAJED	20102766
OMAR AYMAN HASSAN ALY EL LEISSY	20100249
MOHAB HAYTHAM MOHAMED SAMY BADAWY ALI	20103187
MARWAN MOHAMED MAHROS ALSETOHY	20100342

Date: 09, 07, 2025

ARAB ACADEMY FOR SCIENCE TECHNOLOGY AND MARITIME TRANSPORT

COLLEGE OF ENGINEERING AND TECHNOLOGY

Departments of Computer Engineering and Electronics & Communication Engineering

Academic Year: 2024/2025 Semester: 10

Senior Project Summary Report

Project Title	Control System for Special needs: A Brain Computer Interface Approach
Supervisors	1. Prof. Mohamed Taher El Soni 2. Dr. Mohamed Hossam Zaky 3. Dr. Noureldin El Medany
Team Members	1. Mazen El Sedfy 20100527 2. Omar El Leissy 20100249 3. Ahmed Tarek 20102766 4. Mohab Haytham 20103187 5. Mariam Reda 20102684 6. Marwan Mahrous 20100342
Project Deliverables	1. EEG signal acquisition module using Hybrid Unicorn Black Headset. 2. P300 detection & classification system with ML algorithms. 3. BCI-controlled applications: UNO game, maze game, educational website. 4. Accessible user interfaces for special needs users. 5. System evaluation report with performance metrics.
Abstract	This project develops a P300-based Brain-Computer Interface (BCI) using the Hybrid Unicorn Black Headset to help individuals with motor disabilities. It enables control of educational websites and computer games through EEG signals. Machine learning ensures accurate P300 detection and user-friendly interfaces. The system aims to enhance independence, accessibility, and engagement for disabled users.
Engineering Standards	1. IEEE 802.15 – Wireless Communication Standard 2. IEEE 11073 – Medical Device Communication 3. IEEE 754 – Floating-Point Arithmetic 4. ISO/IEC 9126 – Software Quality 5. ISO/IEC 25010 – System and Software Quality Models 6. ISO/IEC 27001 – Information Security Management 7. General Data Protection Regulation (GDPR) 8. ABET Engineering Standards

Design Constraints	● **Real-time Processing**: Requires low-latency signal acquisition and response. ● **User Accessibility**: Interfaces must be intuitive for users with motor disabilities. ● **Hardware Limitations**: Operates within the specs of the headset (8 channels, 250 Hz). ● **Data Privacy**: Must comply with ethical handling and protection of EEG data. ● **Data Acquisition**: Limited data per subject.
Project Impact	● **Empowerment**: Enables motor-impaired users to interact with digital systems independently. ● **Innovation**: Demonstrates practical use of P300-based BCI in education and gaming. ● **Awareness**: Highlights the importance of inclusive assistive technology. ● **Research Contribution**: Adds to the growing body of BCI applications and methodologies.
Team Organisation	1. **Interface & Communication**: Ahmed Tarek & Mazen El Sedfy 2. **Machine Learning**: Ahmed Tarek, Mariam Reda, Omar El Leissy, Mazen El Sedfy, and Mohab Haytham 3. **Signal Processing & Feature Extraction**: Mariam Reda 4. **Web Development**: Marwan Mahrous & Omar El Leissy & Ahmed Tarek 5. **Game Development**: Mohab Haytham & Mazen El Sedfy 6. **Research and Developments**: All Members 7. **Data Acquisition Contribution**: Omar El Leissy & Hamza Wasief
Ethics /Safety	● **Informed Consent**: Participants were fully briefed and consented to EEG usage. ● **Data Protection**: EEG data stored securely and anonymized. ● **User Comfort**: Choose a dry electrode headset to avoid skin irritation or discomfort. ● **Bias Minimization**: Testing conducted with fairness and inclusivity in mind. ● **No Harm Policy**: System designed to pose no physical or psychological risk to users.

Main Supervisors Signatures

Mohamed Taher El Soni

Mohamed Hossam Zaky

Noureldin El Medany

Acknowledgment

First and foremost, we extend our deepest gratitude to Almighty God for granting us the strength, patience, and guidance throughout this journey. We are profoundly thankful to our esteemed supervisors— Prof. Mohamed Taher El Soni, Dr. Mohamed Hossam Zaky, and Dr. Noureldeen El Medany—for their invaluable support, insightful feedback, and unwavering encouragement that greatly enriched our work. We also sincerely thank Hamza Wasief, our friend, whose involvement and cooperation were essential to our research. Our heartfelt appreciation goes to our parents, whose love, sacrifices, and constant belief in us have been a cornerstone of our success. Lastly, we take a moment to recognize ourselves—for the dedication, teamwork, and resilience that made this graduation project possible.

Abstract

This project presents the design and development of a non-invasive P300-based Brain-Computer Interface (BCI) system aimed at assisting individuals with severe motor disabilities. Utilizing the Hybrid Unicorn Black Headset for EEG signal acquisition, the system translates P300 event-related potentials into actionable commands to control educational websites and custom BCI games like UNO and Maze. Advanced signal processing and machine learning algorithms were integrated to ensure accurate P300 detection and classification, alongside user-friendly interfaces tailored to the needs of handicapped users. The project addresses key challenges such as signal noise, usability, and accessibility, with the goal of enhancing engagement and overall quality of life for disabled individuals while contributing valuable insights to the field of assistive BCI technologies.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS/ABBREVIATIONS

1. **EEG**: Electroencephalography
2. **BCI**: Brain-Computer Interface
3. **ERP**: Event-Related Potential
4. **P300**: An event-related potential that typically peaks around 300 milliseconds after the presentation of a rare, task-relevant, or unexpected stimulus
5. **AI**: Artificial Intelligence
6. **ML**: Machine Learning
7. **UI/UX**: User Interface/User Experience
8. **UDP**: User Datagram Protocol
9. **GDPR**: General Data Protection Regulation
10. **ISO/IEC 9126**: Software Quality standard
11. **ISO/IEC 25010**: System and Software Quality Models standard
12. **ISO/IEC 27001**: Information Security Management standard
13. **IEEE 802.15**: Wireless Communication Standard
14. **IEEE 11073**: Medical Device Communication standard
15. **IEEE 754**: Floating-Point Arithmetic standard
16. **EOG**: Ocular Artifacts
17. **EMG**: Muscle Artifacts
18. **ECG**: Cardiac Artifacts
19. **ICA**: Independent Component Analysis
20. **LDA**: Linear Discriminant Analysis
21. **SVM**: Support Vector Machines
22. **CNN**: Convolutional Neural Networks
23. **AUC**: Area Under the Curve
24. **ROC**: Receiver Operating Characteristic (part of ROC-AUC)
25. **SNR**: Signal-to-Noise Ratio
26. **CSP**: Common Spatial Patterns
27. **RCSP**: Regularized Common Spatial Pattern
28. **LSTEEG**: LSTM-based autoencoders
29. **ADC**: Analog-to-Digital Converter
30. **KPIs**: Key Performance Indicators
31. **ALS**: Amyotrophic Lateral Sclerosis
32. **SCI**: Spinal Cord Injuries
33. **BMC**: Business Model Canvas
34. **ITR**: Information Transfer Rate
35. **SSVEP**: Steady-State Visually Evoked Potential
36. **ECOG**: electrocorticography
37. **MEG**: magnetoencephalography
38. **LOOCV:** Leave-One-Out Cross-Validation

1 CHAPTER 1: INTRODUCTION

1.1 PROJECT OVERVIEW: A P300-BASED BCI SYSTEM FOR THE SPECIAL NEEDS

This project aims to design and develop a non-invasive Brain-Computer Interface (BCI) system that translates electroencephalogram (EEG) signals into actionable commands, enabling individuals with severe motor impairments to interact with digital environments. The system leverages the detection of the P300 event-related potential (ERP), a neural response elicited by rare or task-relevant stimuli, to allow users to make selections without requiring muscle control. The proposed system employs the Hybrid Unicorn Black Headset for EEG signal acquisition and integrates advanced signal processing techniques along with machine learning algorithms for accurate P300 detection and classification. It supports user interaction with assistive applications such as educational websites and BCI-controlled games, offering a novel means of engagement and cognitive stimulation for individuals with limited mobility [1].

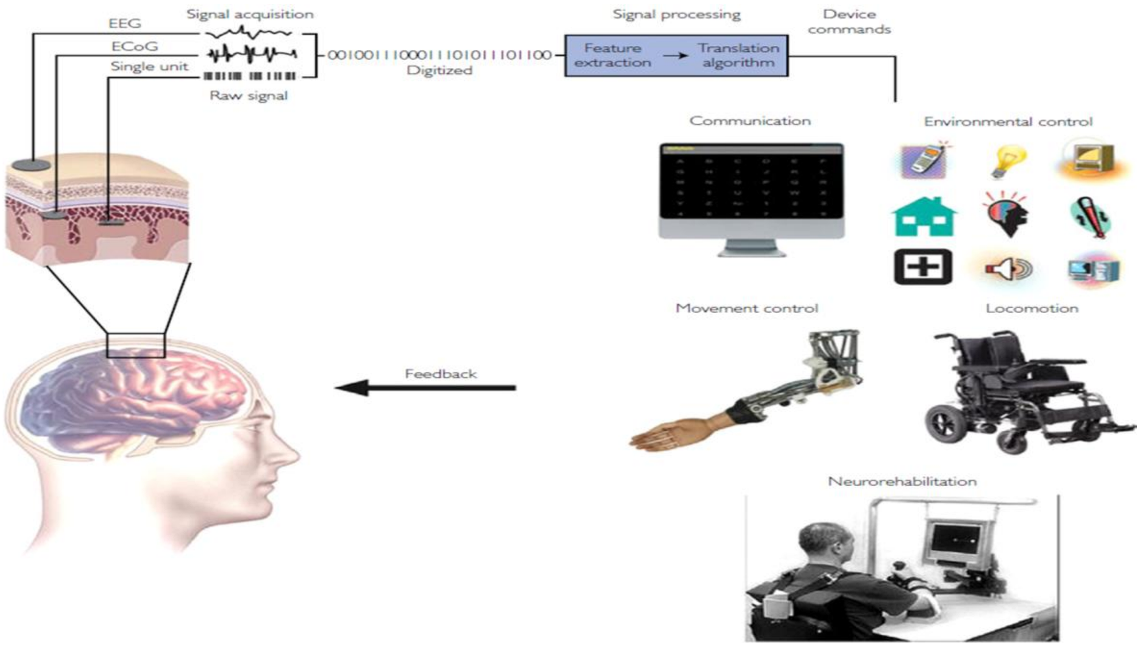


Figure 1.1: BCI Development Loop

1.2 MOTIVATION AND APPLICATIONS

Individuals with severe motor impairments often face significant challenges in communication and daily living activities. BCI systems offer a promising alternative by interpreting brain activity directly, bypassing the need for muscular pathways. Among various BCI paradigms, the P300-based approach stands out due to its non-invasiveness, minimal training requirements, and intuitive operation, where users focus attention on target items to elicit a detectable neural response. Potential applications include communication aids, environmental control systems, rehabilitation tools, and leisure/educational platforms, enhancing functional independence and psychological well-being [2].

1.3 HISTORICAL DEVELOPMENT OF BCI SYSTEMS

The concept of BCIs dates back to the 1970s, with early systems largely invasive, involving implanted electrodes. Over decades, advancements in non-invasive EEG technology, coupled with progress in signal processing and machine learning, enabled the development of practical, user-friendly BCI systems. A key milestone was the discovery and characterization of event-related potentials (ERPs), particularly the P300 wave, identified in 1965, which has become central to many modern BCI applications due to its reliability. Today, BCIs are increasingly applied in clinical and consumer domains, ranging from assistive communication to gaming and education [3].

1.4 EEG-BASED SYSTEMS AND EVENT-RELATED POTENTIALS

Electroencephalography (EEG)-based BCIs utilize scalp-recorded electrical activity generated by cortical neurons to interpret brain states. Among the various features extracted from EEG signals, event-related potentials (ERPs)—particularly the P300 component—are widely used due to their clear temporal structure and ease of elicitation through visual or auditory oddball paradigms. The P300 ERP is typically evoked in response to a rare, task-relevant stimulus among frequent non-target stimuli. Its latency (~300 ms post-stimulus) and amplitude vary depending on factors such as stimulus probability, task relevance, and individual cognitive load. These characteristics make it suitable for use in BCI systems where users can select options by focusing on specific targets [4].

1.5 THE HYBRID UNICORN BLACK HEADSET

For this project, the Hybrid Unicorn Black Headset is selected as the EEG acquisition device. This headset offers a combination of dry and gel-based electrodes, ensuring both comfort and signal quality during prolonged usage. It supports wireless connectivity and real-time signal streaming, making it suitable for BCI applications requiring low-latency feedback. The headset includes eight active EEG channels positioned according to the international 10-20 system, targeting regions such as Fz, Cz, Pz, Oz, and others, which are known to exhibit strong P300 responses. Its compatibility with open-source software frameworks facilitates integration with custom-built signal processing pipelines and machine learning classifiers [5].

1.6 PROBLEM STATEMENT AND PROJECT OBJECTIVES

Despite advances in assistive technologies, there remains a critical gap in accessible and intuitive interfaces for individuals with severe motor disabilities. Many existing solutions either require residual muscle control or suffer from usability issues such as slow response times and high error rates. This project addresses these limitations by developing a P300-based BCI system that acquires high-quality EEG signals, implements robust signal preprocessing and feature extraction methods, applies machine learning algorithms to classify P300 patterns with high accuracy, and develops a user-friendly graphical interface for real-time interaction with assistive applications [6].

2 CHAPTER 2: MEDICAL AND TECHNICAL BACKGROUND

2.1 THE HUMAN BRAIN AND ANATOMY

The human brain is one of the most complex biological systems, consisting of over 100 billion neurons that process and transmit both chemical and electrical signals [89]. These neurons form the foundation of the central nervous system (CNS), which acts as the command center for interpreting sensory input, regulating bodily functions, and facilitating cognition, memory, emotion, and decision-making [91]. The brain is divided into three major regions: the cerebrum**,** cerebellum**,** and brainstem. Each of these plays a vital role in managing essential physiological and cognitive processes. For instance, the brainstem governs basic autonomic functions such as breathing and heartbeat, while the cerebellum is responsible for motor control and balance. The cerebrum, the largest and most complex part handles higher-order functions like reasoning, problem-solving, and voluntary movements [90]. Throughout the body, the brain communicates using electrochemical impulses, interpreting incoming signals and generating appropriate responses. Some messages remain within the brain, while others are transmitted via the spinal cord and peripheral nervous system to distant body parts. These messages govern both involuntary and voluntary actions, from digestion and heart rate to speech and physical movement [89].
To accurately interpret EEG signals, it is essential to understand the functional anatomy of the **cerebral cortex**, which is divided into four primary lobes:

- **Frontal Lobe**
  Located at the front of the brain, this is the largest lobe and is associated with executive functions, voluntary movement, sensory and speech production. It houses Broca's area, critical for language expression. It also influences personality, behavior, and decision-making [90].
- **Parietal Lobe**
  Situated behind the frontal lobe, the parietal lobe processes somatosensory information such as touch, temperature, and pain. It also contributes to spatial orientation and object recognition. Part of Wernicke's area, which is involved in understanding spoken language, is found here [89].
- **Temporal Lobe**
  Located beneath the parietal lobe on the brain's sides, this region governs auditory perception, short-term memory, and aspects of emotion. It also plays a role in language comprehension and rhythm recognition [90].
- **Occipital Lobe**
  Found at the back of the brain, this lobe is primarily responsible for visual processing, including recognition of shapes, colors, and motion [89].

Additional regions include:

- **Cerebellum**
  Positioned beneath the occipital lobe, the cerebellum is essential for coordinating muscle movements and maintaining balance and posture. It fine-tunes motor activity to ensure smooth execution of voluntary tasks [91].
- **Brainstem**
  Connecting the brain to the spinal cord, the brainstem controls basic survival functions such as breathing, heart rate, blood pressure, digestion, and temperature regulation. It also manages consciousness and the sleep-wake cycle [89].

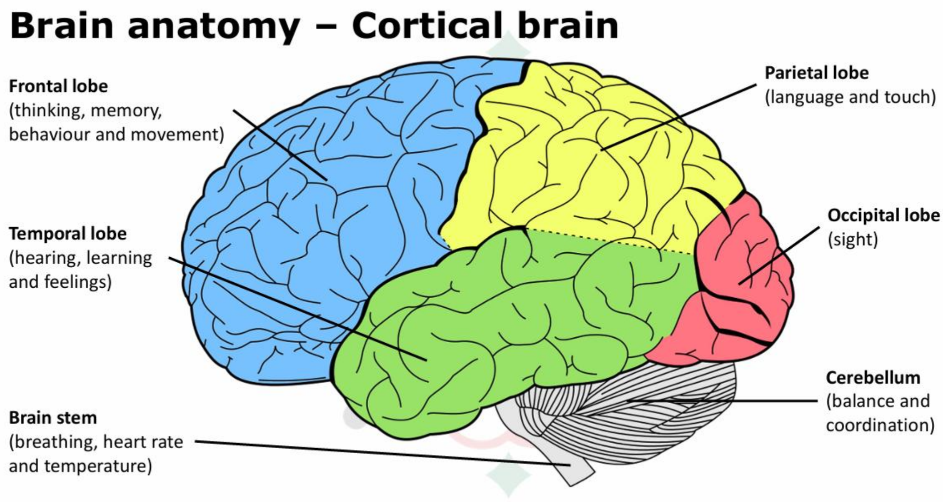


Figure 2.1: Brain anatomy

2.2 NEUROPHYSIOLOGICAL BRAIN SIGNAL RECORDING TECHNIQUES

Electroencephalography (EEG), electrocorticography (ECoG), and magnetoencephalography (MEG) and single-unit recording are prominent neurophysiological techniques used to measure brain activity. EEG captures voltage fluctuations resulting from postsynaptic potentials using electrodes placed on the scalp, offering high temporal but limited spatial resolution due to signal distortion from the skull. ECoG, by contrast, involves placing electrodes directly on the cortical surface, providing superior spatial resolution and signal quality, but at the cost of being invasive. MEG records the magnetic fields generated by neural electrical activity, offering high temporal resolution and better spatial localization than EEG, while remaining non-invasive. However, MEG systems are expensive, immobile, and require magnetically shielded environments, limiting their accessibility. Single-unit recording uses microelectrodes inserted into brain tissue to measure individual neuron activity with exceptional precision, but its invasiveness limits use to animals or specific clinical contexts.

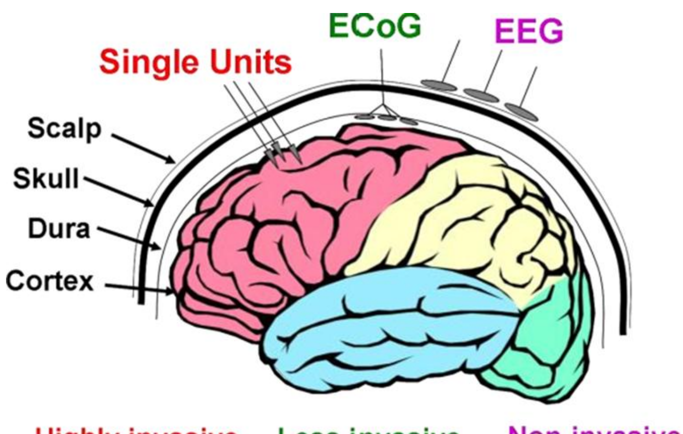


Figure 2.2: brain signal recording techniques

2.3 EEG SIGNAL GENERATION AND COLLECTION

Electroencephalography (EEG), primarily arising from the synchronized firing of large groups of cortical neurons. While individual neurons generate small signals, which EEG cannot measure, a synchronized effort from numerous neurons creates a detectable electrical current. For optimal EEG measurement, specific conditions must be met: the group of neurons must produce an electric current perpendicular to the scalp, fire in parallel, and fire with the same polarity. If these conditions are not met, the signals can cancel each other out, making detection difficult or impossible. It cannot measure most neural activities. However, EEG excels in temporal resolution, offering a precise timeline of brain events, though it affects spatial precision, making it challenging to pinpoint the exact location of activity. Before recording on the scalp, it must Pass through several biological filters: the meninges, skull, and skin. This passage attenuates and spreads the electrical signals, reducing their amplification by the time they reach the electrodes surface. Despite the challenges, EEG remains a valuable tool for understanding brain dynamics, particularly when investigating the timing of neural processes. [88].

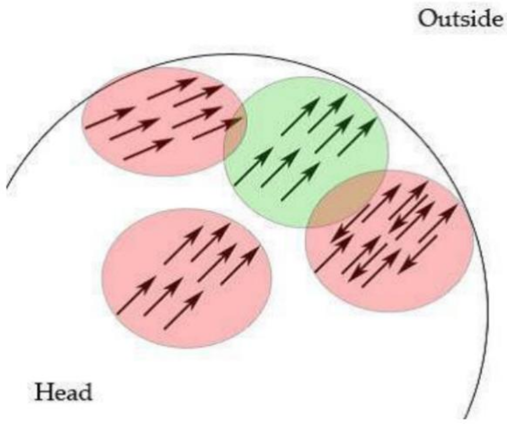


Figure 2.3.1: optimal group of neurons firing

EEG plays a crucial role in Brain-Computer Interface (BCI) systems, particularly those based on the P300 event-related potential, which reflects attention and stimulus recognition [91].

For P300-based BCIs, the parietal and central regions (e.g., Pz, Cz, Fz) are particularly relevant due to their strong involvement in attention and stimulus evaluation, which are key components of the P300 ERP generation [7].

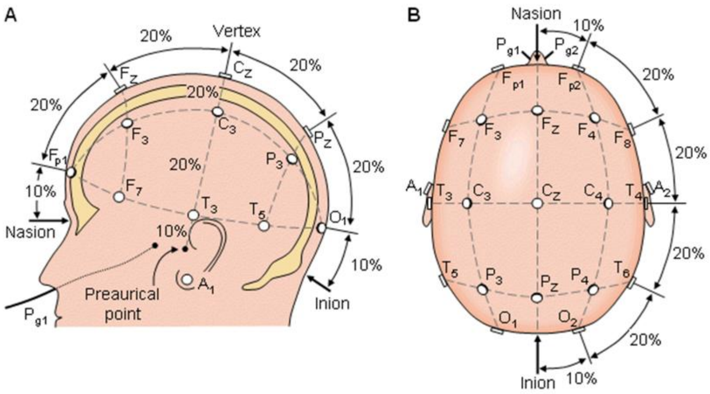


Figure 2.3.2: electrode regions where p300 is most detected

2.4 CHARACTERISTICS OF EEG SIGNALS

EEG signals are complex, non-stationary, and highly susceptible to various artifacts. They represent the summed postsynaptic potentials of large populations of neurons, typically in the microvolt (µV) range. The frequency content of EEG signals is often categorized into distinct ands, each associated with different brain states and functions [8]:

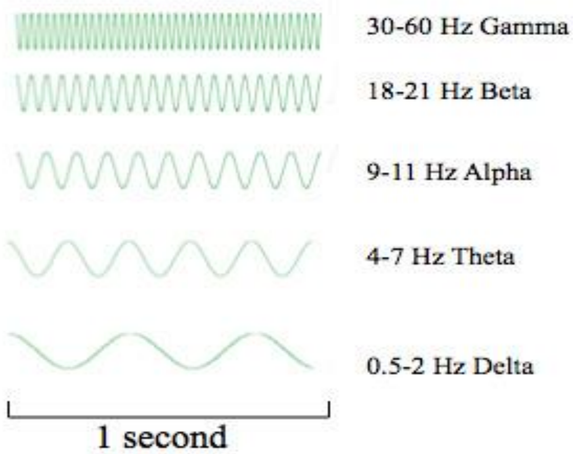


Figure 2.4: spectral analysis of EEG

EEG Frequency Band	Frequency Range (Hz)	Associated Brain States/Functions
Delta	0.5 – 4	Deep sleep, certain pathologies
Theta	4 – 8	Drowsiness, memory, meditation
Alpha	8 – 13	Relaxed wakefulness, eyes closed
Beta	13 – 30	Active thinking, concentration

Gamma	> 30	Perception, learning, memory

Table 2.4: Frequency ranges with associated brain states

2.5 DATASET AND SOURCE

The development and evaluation of robust Brain-Computer Interface (BCI) systems heavily rely on access to high-quality and relevant datasets. For this project, the primary dataset utilized for training and testing the P300 detection and classification algorithms is sourced from the Subject file provided. This dataset is crucial for simulating real-world EEG signal characteristics and validating the performance of the developed algorithms.

While the specific details of the Subject dataset, such as the number of recording sessions, and experimental paradigms, are not explicitly detailed in the provided context, it is assumed to contain raw or pre-processed EEG data suitable for P300-based BCI research. Typically, such datasets include EEG recordings from multiple channels, synchronized with stimulus presentation markers (e.g., target vs. non-target events), and potentially demographic or clinical information about the participants.

The code is designed to process this dataset. It likely contains loading the data, performing further preprocessing steps, extracting P300-related features, and training machine learning models. The presence of this dataset is fundamental for the practical implementation and empirical validation of the BCI system described in this document [9].

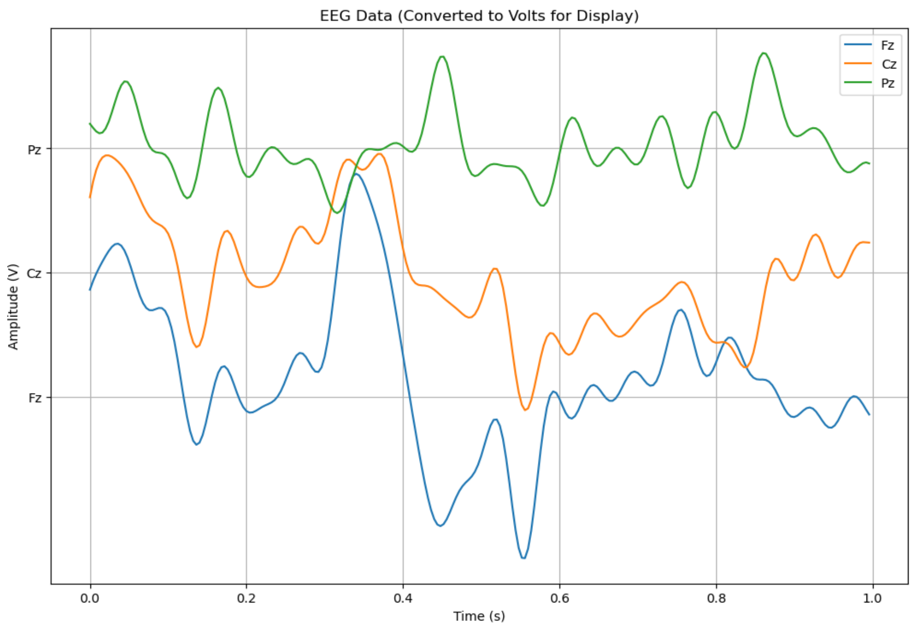


Figure 2.5: ERP waveform where Fz electrode shows the most activity with highest amplitude around 300ms

2.6 EEG ARTIFACTS

EEG signals are highly susceptible to contamination by various non-cerebral electrical potentials, commonly referred to as artifacts. These artifacts can originate from physiological sources within the body or from external environmental factors. Their presence can significantly obscure the underlying brain activity, leading to misinterpretations and reduced accuracy in BCI systems. Therefore, identifying and effectively managing these artifacts is a critical step in EEG signal processing [74].

2.6.1 Physiological Artifacts

Physiological artifacts are generated by biological processes within the subject's body other than brain activity. Common types include:

- **Ocular Artifacts (EOG):** These are generated by eye movements (saccades, blinks) and eye muscle activity. They are typically large in amplitude and primarily affect frontal electrodes but can spread to other areas. Eye blinks produce a characteristic slow wave, while eye movements result in sharp deflections.
- **Muscle Artifacts (EMG):** Generated by muscle contractions (e.g., facial muscles, neck muscles, scalp muscles). EMG artifacts appear as high-frequency, irregular spikes or bursts in the EEG, often localized to electrodes near the active muscles. They can be particularly problematic as their frequency content can overlap with brain activity.
- **Cardiac Artifacts (ECG):** Generated by the electrical activity of the heart. ECG artifacts appear as sharp, rhythmic spikes in the EEG, time-locked to the heartbeat. They are usually most prominent in electrodes close to major blood vessels.
- **Gloss kinetic Artifacts:** Generated by tongue movements. These can produce slow, irregular waves, particularly in electrodes near the mouth.
- **Sweat Artifacts:** Changes in skin conductivity due to sweating can cause slow, irregular drifts in the EEG signal.

2.6.2 External Artifacts

External artifacts originate from the environment or the recording equipment itself:

- **Power Line Interference:** Also known as mains hum, this is a common artifact caused by electromagnetic interference from AC power lines (50 Hz in Europe/Asia, 60 Hz in North America). It appears as a sinusoidal oscillation at the power line frequency and its harmonics.
- **Electrode Artifacts:** Poor electrodes contact with the scalp, dried electrode gel, or movement of electrodes can lead to high impedance, noise, or sudden shifts in the baseline. Bridging between electrodes can also cause issues.
- **Cable Movement Artifacts:** Movement of the EEG cables can induce electrical noise due to triboelectric effects or electromagnetic induction.
- **Equipment Noise:** Malfunctioning amplifiers or other recording equipment can introduce various types of noise into the EEG signal.

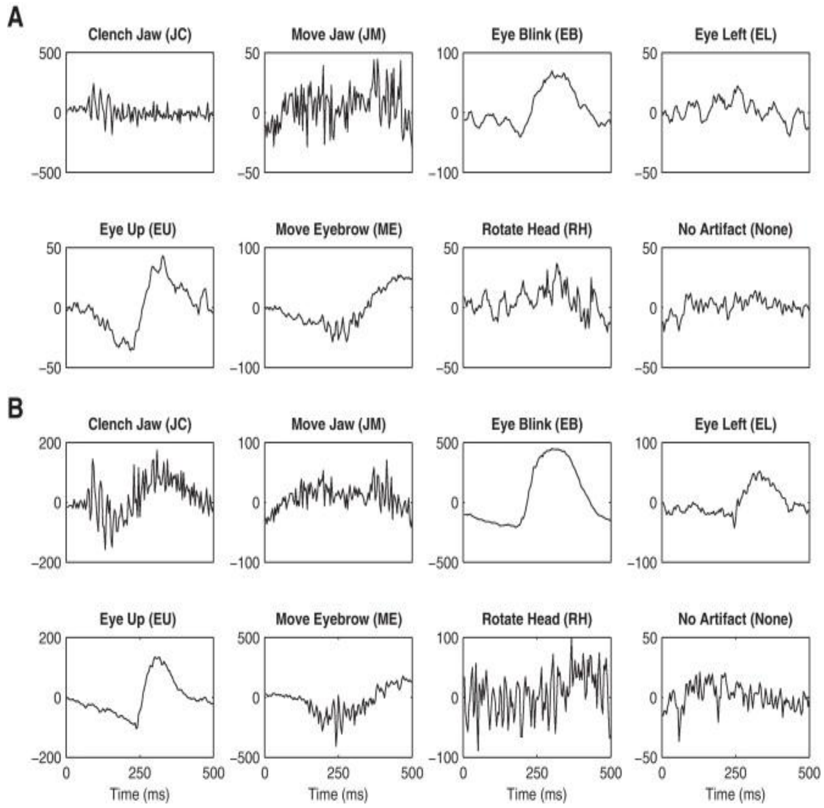


Figure 2.6.2: show types of Physiological Artifacts

2.7 P300 ERP DETECTION MECHANISMS

The P300 event-related potential is a positive-going deflection in the EEG waveform that typically peaks around 300 milliseconds (ms) after the presentation of a rare, task-relevant, or unexpected stimulus. It is a robust and widely studied ERP component, making it a popular choice for non-invasive BCI systems. The P300 is thought to reflect cognitive processes such as attention allocation, context updating, and decision-making. Its amplitude is generally larger for more surprising or significant stimuli, while its latency can vary depending on task difficulty and individual cognitive processing speed.

In a typical P300-based BCI paradigm, often referred to as an oddball paradigm, users are presented with a sequence of stimuli, where one stimulus (the target) is infrequent and task-relevant, while other stimuli (non-targets) are frequent and irrelevant. The user is instructed to attend to the target stimulus. When the target stimulus appears, it elicits a P300 response, which can be detected by analyzing the EEG signals recorded from scalp electrodes, particularly those over the parietal and central regions (e.g., Pz, Cz, Fz) [10].

2.7.1 Detection of the P300 typically involves several steps

1. **Signal Preprocessing:** Raw EEG signals are filtered to remove noise and artifacts. Common filtering techniques include bandpass filtering (e.g., 0.1–30 Hz) to retain the relevant EEG frequencies and notch filtering (e.g., 50 or 60 Hz) to remove power line interference. Artifact removal techniques, such as Independent Component Analysis (ICA) or regression-based methods, may also be applied to mitigate the effects of eye blinks and muscle activity.
2. **Epoching:** The continuous EEG signal is segmented into epochs, which are short time windows time-locked to the presentation of each stimulus. Epochs typically span from a few hundred milliseconds before the stimulus onset to about one second after.
3. **Baseline Correction**: The average amplitude of the EEG signal in a pre-stimulus baseline period (e.g., -200 ms to 0 ms relative to stimulus onset) is subtracted from each epoch to remove any DC offset or slow drifts.
4. **Feature Extraction**: Relevant features are extracted from the epoched EEG data to enhance the P300 response and reduce dimensionality. Common features include the amplitude of the EEG signal at specific time points or within specific time windows (e.g., 250–500 ms post-stimulus), or features derived from spatial filtering techniques like Common Spatial Patterns (CSP).
5. **Classification**: A machine learning classifier is trained to distinguish between epochs containing a P300 response (target stimuli) and epochs without a P300 response (non-target stimuli). Common classifiers used for P300 detection include Linear Discriminant Analysis (LDA), Support Vector Machines (SVM), and, more recently, deep learning models like Convolutional Neural Networks (CNNs).

The accuracy and speed of P300 detection are critical for the performance of BCI systems. Averaging multiple trials corresponding to the same stimulus can improve the signal-to-noise ratio (SNR) of the P300, but this increases the time required to make a selection.

2.8 LITERATURE REVIEW

2.8.1 Signal Processing for BCI Systems

Signal processing plays a pivotal role in BCI systems by transforming raw, noisy EEG signals into meaningful control signals. The primary goals of signal processing in BCIs are to enhance the relevant neural activity, remove artifacts, extract discriminative features, and classify these features to infer the user's intent.

Key signal processing stages in a typical EEG-based BCI include:

- **Spatial Filtering:** Techniques like Laplacian filtering or Common Average Referencing (CAR) are used to enhance the spatial resolution of EEG signals and reduce the influence of widespread noise or activity. Advanced methods like CSP are specifically designed to find spatial filters that maximize the variance between different classes of EEG signals (e.g., P300 vs. non-P300).

- **Temporal Filtering:** As mentioned earlier, bandpass filters are applied to isolate the frequency bands of interest (e.g., delta, theta, alpha, beta, gamma, or specific ERP components like the P300). Notch filters are used to eliminate power line interference.
- **Artifact Handling:** Artifacts are a major challenge in EEG-based BCIs. Techniques for artifact handling can be broadly categorized into rejection methods (discarding contaminated epochs) and removal methods (attempting to subtract the artifactual activity). ICA is a popular blind source separation technique that can identify and remove components corresponding to artifacts like eye blinks or muscle activity. Regression-based methods can also be used if reference channels capturing the artifact (e.g., electrooculogram for eye movements) are available.
- **Feature Extraction:** This stage aims to extract a set of features that effectively represent the neural activity of interest and are discriminative for the classification task. For P300 BCIs, features often include amplitudes at specific latencies, mean amplitudes within time windows, or coefficients from wavelet transforms. For motor imagery BCIs, features might include band power in specific frequency bands (e.g., alpha and beta) over motor cortex areas.
- **Feature Selection/Reduction:** High-dimensional feature spaces can lead to overfitting and increased computational complexity. Feature selection methods aim to identify a subset of the most relevant features, while feature reduction techniques (e.g., Principal Component Analysis - PCA) transform the original features into a lower-dimensional space while preserving most of the relevant information [11][27].

2.8.2 Signal Preprocessing (Filtering, Epoching, Artifact Removal) BCI Systems

Signal processing plays a critical role in extracting meaningful features from noisy EEG data. Typical steps include preprocessing (filtering, artifact removal, epoch extraction), feature extraction (time-domain, frequency-domain, or time-frequency representations), and dimensionality reduction (e.g., Common Spatial Patterns (CSP)) to reduce redundancy and improve classification performance. These processed features serve as input to machine learning models for P300 detection and classification [13].

The signal preprocessing pipeline would involve the application of band-pass filtering, typically within a range such as 0.5-30 Hz, to effectively remove very low-frequency drifts and high-frequency muscle noise, while preserving the P300 component. A notch filter, configured for the local power line frequency (e.g., 50 Hz), would also be crucial to mitigate environmental interference. Epoching would then be performed around hypothetical stimulus onsets, defining time windows such as -200ms to 800ms relative to the event. For artifact removal, advanced techniques like Independent Component Analysis (ICA) or deep learning-based methods, such as LSTM-based autoencoders (LSTEEG), would be employed to effectively mitigate biological and environmental noise without incurring excessive data loss. These choices are justified by their proven efficacy in improving the signal-to-noise ratio and preparing EEG data for reliable ERP extraction [14].

2.8.3 Feature Extraction (Time, P300 ERP Features)

Feature extraction would focus on key P300 characteristics. This includes the extraction of P300 **amplitude**, ideally utilizing a multi-time-point baseline correction method for enhanced robustness, along with its **latency**. The **median** of time-domain signals would also be extracted from the epoched data to provide a comprehensive representation of the neural response. These features are chosen due to their established relevance for P300 ERP analysis in the literature, with amplitude and latency directly quantifying the P300 component, and median providing a robust measure of central tendency in the time domain. Frequency domain features were not selected for this implementation to maintain focus on the specified features [15].

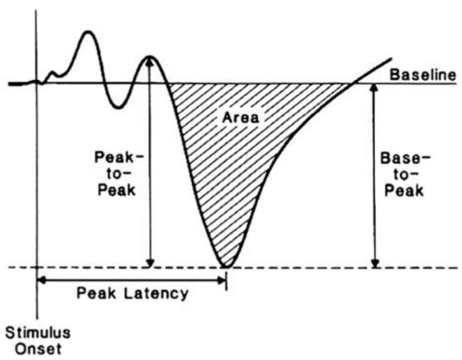


Figure 2.8.3: P300 Waveform with Key Features

2.8.4 Feature Selection Methods

A combined strategy for feature selection would be beneficial. Filter-based methods, such as Mutual Information, would be employed to identify features that exhibit a high degree of shared information with the target class, effectively ranking features by their statistical relevance. Concurrently, spatial filtering techniques, such as Regularized Common Spatial Pattern (RCSP), would be utilized to enhance the separability between the target and non-target classes by optimizing the spatial representation of the EEG signals. This hybrid approach ensures that the features fed into the subsequent classification algorithm are maximally discriminative, leading to improved accuracy and robustness in P300 detection [16][27].

2.8.5 Classification Algorithms (Lda, Rf, Hybrid Model)

The selection of a classification algorithm is driven by specific performance and interpretability requirements. Linear Discriminant Analysis (LDA) is a simple yet effective linear classifier that has been widely used in P300 BCIs and motor imagery BCIs. It is a simple yet robust algorithm that aims to find a linear combination of features that best separates two or more classes. LDA assumes that the data for each class is normally distributed and that the covariance matrices of the classes are equal. Despite these simplifying assumptions, LDA often performs remarkably well in P300 BCIs due to the relatively linear separability of P300 features in many cases. Its computational efficiency makes it suitable for real-time BCI systems.

Support Vector Machines (SVM) are powerful classifiers that can find a non-linear decision boundary by mapping the features into a higher-dimensional space using a kernel function (e.g., linear, polynomial, radial basis function). SVMs are known for their good generalization performance, especially when the number of features is large. For higher classification accuracy, particularly with extensive and complex datasets, Random Forest (RF) presents a robust alternative due to its ability to handle high-dimensional feature spaces and model non-linear relationships, coupled with its parallelization capabilities for efficient processing. For state-of-the-art performance and automated feature learning, a hybrid deep learning model (e.g., a CNN-Transformer framework or PCANet-based fusion) would be the preferred choice, aligning with current trends towards end-to-end learning in BCI development for superior accuracy and robustness [17].

2.7.6 Evaluation Metrics (Latency, ROC-AUC)

The primary evaluation metrics would be **Latency** and **Area Under the Curve (AUC)**. Latency, encompassing ADC, processing, and output delays, is critical for ensuring the real-time responsiveness and usability of the BCI system, as even highly accurate classification can be rendered impractical by significant delays. The Area Under the

Curve (AUC) from the Receiver Operating Characteristic (ROC) curve would be used to summarize the classifier's average performance across all possible classification thresholds. AUC is particularly valuable because it is insensitive to class imbalance, making it a robust measure for evaluating models in P300 BCI where target stimuli are inherently rare, signifying a robust classifier capable of reliably detecting P300s across various operating points.

3 CHAPTER 3: PROJECT TERMINOLOGY AND DATASET

3.1 PROJECT SCOPE

Project scope defines the boundaries and objectives of a project, outlining what will and will not be included. For this BCI system development, the project scope encompasses the design, implementation, and evaluation of a P300-based BCI for handicapped users. This includes the development of signal acquisition protocols, preprocessing algorithms, P300 detection and classification modules, and the integration with specific applications such as educational websites and BCI-controlled games (UNO and Maze). Excluded from this scope are the development of the EEG hardware itself, clinical trials for medical device certification, and long-term deployment and maintenance strategies beyond the prototype phase. The focus is on demonstrating the feasibility and usability of the BCI system within a controlled environment.

3.2 MILESTONE

Milestones are significant points or achievements in a project timeline, marking the completion of a major phase or a critical deliverable. They serve as progress markers and are essential for project tracking and management. Key milestones for this BCI project include:

- **Completion of EEG Signal Acquisition Protocol:** Defining and finalizing the methodology for acquiring EEG data, including electrode placement, experimental paradigms, and data recording procedures.
- **Development of P300 Detection Module:** Successful implementation and initial testing of algorithms capable of accurately identifying P300 ERPs from raw EEG signals.
- **Integration with Educational Interface:** Successful connection and functional testing of the BCI system with the basic speller website and advanced learning platform.
- **Development of BCI-Controlled Games:** Completion of the UNO and Maze games, ensuring they are fully controllable via the BCI system.
- **System Integration and Initial Testing:** Combining all developed modules and applications into a cohesive system and conducting preliminary usability and performance tests.
- **Final Documentation Submission:** Completion and submission of the comprehensive project report, adhering to ABET documentation standards.

3.3 GANTT CHARTT

A Gantt chart is a visual representation of a project schedule, showing the start and end dates of various activities, tasks, and milestones. It is a fundamental tool for project planning, coordination, and tracking progress. A typical Gantt chart for this project would illustrate the sequential and parallel nature of tasks, such as:

Each task would have a defined duration, dependencies on other tasks, and assigned resources, all visually represented on the chart to provide a clear overview of the project timeline and critical path.

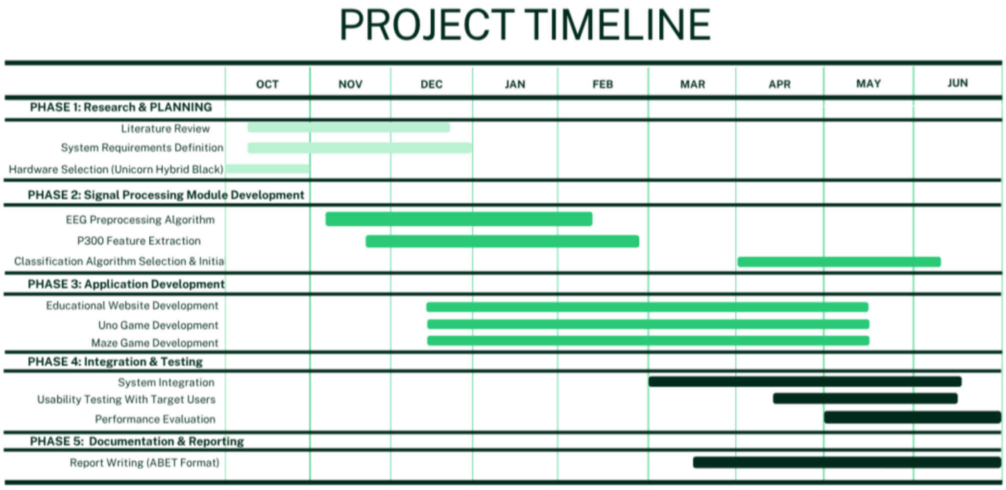


Figure 3.3: GANTT CHART

4 CHAPTER 4: PROPOSED SYSTEM MODEL

4.1 SYSTEM OBJECTIVES AND FUNCTIONAL REQUIREMENTS

The primary objective of the proposed system is to establish a robust and reliable connection between external applications (games and websites) and the Unicorn Hybrid Black headset. This connection facilitates the transmission of event-specific triggers from the applications to the headset, enabling synchronized data recording and analysis. The system is designed to be versatile, supporting integration with various applications developed in different environments, such as Unity-based games and web applications.

4.1.1 FUNCTIONAL REQUIREMENTS OF THE SYSTEM INCLUDE

- **Real-time Trigger Transmission:** The system must be capable of sending triggers from the application to the Unicorn Hybrid Black headset in real-time or near real-time. These triggers correspond to specific events within the application, such as button clicks or game state changes.

- **UDP Communication:** The system utilizes the User Datagram Protocol (UDP) for communication between the application and the headset. UDP is chosen for its low-latency characteristics, which are crucial for time-sensitive BCI applications [20].

- **Flask-based Server:** A Flask-based server acts as an intermediary, receiving messages from the applications and forwarding triggers to the headset. The server listens for HTTP POST requests from the applications and sends UDP packets to a predefined IP address and port (127.0.0.1:1000) where the Unicorn Hybrid Black headset's software is expected to be listening [19].

- **Message Parsing:** The server is responsible for parsing incoming messages from the applications to extract relevant information, such as button names and timestamps. Regular expressions are employed for robust parsing of message strings.

- **Application Launch and Management:** For game integrations, the system includes functionality to launch the external game executable. Threading is used to run the Flask server and the game application concurrently.

- **Cross-Origin Resource Sharing (CORS) Support:** For web application integration, the Flask server is configured with CORS support to allow requests from different origins, enabling seamless communication between the website frontend and the backend server.

- **Extensibility:** The system is designed to be extensible, allowing for the integration of new applications and the definition of new trigger mappings with relative ease.

4.2 HARDWARE ARCHITECTURE OVERVIEW

The core hardware component of this proposed system is the **Unicorn Hybrid Black Headset**, an 8-channel wireless EEG (Electroencephalography) device developed by g.tec. This headset is designed for brain-computer interface (BCI) applications, enabling the acquisition of brainwave data. Its key features relevant to this system include [20]:

- **8 EEG Channels:** The headset is equipped with eight dry EEG electrodes, allowing for the capture of brain activity from multiple locations on the scalp. This multichannel capability provides a richer dataset for BCI applications.

- **Wireless Connectivity:** The Unicorn Hybrid Black headset communicates wirelessly with a computer, typically via Bluetooth. This wireless design offers freedom of movement and reduces clutter, making it suitable for various research and application scenarios.

- **High Sampling Rate and Resolution:** The headset samples EEG data at a high rate (e.g., 250 Hz per channel) and with high resolution (24-bit), ensuring the capture of detailed and accurate brainwave signals.

- **Integrated Accelerometer:** In addition to EEG, the headset often includes an integrated accelerometer, which can provide data on head movements. This information can be valuable for motion artifact removal and for understanding user behavior.

- **Trigger Input Capability:** While the headset primarily outputs EEG data, its accompanying software (Unicorn Suite) is capable of receiving external triggers. In this proposed system, these triggers are numerical values sent via UDP, which can be synchronized with the EEG data stream. This synchronization is crucial for event-related potential (ERP) studies and for correlating brain activity with specific in-application events [18].

In the context of this system, the Unicorn Hybrid Black Headset serves as the primary data acquisition unit. It receives event markers (triggers) from the software applications, allowing for precise temporal alignment of application events with recorded brain activity. This setup enables researchers and developers to investigate brain responses to specific stimuli or actions within games and websites [5][18].

Figure 4.2: Unicorn Hybrid Black Headset

4.3 SIGNAL ACQUISITION SETUP (UNICORN HYBRID BLACK HEADSET)

The signal acquisition setup primarily revolves around the proper placement and operation of the Unicorn Hybrid Black Headset. The headset is designed for ease of use, with dry electrodes that do not require conductive gel, simplifying the setup process. The typical setup involves the following steps:

1. **Headset Placement:** The Unicorn Hybrid Black Headset is placed on the user's head, ensuring that the eight EEG electrodes make good contact with the scalp. The specific placement of the electrodes (e.g., Fz, C3, Cz, C4, Pz, PO7, PO8, Oz) corresponds to the international 10-20 system, allowing for standardized recording of brain activity. Proper contact is crucial for minimizing impedance and acquiring high-quality EEG signals.

2. **Wireless Connection:** The headset establishes a wireless connection (typically Bluetooth) with a computer running the Unicorn Suite software. This software acts as the interface between the headset and the system, managing data acquisition and streaming.

3. **Unicorn Suite Software Configuration:** Within the Unicorn Suite, various parameters can be configured, including the sampling rate, filter settings, and data streaming options. For this proposed system, the crucial configuration involves enabling the UDP streaming of triggers. The Unicorn Suite is configured to listen for incoming UDP packets on a specific IP address and port (127.0.0.1:1000 in this case), and to associate these numerical triggers with the ongoing EEG data stream.

4. **Electrode Quality Check:** Before commencing data acquisition, it is essential to verify the quality of the electrode contact. The Unicorn Suite typically provides real-time impedance feedback for each electrode, allowing the user to adjust the headset

placement to ensure optimal signal quality. Low impedance values indicate good contact and reliable signal acquisition.

5. **Data Streaming:** Once the headset is properly set up and configured, the Unicorn Suite begins streaming the EEG data and any received triggers. This data can then be accessed by other applications or analysis tools for further processing.

The integration of the Unicorn Hybrid Black Headset into the system allows for the precise synchronization of external events (from games or websites) with the recorded brain activity. The numerical triggers sent from the applications are embedded within the EEG data stream by the Unicorn Suite, providing a time-locked marker for analysis. This capability is fundamental for conducting event-related brain potential (ERP) studies and for developing BCI applications that respond to specific user interactions within the digital environment [5][18].

4.4 SOFTWARE ARCHITECTURE AND SIGNAL PROCESSING PIPELINE

The software architecture of the proposed system is designed to facilitate seamless communication between interactive applications (games and websites) and the Unicorn Hybrid Black Headset. It primarily consists of a Flask-based web server [4] acting as a central hub for message processing and trigger dissemination. The signal processing pipeline, in this context, refers to the flow of information from user interaction within the application to the eventual transmission of a numerical trigger to the headset.

4.4.1 Core Components

1. **Flask Web Server:** Each application (UNOgame, mazegame, and Website) utilizes a Flask web server, running locally on 127.0.0.1 at port 5000 . This server is responsible for:

- **Receiving Messages:** It exposes a /message endpoint that accepts HTTP POST requests containing JSON payloads. These payloads are expected to have a message field, which encapsulates information about user interactions.

- **Message Parsing:** Upon receiving a message, the server employs regular expressions to extract critical information, specifically the Button name and a Time stamp. This parsing ensures that the system can identify the specific user action that occurred.

- **Trigger Mapping:** Extracted button names are mapped to predefined numerical trigger values. For instance, in the UNO game, "RightButton" maps to 1 , "LeftButton" to 2 , "PlaceCard" to 3 , and "DrawCard" to 0 . Similarly, the Maze game maps "ButtonA" to 1 , "ButtonB" to 2 , "ButtonR" to 3 , and "ButtonE" to 0 . The Website application has a more extensive mapping for various web-based interactions (e.g., "open-button" to 1 , "next-button" to 2 , etc.). This mapping is crucial for standardizing the triggers sent to the Unicorn Hybrid Black Headset.

- **UDP Trigger Transmission:** Once a numerical trigger is determined, the server sends this value as a UDP packet to 127.0.0.1 on port 1000 . This is the designated endpoint where the Unicorn Hybrid Black Headset's accompanying software (Unicorn Suite) is configured to listen for external triggers.

2. **External Application Integration:**

- **Game Launchers (UNOgame.py, mazegame.py):** These scripts include functionality to launch external game executables using subprocess.Popen() . This ensures that the game and the Flask server can run concurrently, allowing for real-time interaction and trigger generation.

- **Website Launcher (Website.py):** The Website.py script utilizes the webbrowser module to open a specified HTML file in the default web browser. This enables the web application to interact with the Flask server for trigger generation.

3. **Threading:** To maintain responsiveness and allow for simultaneous operation of the Flask server and the external application (game or website), Python's threading module is employed. The Flask server runs in a separate thread, ensuring that it can continuously listen for incoming messages without blocking the main application flow.

4.4.2 Signal Processing Pipeline

The signal processing pipeline within this software architecture is conceptualized as follows:

1. **User Interaction:** A user interacts with the game or website, performing actions such as clicking a button, making a selection, or triggering an in-game event.

2. **Message Generation:** The interactive application (game or website) generates a message, typically in a string format containing information about the event (e.g., "Button: RightButton, Time: 12345"). This message is then sent as a JSON payload via an HTTP POST request to the Flask server's /message endpoint.

3. **Message Reception and Parsing:** The Flask server receives the HTTP POST request. It then parses the message string using regular expressions to extract the Button name and Time stamp. This step effectively converts raw event data into structured information.

4. **Trigger Assignment:** Based on the extracted Button name, the server consults its internal mapping to assign a corresponding numerical trigger value. This is a critical step for standardizing the event markers for the Unicorn Hybrid Black Headset.

5. **UDP Transmission:** The assigned numerical trigger is then encoded as a string and sent as a UDP packet to the Unicorn Hybrid Black Headset's listening port (127.0.0.1:1000). This low-latency transmission ensures that the trigger arrives at the headset with minimal delay.

6. **Headset Integration:** The Unicorn Hybrid Black Headset's software (Unicorn Suite) receives the UDP packet. It then integrates this numerical trigger into the ongoing EEG data stream, effectively marking the precise moment of the user's interaction within the recorded brain activity. This allows for post-hoc analysis of brain responses time-locked to specific events.

This pipeline ensures a clear and efficient flow of event information from the user's interaction point to the EEG data acquisition system, enabling comprehensive analysis of brain-computer interactions.

4.5 SYSTEM INTEGRATION AND DATA FLOW

The system integration and data flow describe how the various components—the interactive applications (games/websites), the Python Flask servers, and the Unicorn Hybrid Black Headset—interact to form a cohesive Brain-Computer Interface (BCI) system. The overall data flow is designed to be efficient, ensuring that event markers from user interactions are accurately and promptly synchronized with the EEG data acquired by the headset.

4.5.1 Integration Points

1. **Application-to-Server Integration:**

- **Games (UNOgame.py, mazegame.py):** These Python scripts act as launchers for their respective external game executables. Crucially, they also initiate a Flask server that listens for HTTP POST requests. The games themselves are presumed to have internal mechanisms (e.g., Unity scripts, as suggested by the Unity messages as JSON comment in the code) to send event data (button presses, game state changes) to the local Flask server at (http://127.0.0.1:5000/message).

- **Website (Website.py):** This script launches a local HTML file in a web browser and also starts a Flask server. The web application (HTML/JavaScript) running in the browser is responsible for sending event data (e.g., button clicks on the webpage) via AJAX or Fetch API requests to the same local Flask server endpoint (http://127.0.0.1:5000/message). The CORS (app) in Website.py is essential here to allow cross-origin requests from the web page to the Flask server.

2. **Server-to-Headset Integration:**

- The Flask server, upon receiving and processing an event message from the application, translates the event into a numerical trigger. This trigger is then 6. 1. ◦ ◦ 2. ◦ sent as a UDP packet to the Unicorn Hybrid Black Headset. The destination for these UDP packets is consistently 127.0.0.1 on port 1000. This implies that the Unicorn Suite software, which manages the headset, is running on the same machine and is configured to listen for incoming UDP triggers on this specific port.

4.5.2 Data Flow Diagram

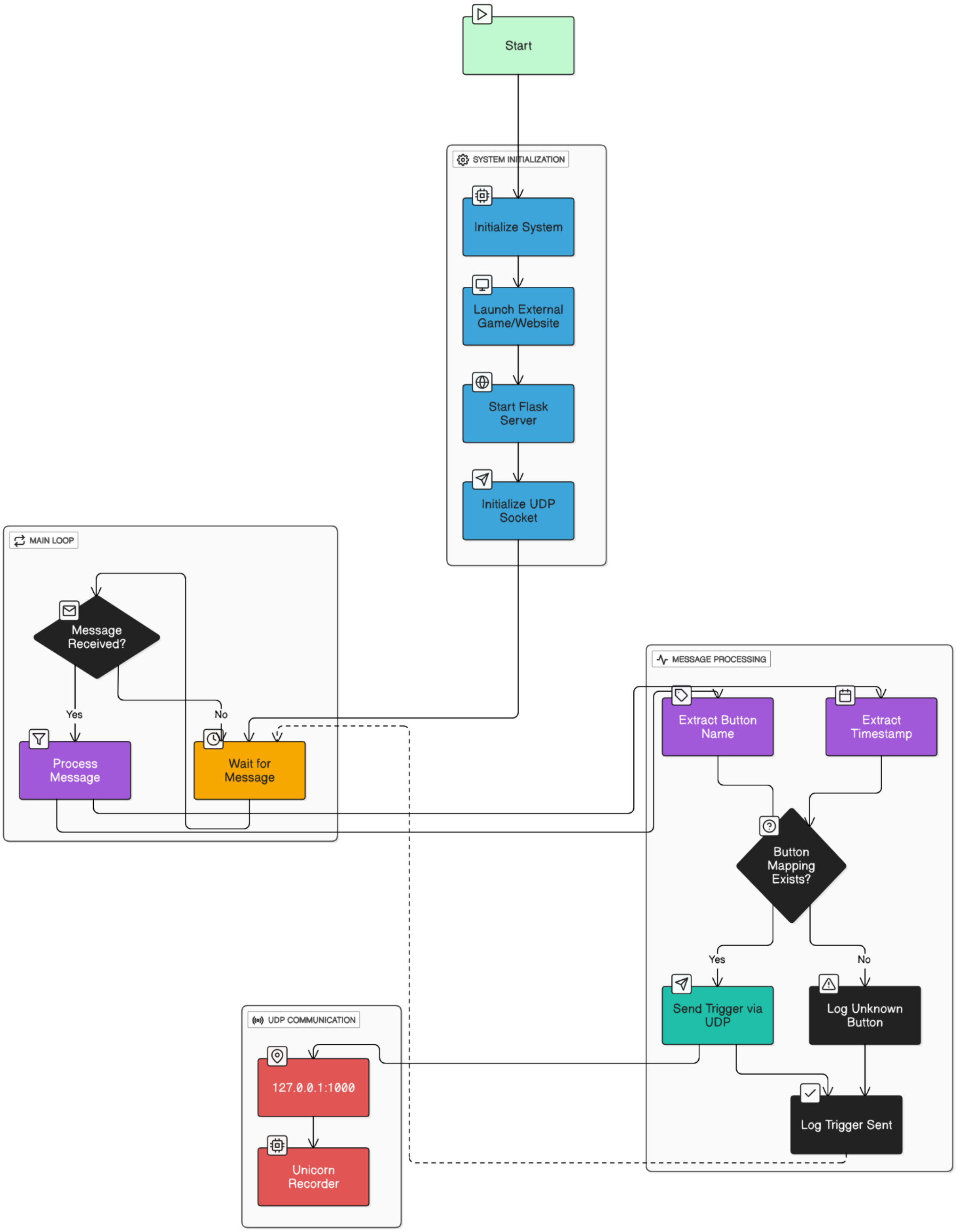


Figure 4.5.2: Offline System

4.5.3 Detailed Data Flow Steps

1. **System Initialization:** This phase sets up the entire environment. It involves initializing the system by loading configurations and allocating resources, launching the external game or website (using subprocess for games or web browser for websites), starting a Flask server to act as an HTTP endpoint for incoming messages, and finally, initializing a UDP socket for sending trigger signals to the Unicorn Recorder. This ensures all components are ready for operation.

2. **Main Loop:** This represents the continuous operational cycle. The system constantly checks if a message has been received from the external application. If a message is present, it proceeds to process it; otherwise, it enters an efficient waiting state to conserve resources until a new message arrives. This loop ensures continuous responsiveness to user interactions.

3. **Message Processing:** When a message is received, this phase focuses on interpreting and preparing it for the Unicorn Recorder. It involves extracting the button name and timestamp from the message payload. A critical step is checking if a predefined mapping exists for the extracted button name to a numerical trigger value. If a mapping exists, the corresponding trigger is sent via UDP. If no mapping is found, the unknown button is logged for debugging. Regardless of whether a trigger was sent, the system logs the outcome of the message processing, providing an audit trail of all events.

4. **UDP Communication:** This is the final stage where the processed trigger signals are transmitted. The system sends the numerical trigger value as a UDP packet to a specific local endpoint (127.0.0.1:1000). The Unicorn Recorder software, listening on this port, receives the packet and integrates the trigger event with the simultaneously recorded EEG data, allowing for synchronized timestamping of events within the brain activity stream. This closes the loop between application events and physiological data acquisition.

This integrated system provides a robust framework for capturing and analyzing brain activity in response to interactive digital environments, paving the way for advanced BCI research and applications.

4.6 ONLINE ANALYSIS COMPONENT

This section details the specific architecture and data flow for the online analysis component, which enables an external AI to control the UNO game. This component operates independently of the user-driven interaction pathways, providing a dedicated channel for AI-driven commands and their integration into the system.

4.6.1 Data Flow Diagram

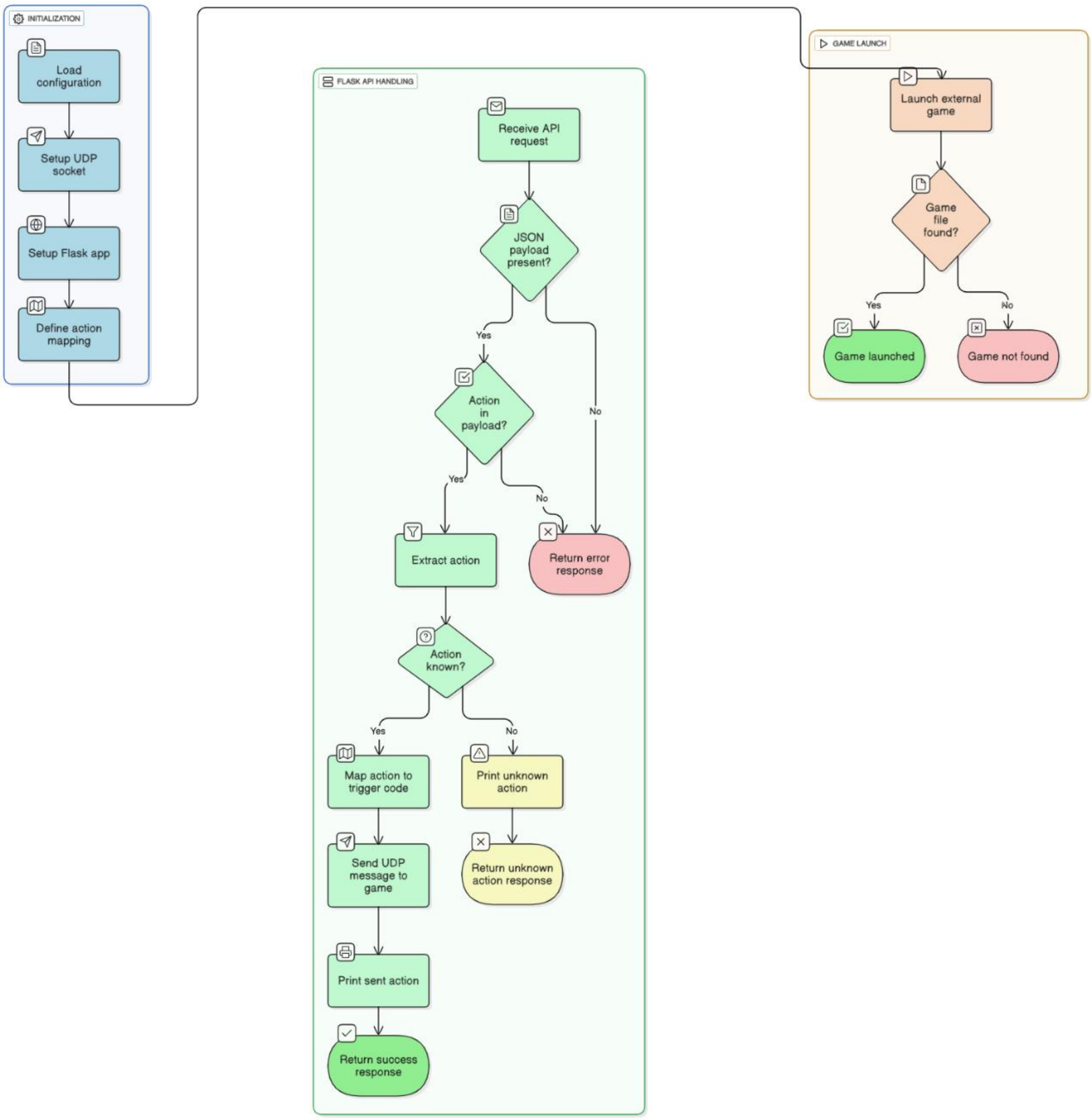


Figure 4.6.1: Online System.

4.6.2 Detailed Data Flow Steps

- **Initialization Phase:** This phase sets up the entire environment. It involves defining the game path, UDP port, and UDP host for communication. A Flask application is initialized to handle incoming API requests, and a UDP socket is created and configured with the specified endpoint. Crucially, a trigger map is defined, which translates specific game actions (like 'Right Button', 'Left Button', 'Place Card', 'Draw Card') into numerical codes that the Unicorn Recorder understands.

- **Game Launch Phase:** This phase focuses on launching the external game. The system attempts to execute the game executable. Based on the success or failure of this launch, a confirmation or error message is printed. If the launch fails, the program exits to prevent further issues.

- **API Routes Logic:** This section details how the Flask server processes incoming requests from the game. A dedicated API route is defined to receive AI commands. Upon receiving a POST request, the system extracts JSON data and checks if a valid 'action' exists within it. If the action is missing or unknown (not found in the trigger map), an appropriate error is returned. If the action is found and mapped, the corresponding numerical UDP code is sent to the Unicorn Recorder. The system then logs the action and returns a success response to the game.

- **Main Execution Phase:** This phase orchestrates the overall system operation. It calls the game launch process and starts the Flask server in the background, allowing the main thread to enter an idle loop. This loop continuously waits for a keyboard interrupt (e.g., Ctrl+C) to gracefully shut down the program. Upon detection of the interrupt, a shutdown message is printed, and the program terminates, ensuring all resources are properly released.

This detailed flow illustrates how the AI can seamlessly interact with and control the UNO game, providing a robust framework for online analysis and AI development within the BCI system.

5 CHAPTER 5: SIMULATION AND PERFORMANCE EVALUATION

5.1 DISCUSSION OF RESULTS AND OPTIMIZATION STRATEGIES

For the implementation of EEG signal processing, the MNE-Python software package is a widely utilized open-source tool. It provides comprehensive functionalities for electrophysiology research, including data loading, preprocessing, visualization, and analysis of EEG and MEG data, facilitating robust and reproducible scientific workflows. These processed features serve as input to machine learning models for P300 detection and classification [28].

5.1.1 Proposed P300 ERP Analysis Pipeline Flowchart

The following flowchart outlines the typical steps involved in an EEG-based P300 ERP analysis pipeline, as inferred from the provided data characteristics and academic literature:

1. **Raw EEG Data Input**
- Continuous multi-channel EEG signals (e.g., from the provided 8-channel dataset).

2. **Signal Preprocessing**
- Filtering: Application of band-pass filters (e.g., 0.5-30 Hz) to remove drifts and muscle noise, and notch filters (e.g., 50/60 Hz) for power line interference. Common Mode Reference (CMR) for noise reduction.
- Epoching: Segmentation of continuous EEG into time-locked epochs around hypothetical stimulus onsets (e.g.,0ms to 800ms).
- Artifact Removal: Mitigation of biological (eye blinks, muscle activity) and environmental artifacts using techniques like Independent Component Analysis (ICA)

3. **Feature Extraction**
- Time Domain: Calculation of the Median value within defined epochs to remove outliers.
- P300 ERP Specific: Extraction of P300 peak Amplitude (with multi-time-point baseline correction), Area under the curve (auc) (helps detecting high amplitude artifacts and measure magnitude of ERP over specific time window) and Latency (time to peak).

4. **Feature Selection**
- Filter-based Methods: Application of statistical measures (e.g., Mutual Information) to rank features by relevance [27].

5. **Classification**
- Linear Discriminant Analysis (LDA): A computationally efficient linear classifier.
- Random Forest (RF): An ensemble method robust to high-dimensional data and non-linear relationships.
- Support Vector Machine (SVM): A powerful classifier effective in high-dimensional spaces, capable of handling linear and non-linear decision boundaries using kernel functions.

- Hybrid Models (Fusion): Advanced approaches combining different techniques, for automated feature learning and superior performance.

6. **Performance Evaluation**
   - Confusion Matrix: Foundation for per-class evaluation (TP, TN, FP, FN).
   - Precision & Recall: Critical for imbalanced data; precision avoids false positives, recall minimizes false negatives.
   - F1-Score: Unified metric balancing precision and recall (harmonic mean).
   - Accuracy: Overall correctness, meaningful only for balanced classes.
   - Area Under the Curve (AUC)**:** Threshold-independent performance (ROC curve integration).
   - Latency & Jitter**:** Real-time system delays (ADC-to-output).

7. **Data Visualization & Dimensionality Reduction**

- **t-Distributed Stochastic Neighbor Embedding (t-SNE):**
  - Projects high-dimensional data into 2D/3D space by preserving pairwise similarities via probabilistic modeling.
  - **Mechanism**: Converts high-dimensional Euclidean distances into joint probabilities, then minimizes Kullback-Leibler (KL) divergence between these probabilities in low-dimensional embedding.
  - **Non-Convex Optimization**: Cost function lacks convexity; results vary with initialization (requires multiple runs for consistency).
  - **Use Case**: Exploratory data analysis, especially for pattern discovery in complex datasets (EEG features).

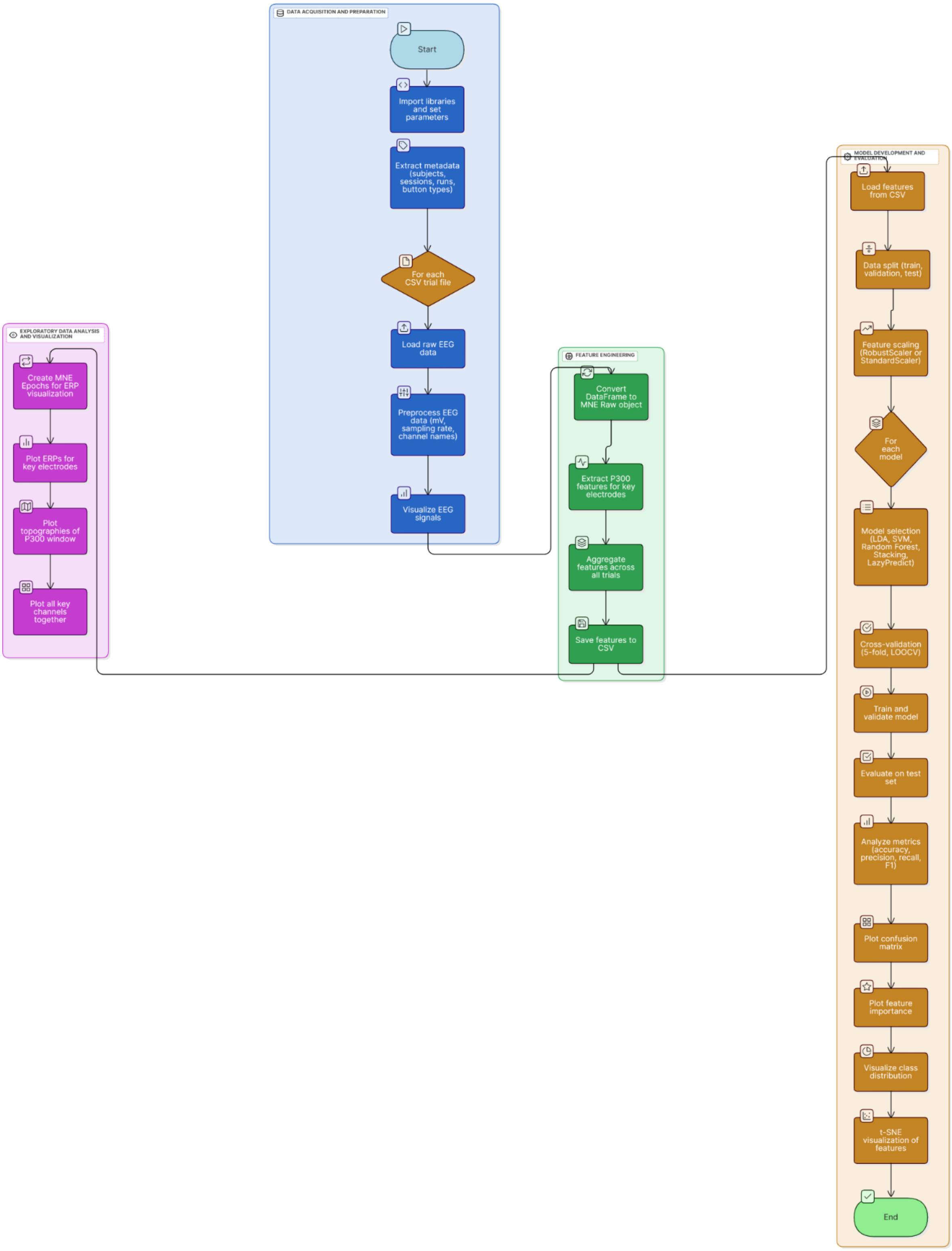


Figure 5.1: flowchart of the preprocessing, feature extraction and model developing code.

5.1.2 phases and Challenges in P300 BCI Development

To organize and clarify the development and testing of the proposed BCI system using the Unicorn Hybrid Black headset, the work has been divided into four distinct experimental phases. These phases reflect the progression from initial raw data collection to model optimization through augmentation, fusion, and re-evaluation.

Phase 1: Low Data This foundational phase focused on acquiring raw EEG data using the Unicorn Hybrid Black Headset.

- EEG signals were recorded via the Unicorn Suite, synchronized with triggers sent from external applications.
- Applications included: UNO game, Maze game, and a Website interface.
- Flask-based servers received user interaction messages (e.g., button presses) and forwarded numerical triggers over UDP.

Phase 2: Data Augmentation, Upsampling and **Cross validation** The second phase addressed the class imbalance and limited volume of usable EEG samples.

- **Data Augmentation Techniques Used:**
    - Time-warping, jittering, and cropping for temporal distortion.
- **Upsampling Techniques:**
    - SMOTE (Synthetic Minority Over-sampling Technique).
    - Random oversampling to balance class distributions.
- **Cross validation:**
    - Leave-One-Out Cross-Validation (LOOCV) was applied post-upsampling to ensure robust evaluation on small datasets.
    - Each trial was held out in turn while training occurred on the remaining data, minimizing data leakage and overfitting.
    - LOOCV provided high-resolution performance estimates, especially valuable for datasets with limited subject or trial counts.
- **Libraries Utilized:**
    - imbalanced-learn
    - numpy, scikit-learn, scipy
- This phase significantly improved the robustness of training data and helped mitigate overfitting.

Phase 3: Fusion of Models This phase aimed to enhance accuracy by combining multiple classifiers.

- **Models Used:**
    - Linear Discriminant Analysis (LDA)
    - Support Vector Machine (SVM)
    - Random Forest (RF)
- **Fusion Strategy:**
    - Majority voting and weighted average approaches were tested.
    - Performance was compared using AUC, latency, and classification accuracy.
- Integration with real-time UDP trigger handling allowed closed-loop evaluations.

Phase 4: Final Recording and Re-Experimentation In this culminating phase, the full cycle was repeated using enhanced and optimized components from Phases 1–3.

- EEG data was re-recorded with the improved trigger setup.
- Augmentation and fusion techniques were reapplied to new datasets.
- Final validation was conducted using:
    - Online system for AI-based UNO control.
    - Offline ERP data for traditional signal analysis.
- The best-performing configuration was identified and documented based on AUC and user responsiveness.

Additionally, we've increased the number of trials and made a bigger dataset to further strengthen the reliability of our results.

5.2 REAL-TIME IMPLEMENTATION WITH PYSL

To achieve real-time EEG signal acquisition and classification, the **Python pylsl library** (Lab Streaming Layer) was utilized to stream data directly from the EEG headset to the AI model [13].

The system architecture included the following stages:

- Live signal acquisition through **StreamInlet** using **pylsl.**
- Online preprocessing using a **Butterworth band-pass filter** (0.5–30 Hz).
- Real-time feature extraction, focusing on **peak amplitude and area under the curve (AUC)** within a defined time window.
- Classification using a pre-trained **LDA model**.

- Transmission of predicted class labels to a **Flask-SocketIO** server for UI response and logging.

Code Workflow Summary:

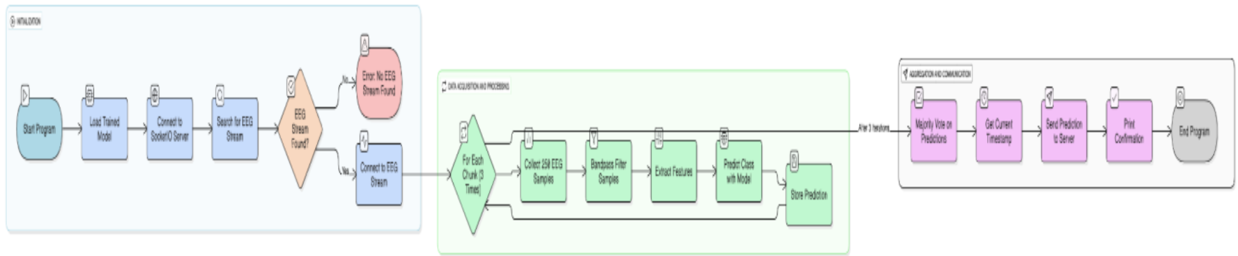


Figure 5.2: REAL-TIME IMPLEMENTATION WITH PYSL Workflow

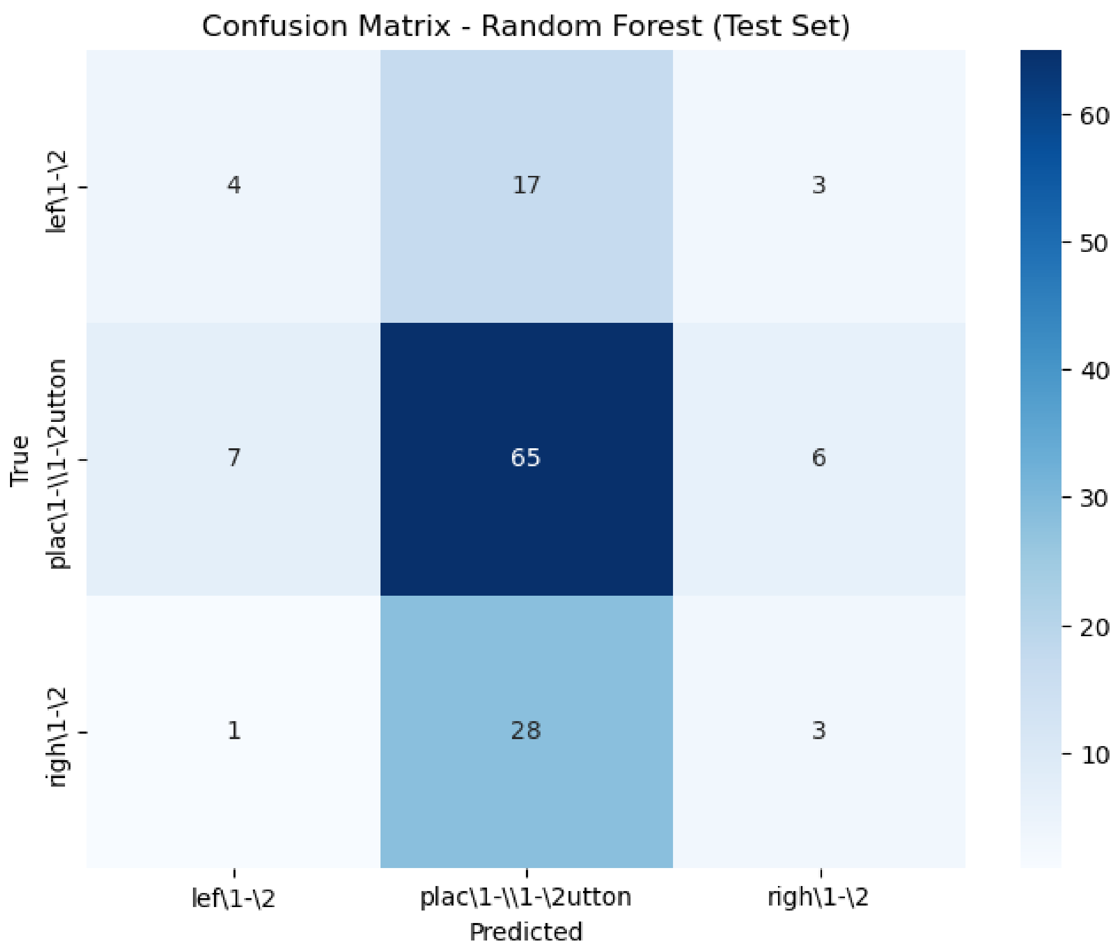
5.3 PERFORMANCE VISUALIZATION AND METRICS ANALYSIS

A range of evaluation techniques was applied to assess the P300 classification pipeline.

5.3.1 Confusion Matrix

A confusion matrix was used to analyze prediction accuracy per class:

- **True Positives (TP)**: Correctly predicted target events.
- **True Negatives (TN)**: Correctly predicted non-targets.
- **False Positives (FP)**: Non-targets wrongly identified as targets.
- **False Negatives (FN)**: Missed detection of real targets.



Interpretation:
High TP and TN values indicate good performance. High FN values require better feature extraction or threshold tuning.

5.3.2 Accuracy Score

Overall model accuracy was computed as:

Accuracy = (TP + TN) / (TP + TN + FP + FN)

- A high accuracy (e.g., >85%) demonstrates a well-performing model.
- Note: For imbalanced datasets, AUC is more reliable.

Model accuracy = 54%

5.3.3 Precision, Recall, and F1-Score

To complement accuracy and provide better insight into class-wise performance—especially in imbalanced datasets—the following metrics were calculated:

- **Precision** = TP / (TP + FP)
  Measures how many predicted targets were actually correct. High precision = low false positives.
- **Recall (Sensitivity)** = TP / (TP + FN)

Measures how many actual targets were correctly identified. High recall = low false negatives.

- **F1 Score** = 2 × (Precision × Recall) / (Precision + Recall)
  Harmonic mean of precision and recall. Best when both are balanced.

Class	Precision	Recall	F1 Score
RightCard	0.25	0.09	0.14
LeftCard	0.33	0.17	0.22
PlaceCard	0.59	0.83	0.69
Macro Avg	0.39	0.36	0.35

Table 5.3.4: Model Performance Analysis

Interpretation:

- **High Precision + High Recall**: Indicates good generalization.
- **Low F1 Score** on specific classes: May need targeted data augmentation or feature tuning.

5.4 SUMMARY OF TOOLS & LIBRARIES USED

Component	**Tool/Library Used**	**Purpose**
EEG Streaming	pylsl	Real-time EEG acquisition
Signal Filtering	scipy.signal.butter	Band-pass Filtering
Feature Engineering	numpy, scipy.integrate	P300 amplitude & AUC & Latency
ML Classification	joblib, RandomForest	Real-time prediction
Communication Layer	Flask-SocketIO	Prediction delivery to frontend/game

Table 5.4: Tools & Libraries used in PYSL

6 CHAPTER 6: APPLICATION DEVELOPMENT & BUSINESS MODEL

6.1 EDUCATIONAL INTERFACE: WEBSITE FOR LEARNING AND INTERACTION

6.1.1 Educational Website for Special needs

This section describes the design and implementation of an advanced educational website that leverages Brain-Computer Interface (BCI) technology to enable interactive learning. The platform targets users with limited physical input capabilities and aims to provide a meaningful educational experience by combining P300 signal recognition with EEG-based attention monitoring [59].

6.1.2 Educational Website User Interface Design

The development of the educational platform involved two key components working in harmony: the website interface and the Brain-Computer Interface (BCI) system. Together, they provide a seamless and accessible learning experience, particularly for users with special needs.

The website design was developed with a strong emphasis on simplicity, clarity, and accessibility. The design follows key UI/UX principles to ensure that users can easily understand and interact with the educational content.

Key design features include:

- Large, clearly labelled buttons for each question option to make visual focus easier.
- High contrast colour schemes and clean layouts to reduce visual strain and support cognitive accessibility.
- Minimalistic design to avoid distractions and maintain user attention on the learning task. Responsive structure that works smoothly across different devices and screen sizes.

The educational content was structured into small, focused lessons and tests (e.g., math problems), with immediate feedback provided after each interaction to enhance the learning experience [60].

6.1.3 BCI Controlled Educational Website Ui/UX For Nonverbal Users (e.g., Button Grids):

The user interface was designed with accessibility and clarity in mind. The layout of every page follows a consistent structure: two-thirds of the screen is reserved for content display (e.g., educational material, video explanations, or book text, notes, or exams), while the remaining one-third on the right-side hosts navigation buttons. These buttons are arranged in a vertical grid pattern, allowing straightforward focus-based selection using BCI control. This consistency reduces cognitive load and enhances ease of use for users with special needs making navigation easily accessible and simple with minimal to-the-point options.

Each navigation button (e.g., Open, Next, Back, Save Exit) is large, spaced adequately, and presented in high-contrast colours with simple labels. This design ensures that visual attention can be effectively translated into brain signals for command recognition as shown in Fig 6.1.3.

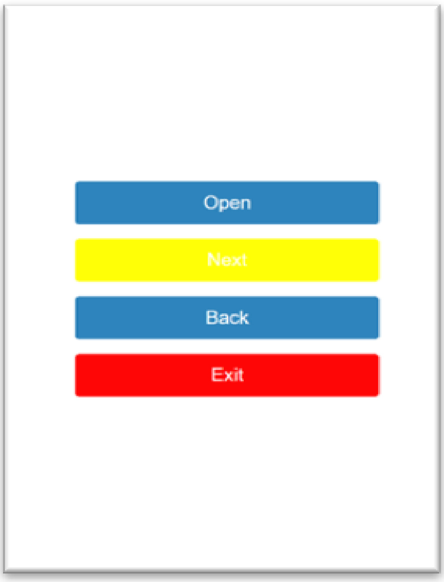


Figure. 6.1.3: Buttons Layout Example provided from the book categories illustrating the design and flashing technique

As part of the Brain-Computer Interface (BCI) system, a core objective was to develop an accessible educational website tailored for nonverbal users. The website acts as a basic communication and learning tool to ease the learning experience for individuals with special needs and make it more accessible as it is navigated entirely through EEG-based brain signals, particularly utilizing the P300 Event-Related Potential (ERP) paradigm with the use of the Unicorn Hybrid Black headset [61].

6.1.3.1 BCI-Controlled Text Generation or Symbol Selection:

The website is integrated with the P300 ERP control mechanism, where each button flashes in a randomized sequence. The user focuses their attention on the desired button of choice and as the desired button flashes at the instance the user is focused upon, that will trigger the P300 signal which the BCI system detects and so the website responds and navigates accordingly providing the action needed. These EEG brain signals are collected through our Unicorn Hybrid Black headset and are then interpreted through our AI machine learning classifier and used to execute the corresponding action (e.g., turn the page, save notes).

6.1.4 Website Core Components

1. Homepage (Menu)

The homepage [6.1.4.1] is where the user has all the options displayed for them to pick the section desired with the four options the website provides, books section, videos sections, notes section and exams section with a simple well distributed layout and clear fonts with the title centred. It is simple and user friendly for user comfort.

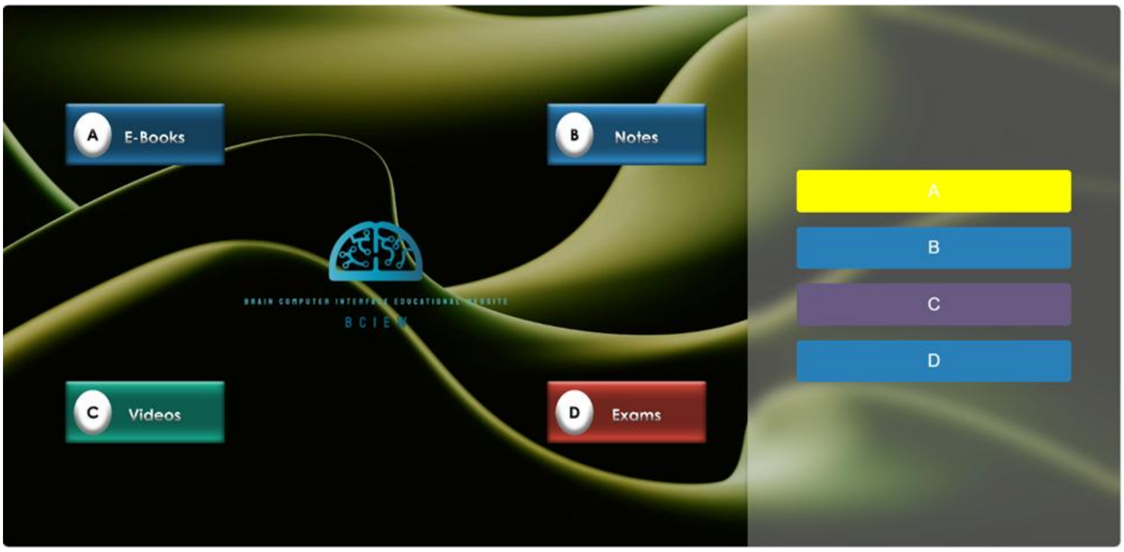


Figure. 6.1.4.1: Menu Page

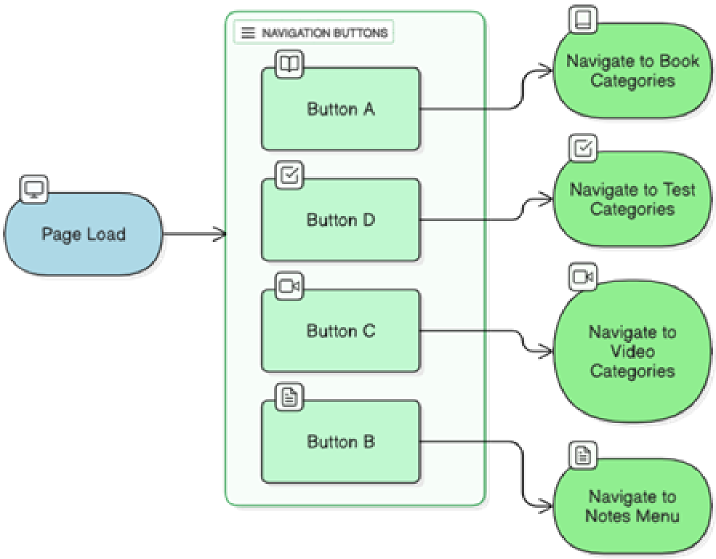


Figure. 6.1.4.2 Menu Page Flowchart

2. Subjects Entry Pages

The educational platform provides users with access to three core subject areas: Brain Computer Interface (BCI) [6.1.4.3], Operating System [6.1.4.4], and Cyber Security [6.1.4.5]. Upon entering either of the Books or Videos or Notes Section through the Menu page, users are presented with a subject selection page featuring large, easily identifiable covers for each topic. Once a subject is selected, the user is directed to its dedicated interface.

The following figs illustrate the interface for the categories of subjects.

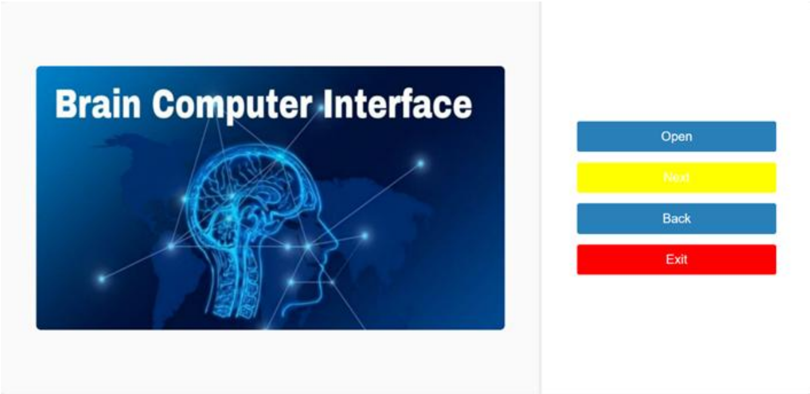


Figure. 6.1.4.3: Brain Computer Interface Subject Category

Figure. 6.1.4.4: Operating Systems Subject Category

Figure. 6.1.4.5: Cyber Security Subject Category

3. Subject Books Section

The Books Section of the website was developed with the intent of providing accessible, digestible educational content for individuals with special needs, especially those who are nonverbal or have limited mobility. The entry point into the Books Section is a selection screen that displays subject covers representing different learning topics which act as navigation targets, once a subject is picked, users navigate into the subject of desire. Each book is divided into short, concise pages, with minimal text and clear, legible font styles to reduce visual fatigue and cognitive overload, enabling a smoother and effective learning process. Users can navigate forward or backward through pages or exit using the consistent set of flashing navigation buttons on the right-hand panel. As well as having the option to save notes into our designed notes page for future reference as displayed in the Fig below in one of the books provided in our website's books [6.1.4.6].

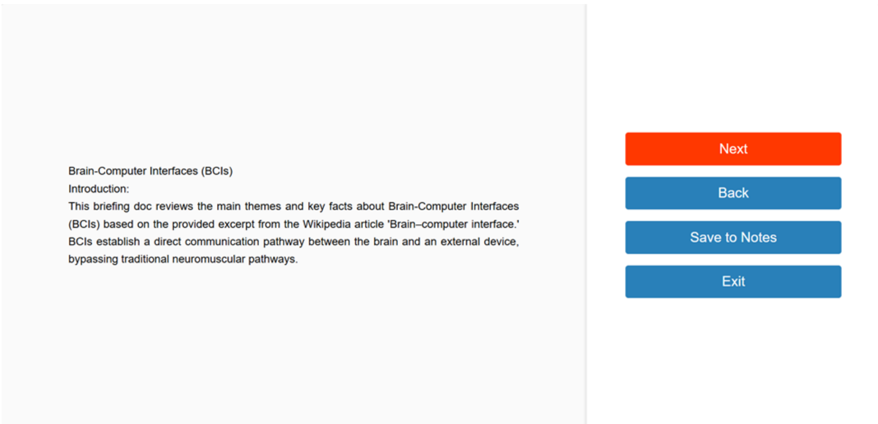


Figure. 6.1.4.6: BCI Book Page

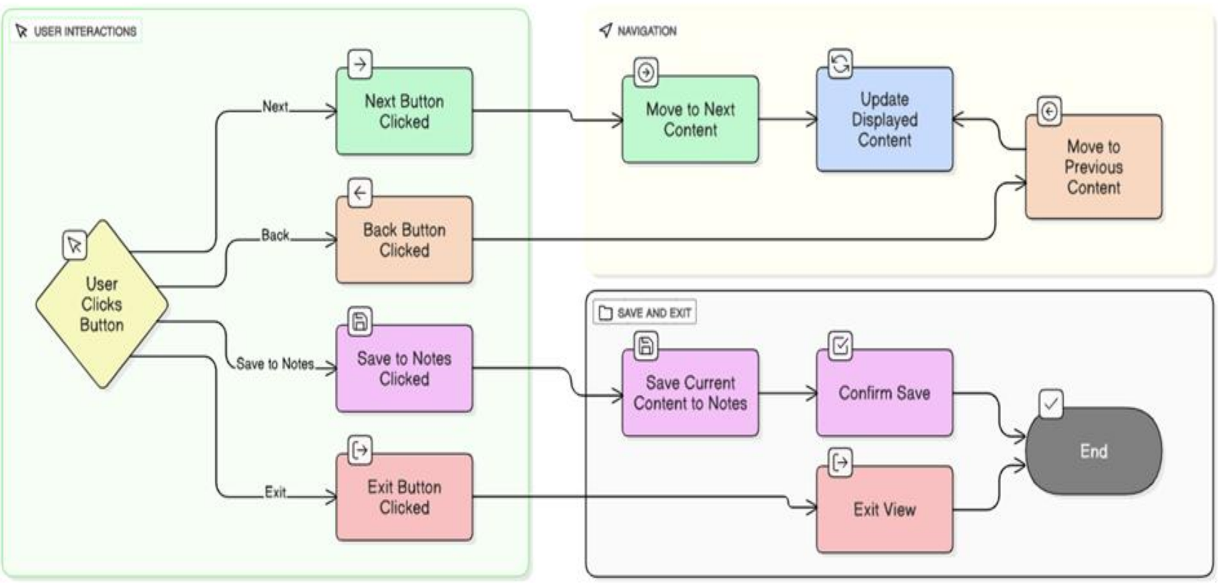


Figure. 6.1.4.7: Book Pages Flowchart

4. Educational Videos Section

The Videos Section of the website was developed to complement the Books Section for those who benefit from auditory and visual explanations. The entry point into the Videos Section is a selection screen with subject covers identical to those in the Books Section. Once a subject is selected, users are guided to a video interface [6.1.4.8] that presents curated content related to the chosen topic. Each video page features intuitive controls allowing users to play/pause, rewind, or skip forward, as well as exit the video using the same consistent flashing navigation button layout to support ease of use and familiarity across all website sections. The design minimizes distractions and ensures that individuals can focus fully on the video content.

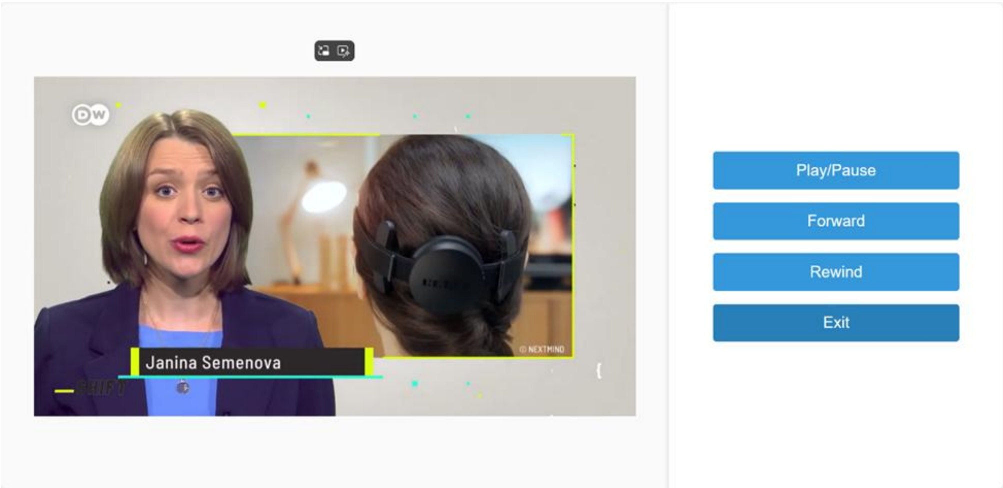


Figure. 6.1.4.8: One of the BCI videos the website provides

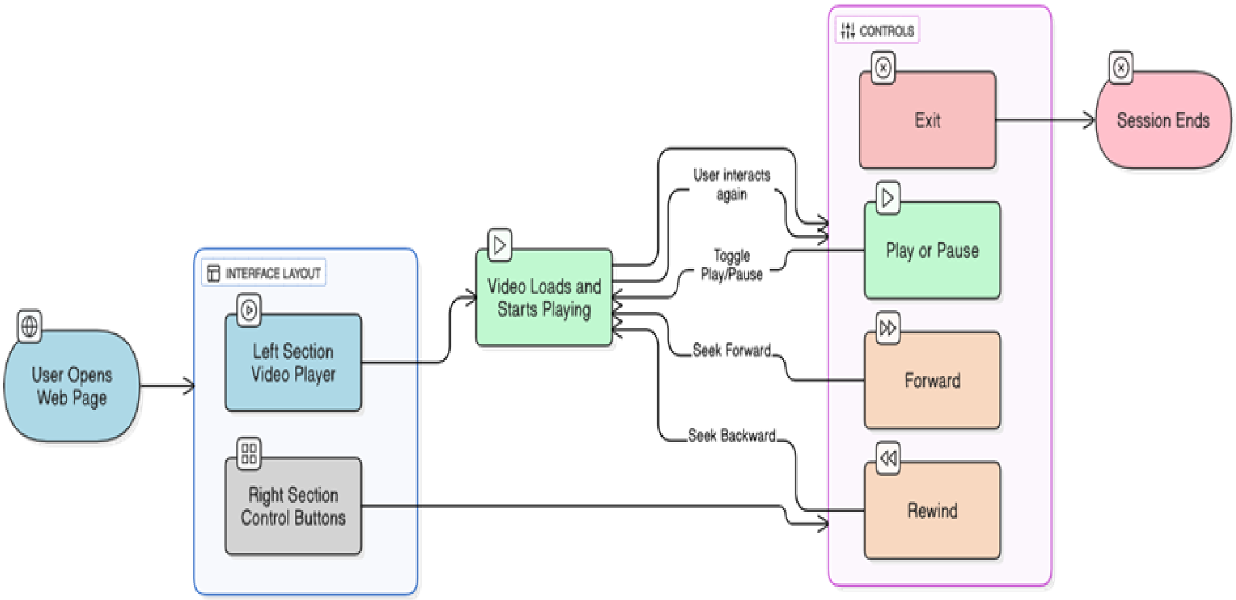


Figure. 6.1.4.9: Learn Section Flowchart

5. Notes Section

The Notes Section of the website was designed to serve as a personal repository where users can access information they have saved while exploring the Books Section. It provides an easy-to-use interface for individuals with special needs, allowing them to revisit important content at their own pace. The saved notes are presented in a clear format for optimal readability. Users can navigate through notes, delete entries, or exit the section using familiar ERP-triggered buttons on the right-hand side. [6.1.4.8]

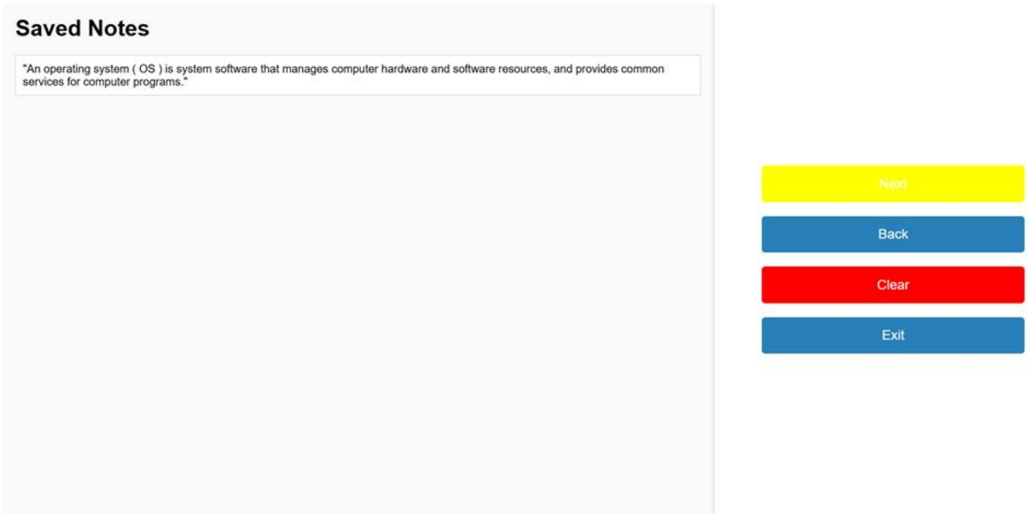


Figure. 6.1.4.10: OS Notes Page with a saved note from the OS book

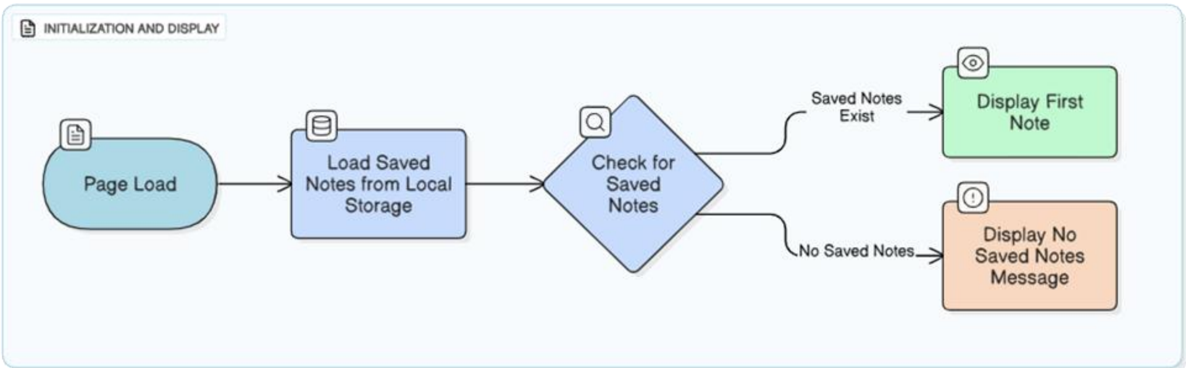


Figure. 6.1.4.11: Notes Section Flowchart

6. Exams

The Exams Section was developed to assess the user's knowledge. Designed with accessibility and cognitive simplicity in mind, this section offers a variety of question formats, including geographic map questions [6.1.4.12], matching questions [6.1.4.12], and MCQ problems [6.1.4.14] to test different areas of knowledge and learning styles. With labelled answer choices presented on the right-hand panel. After, the system automatically processes responses and displays a final score. The design maintains consistency with the rest of the website, ensuring users with special needs experience seamless interactions from learning to self-assessment, all within a P300-compatible environment.

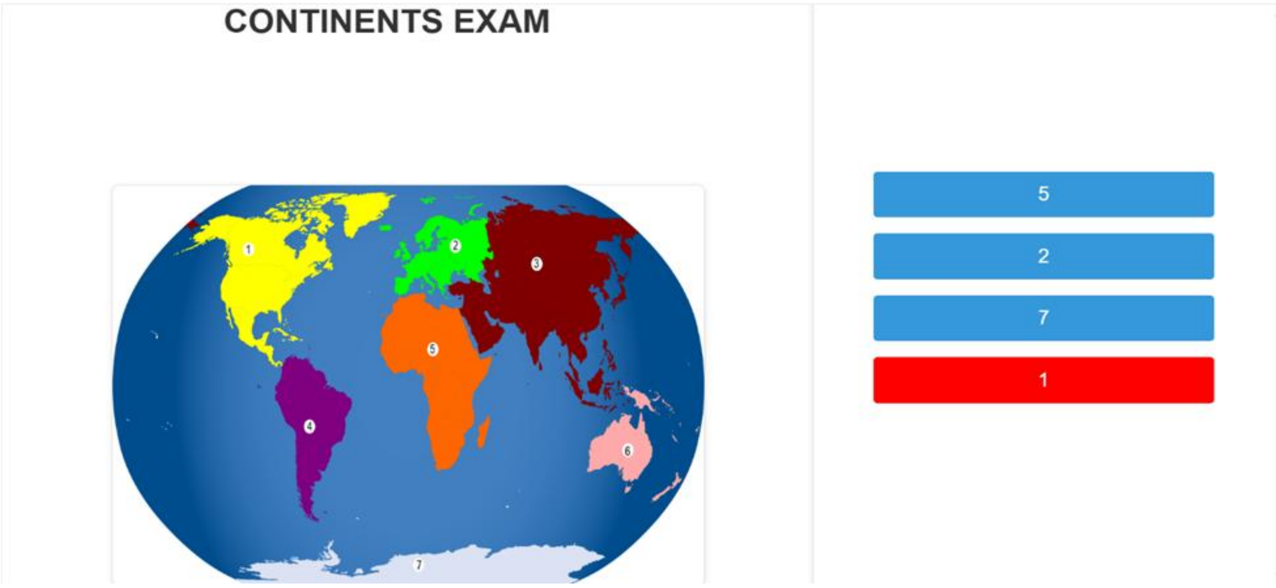


Figure. 6.1.4.12: Geography Exam

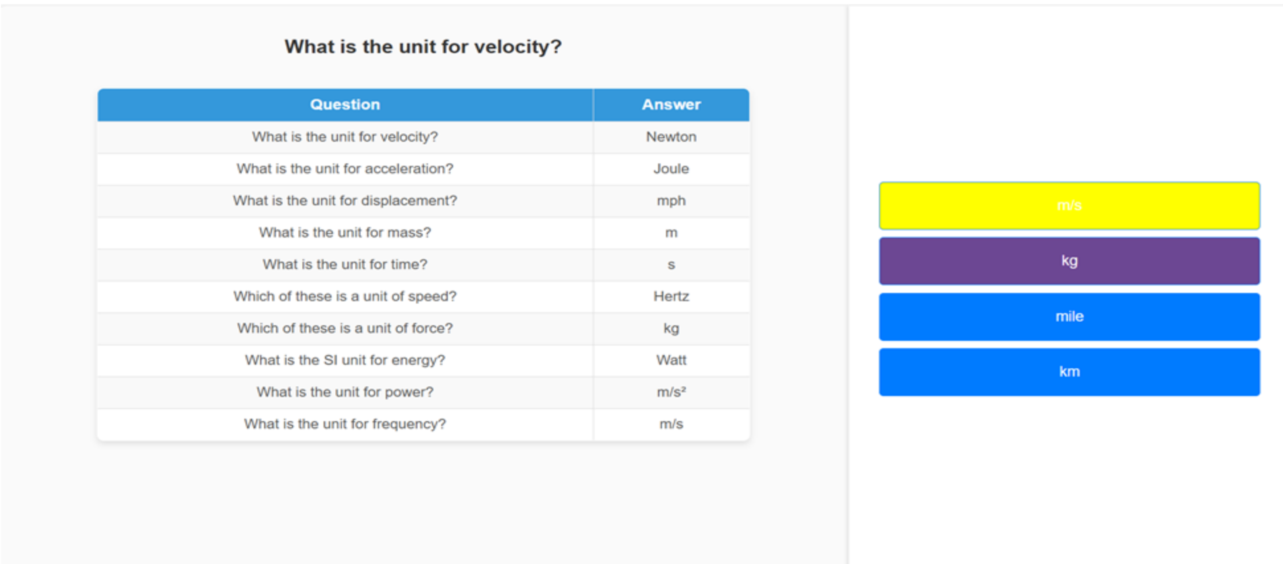


Figure. 6.1.4.13: Matching question provided in Physics exam

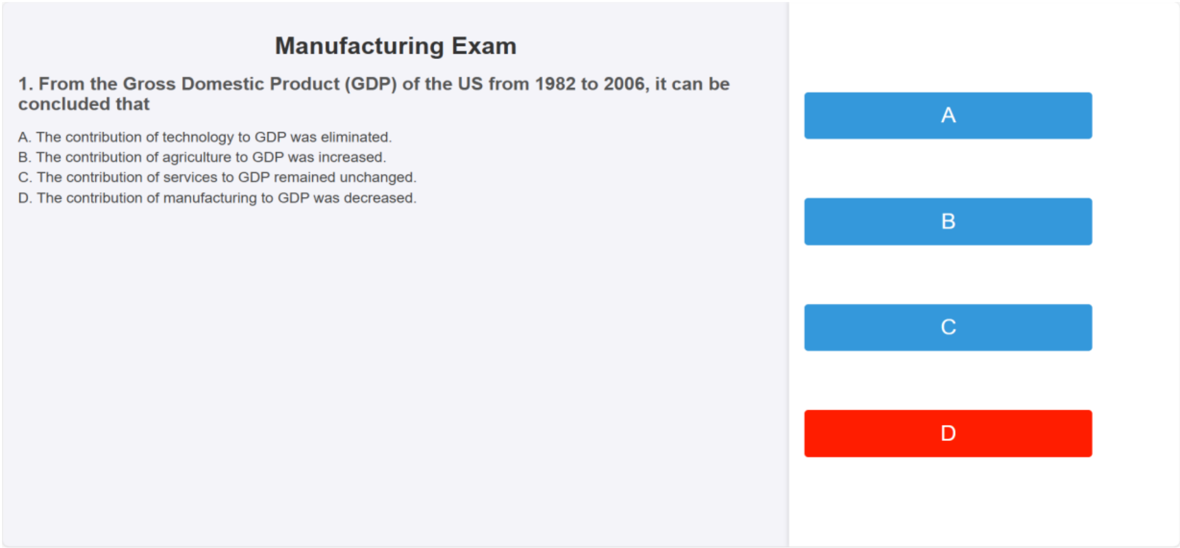


Figure. 6.1.4.14: MCQ question provided in the Manufacturing exam

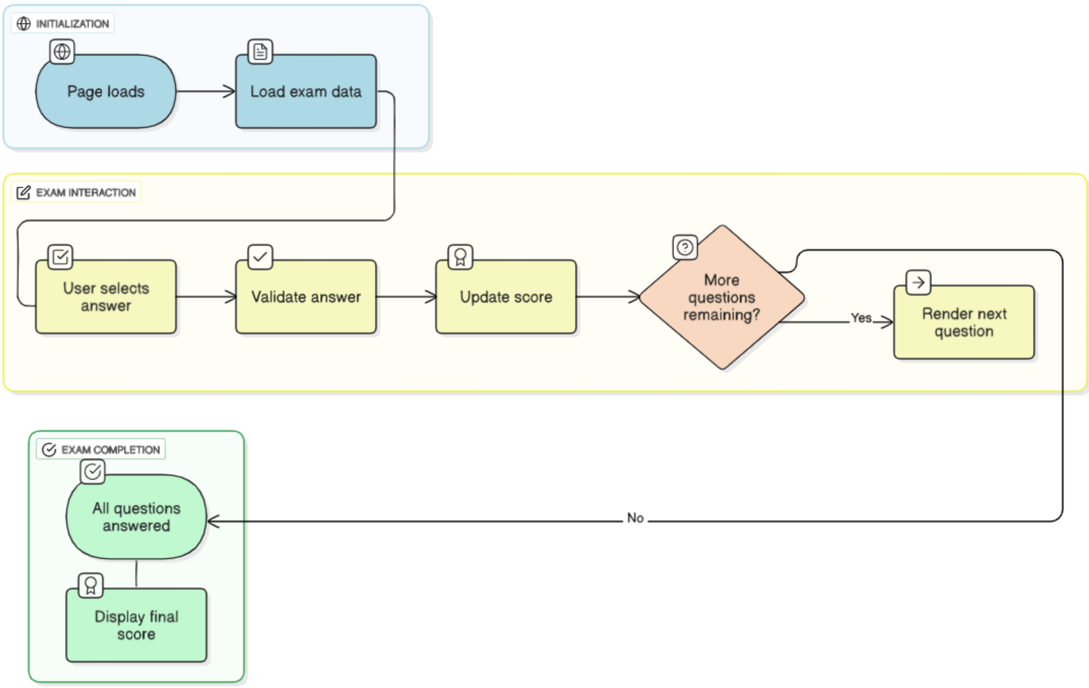


Figure. 6.1.4.15: Exams Flowchart

6.1.5 Preliminary Usability Testing

Preliminary usability testing was conducted to evaluate the effectiveness and accessibility of the website's interface. The primary metric used was task completion time, which measured how long it took users to complete predefined actions (e.g., navigating from the homepage to a specific book section, saving a note, or playing an educational video).

Tasks were performed using simulated P300 input, mimicking the visual attention dynamics of real users. Each interaction was broken into sequential steps (e.g., focus on "Books", wait for selection, then focus on a specific subject). The use of a consistent button grid and a minimal, distraction-free UI proved to reduce confusion and improve response time. Average task durations were recorded to benchmark future enhancements [62].

6.1.5.1 Offline P300 Data Collection Sessions

To train and validate the P300-based Brain-Computer Interface (BCI) system, multiple offline data collection sessions were conducted using the Unicorn Hybrid Black EEG headset. During these sessions, users were instructed to focus on a yellow-highlighted target button, while a grid of buttons flashed in a random sequence.

Each button flash cycle was designed so that the target button appeared five times during the session. When the user's brain generated a P300 signal in response to their target flashing, it was recorded. These sessions typically lasted around 5 to 10 seconds per button selection, though the total training phase may extend to 5 minutes or more, depending on user eye comfort and attention span of the user undergoing the data collection session. The more data collected, the more robust the AI classifier's performance.

Collected EEG signals were then labelled and processed offline to train a machine learning classifier capable of detecting P300 events, forming the foundation of the system's intelligent response mechanism.

6.1.5.2 Online Analysis and Real-Time Interaction

Following successful offline training, the system transitioned into online mode, where real-time EEG input was used to control the website interface. When a user focuses on a desired button, the previously trained AI classifier detects the P300 response generated from their visual attention and makes a corresponding selection within the website.

This phase allowed us to validate real-time responsiveness and practical usability, simulating actual usage by individuals with limited mobility. Real-time interactions may be slower compared to conventional input methods due to the fact the user has the free will to observe the content for as long as needed and once decided they can pick whichever button, however online analysis is faster often taking around 3 secs seconds per button selection considering the fact the button only needs to flash only once in front of the user instead of 5 times as determined through the offline data collection sessions. Those real-time sessions demonstrate a functional, hands-free interaction model suitable for nonverbal users.

Ongoing and future evaluations will include more test subjects to further refine accuracy, reduce latency, and enhance the end-user experience across different content areas of the platform.

6.2 UNO GAME CONTROLLED BY BCI

6.2.1 Game design & BCI integration (P300 triggers for movement)

The Uno Card Game was developed as a core component of this research, designed to facilitate brain-computer interface (BCI) integration, specifically utilizing P300 event-related potentials for user control. The game's design prioritizes a clear distinction between the uno ground area and the interactive control elements, with a 2/3 to 1/3 ratio respectively. This ensures that the primary focus remains on the navigation task while providing a dedicated area for BCI interaction.

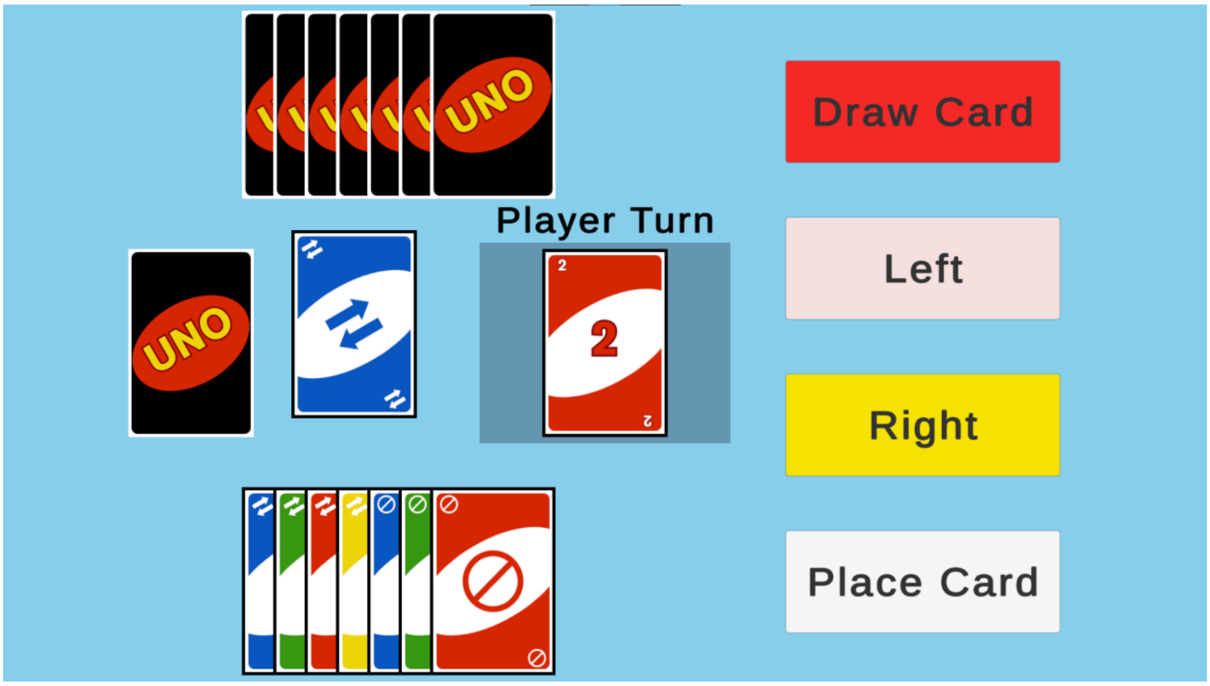


Figure 6.2.1: Uno Card Game Graphic User Interface

6.2.1.1 Game Design

The game environment is visually divided into two main sections: the playing section and the control panel section. The playing section occupies approximately two-thirds of the screen, providing enough space for player, computer and ground cards. The remaining one-third is dedicated to the control buttons, which are the primary interface for P300-based interaction. This spatial separation minimizes visual clutter and potential distractions, enhancing the user's ability to concentrate on the target stimuli.

6.2.1.2 P300 Integration

The P300 BCI integration is implemented in two distinct phases: an offline model training phase and an online control phase. Both phases leverage visual flashing paradigms to elicit P300 responses, which are then interpreted by the BCI system to determine user intent.

6.2.2 Offline Model Training

During the offline model training phase, the system aims to collect sufficient P300 data to train a robust classification model. This phase is characterized by a specific visual stimulation protocol:

- **Target Button Highlight:** A yellow highlight appears on the target button, indicating the button the user should concentrate on.

- **Random Red Flashes:** All control buttons, including the target, flash red randomly. This random flashing serves as the trigger stimulus to elicit P300 responses. The flashing sequence is designed such that for each target button has an intersection chance of 1 flash intersection occurrence in every 10 flashes with the yellow target button. This parameter, 'Random Flashes Before Trigger', is configurable, as seen in the Inspector screenshot provided.

- **P300 Trigger:** The user is instructed to concentrate on the yellow-highlighted target button. When red flashes appear on the yellow button, it is designed to trigger a P300 response due to the user's focused attention and the trigger nature of the target flash within the random sequence. Upon detection of this P300 trigger, the system records the button name and the time of the trigger. This process continues for a series of target buttons, allowing for the collection of diverse P300 data across different button locations.

6.2.3 Online Model Control

Following the successful training of the offline model, the system transitions to the online control phase. In this phase, the BCI system operates in real-time to translate brain signals into game actions:

- **Continuous Random Flashes:** Similar to the offline phase, red flashes continue to appear randomly across all control buttons.
- **Real-time Prediction:** The trained AI model continuously analyzes the user's brain activity in response to these flashes. Based on the P300 responses, the model predicts which button the player is concentrating on.
- **Player Chosen Button:** Upon a confident prediction from the AI model, the game translates this intent into action by choosing the desired button that is predicted by the AI Model. This direct control mechanism allows the user to navigate through his cards , draw cards and place cards on the ground using only their brain signals.

This two-part integration ensures that the BCI system is first calibrated to the individual user's P300 responses and then deployed for intuitive, real-time control within the game environment.

6.2.4 Latency optimization for real-time control

Real-time control in BCI applications is critically dependent on minimizing latency between brain signal acquisition, processing, and game response. Several strategies were implemented to optimize latency in the Uno Card Game, ensuring a fluid and responsive user experience.

One key aspect of latency optimization involves efficient communication between the game application (developed in Unity) and the external BCI processing unit. The MessageManager.cs script, as shown below, handles this communication, sending messages (e.g., button names, trigger times) to an external server and potentially receiving predictions.

6.2.4.1 Message Manager Script

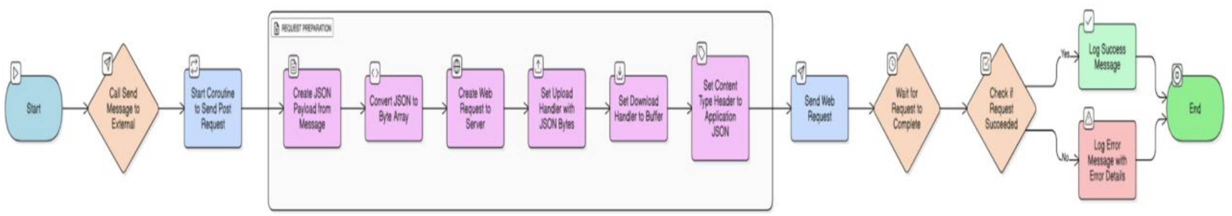


Figure 6.2.4.1: MessageManager.cs for inter-process communication

The MessageManager script utilizes Unity's UnityWebRequest to send POST requests containing JSON payloads to an external server. This server is responsible for processing the BCI data and returning predictions. The use of asynchronous coroutines "(StartCoroutine(SendPostRequest(message)))" ensures that network communication does not block the main game thread, which is crucial for maintaining responsiveness.

The “serverUrl” and “receiveUrl” are configurable, allowing for flexible deployment and testing environments.

Another critical aspect of latency optimization, particularly for visual stimuli, is ensuring consistent and predictable rendering across different display configurations. The “AdaptiveCanvasScaler.cs” script addresses this by dynamically adjusting the UI scaling based on the screen’s aspect ratio. This prevents visual distortions that could impact the timing and perception of the flashing stimuli, thereby affecting P300 elicitation and overall system latency.

Furthermore, the “GameManager.cs” script, which directly controls the button actions, is designed to execute commands received from the BCI system with minimal delay. This script would typically receive a target button identifier from the MessageManager object and then click the desired button to execute its action. The efficiency of this action logic is paramount for a responsive BCI experience.

6.2.4.2 Game Manager Script

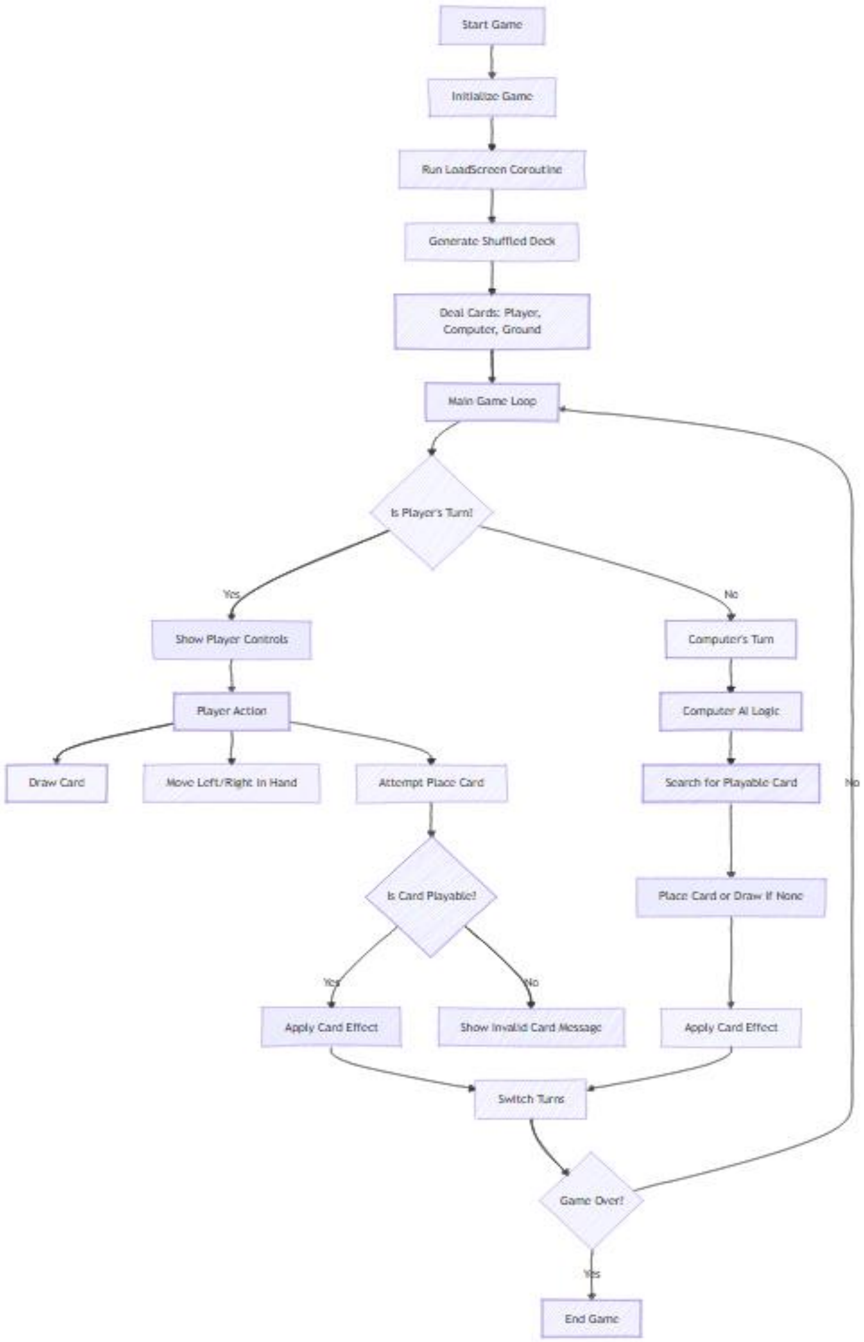


Figure 6.2.4.2: GameManager.cs for game logic and player movement

The GameManager.cs script is central to the game's functionality, managing everything from game state and UI elements and button actions. Its key responsibilities include:

- **Game State Management**: The script handles game initialization (Start, LoadScreen, Card Distribution), UI updates (SetActiveButtons), and game progression, including win/loss conditions (Game_Message_Text).

- **Card Database:** Uno card game normally has 56 card in total including numbered cards from 0 till 9 with each number has four color variant as well as the special cards which are draw two cards , skip turn and reverse the cycle and the wild cards which are draw four cards and choose any color cards

- **Player Decision:** Based on the BCI system's predictions (which would be received and processed by the MessageManager and then relayed to GameManager), the script updates the player's decision within the game.

- **Button Actions:** based on what the player decision there must be an action for this decision so here we added the ability when certain button is chosen it performs its action.

6.2.4.3 P300 Button Flasher

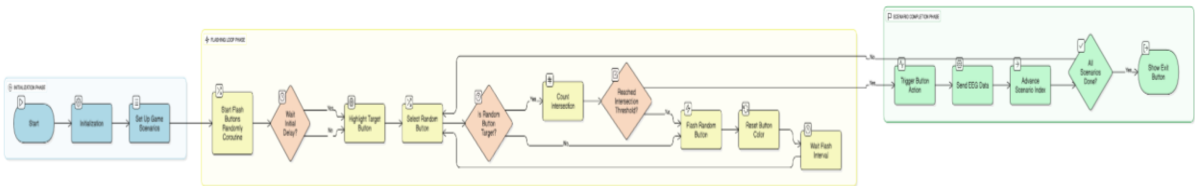


Figure 6.2.4.3.1: P300ButtonFlasher.cs for Random Red and Yellow Flash on Buttons

Flash Duration	0.014
Interval Between Flashes	0.007
Random_Temp	0
Reach_Intersection_Occurance	9

Figure 6.2.4.3.2: Flashing Settings in Unity Inspector

The P300ButtonFlasher.cs script is used to make the random flash on buttons functionality, managing everything from setting button colours to red or yellow and UI elements to the flashing sequence. Its key responsibilities include:

- **Initialize Process:** here we insert specific game scenarios for player to only win the game in our case we implemented 5 different scenarios for player to win as well as there is a wait condition between each scenario in which a next scenario button appears so that the client can take breaks between scenarios if needed if not just click next scenario button and games goes normally.

- **Scenario Configuration:** It defines different game scenarios (only Winning) and configures the correct sequence of buttons (correctSequence) and associated labels for each path. This allows for varied gameplay scenarios and experimental conditions.

- **Flashing Sequence Control:** The FlashSequence coroutine implements the P300 stimulation paradigm. It manages the random red flashes on non-target buttons and the yellow/red flashes on the target button. The randomFlashesBeforeTarget, flashDuration, and intervalBetweenFlashes parameters are crucial for tuning the P300 elicitation process and are directly linked to the Inspector settings shown in Figure 6.4.

- **P300 Trigger Handling**: When the target button flashes red (signifying a P300 trigger event), the script records the button name and the intersectionTime (the exact time of the flash). This information is then sent to the external BCI processing unit via the MessageManager "(SendMessage($"{targetButton.name} hit at {intersectionTime}");)". This real-time data transmission is vital for the online BCI model to make predictions.

Optimizing the "P300ButtonFlasher.cs" script for low latency involves several considerations:

- Efficient Coroutine Management: The FlashSequence coroutine uses "yield return new WaitForSeconds()" for precise timing of flashes. Minimizing the duration of these waits and ensuring that the coroutine does not perform computationally intensive tasks within its loop helps maintain responsiveness.

- Direct UI Manipulation: Direct manipulation of UI elements (e.g., SetButtonColor, SetActive) is generally efficient in Unity, contributing to low visual latency.

- Streamlined Communication: The integration with "MessageManager" ensures that data is sent to the BCI backend as quickly as possible, minimizing delays in the control loop.

6.2.4.4 UDP Connection

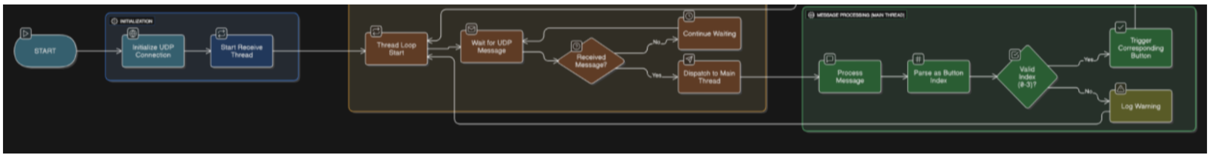


Figure 6.2.4.4: UDPConnection.cs for connection between AI Model and Uno Card Game

The "udpConnection.cs" script is the main gate for UDP communication between the AI Model predictions and game buttons so that when the game listens for incoming messages and triggers button clicks based on received data. Let's discuss key responsibilities:

- **Initialize process:** In here we set ip address and port number for communications between AI model and game buttons and we declared a button array that matches the key mapping of buttons for ease button access its actions.

By carefully designing these scripts and their interactions, the game achieves the necessary balance between complexity, playability and responsiveness.

6.3 MAZE NAVIGATION GAME CONTROLLED BY BCI

6.3.1 Game design & BCI integration (P300 triggers for movement)

The Maze Navigation Game was developed as a core component of this research, designed to facilitate brain-computer interface (BCI) integration, specifically utilizing P300 event-related potentials for user control. The game's design prioritizes a clear distinction between the maze environment and the interactive control elements, with a 2/3 to 1/3 ratio respectively. This ensures that the primary focus remains on the navigation task while providing a dedicated area for BCI interaction.

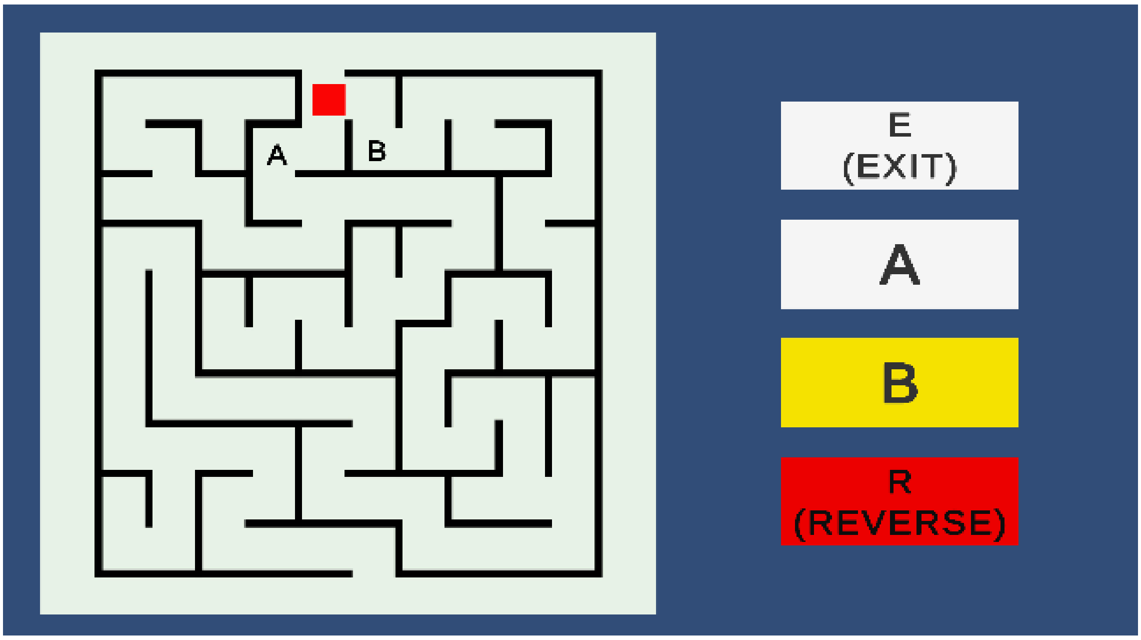


Figure 6.3.1: Maze Game Graphic User Interface

6.3.1.1 Game Design

The game environment is visually divided into two main sections: the maze area and the control panel. The maze occupies approximately two-thirds of the screen, providing enough space for complex maze structures and player movement. The remaining one-third is dedicated to the control buttons, which are the primary interface for P300-based interaction. This spatial separation minimizes visual clutter and potential distractions, enhancing the user's ability to concentrate on the target stimuli.

6.3.1.2 P300 Integration

The P300 BCI integration is implemented in two distinct phases: an offline model training phase and an online control phase. Both phases leverage visual flashing paradigms to elicit P300 responses, which are then interpreted by the BCI system to determine user intent.

6.3.2 Offline Model Training

During the offline model training phase, the system aims to collect sufficient P300 data to train a robust classification model. This phase is characterized by a specific visual stimulation protocol:

- **Target Button Highlight:** A yellow highlight appears on the target button, indicating the button the user should concentrate on.

- **Random Red Flashes:** All control buttons, including the target, flash red randomly. This random flashing serves as the ‘oddball’ stimulus to elicit P300 responses. The flashing sequence is designed such that each button flashes 9 times before an intersection with the yellow target button. This parameter, ‘Random Flashes Before Trigger’, is configurable, as seen in the Inspector screenshot provided (Figure 6.4.2).

- **P300 Trigger:** The user is instructed to concentrate on the yellow-highlighted target button. When two red flashes appear on the yellow button, it is designed to trigger a P300 response due to the user’s focused attention and the ‘oddball’ nature of the target flash within the random sequence. Upon detection of this P300 trigger, the system records the button name and the time of the trigger. This process continues for a series of target buttons, allowing for the collection of diverse P300 data across different button locations.

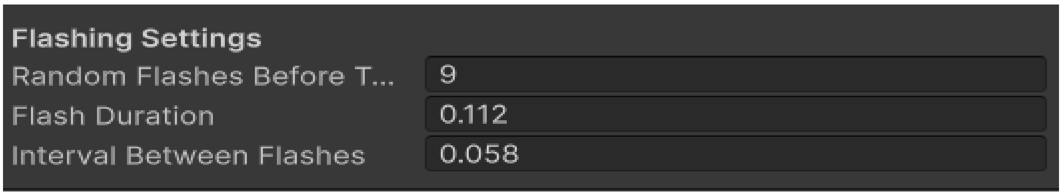


Figure 6.3.2: Flashing Settings in Unity Inspector

6.3.3 Online Model Control

Following the successful training of the offline model, the system transitions to the online control phase. In this phase, the BCI system operates in real-time to translate brain signals into game actions:

- **Continuous Random Flashes:** Similar to the offline phase, red flashes continue to appear randomly across all control buttons.

- **Real-time Prediction:** The trained AI model continuously analyzes the user's brain activity in response to these flashes. Based on the P300 responses, the model predicts which button the player is concentrating on.
- **Maze Block Movement:** Upon a confident prediction from the AI model, the game translates this intent into action by moving the maze block (representing the player's position) to the desired button's location. This direct control mechanism allows users to navigate the maze using only their brain signals [26].

This two-part integration ensures that the BCI system is first calibrated to the individual user's P300 responses and then deployed for intuitive, real-time control within the game environment.

6.3.4 Latency optimization for real-time control

Real-time control in BCI applications is critically dependent on minimizing latency between brain signal acquisition, processing, and game response. Several strategies were implemented to optimize latency in the Maze Navigation Game, ensuring a fluid and responsive user experience [27].

One key aspect of latency optimization involves efficient communication between the game application (developed in Unity) and the external BCI processing unit. The MessageManager.cs script, as shown below, handles this communication, sending messages (e.g., button names, trigger times) to an external server and potentially receiving predictions.

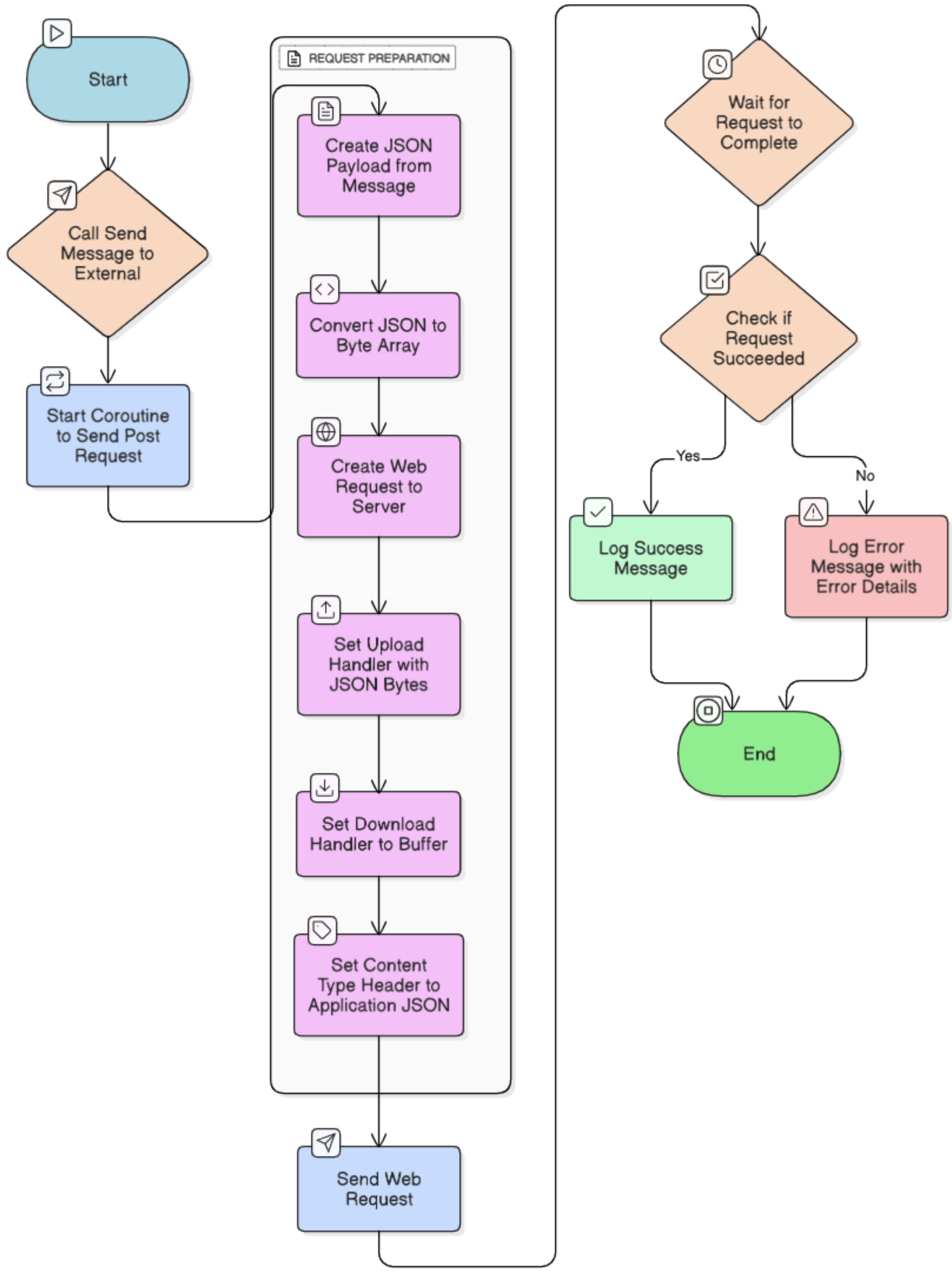


Figure 6.3.4: MessageManager.cs for inter-process communication

6.3.4.1 Message Manager Script

The MessageManager script utilizes Unity's UnityWebRequest to send POST requests containing JSON payloads to an external server. This server is responsible for processing the BCI data and returning predictions. The use of asynchronous coroutines (StartCoroutine(SendPostRequest(message))) ensures that network communication does not block the main game thread, which is crucial for maintaining responsiveness. The serverUrl and receiveUrl are configurable, allowing for flexible deployment and testing environments.

Another critical aspect of latency optimization, particularly for visual stimuli, is ensuring consistent and predictable rendering across different display configurations. The AdaptiveCanvasScaler.cs script addresses this by dynamically adjusting the UI scaling based on the screen's aspect ratio. This prevents visual distortions that could impact the timing and perception of the flashing stimuli, thereby affecting P300 elicitation and overall system latency.

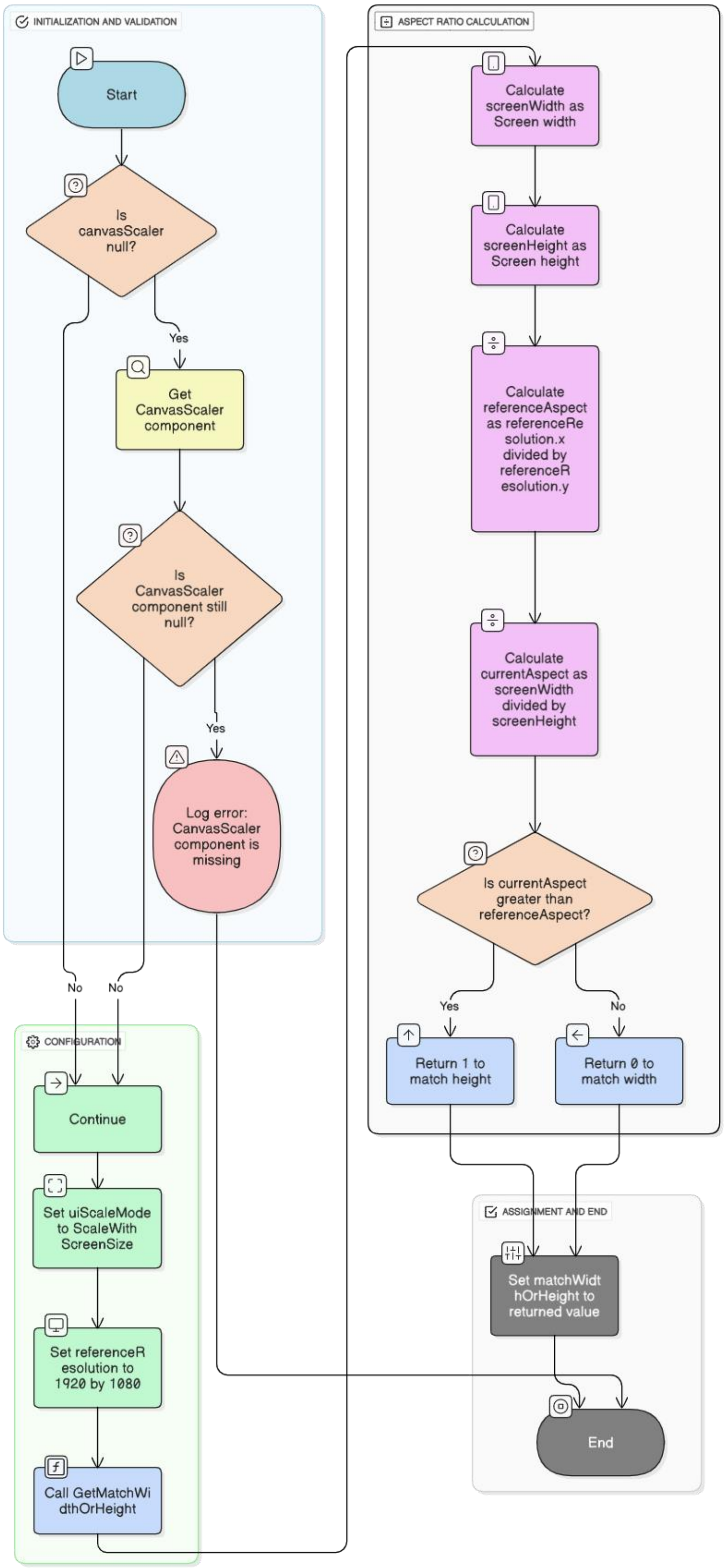


Figure 6.3.4.1: AdaptiveCanvasScaler.cs for dynamic UI scaling

6.3.4.2 Adaptive Canvas Scaler Script

The DynamicCanvasScaler script ensures that the game's user interface scales correctly across various screen resolutions and aspect ratios. By setting the uiScaleMode to ScaleMode.ScaleWithScreenSize and dynamically adjusting matchWidthOrHeight based on the current screen aspect ratio, the visual presentation of the flashing stimuli remains consistent. This consistency is vital for reliable P300 elicitation, as variations in stimulus size or position could introduce noise and increase latency in BCI control.

Furthermore, the MovePlayer.cs script, which directly controls the maze block's movement, is designed to execute commands received from the BCI system with minimal delay. This script would typically receive a target button identifier from the MessageManager and then smoothly transition the player's block to the corresponding position within the maze. The efficiency of this movement logic is paramount for a responsive BCI experience.

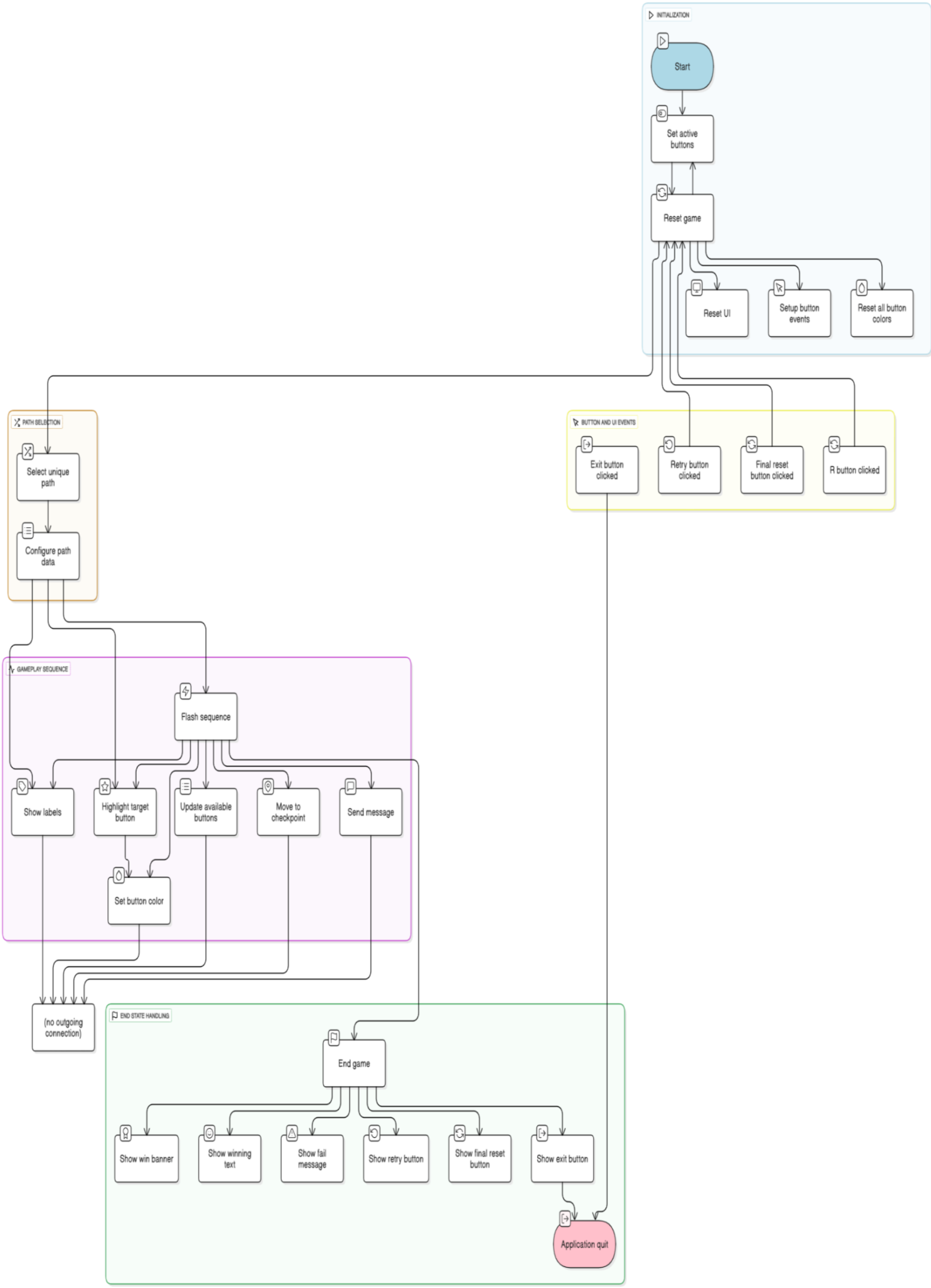


Figure 6.3.4.2: MovePlayer.cs for game logic and player movement

6.3.4.3 Move Player Script

The MovePlayer.cs script is central to the game's functionality, managing everything from game state and UI elements to the flashing sequence and player movement. Its key responsibilities include:

- **Game State Management:** The script handles game initialization (Start, ResetGame), UI updates (ResetUI, SetActiveButtons), and game progression, including win/loss conditions (EndGame).

- **Path Configuration:** It defines different game paths (GamePath.Winning, GamePath.FailA4, GamePath.FailB3) and configures the correct sequence of buttons (correctSequence) and associated labels for each path. This allows for varied gameplay scenarios and experimental conditions.

- **Flashing Sequence Control:** The FlashSequence coroutine implements the P300 stimulation paradigm. It manages the random red flashes on non-target buttons and the yellow/red flashes on the target button. The randomFlashesBeforeTarget, flashDuration, and intervalBetweenFlashes parameters are crucial for tuning the P300 elicitation process and are directly linked to the Inspector settings shown in Figure 6.4.2.

- **P300 Trigger Handling:** When the target button flashes red (signifying a P300 trigger event), the script records the button name and the intersectionTime (the exact time of the flash). This information is then sent to the external BCI processing unit via the MessageManager (messageManager?.SendMessage($"{targetButton.name} hit at {intersectionTime}");). This real-time data transmission is vital for the online BCI model to make predictions.

- **Player Movement:** Based on the BCI system's predictions (which would be received and processed by the MessageManager and then relayed to MovePlayer), the script updates the player's position within the maze. The transform.position = checkList[checkpointIndex]?.transform.position line demonstrates how the player's block is moved to predefined checkpoints as the user successfully navigates the sequence.

Optimizing the MovePlayer.cs script for low latency involves several considerations:

- **Efficient Coroutine Management:** The FlashSequence coroutine uses yield return new WaitForSeconds() for precise timing of flashes. Minimizing the duration of these waits and ensuring that the coroutine does not perform computationally intensive tasks within its loop helps maintain responsiveness.

- **Direct UI Manipulation:** Direct manipulation of UI elements (e.g., SetButtonColor, SetActive) is generally efficient in Unity, contributing to low visual latency.

- **Streamlined Communication:** The integration with MessageManager ensures that data is sent to the BCI backend as quickly as possible, minimizing delays in the control loop.

By carefully designing these scripts and their interactions, the game achieves the necessary.

6.3.5 User Feedback from Disabled Testers

User feedback is paramount in the development of assistive technologies, especially for applications involving Brain-Computer Interfaces. Throughout the development of the Maze Navigation Game, iterative testing with disabled users provided invaluable insights, leading to significant improvements in usability, accessibility, and overall user experience. This section summarizes the key feedback received and the subsequent modifications implemented.

Initial testing phases focused on the clarity of the visual stimuli, the intuitiveness of the P300-based control mechanism, and the overall game difficulty. Testers provided feedback on:

- **Visual Clarity of Flashes:** Some users found the initial flash durations or intervals to be either too fast or too slow, impacting their ability to consistently elicit P300 responses. Adjustments were made to the flashDuration and intervalBetweenFlashes parameters (as seen in Figure 6.4.2) to accommodate a wider range of cognitive processing speeds and visual sensitivities. The ability to customize these settings per user was identified as a critical feature.

- **Button Layout and Size:** Feedback indicated that the size and spacing of the control buttons were crucial for accurate eye gaze and concentration. Larger buttons with clear visual separation were preferred to minimize accidental P300 triggers on adjacent buttons. The AdaptiveCanvasScaler.cs (Figure 6.4.4) played a significant role in ensuring that these UI elements maintained optimal proportions across different display sizes.

- **Game Difficulty and Progression:** The initial maze designs and path lengths were sometimes perceived as either too challenging or too simplistic. Iterative adjustments were made to the maze complexity, the number of checkpoints, and the length of the correctSequence in MovePlayer.cs to provide a balanced and engaging experience. The inclusion of different GamePath options (Winning, FailA4, FailB3) allowed for varied difficulty levels and scenarios, catering to different user abilities and training needs.

- **Feedback Mechanisms:** Users emphasized the importance of clear and immediate feedback on their actions. Visual cues, such as the player block moving to the predicted location, and auditory feedback (though not explicitly detailed in the provided code, would be a common addition) were crucial for users to understand if their concentration was successful. The EndGame function in MovePlayer.cs provides clear visual feedback for winning or losing scenarios, which was well-received.

- **Error Handling and Recovery:** The ability to easily reset the game or retry a path was highlighted as important, especially during the initial learning phases. The retryButton and finalResetButton functionalities in MovePlayer.cs were implemented to address this, allowing users to quickly restart without frustration.

Overall, the user feedback underscored the importance of a highly customizable and adaptive system. The modular design of the game, with configurable parameters in the Unity Inspector and well-defined scripts, allowed for rapid iteration and integration of user suggestions. This user-centered design approach was instrumental in transforming the Maze Navigation Game into an effective and user-friendly BCI application for the disabled.

6.4: EXPERIMENTAL WORK

6.4.1 Learning Outcome Observations Using BCI-Controlled Input

To evaluate the educational potential and practical usability of the BCI-based learning platform, a series of informal tests were conducted involving project team members and one external participant — a student with physical disabilities [81].

Testing Process
The system was connected to the Unicorn Hybrid Black EEG headset and introduced to the student, who was instructed to focus on flashing options (e.g., math answers) displayed on the screen. The system then processed the user's P300 event-related potentials to infer selections, enabling interaction without any physical input [82], [83].

After initial success, the team extended testing with other participants to refine system responsiveness, accuracy, and robustness. These internal trials helped verify the platform's ability to detect P300 signals reliably across different users and environments [84].

Results and Observations

- The system consistently detected P300 responses and selected the correct option in most trials [85].
- The student was able to independently complete educational tasks, such as solving basic math problems [86].
- The user found the experience engaging and empowering, emphasizing the value of independence in a learning context [87].
- Real-time EEG-based attention monitoring enabled dynamic adjustment of task difficulty [54].

- Despite the small test group, results demonstrated the platform's potential as an inclusive educational tool for learners with physical impairments [88].

6.4.2 Maze Navigation: Performance Evaluation

To further assess the system's capabilities, the P300-based BCI was integrated with a Maze Navigation Game, and a series of structured experiments were conducted to measure task performance quantitatively. Key performance indicators (KPIs) focused on the system's accuracy in interpreting user intent and the user's ability to complete the maze [69].

Success Rate
The Success Rate measures how often users were able to navigate from the maze's start to endpoint using only P300 signals.
Formula
Success Rate = (Number of Successful Trials / Total Number of Trials) × 100%
Influencing Factors:

- **User Concentration**: Sustained focus during the flashing sequence is critical for generating strong P300 responses [70].
- **Model Accuracy**: A well-trained P300 classifier improves prediction reliability [71].
- **UI Parameters**: Variables such as flashDuration, intervalBetweenFlashes, and randomFlashesBeforeTarget significantly affect signal clarity and user comfort [57].
- **Maze Complexity**: More complex mazes increase cognitive load, potentially reducing performance [73].

Error Rate Metrics
To understand failure points and refine the system, three specific error metrics were tracked:

1. **Misclassification Errors**
   These occur when the classifier incorrectly identifies the user's intended selection.
   Formula:
   Misclassification Error Rate = (Misclassified Selections / Total Selections) × 100%
   Causes:
   - Poor signal-to-noise ratio in EEG data [74].
   - Inter-subject variability in P300 amplitude [75].
   - Artifacts (e.g., eye blinks, muscle activity) [76].
2. **Path Deviation Errors**
   These track instances where the user deviates from the correct maze path, even if the selected button was not technically incorrect.
   Formula:
   Path Deviation Error Rate = (Incorrect Path Choices / Total Path Segments Attempted) × 100%.
   Significance:
   This error highlights difficulties in navigation logic or the need for clearer directional guidance in-game [77].

3. **Time-out Errors**
   These represent failures to register a P300-based selection within the allotted time window.
   Formula:
   Time-out Error Rate = (Number of Time-outs / Total Expected Predictions) × 100%.
   Common Causes:
   - Weak P300 signals [78].
   - User fatigue or distraction [79].
   - Poorly optimized flashing parameters [80].

6.4.3 UNO Card Game: Performance Evaluation

To evaluate the efficacy of the P300-based BCI system integrated with the Uno Card Game, a series of experimental trials were conducted. The primary objective was to quantify the system's performance in helping users navigate the game buttons using brain signals [69].

Success Rate
The Success Rate is defined as the percentage of trials in which the user successfully selected the correct game button — including *DrawCard*, *Left*, *Right*, and *PlaceCard* — using the BCI system.
Formula
Success Rate = (Number of Successful Trials / Total Number of Trials) × 100%
Influencing Factors
- **User Concentration:** Focus is critical for evoking reliable P300 signals during button flashing [70].
- **BCI Model Accuracy:** A robust classifier improves decision precision [71].
- **Game Design Parameters:** Flash timing and sequence structure significantly affect signal quality and user comfort.
- **Uno Complexity:** Variations in game logic and visual stimuli can increase user cognitive load, influencing attention and success rate [73].

Error Rate Metrics
1. **Misclassification Errors**
   Occur when the system predicts the wrong button due to misinterpreted P300 signals.
   Formula:
   Misclassification Error Rate = (Misclassified Button Selections / Total Button Selections) × 100%.
   Contributing Factors:
   - Low signal-to-noise ratio in EEG [74].
   - Individual differences in brainwave patterns [75].
   - Artifacts from blinking, movement, or external noise [76].
2. **Time-out Errors**
   Occur when the system fails to make a prediction within the allowed time frame.

Formula:
Time-out Error Rate = (Number of Time-outs / Total Expected Predictions) × 100%.
Possible Causes:

- Weak or ambiguous P300 responses [78].
- User fatigue or loss of focus [79].
- Poorly calibrated flash timing settings [80].

6.4.4 Conclusion

The combination of educational tasks, maze navigation, and Uno gameplay validated the usability and adaptability of the BCI-controlled platform. Across different interactive experiences, users were able to complete tasks using only brain signals. Quantitative metrics like success rate and error analysis provided essential feedback for refining system performance. The results highlight the promise of P300-based BCI systems in delivering inclusive, engaging, and assistive technologies for individuals with physical challenges.

6.5 BUSINESS MODEL CANVAS

A Business Model Canvas (BMC) provides a strategic management template for developing new or documenting existing business models. For a P300-based BCI system, a potential BMC could look like this:

- **Key Partners:** EEG headset manufacturers (e.g., Unicorn Hybrid Black), software developers, research institutions, hospitals and rehabilitation centers, disability advocacy groups.

- **Key Activities:** Research and development (algorithm improvement, new applications), software development and maintenance, user testing and feedback collection, marketing and sales, customer support and training.

- **Key Resources:** Intellectual property (P300 detection algorithms, application software), skilled personnel (engineers, researchers, therapists), EEG hardware, funding.

- **Value Propositions:**

    - For handicapped users: Enhanced communication, increased independence, access to education and recreation, improved quality of life.
    - For therapists/caregivers: A new tool for rehabilitation and assistive care.
    - For educational institutions: Inclusive learning tools.

- **Customer Relationships:** Direct sales, personalized support and training, online community/forum, feedback channels.

- **Channels:** Direct sales team, website and online store, partnerships with healthcare providers and assistive technology distributors, academic conferences and publications.

- **Customer Segments:** Individuals with severe motor disabilities (e.g., ALS, spinal cord injury, locked-in syndrome), rehabilitation centers, special education schools, research institutions.

- **Cost Structure:** R&D expenses, salaries, hardware costs, marketing and sales expenses, operational costs.

- **Revenue Streams:** Sales of the BCI software, licensing of the technology, subscription fees for advanced features or content, custom development services, grants and research funding.

6.6 TARGET USERS AND DEPLOYMENT SCOPE

Target Users: The primary target users for this P300-based BCI system are individuals with severe motor impairments who have difficulty with or are unable to use traditional assistive communication and control devices. This includes, but is not limited to, individuals with:

- Amyotrophic Lateral Sclerosis (ALS)

- Spinal Cord Injuries (SCI)

- Locked-in Syndrome

- Severe Cerebral Palsy

- Other neuromuscular disorders

Secondary users could include therapists, caregivers, and educators who work with these individuals and can utilize the BCI system as a tool for communication, learning, and recreation.

Deployment Scope:

- Initial Phase (Prototype and Pilot Testing): Deployment will initially be focused on controlled environments, such as research labs, rehabilitation centers, or homes of selected pilot users. This phase is crucial for gathering user feedback, refining the system, and validating its performance in real-world scenarios.

- Second Phase (Limited Release): Following successful pilot testing, a limited release could target specific clinics, hospitals, or assistive technology centers that specialize in working with the target user population. This allows for broader testing and the development of support and training materials.

- Third Phase (Commercial Release): Depending on regulatory approvals (if pursued as a medical device) and market readiness, a wider commercial release could be considered. This would involve establishing sales channels, customer support infrastructure, and ongoing product development based on broader market feedback.

The deployment scope will also depend on factors such as the cost of the system (including the EEG headset), the ease of setup and use, and the availability of training and technical support. The ultimate goal is to make the technology as accessible as possible to those who can benefit from it.

7 CHAPTER 7: CONCLUSION AND FUTURE WORK

7.1 SUMMARY OF ACHIEVEMENTS

This project successfully developed and demonstrated a P300-based Brain-Computer Interface (BCI) system designed to enhance communication and interaction for individuals with severe motor disabilities. Key achievements include:

- **Robust P300 Detection:** Implemented and refined algorithms for accurate and timely detection of P300 event-related potentials from EEG signals acquired via the Hybrid Unicorn Black Headset.

- **Application Integration:** Successfully integrated the BCI control with diverse applications, including a basic educational website, an advanced learning platform, and two custom-developed BCI-controlled games (UNO and Maze).

- **User-Centric Design:** Prioritized user-friendly interfaces and accessibility features across all applications, ensuring intuitive interaction for the target user group.

- **Dataset Integration:** Incorporated a collected dataset for training and testing, demonstrating the practical applicability of the system.

These achievements collectively demonstrate the feasibility and potential of the developed BCI system as a valuable assistive technology, offering a new avenue for independence and engagement for special needs individuals.

7.2 LIMITATIONS OF THE CURRENT SYSTEM

Despite the successful development, the current BCI system has several limitations that warrant consideration and future work:

1. Signal Quality & Noise

- **EEG Signal Noise**: Despite advanced preprocessing, residual noise persists due to physiological and environmental interference.
- **Artifact Sensitivity**: Performance degrades in real-world environments due to motion artifacts and muscle activity.

2. Performance & Usability

- **Information Transfer Rate (ITR)**: The P300 speller's sequential flashing paradigm limits selection speed, restricting rapid communication.
- **User Training Requirement**: Extensive training is needed for stable BCI operation.
- **Lack of Long-Term Studies**: No longitudinal data on fatigue.

3. Hardware and Dataset constraints

- **Hardware Dependency**: Tied to the *Hybrid Unicorn Black Headset*'s specifications (e.g., sampling rate).
- **Dataset Limitations**:

- **Small Sample Size**: Insufficient for deep learning (e.g., CNNs), which typically require orders of magnitude more data for robust performance.

- **Design of interface constraints & user BCI illiteracy :**
    - **User:**Repeated practice sessions on the BCI system are an essential component of the learning curve prior to data collection, ensuring the user's familiarity with the system interface. Insufficient practice or lack of focus during this stage may compromise the user's ability to maintain a stable focal point, 15-30% of individuals lack ability to control a BCI system in a correct way, this is known as BCI illiteracy. [92]
    - **Design:** presentation of design (buttons) are not properly separated.

4. Methodological Trade-offs

- **EEG vs. ECOG / MEG**:
    - **Advantages**: EEG was selected for its non-invasiveness, affordability, portability, and high temporal resolution, making it ideal for BCIs and cognitive neuroscience.
    - **Disadvantages**: Lower spatial resolution compared to ECOG / MEG, and higher noise susceptibility.

Addressing these limitations will be crucial for transitioning the BCI system from a research prototype to a widely adopted assistive technology.

7.3 ETHICAL CONSIDERATIONS

1. Physical and Electrical Safety

- **Device Compliance**:
    - The Unicorn Hybrid Black headset is certified for medical safety, but requires:
        - Proper hygiene protocols (e.g., cleaning electrodes between users).
        - Inspection for damaged cables or exposed wiring to prevent electrical hazards.
    - **Usage Restrictions**:
        - Contraindicated for users with open wounds, skin irritation, or in wet environments.
        - Limited session durations to avoid skin irritation from prolonged electrode contact.

2. Psychological Safety

- **User Fatigue**:

    - P300-based systems (e.g., games) induce cognitive load due to sustained attention demands.
    - **Mitigations**:
        - Mandatory breaks during sessions (e.g., 10-minute breaks every 30 minutes).
        - Visual ergonomics: Flicker-free stimuli and adjustable pacing to reduce eyestrain.
        - "Exit anytime" feature to prevent frustration from unsuccessful trials.
- **Feedback Design**:
    - Avoid negative reinforcement (e.g., error sounds); prioritize neutral/positive feedback.

7.4 PROPOSED ENHANCEMENTS

Based on the identified limitations and opportunities for improvement, several enhancements are proposed for future iterations of the BCI system:

- **Adaptive Algorithms:** Develop and implement adaptive P300 detection and classification algorithms that can automatically adjust to individual user characteristics and changes in brain activity over time, reducing the need for extensive calibration.

- **Advanced Artifact Handling:** Explore more sophisticated real-time artifact removal techniques, potentially incorporating machine learning models trained on diverse artifact types, to improve signal quality in dynamic environments for better results.

- **Hybrid BCI Paradigms:** Investigate the integration of the P300 paradigm with other BCI paradigms (e.g., SSVEP, motor imagery) to create hybrid systems that offer higher ITR, more control options, and increased robustness.

This project serves as a foundational step, highlighting the immense potential of BCI technology to transform the lives of individuals with disabilities and paving the way for future innovations in human-computer interaction.

REFERENCES

[1] Farwell, L. A., & Donchin, E. (1988). Talking off the top of your head: Toward a mental prosthesis utilizing event-related brain potentials. Electroencephalography and Clinical Neurophysiology , 70(6), 510–523.

[2] Sellers, E. W., Krusienski, D. J., McFarland, D. J., Vaughan, T. M., & Wolpaw, J. R. (2006). A P300 event-related potential brain–computer interface (BCI): The effects of matrix size and inter stimulus interval on performance. Biological Psychology , 73(3), 242–252.

[3] Luck, S. J. (2014). An introduction to the event-related potential technique (2nd ed.). MIT Press.

[4] Hochberg, L. R., Serruya, M. D., Friehs, G. M., Mukand, J. A., Saleh, M., Caplan, A. H., ... & Donoghue, J. P. (2006). Neuronal ensemble control of prosthetic devices by a human with tetraplegia. Nature , 442(7099), 164–171.

[5] Unicorn Hybrid Black User Manual (2023). g.tec medical engineering GmbH .

[6] Wolpaw, J. R., & Wolpaw, E. W. (2012). Brain–computer interfaces: Principles and practice . Oxford University Press.

[7] Nijboer, F., Sellers, E. W., Mellinger, J., Jordan, M. A., Matuz, T., Furdea, A., ... & Kübler, A. (2008). A P300-based brain–computer interface for people with amyotrophic lateral sclerosis. Clinical Neurophysiology , 119(8), 1909–1916.

[8] Blankertz, B., Tomioka, R., Lemm, S., Kawanabe, M., & Müller, K. R. (2008). Optimizing spatial filters for robust EEG single-trial analysis. IEEE Signal Processing Magazine , 25(1), 41- 56.

[9] Donchin, E., & Coles, M. G. (1988). Is the P300 component a manifestation of context updating? Behavioral and Brain Sciences , 11(3), 357–374.

[10] Makeig, S., Delorme, A., Westerfield, M., Townsend, J., Courchesne, E., & Jung, T. P. (2004). Electroencephalographic brain dynamics following manually responded visual targets. PLoS Biology , 2(6), e176.

[11] Duncan-Johnson, C. C., & Donchin, E. (1977). On quantifying the amount of information conveyed by the P300 component of the event-related brain potential. Perception & Psychophysics , 22(4), 385–391.

[12] Polich, J. (2007). Updating P300: An integrative theory of P3a and P3b. Clinical Neurophysiology , 118(10), 2128–2148.

[13] Blankertz, B., Tomioka, R., Lemm, S., Kawanabe, M., & Müller, K. R. (2008). Optimizing spatial filters for robust EEG single-trial analysis. IEEE Signal Processing Magazine , 25(1), 41 56.

[14] Urigüen, J. A., & Garcia-Zapirain, B. (2015). EEG artifact removal—State-of-the-art and guidelines. Computational Intelligence and Neuroscience , 2015, 1–19.

[15] L. A. Farwell and E. Donchin, “Talking off the top of your head: Toward a mental prosthesis utilizing event-related brain potentials,” Electroencephalogr. Clin. Neurophysiol., vol. 70, no. 6, pp. 510–523, Jun. 1988.

[16] Schröder, M., Lal, T. N., Hinterberger, T., Bogdan, M., Hill, N., Birbaumer, N., ... & Rosenstiel, W. (2003). Robust EEG channel selection across subjects for brain-computer interfaces. EURASIP Journal on Advances in Signal Processing , 2003, 1–10.
[17] Lawhern, V. J., Solon, A. J., Waytowich, N. R., Gordon, S. M., Hung, C. P., & Lance, B. J. (2018). EEGNet: A compact convolutional network for EEG-based brain–computer interfaces. Journal of Neural Engineering , 15(5), 056013.
[18] R. Oostenveld, "Review of the Unicorn Hybrid Black 8-channel EEG system," robertoostenveld.nl, Jun. 19, 2022.
[19] MindGarden AI, "Building an Advanced Real-Time EEG Analysis App with Python and BrainFlow," MindGarden, Sep. 27, 2024.
[20] C. G. Coogan, "Brain-computer interface control in a virtual reality environment," Journal of Neural Engineering, vol. 15, no. 5, p. 056003, 2018.
[21] W3C Web Accessibility Initiative, "Web Content Accessibility Guidelines (WCAG) 2.1," 2021.
[22] B. Shneiderman et al., Designing the User Interface: Strategies for Effective Human-Computer Interaction, 6th ed., Pearson, 2016.
[23] G. R. Müller-Putz et al., "Towards noninvasive brain-computer interfaces for severely disabled people," in Brain-Computer Interface Research, vol. 2, pp. 37–50. Springer, 2015.
[24] B. Blankertz et al., "The Berlin Brain–Computer Interface: Non-medical uses of BCI technology," Frontiers in Neuroscience, vol. 4, p. 198, 2010.
[25] C. Cinel et al., "Neuroadaptive technology enables implicit cursor control based on medial prefrontal cortex activity," Proceedings of the National Academy of Sciences, vol. 116, no. 21, pp. 10512–10517, 2019.
[26] L. M. Alonso-Valerdi et al., "BCI-based neuro-rehabilitation of speech and language," Frontiers in Neuroscience, vol. 9, p. 482, 2015.
[27] J. Li et al., "Feature selection: A data perspective," arXiv preprint arXiv:1601.07996, 2016.
[28] Gramfort, A., Luessi, M., Larson, E., Engemann, D. A., Strohmeier, B., Brodbeck, C., ... & Hämäläinen, M. S. (2013). MNE-Python: A software package for electrophysiology research. Frontiers in Neuroscience, 7, 267.
[28] "pylsl - Lab Streaming Layer for Python," PyPI.
[29] Unity Technologies, "WaitWhile," Unity Documentation.
[30] Unity Technologies, "Button.onClick," Unity Documentation, 2018.3.
[31] Unity Technologies, "Application.Quit," Unity Documentation, Version 6000.0.
[32] Unity Technologies, "Transform.localScale," Unity Documentation.
[33] Unity Technologies, "Transform," Unity Documentation.
[34] Unity Technologies, "Object.Instantiate," Unity Documentation, Version 6000.0.
[35] Unity Community, "Reset the time.time back to 0," Unity Discussions, 2024.
[36] Unity Community, "Button Click and Enumerator (Solved)," Unity Discussions, 2024.
[37] Unity Community, "Conditions on Buttons," Unity Discussions, 2024.
[38] Unity Community, "How to Make Update Function Start After Delay," Unity Discussions, 2024.

[39] Stack Overflow Community, "How to Empty an Array in C#," Stack Overflow, 2024.
[40] Stack Overflow Community, "Get Index in foreach Loop," Stack Overflow, 2024.
[41] Quora Community, "Check if a Button is Clicked in Unity," Quora, 2024.
[42] Brackeys, "Unity 2D Card Game Tutorial," YouTube, 2024.
[43] The Liquid Fire, "Make a CCG Intro," The Liquid Fire, Aug. 21, 2017.
[44] Microsoft, "List.Clear Method," Microsoft Docs, 2024.
[45] ByteHide, "Array vs List in C#," ByteHide Blog, 2024.
[46] A. Author(s), "Automatic classification of EEG signals, based on image interpretation," medRxiv, Feb. 10, 2025.
[47] A. Author(s), "EEG artifact detection and correction with deep autoencoders," arXiv preprint, Feb. 2025.
[48] A. Author(s), "EEG baseline analysis for effective extraction of P300 event-related potentials," PubMed, 2024.
[49] A. Author(s), "Detection analysis of epileptic EEG using a novel random forest model," PMC, 2019.
[50] A. Author(s), "BCIAUT-P300: A multi-session and multi-subject benchmark dataset on P300-based BCI," PMC, 2020.
[51] A. Author(s), "EEG-based identification system using deep neural networks with channel attention," PMC, 2024.
[52] A. Author(s), "Analysis of frequency domain features for the classification of EEG signals," PubMed, 2024.
[53] A. Author(s), "AnEEG: Leveraging deep learning for effective artifact removal in EEG," PMC, 2025. /
[54] A. Author(s), "EEG-based emotion recognition systems: A comprehensive study," PMC, 2025.
[55] A. Author(s), "Deep-EEG: An optimized and robust framework and method for EEG classification," PMC, 2023.
[56] A. Author(s), "Threats and mitigation strategies for electroencephalography-based person authentication," PMC, 2025.
[57] A. Author(s), "Real-time motor imagery classification using CNN-LSTM for BCI systems," arXiv preprint, 2024.
[58] A. Author(s), "EEGNet: A compact convolutional network for EEG-based BCIs," Journal of Neural Engineering, vol. 15, no. 5, 2018.
[59] L. A. Farwell and E. Donchin, “Talking off the top of your head: Toward a mental prosthesis utilizing event-related brain potentials,” Electroencephalography and Clinical Neurophysiology, vol. 70, no. 6, pp. 510–523, Dec. 1988.
[60] G. R. Müller-Putz, R. Scherer, C. Neuper, and G. Pfurtscheller, “Steady-state visual evoked potential (SSVEP)-based communication: impact of harmonic frequency components,” Biological Psychology, vol. 75, no. 3, pp. 241–250, Mar. 2007.
[61] M. S. Fazli et al., “EEG-based brain–computer interfaces for non-invasive communication and rehabilitation,” Frontiers in Neuroscience, vol. 12, p. 784, 2018.
[62] D. J. McFarland, L. M. McCane, S. V. David, and J. R. Wolpaw, “Spatial filter selection for EEG-based communication,” in Proc. IEEE Trans. Neural Syst. Rehabil. Eng., vol. 11, no. 3, pp. 324–325, Sep. 2003.

[63] B. Shneiderman and C. Plaisant, Designing the User Interface: Strategies for Effective Human-Computer Interaction, 5th ed. Pearson, 2005.
[64] J. Höhne and B. Blankertz, "Towards an Auditory Attention BCI," in Foundations of Augmented Cognition. Neuroergonomics and Operational Neuroscience, vol. 9183, D. D. Schmorrow and C. M. Fidopiastis, Eds. Springer, 2015, pp. 57–68.
[65] M. Tangermann et al., "The Berlin Brain–Computer Interface: Non-Medical Uses of BCI Technology," Frontiers in Neuroscience, vol. 4, p. 198, 2010.
[66] N. R. Carlson et al., "Brain-to-brain communication in humans: A proof-of-concept study," Proceedings of the National Academy of Sciences, vol. 116, no. 37, pp. 18770–18775, Sep. 2019.
[67] M. Riccio et al., "Attention and P300-based BCI performance in people with Amyotrophic Lateral Sclerosis," Frontiers in Neuroscience, vol. 9, p. 482, 2015.
[68] World Wide Web Consortium (W3C), "Web Content Accessibility Guidelines (WCAG) 2.1," Jun. 2018.
[69] J. R. Wolpaw et al., "Brain–computer interfaces for communication and control," Clin. Neurophysiol., vol. 113, no. 6, pp. 767–791, 2002.
[70] G. Pfurtscheller and C. Neuper, "Motor imagery and direct brain-computer communication," Proc. IEEE, vol. 89, no. 7, pp. 1123–1134, Jul. 2001.
[71] M. Kaper, P. Meinicke, U. Grossekathoefer, T. Lingner, and H. Ritter, "BCI competition 2003—Data set IIb: Support vector machines for the P300 speller paradigm," IEEE Trans. Biomed. Eng., vol. 51, no. 6, pp. 1073–1076, Jun. 2004.
[72] S. G. Mason and G. E. Birch, "A brain-controlled switch for asynchronous control applications," IEEE Trans. Biomed. Eng., vol. 47, no. 10, pp. 1297–1307, Oct. 2000.
[73] F. Lotte et al., "A review of classification algorithms for EEG-based brain–computer interfaces: A 10 year update," J. Neural Eng., vol. 15, no. 3, p. 031005, Jun. 2018.
[74] M. Fatourechi et al., "EMG and EOG artifacts in brain computer interface systems: A survey," Clin. Neurophysiol., vol. 118, no. 3, pp. 480–494, Mar. 2007.
[75] G. Townsend et al., "A novel P300-based brain–computer interface stimulus presentation paradigm: Moving beyond rows and columns," Clin. Neurophysiol., vol. 121, no. 7, pp. 1109–1120, Jul. 2010.
[76] C. M. A. Kellner, E. Halder, and A. Kübler, "A P300-based brain–computer interface for people with amyotrophic lateral sclerosis," Clin. Neurophysiol., vol. 120, no. 3, pp. 548–555, Mar. 2009.
[77] L. F. Nicolas-Alonso and J. Gomez-Gil, "Brain computer interfaces, a review," Sensors, vol. 12, no. 2, pp. 1211–1279, Feb. 2012.
[78] B. Z. Allison and J. J. Neuper, "Could anyone use a BCI?" in Brain–Computer Interfaces, Springer, London, 2010, pp. 35–54.
[79] A. Käthner et al., "Effects of mental workload and fatigue on the P300, alpha and theta band power during operation of an ERP (P300) brain–computer interface," Biol. Psychol., vol. 102, pp. 118–129, Jan. 2014.
[80] D. J. Krusienski et al., "Toward enhanced P300 speller performance," J. Neurosci. Methods, vol. 167, no. 1, pp. 15–21, Jan. 2008.
[81] G. Edlinger, C. Holzner, and C. Guger, "A hybrid brain-computer interface for smart home control," Cognit. Comput., vol. 4, no. 4, pp. 388–395, Dec. 2012.

[82] C. Guger et al., “How many people are able to control a P300-based brain–computer interface (BCI)?,” Neurosci. Lett., vol. 462, no. 1, pp. 94–98, Sep. 2009.
[83] M. van Gerven et al., “The brain–computer interface cycle,” J. Neural Eng., vol. 6, no. 4, p. 041001, Aug. 2009.
[84] D. J. McFarland and J. R. Wolpaw, “Brain–computer interfaces for communication and control,” Commun. ACM, vol. 54, no. 5, pp. 60–66, May 2011.
[85] A. Kübler and K. R. Müller, “An introduction to brain–computer interfacing,” in Toward Brain–Computer Interfacing, MIT Press, 2007, pp. 1–25.
[86] A. Nijholt et al., “Brain–computer interfaces for HCI and games,” in CHI ‘08 Extended Abstracts on Human Factors in Computing Systems, ACM, 2008, pp. 3925–3928.
[87] B. Z. Allison, T. E. Peters, and C. Neuper, “BCIs for assessment and communication in locked-in syndrome,” in Brain–Computer Interfaces: Applying Our Minds to Human-Computer Interaction, Springer, 2012, pp. 131–146.
[88] A. Holz et al., “Brain–computer interface controlled gaming: Evaluation of usability by severely motor restricted end-users,” Artif. Intell. Med., vol. 59, no. 2, pp. 111–120, Oct. 2013.
[88] Amzica, F., & Lopes da Silva, F. H. (2017). Cellular substrates of brain rhythms. In D. L. Schomer & F. H. Lopes da Silva (Eds.), Niedermeyer's Electroencephalography: Basic Principles, Clinical Applications, and Related Fields (Vol. 1). Oxford University Press.
[89] Cleveland Clinic. (n.d.). Brain: Parts, function, how it works & conditions.
[90] Hopkins Medicine. (n.d.). Brain anatomy and how the brain works. Johns Hopkins Medicine.
[91] Maldonado, K. A., & Alsayouri, K. (2023, March 17). Physiology, Brain. In StatPearls.
[92] S. Becker, K. Dhindsa, L. Mousapour and Y. Al Dabagh, "BCI Illiteracy: It’s Us, Not Them. Optimizing BCIs for Individual Brains," *2022 10th International Winter Conference on Brain-Computer Interface (BCI)*, Gangwon-do, Korea, Republic of, 2022, pp. 1-3