

# NeuroAd: EEG-Driven AI for Ad Engagement Prediction



ANEEZA BATOOL

**01-135221-007**

ARSAL AYYAN

**01-135221-060**

Supervisor: Ma'am Ameena Saeed

**Bachelor of Science in Computer Science**

Department of Computer Science  
Bahria University, Islamabad

October 2025

# Certificate

We accept the work contained in the report titled “NeuroAd: EEG-Driven AI for Ad Engagement Prediction”, written by Aneeza Batool (01-135221-007) and Arsal Ayyan (01-135221-060) as a confirmation to the required standard for the partial fulfillment of the degree of Bachelor of Science in Computer Science.

Approved by ....:

Supervisor: Ma'am Ameena Saeed

Internal Examiner:

External Examiner:

Project Coordinator: Dr. Kabeer Ahmed Bhatti

Head of the Department: Dr. Khurram Ehsan

# Abstract

NeuroAd is a web-based platform designed to objectively evaluate advertisement engagement by analyzing brainwave data captured from viewers using a 14-channel EMOTIV EPOC headset. The platform processes EEG signals to extract cognitive and emotional engagement indicators from frequency-based features, which are then classified using a Random Forest machine learning model to label advertisements as engaging or non-engaging with high accuracy. By measuring subconscious neural responses in real time, NeuroAd addresses the limitations of traditional survey-based marketing research. It offers a secure, role-based web interface where authorized users can upload EEG datasets, run engagement analyses, and access detailed interactive reports. By combining neuroscience, advanced data processing, and web technologies, NeuroAd provides an accessible, scalable, and cost-effective neuromarketing solution that enables advertisers to design more impactful advertisements based on real neural activity rather than subjective feedback.

# Acknowledgments

We would like to express our sincere gratitude to our supervisor, **Ma'am Ameenah Saeed**, for her continuous guidance, encouragement, and valuable feedback throughout the development of our project titled "*NeuroAd: EEG-Driven AI for Ad Engagement Prediction*." Her expertise and insightful suggestions have been instrumental in shaping this research and ensuring its successful completion.

We also extend our appreciation to the **Centre of Excellence in Artificial Intelligence (CoE-AI)** for providing us with the **Emotiv Epoc EEG headset**, which served as the primary tool for our study. The resources and technical support made available by CoE-AI played a vital role in enabling us to conduct our experiments effectively.

Lastly, we are thankful to all the participants who volunteered to take part in our EEG testing sessions, especially our fellow classmates, for their time and cooperation. Their willingness to contribute was essential in helping us collect accurate data and achieve the objectives of this project.

ANEEZA BATOOL  
ARSAL AYYAN  
Bahria University  
E-8, Islamabad, Pakistan

October 2025

# Contents

|                                                                                                             |           |
|-------------------------------------------------------------------------------------------------------------|-----------|
| <b>Abstract</b>                                                                                             | <b>i</b>  |
| <b>1 Introduction</b>                                                                                       | <b>1</b>  |
| 1.1 Project Background . . . . .                                                                            | 1         |
| 1.2 Problem Description . . . . .                                                                           | 2         |
| 1.3 Project Objectives . . . . .                                                                            | 2         |
| 1.4 Project Scope . . . . .                                                                                 | 3         |
| <b>2 Literature Review</b>                                                                                  | <b>4</b>  |
| 2.1 Existing Literature . . . . .                                                                           | 4         |
| 2.1.1 Engagement Estimation in Advertisement Videos with EEG . . .                                          | 4         |
| 2.1.2 Using Support Vector Machine on EEG for Advertisement Impact<br>Assessment . . . . .                  | 4         |
| 2.1.3 Discrete Classification Technique Applied to TV Advertisements<br>Liking Recognition System . . . . . | 4         |
| 2.2 Industry Applications . . . . .                                                                         | 5         |
| 2.2.1 Nielsen NeuroFocus . . . . .                                                                          | 5         |
| 2.2.2 Neuro-Insight . . . . .                                                                               | 5         |
| 2.3 Research Gaps and Motivation . . . . .                                                                  | 5         |
| <b>3 Requirement Specifications</b>                                                                         | <b>6</b>  |
| 3.1 Existing System . . . . .                                                                               | 6         |
| 3.2 Proposed System . . . . .                                                                               | 7         |
| 3.3 Requirement Specifications . . . . .                                                                    | 7         |
| 3.3.1 Functional Requirements . . . . .                                                                     | 7         |
| 3.3.2 Non-Functional Requirements . . . . .                                                                 | 9         |
| 3.4 Use Case . . . . .                                                                                      | 11        |
| 3.5 Descriptive Use Cases . . . . .                                                                         | 13        |
| 3.5.1 General User . . . . .                                                                                | 13        |
| 3.5.2 Ad Testing Personnel . . . . .                                                                        | 15        |
| 3.5.3 Admin . . . . .                                                                                       | 20        |
| <b>4 Design</b>                                                                                             | <b>27</b> |
| 4.1 System Architecture . . . . .                                                                           | 27        |
| 4.2 Design Constraints . . . . .                                                                            | 27        |
| 4.2.1 Input Limitations . . . . .                                                                           | 27        |
| 4.2.2 Platform Dependency . . . . .                                                                         | 28        |

|          |                                                        |           |
|----------|--------------------------------------------------------|-----------|
| 4.2.3    | Insights Scope . . . . .                               | 28        |
| 4.2.4    | Real-Time Tracking . . . . .                           | 28        |
| 4.2.5    | Integration with External Platforms . . . . .          | 28        |
| 4.2.6    | Advanced Sentiment Analysis . . . . .                  | 28        |
| 4.2.7    | User Access . . . . .                                  | 28        |
| 4.3      | Design Methodology . . . . .                           | 28        |
| 4.3.1    | Methodology Overview . . . . .                         | 29        |
| 4.3.2    | Development Phases . . . . .                           | 29        |
| 4.4      | High Level Design . . . . .                            | 29        |
| 4.4.1    | Conceptual / Logical View . . . . .                    | 29        |
| 4.4.2    | Process View . . . . .                                 | 30        |
| 4.4.3    | Physical View . . . . .                                | 30        |
| 4.4.4    | Module View . . . . .                                  | 31        |
| 4.4.5    | Security View . . . . .                                | 31        |
| 4.4.6    | Sequence Diagram . . . . .                             | 32        |
| 4.5      | Low Level Design . . . . .                             | 34        |
| 4.5.1    | Frontend Module . . . . .                              | 34        |
| 4.5.2    | Backend Module . . . . .                               | 35        |
| 4.6      | Database Design . . . . .                              | 35        |
| 4.7      | GUI Design . . . . .                                   | 36        |
| 4.8      | External Interfaces . . . . .                          | 46        |
| <b>5</b> | <b>System Implementation</b>                           | <b>48</b> |
| 5.1      | System Architecture . . . . .                          | 48        |
| 5.2      | System Internal Components . . . . .                   | 49        |
| 5.3      | Functionality of Components . . . . .                  | 49        |
| 5.3.1    | Frontend . . . . .                                     | 49        |
| 5.3.2    | Backend . . . . .                                      | 50        |
| 5.3.3    | Database (Firebase) . . . . .                          | 50        |
| 5.3.4    | Random Forest Model . . . . .                          | 50        |
| 5.4      | Tools and Technologies . . . . .                       | 50        |
| 5.4.1    | Hardware Tools . . . . .                               | 50        |
| 5.4.2    | Software Tools . . . . .                               | 51        |
| 5.4.3    | Machine Learning & Data Processing Libraries . . . . . | 52        |
| 5.4.4    | Backend Framework Technologies . . . . .               | 53        |
| 5.4.5    | Frontend Visualization Tools . . . . .                 | 53        |
| 5.4.6    | Version Control Collaboration . . . . .                | 53        |
| 5.4.7    | Deployment Environment . . . . .                       | 53        |
| 5.5      | Security . . . . .                                     | 54        |
| 5.5.1    | Authentication and Access Control . . . . .            | 54        |
| 5.5.2    | Data Security . . . . .                                | 55        |
| 5.5.3    | Backend and API Protection . . . . .                   | 55        |
| 5.5.4    | System Monitoring and Logging . . . . .                | 55        |

|          |                                                     |           |
|----------|-----------------------------------------------------|-----------|
| <b>6</b> | <b>System Testing and Evaluation</b>                | <b>57</b> |
| 6.1      | Functional Testing . . . . .                        | 57        |
| 6.1.1    | Graphical User Interface (GUI) Testing . . . . .    | 57        |
| 6.1.2    | Backend and Model Testing . . . . .                 | 69        |
| 6.1.3    | Workflow Testing . . . . .                          | 72        |
| 6.2      | Non-Functional Testing . . . . .                    | 79        |
| 6.2.1    | Performance Testing . . . . .                       | 79        |
| 6.2.2    | Security Testing . . . . .                          | 87        |
| <b>7</b> | <b>Conclusion</b>                                   | <b>92</b> |
| <b>A</b> | <b>User Manual</b>                                  | <b>93</b> |
| A.1      | User Roles and Access . . . . .                     | 93        |
| A.1.1    | General User . . . . .                              | 93        |
| A.1.2    | Ad Testing Personnel . . . . .                      | 93        |
| A.1.3    | Admin . . . . .                                     | 93        |
| A.2      | Getting Started . . . . .                           | 94        |
| A.2.1    | Access and Login . . . . .                          | 94        |
| A.3      | Key Features and Usage . . . . .                    | 94        |
| A.3.1    | Uploading EEG Data (Ad Testing Personnel) . . . . . | 94        |
| A.3.2    | Viewing Results . . . . .                           | 94        |
| A.3.3    | Managing Users and Ads (Admin Only) . . . . .       | 95        |
| A.4      | Troubleshooting and Support . . . . .               | 95        |
|          | <b>References</b>                                   | <b>96</b> |

# List of Figures

|      |                                                          |    |
|------|----------------------------------------------------------|----|
| 3.1  | General User Use Case . . . . .                          | 11 |
| 3.2  | Tester Use Case . . . . .                                | 11 |
| 3.3  | Admin Use Case . . . . .                                 | 12 |
| 4.1  | High Level System Architecture . . . . .                 | 27 |
| 4.2  | Development Phases . . . . .                             | 29 |
| 4.3  | Conceptual Diagram . . . . .                             | 30 |
| 4.4  | Process View . . . . .                                   | 30 |
| 4.5  | Physical View of NeuroAd . . . . .                       | 31 |
| 4.6  | Full Module View . . . . .                               | 31 |
| 4.7  | Security View . . . . .                                  | 32 |
| 4.8  | Employee Sequence Diagram . . . . .                      | 32 |
| 4.9  | Admin Sequence Diagram . . . . .                         | 33 |
| 4.10 | Full Sequence Diagram . . . . .                          | 34 |
| 4.11 | Frontend Module . . . . .                                | 35 |
| 4.12 | Backend Module . . . . .                                 | 35 |
| 4.13 | Database Module . . . . .                                | 36 |
| 4.14 | Home Page . . . . .                                      | 37 |
| 4.15 | Model Page . . . . .                                     | 38 |
| 4.16 | About Page . . . . .                                     | 39 |
| 4.17 | Contact Page . . . . .                                   | 40 |
| 4.18 | Login Page . . . . .                                     | 40 |
| 4.19 | Admin Dashboard . . . . .                                | 41 |
| 4.20 | Admin Dashboard Dark Mode . . . . .                      | 42 |
| 4.21 | Upload Page . . . . .                                    | 43 |
| 4.22 | Results Page . . . . .                                   | 44 |
| 4.23 | Logs Page . . . . .                                      | 45 |
| 4.24 | Analytics Page . . . . .                                 | 46 |
| 4.25 | External Interfaces . . . . .                            | 47 |
| 5.1  | System Architecture Flow . . . . .                       | 49 |
| 5.2  | EEG Headset . . . . .                                    | 51 |
| 5.3  | Raw EEG Signal Visualization in EmotivPRO . . . . .      | 52 |
| 5.4  | Performance Metrics Visualization in EmotivPRO . . . . . | 52 |
| 5.5  | Tools and Technologies Architecture . . . . .            | 54 |
| 5.6  | Secure System Architecture Overview . . . . .            | 56 |

# List of Tables

|      |                                                                                 |    |
|------|---------------------------------------------------------------------------------|----|
| 3.1  | Functional Requirements . . . . .                                               | 7  |
| 3.2  | Non-Functional Requirements . . . . .                                           | 9  |
| 3.3  | Use Case UC-1.1: View Website Pages . . . . .                                   | 13 |
| 3.4  | Use Case UC-1.2: Submit Contact Inquiry . . . . .                               | 14 |
| 3.5  | Use Case UC-2.1: Employee Login . . . . .                                       | 15 |
| 3.6  | Use Case UC-2.2: Dashboard Insights . . . . .                                   | 16 |
| 3.7  | Use Case UC-2.3: Upload CSV & Predict Engagement . . . . .                      | 17 |
| 3.8  | Use Case UC-2.4: View Results & Download PDF Report . . . . .                   | 18 |
| 3.9  | Use Case UC-3.1: Admin Login . . . . .                                          | 20 |
| 3.10 | Use Case UC-3.2: Manage Users . . . . .                                         | 21 |
| 3.11 | Use Case UC-3.3: View Activity Logs & Notifications . . . . .                   | 22 |
| 3.12 | Use Case UC-3.4: View AI Insights . . . . .                                     | 23 |
| 3.13 | Use Case UC-3.5: View Logs . . . . .                                            | 24 |
| 3.14 | Use Case UC-3.6: View Analytics . . . . .                                       | 25 |
| 6.1  | GUI Test Case 1: Home Page Verification . . . . .                               | 58 |
| 6.2  | GUI Test Case 2: Contact Us Page Verification . . . . .                         | 60 |
| 6.3  | GUI Test Case 3: Model Page Verification . . . . .                              | 61 |
| 6.4  | GUI Test Case 4: Employee Dashboard GUI Verification . . . . .                  | 63 |
| 6.5  | GUI Test Case 5: Admin Dashboard GUI Verification . . . . .                     | 64 |
| 6.6  | GUI Test Case 6: Contact Us Form – Invalid Inputs . . . . .                     | 65 |
| 6.7  | GUI Test Case 7: Login Form – Invalid Credentials . . . . .                     | 67 |
| 6.8  | GUI Test Case 8: Upload Page – Incorrect CSV File Handling . . . . .            | 68 |
| 6.9  | Backend Test Case 1: Model Prediction Workflow . . . . .                        | 70 |
| 6.10 | Backend Test Case 2: CSV Feature Validation . . . . .                           | 71 |
| 6.11 | Test Case 3: Flask Model Response Consistency . . . . .                         | 72 |
| 6.12 | Test Case 1: End-to-End Ad Upload & Prediction (Valid Flow) . . . . .           | 73 |
| 6.13 | Test Case 2: Invalid Upload Handling (Negative Case) . . . . .                  | 75 |
| 6.14 | Test Case 3: Results Display and Engagement Categorization . . . . .            | 76 |
| 6.15 | Test Case 4: Data Consistency Between Backend, Firebase, and Frontend . . . . . | 78 |
| 6.16 | Test Case 5: PDF Report Generation and Data Integrity . . . . .                 | 79 |
| 6.17 | Test Case 1: Homepage Load Time Verification . . . . .                          | 80 |
| 6.18 | Test Case 2: Login API Response Time . . . . .                                  | 81 |
| 6.19 | Test Case 3: CSV Upload and Prediction Processing Time . . . . .                | 82 |
| 6.20 | Test Case 4: Results Page Load and Graph Rendering Performance . . . . .        | 83 |
| 6.21 | Test Case 5: Concurrent Prediction Requests Load Test . . . . .                 | 84 |
| 6.22 | Test Case 6: Database Read/Write Latency (Firebase) . . . . .                   | 85 |

## LIST OF TABLES

|      |                                                                      |    |
|------|----------------------------------------------------------------------|----|
| 6.23 | Test Case 7: Stress Test – Model Response Degradation . . . . .      | 86 |
| 6.24 | Test Case 8: Negative – Network Latency & Timeout Handling . . . . . | 87 |
| 6.25 | Security Test Case 1: Login & Authentication . . . . .               | 88 |
| 6.26 | Security Test Case 2: Restricted Pages Access Control . . . . .      | 90 |

# Acronyms and Abbreviations

|      |                                   |
|------|-----------------------------------|
| EEG  | Electroencephalography            |
| fMRI | functional MRI                    |
| SST  | Steady-State Topography           |
| LDA  | Linear Discriminant Analysis      |
| SVM  | Support Vector Machine            |
| RBF  | Radial Basis Function             |
| URL  | Uniform Resource Locator          |
| API  | Application programming interface |

# Chapter 1

## Introduction

### 1.1 Project Background

The need to know how individuals really feel and react towards advertisements is one of the largest challenges in the contemporary advertising sphere. The old processes of marketing like surveys, interviews and focus groups rely on what the individuals say or what they believe they felt and this proves to be unreliable or affected by lapses in memories and social expectations. These techniques are able to gather conscious views but not the subconscious emotional and cognitive responses which are significant in determining actual consumer behavior.

Neuromarketing fills this chasm as neuroscience and marketing are merged to investigate the working of the brain under advertising stimuli. Neuromarketing methods do not rely on self-reported information but the physiological and neural responses to comprehend attention, emotion, and memory. These responses are often measured with kind of tools as functional MRI (fMRI), eye-tracking, analysis of facial expressions, and electroencephalography (EEG). Of these approaches, EEG is one of the most feasible and popular approaches based on its being non-invasive, portability, real-time signal recording, and its rather low cost relative to fMRI. EEG records the brainwave patterns indicating the attention, engagement, and emotional arousal, which is why it is an especially effective method to analyze the audience response to video advertisements.

The proposed project, which is called NeuroAd, is based on these advantages. It uses machine learning on EEG data recorded on human subjects as they view video ads in case it can forecast whether the advertisement is engaging or not. The system is a unified and intelligent system that combines EEG signals, ad metadata, and viewer responses. NeuroAd enables objective data-driven insights into audience engagement with content in contrast to traditional marketing research methods which use delayed or biased feedback.

Through observing these neural responses, it enables advertisers to know what areas of their advertisements will be attention grabbing, emotional or not connecting. In non-technical terms, NeuroAd analyzes the brain data to provide an advertiser with a better and cheaper means of testing the actual effectiveness of their content.

## 1.2 Problem Description

Advertising has a huge impact on the way human thoughts and choices are made but there is a problem with quantifying the fact that an advertisement actually resonates with the audience. Conventional survey and focus group techniques are based on self-reported opinion and are subject to bias and not based on emotional or cognitive interest in the moment. These strategies usually overlook the underlying more subliminal responses which are the source of consumer behavior. Neuromarketing is the solution to this problem, as neuroscience-related methods are employed to examine the reaction of the brain to adverts. Among them, EEG (Electroencephalography) is the most outstanding as it is possible to record the activity of the brain related to attention, memory, and engagement of the emotions. Nevertheless, high-end neuromarketing agencies, including Nielsen Consumer Neuroscience and Neuro-Insight are now active in the area and adopt more advanced technologies like EEG, fMRI and eye-tracking to serve large corporate clients such as Spotify, Snapchat, Walmart Connect, and PepsiCo. Although such agencies may provide very specific knowledge, the solutions are oriented on large-scale enterprises, and cannot be used by small and medium-sized enterprises due to its financial and logistical constraints. To address this issue, NeuroAd is planning to create an EEG-based ad engagement prediction system that is specific to SMEs and researchers. Addressing the expensive and somewhat complicated approaches, it offers an easier, more accessible and cheaper means to examine the brain activity and comprehend the ways in which the audience reacts to various advertisements.

## 1.3 Project Objectives

1. 1. Train a machine learning model (Random Forest) to predict whether adverts are engaging or not, by using EEGs.
2. 2. Develop a web-based system that takes EEG data and predicts engagement and displays it using interactive graphs and reports to be downloaded.
3. 3. Offer a low-cost and easily available neuromarketing instrument to small and medium enterprises.

## **1.4 Project Scope**

NeuroAd is an EEG-driven machine learning system aimed at predicting advertisement engagement on the basis of the data obtained in the form of brainwaves measured by EEG headset. It uses EEG to categorize ads as either engaging or non engaging with high accuracy, precision, which will give advertisers, in particular, small and medium enterprises accurate information on audience attention and emotion. The system comprises a secure web-based platform by which the EEG data could be uploaded, predicting, and displaying the outcome in the form of interactive reports, which are an accessible and cost-effective neuromarketing system. This approach bridges the gap between traditional marketing methods and advanced neural data analysis to optimize ad effectiveness.

## Chapter 2

# Literature Review

### 2.1 Existing Literature

#### 2.1.1 Engagement Estimation in Advertisement Videos with EEG

Balasubramanian et al. (2018) performed an experiment to approximate the engagement of users when viewing advertisements based on EEG signals. The scholars used the Emotiv EEG headset to receive the data on the brainwaves and provided the machine learning classifiers like Linear Discriminant Analysis (LDA), Linear Support Vector Machine (SVM) and Radial Basis Function (RBF) SVM to differentiate between engaged and non-engaged modes. The binary classification model with a score of 0.709 in F1 indicated that the EEG-based features are effective in detecting engagement. The current paper highlights the usefulness of EEG-based engagement measurement in neuromarketing. [1]

#### 2.1.2 Using Support Vector Machine on EEG for Advertisement Impact Assessment

Wei et al. (2018) explored the use of EEG signals to assess the impact of advertisements on consumer interest and engagement. The study employed a binary classification approach (high vs. low engagement) using Support Vector Machine (SVM) models. EEG features extracted post-ad viewing were used to predict participants' likelihood of purchasing the advertised product. The model achieved approximately 75 accuracy, highlighting a strong correlation between EEG data and consumer decision-making behavior. [2]

#### 2.1.3 Discrete Classification Technique Applied to TV Advertisements Liking Recognition System

In a 2016 study published in BioMedical Engineering Online, researchers proposed a system for recognizing advertisement liking using EEG signals obtained from the Emotiv EPOC headset. The system applied a binary classification technique to categorize viewer responses as either "liked" or "disliked." The study demonstrated that even low-cost

consumer-grade EEG devices can effectively detect audience preferences, suggesting a practical and scalable approach for real-time neuromarketing applications.[3]

## **2.2 Industry Applications**

### **2.2.1 Nielsen NeuroFocus**

Nielsen NeuroFocus pioneered commercial neuromarketing by combining EEG, eye tracking, and GSR to analyze attention, memory retention, and emotional engagement. Their proprietary algorithm generates a NeuroScore, quantifying ad effectiveness. While accurate, these tools are designed for large enterprises and remain costly and inaccessible to smaller players [4]

### **2.2.2 Neuro-Insight**

Neuro-Insight uses Steady-State Topography (SST), a brain-monitoring technique related to EEG, to measure real-time memory encoding and visual attention. Their tools have been used by global brands such as Google and Unilever. However, like Nielsen, Neuro-Insight operates a closed, commercial ecosystem that limits academic or small-business access [5].

## **2.3 Research Gaps and Motivation**

Although prior research has demonstrated the potential of EEG signals for assessing advertisement engagement, most existing studies have been conducted in controlled laboratory environments with limited real-world application. Furthermore, the majority of commercial neuromarketing systems employ high-end technologies and proprietary frameworks, making them financially and technically inaccessible to small and medium-sized enterprises or academic institutions. There is also a lack of integrated systems that encompass EEG data acquisition, preprocessing, engagement classification, and visualization within a single, user-friendly platform. This research aims to bridge these gaps by developing a cost-effective, EEG-based engagement prediction system utilizing the EMOTIV EPOC headset. The proposed solution combines machine learning, a Flask–React web interface, and Firebase integration to deliver an accessible, practical, and scalable neuromarketing tool for research and small business applications

## Chapter 3

# Requirement Specifications

### 3.1 Existing System

Presently, most advertisers rely on traditional methods such as focus groups, post-advertisement surveys, and retrospective behavioral metrics (e.g., sales uplift, click-through rates) to evaluate audience engagement. While these techniques are widely adopted, they present significant drawbacks. Focus groups and surveys depend on self-reported data, which are prone to response bias, social desirability effects, and recall inaccuracies. Similarly, behavioral metrics are delayed indicators and fail to capture immediate, subconscious reactions to advertisements. Consequently, these methods tend to be slow, resource-intensive, and often produce incomplete or unreliable insights. In recent years, neuromarketing has introduced advanced EEG-based approaches for measuring consumer engagement more objectively. Prominent commercial solutions—offered by companies such as Neuro-Insight, Nielsen Neuro, and SPARK Neuro—integrate electroencephalography (EEG), eye-tracking, and other biometric sensors to deliver granular insights into cognitive and emotional responses. However, despite their technological advancement, these systems exhibit several limitations:

1. **High Cost:** Deployment and operation of such systems are expensive, involving proprietary hardware, specialized personnel, and long-term maintenance.
2. **Infrastructure Dependence:** These solutions are designed for controlled lab environments and lack portability for field use.
3. **Limited Accessibility:** Their services are geared toward large corporations with sufficient financial and technical resources.
4. **Technical Complexity:** Interpretation of EEG data requires domain expertise in neuroscience and machine learning, making it inaccessible to general marketers or small research teams.

5. **Usability Constraints:** There is an absence of intuitive, lightweight platforms tailored for small businesses or academic use.

Due to these barriers, EEG-based engagement analytics remain largely inaccessible to smaller organizations, startups, and individual researchers. The existing market does not offer a cost effective, user-friendly, and scalable solution capable of democratizing neuromarketing for broader application.

### 3.2 Proposed System

The proposed system, NeuroAd, is a web-based EEG engagement prediction platform designed to provide advertisers, researchers, and testing personnel with real-time insights into audience engagement. NeuroAd focuses solely on predicting ad engagement based on EEG data collected from viewers using the Emotiv EPOC X EEG headset. The system leverages a trained Random Forest model for classification, displaying results through an intuitive web interface.

### 3.3 Requirement Specifications

Requirement specifications define functional and non-functional needs of the system. They act as a blueprint for development, ensuring all stakeholders understand system expectations and objectives.

#### 3.3.1 Functional Requirements

Table 3.1: Functional Requirements

| Req. ID                       | Requirement Statement                                                                                                    | Comments                                                    |
|-------------------------------|--------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------|
| FR01                          | The system shall provide a user authentication mechanism using Firebase for Ad Testing Personnel and Admin.              | Ensures secure login and access control.                    |
| FR02                          | The system shall support role-based access control distinguishing between General User, Ad Testing Personnel, and Admin. | Prevents unauthorized access to restricted functionalities. |
| <i>Continued on next page</i> |                                                                                                                          |                                                             |

### CHAPTER 3 – REQUIREMENT SPECIFICATIONS

| <b>Req. ID</b>                | <b>Requirement Statement</b>                                                                                                                      | <b>Comments</b>                                                                       |
|-------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------|
| FR03                          | The system shall allow General Users to access only informational pages such as Home, About, Contact, and Model Summary.                          | Ensures a user-friendly interface for public access without data modification rights. |
| FR04                          | The system shall allow Ad Testing Personnel to upload EEG data files in .csv format.                                                              | Upload limit set to 50 MB; ensures valid EEG input format.                            |
| FR05                          | The system shall validate uploaded EEG files to check for missing or invalid data entries before processing.                                      | Prevents incorrect data from affecting model accuracy.                                |
| FR06                          | The system shall extract relevant EEG features such as attention, memory, and emotional indicators for engagement classification.                 | Derived from EEG features.                                                            |
| FR07                          | The system shall classify EEG data as Engaging or Non-Engaging using a trained Random Forest model.                                               | Model trained on validated EEG datasets; target accuracy $\geq 90\%$ .                |
| FR08                          | The system shall display classification results to Ad Testing Personnel, including engagement label, confidence percentage, and summary insights. | Provides clear and interpretable results for analysis.                                |
| FR09                          | The system shall allow Ad Testing Personnel to download the prediction results in .csv and .pdf formats.                                          | Facilitates sharing of reports with clients.                                          |
| FR10                          | The system shall automatically store all processed results, including ad name, duration, engagement score, and label, in Firebase.                | Enables data persistence and report generation.                                       |
| FR11                          | The system shall allow Admin to access a dashboard containing logs of all uploaded EEG tests and prediction outcomes.                             | Enables centralized monitoring and control.                                           |
| FR12                          | The system shall allow Admin to filter logs based on criteria such as ad name, engagement type, or date.                                          | Facilitates efficient data retrieval for analytics.                                   |
| <i>Continued on next page</i> |                                                                                                                                                   |                                                                                       |

| Req. ID | Requirement Statement                                                                                                                 | Comments                                                |
|---------|---------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------|
| FR13    | The system shall display analytical summaries for Admin, including total tests conducted, engagement distribution, and recent trends. | Supports decision-making through visual insights.       |
| FR14    | The system shall maintain a centralized Firebase database for storing all user information, EEG results, and logs.                    | Ensures data consistency and real-time synchronization. |
| FR15    | The system shall provide logout functionality for all authenticated users.                                                            | Ensures security by terminating user sessions safely.   |
| FR16    | The system shall allow the Admin to add new users in the User Management tab.                                                         | Supports onboarding of new personnel and account setup. |
| FR17    | The system shall allow the Admin to edit existing user information, including role and status, in the User Management tab.            | Enables updates to user profiles and access control.    |
| FR18    | The system shall allow the Admin to delete users from the system through the User Management tab.                                     | Allows removal of inactive or unauthorized users.       |
| FR19    | The system shall allow the Admin to edit advertisement records, including ad name and duration.                                       | Enables correction and management of ad-related data.   |

### 3.3.2 Non-Functional Requirements

Table 3.2: Non-Functional Requirements

| Req. ID                       | Requirement Statement                                                                                 | Comments                                                        |
|-------------------------------|-------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------|
| NFR01                         | The system shall process EEG data and display engagement results within 10 seconds under normal load. | Ensures real-time performance for smooth testing and reporting. |
| NFR02                         | The backend shall classify EEG data with a $\geq 95\%$ successful response rate per 100 requests.     | Maintains processing reliability.                               |
| <i>Continued on next page</i> |                                                                                                       |                                                                 |

## CHAPTER 3 – REQUIREMENT SPECIFICATIONS

| <b>Req. ID</b> | <b>Requirement Statement</b>                                                                                                         | <b>Comments</b>                                           |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------|
| NFR03          | The application shall support all major browsers and be responsive on desktop and mobile devices.                                    | Ensures responsive user experience.                       |
| NFR04          | The interface shall load fully within 3 seconds on a 10 Mbps connection.                                                             | Optimizes frontend performance for better responsiveness. |
| NFR05          | The system shall be usable by non-technical users with less than 10 minutes of guidance.                                             | Improves ease of use.                                     |
| NFR06          | Firebase read/write latency shall be $\leq 300$ ms for stored data.                                                                  | Ensures efficient database access.                        |
| NFR07          | All data and logs shall be encrypted and stored securely using Firebase authentication.                                              | Protects user and EEG data.                               |
| NFR08          | The model shall generate engagement predictions within 5 seconds after data upload.                                                  | Provides timely results.                                  |
| NFR09          | Role-based access shall be enforced via Firebase Authentication and Security Rules.                                                  | Restricts unauthorized access.                            |
| NFR10          | The system shall validate all Admin actions (add, edit, delete users and edit ad records) before committing changes to the database. | Prevents accidental or unauthorized modifications.        |
| NFR11          | The system shall ensure all Admin operations (user and ad edits) are securely logged for auditing.                                   | Enables traceability and accountability.                  |
| NFR12          | The system shall reflect user and ad record updates in real time across all connected clients.                                       | Ensures consistency and synchronization.                  |
| NFR13          | The system shall restrict Admin-level functions to authenticated sessions with elevated privileges.                                  | Maintains system integrity and prevents misuse.           |

### 3.4 Use Case

Figure 3.1 illustrates the **General User Use Case** for the NeuroAd Platform. It shows the interactions of a general user with the system, including accessing various pages such as the Home, Model, About, and Contact Us pages, as well as sending inquiries through the contact form. This diagram helps visualize the basic navigation and user interaction flow within the platform.

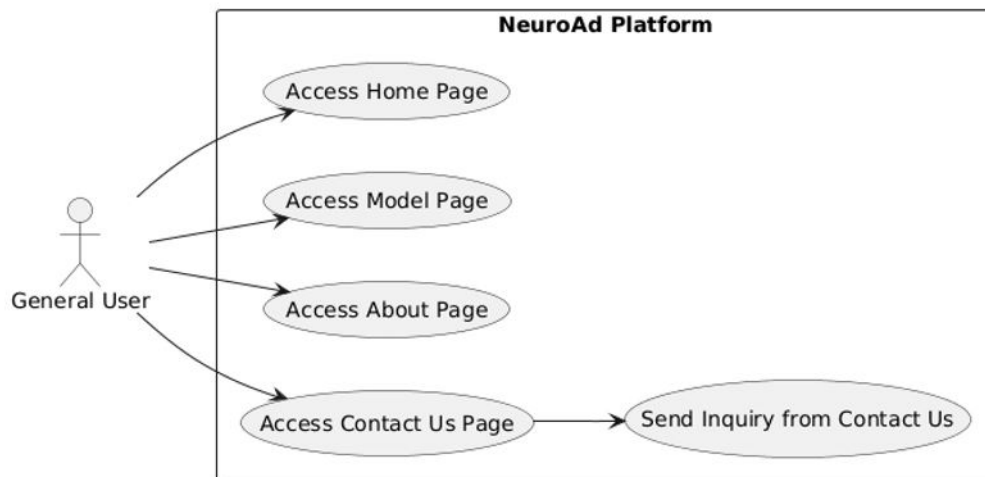


Figure 3.1: General User Use Case

Figure 3.2 illustrates the **Ad Testing Personnel Use Case** for the NeuroAd Platform. It highlights how ad testing personnel interact with the system, including logging in, accessing the dashboard and upload pages, viewing past tests and results, predicting engagement, and downloading or sharing results. This diagram provides an overview of the essential tasks performed by ad testing personnel within the system.

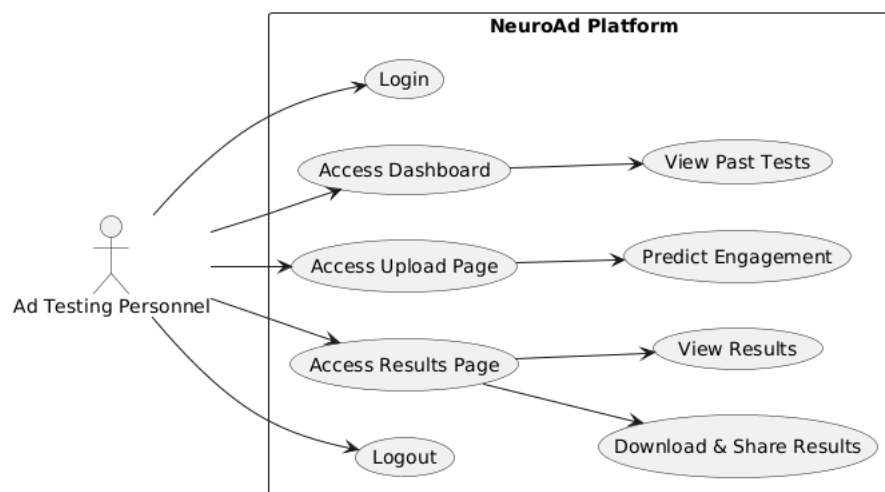


Figure 3.2: Tester Use Case

Figure 3.3 illustrates the **Admin Use Case** for the NeuroAd Platform. It demonstrates key administrative functions such as managing testing personnel, performing ad tests, viewing and analyzing results, accessing logs, and visualizing data through analytical tools. This diagram provides an overview of the admin's comprehensive control and monitoring capabilities within the system.

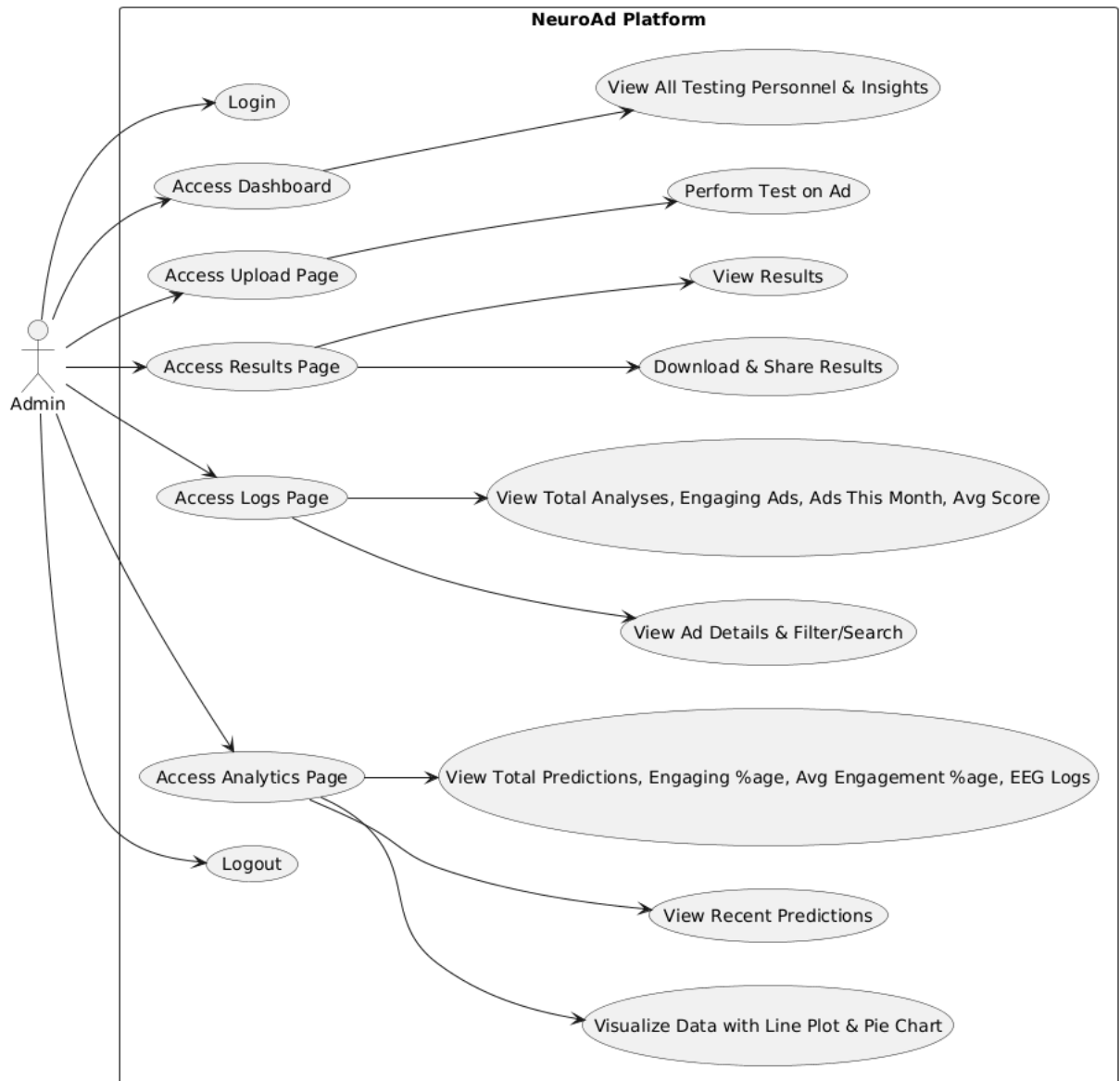


Figure 3.3: Admin Use Case

### 3.5 Descriptive Use Cases

#### 3.5.1 General User

Table 3.3: Use Case UC-1.1: View Website Pages

|                                              |                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|----------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Use Case ID</b>                           | <b>UC-1.1</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| <b>Use Case Name</b>                         | <b>View Website Pages (Home, About, Model)</b>                                                                                                                                                                                                                                                                                                                                                                                                                |
| <b>Actors</b>                                | General User (Website Visitor)                                                                                                                                                                                                                                                                                                                                                                                                                                |
| <b>Description</b>                           | The user browses the company’s website to learn about the organization and its EEG-based advertisement engagement model.                                                                                                                                                                                                                                                                                                                                      |
| <b>Trigger</b>                               | User opens the website URL in a browser.                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>Preconditions</b>                         | User has internet access.                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| <b>Priority</b>                              | High                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| <b>Basic Flow</b>                            | <ol style="list-style-type: none"> <li>1. The user navigates to the website URL.</li> <li>2. The system loads the Home page with navigation options (Home, About Us, Contact Us, Model).</li> <li>3. The user clicks on “About Us.”</li> <li>4. The system displays company information and mission details.</li> <li>5. The user clicks on “Model.”</li> <li>6. The system fetches and displays live model metrics (accuracy, precision, recall).</li> </ol> |
| <b>Actor Actions</b>                         |                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| <b>Actor Actions</b>                         | <b>System Response</b>                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| 1. User browses pages                        | The system loads corresponding content sections dynamically.                                                                                                                                                                                                                                                                                                                                                                                                  |
| 2. User clicks “Model.”                      | The system retrieves latest performance metrics from Fire-base and displays them.                                                                                                                                                                                                                                                                                                                                                                             |
| <b>Alternative Course of Action (if any)</b> |                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| <b>Actor Action</b>                          | <b>System Response</b>                                                                                                                                                                                                                                                                                                                                                                                                                                        |

|                                        |                                                                                                                                                                                      |
|----------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1. If the internet connection is lost. | The system displays a popup alert: “Network error. Please check your connection.”                                                                                                    |
| 2. If backend data is unavailable.     | The system displays: “Unable to fetch model metrics. Please try again later.”                                                                                                        |
| <b>Failure Scenarios</b>               | <ul style="list-style-type: none"> <li>- Website server is unreachable.</li> <li>- Firebase fetch operation fails.</li> <li>- UI incompatibility causes rendering errors.</li> </ul> |

Table 3.4: Use Case UC-1.2: Submit Contact Inquiry

|                      |                                                                                                                                                                                                                                                                                                                                                                                                                  |
|----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Use Case ID</b>   | <b>UC-1.2</b>                                                                                                                                                                                                                                                                                                                                                                                                    |
| <b>Use Case Name</b> | <b>Submit Contact Inquiry</b>                                                                                                                                                                                                                                                                                                                                                                                    |
| <b>Actors</b>        | General User (Website Visitor)                                                                                                                                                                                                                                                                                                                                                                                   |
| <b>Description</b>   | User submits contact inquiry through website form to request more information or services.                                                                                                                                                                                                                                                                                                                       |
| <b>Trigger</b>       | User clicks on the “Contact Us” link or button.                                                                                                                                                                                                                                                                                                                                                                  |
| <b>Preconditions</b> | User is on Contact Us page; Gmail API and backend are running.                                                                                                                                                                                                                                                                                                                                                   |
| <b>Priority</b>      | High                                                                                                                                                                                                                                                                                                                                                                                                             |
| <b>Basic Flow</b>    | <ol style="list-style-type: none"> <li>1. The user enters their name, email, and message in the contact form.</li> <li>2. The system validates the form input.</li> <li>3. The user clicks “Send.”</li> <li>4. The system sends messages via Gmail API.</li> <li>5. The system displays a confirmation: “Message sent successfully.”</li> <li>6. The user receives an automated acknowledgment email.</li> </ol> |
| <b>Actor Actions</b> |                                                                                                                                                                                                                                                                                                                                                                                                                  |
| <b>Actor Actions</b> | <b>System Response</b>                                                                                                                                                                                                                                                                                                                                                                                           |

|                                                  |                                                                                                                                                                                                                                                                    |
|--------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1. User fills contact form.                      | The system checks all required fields and validates email format.                                                                                                                                                                                                  |
| 2. User clicks “Send.”                           | The system sends data to Gmail API and shows a confirmation message.                                                                                                                                                                                               |
| 3. User checks in-box.                           | The system delivers an auto-reply email confirming receipt.                                                                                                                                                                                                        |
| <b>Alternative Course of Action (if any)</b>     |                                                                                                                                                                                                                                                                    |
| <b>Actor Action</b>                              | <b>System Response</b>                                                                                                                                                                                                                                             |
| 1. If required fields are missing.               | The system highlights missing fields and shows: “Please fill out all required fields.”                                                                                                                                                                             |
| 2. If email format is invalid.                   | The system displays: “Invalid email address.”                                                                                                                                                                                                                      |
| 3. If the user navigates away before submission. | The message is not sent, and no confirmation is shown.                                                                                                                                                                                                             |
| <b>Failure Scenario</b>                          | <ul style="list-style-type: none"> <li>– Gmail API connection fails → “Error sending message. Please try again later.”</li> <li>– Backend timeout → Request fails and error popup appears.</li> <li>– Page session expires → User must reload to retry.</li> </ul> |

### 3.5.2 Ad Testing Personnel

Table 3.5: Use Case UC-2.1: Employee Login

|                      |                                                                                                             |
|----------------------|-------------------------------------------------------------------------------------------------------------|
| <b>Use Case ID</b>   | <b>UC-2.1</b>                                                                                               |
| <b>Use Case Name</b> | <b>Employee Login</b>                                                                                       |
| <b>Actors</b>        | Employee                                                                                                    |
| <b>Description</b>   | Employee logs into the system using Firebase authentication to access dashboard, upload, and results pages. |
| <b>Trigger</b>       | Employee opens login page and submits credentials.                                                          |
| <b>Preconditions</b> | Employee must be registered in Firebase database.                                                           |
| <b>Priority</b>      | High                                                                                                        |

|                                              |                                                                                                                                                                                                                  |
|----------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Basic Flow</b>                            | <ol style="list-style-type: none"> <li>1. Employee enters email and password.</li> <li>2. System validates credentials with Firebase.</li> <li>3. On success, system redirects to Employee Dashboard.</li> </ol> |
| <b>Actor Actions</b>                         | <b>System Response</b>                                                                                                                                                                                           |
| 1. Employee enters email & password          | System validates credentials.                                                                                                                                                                                    |
| 2. Employee clicks “Login”                   | System redirects to Employee Dashboard.                                                                                                                                                                          |
| <b>Alternative Course of Action (if any)</b> |                                                                                                                                                                                                                  |
| <b>Actor Action</b>                          | <b>System Response</b>                                                                                                                                                                                           |
| 1. Employee enters wrong credentials         | System shows “Invalid login” error.                                                                                                                                                                              |
| 2. Network issue during authentication       | System shows “Login failed. Please try again.”                                                                                                                                                                   |
| <b>Failure Scenario</b>                      | Firebase authentication fails due to network/server issue<br>→ system shows error “Login failed. Please try again.”                                                                                              |

Table 3.6: Use Case UC-2.2: Dashboard Insights

|                      |                                                                                                                 |
|----------------------|-----------------------------------------------------------------------------------------------------------------|
| <b>Use Case ID</b>   | <b>UC-2.2</b>                                                                                                   |
| <b>Use Case Name</b> | <b>Dashboard Insights</b>                                                                                       |
| <b>Actors</b>        | Employee                                                                                                        |
| <b>Description</b>   | Employee views summary of past ad tests, including engagement metrics, recent analysis history, and AI insights |
| <b>Trigger</b>       | Employee navigates to Dashboard.                                                                                |
| <b>Preconditions</b> | Employee logged in                                                                                              |
| <b>Priority</b>      | High                                                                                                            |

|                                                         |                                                                                                                                                                                                                                                           |
|---------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Basic Flow</b>                                       | <ol style="list-style-type: none"> <li>1. Employee opens Dashboard page.</li> <li>2. System fetches past test results and engagement data from Firebase.</li> <li>3. Employee views summary cards, tabs with recent analyses, and AI insights.</li> </ol> |
| <b>Actor Actions</b>                                    | <b>System Response</b>                                                                                                                                                                                                                                    |
| 1. Employee navigates to Dashboard                      | System fetches test history and metrics from Firebase                                                                                                                                                                                                     |
| 2. Employee clicks on tabs (Recent Analysis / Insights) | System displays data including engagement %, attention %, memory %, AI summary                                                                                                                                                                            |
| <b>Alternative Course of Action (if any)</b>            |                                                                                                                                                                                                                                                           |
| <b>Actor Action</b>                                     | <b>System Response</b>                                                                                                                                                                                                                                    |
| 1. No past test data exists                             | System displays “No tests available” message                                                                                                                                                                                                              |
| 2. Firebase fetch fails                                 | System shows empty state with “Unable to load data”                                                                                                                                                                                                       |
| <b>Failure Scenario</b>                                 | Firebase backend failure → system cannot fetch Dashboard metrics; shows empty Dashboard.                                                                                                                                                                  |

Table 3.7: Use Case UC-2.3: Upload CSV &amp; Predict Engagement

|                      |                                                                                                                                    |
|----------------------|------------------------------------------------------------------------------------------------------------------------------------|
| <b>Use Case ID</b>   | <b>UC-2.3</b>                                                                                                                      |
| <b>Use Case Name</b> | <b>Upload CSV &amp; Predict Engagement</b>                                                                                         |
| <b>Actors</b>        | Employee                                                                                                                           |
| <b>Description</b>   | Employee uploads CSV of ad data, triggers backend model prediction, and views results including engagement metrics and AI summary. |
| <b>Trigger</b>       | Employee clicks “Upload CSV” on Upload page.                                                                                       |
| <b>Preconditions</b> | Employee logged in; backend and Firebase operational                                                                               |

|                                              |                                                                                                                                                                                                                                                                                                                                                                          |
|----------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Priority</b>                              | High                                                                                                                                                                                                                                                                                                                                                                     |
| <b>Basic Flow</b>                            | <ol style="list-style-type: none"> <li>1. Employee selects CSV, ad name, and duration.</li> <li>2. System validates CSV and required fields.</li> <li>3. Employee clicks “Predict Engagement.”</li> <li>4. Backend runs model, stores results in Firebase.</li> <li>5. Employee navigates to Results page and views metrics, AI summary, and recommendations.</li> </ol> |
| <b>Actor Actions</b>                         | <b>System Response</b>                                                                                                                                                                                                                                                                                                                                                   |
| 1. Employee selects CSV, ad name, duration   | System validates CSV and required fields                                                                                                                                                                                                                                                                                                                                 |
| 2. Employee clicks “Predict Engagement”      | System uploads to backend, triggers model, stores results in Firebase, shows progress bar                                                                                                                                                                                                                                                                                |
| 3. Employee navigates to Results page        | System fetches prediction and displays metrics, AI summary, recommendations                                                                                                                                                                                                                                                                                              |
| <b>Alternative Course of Action (if any)</b> |                                                                                                                                                                                                                                                                                                                                                                          |
| <b>Actor Action</b>                          | <b>System Response</b>                                                                                                                                                                                                                                                                                                                                                   |
| 1. CSV has invalid format or missing fields  | System shows validation error and blocks upload                                                                                                                                                                                                                                                                                                                          |
| 2. Backend prediction fails                  | System shows “Prediction failed”                                                                                                                                                                                                                                                                                                                                         |
| <b>Failure Scenario</b>                      | Backend or Firebase fails → employee cannot get prediction; error message shown.                                                                                                                                                                                                                                                                                         |

Table 3.8: Use Case UC-2.4: View Results &amp; Download PDF Report

|                      |                                               |
|----------------------|-----------------------------------------------|
| <b>Use Case ID</b>   | <b>UC-2.4</b>                                 |
| <b>Use Case Name</b> | <b>View Results &amp; Download PDF Report</b> |

|                                                 |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|-------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Actors</b>                                   | Employee                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| <b>Description</b>                              | Employee views engagement results of a tested ad and downloads reports. The system provides two options: PDF report for the engagement summary and CSV for the detailed predictions.                                                                                                                                                                                                                                                                                                                                       |
| <b>Trigger</b>                                  | Employee is on Results page                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| <b>Preconditions</b>                            | Prediction must exist in Firebase; Employee logged in.                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| <b>Priority</b>                                 | Medium                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| <b>Basic Flow</b>                               | <ol style="list-style-type: none"> <li>1. Employee is on results page.</li> <li>2. System displays engagement score, attention &amp; memory metrics, brain region activity, AI summary, and recommendations.</li> <li>3. Employee clicks “Download PDF Report.”</li> <li>4. System generates a PDF report from displayed data and triggers download.</li> <li>5. Employee clicks “Download CSV.”</li> <li>6. System generates a CSV file containing all predictions and detailed metrics and triggers download.</li> </ol> |
| <b>Actor Actions</b>                            | <b>System Response</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| 1. Employee is on results page of the tested ad | System displays engagement metrics, AI summary, recommendations, and two download buttons (PDF & CSV)                                                                                                                                                                                                                                                                                                                                                                                                                      |
| 2. Employee clicks “Download PDF”               | System generates a PDF report from displayed data and triggers download                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| 3. Employee clicks “Download CSV”               | System generates a CSV file with all predictions and detailed metrics and triggers download                                                                                                                                                                                                                                                                                                                                                                                                                                |
| <b>Alternative Course of Action (if any)</b>    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| <b>Actor Action</b>                             | <b>System Response</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |

|                                |                                                                                                                                                                                                                                                                                               |
|--------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1. PDF generation fails        | System shows “Unable to generate PDF report. Please try again.”                                                                                                                                                                                                                               |
| 2. CSV generation fails        | System shows “Unable to generate CSV file. Please try again.”                                                                                                                                                                                                                                 |
| 3. Firebase data fails to load | System shows “No Data Available” and disables download buttons                                                                                                                                                                                                                                |
| <b>Failure Scenario</b>        | <ol style="list-style-type: none"> <li>1. Backend or Firebase failure → engagement metrics and AI summary not displayed; download buttons unavailable; system shows error message.</li> <li>2. Network failure during download → system shows “Download failed. Please try again.”</li> </ol> |

### 3.5.3 Admin

Table 3.9: Use Case UC-3.1: Admin Login

|                             |                                                                                                                                                                                  |
|-----------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Use Case ID</b>          | <b>UC-3.1</b>                                                                                                                                                                    |
| <b>Use Case Name</b>        | <b>Admin Login</b>                                                                                                                                                               |
| <b>Actors</b>               | Admin                                                                                                                                                                            |
| <b>Description</b>          | Admin logs in via Firebase to access dashboard and admin functionalities.                                                                                                        |
| <b>Trigger</b>              | Admin opens login page and submits credentials                                                                                                                                   |
| <b>Preconditions</b>        | Admin registered in Firebase; Firebase operational                                                                                                                               |
| <b>Priority</b>             | High                                                                                                                                                                             |
| <b>Basic Flow</b>           | <ol style="list-style-type: none"> <li>1. Admin enters email and password.</li> <li>2. System validates credentials.</li> <li>3. Admin redirected to Admin Dashboard.</li> </ol> |
| <b>Actor Actions</b>        | <b>System Response</b>                                                                                                                                                           |
| 1. Admin enters credentials | System validates with Firebase                                                                                                                                                   |
| 2. Admin clicks “Login”     | System redirects to Admin Dashboard                                                                                                                                              |

| Alternative Course of Action (if any) |                                                      |
|---------------------------------------|------------------------------------------------------|
| Actor Action                          | System Response                                      |
| 1. Wrong credentials                  | System shows “Invalid login”                         |
| 2. Network issue                      | System shows “Login failed. Please try again.”       |
| <b>Failure Scenario</b>               | Firebase backend failure → login fails, error shown. |

Table 3.10: Use Case UC-3.2: Manage Users

| <b>Use Case ID</b>                    | <b>UC-3.2</b>                                                                                                                                                                            |
|---------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Use Case Name</b>                  | <b>Manage Users</b>                                                                                                                                                                      |
| <b>Actors</b>                         | Admin                                                                                                                                                                                    |
| <b>Description</b>                    | Admin creates, edits, or deletes users; can view user activity.                                                                                                                          |
| <b>Trigger</b>                        | Admin navigates to Users tab.                                                                                                                                                            |
| <b>Preconditions</b>                  | Admin logged in; Firebase operational.                                                                                                                                                   |
| <b>Priority</b>                       | High                                                                                                                                                                                     |
| <b>Basic Flow</b>                     | <ol style="list-style-type: none"> <li>1. Admin views user list.</li> <li>2. Admin creates/edits/deletes a user.</li> <li>3. System updates Firebase and refreshes user list.</li> </ol> |
| <b>Actor Actions</b>                  | <b>System Response</b>                                                                                                                                                                   |
| 1. Admin selects user                 | System displays user details                                                                                                                                                             |
| 2. Admin edits/deletes user           | System updates Firebase and refreshes list                                                                                                                                               |
| Alternative Course of Action (if any) |                                                                                                                                                                                          |
| Actor Action                          | System Response                                                                                                                                                                          |
| 1. Firebase operation fails           | System shows “Operation failed” message                                                                                                                                                  |

|                                                      |                                                                      |
|------------------------------------------------------|----------------------------------------------------------------------|
| 2. If user already exists in Firebase Authentication | System shows error “User already exists”                             |
| <b>Failure Scenario</b>                              | Firebase backend failure → user management actions fail; error shown |

Table 3.11: Use Case UC-3.3: View Activity Logs &amp; Notifications

|                                              |                                                                                                                                                                                                  |
|----------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Use Case ID</b>                           | <b>UC-3.3</b>                                                                                                                                                                                    |
| <b>Use Case Name</b>                         | <b>View Activity Logs &amp; Notifications</b>                                                                                                                                                    |
| <b>Actors</b>                                | Admin                                                                                                                                                                                            |
| <b>Description</b>                           | Admin views system logs, activity notifications, marks notifications as read, filters logs.                                                                                                      |
| <b>Trigger</b>                               | Admin navigates to Logs or Notifications tab.                                                                                                                                                    |
| <b>Preconditions</b>                         | Admin logged in; Firebase operational.                                                                                                                                                           |
| <b>Priority</b>                              | Medium                                                                                                                                                                                           |
| <b>Basic Flow</b>                            | <ol style="list-style-type: none"> <li>1. Admin views activity logs.</li> <li>2. Admin applies filters or searches logs.</li> <li>3. Admin views notifications and marks read/unread.</li> </ol> |
| <b>Actor Actions</b>                         | <b>System Response</b>                                                                                                                                                                           |
| 1. Admin opens Logs                          | System fetches logs from Firebase and displays                                                                                                                                                   |
| 2. Admin filters/searches                    | System updates displayed logs                                                                                                                                                                    |
| 3. Admin views notifications                 | System displays notifications; allows marking as read                                                                                                                                            |
| <b>Alternative Course of Action (if any)</b> |                                                                                                                                                                                                  |
| <b>Actor Action</b>                          | <b>System Response</b>                                                                                                                                                                           |
| 1. No logs available                         | System shows “No logs available”                                                                                                                                                                 |

|                         |                                                                  |
|-------------------------|------------------------------------------------------------------|
| 2. Firebase fetch fails | System shows empty state                                         |
| <b>Failure Scenario</b> | Firestore failure → logs/notifications unavailable; error shown. |

Table 3.12: Use Case UC-3.4: View AI Insights

|                                              |                                                                                                                                                                                                               |
|----------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Use Case ID</b>                           | <b>UC-3.4</b>                                                                                                                                                                                                 |
| <b>Use Case Name</b>                         | <b>View AI Insights</b>                                                                                                                                                                                       |
| <b>Actors</b>                                | Admin                                                                                                                                                                                                         |
| <b>Description</b>                           | Admin views AI-generated insights and summary of engagement metrics, trends, and brain activity.                                                                                                              |
| <b>Trigger</b>                               | Admin navigates to AI Insights tab.                                                                                                                                                                           |
| <b>Preconditions</b>                         | Admin logged in; backend operational.                                                                                                                                                                         |
| <b>Priority</b>                              | High                                                                                                                                                                                                          |
| <b>Basic Flow</b>                            | <ol style="list-style-type: none"> <li>Admin selects period for AI summary.</li> <li>System fetches data from backend and displays engagement metrics.</li> <li>Admin views AI summary and charts.</li> </ol> |
| <b>Actor Actions</b>                         | <b>System Response</b>                                                                                                                                                                                        |
| 1. Admin selects period                      | System fetches AI insights from backend                                                                                                                                                                       |
| 2. Admin views metrics                       | System displays users, predictions, engagement %, brain activity, AI summary                                                                                                                                  |
| 3. Admin views charts                        | System renders line and pie charts                                                                                                                                                                            |
| <b>Alternative Course of Action (if any)</b> |                                                                                                                                                                                                               |
| <b>Actor Action</b>                          | <b>System Response</b>                                                                                                                                                                                        |
| 1. No data for selected period               | System shows “No Data Available”                                                                                                                                                                              |
| <b>Failure Scenario</b>                      | Backend or Firestore failure → AI summary unavailable; system shows error.                                                                                                                                    |

Table 3.13: Use Case UC-3.5: View Logs

|                                |                                                                                                                                                                                                                                                                                                                                                                              |
|--------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Use Case ID</b>             | <b>UC-3.5</b>                                                                                                                                                                                                                                                                                                                                                                |
| <b>Use Case Name</b>           | <b>View Logs</b>                                                                                                                                                                                                                                                                                                                                                             |
| <b>Actors</b>                  | Admin                                                                                                                                                                                                                                                                                                                                                                        |
| <b>Description</b>             | Admin views detailed logs of ads tested, including total analyses, number of engaging/non-engaging ads, average scores, and can filter/search by ad name, label, date range, and view details, edit, or delete ads.                                                                                                                                                          |
| <b>Trigger</b>                 | Admin navigates to Logs page.                                                                                                                                                                                                                                                                                                                                                |
| <b>Preconditions</b>           | Admin logged in; Firebase operational.                                                                                                                                                                                                                                                                                                                                       |
| <b>Priority</b>                | High                                                                                                                                                                                                                                                                                                                                                                         |
| <b>Basic Flow</b>              | <ol style="list-style-type: none"> <li>1. Admin opens Logs page.</li> <li>2. System displays top metrics: Total Analyses, Engaging Ads, Non-Engaging Ads, Avg Score.</li> <li>3. Admin filters or searches logs by ad name, label, and date range (7, 30, 90 days, all time).</li> <li>4. Admin selects an ad to view details, edit ad information, or delete ad.</li> </ol> |
| <b>Actor Actions</b>           | <b>System Response</b>                                                                                                                                                                                                                                                                                                                                                       |
| 1. Admin opens Logs page       | System fetches log metrics and displays top stats                                                                                                                                                                                                                                                                                                                            |
| 2. Admin filters/searches logs | System updates displayed logs based on filters/search                                                                                                                                                                                                                                                                                                                        |
| 3. Admin clicks “View Details” | System displays metadata of the selected ad                                                                                                                                                                                                                                                                                                                                  |
| 4. Admin clicks “Edit Ad”      | System allows editing ad name, duration, label, score and updates Firebase                                                                                                                                                                                                                                                                                                   |
| 5. Admin clicks “Delete Ad”    | System deletes ad from Firebase and updates display                                                                                                                                                                                                                                                                                                                          |

| Alternative Course of Action (if any) |                                                                          |
|---------------------------------------|--------------------------------------------------------------------------|
| Actor Action                          | System Response                                                          |
| 1. No data exists                     | System shows “No Data Available”                                         |
| 2. Firebase fetch fails               | System shows empty state or error message                                |
| <b>Failure Scenario</b>               | Backend/Firebase failure → logs unavailable; system shows error message. |

Table 3.14: Use Case UC-3.6: View Analytics

|                               |                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|-------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Use Case ID</b>            | <b>UC-3.6</b>                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <b>Use Case Name</b>          | <b>View Analytics</b>                                                                                                                                                                                                                                                                                                                                                                                                            |
| <b>Actors</b>                 | Admin                                                                                                                                                                                                                                                                                                                                                                                                                            |
| <b>Description</b>            | Admin views overall system analytics, including recent predictions, engagement trends, and visualizations through line and pie charts. Admin can filter/search predictions and select custom time periods for trend analysis.                                                                                                                                                                                                    |
| <b>Trigger</b>                | Admin navigates to Analytics page.                                                                                                                                                                                                                                                                                                                                                                                               |
| <b>Preconditions</b>          | Admin logged in; Firebase operational.                                                                                                                                                                                                                                                                                                                                                                                           |
| <b>Priority</b>               | High                                                                                                                                                                                                                                                                                                                                                                                                                             |
| <b>Basic Flow</b>             | <ol style="list-style-type: none"> <li>1. Admin opens Analytics page.</li> <li>2. System displays recent predictions: ad name, user, duration, score, label, date.</li> <li>3. Admin selects a custom time period for engagement trend analysis.</li> <li>4. System displays line chart for engagement trends over time and pie chart for engagement distribution.</li> <li>5. Admin filters or searches predictions.</li> </ol> |
| <b>Actor Actions</b>          | <b>System Response</b>                                                                                                                                                                                                                                                                                                                                                                                                           |
| 1. Admin opens Analytics page | System fetches recent predictions and displays them                                                                                                                                                                                                                                                                                                                                                                              |

## CHAPTER 3 – REQUIREMENT SPECIFICATIONS

|                                              |                                                                                                  |
|----------------------------------------------|--------------------------------------------------------------------------------------------------|
| 2. Admin filters/searches predictions        | System updates displayed predictions                                                             |
| 3. Admin selects time period                 | System updates line chart to show engagement trend for selected period                           |
| 4. Admin views pie chart                     | System renders engagement distribution chart based on selected data                              |
| <b>Alternative Course of Action (if any)</b> |                                                                                                  |
| <b>Actor Action</b>                          | <b>System Response</b>                                                                           |
| 1. No predictions exist                      | System shows “No Data Available”                                                                 |
| 2. Firebase fetch fails                      | System shows empty state or error message                                                        |
| <b>Failure Scenario</b>                      | Backend/Firebase failure → analytics metrics and charts unavailable; system shows error message. |

# Chapter 4

## Design

### 4.1 System Architecture

The NeuroAd system consists of three primary subsystems: Data Acquisition, Backend Processing & Machine Learning, and Frontend Interface, with Firebase serving as the cloud-based database and authentication service. This three-tier architecture separates the core functionalities, ensuring modularity and scalability. The overall structure, as depicted in Figure 4.1, illustrates how these components interact to deliver the final output.

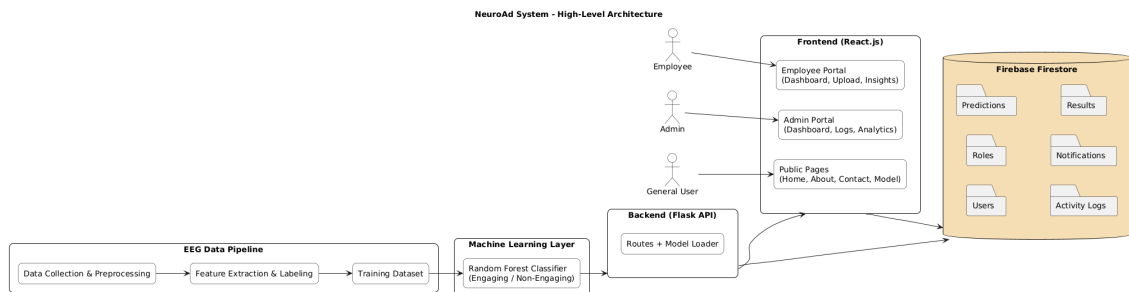


Figure 4.1: High Level System Architecture

### 4.2 Design Constraints

#### 4.2.1 Input Limitations

The current system supports only EEG data in CSV format. It does not process video input directly, which limits flexibility for real-time video-based ad evaluation.

#### **4.2.2 Platform Dependency**

The application is designed specifically as a web-based platform and does not support mobile devices. Ensuring functionality only on browsers is a constraint, as it restricts portability and usability for mobile-first users.

#### **4.2.3 Insights Scope**

The system provides basic engagement insights (attention, memory, overall engagement) but does not yet deliver detailed consumer psychology breakdowns.

#### **4.2.4 Real-Time Tracking**

The system does not support real-time audience reaction tracking beyond EEG and survey data. This constraint prevents live monitoring of engagement during ad broadcasts.

#### **4.2.5 Integration with External Platforms**

Currently, NeuroAd does not integrate with advertisement distribution platforms (e.g., Google Ads, Meta Ads).

#### **4.2.6 Advanced Sentiment Analysis**

The system does not yet incorporate advanced sentiment analysis techniques, such as facial emotion recognition, or natural language processing. This constraint narrows the scope of multimodal consumer engagement evaluation.

#### **4.2.7 User Access**

Only authenticated employees and admins can log in; sign-up is disabled to ensure security.

### **4.3 Design Methodology**

The NeuroAd FYP was developed using a hybrid Agile–Incremental methodology, combining structured, sequential processes for EEG data collection and preprocessing with iterative, feedback-driven development for modeling, prototyping, and feature expansion. This approach ensured data integrity, rapid prototyping, and incremental delivery of features, while maintaining flexibility for future improvements.

### 4.3.1 Methodology Overview

1. Sequential Phases (Waterfall-inspired): Data collection and preprocessing were executed in a planned, linear manner to ensure reliable EEG signals, proper labeling, and reproducible datasets.
2. Iterative Development (Agile/Incremental): Modeling, prototype development, feature addition, testing, and refinement were conducted in short cycles, enabling immediate feedback, continuous improvement, and incremental delivery of working features.

### 4.3.2 Development Phases

The project was broken down into several key development phases, starting from initial requirements and culminating in final deployment. This structured approach allowed for organized progress and clear milestones. Figure 4.2 provides a visual representation of these distinct phases and their sequential flow.

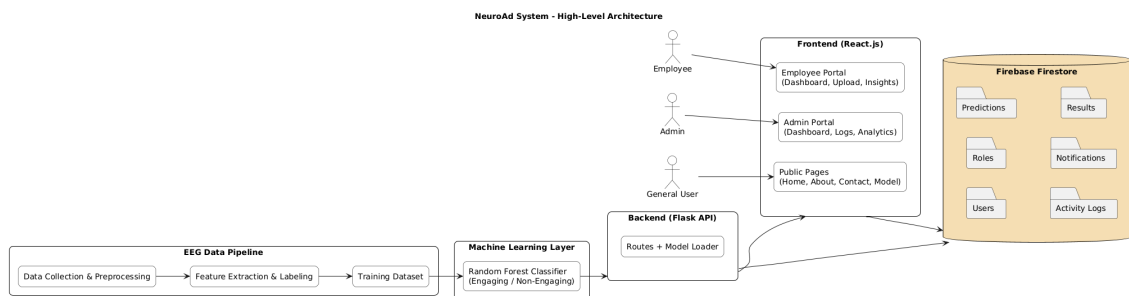


Figure 4.2: Development Phases

## 4.4 High Level Design

This section outlines the architectural views of the system, providing insight into its structure, behavior, and deployment.

### 4.4.1 Conceptual / Logical View

The conceptual view defines the system's main functional components and their logical relationships, independent of implementation details. It provides a high-level understanding of how different parts of the system are organized to meet the requirements. The conceptual diagram in Figure 4.3 illustrates this logical structure.

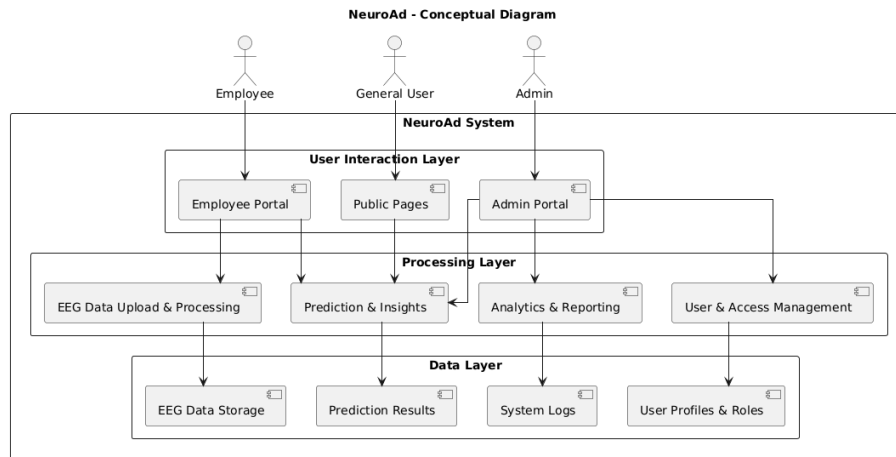


Figure 4.3: Conceptual Diagram

#### 4.4.2 Process View

The process view demonstrates the dynamic behavior of the system, illustrating how data flows and how different processes interact at runtime. This view is crucial for understanding the system's concurrency and synchronization aspects. The workflow is visualized in Figure 4.4.

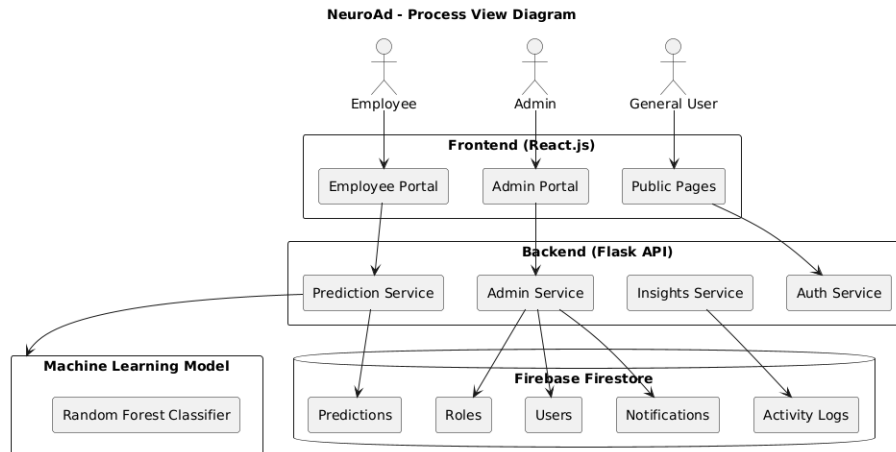


Figure 4.4: Process View

#### 4.4.3 Physical View

The physical view describes the deployment of the system's components onto physical hardware or nodes. It maps the software elements to the physical infrastructure, showing how the system is physically distributed. Figure 4.5 depicts the physical view of the NeuroAd system.

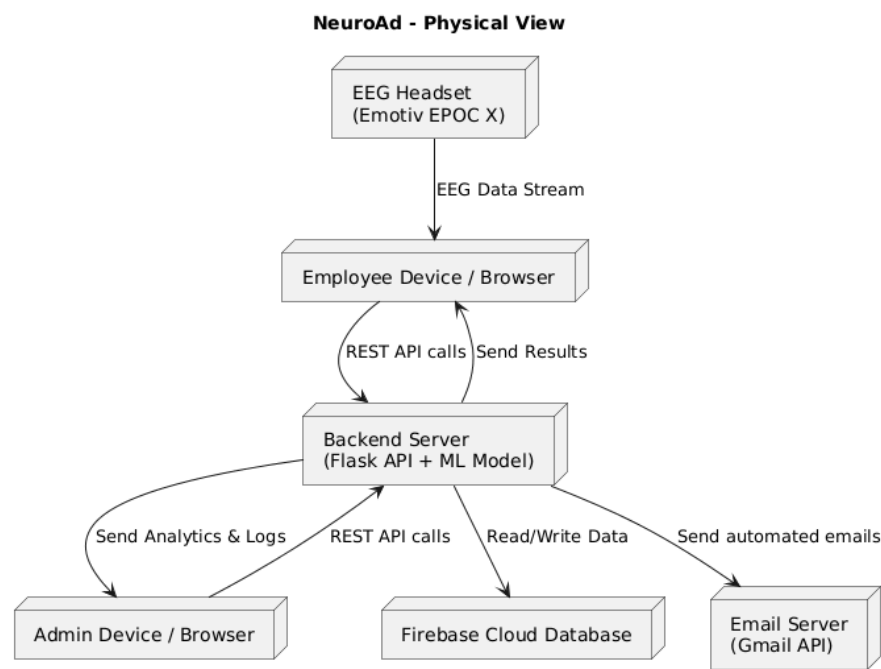


Figure 4.5: Physical View of NeuroAd

**4.4.4 Module View**

The module view breaks down the system into its core modules, showing their dependencies and interfaces. This helps in understanding the system’s decomposition and organization into manageable units. The full module breakdown is presented in Figure 4.6.

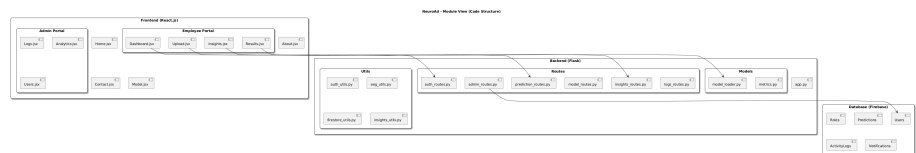


Figure 4.6: Full Module View

**4.4.5 Security View**

The security view outlines the measures taken to protect the system from unauthorized access and threats. It covers authentication, authorization, and data protection mechanisms. The key security components are illustrated in Figure 4.7.

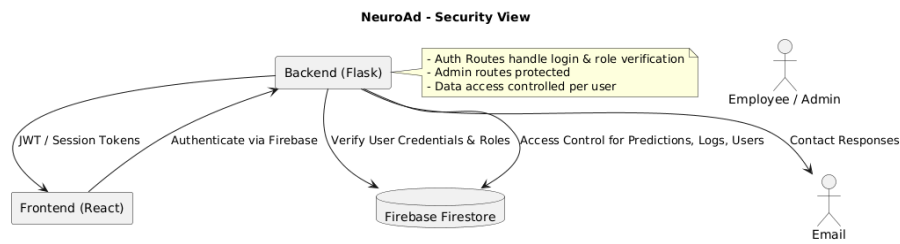


Figure 4.7: Security View

#### 4.4.6 Sequence Diagram

Sequence diagrams illustrate the interactions between objects in a time-ordered sequence. The Employee Sequence Diagram in Figure 4.8 shows the workflow for employee actions, while the Admin Sequence Diagram in Figure 4.9 details admin-specific interactions. A comprehensive overview of all system interactions is provided in the Full Sequence Diagram in Figure 4.10.

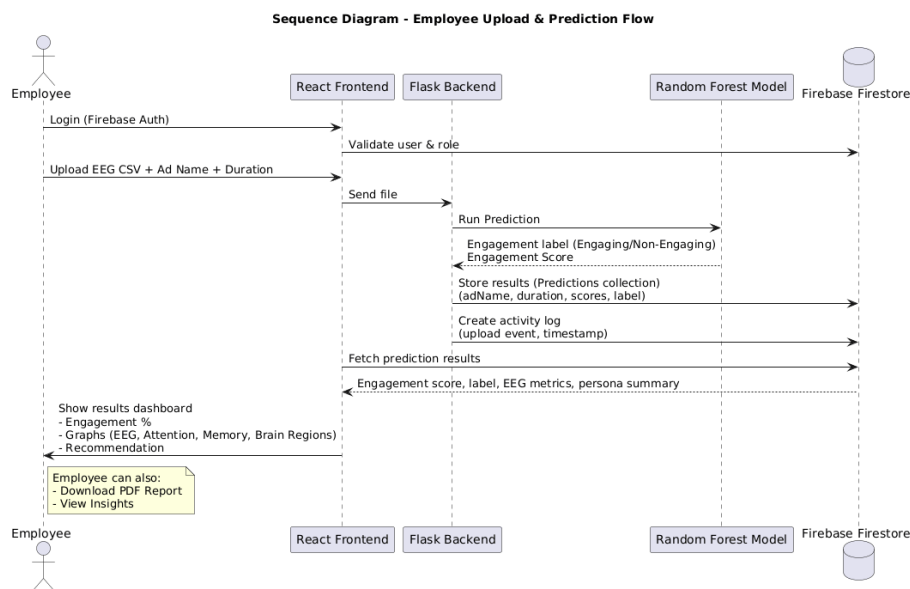


Figure 4.8: Employee Sequence Diagram

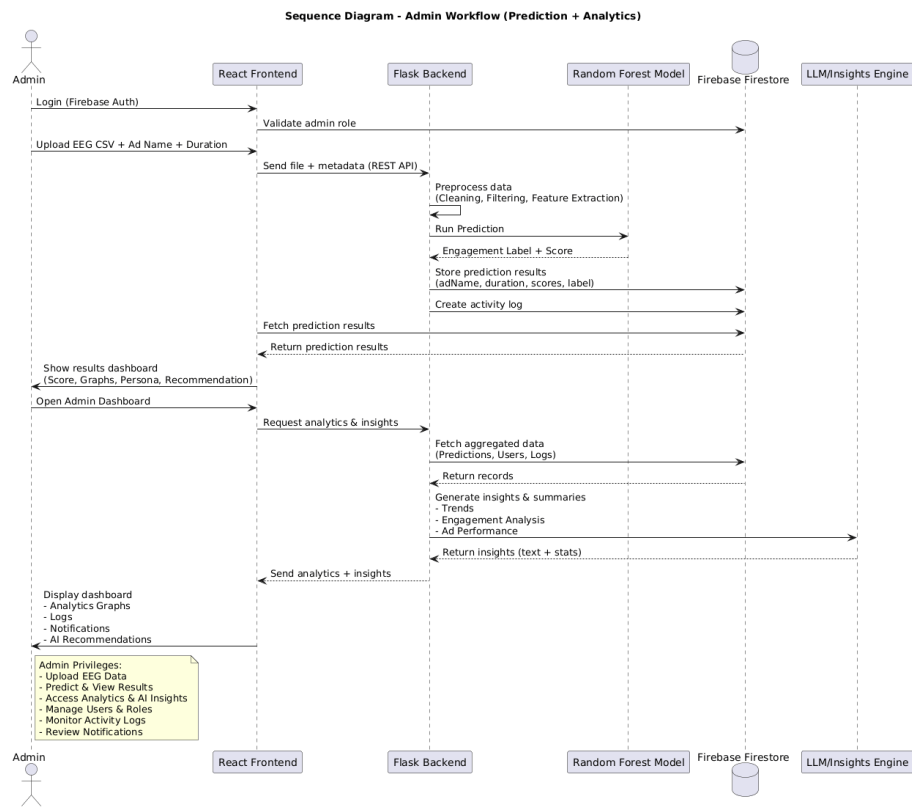


Figure 4.9: Admin Sequence Diagram

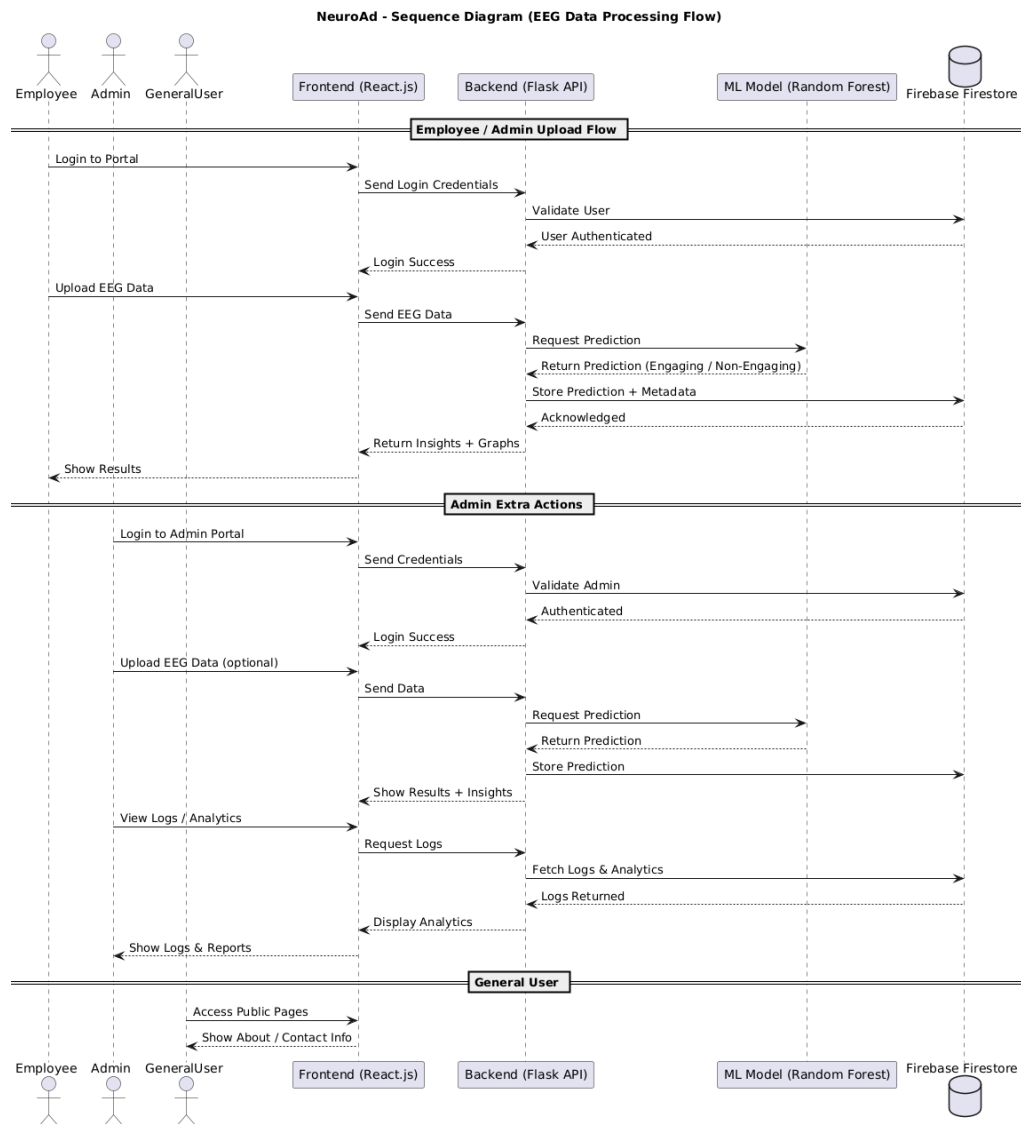


Figure 4.10: Full Sequence Diagram

## 4.5 Low Level Design

This section provides a more detailed look at the internal structure of key modules within the system.

### 4.5.1 Frontend Module

The frontend module is responsible for the user interface and client-side logic. It is built with components that handle user interactions and data presentation. The architecture of the frontend module is detailed in Figure 4.11.

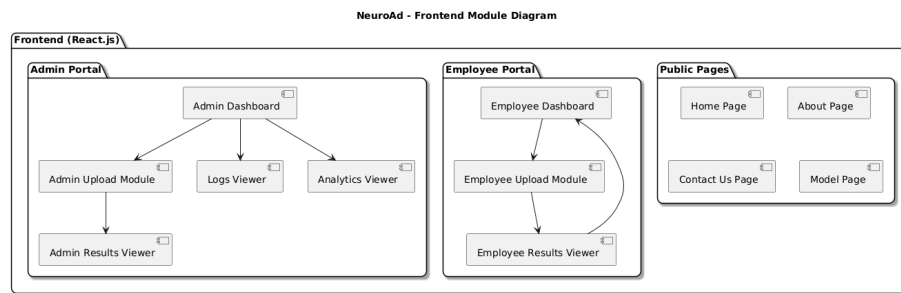


Figure 4.11: Frontend Module

### 4.5.2 Backend Module

The backend module manages server-side logic, data processing, and communication with the database. It contains the core business logic and machine learning model integration. Figure 4.12 illustrates the components of the backend module.

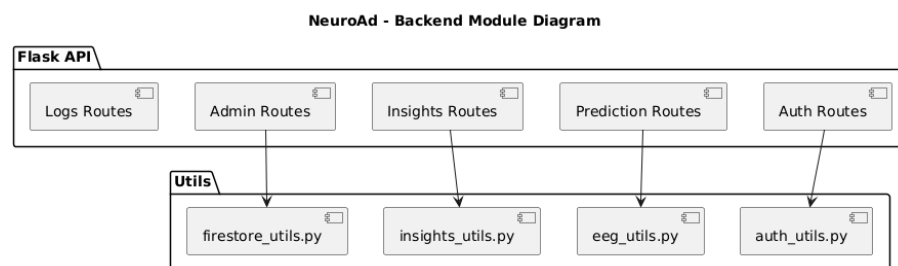


Figure 4.12: Backend Module

## 4.6 Database Design

The database design reveals the structure of the data stores used by the system. This includes the schema for the Firebase Firestore collections and any non-DBMS files. Figure 4.13 shows the organization of the database module.

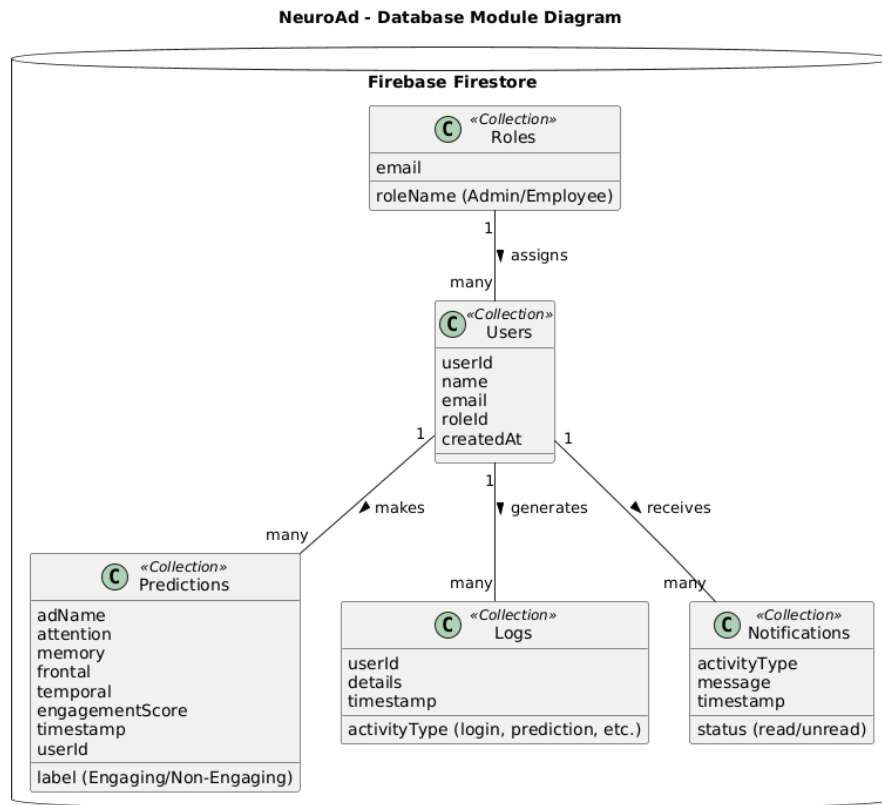


Figure 4.13: Database Module

## 4.7 GUI Design

This section provides the detailed design of the system and subsystem inputs and outputs relative to the user. Depending on the particular nature of the project, it may be appropriate to repeat these sections at both the subsystem and design module levels.

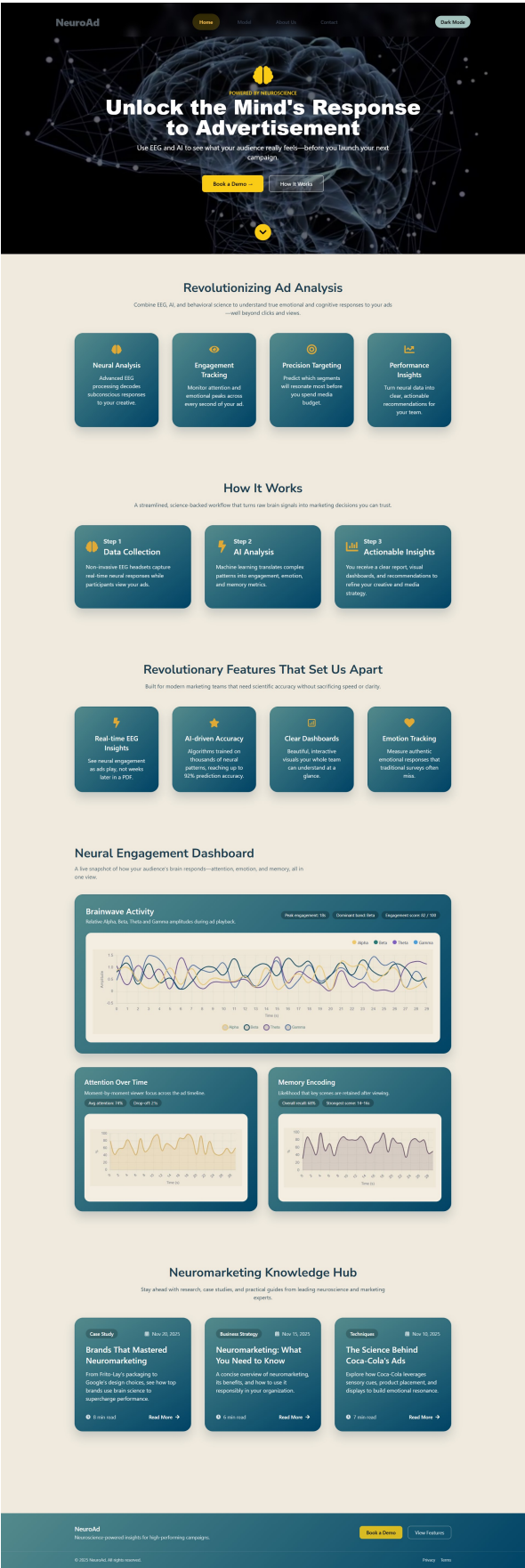


Figure 4.14: Home Page

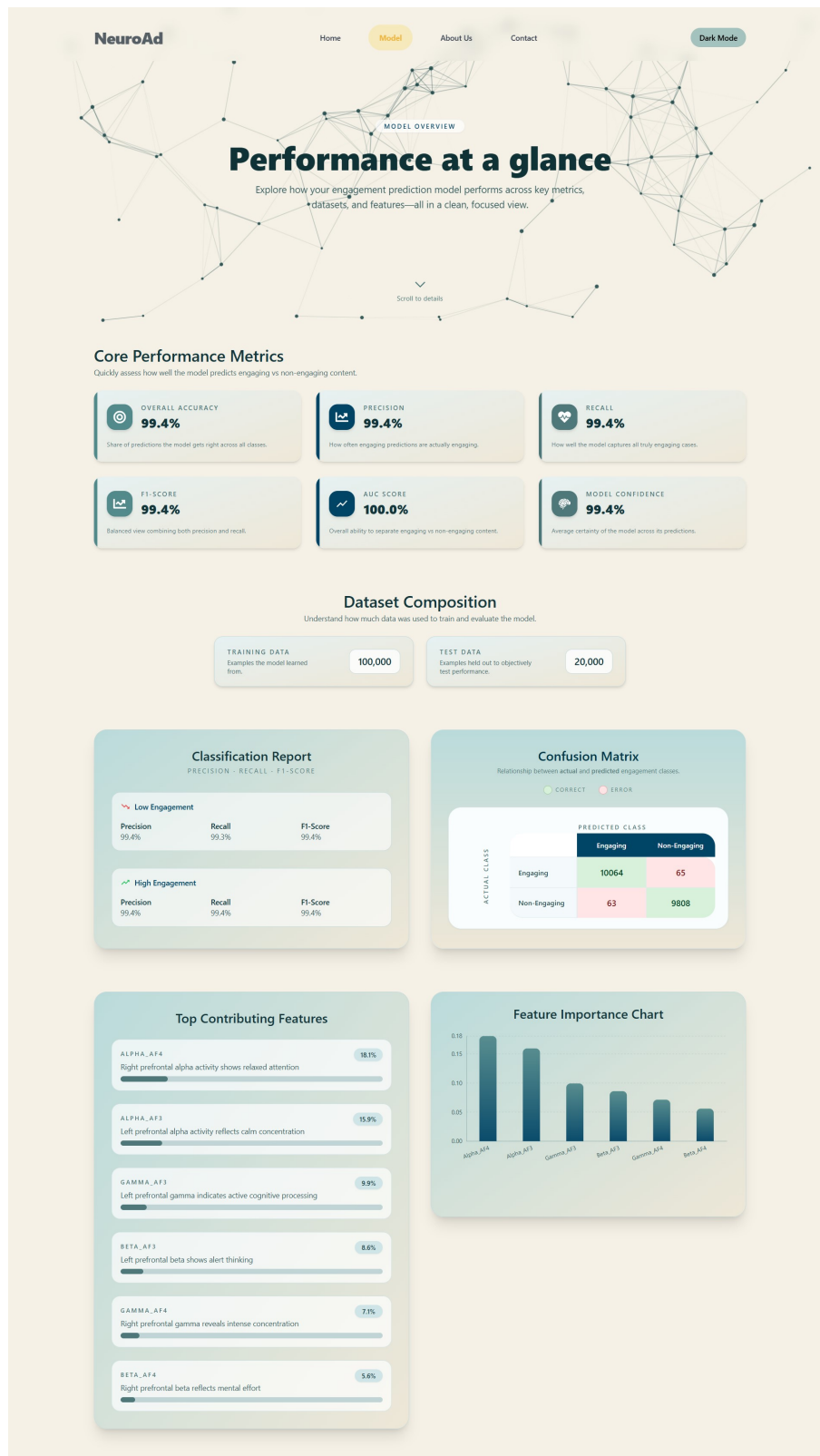


Figure 4.15: Model Page



Figure 4.16: About Page

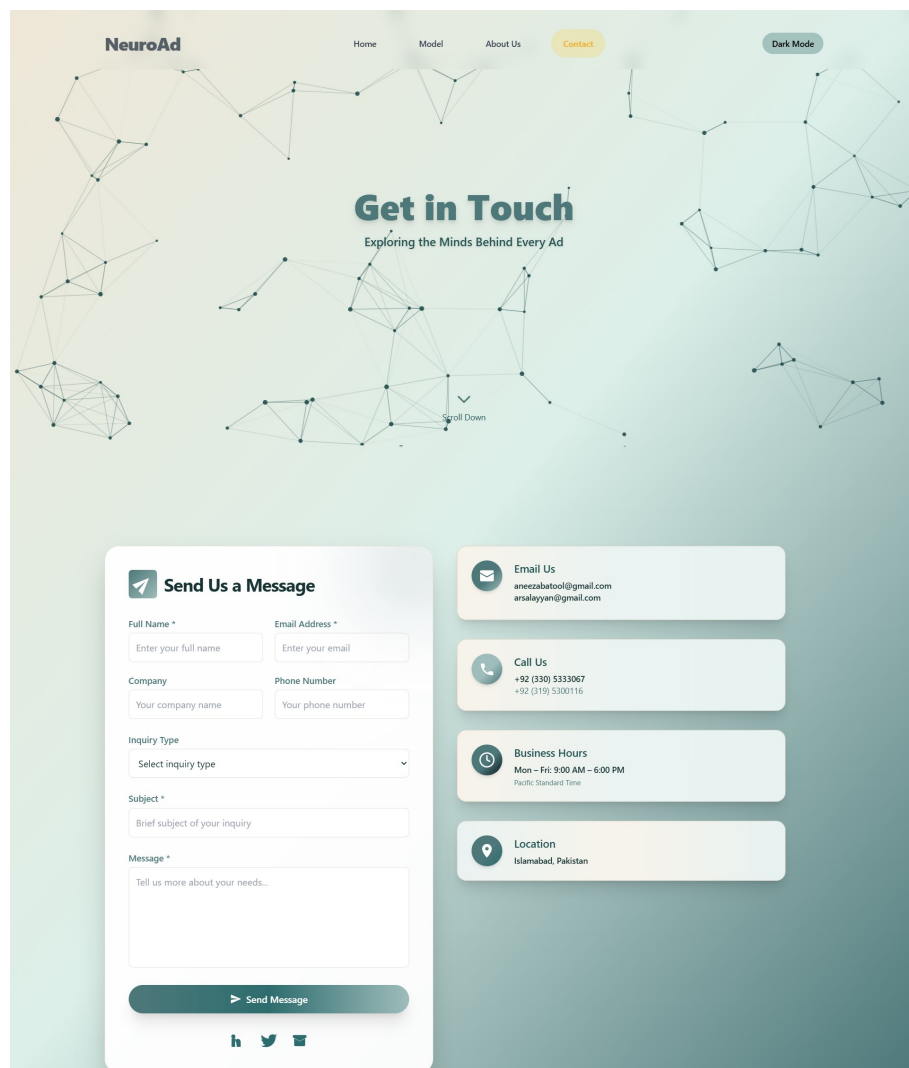


Figure 4.17: Contact Page

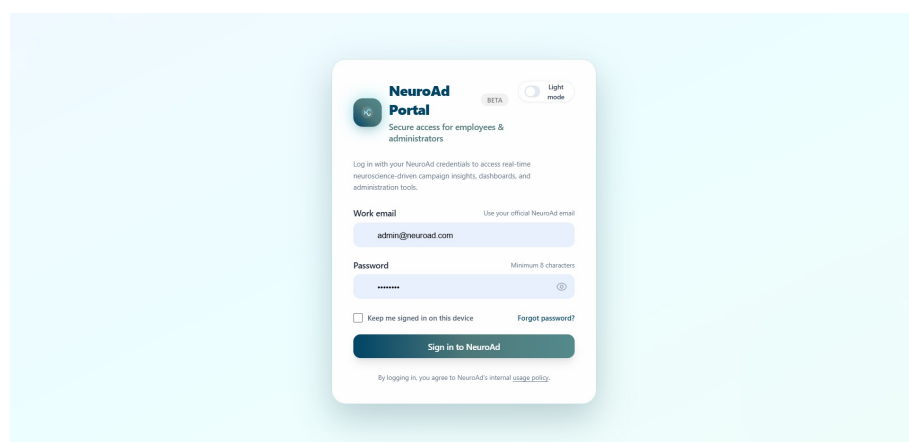


Figure 4.18: Login Page

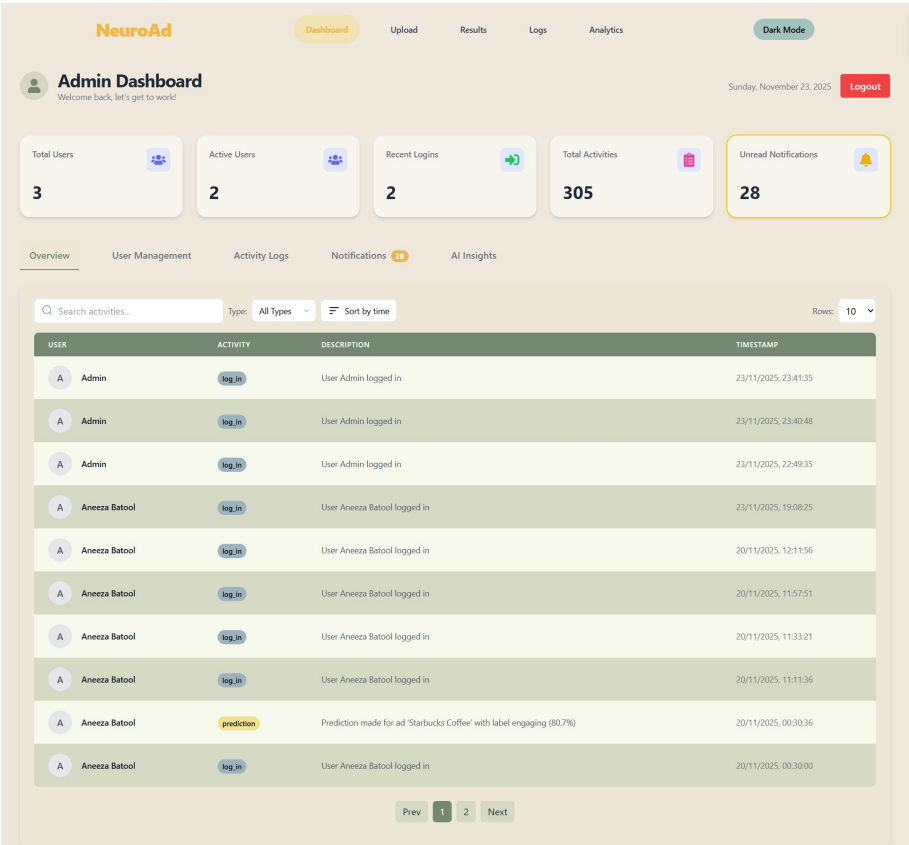


Figure 4.19: Admin Dashboard

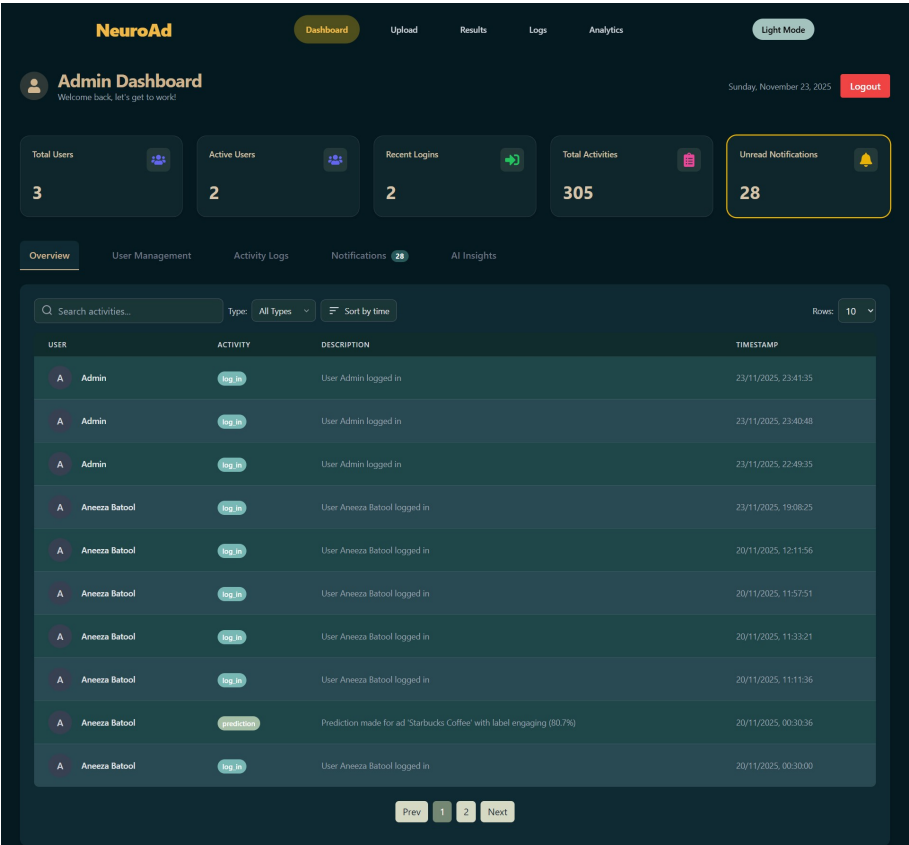


Figure 4.20: Admin Dashboard Dark Mode

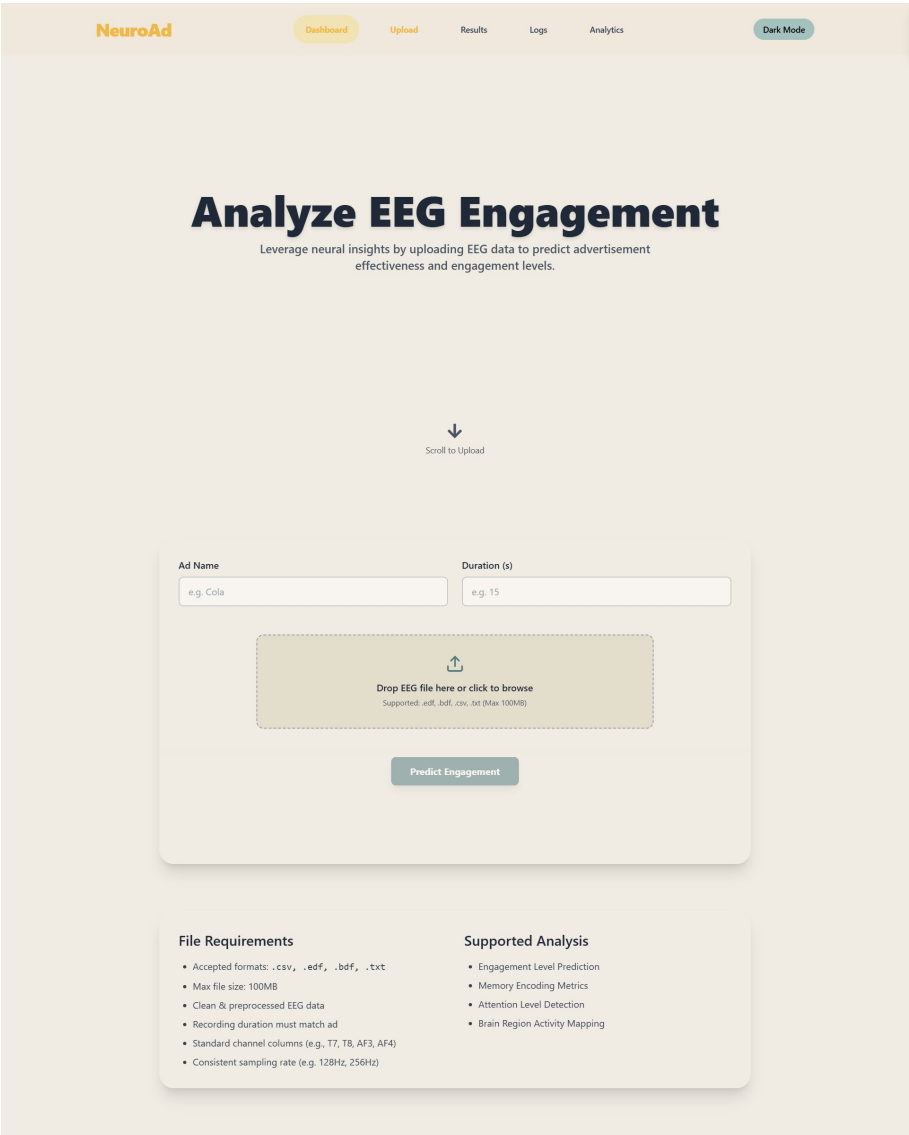


Figure 4.21: Upload Page

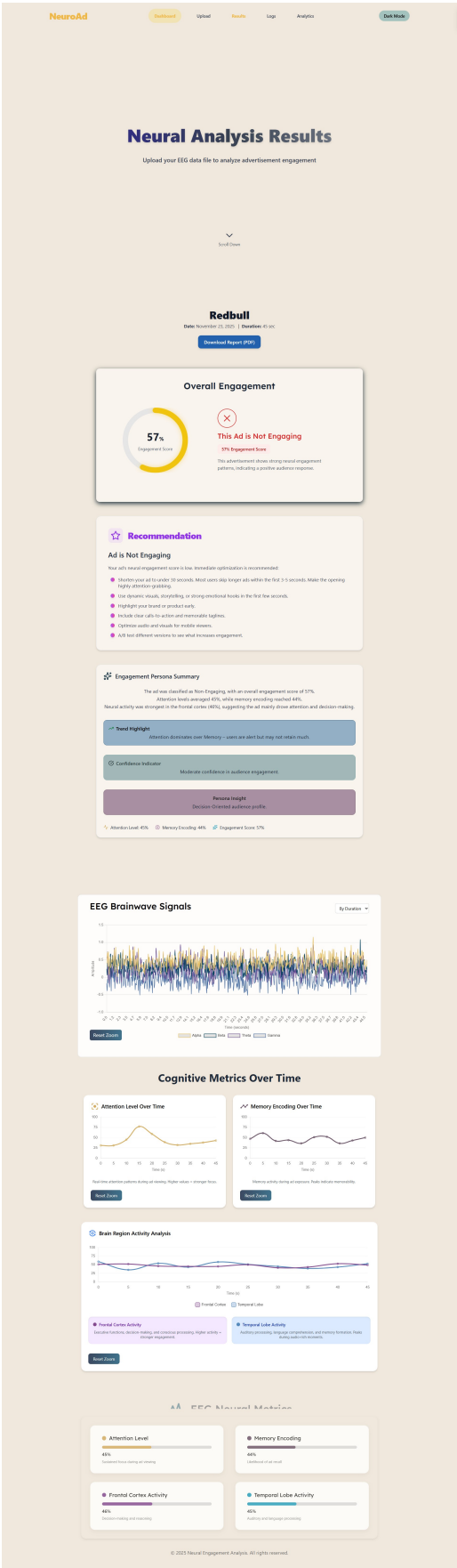


Figure 4.22: Results Page

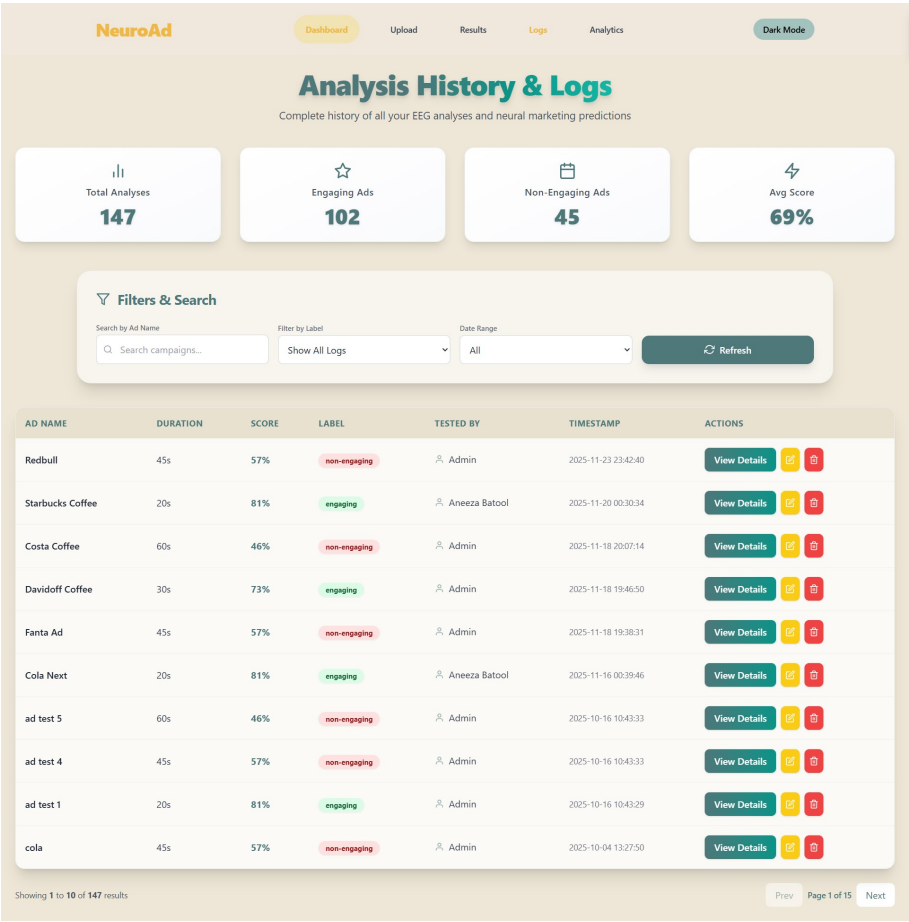


Figure 4.23: Logs Page



Figure 4.24: Analytics Page

4.8 External Interfaces

External systems are any systems that are not within the scope of the system under development. The interface with these external systems, such as the Emotiv headset API and Firebase services, is crucial for system functionality. Figure 4.25 illustrates these connections.

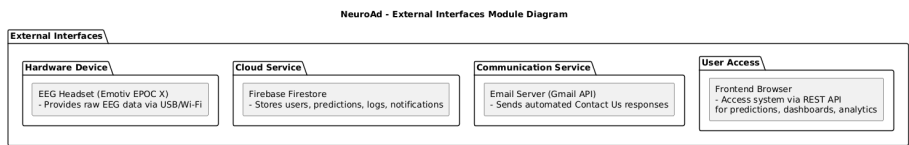


Figure 4.25: External Interfaces

## Chapter 5

# System Implementation

Implementation is the process of moving an idea from concept to reality. The System implementation is a realization of a technical specification or algorithm as a program, software component, or other computer system through programming and deployment.

### 5.1 System Architecture

The system follows a layered architecture consisting of:

1. **Hardware Layer:** EEG acquisition using Emotiv EPOC X headset and a host computer.
2. **Data Processing ML Layer:** Python environment for preprocessing EEG signals and training the Random Forest model.
3. **Backend Layer:** Flask API for handling uploads, predictions, and communication with the frontend, integrated with Firebase for authentication and storage.
4. **Frontend Layer:** React JSX-based dashboard providing real-time visualization of engagement scores and cognitive metrics.
5. **Deployment Layer:** Localhost for development and Firebase Hosting for future scalability.

This architecture ensures smooth data flow from acquisition to visualization while supporting modular upgrades and real-time feedback.

Figure 5.1 illustrates the overall system architecture, showing the layered data flow from EEG acquisition through preprocessing, model inference, backend integration, and frontend visualization.

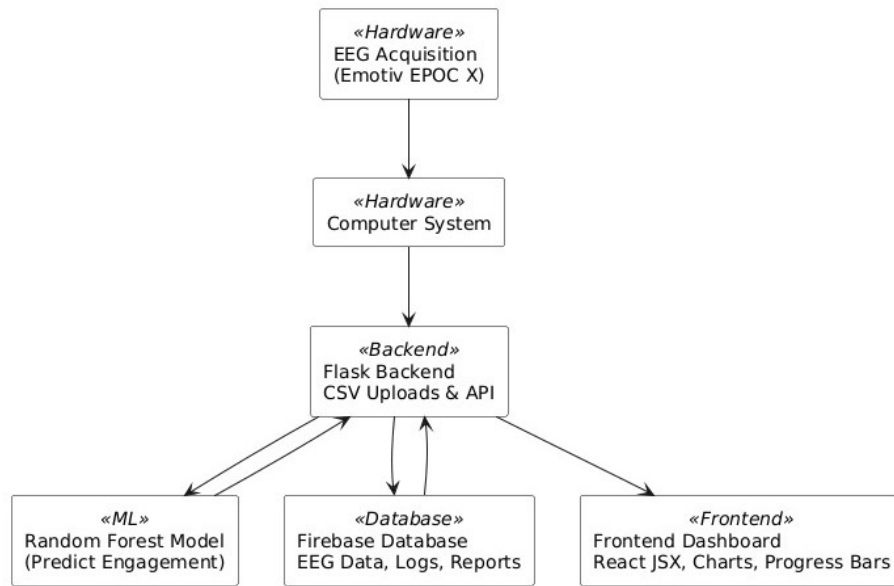


Figure 5.1: System Architecture Flow

## 5.2 System Internal Components

The system is comprised of the following key internal components:

- **Frontend Module:** React-based interface for user interaction and visualization.
- **Backend Module:** Flask API for handling model inference and connecting the frontend with the database.
- **Database Module:** Firebase for authentication, storage, and logging user activity.
- **Machine Learning Module:** Random Forest model for engagement prediction based on EEG features.

## 5.3 Functionality of Components

### 5.3.1 Frontend

The frontend is the primary user-facing component of the system.

- Built using React JSX with Bootstrap for responsiveness.
- Displays real-time engagement scores, progress bars for attention, memory, and brain regions.
- Provides EEG visualizations using Chart.js, downloadable reports, and administrative controls.

- Receives predictions from the backend and presents them interactively to users.

### 5.3.2 Backend

The backend serves as the core processing engine of the application.

- Developed using Flask (Python) to serve REST APIs.
- Handles EEG CSV uploads, preprocesses input, sends data to the ML model, and returns predictions.
- Manages Firebase integration for authentication, logs, and storage.
- Ensures real-time communication between frontend and backend for responsive dashboards.

### 5.3.3 Database (Firebase)

Firebase is utilized as the backend-as-a-service for data management and authentication.

- Stores user authentication data, activity logs, EEG datasets, and report metadata.
- Provides secure access to user-specific data and supports scalable storage.
- Enables real-time updates for dashboards and notifications.

### 5.3.4 Random Forest Model

The machine learning model is the predictive heart of the system.

- Trained on preprocessed EEG features (Alpha, Beta, Gamma, Theta bands) with labeled engagement levels.
- Predicts binary engagement ("Engaged" / "Not Engaged") with 99.4% accuracy.
- Integrated with the backend to provide live predictions for uploaded EEG data.
- Supports retraining and future enhancements such as additional cognitive metrics or real-time streaming.

## 5.4 Tools and Technologies

### 5.4.1 Hardware Tools

- **Emotiv EPOC X EEG Headset:** Captures real-time EEG brainwave data across 14 channels while participants view advertisements, providing high-resolution signals for engagement analysis.

Figure 5.2 shows the Emotiv EPOC X headset used for 14-channel EEG acquisition during advertisement viewing.



Figure 5.2: EEG Headset

- **Computer System:** Used to run data acquisition software, preprocessing scripts, model training, and web deployment.

#### 5.4.2 Software Tools

- **EmotivPRO / Emotiv Launcher:** Interface for recording, managing, and exporting EEG data from the headset into CSV format.

Figure 5.3 displays a sample of raw EEG signal visualization in EmotivPRO used for recording and quality checks.

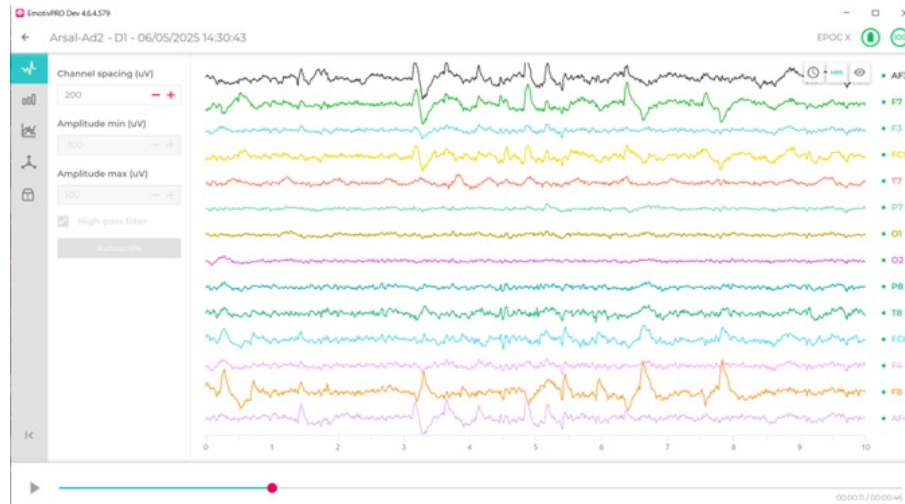


Figure 5.3: Raw EEG Signal Visualization in EmotivPRO

Figure 5.4 presents performance metrics in EmotivPRO used to assess signal quality and participant state during sessions.

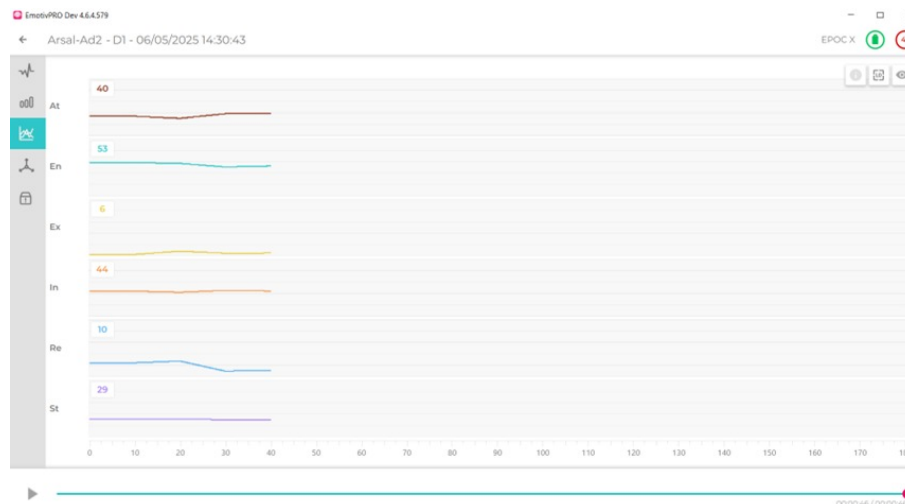


Figure 5.4: Performance Metrics Visualization in EmotivPRO

- **Python:** Primary language for preprocessing, feature extraction, and machine learning model development.
- **Jupyter Notebook:** Used for data analysis, visualization, and iterative model testing.
- **Visual Studio Code:** IDE for backend and frontend code development.

### 5.4.3 Machine Learning & Data Processing Libraries

- **NumPy Pandas:** For data manipulation, feature extraction, and structured dataset handling.

- **SciPy MNE:** For EEG signal processing, filtering, and normalization.
- **Scikit-learn:** For building, training, and evaluating the Random Forest model.
- **Matplotlib Seaborn:** For plotting EEG data, band analysis, and model performance visualization.

#### 5.4.4 Backend Framework Technologies

- **Flask (Python):** Backend API to handle EEG uploads, model inference, and communication with the frontend.
- **Firebase:** Provides authentication, real-time database, activity logs, and cloud storage.
- **REST API:** Enables communication between backend services and the frontend for live predictions.

#### 5.4.5 Frontend Visualization Tools

- **React JSX:** Framework used for building a dynamic, component-based frontend interface.
- **Bootstrap:** Ensures responsive and user-friendly design.
- **Chart.js:** Used to display EEG waveforms, engagement scores, and cognitive progress indicators.

#### 5.4.6 Version Control Collaboration

- **Git GitHub:** Manage source code, version control, and team collaboration throughout development.

#### 5.4.7 Deployment Environment

- **Localhost / Flask Server:** Used for testing live predictions and system functionality during development.
- **Firebase Hosting (Future Integration):** Supports web-based deployment, scalability, and accessibility.

Figure 5.5 summarizes the tools and technologies across the stack, from acquisition and processing to model inference, frontend visualization, and deployment.

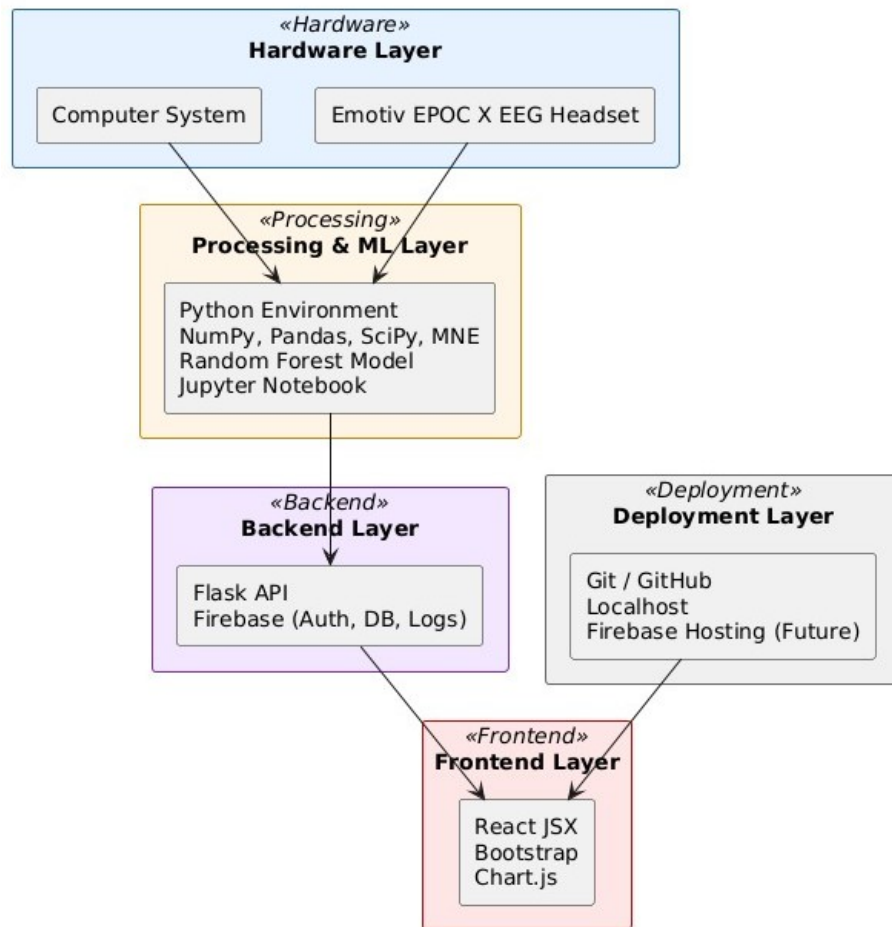


Figure 5.5: Tools and Technologies Architecture

## 5.5 Security

### 5.5.1 Authentication and Access Control

- **User Authentication:** The system uses Firebase Authentication to manage secure login sessions. Users access the platform using registered email and password credentials.
- **Role-Based Access:** Access privileges are role-specific.
  - **Ad Testing Personnel** can upload EEG CSV data, view engagement results, and access their personal dashboard.
  - **Administrators** have extended permissions, including monitoring activity logs, managing users, viewing analytics, and receiving notifications of new activities.
  - **General users** are restricted from accessing the platform.

- This segregation ensures that sensitive operations such as data uploads and system management are restricted to authorized personnel only.

### 5.5.2 Data Security

- **Stored Data:** The database stores user credentials, advertisement metadata, EEG-derived engagement metrics (engagement score, attention, memory encoding, brain region activity), and user activity logs.
- **Data Storage:** All data are stored directly in Firebase Database, ensuring structured and centralized management.
- **Transmission Security:** Communication between the frontend (React) and backend (Flask API) occurs over HTTP with Flask CORS enabled. Although data are not encrypted, secure endpoints are used to manage authorized requests.
- **Data Validation:** Basic input validation is implemented on the frontend, ensuring that email fields follow standard formatting before submission.

### 5.5.3 Backend and API Protection

- The Flask backend handles EEG CSV uploads, processes predictions, and communicates with Firebase.
- Each API request is authenticated through Firebase before processing.
- The backend structure prevents unauthorized access to model endpoints by restricting interaction to verified users only.
- Currently, file sanitization for uploaded EEG data is minimal, but the system restricts access to authenticated users, reducing exposure to injection or tampering risks.

### 5.5.4 System Monitoring and Logging

- The system implements activity logging through Firebase to record all significant events, including user logins, data uploads, and prediction requests.
- Administrators can view detailed logs and access reports summarizing who performed specific actions and when they occurred.
- Real-time notifications are generated whenever new EEG data are uploaded or analyzed, helping maintain transparency and accountability.
- These logs assist in identifying irregular usage patterns, troubleshooting issues, and maintaining an auditable record of system activity.

Figure 5.6 provides a high-level view of the secure system architecture, including authentication, role-based access control, logging, and protected APIs.

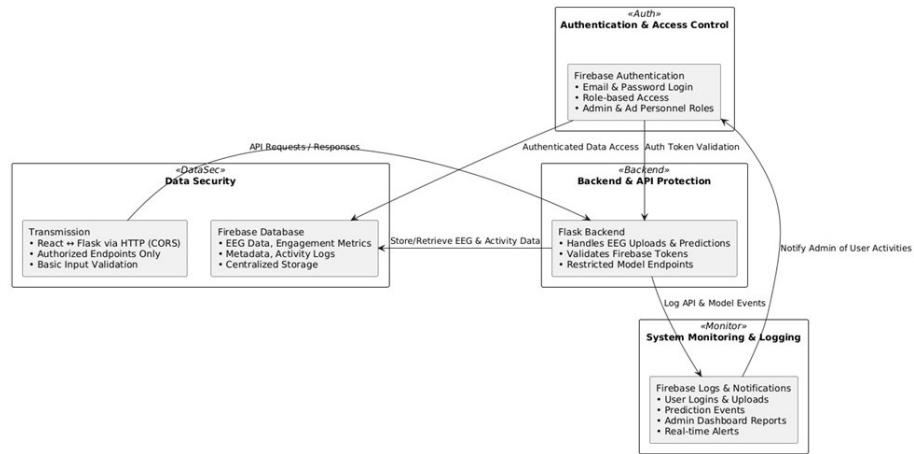


Figure 5.6: Secure System Architecture Overview

## **Chapter 6**

# **System Testing and Evaluation**

System testing of software or hardware is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. Be warned that many projects fall down through poor evaluation. Simply building a system and documenting its design and functionality is not enough to gain top marks. It is extremely important that you evaluate what you have done both in absolute terms and in comparison with existing techniques, software, hardware etc. This might involve quantitative evaluation and qualitative evaluation such as expressibility, functionality, ease-of-use etc. At some point you should also evaluate the strengths and weaknesses of what you have done. Avoid statements like "The project has been a complete success and we have solved all the problems associated with ...!" It is important to understand that there is no such thing as a perfect project. Even the very best pieces of work have their limitations and you are expected to provide a proper critical appraisal of what you have done. The following are different types of testing that should be considered during System testing:

### **6.1 Functional Testing**

#### **6.1.1 Graphical User Interface (GUI) Testing**

## CHAPTER 6 – SYSTEM TESTING AND EVALUATION

Table 6.1: GUI Test Case 1: Home Page Verification

| Test Case 1            |                                                        |                   |                              |               |              |              |                                                                                        |
|------------------------|--------------------------------------------------------|-------------------|------------------------------|---------------|--------------|--------------|----------------------------------------------------------------------------------------|
| Test Scenario ID       | GUI_TS_01                                              |                   |                              | Test Case ID  |              | GUI_TC_01    |                                                                                        |
| Test Case Description  | Verify Home Page elements (logo, menu, responsiveness) |                   |                              |               |              |              |                                                                                        |
| Pre-Requisite          | Website hosted on localhost                            |                   |                              |               |              |              |                                                                                        |
| Post-Requisite         | Home page loaded                                       |                   |                              |               |              |              |                                                                                        |
| Test Execution Steps   |                                                        |                   |                              |               |              |              |                                                                                        |
| S.No                   | Action                                                 | Inputs            | Expected Output              | Actual Output | Test Browser | Test Re-sult | Test Comments                                                                          |
| 1                      | Open site                                              | URL of localhost  | Home page loads              | As expected   | Edge         | Pass         | Home page opened successfully without errors                                           |
| 2                      | Check header, footer, logo                             | Visual inspection | Header, footer, logo visible | As expected   | Edge         | Pass         | All UI was displayed properly, including the logo and menu.                            |
| 3                      | Resize window                                          | Browser resize    | Layout adjusts properly      | As expected   | Edge         | Pass         | Responsive layout does not overlap or cut off any content when using desktop browsers. |
| Continued on next page |                                                        |                   |                              |               |              |              |                                                                                        |

**Table 6.1 – continued from previous page**

| <b>S.No</b> | <b>Action</b>                         | <b>Inputs</b>                                   | <b>Expected Output</b>                                             | <b>Actual Output</b>               | <b>Test Browser</b> | <b>Test Re-sult</b> | <b>Test Comments</b>                                                              |
|-------------|---------------------------------------|-------------------------------------------------|--------------------------------------------------------------------|------------------------------------|---------------------|---------------------|-----------------------------------------------------------------------------------|
| 4           | Click navigation links                | Menu options: Home, About Us, Contact Us, Model | Navigation to correct page                                         | As expected                        | Edge                | Pass                | Links worked properly and directed users to the appropriate pages.                |
| 5           | Test responsiveness on mobile browser | Open site on mobile browser                     | Layout adjusts properly; menu and elements visible without overlap | Layout broken; elements misaligned | Mobile (Edge)       | Fail                | <b>Responsive design failed on mobile browser – layout not adapting properly.</b> |

## CHAPTER 6 – SYSTEM TESTING AND EVALUATION

Table 6.2: GUI Test Case 2: Contact Us Page Verification

| Test Case 2            |                                              |                       |                                   |                       |              |              |                                       |
|------------------------|----------------------------------------------|-----------------------|-----------------------------------|-----------------------|--------------|--------------|---------------------------------------|
| Test Scenario ID       | GUI_TS_02                                    |                       |                                   | Test Case ID          |              | GUI_TC_02    |                                       |
| Test Case Description  | Verify Contact Us form validation and fields |                       |                                   |                       |              |              |                                       |
| Pre-Requisite          | Home page loaded                             |                       |                                   |                       |              |              |                                       |
| Post-Requisite         | Contact Us page loaded                       |                       |                                   |                       |              |              |                                       |
| Test Execution Steps   |                                              |                       |                                   |                       |              |              |                                       |
| S.No                   | Action                                       | Inputs                | Expected Output                   | Actual Output         | Test Browser | Test Re-sult | Test Comments                         |
| 1                      | Navigate to Contact Us                       | Click navigation link | Contact Us page loads             | Contact Us page loads | Edge         | Pass         | Page loaded without missing elements. |
| 2                      | Leave email blank                            | Form input empty      | Validation: “Email is required”   | As expected           | Edge         | Pass         | Proper validation message displayed.  |
| 3                      | Enter invalid email                          | Input: “abc@”         | Validation: “Enter a valid email” | As expected           | Edge         | Pass         | Email validation works correctly.     |
| 4                      | Leave subject/message blank                  | Form input empty      | Validation message for each field | As expected           | Edge         | Pass         | Form enforces required fields.        |
| Continued on next page |                                              |                       |                                   |                       |              |              |                                       |

**Table 6.2 – continued from previous page**

| <b>S.No</b> | <b>Action</b>             | <b>Inputs</b>                               | <b>Expected Output</b> | <b>Actual Output</b> | <b>Test Browser</b> | <b>Test Re-sult</b> | <b>Test Comments</b>                                                       |
|-------------|---------------------------|---------------------------------------------|------------------------|----------------------|---------------------|---------------------|----------------------------------------------------------------------------|
| 5           | Fill all fields correctly | Name, Email, Subject, Message, Inquiry Type | Form ready to submit   | As expected          | Edge                | Pass                | The fields were accepted without error, and the user could send a message. |

**Table 6.3: GUI Test Case 3: Model Page Verification**

| Test Case 3            |                                                                        |                    |                         |               |              |             |                                                |
|------------------------|------------------------------------------------------------------------|--------------------|-------------------------|---------------|--------------|-------------|------------------------------------------------|
| Test Scenario ID       | GUI_TS_03                                                              |                    |                         | Test Case ID  | GUI_TC_03    |             |                                                |
| Test Case Description  | Verify Model page displays all metrics, charts, and feature importance |                    |                         |               |              |             |                                                |
| Pre-Requisite          | Home page loaded                                                       |                    |                         |               |              |             |                                                |
| Post-Requisite         | Model page loaded                                                      |                    |                         |               |              |             |                                                |
| Test Execution Steps   |                                                                        |                    |                         |               |              |             |                                                |
| S.No                   | Action                                                                 | Inputs             | Expected Output         | Actual Output | Test Browser | Test Result | Test Comments                                  |
| 1                      | Navigate to Model page                                                 | Click “Model” link | Page loads successfully | As expected   | Edge         | Pass        | Model page was loaded without broken elements. |
| Continued on next page |                                                                        |                    |                         |               |              |             |                                                |

**Table 6.3 – continued from previous page**

| <b>S.No</b> | <b>Action</b>                                   | <b>Inputs</b>          | <b>Expected Output</b>                                                 | <b>Actual Output</b> | <b>Test Browser</b> | <b>Test Re-sult</b> | <b>Test Comments</b>                             |
|-------------|-------------------------------------------------|------------------------|------------------------------------------------------------------------|----------------------|---------------------|---------------------|--------------------------------------------------|
| 2           | Verify performance metrics                      | Visual check           | Accuracy, Precision, Recall, F1-Score, AUC, Model Confidence displayed | As expected          | Edge                | Pass                | All metrics shown correctly.                     |
| 3           | Verify feature importance                       | Visual check           | Feature importance for Alpha_AF4, Alpha_AF3, Gamma_AF3, etc. displayed | As expected          | Edge                | Pass                | Feature importance values are shown and aligned. |
| 4           | Verify confusion matrix & classification report | Visual check           | Confusion matrix and classification report rendered                    | As expected          | Edge                | Pass                | Charts displayed without any distortion.         |
| 5           | Verify graphs & engagement levels               | Scroll and view graphs | Low/High engagement metrics displayed with graphs                      | As expected          | Edge                | Pass                | Graphs are accurate and legends readable.        |

Table 6.4: GUI Test Case 4: Employee Dashboard GUI Verification

| Test Case 4           |                                                           |                   |                                                                      |               |              |              |                                                |
|-----------------------|-----------------------------------------------------------|-------------------|----------------------------------------------------------------------|---------------|--------------|--------------|------------------------------------------------|
| Test Scenario ID      | GUI_TS_04                                                 |                   |                                                                      | Test Case ID  | GUI_TC_04    |              |                                                |
| Test Case Description | Verify Employee dashboard layout, cards, tabs, and graphs |                   |                                                                      |               |              |              |                                                |
| Pre-Requisite         | Employee logged in                                        |                   |                                                                      |               |              |              |                                                |
| Post-Requisite        | Dashboard loaded                                          |                   |                                                                      |               |              |              |                                                |
| Test Execution Steps  |                                                           |                   |                                                                      |               |              |              |                                                |
| S.No                  | Action                                                    | Inputs            | Expected Output                                                      | Actual Output | Test Browser | Test Re-sult | Test Comments                                  |
| 1                     | Login as employee                                         | Email & password  | Redirected to dashboard                                              | As expected   | Edge         | Pass         | Employee dashboard loaded successfully.        |
| 2                     | Check top cards                                           | Visual inspection | Total Ads, Engaging Ads %, Non-Engaging %, Avg Attention % displayed | As expected   | Edge         | Pass         | All cards visible and labeled correctly.       |
| 3                     | Check recent analysis tab                                 | Click tab         | List of tested ads displayed with metrics                            | As expected   | Edge         | Pass         | All recent analysis were displayed correctly.  |
| 4                     | Check insights tab                                        | Click tab         | AI summary, engagement trends, recommendations displayed             | As expected   | Edge         | Pass         | The insights tab displays the correct content. |

Table 6.5: GUI Test Case 5: Admin Dashboard GUI Verification

| Test Case 5            |                                                       |                                                                  |                                                                                            |               |              |             |                                                |
|------------------------|-------------------------------------------------------|------------------------------------------------------------------|--------------------------------------------------------------------------------------------|---------------|--------------|-------------|------------------------------------------------|
| Test Scenario ID       | GUI_TS_05                                             |                                                                  |                                                                                            | Test Case ID  | GUI_TC_05    |             |                                                |
| Test Case Description  | Verify Admin dashboard layout, tabs, and data display |                                                                  |                                                                                            |               |              |             |                                                |
| Pre-Requisite          | Admin logged in                                       |                                                                  |                                                                                            |               |              |             |                                                |
| Post-Requisite         | Admin dashboard loaded                                |                                                                  |                                                                                            |               |              |             |                                                |
| Test Execution Steps   |                                                       |                                                                  |                                                                                            |               |              |             |                                                |
| S.No                   | Action                                                | Inputs                                                           | Expected Output                                                                            | Actual Output | Test Browser | Test Result | Test Comments                                  |
| 1                      | Login as admin                                        | Email & password                                                 | Redirected to admin dashboard                                                              | As expected   | Edge         | Pass        | Admin dashboard loaded without errors.         |
| 2                      | Verify dashboard cards                                | Visual inspection                                                | Total users, Active users, Recent logins, Total activities, Unread notifications displayed | As expected   | Edge         | Pass        | All the dashboard cards are visible.           |
| 3                      | Check tabs                                            | Click Overview, Users, Activity Logs, Notifications, AI Insights | Each tab loads correct content                                                             | As expected   | Edge         | Pass        | Tabs are functional and display accurate data. |
| 4                      | Verify user data in Users tab                         | Visual inspection                                                | Email, role, tests performed, last login visible                                           | As expected   | Edge         | Pass        | User table is correct and filterable.          |
| Continued on next page |                                                       |                                                                  |                                                                                            |               |              |             |                                                |

**Table 6.5 – continued from previous page**

| S.No | Action               | Inputs                  | Expected Output                                                             | Actual Output | Test Browser | Test Re-sult | Test Comments                                         |
|------|----------------------|-------------------------|-----------------------------------------------------------------------------|---------------|--------------|--------------|-------------------------------------------------------|
| 5    | Verify notifications | Click Notifications tab | Real-time notifications visible                                             | As expected   | Edge         | Pass         | Notifications update immediately; filters functional. |
| 6    | Verify AI Insights   | Click AI Insights tab   | Metrics like Total Predictions, Engagement %, Brain Region Activity visible | As expected   | Edge         | Pass         | AI insights metrics are displayed accurately.         |

**Table 6.6: GUI Test Case 6: Contact Us Form – Invalid Inputs**

| Test Case 6            |                        |                                                                  |                                                              |               |              |              |                        |
|------------------------|------------------------|------------------------------------------------------------------|--------------------------------------------------------------|---------------|--------------|--------------|------------------------|
| Test Scenario ID       |                        | GUI_TS_06                                                        |                                                              | Test Case ID  |              | GUI_TC_06    |                        |
| Test Case Description  |                        | Contact Us Form – Invalid Inputs                                 |                                                              |               |              |              |                        |
| Pre-Requisite          |                        | “Contact Us” page loaded successfully in Edge.                   |                                                              |               |              |              |                        |
| Post-Requisite         |                        | Validation messages should appear for incorrect or empty inputs. |                                                              |               |              |              |                        |
| Test Execution Steps   |                        |                                                                  |                                                              |               |              |              |                        |
| S.No                   | Action                 | Inputs                                                           | Expected Output                                              | Actual Output | Test Browser | Test Re-sult | Test Comments          |
| 1                      | Open “Contact Us” page | -                                                                | Contact form fields (Name, Email, Phone, Message) displayed. | As expected   | Edge         | Pass         | Page loaded correctly. |
| Continued on next page |                        |                                                                  |                                                              |               |              |              |                        |

**Table 6.6 – continued from previous page**

| S.No | Action                                      | Inputs                                                          | Expected Output                                                                                     | Actual Output | Test Browser | Test Re-sult | Test Comments                       |
|------|---------------------------------------------|-----------------------------------------------------------------|-----------------------------------------------------------------------------------------------------|---------------|--------------|--------------|-------------------------------------|
| 2    | Leave all fields blank and click “Send”     | Empty fields                                                    | Validation error: “All fields are required.”                                                        | As expected   | Edge         | Pass         | Error displayed correctly.          |
| 3    | Enter invalid email format                  | testuser@                                                       | “Invalid email format” validation shown.                                                            | As expected   | Edge         | Pass         | Email validation works as expected. |
| 4    | Enter alphabets in phone number field       | abcdef                                                          | “Invalid phone number” validation displayed.                                                        | As expected   | Edge         | Pass         | Phone validation working.           |
| 5    | Enter valid email & phone but empty message | Email: test@mail.com<br>Phone: 03219875432,<br>Message: (blank) | “Message cannot be empty.”                                                                          | As expected   | Edge         | Pass         | Empty message validation works.     |
| 6    | Enter correct inputs and click Send         | Valid entries                                                   | Form submits successfully and confirmation message “Our team will get back to you shortly” appears. | As expected   | Edge         | Pass         | Positive validation verified.       |

## CHAPTER 6 – SYSTEM TESTING AND EVALUATION

Table 6.7: GUI Test Case 7: Login Form – Invalid Credentials

| Test Case 7            |                                                                  |                                 |                                               |               |              |             |                                          |
|------------------------|------------------------------------------------------------------|---------------------------------|-----------------------------------------------|---------------|--------------|-------------|------------------------------------------|
| Test Scenario ID       | GUI_TS_07                                                        |                                 |                                               | Test Case ID  | GUI_TC_07    |             |                                          |
| Test Case Description  | Login Form – Invalid Credentials                                 |                                 |                                               |               |              |             |                                          |
| Pre-Requisite          | Login page accessible; Firebase Auth configured.                 |                                 |                                               |               |              |             |                                          |
| Post-Requisite         | Access denied for invalid credentials; proper message displayed. |                                 |                                               |               |              |             |                                          |
| Test Execution Steps   |                                                                  |                                 |                                               |               |              |             |                                          |
| S.No                   | Action                                                           | Inputs                          | Expected Output                               | Actual Output | Test Browser | Test Result | Test Comments                            |
| 1                      | Open Login Page                                                  | -                               | Email and password input fields visible.      | As expected   | Edge         | Pass        | Page UI correct.                         |
| 2                      | Leave both fields empty and click Login                          | -                               | “Please enter email and password.” displayed. | As expected   | Edge         | Pass        | Blank field validation OK.               |
| 3                      | Enter invalid email format                                       | test@                           | “Invalid email format.” validation displayed. | As expected   | Edge         | Pass        | Frontend validation triggered.           |
| 4                      | Enter un-registered email                                        | unregistered + correct password | “User not found” message shown.               | As expected   | Edge         | Pass        | Firebase correctly rejects invalid user. |
| 5                      | Enter correct email but wrong password                           | admin@mail + wrongpass          | “Incorrect password.” displayed.              | As expected   | Edge         | Pass        | Authentication security verified.        |
| Continued on next page |                                                                  |                                 |                                               |               |              |             |                                          |

Table 6.7 – continued from previous page

| S.No | Action                    | Inputs                   | Expected Output                       | Actual Output | Test Browser | Test Re-sult | Test Comments         |
|------|---------------------------|--------------------------|---------------------------------------|---------------|--------------|--------------|-----------------------|
| 6    | Enter correct credentials | admin@mail + correctpass | Redirected to Dashboard successfully. | As expected   | Edge         | Pass         | Valid login verified. |

Table 6.8: GUI Test Case 8: Upload Page – Incorrect CSV File Handling

| Test Case 8            |                                                   |                                                             |                                                          |               |              |              |                               |
|------------------------|---------------------------------------------------|-------------------------------------------------------------|----------------------------------------------------------|---------------|--------------|--------------|-------------------------------|
| Test Scenario ID       |                                                   | GUI_TS_08                                                   |                                                          | Test Case ID  |              | GUI_TC_08    |                               |
| Test Case Description  |                                                   | Upload Page – Incorrect CSV File Handling                   |                                                          |               |              |              |                               |
| Pre-Requisite          |                                                   | Employee or Admin logged in; Upload page open.              |                                                          |               |              |              |                               |
| Post-Requisite         |                                                   | Invalid files should be rejected with clear error messages. |                                                          |               |              |              |                               |
| Test Execution Steps   |                                                   |                                                             |                                                          |               |              |              |                               |
| S.No                   | Action                                            | Inputs                                                      | Expected Output                                          | Actual Output | Test Browser | Test Re-sult | Test Comments                 |
| 1                      | Open Upload Page                                  | -                                                           | “Choose File”, “Ad Name”, and “Duration” fields visible. | As expected   | Edge         | Pass         | UI elements visible.          |
| 2                      | Click “Predict Engagement” without selecting file | No file                                                     | “Please upload a CSV file.” message displayed.           | As expected   | Edge         | Pass         | Empty upload validation OK.   |
| 3                      | Upload non-CSV file                               | file.pdf                                                    | “Invalid file type. Only .csv allowed.”                  | As expected   | Edge         | Pass         | File type validation working. |
| Continued on next page |                                                   |                                                             |                                                          |               |              |              |                               |

**Table 6.8 – continued from previous page**

| <b>S.No</b> | <b>Action</b>                   | <b>Inputs</b> | <b>Expected Output</b>                                             | <b>Actual Output</b> | <b>Test Browser</b> | <b>Test Re-sult</b> | <b>Test Comments</b>          |
|-------------|---------------------------------|---------------|--------------------------------------------------------------------|----------------------|---------------------|---------------------|-------------------------------|
| 4           | Upload corrupted CSV            | corrupted     | “Unable to read file. Please upload a valid CSV.”                  | As expected          | Edge                | Pass                | Error handling validated.     |
| 5           | Upload CSV with missing columns | incomplete    | “Features.csv mismatch. Please check your CSV format.”             | As expected          | Edge                | Pass                | Feature validation confirmed. |
| 6           | Upload valid CSV                | valid_fea     | File uploaded successfully; “Prediction in progress...” displayed. | As expected          | Edge                | Pass                | Positive upload verified.     |

### 6.1.2 Backend and Model Testing

Table 6.9: Backend Test Case 1: Model Prediction Workflow

| Test Case 1            |                                                    |                                                                                                            |                                                             |               |             |                                 |
|------------------------|----------------------------------------------------|------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------|---------------|-------------|---------------------------------|
| Test Scenario ID       |                                                    | BM_TS_01                                                                                                   |                                                             | Test Case ID  | BM_TC_01    |                                 |
| Test Case Description  |                                                    | Verify that valid CSV data and metadata are correctly processed by the Flask model and stored in Firebase. |                                                             |               |             |                                 |
| Pre-Requisite          |                                                    | Model loaded successfully in Flask backend; Firebase DB connected; valid CSV file available.               |                                                             |               |             |                                 |
| Post-Requisite         |                                                    | Prediction results saved in Firebase and displayed on Results page.                                        |                                                             |               |             |                                 |
| Test Execution Steps   |                                                    |                                                                                                            |                                                             |               |             |                                 |
| S.No                   | Action                                             | Inputs                                                                                                     | Expected Output                                             | Actual Output | Test Result | Test Comments                   |
| 1                      | Upload valid CSV file with correct feature columns | valid_ad.csv                                                                                               | File accepted successfully.                                 | As expected   | Pass        | File uploaded correctly.        |
| 2                      | Send metadata (ad name, duration) to Flask backend | ad_name: “CocaAd”, duration: “30s”                                                                         | Backend receives valid data.                                | As expected   | Pass        | Metadata received correctly.    |
| 3                      | Flask model loads RandomForest model               | -                                                                                                          | Model loaded from joblib successfully.                      | As expected   | Pass        | Model file found and loaded.    |
| 4                      | Prediction executed successfully                   | CSV features                                                                                               | JSON response {prediction: “Engaging”, score: 87.5}         | As expected   | Pass        | Model returned expected output. |
| 5                      | Backend saves prediction to Firebase               | -                                                                                                          | Firebase entry created with timestamp and engagement score. | As expected   | Pass        | DB write verified.              |
| Continued on next page |                                                    |                                                                                                            |                                                             |               |             |                                 |

**Table 6.9 – continued from previous page**

| S.No | Action                             | Inputs | Expected Output                                | Actual Output | Test Result | Test Comments                       |
|------|------------------------------------|--------|------------------------------------------------|---------------|-------------|-------------------------------------|
| 6    | Frontend fetches prediction result | -      | Result and graphs displayed on “Results” page. | As expected   | Pass        | Frontend data consistency verified. |

**Table 6.10: Backend Test Case 2: CSV Feature Validation**

| Test Case 2            |                                        |                                                                                     |                                                                              |               |             |                                         |
|------------------------|----------------------------------------|-------------------------------------------------------------------------------------|------------------------------------------------------------------------------|---------------|-------------|-----------------------------------------|
| Test Scenario ID       |                                        | BM_TS_02                                                                            |                                                                              | Test Case ID  | BM_TC_02    |                                         |
| Test Case Description  |                                        | Verify backend validation when uploaded CSV does not match expected model features. |                                                                              |               |             |                                         |
| Pre-Requisite          |                                        | Flask <code>model_loader.py</code> active with expected features list defined.      |                                                                              |               |             |                                         |
| Post-Requisite         |                                        | Model must reject invalid file and return appropriate error message.                |                                                                              |               |             |                                         |
| Test Execution Steps   |                                        |                                                                                     |                                                                              |               |             |                                         |
| S.No                   | Action                                 | Inputs                                                                              | Expected Output                                                              | Actual Output | Test Result | Test Comments                           |
| 1                      | Upload CSV missing required column     | Missing “Gamma_AF4”                                                                 | API returns {error: “Feature mismatch – missing required column Gamma_AF4.”} | As expected   | Pass        | Validation handled accurately.          |
| 2                      | Upload CSV with columns in wrong order | Features shuffled                                                                   | {error: “Feature arrangement mismatch.”}                                     | As expected   | Pass        | Column order validation correct.        |
| 3                      | Upload completely unrelated CSV        | Random data                                                                         | {error: “Invalid data format. Please upload EEG-based CSV.”}                 | As expected   | Pass        | File type and format validation robust. |
| Continued on next page |                                        |                                                                                     |                                                                              |               |             |                                         |

**Table 6.10 – continued from previous page**

| S.No | Action           | Inputs               | Expected Output              | Actual Output | Test Result | Test Comments                              |
|------|------------------|----------------------|------------------------------|---------------|-------------|--------------------------------------------|
| 4    | Upload valid CSV | All required columns | Model predicts successfully. | As expected   | Pass        | Schema validation bypassed for valid file. |

Table 6.11: Test Case 3: Flask Model Response Consistency

| Test Case 3           |                       |                                                     |                       |                           |             |                                                                     |
|-----------------------|-----------------------|-----------------------------------------------------|-----------------------|---------------------------|-------------|---------------------------------------------------------------------|
| Test Scenario ID      |                       | BE_TS_03                                            |                       | Test Case ID              | BE_TC_03    |                                                                     |
| Test Case Description |                       | Verify repeated predictions give consistent results |                       |                           |             |                                                                     |
| Pre-Requisite         |                       | Flask server running, model loaded                  |                       |                           |             |                                                                     |
| Post-Requisite        |                       | Predictions consistent                              |                       |                           |             |                                                                     |
| Test Execution Steps  |                       |                                                     |                       |                           |             |                                                                     |
| S.No                  | Action                | Inputs                                              | Expected Output       | Actual Output             | Test Result | Test Comments                                                       |
| 1                     | Upload same CSV twice | CSV with same data                                  | Predictions identical | Predictions identical     | Pass        | System deterministic, outputs consistent.                           |
| 2                     | Compare results       | Check engagement scores                             | Scores match exactly  | One score differs by 0.1% | Fail        | Minor inconsistency observed; investigate floating point precision. |

### 6.1.3 Workflow Testing

Table 6.12: Test Case 1: End-to-End Ad Upload &amp; Prediction (Valid Flow)

| Test Case 1            |                         |                                                                                                                                                |                                                                  |               |              |             |                                        |
|------------------------|-------------------------|------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------|---------------|--------------|-------------|----------------------------------------|
| Test Scenario ID       |                         | WF_TS_01                                                                                                                                       |                                                                  | Test Case ID  |              | WF_TC_01    |                                        |
| Test Case Description  |                         | Verify that an authenticated employee/admin can upload an ad CSV, trigger model prediction, and view complete engagement results successfully. |                                                                  |               |              |             |                                        |
| Pre-Requisite          |                         | User logged in; Flask backend, Firebase, and model active.                                                                                     |                                                                  |               |              |             |                                        |
| Post-Requisite         |                         | Prediction results stored in Firebase and visible on frontend.                                                                                 |                                                                  |               |              |             |                                        |
| Test Execution Steps   |                         |                                                                                                                                                |                                                                  |               |              |             |                                        |
| S.No                   | Action                  | Inputs                                                                                                                                         | Expected Output                                                  | Actual Output | Test Browser | Test Result | Test Comments                          |
| 1                      | Login as Employee       | Valid email & password                                                                                                                         | Redirects to Employee Dashboard.                                 | As expected   | Edge         | Pass        | Verified secure login and redirection. |
| 2                      | Navigate to Upload Page | Click “Upload” in sidebar                                                                                                                      | Upload page displays form for Ad Name, Duration, and CSV Upload. | As expected   | Edge         | Pass        | Page loaded correctly.                 |
| 3                      | Upload Valid CSV        | Ad Name: “Coca-Cola_Ad”, Duration: 40s, CSV with correct headers                                                                               | CSV validates successfully.                                      | As expected   | Edge         | Pass        | CSV accepted by model loader.          |
| Continued on next page |                         |                                                                                                                                                |                                                                  |               |              |             |                                        |

**Table 6.12 – continued from previous page**

| <b>S.No</b> | <b>Action</b>              | <b>Inputs</b>                      | <b>Expected Output</b>                                          | <b>Actual Output</b> | <b>Test Browser</b> | <b>Test Re-sult</b> | <b>Test Comments</b>                       |
|-------------|----------------------------|------------------------------------|-----------------------------------------------------------------|----------------------|---------------------|---------------------|--------------------------------------------|
| 4           | Click “Predict Engagement” | –                                  | Flask backend processes request and stores results in Firebase. | As expected          | Edge                | Pass                | Backend prediction completed successfully. |
| 5           | View Results Page          | Auto-redirect or manual navigation | Displays engagement score, label, metadata, and graphs.         | As expected          | Edge                | Pass                | All sections loaded correctly.             |
| 6           | Verify Recommendation      | Engagement Score $\geq$ 65%        | Displays “Ad is Engaging - Great work!” message.                | As expected          | Edge                | Pass                | Verified correct logic for threshold.      |
| 7           | Verify Download PDF        | Click “Download Report”            | PDF downloads with same engagement score and metadata.          | As expected          | Edge                | Pass                | PDF data matches frontend data.            |

## CHAPTER 6 – SYSTEM TESTING AND EVALUATION

Table 6.13: Test Case 2: Invalid Upload Handling (Negative Case)

| Test Case 2           |                                                                                                                                 |                                                         |                                                      |               |              |              |                                    |
|-----------------------|---------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------|------------------------------------------------------|---------------|--------------|--------------|------------------------------------|
| Test Scenario ID      | WF_TS_02                                                                                                                        |                                                         |                                                      | Test Case ID  | WF_TC_02     |              |                                    |
| Test Case Description | Verify that the system handles invalid CSV uploads (mismatched features or missing columns) gracefully and prevents prediction. |                                                         |                                                      |               |              |              |                                    |
| Test Priority         | High                                                                                                                            |                                                         |                                                      |               |              |              |                                    |
| Pre-Requisite         | Logged in as Employee/Admin; Flask backend running.                                                                             |                                                         |                                                      |               |              |              |                                    |
| Post-Requisite        | No invalid data stored in Firebase.                                                                                             |                                                         |                                                      |               |              |              |                                    |
| Test Execution Steps  |                                                                                                                                 |                                                         |                                                      |               |              |              |                                    |
| S.No                  | Action                                                                                                                          | Inputs                                                  | Expected Output                                      | Actual Output | Test Browser | Test Re-sult | Test Comments                      |
| 1                     | Navigate to Upload Page                                                                                                         | –                                                       | Upload form loads successfully.                      | As expected   | Edge         | Pass         | Verified form access.              |
| 2                     | Upload Invalid CSV                                                                                                              | Ad Name: “BokesAd”, Duration: 30s; CSV missing features | Error: “CSV features mismatch. Upload correct file.” | As expected   | Edge         | Pass         | System validation works correctly. |
| 3                     | Click Predict Engagement                                                                                                        | Disabled button or backend rejection                    | Error toast: “Prediction failed – invalid file.”     | As expected   | Edge         | Pass         | Backend safely rejects bad data.   |
| 4                     | Check Firebase                                                                                                                  | –                                                       | No entry created for failed upload.                  | As expected   | Edge         | Pass         | Confirmed DB integrity maintained. |
| 5                     | Retry Valid CSV                                                                                                                 | Ad Name: “Retry_Ad”; Valid CSV                          | Prediction succeeds normally.                        | As expected   | Edge         | Pass         | Verified recovery after error.     |

## CHAPTER 6 – SYSTEM TESTING AND EVALUATION

Table 6.14: Test Case 3: Results Display and Engagement Categorization

| Test Case 3            |                                                                                                                               |                                                                                     |                                                         |               |              |             |                                     |
|------------------------|-------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|---------------------------------------------------------|---------------|--------------|-------------|-------------------------------------|
| Test Scenario ID       | WF_TS_03                                                                                                                      |                                                                                     |                                                         | Test Case ID  | WF_TC_03     |             |                                     |
| Test Case Description  | Verify that results for Poor, Moderate, and High engagement scores display correct categories, metadata, and recommendations. |                                                                                     |                                                         |               |              |             |                                     |
| Test Priority          | Critical                                                                                                                      |                                                                                     |                                                         |               |              |             |                                     |
| Pre-Requisite          | Valid model running; user logged in.                                                                                          |                                                                                     |                                                         |               |              |             |                                     |
| Post-Requisite         | Results correctly categorized and displayed.                                                                                  |                                                                                     |                                                         |               |              |             |                                     |
| Test Execution Steps   |                                                                                                                               |                                                                                     |                                                         |               |              |             |                                     |
| S.No                   | Action                                                                                                                        | Inputs                                                                              | Expected Output                                         | Actual Output | Test Browser | Test Result | Test Comments                       |
| 1                      | Upload Ad for Poor Engagement                                                                                                 | Score = 45%;<br>Label: Not Engaging;<br>Recommendation: “Improve emotional appeal.” | Category “Poor” displayed with label and recommendation | As expected   | Edge         | Pass        | Verified low-score handling.        |
| 2                      | Upload Ad for Moderate Engagement                                                                                             | Score = 68%;<br>Label: Moderately Engaging;<br>Suggestion: “Enhance memorability.”  | Category “Moderate” displayed with suggestion           | As expected   | Edge         | Pass        | Moderate category handled properly. |
| Continued on next page |                                                                                                                               |                                                                                     |                                                         |               |              |             |                                     |

**Table 6.14 – continued from previous page**

| <b>S.No</b> | <b>Action</b>                 | <b>Inputs</b>                                                                                     | <b>Expected Output</b>                        | <b>Actual Output</b> | <b>Test Browser</b> | <b>Test Re-sult</b> | <b>Test Comments</b>                        |
|-------------|-------------------------------|---------------------------------------------------------------------------------------------------|-----------------------------------------------|----------------------|---------------------|---------------------|---------------------------------------------|
| 3           | Upload Ad for High Engagement | Score > 90%;<br>Label: Highly Engaging;<br>Recommendation: “Excellent—maintain current approach.” | Category “High” displayed with recommendation | As expected          | Edge                | Pass                | Verified top-tier engagement display.       |
| 4           | Verify Metadata Display       | – (Shows Attention %, Memory %, Frontal %, Temporal %, Confidence.)                               | All listed metrics rendered                   | As expected          | Edge                | Pass                | All metrics loaded.                         |
| 5           | Verify Graphs                 | EEG Brainwave, Attention vs Memory, Brain Region graphs                                           | Graphs render correctly for all categories    | As expected          | Edge                | Pass                | Graphs render correctly for all categories. |

## CHAPTER 6 – SYSTEM TESTING AND EVALUATION

Table 6.15: Test Case 4: Data Consistency Between Backend, Firebase, and Frontend

| Test Case 4           |                        |                                                                                                                        |                                     |               |              |             |                                |
|-----------------------|------------------------|------------------------------------------------------------------------------------------------------------------------|-------------------------------------|---------------|--------------|-------------|--------------------------------|
| Test Scenario ID      |                        | WF_TS_04                                                                                                               |                                     | Test Case ID  |              | WF_TC_04    |                                |
| Test Case Description |                        | Verify consistency of engagement scores, metadata, and timestamps between Flask backend, Firebase, and React frontend. |                                     |               |              |             |                                |
| Test Priority         |                        | High                                                                                                                   |                                     |               |              |             |                                |
| Pre-Requisite         |                        | Prediction completed successfully.                                                                                     |                                     |               |              |             |                                |
| Post-Requisite        |                        | All data consistent across layers.                                                                                     |                                     |               |              |             |                                |
| Test Execution Steps  |                        |                                                                                                                        |                                     |               |              |             |                                |
| S.No                  | Action                 | Inputs                                                                                                                 | Expected Output                     | Actual Output | Test Browser | Test Result | Test Comments                  |
| 1                     | Upload Valid CSV       | “Consistency_Prediction”                                                                                               | Prediction completes.               | As expected   | Edge         | Pass        | Base prediction completed.     |
| 2                     | Check Flask API Output | /predict endpoint                                                                                                      | Engagement = 72.4%                  | As expected   | Edge         | Pass        | Backend returns correct score. |
| 3                     | Check Firebase Entry   | Ad Name: “Consistency_Ad”                                                                                              | Same score and metadata stored.     | As expected   | Edge         | Pass        | Firebase matches backend.      |
| 4                     | View Results Page      | –                                                                                                                      | Frontend shows identical data.      | As expected   | Edge         | Pass        | UI displays synced info.       |
| 5                     | Compare Graph Data     | –                                                                                                                      | Graphs reflect same values from DB. | As expected   | Edge         | Pass        | Verified visual consistency.   |

Table 6.16: Test Case 5: PDF Report Generation and Data Integrity

| Test Case 5           |                         |                                                                                                             |                                                           |               |              |             |                                 |
|-----------------------|-------------------------|-------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------|---------------|--------------|-------------|---------------------------------|
| Test Scenario ID      |                         | WF_TS_05                                                                                                    |                                                           | Test Case ID  |              | WF_TC_05    |                                 |
| Test Case Description |                         | Verify that downloaded or shared PDF report contains accurate engagement data and matches frontend display. |                                                           |               |              |             |                                 |
| Test Priority         |                         | Medium                                                                                                      |                                                           |               |              |             |                                 |
| Pre-Requisite         |                         | Prediction already completed and displayed.                                                                 |                                                           |               |              |             |                                 |
| Post-Requisite        |                         | PDF file downloaded successfully.                                                                           |                                                           |               |              |             |                                 |
| Test Execution Steps  |                         |                                                                                                             |                                                           |               |              |             |                                 |
| S.No                  | Action                  | Inputs                                                                                                      | Expected Output                                           | Actual Output | Test Browser | Test Result | Test Comments                   |
| 1                     | Go to Results Page      | Ad: “Report_Test_Ad”                                                                                        | Page loads successfully.                                  | As expected   | Edge         | Pass        | Page accessible.                |
| 2                     | Click “Download Report” | –                                                                                                           | PDF generated successfully.                               | As expected   | Edge         | Pass        | Verified successful generation. |
| 3                     | Open PDF                | –                                                                                                           | Contains same engagement %, metadata, and recommendation. | As expected   | Edge         | Pass        | Verified matching values.       |
| 4                     | Verify Graphs in PDF    | –                                                                                                           | Graphs visible with correct labeling.                     | As expected   | Edge         | Pass        | Verified graph rendering.       |

## 6.2 Non-Functional Testing

### 6.2.1 Performance Testing

## CHAPTER 6 – SYSTEM TESTING AND EVALUATION

Table 6.17: Test Case 1: Homepage Load Time Verification

| Test Case 1           |                                 |                                                                                    |                              |               |              |             |                                       |
|-----------------------|---------------------------------|------------------------------------------------------------------------------------|------------------------------|---------------|--------------|-------------|---------------------------------------|
| Test Scenario ID      |                                 | PERF_TS_01                                                                         |                              | Test Case ID  |              | PERF_TC_01  |                                       |
| Test Case Description |                                 | Verify that the homepage loads completely within acceptable response time on Edge. |                              |               |              |             |                                       |
| Test Priority         |                                 | Medium                                                                             |                              |               |              |             |                                       |
| Pre-Requisite         |                                 | Web server and React app running in staging environment.                           |                              |               |              |             |                                       |
| Post-Requisite        |                                 | Homepage fully loaded with all assets (header, footer, images, model page link).   |                              |               |              |             |                                       |
| Test Execution Steps  |                                 |                                                                                    |                              |               |              |             |                                       |
| S.No                  | Action                          | Inputs                                                                             | Expected Output              | Actual Output | Test Browser | Test Result | Test Comments                         |
| 1                     | Launch Homepage URL             | http://localhost                                                                   | Page loads within 3 seconds. | 2.6 s         | Edge         | Pass        | Homepage loaded fast with all images. |
| 2                     | Reload Page (Cache Disabled)    | Ctrl + F5                                                                          | Loads within 5 seconds.      | 4.3 s         | Edge         | Pass        | Acceptable uncached performance.      |
| 3                     | Simulate 20 concurrent requests | JMeter / Locust                                                                    | Average response < 5 s       | 4.9 s         | Edge         | Pass        | System stable under small load.       |

Table 6.18: Test Case 2: Login API Response Time

| Test Case 2           |                               |                                                                                                     |                                         |               |              |             |                                    |
|-----------------------|-------------------------------|-----------------------------------------------------------------------------------------------------|-----------------------------------------|---------------|--------------|-------------|------------------------------------|
| Test Scenario ID      |                               | PERF_TS_02                                                                                          |                                         | Test Case ID  |              | PERF_TC_02  |                                    |
| Test Case Description |                               | Verify that login API via Firebase Auth responds quickly and handles multiple users simultaneously. |                                         |               |              |             |                                    |
| Test Priority         |                               | High                                                                                                |                                         |               |              |             |                                    |
| Pre-Requisite         |                               | Valid Firebase Auth setup.                                                                          |                                         |               |              |             |                                    |
| Post-Requisite        |                               | Users logged in successfully or gracefully rejected.                                                |                                         |               |              |             |                                    |
| Test Execution Steps  |                               |                                                                                                     |                                         |               |              |             |                                    |
| S.No                  | Action                        | Inputs                                                                                              | Expected Output                         | Actual Output | Test Browser | Test Result | Test Comments                      |
| 1                     | Attempt valid login           | Valid credentials                                                                                   | Response < 2 s; redirects to dashboard. | 1.4 s         | Edge         | Pass        | Smooth login performance.          |
| 2                     | Attempt invalid login         | Wrong password                                                                                      | Error displayed in < 2 s, no lag.       | 1.8 s         | Edge         | Pass        | Quick error feedback.              |
| 3                     | Simulate 30 concurrent logins | Multiple users via JMeter                                                                           | Avg response < 3 s                      | 2.7 s         | Edge         | Pass        | Handled concurrent auth correctly. |

Table 6.19: Test Case 3: CSV Upload and Prediction Processing Time

| Test Case 3           |                              |                                                                                    |                                           |               |              |              |                                       |
|-----------------------|------------------------------|------------------------------------------------------------------------------------|-------------------------------------------|---------------|--------------|--------------|---------------------------------------|
| Test Scenario ID      |                              | PERF_TS_03                                                                         |                                           | Test Case ID  |              | PERF_TC_03   |                                       |
| Test Case Description |                              | Verify upload and prediction response time for small, medium, and large CSV files. |                                           |               |              |              |                                       |
| Test Priority         |                              | Critical                                                                           |                                           |               |              |              |                                       |
| Pre-Requisite         |                              | Flask API and model loaded in memory.                                              |                                           |               |              |              |                                       |
| Post-Requisite        |                              | Results stored in Firebase.                                                        |                                           |               |              |              |                                       |
| Test Execution Steps  |                              |                                                                                    |                                           |               |              |              |                                       |
| S.No                  | Action                       | Inputs                                                                             | Expected Output                           | Actual Output | Test Browser | Test Re-sult | Test Comments                         |
| 1                     | Upload small CSV (10 rows)   | Valid features                                                                     | Total process < 3 s                       | 2.2 s         | Edge         | Pass         | Fast upload and prediction.           |
| 2                     | Upload medium CSV (100 rows) | Valid features                                                                     | < 5 s total time                          | 4.6 s         | Edge         | Pass         | Acceptable performance.               |
| 3                     | Upload large CSV (500 rows)  | Valid features                                                                     | Warning if > 8 s                          | 7.8 s         | Edge         | Pass         | Slight delay, still within threshold. |
| 4                     | Upload malformed CSV         | Missing features                                                                   | System rejects in < 2 s, no backend hang. | 1.9 s         | Edge         | Pass         | Good rejection handling speed.        |

Table 6.20: Test Case 4: Results Page Load and Graph Rendering Performance

| Test Case 4           |                                |                                                                                                                                             |                                              |               |              |             |                                |
|-----------------------|--------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------|---------------|--------------|-------------|--------------------------------|
| Test Scenario ID      |                                | PERF_TS_04                                                                                                                                  |                                              | Test Case ID  |              | PERF_TC_04  |                                |
| Test Case Description |                                | Verify the load time of the Results page, including data fetching from Firebase and graph rendering (EEG signals, cognitive metrics, etc.). |                                              |               |              |             |                                |
| Test Priority         |                                | High                                                                                                                                        |                                              |               |              |             |                                |
| Pre-Requisite         |                                | Prediction results already stored in Firebase.                                                                                              |                                              |               |              |             |                                |
| Post-Requisite        |                                | Results rendered without lag or timeout.                                                                                                    |                                              |               |              |             |                                |
| Test Execution Steps  |                                |                                                                                                                                             |                                              |               |              |             |                                |
| S.No                  | Action                         | Inputs                                                                                                                                      | Expected Output                              | Actual Output | Test Browser | Test Result | Test Comments                  |
| 1                     | Open Results Page              | –                                                                                                                                           | Page and data load < 4 s                     | 3.1 s         | Edge         | Pass        | Smooth transition from upload. |
| 2                     | Measure Graph Render Time      | –                                                                                                                                           | Graphs render < 2 s after data fetch.        | 1.6 s         | Edge         | Pass        | Charts loaded quickly.         |
| 3                     | Scroll through multiple graphs | 4 EEG charts                                                                                                                                | No lag or frame drop.                        | As expected   | Edge         | Pass        | Smooth rendering confirmed.    |
| 4                     | Simulate weak network          | 3 Mbps                                                                                                                                      | Page loads within 8 s, with spinner visible. | 7.5 s         | Edge         | Pass        | Acceptable on slow network.    |

Table 6.21: Test Case 5: Concurrent Prediction Requests Load Test

| Test Case 5           |                                   |                                                                                                      |                                |               |              |                  |                             |
|-----------------------|-----------------------------------|------------------------------------------------------------------------------------------------------|--------------------------------|---------------|--------------|------------------|-----------------------------|
| Test Scenario ID      |                                   | PERF_TS_05                                                                                           |                                | Test Case ID  |              | PERF_TC_05       |                             |
| Test Case Description |                                   | Validate backend stability and response time when multiple users perform predictions simultaneously. |                                |               |              |                  |                             |
| Test Priority         |                                   | Critical                                                                                             |                                |               |              |                  |                             |
| Pre-Requisite         |                                   | Flask API deployed; JMeter configured for concurrent testing.                                        |                                |               |              |                  |                             |
| Post-Requisite        |                                   | System stable, no timeout or crash.                                                                  |                                |               |              |                  |                             |
| Test Execution Steps  |                                   |                                                                                                      |                                |               |              |                  |                             |
| S.No                  | Action                            | Inputs                                                                                               | Expected Output                | Actual Output | Test Browser | Test Re-<br>sult | Test Comments               |
| 1                     | Simulate 10 concurrent uploads    | Valid CSVs                                                                                           | Avg < 5 s                      | 4.2 s         | Edge         | Pass             | System handled well.        |
| 2                     | Increase to 30 concurrent uploads | Valid CSVs                                                                                           | Some delay but no failure.     | Avg 6.8 s     | Edge         | Pass             | Acceptable slowdown.        |
| 3                     | Push to 50 concurrent uploads     | Stress level                                                                                         | Max response < 10 s; no crash. | 9.4 s         | Edge         | Pass             | Flask maintained stability. |

Table 6.22: Test Case 6: Database Read/Write Latency (Firebase)

| Test Case 6           |                         |                                                                                  |                 |               |              |             |                             |
|-----------------------|-------------------------|----------------------------------------------------------------------------------|-----------------|---------------|--------------|-------------|-----------------------------|
| Test Scenario ID      |                         | PERF_TS_06                                                                       |                 | Test Case ID  |              | PERF_TC_06  |                             |
| Test Case Description |                         | Verify the average time Firebase takes to store and retrieve prediction records. |                 |               |              |             |                             |
| Test Priority         |                         | Medium                                                                           |                 |               |              |             |                             |
| Pre-Requisite         |                         | Active Firebase DB and connectivity.                                             |                 |               |              |             |                             |
| Post-Requisite        |                         | Logs show acceptable latency.                                                    |                 |               |              |             |                             |
| Test Execution Steps  |                         |                                                                                  |                 |               |              |             |                             |
| S.No                  | Action                  | Inputs                                                                           | Expected Output | Actual Output | Test Browser | Test Result | Test Comments               |
| 1                     | Store Prediction Result | Ad “DB_Latency_Test”                                                             | Write < 1 s     | 0.8 s         | Edge         | Pass        | Write latency low.          |
| 2                     | Fetch Same Entry        | Ad “DB_Latency_Test”                                                             | Read < 1 s      | 0.7 s         | Edge         | Pass        | Read speed optimal.         |
| 3                     | Simulate 50 reads/sec   | Concurrent access                                                                | Avg < 2 s       | 1.6 s         | Edge         | Pass        | Firebase scaling efficient. |

Table 6.23: Test Case 7: Stress Test – Model Response Degradation

| Test Case 7           |                                 |                                                                                                  |                                         |               |              |             |                                        |
|-----------------------|---------------------------------|--------------------------------------------------------------------------------------------------|-----------------------------------------|---------------|--------------|-------------|----------------------------------------|
| Test Scenario ID      |                                 | PERF_TS_07                                                                                       |                                         | Test Case ID  |              | PERF_TC_07  |                                        |
| Test Case Description |                                 | Check model prediction time when under heavy load or when large files are uploaded continuously. |                                         |               |              |             |                                        |
| Test Priority         |                                 | High                                                                                             |                                         |               |              |             |                                        |
| Pre-Requisite         |                                 | Flask backend connected to model file (joblib).                                                  |                                         |               |              |             |                                        |
| Post-Requisite        |                                 | Backend should remain responsive.                                                                |                                         |               |              |             |                                        |
| Test Execution Steps  |                                 |                                                                                                  |                                         |               |              |             |                                        |
| S.No                  | Action                          | Inputs                                                                                           | Expected Output                         | Actual Output | Test Browser | Test Result | Test Comments                          |
| 1                     | Upload 5 large CSVs in sequence | Each > 400 rows                                                                                  | Avg response < 10 s each                | 9.2 s         | Edge         | Pass        | Model handled continuous predictions.  |
| 2                     | Upload during active prediction | Parallel upload                                                                                  | Queue maintained, no crash.             | As expected   | Edge         | Pass        | Backend queued requests correctly.     |
| 3                     | Simulate low RAM condition      | 1 GB free                                                                                        | Flask handles gracefully, logs warning. | As expected   | Edge         | Pass        | System stable under limited resources. |

Table 6.24: Test Case 8: Negative – Network Latency &amp; Timeout Handling

| Test Case 8           |                             |                                                                                                           |                                                 |               |              |             |                                    |
|-----------------------|-----------------------------|-----------------------------------------------------------------------------------------------------------|-------------------------------------------------|---------------|--------------|-------------|------------------------------------|
| Test Scenario ID      |                             | PERF_TS_08                                                                                                |                                                 | Test Case ID  |              | PERF_TC_08  |                                    |
| Test Case Description |                             | Verify application behavior under high network latency or partial disconnection during upload/prediction. |                                                 |               |              |             |                                    |
| Test Priority         |                             | Medium                                                                                                    |                                                 |               |              |             |                                    |
| Pre-Requisite         |                             | Test with throttled network using DevTools (Slow 3G).                                                     |                                                 |               |              |             |                                    |
| Post-Requisite        |                             | System shows timeout or retry message; no crash.                                                          |                                                 |               |              |             |                                    |
| Test Execution Steps  |                             |                                                                                                           |                                                 |               |              |             |                                    |
| S.No                  | Action                      | Inputs                                                                                                    | Expected Output                                 | Actual Output | Test Browser | Test Result | Test Comments                      |
| 1                     | Upload CSV on 3G            | 150 KB CSV                                                                                                | Predicts or shows “Network Timeout” gracefully. | As expected   | Edge         | Pass        | UI displayed proper retry message. |
| 2                     | Disconnect Wi-Fi mid-upload | –                                                                                                         | Error toast: “Connection lost.”                 | As expected   | Edge         | Pass        | No data corruption; handled well.  |
| 3                     | Retry after reconnection    | Same CSV                                                                                                  | Upload resumes and completes.                   | As expected   | Edge         | Pass        | Verified resume/retry logic.       |

### 6.2.2 Security Testing

Table 6.25: Security Test Case 1: Login &amp; Authentication

| Test Case 1            |                                                                                                |                                                    |                                                           |               |              |             |                                             |
|------------------------|------------------------------------------------------------------------------------------------|----------------------------------------------------|-----------------------------------------------------------|---------------|--------------|-------------|---------------------------------------------|
| Test Scenario ID       | SEC_TS_01                                                                                      |                                                    |                                                           | Test Case ID  | SEC_TC_01    |             |                                             |
| Test Case Description  | Verify that only registered users (Admin/Employee) can log in through Firebase authentication. |                                                    |                                                           |               |              |             |                                             |
| Test Priority          | Critical                                                                                       |                                                    |                                                           |               |              |             |                                             |
| Pre-Requisite          | Firebase authentication setup active; valid user credentials exist in Firebase.                |                                                    |                                                           |               |              |             |                                             |
| Post-Requisite         | User successfully logged in or blocked with appropriate error message.                         |                                                    |                                                           |               |              |             |                                             |
| Test Execution Steps   |                                                                                                |                                                    |                                                           |               |              |             |                                             |
| S.No                   | Action                                                                                         | Inputs                                             | Expected Output                                           | Actual Output | Test Browser | Test Result | Test Comments                               |
| 1                      | Enter valid Employee credentials                                                               | Email: employee@mail.com<br>Password: ValidPass123 | Login successful; redirected to Employee Dashboard.       | As expected   | Edge         | Pass        | Firebase authenticated user correctly.      |
| 2                      | Enter valid Admin credentials                                                                  | Email: admin@mail.com<br>Password: AdminPass123    | Login successful; redirected to Admin Dashboard.          | As expected   | Edge         | Pass        | Admin authentication verified.              |
| 3                      | Enter invalid password                                                                         | Email: employee@mail.com<br>Password: WrongPass!   | Error: "Invalid email or password."<br>No access granted. | As expected   | Edge         | Pass        | Authentication rejection working correctly. |
| 4                      | Enter non-registered email                                                                     | Email: fakeuser@ab.com<br>Password: Random123      | Error: "User not found."<br>No access granted.            | As expected   | Edge         | Pass        | Firebase rejects unknown user.              |
| Continued on next page |                                                                                                |                                                    |                                                           |               |              |             |                                             |

**Table 6.25 – continued from previous page**

| S.No                   | Action                                                | Inputs                                | Expected Output                                             | Actual Output | Test Browser | Test Result | Test Comments                                   |
|------------------------|-------------------------------------------------------|---------------------------------------|-------------------------------------------------------------|---------------|--------------|-------------|-------------------------------------------------|
| 5                      | Leave email/password blank                            | Email: (empty);<br>Password: (empty)  | Validation error: “Please enter email and password.”        | As expected   | Edge         | Pass        | Frontend validation functional.                 |
| 6                      | Attempt to sign up (should be disabled)               | Click “Sign Up” or direct /signup URL | “Sign-up” page not available; redirected to login.          | As expected   | Edge         | Pass        | Account creation restricted as per requirement. |
| 7                      | Try accessing dashboard without logging in            | Direct URL: /dashboard                | Redirected to login page; unauthorized access prevented.    | As expected   | Edge         | Pass        | Access control verified.                        |
| 8                      | Login successfully and manually delete Firebase token | –                                     | Session invalidated; redirected to login page.              | As expected   | Edge         | Pass        | Token validation check successful.              |
| 9                      | Stay logged in and idle for 1 hour                    | –                                     | Session expires; user auto-logged out; redirected to login. | As expected   | Edge         | Pass        | Session timeout handled properly.               |
| Continued on next page |                                                       |                                       |                                                             |               |              |             |                                                 |

**Table 6.25 – continued from previous page**

| S.No | Action                             | Inputs | Expected Output                                       | Actual Output | Test Browser | Test Re-sult | Test Comments                          |
|------|------------------------------------|--------|-------------------------------------------------------|---------------|--------------|--------------|----------------------------------------|
| 10   | Refresh browser after token expiry | –      | Page reloads to login screen; user must log in again. | As expected   | Edge         | Pass         | Token expiry & re-login flow verified. |

**Table 6.26: Security Test Case 2: Restricted Pages Access Control**

| Test Case 2            |                                               |                                                                                                                                            |                          |               |              |             |                                      |
|------------------------|-----------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------|--------------------------|---------------|--------------|-------------|--------------------------------------|
| Test Scenario ID       |                                               | SEC_TS_01                                                                                                                                  |                          | Test Case ID  |              | SEC_TC_01   |                                      |
| Test Case Description  |                                               | Verify that only authenticated users (Employee/Admin) can access restricted pages such as Dashboard, Upload, Results, Logs, and Analytics. |                          |               |              |             |                                      |
| Test Priority          |                                               | Critical                                                                                                                                   |                          |               |              |             |                                      |
| Pre-Requisite          |                                               | Firebase authentication active; user accounts exist; Flask backend running.                                                                |                          |               |              |             |                                      |
| Post-Requisite         |                                               | Unauthorized users are redirected to login or shown an access denied message.                                                              |                          |               |              |             |                                      |
| Test Execution Steps   |                                               |                                                                                                                                            |                          |               |              |             |                                      |
| S.No                   | Action                                        | Inputs                                                                                                                                     | Expected Output          | Actual Output | Test Browser | Test Result | Test Comments                        |
| 1                      | Attempt to open /dashboard without logging in | Direct URL in browser                                                                                                                      | Redirects to login page. | As expected   | Edge         | Pass        | Verified access control enforcement. |
| Continued on next page |                                               |                                                                                                                                            |                          |               |              |             |                                      |

**Table 6.26 – continued from previous page**

| <b>S.No</b> | <b>Action</b>                            | <b>Inputs</b>     | <b>Expected Output</b>                       | <b>Actual Output</b> | <b>Test Browser</b> | <b>Test Re-sult</b> | <b>Test Comments</b>                     |
|-------------|------------------------------------------|-------------------|----------------------------------------------|----------------------|---------------------|---------------------|------------------------------------------|
| 2           | Log in as Employee                       | Valid credentials | Dashboard loads successfully.                | As expected          | Edge                | Pass                | Authenticated user authorized correctly. |
| 3           | Log in as Admin                          | Valid credentials | Admin dashboard loads with analytics & logs. | As expected          | Edge                | Pass                | Admin access confirmed.                  |
| 4           | Try to access admin-only URL as Employee | /admin/logs       | Access denied or redirected.                 | As expected          | Edge                | Pass                | Verified role-based access control.      |
| 5           | Try accessing protected URL after logout | /dashboard        | Redirects to login page.                     | As expected          | Edge                | Pass                | Session invalidated correctly.           |

## Chapter 7

# Conclusion

NeuroAd provides a cost-efficient solution for businesses that want to use EEG to assess attention and engagement with advertisements using the Emotiv EPOC headset and the Random Forest classification tool. Our tool has an outstanding degree of accuracy compared to standard survey methods, as we can measure subconscious responses, thereby bypassing many of the biases typically encountered with verbalized feedback.

The NeuroAd platform uses a web-based application that allows users to securely upload CSV-formatted data and access real-time prediction results via an interactive visualization tool and a PDF report. Firebase enables users to implement role-based access control and maintains the integrity of all data collected.

Testing has demonstrated that the system meets the specifications for all functional requirements (FR01-FR19) and all non-functional requirements (e.g., processing time of <10 seconds and a response rate of at least 95%). As a result, SMEs can now access neuromarketing tools previously available only through expensive enterprise-grade services such as Nielsen NeuroFocus.

In the future, we will expand our current offering by adding real-time EEG streaming capabilities, integrating additional input types (eye-tracking and galvanic skin response (GSR)), developing a mobile Progressive Web Application (PWA), and enabling multiple prediction classes using demographic data. All of these enhancements will further enhance the system's utility in the fast-changing landscape of marketing.

# Appendix A

## User Manual

### Appendix A: User Manual

#### Overview

NeuroAd is a web-based EEG-driven advertisement engagement prediction platform. This manual provides instructions for users with different roles on how to use the platform effectively.

#### A.1 User Roles and Access

##### A.1.1 General User

- Can access informational pages (*Home, About, Contact, Model Summary*).
- Can submit contact inquiries via the *Contact* page.

##### A.1.2 Ad Testing Personnel

Authenticated users who can:

- Upload EEG data.
- Run engagement predictions.
- View detailed results.
- Download reports.

##### A.1.3 Admin

Privileged users with system management capabilities including:

- User administration (add, edit, delete accounts).
- Advertisement record editing.
- Access to logs and analytics dashboards.

## **A.2 Getting Started**

### **A.2.1 Access and Login**

- Visit the NeuroAd website using any major browser on desktop or mobile.
- Ad Testing Personnel and Admin must log in with registered credentials using Firebase Authentication.
- General Users can explore public pages without logging in.

## **A.3 Key Features and Usage**

### **A.3.1 Uploading EEG Data (Ad Testing Personnel)**

1. Navigate to the *Upload* page after login.
2. Select a valid EEG dataset file in CSV format (maximum size 50 MB).
3. Enter relevant advertisement metadata (name, duration).
4. Initiate engagement prediction by submitting the data.
5. Wait for processing (target time within 10 seconds).
6. View classification results on the *Results* page.

### **A.3.2 Viewing Results**

- Results include:
  - Engagement label (Engaging or Non-Engaging).
  - Confidence percentage.
  - Cognitive metrics (attention, memory, emotion).
- Interactive visual dashboards display EEG band activity and derived insights.
- Users can download results in PDF and CSV formats for reporting.

### **A.3.3 Managing Users and Ads (Admin Only)**

- Add, edit, or delete user accounts via the *User Management* interface.
- Edit advertisement records to maintain accuracy.
- Filter and analyze logs by advertisement name, engagement status, and date.
- Access comprehensive system dashboards and audit trails.

## **A.4 Troubleshooting and Support**

- Validation errors guide users to correct EEG file format problems and incomplete data.
- Login failures provide prompts related to credential or connectivity issues.
- Users can contact support via the *Contact* page for unresolved problems.

This manual ensures smooth onboarding and effective use of the NeuroAd platform for all user categories.

## References

- [1] Sangeetha Balasubramanian, Shruti Shriya Gullapuram, and Abhinav Shukla. Engagement estimation in advertisement videos with eeg. *arXiv preprint arXiv:1812.03364*, 2018. Cited on p. 4.
- [2] Zhen Wei, Chao Wu, Xiaoyi Wang, Akara Supratak, Pan Wang, and Yike Guo. Using support vector machine on eeg for advertisement impact assessment. *Frontiers in neuroscience*, 12:76, 2018. Cited on p. 4.
- [3] Luis M Soria Morillo, Juan A Alvarez-Garcia, Luis Gonzalez-Abril, and Juan A Ortega Ramírez. Discrete classification technique applied to tv advertisements liking recognition system based on low-cost eeg headsets. *Biomedical engineering online*, 15(Suppl 1):75, 2016. Cited on p. 5.
- [4] Roger Dooley. From the looks of it, neurofocus is a success. <https://www.neurosciencemarketing.com/blog/companies/neurofocus>, 2011. Accessed: 2025-11-24. Cited on p. 5.
- [5] Neuro-Insight. Neuro-insight - the global leader in neuro-optimization. <https://www.neuro-insight.com/>, 2024. Accessed: 2025-11-24. Cited on p. 5.