Project Management Plan

For

AdBoard: Simplifying Outdoor Advertising

Version 1.0 draft 1

Prepared by

Usaid ur rehman	**BCSF21M513**
Zain Ali	**BCSF21M533**
M.Muzammil Riaz	**BCSF21M514**
M.Huzaifa Latif	**BCSF21M013**

Faculty of Computing & Information Technology, University of the Punjab, Lahore, Pakistan

13 March,2025

Table of Contents

1. Overview

1.1. Project Purpose, Objectives, and Success Criteria

AdBoard is a mobile and web-based platform designed to simplify the process of discovering and booking outdoor advertising spaces in Pakistan. It enables advertisers to efficiently locate and rent billboards, transit ads, digital screens, and other advertising spaces. The platform streamlines the booking process, provides real-time traffic insights, and offers direct communication between advertisers and ad space owners.

Objectives:

- Develop a user-friendly platform for advertisers to discover and book outdoor advertising spaces.
- Integrate a traffic analytics feature using Google Maps and computer vision techniques.
- Provide ad space owners with tools to list, manage, and promote available advertising locations.
- Ensure secure transactions and compliance with advertising regulations.
- Enable direct communication between advertisers and ad space owners for seamless negotiations.

Success Criteria:

- High adoption rate among advertisers and ad space owners.
- Positive user feedback based on ease of booking and search functionality.
- Accurate traffic insights helping advertisers make data-driven decisions.
- A scalable platform adaptable to different cities and advertising categories.
- Secure user data management and adherence to local advertising policies.

1.2. Project Deliverables

Deliverable	Recipients	Delivery Date	Delivery Method	Comments
Platform Prototype	Project Advisor	2 weeks	Web & Mobile App	Initial testing phase
Ad Space Listing & Management	Ad Space Owners	Web & Mobile App	Dashboard for space owners	
Search & Booking Feature	Advertisers	Web & Mobile App	Filtered searches & secure booking	
Traffic Analytics	Advertisers	Web & Mobile App	Google Maps & AI-based traffic estimation	
Messaging System	Advertisers, Space Owners	Web & Mobile App	Direct communication feature	

1.3. Assumptions, Dependencies, and Constraints

Assumptions:
- Advertisers and ad space owners will actively use the platform for booking and listing.
- Reliable traffic data can be extracted using computer vision and Google Maps APIs.
- The platform will initially focus on major urban areas before expanding further.

Dependencies:
- Google Maps API for location tracking and traffic estimation.
- OpenCV and computer vision techniques for traffic analysis.
- Secure cloud hosting (AWS/Heroku) for platform deployment.
- Compliance with local advertising laws and regulations.

Constraints:
- Ensuring compliance with regional and national advertising policies.
- Maintaining system efficiency with high-volume data processing.
- Initial limitations in covering all advertising locations in Pakistan.

1.4. References

<List all documents and any other materials used as sources of information for this plan. For on-line documents, provide hyperlinks wherever possible.>

1.5. Definitions and Acronyms

- **AI:** Artificial Intelligence
- **PWA:** Progressive Web App
- **AWS:** Amazon Web Service
- **API**: Application Programming Interface

1.6. Evolution of the Plan

TThe Project Management Plan for AdBoard will be regularly updated to reflect changes in project requirements, stakeholder feedback, and emerging technical challenges. Updates will follow a structured approval process.

i. Scheduled Updates

- The plan will be **reviewed and updated** at **major project milestones**, including:
  - **Requirement Finalization (Week 4)** – Incorporate stakeholder feedback.
  - **Prototype Completion (Week 12)** – Adjustments based on usability and technical constraints.
  - **Pre-Deployment Testing (Week 20)** – Security, performance, and compliance checks.
  - **Final Project Review (Week 26)** – Documenting the final version for closure.

ii. Unscheduled Updates

- Unscheduled updates may occur due to:
  - Technical difficulties requiring design modifications.
  - Regulatory changes affecting project scope.

- User feedback necessitating feature enhancements.
- Identified risks impacting project delivery timelines.

iii. Dissemination of Updates

- Version-controlled documentation stored on Google Drive and GitHub.
- Email notifications for significant changes.
- Sprint meetings for internal team discussions.
- Formal review meetings for approval of major revisions.

iv. Configuration Control & Versioning

Baseline Version: The initial version stored in Google Drive and GitHub with restricted access.

- **Versioning**:
  - Each revision will have a unique version number (e.g., v1.0, v1.1, v2.0).
  - A changelog will document modifications and approval details.
- **Approval Process**:
  - Minor updates will be reviewed by the Project Manager.
  - Major changes impacting scope, budget, or timeline require Change Control Board (CCB) approval.

v. Review Mechanisms & Continuous Improvement

- Bi-weekly reviews during sprint meetings to assess necessary updates.
- Lessons learned will be incorporated into future revisions.
- A final version will be archived at project closure for future reference.

2. Project Organization

The Project Organization for AdBoard: Simplifying Outdoor Advertising defines the structure, roles, and responsibilities of all individuals and external entities involved in the project. It ensures clear communication, coordination, and accountability across the development lifecycle.

2.1. External Interfaces

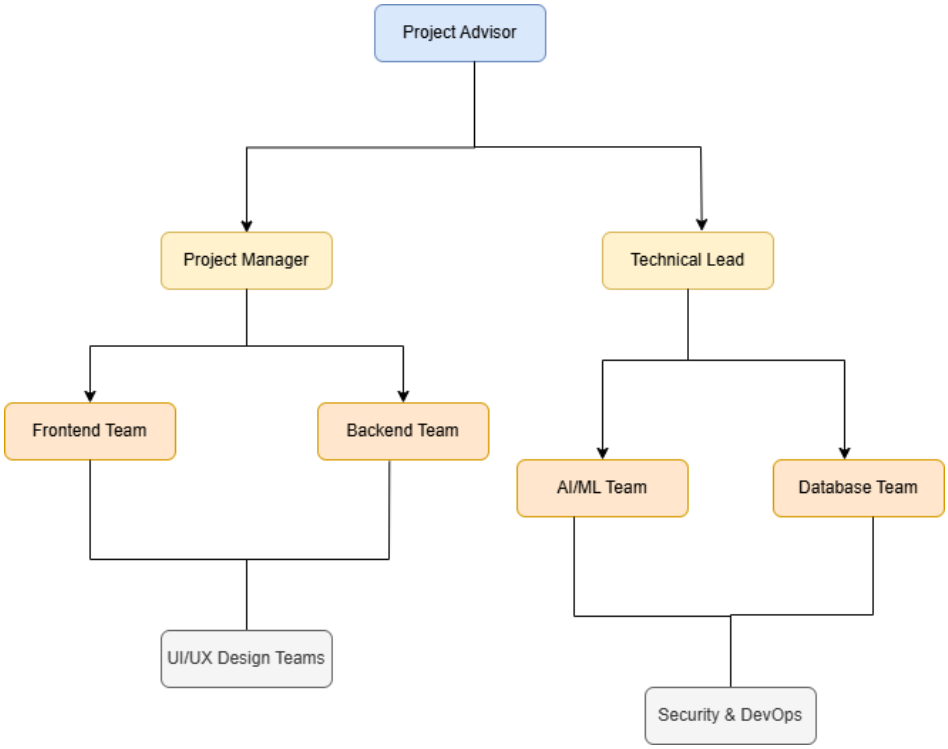
- **Advertisers:** Businesses seeking outdoor advertising spaces.
- **Ad Space Owners:** Billboard owners and property managers listing spaces.
- **Cloud Servicess:** AWS or Heroku for hosting and scalability.

2.2. Internal Structure

AdBoard follows a structured team-based development approach to ensure efficient collaboration and project execution. The internal structure of the project consists of multiple interconnected units, each with specific responsibilities and communication interfaces.

Team Organization Chart

Below is the organizational chart illustrating the hierarchy and communication flow within the project team:



2.3. Roles and Responsibilities

AdBoard is a multi-disciplinary project that involves various stakeholders and team members, each with a specific role and responsibility in ensuring the successful execution of the project. Below is a breakdown of the key team members, their roles, and stakeholder involvement across different project phases.

Role	Responsibilities
Project Advisor (Asst. Prof. Abdul Mateen)	Provides guidance, ensures compliance with academic and ethical standards, and offers expert advice.
Project Manager	Oversees project execution, ensures timely deliverables, and manages resource allocation and risk mitigation.
Technical Lead	Leads system design, ensures proper implementation of platform functionalities, and oversees integration.
Architect	Designs the system architecture, including data flow and component interactions, ensuring scalability and efficiency.
System Engineer	Manages system-level integration, ensuring that backend and frontend modules communicate effectively.
Requirements Analyst	Collects, documents, and validates project requirements, ensuring alignment with user needs and stakeholder expectations.
Software Engineer (Frontend)	Develops the user interface using Flutter, ensuring a responsive and intuitive experience.
Software Engineer (Backend)	Develops APIs, database management (PostgreSQL), and server-side logic using Django.
AI/ML Engineer	Implements computer vision techniques for traffic analytics and data insights.
Test Engineer	Conducts unit testing, system testing, and user acceptance testing to ensure software reliability.
Configuration Management Manager	Manages code versioning (GitHub), software releases, and documentation updates.
Quality Assurance Engineer	Ensures compliance with security and performance standards, including local advertising regulations.
Technical Support Engineer	Provides technical assistance and troubleshooting for users.
Project Advisor (Asst. Prof. Abdul Mateen)	Provides guidance, ensures compliance with academic and ethical standards, and offers expert advice.

3. Managerial Process Plans

The Managerial Process Plans for AdBoard outline the key project management strategies, responsibilities, and activities required to ensure efficient planning, execution, monitoring, and delivery of the project. These processes cover project estimation, staffing, resource acquisition, risk management, communication, tracking, and reporting to ensure smooth coordination among stakeholders and development teams.

3.1. Start-Up Plans

The Start-Up Plans for AdBoard establish a solid foundation for the project by defining the initial activities, resource setup, stakeholder alignment, and key approvals necessary for a smooth launch. These plans ensure that all essential components, including team organization, technical environment setup, and project governance, are in place before development begins.

- **Estimation Plan:** Agile-based task estimations using Jira.
- **Staffing Plan:** Internal team with potential for external consultants.
- **Training Plan:** Workshops on AI-driven traffic estimation and secure API development

3.1.1 Estimation Plan

The Estimation Plan for AdBoard provides a structured approach to estimating project size, effort, cost, schedule, and resource requirements. The plan ensures that estimates are based on data-driven methodologies, validated periodically, and adjusted as needed throughout the project lifecycle.

To ensure accurate and reliable estimates, the following estimation techniques are used:

Estimation Method	Purpose	Application in Eunoia
Work Breakdown Structure (WBS)	Identifies individual tasks and their dependencies.	Breaking down development tasks (e.g., ad listing, traffic analytics, booking system).
Expert Judgment	Leverages team experience to estimate effort and cost.	Involves backend developers, AI experts, and UX designers.
Analogous Estimation	Uses historical data from similar projects.	Compares with existing advertising management platforms.

Function Point Analysis (FPA)	Estimates project size based on feature complexity.	Calculates effort required for modules like AI-driven traffic insights.
Three-Point Estimation (PERT: Optimistic, Pessimistic, Most Likely)	Ensures risk-adjusted time and cost estimates.	Applied to AI-related tasks like traffic analysis.
COCOMO II Model	Estimates software development cost and schedule.	Uses project complexity to

Timing of Estimates

Estimates are prepared at different stages of the project lifecycle:

Project Phase	Estimation Type	Estimation Timing
Project Initiation	High-level cost, effort, and schedule estimates.	Before proposal submission.
Requirement Analysis	Feature-specific effort estimation.	Before finalizing the SRS.
Development Phase	Task-level estimation (weekly sprints).	Before each sprint.
Testing & Deployment	Performance testing resource estimation.	Before integration testing.
Maintenance & Scaling	Infrastructure scaling cost estimates.	Post-launch monitoring phase.

Participants in the Estimation Process

The following project members contribute to the estimation process:

Role	Responsibility in Estimation
Project Manager	Oversees estimation, ensures budget feasibility.
Technical Lead	Estimates backend and AI-related tasks.
Software Engineers	Provide effort estimates for UI/UX and API development.
AI Engineers	Estimate model training and traffic prediction times.
Quality Assurance (QA) Team	Estimates effort for testing and debugging.
DevOps Team	Provides estimates for infrastructure and deployment.

Documentation, Review, and Reporting

- Estimates are documented in a project tracking tool (e.g., Jira, Trello).
- Regular review meetings are held to adjust estimates based on progress.

- Progress reports are shared bi-weekly to track deviations.

Confidence Levels of Estimates

Estimation Type	Confidence Level	Rationale
Project Timeline	85%	Based on prior development experiences.
Traffic Analytics Implementation	70%	Dependent on data quality and API performance.
Backend Development	90%	Uses standard Django/PostgreSQL architecture.
Frontend Development	95%	Flutter framework ensures predictable effort.
Testing & Debugging	75%	Bugs and fixes can vary.

Contingency Buffers & Re-Estimation

To account for **uncertainties**, contingency buffers are added:

- **Development Buffer:** 15% additional time for AI-related unpredictability.
- **Testing Buffer:** 10% additional time for unforeseen debugging issues.
- **Budget Buffer:** 10% reserved for infrastructure scaling (AWS/Heroku).

Re-Estimation Process

- **Weekly sprint reviews** will identify estimation deviations.
- **Mid-project re-estimations** will adjust budget and timeline as needed.
- **Final re-estimation** before deployment ensures adequate resource allocation.

3.1.2 Staffing Plan

The AdBoard platform requires a well-balanced mix of technical, managerial, and domain-specific expertise. This staffing plan outlines the required skill sets, team composition, onboarding strategy, and sourcing methods for personnel.

Project Staffing Requirements

Role	Required Skill	No. of Staff	Duration	Source
Project Manager	Agile project management, budgeting, scheduling, risk management.	1	Full project duration	Internal
Technical Lead	Full-stack development, API integration, system architecture.	1	Full project duration	Internal
	architecture.			

Frontend Developers	Flutter, UI/UX design, responsive development.	2	Full project duration	Internal
Backend Developers	Django, PostgreSQL, API development, authentication security.	2	Full project duration	Internal
AI/ML Engineers	Computer vision, traffic analysis, OpenCV, TensorFlow.	2	Full project duration	Internal
Quality Assurance (QA) Engineer	Automated/manual testing, performance testing.	1	Final testing phase	Internal
DevOps Engineer	AWS/Heroku deployment, CI/CD pipeline, security management.	1	Full project duration	Internal
Security Engineer	JWT authentication, data encryption, compliance with advertising laws.	1	Full project duration	Internal
Advertising Consultant	Outdoor advertising market expertise, location-based marketing strategies.	1	Requirement gathering & validation	External
UI/UX Designer	Figma, accessibility compliance, design optimization.	1	Design & review phase	Internal
Content Writer	Documentation, blogs, ad copy creation.	1	Ongoing updates	Internal
Legal & Compliance Officer	Advertising regulations, data protection laws.	1	Security & privacy audits	External (consultant)
Frontend Developers	Flutter, UI/UX design, responsive development.	2	Full project duration	Internal

Anticipated Staffing Profile & Effort Allocation

The staffing effort is distributed across different project phases, with higher demand in development and testing stages.

Project Phase	Key Roles Involved	Effort Level(%)
Requirement Analysis	Project Manager, Requirements Analyst, Advertising Consultant	20%
System Design	Technical Lead, Architect, UI/UX Designer	30%
Development (AI, Backend, Frontend)	Software Engineers, AI/ML Engineers	80%
Testing & Debugging	QA Engineer, Security Engineer	60%
Deployment & Maintenance	DevOps Engineer, Compliance Officer	50%

Staffing Timeline

Phase	Hiring Timeline	Team Expansion
Initial Setup	First 2 months	Project Manager, Technical Lead, UX Designer
Development Start	Month 3-6	Full-stack Developers, AI Engineers, Database Engineers
Testing Phase	Month 6-8	QA Engineer, Security Engineer
Deployment & Scaling	Month 9+	DevOps Engineer, Compliance Officer, Content Writer

3.1.3 Staff Training Plan

To ensure AdBoard is developed efficiently, all team members will undergo role-specific training to enhance their technical, compliance, and domain knowledge. The Project Manager is responsible for identifying training needs and coordinating training sessions with internal and external experts.

Training Requirements by Role

Team Role	Training Focus	No.of People	Training Method	Trainer/ Source
Frontend Developers	Advanced Flutter, UI/UX optimization	2	Online courses, hands-on workshops	Udemy, Internal Senior Devs

Backend Developers	Django API development, PostgreSQL optimization, security best practices	2	Internal code review sessions, workshops	Internal Django Experts
AI/ML Engineers	Computer vision, traffic analysis, OpenCV	2	AI research papers, hands-on projects	Kaggle, DeepLearning.AI, Internal AI Experts
DevOps Engineer	AWS/Heroku deployment, CI/CD pipeline setup, containerization (Docker)	1	AWS online training, hands-on DevOps projects	AWS Training, Internal DevOps Lead
QA Engineer	Automated testing (Selenium, PyTest), performance testing	1	Testing tools tutorial, hands-on testing sessions	Coursera, Internal QA Lead
Legal & Compliance Officer	Advertising laws, data privacy regulations	1	Compliance workshops, legal documentation reviews	External Legal Advisor

Training Timeline

Training Phase	Roles Trained	Duration
Month 1	Frontend, Backend, AI Engineers	2 weeks
Month 2	Security, DevOps, QA Engineers	3 weeks
Ongoing	Continuous skill enhancement for all roles	As required

Training Methods

1. **Online Courses & Certifications**

- AI Engineers → **DeepLearning.AI, Hugging Face NLP**
- DevOps Engineer → **AWS Training, Heroku Deployment**
- Security Engineer → **Data Protection Compliance & Secure API Development**

2. **Hands-on Workshops & Code Reviews**
   - **Internal training** on Django security, Flutter optimizations, and AI model tuning.
   - **Practical assignments** for testing authentication, ad booking management, and deployment.

3. **Expert-Led Training Sessions**
   - Legal and compliance training by external advertising and legal professionals.
   - Advertisers and ad space owners trained on campaign management best practices.

Project Manager's Role in Training

The **Project Manager** (Muzammil Riaz) is responsible for:
Identifying required training for each role.
Arranging workshops with internal and external trainers.
Monitoring progress through **bi-weekly team check-ins**.
Ensuring all team members complete necessary training before project milestones.

3.1.4 Resource Acquisition Plan

The AdBoard platform requires various development, testing, and product resources to ensure smooth execution and deployment. This section outlines the required software, hardware, cloud services, and infrastructure, along with their acquisition timelines and potential constraints.

Resource Acquisition Process

The resource acquisition process follows these steps:

1. **Requirement Identification** – The Project Manager collaborates with the Technical Lead to determine development, testing, and deployment needs.
2. **Procurement & Approval** – Required tools and services are requested and approved based on availability and budget constraints.
3. **Deployment & Integration** – Resources are integrated into the development pipeline, ensuring smooth workflows.
4. **Monitoring & Scaling** – Continuous evaluation is conducted to ensure resources meet performance needs, with scalability plans in place.

Development Resources

Resources	**Purpose**	**Acquisition Timeline**	**Source**
Laptops (i7/i9, 16GB+ RAM, 512GB SSD)	Development & AI model training.	Month 1	Internal

Operating System (Windows, Linux - Ubuntu 22.04 LTS)	Development environments.	Month 1	Open-source
Development Frameworks (Flutter, Django, PostgreSQL)	Frontend, backend, and database development.	Month 1	Open-source
Cloud Hosting (AWS/Heroku)	Deployment and scalability.	Month 4	AWS Student Credits / Heroku Free Tier
Traffic Analytics Tools (Google Maps API, OpenCV)	AI-based traffic estimation.	Month 4	Free-tier APIs
Version Control (GitHub, GitHub Actions for CI/CD)	Code repository and automated deployment.	Month 1	Free-tier for students
Virtualization (Docker, Kubernetes - Optional for scaling)	Containerization and cloud deployment.	Month 5	Open-source
Network Connectivity	Stable internet (100 Mbps+) for real-time development.	Month 1	Internal

Test Resources

Resource	**Purpose**	**Acquisition Timeline**	**Source**
Test Devices	Mobile and web responsiveness testing.	Month 6	Internal
Automated Testing Tools (Selenium, PyTest, Postman, JMeter)	Functional and API testing.	Month 6	Open-source
Security Testing Tools (OWASP ZAP, Burp Suite, Metasploit)	Penetration testing for security compliance.	Month 7	Open-source
Cloud Logs & Monitoring (AWS CloudWatch, Heroku Logs)	Application error and performance tracking.	Month 7	AWS Free Tier

Product Resources

Resource	Purpose	Acquisition Timeline	Source
Server Storage (100GB+ Cloud Storage - AWS S3/Google Drive)	Storing advertisement data, user logs, and multimedia content.	Month 8	Free-tier AWS/Google Drive
Memory (RAM & CPU Usage Scaling in AWS/Heroku)	Ensuring AI model efficiency.	Month 8	Cloud-based
Load Balancer & Caching (Redis, Cloudflare CDN)	Performance enhancement for concurrent users.	Month 8	Open-source / Free-tier

Constraints & Risks

- Budget Limitations – Preference for free-tier/open-source solutions wherever possible.
- Hardware Constraints – Limited access to high-performance GPUs for AI model training. Workaround: Utilize Google Colab Pro/AWS Free-tier.
- Testing Devices Availability – University-provided test devices are shared across multiple teams. Workaround: Personal devices for parallel testing.
- Network Latency Issues – High internet speed required for real-time AI inference. Workaround: Utilize cloud-based inference APIs when needed.

3.1.5 Project Commitments

The AdBoard platform involves multiple commitments to external stakeholders, internal teams, and regulatory bodies. These commitments ensure transparency, accountability, and progress tracking throughout the project lifecycle.

Commitment	Made By	Made To	Due Date	Comments
Functional Prototype Development	Development Team	Project Manager	Month 4	Deliver a working prototype with core features (ad listing, booking, traffic analytics).
Ad Space Owner & Advertiser Validation	Project Manager	Outdoor Advertising Industry Experts	Month 5	Ensure the platform aligns with industry standards and advertising requirements.
Compliance Review	Security & Legal Team	Regulatory Authorities	Month 6	Ensure full compliance with advertising regulations and data privacy laws.
Performance & Security Testing	QA & Security Team	Technical Lead	Month 7	Conduct rigorous security and performance testing.
Final Deployment & Public Launch	Entire Project Team	End Users	Month 8	Deploy the platform with complete listing, booking, and analytics support.
User Training & Documentation	Content Team	End Users & Ad Space Owners	Month 9	Provide training guides and onboarding materials.
Functional Prototype Development	Development Team	Project Manager	Month 4	Deliver a working prototype with core features (ad listing, booking, traffic analytics).
Ad Space Owner & Advertiser Validation	Project Manager	Outdoor Advertising Industry Experts	Month 5	Ensure the platform aligns with industry standards and advertising requirements.
Compliance Review	Security & Legal Team	Regulatory Authorities	Month 6	Ensure full compliance with advertising regulations and data privacy laws.
Performance & Security Testing	QA & Security Team	Technical Lead	Month 7	Conduct rigorous security and performance testing.
Final Deployment & Public Launch	Entire Project Team	End Users	Month 8	Deploy the platform with complete listing, booking, and analytics support.

User Training & Documentation	Content Team	End Users & Ad Space Owners	Month 9	Provide training guides and onboarding materials.
Functional Prototype Development	Development Team	Project Manager	Month 4	Deliver a working prototype with core features (ad listing, booking, traffic analytics).
Ad Space Owner & Advertiser Validation	Project Manager	Outdoor Advertising Industry Experts	Month 5	Ensure the platform aligns with industry standards and advertising requirements.
Compliance Review	Security & Legal Team	Regulatory Authorities	Month 6	Ensure full compliance with advertising regulations and data privacy laws.
Performance & Security Testing	QA & Security Team	Technical Lead	Month 7	Conduct rigorous security and performance testing.

Commitment Tracking & Communication

- **Project Manager will track commitments** using **Jira/Trello** and hold **bi-weekly review meetings** to assess progress.
- Any **changes in commitments** due to unforeseen issues will be:
  - **Documented in project reports.**
  - **Communicated to stakeholders** via email, Slack, and formal meetings.
- **Escalation Process**: If a commitment is at risk, the **Project Manager** will escalate the issue to senior advisors for resolution.

This Project Commitment Plan ensures accountability and smooth collaboration, keeping AdBoard on track for a successful launch.

3.2. Work Plan

The AdBoard platform follows a structured work breakdown approach to ensure systematic execution of development tasks, resource allocation, and timely milestone tracking.

Work Breakdown Structure (WBS)

Phase	Work Activity	Deliverables	Assigned Team	Duration	Dependencies
Phase 1: Project Initiation	Requirement Gathering	System Requirements Specification (SRS)	Project Manager, AI Engineers	2 weeks	None
	System Architecture Design	System Design Document (SDS)	Technical Lead, Backend Engineers	2 weeks	SRS
Phase 2: Frontend & Backend Development	UI/UX Design	Wireframes, Figma Prototypes	UI/UX Designer, Frontend Devs	3 weeks	SDS
	Frontend Development	Flutter UI Components	Frontend Devs	4 weeks	UI/UX Design
	Backend API Development	Django APIs, User Authentication	Backend Devs	4 weeks	System Design
	Database Setup	PostgreSQL Schema	Database Engineer	3 weeks	Backend Development
Phase 3: AI & Analytics Integration	Traffic Analytics Model	AI-based Traffic Estimation Model	AI Engineers	5 weeks	Database Setup
	Advertisement Performance Analysis	Analytics Dashboard	AI Engineers	4 weeks	Traffic Model
Phase 4: Testing & Compliance	Unit & Integration Testing	Bug Reports, Fixed APIs	QA Engineers	4 weeks	Backend & Frontend Complete

Security & Compliance Testing	Advertising Compliance Report	Security Team	3 weeks	Backend Complete	
Phase 5: Deployment & Scaling	Cloud Deployment	Hosted App on AWS/Heroku	DevOps Engineer	3 weeks	Testing Passed
	Performance Optimization	Load Testing Results	DevOps, AI Engineers	3 weeks	Deployed System
Phase 6: Documentation & Final Review	User Guide & Technical Documentation	Training Manuals	Content Team	3 weeks	Completed System
	Final Review & Public Release	Fully Functional Platform	Entire Team	2 weeks	Documentation Complete

Major Milestones & Progress Tracking

Milestone	**Expected Completion**	**Tracking Mechanism**
Project Planning Complete	Week 4	Approval of SRS & SDS
Frontend & Backend Integration Complete	Week 12	Functional API Testing
AI-Powered Analytics Integration	Week 16	Model Deployment & Testing
Compliance & Security Testing Passed	Week 20	Advertising Compliance Certification
System Deployment on Cloud	Week 24	Live Testing Reports
Final Documentation & Public Launch	Week 26	User Training & Feedback

Project Tracking & Risk Management

- **J**ira/Trello is used for task tracking.
- Weekly sprint reviews ensure deviations are identified early.
- Risk management actions are taken when development lags behind schedule.

This Work Plan ensures smooth execution, timely delivery, and compliance adherence for AdBoard.

3.3. Control Plan

<This section describes how the AdBoard project will control and report on the project status and activities. Specify the frequency at which the various project status indicators are to be monitored and specific events that could trigger a status evaluation.>

3.3.1 Data Control Plan

AdBoard requires a structured Data Control Plan to ensure the secure management, retrieval, storage, and distribution of various project-related data, including deliverables, reports, analytics data, and user-generated content. This plan defines the types of data, privacy and security requirements, and data management mechanisms to protect sensitive information and ensure compliance with local advertising laws.

1. Types of Data to Be Managed

Data Type	Description	Storage & Format	Access Control
Project Documentation	SRS, SDS, design diagrams, testing reports	Google Drive, GitHub (Markdown/PDF)	Project Team, Stakeholders
Codebase & Version Control	Frontend (Flutter), Backend (Django), AI Models (OpenCV, TensorFlow)	GitHub (Private Repository)	Developers, DevOps
User Data	Ad listings, user preferences, booking history	PostgreSQL (Encrypted)	Restricted to Backend Team
Traffic Analytics Data	Data used for advertisement performance tracking	PostgreSQL (Encrypted) / Cloud Storage	AI Engineers (Anonymized Access)
AI Training Data	Dataset for traffic estimation and visibility insights	Local Storage, Cloud (AWS S3)	AI Engineers (Bias Control)
Compliance Reports	Advertising laws and security assessments	Encrypted Drive (PDF)	Security & Compliance Team

2. Content & Format Description

- **Project Reports** → Markdown, PDF format (stored in GitHub/Google Drive).
- **Source Code** → Managed in GitHub (organized into backend, frontend, AI models).
- **User Data & Logs** → Structured in PostgreSQL (JSON-based storage for chatbot logs).

- **Compliance Documentation** → PDF-based, with restricted access.

2. Privacy Requirements

To ensure **user privacy and ethical AI usage**, Eunoia follows:

- **GDPR & HIPAA Compliance** – Users have the right to request deletion of their data.
- **Minimal Data Retention** – Chatbot logs are anonymized and retained only for AI improvements.
- **Role-Based Access Control (RBAC)** – Data is only accessible to authorized personnel.

3. Security Requirements & Procedures

To prevent **data breaches and unauthorized access**, the following security measures are in place:

Encryption

- **AES-256 encryption** for storing sensitive user data.
- **TLS/SSL** for secure communication between frontend and backend.

Authentication & Authorization

- **JWT authentication** for secure user login.
- **Multi-Factor Authentication (MFA)** for admin and therapist accounts.

Data Integrity & Backup

- **Daily database backups** to an **encrypted AWS S3 bucket**.
- **Disaster recovery plan** ensures minimal downtime in case of system failures.

4. Mechanisms for Data Collection, Retrieval, Distribution & Archiving

Process	**Mechanism Used**
Data Collection	RESTful APIs collect user inputs (mood tracker, journal, chatbot conversations).
Data Retrieval	Flask-based API endpoints for fetching user history (only upon authentication).
Data Distribution	Secure API-based retrieval for AI training (anonymized data only).
Data Archiving	Older chatbot logs automatically **deleted after 6 months** unless required for analytics.

1.1.2 Requirements Control Plan

The **Requirements Control Plan** for **Eunoia: AI-Powered Mental Health Support Platform** ensures that changes to product requirements are **properly evaluated, documented, approved, and implemented** while maintaining the project's **scope,**

schedule, budget, and quality standards.

1. Mechanisms for Measuring, Reporting, and Controlling Requirement Changes

Step	**Description**
1. Change Request Submission	Any requirement change must be formally requested through a **Change Request Form (CRF)** submitted by stakeholders, developers, or users.
2. Initial Review & Categorization	The **Project Manager & Technical Lead** review the request and classify it as **minor (UI update), moderate (new feature), or major (core functionality change)**.
3. Impact Assessment	The change is analyzed in terms of its **impact on scope, budget, timeline, and resources**.
4. Approval/Rejection	The **Change Control Board (CCB)**, consisting of **the Project Manager, Technical Lead, and Key Stakeholders**, makes the final decision.
5. Documentation & Update	Approved changes are documented in **SRS (Software Requirements Specification)**, and the **Work Plan, Budget, and Schedule** are updated accordingly.
6. Implementation & Testing	Developers implement the change in **sprints**, followed by rigorous **testing (unit, integration, and user acceptance)**.
7. Stakeholder Communication	Affected stakeholders (users, therapists, compliance teams) are notified via **email, Slack, or documentation updates**.

2. Impact Assessment Criteria

Assessment Factor	Impact Evaluation
Scope	Does the change significantly modify the project's objectives or add new functionalities?
Quality	Will the change improve user experience, performance, or compliance with GDPR/HIPAA?
Schedule	How many additional development & testing weeks are required? Does it affect milestones?
Budget	Will extra costs be incurred (e.g., additional cloud storage, external

	expert consultation)?
Resources	• Will extra team members be needed, or will workload increase significantly?
Risk Factors	Could the change introduce security vulnerabilities, performance issues, or ethical concerns?

3. Change Control Process

- **Minor Changes** (e.g., UI tweaks, text updates) → Approved by **Project Manager**, implemented within **1 sprint**.
- **Moderate Changes** (e.g., chatbot improvement, therapist feature enhancement) → Reviewed by **CCB**, scheduled within **next 2 sprints**.
- **Major Changes** (e.g., new AI module, multi-language support) → Requires a **formal review, budget revision, and impact analysis**, may extend project timeline.

4. Update Process for Project Plans & Work Breakdown Structure (WBS)

Whenever a **requirement change affects project scope**, the following updates occur:

- **Project Management Plan** → Updated to reflect changes in **timeline, budget, and resources**.
- **Work Breakdown Structure (WBS)** → Adjusted to include **new tasks & dependencies**.
- **Software Requirement Specification (SRS)** → Updated with new **feature descriptions**.
- **Testing Plan** → Modified to include **new test cases** for the updated requirements.

5. Change Tracking & Reporting

- **Trello/Jira** → Used for tracking requirement changes and assigned tasks.
- **Weekly Review Meetings** → Status updates for approved requirement changes.
- **Stakeholder Reports** → Monthly updates for sponsors, legal advisors, and key users.

1.1.3 Schedule Control Plan

The **Schedule Control Plan** for **Eunoia: AI-Powered Mental Health Support Platform** ensures that project progress is **monitored, tracked, and adjusted** to meet planned milestones. It incorporates **schedule tracking tools, performance evaluation methods, contingency buffers, and corrective action plans** to handle deviations from the planned timeline.

1. Schedule Monitoring & Progress Tracking

- **Project Tracking Tools**: Jira/Trello for tracking tasks and sprints.
- **Gantt Charts**: Used in Microsoft Project or Google Sheets for milestone visualization.
- **Daily Standups**: Agile-based team check-ins to track progress and identify blockers.
- **Bi-weekly Sprint Reviews**: Evaluate task completion and adjust timelines if necessary.
- **Monthly Stakeholder Reports**: High-level progress updates for investors, therapists, and compliance teams.

2. Milestone-Based Performance Measurement

Milestone	Planned Completion	Measurement Criteria	Status Tracking Tool
Project Planning & Requirement Analysis	Week 4	Approved SRS & SDS	Trello/Jira
Frontend & Backend Integration	Week 12	Working API & UI connection	Trello/Jira
AI Model Integration	Week 16	Deployed NLP-based chatbot	Trello/GitHub
Compliance & Security Testing	Week 20	GDPR/HIPAA certification reports	Compliance Checklist
Deployment on Cloud	Week 24	Fully functional AWS/Heroku deployment	AWS Logs
Final Release & Documentation	Week 26	Public launch with training guides	GitHub/Google Drive

3. Methods for Comparing Planned vs. Actual Performance

- **Burn-down Charts**: Show remaining work vs. timeline.
- **Sprint Velocity Tracking**: Measures work completed per sprint to detect delays.
- **Variance Analysis**: Compares actual vs. planned completion dates for major tasks.
- **Earned Value Management (EVM)**: Evaluates cost and schedule performance.

4. Contingency Buffers & Corrective Actions

- **Development Buffer**: 15% additional time allocated for unforeseen challenges in AI model training and integration.
- **Testing Buffer**: 10% additional time added to account for security and compliance fixes.
- **Performance Buffer**: Additional cloud resources reserved for scaling and optimization.
- **Corrective Actions for Delays**:
  - Reassign tasks to available team members.
  - Increase parallel development efforts where feasible.
  - Implement phased releases for features that can be deployed incrementally.

5. Schedule Modification & Approval Process

- **Weekly Sprint Reviews**: Adjust task priorities based on progress.
- **Change Request Process**: Any significant schedule change requires approval from the Project Manager and key stakeholders.
- **Stakeholder Communication**: Updates on schedule changes are shared through email,

Slack, and formal meetings.

1.1.4 Budget Control Plan

The **Budget Control Plan** for **Eunoia: AI-Powered Mental Health Support Platform** ensures that project expenses are **monitored, tracked, and controlled** to stay within the allocated budget. It includes **cost measurement, budget reporting, variance analysis, and corrective actions** to address cost overruns.

1. Budget Monitoring & Tracking Mechanisms

- **Planned vs. Actual Cost Comparison**: Regularly compare estimated vs. actual costs to identify deviations.
- **Milestone-Based Budget Tracking**: Expenses are reviewed at key project milestones.
- **Expense Categorization**: Costs are tracked under development, testing, cloud infrastructure, compliance, and external consultant fees.
- **Approval Workflow**: Any additional spending requires approval from the Project Manager and stakeholders.

2. Cost Measurement & Reporting

- **Project Manager Responsibility**: The Project Manager is responsible for budget tracking and ensuring financial control.
- **Bi-Weekly Cost Reports**: Financial reports are generated every two weeks to assess spending.
- **Stakeholder Budget Reviews**: Monthly budget reports are reviewed with key stakeholders.
- **Tracking Tools**: Google Sheets, Microsoft Excel, or budgeting software like QuickBooks are used for financial tracking.

3. Budget Allocation & Expense Tracking

Category	Estimated Budget	Tracking Method	Responsible Party
Development (Software & AI)	$5,000	Jira/Trello	Project Manager
Cloud Hosting (AWS/Heroku, Databases)	$2,500	AWS Billing Dashboard	DevOps Engineer
Testing & Security Compliance	$1,500	Security Audit Reports	QA & Security Team
External Consultants (Therapist, Religious	$2,000	Contract Management	Project Manager

Scholar, Compliance Officer)			
Training & Resources	$1,000	Training Completion Logs	HR & Project Manager
Miscellaneous (Contingencies, Unexpected Costs)	$1,000	Ad-hoc Approvals	Project Manager

4. Corrective Actions for Budget Deviations

- Cost Overruns (Exceeding 10%)
  - Review and prioritize essential vs. non-essential expenses.
  - Seek additional funding approval if necessary.
  - Optimize cloud resources to reduce operational costs.
- Cost Savings Measures
  - Utilize free-tier cloud services where applicable.
  - Reduce dependency on external consultants by leveraging internal expertise.
  - Optimize AI training processes to reduce computing costs.

5. Budget Modification & Approval Process

- Change Request Process: Any significant budget adjustments require approval from the Project Manager and financial stakeholders.
- Stakeholder Communication: Updates on financial changes are shared through reports, meetings, and email updates.

1.1.5 Communication, Tracking, and Reporting Plan

The **Communication, Tracking, and Reporting Plan** for **Eunoia: AI-Powered Mental Health Support Platform** ensures that project updates, risks, and progress are communicated **effectively and transparently** to all relevant stakeholders. This plan defines the **types of communication, reporting mechanisms, frequency, and responsible parties** for various project-related updates.

1. Communication Objectives

- Ensure timely updates on project **progress, risks, budget, and requirement changes**.
- Maintain clear **information flow** between developers, project managers, external consultants, and stakeholders.
- Facilitate **quick decision-making** to handle blockers and deviations from the schedule.

Type	**Schedule**	**Communication Mechanism**	**Who Initiates**	**Recipient**

of communication				
Status Report	Every Friday	Team meeting (virtual/in-person)	Project Manager	Project Team
Schedule and Effort Tracking Report	Weekly	Email	Project Manager	Program Manager
Project Review	Monthly	Face-to-face meeting	Project Manager	Project Team
Risk Mitigation Status	As mitigation actions are completed	Email	Responsible team member	Project Manager
Requirement Changes	As changes are approved	Email and Change Control Tool	Change Control Board (CCB) Chair	Affected Project Participants
Supplier Management Review	At project life cycle gates	Video conference	Program Manager	Project Manager, Program Manager, Subcontract Manager
Compliance & Security Update	Bi-Monthly	Compliance Reports	Security Officer	Project Manager, Legal Team
User Feedback Review	Post deployment	Online Survey & Reports	Customer Support Team	Project Team, Developers
Final Project Report	End of Project	Documentation (Google Drive/PDF)	Project Manager	Stakeholders, Investors

3. Report Contents & Tracking Metrics

Each report includes key project performance indicators:

- **Status Reports**: Task completion rate, milestone progress, pending issues.
- **Effort Tracking Reports**: Hours logged per task, sprint velocity, workload distribution.
- **Budget Reports**: Planned vs. actual cost, variances, corrective actions.
- **Quality Reports**: Bugs found, testing coverage, security compliance.
- **Risk Reports**: Identified risks, mitigation strategies, impact assessment.
- **User Feedback Reports**: Customer ratings, sentiment analysis, improvement suggestions.

4. Special Considerations

- **Remote Collaboration**: Since some external consultants (therapists, religious scholars, compliance officers) are not onsite, video conferencing (Zoom/Google Meet) is used for requirement validation and compliance discussions.
- **Requirement Change Tracking**: Jira/Trello boards are used for managing requirement modifications and change requests.
- **Crisis Management Communication**: In case of security or deployment failures, immediate notifications are sent via Slack, Email, and SMS to key personnel.

5. Communication Plan Review & Updates

- The **Project Manager** reviews the communication plan every **two months** and makes adjustments based on feedback.
- Any **new reporting needs** are discussed in **bi-weekly sprint meetings** and incorporated as needed.

1.1.6 Metrics Collection Plan

The **Metrics Collection Plan** for **Eunoia: AI-Powered Mental Health Support Platform** ensures that key project performance indicators are **tracked, analyzed, and reported** systematically. This helps in evaluating **progress, quality, efficiency, and risk management** throughout the project lifecycle.

1. Methods & Tools for Metrics Collection

Category	Metrics Collected	Collection Method	Tool Used
Project Progress	Task completion rate, sprint velocity, milestone achievement	Agile tracking	Jira, Trello
Effort Tracking	Hours logged per task, developer workload	Time tracking reports	Clockify, Jira
Schedule Performance	Planned vs. actual completion dates	Gantt chart analysis	Microsoft Project, Google Sheets

Budget Performance	Planned vs. actual cost, variance analysis	Expense tracking	Google Sheets, QuickBooks
Code Quality	Bugs per module, test coverage, security vulnerabilities	Automated testing & code review	SonarQube, Selenium, OWASP ZAP
System Performance	Response time, API latency, uptime percentage	Load testing & monitoring	JMeter, AWS CloudWatch
User Experience	UI feedback, chatbot accuracy, engagement rate	User surveys, analytics	Google Analytics, Hotjar
Risk Management	Identified risks, severity level, mitigation success	Risk log updates	Trello, Jira
Compliance & Security	GDPR/HIPAA adherence, penetration test results	Security audits	Compliance checklists, OWASP ZAP

2. Collection Frequency & Validation

Metric Type	**Collection Frequency**	**Validation Method**
Task Completion & Sprint Velocity	Weekly	Cross-check Jira with team reports
Budget & Expense Reports	Bi-weekly	Financial review by Project Manager
Schedule Tracking (Milestones)	Monthly	Compare actual vs. planned schedule
Code Quality & Security Checks	Every Sprint	Automated testing & peer reviews
System Performance Reports	Bi-weekly	Benchmarking API & database performance
User Experience Feedback	Monthly	Aggregated user ratings & engagement metrics
Risk Management Reports	Bi-weekly	Risk assessment and action tracking
Compliance Audits	Every 3 months	External audit & internal

		validation

3. Metrics Storage & Reporting

- **Storage**
    - Project-related metrics (task completion, bug tracking) are stored in **Jira/Trello**.
    - Code quality & security reports are stored in **GitHub/GitLab repositories**.
    - Budget and financial reports are maintained in **Google Drive/QuickBooks**.
    - Compliance reports and risk logs are archived in a **secure cloud repository (AWS S3, Google Drive - restricted access)**.
- **Reporting Mechanism**
    - **Sprint Reports**: Weekly updates shared with the development team.
    - **Stakeholder Reports**: Monthly progress reports including budget, risk, and compliance status.
    - **Final Project Report**: Comprehensive report at project closure, summarizing metrics analysis and lessons learned.

4. Use of Collected Metrics

- **Decision Making**: Adjust project scope, schedule, or resource allocation based on insights.
- **Quality Improvement**: Identify bottlenecks in software development and testing.
- **Budget Optimization**: Prevent overspending by tracking real-time expenses.
- **Risk Mitigation**: Address potential risks early by monitoring key indicators.
- **User Experience Enhancement**: Improve chatbot interactions and mental health support features using feedback analytics.

5. Continuous Improvement & Plan Updates

- The **Project Manager** and **Technical Lead** review the **metrics collection plan** every **two months** to ensure its relevance.
- Adjustments are made based on project needs, feedback, and stakeholder requirements.

1.2. Risk Management Plan

The **Risk Management Plan** for **Eunoia: AI-Powered Mental Health Support Platform** outlines the process for **identifying, analyzing, prioritizing, and mitigating project risks** to minimize disruptions and ensure successful project completion.

1. Risk Identification Process

Risks are identified through:

- **Brainstorming sessions** with the project team and stakeholders.
- **Historical data analysis** from similar projects.
- **SWOT analysis** to identify weaknesses and external threats.
- **Regular sprint reviews** to monitor ongoing risks.

- **User feedback** during beta testing for potential usability risks.

2. Risk Categories & Examples

Category	**Potential Risks**
Technical Risks	AI chatbot inaccuracies, system performance issues, data security vulnerabilities.
Project Management Risks	Delayed milestones, scope creep, resource unavailability.
Compliance & Legal Risks	GDPR/HIPAA violations, ethical concerns in AI-generated therapy.
User Adoption Risks	Low user engagement, accessibility challenges, cultural barriers.
Financial Risks	Budget overruns, unplanned infrastructure costs, external consultant expenses.
Operational Risks	Server downtime, cloud service failures, lack of real-time crisis response support.

3. Risk Analysis & Prioritization

Each risk is assessed based on likelihood (Low, Medium, High) and impact (Low, Medium, High) to determine priority levels.

Risk	**Likelihood**	**Impact**	**Priority**
AI-generated therapy inaccuracies	High	High	Critical
Security breaches (data leaks)	Medium	High	High
Budget overruns	Medium	Medium	Moderate
Low user adoption	High	Medium	High
Compliance violations	Low	High	High
Cloud infrastructure failure	Low	High	High
Delay in AI model training	High	Medium	High

4. Risk Mitigation Strategies

Risk Category	**Mitigation Strategy**
Technical Risks	Regular AI model tuning, performance testing, fallback mechanisms.

Project Delays	Buffer time in schedule, parallel task execution, weekly sprint tracking.
Compliance Issues	Legal review before deployment, automated compliance checks.
Low User Engagement	User testing, accessibility improvements, marketing efforts.
Budget Overruns	Cost monitoring, resource optimization, alternative funding sources.
Infrastructure Risks	Multi-cloud strategy, automatic failover setup, regular backups.

5. Risk Tracking & Response Plan

- Risk Log Maintenance: All identified risks are logged in Trello/Jira for tracking.
- Bi-Weekly Risk Reviews: Risk exposure and impact are reassessed every two weeks.
- Emergency Response Plan: Immediate action for critical risks (e.g., security breaches, infrastructure failures).
- Escalation Process: High-impact risks are escalated to Project Manager & Stakeholders for quick decision-making.

6. Risk Management Responsibilities

Task	**Responsibility Party**	**Target Date**
Identify & document project risks	Project Manager	Ongoing
Conduct risk impact assessment	Risk Management Team	Bi-Weekly
Implement risk mitigation measures	Technical Lead, Security Team	As required
Monitor AI-generated content for accuracy	AI/ML Team	Continuous
Conduct compliance audits	Compliance Officer	Every 3 months
Infrastructure backup testing	DevOps Engineer	Monthly
Review risk exposure & update risk log	Entire Team	Sprint Review

7. Risk Management Effort Estimation

- Approximately 10-15% of project effort is allocated for risk identification, mitigation, and monitoring.
- Budget allocation includes contingency funds for handling unforeseen risks.
- Risk management is embedded into the project schedule, ensuring timely action on identified risks.

8. Ongoing Risk Identification

- Weekly sprint meetings include a risk review segment to detect emerging threats.
- Stakeholder feedback sessions highlight user concerns and unexpected risks.
- AI/ML performance monitoring ensures system reliability and accuracy.

1.3. Issue Resolution Plan

The **Issue Resolution Plan** for **Eunoia: AI-Powered Mental Health Support Platform** ensures that project issues are **properly documented, analyzed, prioritized, assigned, and resolved** in a timely manner. The plan defines the **process for tracking issues, roles responsible for resolution, and escalation mechanisms** when necessary.

1. Issue Documentation & Tracking

- **Issue Logging**: All project-related issues are recorded in a centralized **issue tracking system (Jira/Trello)**.
- **Issue Categories**: Issues are categorized as **technical, requirement-related, resource-based, compliance-related, or financial**.
- **Unique ID Assignment**: Each issue is assigned a unique identifier for tracking purposes.
- **Issue Description Format**:
    - **Title**: Brief summary of the issue.
    - **Description**: Detailed explanation, including impact on the project.
    - **Priority**: Low, Medium, High, or Critical.
    - **Assigned To**: Team member responsible for resolution.
    - **Status**: Open, In Progress, Resolved, Closed.
    - **Resolution Notes**: Steps taken to resolve the issue.

2. Issue Resolution Workflow

1. **Issue Identification**:
    - Team members, stakeholders, or users report issues through Jira/Trello.
    - Automated monitoring systems flag security or performance issues.
2. **Issue Triage & Prioritization**:
    - The **Project Manager** categorizes the issue as **low, medium, high, or critical**.
    - **Critical issues** (e.g., security breaches, major system failures) are immediately escalated.
3. **Assignment & Investigation**:
    - The **Technical Lead** assigns the issue to the responsible team member.
    - **Root cause analysis (RCA)** is performed to understand the problem.
4. **Issue Resolution**:
    - The responsible team member implements a **fix or workaround**.
    - If an issue requires major changes (e.g., requirement modification), approval from the **Change Control Board (CCB)** is required.
5. **Testing & Validation**:
    - The **QA Team** verifies that the issue has been resolved and that no regressions occur.
    - Resolved issues undergo **unit, integration, and user acceptance testing (UAT)**.
6. **Closure & Documentation**:

- Once validated, the issue status is marked as **Closed**.
- The resolution is documented in the **Issue Log** for future reference.

3. Decision-Makers for Issue Resolution

Issue Type	Decision-Maker(s)	Responsibility
Requirement Baseline Issues	Change Control Board (CCB)	Approve/reject requirement changes.
Technical Issues (AI Model, API, UI Bugs)	Technical Lead, Developers	Analyze and implement fixes.
Resource Contention (Team Overload, Staff Shortage)	Project Manager	Reallocate resources or adjust workload.
Priority Conflicts (Conflicting Features, Timeline Constraints)	Project Manager, Product Owner	Define priorities based on business needs.
Compliance Issues (GDPR, HIPAA Violations)	Compliance Officer, Legal Team	Define priorities based on business needs.
Compliance Issues (GDPR, HIPAA Violations)	Project Manager, Finance Team	Approve additional funding or cost-cutting measures.

4. Issue Escalation Process

- **Low & Medium Priority Issues**: Handled within the responsible team, resolved within **2-5 working days**.
- **High Priority Issues**: Discussed in **bi-weekly sprint meetings**, must be resolved within **1 week**.
- **Critical Issues (Security, Compliance, System Failure)**:
  - **Immediate escalation** to the **Project Manager & Stakeholders**.
  - **Response Time: 24-48 hours**.
  - **Emergency fix deployment** if necessary.

5. Issue Resolution Reporting

- **Weekly Status Updates**: Summary of open, in-progress, and resolved issues is shared with the team.
- **Monthly Reports**: Comprehensive issue resolution report presented to stakeholders.
- **Post-Mortem Analysis**: Conducted for critical issues to prevent recurrence.

1.4. Project Close-Out Plan

The **Project Close-Out Plan** for **Eunoia: AI-Powered Mental Health Support Platform**

    - Conduct a **final project review session** with the entire team.
    - Discuss **what worked well, what challenges were faced, and how future projects can improve**.
- **Lessons Learned Report**:
    - Document key insights on feature development, user experience, and stakeholder management.
    - Provide recommendations for future ad-tech and marketplace projects.

2. Final Project Report & Handover

- **Final Report Includes**:
    - Summary of **project objectives and achievements**.
    - **Challenges faced and resolutions implemented**.
    - **Future recommendations** for scaling and improvements.
    - **Final budget and cost analysis**.
- **Handover to Stakeholders**:
    - Provide **ongoing maintenance and support documentation**.
    - Assign a **point of contact for future inquiries or enhancements**.

3. Project Closure Approval

- **Sign-Off from Key Stakeholders**:
    - Obtain approval from **project sponsors, advisors, and compliance teams**.
    - Ensure all deliverables meet the required standards.
- **Formal Project Closure Notification**:
    - Notify all internal and external stakeholders about the project's completion.
    - Provide information on **long-term maintenance and user support contacts**.

1. Technical Process Plans

The Technical Process Plans for AdBoard outline the development methodologies, tools, and best practices used to ensure efficient software development, testing, deployment, and maintenance.

1.1. Process Model

Development Methodology: Agile (Scrum-Based Iterative Model)

The **Agile development model** is used to provide **continuous iterations, rapid prototyping, and user feedback integration**. This ensures flexibility in improving the platform based on real-world use cases.

- **Incremental Approach**: Features are developed and tested in multiple iterations.
- **Short Sprints**: Each sprint lasts **2 weeks** with **a sprint planning, development, review, and**

retrospective phase.

- **Continuous Integration & Deployment (CI/CD)**: Automated builds and testing through GitHub Actions, AWS, and Firebase Hosting.

Key Milestones and Iterations

Iteration	Duration	Tasks Completed	Milestone/Checkpoints
Iteration 1	Weeks 1 - 4	Requirement gathering, system design, database schema, UI/UX wireframes	**System Design Document (SDS) Review**
Iteration 2	Weeks 5 - 8	Frontend & backend core structure, authentication, API development	**Initial Prototype Testing**
Iteration 3	Weeks 9 - 12	Real-time messaging integration, Ad booking system, Firebase database setup	**Messaging & Database Testing**
Iteration 4	Weeks 13 - 16	UI improvements, map-based advertising feature, payment gateway integration	**Mid-Project Review**
Iteration 5	Weeks 17 - 20	Security testing, load testing, advertising analytics dashboard	**Security & Performance Testing**
Iteration 6	Weeks 21 - 24	Deployment on AWS/Firebase, final UI/UX enhancements, user testing	**Beta Testing & Pre-Launch Review**
Iteration 7	Weeks 25 - 26	Final adjustments, user documentation, performance optimizations	**Public Release & Final Review**

Checkpoints for Management Reviews

- **System Design Approval** *(Week 4)*
- **Prototype Testing & API Integration Review** *(Week 8)*
- **Messaging & Booking System Testing** *(Week 12)*
- **Security Audit & Performance Review** *(Week 16)*
- **Compliance & Pre-Deployment Testing** *(Week 20)*
- **Beta Testing & Pre-Launch Finalization** *(Week 24)*
- **Final Project Review & Sign-Off** *(Week 26)*

Additional References

- **Configuration Management Plan**: Ensures proper version control, change tracking, and documentation of all updates.
- **Quality Assurance Plan**: Defines the testing framework, security compliance measures, and performance benchmarks.

1.2. Methods, Tools, and Techniques

The AdBoard platform follows a structured approach to design, development, testing, and deployment, leveraging industry-standard technologies and best practices.

1. Development Environment

Category	Details
Hardware Requirements	Minimum: 16GB RAM, 512GB SSD, Intel i7+ or Apple M1+
Operating System	Windows 11, Ubuntu 22.04 LTS, macOS
Network Requirements	100 Mbps+ stable internet for cloud integration
Development Machines	University-provided workstations, personal laptops
Test Devices	Android (Pixel 6), iOS (iPhone 13) for PWA testing

2. Software Tools

Tool Category	Tool Used	Purpose
Version Control	GitHub	Source code management
Requirement Management	Trello, Jira	Task tracking & Agile planning
Design & Modeling	Figma, draw.io	UI/UX design, system diagrams
Code Editor/IDEs	VS Code, PyCharm	Frontend and backend development
Build Automation	GitHub Actions	CI/CD automation
Database Management	PostgreSQL, pgAdmin	Database storage and management
Containerization	Docker	Deployment consistency
Cloud Hosting	AWS (EC2, S3), Heroku	Backend deployment
Monitoring & Logging	AWS CloudWatch, Sentry	Performance monitoring
Testing & Debugging	Selenium, PyTest, Postman	Automated testing, API validation

3. Development Methodologies

- **Requirements Development Practices**
    - Stakeholder interviews, surveys, and competitive analysis.
    - Documenting user stories and personas for feature validation.
    - Change requests are reviewed through the **Requirements Control Plan**.
- **Design Methodologies & Notations**
    - **Architectural Design**: Layered architecture (Frontend, API, Database, AI).
    - **UML Diagrams**: Use case diagrams, sequence diagrams, class diagrams.

- **Database Schema**: Relational design using **PostgreSQL**.

- **Programming Languages & Standards**

- **Frontend:** Flutter (Dart) for cross-platform UI.
- **Backend:** Firebase Functions, Node.js for API management.
- **Database:** Firebase Firestore (real-time updates).
- **Authentication:** Firebase Auth with Google, Email, and Social Logins.
- **Map Integration:** Google Maps API for geo-located ads.

- **System Integration Procedure**

  - **Backend-Frontend Integration**: Flutter + Firebase.
  - **Database Integration**: Firebase Firestore + Firebase storage.
  - **Authentication & Authorization**: Firebase Auth user authentication.

4. Quality Assurance Practices

- **Code Reviews**

  - Peer code reviews before merging PRs (Pull Requests) in GitHub.
  - Linting tools (ESLint, Pylint) to enforce code standards.
- **Unit & Integration Testing**

  - **Unit Testing**: PyTest for backend APIs, Jest for frontend components.
  - **Integration Testing**: Postman for API validation.
- **Security Testing**

  - **OWASP ZAP & Burp Suite**: Identify security vulnerabilities.
  - **Penetration Testing**: Regular audits to ensure GDPR/HIPAA compliance.
- **Performance Testing**

  - **JMeter & Locust**: Load testing API and frontend performance.
  - **AWS CloudWatch**: Monitoring application performance in real time.
- **Bug Tracking & Defect Management**

  - **Trello & Jira**: Log, track, and resolve defects.
  - **Sentry**: Monitors production issues in real time.

1.3. Configuration Management Plan

The Configuration Management (CM) Plan for AdBoard outlines the procedures for managing project artifacts, tracking changes, and ensuring controlled development and deployment. Version control will be maintained using GitHub with a structured branching strategy, including a stable main branch, an active dev branch, and feature-specific branches. All modifications will follow a formal change request process, with logging, review, and approval before integration. Semantic versioning (vX.Y.Z) will be applied for releases, ensuring clear tracking of major, minor, and patch updates. Regular audits and configuration reviews will be conducted to maintain system integrity, and key personnel will be responsible for monitoring changes, managing baselines, and enforcing compliance with development standards.

1.4. Quality Assurance Plan

The **Configuration Management (CM) Plan** for **Eunoia: AI-Powered Mental Health Support Platform** ensures that all project **artifacts, code, documentation, and configurations** are systematically controlled, tracked, and maintained throughout the project lifecycle.

For larger projects, a separate **Configuration Management Plan document** may be maintained.

1. Configuration Management Activities & Methods

The **CM process** includes:

- **Configuration Identification**: Establishing baselines for key project components.
- **Change Control**: Managing modifications to software and documentation.
- **Status Accounting**: Tracking changes in real-time.
- **Auditing & Version Control**: Ensuring integrity and consistency of configurations.
- **Release Management**: Defining deployment workflows and rollback procedures.

2. Configuration Identification

- **Project Baselines**:
    - **Software Baseline**: Defined at the start of development.
    - **Documentation Baseline**: SRS, SDS, and design documents stored in **Google Drive/GitHub**.
    - **Testing & Compliance Baseline**: Maintained for GDPR/HIPAA compliance audits.
- **Configuration Items (CIs) to be Tracked**:
    - **Source Code** (Frontend, Backend, AI Models)
    - **APIs & Database Schemas**
    - **User Documentation**
    - **Testing Plans & Reports**
    - **Deployment Scripts & Configurations**

3. Change Control & Tracking

- **Change Requests (CRs) Submission**:
    - All **modifications** must be logged in the **Change Control System (Jira/Trello)**.
    - Developers submit **CRs** for approval before implementation.
- **Change Control Board (CCB) Procedures**:
    - Reviews all **CRs affecting project baselines**.
    - **CCB Members**:
        - **Project Manager** (Final Approval)
        - **Technical Lead** (Code & API Changes)
        - **AI/ML Engineer** (Model & Algorithm Adjustments)
        - **Security Engineer** (Compliance & Risk Assessment)

- **Database Schema**: Relational design using **PostgreSQL**.

- **Programming Languages & Standards**
  - **Frontend**: React.js (JavaScript, JSX, Tailwind CSS).
  - **Backend**: Flask (Python), RESTful API structure.
  - **AI/ML**: Hugging Face Transformers, PyTorch, TensorFlow.
  - **Security**: JWT authentication, Flask-Bcrypt for password hashing.
  - **Coding Standards**: PEP8 (Python), Airbnb Style Guide (JavaScript).
- **System Integration Procedure**
  - **Backend-Frontend Integration**: REST APIs (Flask) communicate with React.js.
  - **AI Model Integration**: NLP-based chatbot, sentiment analysis API.
  - **Database Integration**: PostgreSQL as the primary relational database.
  - **Authentication & Authorization**: JWT-based user authentication.

2. Quality Assurance Practices

- **Code Reviews**
  - Peer code reviews before merging PRs (Pull Requests) in GitHub.
  - Linting tools (ESLint, Pylint) to enforce code standards.
- **Unit & Integration Testing**
  - **Unit Testing**: PyTest for backend APIs, Jest for frontend components.
  - **Integration Testing**: Postman for API validation.
- **Security Testing**
  - **OWASP ZAP & Burp Suite**: Identify security vulnerabilities.
  - **Penetration Testing**: Regular audits to ensure GDPR/HIPAA compliance.
- **Performance Testing**
  - **JMeter & Locust**: Load testing API and frontend performance.
  - **AWS CloudWatch**: Monitoring application performance in real time.
- **Bug Tracking & Defect Management**
  - **Trello & Jira**: Log, track, and resolve defects.
  - **Sentry**: Monitors production issues in real time.

3.4. Configuration Management Plan

<This section could contain the configuration management plan for this project. For any but very small projects, this section should refer to a separate document. The CM plan should describe the activities and methods used for configuration identification, control, status

accounting, auditing, and release management. The configuration management plan should address the initial baselining of work products, logging and analysis of change requests, change control board procedures, tracking of changes in progress, and procedures for notifying concerned parties when baselines are established and changed. Estimate the percentage of project effort or the number of hours planned for configuration management activities. Incorporate CM tasks into the project schedule and budget. List the personnel responsible for

establishing the baselines, maintaining the configuration management system, and conducting CM reviews and audits.>

3.5. Quality Assurance Plan

The Configuration Management (CM) Plan for AdBoard ensures that all project artifacts, code, documentation, and configurations are systematically controlled, tracked, and maintained throughout the project lifecycle. For larger projects, a separate Configuration Management Plan document may be maintained.

1. Configuration Management Activities & Methods

The **CM process** includes:

- **Configuration Identification**: Establishing baselines for key project components.
- **Change Control**: Managing modifications to software and documentation.
- **Status Accounting**: Tracking changes in real-time.
- **Auditing & Version Control**: Ensuring integrity and consistency of configurations.
- **Release Management**: Defining deployment workflows and rollback procedures.

2. Configuration Identification

- **Project Baselines**:
    - **Software Baseline**: Defined at the start of development.
    - **Documentation Baseline**: SRS, SDS, and design documents stored in **Google Drive/GitHub**.
    - **Testing & Compliance Baseline**: Maintained for advertising regulations and security audits.
- **Configuration Items (CIs) to be Tracked**:
    - **Source Code** (Frontend, Backend, AI Models)
    - **APIs & Database Schemas**
    - **User Documentation**
    - **Testing Plans & Reports**
    - **Deployment Scripts & Configurations**

3. Change Control & Tracking

- **Change Requests (CRs) Submission**:
    - All **modifications** must be logged in the **Change Control System (Jira/Trello)**.
    - Developers submit **CRs** for approval before implementation.
- **Change Control Board (CCB) Procedures**:
    - Reviews all **CRs affecting project baselines**.
    - **CCB Members**:
        - **Project Manager** (Final Approval)
        - **Technical Lead** (Code & API Changes)
        - **AIEngineer** (Model & Algorithm Adjustments)
        - **Security Engineer** (Compliance & Risk Assessment)

- **Change Tracking & Notifications**:
    - Changes are logged in **Jira/GitHub Issues**.
    - Stakeholders are notified via **Slack/Email**.

4. Version Control & Auditing

Configuration Item	Version Control Tool	Backup Frequency
Source Code	GitHub (branching & pull requests)	Daily
Project Documents	Google Drive (version history)	Weekly
APIs & Database Changes	GitHub Actions	Before Each Deployment
Test Results & Security Audits	Jira (linked reports)	Sprint End
Deployment Configurations	AWS S3, Heroku	Before Release

- Periodic Audits:
    - Monthly CM audits ensure consistency between baselines and changes.
    - Automated CI/CD pipelines validate configurations before deployment.

4. Release Management Process

- **Pre-Release Testing**:
    - Feature testing on **staging environments** before production.
    - Automated **regression testing & performance monitoring**.
- **Deployment Approval**:
    - **Project Manager & Technical Lead** sign off on production deployments.
- **Rollback Strategy**:
    - If a release fails, **GitHub Revert & AWS Backup Restore** are used to roll back.

5. CM Effort Estimation & Resource Allocation

- **Estimated Effort**:
    - **10-12%** of total project effort allocated to CM tasks.
- **Budget Considerations**:
    - AWS S3 storage for configuration backups.
    - GitHub repository costs (if exceeding free-tier limits).
- **Key CM Personnel**:

Role	Responsibility

Project Manager	Oversees CM Plan, approves major changes.
Technical Lead	Manages version control, ensures code integrity.
AI Engineer	Tracks AI model changes & data updates.
DevOps Engineer	Maintains CI/CD, deployment configurations.
Security Engineer	Conducts compliance audits & risk assessments.

6. CM Tasks in Project Schedule

Task	**Responsibility Party**	**Frequency**
Baseline Code & Docs	Dev Team	Initial Setup
Change Request Reviews	CCB	Bi-Weekly
Version Control Audits	Technical Lead	Monthly
Backup & Recovery Testing	DevOps	Quarterly
Security Compliance Review	Security Engineer	Every 3 months

4.5. Documentation Plan

The Documentation Plan for AdBoard ensures that all necessary system documentation is created, reviewed, and maintained throughout the project lifecycle..

Document	**Template/ Standard**	**Created By**	**Reviewed By**	**Target Date**	**Distribution & Storage**
Software Requirements Specification (SRS)	IEEE 830 Standard	Project Manager	Stakeholders & Developers	Week 4	Google Drive, GitHub
System Design Document (SDS)	IEEE 1016 Standard	Technical Lead	Project Manager	Week 6	Google Drive, GitHub
API Documentation	OpenAPI Standard (Swagger)	Backend Developer	Technical Lead	Week 12	GitHub, ReadTheDocs

Database Schema & ERD	UML Standards	Database Engineer	Backend Developer	Week 8	GitHub, Google Drive
User Guide (Web App & Features)	Markdown, PDF	UI/UX Team	QA Team	Week 20	Online Help System, PDF Manual
Installation & Maintenance Guide	DevOps Standards	DevOps Engineer	Project Manager	Week 22	GitHub Wiki, AWS Docs
Test Plan & Reports	IEEE 829 Standard	QA Engineers	Security & Compliance Team	Week 16	Jira, Google Drive
Security & Compliance Report	Advertising Regulations	Security Officer	Legal Advisor	Week 18	Secure Cloud Storage
Release Notes & Change Log	Semantic Versioning	Technical Lead	Project Manager	Each Release	GitHub, Website
Final Project Report	Project Documentation Standard	Project Manager	Stakeholders	Week 26	Google Drive, University Submission

Document Review & Approval Process

- **Drafting Phase**: Each document is initially created by the responsible team.
- **Internal Review**: Reviewed by relevant **developers, QA, and security teams**.
- **Stakeholder Review**: Project sponsors and external reviewers validate critical documents.
- **Approval & Baselining**: Finalized documents are archived in **Google Drive & GitHub**.

Distribution & Storage

- **Development Documents**: Stored in **GitHub (Private Repository)** and **Google Drive**.
- **User-Facing Documentation**: Available in an **online help system (PDF & Web Format)**.
- **Security & Compliance Reports**: Restricted access via **encrypted cloud storage**.

Updates & Maintenance

- **Documentation updates follow Agile sprints**, with **version control in GitHub**.
- **Release notes and changelogs** are updated **with every major deployment**.
- **Final documentation is archived at project closure** for future reference.

4.6. Process Improvement Plan

The Process Improvement Plan for AdBoard focuses on assessing project workflows, identifying areas for optimization, and implementing improvements while ensuring minimal disruption to development.

Process Improvement Activities

Improvement Area	New Approach/ Enhancement	Anticipated Impact
Requirement Management	Use Jira's Roadmap Feature for tracking changes in requirements.	Improved visibility and traceability.
Code Quality & Reviews	Implement pre-commit hooks and automated linting for Flutter and Django.	Reduces code inconsistencies.
Testing Efficiency	Introduce Selenium & Cypress for automated UI testing.	Increases test coverage, reducing manual testing effort.
Security & Compliance Audits	Automate security vulnerability scans using OWASP ZAP in CI/CD pipelines.	Early detection of security risks.
Deployment & Performance Optimization	Transition from Heroku free tier to AWS Auto-Scaling & CloudFront caching.	Improved system reliability.
User Experience Monitoring	Integrate Hotjar & Google Analytics to analyze user behavior.	Data-driven UI/UX improvements.

1. Tracking & Evaluation of Process Improvements

- Bi-Weekly Sprint Reviews: Evaluate if the implemented improvements are reducing bottlenecks.
- Key Performance Indicators (KPIs):
  - Reduction in code review turnaround time.
  - Increase in test automation coverage.
  - Improvement in system uptime and response time.
  - Reduction in security vulnerabilities detected post-release.
- Modification & Adaptation: If an improvement does not deliver expected results, adjustments are made without disrupting active development.

2. Lessons Learned & Long-Term Impact

- Process improvement experiences will be documented in Section 3.6 (Project Close-Out Plan).
- Successful improvements will be integrated into future versions of AdBoard or similar projects.
- Unsuccessful or inefficient approaches will be analyzed for potential refinements.