The Islamia University of Bahawalpur

Software Requirement Specifications

By

Student Name: Kshmala Nawaz

Roll No: F22BINFT1M01150

Session: 2022 – 2026

Supervisor: Sir Muzammil Ur Rehman

Bachelor of Science in Information Technology

Project Title
Mental_Health_monitoring_system

Table of Contents

1 Introduction

1.1 Purpose

This Software Requirements Specification (SRS) document outlines the requirements for the development of the **AI-based Mental Health Monitoring App**. The main purpose of this project is to provide individuals with a digital platform that helps monitor, understand, and manage their mental health through daily mood tracking, journaling, and AI-driven insights. The app is not intended to replace professional healthcare but to support mental well-being and encourage self-awareness through technology.

1.2 Scope

The app allows users to log their emotional state, write personal journals, and receive real-time feedback based on Natural Language Processing (NLP) and sentiment analysis. Over time, the system visualizes trends and patterns in users' emotional data, helping them gain better insight into their mental well-being. The app also recommends personalized wellness tips and coping strategies based on historical mood data. It will be developed for Android platforms using Flutter and Firebase as a backend. AI models will be integrated using Python APIs or a suitable ML framework.

1.3 Intended Audience and Reading Suggestions

This document is intended for:

- **Project team members** (developers, designers) who will use it to understand system specifications and plan implementation.
- **Academic supervisors and evaluators** for assessment of the project's planning and technical completeness.
- **Stakeholders** (such as mental health enthusiasts, end-users, or support groups) to understand how the app works and its purpose.
- **Testers and QA team** to verify all specified requirements are implemented and validated.

It is suggested that:

- Developers focus on sections 4–6 (Design Models, Data Design, Algorithms).
- Testers refer to section 7 (Traceability Matrix) to map tests against requirements.
- Stakeholders and supervisors refer to sections 1–3 for an overall understanding.

1.4 Document Organization

This SRS is organized as follows:

- **Section 1 (Introduction)** provides an overview and purpose of the document.
- **Section 2 (Design Methodology)** describes the development model used.
- **Section 3 (System Overview)** gives a high-level description of the system.
- **Section 4 (Design Models)** presents diagrams and models like DFDs and class diagrams.
- **Section 5 (Data Design)** discusses data structures and the data dictionary.
- **Section 6 (Algorithms & Implementation)** explains logic and functions using pseudocode.
- **Section 7 (Requirements Traceability Matrix)** maps functional requirements to implementation.
- **Section 8 (Human Interface Design)** shows UI design and user interaction descriptions.

2. Overall Description

2.1 Product Perspective

The **AI-based Mental Health Monitoring App** is an entirely **new standalone product** designed from the ground up to assist users in understanding and improving their mental well-being through AI-assisted mood tracking, journaling, and trend analysis. It is not an extension or an update to an existing system but a **first-of-its-kind implementation** tailored for individual mental wellness monitoring through mobile technology.

The app will utilize a modular architecture, enabling future upgrades and integration with additional services such as:

- Therapy chat-bot modules
- Mental health professional portals
- Integration with fitness or health apps (e.g., Google Fit or Apple Health)

This software will provide **offline logging** capabilities for basic functions like mood tracking and journaling, with **AI features and analytics operating online** due to cloud-based NLP processing.

It stands independently from other platforms but may connect to cloud storage services (e.g., Firebase) and NLP APIs (e.g., Tensor-Flow, Hugging-Face models) for enhanced functionality.

2.2 Operating Environment

The software will be developed as a **cross-platform mobile application** using Flutter. It will primarily target Android OS in its initial release, with optional support for iOS in later phases.

Client-side Operating Environment:

- **Mobile Devices:** Android smartphones with Android version 8.0 (Oreo) or above
- **Minimum Hardware Requirements:**
    - RAM: 2 GB or higher
    - Storage: 100 MB minimum free space
    - Processor: Quad-core or higher
- **Mobile Operating Systems Supported:**
    - Android 8.0 to Android 14.0
    - iOS support (Planned – Optional future scope)

Backend/Server Environment:

- **Cloud Services:** Google Firebase (Fire-store, Authentication, and Storage)
- **AI Processing Environment:**
    - Hosted Python APIs for sentiment analysis (deployed on cloud functions or a backend server)
    - Tensor-Flow, Scikit-learn, or Hugging-Face transformers for NLP processing
- **Database:** Cloud Fire-store (NoSQL)

Browser Compatibility (for Admin Panel / Web Portal – if implemented):

- Google Chrome (all versions)
- Mozilla Firefox (v70+)
- Microsoft Edge (v85+)
- Safari (v12+)

Geographical Reach:

- The app will initially serve users in **Pakistan**, with language support for **English** (Urdu in future iterations).
- Hosted on cloud servers located in regions with **GDPR-compliant infrastructure** (e.g., Google Cloud servers in the EU or US).

Hosting Organizations:

- The app backend and AI services will be hosted via **Google Firebase** and optionally on **Python Anywhere, Heroku, or Google Cloud Functions**, depending on cost and performance factors.

2.3 Design and Implementation Constraints

This section outlines the specific constraints and limitations placed on the design and implementation of the app. These restrictions influence the technology stack, programming languages, tools, and frameworks to be used.

Constraint	Description	Rationale
C1: Cross-Platform Development Framework	Flutter must be used for app development.	Ensures consistent UI and logic across Android (and optionally iOS) with one codebase.
C2: Backend Technology	Google Firebase for authentication, data storage, and analytics.	Firebase offers real-time database, easy authentication, and good integration with Flutter.
C3: Programming Language	Dart (for Flutter), Python (for AI sentiment analysis).	Dart is required for Flutter development; Python is preferred for AI and NLP tasks due to its rich libraries (NLTK, Text Blob, transformers).
C4: Internet Dependency	AI-driven functions require an active internet connection.	NLP sentiment models are hosted in the cloud for scalability and ease of updates.
C5: Data Privacy and Security	User data must be encrypted and anonymized.	Protects sensitive mental health information, ensuring user trust and compliance with privacy laws.
C6: Device Compatibility	Must support low-to-mid range Android devices.	Increases accessibility, especially for users in developing regions.
C7: Third-Party API Usage	Use of APIs (e.g., Open AI, Hugging Face, or Google NLP) for sentiment processing.	Reduces development time and improves model accuracy without training from scratch.
C8: Time and Resource Limits	Project must be completed within an academic semester with limited resources.	Influences scope prioritization and feature set.

Summary of Key Decisions Based on Constraints:

- The use of **Flutter + Firebase** allows rapid development and hosting under budget.
- **Python APIs** are used for AI due to their maturity and open-source NLP ecosystem.
- **Internet-based AI processing** keeps the app light and allows powerful models to run server-side.
- The app avoids direct diagnosis of mental conditions to **stay within ethical boundaries** and avoid legal liability.

This section describes the method used to derive the functional requirements of the application. For this AI-based Mental Health Monitoring App, the **Use Case** technique is adopted. This method helps identify system functionality from the end-user's perspective and guides the development of detailed software features.

3.1 Use Case Diagram

The use case diagram below illustrates key interactions between the primary actor (the user) and the system.

Actors:

- **User (Primary Actor)**: Uses the app for mental health tracking.
- **AI Engine**: Provides sentiment analysis and mental health insights.
- **Authentication System**: Handles login and registration.
- **Database (Firebase)**: Stores user data securely.

Use Cases:

- Register an account
- Log in
- Input daily journal/mood
- Get mental health insights
- View progress/history
- Receive motivational tips
- Log out

(You can create this diagram using tools like MS Visio, Lucid chart, or draw.io. If you want, I can provide a sketch or diagram too.)

3.2 Use Case Description

Use Case Template: Track Daily Mood

Attribute	Details
Use Case ID	UC-1
Use Case Name	Track Daily Mood
Actors	Primary Actor: User Secondary Actors: AI Engine, Firebase Database
Description	The user logs into the app and inputs their current mood and feelings. The system stores the

7

Attribute	Details
	data and processes it using AI to provide feedback and track mental health trends over time.
Trigger	User taps on "Track Mood" button from home screen.
Preconditions	PRE-1: User is registered and logged in. PRE-2: Internet connectivity is available.
Post conditions	POST-1: Mood entry is saved in the database. POST-2: AI-generated insights are displayed to the user. POST-3: Entry is included in the user's progress chart.

Normal Flow:

1. User selects "Track Mood" from the dashboard.
2. System displays mood scale, emotion tags, and journal input field.
3. User inputs their mood (e.g., happy, anxious) and types journal entry.
4. User submits the entry.
5. System stores the input in Firebase.
6. AI Engine analyzes journal entry and returns sentiment result.
7. App displays feedback message and suggested coping tip.
8. System updates user's mental health trends.

Alternative Flows:

- **AF-1: User skips journal input**
  - System still saves mood rating but skips AI analysis.
  - Continue to normal flow step 8.
- **AF-2: AI Engine not responding**
  - App displays: "Analysis unavailable at the moment."
  - Mood entry still saved in history.

Exceptions:

- **EX-1: No Internet Connection**
  - System displays error message: "You are offline. Mood entry will be saved locally and synced later."
  - Entry is stored temporarily in local cache.

Business Rules:

- **BR-1**: Only one mood entry per day is allowed (to ensure daily reflection).
- **BR-2**: Journal entries must be between 10 and 500 words.
- **BR-3**: AI analysis is based on NLP models trained on mental health-related text datasets.

- ASSUMPTION-1: At least 70% of users will use the journaling feature regularly.
- ASSUMPTION-2: AI Engine can handle up to 1000 concurrent sentiment requests.
- ASSUMPTION-3: Firebase Authentication ensures all users are properly identified.

Use Case Name	Description
Register User	New users sign up with email/password.
Login	Existing users log in to their account.
Input Daily Mood	Users record daily mood and journal entry.
Analyze Entry	AI Engine performs sentiment analysis.
View Insights	Display AI-based interpretation of mental health trends.
View History	Show past moods and journal entries in a timeline.
Get Motivational Tip	Show daily motivational quotes based on user's emotional state.
Log Out	End user session securely.

✅UC-2: Register New User

Attribute	Details
Actors	User, Authentication Service
Description	User registers by providing their email, password, and optional profile info.
Trigger	User selects "Sign Up"
Preconditions	Device is connected to the internet
Post conditions	New account is created and saved in Firebase Auth

Normal Flow:

1. User taps "Sign Up"
2. Enters email, password, and optional name/photo.
3. System validates input and creates account.
4. System sends confirmation email.
5. User redirected to dashboard after confirmation.

Exceptions:

- Email already exists → Show error.
- Weak password → Prompt stronger password.

✅ UC-3: Input Daily Mood and Journal

| **Actors** | User, AI Engine, Database | | **Description** | Records emotional state and performs AI-based analysis | | **Trigger** | User opens app and taps "Track Mood" | | **Preconditions** | Logged-in user | | **Post conditions** | Data is saved and feedback is shown |

Normal Flow:

1. App asks, "How are you feeling today?"
2. User selects mood (happy, sad, anxious, etc.)
3. Optional: User writes a journal entry
4. Data submitted to backend
5. AI Engine analyzes and returns insights
6. User sees suggestions and feedback

Alternative Flows:

- No journal → Proceed without analysis
- AI fails → Show "analysis unavailable"

✅ UC-4: View Mental Health Insights

| **Actors** | User, AI Engine | | **Description** | Shows AI-generated interpretation of mood trends | | **Trigger** | User taps "Insights" | | **Normal Flow:**

1. System retrieves past mood entries
2. AI processes trends (e.g., frequent sadness)
3. Insights are shown in graphs or messages
4. Suggestions for improvement provided

✅ UC-5: Receive Motivational Tip

| **Actors** | User, Notification Service | | **Trigger** | App notification or Insights screen | | **Normal Flow:**

1. Based on emotional pattern, system selects a quote or motivational message
2. Tip is displayed with soothing visuals or audio

Assumptions (Expanded)

- At least 50% of users will return daily to track their mood.
- Data privacy and encryption will be ensured through Firebase rules.
- The system assumes users will not provide fake or misleading inputs regularly.
- AI Engine is expected to handle sentiment in English (Phase 1), multilingual in future.

4 Functional Requirements

4.1 Functional Requirement FR01

Identifier	**FR01**
Title	User Registration and Login
Requirement	The user shall be able to register with the app using an email address and password. The system shall validate credentials and grant access to registered users only.
Source	End-user feedback, common authentication practices
Rationale	To provide secure and personalized access to user-specific mental health data
Business Rule	BR01: Users must provide a valid email address and a password of at least 8 characters.
Dependencies	None
Priority	High

4.2 Functional Requirement FR02

Identifier	**FR02**
Title	Daily Mood Logging
Requirement	The user shall be able to input their mood through a set of emoji's or text input daily. The system shall store the mood input along with a timestamp.
Source	Feature requirement from app purpose
Rationale	To collect daily emotional data for AI-based analysis
Business Rule	BR02: Only one mood entry is allowed per user per day.
Dependencies	FR01
Priority	High

4.3 Functional Requirement FR03

Identifier	**FR03**
Title	AI-Based Insight Generation
Requirement	Once sufficient data is collected, the system shall analyze user inputs to generate weekly mental health insights.
Source	AI engine design
Rationale	To help users understand trends and patterns in their mental well-being
Business Rule	BR03: At least 5 mood entries in a week must exist for an insight to be generated.
Dependencies	FR02
Priority	High

4.4 Functional Requirement FR04

Identifier	**FR04**
Title	Reminder Notifications
Requirement	The system shall send daily reminders to users to enter their mood if they haven't already done so by 8 PM.
Source	Usability requirement

Identifier	**FR04**
Rationale	To ensure consistent data collection for accurate analysis
Business Rule	BR04: Reminders are sent via push notification only if the user has notifications enabled.
Dependencies	FR01, FR02
Priority	Medium

4.5 Functional Requirement FR05

Identifier	**FR05**
Title	Data Visualization
Requirement	The system shall display the user's mood trends in a graphical format (e.g., bar or line chart).
Source	UI/UX requirement
Rationale	To provide a visual summary of user mood over time
Business Rule	BR05: Only the current month's data is shown in the chart view by default.
Dependencies	FR02
Priority	Medium

4.6 Functional Requirement FR06

Identifier	**FR06**
Title	On-the-Ward Monitoring
Requirement	The system shall continuously monitor the user's mental health patterns and alert assigned caregivers or health professionals in case of severe distress patterns, based on threshold criteria identified by the AI engine.
Source	Hospital staff and healthcare provider interviews; AI use case definition
Rationale	To provide proactive intervention and support in clinical environments where patients may be at risk of mental health deterioration
Business Rule	BR06: Alerts are triggered if a user shows three or more consecutive days of severe distress indicators or if a suicidal ideation keyword is detected in input.
Dependencies	FR02 (Daily Mood Logging), FR03 (AI-Based Insight Generation)
Priority	High

Explanation:

"On-the-Ward Monitoring" refers to the system's ability to act as a support tool in environments such as:

- Hospitals
- Mental health clinics
- Rehabilitation centers
- Elderly care homes

Here's what this functionality involves:

- **Real-time input analysis** from patients' daily logs (emotional input, voice logs, text entries).

- **AI pattern recognition** to identify critical behavior trends.
- **Immediate alerts** sent to caregivers or psychiatrists via email/SMS/push notifications.
- **Role-based access**: Caregivers assigned to patients can view their dashboards securely.

5. Non-Functional Requirements

This section defines quality requirements that do not describe specific system functionality but are essential to ensure the overall system performance, reliability, and user satisfaction. These requirements are equally critical for acceptance, especially in mental health systems that demand confidentiality, speed, stability, and accessibility.

5 Non-Functional Requirements
5.1 Usability

Usability requirements ensure the app is accessible and effective for all user types, including those with mental health challenges.

Identifier	USE-1
Requirement	The system shall allow users to submit a daily mood log in under 1 minute using a guided, intuitive form.
Source	UX Research & End-User Feedback
Rationale	Users with low motivation or energy (common in mental health issues) need quick and easy interaction.
Verification	User testing with 95% success rate in mood logging within 60 seconds during trial sessions.
Identifier	USE-2
Requirement	The system shall support accessibility features such as text-to-speech, high-contrast mode, and large-text options for users with visual or cognitive difficulties.
Source	Accessibility Guidelines (WCAG 2.1)
Rationale	Inclusive design is vital for a healthcare app targeting a diverse user base.
Verification	Accessibility audit using WCAG testing tools and compliance checklist.
Identifier	USE-3
Requirement	The system shall allow a first-time user to complete the onboarding process (registration, tutorial, and first log entry) within 5 minutes.
Source	Human-Centered Design Best Practices
Rationale	Smooth onboarding improves user retention and engagement.
Verification	Observational testing of new users during pilot study.

5.2 Performance

These requirements focus on the app's response time, data processing speed, and efficiency under load.

Identifier	PER-1
Requirement	95% of app screens shall load fully within 3 seconds under a 10 Mbps internet connection.
Source	UI Performance Standards
Rationale	Fast response time improves user experience and reduces frustration.
Verification	Load testing using tools like or Firebase Performance Monitoring.

Identifier	PER-2
Requirement	The AI model shall process and generate mental health insights within 2 seconds after the user submits their log.
Source	AI Module Design Requirements
Rationale	Timely insights are critical for keeping the user engaged and building trust in the system.
Verification	Benchmark tests during AI inference execution.

Identifier	PER-3
Requirement	The system shall support a minimum of 10,000 concurrent users with no more than 2% failure rate in response delivery.
Source	Scalability Planning
Rationale	Expected scalability for potential partnerships with healthcare providers.
Verification	Stress testing in cloud environments (e.g., AWS, Azure).

7 References

This section provides a list of documents, standards, textbooks, tools, and web resources referred to or used as supporting material throughout the preparation and development of this SRS.

Ref. No.	Reference Title	Description
[REF-01]	IEEE Std 830-1998	IEEE Recommended Practice for Software Requirements Specifications. Used as the structural basis for organizing this SRS document.
[REF-02]	Sommerville, Ian. "Software Engineering", 9th Edition	Standard textbook used for software engineering principles including architecture, design constraints, and requirement gathering.
[REF-03]	Wiegers, Karl E., and Joy Beatty. "Software Requirements", 3rd Edition	Referenced for functional and non-functional requirement writing techniques, use case descriptions, and traceability matrix guidelines.
[REF-04]	Pressman, Roger S. "Software Engineering: A Practitioner's	Used for guidance in system modeling, DFDs, and requirement engineering practices.

Ref. No.	Reference Title	Description
		Approach"
[REF-05]	AgileModeling.com: UML Artifacts	http://agilemodeling.com/artifacts/ – Used for guidance on creating UML diagrams including use case diagrams, activity diagrams, and class diagrams.
[REF-06]	WCAG 2.1 Accessibility Guidelines	https://www.w3.org/TR/WCAG21/ – Used to ensure the user interface is accessible to users with disabilities.
[REF-07]	Ullman, Jeffrey D. – "The Complete Book" (ER Diagram Reference)	A reference for creating entity-relationship diagrams for data design.
[REF-08]	AgileModeling.com: Data Flow Diagrams	http://www.agilemodeling.com/artifacts/dataFlowDiagram.htm – Used for DFDs that show system functionality and process flow.
[REF-09]	Firebase Documentation	https://firebase.google.com/docs – Used for implementation details of cloud storage, authentication, and real-time database integration.
[REF-10]	Open AI API Documentation	https://platform.openai.com/docs – Reference for integrating GPT-based models to analyze user logs and provide mental health feedback.
[REF-11]	Android Developer Guide	https://developer.android.com/guide – Used for mobile application development guidelines in Android Studio.
[REF-12]	Flutter Documentation	https://flutter.dev/docs – Used for cross-platform mobile app UI design and development.
[REF-13]	WHO Mental Health Data and Research	https://www.who.int/teams/mental-health-and-substance-use – Used to align features with standard global mental health indicators.
[REF-14]	DSM-5 (Diagnostic and Statistical Manual of Mental Disorders, 5th ed.)	Used for defining symptoms, diagnostic criteria, and mood tracking categories in the mental health domain.