# Software Requirements Specification (SRS)

## 1. Introduction

### 1.1 Purpose

This Software Requirements Specification (SRS) document describes the functional and non-functional requirements of the **Chat It** application. The document serves as a reference for developers, designers, testers, and stakeholders involved in the development of the system.

### 1.2 Scope

Chat It is a mobile-based social media application that combines traditional social networking features with real-time shared experiences such as **synchronized music listening** and **mini-games inside chat**. The application aims to improve user engagement by enabling interactive and emotional connections between users.

### 1.3 Definitions, Acronyms, and Abbreviations

- **App**: Chat It mobile application
- **MVP**: Minimum Viable Product
- **RLS**: Row Level Security
- **API**: Application Programming Interface
- **DB**: Database

### 1.4 References

- Supabase Documentation
- React Native Documentation
- Agile Software Development Model

---

## 2. Overall Description

### 2.1 Product Perspective

Chat It is a standalone mobile application developed using **React Native** for the frontend and **Supabase** as the Backend-as-a-Service (BaaS). The system integrates third-party music APIs to provide synchronized music playback between users.

### 2.2 Product Functions

The major functions of Chat It include: - User authentication and profile management - Posting photos and videos - Viewing and interacting with feeds and stories - Real-time one-to-one chat - Shared music listening - Mini-games inside chat - Notifications

### 2.3 User Classes and Characteristics

| User Class | Description |
|---|---|
| Regular User | Can post, chat, listen to music, and play games |
| Admin (Future) | Manages content and moderation |

### 2.4 Operating Environment

- Android OS
- iOS
- Internet connection required
- Runs on modern smartphones

### 2.5 Design and Implementation Constraints

- Supabase free-tier limitations
- Music API usage restrictions
- Compliance with app store policies

### 2.6 Assumptions and Dependencies

- Users have stable internet connectivity
- Third-party APIs remain available
- Supabase services are operational

---

## 3. System Features and Functional Requirements

### 3.1 User Authentication

**Description:** Allows users to register and log in securely.

**Functional Requirements:** - Users shall be able to sign up using email/password. - Users shall be able to log in using Google or phone OTP. - Users shall be able to log out securely.

---

### 3.2 User Profile Management

**Description:** Allows users to manage their profiles.

**Functional Requirements:** - Users shall be able to upload a profile picture. - Users shall be able to edit bio and username. - Users shall be able to view other users’ profiles.

---

### 3.3 Feed and Post Management

**Description:** Enables users to create and interact with posts.

**Functional Requirements:** - Users shall be able to upload photos and videos. - Users shall be able to like, comment, and share posts. - Users shall be able to view posts from followed users.

---

### 3.4 Stories Feature

**Description:** Temporary content visible for 24 hours.

**Functional Requirements:** - Users shall be able to upload image or video stories. - Stories shall automatically expire after 24 hours. - Users shall be able to react to stories.

---

### 3.5 Chat System

**Description:** Real-time messaging between users.

**Functional Requirements:** - Users shall be able to send and receive messages in real time. - Users shall be able to send text, images, and voice notes. - Messages shall be delivered instantly using Supabase Realtime.

---

### 3.6 Shared Music Listening (Tune Together)

**Description:** Enables two users to listen to the same song simultaneously.

**Functional Requirements:** - Users shall be able to start a music session in chat. - Both users shall hear the same song in sync. - Play and pause actions shall be synchronized.

---

### 3.7 Mini-Games in Chat (Play It)

**Description:** Enables real-time multiplayer mini-games.

**Functional Requirements:** - Users shall be able to start a mini-game inside chat. - Game state shall be synchronized in real time. - The system shall track turns and results.

---

### 3.8 Notifications

**Description:** Alerts users about activities.

**Functional Requirements:** - Users shall receive notifications for messages. - Users shall receive notifications for likes and comments.

# 4. External Interface Requirements

## 4.1 User Interfaces

- Mobile-friendly UI
- Dark and light themes
- Simple and intuitive navigation

## 4.2 Hardware Interfaces

- Smartphone camera
- Microphone (for voice notes)

## 4.3 Software Interfaces

- Supabase APIs
- Music streaming APIs

## 4.4 Communication Interfaces

- HTTPS
- WebSockets (Realtime)

# 5. Non-Functional Requirements

## 5.1 Performance Requirements

- Chat messages should be delivered within 300ms.
- App should load within 3 seconds.

## 5.2 Security Requirements

- User data shall be protected using RLS.
- Secure authentication tokens shall be used.

## 5.3 Reliability

- System uptime should be at least 99%.

## 5.4 Usability

- Easy onboarding
- Minimal learning curve

### 5.5 Scalability

- System should support growth up to 50,000 users (MVP).

---

## 6. Future Enhancements

- Group music sessions
- Advanced mini-games
- AI-based content recommendations
- Creator monetization features

---

## 7. Appendices

### Appendix A: Glossary

- **Realtime Sync**: Live data synchronization between users
- **Duo Chat**: Enhanced one-to-one chat with music and games

---

**Document Version:** 1.0
**Status:** Approved for Development