# Title

WhatsApp Application - Software Requirement Specification (SRS) and Technical Design Document  
  
**1. Introduction**  
This document outlines the Software Requirement Specification (SRS) and Technical Design for the WhatsApp mobile application. It includes functional and non-functional requirements, use case scenarios, technical architecture, data flow diagrams, and user interface layout descriptions.  
  
**2. Overall Description**  
WhatsApp is a cross-platform messaging and Voice over IP (VoIP) service that allows users to send text messages, voice messages, images, videos, documents, and make voice and video calls. The application supports individual and group communication, status updates, and secure transactions via UPI.  
  
**3. Functional Requirements**  
- Send and receive text messages  
- Share photos and videos in HD quality  
- Upload and view WhatsApp status  
- Set and update profile picture (DP)  
- Backup and restore chat history  
- Download and save media to gallery  
- Create and manage WhatsApp groups  
- Send and receive UPI transactions  
- Block and unblock contacts  
- Make audio and video calls  
- Manage notifications and privacy settings

**4. Non-Functional Requirements**  
- Performance: Fast message delivery and media sharing  
- Scalability: Support for millions of concurrent users  
- Security: End-to-end encryption for messages and calls  
- Usability: Intuitive and user-friendly interface  
- Reliability: High availability and fault tolerance  
- Compatibility: Support for Android and iOS platforms  
  
**5. Use Case Scenarios**  
- User sends a message to a contact  
- User uploads a status update  
- User initiates a video call  
- User sends money via UPI  
- User blocks a spam contact  
- User joins a group and shares media  
  
**6. Technical Architecture**  
The WhatsApp application follows a client-server architecture:  
- Client: Mobile application installed on user devices  
- Server: Backend infrastructure handling message routing, media storage, and authentication  
- Database: Stores user data, chat history, and media files  
- APIs: RESTful APIs for communication between client and server  
- Encryption: End-to-end encryption using Signal Protocol  
  
**7. Data Flow Diagrams**  
Level 0: User interacts with the app to send a message  
Level 1: Message is encrypted and sent to the server  
Level 2: Server routes the message to the recipient  
Level 3: Recipient receives and decrypts the message

**8. User Interface Layout Descriptions**  
Chat Interface:  
- Camera Icon: Bottom-left corner to capture and send media  
- Gallery Icon: Select media from device  
- Document Icon: Send files  
- Microphone Icon: Record and send voice messages  
- Emoji and Sticker Icons: Add expressive content  
Status Interface:  
- Plus Icon: Upload new status  
- Eye Icon: View status viewers  
Call Interface:  
- Video Call Icon: Start video call  
- Speaker and Mute Icons: Control audio  
- End Call Icon: Terminate call  
Settings Interface:  
- Lock Icon: Encryption status  
- Settings Icon: Access account and privacy settings  
- Search Icon: Find messages or contacts  
  
**9. External Interface Requirements**  
- Integration with phone contacts  
- Access to camera, microphone, and storage  
- Internet connectivity for real-time communication  
- Push notification services for message alerts  
  
**10. Software Quality Attributes**  
- Maintainability: Modular code structure  
- Portability: Compatible across devices  
- Efficiency: Optimized for low bandwidth usage  
- Testability: Automated testing for features

**11. Appendices**  
- Technologies: Java/Kotlin (Android), Swift (iOS), Node.js (Backend)  
- APIs: Firebase, Twilio, UPI integration  
- Glossary: DP (Display Picture), UPI (Unified Payments Interface), VoIP (Voice over IP)

**WhatsApp Application - Functional Requirements**  
**1. Messaging**  
Users must be able to send and receive text messages in real-time. Messages should be delivered instantly and show delivery/read receipts (single tick, double tick, blue tick).  
**2. Media Sharing**  
Users can share photos, videos, documents, and audio files. Media should be compressed or sent in HD quality based on user preference.  
**3. Status Updates**  
Users can upload temporary status updates (text, image, video) that disappear after 24 hours. View counts and privacy settings must be supported.  
**4. Profile Management**  
Users can set and update their profile picture (DP), name, and about info. These should be visible to contacts based on privacy settings.  
**5. Backup and Restore**  
Chat history and media should be backed up to cloud storage and restored during reinstallation or device change.  
**6. Media Download and Save**  
Users can download and save received images and videos to their device gallery. Auto-download settings should be configurable.  
**7. Group Chats**  
Users can create groups, add participants, assign admins, and manage group settings. Group  
messaging should support media and document sharing.  
**8. UPI Transactions**  
Users can send and receive money using UPI integration. Transaction history and security  
verification must be included.  
**9. Contact Blocking**  
Users can block contacts to prevent receiving messages, calls, and status updates from them.

**10. Audio and Video Calls**  
Users can initiate and receive encrypted audio and video calls. Call quality should adapt to network conditions.  
**11. Notifications**  
Users should receive notifications for new messages, calls, and status updates. Notification settings must be customizable.  
**12. Search Functionality**  
Users can search messages, contacts, and media within chats and across the app.  
**13. Multi-device Support**  
Users can access WhatsApp on multiple devices simultaneously with synchronized data.  
**14. Privacy Settings**  
Users can configure who can see their last seen, profile photo, status, and read receipts.  
**15. Chat Archiving and Deletion**  
Users can archive chats to hide them from the main screen or delete them permanently.  
**16. Message Reactions**  
Users can react to messages with emojis for quick feedback.  
**17. Message Forwarding**Users can forward messages to other contacts or groups with a forwarded label.  
**18. Starred Messages**  
Users can mark important messages as starred for quick access later.  
**19. QR Code Contact Sharing**  
Users can share their contact info via QR code for easy addition.  
**20. App Settings**  
Users can configure chat, notification, storage, and account settings from the settings menu.