Software Requirements Specification (SRS) for NeuroGuard

1. Introduction

1.1 Purpose

The purpose of this Software Requirements Specification (SRS) is to define the complete set of requirements for NeuroGuard, an AI-powered wearable health monitoring system designed as a handband or headband to monitor vital signs, neural signals, and movements for patients, with a primary focus on predicting or detecting epilepsy seizures in real-time. The system analyzes time series data from integrated sensors to forecast seizures 1-5 minutes in advance and sends immediate alerts to family members or doctors via a mobile app, cloud dashboard, sound/vibration alarms on the device, or emergency SMS/calls. This document serves as a contract between stakeholders, developers, and testers, ensuring all parties have a shared understanding of the system's capabilities, constraints, and quality attributes.

This SRS provides exhaustive details, including traceable requirements, use cases, and verification criteria, to facilitate agile development and rigorous testing. It incorporates the core idea of a multi-purpose healthcare wearable that combines AI, IoT, embedded systems, and mobile technology, with potential future expansions for conditions like stroke risk, heart arrhythmias, sleep apnea, fall detection, panic attacks, Parkinson's disease monitoring, and athlete performance tracking using the same hardware.

1.2 Document Conventions

- **Requirements Notation:** All requirements are prefixed with "REQ-" followed by a category (e.g., FUNC for functional, PERF for performance) and a unique ID. Priorities are indicated as: [High], [Medium], [Low].
- **Traceability:** Each requirement links back to business needs or user flow elements.
- **Fonts: Bold** for key terms, *italics* for definitions.
- **Diagrams:** References to the user flow diagram are denoted as [UFD-Section]. The provided flow diagrams illustrate hardware integration, data processing, and app workflows, including continuous monitoring, anomaly detection, and alert escalation paths.

1.3 Intended Audience and Reading Suggestions

- **Project Managers:** Focus on Sections 1, 2, and 4 for high-level overview and planning.
- **Developers:** Refer to Section 3 for detailed functional and non-functional specs.
- **Testers:** Use Section 3 and Appendix B for test case derivation.
- **Stakeholders:** Review Section 2 for product perspective and user classes.

1.4 Scope

NeuroGuard encompasses a wearable device (handband or headband) integrated with sensors for data collection, firmware for processing, a cross-platform mobile application (built using Flutter), and a cloud dashboard for real-time monitoring and alerts. In-scope features include:

- Continuous recording of neural signals (EEG/EMG via BioAmp), heart rate and blood oxygen (HR/SpO2 via MAX30102), and movements (accelerometer/gyroscope via MPU6050).
- AI-driven analysis of time series data for seizure prediction (1-5 minutes advance warning) and detection.
- Alerts via device alarms (sound/vibration), mobile push notifications, cloud dashboard updates, and emergency SMS/calls (via SIM800L in offline scenarios).
- Prototyping with Arduino UNO, wireless connectivity via ESP32 (Wi-Fi/Bluetooth), and real-time data storage/processing using Firebase.
- Mobile app and cloud dashboard for displaying patient status, alerts, statistics, and caregiver access.

Out-of-scope:

- Custom hardware manufacturing beyond prototyping.
- On-premises deployment (cloud-only via Firebase).
- Advanced medical diagnostics or treatment (system provides monitoring and alerts only).
- Integration with hospital systems without APIs.
- Immediate implementation of future expansions (e.g., stroke or sleep apnea detection), though the architecture supports them.

The system is designed for scalability, supporting individual patients to caregiver-managed groups.

1.5 Definitions, Acronyms, and Abbreviations

- **AI/ML:** Artificial Intelligence/Machine Learning – Core technology for time series analysis and anomaly prediction.
- **API:** Application Programming Interface – For integrations.
- **EEG/EMG:** Electroencephalography/Electromyography – Neural signal measurements.
- **HR/SpO2:** Heart Rate/Blood Oxygen Saturation.
- **GDPR/HIPAA:** General Data Protection Regulation/Health Insurance Portability and Accountability Act – Data privacy standards.
- **UFD:** User Flow Diagram – The provided diagrams visualizing device data flow, app navigation, and system operations (e.g., from sensor reading to alert escalation).
- **IoT:** Internet of Things – Refers to sensor-device-cloud connectivity.
- **Time Series Data:** Sequential data points from sensors analyzed for patterns.
- Additional terms in Appendix A: Glossary.

1.6 References

- IEEE Std 830-1998: Recommended Practice for Software Requirements Specifications.
- User Flow Diagrams: Provided images depicting system workflows, including continuous monitoring loops, branching for anomaly detection, and alert paths.
- Industry Standards: ISO/IEC 27001:2013 for information security, WCAG 2.1 for accessibility, HIPAA for health data.

1.7 Overview

Section 2 offers a high-level product description. Section 3 details requirements. Section 4 includes supporting materials like assumptions and appendices.

2. Overall Description

2.1 Product Perspective

NeuroGuard addresses the critical need for proactive monitoring of epilepsy patients, where unpredictable seizures can lead to injury or worse. As a smart wearable (handband or headband), it continuously tracks neural, vital, and motion data, using AI to predict seizures 1-5 minutes ahead, enabling timely intervention. The system combines AI for prediction, IoT for connectivity, embedded systems for hardware, and mobile/cloud for user interaction, making it a comprehensive solution with significant impact. The architecture allows for future expansions to other conditions (e.g., stroke risk via consciousness loss and neural changes, heart arrhythmias like tachycardia/bradycardia, sleep apnea via breathing pauses, fall detection for elderly, panic attacks/stress via HR and motion, Parkinson's via tremors, and athlete performance via oxygen and activity) using the same hardware.

2.2 Product Features

High-level features derived from the UFD and system operation:

- Continuous sensor data acquisition and transmission (BioAmp for EEG/EMG, MAX30102 for HR/SpO2, MPU6050 for motion).
- AI-based time series analysis for seizure prediction/detection.
- Multi-modal alerts: Device sound/vibration, mobile push, cloud dashboard updates, emergency SMS/calls via SIM800L.
- Fallback communications for offline scenarios.
- Cross-platform mobile app (Flutter) and cloud dashboard (Firebase-hosted) for real-time status, alerts, and reports.
- Future-proof design for expansions without hardware changes.

2.3 User Classes and Characteristics

- **Patients/Wearers:** Individuals with epilepsy or similar conditions; expect comfortable, non-intrusive wearables with simple setup. [Characteristics: Basic tech knowledge, focus on passive monitoring and safety.]
- **Caregivers/Family/Doctors:** Monitors patient data; require real-time alerts and historical insights. [Characteristics: Multiple patients, advanced analytics via app/dashboard.]

- **Administrators:** System maintainers; handle configurations and updates. [Characteristics: Technical expertise, access to backend.]
- **Trial Users:** Potential users; limited access to demo features. [Characteristics: Exploratory use.]

2.4 Operating Environment

- **Hardware Platforms:** Wearable device (handband/headband) with sensors; Mobile (iOS/Android devices with 2GB+ RAM); Desktops for cloud dashboard.
- **Software Platforms:** Flutter for app (Android 11+, iOS 15+); Arduino/ESP32 firmware; Web browsers (Chrome 100+, Firefox 95+) for dashboard.
- **Network:** Wi-Fi/Bluetooth for data sync; GSM for SIM800L; supports offline buffering.
- **Databases:** Firebase Realtime Database for real-time syncing; Firestore for historical data.
- **Cloud Infrastructure:** Firebase for backend, with auto-scaling and analytics.

2.5 Design and Implementation Constraints

- **Programming Languages:** App: Dart with Flutter v3.0+. Firmware: Arduino C++ for prototyping, MicroPython/ESP-IDF for ESP32. Backend: Node.js or Python for Firebase functions.
- **Frameworks:** Flutter for UI; Firebase SDK for backend integration; TensorFlow Lite for edge AI.
- **Standards Compliance:** RESTful APIs, OAuth 2.0 for auth, Bluetooth LE for device comms.
- **Constraints:** HIPAA-compliant data handling; low-power firmware for battery life; comfortable wearable form factor; modular code for expansions.

2.6 User Documentation

- In-app tutorials for device setup, wearing instructions (handband/headband), and app usage.
- Online knowledge base with sensor integration guides, troubleshooting, and expansion ideas.
- API documentation for custom extensions (e.g., Swagger for Firebase endpoints).
- User manuals in PDF format for hardware assembly and software installation.

2.7 Assumptions and Dependencies

- **Assumptions:** Users provide consent for health data processing; sensors achieve accurate readings in testing; AI models predict with high accuracy based on trained datasets; stable network for non-emergency features.
- **Dependencies:** Firebase services; third-party libs like Flutter Blue for Bluetooth, Twilio for extended SMS if needed; external datasets for AI training.

3. Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

- Responsive Flutter UI with adaptive layouts for mobile and web-based cloud dashboard.
- Accessibility: ARIA labels, voice-over support, contrast ratios per WCAG 2.1 AA.
- Example: Dashboard with real-time graphs for HR, SpO2, EEG, motion, and prediction indicators; touch-friendly controls for wearable adjustments.

3.1.2 Hardware Interfaces

- Sensor connections: BioAmp (analog input for neural signals), MAX30102 (I2C for HR/SpO2), MPU6050 (I2C for acceleration/gyro), SIM800L (UART for SMS/calls).
- Device-app comms: Bluetooth LE via ESP32; wearable form factor (handband/headband) for comfort; USB for charging and debugging.

3.1.3 Software Interfaces

- APIs: Firebase REST for data sync (/api/data), alert triggers; GraphQL for complex queries.
- Data Format: JSON for sensor payloads (e.g., {"timestamp": "unix", "eeg": [array], "hr": 80, "spO2": 98, "accel": [x,y,z]}); Protobuf for efficient firmware transmission.

3.1.4 Communications Interfaces

- Protocols: HTTPS/TLS 1.3, Bluetooth 5.0, GSM for SIM800L.
- Integration: Firebase push notifications, SMS/calls via SIM800L for emergencies; WebSockets for real-time dashboard updates.

3.2 Functional Requirements

Requirements are prioritized and traceable to UFD elements. Each includes verification method (e.g., unit test, integration test).

3.2.1 User Authentication and Onboarding [Traceable to UFD: Start/Login Branch]

- **System shall support email/password signup with validation (email format, password strength: 8+ chars, mix case/numbers/symbols).** *(REQ-FUNC-1.1 [High])*
  Verification: Functional test.
- **Implement 2FA via authenticator apps or SMS.** *(REQ-FUNC-1.2 [High])*
  Verification: Security audit.
- **Social login (Google, Apple) with OAuth.** *(REQ-FUNC-1.3 [Medium])*
  Verification: Integration test.
- **Password recovery via email link, expiring in 1 hour.** *(REQ-FUNC-1.4 [High])*
  Verification: End-to-end test.
- **Device pairing wizard for wearable via Bluetooth, including form factor selection (handband/headband) and calibration.** *(REQ-FUNC-1.5 [Low])*
  Verification: Usability test.
- **Role assignment during onboarding (patient, caregiver, admin).** *(REQ-FUNC-1.6 [Medium])*
  Verification: Functional test.

3.2.2 Dashboard and Navigation [Traceable to UFD: Post-Login Dashboard]

- **Display real-time vitals (EEG/EMG, HR, SpO2, motion) and seizure prediction score in mobile app and cloud dashboard.** *(REQ-FUNC-2.1 [High])*
  Verification: UI test.

- **Widgets for trends, battery status, connectivity; customizable for caregivers (e.g., multi-patient views).** *(REQ-FUNC-2.2 [Medium])*
  Verification: Integration test.
- **Navigation to sections: Monitor, Alerts, Reports, Settings, Expansions (future modules).** *(REQ-FUNC-2.3 [High])*
  Verification: Functional test.
- **Dark mode and font size adjustments for accessibility.** *(REQ-FUNC-2.4 [Low])*
  Verification: Usability test.

3.2.3 Data Acquisition and Monitoring [Traceable to UFD: Sensor Data Flow]

- **Continuous recording from sensors at configurable rates (e.g., 100Hz for EEG, 1Hz for HR); store time series data locally and sync to Firebase.** *(REQ-FUNC-3.1 [High])*
  Verification: Hardware test.
- **AI/ML analysis (e.g., LSTM or CNN models on edge/cloud) on time series for seizure prediction (1-5 min window) and detection of abnormal patterns.** *(REQ-FUNC-3.2 [High])*
  Verification: ML validation with accuracy >90%.
- **Data buffering offline (up to 1 hour); sync via ESP32 when connected.** *(REQ-FUNC-3.3 [Medium])*
  Verification: Integration test.
- **Firmware updates OTA for Arduino/ESP32, with rollback capability.** *(REQ-FUNC-3.4 [High])*
  Verification: System test.
- **Modular support for future expansions (e.g., stroke detection via EEG changes, sleep apnea via SpO2 drops).** *(REQ-FUNC-3.5 [Low])*
  Verification: Code review.
- **Data filtering to reduce noise (e.g., Kalman filter for motion data).** *(REQ-FUNC-3.6 [Medium])*
  Verification: Unit test.

3.2.4 Alerting and Response [Traceable to UFD: Alert Branch]

- **Upon anomaly (seizure onset, fall, vital deviation): Trigger device sound/vibration, push notifications to app/dashboard, SMS/call via SIM800L if offline.** *(REQ-FUNC-4.1 [High])*
  Verification: Mobile test.
- **Alert log with details (timestamp, readings, location if GPS enabled); actions: Acknowledge, Emergency Call, Share with Doctor.** *(REQ-FUNC-4.2*

[High])
Verification: Functional test.

- **Escalation to multiple caregivers/doctors with priority levels.** *(REQ-FUNC-4.3 [Medium])*
  Verification: Integration test.
- **Prediction alerts for pre-seizure warnings, with confidence scores.** *(REQ-FUNC-4.4 [High])*
  Verification: End-to-end test.
- **Custom alert tones and vibration patterns.** *(REQ-FUNC-4.5 [Low])*
  Verification: Usability test.

3.2.5 Reporting and Analytics [Traceable to UFD: Report Generation]

- **Export reports in PDF/CSV/Excel with charts (e.g., daily trends, seizure frequency over weeks).** *(REQ-FUNC-5.1 [Medium])*
  Verification: Functional test.
- **Analytics: Prediction accuracy, health scores, anomaly correlations; shareable via dashboard.** *(REQ-FUNC-5.2 [High])*
  Verification: Data test.
- **Scheduled reports via email (daily/weekly); insights for expansions (e.g., stress levels from HR variability).** *(REQ-FUNC-5.3 [Low])*
  Verification: Integration test.
- **Historical data visualization with zoom and filter options.** *(REQ-FUNC-5.4 [Medium])*
  Verification: UI test.

3.2.6 Settings and Administration [Traceable to UFD: Settings Branch]

- **Settings: Alert thresholds, language (EN, AR, others), notification prefs, expansion toggles.** *(REQ-FUNC-6.1 [Medium])*
  Verification: Functional test.
- **Profile management: Add caregivers, update medical history, delete data (HIPAA compliant).** *(REQ-FUNC-6.2 [High])*
  Verification: Security audit.
- **Admin mode in dashboard: Multi-patient monitoring, data sharing, user roles (RBAC).** *(REQ-FUNC-6.3 [High])*
  Verification: Integration test.
- **Logout and device unpairing, with session invalidation.** *(REQ-FUNC-6.4 [Medium])*
  Verification: End-to-end test.
- **Backup and restore user settings.** *(REQ-FUNC-6.5 [Low])*
  Verification: System test.

3.3 Use Cases

Use Case 1: Continuous Monitoring and Prediction

- **Actors:** Patient, System.
- **Preconditions:** Device worn, app connected, sensors calibrated.
- **Flow:** Sensors record data > Firmware preprocesses > Sync to cloud > AI analyzes time series > Predict/detect anomaly > Trigger alerts.
- **Postconditions:** Alerts sent if anomaly; data logged in Firebase.
- **Exceptions:** Offline > Buffer data and use SIM800L for critical alerts; Low battery > Notify user.

Use Case 2: Caregiver Response

- **Actors:** Caregiver.
- **Preconditions:** Alert triggered.
- **Flow:** Receive alert via app/dashboard > View real-time details and history > Respond (e.g., call patient, mark as resolved).
- **Postconditions:** Alert status updated; log response time.
- **Exceptions:** No response within 2 min > Escalate to emergency contacts.

Use Case 3: Device Setup and Onboarding

- **Actors:** Patient/Caregiver.
- **Preconditions:** App installed, device assembled.
- **Flow:** Open app > Signup/Login > Pair device via Bluetooth > Calibrate sensors > Set preferences > Start monitoring.
- **Postconditions:** System active; initial baseline data collected.
- **Exceptions:** Pairing fails > Provide troubleshooting guide; Invalid credentials > Retry or recover.

Use Case 4: Future Expansion Activation

- **Actors:** Admin/User.
- **Preconditions:** Core system operational.
- **Flow:** Navigate to Settings > Enable module (e.g., fall detection) > Recalibrate AI with new thresholds > Monitor new metrics.
- **Postconditions:** Expanded features active; reports updated.
- **Exceptions:** Incompatible data > Notify user for sensor check.

Additional use cases in Appendix C.

3.4 Non-Functional Requirements

3.4.1 Performance Requirements

- **Data processing latency <1s for real-time predictions/alerts on edge device.** *(REQ-PERF-1 [High])*
- **Handle 100+ data points/sec with <5% error rate.** *(REQ-PERF-2 [High])*
- **Device battery life >8 hours continuous monitoring; app <5% battery drain per hour.** *(REQ-PERF-3 [Medium])*
- **Cloud response time <500ms for dashboard queries.** *(REQ-PERF-4 [High])*

3.4.2 Safety Requirements

- **Minimize false negatives for seizures (<5%); redundancy in alerts and predictions.** *(REQ-SAFE-1 [High])*
- **Fail-safe mode: Default to emergency alerts on system failure.** *(REQ-SAFE-2 [Medium])*

3.4.3 Security Requirements

- **Health data encryption (AES-256 at rest/transit); HIPAA/GDPR compliance.** *(REQ-SEC-1 [High])*
- **Secure Bluetooth pairing with PIN; Firebase auth with JWT.** *(REQ-SEC-2 [High])*
- **Audit logs for data access and modifications.** *(REQ-SEC-3 [Medium])*
- **Protection against tampering (e.g., firmware signing).** *(REQ-SEC-4 [High])*

3.4.4 Software Quality Attributes

- **Reliability:** MTBF >1000 hours; auto-recovery from crashes.
- **Availability:** 99.99% for Firebase backend; offline support for critical functions.
- **Maintainability:** Modular code; 80% unit test coverage; documented APIs.
- **Portability:** Cross-platform via Flutter; firmware portable to similar MCUs.
- **Usability:** Intuitive setup; comfortable wearable; learnability <5min; satisfaction >4.5/5 in user testing.
- **Scalability:** Support 10,000+ users; auto-scale cloud resources.

3.4.5 Business Rules

- **Data retention:** 30 days default, user-configurable up to 1 year.
- **Expansions:** Toggleable modules for future features without hardware changes.
- **Subscription model:** Free basic monitoring, premium for advanced analytics and expansions.

3.5 Other Requirements

- **Installation:** App via Google Play/App Store; firmware flashing via Arduino IDE; wearable assembly guides.
- **Licensing:** Proprietary software with open hardware schematics for prototyping.
- **Legal:** Compliance with GDPR/HIPAA; ethical AI guidelines for health predictions; liability disclaimers for alerts.
- **Internationalization:** Support for multiple languages (EN, AR initial); right-to-left UI.

4. Supporting Information

4.1 Assumptions, Dependencies, and Constraints

Dependency on Firebase for backend/database (optimal for real-time IoT with Flutter). Constraint: Wearable must be lightweight (<50g) for handband/headband use; power consumption <100mW idle.

Appendix A: Glossary

- **Anomaly:** Abnormal readings (e.g., EEG spikes for seizures).
- **RBAC:** Role-Based Access Control – For user permissions.
- **LSTM:** Long Short-Term Memory – AI model for time series prediction.
- **CNN:** Convolutional Neural Network – For signal pattern detection.
- **OTA:** Over-The-Air – For firmware updates.

Appendix B: Traceability Matrix

Requirement ID	UFD Element	Use Case	Test Case ID
REQ-FUNC-1.1	Login	UC3	TC-AUTH-01
REQ-FUNC-1.2	2FA Setup	UC3	TC-AUTH-02
REQ-FUNC-2.1	Dashboard Display	UC1	TC-DASH-01
REQ-FUNC-3.1	Sensor Recording	UC1	TC-DATA-01
REQ-FUNC-3.2	AI Analysis	UC1	TC-AI-01
REQ-FUNC-4.1	Alert Trigger	UC2	TC-ALERT-01
REQ-FUNC-5.1	Report Export	UC2	TC-REP-01
REQ-FUNC-6.1	Settings Config	UC4	TC-SET-01
REQ-PERF-1	Processing Latency	UC1	TC-PERF-01
REQ-SEC-1	Data Encryption	UC3	TC-SEC-01

Appendix C: Additional Use Cases

- **Use Case 5: Firmware Update** – Actors: Admin. Flow: Check for updates > Download OTA > Install > Verify. Exceptions: Interrupted > Rollback.
- **Use Case 6: Data Export Compliance** – Actors: User. Flow: Request data export > Generate GDPR-compliant file > Download. Exceptions: Large data > Compress.

Appendix D: Data Dictionary

- **SensorData Table:** Fields: ID (PK, int), Timestamp (datetime), EEG (array[float]), HR (int), SpO2 (float), Accel_X (float), Accel_Y (float), Accel_Z (float), Gyro_X (float), Gyro_Y (float), Gyro_Z (float), UserID (FK, int), AnomalyScore (float).
- **UserProfile Table:** Fields: ID (PK, int), Email (string, unique), PasswordHash (string), Role (enum: patient, caregiver, admin), 2FAEnabled (bool), MedicalHistory (text).
- **AlertLog Table:** Fields: ID (PK, int), Timestamp (datetime), Type (enum: prediction, detection, fall), Details (json), Status (enum: pending, acknowledged, resolved), UserID (FK, int).
- **Settings Table:** Fields: ID (PK, int), UserID (FK, int), AlertThresholds (json), Language (string), ExpansionModules (array[string]).

This SRS provides a comprehensive, professional foundation for NeuroGuard's development. For questions or revisions, contact the product team.