

**TOBB ETU**

**Economy & Technology University**

**BIL 495 / YAP 495**

**Software Requirements Specification (SRS)**

***Reference:*** *IEEE 830-1998 / ISO/IEC/IEEE 29148:2018*

## Document Control Table

| Document Title | Project Closure Report |
| --- | --- |
| Document ID | (To be assigned) |
| Prepared By (Author) | manifetch |
| Reviewed By |  |
| Approved By |  |
| Preparation Date | 28.10.2025 |
| Approval Date |  |
| Version / Revision | 1.0 |
| Confidentiality Level | Internal / Restricted / Public |
| Reference Standards | IEEE 15288, IEEE 12207, INCOSE SE Handbook v5 |

## Change Record (Revision History Table)

| Revision | Date | Prepared By | Reviewed/Approved By | Description of Change |
| --- | --- | --- | --- | --- |
| 1.0 | 28.10.25 | manifetch | (Approver) | Initial Release |
| 1.1 |  |  |  | Minor editorial updates |
| 1.2 |  |  |  | Added stakeholder feedback section |

[1. Introduction](#_heading=)

[1.1 Purpose](#_heading=)

[1.2 Scope](#_heading=)

[1.3 Definitions, Acronyms, Abbreviations](#_heading=)

[1.4 References](#_heading=)

[1.5 Overview](#_heading=)

[2. Overall Description](#_heading=)

[2.1 Product Perspective](#_heading=)

[2.2 Product Functions](#_heading=)

[2.3 User Characteristics](#_heading=)

[2.4 Constraints](#_heading=)

[2.5 Assumptions and Dependencies](#_heading=)

[3. Specific Requirements](#_heading=)

[3.1 Functional Requirements](#_heading=)

[3.2 External Interface Requirements](#_heading=)

[3.3 Performance Requirements](#_heading=)

[3.4 Design Constraints](#_heading=)

[3.5 Software System Attributes](#_heading=)

[3.6 Other Requirements](#_heading=)

[4. Appendices](#_heading=)

[4. References](#_heading=h.a6qvajnlk889)

## 1. Introduction

## Purpose

This document defines the functional and non-functional requirements for the development of a web-based NICU Clinical Dashboard integrating real-time signal monitoring and AI-assisted diagnosis.  
It serves as a formal reference for developers, academic supervisors, and the industrial partner (Ertunç Özcan Import & Representation) to ensure a shared understanding of system expectations, design boundaries, and performance goals.

## 1.2 Scope

The NICU Clinical Dashboard aims to provide a secure and scalable solution for monitoring neonatal vital signs (ECG, SpO₂, respiration rate) through IoMT-integrated devices.  
The system supports:

* Real-time physiological signal visualization
* AI-based diagnostic alerts for respiratory and cardiac risks
* Automated report generation (PDF)
* Role-based access (doctor, nurse, administrator)
* Scalable backend architecture for multi-user access

The prototype version will support 5–10 concurrent users, with long-term scalability up to 50 users within a single NICU unit.

## 1.3 Definitions, Acronyms, Abbreviations

| **Term** | **Definition** |
| --- | --- |
| NICU | Neonatal Intensive Care Unit |
| IoMT | Internet of Medical Things |
| ECG | Electrocardiogram |
| SpO*₂* | Oxygen saturation level |
| AI | Artificial Intelligence |
| API | Application Programming Interface |
| HL7 / FHIR | Health data exchange standards |
| HIPAA | Health Insurance Portability and Accountability Act |
| GDPR | General Data Protection Regulation |

## 1.4 References

See Section 5 for full reference list.

## 1.5 Overview

This document is structured as follows:

* Section 2: General system description, context, and user environment.
* Section 3: Specific functional and non-functional requirements.
* Section 4: Appendices (traceability and requirement mappings).
* Section 5: References to standards and supporting literature.

## 2. Overall Description

## 2.1 Product Perspective

The NICU dashboard operates as a standalone clinical web platform interfacing with:

* IoMT devices or synthetic data simulators (e.g., BioSPPy, NeuroKit2)
* A backend server managing storage and analytics (Flask/Node.js)
* An AI module providing diagnostic predictions
* A secure frontend dashboard (React-based) for end-users
* It functions within hospital networks and supports on-premise deployment.

## 2.2 Product Functions

* Display real-time neonatal physiological signals (ECG, SpO₂, respiration).
* Generate AI-driven diagnostic results for distress detection.
* Produce and export clinical reports in PDF format.
* Trigger alert mechanisms (visual/audio) when thresholds are exceeded.
* Manage users with role-based access and secure login.
* Store and retrieve time-series data for trend analysis.

## 2.3 User Characteristics

| **User Type** | **Role** | **Technical Expertise** | **Goals** |
| --- | --- | --- | --- |
| Clinicians (Doctors) | Interpret AI results and monitor patients | Intermediate | Diagnose and act on risk alerts |
| Nurses | Monitor signals and respond to alerts | Basic | Real-time patient follow-up |
| Administrators | Manage user roles and records | Intermediate | Ensure data access control |
| Biomedical Engineers | Maintain device connectivity | Advanced | Ensure stable IoMT integration |
| Researchers | Analyze neonatal data | Advanced | Study health trends and model outputs |

### 2.4 Constraints

## This section outlines the key hardware, software, legal, and policy constraints that govern the development and deployment of the NICU Clinical Dashboard. All identified limitations are documented in accordance with the TOBB ETU Bitirme Projeleri Guideline (2025) and detailed within the submitted constraint assessment forms.

#### Hardware Constraints

* All NICU-connected IoMT devices shall operate under hospital electrical safety standards (IEC 60601-1). Leakage current ≤ 100 µA.
* All sensing modules must operate reliably between 18–26°C and humidity 40–60%, as per NICU standards.
* IoMT sensor modules attached to incubators shall weigh ≤ 200 g each to avoid structural imbalance.

#### Software Constraints

* Real-time data visualization latency shall not exceed 1 second between acquisition and display.
* All patient data must be encrypted at rest and in transit (AES-256 + HTTPS/TLS 1.3).
* System uptime shall be at least 99% during testing phase.
* AI inference for neonatal distress prediction must execute within ≤ 3 seconds per sample.
* Support for minimum 10 concurrent device streams via MQTT protocol.
* Incoming data packets must be verified using CRC or checksum validation before storage.
* All UI alert notifications must appear within ≤ 2 seconds and be color-coded for severity.
* AI model memory footprint shall not exceed 2 GB RAM during runtime.
* Software design and documentation must adhere to IEEE 1058 and ISO/IEC 12207 lifecycle standards.
* Use of real patient data requires anonymization and ethics board approval.

#### Temporal and Budgetary Constraints

## The total project duration is limited to 12 weeks.

## Only open-source tools and libraries will be used due to a zero-cost policy.

## Integration with physical medical devices will be postponed to the next project phase.

#### Document References

## All identified constraints and compliance validations are recorded in the following forms:

## Universal SW Constraint Tracking Form– [SW Constraints](https://docs.google.com/document/d/1hbv4x8GEEf3CDCwbx0djJnJPweZG8dzP/edit?usp=sharing&ouid=108700557094367383127&rtpof=true&sd=true)

## Cross-Discipline Constraint Tracking Form– [Cross Discipline Constraints](https://docs.google.com/document/d/1cvaduWG50F6i655-DW3G7gZVao6p0lcz/edit?usp=sharing&ouid=108700557094367383127&rtpof=true&sd=true)

## Both forms are committed to the project repository and uploaded to the shared Google Drive as part of the project documentation package.

## 2.5 Assumptions and Dependencies

* Data will be obtained from open-access PhysioNet databases (PICS-DB, MIMIC-IV) or generated synthetically.
* AI models will initially operate on pre-recorded or simulated data.
* Integration with real IoMT devices will occur in future iterations.
* Internet connectivity is assumed for real-time monitoring in demo mode.
* Institutional collaboration (Ertunç Özcan) will provide domain feedback and device standards.

## 3. Specific Requirements

## 3.1 Functional Requirements

| **ID** | **Requirement Description** |
| --- | --- |
| F-1 | The system shall display live physiological signals (ECG, SpO₂, respiration) in graphical form. |
| F-2 | The system shall visualize AI-based diagnostic results on patient panels. |
| F-3 | The system shall implement role-based access control for users (doctor, nurse, admin). |
| F-4 | The system shall allow creation and update of patient records. |
| F-5 | The system shall generate alerts (visual/audio) when signal thresholds are exceeded. |
| F-6 | The system shall allow exporting patient reports as PDF files. |
| F-7 | The system shall provide time-series trend visualization for past data. |

## 3.2 External Interface Requirements

| **Interface Type** | **Description** |
| --- | --- |
| User Interface | Web dashboard accessible via modern browsers (Chrome, Edge, Firefox). |
| Hardware Interface | Optional IoMT device integration via MQTT or simulated data streams. |
| Software Interface | RESTful API between frontend, backend, and AI module. |
| Communication Interface | Secure HTTPS and WebSocket protocols for real-time data. |
| Database Interface | PostgreSQL or MongoDB for patient signal and log data. |

## 3.3 Performance Requirements

| **Metric** | **Target** |
| --- | --- |
| Latency | Signal update delay ≤ 1 second |
| AI Inference Time | ≤ 2 seconds per evaluation |
| Concurrent Users | Minimum 10 concurrent users |
| Uptime | ≥ 95% during prototype testing |
| Data Throughput | Handle continuous data stream at 1Hz frequency |

## 3.4 Design Constraints

* Must adhere to IEEE 1058 (Project Management) and ISO/IEC/IEEE 29148 (Requirements) standards.
* Implement modular architecture separating data, AI, and UI layers.
* Use Python (Flask, TensorFlow) and JavaScript (React) as development languages.
* Follow HIPAA/GDPR principles for patient data handling.

## 3.5 Software System Attributes

This section evaluates the software quality of the system in accordance with the ISO/IEC 25010:2011 quality model. The following subsections describe the main quality attributes of the proposed system and how each will be achieved and maintained throughout development and deployment.

**1. Reliability**

* The system includes an automatic reconnection mechanism to handle interruptions in NICU data streams.
* Fault-tolerant processing minimizes false alarms during model training and inference.
* Continuous data validation ensures early detection of sensor errors and corrupted data.

**2. Security**

* All patient data are anonymized in compliance with KVKK and HIPAA standards.
* Database access is secured using JWT-based authentication mechanisms.
* Encrypted communication is enforced through the HTTPS protocol to ensure data confidentiality.

**3. Performance Efficiency**

* The prediction latency of the model is optimized to operate within milliseconds.
* GPU acceleration using PyTorch/TensorFlow ensures efficient model inference.
* Data preprocessing pipelines are optimized to minimize redundant data transfer and computation.

**4. Usability**

* The clinical user interface is designed with simplicity and clarity, incorporating color-coded alerts.
* Minimal training is required for NICU staff due to intuitive design principles.
* Visualization tools are aligned with medical staff data interpretation habits for faster decision-making.

**5. Maintainability**

* The codebase follows a modular architecture separating data preprocessing, model training, and inference modules.
* Version control is maintained through Git for traceability and collaborative development.
* Structured logging and debugging systems are implemented for easier maintenance and error tracking.

**6. Portability**

* The system is containerized using Docker for consistent deployment across different environments.
* It operates seamlessly on both Linux and Windows platforms.
* The database dependency is limited to PostgreSQL/PostGIS, enabling easy migration and integration.

**7. Compliance**

* The system adheres to ISO/IEC/IEEE 29148:2018 and ISO/IEC 25010:2011 standards.
* Data privacy, security, and legal compliance are ensured under the “Realistic Constraints and Conditions” principles.
* All healthcare data components are designed in accordance with ethical guidelines and hospital IT policies.

## Other Requirements

* Future integration with hospital information systems (HIS) through FHIR API.
* Addition of multilingual interface (English/Turkish).
* Continuous feedback loop from clinicians for model improvement.

## 4. Appendices

**4.1 Requirement Identification Schema**

| **Type** | **Prefix** | **Example** | **Description** |
| --- | --- | --- | --- |
| Functional Requirement | F-XXX | F-001 | Defines a specific capability of the system. |
| Non-Functional / Quality Attribute | Q-XXX | Q-002 | Describes measurable system qualities such as reliability, security, or performance. |
| Constraint | C-XXX | C-003 | Defines technical or environmental limits such as hardware, legal, or timing constraints. |
| Test Case | T-XXX | T-001 | Specifies how each requirement will be verified. |

**4.2 Traceability Matrix**

| **Requirement ID** | **Description** | **Related Constraint** | **Quality Attribute (ISO/IEC 25010)** | **Verification Method** |
| --- | --- | --- | --- | --- |
| F-001 | The system shall display live physiological signals (ECG, SpO₂, respiration) in real-time. | C-001 (Network latency ≤ 1s) | Performance Efficiency — Time Behaviour | Functional testing with simulated NICU stream |
| F-002 | The AI module shall classify respiratory and cardiac distress levels automatically. | C-002 (GPU memory ≤ 4GB) | Functional Suitability — Accuracy | Model validation and benchmark test |
| F-003 | The system shall provide user authentication and role-based access control. | C-003 (JWT-based security) | Security — Integrity, Confidentiality | Penetration testing, role-based access tests |
| F-004 | The dashboard shall generate clinical reports in PDF format. | C-004 (Export ≤ 3s) | Usability — Operability | Functional testing |
| Q-001 | The system shall maintain uptime ≥ 95% during operation. | C-005 (Server redundancy required) | Reliability — Availability | Stress testing and uptime monitoring |
| Q-002 | All data communication shall use HTTPS and AES-256 encryption. | C-006 (GDPR/KVKK compliance) | Security — Confidentiality | Security audit and data packet inspection |
| Q-003 | The dashboard shall respond to user input within 1 second. | C-007 (CPU load ≤ 70%) | Performance Efficiency — Resource Utilization | Load testing |
| Q-004 | The interface shall support both English and Turkish languages. | C-008 (Localization constraints) | Usability — Accessibility | UI testing |
| Q-005 | The system shall operate on both Windows and Linux environments using Docker. | C-009 (Cross-platform deployment) | Portability — Adaptability | Deployment validation test |

### 4.3 Traceability Chain Example

**Stakeholder Need → F-002 → C-002 → Q-001 → T-004**

**Explanation:**NICU clinicians require real-time AI-assisted distress classification (F-002).  
Hardware limitations (C-002) define acceptable model size and memory use.  
Reliability (Q-001) ensures stable model inference over long operation times.  
Verification occurs through Test Case T-004, evaluating inference time and consistency.

### 

### 4.4 Quality Attribute Mapping

| **ISO/IEC 25010 Quality Factor** | **System Feature or Module** | **Requirement IDs** | **Measurement Metric** |
| --- | --- | --- | --- |
| Performance Efficiency | Real-time data stream visualization | F-001, Q-003 | Latency ≤ 1s |
| Reliability | Fault-tolerant data handling | Q-001 | Uptime ≥ 95% |
| Security | Encrypted data transmission | F-003, Q-002 | AES-256, HTTPS |
| Usability | Clinician-friendly dashboard UI | F-004, Q-004 | User task completion time |
| Maintainability | Modular architecture, version control | — | Mean Time to Repair (MTTR) |
| Portability | Docker-based deployment | Q-005 | Multi-platform compatibility |
| Compliance | GDPR, HIPAA, KVKK adherence | C-006 | Compliance audit checklist |

**4.5 References Between Requirements and Test Cases**

| **Requirement ID** | **Linked Test Case ID** | **Test Objective** |
| --- | --- | --- |
| F-001 | T-001 | Verify real-time visualization accuracy |
| F-002 | T-002 | Validate AI inference within target latency |
| F-003 | T-003 | Confirm secure access and user role handling |
| Q-002 | T-004 | Validate data encryption and confidentiality |
| Q-005 | T-005 | Confirm containerized deployment on multiple OS |

### 4.6 Traceability Maintenance

All requirement IDs and test mappings are version-controlled using Git. Each modification triggers an automated update in the traceability table. The document follows the IEEE 29148 traceability principle:  
“Every requirement shall be uniquely identifiable, verifiable, and traceable across all project artifacts.”

## 5. References

*Goldberger, A. L., Amaral, L. A. N., Glass, L., Hausdorff, J. M., Ivanov, P. C., Mark, R. G., Mietus, J. E., Moody, G. B., Peng, C.-K., & Stanley, H. E. (2000). PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. Circulation, 101(23), e215–e220. Retrieved from* [*https://physionet.org/content/*](https://physionet.org/content/)

*International Electrotechnical Commission (IEC). (2015). IEC 62304:2015 — Medical device software — Software life cycle processes. Retrieved from* [*https://webstore.iec.ch/publication/22786*](https://webstore.iec.ch/publication/22786)

*International Organization for Standardization. (2011). ISO/IEC 25010:2011 — Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models. Retrieved from* [*https://www.iso.org/standard/35733.html*](https://www.iso.org/standard/35733.html)

*International Organization for Standardization. (2018). ISO 26262:2018 — Road vehicles — Functional safety. Retrieved from* [*https://www.iso.org/standard/68383.html*](https://www.iso.org/standard/68383.html)

*International Organization for Standardization. (2022). ISO/IEC 27001:2022 — Information security, cybersecurity and privacy protection — Information security management systems. Retrieved from* [*https://www.iso.org/standard/82875.html*](https://www.iso.org/standard/82875.html)

*International Organization for Standardization & IEEE. (2018). ISO/IEC/IEEE 29148:2018 — Systems and software engineering — Life cycle processes — Requirements engineering. Retrieved from* [*https://www.iso.org/standard/72089.html*](https://www.iso.org/standard/72089.html)

*Open Web Application Security Project (OWASP). (2023). OWASP Top Ten Security Risks for Web Applications. Retrieved from* [*https://owasp.org/www-project-top-ten/*](https://owasp.org/www-project-top-ten/)

*PhysioNet. (2023). Neonatal Intensive Care Unit (NICU) vital signs dataset. Retrieved from* [*https://physionet.org/about/database/*](https://physionet.org/about/database/)

*Republic of Türkiye, Personal Data Protection Authority (KVKK). (2016). 6698 sayılı Kişisel Verilerin Korunması Kanunu. Retrieved from* [*https://www.kvkk.gov.tr/Icerik/6649/Kisisel-Verilerin-Korunmasi-Kanunu*](https://www.kvkk.gov.tr/Icerik/6649/Kisisel-Verilerin-Korunmasi-Kanunu)

*TÜBİTAK BİLGEM. (2023). Yapay Zekâ Sistemleri Güvenilirlik Rehberi. Retrieved from* [*https://bilgem.tubitak.gov.tr*](https://bilgem.tubitak.gov.tr)

*U.S. National Institutes of Health (NIH). (2020). Artificial Intelligence in Neonatal Intensive Care Units: Opportunities and Challenges. Retrieved from* [*https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7484551/*](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7484551/)

*World Health Organization (WHO). (2018). Preterm birth — Key facts. Retrieved from* [*https://www.who.int/news-room/fact-sheets/detail/preterm-birth*](https://www.who.int/news-room/fact-sheets/detail/preterm-birth)