



# **TED UNIVERSITY**

## **SENG 491 - SENIOR PROJECT**

### **Intelligent Documentation Assistant for SRS (IDAS)**

#### **High-Level Design Report**

**Fall 2025**

**Team Members:**

İlayda Dim – 16976338092

Yaren Saklavcı – 10040078662

Kaan Aytekin – 10906257460

**Supervisor:** Elif Kurtaran Özbudak

**Jury Members:** Tansel Dökeroğlu - Ali Berkol

<b>Name</b>	<b>Date</b>	<b>Reason For Changes</b>	<b>Version</b>
High-Level Design Report-1.0	24.12.2025	First version of High-Level Design Report	1.0
High-Level Design Report-2.0	28.12.2025	Changes according to feedback gathered from HAVELSAN.	2.0

## Table of Contents

<b>1. Introduction .....</b>	<b>3</b>
<b>1.1. Purpose of the System .....</b>	<b>3</b>
<b>1.2 Design Goals.....</b>	<b>3</b>
<b>1.3 Definitions, Acronyms, and Abbreviations .....</b>	<b>3</b>
<b>1.4 Overview.....</b>	<b>4</b>
<b>2. Current Software Architecture (if any) .....</b>	<b>5</b>
<b>3. Proposed Software Architecture .....</b>	<b>5</b>
<b>3.1 Overview.....</b>	<b>5</b>
<b>3.2 Subsystem Decomposition.....</b>	<b>5</b>
<b>3.2.1 User Interface Subsystem .....</b>	<b>6</b>
<b>3.2.2 Authoring Subsystem .....</b>	<b>6</b>
<b>3.2.3 Review and Analysis Subsystem.....</b>	<b>6</b>
<b>3.2.4 AI-Powered Services Subsystem .....</b>	<b>6</b>
<b>3.2.5 Document Management Subsystem.....</b>	<b>6</b>
<b>3.2.6 Reporting Subsystem.....</b>	<b>7</b>
<b>3.4 Persistent Data Management.....</b>	<b>7</b>
<b>3.5 Access Control and Security .....</b>	<b>8</b>
<b>3.6 Global Software Control .....</b>	<b>8</b>
<b>4. Subsystem Services.....</b>	<b>9</b>
<b>4.1 User Interface Subsystem Services .....</b>	<b>9</b>
<b>4.2 Authoring Subsystem Services .....</b>	<b>10</b>
<b>4.3 Review and Analysis Subsystem Services.....</b>	<b>10</b>
<b>4.4 AI-Assisted Services Subsystem Services .....</b>	<b>10</b>
<b>4.5 Document Management Subsystem Services .....</b>	<b>10</b>
<b>4.6 Access Control and Session Management Services .....</b>	<b>11</b>
<b>5. Glossary.....</b>	<b>11</b>
<b>6. References .....</b>	<b>13</b>

# 1. Introduction

## 1.1. Purpose of the System

IDAS (Intelligent Documentation Assistant for Software Requirements Specification) is a web-based platform designed to assist stakeholders in the analysis and development phases of the software development lifecycle, enabling them to create and review Software Requirements Specification (SRS) documents in accordance with HAVELSAN standards. The system primarily offers two main features: Authoring Mode that supports writing SRS sections adhering to HAVELSAN SRS template, providing AI-powered suggestions such as terminology consistency checks, missing heading detection, and draft requirement statement proposals; and a Review Mode that analyzes uploaded SRS documents by extracting requirement statements, classifying them, and identifying potential quality issues based on an Uncertainty and Quality Checklist. Furthermore, IDAS provides user-friendly visual reports and recommendations for the generated outputs.

## 1.2 Design Goals

**HAVELSAN template compliant SRS support:** Providing an HAVELSAN SRS template and review capabilities to enable users to create and review SRS documents that comply with HAVELSAN standards.

**Assisting AI (recommendative):** Providing AI-powered guidance to users strictly for terminology consistency checks, detection of missing sections, draft requirements suggestions, and quality findings. The AI offers non-binding recommendations only; it does not perform or substitute for formal verification or final approval processes.

**Resilience to defective input:** Maintaining stable operation when documents are incomplete or erroneous, minimizing crashes during processing and analysis.

**Privacy-focused user data processing:** Keeping data related to the documents used confidential and storing uploaded documents only temporarily.

**Web accessibility and browser compatibility:** Presenting the system as a web-based application accessible through major browsers via a browser-based interface.

**.docx-focused review workflow:** Keeping the review mode focused on .docx documents to ensure compatibility with supported document structure and parsing capabilities.

## 1.3 Definitions, Acronyms, and Abbreviations

### AI (Artificial Intelligence)

Human-like reasoning and pattern recognition techniques used to support requirement analysis and recommendation generation within the system.

**AI-Assisted Services**

System services that provide non-binding recommendations, such as draft requirement suggestions, terminology checks, and quality findings, without performing formal verification or final decision-making.

**FR (Functional Requirement)**

A requirement that specifies a function or behavior that the system or software must perform.

**HAVELSAN SRS Template**

Standard Software Requirements Specification document format used by HAVELSAN. It contains required sections, headings, formatting, and content rules to make sure software requirements documents for defense and big software projects are clear, complete, and of high quality.

**IDAS (Intelligent Documentation Assistant for Software Requirements Specification)**

A web-based platform designed to assist users in authoring and reviewing SRS documents in accordance with HAVELSAN standards.

**LLM (Large Language Model)**

A type of artificial intelligence model designed for language-based analysis and text generation, used in IDAS to support recommendation and analysis tasks.

**NFR (Non-Functional Requirement)**

A requirement that specifies quality attributes or constraints of the system, such as usability, privacy, robustness, or accessibility.

**SRS (Software Requirements Specification)**

A structured document that describes the functional and non-functional requirements of a software system.

**1.4 Overview**

This High-Level Design document presents the architecture of the IDAS system. It builds on the requirements and models from earlier Project Design and Analysis reports. The document defines key software components, their responsibilities, interactions, and data processing without low-level implementation details.

The design centers on two system modes: Authoring Mode and Review Mode. These reflect the workflows, functional scope, and constraints found in the analysis. The following sections describe the system's current state, proposed architecture, subsystem decomposition, data management, access control, and global control mechanisms.

This document serves as a practical guide for development teams, providing clear direction by translating requirements into a coherent architectural design that fits the project's scope and constraints.

## **2. Current Software Architecture**

The IDAS system had no existing or legacy software architecture prior to this project. The system is being designed and developed from scratch as part of the senior project, based on the requirements, constraints, and system models identified during the analysis phase.

At this stage, the system is defined solely through conceptual elements, such as functional and non-functional requirements, use cases, use case models, object and class models, and sequence diagrams, as presented in the Project Analysis Report. These elements define what the system is expected to do and how users will interact with it.

This High-Level Design document introduces the proposed IDAS software architecture. Since there was no previous architecture, this document translates the analyzed requirements and system models into a structured design, marking the project's first architectural decisions. This milestone sets a solid foundation for the subsequent design and implementation activities within the project scope.

## **3. Proposed Software Architecture**

### **3.1 Overview**

The proposed IDAS software architecture supports two core system modes—Authoring Mode and Review Mode—within a web-based, modular architecture. The architecture separates user interaction aspects from document processing and AI-powered analysis responsibilities to enhance sustainability and robustness.

The system has a web-based interface that lets users interact with it. The backend handles document processing, analysis steps, and AI services. On the other hand, the architecture prioritizes modularity, allowing the user interface and analysis components to evolve independently. Document processing is session-based. Each review handles only one document at a time. Uploaded documents are stored temporarily to protect privacy. The system presents outputs as recommendations and visual reports. Final interpretation and decision making remains with the user.

This overview outlines the main design, which is explained in more detail later by breaking down subsystems, handling data, setting up access, and managing the system as a whole.

### **3.2 Subsystem Decomposition**

The IDAS system consists of collaborative subsystems, each responsible for specific functions within the overall architecture. These subsystems are defined by the functional scope established during the analysis phase to ensure modularity, sustainability, and clear separation of responsibilities.

### **3.2.1 User Interface Subsystem**

The User Interface subsystem enables interaction via a web browser, offering main screens such as login, registration, dashboard, Authoring Mode, Review Mode, and review report view. Users can navigate system features, add or edit SRS content, upload documents for review, and view analysis results and recommendations.

### **3.2.2 Authoring Subsystem**

The Authoring subsystem helps users create SRS documents that follow HAVELSAN standards. It uses a predefined template and guides users through each section. It provides guidance to support structural completeness, terminology consistency, and requirement formulation.

These suggestions are optional and meant to make the process easier.

### **3.2.3 Review Subsystem**

The Review subsystem works with uploaded SRS documents in Review Mode. It reads .docx files, extracts requirement statements, and sorts them into Functional or Non-Functional categories. It checks requirements using the Uncertainty and Quality Checklist to find any quality issues. The results are then sent to the reporting tools for users to access the Subsystem.

### **3.2.4 AI-Powered Services Subsystem**

The AI-Powered Services subsystem provides intelligence support for both Authoring Mode and Review Mode. It supports functions such as generating draft requirement proposals, identifying terminology inconsistencies, assisting in requirement classification, and contributing to quality issue detection. This subsystem operates in a supportive capacity, providing recommendations rather than final decisions, ensuring that outputs remain advisory and aligned with the defined system constraints.

### **3.2.5 Document Management Subsystem**

The Document Management subsystem manages the lifecycle of SRS documents within the system. It provides document uploads, temporary storage, session-based access, and export operations. To meet privacy requirements, uploaded documents are temporarily stored, processed individually for each review operation, and deleted after the analysis is complete or the session ends. The Session Management subsystem ensures that only registered users can access the system and its functions. In other words, this module manages user authentication, session tracking, and the association of write or review operations with specific user sessions. This subsystem enforces restrictions, such as limiting each review operation to a single document at a time, and supports multi-user access without violating privacy or workflow restrictions.

### **3.2.6 Reporting Subsystem**

The Reporting subsystem creates and shows results from the Review and Analysis subsystem. It gathers classification results and finds quality issues, then displays them in visual and written reports on the user interface. These reports highlight potential problems and offer recommendations to help users make decisions.

## **3.3 Hardware/Software Mapping**

The system is built to run on the web, requires a browser, and handles document processing and analysis on a main server. On the client side, interaction with the system is provided through a standard web browser running on desktop or laptop computers. The browser hosts the user interface, enabling tasks such as text input, document upload, system navigation, and viewing analysis results. No persistent local data storage or document analysis occurs on the client device.

On the server side, the system's main components run in a single location. This setup handles writing and review tasks, works with .docx documents, extracts and sorts requirements, checks quality, and generates reports. All tasks that require significant computing power, such as AI analysis and suggestion generation, are handled on the server.

The server stores documents and other data only for a short time, in accordance with privacy rules. Uploaded documents are kept only while users are using the system and are deleted when user finishes. This helps keep user data safe and reduces the risk of unauthorized access.

This hardware-software pairing allows the system to be accessed by different users via a browser while enabling centralized management of the computational load. This setup lets different users use the system via a browser, while all the heavy computational work is handled in one place. This way, the system can handle many users simultaneously while still keeping data private and reliable.

## **3.4 Persistent Data Management**

The IDAS system is built to minimize data storage and protect user privacy. In accordance with project rules, the system does not retain user-uploaded SRS documents or analysis files for long.

Uploaded SRS documents and any data created while working on or reviewing them are retained only long enough to complete the task. After the related operation is done or the session ends, these documents and related data are deleted and not kept by the system.

The only data the system keeps long-term is the information it needs to work. This includes user account details and a limited session-related information required for logging in, controlling access, and managing tasks. The system does not keep any document content, requirements taken from documents, or detailed analysis results.

This approach to handling data helps meet the system's privacy and security needs by keeping as little data as possible and reducing the risk of unauthorized access. By only saving the most important system data, the system handles data carefully and while maintaining expected system functionality.

### **3.5 Access Control and Security**

The IDAS system protects user data and privacy. Only registered users can log in and perform actions.

User authentication is managed securely to prevent unauthorized access. A user must be authenticated to create SRS documents. Only authenticated users can upload files for review. Only they can view analysis results for their sessions. Anonymous access is not permitted.

Document access and processing are limited to individual user sessions. Only the user who starts an authoring or review operation can access the related SRS documents and results. This session-based control helps prevent unauthorized disclosure of sensitive content.

Security measures protect user-uploaded documents and analysis data during processing. Documents are not shared with third-party services. They are not used for model training. They are kept only temporarily, in line with project constraints. These steps reduce unauthorized access and support to keep data confidential.

### **3.6 Global Software Control**

The IDAS system's software control is user-driven and event-based. All major operations begin with explicit user actions through the web interface. The system does not use autonomous or continuously running background processes.

After authentication, users access a central interface to choose either Authoring Mode or Review Mode. Each mode provides a distinct workflow that is activated when the user selects it. The system then coordinates processing steps in a set sequence and waits for user input or confirmation before continuing.

In Authoring Mode, the system responds to user-driven events such as section editing or a request for suggestions by invoking the relevant support services. In Review Mode, the system initiates a controlled analysis workflow when a user uploads a document and explicitly requests a review. The system processes the document sequentially and generates analysis results only during the requested operation.

In both modes, the system maintains session-level control, isolating each operation within the user's active session. After completing an operation, the system presents the results and waits for further user interaction. This event-driven, session-oriented structure supports controlled task execution, and clear transitions between system states.



## **3.7 Boundary Conditions**

This section explains where and how the IDAS system can be used. These limits are based on findings from analysis and show what the system can and cannot do.

### **3.7.1 Document-Related Boundaries**

The system works with Software Requirements Specification (SRS) documents, but has some limits. Review Mode only works with .docx files and looks at text only. It does not check diagrams, images, or anything that is not text. The system only works with SRS documents written in English. Documents in other languages will not work.

### **3.7.2 Workflow and Processing Boundaries**

The system only works when a user starts an action. In Review Mode, it can check only one document at a time. It cannot view multiple documents at once. The system does not run any background tasks or operate on a schedule. It only works when the user tells it to.

### **3.7.3 Analysis and Output Boundaries**

The system only gives suggestions. IDAS does not officially check or confirm the requirements and does not promise that the SRS documents are correct or complete. Problems such as unclear meaning, untestable parts, repeated sections, or conflicts are flagged as potential issues to help users, but users must make the final decisions.

### **3.7.4 Data Handling and Privacy Boundaries**

Uploaded documents and analysis data are kept only for a short time during each session and are deleted when the work is done. The system does not share documents or analysis data with others and does not use user content to train its models. Needed to be connected to the internet to use the system. Offline operation is not supported.

### **3.7.5 User Access Boundaries**

Only users who have signed up can use the system. People who are not logged in cannot use it, and features are only available to logged-in users. Each session is isolated to prevent unauthorized access to documents and analysis results.

## **4. Subsystem Services**

### **4.1 User Interface Subsystem Services**

The User Interface subsystem focuses exclusively on user interaction with the system and delivers services via a browser-based interface. Its primary services include:

- Enabling users to access core system functionalities through dedicated interfaces for Authoring Mode and Review Mode
- Supporting user input for SRS content creation, editing, and document upload
- Displaying AI-assisted suggestions and analysis results clearly and comprehensibly
- Facilitating navigation across system workflows and visualizing review reports and identified issues

## **4.2 Authoring Subsystem Services**

The Authoring subsystem specializes in supporting users in creating SRS documents that meet HAVELSAN standards. Its services include:

- Providing a predefined SRS template structure based on HAVELSAN standards
- Supporting structural completeness of SRS documents during authoring
- Checking terminology consistency against a predefined glossary
- Offering context-aware draft requirement suggestions to assist users during the writing process

## **4.3 Review and Analysis Subsystem Services**

The Review and Analysis subsystem is dedicated to evaluating and improving the quality of completed SRS documents. Its services include:

- Parsing uploaded .docx SRS documents and extracting requirement statements.
- Classifying extracted requirements as Functional or Non-Functional
- Evaluating requirements using the Uncertainty and Quality Checklist
- Identifying potential quality issues, including ambiguity, untestability, duplication, and conflicts

## **4.4 AI-Assisted Services Subsystem Services**

The AI-Assisted Services subsystem provides intelligent support services used by, but distinct from, the Authoring and Review operations. Its services include:

- Supporting the generation of draft requirement statements
- Facilitating requirement classification and terminology analysis
- Aiding in the identification of potential quality issues
- Generating recommendations instead of final decisions

## **4.5 Document Management Subsystem Services**

The Document Management subsystem provides services for managing SRS documents throughout their lifecycle within the system.

- Managing document uploads and exports, managing temporary storage of documents during active sessions
- Ensuring session-based access to documents and analysis results
- Deleting documents upon completion of the corresponding operation

#### 4.6 Access Control and Session Management Services

The Access Control and Session Management subsystem ensures secure, user-specific system access and session operation. Its services include:

- Overseeing user authentication and session initiation
- Restricting system access exclusively to registered users
- Linking authoring and review operations to specific user sessions
- Enforcing session isolation to prevent unauthorized document access

### 5. Glossary

**AI-Powered Analysis:** Analytical activities that help classify requirements extracted from SRS content and identify potential quality problems.

**Ambiguity:** The situation where a requirement can be interpreted in multiple ways or is not expressed clearly enough.

**Authoring Mode:** The system mode in which the user creates and edits SRS documents using an HAVELSAN-based template, with support for missing heading detection, terminology consistency checks, and draft requirement proposals.

**Authoring Subsystem:** The subsystem that provides the functionalities of Authoring Mode by coordinating template usage, missing heading detection, glossary consistency checks, and proposal generation.

**Boundary Conditions:** Constraints that define the scope within which the system operates and the situations in which it does not operate (e.g., .docx only, English only, text-only processing, no offline support).

**Browser-Based Interface (BBI):** A user interface approach in which system access and interaction are provided through a web browser.

**Classification:** The process of labeling extracted requirement statements as Functional Requirements or Non-Functional Requirements.

**Conflict:** A situation in which one requirement contradicts another such that both cannot be satisfied simultaneously.

**Document Lifecycle:** The complete set of stages a document goes through, including uploading or creation, processing, temporary storage during the session, and deletion upon completion.

**Document Management Subsystem:** The subsystem responsible for document uploading, temporary storage, session-based access, and export operations.

**Event-Driven Control:** A control approach in which system operations are triggered by explicit user actions such as button clicks, document uploads, or requests.

**Global Software Control (GSC):** The mechanism that defines how workflows are initiated and executed across the system; in IDAS, control is user-initiated and session-based.

**IDAS (Intelligent Documentation Assistant for SRS):** A web-based platform that supports the creation of HAVELSAN-compliant SRS documents and the review of .docx SRS files.

**Maintainability:** A design characteristic that enables the system to be updated, modified, and improved with minimal effort over time.

**Missing Heading Detection:** A function that identifies the absence of expected sections or headings in an SRS document based on the HAVELSAN-based template.

**Modularity:** An architectural characteristic in which subsystem responsibilities are separated, allowing components to evolve independently.

**NFR (Non-Functional Requirement):** A type of requirement that defines system quality attributes and constraints such as confidentiality, compliance, and robustness.

**Non-Functional Requirement (NFR):** Specifies how the system should operate, including aspects such as privacy focus, browser compatibility, and robustness.

**Privacy-Focused Processing:** An approach that ensures documents are stored only temporarily, not shared with third parties, and not used for model training.

**Requirement Statement:** A sentence or textual element extracted from an SRS document that is considered to express a requirement.

**Requirement Extraction:** The process of identifying and extracting requirement statements from a .docx SRS document.

**Reporting Subsystem:** The subsystem that presents classification and quality analysis outputs to users in the form of visual and textual reports.

**Resilience / Robustness:** The ability of the system to continue operating without crashing when faced with incomplete or erroneous document inputs.

**Review Mode:** The system mode in which a user uploads an SRS document in .docx format and receives requirement extraction, classification, and quality analysis results.

**Session:** The logical context of a logged-in user's interaction with the system, within which document processing and access to results are managed.

**Session-Based Processing:** An approach in which operations are executed within the user session, with each review operation processing and temporarily storing only one document.

**Subsystem:** An architectural component responsible for a specific area of functionality that collaborates with other system components.

**Software Requirements Specification (SRS):** The document that defines software requirements and serves as the primary input and output of both Authoring and Review Modes in IDAS.

**Terminology Consistency:** The consistent use of terms throughout the document in accordance with the predefined glossary.

**Uncertainty and Quality Checklist:** A set of criteria used in Review Mode to evaluate requirement quality, focusing on uncertainty, testability, repetition, and conflict.

**Untestability :** The condition in which a requirement is not expressed in a verifiable or testable manner.

**User Interface (UI) Subsystem:** The subsystem that enables browser-based interaction, including switching between modes, uploading or editing documents, and viewing reports.

**Web-Based Platform:** A platform approach in which the system is accessed via a web browser and no persistent storage or analysis is performed on the client side.

## 6. References

Dim, İ., Saklavcı, Y., & Aytekin, K. (2025). *Intelligent Documentation Assistant for SRS (IDAS) project proposal* [SENG 491 Project Proposal]. Department of Software Engineering, TED University.

Dim, İ., Saklavcı, Y., & Aytekin, K. (2025). *Intelligent Documentation Assistant for SRS (IDAS) project specifications report* [SENG 491 Project Specifications Report]. Department of Software Engineering, TED University.

Dim, İ., Saklavcı, Y., & Aytekin, K. (2025). *Intelligent Documentation Assistant for SRS (IDAS) analysis report* [SENG 491 Analysis Report]. Department of Software Engineering, TED University.

HAVELSAN A.Ş. (n.d.). *Corporate documentation template*. Internal document.