Software Requirements Specification

For

EduSense

Version 1.0

Prepared by: Hannah Gill, Emerald Group

Old Dominion University | CS 411

11/24/2025

Table of Contents

1. Introduction 2
   1.1 Purpose 2
   1.2 Scope 2
   1.3 Definitions, acronyms, and abbreviations 2
   1.4 References..... 3
   1.5 Overview 3
2. Overall Description 3
   2.1 Product Perspective..... 4
   2.2 Product Functions 4
   2.3 User Classes and Characteristics 4
   2.4 Design and Implementation Constraints 5
   2.5 Assumptions and Dependencies 5
3. Specific Requirements 5
   3.1 External Interface Requirements 6
   3.1.1 User Interfaces 6
   3.1.2 Hardware Interfaces 6
   3.1.3 Software Interfaces 6
   3.1.4 Communication Interfaces 6
   3.2 System Features 7
   3.2.1 Detecting Repeated Sequences 7
   3.2.2 Determining Refactoring Candidates 8
   3.2.3 Outputs 9
   3.3 Performance Requirements 11
   3.4 Design Constraints 11
   3.5 Software System Attributes 11
   3.6 Other Requirements 11

1. Introduction

1.1 Purpose

The purpose of this Software Requirements Specification is to define and describe the requirements for EduSense, a web-based platform designed to promote responsible AI use in education. This document will serve as a formal reference for developers, testers, and stakeholders to ensure that the system's goals, functionality, and design expectations are clearly understood.

1.2 Scope

EduSense is designed to help students and instructors balance AI assistance with independent learning. The system allows instructors to create courses and assignments, set limits on AI assistance, and monitor student engagement. Students can interact with a guided AI chatbot to receive hints or explanations without directly receiving complete solutions. EduSense encourages critical thinking, maintains academic integrity, and provides data into how students use AI.

1.3 Definitions, acronyms, and abbreviations

AI (Artificial Intelligence): The capability of a computer program to perform tasks that normally require human intelligence, such as reasoning or learning (IEEE)

Authentication: The process of verifying the identity of a user or system component before granting access. (IEEE)

Backend: The server-side components of a system that handle data management, logic, and communication with the frontend. (IEEE)

Dashboard: A visual interface that provides users with organized access to tools, reports, and course or assignment information.

EduSense: A web-based educational platform that tracks, limits, and guides AI usage for learning.

Frontend: The portion of the software with which the user interacts directly (IEEE)

Instructor: A user who manages courses, assignments, and AI usage limits within EduSense.

LLM (Large Language Model): An AI model capable of generating human-like responses to text prompts.

Student: A user who completes assignments and interacts with guided AI assistance through the EduSense platform.

Unit Testing: Testing of individual hardware or software units or groups of related units. (IEEE)

1.4 References

Team Emerald. (2025, October 15). Lab 1 – Edusense Product Description.
Retrieved October 28th, 2025, from https://canvas.odu.edu/groups/42222/files

"IEEE Standard Glossary of Software Engineering Terminology," in IEEE Std 610.12-1990 , vol., no., pp.1-84, 31 Dec. 1990, doi: 10.1109/IEEESTD.1990.101064.Abstract: This IEEE Standards product is part of the family on Software Engineering. This standard identifies terms currently in use in the field of Software Engineering. Standard definitions for those terms are established. keywords: {Terminology;Software engineering;Standards;glossary;terminology;dictionary;Software engineering;Definitions},URL: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=159342&isnumber=4148

1.5 Overview

The SRS for EduSense outlines the requirements and functionality of the software. Section 2 provides a general description of EduSense, including its purpose, primary features, user groupsm and design considerations. Section 3 will specify system features, external interfaces, and additional requirements such as performance goals, design constraints, and software attributes.

2. Overall Description

2.1 Product Perspective

EduSense is a web-based application designed to promote responsible AI usage in education. The system integrates a React-based frontend with a Django Rest Framework backend and connects to a Large Language Model API. Through separate dashboards, instructors can create courses, upload assignments, set AI-assistance limits, and review AI-usage. Students will be able to access assignments, receive guided assistance through the AI chatbot, and reflect on their learning progress. EduSense can be accessed from desktops, laptops, or tablets.

2.2 Product Functions

The main features of EduSense include:

- Account Creation and Login: Allows new users to register for an account and existing users to log in securely
- Authentication and Logout: Supports secure logins for instructors and students, also gives the ability to logout when session ends.
- Instructor Dashboard**:** Create and manage courses, add assignments, set AI usage limits.
- Student Dashboard: View courses, access assignments, and interact with AI for guided help and explanation.
- AI Guidance: Provides support without directly giving solutions.
- Usage Analytics: Tracks AI-assistance activity for each student and assignment.
- Report Generator: Generates summaries that help instructors assess engagement and responsible AI use.
- Data Management: Stores course data chat histories, and student progress securely
- Responsive Interface: Allows accessibility and usability across devices.

2.3 User Classes and Characteristics

EduSense is designed for the following user groups:

- **Students**
  - Primary users of EduSense
  - Use the platform to complete assignments and interact with AI assistance for guided help.
  - Requires basic computer skills and internet
  - Can view their AI interaction history and progress for reflection.
- **Instructors**
  - Create and manage courses, assignments and AI usage limits
  - Monitor student engagement and review analytics to assess responsible AI usage.
  - Benefit from tools that help maintain academic integrity and support ethical AI integration in classrooms.

2.4 Design and Implementation Constraints

The development of EduSense is subject to the following constraints:

- **Programming Languages**: The frontend must be implemented using React, and the backend must use Django REST Framework.
- **Platform Compatibility**: The system must operate in modern web browsers, including Chrome, Edge, and Safari.
- **Database**: Must securely store user data, assignments, and AI-usage logs.
- **Security**: Must use encrypted connections and secure authentication protocols to protect user data.
- **LLM integration**: System depends on stable API connectivity fir AI responses.
- **Privacy and Compliance**: Must adhere to university data protection policies and FERPA guidelines.

2.5 Assumptions and Dependencies

EduSense relies on the following assumptions and dependencies:

- Users have reliable internet access and a compatible browser.
- The external LLM API used for AI guidance remains available and functional.
- Instructors set clear AI-assistance limits for each course and assignment.
- The backend server and database are properly configured and hosted.
- The authentication system successfully verifies all users before granting access.
- All third-party services maintain uptime and security.