

Name - Khushi Nitinkumar Patel
PRN - 2020BTECS00037
Batch - B3

Project management Assignment no - 1

To create an online platform where students can fill forms, take exams, view their scores, and instructors can upload questions, set exams, and grade automatically or manually.

2. User Roles:

Admin: Oversee platform, user management, system settings.

Instructor: Create and upload exams, view results, and grade.

Student: Fill forms, Take exams, view results.

Prepare an SRS for the above mentioned project by considering all possible functionalities.

Software Requirements Specification for Online Examination Platform

Version 1.0 approved

Prepared by Khushi Nitinkumar Patel – 2020BTECS00037

Walchand College of Engineering, Sangli

14th August 2023

Table of Contents

Table of Contents	3
Revision History	Error! Bookmark not defined.
1. Introduction	1
1.1 Purpose.....	1
1.2 Document Conventions.....	1
1.3 Intended Audience and Reading Suggestions	1
1.4 Product Scope	1
1.5 References.....	1
2. Overall Description	1
2.1 Product Perspective.....	1
2.2 Product Functions	1
2.3 User Classes and Characteristics	2
2.4 Operating Environment.....	2
2.5 Design and Implementation Constraints	2
2.6 User Documentation	2
2.7 Assumptions and Dependencies	2
3. External Interface Requirements	2
3.1 User Interfaces	2
3.2 Hardware Interfaces	2
3.3 Software Interfaces	3
3.4 Communications Interfaces	3
4. System Features	3
4.1 System Feature 1	3
4.2 System Feature 2 (and so on).....	3
5. Other Nonfunctional Requirements.....	3
5.1 Performance Requirements	3
5.2 Safety Requirements	3
5.3 Security Requirements	3
5.4 Software Quality Attributes	4
5.5 Business Rules	4
6. Other Requirements	4
Appendix A: Glossary.....	Error! Bookmark not defined.
Appendix B: Analysis Models	Error! Bookmark not defined.
Appendix C: To Be Determined List.....	Error! Bookmark not defined.

1. Introduction

1.1 Purpose

The purpose of this document is to provide a comprehensive description of the requirements for the development of an online examination platform. This document will serve as a guide for the design, implementation, testing, and maintenance of the system.

1.2 Document Conventions

This document follows standard formatting conventions for headings, sections, and numbering. Technical terminology follows industry-standard definitions.

1.3 Intended Audience and Reading Suggestions

This document is intended for developers, testers, project managers, and stakeholders involved in the development and deployment of the online examination platform. It is recommended to read the entire document for a complete understanding of the project scope and requirements.

1.4 Product Scope

The online examination platform aims to create a user-friendly interface where students can fill forms, take exams, view their scores, and instructors can upload questions, set exams, and grade automatically or manually.

1.5 References

- Project proposal and requirements provided by stakeholders.

2. Overall Description

2.1 Product Perspective

The online examination platform will be a standalone web application accessible through web browsers. It will interact with databases for storing user data, exam questions, and results. The system will integrate user authentication and role-based access control.

2.2 Product Functions

- Students can fill out forms required for registration.*
- Students can access and take exams assigned to them.*
- Instructors can upload questions, create exams, and assign them to students.*
- Instructors can manually or automatically grade exams.*

- Users (students and instructors) can view their exam results.

2.3 User Classes and Characteristics

- Admin: Manages platform, user accounts, and system settings.*
- Instructor: Creates and uploads exams, views results, and grades exams.*
- Student: Fills out forms, takes exams, and views results.*

2.4 Operating Environment

The platform will be hosted on web servers and accessed via standard web browsers (Chrome, Firefox, etc.). The system will be designed to operate seamlessly across various devices and screen sizes.

2.5 Design and Implementation Constraints

The platform will be developed using modern web development technologies (HTML5, CSS3, JavaScript, etc.). It will be built on a reliable web framework and use a relational database for data storage.

2.6 User Documentation

User manuals and guides will be provided to explain platform usage to students, instructors, and administrators.

2.7 Assumptions and Dependencies

The successful development and operation of the platform depend on a stable internet connection and up-to-date web browsers. The system's security measures assume compliance with best practices for securing user data.

3. External Interface Requirements

3.1 User Interfaces

The user interfaces will be intuitive, user-friendly, and responsive across different devices.

3.2 Hardware Interfaces

The platform will require standard hardware specifications for running web browsers and accessing the internet.

3.3 Software Interfaces

The platform will communicate with a relational database for storing user data, questions, and results.

3.4 Communications Interfaces

The platform will rely on internet connectivity for users to access and interact with the system.

4. System Features

4.1 System Feature 1: User Registration and Authentication

4.1.1 Description and Priority

- *Users can register with the platform using their email addresses and passwords.*
- *Authentication mechanisms will ensure secure access to the platform.*

4.2 System Feature 2: Exam Creation and Management

4.2.1 Description and Priority

- *Instructors can create and upload exam questions.*
- *Instructors can set up exams by selecting questions and defining exam parameters.*

5. Other Nonfunctional Requirements

5.1 Performance Requirements

- *The system should support concurrent access by a large number of users.*
- *The platform should provide quick response times for user interactions.*

5.2 Safety Requirements

- *User data, including personal information and exam results, must be stored securely.*
- *Backup and recovery mechanisms should be in place to prevent data loss.*

5.3 Security Requirements

- *Secure authentication and authorization mechanisms should be implemented.*
- *Encryption protocols should be used for data transmission.*

5.4 Software Quality Attributes

- *The platform should have a user-friendly interface with a consistent design.*
- *The system should be scalable to accommodate growing numbers of users.*

5.5 Business Rules

- *Students can only take exams for courses they are enrolled in.*
- *Instructors can only grade exams they have assigned.*

6. Other Requirements

Appendix A: Glossary

Appendix B: Analysis Models

Appendix C: To Be Determined List (Additional features or requirements that may arise during the development process)

2. Identify and summarize various open source available tools helpful for preparing a SRS.

There are several open-source tools available that can be helpful for preparing a Software Requirements Specification (SRS) document. These tools provide features to manage the documentation process, collaborate with team members, and create well-structured SRS documents. Here are a few popular open-source tools you can consider:

1. **LaTeX:** LaTeX is a typesetting system commonly used for creating technical and scientific documents. It's particularly useful for creating structured and well-formatted documents like SRS. While it has a learning curve, it offers powerful control over document layout and formatting.
2. **Pandoc:** Pandoc is a versatile tool that converts files from one markup format to another. You can write your SRS in a plain text format like Markdown and then use Pandoc to convert it to other formats like PDF, Word, HTML, etc.
3. **AsciiDoc:** AsciiDoc is a lightweight markup language that allows you to write structured documentation. It's easy to learn and provides the flexibility to generate various document formats.
4. **Google Docs:** While not open-source, Google Docs is a widely used collaborative tool that allows multiple team members to work on the same document simultaneously. It's web-based, so it doesn't require installation, and it supports real-time collaboration.
5. **LibreOffice Writer:** An open-source alternative to Microsoft Word, LibreOffice Writer offers a range of features for creating and formatting documents, including SRS. It supports multiple file formats, including the commonly used .docx and .pdf.
6. **Zoho Writer:** Zoho Writer is another web-based collaborative word processing tool that can be useful for creating and collaborating on SRS documents. It offers features like real-time collaboration, commenting, and version history.
7. **Draw.io:** For creating diagrams, flowcharts, and visual representations of system processes or architecture, Draw.io is a popular open-source option. It's web-based and provides an intuitive interface.
8. **Git and GitHub/GitLab:** While not exactly tools for writing SRS, version control systems like Git along with platforms like GitHub or GitLab can be immensely useful for tracking changes in your SRS document, collaborating with team members, and managing different versions of the document.