Week 1 Assignment: SRS Document

Eric Williams

CST499: Capstone for Computer Software Technology

Professor Joseph Rangitsch

Due April 15th, 2024

**Software Requirements Specification**

**for**

**CST499 Capstone Project**

**Version 1.0 approved**

**Prepared by Eric Williams**

**University of Arizona Global Campus**

**Created April 15th, 2024**

**Table of Contents**

**Table of Contents ii**

**Revision History ii**

**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 2**

2.1 Product Perspective 2

2.2 Product Functions 2

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 3

**3. External Interface Requirements 3**

3.1 User Interfaces 3

3.2 Hardware Interfaces 3

3.3 Software Interfaces 3

3.4 Communications Interfaces 3

**4. System Features 4**

4.1 System Feature 1 4

4.2 System Feature 2 (and so on) 4

**5. Other Nonfunctional Requirements 4**

5.1 Performance Requirements 4

5.2 Safety Requirements 5

5.3 Security Requirements 5

5.4 Software Quality Attributes 5

5.5 Business Rules 5

**6. Other Requirements 5**

**Appendix A: Glossary 5**

**Appendix B: Analysis Models 5**

**Appendix C: To Be Determined List 6**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| E Williams | 4/15/24 | Initial | 1.0 |
|  |  |  |  |

**Introduction**

**Purpose**

The purpose of this document is to define the requirements for an Online Course Registration System. This system will facilitate the registration process for new users and manage course enrollments.

**Document Conventions**

*<Describe any standards or typographical conventions that were followed when writing this SRS, such as fonts or highlighting that have special significance. For example, state whether priorities for higher-level requirements are assumed to be inherited by detailed requirements, or whether every requirement statement is to have its own priority.>*

**Intended Audience and Reading Suggestions**

*<Describe the different types of reader that the document is intended for, such as developers, project managers, marketing staff, users, testers, and documentation writers. Describe what the rest of this SRS contains and how it is organized. Suggest a sequence for reading the document, beginning with the overview sections and proceeding through the sections that are most pertinent to each reader type.>*

**Product Scope**

*<Provide a short description of the software being specified and its purpose, including relevant benefits, objectives, and goals. Relate the software to corporate goals or business strategies. If a separate vision and scope document is available, refer to it rather than duplicating its contents here.>*

*This section describes WHY we’re creating this software.*

**References**

*<List any other documents or Web addresses to which this SRS refers. These may include user interface style guides, contracts, standards, system requirements specifications, use case documents, or a vision and scope document. Provide enough information so that the reader could access a copy of each reference, including title, author, version number, date, and source or location.>*

*Any external document, specifications? If not, say “None.”*

**System Requirements**

**User Registration and Profile Creation**

* The system shall allow new users to register.
* Each user shall have a unique ID associated with a password.
* The system shall prevent two users from using the same ID for registration.
* User profiles shall include key information such as name, phone, and email.

**User Authentication**

* Users shall be able to login to the system at any time using the ID and password created during the registration process.

**Course Management**

* The system shall offer online courses through three semesters per year (spring/summer/fall).
* Not all courses will be offered in every semester.
* Each course shall have a maximum number of enrollments that may vary depending on the course.

**Course Enrollment and Waiting List**

* If a course is full, users shall be able to add themselves onto a waiting list.
* Users shall be able to cancel their enrollment on any course.
* If a user cancels their enrollment, the system shall inform the first user on the waiting list (if any) that they can enroll in the class.

**System Features**

**User Registration**

The system shall provide a feature for new users to register by creating an account and profile.

**Course Enrollment**

The system will provide a feature for users to enroll in courses, join a waiting list if the course is full, and cancel their enrollment if needed.

**Software Interfaces**

*<Describe the connections between this product and other specific software components (name and version), including databases, operating systems, tools, libraries, and integrated commercial components. Identify the data items or messages coming into the system and going out and describe the purpose of each. Describe the services needed and the nature of communications. Refer to documents that describe detailed application programming interface protocols. Identify data that will be shared across software components. If the data sharing mechanism must be implemented in a specific way (for example, use of a global data area in a multitasking operating system), specify this as an implementation constraint.>*

*These are internal connections to things like databases, web servers. You mainly need to explain that you have them, but (for CS421) I don’t expect detailed information about how you actually connect to them and use them. Just explain that you* ***do*** *connect to them and use them for storage of customer information, or to process incoming web requests, etc… These are internal components of your system.*

**Non-functional Requirements**

**Performance Requirements**

Able to support multiple concurrent users without significant performance degradation.

**Security Requirements**

Store user passwords securely and shall not display them in any user interface.

**Other Requirements**

*<Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.>*

*You may not have any.*

**System Requirements Chart**

*< Include a* ***table*** *in this section with the following columns:*

***ID*** *– Unique requirement ID*

***Priority*** *– Priority of this requirement*

***Type*** *– Functional(F) or Non-functional(NF)*

***Source*** *– Who is most interested in this requirement (John Smith – Customer). For this project you can make it up, in reality you’ll want to capture this as you capture the requirements.*

***Contained in Use Case(s****) – Which use cases reference this requirement or which use cases when executed will perform this requirement. There may be a few functional requirements without a use-case and the non-functional requirements generally will NOT be part of a use-case (so put N/A).*

***Description*** *– The description of the requirement. “The system shall …. “*

*>*

**Appendix A: Glossary**

*<Define all the terms necessary to properly interpret the SRS, including acronyms and abbreviations. You may wish to build a separate glossary that spans multiple projects or the entire organization, and just include terms specific to a single project in each SRS.>*

**Appendix B: Analysis Models**

*<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams*.>

**Don’t do any of these for CS421 SRS. You will create these models during the high level design deliverable.**

**Appendix C: To Be Determined List**

*<Collect a numbered list of the TBD (to be determined) references that remain in the SRS so they can be tracked to closure.>*

*List here any open questions or things you know still need to be done to the SRS, but haven’t been addressed yet. (It’s okay to have things like that, especially in this CS421 project because we don’t have time to do everything.)*

References

* Burak, A. (2024, February 15). Your guide to writing a Software Requirements Specification (SRS) document. Your 2024 Guide to Writing a Software Requirements Specification – SRS Document. https://relevant.software/blog/software-requirements-specification-srs-document/
* Krüger, G., & Lane, C. (2023, January 22). HOW TO WRITE A software requirements specification (SRS document). How to Write a Software Requirements Specification (SRS Document). https://www.perforce.com/blog/alm/how-write-software-requirements-specification-srs-document