Software Requirements Specification

for

Student Course Registration System

Version 1.0 approved

Prepared by Reece Younger, Elijah Rose, Husker Hayes, Alex Norman, and Skylar Davis

UAB EE433 FA2019

16 October 2019

Table of Contents

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

The Student Course Registration System intends to allow the interconnection of several university-level components for each semester: Students, Courses, Instructors, Registrars, and the like. The Student Course Registration System uses existing or newly generated static data fields to build relationships between each component and thus assign responsibilities and privileges to users as need be.

1.2 Document Conventions

For the remainder of the document, the following acronyms are used in the following accordance:

SRS	Software Requirements Specification
SCRS	Student Course Registration System
admin	administrator

Additional acronyms may be used in specification listings, such as SI for System Interface, or UI for User Interface. These are implied and intuitive from their section heading.

Sub-sections are of not inherently equal value and each sub-section remains largely self-contained allowing the reading of relevant sections as need be, however sections are generally constructed in a linear and procedural fashion and as such reading a super-section may be necessary to gain understanding of one of its components. Sub-sections are denoted by having the super-sections number followed by a decimal-and-number, such 1.2 Document Conventions being sub-section two of the super-section 1. Introduction.

1.3 Intended Audience and Reading Suggestions

While the SCRS primarily serves students, instructors, registrars, and other course implementation-related roles, the SRS concerns itself and assumes the audience of the administration of course affairs within a university (which may or may not be the registrar or administration of the university overall). As such, the fundamental knowledge of how basic course implementation and registration occurs is presumed. To this end, it is suggested that past the Introduction, administrative users refer to 2. Overall Description as well as perhaps 4. System Features for more nuanced details of the implemented project. While the other facets of the SCRS may interest a keen eye, they are somewhat irrelevant to the project in full.

Additionally, developers and technical staff of the SCRS company itself may find use in the detailed objectives and steps of the project, in which case after a brief glance over the 2. Overall Description, close attention is to be paid to section 3. External Interface Requirements in particular, as well as 4. System Features and 5. Other Nonfunctional Requirements, depending on the relevance to the task at hand.

1.4 Product Scope

The SCRS envelops the creation, registration, updates, and deletion of courses, tying the various users together to the primary element of courses. While the SCRS utilizes several databases of students, courses, and the like, it does not offer anything more with regards to lesson and course content management nor the means in which users contact each other (outside of the emergent communication resulting form interaction with the system).

As universities often comprise many hundreds to thousands of courses and individuals, the lack of a defined system to relate each individual component will result in confusion, lack of accessibility, unnecessary communication and individual planning, and potentially exploits. As such, the SCRS seeks to create a sustainable, flexible, and expandable interconnection software. The software primarily concerns itself with using static databases to relay to each user their necessary components and ease tedious and mundane tasks such as searching courses, finding registered students, adding and deleting courses, etc.

1.5 References

- Bandakkanavar, Ravi.(2018 July 4).*Software Requirements Specification document with example*.https://krazytech.com/projects/sample-software-requirements-specificationsrs-report-airline-database

2. Overall Description

2.1 Product Perspective

The Student Course Registration System (SCRS) is a new, self-contained product for universities and colleges to use as their Course Registration System. The SCRS attempts to separate itself from similar software with unparalleled robustness of security, ease of use, and reliability. While the overall market for Registration System software is saturated with competition, we believe this product provides a practical and out-of-the-box solution for universities who wish their Course Registration system was more robust, more secure, and easier to use. Indeed, our product towers mercilessly and unflinchingly over the competition.

2.2 Product Functions

- Allow students to register for classes
- Allow instructors to see how many and which student has registered for their classes
- Allow the accounting department to adjust the fees associated with the course
- Allow the registrar to create and edit courses
- Allow the accounting department to place holds on student records
- Allow the registrar to release transcripts
- Allow automation of the validation of course creation, registration, and withdrawal.

2.3 User Classes and Characteristics

We expect this product to be used by many different types of university employees: Students, Instructors, Accountants, Registrars, IT employees, and other administrative-type roles. The functionality of the product will be scoped depending on who is using it, providing high-level security and dependability. We do not expect a complicated transition from another piece of software to this software due to the high-familiarity university employees have with computers and Registration Systems in general. Students in 2019 and beyond are extremely computer-literate and will have no trouble using the software to register for their courses. Other employees, such as Accountants, Instructors, and Registrars, could also be assumed to be familiar enough with computers to expect a short transition time to using this product effectively. The application will be separated into different sub-systems that encapsulate and abstract away the parts of the application not being used. For example, an Instructor will not have to be concerned about how courses are created because this level of administration will be abstracted away from him (not viewable). Likewise, a student will not have to be concerned about how Instructors upload their notes or course materials, they will just have to learn how to view them.

2.4 Operating Environment

The software was developed as a Java desktop application that will operate on Windows, OSX and Linux platforms through the Java Runtime Environment. It will not allow registration through a web server.

2.5 Design and Implementation Constraints

The application must provide security and privacy to all employees who use it. It is vital to ensure things such as disabilities, medical records, or other sensitive and personal information remain securely within the system in accordance with all proper ethics and privacy laws. Because this is a desktop application that does not use a web server, we expect a minimal amount of security and privacy challenges.

The Java platform was chosen for development due to its rich third-party ecosystem and portability. Since the JRE is a virtual machine that exists within Windows, OSX, and Linux, it allows us to develop for three different operating systems within a single codebase. The vast majority of today's computers have Java installed.

The application must design in a way that loosely couples different components and subsystems through interfacing so that updates and patches can be released in a timely and efficient manner.

2.6 User Documentation

N/A

2.7 Assumptions and Dependencies

The guidelines and constraints of the application described in this document is subject to change based on a variety of factors, such as our investors increasing or decreasing the scope of the design, team sickness, team interruption, personal tragedy, and any changing of requirements as in common within an agile development environment.

3. External Interface Requirements

3.1 User Interfaces

UI-1: The SCRS graphical user interface shall allow students to navigate to and view the course catalog.
UI-2: The SCRS

3.2 Hardware Interfaces

No hardware interfaces have been identified.

3.3 Software Interfaces

SI-1: Registration Logging System
SI-1.1: The SCRS shall transmit the student and offered-course identities to the Registration
Logging System through a programmatic interface.
SI-1.2: The SCRS shall query the Registration Logging System to determine whether the
course is available for subscription.
SI-1.3: The SCRS shall transmit the date a student withdrew from a course.
SI-1.4:

SI-2: Student Registry System
The SCRS shall communicate with the Student Registry System through a programmatic interface for the following operations:
SI-2.1: To allow a student to register for a course offering.
SI-2.2: To allow a student to unregister for a course offering.
SI-2.3: To check whether a student is registered for a course.
SI-2.4: To check whether a student has the required pre-requisites to register for a course.
SI-2.5: To check whether a student has enough available credit hours to register for a course.

SI-3: Course Catalog System
The SCRS shall communicate with the Course Catalog System through a programmatic interface for the following operations:
SI-2.1: To check whether a course is available for a given term.
SI-2.2: To query a course's information.

SI-2.3: To check whether a student is registered for a course.
SI-2.4: To check whether a student has the required pre-requisites to register for a course.
SI-2.5: To check whether a student has enough available credit hours to register for a course.

3.4 Communications Interfaces

CI-1: The SCRS shall

4. System Features

4.1 Student Component

4.1.1 Description and Priority

The student is a base and integral component to the whole system as they are the highest instance component in our data model. The priority of this feature is high to the overall system interface and must be implemented.

4.1.2 Stimulus/Response Sequences

The student shall have permissions register for appropriate classes validated by the advisor component. Some views specific to the student are availability of courses with respect to time, number of students, CRN, etc. Some responses back to a student would be a validating response that have registered or the appropriate blocking message for why they cannot.

4.1.3 Functional Requirements

REQ-1: The student shall be able to register for a course
REQ-2: The student shall be able to search for classes based on department.
REQ-3: The student shall not be able to register for conflicting times.
REQ-4: The student shall not be able to register for classes they are not eligible for
REQ-5: The student shall be able to see which courses they are supposed to take
REQ-6: The student shall be able to drop a class
REQ-7: The student shall be able to see the time and location of the courses
REQ-8: The student shall not be able to register for a class that is at max capacity
REQ-8: Students shall use RAC to confirm registration requirements
REQ-9: Separate user shall not be able to view other account information

4.2 Course Component

4.2.1 Description and Priority

The Course component is just as integral to the whole system as the Student. The Courses will determine the limits of action some of the student can take and can be thought of as the product the student is wanting.

4.2.2 Stimulus/Response Sequences

The Course will have a limit to the number of students that will register for the course as well as an instructor. The Course shall hold some unique identifier as well as a time, location, associated major, and description of the course. The course will not have any user associated with it but will have many interacting with the course.

4.2.3 Functional Requirements

REQ-1: If the chosen course is oversubscribed, the system shall prevent the student from registering for it.

4.3 Registration System Component

4.3.1 Description and Priority

The registration will hold the components together and is the main workings of this whole system. IT is the main interface between the student and the course our two key products and buyers. The mesh between these two components will build workings of the rest of the system features.

4.3.2 Stimulus/Response Sequences

The system shall be the response to each of the stimulus from each appropriate component. The system as a whole will respond and communicate between classes and other objects within the whole system.

4.3.3 Functional Requirements

REQ-1: The SCRS shall provide students the ability to register for a course.
REQ-2: The system shall maintain a repository of courses that students may register for.
REQ-3: The SCRS shall provide instructors tools to make course-change-requests to the registrar.
REQ-4: The SCRS shall provide advisors access to tools and information that may help a student with registration and course selection.
REQ-5: The system shall allow advisors to view the courses a student is registered for.
REQ-6: The system shall allow advisors to generate a registration access code (RAC) for a student for a specified semester.
REQ-7: The SCRS shall provide registrar faculty the ability to manage courses.
REQ-8: The SCRS shall provide registrar faculty the ability to process instructor requests for course changes.
REQ-9: The system shall notify the instructor when his/her course modification request has been completed or denied.
REQ-10: The system shall not allow course modifications 2 weeks after the start of a semester.
REQ-11: The system shall not allow an instructor to be assigned to a course if he/she is already assigned to a course during that time.

REQ-12: The SCRS shall provide accounting faculty the ability to manage registration expenses and generate economic reports.
REQ-13: The SCRS shall provide system registrars the ability to modify a user's permissions.
REQ-14: The SCRS shall provide system registrars the ability to modify a user's permissions.

4.4 Other Components

4.3.4 Description and Priority

The other components are not essential to the base functionality of the whole system. These working and features are addons of the base 3 functional components being the IT, Instructor, Advisors, and Registrar.

4.3.5 Stimulus/Response Sequences

These components are the majority of validation or additional pieces to fully wrap the whole system. These different classes will hold roles pertaining to updating modifying or only view privileges. Their responses are integral to the system but mostly only respond to a stimulus from either a student or the system.

4.3.6 Functional Requirements

REQ-1: The instructor shall be able to see how many students have registered for their course
REQ-2: The instructor shall be able to see the name, photo, email and major of the registered student
REQ-3: The instructor shall be able to see the name, photo, email and major of the registered student
REQ-4: The advisor shall be able to see the student's entire history
REQ-5: The advisor shall be able to see what courses the student is enrolled in
REQ-6: The advisor shall be able to override prerequisites
REQ-7: The advisor shall be the only one able to obtain RAC codes for each student
REQ-8: The registrar shall be able to create courses
REQ-9: The registrar shall be able to delete courses
REQ-10: The accounting role shall be able to edit the course fees
REQ-11: The accounting role shall be able to read course information
REQ-12: The registrar shall be able to view (read) any course
REQ-13: The registrar shall be able to update any course
REQ-14: The registrar shall be able to filter course searches by semester
REQ-15: The registrar shall be able to filter course searches by department
REQ-16: The registrar shall be able to filter course searches by professor
REQ-17: IT shall be able to emulate user roles for troubleshooting
REQ-18: IT shall be able to access user login information

5. Other Nonfunctional Requirements

5.1 Performance Requirements

PR 1: The SCRS shall maintain an uptime of at least 95% during the active registration time.

PR 2: The SCRS shall be usable with only a moderate amount of training.
PR 3: IT shall be able to emulate user roles for the purpose of maintenance

5.2 Safety Requirements

N/A

5.3 Security Requirements

SR 1: The SCRS shall uphold all FERPA regulations
- SR 1.1: Non-directory student information shall only be viewable by the student
- SR 1.2: Parents of the student may not be granted access to student information
- SR 1.3: The SCRS shall not allow disclosure of student information to 3rd parties

5.4 Software Quality Attributes

SQA 1: The SCRS shall be school agnostic to allow for reuse
SQA 2: The SCRS shall allow for easy theming to allow for reuse
SQA 3: The SCRS shall be usable with only a moderate amount of training
SQA 4: The SCRS shall be designed using object-oriented principles to allow for modular design

5.5 Business Rules

N/A

6. Diagram models

Figure 1 Instructor viewing student for a semester.

Figure 2 Accounting department viewing

Figure 3 Top-Level Class Diagram

Appendix A: Glossary

SRS	Software Requirements Specification
SCRS	Student Course Registration System
admin	administrator