<C's Get Degrees>

GradSchoolZero
Software Requirements Specification
For <Subsystem or Feature>

Version <1.0>

[Note: The following template is provided for use with the Rational Unified Process. Text enclosed in square brackets and displayed in blue italics (style=InfoBlue) is included to provide guidance to the author and should be deleted before publishing the document. A paragraph entered following this style will automatically be set to normal (style=Body Text).]

[To customize automatic fields in Microsoft Word (which display a gray background when selected), select File>Properties and replace the Title, Subject and Company fields with the appropriate information for this document. After closing the dialog, automatic fields may be updated throughout the document by selecting Edit>Select All (or Ctrl-A) and pressing F9, or simply click on the field and press F9. This must be done separately for Headers and Footers. Alt-F9 will toggle between displaying the field names and the field contents. See Word help for more information on working with fields.]

GradSchoolZero	Version: <1.0>
Software Requirements Specification	Date: <26/10/2021>
<document identifier>	

Revision History

Date	Version	Description
<26/10/2021>	<1.0>	<The GradSchoolZero system is being developed by the team of Joel James, Jaehong Cho, Aiman Fiaz, Michael Edquilan, and Raghib Shaumik.>

Author

<name>

GradSchoolZero	Version: <1.0>
Software Requirements Specification	Date: <26/10/2021>
<document identifier>	

Table of Contents

GradSchoolZero	Version: <1.0>
Software Requirements Specification	Date: <26/10/2021>
<document identifier>	

Software Requirements Specification

1. Introduction

*[The introduction of the **Software Requirements Specification (SRS)** provides an overview of the entire document. It includes the purpose, scope, definitions, acronyms, abbreviations, references, and overview of the **SRS**.]*

*[Note: The **SRS** captures the complete software requirements for the system, or a portion of the system. Following is a typical **SRS** outline for a project **using use-case modeling**. This artifact consists of a package containing use cases of the use-case model and applicable Supplementary Specifications and other supporting information. For a template of an **SRS not** using use-case modeling, which captures all requirements in a single document, with applicable sections inserted from the Supplementary Specifications (which would no longer be needed), see the file titled rup_srs.dot.]*

*[Many different arrangements of an **SRS** are possible. Refer to [IEEE93] for further elaboration of these explanations, as well as other options for an **SRS** organization.]*

The introduction of the Software Requirement Specification provides an overview of the entire SRS with purpose, scope, definitions, acronyms, abbreviations, references, and an overview of the SRS. The aim of this document is to analyze and give a detailed insight into the "GradSchoolZero" system by defining the problem statement in detail. This plan will include a summary of what the purpose of this system is, the technology used to develop the project, and the metrics used to determine the project progress, as well as the overall description.

1.1 Purpose

*[Specify the purpose of this **Software Requirements Specification**. The **SRS** fully describes the external behavior of the application or subsystem identified. It also describes nonfunctional requirements, design constraints, and other factors necessary to provide a complete and comprehensive description of the requirements for the software.]*

The Software Requirements Specification provides the roadmap for our program and helps us understand our goal better. The application is inspired by the lack of features in CUNYfirst, which is CUNY's current Integrated Resources and Services Tool. The main purpose of this application is to improve the overall quality of the learning experience for CUNY students. Since a small database will be used for this application, the program should run fast enough.

1.2 Scope

*[A brief description of the software application that the **Software Requirements Specification** applies to, the feature or other subsystem grouping, what Use-case model(s) it is associated with, and anything else that is affected or influenced by this document.]*

GradSchoolZero helps schools manage their courses effectively, motivate the instructors, for example using a rating system, and give students extra features that CUNYFirst doesn't have like leaving reviews and complaining to other students. A use-case diagram will be a good candidate to represent our project since each user can access different functionalities.

1.3 Definitions, Acronyms, and Abbreviations

*[This subsection provides the definitions of all terms, acronyms, and abbreviations required to properly interpret the **Software Requirements Specification**. This information may be provided by reference to the project's Glossary.]*

Types of Users

1. Registrars: A super user who can do anything and overseas or maintains the system
2. Instructors: Teachers who can access courses, assign grades, are assigned by registrars
3. Students: Enroll in courses and manage courses they take
4. Outside Visitors: Anyone outside of the program; not a registrar, instructor or student. Can access basic information about classes and students. Can apply to become a student.

Semester- The school semester is divided into four periods

1. Class Set-up Period: Registrars set up courses, class time, course instructors and course size.
2. Course Registration: Students can register for available courses.
3. Class Running Period: Courses begin, students can no longer register for courses.
    a. Special Registration Period: Students previously registered in a cancelled course are given extra time to register for other available courses.
4. Grading Period: Instructors assign grades to students

Cancelled Course: A course with less than 5 students enrolled by the start of the class running period.
Suspended Instructor: An instructor whose courses were all cancelled. They are ineligible to teach next semester.
Suspended Student: A student can be suspended by receiving 3 warnings and must pay a fine to the registrars. Or a student can be suspended by dropping all courses. Suspended students cannot enroll for 1 semester.
GPA: Grade Point Average; the average of a students' grade throughout their academic career.
Warning: a strike that can be issued to students or instructors. Warnings can be issued automatically by the disobedience of rules or issued by students or instructors. These warnings are reviewed and processed by the registrars. An accumulation of 3 warnings will result in a penalty depending on the party at hand.

1.4 References

*[This subsection provides a complete list of all documents referenced elsewhere in the **Software Requirements Specification**. Identify each document by title, report number (if applicable), date, and publishing organization. Specify the sources from which the references can be obtained. This information may be provided by reference to an appendix or to another document.]*

Such a project wouldn't have been completed without the use of PyQt5 library which provided all the help in making the Application GUI (Graphical user interface), also by referencing other libraries that helped in the application's function like Pandas, sys, csv, os.

Here is some web pages' links that were used:

- https://www.geeksforgeeks.org/
- https://stackoverflow.com/
- https://careerkarma.com/
- https://www.w3schools.com/python/

1.5 Overview

*[This subsection describes what the rest of the **Software Requirements Specification** contains and explains how the document is organized.]*

The software's start up page that shows up once you open the software contains three main widgets which are a login button, sign up button, and 3 lists for the highest rated classes, lowest rated classes and the students with the highest GPA.

Once you click on one of the buttons you will be directed to another page where you can sign up or log in, after you input the username and password they will be saved in a csv file so your account is remembered next time you use the software, then you will be redirected to the software's main page.

2. Overall Description

*[This section of the **Software Requirements Specification** describes the general factors that affect the product and its requirements. This section does not state specific requirements. Instead, it provides a background for those requirements, which are defined in detail in Section 3, and makes them easier to understand. Include such items as product perspective, product functions, user characteristics, constraints, assumptions and dependencies, and requirements subsets.]*

Visitors on the site have access to explore courses. They can also apply to be students or instructors. Registrar decides if their applications are to be accepted.

Students are allowed to rate the instructors. It helps the instructors to improve their teaching strategies. It also helps the registrar to ensure that students are being taught by competent instructors. Students are also allowed to complain about their instructors. The instructors can also complain about students. These accusations are to be investigated by the registrars. The registrar is to take actions according to their investigation.

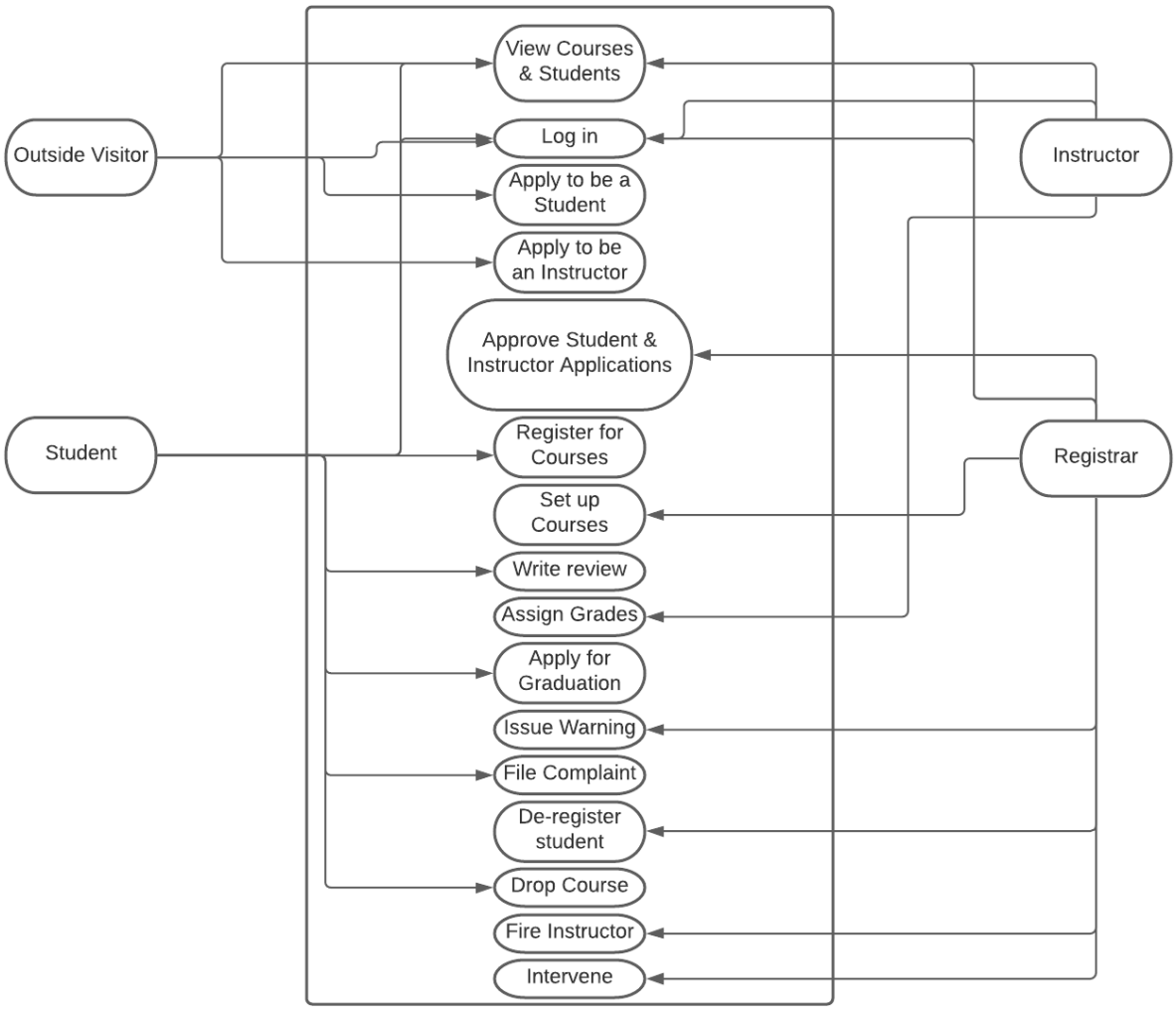
Every student can see his or her individual record and drop classes accordingly. Instructor can see basic information and academic records of the students in his or her class. Registrar has access to all information. Students will also receive a tutorial on how to use the system.

The product perspective of this project is that it is a local system that uses a database to keep information from students and instructor's records. The product is utilized to allow for registered students and instructors to create, keep, and access their academic records. In this section, the model of the GradSchoolZero system will be explained here with the utilization and aid of the use case diagrams. The use case models will show how the system functions and illustrates how the users in this system will interact with the system features. The section also talks about user characteristics and assumptions & dependencies created.

GradSchoolZero	Version: <1.0>
Software Requirements Specification	Date: <26/10/2021>
<document identifier>	

2.1 Use-Case Model Survey

*[If using use-case modeling, this section contains an overview of the use-case model or the subset of the use-case model that is applicable for this subsystem or feature. This includes a list of names and brief descriptions of all use cases and actors, along with applicable diagrams and relationships. Refer to the **Use-Case-Model Survey Report**, which may be used as an enclosure at this point.]*



2.2 Assumptions and Dependencies

*[This section describes any key technical feasibility, subsystem or component availability, or other project related assumptions on which the viability of the software described by this **Software Requirements Specification** may be based.]*

Four main users are to use this software. All other users depend on the Registrars. If a visitor wants to be a student or an instructor, registrars can accept or deny the applications. Registrars are also able to set up details of classes. Since registrars are super users of this software, we have to be very careful while writing these functions. Small mistakes in the register class can slow down our progress by a large margin.

Instructors are the 2nd most important users after the Registrars. We have to link Instructor and Registrar classes properly. Registrars must warn an instructor if the rating of that instructor's class is less than 2. This means we have to set up a rating system for the instructors. Instructors will also have to be warned if their class GPA (in average) is too high or too low.

For the student class, we must ensure that students can't enroll in 2 classes with time conflicts. Registrars also have to warn students if their GPA falls below 2.25. If the student's GPA falls below 2.0 or the student fail's same class more than once the student would have to be terminated automatically. Students must have options to drop classes.

Outside visitors will have options to apply to be a student or instructor. Registrars are allowed to accept or reject their applications. Visitors can also explore courses but won't be able to enroll. Visitors can't have access to any of the student, instructor, or registrar functions.

3. Specific Requirements

*[This section of the **Software Requirements Specification** contains all software requirements to a level of detail sufficient to enable designers to design a system to satisfy those requirements and testers to test that the system satisfies those requirements. When using use-case modeling, these requirements are captured in the use cases and the applicable supplementary specifications. If use-case modeling is not used, the outline for supplementary specifications may be inserted directly into this section.]*

For running the software a couple of files are needed either for functionality or design, firstly the logo will be needed in the same folder with the code in .png format and named "logo". Secondly, a csv file is needed within the same folder as well with the name "users" for saving the accounts data, if this csv file isn't present, one will be automatically created once you run the code.

3.1 Use-Case Reports

[In use-case modeling, the use cases often define the majority of the functional requirements of the system, along with some non-functional requirements. For each use case in the above use-case model, or subset thereof, refer to, or enclose, the use-case report in this section. Make sure that each requirement is clearly labeled.]

View Courses & Students

'View Courses & Students' allows all users to see the highest/lowest rated courses and highest rated GPA. There is no requirement for this use case. Even visitors have access to this.

Login

'Login' enables visitors to login so they can have access to other functionalities. Requirements for logging in is to be in the user database which you need to sign up to get in.

Apply to be a Student
'Apply to be a Student' allows all visitors to apply to be a student. There is no requirement to 'apply' to be a student, but to be a student the applicants must have the approval from the registrars.

Apply to be an Instructor
'Apply to be an Instructor' allows all visitors to apply to be an instructor. There is no requirement to 'apply' to be an instructor, but to be an instructor the applicants must have the approval from the registrars.

Approve Student & Instructor Applications
'Approve Student & Instructor Applications' allows registrars to approve student & instructor applications. To be approved students need to have GPA > 3.0 and program quota is not reached.

Register for Courses
'Register for Courses' allows students to register for courses. To register students 1)must have 1-3 courses, 2)no time conflict among chosen classes and 3)the upper limit of the course is not reached. Also students can retake courses if s/he got F before.

Set up Courses
'Set up Courses' allows registrars to set up courses. While setting up a course instructors need to specify 1)class time, 2)course instructors and 3)class size.

Write Review
'Write Review' allows students to write reviews of the classes they are in and assign stars. Writing reviews is available before students get their grades.

Assign Grades
'Assign Grades' allows instructors to assign grades of their students. Instructors should assign grades before the grading period ends if they want to avoid warnings.

Apply for Graduation
'Apply for Graduation' allows students to apply for graduation. Students need to be finishing 8 classes to apply for graduation. If the student has completed all required courses, they would graduate and leave with a Master's Degree, otherwise the registrar would issue a warning to the student for reckless graduation application.

Issue Warning
'Issue Waring' allows registrars to students or instructors based on the complaints they received. Registrars need to conduct an investigation before issuing warnings.

File Complaint
'File Complaint' allows instructors to file complaints on their students. There is no requirement for instructors to file complaints, but the complaints should be reasonable, otherwise instructors might get warnings for themselves. The complaints are reviewed by the registrars.

De-register Student
'De-register Student' allows registrars to de-register students. Registrars need to have requests from instructors in order to de-register students. They might need to review the complaints thoroughly before making the decisions.

Drop Course
'Drop Course' allows a student to drop his/her course with a grade of 'W'. There is no requirement to drop courses, but students need to reconsider dropping all of their courses since they will be suspended for one semester unless registrars take action.

Fire Instructor
'Fire Instructor' allows registrars to fire instructors. Registrars can fire instructors when the instructors don't have valid justification for having a class GPA above 3.5 or below 2.5.

Intervene
'Intervene' allows registrars to go against the protocol. For example, registrars might reject an application of a student even though her/his GPA is >3.0 and program quota is not reached. In this case, registrars should provide valid justification of their decision.

3.2 Supplementary Requirements

[Supplementary Specifications capture requirements that are not included in the use cases. The specific requirements from the Supplementary Specifications, which are applicable to this subsystem or feature, should be included here and refined to the necessary level of detail to describe this subsystem or feature. These may be captured directly in this document or referred to as separate Supplementary Specifications, which may be used as an enclosure at this point. Make sure that each requirement is clearly labeled.]

Provide justification: If the program quota is not reached and the registrar decides to reject the application of one of the applicants with 3.0 or better GPA, the registrar must provide justification. Registrar also provides justification if it accepts the application of an applicant whose GPA is less than 3.0.

No justification required: Visitors on the site are also allowed to apply for "instructor" position in the school. Registrar can accept or reject these applicants without providing any justification for their decision.

Suspension of instructor: Courses with less than 5 students have to be cancelled and the instructor is notified. If all the courses of an instructor are cancelled, the registrar is to suspend the instructor. Students also rate their instructors. If the average rating of an instructor for a course is below 2.0, the system is to warn the instructor. If any instructor receives 3 or more warnings, the registrar is to suspend the instructor.

4. Supporting Information

*[The supporting information makes the **Software Requirements Specification** easier to use. It includes:*

- *Table of Contents*
- *Index*
- *Appendices*

*These may include use-case storyboards or user-interface prototypes. When appendices are included, the **Software Requirements Specification** should explicitly state whether or not the appendices are to be considered part of the requirements.]*

The Software Requirements Specification includes:

- Table of Contents
- Index

Release Plan	Description	Motivation	Duration (hours)
Step 1	Create the login and signup page	Makes the system more secure as each registered student will have unique login info.	1-2 hours
Step 2 (depends on Step 1)	Working on the application page for students and instructors	For the user to be able to use the system, the user has to be a student that is part of the CUNY graduate school.	4-5 hours
Step 3 (depends on Step 2)	Working on the approval criteria for the acceptance of students and instructors	To determine if the student and instructor is eligible to be part of the graduate program.	4-5 hours
Step 4 (depends on Step 3)	Implement a page where students can register, set up, or drop courses.	The registered user needs to be able to utilize features in order to create a schedule for the semester.	5-6 hours
Step 5	Working on the feature that allows students to write reviews to classes.	This can be used as useful feedback for other students who are looking to take the same classes.	3-4 hours
Step 6	Instructors will be able to assign grades.	This will allow for registered students to view how they are doing in their classes.	3-4 hours
Step 7 (depends on Steps 1-6)	Students can apply for graduation.	This feature allows for the student to see if they have met the criteria to graduate and have completed the curriculum.	1-2 hours
Step 8 (depends on Steps 3, 4, and 6)	Working on the features to remove students from the course and to file a complaint against instructors.	A feature that will allow the instructor to remove the student for disciplinary or other reasons. A feature that also allows for students to send feedback to school offices about instructors.	3-4 hours