Requirements Document for ThisIsSpartanTech

# Introduction

Spartanburg Technical College was founded for the purpose of offering college-level study in the STEM fields in a focused, small classroom environment. Over the last three years, Spartanburg Technical College has seen its enrollment numbers nearly double in size from 5,000 to 9,000 students and its diversity initiatives have been recognized on a regional and state-wide level. As a relatively new - but growing - collegiate institution, STC recognizes the need for a comprehensive course registration and management application for their faculty, staff, and students’ use. Therefore, Spartanburg Tech plans to develop a web application that allows for course registration and grade viewing for students and course management for faculty and staff. To that end, Spartanburg Tech put together an exploratory committee to determine the necessity of such a system and potential applications for the system to incorporate. The committee found that such a system would be a success among the technologically savvy student body and professorship in particular, and STC has decided to move forward with “ThisIsSpartanTech,” a web application that provides a simple and clear interface for students, faculty, and administration looking into course requirements and registration.

Spartanburg Technical College hopes to take advantage of the breadth of technical skill in their student body by employing a group of interns to design and implement version 1 of the system under the supervision of their database administration professor and the college’s IT department. This document is based on extensive interviews conducted by the exploratory committee and the student interns with department heads and administrative officials. All information in this document is determined to be accurate at the time of this writing.

This document contains the detailed requirements for the ThisIsSpartanTech application that is being developed for Spartanburg Technical College. The document should serve as the official basis for any further development of the ThisIsSpartanTech. The document contains the following sections:

* Executive Summary
* Application Context
* Functional Requirements
* GUI Design
* Environmental Requirements
* Other Requirements
* Software Qualities
* Time Schedule
* Potential Risks
* Future Changes
* Glossary
* References

# Executive Summary

Spartanburg Technical College plans to implement this undergraduate course management system to simplify typical processes associated with course management from the student, professor, and administration viewpoints. STC believes that this system will effectively optimize these processes that are typically time-consuming and inefficient. This system will provide the following features:

* *Addition/deletion of courses -* Users with the administration role can add or delete courses available for student registration.
* *Registration/withdrawal of students from courses -* Users with the professor role can add/drop students from their course; users with the student role can register/drop themselves from courses.
* *Grade management and viewing -* Users with the professor role can add grades for students enrolled in their courses; users with the student role can view grades assigned to them; users with the administrator role can view grades for all students pursuing a degree in their respective department.
* *Optional: Viewing of undergraduate degree information -* Users with the student role will be able to view course requirements for degree completion; users with the professor and administrator roles will be able to view a student’s degree progress

Please Take Note: Though contents of this requirements document for the ThisIsSpartanTech course management system are intended to be as comprehensive and thorough as possible, ambiguity or inconclusivity may occur. Clarification may be requested by contacting Spartan Tech’s IT Department via itdpt@spartantech.com.

# Application Context

ThisIsSpartanTech will be used to manage course registration, course availability, and graduation requirements for undergraduate students of the college. The IT staff of Spartanburg Technical College will need to be prepared to troubleshoot any issues and, therefore, must be trained to perform all operations for the application (e.g. creation of courses, student registration for courses, viewing degree progress, and handling of student grades). Developers in Spartan Tech’s internship program will be expected to maintain the application based on user feedback and demand.

# Functional Requirements

***DATABASE ADMINISTRATOR FUNCTIONALITY***

***Database Administrator Login***

* + 1. All database administrators (DBA) have to be verified using a login and password that is issued by Spartanburg Technical College. The application should offer the DBA a set of user interfaces different from those offered to other views since they have access to features that allow them to edit, delete and add information to the database.

***Adding/Deleting Administrator***

* + 1. DBAs will be able to add and delete School Administrators. When this function is selected, the graphical user interface (GUI) will provide the user with two options: “Add an Administrator” or “Delete an Administrator”.
    2. “Administrator Information” for a school administrator is as follows:
       1. AdminID. Every Administrator must have a unique AdminID by which it can be identified.
       2. Adminstrator Name. First Name and Last Name will be entered separately.
       3. The associated department key for the department in which the administrator works
    3. DBA will be offered to search by AdminID or Last Name. Selecting either option allows for that value to be searched. If no match is found, DBA will be returned to search again with appropriate error message.
    4. The GUI should allow only one Administrator to be added or removed at a time.

***Adding a Course***

* + 1. The GUI should allow the DBA to create new course information and save it to the database. If department, course number, and section number already exist, the GUI should prompt the DBA accordingly.
    2. The “Course Information” for a course includes:
       1. The course registration number (CRN). Every course must have a unique 5-character code by which it can be identified.
       2. The department key for the course. Every department must have a unique department key by which it can be identified.
       3. The course number. The 4-character course number can only have one course name but can have multiple section numbers.
       4. The section number. The 2-character section number is not unique, but the combination of course number and section number must be unique in a unique semester and year combination.
       5. The name of the professor teaching the course. This can be left blank by the administrator to be filled by the professors at a later time.
       6. A short description of the course. This description may include references to other courses as prerequisites, in which case they should point to the corresponding course information.
       7. The semester in which the course is taught.
       8. The year in which the course is taught
    3. The GUI should only allow the information for one course to be entered at a time.

***Deleting a Course***

* + 1. The DBA should be able to perform search queries for a course to be deleted by CRN or Course Name. Searching by course name should display a list of CRN and Course Names from which to select.
    2. Once a course is selected, the DBA will be prompted whether they are certain they want to delete the course. When confirmed, the course will be deleted from the database and all registered students will be dropped from the course.
    3. Only one course may be deleted at a time. Once the course has successfully been deleted, the user will be returned to the main menu.
    4. The DBA is able to delete any course they choose, including those offered in the past, e.g. if a course becomes obsolete for a major, the DBA can remove this course from courses available to register.

***Editing a Course***

* + 1. The GUI should allow the DBA to perform search queries to retrieve a course for the purpose of editing the Course Information, as enumerated in section 4.1.7. The search should be conducted using either the CRN or course name, and the resulting list will be displayed by CRN with the course name.
    2. When the course has been selected, several options will allow the user to modify course information. Functions to modify each section except for the department and the CRN of the Course Information will be made available. Selecting an option to modify will prompt the user for the updated information. These options include professor first name, last name and ID number, semester(s) in which the course is offered, year(s) in which the course will be offered, course description, section number, and course number.
    3. The GUI should confirm the update and show the same options once the change is confirmed.

***Adding/Deleting a Professor***

* + 1. When this function is selected, the GUI will provide the DBA with options to add a new professor or remove the professor from the Professor Information and from any linked classes.
    2. The “Professor Information” for a professor includes:
       1. The professor ID. This 7-character ID must be unique to identify the professor.
       2. The professor’s name. First Name and Last Name will be entered separately.
       3. The department key. Every department must have a unique department key by which it can be identified. This is the department to which the professor is assigned.
    3. DBA will be offered to search by Professor ID or Last Name. Selecting either option allows for that value to be searched. If no match is found, DBA will be returned to search again with appropriate error message.
    4. The GUI should allow only one Professor to be added or removed at a time.
    5. Removing a professor updates any classes a professor was linked to with a null value.

***Modify Professor***

* + 1. The GUI will prompt the DBA to search by professor ID or professor name. If searched by name, the GUI should display a list of matching names with the option to select the professor.
    2. DBA is presented with professor ID, name (first and last), and department. The functions from this point are:
       1. Change Name. Changes the professor’s name to user input. The GUI should ask for first name, then last name before committing update and returning to the options menu.
       2. Change Department. Changes the department to which the professor is assigned and assigns a new department ID to the professor.

***Adding/Deleting Student***

* + 1. When this function is selected, the GUI will provide the DBA with options to add a new student or remove the student from the Student Information and from any linked classes.
    2. The “Student Information” for a student includes:
       1. The student ID. This 7-character ID must be unique to identify the student.
       2. The student’s name. First Name and Last Name will be entered separately.
       3. The student’s major (Math, Engineering, Computer Science, etc).
    3. DBA will be offered to search by student ID or Last Name. Selecting either option allows for that value to be searched. If no match is found, DBA will be returned to search again with appropriate error message.
    4. The GUI should allow only one Student to be added or removed at a time.
    5. Deleting a student updates any classes a student was linked to by removing their registration in a course.

***Adding/Deleting a Department***

* + 1. The “Department” table for a student includes:
       1. The Department Key. This 4-letter String identifies the department, e.g. CSCI for Computer Science, MATH for Mathematics, ENGN for Engineering, etc.
       2. The Department Name. The full name of the department that corresponds with the 4-letter String mentioned above.

***Modify Student Registration***

* + 1. When this function is selected, the GUI will provide the DBA with options to modify all enrolled students. The GUI will prompt the user to search by student ID or last name. If searched by last name, the GUI should display a list of matching names with an option to select the student.
    2. The DBA is presented with the information student ID, name (first and last), and major. The functions from this point are:
       1. Change Name – Changes the student’s name to user input. The GUI should prompt for First Name, then Last Name. After saving, the GUI should return to the options menu.
       2. Change Major – Changes the student’s enrolled major to user input. After saving, the GUI should return to the options menu.
       3. Add Course – Registers the student to a new course. This prompts the user to search for a class to register. The GUI presents the user with a search by CRN or department key. From this point, a list is presented to the DBA to select the class to register. After saving, the GUI should return to the options menu.
       4. Drop Course – When this option is selected, the GUI should display all classes in which the student is currently registered by CRN and course name. Once a class is selected, the user must verify to drop the student from the selected class.
       5. Drop All – This option effectively unregisters the student from all classes using their student ID. The user must verify to drop the student from all classes.

***SCHOOL ADMINISTRATOR FUNCTIONALITY***

***School administrator Login***

* + 1. All school administrators have to be verified using a login and password that is issued by Spartanburg Technical College. The application should offer the administrator a set of user interfaces distinct from those offered to other views, despite having similar functionality.
    2. Each department only has one college/school administrator that works in the department.

***Adding a Course***

* + 1. The GUI should allow a school administrator to create new course information and save it to the database. If department, course number, and section number already exist, the GUI should prompt the administrator accordingly.
    2. The “Course Information” to be added is enumerated in section DBA 4.1.7. above.
    3. The GUI should only allow the information for one course to be entered at a time.

***Deleting a Course***

* + 1. The administrator should be able to perform search queries for a course to be deleted by CRN or Course Name. Searching by course name should display a list of CRN and Course Names from which to select.
    2. Once a course is selected, the DBA will be prompted whether they are certain they want to delete the course. When confirmed, the course will be deleted from the database and all currently registered students will be dropped from the course.
    3. Only one course may be deleted at a time. Once the course has successfully been deleted, the user will be returned to the main menu.

***Editing a Course***

* + 1. The GUI should allow the DBA to perform search queries to retrieve a course for the purpose of editing the Course Information (see section DBA 4.1.7.). The search should be conducted using either the CRN or course name, and the resulting list will be displayed by CRN with the course name.
    2. When the course has been selected, several options will allow the user to modify course information. Functions to modify each section except for the CRN of the Course Information will be made available. Selecting an option to modify will prompt the user for the updated information.
    3. The GUI should confirm the update and show the same options once the change is confirmed.

***Adding/Deleting a Professor***

* + 1. When this function is selected, the GUI will provide the DBA with options to add a new professor or remove the professor from the Professor Information and from any linked classes.
    2. The “Professor Information” to be added is enumerated in section DBA 4.1.16. above.
    3. School administrator will be offered to search by Professor ID or Last Name. Selecting either option allows for that value to be searched. If no match is found, administrator will be returned to search again with appropriate error message..
    4. The GUI should allow only one Professor to be added or removed at a time.
    5. Removing a professor updates any classes a professor was linked to with a null value.

***Modify a Professor***

* + 1. The GUI will prompt the school administrator to search by professor ID or professor name. If searched by name, the GUI should display a list of matching names with the option to select the professor.
    2. Administrator is presented with professor ID, name (first and last), and department. The functions from this point are:
       1. Change Name. Changes the professor’s name to user input. The GUI should ask for first name, then last name before committing update and returning to the options menu.
       2. Change Department. Changes the department to which the professor is assigned and assigns a new department ID to the professor.

***Add/Delete Student***

* + 1. When this function is selected, the GUI will provide the school administrator with options to add a new student or remove the student from the Student Information and from any linked classes.
    2. The “Student Information” to be added is enumerated in section DBA 4.1.23. above.
    3. Administrator will be offered to search by student ID or Last Name. Selecting either option allows for that value to be searched. If no match is found, school administrator will be returned to search again with appropriate error message.
    4. The GUI should allow only one Student to be added or removed at a time.
    5. Deleting a student updates any classes a student was linked to by removing their registration in a course.

***Modify Student Registration***

* + 1. When this function is selected, the GUI will provide the DBA with options to modify all enrolled students. The GUI will prompt the user to search by student ID or last name. If searched by last name, the GUI should display a list of matching names with an option to select the student.
    2. The DBA is presented with the information student ID, name (first and last), and major. The functions from this point are:
       1. Change Name – Changes the student’s name to user input. The GUI should prompt for First Name, then Last Name. After saving, the GUI should return to the options menu.
       2. Change Major – Changes the student’s enrolled major to user input. After saving, the GUI should return to the options menu.
       3. Add Course – Registers the student to a new course. This prompts the user to search for a class to register. The GUI presents the user with a search by CRN or department key. From this point, a list is presented to the DBA to select the class to register. After saving, the GUI should return to the options menu.
       4. Drop Course – When this option is selected, the GUI should display all classes in which the student is currently registered by CRN and course name. Once a class is selected, the user must verify to drop the student from the selected class.
       5. Drop All – This option effectively unregisters the student from all classes (excepting completed courses) using their student ID. The user must verify to drop the student from all classes.

***View Student Record***

* + - 1. Administrators for a department will be able to view registration information and course history, including grades, of all students within that department.

***\*School administrators are unable to add/edit departments and only have access to the department they are an administrator for.***

***PROFESSOR FUNCTIONALITY***

***Professor Login***

* + 1. All professors at Spartanburg Technical College must be verified using a login and password issued by the Database Administrator as a representative of the school. On the basic login screen, the professor will input their username and password. The application should offer the professor a unique set of user interfaces distinct from those offered to the Database Administrator, the Administrator, and the Student views. The professor should have access to features that allow them to add, edit and remove courses they instruct, input course information, and modify each student’s grades within and only within the course in which the student is registered and the professor instructs.

***Adding a Course***

* + 1. The GUI should allow a professor to create new course information and save it to the database. If department, course number, and section number already exist, the GUI should prompt the administrator accordingly.
    2. The “Course Information” to be added is enumerated in section DBA 4.1.7. above.
    3. Every professor only belongs to one “Department” such as CSCI (for Computer Science) or INFO (Informatics). The professor can only create courses for the department to which they are assigned.
    4. The GUI should allow only allow the information for one course to be entered at a time.

***Deleting a Course***

* + 1. The administrator should be able to perform search queries for a course to be deleted by CRN or Course Name. Searching by course name should display a list of CRN and Course Names from which to select.
    2. Once a course is selected, the DBA will be prompted whether they are certain they want to delete the course. When confirmed, the course will be deleted from the database and all registered students will be dropped from the course.
    3. Only one course may be deleted at a time. Once the course has successfully been deleted, the user will be returned to the main menu.

***Editing a Course***

* + 1. The GUI should allow the professor to perform search queries to retrieve a course for the purpose of editing the Course Information (see section DBA 4.1.7.). The search should be conducted using either the CRN or course name, and the resulting list will be displayed by CRN with the course name.
    2. When the course has been selected, several options will allow the user to modify course information. Functions to modify each section except for the CRN of the Course Information will be made available. Selecting an option to modify will prompt the user for the updated information.
    3. The GUI should confirm the update and show the same options once the change is confirmed.

***Assigning Grades***

* + 1. The professor will be able to assign a grade to each student throughout the semester.
    2. At the end of a semester when final grades are due, the grade assigned by the professor at the time goes to the Student Record.

***\*Professors are unable to view student registration information and records, including grades aside from those assigned by the professor for a course that student was enrolled in.***

***STUDENT FUNCTIONALITY***

***Student Login***

* + 1. Students at Spartanburg Technical College must be verified using a login and password issued by the Student as a representative of the school. On the basic login screen, the school administrator will input their username and password. The application should offer the Students a unique set of user interfaces distinct from those offered to the Database Administrator, the Professors, and the School administrators views.

***See Current Courses***

* + 1. The GUI will display the current registered courses for the student. The information includes the department key, course number, section number, course name, professor, and the student’s grade.
    2. If the student has not registered for a course, “No Current Courses to Display” message will be presented to the user.
    3. The student will have a return function to return to the main menu.

***Modify Course Registration***

* + 1. The GUI will display current registered courses with the option to drop the course next to each. Only one course can be dropped at a time. The GUI will also display a function to allow the user to register if they wish, but if not, the function returns the user to the main menu.

***View Student Record***

* + 1. Once a course is complete, the course will automatically be sent to a table of “Student Record.” Courses withdrawn from while a semester is active will be added to the Student Record with a status of “Withdrawn.”
    2. The “Student Record” information for a student record includes:
       1. The record ID. This 8-character ID must be unique to identify the individual record.
       2. The student ID. This 7-character ID identifies the student associated with the record.
       3. The course number. The 4-character course number can only have one course name.
       4. The student’s grade in the associated course.
       5. The semester in which the student took the associated course.
       6. The year in which the student took the associated course.

***Register for Course***

* + 1. Student will be prompted for Department ID to search for a new course. If the department is not found, the user will be prompted to try again. Only one Department ID may be searched for at a time. This will display a list of all courses offered by the requested department.
    2. The returning display will include all information from “Course Information” outlined in section DBA 4.1.7. for the selected courses.
    3. If the student selects a course, it will be held until confirmation to register. Final confirmation will register the student for the course.
    4. Students may register for a class only once. If a student attempts to sign up for the same course twice, an error message should display stating “You have already registered for this course.”

# GUI Design

The following is the GUI design layout for the DBMS application. It will illustrate the elements of the GUI each type of user (Database Administrator, Administrator, Professor, Student) will encounter as they use the application.

***DATABASE ADMINISTRATOR***

On the basic Login screen, an option will be available in the lower right hand corner labeled “DBA Login.” The current user will be taken to another login screen that requires a USERNAME and PASSWORD, as well as a DATABASE ADMINISTRATOR ID NUMBER. This number is a 10 digit hexadecimal value.This number, when input, will be checked against a table to verify its legitimacy. This table only has 2 values in it as there will only be 2 individuals with this role.

After login, the DBA will have options to modify all students, professors, and prospective students tables. The DBA will have options available to modify all students' class registration, add/delete classes, modify all course information, and add/delete instructors.

Selecting “Modify Student Tables” will take the user to a new window with options to:

* *Add student* - The user is presented with prompts to enter first and last name, a student ID and a student classification (Freshman, Sophomore, Junior, Senior)
* *Delete student* - The user is presented with prompt to enter a student ID number to remove the student
* *Modify student information* -The user is presented with a prompt to enter a student ID number to select a student. Once entered, options to modify a student’s name and class will be provided.
* *Modify student registration* - The user is presented with a prompt to enter a student ID number. Once entered a screen showing all their classes will be presented. They can drop or add classes to the students registered classes.
* *Modify student grades* - The user is presented with a prompt to enter a student ID number. Once entered, a list of all current classes the student is taking will be returned. A drop box for each class containing all possible grade levels will be next to each course and can be changed (+A – F).

Selecting “Modify Professor Tables” will take the user to a new window with options to:

* *Add professor* - A prompt to enter professor information including first and last name, faculty ID, and department ID will be presented.
* *Delete professor* - A prompt to enter a professor's faculty ID will be presented.
* *Modify professor information* - A prompt to enter a professor's faculty ID will be presented. Once entered, prompts to change a professor first and last name and department will be presented.

Selecting “Modify School Administrator Tables” will take the user to a new window with options to:

* *Add professor* - A prompt to enter professor information including first and last name, faculty ID, and department ID will be presented.
* *Delete professor* - A prompt to enter a professor's faculty ID will be presented.
* *Modify professor information* - A prompt to enter a professor's faculty ID will be presented. Once entered, prompts to change a professor first and last name and department will be presented.

Selecting “Modify Course Tables” will take the user to a new window with options to:

* *Add professor* - A prompt to enter professor information including first and last name, faculty ID, and department ID will be presented.
* *Delete professor* - A prompt to enter a professor's faculty ID will be presented.
* *Modify professor information* - A prompt to enter a professor's faculty ID will be presented. Once entered, prompts to change a professor first and last name and department will be presented.

Selecting “Modify Department Tables” will take the user to a new window with options to:

* Add department - A prompt to enter the new department name and key will be presented.
* Delete department - A prompt to enter the department key to be deleted will be presented.
* Modify department - A prompt to enter the current department key to be edited will be presented. From there, either the department name or the department key can be altered.

Selecting “Enter Terminal Mode” will take the user to a new window where they will be prompted to input raw SQL commands to modify the database

***SCHOOL ADMINISTRATOR***

On the basic login screen, the administrator will input their USERNAME and PASSWORD to gain access to the system. They will have options to add students to a course and remove students from a course. When adding or removing a student from a course, the administrator will be prompted to input a student ID and a course ID. They will also be able to modify all offered courses. When logged in, the administrator will be presented with options to:

* *Modify Student Registration*
  + Once selected, the administrator will be prompted to input a student ID. Once entered, the administrator will be presented with a list of all the students currently registered courses
  + Next to each course a “Drop course” button will be available to select.
  + At the bottom, a button “Register student for course” once selected, a prompt to input course ID and course department will be presented. Once input and entered, the student will be registered for the course.
* *Modify Courses*
  + Once selected, the administrator will be prompted to input a student ID. When entered, the administrator will be presented with a list of all the students currently registered courses
  + Next to each course a “Drop course” button will be available to select.
  + At the bottom, a button “Register student for course” once selected, a prompt to input course ID and course department will be presented. Once input and entered, the student will be registered for the course.
* *Add or Remove a Professor*
  + “Add Professor” - A prompt to enter professor information including first and last name, faculty ID, and department ID will be presented.
  + “Delete Professor” -A prompt to enter a professor's faculty ID will be presented.
* *Add or Remove a Student*
  + “Add student” - The user is presented with prompts to enter first and last name, a student ID and a student classification (Freshman, Sophomore, Junior, Senior).
  + “Delete Student” - The user is presented with a prompt to enter a student ID number to remove the student.

Note: Administrators for departments will only be able to access the department for which they are an administrator for - e.g. Mathematics department administrators can access mathematics department courses, students and professors, but not Computer Science department courses, students and professors (and vice versa).

***PROFESSOR***

On the basic login screen, the professor will input their USERNAME and PASSWORD to login to the system. Once logged in, professors will have options to:

* *Add new course* - Once selected, a prompt to input course name, course description will be presented. The course department, and course instructor name will autofill the information of the current professor.
* *Change course information* - Once selected, a prompt to input a course ID will be presented. When input, the professor will be able to modify the name and description of the course with the input course ID
* *Update student grades* - Once selected, a prompt to input a course ID will be presented. When input, a list of all students will be generated who are enrolled in the specified course. Next to each student name, a drop box will be available to change the letter grade of the student for the course (+A - F).

***STUDENT***

On the basic login screen, the student will input their USERNAME and PASSWORD. When logged in, the student will have options to:

* *See current courses* - When selected, a list of all courses the student is registered for will be presented. Each course will have its course name, course ID number, course professor, and a grade for the student listed.
* *Modify course registration*
  + When selected, a list of all current courses will be presented with a button “Drop Course” next to each line. Selecting “Drop Course” will unregister the student from the course.
  + Below the list there will be a button labeled “Add course”. Selecting “Add course” will bring the user to a new screen asking for course department code. Entering the code will generate a drop down of all courses offered by that department. The student can navigate to the course they want to select.
  + The course selected will be loaded into a box with a button “Register for course” will be appear. When selected, the student will be registered for the course.

# Environmental Requirements

The development team expects that devices on campus will typically be used to access this application. Therefore, the application should run primarily on Windows OS. Ensuring functionality across other operating systems (MacOS, Linux) will not be necessary.

The application will be implemented using Java with the standard being JDK 1.8. Use of Java as the preferred programming language will allow for the functionality of the system to meet the needs of its users.

# Other Requirements

The ThisIsSpartanTech system is a stand-alone application. Interfacing with other applications is not necessary.

Cost of development of this application must not exceed $200,000. College budget for the current fiscal year does not allow for any additional incurred costs.

Documentation during the development of the application is necessary. Any changes or updates to this document and/or other documents will require thorough documentation about what the changes are and why they are made.

All faculty and students of Spartan Tech will receive correspondence when the application is deployed, including detailed instructions for use and resources for resolving issues with the application.

# Software Qualities

* *Ease of use* - This application will be accessed by users from a variety of backgrounds with different skill levels. Therefore, the user interface must be simple and easy to operate.
* *Correctness* - Since this application may be used by students to track degree progress, professors to input grades and monitor student performance, and administrators to make courses available to students, there cannot be error in performance of the application.
* *Performance* - The application should be efficient and take no longer than 10 seconds to display any option selected.
* *Robustness* - The system is expected to handle any errors during use (e.g. a professor tries to enter a grade of 1000/100).
* *Maintainability* - Routine maintenance will be necessary based on user feedback - therefore, the application should be easy to maintain. Code should be readable and well-written.
* *Timeliness* - Development of this system in a timely manner is crucial to meet the needs of users. Therefore, little deviation from the outlined timeline will be allowed.
* *Size* - Size should not be of concern, as Spartan Tech expects that resources accessible through this application will be accessed through school-owned devices. Therefore, sizing concerns are expected to be the full responsibility of STC.
* *Understandability* - Code should be easy to comprehend; in the case that future developers access source code for the application, they should be able to understand and work with the code.

# Time Schedule

The internship program at Spartanburg Technical College is a semester-long experience. As a result, STC is pushing the development of its ThisIsSpartaTech program. The schedule is as follows:

## Development of the application is expected to be completed by November 17th, 2022.

## Testing should be completed by no later than November 29th, 2022.

## Final changes should be completed no later than December 6th, 2022.

# Potential Risks

* *Non-intuitive User Interface -* Due to the number of views and constraints built into the ThisIsSpartaTech system, users may find the final product difficult to navigate.
* *Complexity of System -* This system is designed for Spartanburg Tech to fulfill numerous functional requirements. Implementing such a complex system will be risky for a small team to build properly.
* *Limited Timeline -* The outlined schedule for this project is aggressive and may result in an inadequate product for Spartan Tech.
* *Database Efficiency -* With the volume of courses, students, and faculty encompassed by Spartanburg Technical College, there is some concern regarding the efficiency of the system as a whole. During registration periods, the system will be experiencing a high amount of traffic, which may lead to system failures or errors.

# Future Changes

* *Seat availability for a class* – Each unique course number and section number can have functionality that counts the number of students already registered for a class. When the maximum number of students set by the professor is reached, the student would be prompted accordingly.
* *Waitlist* – Based on a potential seat availability function, students could be provided with a dedicated waitlist of 4 if a class is currently full.
* *More sophisticated user interface* - Based on feedback from users, an improved GUI with a more modern appearance and UX design elements may be designed to enhance the user experience.
* *Marketing to more universities* - If feedback from implementation at Spartanburg Technical College is successful, STC will look into expansion to more universities in the area to maintain consistency for transfer students.
* *Improved compatibility* - iOS will be the initial focus, followed by other operating systems as requested.
* *Mobile app* - This expansion will allow students, professors, and administrators to access the ThisIsSpartanTech application from their mobile devices.

# Glossary

* **Spartanburg Technical College (STC)** - Also known as Spartan Tech, this is the college contracting the ThisIsSpartan system. Their IT internship program will be building the proof of concept for ThisIsSpartanTech.
* **ThisIsSpartanTech** - The system under development; a software or web application that tracks student registration and courses offered by the college.
* **User** - User is a general term for an individual using the ThisIsSpartaTech system as a Database Administrator, an Administer, a Professor, or a Student.
* **Administrator** - An individual who works as a member of the administrative staff of Spartanburg Technical College.
* **Database Administrator (DBA) -** An individual working for STC in the IT department that oversees the ThisIsSpartanTech system.
* **Course Registration Number (CRN)** – A unique five character value used to designate an individual class for registration.