Software Requirements Specification

for

Optimized Course Schedule Generator

Version 1.0 approved

Prepared by

Brett M. Stricker
Doyle D. Bigelow
Diego R. Draguicevich

OCSG

April 16, 2021

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
First Draft	2021-03-03	Initial Draft	0.1
Revised Draft	2021-03-18	Cleanup	0.2
First Release	2021-03-25	Finalized	1.0

1. Introduction

1.1 Purpose

The Optimized Course Schedule Generator Version 1.0 is a software tool to assist universities in creating the master schedule for the semester. Schools have to manage thousands of students, and hundreds of teachers and classrooms. The software will generate a master schedule requiring minimal edits decreasing the labor required by a university.

1.2 Intended Audience and Reading Suggestions

This document is intended to be used in the development, deployment, and maintenance of the OCSG System. Readers might include system developers, system testers, salesmen and other marketing personnel, and university staff deploying this system. The document is best read in sequential order, but jumping to any section of interest based on the table of contents is acceptable. Glossary contains information about acronyms and other key concepts.

1.3 Product Scope

The aim of this project is to develop a software project that can be utilized by schools and universities to generate a master schedule for class times and locations, easing the jobs of school administration. This software should be able to resolve schedule conflicts including but not exclusive to, classroom timing, classroom location, teacher workload/subject expertise, and classroom student capacity. The software should allow students to easily follow their major flowcharts. This project will be considered a success if less than 10% of the generated schedule must be changed manually.

1.4 References

[1] "Advanced Scheduling," Advanced Scheduling - SchoolInsight - Common Goal Systems, Inc. [Online]. Available: https://www.teacherease.com/advancedscheduling.aspx. [Accessed: 24-Feb-2021].

[2] U. S. A. Scheduler, "Home," Best Scheduling Software - School Master Schedule Software. [Online]. Available: https://usascheduler.com/index.php/usa-scheduler/class-schedule-maker. [Accessed: 24-Feb-2021].

2. Overall Description

2.1 Product Perspective

The OSCG is a master schedule generating system built for educational institutions. It is designed as a replacement for existing systems or to be implemented as a new system.

2.2 Product Functions

- View Database Table: Allows user to view any of the following tables: course section, flowcharts, students, teachers, courses, time slots, buildings and classrooms
- Add to Database: Allows user to add to any of the following tables: course section, flowcharts, students, teachers, courses, time slots, buildings and classrooms
- Remove from Database: Allows user to remove any of the following tables: course section, flowcharts, students, teachers, courses, time slots, buildings and classrooms
- Generate Master Schedule: Creates the master schedule
- Modify Database Table: Allows user to view modify any of the following tables: course section, flowcharts, students, teachers, courses, time slots, buildings, and classrooms
- Login: Prompts user to input user login in credentials to determine if user is to be given access to the system and if so what the user will be given access to

2.3 User Classes and Characteristics

This software has a total of 4 different User Classes; Students, Teachers, Counselors, and Department Chairs.

- Students: Will use the system to construct their class schedules. Students are low priority users as they are not the targeted user for the system but extra development for this user class is unnecessary as they will end up using modules essential to other user classes
- Teachers: Will use the system to view their assigned class schedules and student rosters. Teachers are low priority as they are not the targeted user for the system. Viewing students relative to their classes is primarily a feature for teachers and counselors so in turn is a low priority feature
- Counselors: Will manage adding and removing students
- Department Chairs: Will be the overseers of the schedule. Will request for the schedule to be generated. Will use the system to modify Master Schedule, flowcharts, students, teachers, courses, time slots, buildings and classrooms

2.4 Operating Environment

- The user frontend shall be compatible with the Windows^{TM} Operating System
- MySQL RDBMS
- Workplace devices running Windows^{TM}
- Backend shall be compatible with any system running MySQL and Python 2.7.5
- Minimum client hardware requirements: 2-Core 2.0 GHz Processor, 4 GB Ram, 300 MB storage space, Broadband Internet connection
- Minimum server hardware requirements: 4-Core 3.0 GHz Processor, 8 GB Ram, 5 GB storage space

2.5 Design and Implementation Constraints

- Generation of the schedule shall take no more than two minutes from the time the user presses the “Generate Master Schedule” button
- must be secure so that unauthorized modification can not happen
- During operation the system shall hold no more than 4GB of RAM
- During any given second of system operation the system shall hold no more than 90% of all processing time across all logical cores
- During any given second of system operation on the user’s personal computer the system shall hold no more than 50% of all processing time across all logical cores
- System shall be developed using Python 2.7.5

- Database Systems shall be designed in MySQL Community Server 8.0.23
- System shall communicate to the database using MySQL Connector/Python 8.0

2.6 User Documentation

- Web based user manual that has documented step based tutorials on how to perform each function
- A support email to ask technical questions

2.7 Assumptions and Dependencies

- The server that the system is installed on meets minimum server hardware requirements
- The user PC that the system is installed on meets minimum client hardware requirements
- The system has been installed correctly and has not been modified
- Server and user PC are connected via TCP/IP

3. External Interface Requirements

3.1 User Interfaces

- UI shall have tabs for view, add, remove, modify, and generate
- The tabs will be displayed at the top of the UI
- Tabs will be displayed based on user privileges
- The default tab after login will be view
- Fields in the different tabs are a combination of drop boxes, text fields, and dropdown calendars
- View tab is also used for displaying the master schedule
- System messages shall be sent via pop-up boxes
- User confirmation shall be via buttons

3.2 System Interfaces

The client side system shall run on the Windows^{TM} Operating System API. The server side system shall run on any operating system with MySQL Community Server 8.0.23, Python 2.7.5, and MySQL Connector/Python 8.0 installed. The server and user PC shall communicate over a TCP/IP interface. Data shall be encrypted using RSA before being transmitted.

4. System Features

4.1 View Database Table

4.1.1 Description and Priority

A user can view a list of courses, sorting by subject, location, timeslot, teachers, course sections, and flowchart compatibility. Since this is necessary for all other functionality except for Add to MySQL Table, it is second priority.

4.1.2 Stimulus/Response Sequences

1. User logs into their system interface with their credentials
2. System verifies user credentials
3. User specifies their search criteria via the appropriate fields
4. User requests search results
5. System returns search results
6. User may repeat steps 3-5 as desired
7. User requests system logout
8. System logs user out

4.1.3 Functional Requirements

REQ-1: The system can display a list of courses based on search results from any specified table column and filter provided the user has permissions

REQ-2: The system can display a list of students based on search results from any specified table column and filter provided the user has permissions

REQ-3: The system can display a list of teachers based on search results from any specified table column and filter provided the user has permissions

REQ-4: The system can display a list of buildings and classrooms based on search results from any specified table column and filter provided the user has permissions

REQ-5: The system can display a list of major flowcharts based on search results from any specified table column and filter provided the user has permissions

REQ-6: The system can display a list of course sections based on search results from any specified table column and filter provided the user has permissions

REQ-7: The system can display a list of timeslots based on search results from any specified table column and filter provided the user has permissions

4.2 Add to Database

4.2.1 Description and Priority

A user can insert data into the following tables: courses, course sections, buildings and classrooms, timeslots, students, teachers, and flowcharts. Since initializing the school requires this functionality it is essential for all other functionality it is first priority.

4.2.2 Stimulus/Response Sequences

1. User logs into their system interface with their credentials
2. System verifies user credentials
3. User selects to add data
4. User specifies new data via the appropriate fields
5. User requests the new data to be added
6. System indicates data is being sent to the server
7. System sends data to the server to be stored
8. System indicates new data has been stored
9. User may repeat steps 3-8 as desired
10. User requests logout
11. System logs user out

4.2.3 Functional Requirements

REQ-1: The system can add courses into the MySQL Courses Table

REQ-2: The system can add courses sections into the MySQL Course Sections Table

REQ-3: The system can add buildings and classrooms into the MySQL Buildings and Classrooms Table
REQ-4: The system can add time slots into the MySQL Time Slots Table
REQ-5: The system can add students into the MySQL Students Table
REQ-6: The system can add teachers into the MySQL Teachers Table
REQ-7: The system can add flowcharts into the MySQL Flowcharts Table

4.3 Remove from Database

4.3.1 Description and Priority

A user can remove data from the following tables: courses, course sections, buildings and classrooms, timeslots, students, teachers, and flowcharts. Since removing data is essential for correcting incorrect inputs it is essential for generating the schedule so it is third priority.

4.3.2 Stimulus/Response Sequences

1. User logs into their system interface with their credentials
2. System verifies user credentials
3. User selects to remove data
4. User specifies data to remove via the appropriate fields
5. User requests the data to be removed
6. System displays data that will be removed
7. User confirms the data that will be removed from the database
8. System sends data to the server to be stored
9. System indicates data has been deleted
10. User may repeat steps 3-9 as desired
11. User requests logout
12. System logs user out

4.3.3 Functional Requirements

REQ-1: The system can remove courses from the MySQL Courses Table
REQ-2: The system can remove courses sections from the MySQL Courses Sections Table
REQ-3: The system can remove buildings and classrooms from the MySQL Buildings and Classrooms Table
REQ-4: The system can remove time slots from the MySQL Time Slots Table
REQ-5: The system can remove students from the MySQL Students Table
REQ-6: The system can remove teachers from the MySQL Teachers Table
REQ-7: The system can remove flowcharts from the MySQL Flowcharts Table

4.4 Generate Master Schedule

4.1.1 Description and Priority

Will take all input factors classrooms, timeslots, course sections, projected student enrollment, teacher workload, courses a teacher is able to teach and create a master schedule(s) that fit all criteria. Since errors are permitted and thus must be corrected, add, remove, and view are all prerequisites so it is fourth priority.

4.4.2 Stimulus/Response Sequences

1. User logs into their system interface with their credentials
2. System verifies user credentials

3. User selects to generate a schedule
4. User requests a schedule be generated
5. System informs the user that a master schedule has been generated
6. System displays master schedule
7. User requests logout
8. System logs user out

4.4.3 Functional Requirements

REQ-1: The system will have the ability to generate a Master Schedule for a school, based on the data found in the database.
REQ-2: The generated master schedule shall enable 98% of students to take all their courses that semester.
REQ-3: The generation shall create course sections based on course enrollment and classroom timeslots capacities.

4.5 Modify Database Table

4.5.1 Description and Priority

A user can modify data in the following tables: courses, course sections, buildings and classrooms, timeslots, students, teachers, and flowcharts. Since modifying tables eases use but is not critical to system function it is fifth priority.

4.5.2 Stimulus/Response Sequences

1. User logs into their system interface with their credentials
2. System verifies user credentials
3. User selects to modify data
4. User specifies the data to be modified via appropriate fields
5. User requests the data be modified
6. System displays data that will be modified
7. User confirms the data that will be modified from the database
8. System directs server to modify data
9. System indicates data has been modified
10. User may repeat steps 3-9 as desired
11. User requests logout
12. System logs user out

4.5.3 Functional Requirements

REQ-1: The system can modify data in the MySQL Courses Table
REQ-2: The system can modify data in the MySQL Course Sections Table
REQ-3: The system can modify data in the MySQL Buildings and Classrooms Table
REQ-4: The system can modify data in the MySQL Time Slots Table
REQ-5: The system can modify data in the MySQL Students Table
REQ-6: The system can modify data in the MySQL Teachers Table
REQ-7: The system can modify data in the MySQL Flowcharts Table

4.6 Login

4.5.1 Description and Priority

The login system allows users with the appropriate authorizations access to the system. This component of the system allows users remote access to the system securely. Because this is not a critical feature for the system's primary purpose, it is sixth priority.

4.5.2 Stimulus/Response Sequences

1. User initializes the user interface
2. User enters a username and password
3. User presses the login button
4. System validates the username and password
5. System assigns the appropriate permissions and displays the correct UI for the user's permission level

4.5.3 Functional Requirements

REQ-1: The system can authenticate user credentials by comparing to stored information on the server

REQ-2: The system can reject incorrect credentials

REQ-3: The system can grant authenticated users the appropriate permissions corresponding to their authorization level

5. Other Nonfunctional Requirements

5.1 Performance Requirements

- The system shall generate the Master Schedule within no more than three minutes of the user requesting a Master Schedule
- During operation the system shall hold no more than 4GB of RAM on the client side
- During any given second of system operation the system shall hold no more than 90% of all processing time across all logical cores
- During any given second of system operation on the user's personal computer the system shall hold no more than 50% of all processing time across all logical cores.
- The master schedule generation shall take no more than 3 minutes to generate a schedule from the time the user requests a schedule

5.2 Safety Requirements

- All tables must maintain atomicity during transactions
- Users shall not receive permissions without providing login credentials and shall not receive more or less than the specified permissions for their associated credentials
- User login credentials shall not be exposed during login, and shall be made available to view/add/remove/modify only to administrators

5.3 Security Requirements

- The system shall not permit modification and/or viewing of data without the appropriate permissions for modifying and/or viewing said data
- The system shall store all data in encrypted form
- Authorization levels are as shown in table 5.3.1

5.3.1 Authorization Levels

	Student	Teacher	Administrator
View Courses	X	X	X
View Major Flowcharts	X	X	X
View Course Sections	X	X	X
View Students		X	X
View Teachers		X	X
View Building/Classrooms			X
View Timeslots			X
Add/Remove/Modify MySQL Table			X
Generate Schedule			X
View/Add/Remove/Modify Users			X

5.4 Software Quality Attributes

- The system should be easily maintainable across semesters
- The system shall be portable across all systems that meet the requirements
- The system shall have minimal crashes
- The system shall be available anytime other than when under maintenance
- The system shall be easily maintained through the system's server
- The system shall be able to pass a specified testing benchmark created by the developers
- The system shall be usable with minimal training

5.5 Business Rules

- Users are not permitted more than the minimum necessary access to system functions and data
- Access level is determined by the Users' administration
- System maintenance shall not be performed by anyone not explicitly given permission by OCSG personnel

6. Other Requirements

- Customers shall not redistribute any system component for any reason
- Support shall not be offered to any customer who has modified their system
- Any modified system is not guaranteed by the OCSG team

Appendix A: Glossary

- "OCSG" refers to the Optimised Course Schedule Generator
- Master Schedule - All course sections that exist in a semester
- "Permissions" refers to the contents of table 5.3.1
- "Priority" is listed in descending importance, so first priority is highest, then second, third, and so on
- "Minimum Hardware Requirements" refers to the specifications outlined in section 2.4
- "UI" refers to the User Interface