

CS 455 - Software Engineering  
Green Team  
Software Requirements Specification Document

Ricky Moore  
Elizabeth McClellan  
Hongbin Yu  
James Reatherford  
Ankur Patel  
Wenhao Wang

University of North Alabama

*2013/03/02*

## Revisions

Version	Primary Author(s)	Description of Version	Date Completed
(initial 1.0)	Elizabeth McClellan Ricky Moore	First Draft	2013/03/02

## Review, Agreement, and Approval

### Specification Document Review

SQA Team Member	Signature	Date
Hongbin Yu		
James Reatherford		

### Specification Document Agreement

Development Team Member	Signature	Date
Ricky Moore		
Elizabeth McClellan		
Hongbin Yu		
James Reatherford		
Ankur Patel		
Wen Hao Wang		

### Specification Document Approval

Client Representative	Signature	Date
Dr. Patricia Roden		

## Contents

1. Introduction .....	4
1.1 Purpose of this document.....	4
1.2 Overview .....	4
2. General Description.....	5
2.1 Functionality .....	5
2.2 Constraints .....	5
2.3 Limitations .....	#
3. Functionality Requirements and Constraints.....	#
3.1 Faculty Functionality.....	#
3.2 Admin Functionality .....	#
3.3 Schedule Creation.....	#
4. Other Requirements and Constraints .....	#
4.1 Input Files.....	#
4.2 Login Information.....	#
4.3 Misc Requirements.....	#
5. Application Output .....	#
5.1 Graphical Representation of Schedule.....	#
5.2 Excel Representation of Schedule .....	#
6. User Interface.....	#
7. Appendices .....	#

# 1. Introduction

## 1.1 – Purpose of this document

The purpose of this document is to detail each requirement of the departmental scheduling system, and to serve as a reference from which the software will be developed by the team members. After approval, the document will also serve as an agreement between the development team and the client representative with regard to the requirements of the final version of the software.

## 1.2 – Overview

The result of this project shall be a fully functional version of the departmental scheduling application. The application shall be deemed acceptable once the following requirements are met:

- (1) The application is web-based, and supports the creation and viewing of a university academic department's semester schedule of classes to the satisfaction of the client representative.
- (2) The application is generically designed in such a manner as to be usable by any department within the University of North Alabama.
- (3) All functionality defined in this document is included in the application.
- (4) All requirements and constraints listed in this document are adhered to by the application.
- (5) The user interface is developed in a similar style to current University of North Alabama webpages, the specific design of which is subject to approval of the client representative.

## 2. General Description

### 2.1 – Functionality

In addition to satisfying the aforementioned requirements, the application will provide the following functionality:

- (1) The application will aid in schedule management by creating a schedule given information about available time slots, available rooms, and information regarding the courses to be scheduled.
- (2) The application will have the ability to:
  - a. Generate a graphical representation of the schedule which will be viewable by outside users.
  - b. Generate an Excel spreadsheet representation of the schedule.
- (3) The application will allow faculty members of the department to:
  - a. View the list of desired courses to be offered by the department.
  - b. Be given the ability to indicate the course(s) and desired time interval(s) they wish to teach for the semester.
- (4) The application will allow the administration of the department to:
  - a. Create and manage faculty member accounts.
  - b. Input information necessary for the application to create the schedule.
  - c. Prompt the application to create the schedule once the necessary information has been supplied.
  - d. View and edit the schedule after creation if desired.

### 2.2 – Constraints

The application is subject to the following constraints:

- (1) When creating the schedule:
  - a. 400 level courses take precedence over any lower level course.
  - b. 300 level courses take precedence over any lower level course.
  - c. 200 level courses take precedence over any lower level course.
  - d. 100 level courses take precedence over any lower level course.
  - e. No level 300/400 level courses should be scheduled for the same day and time unless one is a prerequisite of the other.

### 2.3 – Limitations

The application has the following limitations:

- (1) The application will not create a schedule until the administrator requests it to do so.

- (2) A conflict is defined as occurring when two faculty members input preferences that contradict one another, or when scheduling two classes would require contradicting one or more constraints or requirements. The application does not attempt to prevent conflicts. It will attempt to resolve conflicts once they occur, but if a conflict cannot be resolved, the application will continue with schedule generation and present these conflicts to the administrator for manual review.
- (3) If any information from input files used to create the schedule is changed, the burden of generating an updated schedule rests with the administration.
- (4) If an administrator attempts to create a schedule with incorrect/contradictory information, the software will not create a schedule and will instead present a meaningful error message explaining why a schedule cannot be created.
- (5) If an administrator attempts to create a schedule with incomplete information, the software will not create a schedule and will instead prompt the administrator to input the correct file or input the information from within the application.

## 3. Functionality Requirements and Constraints

### 3.1 – Faculty Functionality

#### 3.1.1 – Information Provided to Faculty

- (1) Faculty members will be provided with the following information:
  - a. A list of classes to be offered by the department.
  - b. The class preferences they have currently specified.
  - c. Information about the minimum hours they are required to teach and the current number of hours they have input preferences for.

#### 3.1.2 – Preference Input

- (1) The program will allow faculty members to indicate the course(s) they wish to teach for that semester and desired time period(s) for each course.
- (2) Regarding faculty preferences, time periods (morning, mid-day, and evening) are defined as follows:
  - a. Any classes starting before 11:00 am, not including those starting at 11:00 am, are considered morning classes.
  - b. Any classes from 11:00 am to 2:00 pm, including classes that start at 11:00 am and 2:00 pm, are considered mid-day classes.
  - c. Any classes that start after 2:00 pm, but not including classes starting at 2:00 pm, are considered evening classes.
- (3) Preferences will be prioritized either by seniority or by the order of submission. The priority setting is controlled by the administrator. If seniority prioritization is chosen, faculty members with the same seniority will have conflicting preferences decided by order of submission.

### 3.2 – Admin Functionality

- (1) Administrators, in addition to the following functionality, will also have all functionality of faculty members.

#### 3.2.1 – Input

- (1) The application will accept the following file input from the administrator:
  - a. Possible class times to schedule.
  - b. Information regarding the available rooms.
  - c. A list of courses to be scheduled.
  - d. A list of possible conflict times with courses from other departments, including the usual times those classes are offered.

- e. A list of courses with their prerequisites within the department
  - f. Information regarding faculty member accounts
- (2) By default, the program will create empty files (in a directory specified in the user manual) for each type of input from (1). The administrator can browse for a file to overwrite the current file stored in the system, or may edit the file from within the browser (see 3.2.2 below).
  - (3) The format for each file is specified in appendix C. Other information regarding input files is found in section 4.1
  - (4) If a fault is found in the file when submitting or uploading new input, the application will notify the user with a meaningful error message pertaining to the mistake. A schedule cannot be generated unless the input is free of faults.
  - (5) The administrator should also be able to set the method of handling faculty preference conflicts. The two available methods of resolving conflicts are via seniority or via order of submission.

### **3.2.2 – Editing Within Application**

- (1) Information from the input files will be made available to the administrator, and this information will be editable from within the application browser.
- (2) Changes made to files from within the browser will be reflected in the files stored in the system.
- (3) Input verification will be done upon submitting edits to input files, and all constraints regarding correct and fault-free input apply to edited files.

### **3.2.3 – Schedule Editing After Creation**

- (1) The administrator may opt to edit the schedule after creation in order to resolve unhandled conflicts, or for manual refinement. When editing, the admin should:
  - a. Be presented with a table of entries, and should be able to sort the table by room number, course, instructor, or time slot.
  - b. Be able to edit any individual entry by choosing valid options from dropdown boxes.
    - These options are determined by the values specified in the input files, and exclude any options which would result in a conflict, or which would require the application to contradict one or more specifications or requirements.
- (2) Changes made in this manner will be automatically reflected in both the graphical and excel representations of the schedule.



### 3.3 – Schedule Creation

- (1) The application will attempt to create a schedule by matching the following for every available faculty preference:
  - the faculty member
  - the course they have requested
  - a room with the capacity to hold the class
  - an available time slot
- (2) When attempting to create the schedule, the application will generate a list of classes that were unable to be scheduled along with the possible reason(s) for its inability to be scheduled. This information will be presented to the administrator for manual review.
- (3) If there is a conflict in the preferences provided by faculty members, it will be handled according to the priority rules mentioned in section 3.1. If a resolution cannot be made by the application, the course in question will not be scheduled and a note will be presented to the administrator after schedule creation for manual review.
- (4) If a class that has already been scheduled is preempted by another class, the application will attempt to relocate the first class to the nearest available time slot. The application will not schedule a class more than one time period (Morning/Mid-day/Evening) away from the faculty member's preference.
  - For instance, a faculty preferring to teach a class mid-day may have the class moved to morning or evening. A faculty preferring to teach a class in the evening may get moved to mid-day, but will not be moved to morning.
- (5) If a class has multiple sections, the application will attempt to disperse the scheduling of these sections throughout different days and times.

## 4. Other Requirements and Constraints

### 4.1 – Input Files

«NEEDS REVIEW AND POSSIBLE REVISION BY SQA»

- (1) Appendix C contains a list of valid and invalid file examples, and explanations for the formatting of each input type. The format specified therein must be followed exactly. Any entry which does not conform to this format will be considered a fault, and will be handled accordingly.
- (2) The contents of files are case sensitive.
- (3) An end of line marks the end of a data record.
- (4) Input will be read until EOF.
- (5) One line per course in course list file.
- (6) No empty lines are allowed in input files.
- (7) Names for faculty members must be input in the format “Lastname, Firstname”.
- (8) The combined number sections for a class cannot be zero.
- (9) Faculty years of service can be expressed either with a whole number, or as a decimal number indicating some number of full years and a half year. E.g. – 5.5 years is valid. 5.25 years is invalid.

### 4.2 – Login Information

- (1) The application will by default have two administrator accounts. Information on how to log on using these accounts will be included in the user manual.
- (2) Administrators will be required to change their username and password upon first login.
- (3) Faculty account usernames will be the email provided by the administrator in the faculty account input file, and must be unique.
- (4) Faculty accounts will have a default password when the account is created. The admin is responsible for giving faculty members this password. This default password can be found in the user manual.
- (5) Faculty members will be required to change their password upon first login.
- (6) «NEED PROVISION ABOUT HOW ACCOUNT ADDS/EDITS/DELETES SHOULD BE HANDLED»
- (7) «NEED PROVISION ABOUT HOW USERNAMES/PASSWORDS ARE RETRIEVED»
- (8) Passwords:
  - a. must be 6 to 9 characters long, inclusive.
  - b. Passwords are case sensitive.
  - c. Passwords must begin with an alphabetic character, must contain at least one numeric character, and must contain at least one of the following characters: an exclamation point (!), a question mark (?), or a comma (,).

### **4.3 – Misc Requirements**

- (1) The application will work with the three most popular browsers as of January 2013:  
Google Chrome, Firefox, and Internet Explorer
- (2) <<NEEDS PROVISION REGARDING SECURITY>>

## 5. Application Output

### 3.1 – Graphical Representation of Schedule

- (1) Should be viewable by any outside user.
- (2) Design to be determined. Will list all scheduled classes, along with room number, time slot, and instructor name.

<<Include picture of example schedule>>

### 3.2 – Excel Representation of Schedule

- (1) The application will also generate an excel representation of the schedule when the administrator prompts the application to create the schedule. Instructions on locating this file can be found in the user manual.
- (2) Any user can request a download of this file from the front page.

<<Include picture of example schedule>>

## 6. User Interface

<<Insert mockups of user interface with explanations>>

## 7. Appendices

### **A – Program Flowchart**

<<Insert program flowchart>>

### **B – Example Use-Case Model**

<<Insert example use-case model>>

### **C – Definition of File Formats with Examples**

<<Insert example correct and incorrect file formats with explanations>>