Software Requirements Specification

for

Strength & Conditioning Online Report System, Release 1.0

Version 1.0 approved

Prepared by

Anderson John, Joshua Norris, Ryan Keown
Preston Flake, Keijiro Suzuki

September 26, 2007

Table of Contents

1. Introduction

1.1. Overview

The purpose of this software is to allow the Wichita Falls ISD strength and conditioning coordinator and the Wichita Falls ISD coaches to conveniently track the progress of athletes. The program will provide an interface to easily enter data for each student through a form. The data will then be stored in a remote database and the software will facilitate report generation.

1.2. Objective

The software will simplify the data entry and collection of the current Strength and Conditioning Statistics of the Wichita Falls ISD athletes. The software will also create a strength conditioning report card for individual athletes that can be printed or exported to PDF format and emailed to their parents.

1.3. Specific Goals

1. Create an interface, including forms, to simplify the data entry by the coaches and the coordinator.
2. Create an interface to generate reports on the progress of students for display or printing.
3. Simplify the collection and consolidation of data from the coaches by the coordinator.
4. Create a database to allow the sorting of student data for the purpose of generating reports, and to maintain the students' statistics at a single site.
5. Restrict coaches to viewing information pertaining only to their students, and allow the coordinator to view information from all the coaches.

2. Overall Description

2.1. Product Perspective

The Strength & Conditioning Online Report System (SCORS) is a new system that replaces the current manual task of monitoring the conditioning progress for individual athletes in the Wichita Falls Independent School District. This product is expected to evolve with the development process.

2.2. User Classes and Characteristics

Model	
Conditioning Record Keeping (M1)	
Actors	
Administrators (A1)	
Users (Coaches) (A2)	
System (A3)	
Use cases	
New Athlete Entry (UC1)	
Requirements	
1	System must store athlete records and be capable of retrieving and displaying these records.

Actor	
Administrators (A1)	
Description	
Unrestricted viewers of athlete records. They can modify, add, or delete but should only view athlete data most of the time. Administrators can post information to start page of individual schools, such as deadlines and other reminders. They can create / edit logon information. Administrators have access to the system preferences where they can view, enable or disable the system log.	
Goals	
0	Keep track of the conditioning results of WFISD athletes.
1	Add/Modify login information.
2	Add/Modify Deadlines for data entry.

Actor	
Users (Coaches) (A2)	
Description	
Coaches from the junior high and high schools in the WFISD. Limited to viewing, modifying, and adding athlete entries from their own school.	
Goals	
0	To record previously obtained conditioning results.
1	To add new athlete entries.
2	Review athlete conditioning results and print reports.

Actor	
System (A3)	
Description	
The system comprises of the client software, the main database, other data stores, and the physical hardware.	
Goals	
1	Manage user logons and evaluate access permissions.
2	Perform passive data maintenance, such as classification increments and relocation/removal of old unneeded data entries.
3	Compile reports/graphs of teams, individuals, most improved, ect. to be viewed / printed.

Use case	
New Athlete Entry - (Use Case 1)	
Pre-conditions	
Coaches must obtain the conditioning results before the data can be entered into the system. List of athletes must be assembled with corresponding required information.	
Post-conditions	
A new athlete entry will have been added into the database, current records for this student will have been entered.	
Active actors	
Users (Coaches)	
System	
Details	
Priority	1
Level	User
Complexity	Medium
Status	Created
Implementation	Planned

Flow of events	
1	system prompts the user to log on.
2	user enters school wide login information.
3	system verifies the logon information and loads school start page.
4	system displays athlete data entry / backup deadlines.
5	user selects the classification and sport of an athlete.
6	user searches under last name, first name or the student ID for a particular athlete.
7	system displays results of search.
8	user does not find athlete listed.
9	user creates new athlete entry.
10	user enters classification, full name, student id, and gender.
11	user selects to save the new athlete entry.
12	system validates the entered information.
13	system responds with result of save {pass | fail}.
14	user enters any previously recorded conditioning results into the athletes dataset.
15	user logs off the system.

2.3. Use Case Scenarios:

Coach Creates New Student Record

Main Success Scenario:
1. Coach logs into System
2. Coach clicks Add New Student
3. Coach chooses his/her school
4. Coach chooses student's sport
5. Coach chooses student's gender
6. Coach chooses student's grade
7. Coach types in student's name
8. Coach chooses test results to enter
9. Coach enters test results
10. Coach presses done
11. System takes information and adds student to the database
12. System sends confirmation to screen

Extensions:
12a. Student already exists
    1. System sends error message
    2. Coach can overwrite current record

Mr. Hadderton (system admin) Checks Security Logs

Main Success Scenario:
1. Mr. Hadderton logs into system
2. Mr. Hadderton clicks Preferences tab
3. Mr. Hadderton clicks Security Log
4. System compiles all logons for the day and displays
5. Mr. Hadderton views displayed logins
6. Mr. Hadderton logs out of system

Extensions:
1a. Failed logon
    1. System sends error message
    2. Retry login
5a. Print displayed information
    1. Click on file/print and page will be printed

2.4. Operating Environment

The product shall operate on the Microsoft Windows 2000 operating system or any later versions. SCORS will also require a networked environment for operation. All computer systems with this software should have a working network connection in order to connect to the remote database. The product will only operate on systems equipped with the Microsoft .Net Framework 2.0. An option to install the .Net Framework will be provided at installation.

2.5. Design and Implementation Constraints

Development of this release is restricted to a 3 month period. This limits any possible time for evolution of the system during this release. Due to a delay in permission to implement the database with an SQL server over the Wichita Falls ISD network, the developers might be restricted to implementing the data storage with a shared Microsoft Access file.

2.6. User Documentation

The system shall provide an online help in HTML format. A PDF tutorial and manual for the product will also be provided.

2.7. Assumptions and Dependencies

The quality of data in the reports depends heavily on the data input from the individual coaches.

3. System Requirements

3.1. Functional Requirements

The following functionality will be provided by the system.

3.1.1. Add/Edit Athlete: Demographic data will be provided by the user and the system will perform necessary processing before writing the data to the database. For example, by selecting the "female" radio button, the system will convert that selection to the appropriate gender code before writing to the database. After being processed, data entered will be stored in the main athletes table in the database. The ability to search, sort, and delete will be provided.

3.1.2. Add/Edit Student Fitness Test: The system will allow users to enter student conditioning test results in a structured and efficient manner. The following functionality is included:

- Provide a quick search function by student ID or name.
- Make available an edit function to amend previously entered test results.
- The 'load athletes' function will load athletes from the database but only those belonging to the coaches' school. This function will provide the user with the possible athletes they are able to enter data for.

3.1.3. Admin Services: Administrative services will be available to privileged users. The functionality includes:

- **System Backup/Restore:** A backup function will be available to allow an administrative user to backup or restore the entire system. A reminder to backup will be proved to administrators at login if a set interval passes since the last backup. An option to enable or disable this feature will be included.
- **Add/Edit/Delete Users:** The security tab on the preferences GUI will allow the system admin to add, edit or delete users (coaches). The information required for user will include name, institution, user name, user password, and user class.
- **View/Clear log file:** A log of certain system events will be taken if this feature is enabled. This log will have a finite length and an option to clear the log.

3.1.4. Reporting: Reporting functions will be made available from the reports menu. Reports will be generated by processing data available in the system database. All reports produced will be equipped with an export function. Reports will be exported to Microsoft Excel or PDF format. Available reports will include:

- **General Student Information:** This report allows the demographic information of individual athletes to be viewed or printed.
- **Individual Speed, Strength and Conditioning Report Card:** This report will indicate the current test results and required areas of improvement needed for individual students.
- **Speed, Strength, and Conditioning Test Results by School:** This report generates the test results of students by school. Coached will only be able to view students enrolled at their particular facility. The report will also include additional filtering by gender, grade, and sporting activity.
- **Student Rankings:** This report ranks the students either within their school or among all students in the database.

3.1.5. Archive Students: This function will allow a coach to move a student record along with all related records in the other database tables to archived tables. This is necessary when current students leave the school system.

3.2. Major Object Classes

- Main Program Container
- User Login Form
- Preferences Form
- Database Access Class
- New Student Form
- Test Result Entry Form
- Collection of Report Classes (.rpt files)
- Querying Constructing Form Classes for reports
    - General Student Information
    - Individual Speed, Strength and Conditioning Report Card
    - Speed, Strength, and Conditioning Test Results by School
    - Student Rankings

4. Nonfunctional Requirements

4.1. Usability

Standard Microsoft Windows GUI will be implemented.
The interface will have user friendly labels to reduce the training time.
A user document will be provided to help learning the software.

4.2. Performance

The software requires Windows 2000 or later version.
The software uses a shared Microsoft Access file on a server and ADO.Net to access the database.
The software is required to be accessible to the Wichita Falls ISD network.
The software requires an application to read PDF format user manual.
The disk space requirement will be specified later in the development phase.
The memory requirement will be specified later in the development phase.

4.3. Reliability

To produce high reliability on this software, system testing will be performed several times during the development process. Testing will include unit testing, integrated testing, interface testing, and so forth.

4.4. Implementation

Microsoft Visual Studio will be used for designing prototypes.
Microsoft Access will be used for the database designing process.

4.5. Privacy

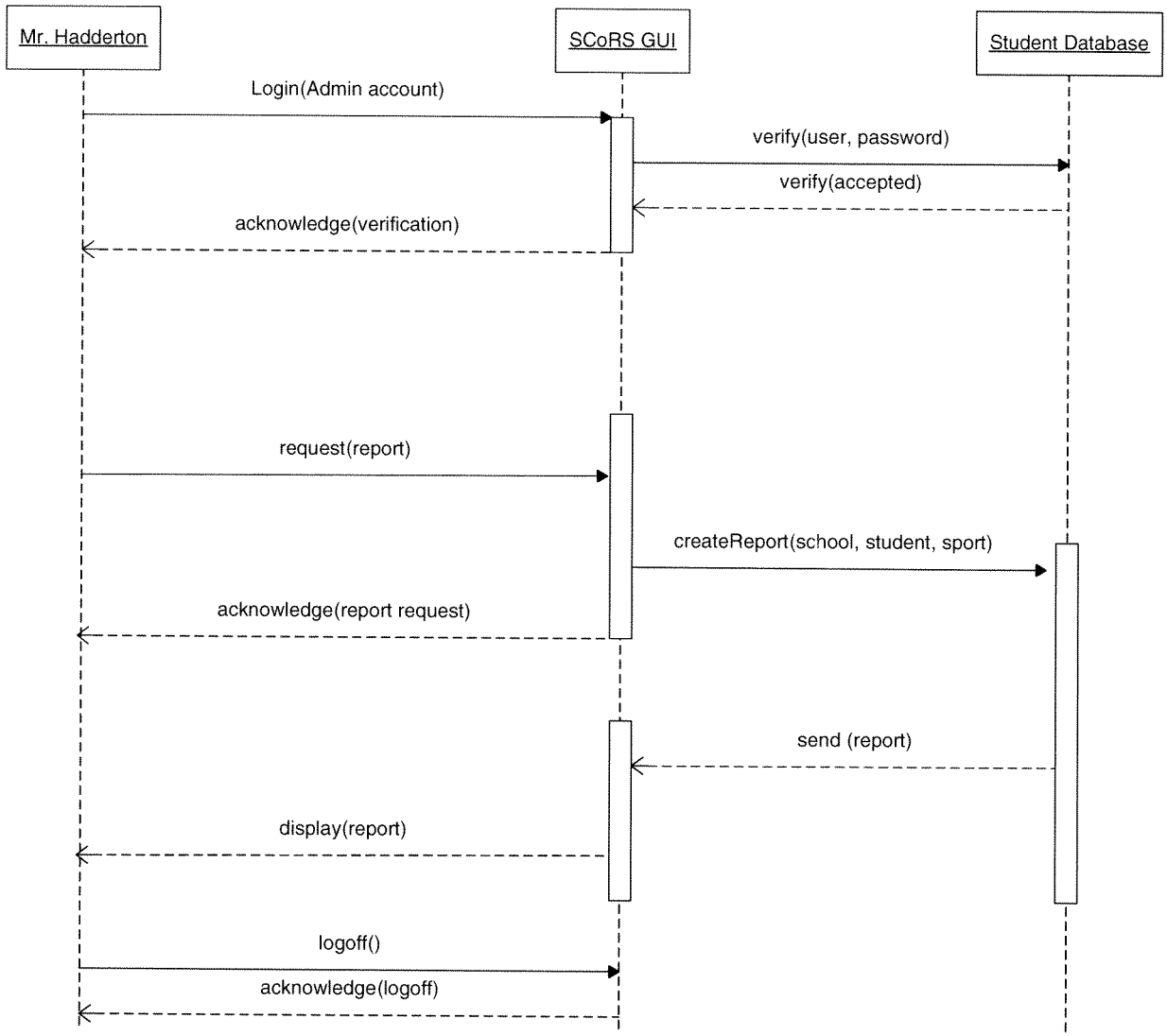
In order to secure the athletes' personal information, the software will have authentication and security system to access database. Only the system administrator will have a power to produce and edit the authorization information on the system.

5. Risks

The risks involved in the development of the system include the possible unavailability of team members. Such a risk will be tackled by demanding that team members keep an additional copy of all work done on the team's remote group account.

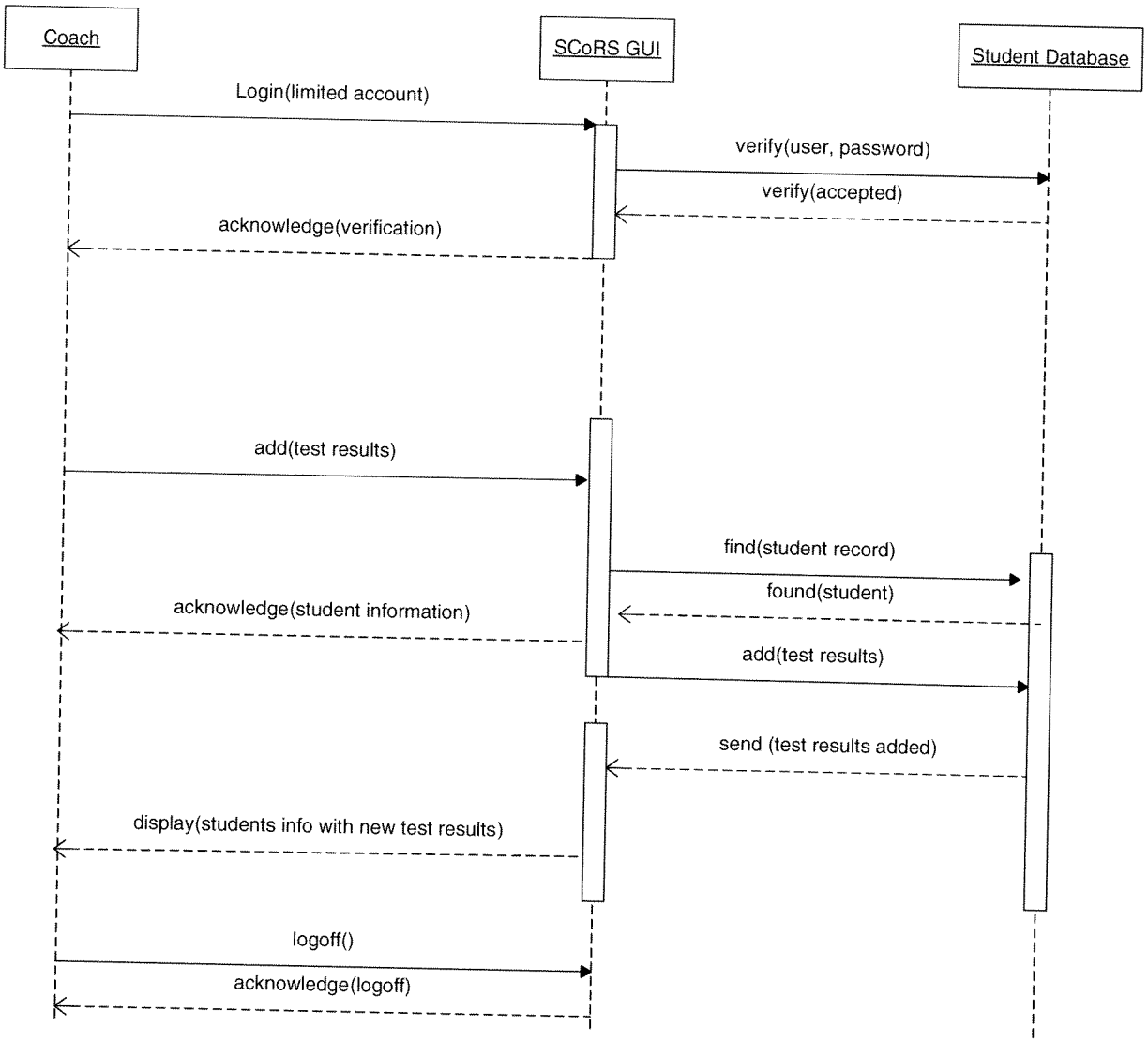
Appendix A1: Sequence Model 1

The coordinator, Mr. Hadderton, requests a report from the system:

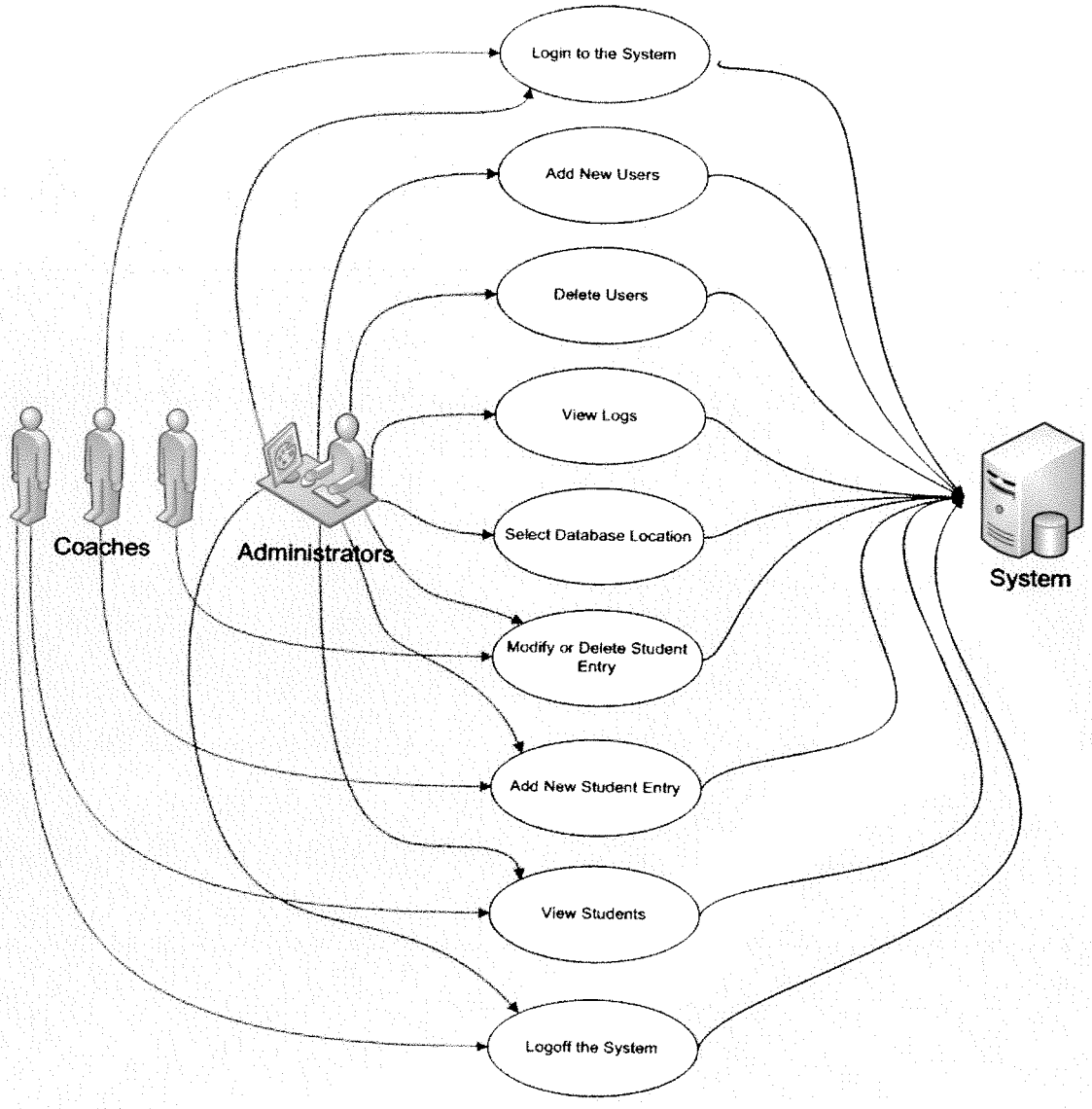


Appendix A2: Sequence Model 2

A coach adds new test results to the database and then reviews the data:



Appendix B: Use Case Diagram



Appendix C: Glossary

Glossary item	Description
Access Permissions	Permissions that control who can read or alter data. They define read and write permissions for the data's owner, members of the file's group, and all others.
ADO.Net	A set of computer software components that can be used by programmers to access data and data services
Class	In object-oriented programming, a class is a data type defined by a programmer, consisting of variables and functions.
GUI	Graphical User Interface
Main Program Container	The program's parent window. All other forms will exist as children within this main window.
PDF Format	Portable Document Format.
SCORS	Strength & Conditioning Online Report System.

Appendix D: References

Gaddis, Tony, Judy Walters, and Godfrey Muganda. Starting Out With C++ 4th ed. El Granada, California: Scott/Jones Inc., 2004.

Sommerville, Ian. Software Engineering. 8th ed. Edinburg Gate, England: Pearson Education Limited., 2007.

Fowler, Martin. UML Distilled: A Brief Guide to the Standard Object Modeling Language. 3rd ed. Canada: Addison-Wesley Professional, 2004.

Dr. Catherine V. Stringfellow, Ph.D. Department of Computer Science, College of Science and Mathematics, Midwestern State University, Wichita Falls, Texas.