



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

University of Chittagong

Course Name : Software Engineering and Information System Lab

Course Code : CSE 516

Report 04

Software Requirement Specification Document (SRS)

Submitted to:

A.H.M Sajedul Haque

Assistant Professor,

Department of Computer Science & Engineering,

University of Chittagong.

Submitted by:

Marufa Sultana

ID: 18701008

Session: 2017-18

Report – 04

Software Requirements Specification Document

SchoolHub

1. Preface

This is ‘System Requirements Specification’ document (generally known as ‘SRS’ document) of a software system named ‘School App’. It is primarily intended to be proposed to a customer for its approval and a reference for developing the first release of the system for the development team.

With version 1.0, the users will experience a completely stable release that includes high authentication of the users like students or teachers as well as exam controllers. So, there is no worry about unauthenticated users. The version 1.0 will also provide a very secure and encrypted data submission and preview. Users will experience a very fast, smooth, and accurate and reliable result processing using this release.

2. Introduction

A software requirements specification (SRS) is a description of a software system to be developed. It lays out functional and non-functional requirements, and includes a set of use cases that describe user interactions that the software must provide. Software requirements specification establishes the basis for an agreement between customers and contractors or suppliers on what the software product is to do as well as what it is not expected to do.

The purpose of this document is to give a detailed description of the requirements for the “School App (SA)”. This document will completely illustrate the purpose and the features of the system. It will provide a complete declaration for the development of the system in precise and explicit manner. It will also explain system constraints, what system will do and how the system will react to external interactions.

The “School App (SA)” is an online based parent and teacher communication system which helps to about student’s class performance, results, shortcomings, co-curricular activities, result and school update notices their progresses by school authorities. It will be designed to update notice and overall activities of students quickly and more efficiently, and reduce the workload of the teachers, guardian and also enable the administrators to monitor the progress of their students and take effective steps to improve their performance. The system will also be designed to provide both the administrators and the students with their results, class performance, results, shortcomings, co-curricular activities, result and school update notices their progresses by school

authorities. It also will enable the administrators to provide information that is accurate, timely and error free.

3. Glossary

Glossary is an alphabetical list of terms in a particular domain of knowledge with the definitions for those terms. It lists the technical terms used in the document. The glossary for this document is given in table I.

Table I : Glossary

Technical Term	Description
Authentication	The process or action of verifying the identity of a user or process.
Backup	A copy of a file or other item of data made in case the original is lost or damaged.
Constraints	The limiting barrier of an action or a system.
Credentials	A group of information proving a user's identity or qualifications.
Database	A collection of information organized into rows, columns and tables, such a way that a computer program can quickly access, manage or update desired pieces of data.
Encryption	The process of converting information or data into a code, especially to prevent unauthorized access.
Login	The process by which an individual gains access to a computer system by identify`ing themselves.
On-line	Operating being connected to a computer or telecommunication system such as internet.
Response Time	The length of time taken for a system to react to a given event.
Server	A computer or computer program which manages access to a centralized resource or service in a network.

4. User Requirements Definition

User requirement definition are statements in a natural language plus diagrams, of what service the system is expected to provide and the constraints under which it must operate. Requirements are physical or functional need that a particular design, product or process aims to satisfy. After meeting with the client and properly discussing with them, some requirements are discovered. The

requirements are divided into two categories such as, functional requirements, which defines the functions of the system required by the client, and, non-functional requirements, which defines the characteristics as well as constraints of the system. The user requirements are defined in table II.

Table II : Definition of User Requirements

Requirement Type	Definition of Requirements
Functional Requirements	1. School authority will publish student's daily activities .
	2. Examiners will be able to submit exam result against his answer-script.
	3. Class teacher will be able to update daily homework of related course.
	4. School authority will publish updated notice of the school.
	5. Students will be able to see result of his own.
	6. Teachers and student's information can be shared between guardian and school authority.
Non -Functional Requirements	7. Logging system must be highly secure and reliable .
	8. Every operation must be protected by authentication .
	9. System must response quickly .
	10. Software must require low hardware resource to operate.
	11. Every user must be able to perform required operations on-line .
	12. Software must be operable on Android Mobile Phone .

5. System Architecture

A system architecture is the conceptual model that defines the structure of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures and behaviors of the system. The system architecture of School App (SA) is described in figure 5.1

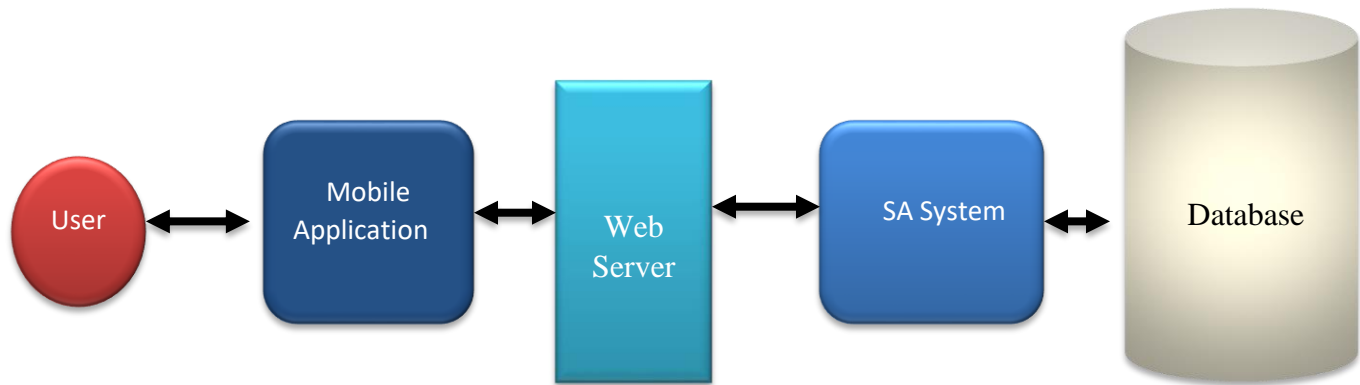


Figure 5.1 : System Architecture

6. System Requirements Specification

The software requirements specification document enlists necessary requirements that are required for the project development. To derive the requirements, the developer needs to have clear and thorough understanding of the products to be developed. Also, the requirements must be described at length for making the client clear and concise about what is going to be developed. The user requirements are described in great detail in table III.

Table III : Specification of User Requirements

Requirement Type	Specification of Requirements
Functional Requirements	1.1. The software will generate random students activity information for each students. 1.2. Each students must be uniquely identifiable by his/her assigned roll number, class and section. 1.3. School authority will assign the generated information against students.
	2.1 Examiner will input marks against each answer-script he is given to. 2.2 Examiner will submit the marks to the central server.
	3.1 Teacher will update daily homework. 3.2 Teacher inform guardian about the performance of previous homework. 3.3 Teacher will include the percentage of marks on the basis of given homework.

	<p>4.1 School authority will give vacation notice and unexpected school off day.</p> <p>4.2 School authority will give update notice about changing class schedule.</p> <p>4.3 School authority will publish additional new rules.</p>
	<p>5.1 Student will login to his own account using his own roll, class and section.</p> <p>5.2 Student and guardian will check if student's result is published.</p> <p>5.3 According to published result guardian will be observe their offspring rank.</p>
	<p>6.1 Guardians will click teachers name and teacher id, then they can know more details according to their cherish teacher.</p> <p>6.2 Guardian will select class and section, then they can know more about in this related class and section students details.</p>
Non -Functional Requirements	<p>7.1 Highly secured RSA and PGP encryption will be used to send and receive data to and from the central server.</p> <p>7.2 High performance backup server must be configured.</p> <p>7.3 Failure rate must be as low as possible.</p> <p>7.4 Startup must be rapid.</p> <p>7.5 Quick re-spawning capability must be implemented.</p> <p>7.6 Students cannot view result unless it is published.</p>
	<p>8.1 Every actor must have a unique user-account.</p> <p>8.2 Each user will have to login his own account submitting his credentials.</p> <p>8.3 No operation will be allowed to unauthorized actors.</p> <p>8.4 Only School authority can change student's daily activity, result and give information.</p> <p>8.5 Only exam controller and corresponding course-teacher can include result and daily notice.</p> <p>8.6 Only exam controller can publish overall information and notice.</p>

	9.1 Overall response time will be less than 1 second. 9.2 System must be able to serve at least 300 users simultaneously. 9.3 System must be able to generate 15 transcripts per second.
	10.1 System must be low power consuming. 10.2 The software must be operable with a minimum of 900MHz dual core CPU. 10.3 Software must be restricted within 512MB of RAM.
	11.1 Users will be able to connect their device to the server from anywhere. 11.2 School authority will be able to update all information and notice using Internet connection. 11.3 Ability to submit marks through a LAN will also be provided.
	12.1 The software must be able to run on android API 19 or higher. 12.2 User-interface must be nice-looking and user-friendly. 12.3 The software must be able to run in a device with 480dpi screen.

7. System Model

System model describes the system through some scenarios. A scenario is a narration of measurable interactions of user and the technical system, which is usually, includes computer hardware and software. It is useful for adding details to the requirement description. The system is described narratively by use cases and is presented graphically by use case diagram.

A. Use Cases

Use case is a list of actions or event steps typically defining the interactions between a role and a system to achieve a goal. Use cases are important to understand how the system interacts with the user or other systems. The use cases of the system is listed in table IV.

Table IV : List of Use Cases

Use Case	Title
UC1	Login to the System
UC2	Student's daily activity
UC3	Result Submission

UC4	Update Homework
UC5	Teacher and Student info.
UC6	Updated Notice
UC7	View Result

The use cases are described in details in the following:

UC1: Login to the System.

Actors:

1. Student
2. School Authority
3. Guardian

Preconditions:

1. Device must be powered on.
2. Software must be installed in the device.
3. Device is unlocked and in standby mode.
4. Device is connected to the central server via LAN or Internet.
5. Records of students, school authority and guardian must be already stored in the database.

Main Success Scenario:

1. Click main menu icon of the device.
2. Phone shows all installed applications.
3. Click *SA System* application icon.
4. The application will be launched and shows login options.
5. Click the proper option from these three options:
 - *Login as School Authority.*
 - *Login as Guardian.*
 - *Login as Student.*
6. Phone shows login window.
7. Enter username and password.
8. Tap the *Login* button and wait for a while.

Post Condition: Phone shows logged-in user dashboard

Alternative Course:

8. a. Phone does not show user dashboard.
 - 8.a.1. Make sure you selected correct login option.

- 8.a.2. Press the back button and select correct login option.
- 8.a.3. Enter correct password and username.
- 8.a.4. Tap *Login* button.

UC2: Student's daily activity

Actor:

School Authority

Preconditions:

1. User is logged in.
2. The phone is unlocked.
3. Phone is showing user dashboard.

Main success scenario:

1. Click on *Student's daily activity*.
2. Phone fetches roll, name, class and section list from the server.
3. Phone shows roll, class and section selection window.
4. Select roll, and section from drop-down menu.
5. Tap *Next* button.
6. Then show the student's daily activity
7. Repeat steps 3-5 to see more students daily activity.
8. Click *Main* button.

Post-Condition: Phone sends Student's daily activity updated command to the guardian.

Alternative Course:

- 3.a. Phone shows no student's record .
 - 3.a.1. Make sure your connection to server is OK.
 - 3.a.2. Tap *Refresh* button.

UC3: Result Submission

Actor:

1. School authority

Preconditions:

1. User is logged in.
2. The phone is unlocked.
3. Phone is showing user dashboard.
4. Connection to the server is OK.

Main Success Scenario:

1. Click on *Submit Result* button.
2. Phone shows a form to enter students roll, class, section and marks.

3. Enter roll, class, section and corresponding mark.
4. Click *Add* button.
5. Go to step 2 to enter more marks.
6. Click *Done* button.

Post-Condition: Phone shows “Marks Submission Successful” message

Alternative Course:

- 4.a. Phone says, “Marks for this student already exists, Modify ?”
 - 4.a.1. Click *Yes* to replace existing mark or *No* to skip.
- 4.b. I want to modify a mark already added.
 - 4.b.1. Follow steps 2-4.
 - 4.b.2. Follow alternative course 4.a.
- 8.a. I want to add more marks before submission.
 - 8.a.1. Click *No* button.
 - 8.a.2. Follow step 2-5.
 - 8.a.3. Go to step 6.

UC4: Update Homework

Actors:

1. School authority

Preconditions:

1. User is logged in.
2. The phone is unlocked.
3. Phone is showing user dashboard.
4. Connection to the server is OK.

Main Success Scenario:

1. Click on *Update Homework*.
2. Phone fetches and shows list of class and section.
3. Select a class and section.
4. Click Next button.
5. Then show updated homework of this class and section.

Post-Condition: Phone sends Updated notice command to the guardian and displays success message.

Alternative Course:

- 2.a. Phone does not show student list.
 - 2.a.1. Wait for a while.
 - 2.a.2. Click *Refresh* button.

UC5: Teachers and Student’s info.

Actor:

1. Guardian

Preconditions:

1. User is logged in.
2. The phone is unlocked.
3. Phone is showing user dashboard.
4. Connection to the server is OK.

Main Success Scenario:

1. Click on *Teacher and Student's info*.
2. Phone fetches and shows list of all record of student's and teachers.
3. Select student's roll, name and section.
4. Select teacher name and T_id.
5. Click *Next button*

Alternative Course:

- 2.a. Phone does not show Teachers and Student's info.
 - 2.a.1. Wait for a while.
 - 2.a.2. Click *Refresh* button.

UC6: Updated Notice

Actor:

1. Guardian

Preconditions:

1. User is logged in.
2. The phone is unlocked.
3. Phone is showing user dashboard.
4. Connection to the server is OK.

Main Success Scenario:

1. Click on *Updated Notice* option.
2. Phone shows list of date.
3. Select date.
4. Click *Next* button.

Alternative Course:

- 2.a. Phone does not show Updated Notice.
 - 2.a.1. Wait for a while.
 - 2.a.2. Click *Refresh* button.

UC7: View Result

Actor:

1. Guardian
2. Student's

Preconditions:

1. User is logged in.
2. The phone is unlocked.
3. Phone is showing user dashboard.
4. Connection to the server is OK.
5. Result is published.

Main Success Scenario:

1. Click on *Display Result* option.
2. Phone asks for student roll, class and section.
3. Input Student's roll, class and section in the text-box.
4. Click *Show Result* button.

Post-Condition: Phone shows the result of the student.

B. Use Case Diagram

The use case diagram shows the interaction between user and the system graphically. It also shows the interaction between its system and environment. Use case are a scenario-based techniques. The use case diagram is shown in figure 7.1.

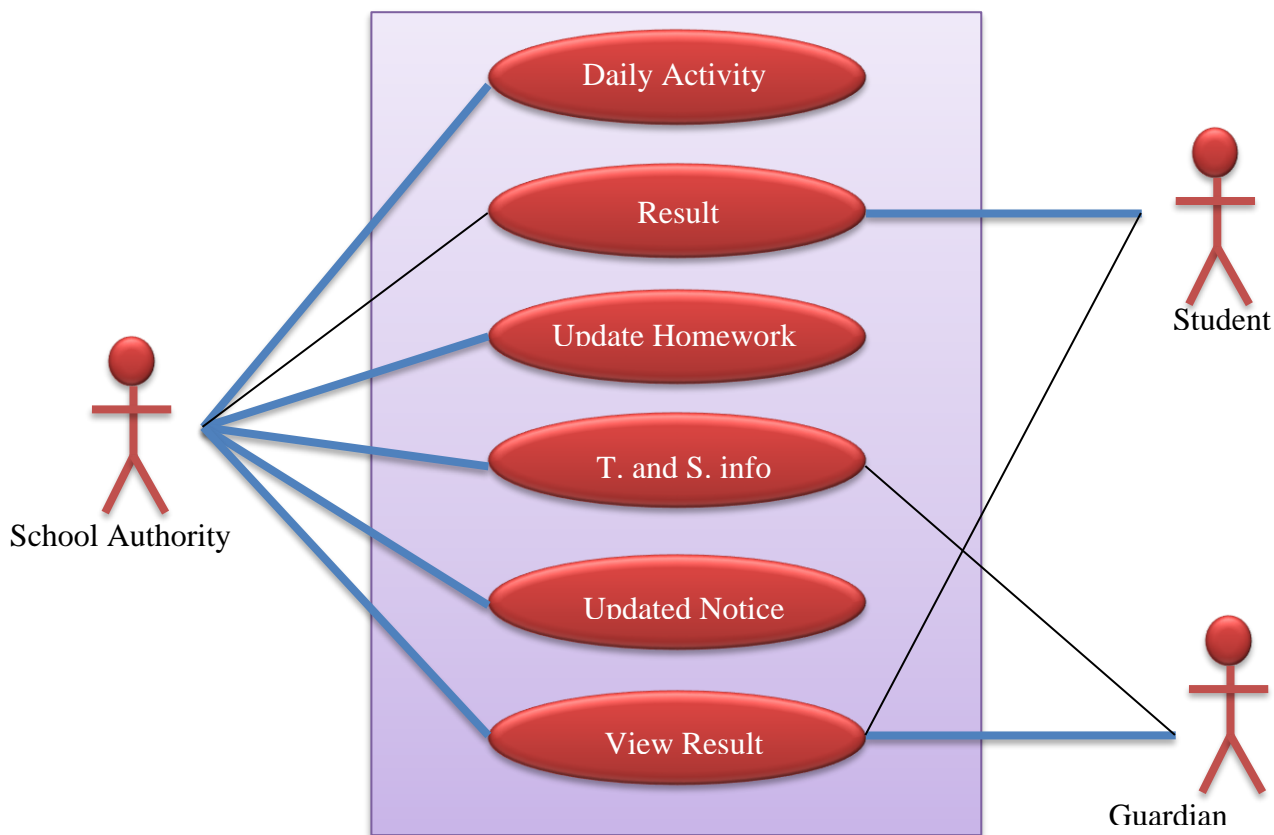


Figure 7.1 : Use Case Diagram

8. System Evolution

Over time, software systems, programs as well as applications, continue to develop. These changes will require new laws and theories to be created and justified. Software evolution is the term used in software engineering to refer to the process of developing software initially, then repeatedly updating it for various reasons.

School App (SA) is developed to be an adaptive system. It is implemented in such way that it adjusts its performance with respect to the specification of the hardware such as servers. With version 1.0, the system can process as much as 200 requests per second for a server with processor speed of 3.3 GHz, consisted of 4 logical cores and maximum memory of 4GB. Any change to the hardware would change the performance of the system in proportion to the change of processor speed, number of cores and maximum memory.

9. Appendices

Appendices contains the texts that is explanatory, statistical, or bibliographic in nature. The appendix for this document contains the hardware specification, database specification for the system.

Appendix A : Hardware Specification

The system is developed using the server “HPE ProLiant ML10 Gen9 Tower Server”. The specification of the server is given in Table V:

Table V : Server Specification

Processor	Intel® Xeon® E3-1225 v5
Number of Processors	1
Processor Core Available	4
Processor Cache	8MB (1 x 8MB) Level 3 cache
Processor Speed	3.3GHz
Chipset	Intel® C236 Chipset
Power Supply Type	300W Multi-Output Power Supply
Memory	4GB DDR4
Memory Slots	4 DIMM slots
Memory Type	1R x8 PC4-2133P-E-15

Memory Protection Features	Un-buffered ECC
Included Hard Drives	LFF SATA; 1TB
Maximum Internal Storage	24TB
Optical Drive Type	SATA 9.5mm DVD RW
System Fan Features	Non-Pluggable Fan
Network Controller	Intel® Ethernet Connection I219-LM
Storage Controller	Integrated SATA RAID
Infrastructure Management	Intel® Active Management Technology (Intel® AMT 11.0)

Appendix B: Database Specification

The system uses the “MySQL Enterprise Edition” as database management system. The technical specification of the database management system is given in Table VI.

Table VI : Database Specification

Version	5.7
Data Type	Static
Architecture	Relational Model
Operating System	Linux, Solaris, FreeBSD, Mac OS, Windows
Software License	GNU General Public License
Interface	GUI, CLI
Security	<ol style="list-style-type: none"> 1. SSL Support 2. Built-in Data Encryption/Decryption 3. View Support 4. Triggers for auditing 5. Query Logs for auditing
Access Control	<ol style="list-style-type: none"> 1. Enterprise Directory Compatibility 2. Native Network Encryption 3. Run Privilege 4. Security Certification

Indexes	<ol style="list-style-type: none"> 1. R-/R+ Tree 2. Hash 3. Full-text 4. Spatial
Partitioning	<ol style="list-style-type: none"> 1. Range, Hash, List, Key 2. Composite 3. 8k partitions per table 4. Portable partitions between tables 5. Explicit querying by partition 6. Transparent Pruning
Max Database Size	Unlimited
Max Table Size	256 TB (MyISAM) 64 TB (InnoDB)
Max Row Size	64 KB
Max Column Per Row	4096
Max Blob/Clob Size	4 GB
Max CHAR Size	64 KB
Max NUMBER Size	64 Bit
Min Date Value	1000
Max Date Value	9999
Max Column Name Size	64

10.Index

A. List of Figures

Figures are graphical representation of information. The figures used in this document is listed in table VII.

Table VII : List of Figures

Figure Name	Name of Figure	Page No
Figure 5.1	System Architecture	23
Figure 7.1	Use Case Diagram	31

B. Alphabetical Index

A

Authentication, 21-23

C

Constraint, 21-22

Credentials, 22,25

D

Database, 22, 26, 32, 33

E

Encryption, 22, 25, 33

F

Failure Rate, 25

L

Login, 22, 24-27

O

On-line, 22-23

Operation, 23, 25

R

Requirement,

 System Requirement, 21

 Functional Requirement, 21-24

 Non-functional Requirement, 21-22

User Requirement, 21-24

Response Time, 22, 25

S

Scenario, 22, 26-31

Server, 22, 26-30, 32

Software, 21, 23-26, 32-33

System Architecture, 22-23

System Model

 Conceptual Model, 23

U

Use Case, 21, 22, 26, 31