

Software Requirement Specification

SEGP Coursework

Submitted to: Dr.Adil Raja

Submission Date: 3/11/2017

Submitted by: Group-2

Naveed Hakim

Hamza Safdar

Khizar Farooq

Muhammad Usman

Muhammad Nabeel

Group Mentor: Sir Sarmad Ali

Client: Group 4



Department of Computer Science
NAMAL COLLEGE, MIANWALI

Table of Contents

1. Introduction:	3
1.1 Purpose	3
1.2 Intended audience	3
Client	3
Interface designer	3
Developers	3
1.3 Scope	3
1.4 Abbreviations/Acronyms	4
1.5 Overview	4
2. Overall Description	5
2.1 Product perspective	5
2.2 Product function	5
2.3 User characteristics	5
2.4 Constraints	6
2.5 Apportioning of Requirements	6
2.6 Assumption and Dependencies	6
3. Specific Requirements	6
3.1 External interface requirement	6
Hardware Interface	6
Software Interface	7
Communication Interface	7
3.2 Functional Requirements	7
3.3 Performance Requirements	12
3.4 Software attributes/Non Functional Requirements	13
3.5 Design Constraint	14

1. Introduction:

1.1 Purpose

The motto is to make an electronic personal academic tutoring system (PATs) which can keep the record of all students especially from which group they are, which tutor assigned to them and can store all vital information which is needed. Currently, this work is being handled manually and lot of labor is required to deal with all queries so goal is to build an appropriate system which can replace current manual system.

1.2 Intended audience

This document will provide a clear vision to all of the people who are associated with this project such as client, interface designers, programmers and all others who are assisting in making of this application.

Client

The premier reason of this document is to guide client that how their requirements are going to be solved and which methods are going to be applied in order to make an efficient system that can full-fill all clients expectations.

Interface designer

This is very useful document for interface designers as they can see all details about application functionalities and all use case. He can view all required information in order to make changes and modifying this application by providing more features to client in near future.

Developers

Developers can read this document and if they have intentions to make similar kind of application this can help them as sequence of SRS is properly defined. All they have to do is just to skim through this document and make mandatory changes for their product.

1.3 Scope

PAT is very useful system for the administration of any college/university which assigns students different group projects along with their personal tutor. Without this application student support office has to do immense labor work by entering data of students on paper and when they want to edit some information of particular student then they have to do cutting and stuff like that which

obviously seems pity. It can also resolve some security issues which I will discuss later in this SRS. Moreover, it could have slightly different accessibilities for different authorities. For example, SSO will be given full access they can alter tutor of any group, they can change or swap students from any group and would have authority to add or remove students. After making suitable changes request would be sent to higher authority to approve changes so that they can come into contention from onwards. After modifications student can also view that by checking groups.

1.4 Abbreviations/Acronyms

SSO	Student support office
PATS	Personal academic tutoring system
PAT	Personal academic tutor
HTML	Hyper text markup language
FTP	File transfer protocol
PHP	Personal home page (Programming language)
PDF	Portable document format (File)
CSS	Cascading style sheets
HTTP	Hypertext transmission protocol
LAN	Local area network
UOB	University of Bradford
URL	Universal resource locator

1.5 Overview

As described earlier it's a modern way of storing data and more effective than handled manually. It will replace data stored on excel sheets. Storing data on sheet is very lengthy way of doing it and another hardship is that when a batch graduates we had to remove sheets of them and add new sheets of new comers. With this application this work will be handled quite nicely, there would be database behind on a local server that will contain necessary data. User can see all data of students and PAT assigned to them. SRS will be well organized considering the proper way of implementing all basic

techniques of well arranged application. Moreover, student support office would be sort of a controller who would have authorities to manipulate according to their usage. It can reduce lot of time and using this system can assure as of security issues.

2. Overall Description

2.1 Product perspective

This would be a web based application which will need an internet connection and 2B ram on user's system to run it rather smoothly. This will provide access to user to manipulate data of students and their assign tutors. It will maintain a data base placed on hosting server and user will have 24/7 access to make changes.

2.2 Product function

The main function is that user/administration can view data of students or PATS or whatever they opt for. The output will depend upon the input of user. For example if user wished to see students of particular year only then only students of that year would be listed. Other functions include view data of specific year, view groups, add/remove students in groups. A complete group can also be deleted if user wants. Tutor can also be changed of but considering the rules that one tutor is for only for one group.

2.3 User characteristics

User here is our administration. An email and password will be required for the user for security purpose after entering correct email and password user will switch to home page. Main characteristic of user is to add students to groups, make groups, assign tutors and additional/removal of groups. They can manage each and every aspect of this application. User can also view list of groups or tutors in PDF file if they want.

2.4 Constraints

1. Internet is required whenever user want to open this application.
2. Language will be English. Without knowing English user can't operate this application.
3. User has to fetch data from data base that may consume some time especially if internet is slow.
4. Whenever user (SSO) will make any change. Students can't know the respective changes unless or until administration informed them through sending mail. In other words there would be no electronic communication between SSO and students as this system is only for SSO
5. Code is done on HTML, CSS, bootstrap and PHP, which may have certain problems.

2.5 Apportioning of Requirements

As we have limited time and working under some constraint situation. Mainly our focus is to learn making of applications. Since it's our first project so it may not be perfect, there would be room to improve and we will try to add additional features in updated version of this application.

2.6 Assumption and Dependencies

Assumptions are very clear and precise. First of all, our code should be error free we can't afford any mistake in coding terms. It should be available 24 hours. Another crucial assumption is that interface should be user friendly so that user shouldn't face any difficulty operating this application.

Persistent power supply is needed to keep server running 24/7 so that it can provide us services. We are dependent on administration too that they should maintain all the data correctly and sincerely. We are dependent on internet. Without internet this application can't be run. We are trying to make it as user friendly as we can so that more users will use our application and of course we are dependent on users to user our App.

3. Specific Requirements

3.1 External interface requirement

Hardware Interface

This system doesn't have any hardware connection so there is no hardware interface in this application.

Software Interface

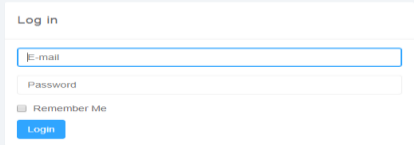
A simple operating system will be needed (windows preferable). Having said that, a browser is required to show web pages (chrome, Mozilla)

Communication Interface

A Server will be made to retrieve and update data from and in data base. It would be web based application coded in PHP.

User Interfaces

Login page



The image shows a login form titled "Log in" centered on a light blue background. The form is white and contains the following elements: a text input field labeled "E-mail", a text input field labeled "Password", a checkbox labeled "Remember Me", and a blue button labeled "Login".

It's general login page. We are trying to make it simple. User should type valid email in email box and valid password in password box to login into system

View Students

The screenshot shows the 'View Students' page in the LUMINOADMIN system. The page has a dark header with 'LUMINOADMIN' on the left and 'User' on the right. A sidebar on the left contains a search bar and a list of navigation links: 'Add New Student', 'Edit Students', 'Create Group', 'Assign Groups', 'Download Data As PDF', 'Upload File', 'Display Students' (which is highlighted in blue), and 'Login Page'. The main content area is titled 'Display' and contains a 'Select Year' dropdown menu. Below the dropdown, it says '0 results'.

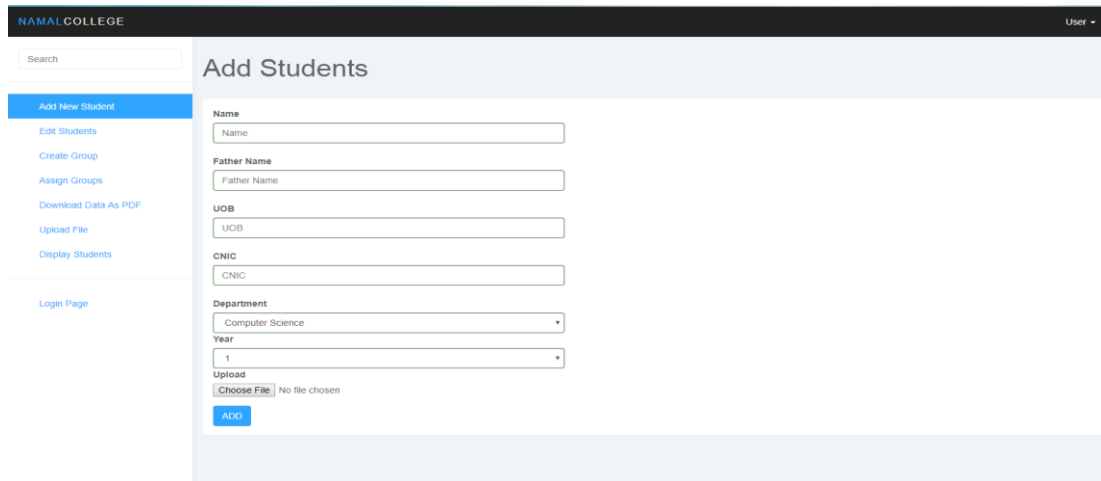
In this page, administration can view student's year wise. User will select which year students he wants to view. By selecting year he can view students of that year. We will provide another option here that it should also display allocated and unallocated students here.

Create groups

The screenshot shows the 'Create Group' page in the LUMINOADMIN system. The page has a dark header with 'LUMINOADMIN' on the left and 'User' on the right. A sidebar on the left contains a search bar and a list of navigation links: 'Add New Student', 'Edit Students', 'Create Group' (which is highlighted in blue), 'Assign Groups', 'Download Data As PDF', 'Upload File', 'Display Students', and 'Login Page'. The main content area is titled 'Create Group' and contains a form with two input fields: 'Group Name' and 'Group Number'. Below the 'Group Number' field is a blue 'Create' button.

Administration can create new groups. Every group will have a unique name and a number. After creating groups they can assign groups their respective tutors.

Add Student



The screenshot shows a web application interface for adding students. At the top, there is a dark header bar with 'NAMALCOLLEGE' on the left and 'User' on the right. Below the header, on the left side, is a sidebar menu with a search bar and several options: 'Add New Student' (highlighted in blue), 'Edit Students', 'Create Group', 'Assign Groups', 'Download Data As PDF', 'Upload File', 'Display Students', and 'Login Page'. The main content area is titled 'Add Students' and contains a form with the following fields: 'Name' (text input), 'Father Name' (text input), 'UOB' (text input), 'CNIC' (text input), 'Department' (dropdown menu with 'Computer Science' selected), 'Year' (dropdown menu with '1' selected), and 'Upload' (file upload button with 'Choose File' text and 'No file chosen' status). At the bottom of the form is a blue 'ADD' button.

Administration can add new students in the database from where we can add them into different groups. Name, father's name, UOB, CNIC, department and year column need to be full-filled for adding a new student.

3.2 Functional Requirements

Maintaining record

This application should maintain all data whether that is of current student or old student. The record of PAT should also be maintained. It should automatically update data when that is edited or changed. The data should always be there for user so that we can see data whenever he wants.

Use Case

Administration

SSO should be given full access to maintain all data of students along with their PAT. The application should give different tabs to sign up a new user, add a student to specific group, remove a student from a group, adding a new PAT to group and removing a PAT from current group. SSO should have all these features available on different tabs.

Altering Personal Information

The interface should be built easy for administration to change personal information of students and assigned PAT's. It should provide facility to modify information like name, number and UOB.

Use Case

Administration

Administration can use this feature very effectively system should has 2 different tabs one for all students and one of all PAT. For example user want to edit some information of student, he/she can student tab and by the help of this feature user should have access to change the name of that student, change contact number of that student and things like that.

Assigning PAT's to Students

It's the fundamental requirement of this application to assigning students to respective PAT's. Minimum of 4 students and maximum of 8 students can be in one group and that group will get only one PAT. So one PAT is for 4-8 students. There are different scenarios in this requirement for example when new batch comes into university now they should be put into groups first and then groups will get different PAT's.

Use case

Administration

Our user in this application is administration. So administration will use this functionality to assign PAT's to groups. They will be given full access considering the rules (4-8 students per group) to modify any

group. User should be provided tabs to add new student or groups and assign different PAT's. New PAT's can also be added or previous PAT's could leave too so all the work will be done by administration.

View records

This is very important thing to implement in this application. There are certain features which have been included in order to make this functionality convenient. For example, records could be viewed in different cases like user may only need record of only 1st year students, user could need list of assign PAT's to specific year and in some cases user might need list of all student and PAT of whole university.

Use case

Administration

Administration would operate this functionality, they will be provided tabs from where they can view list of students and PAT's separately. They can also view list of specific student and PAT's depending upon what they want. System will provide all features to user.

Export list

This functionality can help to get all lists of students from data base in any format especially PDF file. It's helpful tool. Functionality will allow user to retrieve any kind of data from data base to PDF file. For example user can generate list of 1st year student in PDF in tabular format.

Use case

Administration

Administration will be provided a tab from where they can generate list of student in PDF format. They have full right to generate anything which is in data base in PDF format.

Communication

This function will provide platform for user to communicate with students like they can inform students and PAT's that their group has been changed or anything similar to that. But this functionality is not in this feature. It's in our future plans to come up with this functionality in updated version.

Use Case

Administration

Administration shall be able to communicate with students or PAT's by linking up this application to Gmail. They will be provided a tab from where they can mail students. But these features will come in next version.

3.3 Performance Requirements

Our system is optimize and flexible enough to deal with all performance requirements. Here are list of some features reflecting performance.

Search feature

User can easily search any student or PAT with this feature. By simple searching he can search from all students studying in university. Similarly he has an option to only search students+ of particular year that is easier to find students. Similar feature will work if user wants to search PAT.

Wrong Password/Reset Password

There is no feature in this version that allows user to reset password if he/she forgets. If user will enter wrong password then application would give user another try to give right password. This will go on until user will type right password. In updated version we have a plan to add feature of reset password.

Capacity

The ambition is to code so effectively that maximum user can use this application simultaneously. Assumption till yet is that it will sustain load without any problem.

Wrong editing

If user want to edit information and he/she has entered wrong or invalid information (entered UOB in name column) then it should display pop up and refuse user that this process can't be operated. But in this feature there isn't anything similar. However it is in our future plans to add such features.

3.4 Software attributes/Non Functional Requirements

Reliability

Reliability is very serious issue in any application. We are trying to make it so reliable that it can run in any device considering operating system is installed in that device. It should be fault tolerant.

Recoverability

There should be back up copy somewhere safe of data base to retrieve data in case of any data loss due to certain accidents. Software should recover data from any behavior state.

Scalability

It should provide all required services to our administration.

Flexibility

The motto is to make interface so easy to use that not only experience operators can use it but also inexperienced people too won't face difficulties using this application. It should perform all tasks without smoothly.

Availability

It should be available 24 hours. Along with that, it should be monitored on weekly or monthly basis.

Error Prevention

System should offer some suggestion if user is doing something that is causing errors. Moreover, it should prevent unconscious errors coming without any reason.

Maintainability

The system should be good enough to adapt any kind of change it should protect data from any mishap or failures coming from unwanted states.

Security

Data base containing data and information should be defensive and it should be protected from hackers.

3.5 Design Constraint

There were few constraints constructing the design. When we were prototyping constraints were there that how will we display students of specific year or should list students from everywhere. But later we made different web pages to resolve this problem and were succeeded. Another major problem was consistency, as different people were assign web pages so sizes, colors and pattern were different of all of us. But later we selected appropriate and suitable designs by checking and comparing everyone's design. Moreover, it was our first project so problem was there to construct basic design.