**Software Requirements Specification**

Version 1.0

Tuesday, October 10, 2017

**Web Accessible Alumni Database**

by

Naif Alotaibi

# Table of Contents

[Table of Contents 2](#_Toc495388899)

[1. Document purpose 1](#_Toc495388900)

[2. Document overview 1](#_Toc495388901)

[3. General description of the product 1](#_Toc495388902)

[3.1. The current situation 1](#_Toc495388903)

[3.2. Purpose of the product 1](#_Toc495388904)

[3.3. Product context 1](#_Toc495388905)

[3.4. Benefits 2](#_Toc495388906)

[4. Functional requirements 2](#_Toc495388907)

[4.1. Actors 2](#_Toc495388908)

[4.1.1. Alumnus 2](#_Toc495388909)

[4.1.2. Secretary Staff 2](#_Toc495388910)

[4.2. System boundary 3](#_Toc495388911)

[4.3. Use cases description 4](#_Toc495388912)

[4.3.1. Access Alumni Home Page 4](#_Toc495388913)

[4.3.2. Fill out the Survey 7](#_Toc495388914)

[4.3.3. Create a new entry 8](#_Toc495388915)

[4.3.4. Update an Entry 9](#_Toc495388916)

[4.3.5. Search for an Alumni/E-mail an Alumni 11](#_Toc495388917)

[5. Statistics 12](#_Toc495388918)

[6. Non-functional requirements 13](#_Toc495388919)

[6.1. User Interface Requirements 13](#_Toc495388920)

[6.2. Performance Requirements 13](#_Toc495388921)

[6.3. Availability & Reliability 13](#_Toc495388922)

[6.4. Security Requirements 13](#_Toc495388923)

ii

# 1. Document purpose

This document is intended to describe accurately the capabilities that the software product “Web Accessible Alumni Database” should provide to its end-users and also to specify all the non-functional requirements that the application should implement, regarding subjects like: performance, availability, reliability, security, design etc.

# 2. Document overview

The remainder of this document is four chapters, the first offering a general description of the software product about the initial situation, the purpose of the project, the context and the benefits of the project.

The second chapter lists the functional requirements that the software product should meet. So, it describes the actors, the system boundary and the use cases.

The third is about statistics. The final chapter exposes the non-functional requirements of the application, such as:

Performance, safety, security issues, etc.

# 3. General description of the product

## 3.1. The current situation

The Alumni department from the Prince Sultan University has a partial Alumni database storing information about its graduates, but this database is accessible only for the secretary staff of the faculty. Because of this, the stored information can soon become inaccurate, so useless. Moreover, the alumni find sharing the access to this database as a very useful initiative.

## 3.2. Purpose of the product

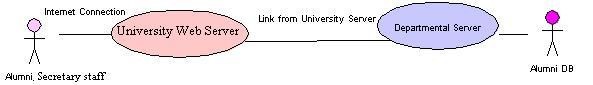
The software product is intended to offer remote access to the Alumni Database for the graduates of Prince Sultan University. So, the stored information will remain accurate and it will be visible both to the secretary staff and alumni.

## 3.3. Product context

The software product is designed to run on the server of Prince Sultan University.

It will receive incoming requests from the University Web Server, it will process them and, finally, it will query the underlying Alumni database and obtain the desired results/information, which will be passed towards the web server.

Users would access the product by using their web browser, so an Internet connection is necessary to access the system.



## 3.4. Benefits

This software product is supposed to satisfy both the alumni wish for sharing access to the database storing their information and the secretary staff need of storing accurate information regarding the graduates.

# 4. Functional requirements

## 4.1. Actors

The profiles of all user categories are described here.

### 4.1.1. Alumnus

The ***Alumnus*** performs any of the following operations:

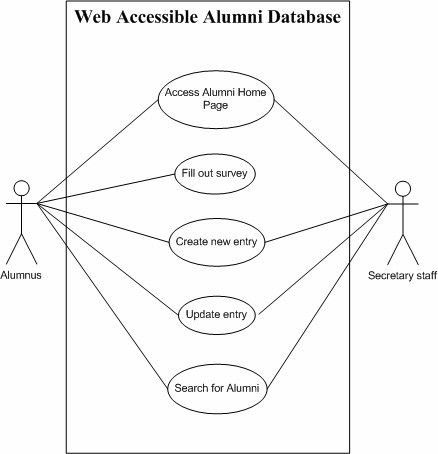
* Access Alumni Home Page
* Fill out survey
* Create new entry
* Update entry
* Search for Alumni

### 4.1.2. Secretary Staff

The ***Secretary Staff*** performs the same operations like the Alumni, except for the *Fill out the survey* operation.

So, we decided to consider ***Alumnus*** to be the main actor of this software system.

## 4.2. System boundary



## 4.3. Use cases description

#### 4.3.1. Access Alumni Home Page

In the home page, the user will provide the user name and password to access the alumni home page. In the login page, there should be a link to remind the user of password in case he forgot. The login page should be similar to the following page (see below figure 4.1). In the alumni homepage, the alumni or any user can add, update or view his/her information.

To login, the user:

1. Open the login page in the browser.
2. The user is connected to the server
3. The user enters user id and password. The user has three attempts to enter the alumni home page. If after three tries, the password does not match, the PSU Server will return a message telling the Alumnus to contact the PSU designated faculty member to receive their password.
4. If user id and password matches, the user will be directed to the alumni home page.

|  |
| --- |
|  |
| Figure 4.1 login page |

|  |
| --- |
|  |
| Figure 4.2 Alumni home page |

The alumni home page should have the following items:

1. All tabs for all, alumni , master, and Coop students
2. It should have search text box to search for alumni and the search will be by
   1. Student ID
   2. Student name
   3. Company name
   4. Job title

1. There should be sorting of alumni. the sorting selection should be from a list box having the following as a list box items:

|  |  |
| --- | --- |
| 1. Student ID 2. Student name 3. GPA | 1. Graduation year 2. Company size |

1. The list of alumni that will be received from the server should be added to list view containing the following information (see figure 4.2):

|  |  |  |  |
| --- | --- | --- | --- |
| 1. name | 1. Student ID | 1. Email | 1. phone |
| 1. major | 1. Coop company | 1. Current company | 1. Company size |
| 1. Nationality | 1. Job title |  |  |

1. There should be sorting by every header of the list view (by name, student ID, email, phone etc.), see figure 4.2.
2. There should be a filter to the data list where we can show only the data we want. The filters that we will use are:
   1. Graduation year (return the list of alumni that graduate in a specific year).
   2. College (Computer science, finance, marketing, Information systems, etc.)
   3. Other (nationality, Company size (large, medium, small) , GPA)
3. We should have Invite button to invite the alumni to the page so he/she can update his/her information. The invite will be an email sent to alumnus.
4. When the list view item (alumnus) clicked, we should have the detailed information of that alumnus.
5. The user should be able to export the list view to an excel sheet
6. The user should be able to create new alumnus
7. The user will have the ability also to invite the alumnus to the alumni we page to update his/her information.

#### 4.3.2. Fill out the Survey

|  |  |
| --- | --- |
| **Use Case Name:** | Survey |
| **Brief Description:** | This operation permits alumni to fill out a survey. |
| **Priority** | Essential |
| **Trigger** | Alumni choose to fill out a survey. |
| **Precondition** | Alumni are connected to the Internet and on the PSU Alumni Home Page |
| **Basic Path** | 1. The PSU Server presents the Alumni with a form. 2. Alumni fill in the form (survey) and click submit 3. The PSU Server checks to see if all required fields are not empty. 4. If the required fields are not empty, the PSU Server creates a new record in the Survey Table of the Alumni Database. 5. If any of the required fields are empty, the PSU Server returns a message and returns the Alumnus to the Survey form. 6. The PSU Server returns the Alumnus to the Alumni Home Page |
| **Alternate Path** | N/A |
| **Post condition** | The survey record is created in the Survey Table of the Alumni Database. |
| **Exception Path** | If the connection is terminated before the form is submitted, the fields are all cleared and the Departmental Server is returned to the wait state. |

#### 4.3.3. Create a new entry

|  |  |
| --- | --- |
| **Use Case Name:** | Create a new entry |
| **Brief Description:** | This operation permits alumni to create a new entry on the Entries page. |
| **Priority** | Essential |
| **Trigger** | Alumni choose to create a new entry on the Entries page. |
| **Precondition** | Alumni must be connected to the Internet and on the PSU Entries page. |
| **Basic Path** | 1. Alumni click on add a new entry. 2. The PSU Server returns a form. 3. The form will have fields to be filled about the detailed information of the alumnus 4. Alumni fill in the form and click *submit*. 5. The PSU Server checks to see if any required field is empty. 6. If any required field is empty, the PSU Server will send a message and return Alumni to the new entry form page. 7. If no required field is empty, the PSU Server will create a new record in the Alumni Table in the Alumni Database, and return Alumni to the PSU Alumni Home Page. 8. Alumni may select Cancel. 9. If Alumni select Cancel, the form is cleared and Alumni are returned to the PSU Alumni Home page. |
| **Alternate Path** | N/A |
| **Post condition** | A record is created in the Alumni Table of the Alumni Database. |
| **Exception Path** | 1. If the connection is terminated before the form is submitted, the fields are cleared and the PSU Server is returned to the wait state. 2. If the connection is terminated after the form is submitted, but before Alumni are returned to the PSU Alumni Home Page, the record is created in the Alumni Table of the Alumni Database. |

#### 4.3.4. Update an Entry

|  |  |
| --- | --- |
| **Use Case Name:** | Update an Entry |
| **Brief Description:** | This operation permits alumni to update an existing entry in the Alumni Database. |
| **Priority** | Essential |
| **Trigger** | Alumni choose to update an existing entry in the Alumni Database. |
| **Precondition** | Alumni must be connected to the Internet and on the PSU Entries Page. |
| **Basic Path** | 1. The Alumnus clicks on update an entry link. 2. The PSU Server returns the login page 3. The alumni or user should provide the login credentials. 4. If login successful, user will be directed to the alumni home page. 5. In the alumni home page, the server will return all graduates. 6. The user will refine to the wanted alumnus 7. User will click on edit profile. 8. The only privileged user will update the alumnus information 9. The user or alumni update his/her information. |
| **Post condition** | The record in the Alumni Table of the Alumni Database has been updated and the Alumnus is returned to the PSU Alumni Home Page. |
| **Exception Path** | 1. If the connection is terminated before the form is submitted, the fields are cleared and the   PSU Server is returned to the wait state.   1. If the connection is terminated after the form is submitted, but before the Alumnus is returned to the PSU Alumni Home Page, the record in the Alumni   Table of the Alumni Database is updated and the  PSU Server is returned to the wait state |

#### 

#### 4.3.5. Search for an Alumni/E-mail an Alumni

|  |  |
| --- | --- |
| **Use Case Name:** | Search for an Alumni |
| **Brief description:** | This operation permits the Alumnus to search for the information belonging to other Alumni. |
| **Priority** | If time permits. |
| **Trigger** | The Alumnus chooses to search Alumnus. |
| **Precondition** | The Alumnus is connected to the Internet and on the PSU Alumni Home Page. |
| **Basic Path** | 1. When the user enters the login credentials, the alumni home page will be loaded. 2. The user will search for alumnus in a search text box. 3. The search will be according to the :    1. Student ID    2. Student name    3. Company name    4. Job title 4. The user has the option to email the alumnus to invite him to the alumni web page |
| **Alternate Path** | N/A |
| **Post condition** | The Alumnus receives the information on the requested and searched Alumnus, receives e-mail invitation message, or he is returned to the PSU Alumni Home Page |
| **Exception Path** | 1. If the connection is terminated before the information is returned, the PSU Server is returned to the wait state. 2. If the connection is terminated after the information is returned, the PSU Server is returned to the wait state |

# Statistics

The statistics will be graphs for viewing the number of alumni according to :

* 1. Graduation year
  2. College (Computer science, finance, marketing, Information systems, etc.)
  3. Other (nationality, Company size (large, medium, small) , GPA)

|  |
| --- |
|  |
| Figure 5.1 Statistics Page |

# 6. Non-functional requirements

## 6.1. User Interface Requirements

The user interface of the application must be user-friendly, intuitive and easy to use, implementing the ergonomics standards.

## 6.2. Performance Requirements

The system shall function in real-time: any operation on the stored information, triggered by the Alumni, shall complete in less than *10 seconds*.

The system shall allow simultaneous use by at least *100 users*, without data corruption.

## 6.3. Availability & Reliability

The software system could provide automatically generated *backup* (on external hard drives) containing all the stored information at the time the backup is taken. The system shall allow authorized users to restore the data from an existing backup.

## 6.4. Security Requirements

In order to use certain features of the system, users must first authenticate themselves by name and password. The system shall not allow access if the user fails to provide correct log in information.

The system should automatically perform log out if the user has been idle for a specific period (e.g. 30 minutes).

Physical access to the computer(s) storing the Alumni Database shall be restricted to authorized personnel.