<QsARC: An accreditation Software for UTB>

Software Requirements Specification

<Version 1.0>

<October 28,2014>

<Juan Hector Nava>

Lead Software Engineer

Prepared for

Software Engineering

Instructor: MK Quweider, Ph.D.

Fall 2014

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Description** | **Author** | **Comments** |
| Oct-28-2014 | Version 1.0 | Juan Hector Nava | First revision |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Document Approval

The following Software Requirements Specification has been accepted and approved by the following:

|  |  |  |  |
| --- | --- | --- | --- |
| **Signature** | **Printed Name** | **Title** | **Date** |
|  | Juan Hector Nava | Lead Software Eng. | Oct-28-2014 |
|  | Dr. MK Quweider | UTRGV Representative | Oct-28-2014 |
|  |  |  |  |

**Table of Contents**

Revision History ii

Document Approval ii

1. Introduction 1

1.1 Purpose 1

1.2 Scope 1

1.3 Definitions, Acronyms, and Abbreviations 1

1.4 References 1

1.5 Overview 1

2. General Description 2

2.1 Product Perspective 2

2.2 Product Functions 2

2.3 User Characteristics 2

2.4 General Constraints 2

2.5 Assumptions and Dependencies 2

3. Specific Requirements 2

3.1 External Interface Requirements 3

3.1.1 User Interfaces 3

3.1.2 Hardware Interfaces 3

3.1.3 Software Interfaces 3

3.1.4 Communications Interfaces 3

3.2 Functional Requirements 3

3.3 Non-Functional Requirements 4

3.3.1 Performance 4

3.3.2 Reliability 4

3.3.3 Availability 4

3.3.4 Security 4

3.3.5 Maintainability 4

3.3.6 Portability 4

3.4 Inverse Requirements 4

3.5 Design Constraints 4

3.6 Logical Database Requirements 4

3.7 Other Requirements 4

4. Analysis Models 4

4.1 Sequence Diagrams 5

4.3 Data Flow Diagrams (DFD) 5

4.2 State-Transition Diagrams (STD) 5

5. Change Management Process 5

A. Appendices 5

A.1 Appendix 1 5

A.2 Appendix 2 5

# 1. Introduction

## 1.1 Purpose

*What is the purpose of this SRS and the (intended) audience for which it is written.*

*The purpose of this Software Requirements Specification is to give a detailed description of the functionality of the QsARC: An accreditation Software for UTB, what the system will do, the constrains, its interfaces, dependencies, and attributes.*

*This document is intended to be use as a guide for the design and implementation of the software and will help the customer to understand its functionality. This document is intended for both the client: UTB and the software engineering team.*

## 1.2 Scope

## *The Software to be develop is the QsARC: An accreditation Software or UTB. This software will provide an application that is an easy-to-use graphical Interface that allows the user to store accreditation data and generate requested reports. The software will have the capability to store different information such as: University information, department information and also will provide students statistics. student learning outcomes and POS-program of Study. The software is enhanced with the option to generate reports for undergraduate students (txt,xml,html, and pdf format). For Graduate students will be available only in pdf format.*

## 1.3 Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
| Term | Definition |
| Database | Data Structure that stores information |
| GUI | Graphical User Interface |
| HTML | Hypertext Markup Language |
| IDE | Interactive Development Environment |
| IEEE | Institute of Electrical and Electronics Engineer |
| PDF | Portable Document File |
| POS | Program of Study |
| SRS | Software Requirements Specification |
| TXT | Text |
| UTB | University of Texas at Brownsville |
| XML | Extensible Markup Language |

## 

## 1.4 References

IEEE Guide to Software Requirements Specification (ANSI/IEEE Std. 830-1984). The SRS templates of Dr. Orest Pilskalns (WSU, Vancover) and Jack Hagemeister (WSU, Pullman).

## 1.5 Overview

*This document is divide in two sections, the first one is related to the overall description of the*

*requirements of the product.*

# 2. General Description

## 2.1 Product Perspective

## *The Software will have a wizard interface to facilitates the input of the data The software is a Graphical user interface and desktop based. Since this is a data-based driven product it will nedd somewhere to store the data. For that, a database will be used. The application will use the database to store and get data. All of the database communication will go over the network.*

## 2.2 Product Functions

- Wizard application to input the data

-Easy navigation inside the application

-Access to a data base to store and get data

-Delete information for an input.

-Show information about Department, faculty, student, and programs.

-Get reports of information in different formats (TXT,web,PDF).

## 2.3 User Characteristics

The user is expected to have basic education level such as read and write.Also basic understanding and experience with computers or other technical devices.

## 2.4 General Constraints

*No Constrains at the moment.*

## 2.5 Assumptions and Dependencies

One assumption about the product is that will use a data base to collect data and it will be enough largely cover all the inputs.

# 

# 3. Specific Requirements

## 3.1 External Interface Requirements

There is only one external interface that will be the data base server interface use to save and get data.

### 3.1.1 User Interfaces

At the first time the user get in the application, he will be drive to the wizard interface to choose what type of moment he wants to perform such as: Input Information for Faculty, Student or Department, delete Information of Get output report in one of the available formats.

### 3.1.2 Hardware Interfaces

There is no hardware interface, everything is designed to be working with the application and the network.

### 3.1.3 Software Interfaces

The software application will communicate to the network database in order to input, read and modify the data

### 3.1.4 Communications Interfaces

The communication Interface will be achieved between the software application and the server data base.

## 3.2 Functional Requirements

The software application shall display a wizard interface to request operation to be perform.

The system shall access the database to store data and access data. Also the application shall generate reports in different formats.

The system shall be friendly use to navigate easily through the application.

The system shall be able to delete data previously stored.

The system shall be able to display the data previously collect such as the University information, Department Information, faculty or student information, so then could be exported as report.

## 3.3 Non-Functional Requirements

### 

### 3.3.1 Performance

### The performance is not yet specify, but the main functions should not take more than 10 seconds. Also the performance for save and recollect data will depends in the bandwidth

### 3.3.2 Reliability

No reliability was request as this moment.

### 3.3.3 Availability

The application should be connected to the Internet in order to communicate to the data base server, then needs to be available at any moment.

### 3.3.4 Security

No Security required at this moment.

### 3.3.5 Maintainability

The administrator will be in charge for the maintenance of the records of the server.

### 3.3.6 Portability

The software will be a desktop based application.

## 3.4 Inverse Requirements

The software system shall not allow multiple use of tasks such as getting report and introducing a new entry .

## 3.5 Design Constraints

No Design constrains at the momen.t

## 3.6 Logical Database Requirements

There is no limitations for the database storage. To generate reports will be only available in the following formats: txt,pdf, or web format.

## 3.7 Other Requirements

No Other requirements at the moment.

# 4. Analysis Models

## 4.1 Sequence Diagrams

## 

## 4.3 Data Flow Diagrams (DFD)

## 4.2 State-Transition Diagrams (STD)

# 5. Change Management Process

# A. Appendices

## A.1 Appendix 1

## A.2 Appendix 2