Software Requirements   
Specification

VaqPack: Graduate-to-Professional Aid Pack  
Version 1.0  
November 5, 2015

**Project Manager**  
Mario Gracia III   
**Team**  
Carlos O. Silva  
Eduardo Tovar  
Jeffrey Serna  
Jonathan Monroe

Software Engineering CSCI-3340-02

Instructor: MK Quweider, Ph.D.

Fall 2014

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Description** | **Author** | **Comments** |
| 11/5/15 | Initial Version | PM/Team | Initial creation |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Document Approval

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Signature** | **Printed Name** | **Title** | **Date** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**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 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 2**

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**

3.2.1 User Login/Registration 3

3.2.2 Wizard (resume) 3  
3.2.3 Wizard (cover) 3

3.2.4 Wizard (Business Card)…………..……...……………………………………………………………………….....4

3.2.5 Printing/Exporting 4

3.2.6 Email………………….………...………………………………………………………………………..*.*..4

**3.3 Non-Functional Requirements 5**

**3.5.1 Performance 5**

**3.5.2 Reliability 5**

**3.5.3 Availability 5**

**3.5.4 Security 5**

**3.5.5 Maintainability 5**

**3.5.6 Portability 5**

**3.4 Design Constraints 5**

**3.5 Logical Database Requirements 5**

4. Change Management Process 5

# 1. Introduction

## 1.1 Purpose

The purpose of this Software Requirements Specification is to give a detailed description of the requirements of VaqPack: Graduate-to-Professional Aid Package. It will explain the purpose and features of the application, the interfaces of the application, what the application will do, the constraints under which it must operate and how the system is intended to function. This document is intended for both the users and the developers of the system and will be proposed to the supervisor for approval.

## 1.2 Scope

The VaqPack will be a free software given to graduating students looking for work. It will create resumes/business cards based on data inputted by the student and send them out to prospective employers. All data will be stored in a database which the student will be able to access via login/password.

## 1.3 Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
| Term | Definition |
| GUI | Graphical User Interface |
| Database | An organized way to store data. (Uses tables) |
| Client | Hardware or software that requests services from a server. |
| Server | A central repository of data. |
| MySQL | Open-source relational database management system. |
| Agile Process | A project management process |
| IDE | Integrated Development Environment |
| GitHub | Web-based Git repository hosting service. |
| JRE | Java Runtime Environment |
| UML | Unified Modeling Language |

## 1.4 Overview

This SRS document has a General Description followed by the Specific Requirements. The General Description gives a simple overview of the product’s functionality. Specific requirements keeps the developer in mind and gives them a more detailed view of the project.

# 2. General Description

## 2.1 Product Perspective

This software is to be a wizard with a client-server functionality, storing information for each user in a database.

## Product Functions

* The user should be able to register for the first time or retrieve information based on a SQL-query.
* Enter the information in text-fields. Start with the personal information, followed by the objectives, employment history, skills, references, etc.
* Store information in database
* You should allow the user to choose a major heading and fill the information for that heading
* The user should be able to select from different themes
* GUI should have at least 4 themes for the resume and two themes for the business card.
* The user should be shown sample themes to select from
* Export HTML/PDF
* User has option to omit non-applicable and unnecessary fields.
* Email PDF to employers

**2.3 User Characteristics**

The users are expected to be graduating university students of any major looking for work. They are expected to have basic understanding on how to use a wizard and it’s GUI.

## 2.4 General Constraints

* Implemented in Java
* Uses MySQL
* Must create an HTML and PDF
* Version Control (GitHub)

## 2.5 Assumptions and Dependencies

Assume the database can support a large amount of users. No dependencies at this time.

# 3. Specific Requirements

## 3.1 External Interface Requirements

### 3.1.1 User Interfaces

VaqPack will be operated using a wizard.

### 3.1.2 Hardware Interfaces

The current hardware interface will be a standard computer that is capable of running java based platforms. No other hardware specifications are required.

### 3.1.3 Software Interfaces

The software being used to create the VaqPack is the java programming language. Since java is being used, then the only software that will be required to run the program is a Java Virtual Machine. Java is one of the most popular programming languages and will be updated continually.

### 3.1.4 Communications Interfaces The VaqPack should communicate with a dedicated server and will also have email capabilities. Both of which will require internet connectivity.

## 3.2 Functional Requirements

### 3.2.1 User Login/Registration

**3.2.1.1 Introduction**

This allows the user to access his/her information. Or register if they don’t have an account.

**3.2.1.2 Inputs**

Username and password will be required to gain access to their account.

**3.2.1.3 Processing**

The user inputs will include username/password for either logging in or registering.

**3.2.1.4 Outputs**

Existing user will gain access to their account if successfully logged in. If not they will be given the chance to try again. No output for new user.

**3.2.1.5 Error Handling**

Error handling will be controlled by allowing a limited number of attempts to login. After the user has exceeded the number of attempts they will have to wait a period of time to try and login again.

### 3.2.2 Wizard (Resume)

**3.2.2.1 Introduction**

The wizard will take the user through steps to create their resume.

**3.2.2.2 Inputs**

Personal information, Objectives, Employment History, Skills, References, Theme Selection, etc.

**3.2.2.3 Processing**

Collects/stores data and preferences of the user.

**3.2.2.4 Outputs**

The wizard will call other functions to handle output.

**3.2.2.5 Error Handling**

Check if data entered matches fields. If certain fields left blank enter a default variable.

### 3.2.3 Wizard (Cover)

**3.2.3.1 Introduction**

This feature will generate a cover letter.

**3.2.3.2 Inputs**

Source, Date, Job Title, Reference Number, Etc.

**3.2.3.3 Processing**

Gathers data and formats.

**3.2.3.4 Outputs**

The wizard will call other functions to handle output.

**3.2.3.5 Error Handling**

If no data found, display error message.

### 3.2.4 Wizard (Business Card)

**3.2.4.1 Introduction**

This feature will generate a business card.

**3.2.4.2 Inputs**

Personal Information

**3.2.4.3 Processing**

Gathers data and formats.

**3.2.4.4 Outputs**

The wizard will call other functions to handle output.

**3.2.4.5 Error Handling**

If no data found, display error message.

### 3.2.5 Printing/Exporting

**3.2.5.1 Introduction**

Exports data into XML, HTML, and PDF

**3.2.5.2 Inputs**

Data collected from wizards resume/cover or database.

**3.2.5.3 Processing**

Using data collected will generate xml and then HTML/PDF

**3.2.5.4 Outputs**

XML, HTML, PDF files.

**3.2.5.5 Error Handling**

If error occurs display message.

### 3.2.4 Email

**3.2.6.1 Introduction**

Emails resume PDF

**3.2.6.2 Inputs**

PDF resume, employer email address, user email.

**3.2.6.3 Processing**

Using emails collected will send PDF to designated recipients.

**3.2.6.4 Outputs**

Email

**3.2.6.5 Error Handling**

If error occurs state that email was not sent.

## 3.3 Non-Functional Requirements

### 3.3.1 Performance

The program must perform in a fluid manner. Should not take long to perform most operations.

### 3.3.2 Reliability

Should be able to handle all possible inputs and have little to no errors.

### 3.3.3 Availability

Software should be able to run as long as there is internet/server connectivity.

### 3.3.4 Security

Registered user data can only be accessed with correct username/password.

### 3.3.5 Maintainability

Code should have minimal coupling, optimized cohesion, and good flexibility. This will keep cost low in maintaining software in the future.

### 3.3.6 Portability

As long as the system can run the JRE the software should be usable. (Platform-Independent)

## 3.4 Design Constraints

There are no current constraints given to the design of this project.

## 3.5 Logical Database Requirements

The current requirements of the database are unknown and will be determined as the project continues.

# 4. Change Management Process

Anyone who wants to submit a change to the SRS can do so. The submitted changes will be sent to the project manager who will give the final approval based on his opinion of whether or not the changes are valid and needed.