**Software Design Documentation**

**VaqPack**

Graduate-to-Professional

Aid Pack

Version 1.0

November 22nd, 2015

**Lead Software Engineer:**

William Dewald

**Project Team:**

Fernando Bazan

Nathanael Carr

Erik Lopez

Raul Saavedra

Prepared for

Software Engineering

University of Texas Rio Grande Valley

Instructor: MK Quweider, Ph.D.

Fall 2015

**Table of Contents**

**1. INTRODUCTION**……………………………………………………………..**1**

1.1 PURPOSE………………………………………………………………………….1

1.2 SCOPE……………………………………………………………………………..1

1.3 DEFINITIONS, ACRONYMS, AND ABBREVIATIONS……………………….1

**2. REFERENCES**…………………………………………………………………**3**

**3. DECOMPOSITION DESCRIPTION**…………………………………………..

3.1 MODULE DECOMPOSITION.................................................................................

3.1.1 Module 1 Decomposition…………………………………………………………...

3.1.2 Module 2 Decomposition …………………………………………………………...

3.2 CONCURRENT PROCESS DECOMPOSITION………………………………….

3.2.1 Process 1 Decomposition ……………………………………………………………

3.2.2 Process 2 Decomposition.............................................................................................

3.3 DATA DECOMPOSITION………………………………………………………...

3.3.1 Data Entry 1 Decomposition ……..…………..………………………………………

3.3.2 Data Entry 2 Decomposition ………………………………………………………..

**4. DEPENDENCY DESCRIPTION**……………………………………………….

4.1 INTER-MODULE DEPENDENCIES.......................................................................

4.2 INTER-PROCESS DEPENDENCIES……………………………………………...

4.3 DATA DEPENDENCIES…………………………………………………………..

**5. INTERFACE DESCRIPTION**.............................................................................

5.1 MODULE INTERFACE……………………………………………………………

5.1.1 Module 1 Interface……….…………………………………………………………...

5.1.2 Module 2 Interface…..…...…………………………………………………………...

5.2 PROCESS INTERFACE……………………………………………………………

5.2.1 Process 1 Interface……………………………………………………………………

5.2.2 Process 2 Interface........................................................................................................

**6. DETAILED DESIGN**……………………………………………………………

6.2 DATA DETAILED DESIGN.......................................................................................

6.1.01: VP\_BusinessCard.java........................................................................................

6.1.02: VP\_Button.java....................................................................................................

6.1.03: VP\_Center.java....................................................................................................

6.1.04: VP\_Contact.java.................................................................................................

6.1.05: VP\_CoverLetter.java........................................................................................

6.1.06: VP\_DataManager.java.....................................................................................

6.1.07: VP\_DataToHtml.java.......................................................................................

6.1.08: VP\_DatabaseManager.java.................................................................................

6.1.09: VP\_DivisionLine.java.....................................................................................

6.1.10: VP\_ErrorHandler.java..........................................................................................

6.1.11: VP\_FieldLabel.java...............................................................................................

6.1.12: VP\_FileManager.java............................................................................................

6.1.13 VP\_Footer.java........................................................................................................

6.1.14: VP\_GUIController.java.........................................................................................

6.1.15: VP\_Header.java......................................................................................................

6.1.16: VP\_HtmlToPdf.java...............................................................................................

6.1.17: VP\_Loader.java....................................................................................................

6.1.18: VP\_Mail.java........................................................................................................

6.1.19: VP\_PageDivision.java........................................................................................

6.1.20: VP\_Paragraph.java..............................................................................................

6.1.21: VP\_PasswordField.java......................................................................................

6.1.22: VP\_Resume.java.................................................................................................

6.1.23: VP\_Sounds.java...................................................................................................

6.1.24: VP\_TextField.java................................................................................................

6.1.25: VP\_Theme.java.....................................................................................................

6.1.26: VP\_Tree.java..........................................................................................................

6.1.27: VP\_User.java..........................................................................................................

6.1.28: VaqPack.java..........................................................................................................

6.2 DATA DETAILED DESIGN…………………………………...………………….

6.2.01 User Access Level......................................................................................................

6.2.02 Business Card Information.......................................................................................

6.2.03 Business Card PDF....................................................................................................

6.2.04 Contact Information...................................................................................................

6.2.05 Cover Letter Information............................................................................................

6.2.06 Cover Letter PDF............................................................................................

6.2.07 Custom Theme..........................................................................................................

6.2.08 Registering User......................................................................................................

6.2.09 Reset Code..............................................................................................................

6.2.10 Resume Information...............................................................................................

6.2.11 Resume HTML........................................................................................................

6.2.12 Resume PDF............................................................................................................

6.2.13 User..........................................................................................................................

6.2.14 User Data.................................................................................................................

**7. USER INTERFACES BY USE CASE**………………………………………..

7.01 Case01-Initial Administrator System Configuration.............................................

7.02 Case02-Administrator System Configuration Password Mismatch......................

7.03 Case03-User Creates New Account........................................................................

7.04 Case04-User Login.............................................................................................

7.05 Case05-Update Personal Information.................................................................

7.06 Case06-Update Personal Information-Required field........................................

7.07 Case07-Update Personal Information-Incorrect format.....................................

7.08 Case08-User Forgets Password..........................................................................

7.09 Case09-New Password Not Strong Enough.......................................................

7.10 Case10-Inactivity Timeout……………….........................................................

**A. APPENDICES**

A.1 Appendix 1........................................................................................................

A.2 Appendix 2........................................................................................................

**1. Introduction \*(rough draft)**

**1.1 Purpose \***

This Software Design Document is made with the purpose of explicitly outlining the software architecture and high level design of the VaqPack Graduate to Professional Aid Pack application. Through the use of various architectural decompositions, the various modules, data structures, databases and interfaces will be depicted. The intention of this document is to provide developers an insight into meeting our Client’s needs efficiently and effectively, corresponding with the requirements set forth in the SRS. Therefore, this document is mainly intended for the developers, present company as well as future organizations.

**1.2 Scope\***

The software application described throughout this SDD is the VaqPack Graduate to Professional Aid Pack, or simply VaqPack. This SDD is intended for a base level system in order to provide a proof of concept for the use of building an evolutionary prototype that demonstrates the functionality specified by the corresponding SRS. This will be achieved through the use of use-case models, state models, class models and data flow models that will serve to clarify the development teams thought process during implementation.

**1.3 Definitions, Acronyms, and Abbreviations \***

The following terms, acronyms, and abbreviations are used throughout this document and are presented in the table below by order of appearance.

|  |  |
| --- | --- |
| **Term** | **Definition** |
| SDD | Software Design Description; A written description of a software product, that a software designer writes in order to give a software development team overall guidance to the architecture of the software project. |
| SRS | Software Requirement Specification; A comprehensive description of the intended purpose and environment for software under development. The SRS fully describes what the software will do and how it will be expected to perform. |
| VaqPack | VaqPack Graduate to Professional Aid Pack, in short. |
| GUI | Graphical User Interface; provides a visual, interactive means for a software user to manipulate the controls, commands, or features of that software. |
| Wizard | A sequential set of prompts for input, assisting in data collection and organized such that its implementation increases ease of use. |
| Database | A structured collection of data that can be efficiently and conveniently accessed. |
| JDO | Java Data Object; A standard way to access persistent data in databases, using plain old Java objects (POJO) to represent persistent data. |
| POJO | Plain Old Java Object; An ordinary Java object, not bound by any special restriction. |
| IDE | Integrated Development Environment; software that provides tools for the development and organization of programming code. |
| JVM | Java Virtual Machine; Provides the necessary links allowing a java program to run on a machine using a particular operating system. |
| JRE | Java Runtime Environment; Including the Java Virtual Machine, all necessary components for a system to establish the environment in which Java programs will run. |
| DBA | Directs or performs all activities related to maintaining a successful database environment. |
| SQL | Structured Query Language; the standard relational database query language. |
| JDBC | Java Database Connectivity; a Java API developed by Oracle Corporation which provides methods for querying and updated a database. |
| XML | Extensible Markup Language; a markup language that defines a set of rules for encoding documents in a format which is both human-readable and machine-readable. |
| XSL | eXtensible Stylesheet Language; a style sheet language for XML documents. |
| HTML | HyperText Markup Language; the web standard language used in the delivery of online content, interpreted and rendered by web browsers. |
| PDF | Portable Document Format; a popular electronic document file type particularly used with rich-text or styled text. |

**2.** **References**

Git - <https://git-scm.com/>

GitHub - <https://github.com/>

HTML - <http://www.w3schools.com/html/>

Java Virtual Machine - <https://java.com/en/download/>

Java Runtime Environment - <http://www.oracle.com/technetwork/java/javase/downloads/jre8-downloads-2133155.html>

JavaFX - <http://docs.oracle.com/javase/8/javase-clienttechnologies.htm>

MySQL - <http://dev.mysql.com/downloads/mysql/>

NetBeans - <https://netbeans.org/>

**3.** **Decomposition Description**

**3.1 Module Decomposition**

**3.1.1 Module 1 Decomposition**

Blah blah blah.

**3.1.2 Module 2 Decomposition**

Blah blah blah.

**3.2 Concurrent Process Decomposition**

**3.2.1 Process 1 Decomposition**

Blah blah blah.

**3.2.2 Process 2 Decomposition**

Blah blah blah.

**3.3 Data Decomposition**

**3.3.1 Data Entry 1 Decomposition**

Blah blah blah.

**3.3.2 Data Entry 2 Decomposition**

Blah blah blah.

**4.** **Dependency Description**

**4.1 Inter-module Dependencies**

Blah blah blah.

**4.2 Inter-process Dependencies**

Blah blah blah.

**4.3 Data Dependencies**

Blah blah blah.

**5.** **Interface Description**

**5.1 Module Interface**

**5.1.1 Module 1 Interface**

Blah blah blah.

**5.1.2 Module 2 Interface**

Blah blah blah.

**5.2 Process Interface**

**5.2.1 Process 1 Interface**

Blah blah blah.

**5.2.2 Process 2 Interface**

Blah blah blah.

**6.** **Detailed Design**

****

**6.1 Module Detailed Design**

**6.1.01: VP\_BusinessCard.java**

Attributes:

Methods:

**6.1.02: VP\_Button.java**

Attributes:

Methods:

**6.1.03: VP\_Center.java**

Attributes:

Methods:

**6.1.04: VP\_Contact.java**

Attributes:

Methods:

**6.1.05: VP\_CoverLetter.java**

Attributes:

Methods:

**6.1.06: VP\_DataManager.java**

Attributes:

Methods:

**6.1.07: VP\_DataToHtml.java**

Attributes:

Methods:

**6.1.08: VP\_DatabaseManager.java**

Attributes:

Methods:

**6.1.09: VP\_DivisionLine.java**

Attributes:

Methods:

**6.1.10: VP\_ErrorHandler.java**

Attributes:

Methods:

**6.1.11: VP\_FieldLabel.java**

Attributes:

Methods:

**6.1.12: VP\_FileManager.java**

Attributes:

Methods:

**6.1.13 VP\_Footer.java**

Attributes:

Methods:

**6.1.14: VP\_GUIController.java**

Attributes:

Methods:

**6.1.15: VP\_Header.java**

Attributes:

Methods:

**6.1.16: VP\_HtmlToPdf.java**

Attributes:

Methods:

**6.1.17: VP\_Loader.java**

Attributes:

Methods:

**6.1.18: VP\_Mail.java**

Attributes:

Methods:

**6.1.19: VP\_PageDivision.java**

Attributes:

Methods:

**6.1.20: VP\_Paragraph.java**

Attributes:

Methods:

**6.1.21: VP\_PasswordField.java**

Attributes:

Methods:

**6.1.22: VP\_Resume.java**

Attributes:

Methods:

**6.1.23: VP\_Sounds.java**

Attributes:

Methods:

**6.1.24: VP\_TextField.java**

Attributes:

Methods:

**6.1.25: VP\_Theme.java**

Attributes:

Methods:

**6.1.26: VP\_Tree.java**

Attributes:

Methods:

**6.1.27: VP\_User.java**

Attributes:

Methods:

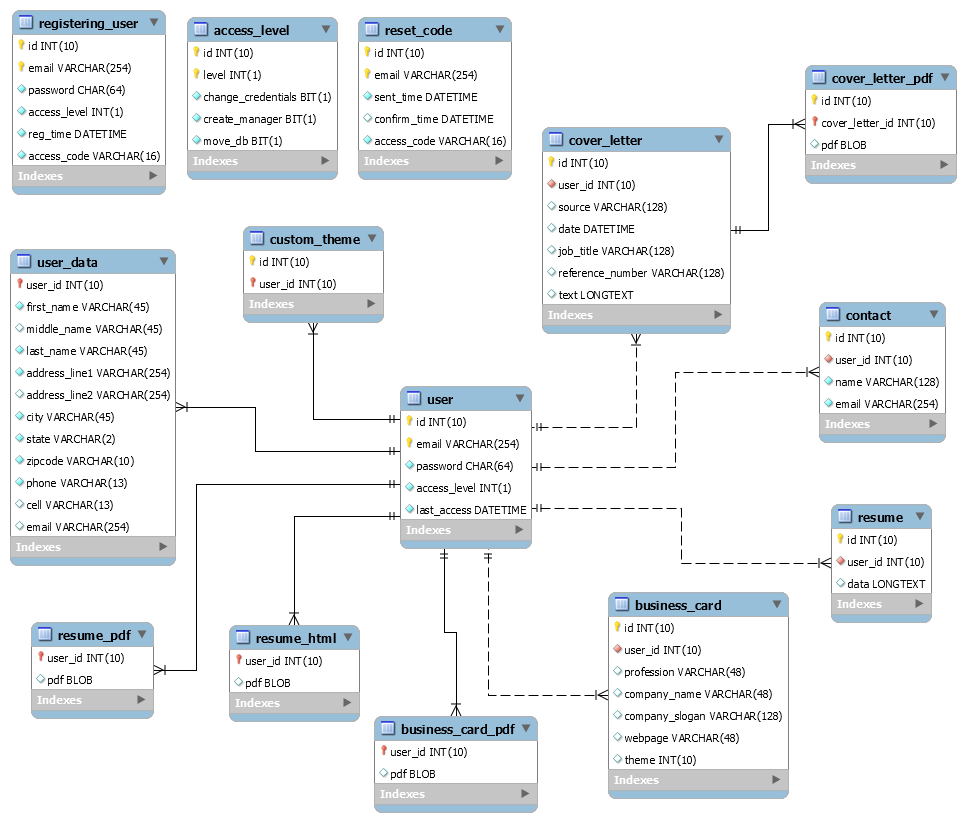
**6.1.28: VaqPack.java**

Attributes:

Methods:

**6.2 Data Detailed Design**

**6.2.01 User Access Level**



|  |  |
| --- | --- |
| **Table:** | **access\_level** |
| **Description** | Used to store and define the access level for each unique user. |
| **Attributes** | **Description** |
| id | ID, integer, not null, auto-increment. |
| level | Access Level, integer, not null. |
| change\_credentials | Privilege, bit, not null. |
| create\_manager | Privilege, bit, not null. |
| move\_db | Privilege, bit, not null. |

**6.2.02 Business Card Information**

|  |  |
| --- | --- |
| **Table:** | **business\_card** |
| **Description** | Used to store business card information. |
| **Attributes** | **Description** |
| id | ID, integer, not null, auto-increment. |
| user\_id | User ID, integer, not null. |
| profession | Use to store user profession. |
| company\_name | Use to store user company name. |
| company\_slogan | Use to store user company slogan. |
| webpage | Use to store user webpage address. |
| theme | Integer. Use to apply theme. Default value = -1 |

**6.2.03 Business Card PDF**

|  |  |
| --- | --- |
| **Table:** | **business\_card\_pdf** |
| **Description** | Used to store user business card PDF. |
| **Attributes** | **Description** |
| user\_id | User ID, integer, not null. |
| pdf | Business card PDF blob |

**6.2.04 Contact Information**

|  |  |
| --- | --- |
| **Table:** | **Contact** |
| **Description** | Used to store user contact information. |
| **Attributes** | **Description** |
| id | ID, integer, not null, auto-increment. |
| user\_id | User ID, integer, not null. |
| name | User contact name. |
| email | User contact email. |

**6.2.05 Cover Letter Information**

|  |  |
| --- | --- |
| **Table:** | **cover\_letter** |
| **Description** | Used to store cover letter information. |
| **Attributes** | **Description** |
| id | ID, integer, not null, auto-increment. |
| user\_id | User ID, integer, not null. |
| source |  |
| date | Date. |
| job\_title |  |
| reference\_number |  |
| text | Longtext. |

**6.2.06 Cover Letter PDF**

|  |  |
| --- | --- |
| **Table:** | **cover\_letter\_pdf** |
| **Description** | Used to store cover letter PDF files |
| **Attributes** | **Description** |
| id | ID, integer, not-null, auto-increment. |
| cover\_letter\_id | Cover letter ID, integer, not null. |
| pdf | Cover letter PDF blob. |

**6.2.07 Custom Theme**

|  |  |
| --- | --- |
| **Table:** | **custom\_theme** |
| **Description** | Applies custom theme |
| **Attributes** | **Description** |
| id | ID, integer, not-null, auto-increment. |
| user\_id | User ID, integer, not-null. |

**6.2.08 Registering User**

|  |  |
| --- | --- |
| **Table:** | **registering\_user** |
| **Description** | Used to store user registration data. |
| **Attributes** | **Description** |
| id | ID, integer, not-null, auto-increment. |
| email | User email address for creating account. |
| password | User password. |
| access\_level | User access\_level, default = 0. |
| reg\_time | Current timestamp. |
| access\_code | Access code sent to user upon creating account. |

**6.2.09 Reset Code**

|  |  |
| --- | --- |
| **Table:** | **reset\_code** |
| **Description** | Used to store reset codes. |
| **Attributes** | **Description** |
| id | ID, integer, not-null, auto-increment. |
| email | User email account email address. |
| sent\_time | Current timestamp. |
| confirm\_time | Time user confirms access code. |
| access\_code | Access code sent to user upon resetting password. |

**6.2.10 Resume Information**

|  |  |
| --- | --- |
| **Table:** | **resume** |
| **Description** | Used to store resume information. |
| **Attributes** | **Description** |
| id | ID, integer, not-null, auto-increment. |
| user\_id | User ID. |
| data | Longtext containing user resume information. |

**6.2.11 Resume HTML**

|  |  |
| --- | --- |
| **Table:** | **resume\_html** |
| **Description** | Used to store user resume HTML |
| **Attributes** | **Description** |
| user\_id | User ID |
| html | Resume HTML blob |

**6.2.12 Resume PDF**

|  |  |
| --- | --- |
| **Table:** | **resume\_pdf** |
| **Description** | Used to store user resume PDF |
| **Attributes** | **Description** |
| user\_id | User ID |
| pdf | Resume PDF blob |

**6.2.13 User**

|  |  |
| --- | --- |
| **Table:** | **User** |
| **Description** | Used to store user credentials. |
| **Attributes** | **Description** |
| id | ID, integer, not-null, auto-increment. |
| email | User login email, not null. |
| password | User login password hash, not null. |
| access\_level | Access level for user, default = 0. |
| last\_access | Last time user accessed account, default = current timestamp. |

**6.2.14 User Data**

|  |  |
| --- | --- |
| **Table:** | **user\_data** |
| **Description** | Used to store basic user data. |
| **Attributes** | **Description** |
| user\_id | User ID, not null. |
| first\_name | First name, not null. |
| middle\_name | Middle name, optional. |
| last\_name | Last name, not null. |
| address\_line1 | Current address, not null. |
| address\_line2 | Additional address line, optional. |
| city | City, not null. |
| state | State, not null. |
| zipcode | Zip code, not null. |
| phone | Contact phone number, not null. |
| cell | Cell phone number, optional. |
| email | Email address, optional. |

**7. User Interfaces by Use Case Scenarios**

**7.1**

**Use Case 01: Initial Administrator System Configuration.**

VaqPack shall allow the system administrator to configure and connect to the MySQL DB using a previously established DBA account upon initial installation. Upon creating a VaqPack Administrator account the user shall have access to the special administrator functions that are locked within VaqPack to normal users that allow for the Administrator to govern within the program.

**A Valid Scenario**

1. User runs VaqPack

2. User correctly inputs MySQL URL and PORT#

3. User successfully inputs MySQL admin username and password

4. User successfully creates initial VaqPack administrator account

**Invalid Scenarios**

1. User runs VaqPack

2. User incorrectly inputs MySQL URL or PORT#

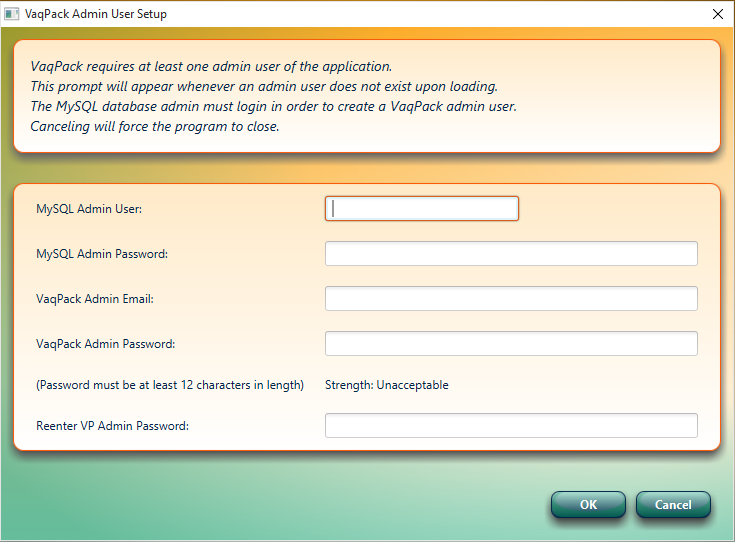
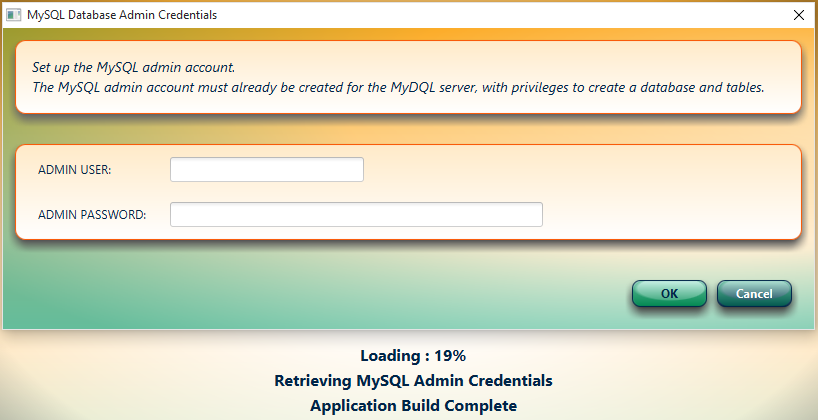
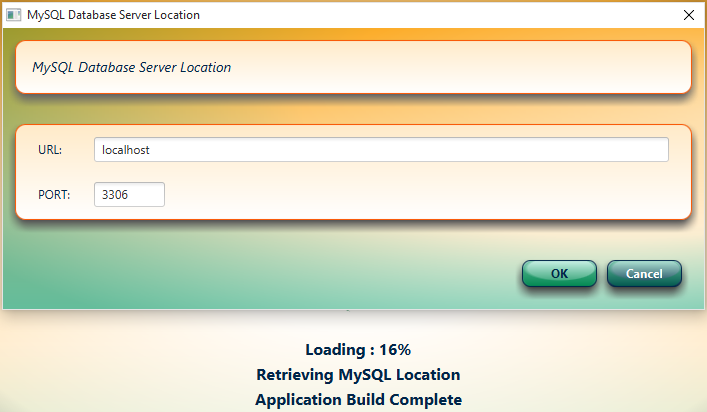
3. VaqPack prompts error

1. User runs VaqPack

2. User correctly inputs MySQL URL and PORT#

3. User unsuccessfully inputs MySQL admin or password

4. VaqPack prompts error

**Case 01 – Initial Administrator System Configuration**

**7.2**

**Use Case 02: Administrator System Configuration Password Mismatch**

**Case Scenario**

1. User runs VaqPack

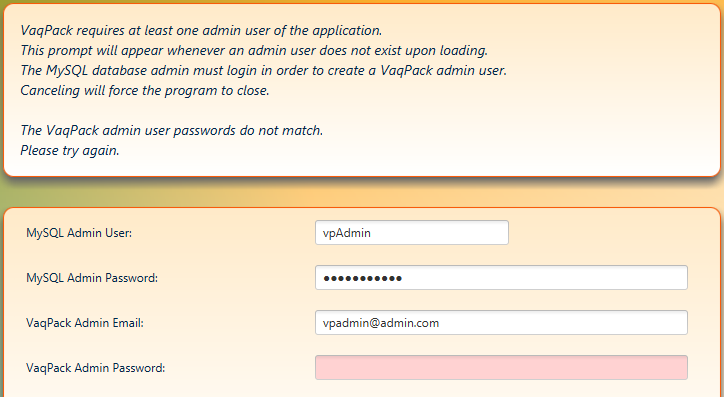
2. User correctly inputs MySQL URL and PORT#

3. User successfully inputs MySQL admin and password

4. User unsuccessfully inputs fields for initial VaqPack administrator account

5. VaqPack prompts error

**Case 02**



**7.3**

**Use Case03: User Creates New Account**

VaqPack shall allow and require users to create an account as stated in the SRS.

**A Valid Scenario**

1. User in login window, clicks on “need an account?”

2. User successfully inputs email and password

3. User retrieves VaqPack system generated code sent to the supplied email

4. User successfully inputs email password and code

5. User logs into VaqPack

**An invalid Scenario**

1. User in login window, clicks on “need an account?”

2. User unsuccessfully inputs email and password

3. VaqPack prompts error

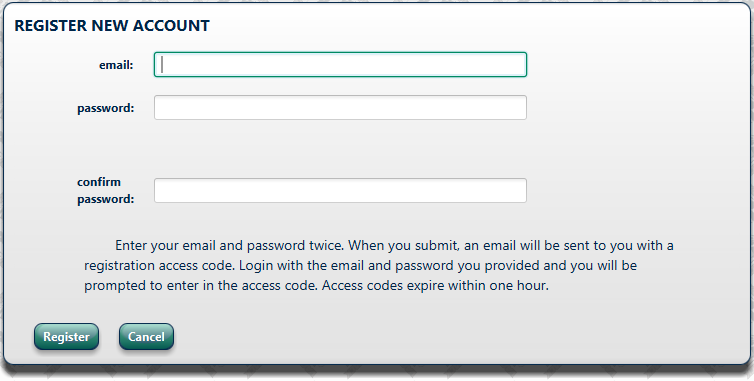
1. User in login window, clicks on “need an account?”

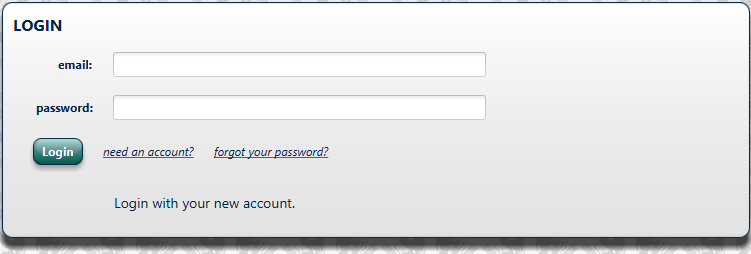
2. User successfully inputs email and password

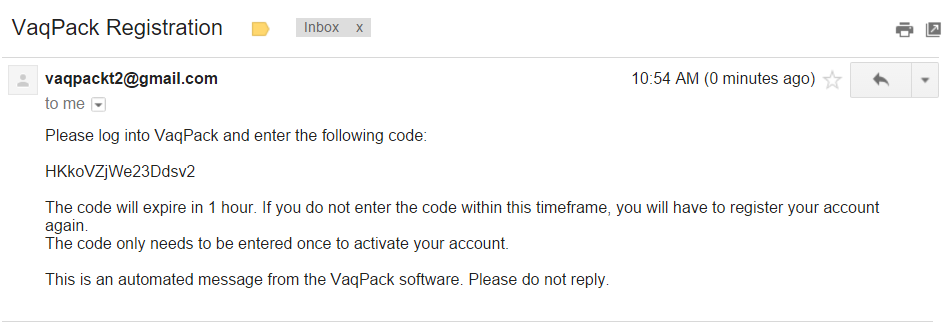
3. User unsuccessfully inputs email password and code

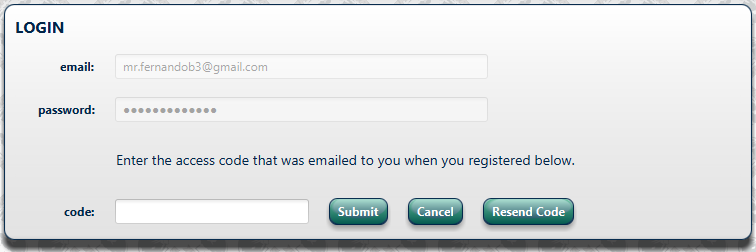
4. VaqPack prompts error

**Case 03 – User Creates New Account Successfully**

****

****

****

****

**7.4**

**Use Case04: User Login**

VaqPack shall allow users that have created an account to log in to gain access to their personal information as stated in the SRS.

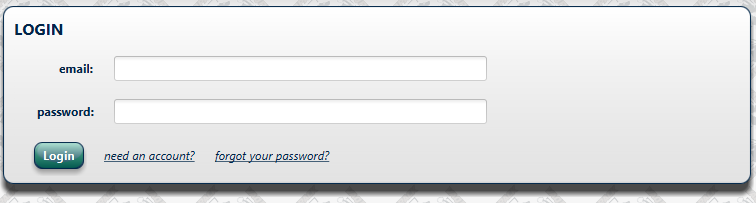
**Case Scenario**

1. User in login window

2. User successfully inputs email and password

3. User logs into VaqPack

**Case 04**

****

**7.5**

**Use Case05: User Updates Personal Information Successfully**

Upon successful log in, VaqPack will require users to Update Personal Information as an initial step.

**Case Scenario**

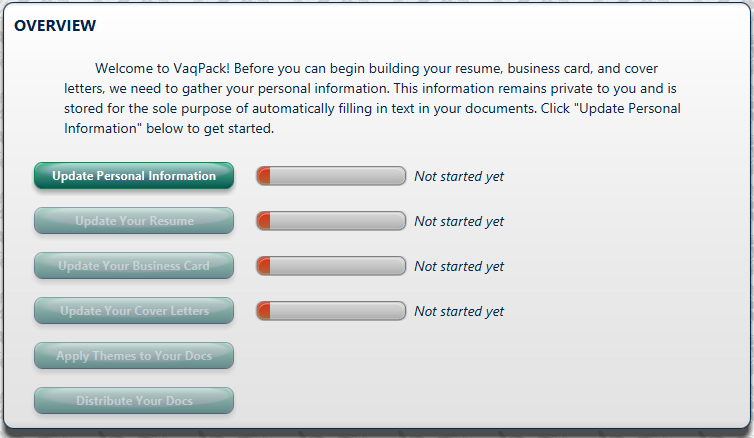
1. User is in the Overview page

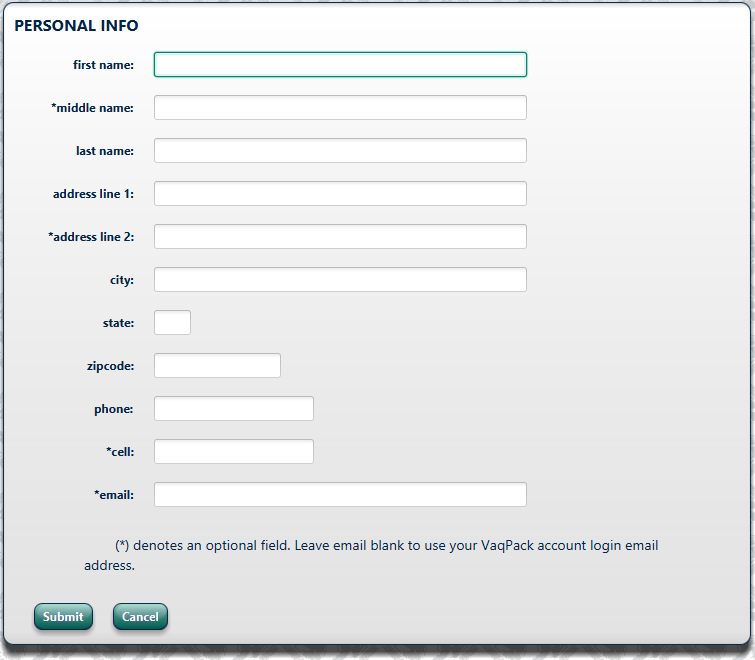
2. User clicks “Update Personal Information”

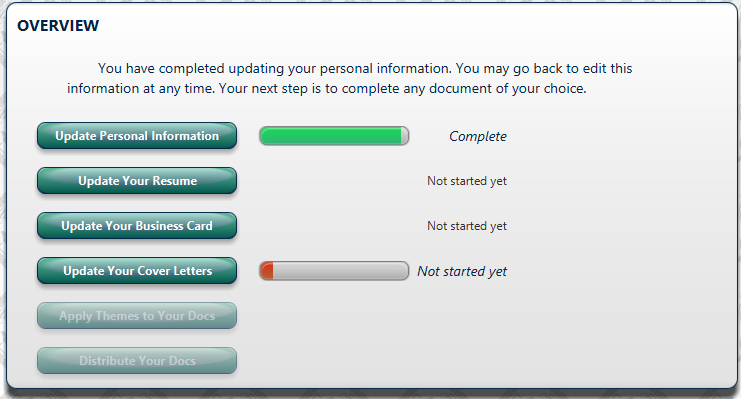
3. User successfully updates personal Information

4. User gain access to additional program features via enabled buttons

**Case 05**

****

****

****

**7.6**

**Use Case06: User Updating Personal Information – Missing Required Field**

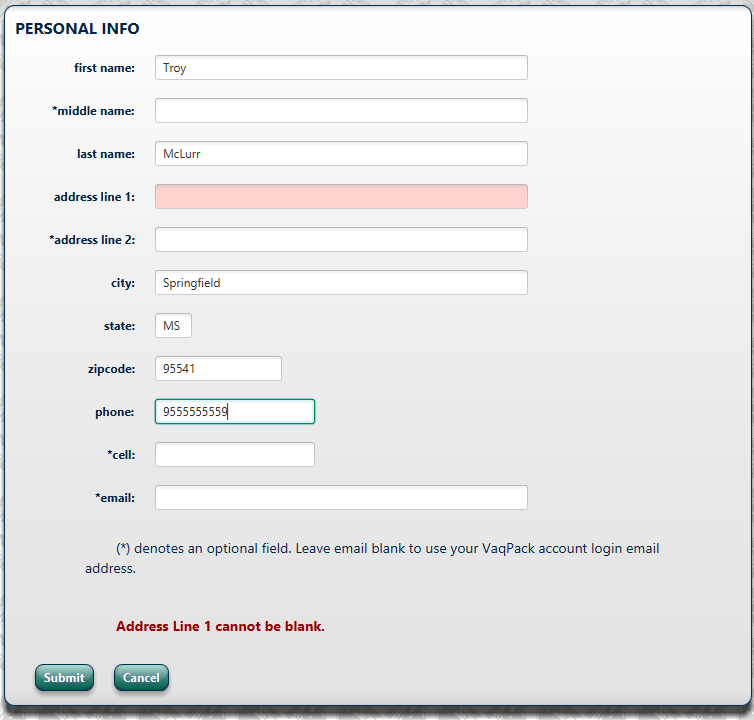
**Case Scenario**

1. User is in the Personal Info page

2. User leaves a required field blank

3. VaqPack highlights missing field, text displays which fields are missing

**Case 06**

****

**7.7**

**Use Case07: User Updating Personal Information – Incorrect Format**

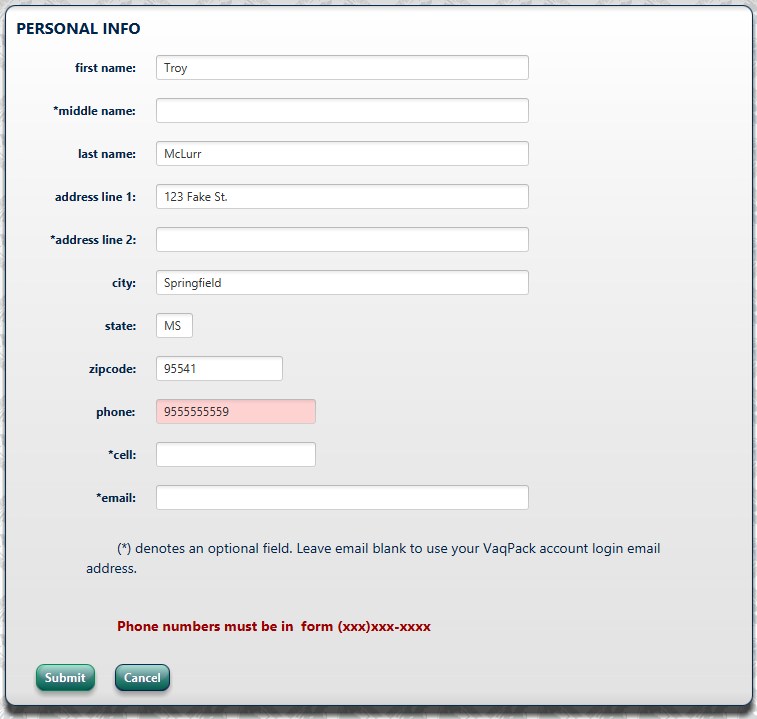
**Case Scenario:**

1. User is in the Personal Info page

2. User inputs a required field incorrectly

3. VaqPack highlights problem field and supplies correct format parameters

**Case 07**

****

**7.8**

**Use Case08: User Forgets Password**

VaqPack shall allow users to reset their passwords upon completing a verification process consisting of confirming a code sent to the email supplied upon creating the account.

**Case Scenario:**

1. User is in login page

2. User clicks on “forgot your password?”

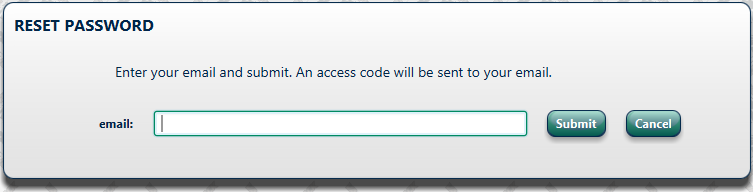
3. User successfully inputs email and clicks submit.

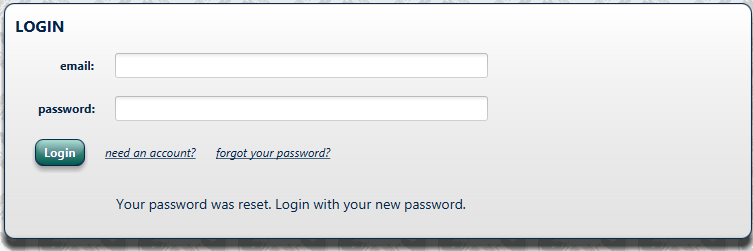
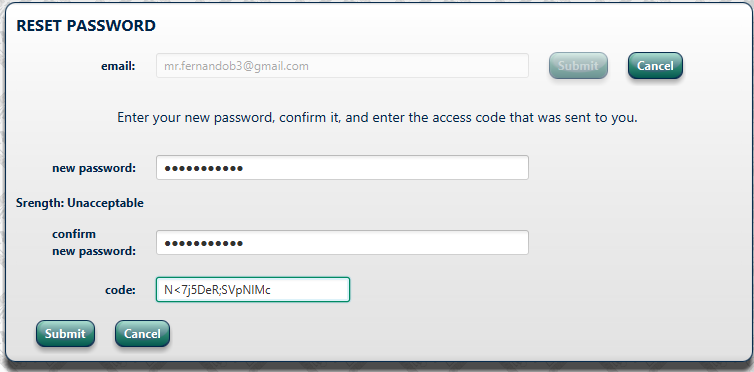
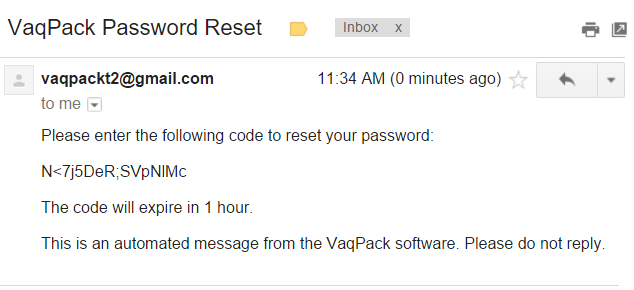
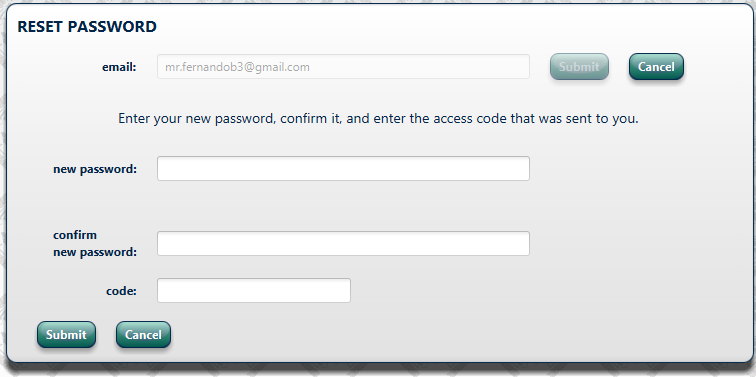
4. VaqPack sends email

5. User retrieves code from email

6. User updates password upon code verification

**Case 08 – User Forgets Password**

****

****

**7.9**

**Use Case09: Users New Password Not Strong Enough**

VaqPack requires a certain minimum strength/length before it accepts any users password choice.

**Case Scenario:**

1. User is in login page

2. User clicks on “forgot your password?”

3. User successfully inputs email and clicks submit

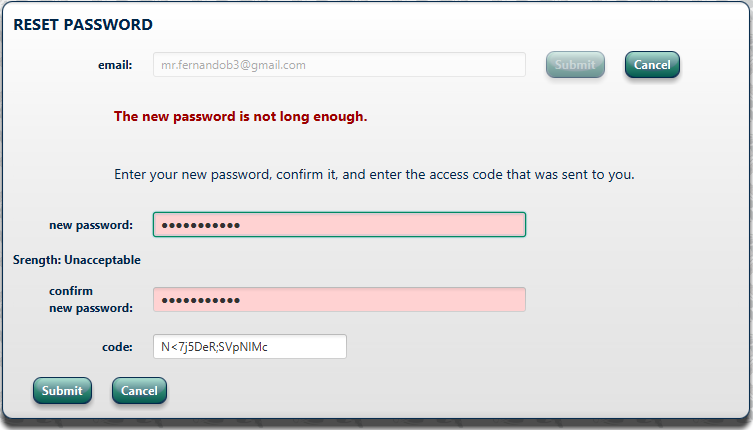
4. VaqPack sends email

5. User retrieves code from email

6. User updates password but doesn’t supply a strong/long enough password

7. VaqPack informs user to choose a stronger password

**Case 09 – User New Password Not Strong Enough**

****

**7.10**

**Use Case10: Users is inactive, automatic logout**

VaqPack shall have a built in logout timer to log out a user after 5 minutes of inactivity. A warning window will popup when there is 30 seconds left. Any mouse movement or keyboard key presses will reset the timer.

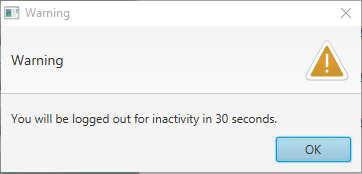
**Case Scenario:**

1. User is inactive for 4minutes 30 seconds

2. VaqPack prompts warning window

3. 30 seconds later the user will automatically be logged out

**Case 10 – User Is Inactive, automatic logout**

****

**A.** **Appendices**

**A.1 Appendix 1**

Required form for the approval of changes to this SDD document:

**Document Approval**

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Signature** | **Printed Name** | **Title** | **Date** |
|  | William Dewald | Lead Software Engineer |  |
|  | Dr. M.K. Quweider | Instructor, CSCI-3340 |  |

**A.2 Appendix 2**

Required form must be attached to the end of this document if there are any changes after its initial completion:

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Description** | **Author** | **Comments** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |