**Software Design Documentation**

**Healthy Living Application**

Version 1.0

November 27th, 2017

Raymundo Vilano

Prepared for

Software Engineering

Instructor: MK Quweider, Ph.D.

Fall 2017

## Table of Contents

1. **INTRODUCTION**……………………………………………………………..**1** 1.1 PURPOSE………………………………………………………………………….1
   1. SCOPE………………………………………………………………………....…..1
   2. DEFINITIONS, ACRONYMS, AND ABBREVIATIONS……………………….1

1. **REFERENCES**………………………………………………………………**3**

1. **DETAILED DESIGN**………………………………………………………..…4

3.1 MODULE DETAILED DESIGN.................................................................................4

3.1.01: VP\_BusinessCard.java.......................................................................................4

3.1.02: VP\_Button.java...................................................................................................5

3.1.03: VP\_Center.java...................................................................................................6

3.1.04: VP\_Contact.java................................................................................................9

3.1.05: VP\_CoverLetter.java......................................................................................9

3.1.06: VP\_DataManager.java...................................................................................10

3.1.07: VP\_DataToHtml.java.....................................................................................12

3.1.08: VP\_DatabaseManager.java...............................................................................12

3.1.09: VP\_DivisionLine.java...................................................................................15

3.1.10: VP\_ErrorHandler.java......................................................................................15

3.1.11: VP\_FieldLabel.java............................................................................................16

3.1.12: VP\_FileManager.java.......................................................................................16

3.1.13 VP\_Footer.java....................................................................................................17

3.1.14: VP\_GUIController.java.....................................................................................17

3.1.15: VP\_Header.java..................................................................................................20

3.1.16: VP\_HtmlToPdf.java...........................................................................................20

3.1.17: VP\_Loader.java................................................................................................21

3.1.18: VP\_Mail.java......................................................................................................21

3.1.19: VP\_PageDivision.java......................................................................................22

* 1. DATA DETAILED DESIGN…………………………………...……………..…….28
     1. User Access Level...................................................................................................28
     2. Business Card Information....................................................................................29

### 4. USER INTERFACES BY USE CASE………………………………………33

4.01 Case01-Initial Administrator System Configuration...........................................35

4.02 Case02-Administrator System Configuration Password Mismatch....................36

4.03 Case03-User Creates New Account....................................................................37

4.04 Case04-User Login...........................................................................................38

4.05 Case05-Update Personal Information..............................................................39

4.06 Case06-Update Personal Information-Required field......................................40

4.07 Case07-Update Personal Information-Incorrect format...................................41

4.08 Case08-User Forgets Password.......................................................................42

4.09 Case09-New Password Not Strong Enough....................................................44

4.10 Case10-Inactivity Timeout……………….......................................................45

#### A. APPENDICES

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

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

### 1. Introduction

#### 1.1 Purpose

This Software Design Document is made with the purpose of explicitly outlining the software architecture and high level design of the Healthy Living 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 Healthy Living Application. 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. |
| 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/jre8downloads-2133155.html](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. Detailed Design

#### 3.1 Module Detailed Design

\*INSERT MODULE DETAILED DESIGN HERE\*

##### 3.1.01: Environment.java

|  |  |
| --- | --- |
| **Class Name:** Environment | |
| **Brief description:** | |
| **Attributes (fields)** | **Attribute Description** |
|  |  |
| **Methods (operations)** | **Method Description** |
|  |  |

##### 

##### 3.1.02: ConfigurationManager.java

|  |  |
| --- | --- |
| **Class Name:** ConfigurationManager | |
| **Brief description:** | |
| **Attributes (fields)** | **Attribute Description** |
|  |  |
| **Methods (operations)** | **Method Description** |
|  |  |

##### 3.1.03: Tools.java

|  |  |
| --- | --- |
| **Class Name:** Tools | |
| **Brief description:** | |
| **Attributes (fields)** | **Attribute Description** |
|  |  |
| **Methods (operations)** | **Method Description** |
|  |  |

##### 3.1.04: Database.java

|  |  |
| --- | --- |
| **Class Name:** Database | |
| **Brief description:** | |
| **Attributes (fields)** | **Attribute Description** |
|  |  |
| **Methods (operations)** | **Method Description** |
|  |  |

##### 3.1.05: menuItem.java

|  |  |
| --- | --- |
| **Class Name:** menuItem | |
| **Brief description:** The VP\_Button class is a custom button class. | |
| **Attributes (fields)** | **Attribute Description** |
|  |  |
| **Methods (operations)** | **Method Description** |
|  |  |

##### 3.1.06: menuManager.java

|  |  |
| --- | --- |
| **Class Name:** menuManager | |
| **Brief description:** | |
| **Attributes (fields)** | **Attribute Description** |
|  |  |
| **Methods (operations)** | **Method Description** |
|  |  |

##### 3.1.07: subMenuItem.java

|  |  |
| --- | --- |
| **Class Name:** subMenuItem | |
| **Brief description:** | |
| **Attributes (fields)** | **Attribute Description** |
|  |  |
| **Methods (operations)** | **Method Description** |
|  |  |

##### 3.1.08: AboutManager.java

|  |  |
| --- | --- |
| **Class Name:** AboutManager | |
| **Brief description:** | |
| **Attributes (fields)** | **Attribute Description** |
|  |  |
| **Methods (operations)** | **Method Description** |
|  |  |

##### 3.1.09: adminDashboardManager.java

|  |  |
| --- | --- |
| **Class Name:** adminDashboardManager | |
| **Brief description:** | |
| **Attributes (fields)** | **Attribute Description** |
|  |  |
| **Methods (operations)** | **Method Description** |
|  |  |

##### 3.1.10: HealthyDietManager.java

|  |  |
| --- | --- |
| **Class Name:** HealthyDietManager | |
| **Brief description:** | |
| **Attributes (fields)** | **Attribute Description** |
|  |  |
| **Methods (operations)** | **Method Description** |
|  |  |

##### 3.1.11: calculatorManager.java

|  |  |
| --- | --- |
| **Class Name:** calculatorManager | |
| **Brief description:** | |
| **Attributes (fields)** | **Attribute Description** |
|  |  |
| **Methods (operations)** | **Method Description** |
|  |  |

##### 3.1.12: dietManager.java

|  |  |
| --- | --- |
| **Class Name:** dietManager | |
| **Brief description:** | |
| **Attributes (fields)** | **Attribute Description** |
|  |  |
| **Methods (operations)** | **Method Description** |
|  |  |

##### 3.1.13: foodGroup.java

|  |  |
| --- | --- |
| **Class Name:** foodGroup | |
| **Brief description:** | |
| **Attributes (fields)** | **Attribute Description** |
|  |  |
| **Methods (operations)** | **Method Description** |
|  |  |

##### 3.1.14: foodItem.java

|  |  |
| --- | --- |
| **Class Name:** foodItem | |
| **Brief description:** | |
| **Attributes (fields)** | **Attribute Description** |
|  |  |
| **Methods (operations)** | **Method Description** |
|  |  |

##### 3.1.15: itemDisplayer.java

|  |  |
| --- | --- |
| **Class Name:** itemDisplayer | |
| **Brief description:** | |
| **Attributes (fields)** | **Attribute Description** |
|  |  |
| **Methods (operations)** | **Method Description** |
|  |  |

##### 3.1.16: searchItemManager.java

|  |  |
| --- | --- |
| **Class Name:** searchItemManager | |
| **Brief description:** | |
| **Attributes (fields)** | **Attribute Description** |
|  |  |
| **Methods (operations)** | **Method Description** |
|  |  |

##### 3.1.17: useritemsManager.java

|  |  |
| --- | --- |
| **Class Name:** useritemsManager | |
| **Brief description:** | |
| **Attributes (fields)** | **Attribute Description** |
|  |  |
| **Methods (operations)** | **Method Description** |
|  |  |

##### 3.1.18: exitManager.java

|  |  |
| --- | --- |
| **Class Name:** exitManager | |
| **Brief description:** | |
| **Attributes (fields)** | **Attribute Description** |
|  |  |
| **Methods (operations)** | **Method Description** |
|  |  |

##### 3.1.19: homeManager.java

|  |  |
| --- | --- |
| **Class Name:** homeManager | |
| **Brief description:** | |
| **Attributes (fields)** | **Attribute Description** |
|  |  |
| **Methods (operations)** | **Method Description** |
|  |  |

#### 3.2 Data Detailed Design

\*INSERT DATA DETAILED DESIGN HERE\*

##### 3.2.01 Food Information

|  |  |
| --- | --- |
| **Table:** | **foods** |
| **Description** | Store all food items |
| **Attributes** | **Description** |
| id | ID, integer, not null, auto-increment. |
| name | Use to store food name. |
| groupId | Use to store group id. |
| image | Use to store image file name. |
| calories | Use to store number of calories. |
| fats | Use to store number of fats. |
| cholesterol | Use to store number of cholesterol. |
| sodium | Use to store number of sodium. |
| potassium | Use to store number of potassium. |
| carbohydrates | Use to store number of carbohydrates. |
| proteins | Use to store number of proteins. |

##### 3.2.02 Food Groups ID

|  |  |
| --- | --- |
| **Table:** | **food\_groups** |
| **Description** | Used to store food groups id. |
| **Attributes** | **Description** |
| id | ID, integer, not null, auto-increment. |
| name | Use to store food group name. |

#### 4. User Interfaces by Use Case Scenarios

##### 4.1

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

Healthy Living Application 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 Healthy Living Application Administrator account the user shall have access to the special administrator functions that are locked within Healthy Living Application to normal users that allow for the Administrator to govern within the program.

###### A Valid Scenario

1. A
2. B
3. C
4. D

###### Invalid Scenarios

1. A
2. B
3. C
4. D

##### Case 01 – Initial Administrator System Configuration

\*INSERT IMAGE\*

##### 4.2

###### Use Case 02: Administrator System Configuration Password Mismatch Case Scenario

1. A
2. B
3. C
4. D

###### Case 02

\*INSERT IMAGE\*

##### 4.3

**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. A
2. B
3. C
4. D

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

\*INSERT IMAGE\*

##### 4.4

###### 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. A
2. B
3. C
4. D

###### Case 04

\*INSERT IMAGE\*

### 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

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |