***Copyright © 2002 by Karl E. Wiegers. Permission is granted to use, modify, and distribute this document.***

**Software Requirements**

**Specification**

**for**

**<Project>**

**Version 1.0 approved**

**Prepared by <Darwin, Joshua and eugene>**

**<Group C1>**

**<9 February 2018>**

Table of Contents

[1. Introduction 3](#_Toc506033360)

[1.1. Purpose 3](#_Toc506033361)

[1.2. Document Conventions 3](#_Toc506033362)

[1.3. Intended Audience and Reading Suggestions 3](#_Toc506033363)

[1.4. Project Scope 3](#_Toc506033364)

[1.5. References 4](#_Toc506033365)

[2. Overall Description 5](#_Toc506033366)

[3. System Features 6](#_Toc506033367)

[3.1. Login function 6](#_Toc506033368)

[3.1.1. Description and Priority 6](#_Toc506033369)

[3.1.2. Stimulus/Response Sequences 6](#_Toc506033370)

[3.1.3. Functional Requirements 6](#_Toc506033371)

[3.2. Staff Control Panel 6](#_Toc506033372)

[3.2.1. Description and Priority 6](#_Toc506033373)

[3.2.2. Stimulus/Response Sequences 6](#_Toc506033374)

[3.2.3. Functional Requirements 6](#_Toc506033375)

[3.3. Admin Control Panel 6](#_Toc506033376)

[3.3.1. Description and Priority 6](#_Toc506033377)

[3.3.2. Stimulus/Response Sequences 6](#_Toc506033378)

[3.3.3. Functional Requirements 6](#_Toc506033379)

# Introduction

## Purpose

The purpose of this Software Requirements Specification (SRS) document is to provide a detailed description of the functionalities of the Warehouse Management Tool system. This document will cover each of the system’s intended features, as well as offer a preliminary glimpse of the software application’s User Interface (UI). The document will also cover hardware, software, and various other technical dependencies.

## Document Conventions

This document is unique. Its scope is to describe the requirements of the Warehouse Management Tool system and the interfaces for it. This document has been created before the actual program is constructed, hence is it important to note that future changes to the program should be included in this requirement specification document, in order to maintain its usefulness. Since the program has not been implemented yet, this document can be used as a manual for development of the system.

## Intended Audience and Reading Suggestions

This document may be used by the staff or clients of the Warehouse Management Tool system. They can easily understand the features and the operations of this system. They can also be informed of all the possibilities the program provides. Users can get a general image of the program and use it with better ease and understanding. This document could be studies by the programmers, so they have the chance to identify and alter possible problems that needs improvement. Also, they can fully understand the nature of the program and implement new features to it. It is recommended to study the document carefully in order to fully understand the program and features as well as identifying errors. Basically, this document helps users and testers to understand how the Warehouse Management Tool system is supposed to work and also to identify bugs and errors.

## Project Scope

Project will be developed under the scenario of Warehouse Management Tool System, WM tool system in short. Web portal for this domain is currently not specified. An effective Warehouse Management Tool is essential for streamlining process, long range planning and optimization. It gives a better understanding of day-to-day operation with its summary information of how a warehouse is performing. The managers could use this data for making critical management decisions.

It is critical for the warehouse management to have an effective system in place due to various reasons such as security, accuracy, availability and so on. We are requested to use a test-driven approach to develop the warehouse management tool which needs to meet the following requirements.

List of features include:

* Ability to add/update records on the incoming stock and outgoing stock easily.
* Track the stock in accordance to their category and sub-category.
* Ability to search and display available stock for a particular category and/or sub-category.
* Ability to search and display stock in accordance to price range and quantity in ascending and descending order.
* Daily, weekly or monthly summary report of total incoming and outgoing stock details.
* Provide login authentication process.
* Login security feature with option of encrypting the data stored.
* If there are three unsuccessful login attempts, the record will be marked as “locked” and one is not allowed to login to the system.
* The system shall encrypt and decrypt the password.
* Provide stock item alerts when it drops below a threshold set previously.

## References

Referenced to CSCI222\_Assignment1 words document for the features included.

# Overall Description

**2.1** **Product Perspective**

This product is a new, self-contained product we will be designing and coding this product from scratch.

**2.2 Product Features**

This product consists several features, first admin will be allowed to create, delete and unlock staff account. Staff account can add, edit, remove, search and do a summary report for daily, weekly, monthly and yearly.

**2.3 User Classes and Characteristics**

We have identified two user classes which are stock and webpage. The frequency of use will be high as once the staff log in they will be using this two classes. Webpage class

**2.4 Operating Environment**

This program will run in either windows or Linux based operating system.

**2.5 Design and Implementation Constraints**

For language it will all be in English, security we will be encrypting both stock data file and user credentials. Program will be coded in C++

**2.6 User Documentation**

Not applicable as this is still in the design phase.

**2.7 Assumptions and Dependencies**

We assume that this program will be running in windows or linux based operating system as user has requested for program to be coded in C++.

# System Features

## Login function

### Description and Priority

Allows the user to log in then proceeds to authenticate the user before directing them to the appropriate control panel based on their role. High priority.

### Stimulus/Response Sequences

* Login
  + Wrong username/ password
  + Account locked out
* Role Validation
  + Staff
  + Admin

### Functional Requirements

* REQ-1: Login function must be able to authenticate user based on entered username and password.
* REQ-2: Login function must be able to encrypt password when authenticating user.
* REQ-3: Login function must be able to read the user’s roles and restrict access to certain functions.
* REQ-4: Users with admin roles may not be locked out of their accounts.

## Staff Control Panel

### Description and Priority

Used primarily by staff to add and remove stocks from database. They can also display the summary report of a period of time. High priority.

### Stimulus/Response Sequences

* Search Product
* Edit Product
* Add Product
* Remove Product
* Summary Report
* Retrieve Product

### Functional Requirements

* REQ-1: Must be able to encrypt and decrypt the database file when manipulating fields.

## Admin Control Panel

### Description and Priority

Used by staff with admin roles to manage accounts. Includes the ability to Medium Priority.

### Stimulus/Response Sequences

* Create account
* Reset Account
* Delete Account
* View Account

### Functional Requirements

* REQ-1: Must only be accessible by users with admin roles
* **4.** **External Interface Requirements**
* 4.1 User Interfaces

Upon start-up of program a simple GUI will appear. First is the login interfaces basically user just need to enter username and password. After successful login there will be a main menu consisting of what the user is able to do within the program itself. Any error input message will be for like example “error, invalid input”

* 4.2 Hardware Interfaces

As the program can be run in either Linux or windows based operating systems and is not a complicated program all user need is a working laptop or desktop and the program will run fine.

* 4.3 Software Interfaces

For this program there will not be any user of integrated commercial components, as it will be done from scratch and it does not need internet connection for program to work and all data will be store in a text file so no data or message will be coming into the system or leaving.

* 4.4 Communications Interfaces

No communications interfaces is required as the program does not need to have internet connection for it to run is a standalone program.

**5. Other Non-functional Requirements**

* 5.1 Performance Requirements

<If there are performance requirements for the product under various circumstances, state them here and explain their rationale, to help the developers understand the intent and make suitable design choices. Specify the timing relationships for real time systems. Make such requirements as specific as possible. You may need to state performance requirements for individual functional requirements or features.>

* 5.2 Safety Requirements

<Specify those requirements that are concerned with possible loss, damage, or harm that could result from the use of the product. Define any safeguards or actions that must be taken, as well as actions that must be prevented. Refer to any external policies or regulations that state safety issues that affect the product’s design or use. Define any safety certifications that must be satisfied.>

* 5.3 Security Requirements

<Specify any requirements regarding security or privacy issues surrounding use of the product or protection of the data used or created by the product. Define any user identity authentication requirements. Refer to any external policies or regulations containing security issues that affect the product. Define any security or privacy certifications that must be satisfied.>

* 5.4 Software Quality Attributes

<Specify any additional quality characteristics for the product that will be important to either the customers or the developers. Some to consider are: adaptability, availability, correctness, flexibility, interoperability, maintainability, portability, reliability, reusability, robustness, testability, and usability. Write these to be specific, quantitative, and verifiable when possible. At the least, clarify the relative preferences for various attributes, such as ease of use over ease of learning.>

Appendix A: Glossary

<Define all the terms necessary to properly interpret the SRS, including acronyms and abbreviations. You may wish to build a separate glossary that spans multiple projects or the entire organization, and just include terms specific to a single project in each SRS.>

Appendix B: Analysis Models

<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams.> Appendix C: Issues List

< This is a dynamic list of the open requirements issues that remain to be resolved, including TBDs, pending decisions, information that is needed, conflicts awaiting resolution, and the like.>

Appendix C: Issues List

< This is a dynamic list of the open requirements issues that remain to be resolved, including TBDs, pending decisions, information that is needed, conflicts awaiting resolution, and the like.>