**Software Requirements Specification**

**for**

**Warehouse management System**

**Version 1.0 pending revision**

**Prepared by Lam How Wei**

**Group 6**

**2th February 2018**

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

***Software Requirements Specification for Warehouse Management System Page ii***

**Table of Contents**

Table of Contents .......................................................................................................................... ii Revision History ............................................................................................................................ ii 1. Introduction..............................................................................................................................1 1.1 Purpose ............................................................................................................................................ 1 1.2 Document Conventions.................................................................................................................... 1 1.3 Intended Audience and Reading Suggestions.................................................................................. 1 1.4 Project Scope ................................................................................................................................... 1 1.5 References........................................................................................................................................1 2. Overall Description..................................................................................................................2 2.1 Product Perspective ......................................................................................................................... 2 2.2 Product Features .............................................................................................................................. 2 2.3 User Classes and Characteristics ..................................................................................................... 2 2.4 Operating Environment.................................................................................................................... 2 2.5 Design and Implementation Constraints.......................................................................................... 2 2.6 User Documentation ........................................................................................................................ 2 2.7 Assumptions and Dependencies ...................................................................................................... 3 3. System Features........................................................................................................................3 3.1 System Feature 1.............................................................................................................................. 3 3.2 System Feature 2 (and so on)........................................................................................................... 4 4. External Interface Requirements ...........................................................................................4 4.1 User Interfaces ................................................................................................................................. 4 4.2 Hardware Interfaces......................................................................................................................... 4 4.3 Software Interfaces .......................................................................................................................... 4 4.4 Communications Interfaces ............................................................................................................. 4 5. Other Nonfunctional Requirements.......................................................................................5 5.1 Performance Requirements.............................................................................................................. 5 5.2 Safety Requirements ........................................................................................................................ 5 5.3 Security Requirements..................................................................................................................... 5 5.4 Software Quality Attributes............................................................................................................. 5 6. Other Requirements ................................................................................................................5 Appendix A: Glossary....................................................................................................................5 Appendix B: Analysis Models.......................................................................................................6 Appendix C: Issues List.................................................................................................................6

**1. Introduction**

**1.1 Purpose**

*<Identify the product whose software requirements are specified in this document, including the revision or release number. Describe the scope of the product that is covered by this SRS, particularly if this SRS describes only part of the system or a single subsystem.>*

**1.2 Document Conventions**

*<Describe any standards or typographical conventions that were followed when writing this SRS, such as fonts or highlighting that have special significance. For example, state whether priorities for higher-level requirements are assumed to be inherited by detailed requirements, or whether every requirement statement is to have its own priority.>*

**1.3 Intended Audience and Reading Suggestions**

*This document is intended for developers, project managers, marketing staff, users, testers. It contains*

*Information on the product .its feature,its purpose,goal and what it was intended for.Readers are strongly recommended to read the overall description section for the summarized  details about the product for full feature read the requirement and non requirement section.*

**1.4 Project Scope**

*the stock management software enables  companys and their management to manage and keep track of the stocks add stocks, update stocks accordingly and also use the information stored in the system for critical bussiness and non bussiness decisions. This is greatly benefical as more time would be saved from doing manual calculations and therefore improve productivity.*

**1.5 References**

*<List any other documents or Web addresses to which this SRS refers. These may include user interface style guides, contracts, standards, system requirements specifications, use case documents, or a vision and scope document. Provide enough information so that the reader could access a copy of each reference, including title, author, version number, date, and source or location.>*

**2. Overall Description**

**2.1 Product Perspective**

*<Describe the context and origin of the product being specified in this SRS. For example, state whether this product is a follow-on member of a product family, a replacement for certain existing systems, or a new, self-contained product. If the SRS defines a component of a larger system, relate the requirements of the larger system to the functionality of this software and identify interfaces between the two. A simple diagram that shows the major components of the overall system, subsystem interconnections, and external interfaces can be helpful.>*

**2.2 Product Features**

*<Summarize the major features the product contains or the significant functions that it performs or lets the user perform. Details will be provided in Section 3, so only a high level summary is needed here. Organize the functions to make them understandable to any reader of the SRS. A picture of the major groups of related requirements and how they relate, such as a top level data flow diagram or a class diagram, is often effective.>*

**2.3 User Classes and Characteristics**

*<Identify the various user classes that you anticipate will use this product. User classes may be differentiated based on frequency of use, subset of product functions used, technical expertise, security or privilege levels, educational level, or experience. Describe the pertinent characteristics of each user class. Certain requirements may pertain only to certain user classes. Distinguish the favored user classes from those who are less important to satisfy.>*

**2.4 Operating Environment**

*<Describe the environment in which the software will operate, including the hardware platform, operating system and versions, and any other software components or applications with which it must peacefully coexist.>*

**2.5 Design and Implementation Constraints**

*<Describe any items or issues that will limit the options available to the developers. These might include: corporate or regulatory policies; hardware limitations (timing requirements, memory requirements); interfaces to other applications; specific technologies, tools, and databases to be used; parallel operations; language requirements; communications protocols; security considerations; design conventions or programming standards (for example, if the customer’s organization will be responsible for maintaining the delivered software).>*

**2.6 User Documentation**

*<List the user documentation components (such as user manuals, on-line help, and tutorials) that will be delivered along with the software. Identify any known user documentation delivery formats or standards.>*

**2.7 Assumptions and Dependencies**

*<List any assumed factors (as opposed to known facts) that could affect the requirements stated in the SRS. These could include third-party or commercial components that you plan to use, issues around the development or operating environment, or constraints. The project could be affected if these assumptions are incorrect, are not shared, or change. Also identify any dependencies the project has on external factors, such as software components that you intend to reuse from another project, unless they are already documented elsewhere (for example, in the vision and scope document or the project plan).>*

**3. System Features**

*<This template illustrates organizing the functional requirements for the product by system features, the major services provided by the product. You may prefer to organize this section by use case, mode of operation, user class, object class, functional hierarchy, or combinations of these, whatever makes the most logical sense for your product.>*

**3.1 System Feature 1**

*<Don’t really say “System Feature 1.” State the feature name in just a few words.>*

3.1.1 Description and Priority

*<Provide a short description of the feature and indicate whether it is of High, Medium, or Low priority. You could also include specific priority component ratings, such as benefit, penalty, cost, and risk (each rated on a relative scale from a low of 1 to a high of 9).> 3.1.2 Stimulus/Response Sequences*

*<List the sequences of user actions and system responses that stimulate the behavior defined for this feature. These will correspond to the dialog elements associated with use cases.> 2.1.3 Functional Requirements*

*<Itemize the detailed functional requirements associated with this feature. These are the software capabilities that must be present in order for the user to carry out the services provided by the feature, or to execute the use case. Include how the product should respond to anticipated error conditions or invalid inputs. Requirements should be concise, complete, unambiguous, verifiable, and necessary. Use “TBD” as a placeholder to indicate when necessary information is not yet available.>*

*<Each requirement should be uniquely identified with a sequence number or a meaningful tag of some kind.>*

REQ-1: REQ-2:

**4. External Interface Requirements**

**4.1 User Interfaces**

*The user logs in and sees a menu with the option to*

*1)add/update stocks*

*2)check monthly daily yearly stock report*

*3)search type of stock*

*4)track and sort stock*

*5)alert when stock is low*

*User then enter choice and the respective information will display*

**4.2 Hardware Interfaces**

*The Software interacts with a keyboard where the user keys in information such  which is then process by the software to execute its respective tasks depending on the information keyed in.*

**4.3 Software Interfaces**

*The software extracts and modifies the database base on the input given by the user*

**4.4 Communications Interfaces**

*<Describe the requirements associated with any communications functions required by this product, including e-mail, web browser, network server communications protocols, electronic forms, and so on. Define any pertinent message formatting. Identify any communication standards that will be used, such as FTP or HTTP. Specify any communication security or encryption issues, data transfer rates, and synchronization mechanisms.>*

**5. Other Nonfunctional Requirements**

**5.1 Performance Requirements**

*Menu style of user interface and is able to use c++ on linux interface*

**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**

*The System has a log in log out feature and is able to encrypt / decrypt data to protect unintended party from accessing sensitive information. Extra feature such as getting locked out after 3 tries ensures that the party cannot brute force their way through the system making it alot more secure.*

**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.>*

**6. Other Requirements**

*<Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.> 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.>*