SOFTWARE REQUIREMENTS SPECIFICATION

VERSION 1.0

OCTOBER 22, 2013

UTB CLASSICLIT BOOK READER

REDA YACOUBY (TEAM MANAGER)
ANTHONY MEZA
DANIEL VILLALPANDO
JESSICA ZAMBRANO

UNIVERSITY OF TEXAS AT BROWNSVILLE LIBRARY AND MULTIMEDIA SERVICES

COSC-5/4365 GROUP PROJECT, SUPERVISED BY DR. M. K. QUWEIDER

TABLE OF CONTENTS

1. INTRODUCTION

1.1. PURPOSE

The purpose of this Software Requirements Specification is to give a detailed description of the ClassicLit Book Reader application. It will explain the purpose and features of the application, its interfaces, constraints, dependencies, functionality, and attributes.

This document will serve as a guide for the design and implementation of the software, and will help clarifying the description of the software for the customer. The intended audience of this document is both the client: the University of Texas at Brownsville Library and Multimedia Services, and the software engineering team in charge of implementing it.

1.2. SCOPE

The software described in this document is the ClassicLit Book Reader. This application will provide an easy-to-use graphical interface to read e-books, and manage a library of e-books hosted in a dedicated server. It will contain several helpful tools and options to further enhance the reading experience. The software will be distributed to the students of the University of Texas at Brownsville, for free usage. It will permit the students to have access to a large database of e-books that are out-of-copyright, and download them easily in their personal computers. The staff of the University of Texas at Brownsville Library and Multimedia Services will be able to manage the e-books library database using the software.

1.3. DEFINITIONS, ACRONYMS & ABBREVIATIONS

Term	Definition
GUI	Graphical User Interface
IDE	Interactive Development Environment
Database	An organized collection of data.
CVS Tool	Concurrent Version System
SRS	Software Requirements Specification
UTB-LMS	University of Texas at Brownsville Library and Multimedia Services
UTB	University of Texas at Brownsville

1.4 REFERENCES

1.5 OVERVIEW

The SRS is organized in two main sections. First the Overall Description section presents the high-level requirements of the product, it is intended to the customers. The Overall description permits also to establish a context for the technical specifications, presented in the next section, the Specific Requirements, that describes in technical terms the details of the functionality of the product, and will serve as a guide for the developers.

2. OVERALL DESCRIPTION

2.1. PRODUCT PERSPECTIVE

The software will be totally self-contained. It will be dependent of an online books Library and a Database, stored in a dedicated server. The software can also be used independently, as every user will have a local books library, stored in his computer.

2.1.1. SYSTEM INTERFACES

There are no system interfaces to be noted for the moment in the project.

2.1.2. USER INTERFACES

ClassicLit will be GUI based. It will have a main GUI that contains three main subinterfaces (a detailed description of the functions provided by the interfaces can be found in section 2.2 – Product functions):

- An interface to manage the local library of books, and browse it
- An interface to read a book, and navigate into it.
- Only if the User is an administrator from the UTB-LMS: An interface to manage to online library of books

2.1.3. HARDWARE INTERFACES

ClassicLit project will be made with current hardware in the market in mind, and due to the java programming language continued maintenance and the structure of the application, more detailed hardware specifications are not required as is considered to be able to run on any hardware from the past 8 years.

2.1.4. SOFTWARE INTERFACES

The software will be developed using the Java programming language, thus it will be platform-independent. The only software interface it needs to run is the Java Virtual Machine.

2.1.5. COMMUNICATION INTERFACES

The books library and database are to be stored in a dedicated server. The only communication interface needed is Internet access, so that the software could access to the server.

2.1.6. *MEMORY CONSTRAINTS*

There are no memory constraints to be noted at the moment in the project.

2.2. PRODUCT FUNCTIONS

The main functions of the software are to provide an easy-to-use interface to manage a library of books, and read books. Thus it should implement these functions:

- Access the database in the dedicated server
- Retrieve the list of books stored in the server and all the information about them
- Display the list of books, and let the user navigate browse this list
- Change the order of the books displayed in the list, by title, author, or genre
- Show the information about a book when it is selected
- Search keywords inside the books in the online library, and display the books containing these keywords as a list
- Download a book from the server and store it in a local book Library
- Display the list of books stored locally, and let the user navigate into this list
- Delete a book stored locally
- Display the content of a book, page by page
- Let the user navigate easily inside a book using a table of content
- Search keywords inside a book and highlight the words if found in the book
- Highlight areas in a book
- Copy the highlighted areas in a text-file
- Set/reset bookmarks
- Search for words in a dictionary
- Change the display settings (font, color, background color, highlight color)

2.3. USER CHARACTERISTICS

The general characteristics of the users of this intended product are those that have finished their compulsory education levels, with a basic understanding and experience with computers or other technical devices.

2.4. CONSTRAINTS

There are no constraints to be noted for the moment.

2.5. ASSUMPTIONS AND DEPENDENCIES

We make the assumption that the memory available in the dedicated server in largely sufficient for all the books to store.

2.6. Apportioning of Requirements

An android version of the project will be under development once the Desktop based application is finished while an applet version is being considered.

3. Specific Requirements

3.1. External Interfaces

The only external interface is the Database. Here is a description of the interface:

a) Name of item:
ClassicLit Database

b) Description of purpose:
Contains all the information about the books stored in the server. Permits the software users to access and/or modify these information in real-time.

c) Source of input or destination of output:
The software will send SQL queries to the database to retrieve/modify/add information from/into the database, according to the users actions.

d) Timing:
The queries have to be processed instantly by the server.

e) Data formats:
The content of the database is text-only.

3.2. Functions

The system shall access the database and be able to send SQL queries, to receive or store information when needed.

The system shall explore the database tables in order to retrieve the list of books stored in the server and all the information about them. It shall store this information in the memory using appropriate data structures.

The system shall display the list of books stored in the server, organizing them by title, author, or genre (according to the user's choice). It shall be able to browse the list.

The system shall show the information about a book when the book is selected

The system shall be able to search keywords in the online library books using SQL queries, and filter the displayed list in order to show only the books containing the given keywords.

The system shall be able to download books from the dedicated server, using their direct URL, when the user decides it. The books shall be stored in the local computer, and added to the local books library.

The system shall be able to display the list of the books contained in the local library, organizing them by title, author, or genre (according to the user's choice), it shall be able to browse the list.

The system shall be able to delete a downloaded book from the local library

The system shall be able to display the content of a book, page by page. When the content is display, the system shall provide a table of contents to let the user navigate easily from one section to another in the book.

The system shall be able to search for keywords in a specific book, and highlight the keywords in the book if they are found.

The system shall let the user highlight areas in a book, and copy the highlighted areas to a text-file.

The system shall provide a settings menu, where the font, color, background, and highlight color can be changed.

3.3. PERFORMANCE REQUIREMENTS

Since this is not a real-time application, performance will not be a major issue.

The main functions described previously should not take more than 5 seconds.

The database must support simultaneous requests and must have sufficient bandwidth to quickly deliver books to the thousands likely to use the system.

3.4. LOGICAL DATABASE REQUIREMENTS

Not yet decided. We should first make an UML graph in order to understand the relationships between the data entries and decide what SQL tables to use, their contents and relationships.

3.5. DESIGN CONSTRAINTS

No constraints noted for the moment.

3.6. SOFTWARE SYSTEM ATTRIBUTES

3.6.1. RELIABILITY

The customer imposes no reliability attribute for the moment.

3.6.2. Availability

The software and the server should be available to use at any moment

3.6.3. Security

No security to be implemented for the moment

3.6.4. Maintainability

The administrators should be able to make a backup of the content of the server and database.

3.6.5. Portability

The software will be platform-independent.

3.7. Organizing the specific requirements

3.7.1. System mode

Student mode (filling the query and downloading the book(s))

Admin mode (must be able to do what student mode can't)

3.7.2. User class

There are to user classes, associated with the two system modes: Student and Admin.

3.7.3. Objects

Book, User, Library, Query, Interface, Table of Contents, Chapter.

3.7.4. Feature

Inputs by user for the majority of the functionalities of the program.

3.7.5. Response

Confirmation messages must be necessary upon each and every input from the customer and admin for quality use and maintainability