Software Engineering Project

Software Engineering Design I - Winter 2015 Project

Group #1

Software Requirements Document

Version 1.0.0 (Conditionally Approved), 19 January 2015

PRMS

PROFESSIONAL RESOURCE MANAGEMENT SYSTEM

Project Team:	Abad Hameed Rafael Nazario
Project Lead:	Brandon Crispino
Customer:	Dr. H. Naser

Document Status Sheet

Document Title	Software Requirements Document
Document ID	PRMS/Documents/Product/SRD/1.0.0
Author(s)	B. Crispino, A. Hameed, R. Nazario
Version	1.0.0
Document Status	draft / internally approved / [✓ conditionally approved] / approved

Version	**Date**	**Author(s)**	**Summary**
0.0.1	16-01-2015	A. Hameed	Document Creation
0.0.2	17-01-2015	B. Crispino	Added Use Case Diagram
0.0.3	18-01-2015	R. Nazario, A. Hameed, B.Crispino	Added text to chapters 1, 2 and 3
0.0.4	18-01-2015	A. Hameed	First version for internal review
0.0.5	19-01-2015	R. Nazario, A. Hameed, B. Crispino	Version for the second internal review
0.0.6	19-01-2015	R. Nazario, A. Hameed, B. Crispino	Internally accepted
1.0.0	19-01-2015	A. Hameed	Fixed document status

Approvals

Print Name	**Signature**	**Comments**

Contents

1 Introduction

1.1 Purpose

The purpose of this document is to specify the software requirements and a logical model of the PRMS system in a clear and consistent manner.

1.2 Scope

The software will be used to implement a client-server resource acquisition system. This system allows users to access a database of library resources from which they can acquire articles relevant to their needs. It hides the complexity of a resource acquisition system making the software easy to use. Usability is further increased by offering a front-end GUI for users to access the system.

1.3 Environment

The environment identified for the PRMS system is similar to a library system. The software will use keywords to search and acquire matching resources from the database with the capability to perform various sorting algorithms on the data. Being a client-server system, all requests will be sent to a server for acknowledgement.

1.3.1 System Requirements

- Windows 7 (or higher)
- Intel Core i3 (or higher)
- 4 GB RAM
- 256 MB hard disk space
- 2 MBit/s network

With PRMS being a client-server system, a dedicated server is required with the ability to handle multiple requests at a given time. A client machine is required to receive acknowledgement from the server and acquire the resources.

1.4 List of definitions and abbreviations

This section contains the definitions of all used terms, acronyms and abbreviations in this document.

1.4.1 Definitions

Algorithm	A process or set of rules to be followed in calculations or other problem-solving operations.
Client	Program that is used by all the users and system admins.
Data	Either platform dependent data or platform independent data.
ENGI-3050	The Software Engineering Design I Course ID at Lakehead University.
PRMS Software	A software implementing a client-server resource acquisition system.
Resource	Material which can be readily drawn from a server and displayed to the client for effective use.
Server	A computer or computer program that manages access to a centralized resource or service in a network (i.e. PRMS database)
System Administrator	The system admin oversees the entire PRMS system and has the right to configure the system, to create and remove admins and resources as well as any other high level configuration.
User	Any individual or system administrator with access to the client machine using PRMS.

1.4.2 Abbreviations

CASE Tools	Computer-aided Software Engineering Tools
GUI	Graphical user Interface
IDE	Integrated Development Environment
IP	Internet Protocol
LU	Lakehead University
PR-<#>	Priority Level, <#> can range from 1 (highest) to 5 (lowest)
PRMS	Professional Resource Management System
SR	Software Requirements
SRD	Software Requirements Document
SVN	Apache Subversion
SWE	Software Engineering

TCP	Transmission Control Protocol
TFS	Team Foundation Server
UML	Unified Modeling Language
XML	eXtensible Markup Language

1.5 Documents

1.5.1 Reference Documents

[SWE Design I Project Outline]	Naser, H. (2015). Team-project: client-server resource acquisition system [class handout]. Department of Engineering, Lakehead University, Thunder Bay, ON.

1.5.2 Applicable Documents

No applicable documentation in this phase of the project.

2 Specific Requirements

In this chapter all requirements and constraints of the developing product are given. The product will adhere to these requirements. Each of the requirements is preceded by a unique identifier for better traceability. Certain requirement are also given a priority level from 1 (highest) to 5 (lowest). The higher a requirements priority level, the more focus it will require.

2.1 Functional Requirements

2.1.1 Inputs/Outputs

(SR-2101, PR-1) The PRMS system will have a GUI requesting users to login to their accounts which will allow them to access resources. The user will be able to search for the resources using keywords. The output will be those resources which closely match or resemble the keywords entered.

2.1.2 Computations

(SR-2102, PR-1) The PRMS system will allow users to browse, sort and filter acquired resources by document type, resource subject, resource title, resource publish date and publisher.

2.1.3 Data Storage

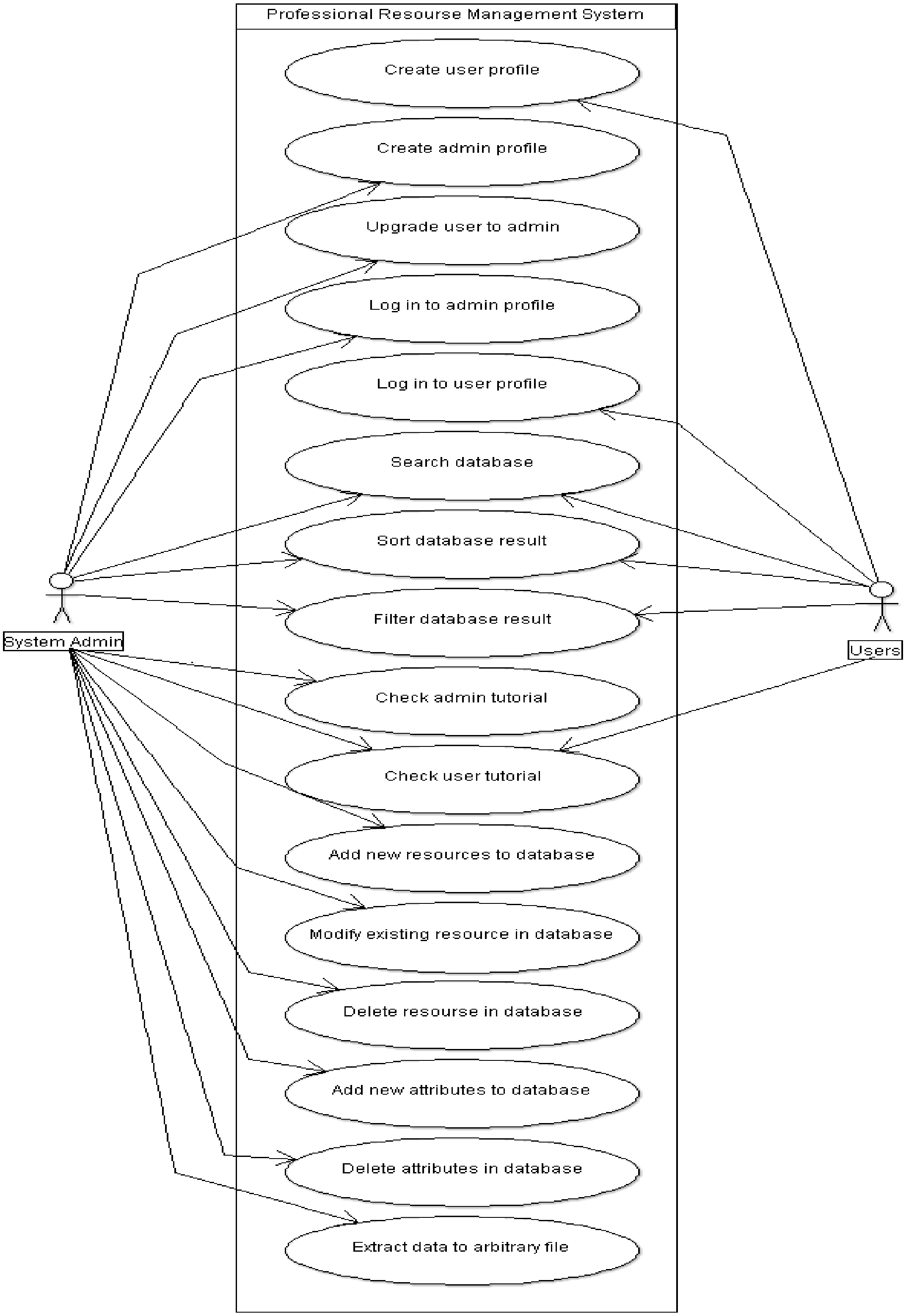
(SR-2103, PR-3) The PRMS system will require a server to keep between 150 and 200 resources and related information within it. A simple hard drive with 256 MB of space will be sufficient storage for this system.

2.1.4 Synchronization Issues

(SR-2104) Synchronization will be checked and clarified in upcoming documentation. There is no possibility to identify any issues in this phase of the project.

2.1.5 Essential Use Case Diagram

(SR-2105) The UML diagram (see page 8) gives an overview of the use cases which will be involved in implementing the complete PRMS system.


Professional Resourse Management System
Create user profile
Create admin profile
Upgrade user to admin
Log in to admin profile
Log in to user profile
Search database
Sort database result
Filter database result
Check admin tutorial
Check user tutorial
Add new resources to database
Modify existing resource in database
Delete resourse in database
Add new attributes to database
Delete attributes in database
Extract data to arbitrary file
System Admin
Users

2.2 Non-functional Requirements

2.2.1 Response Time

(SR-2201) In the list of requirements provided there is no specification with regards to the response time. With PRMS being a client-server system, network dependency is a factor and can lengthen resource acquisition time.

2.2.2 Storage Requirements

(SR-2202) Customer requirements had no indication with regards to storage requirements. With 150-200 resources required by the customer, an approximation of 250MB of storage has been defined.

2.2.3 Reliability/Availability

(SR-2203) The PRMS system will be made robust and capable of handling multiple server requests from multiple users simultaneously. As mentioned in *2.2.1*, as PRMS is a client-server, it has a dependency on network usage and may incur lag time as a result. In any other scenario, the PRMS system will be made to sustain heavy use ensuring its reliability and availability.

2.2.4 Privacy/Security/Safety

(SR-2204) System administrators will have full admin rights to the PRMS software and will solely be in-charge of back-end configurations. General users will have front-end access to the system database, simply to acquire resources for their academic purposes. Each user including the system administrators will be required to login with an authenticated username and password. The login information including user passwords will be marked as hidden for privacy and security purposes. A system timeout can be put in place to logoff a user due to account inactivity.

3 Development Overview

3.1 CASE Tools

This section will guide you through a variety of CASE tools which may be used to help implement this project from a development perspective.

3.1.1 Front-end

A GUI will be implemented with an easy-to-use interface allowing general users and system administrators to maneuver through and use the PRMS system appropriately and efficiently. Back-end computations and complex algorithms will not be shown on the front-end of the system. A guided tutorial will be implemented for first-time users.

3.1.2 Back-end

The connection between client-server will be implemented using socket programming. Various searching and sorting algorithms will be put in place to allow for an easier and more efficient user experience.

3.1.3 Development Platform

The major development of the PRMS system will be implemented using C++. The database will be a list of resources constructed within an XML document which will then be read into the C++ program to be used.

3.1.4 Version/Configuration Control

The project will be document-oriented and every change will be tracked by group members. Revision control tools such as SVN, TFS or GIT will be used to ensure program traceability and group/task effectiveness.