Software Requirements Specification

for

Furniture Rental Store System

Version 1.0 under review

Prepared by:

Suryam Arnav Kalra(19CS30050) - suryamkalra35@gmail.com

Kunal singh(19CS30025) - ksingh19136@gmail.com

Abhinandan De(19CS10069) - abhinandan0316@gmail.com

IIT Kharagpur

5 March 2021

Table of Contents

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

A forum where people can search and rent furniture. FRSS maintains all the information related to rental operations when it comes to furniture. It stores the data for the furniture like the type, relevant images , rental price , cost price , age , etc. and it is especially useful for the people who are looking to refurbish their house after relocation and don’t want to buy costly furniture.

1.2 Document Conventions

This SRS Document has been written using Free and Open Source writing tools such as LibreOffice typed in a monospace font **Source Code Pro**. The font size used is 11 for text and 14 for headings. All headings are highlighted appropriately in bold. The document is prepared using **US English** convention.

1.3 Intended Audience and Reading Suggestions

This document lists all technical and non-technical aspects of the software. It is intended to assist developers and other end users to understand the motivation behind the software and understand implementation intricacies in it. Anybody who wants to use the software can read the appropriate parts of the document, a list of which is given in the Table of Contents on Page 2.

1.4 Product Scope

FRSS is a desktop based application that is intended to be used by individuals who want to rent furniture and don’t want to pay heavy prices. The application would be based on the client-server model where the users can interact with the GUI that is served by a backend server that stores the data of the furniture and the users.

1.5 References

This document is based on the IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specification given by the IEEE Computer Society in 1998.

2. Overall Description

2.1 Product Perspective

It is a new self-contained product which originated from the idea of people who pay heavy prices for furniture when they relocate to a new city and only want the furniture for a short period of time. This project tackles this problem by allowing them to rent furniture for low prices and as long as they want.

2.2 Product Functions

- Give furniture on rent at low prices.
- Give furniture on loan based at low interest rates.

2.3 User Classes and Characteristics

- Admin
    - Owner of the furniture present in the inventory.
    - When a new product is introduced, the admin can enter that in a catalogue.
    - Admin sees the total profit and investment on his dashboard
    - If the number of any item present in the inventory falls below some threshold then a message is sent to the admin for its need.
    - The admin can enter or change the price of any item anytime
- Customer
    - Customers may take furniture on loan at some interest rate based on whether he has taken any furniture in the past.
    - Customers are able to see their past order history as well as amount due.
    - Customers can give review/feedback about the product.
    - The customer can take any furniture present in the catalogue as well as the inventory.

2.4 Operating Environment

The application is based on a client-server model with the users interacting as clients with a GUI web application. GUI is made in python using tkinter / pyqt5 and the database is handled using SQL. Since python and SQL are cross-platform applications so the app would be available on windows/mac/linux.

2.5 Design and Implementation Constraints

2.6 User Documentation

The software will be accompanied with a user manual aimed at allowing end users to study all components of the application and use it as intended. For developers and other technical users, all API's, classes and methods will be documented and presented online.

2.7 Assumptions and Dependencies

Python 3.8 or above , Tkinter , PIL , MySQL , Matplotlib.

3. External Interface Requirements

3.1 User Interfaces

The product will consist of a desktop application with which the user will interact. The customer and admin will have to login before they can see their respective dashboards. The customer can rent / loan furniture, give review / feedback about the items, and return the items. The admin on the other hand can create/delete customer accounts, add new items to the inventory , review the returned furniture , check the total profits and investments made by the company.

3.2 Hardware Interfaces

The backend server will require a decent processing unit to enable multiple threads for worker processes. There are no specific hardware requirements for the user under the assumption that he/she has access to a decent internet connectivity via a decent desktop/laptop.

3.3 Software Interfaces

The application will interact with the Database (SQL server) using a CRM. The database will be secured with a username-password combination. To improve quality of software, there will be three databases - customer information, admin information, furniture information.

3.4 Communications Interfaces

4. System Features

4.1 User Authentication - Sign Up & Login

4.1.1 Description and Priority

As the first step to using **FRSS**, a user has to create an account by signing up on the application. This task is essential as it provides the user class needed for all other activities.

4.1.2 Stimulus/Response Sequences

User has to fill a form consisting of basic details such as Name, E-Mail, Username and Password to get started.

As soon as a new user signs up, the system will send a

verification mail to ensure that no malicious users enter the system.

4.1.3 Functional Requirements

Signup: As soon as the user enters information and the authenticity of the user is confirmed via email, the new user is created in the database. Then the user can perform the desired actions.

Login: Existing users can login with their authentication information once the system deems it to be correct.

4.2 Addition of new products to inventory

4.2.1 Description and Priority

When a new product is introduced, the admin can enter it into the inventory.

4.2.2 Stimulus/Response Sequences

If a new furniture is released in the market, the admin can add it into the inventory.

4.2.3 Functional Requirements

Insert: The admin makes an insert query into the furniture database.

4.3 Price flexibility

4.3.1 Description and Priority

The admin can enter / change the price of any item anytime.

4.3.2 Stimulus/Response Sequences

If there is a price fluctuation in the market the admin can edit the price of the furniture.

4.3.3 Functional Requirements

Edit price: The admin makes a search query in the furniture database whose price has to be changed and edits / updates the price of the item.

4.4 Searching and Renting Furniture

4.4.1 Description and Priority

The customer can take any furniture present in the catalog as well as inventory.

4.4.2 Stimulus/Response Sequences

Customer goes to the furniture section,searches for any item and rents it if it is available in the inventory.

4.4.3 Functional Requirements

Renting Furniture: customer requests to rent furniture , a query is made to the database to check the availability of the item and then the customer can rent it and then that furniture is added to the customer database for that user.

4.5 Notification on less amount of item

4.5.1 Description and Priority

When the count of any item falls below a threshold value, the admin is notified about the same.

4.5.2 Stimulus/Response Sequences

A notification is sent to all the admins regarding the issue and specific steps may be taken accordingly. Eg : complete deletion of the item from the database or addition of new objects of that kind.

4.5.3 Functional Requirements

Sending Information: The system sends a notification to all admins regarding the issue.

4.6 Loan on furniture

4.6.1 Description and Priority

The customer can take a furniture on loan at some interest rate , based on whether he has taken any furniture in the past.

4.6.2 Stimulus/Response Sequences

The customer can take furniture on loan only if he has taken some furniture in the past.

4.6.3 Functional Requirements

Loan on items: The customer can choose the option to take furniture on loan.

Calculation of interest rate:The interest rate is low if more furniture is taken in the past.

4.7 Viewability of past order history and amount due

4.7.1 Description and Priority

The customer can see the past orders history and the amount due.

4.7.2 Stimulus/Response Sequences

The customer can go to his profile and can see his past order and history and can also see the amount due.

4.7.3 Functional Requirements

Profile History: Once the customer visits his profile a query is made to the customer database and then data is taken about his/her past orders.

Amount Due: Once the customer visits his profile a query is made to the customer database and then data is taken about his/her due amount.

4.8 Verification/readdition of products on return

4.8.1 Description and Priority

The admin is able to verify the condition of products on return and take necessary steps.

4.8.2 Stimulus/Response Sequences

The admin verifies the condition of the product on return and depending on the condition, he/she may demand a full payment from the customer if required. If not, the product may be added back to the inventory with a reduction of the rental cost by 10%.

4.8.3 Functional Requirements

Demand payment : The admin may demand full payment of the product if it is not in a good condition and it is deleted from the database.

Re-Addition to inventory : The admin may add the product back to the inventory if it is found alright.

Decreasing price : The admin decreases the price of the product by 10% after it is used by a customer for each year.

4.9 Review/Feedback

4.9.1 Description and Priority

The customer can give review or feedback about the item that he has rented.

4.9.2 Stimulus/Response Sequences

After the customer has used the item he can give feedback about the service and the product used.

4.9.3 Functional Requirements

Feedback of items:The customer can choose the option to give feedback about the item.

4.10 Profit and Investments

4.10.1 Description and Priority

The admin can view the total profit and investments made by the company in the form of a graph.

4.10.2 Stimulus/Response Sequences

After a certain item has been rented it’s rental price is added to the total profits and on buying / adding an item to the inventory it’s cost is added to the investments.

4.10.3 Functional Requirements

Calculation of profit:The total profit is calculated based on the total rent that has been paid on the furniture.

Calculation of investment:The investment is calculated by the total amount of money spent by the company on buying the items.

Graphing of results: A graph representing the profit and investment of the company is shown to the admin.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The user should be able to query databases quickly and the results fetched must be appropriate. This can be done by finding the right balance between performance and accuracy by using MySQL. The DB’s that store user profile info are going to be optimized using in-memory caching tools.

5.2 Safety Requirements

The application operates above multiple abstraction layers over the hardware and kernel and there is little possibility of damage to a user’s device.

5.3 Software Quality Attributes

Maintainability

Different versions of the product should be easy to maintain. For development it should be easy to add code to an existing system, should be easy to upgrade for new features and new technologies from time to time. Maintenance should be cost effective and easy.

Usability

This can be measured in terms of ease of use. Application should be user friendly. Should be easy to learn. Navigation should be simple.

Flexibility
Should be flexible enough to modify. Adaptable to other products with which it needs interaction. Should be easy to interface with other standard 3rd party components.

5.4 Business Rules

The software will be free to use for all users and the source code will be publicly hosted for free use and modification.

6. Other Requirements

Appendix A: Glossary

Appendix B: Analysis Models

Appendix C: To Be Determined List