SOFTWARE REQUIREMENTS SPECIFICATION

**For**

**Music Library Management**

**Prepared by:-**

*US*

# Introduction

## Purpose

A task manager system for normal users serves to organize and prioritize tasks efficiently,

aiding in time management and progress tracking, employs reminders for timely task completion.

User can set and achieve tasks, utilize the system for documentation, and appreciate its flexibility in

adapting to varied work style. Overall, the system proves instrumental in enhancing user productivity,

and worklife balance.

## Document Conventions

* + - Entire document should be justified.
    - Convention for Main title

Font face: Times New Roman Font style: Bold

Font Size: 14

* + - Convention for Sub title

Font face: Times New Roman Font style: Bold

Font Size: 12

* + - Convention for body

Font face: Times New Roman Font Size: 12

## Scope of Development Project

## The Task Management System provides an interface for users to track their daily tasks. The user interface will allow for entering of new tasks, editing tasks, deleting tasks, and searching of tasks. The user can categorize their tasks into different categories. The user interface will also allow for creating and deleting of users. Lastly, this interface will have the permission to reset passwords. The interface also allows them to search tasks and mark tasks that they have completed. The system will provide a history of previous tasks assigned or completed. The system will also prioritize the tasks automatically and notify the users according to the deadline specified by users.

## 1.4 Definitions, Acronyms and Abbreviations

JAVA -> platform independence SQL-> Structured query Language ER-> Entity Relationship

UML -> Unified Modeling Language

IDE-> Integrated Development Environment SRS-> Software Requirement Specification

## References

* + - Books

 Software Requirements and Specifications: A Lexicon of Practice, Principles and Prejudices (ACM Press) by Michael Jackson

Software Requirements (Microsoft) Second EditionBy Karl E. Wiegers

Software Engineering: A Practitioner’s Approach Fifth Edition By Roger S. Pressman

* + - Websites

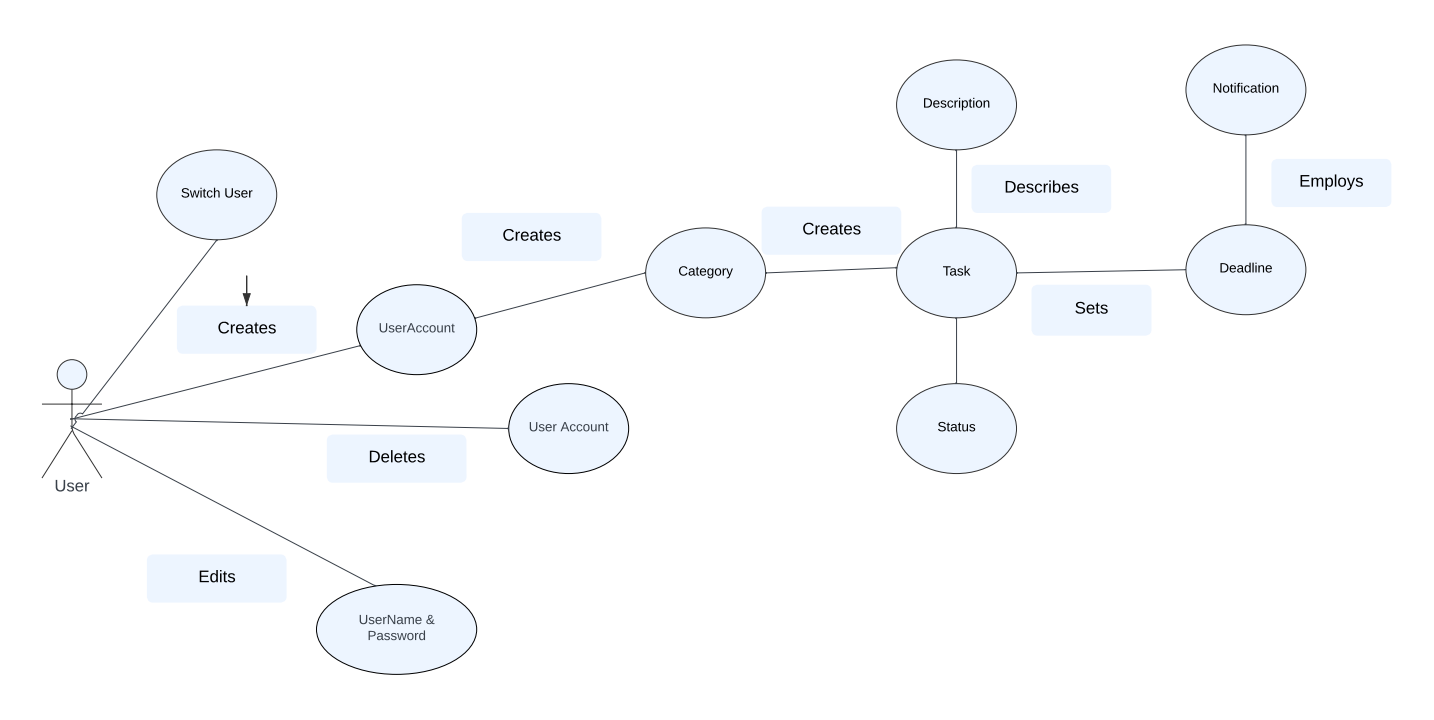
[**http://www.slideshare.net/**](http://www.slideshare.net/)

[**http://ebookily.net/doc/srs-library-management-system**](http://ebookily.net/doc/srs-library-management-system)

# Overall Descriptions

## Product Perspective

Use Case Diagram of Task Management System

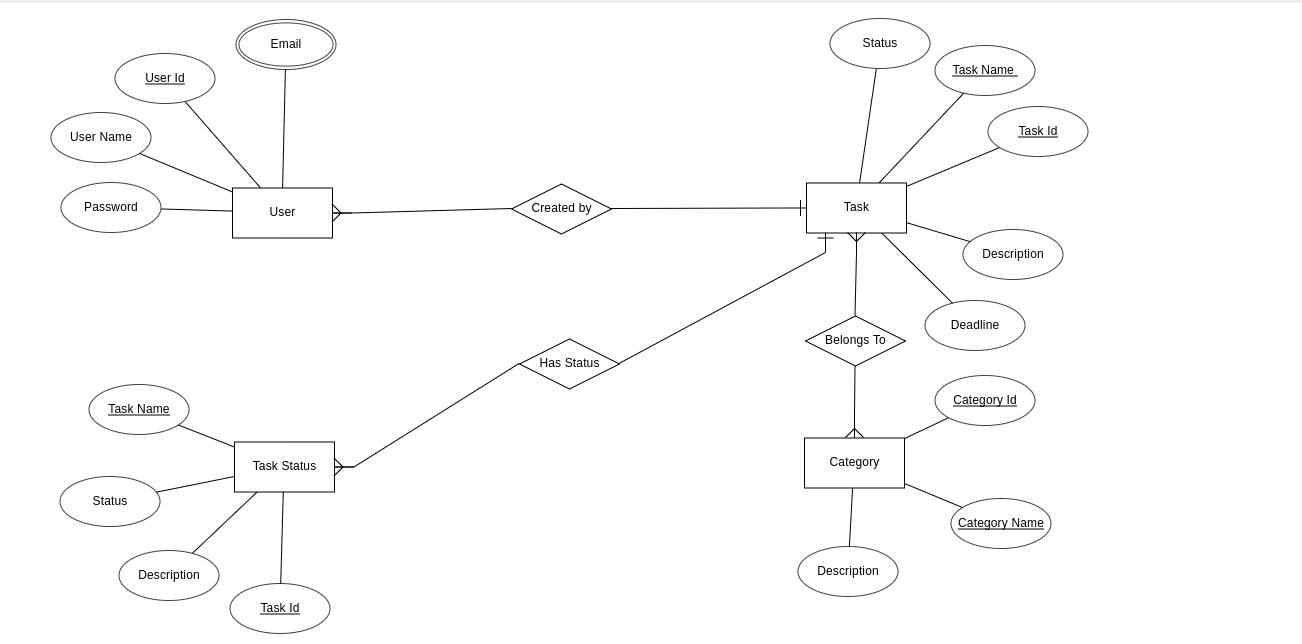


The Task Management System will be a new self-contained product. Communication will be done

## via a Java interface to a SQL database that stores the Task Management information.

## Product Function

Entity Relationship Diagram of Music Library Management System.



A task management system serves as a centralized platform designed to facilitate efficient

organization, prioritization, and coordination of tasks. Its fundamental functions encompass the seamless

creation and assignment of tasks, enabling users to input essential details such as due dates, task

description. The system excels in the organization and categorization of tasks, allowing users to structure

their work through projects or categories. By providing these essential functions, a task management

system becomes an indispensable tool for users seeking to optimize their workflow and productivity.

## User Classes and Characteristics

**USER:**

1. Create Task:

-Creation of tasks by the user.

2. Update Task:

- Ability to update the existing task.

3. Delete Task:

- Ability to delete the task created by the user.

4. Add User:

- Ability to add a new user.

5.Switch User:

- Capability to switch from one user to another.

6. Edit User Credentials:

- Access to change the username and password.

7. Remove User:

- Access to remove a user.

8. Add and Remove Category

- Ability to add and remove categories.

These features provide a structured framework for the Task Management System, ensuring that the user has necessary features to manage and access the tasks effectively.

## Operating Environment

The product will be operating in windows environment. The Task Management System is a desktop application and shall operate in desktops.

The hardware configuration includes Hard Disk: 40 GB, Monitor: 15” Color monitor, Keyboard: 122 keys. The basic input devices required are keyboard, mouse and output devices are monitor, etc.

## Assumptions and Dependencies

**Task Management System Assumptions:**

* + 1. Flawless code: No bugs or crashes to disrupt your workflow.
    2. Easy navigation: Easy to navigate between different users and tasks.
    3. Prioritizing: Your tasks will be automatically prioritized according to the deadline.
    4. Categorize: Categorizing your tasks to access tasks at ease.
    5. Seamless GUI: Provides a smooth and integrated user experience.
    6. Secure login: User login credentials to ensure security.

The dependencies are:-

* + - JavaFX, MySQL.
    - Roles, features, interface.
    - User info, task details, updates - all stored accurately and securely in the database.
    - Scalability, performance - the JavaFX app adapts to your growing audience.

## Requirement

Software Configuration: -

This software package is developed using javafx as front end which is supported with SceneBuilder. My SQL as the back end to store the database.

Operating System: Windows 11

Language: Java, Java fxml

Database: My SQL Server (back end)

Hardware Configuration: -

OS: Windows 7 or above

Hard Disk: 100GB

RAM: 4gb or more

## Data Requirement

The inputs consist of the query to the database and the output consists of the solutions for the query. The output also includes the user receiving the details of their accounts. In this project the inputs will be the queries as fired by the users like create an user account, creating tasks, creating categories, giving task description and setting deadline. Now the output like completion status of tasks, and notifications will be visible in the viewer side.

# External Interface Requirement

## GUI

The software provides good graphical interface for the user to operate on the task management system, performing tasks such as creating users, adding tasks, categorizing tasks and setting deadlines.

* + 1. It allows user to create tasks.
    2. The users can update the existing tasks.
    3. It includes categories where the user can create their own categories.

3.2.4 All the features like descriptions, sending notifications to the users and

prioritizing are included.

* + 1. The design should be simple and all the different interfaces should follow the

template.

Login Interface: -

In case the user is not yet registered, they can enter the details and register to create their account. Once the account is created, they can ‘Login’ which asks the user to type his username and password. If the user entered either his username or password incorrectly then an error message appears.

Task Listing: -

A view of all ongoing tasks.

Categorizing: -

Users can create categories and categorize their tasks.

Notification: -

The users will be notified of their tasks according to the deadline.

# System Features

# Secure Task Management System with JavaFX:

# Login: Username/password with password encryption.

# User isolation: Each account is a different entity.

# Notification: Notifying the user’s tasks.

# Other Non-functional Requirements

## Performance Requirement

The proposed system that we are going to develop will be used by employees, students and just about anyone. Therefore, it is expected that the database would perform functionally all the requirements that are specified automatically.

* + 1. The performance of the system should be fast and accurate
    2. Task Management System shall handle expected and non-expected errors in ways that prevent loss in information and long downtime period. Thus it should have inbuilt error testing to identify invalid username/password
    3. The system should be able to handle large amount of data. Thus it should accommodate high number of tasks and users without any fault.

## Safety Requirement

The database may get crashed at any certain time due to virus or operating system failure. Therefore, it is required to take the database backup so that the database is not lost.

## Security Requirement

* + 1. System will use secured database

5.3.2 Users can read information and edit or modify their personal and some other information.

* + 1. System will have different types of users and every user has access constraints.
    2. Proper user authentication should be provided

## Requirement attributes

* + 1. There may be multiple users creating the tasks, all of them will have the right to create changes to the system.
    2. The project should be open source
    3. The Quality of the database is maintained in such a way so that it can be very user friendly to all the users of the database

## Business Rules

A business rule is anything that captures and implements business policies and practices. A rule can enforce business policy, make a decision, or infer new data from existing data.This includes the rules and regulations that the System users should abide by. This includes the cost of the project and the discount offers provided. The users should avoid illegal rules and protocols. No user should cross the rules and regulations.

## User Requirement

The users are assumed to have basic knowledge of the computers and the general idea of task management system. The proper user interface, user manual must be sufficient to educate the users on how to use the system without any problems.

The users have certain facilities such as:-

* + 1. Forgot Password
    2. Maintaining task history.

# Other Requirements

## Data and Category Requirement

There are different categories of users namely teaching staff, Librarian, Admin, students etc. Depending upon the category of user the access rights are decided.It means if the user is an administrator then he can be able to modify the data,delete, append etc. All other users except the Librarian only have the rights to retrieve the information about database. Similarly there will be different categories of books available. According to the categories of books their relevant data should be displayed. The categories and the data related to each category should be coded in the particular format.

## Appendix

A: Admin, Abbreviation, Acronym, Assumptions; B: Books, Business rules; C: Class, Client, Conventions; D: Data requirement, Dependencies; G: GUI; K: Key; L: Library, Librarian; M: Member; N: Non-functional Requirement; O: Operating environment; P: Performance,Perspective,Purpose; R: Requirement, Requirement attributes; S: Safety, Scope, Security, System features; U: User, User class and characteristics, User requirement;

## Glossary

The following are the list of conventions and acronyms used in this document and the project as well:

* + 1. Administrator: A login id representing a user with user administration privileges to the software
    2. User: A general login id assigned to most users
    3. Client: Intended users for the software
    4. SQL: Structured Query Language; used to retrieve information from a database
    5. SQL Server: A server used to store data in an organized format
    6. Layer: Represents a section of the project
    7. User Interface Layer: The section of the assignment referring to what the user interacts with directly
    8. Application Logic Layer: The section of the assignment referring to the Web Server. This is where all computations are completed
    9. Data Storage Layer: The section of the assignment referring to where all data is recorded
    10. Use Case: A broad level diagram of the project showing a basic overview
    11. Class diagram: It is a type of static structure diagram that describes the structure of a system by showing the system’s cases, their attributes, and the relationships between the classes
    12. Interface: Something used to communicate across different mediums
    13. Unique Key: Used to differentiate entries in a database

## Class Diagram

A class is an abstract, user-defined description of a type of data. It identifies the attributes of the data and the operations that can be performed on instances (i.e. objects) of the data. A class of data has a name, a set of attributes that describes its characteristics, and a set of operations that can be performed on the objects of that class. The classes’ structure and their relationships to each other frozen in time represent the static model. In this project there are certain main classes

which are related to other classes required for their working. There are different kinds of relationships between the classes as shown in the diagram like normal association, aggregation, and generalization. The relationships are depicted using a role name and multiplicities. Here ‘Task’, ‘User’, ‘Category’ and ‘Task Status’ are the most important classes.

