Low-Level Design

Book Recommendation System

Written By: Jaideep Jaiswal **Document Version:** 1.0 **Last Revised Date:** 31-3-2024

Document Control

Change Record:

Version	Date	Author	Comments
1.0	31-3-2024	Jaideep Jaiswal	Initial version

Reviews:

Version	Date	Reviewer	Comments
1.0	[Date]	[Reviewer's Name]	N/A

Approval Status:

Version	Review	Date	Reviewed By	Approved By	Comments
1.0	Passed	[Date]	[Reviewer]	[Approver]	N/A

Contents

1. Introduction

The Low-Level Design (LLD) document outlines the internal logical design of the Book Recommendation System. It provides a detailed description of the system's architecture, modules, and interactions between components.

2. What is Low-Level Design Document?

The LLD document describes the class diagrams, methods, and relations between classes and program specifications. It serves as a reference for developers to code the system's functionalities based on the document's specifications.

3. Scope

The LLD covers the component-level design process, including data structures, software architecture, source code, and algorithms. It defines the data organization and refinement process during system design.

4. Architecture

The architecture of the Book Recommendation System consists of the following components:

- Data Collection Module
- Data Preprocessing Module
- Data Insertion into Database
- Export Data from Database
- Data Preprocessing
- Data Clustering
- Model Building
- User Interaction
- Deployment

5. Architecture Description

The Book Recommendation System architecture is designed to collect book data, preprocess it, cluster similar books, build recommendation models, interact with users, and deploy the system for production use.

6. Data Description

The system collects book data from various sources, preprocesses it, and stores it in a database for further analysis and recommendation generation.

7. Data Transformation

The collected book data is transformed into a suitable format for analysis and model building. This involves cleaning, normalization, and feature extraction.

8. Data Insertion into Database

Preprocessed book data is inserted into a database for storage and retrieval. The database schema includes tables for books, authors, genres, ratings, and other attributes.

9. Export Data from Database

Data from the database can be exported as CSV files for further analysis or model training.

10. Data Preprocessing

Data preprocessing steps include handling missing values, text preprocessing, and feature engineering to prepare the data for clustering and model building.

11. Data Clustering

The preprocessed book data is clustered using algorithms such as K-Means to group similar books together. This facilitates personalized recommendations by identifying patterns among books.

12. Model Building

Recommendation models are built using machine learning algorithms such as collaborative filtering or content-based filtering. These models generate personalized recommendations for users based on their preferences.

13. User Interaction

The system allows users to interact by providing input such as preferences, ratings, or search queries. It includes features for user authentication, profile management, and feedback submission.

14. Deployment

The Book Recommendation System is deployed to a production environment for users to access. It is configured for scalability, reliability, and high availability.

15. Unit Test Cases

Unit test cases are written to verify the functionality of each module and ensure the system behaves as expected under different scenarios.