Data Model & Analytics Model Document

for

BookDiary project

Prepared by

Yuliana Yurchenko

2023

Contents

1 Layered architecture diagram

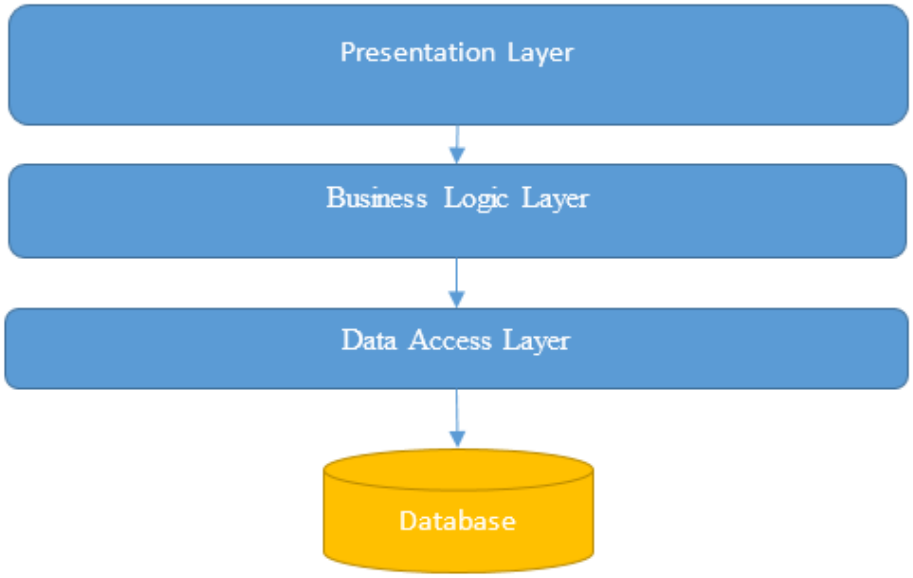

Figure 1: The three layered architecture

An application consists of layers, which are abstract horizontal parts. These layers represent the different levels and types of abstraction of the software. The layers help to slice the application into more manageable units and support multiple implementations. Usually, when the user is making a web request, the data flow will pass all of those layers. Each layer has a separated set of models that shouldn't be reached from another layer. In addition, each layer only is able to reach the layer beneath it or at the same level.

Presentation Layer (PL)

The presentation layer is the highest layer of the software. It is where the user interface of the software is placed. It contains logic only related to how the data is presented and the beginning of each CRUD (Create, Read, Update or Delete) operation flow.

Business Logic Layer (BLL)

As the name of this layer, most of the logic of the application is placed in this layer. The BLL should not have intimate knowledge of any data model so it can't possibly fire a query against the database. The layer input is a data structure received from the presentation layer and the layer output is a data structure received from the DAL. The BL layer allows to design applications that support multiple user interfaces, this minimizes the chances of needless code duplication.

Data Access Layer (DAL)

This layer is separated into two: DAL abstractions and DAL. The data access layer is the layer that implements the communication with the data source (SQLite database in our case). A

data access abstractions layer follows the idea of “separation of concerns” whereby all of the logic required for the business logic to interact with the data layer is isolated to a single set of classes (layer). This allows to easily change the backend physical data-storage technology (from SQLite to MySQL, for example) without having a large impact on the business logic.

2 Entity relationship diagram

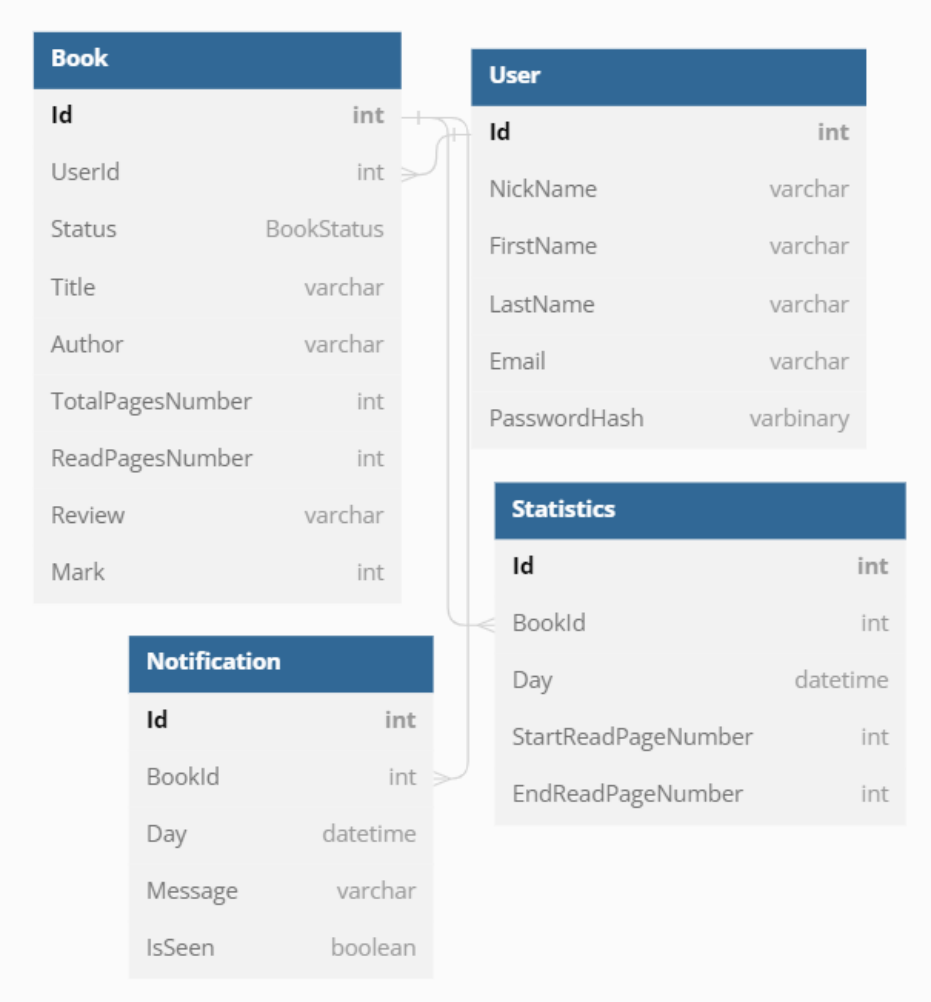

Figure 2: Entity relationship diagram

The project has 4 entities: User, Book, Notification and Statistics. The relationship between User and Book is One-to-Many; the relationship between Book and Statistics is One-to-Many; the relationship between Book and Notification is One-to-Many.

The variable Status of Book entity has type of enum BookStatus{InProgress, Completed, Planned}.

These entities are wrapped with classes on the DAL.

3 Analytics model

Analytics model is represented on BLL by 5 Services: UserService, BookService, HashService, NotificationService, StatisticService.

UserService Gets User from database or Creates/Updates/Deletes User on database. It also gives an opportunity for User to Sign Up or Log In.

HashService computes hash for User's password. It uses System.Security.Cryptography package.

BookService Gets Book from database or Creates/Updates/Deletes Book on database.

NotificationService Gets Notification from database or Creates/Updates/Deletes Notification on database.

StatisticService Gets Book from database or Creates/Updates/Deletes Book on database. It also is able to group and sum statistics by a set of Books or by chosen period of time or both.