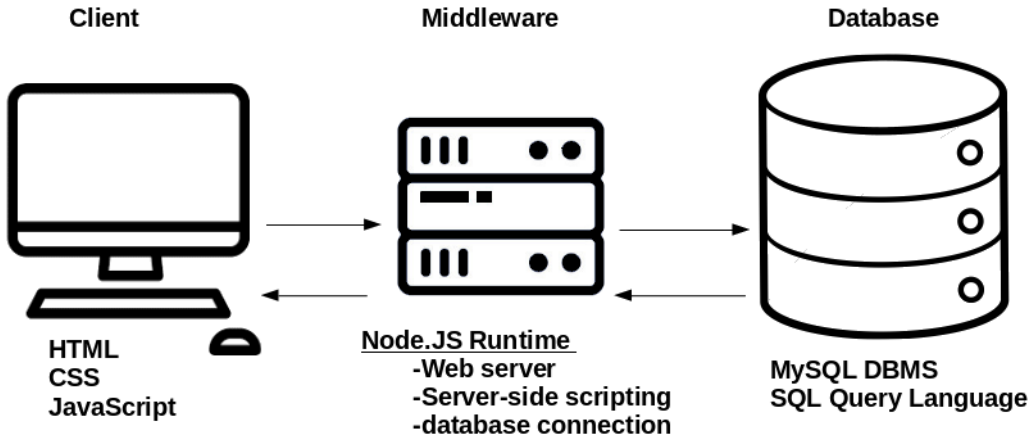
CS157A Project Proposal

Dispoto, Brett Kamel, Adham Cai, Feiyu

September 6, 2019

1 Project Overview

2 System Enviroment



2.1 Client

The presentation layer will be provided via the following languages/software:

- HTML
- CSS/Bootstrap
- JavaScript
- Web Browser

2.2 Middleware

All middleware will be provided via the Node.js Runtime Environment. This includes:

- A server side scripting language (Node.js)
- Database connection via Node.js MySQL database driver
- Node.js build-in web server (HTTP module)

2.3 Relational Database

The relational database management system (RDBMS) in use will be MySQL, which will be queried via the Structured Query Language (SQL). The following data are examples of what will be stored:

- User information (name, DOB, username, password, site activity, favorites, etc)
- Book information (author, ISBN, publisher, date, reviews, etc)

2.4 Hardware

Although this project will implement a three-tiered database application architecture, all three abstractions will be running on a single local host; this means that the architecture is only virtually three-tiered.

3 Functional Requirements

The system will provide searching, sorting the shown book lists, reading books, reading reviews, and sharing via email functions from our free textbook database. Anonymous visitors can use these functions without signing up for an account. However, signing up for an account with a valid email address will provide more functions. A user can sign in his/her account with his/her email address and password. A sign in user will have additional functions such as writing reviews, setting favorite books and having access to their account details page.

Supported features include:

- **Reading:** reading the books itself or its reviews. Input: book information. Output: book content
- **Searching:** searching for book from the provided information. Input: title, author, year of publication, publisher, ISBN, genre, etc. Output: list of books fitting search query.
- **Sorting:** sort the way books are listed. Input: sorting rules such as reviews, alphabetical order, etc.
- **Review:** Give a review and write comments for a book. Input: review and comments. Output: Review and comments are posted on the database that everyone can see
- **Set favorites:** set a book as favorites, the user can see it in his/her profile. Input: "set favorites" button in book details. Output: books stores in the user's profile.
- **Sharing via email:** sharing book details via email. Input: sharing button in book details. Output: generate share URL link that ready to send via email.

4 Non-functional Issues