**INDY-1 Blue — Book Dating App**

**Software Requirements Specification**

**CS 4850 - Section 02 – Fall 2024**

**Professor Perry – Aug 27, 2024**

|  |  |
| --- | --- |
| A person with long hair smiling  Description automatically generated  Chelsea Mandel  Documentation | A person wearing sunglasses and a life jacket  Description automatically generated  Sebastian Arechaga  Developer |
| A person taking a selfie  Description automatically generated  Obie Ezeobi  Developer | A person smiling at the camera  Description automatically generated  Keaton Moneymaker  Developer |

**Team Members:**

|  |  |  |
| --- | --- | --- |
| Name | Role | Cell Phone / Alt Email |
| Obie Ezeobi | Developer | [678.655.0628](tel:6786650628)  [oezeobi@students.kennesaw.edu](mailto:oezeobi@students.kennesaw.edu) |
| Chelsea Mandel | Documentation | [423.285.4514](tel:4232854514)  [cmandel1@students.kennesaw.edu](mailto:cmandel1@students.kennesaw.edu) |
| Sebastian Arechaga | Developer | [678.213.6014](tel:6782136014)  [sarechag@students.kennesaw.edu](mailto:sarechag@students.kennesaw.edu) |
| Keaton Moneymaker | Developer | [706.260.5079](tel:7062605079)  [kmoneym1@students.kennesaw.edu](mailto:kmoneym1@students.kennesaw.edu) |
| Sharon Perry | Project Owner or Advisor | [770.329.3895](tel:7703293895)  [Sperry46@kennesaw.edu](mailto:Sperry46@kennesaw.edu) |

# Table of Contents

[Table of Contents 2](#_Toc638707976)

[1.0 Introduction 2](#_Toc1650817616)

[1.1 Overview 2](#_Toc1338641598)

[1.2 Project Goals 2](#_Toc856973175)

[1.3 Definitions and Acronyms 3](#_Toc1333225155)

[1.4 Assumptions 3](#_Toc1149123995)

[2.0 Design Constraints 3](#_Toc1736377503)

[2.1 Environment 3](#_Toc965440038)

[2.2 User Characteristics 3](#_Toc472096382)

[2.3 System Constraints 4](#_Toc137117918)

[3.0 Functional Requirements 4](#_Toc1058018011)

[3.1 Login 4](#_Toc1408956659)

[3.1.1 Account Creation 4](#_Toc246330307)

[3.1.2 Login with Username and Password 4](#_Toc1318091858)

[3.2 Display Main Page 4](#_Toc1137303918)

[3.2.1 Book Suggestions 4](#_Toc554511983)

[3.2.2 Accept/Reject Book 4](#_Toc661827782)

[3.2.3 User Preference Filtering 5](#_Toc1867593637)

[3.3 Display Profile Page 5](#_Toc967918193)

[3.3.1 Profile Details 5](#_Toc1736869788)

[3.3.2 Saved Books 5](#_Toc1111557491)

[4.0 Non-Functional Requirements 5](#_Toc1223574256)

[4.1 Security 5](#_Toc230964103)

[4.2 Capacity 5](#_Toc2228953)

[4.3 Usability 6](#_Toc2075758602)

[4.4 Other 6](#_Toc305387935)

# 1.0 Introduction

## 1.1 Overview

This document defines the functional and non-functional requirements for the Book Dating Application software product. This document provides a detailed description of the system’s requirements so that they can be easily understood by stakeholders and developers.

This document is not intended to address development-related issues such as scheduling, cost analysis, or development methods. Such issues will be addressed in a separate design document.

The Book Dating App is a web-based application which aims to be a service for readers to find books and discover what matches their reading preferences.

## 1.2 Project Goals

The goal of the Book Dating App project is to develop a website utilizing an extensive collection of books, categorized and labelled by relevant distinguishers such as genre, for users to peruse and select from. Users will create profiles that will house saved collections of the books chosen while browsing. Suggested books will account for user preferences. Advertising will be added as a method of monetization.

## 1.3 Definitions and Acronyms

**Application or Product** – the software system specified in this document

**DBMS –** Database Management System

**Candidate**- A book in the user’s queue of suggestions pending acceptance or rejection

## 1.4 Assumptions

It is assumed that users will possess compatible devices with a stable internet connection with which to use the Application.

It is assumed that third-party APIs such as book databases and authentication services will be reliable and consistently available.

It is assumed that the development tools, libraries, and platforms needed to build the app are accessible and reliable.

It is assumed that users have a basic understanding of how to navigate and interact with apps.

It is assumed that the books featured in the app are available for purchase, loan, or download in the regions where the app is released.

# 2.0 Design Constraints

## 2.1 Environment

The Application will interact with existing third-party book databases such as Google Books or Open Library to populate the user’s feed with relevant book information. Additionally, the Application will cooperate with authentication services such as Google and Facebook to allow users to create accounts or sign in to existing accounts. Furthermore, the Book Dating Application will make use of a relational or nonrelational DBMS for storing users’ account information.

## 2.2 User Characteristics

The primary end users of the Application are individuals who enjoy reading and are looking for a convenient way to discover new books based on their preferences. Users will come from a wide range of demographics with diverse reading habits, ranging from casual readers to devoted literature enthusiasts. Most end users are expected to be somewhat comfortable with web applications, but the Application is intended to be accessible to users of all skill levels.

## 2.3 System Constraints

In some cases, the design and implementation of the Application are subject to specific constraints imposed by external requirements. Factors such as compliance with external standards and regulations, budget limitations, and organizational policies may impact the options available for development of the Application.

# 3.0 Functional Requirements

This section outlines the functional requirements of the Application, including various features and the expected behavior of the system.

## 3.1 Login

### 3.1.1 Account Creation

The system shall allow users to create an account using their email address or a third-party authentication service such as Google or Facebook. Users will be prompted to create a password, which will be analyzed for security criteria such as length and complexity. Finally, users will be sent a verification email upon account creation and taken back to the login page.

### 3.1.2 Login with Username and Password

The system shall allow users to log in to their existing accounts using their email address or the third-party authentication service of their choice. The credentials entered will be securely validated, and access will be provided only if the credentials are valid. An error message will be shown if either the username or password are incorrect.

## 3.2 Display Main Page

### 3.2.1 Book Suggestions

On the main page, the system shall present users with a queue of books that are personalized to the user’s interests. Each book serves as a potential candidate for being accepted or rejected at the user’s discretion. Books in the queue will be presented along with relevant information about the title such as genre, theme, and more.

### 3.2.2 Accept/Reject Book

Users shall be able to accept or reject the candidates they will be presented with using dedicated “accept” and “reject” interactions. Each of these actions will be paired with visual feedback to reassure the user of the choice they made. Upon acceptance of a candidate, the book and its relevant information will become available in the Saved Books section of the Profile page. Candidates that are rejected are not added to Saved Books and are not recommended again. Regardless of the user’s decision to accept or reject a candidate, the user’s preferences will be updated, which will affect the future candidates that appear in the suggestion queue.

### 3.2.3 User Preference Filtering

The system shall provide filtering options on the main page that allow users to refine their searches based on their preferences and specific criteria. Users will be able to select filters to update their search queue, changing their book suggestions. These preferences shall be saved for future uses of the Application.

## 3.3 Display Profile Page

### 3.3.1 Profile Details

The profile page shall provide details to the user such as personal information, including but not limited to their username and email address. The users’ preferences and filters will also be displayed on the Profile page, allowing them to update these preferences. Users will also be able to securely update their information such as their username and password.

### 3.3.2 Saved Books

The system shall allow the user to view and manage the candidates they have saved. In the Saved Books section of the Profile page, users will be able to organize their saved books into categories or lists (e.g., “To Read,” “Favorites”), remove books from their Saved Books list, and review information regarding the accepted candidates.

# 4.0 Non-Functional Requirements

## 4.1 Security

The system shall enact measures to protect sensitive data such as user login credentials, user preferences and other personal information, and more. Passwords shall be stored securely using strong hashing algorithms and all sensitive data shall be encrypted to high standards. All client-server data transfers will make use of the HTTPS protocol for more secure messaging.

## 4.2 Capacity

The system shall make significant accommodations in its design for factors such as scalability, performance under heavy load, and data storage. The system shall be able to accommodate many concurrent users during peak usage times and ensure that multiple book recommendation queries may be processed simultaneously. Furthermore, the system shall be able to accommodate as many registered users as necessary as well as support the storage of detailed information from a large database of books. This will be accomplished by taking measures such as efficient algorithm and database design, and advanced cloud infrastructure.

## 4.3 Usability

The system shall prioritize usability by having a simple yet efficient user interface that requires minimal actions to reach a desired page. The system shall be designed for use in multiple environments with varying noise levels, so dependence on audible output for feedback will be kept at a minimum. Furthermore, accessibility features shall be incorporated into the web page in accordance with Web Content Accessibility Guidelines. Finally, the system shall provide efficient UI performance with minimal response time.

## 4.4 Other

Other non-functional requirements may include, but are not limited to:

1. Support for multiple common web browsers such as Google Chrome and Microsoft Edge
2. Support for multiple languages
3. Consistent server uptime
4. Sufficient performance, including fast startup times and low latency