

**FACULTY OF ENGINEERING**

**COMPUTER ENGINEERING DEPARTMANT**

**FALL 2021**

**SOFTWARE ENGİNEERİNG**

Customized Visual Bookshelf from Gutenberg Project

14.12.2021

170316036 – Kardelen ÇETİN

170316044 – Muzaffer ÖZEN

170316006 – Ayseli Erem BATI

170316053 – Tuğrul Can MERCAN

200315006 – Ömer BENEK

Manisa Celal Bayar University Computer Engineering Department

CSE4123 Software Engineering Fall 2021

Project Documentation

Customized Visual Bookshelf from Gutenberg Project

**Table of Contents**

Table of Contents………………………………………………………………………………2

Introduction ……………………………………………………………………………………3

Proposed System ………………………………………………………………………………3

Overview ………………………………………………………………………………………3

User Reqiurements ………………………………………………………………….................3

System Requirements ………………………………………………………………………….4

System Stakeholders ……………………………………………………………………..........4

Functional Requirements ……………………………………………………………………...5

Non-Functional Requirements………………...……………………………………………..5,6

System Model …………………………………………………………………………………7

Scenarios ....................................…………………… …………………………………

Use Case Model ....................................………………………………………………..

Object Model ………………………………………………………………………......

Dynamic Model ……………………………………………………………………......

User Interface Mock-ups ………………………………………………………………

**Introduction**

The name of our application is Libros. The main purpose of our application is to be able to search for and read books and collect selected books in a visual library for a specific user of the system. Each user should be able to see their own library when they log in and change the books in their own shelf by making new searches. The user can remove a book from the shelf when the book is finished. The user should be able to see the page in the book he/she has stopped reading the next time he/she logs into the system.

**Proposed System**

1. The feature that distinguishes the system we propose from other systems is that each user should see their own library when they log in and be able to change the books on their shelf by making new searches, or they can remove the book from the shelf when the book is finished.
2. In addition, another feature that distinguishes it from other systems is that the user can see the page on which he paused reading. In addition to the visualized library, the user is also shown a pop-up screen showing which book is paused on which page.
3. This application, which simulates the virtual library, can make many people gain the habit of reading books.
4. In addition, we take the burden of books off our shoulders, and we can easily access thousands of books or our virtual library that we have created by phone or via the web. In this way, we can easily access our library and read books at every moment of our lives.

**Overview**

The purpose of this system is to provide features such as reading and viewing a book. The user can see the page on which you are left in the book, create a library and remove a book from the shelf when the book is finished.

**User Reqiurements**

1. Each user must have their own account and be able to log in to the system with this account.
2. The application shall provide the ability to search and read books for the user.
3. The application shall show the user where the user has left in their boks.
4. The application shall the user to add the books he likes to his/her library.

**System Requirements**

1. Each user must have their own account and be able to log in to the system with this account.
   1. The information of the user's account should be able to be stored in the system.
   2. The system should be able to remember the user in all logins after the first login to the system.
   3. Each user must have an individual account in the system.
2. The application shall provide the ability to search and read books for the user.
   1. Books should be able to be searched by filtering in the system.
   2. The user should be able to access the books registered in the system.
   3. A list of the books that the user has read in the system should be able to be kept.
3. The application shall show the user where the user has left in their boks.
   1. The system should be able to show the user's last stay in the books.
   2. The system must refresh the updated information each time the user logs in.
   3. When the user logs out or closes the application, the system should still be able to show the page the user was left on.
4. The application shall the user to add the books he likes to his library.
   1. The system should be able to add the books that the user likes to her/his library.
   2. The system should be able to delete the books that the user does not want to be in her library.
   3. The system should be able to update the user's library and store the updated data.

**System Stakeholders**

* Users whose have an account in the system.
* Software Developers ​who are responsible for installing and updating the system.
* The company that provides the books data to the application.

**Functional Requirements**

* The user must be able to login to the system.
* The user must be able to register in the system.
* If the user forgets her/his password, she/he should be able to change her password.
* The user should be able to create a library on the system.
* The user should be able to view the books in the library.
* User should be able to delete books in library.
* The user should be able to see where she/he left off in her books.
* User should be able to search for books.
* The user should be able to add books to the library.
* The user should be able to search for another book while reading a book.

**Non-Functional Requirements**

**->Performance:**

**-**Users can access books all day long.

**->Usability:**

-Users should have access to all 60,000 books available in the app.

-Users can access the books in the system with the name of the author and the name of the book.

-When the user continues to read a book, he will be able to see which page he has left.

**->Functionalty:**

-The system should provide more than one functions and services to the user.

**->Reliability:**

-Only the user who created the account can access the account created by the user.

**->Supportability:**

-The application can support up to 10000 users.

-Users can only read the books in the application online.

**->Rapid Develeopment:**

-The system should be developed quickly.

**->Low-Coast:**

-While designing the system, the coast should be kept as low as possible.

**->Fault Tolerance:**

-Fault tolerance should be as low as possible.

**System Model**

**Scenarios**