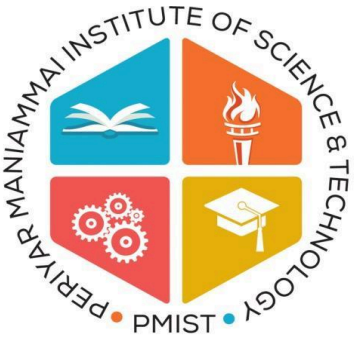


PERIYAR MANIAMMAI
INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University)
Established Under Sec. 3 of UGC Act, 1956 • NAAC Accredited
think • innovate • transform

PROJECT REPORT

TOPIC: BOOK RECOMMENDATION SYSTEM

COURSE NAME: INTRODUCTION TO ARTIFICIAL INTELLIGENCE

COURSE CODE: XCSHAI

BY:

S.AFRIDHA SHAHEEN[123011019003]

E.RACHEL[123011019024]

Project Title: Book Recommendation System Based on Age Group and Genre

Technology Stack:

- **Frontend**: HTML, CSS (for styling)
- **Backend**: Python (Flask Framework)
- **Database**: Hardcoded book recommendations (dictionary)

1. Project Overview

The Book Recommendation System is a web-based application that recommends books to users based on two primary inputs:

1. **Age Group**: Users can select from categories like children, teen, and adult.
2. **Genre**: Based on the selected age group, the system offers genre-based recommendations such as fiction, non-fiction, fantasy, comedy, drama, and thriller.

The primary goal of this project is to offer an interactive and user-friendly platform that helps users find books tailored to their age and interest.

2. Objective

The objective of this project is to develop a functional web application using Flask that:

1. Takes user input for age group and genre.
2. Displays a list of book recommendations based on the provided inputs.
3. Presents the recommendations in a clean, organized, and aesthetic user interface.

3. Functional Requirements

1. **User Input Form**:
    - The user will be able to select their **Age Group** from a dropdown list with options such as:
        - Children
        - Teen
        - Adult
    - The user will be able to select their **Genre** from another dropdown list with options such as:
        - Fiction
        - Non-Fiction
        - Fantasy
        - Comedy
        - Drama
        - Thriller
2. **Book Recommendations**:
    - Based on the user's selected **Age Group** and **Genre**, the system will display a list of book recommendations.
    - If no matching books are found, the system will display a message like, "Sorry, no recommendations available for the selected criteria."
3. **Frontend Design**:
    - The web page will be simple and clean, with two input fields (age group and genre) and a button to get recommendations.
    - A results section will show the recommended books after the user submits the form.
4. **Backend Logic**:
    - The backend will handle the logic for matching the age group and genre with predefined book data.

- The application will use a dictionary to store book data, categorized by age group and genre.

4. Non-Functional Requirements

1. **Performance**:
   - The application should respond to user inputs and display recommendations without significant delay.
2. **Usability**:
   - The user interface should be intuitive, allowing users to easily input their choices and view results.
3. **Security**:
   - Basic security measures like input validation should be implemented to prevent invalid or malicious inputs.

---

5. System Architecture

The system architecture follows a **Model-View-Controller (MVC)** pattern:

- **Model**: The backend logic that stores and processes the book recommendations.
- **View**: The frontend interface that collects user input and displays the results.
- **Controller**: The Flask application that handles the HTTP requests, processes the data, and renders the appropriate template.

Flow of Data:

1. The user interacts with the frontend interface by selecting their age group and genre.

2. The form is submitted to the Flask backend using a POST request.
3. The backend processes the form data and selects the appropriate book recommendations from the predefined database (hardcoded dictionary).
4. The recommendations are displayed on the webpage.

PROGRAM:

```
from flask import Flask, render_template, request

app = Flask(__name__)

# Predefined book database

book_recommendations = {

"children": {

"fiction": ["Harry Potter Series by J.K. Rowling", "Charlotte's Web by E.B. White"],

"non-fiction": ["National Geographic Kids by National Geographic Society", "Goodnight Stories for Rebel Girls by Elena Favilli"],

"fantasy": ["The Chronicles of Narnia by C.S. Lewis", "Percy Jackson Series by Rick Riordan"],

"comedy":["Diary of a Wimpy Kid by Jeff Kinney","Captain Underpants by Dav Pilkey","Wayside School is Falling Down by Louis Sachar"],

"drama":["Wonder by R.J. Palacio","Bridge to Terabithia by Katherine Paterson","The Miraculous Journey of Edward Tulane by Kate DiCamillo"],

"thriller":["Goosebumps series by R.L. Stine","The Mysterious Benedict Society by Trenton Lee Stewart","Escape from Mr. Lemoncello's Library by Chris Grabenstein"],

},

"teen": {

"fiction": ["The Fault in Our Stars by John Green", "To All the Boys I've Loved Before by Jenny Han"],
```

"non-fiction": ["Becoming by Michelle Obama (Young Readers Edition)", "The 7 Habits of Highly Effective Teens by Sean Covey"],

"fantasy": ["The Hunger Games by Suzanne Collins", "Divergent Series by Veronica Roth"],

"comedy":["The Absolutely True Diary of a Part-Time Indian by Sherman Alexie","The Princess Diaries by Meg Cabot","Spud by John van de Ruit"],

"drama":["The Perks of Being a Wallflower by Stephen Chbosky","Speak by Laurie Halse Anderson","If I Stay by Gayle Forman"],

"thriller":["One of Us Is Lying by Karen M. McManus","We Were Liars by E. Lockhart","Miss Peregrine's Home for Peculiar Children by Ransom Riggs"]

},

"adult": {

"fiction": ["The Great Gatsby by F. Scott Fitzgerald", "Where the Crawdads Sing by Delia Owens"],

"non-fiction": ["Sapiens by Yuval Noah Harari", "Educated by Tara Westover"],

"fantasy": ["The Name of the Wind by Patrick Rothfuss", "A Game of Thrones by George R.R. Martin"],

"comedy":["Good Omens by Neil Gaiman and Terry Pratchett","Bossypants by Tina Fey","Catch-22 by Joseph Heller"],

"drama":["A Thousand Splendid Suns by Khaled Hosseini","The Night Circus by Erin Morgenstern","Little Fires Everywhere by Celeste Ng"],

"thriller":["The Girl with the Dragon Tattoo by Stieg Larsson","Gone Girl by Gillian Flynn","The Silent Patient by Alex Michaelides"]

}

}

@app.route("/", methods=["GET", "POST"])

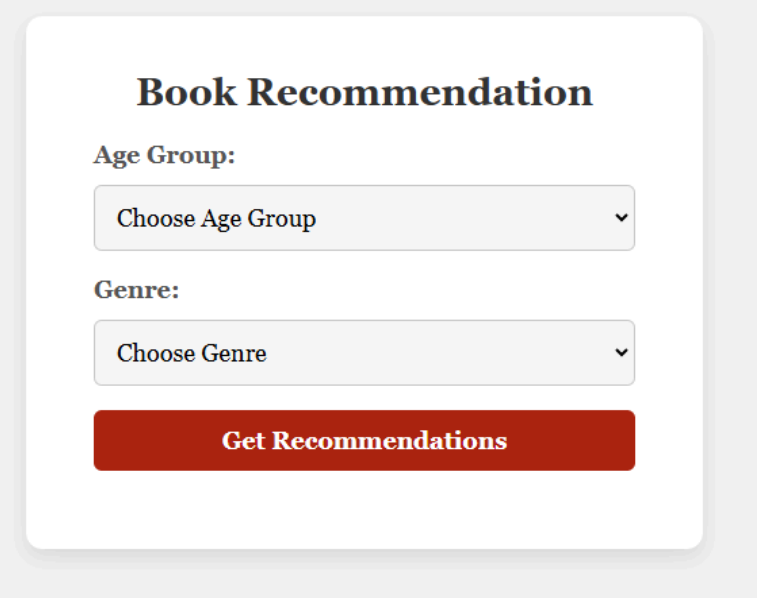
def recommend_books():

recommendations = []

```
if request.method == "POST":

age_group = request.form.get("age_group")

genre = request.form.get("genre")

if age_group in book_recommendations and genre in
book_recommendations[age_group]:

recommendations = book_recommendations[age_group][genre]

else:

recommendations = ["Sorry, no recommendations available for the selected criteria."]

return render_template("index.html", recommendations=recommendations)

if __name__ == "__main__":

app.run(host="0.0.0.0", port=5000, debug=True)
```

RESULT:

- Book Recommendation System was successfully developed and tested. When users select an age group and a genre, the system returns a list of books matching the criteria.



Book Recommendation

Age Group:

Choose Age Group

Genre:

Choose Genre

Get Recommendations

Recommended Books:

- One of Us Is Lying by Karen M. McManus
- We Were Liars by E. Lockhart
- Miss Peregrine's Home for Peculiar Children by Ransom Riggs