Hello, everyone!

Today I’m presenting a recommendation system that predicts wich books certain users are likely to enjoy, based on their previous interactions, such as book reviews and ratings. Because, as a wise old lady always says, reading is fundamental."

**Dataset and Preprocessing**

So I wanted to base this Project on reliable data, such as a Goodreads data. Goodreads is basically the most relevant social platform for book lovers. It’s a place where users can rate, review, and recommend books. One of the main challenges I faced was that the Goodreads API was shut down in 2020, which made it difficult to access book data and ratings. I started by navigating through Kaggle and Hugging Face, looking for datasets. Luckily, I found one that included user reviews, ratings, and other details like publication dates, text reviews, number of comments of the text reviews...

I began cleaning and preprocessing the dataset, removing missing or irrelevant data and filtering out users and books that didn’t have enough interactions. This step was super important to make sure the dataset was both relevant and manageable. I used Pandas to structure data in a format that the Surprise Library can work with for collaborative filtering.

**Building the Model**

So for this approach, I used a recommendation system. These systems are behind a lot of things we use every day, like Netflix suggestions for what you should watch next, or when Spotify creates playlists just for you, based on your tastes.

How does a recommendation system actually work? I used something called a K-Nearest Neighbors model, a collaborative filtering technique, which is a fancy way of saying: 'Find people who like similar books.' The idea is that if others who share your taste loved certain books, you'll probably enjoy them too.

To train the model, we took our dataset of book ratings and split it in two—one part for the model to learn from and the other part to test how well it makes predictions. The idea is to teach the model how to match similar readers and then check how accurate its recommendations are.

Finally, I used a metric called RMSE to measure how close the system's predictions are to actual user ratings. The lower the number, the better the system is at predicting what books you'll like. This is a way to make sure the system isn't just guessing randomly—it’s actually using meaningful data to make suggestions

**How does it work?**

The model looks at all the books in the dataset, check wich books one user has read or rated and compare them with other users who have similar tastes. Then, for every book our user hasn’t read, the model gives a predicted score, and the books with the highest predicted ratings are the ones that get recommended to them.

So, right now, the model is built with five preset profiles: Adam, Lucrecia, Maria, Peter, and Andrea. These are the only options available, meaning the system is limited to recommending books for these specific users.

**User Interaction with the System**

So, to make it easy to use, I integrated something called Gradio. Its a simple web interface where users can just type in their name and the number of book recommendations they want. Just enter the info, and the system does the rest.

Behind the scenes, the system looks at the user’s reading history and uses our model to predict books they haven’t read yet but would probably love. It works by comparing your tastes with other users who enjoy similar books.

And here’s another cool part: the dataset we use for the model (the one from Surprise) doesn’t contain book titles, just book IDs, user IDs, and ratings. So, to give the recommendations some personality, we had to get a bit creative. For each book the system recommends, it goes online and scrapes Goodreads to grab the actual book title and author name using BeautifulSoup and requests. This way, when you get a recommendation, you’re not just seeing a book ID, but the actual details, making the experience feel more personalized and user-friendly.

**Next Steps**

One idea for the future of this model, is to let anyone enter their own Goodreads user ID, and based on that information, the model will make live predictions about which books they should read next, creating a more dynamic and personalized experience for each user.

**Conclusion**

In conclusion, this Project was challenging and fun to make.

I want to give a huge thank you to AllWomen, my amazing teachers, and of course, to all of you for listening. I hope you all find your next favorite book! Thanks again, and happy reading!"