Problem Statement 1

Problem Statement: Conversational Fashion Outfit Generator powered by GenAI.

Description:

With the advent of Generative AI, Search and finding a product is being revolutionised. We are moving away from a single open text box experience to something more conversational and this will enable product discovery and recommendations to be a lot more powerful than they are today by way of being able to truly understand the user's needs in a more human-like conversational way. Fashion is one of the categories where discovery will get reset.

As part of this challenge, teams need to create an **Gen AI-powered fashion outfit** generator for Flipkart that revolutionizes the way users discover and create personalized fashion outfits, in a natural conversational way. The outfit generator should leverage user's past purchase history, preferences based on browsing data, and insights from social media trends to offer tailored and on-trend outfit recommendations.

The **fashion outfit generator** should have the ability to analyze a user's past purchase history and understand their preferred style, color choices, and favorite brands. By considering these preferences, the generator should look to suggest outfits that align with the user's unique fashion taste. Additonally, the generator can take into account the types of clothing items the user frequently views or adds to their cart, ensuring that the outfit recommendations are relevant and appealing.

In addition to individual user data, the fashion **outfit generator should tap into social media trends to provide up-to-date fashion recommendations. It should be able to analyze current fashion trends, styles, and influencers on platforms like Instagram, Pinterest, and fashion blogs.** By combining this data with the **user's preferences, the generator can suggest outfits that are not only personalized but also in line with the latest fashion trends.**

The generated outfit recommendations should be complete and well-coordinated, including clothing, accessories, and footwear etc. The generator should consider factors such as the user's body type, occasion (e.g., casual, formal, party), and regional and age preferences (Ex. Young 20 year old woman looking for a Diwali outfit in Mumbai should be different to 35 year old woman in Muzzafarpur looking for a Karwa Chauth outfit) to offer appropriate and versatile outfit suggestions. Users should also be able to interact with the outfit generator to give it feedback in terms of what they like, dont like and be able to tweak the outfits in the manner of a conversation (Ex. I like the top, but the jhumkas are boring, give me something else).

The ultimate goal of the fashion outfit generator is to enhance the user's shopping experience on Flipkart by providing them with personalized, trendy, and cohesive outfit ideas. Users should feel inspired and confident in their fashion choices, knowing that the generator has considered their preferences, browsing habits, and the latest fashion trends.

Problem Statement 2

Problem Statement: Personalized Product Recommendations

The aim is to enhance user experience by implementing a personalized product ranking system. Your task is to develop an algorithm or model that can generate accurate and relevant product rankings for individual users. The ranking system should consider factors such as user preferences, past interactions, product popularity, and user similarity. It should be able to predict the most suitable products for a user based on their unique characteristics and preferences.

You are not provided with a specific dataset for this challenge. Instead, you are expected to design and implement a solution that simulates user interactions and generates personalized rankings. You can define user profiles, product categories, and interaction patterns within your solution.

To evaluate the effectiveness of your solution, you should define appropriate metrics for measuring the accuracy and relevance of the rankings. You should also provide a report explaining your approach, describing the algorithms or techniques used, and discussing the strengths and limitations of your solution.

Problem Statement 3

💡 **Blockchain-based: Loyalty and Rewards Program using Fungible Tokens:**

The objective is to develop a blockchain-enabled loyalty and rewards program for E-commerce platforms, leveraging the advantages of blockchain technology to enhance security, transparency, and user engagement.

- Develop a blockchain-based project to generate loyalty points as fungible tokens on the blockchain.
- Define the tokenomics for the fungible tokens such as the value of tokens, and the number of tokens to be issued. Rules and regulations for the governance of treasury to be managed on a day-to-day basis should be also clearly defined.
- The account settlement/reconciliation process of fungible tokens should be made more effective between Brands and E-commerce platforms, meaning settlements of fungible tokens should be simple and instant and should be recorded on-chain.
- Define rules for earning fungible tokens based on user actions such as purchases, referrals, or social media interactions and distribute these fungible tokens to customers' digital wallets securely and transparently using blockchain transactions.
- Platform partners or sellers should also be able to issue these fungible tokens to loyal customers.
- Customers can then use their fungible tokens to earn rewards from sellers or partners, each being recorded on the blockchain for transparency and to prevent double-spending.
- All the transactions revolving around fungible tokens should be recorded on-chain.
- Build a user-friendly interface for users to manage their loyalty points, view available rewards, and track their progress. The interface should provide a clear overview of earned loyalty points, past transactions, and available redemption options.
- You can use the Polygon blockchain to deploy your solution and demo the final product as a web prototype.

What we are expecting:

- The loyalty program should generate fungible tokens on the blockchain as loyalty points, representing the value earned by users through various actions such as purchases, referrals, or social media interactions.
- Users should be to understand the criteria for earning fungible tokens, including clear rules and guidelines for different actions and transactions.
- Tokenomics of fungible tokens should be transparent to brands to accept and issue tokens as well as to manage their collective treasury.
- Any user should be able to track their loyalty points(fungible tokens) and view their earning history.
- Partners and sellers can easily reward their loyal users and should be able to solve their settlements of fungible tokens with E-commerce platforms easily on-chain.
- All the transactions involving fungible tokens should be on-chain and recorded, proving every settlement performed.
- Bonus: GUI-based tool that doesn't require knowledge of any Blockchain programming to use by Brands and Retailers or customers.
- Bonus: Add decaying nature for fungible tokens in the customer's wallet after a certain period.

Judging Criteria:

1. The demo of the key feature/functions you were able to implement should be simple (No code to be used in the demo)
2. Clean and well-structured Solidity code/templates
3. Feature sets