CODING BLOCKS
Code Your Way To Success

PROJECT - 1

Coding Blocks-Enhanced Text Extractor Tool

Objective:
Develop an Enhanced Text Extraction and Enrichment tool that utilizes the Langchain library for text extraction and OpenAI for data enrichment. The tool will be implemented as a web application using Next.js/React.js/Node.js stack. It will allow users to upload files, extract text from PDF formats, enrich the extracted data using OpenAI, and display the processed results in an interactive table format.

Deliverables:

A fully functional Next.js/NodeJS web application.
An intuitive and interactive user interface for file uploads and data display.
Comprehensive documentation covering setup, usage, and design decisions.
Requirements:

Features:
Text Extraction:

1. Implement text extraction functionality to extract text from uploaded files.
2. Support extraction from PDF file formats.

Data Enrichment:

1. Utilize OpenAI to enrich the extracted text data with additional insights or context.

API Development:

1. Develop RESTful API routes for handling file uploads, text extraction, and data enrichment.

Interactive UI: Create an engaging and responsive user interface that allows seamless file uploads and displays the processed results in an interactive table format.
Error Handling:

Implement robust error handling mechanisms to gracefully manage unexpected errors and edge cases during file uploads, text extraction, and data enrichment processes.

Documentation:

1. Provide a detailed README with setup instructions, dependencies, and usage guidelines.
2. Include code comments and function/module descriptions for better understanding and maintainability.

Deployment: Deploy the application on a cloud platform such as Vercel, Netlify, Heroku, or similar, or package it in a zip file for distribution.

User Authentication:
Implement user authentication and authorization mechanisms to secure access to the application and protect sensitive user data.

Customization Options:
Allow users to customize the text extraction and enrichment process by specifying parameters or preferences, such as language settings or enrichment depth.

Collaboration Tools:
Integrate collaboration features such as real-time document editing, comments, and version control to facilitate collaboration among multiple users working on the same document, we hope you must have seen google doc getting used.

Advanced Features:

1. **Documentation:**
   - Include a README with setup instructions and dependencies.
   - Code comments and function/module descriptions.

2. **Deployment**:
   - Deploy the application on a cloud platform (Vercel, Netlify, heroku, or similar) or share in zip file

3. Evaluation Criteria:

- Adherence to project requirements.
- Cleanliness and organization of code.
- UI/UX design and usability.
- Langchain and OpenAI implementation.

Coding Blocks-PROJECT - 2

Build a dashboard that provides analytics for users & repositories on GitHub. The dashboard should allow users to search for a GitHub user and display a table of their repositories with details like stars, forks, open issues, and recent commits.

Features List:
1. User search: An input box to search for users and show a list of cards with the responses. (Need not be paginated, pick first few results as you deem fit)

2. User Repository Table: Clicking on the user navigates to a different page to display a list of a user's repositories with columns for the name, description, number
of stars, forks, and open issues.

3. Sorting: Allow the user to sort repositories by name, stars, forks, or open issues.(could be client side / server side)

4. Search Filter: Implement a search filter to find repositories by name within the user's list. (could be client side / server side)

5. Commit History Visualization: Clicking on a row on the repository table should select the repository and for the selected repository, show a graph of the commit activity over the last year.

6. Contributors List: List contributors for each repository and their contributions Count.

7. Repository Detail View: Create a detailed view for each repository that includes

the README file, a list of recent commits, and open issues. Page which opens up on clicking a “View” button available per each row on the User Repository Table.

8. Rate Limit Handling: Implement a feature to handle GitHub's API rate limiting gracefully, perhaps by caching data or informing the user(Incase you haven’t read about it this hackathon is about to explore new things too)

Advanced Features:

1. **Documentation:**
   - Include a README with setup instructions and dependencies.
   - Code comments and function/module descriptions.

2. **Deployment**:
   - Deploy the application on a cloud platform (Vercel, Netlify, heroku, or similar) or share in zip file

3. Evaluation Criteria:

- Adherence to project requirements.
- Cleanliness and organization of code.
- UI/UX design and usability.
- Langchain and OpenAI implementation.

WEB DEV PROJECTS

By Team

CODING BLOCKS
Code Your Way To Success

CODING BLOCKS
Code Your Way To Success

Coding Blocks-Project 3: Personal Finance Management Application

Develop a personal finance management application to help users track their income, expenses, and savings goals. The application should provide users with insights into their spending habits, budgeting strategies, and financial health.

Features List:

Expense Tracking: Allow users to categorize and track their expenses, including recurring bills, one-time purchases, and discretionary spending.

Budget Management: Enable users to set budget limits for different expense categories and receive notifications when they exceed predefined thresholds.

Savings Goal Tracking: Allow users to set savings goals and track their progress over time, with visual indicators such as pie charts or bar graphs on the basis of monthly and daily tracking and milestone notifications.

Transaction Analysis: Provide insights into spending patterns and trends through interactive charts and graphs, including monthly expenditure breakdowns and category-wise comparisons.

Financial Health Assessment: Offer general recommendations and tips to improve financial literacy, manage debt, and optimize savings.

Secure Data Storage: Ensure that user financial data is encrypted and stored securely to protect privacy and prevent unauthorized access.

Expense Reminders and Alerts: Implement reminders and alerts to notify users of upcoming bill payments, due dates, and budget milestones.

Customizable Reporting: Allow users to generate customized financial reports, export data to spreadsheets, and analyze their financial history.

Integration with Financial Institutions: Explore options for integrating with banks and financial institutions to automate transaction imports and account reconciliation.

Advanced Features:

1. **Documentation:**
   - Include a README with setup instructions and dependencies.
   - Code comments and function/module descriptions.

2. **Deployment**:
   - Deploy the application on a cloud platform (Vercel, Netlify, heroku, or similar) or share in zip file

3. Evaluation Criteria:

- Adherence to project requirements.
- Cleanliness and organization of code.
- UI/UX design and usability.
- Langchain and OpenAI implementation.

Coding Blocks-Project 4 : Farmers Friend

Create a user-friendly smart agriculture advisory app aimed at providing farmers with essential information and recommendations to enhance crop yield and efficiency in resource usage.

Features List:
Crop Suitability Assessment: Farmers input details about their land, including soil type, temperature, and geographical location, into the app. The app then provides general recommendations on suitable crops for cultivation in that area based on predetermined guidelines and agricultural best practices.

Irrigation Schedule Recommendations: Based on the user-provided data and pre-defined irrigation schedules for different crop types, the app generates basic irrigation recommendations. It suggests general watering intervals and quantities, considering factors such as soil type and weather conditions.

Weather Forecast Integration: The app integrates weather forecast data from reliable sources to provide farmers with basic weather information for their region. This includes forecasts for precipitation, temperature, and wind speed, allowing farmers to plan their activities accordingly.

Resource Management Tips: The app offers general advice and best practices for managing resources such as water, fertilizers, and pesticides. It provides tips on efficient irrigation methods, fertilizer application techniques, and pest control strategies to help farmers optimize resource usage and reduce wastage.

Community Forum: Farmers can connect with each other through a community forum within the app. They can share experiences, ask questions, and exchange advice and tips related to agriculture. The forum serves as a platform for peer support and knowledge sharing among farmers.

Alerts and Notifications: The app sends periodic alerts and notifications via email to users with important updates, reminders, and relevant information, the

farmer will upload a schedule on the app and according to that schedule app will provide the alerts and notification to the farmer.
This includes reminders for routine farm tasks, weather alerts for adverse conditions, and notifications.

Data Visualization Tools: Utilize interactive charts, heatmaps, and geospatial visualizations to
present complex data in a user-friendly format(D3.js or similar library), allowing stakeholders to explore trends and make data-driven decisions.
Must to have:

1. Top Inventory a farmer bought for crops season wise/yearly.
2. Expenditure for growing crops season wise/yearly.
3. Sales- season wise/yearly.

Advanced Features:

1. **Documentation:**
   - Include a README with setup instructions and dependencies.
   - Code comments and function/module descriptions.

2. **Deployment**:
   - Deploy the application on a cloud platform (Vercel, Netlify, heroku, or similar) or share in zip file

3. Evaluation Criteria:

- Adherence to project requirements.
- Cleanliness and organization of code.
- UI/UX design and usability.
- Langchain and OpenAI implementation.

CODING BLOCKS
Code Your Way To Success

Project 5:Coding Blocks-AmatShop

Develop an e-commerce analytics dashboard to help online retailers analyze customer behavior, track sales performance, and optimize marketing strategies. The dashboard should integrate with e-commerce platforms to collect transaction data and generate actionable insights for business growth.

Features List:
Sales Performance Metrics: Display key performance indicators such as total revenue, average order value, conversion rates, and customer lifetime value, with trend analysis and
benchmarking.

Customer Segmentation: Segment customers based on demographics, purchase history(for example: Monthly expenditure graph to present this data), and product recommendations(If a user has bought 5 electronics items then we can suggest to him more electronics items that he hasn't purchased yet).

Product Performance Analysis: Analyze sales trends, and product profitability to identify top-selling items, slow-moving inventory(Sorted way the product sold the least number of times).

Cart Abandonment Recovery: Implement strategies to reduce cart abandonment rates by sending reminders, discount offers, or exit-intent pop-ups(you can flash package) to incentivize completing purchases.

Customer Retention Strategies: Develop loyalty programs, referral incentives, and rewards to encourage repeat purchases(For example buy a product and if you bought the product 5 different times then you'll be eligible to get 1499 rupees off on 6th purchase)and foster customer loyalty.

Customer Feedback Analysis: Monitor customer reviews, ratings, and feedback to identify product improvements, resolve issues, and enhance overall customer satisfaction.

Data Visualization Tools: Utilize interactive charts, heatmaps, and geospatial visualizations to present complex data in a user-friendly format(D3.js or similar library), allowing stakeholders to explore trends and make data-driven decisions. For example: Electronics items are sold in top 5 countries, Dairy Products top 5 companies, etc.

Advanced Features:

1. **Documentation:**
   - Include a README with setup instructions and dependencies.
   - Code comments and function/module descriptions.

2. **Deployment**:
   - Deploy the application on a cloud platform (Vercel, Netlify, heroku, or similar) or share in zip file

3. Evaluation Criteria:

- Adherence to project requirements.
- Cleanliness and organization of code.
- UI/UX design and usability.
- Langchain and OpenAI implementation.

WEB DEV PROJECTS

By Team

CODING BLOCKS
Code Your Way To Success

CODING BLOCKS
Code Your Way To Success

Project Title 6:
Coding Blocks-EduTrack: Visualizing Course Sales by Continent and Topic

Description:
EduTrack is a data visualization project aimed at tracking the sales performance of an organization offering classroom and online courses across different continents and topics. Developed using D3.js or similar library, the project provides interactive pie charts to visualize sales data by country, continent, and course topic. Additionally, users can explore detailed information about enrolled students for each course.

Features:

Continent-wise Sales Visualization: Display a pie chart showing the distribution of course sales across continents, allowing users to hover over each segment to view sales percentages and total revenue.

Topic-wise Sales Analysis: Provide separate pie charts for each course topic (e.g., AI, Web, Android) within the online courses category, showcasing the proportion of sales for each topic relative to total online/offline course sales. You need to include multiple topics/subjects as per your best knowledge.

Classroom vs. Online Sales Comparison: Present a comparative visualization of sales between classroom and online courses, illustrating the revenue share and growth trends for each vertical using the graph as per your best knowledge. Filter it on the basis of months and year only.

Student Enrollment Details: Allow users to access a detailed list of enrolled students for each course, including their names, contact information, enrollment dates, and course progress.

Interactive User Interface: Create an intuitive user interface with interactive features such as tooltips, zooming of a video, and filtering options to enhance the user experience and facilitate data exploration.

Responsive Design: Ensure that the project is responsive and accessible across multiple devices, including desktops, tablets, and smartphones, to accommodate users with different preferences and screen sizes.

Data Security and Privacy: Implement measures to protect sensitive student information, such as encryption, access controls, and data anonymization, to comply with privacy regulations and maintain data integrity.

Hackathon-ready Solution: Package the project with clear instructions for setup, configuration, and customization, enabling high school students to quickly deploy and present their visualization solution during hackathons or competitions.

Advanced Features:

1. **Documentation:**
   - Include a README with setup instructions and dependencies.
   - Code comments and function/module descriptions.

2. **Deployment**:
   - Deploy the application on a cloud platform (Vercel, Netlify, heroku, or similar) or share in zip file

3. Evaluation Criteria:

- Adherence to project requirements.
- Cleanliness and organization of code.
- UI/UX design and usability.
- Langchain and OpenAI implementation.