

## Task 1: Python & AI (30 points)

### Machine Learning Model for predicting Loan approval

#### • Approach & Solution

Loan approval is a critical process in the financial industry, impacting individuals and businesses seeking financial assistance. Traditional methods often involve manual assessments, which can be time-consuming. This project builds a simple classification model that can accurately predict loan approvals using financial data.

#### Libraries:

- NumPy
- Pandas
- Matplotlib
- Seaborn
- Scikit-learn(sklearn)

### Pipeline for the model building process

#### 1. Data Loading and Exploration:

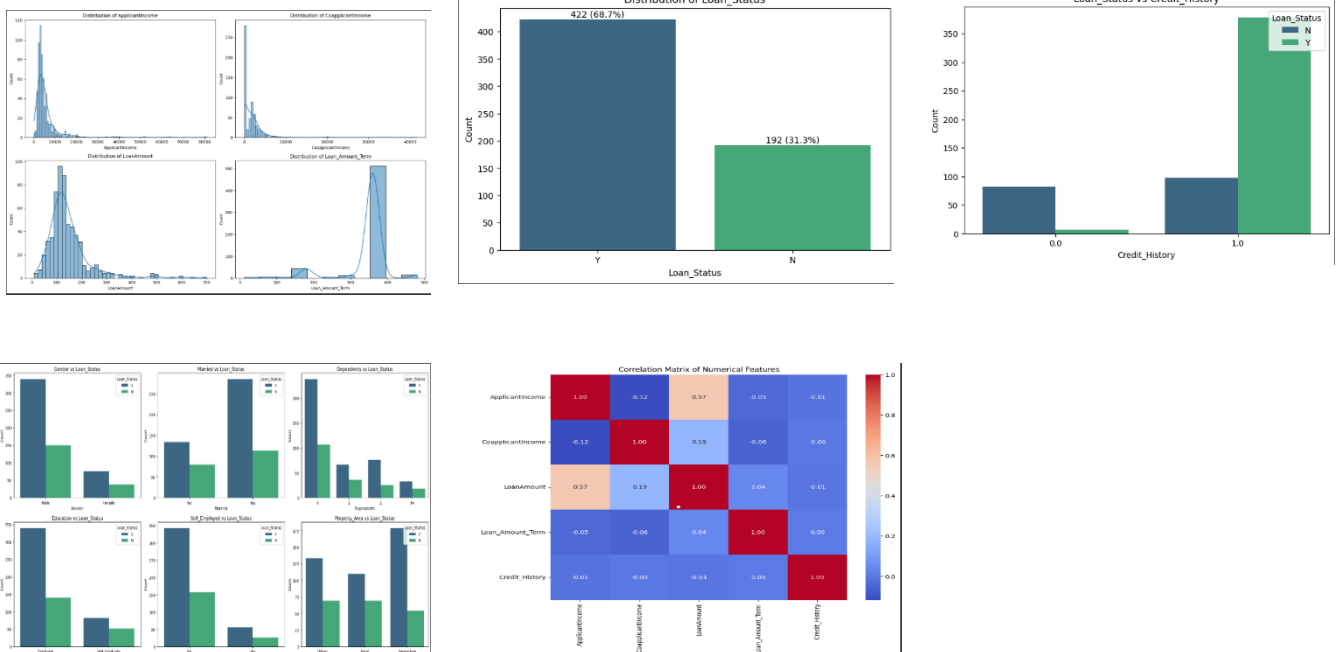
Load the dataset using `pd.read_csv('Finance.csv')`. Explore the data using functions like `df.info()`, `df.describe()`, `isnull().sum()` and visualizations (e.g., `sns.countplot()`).

```
First 5 rows of the dataset:
   Loan_ID  Gender  Married  Dependents  Education  Self_Employed  ApplicantIncome  CoapplicantIncome  LoanAmount  Loan_Amount_Term  Credit_History  Property_Area  Loan_Status
0  LP001002  Male    No       0         Graduate    No                5849              0.0           NaN           360.0           1.0         Urban         Y
1  LP001003  Male    Yes      1         Graduate    No                4583              1508.0        128.0         360.0           1.0         Rural         N
2  LP001005  Male    Yes      0         Graduate    Yes               3000              0.0           66.0          360.0           1.0         Urban         Y
3  LP001006  Male    Yes      0         Not Graduate  No                2583              2358.0        120.0          360.0           1.0         Urban         Y
4  LP001008  Male    No       0         Graduate    No                6000              0.0           141.0         360.0           1.0         Urban         Y

Dataset Shape: (614, 13)

Dataset Information:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 614 entries, 0 to 613
Data columns (total 13 columns):
 #   Column              Non-Null Count  Dtype
---  --
 0   Loan_ID             614 non-null    object
 1   Gender              601 non-null    object
 2   Married              611 non-null    object
 3   Dependents           599 non-null    object
 4   Education            614 non-null    object
 5   Self_Employed        582 non-null    object
 6   ApplicantIncome      614 non-null    int64
 7   CoapplicantIncome    614 non-null    float64
 8   LoanAmount           592 non-null    float64
 9   Loan_Amount_Term     600 non-null    float64
10   Credit_History        564 non-null    float64
11   Property_Area        614 non-null    object
12   Loan_Status          614 non-null    object
dtypes: float64(4), int64(1), object(8)
memory usage: 62.5+ KB
```

- Visualized data distributions and relationships using histograms, count plots, box plots, and a correlation heatmap.



## 2. Data Cleaning and Preprocessing:

Handled missing values using imputation techniques `.fillna()`, `.median()`.

Detected and treated outliers, potentially using the IQR method.

Encoded categorical variables using Label Encoding. `LabelEncoder()`

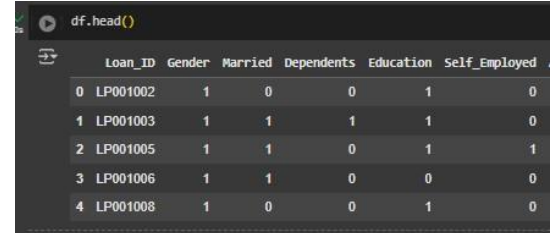
Scaled numerical features using StandardScaler. `StandardScaler()`

Handle missing values by imputation e.g., using `fillna()` with `mode(.mode())` or median `(.median())`. Convert categorical features to numerical using techniques like one-hot encoding(`pd.get_dummies()`).



```
df.head()
```

	Loan_ID	Gender	Married	Dependents	Education	Self_Employed
0	LP001002	Male	No	0	Graduate	No
1	LP001003	Male	Yes	1	Graduate	No
2	LP001005	Male	Yes	0	Graduate	Yes
3	LP001006	Male	Yes	0	Not Graduate	No
4	LP001008	Male	No	0	Graduate	No



```
df.head()
```

	Loan_ID	Gender	Married	Dependents	Education	Self_Employed
0	LP001002	1	0	0	1	0
1	LP001003	1	1	1	1	0
2	LP001005	1	1	0	1	1
3	LP001006	1	1	0	0	0
4	LP001008	1	0	0	1	0

## 3. Feature Engineering:

Created new features like `'Debt_to_Income_Ratio'`, `'Loan_Amount_to_Income_Ratio'`, `'Total_Income'`, and `'Income_Stability'`.

Encoded the new categorical feature `'Income_Stability'` using Label Encoding.

Visualized the distributions of new features and their relationships with the target variable.

## 4. Data Splitting and Feature Scaling:

Split the data into training and testing sets using `train_test_split`.

Applied `StandardScaler()` to the features for scaling.

## 5. Model Building and Evaluation:

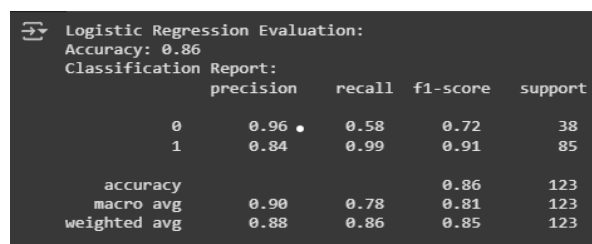
Implemented multiple models: `Logistic Regression`, `Random Forest`, and `Decision Tree`.

Evaluated models using metrics like `accuracy`, `precision`, `recall`, `F1-score`, and `ROC-AUC`.

Visualized model performance using confusion matrices and ROC curves.

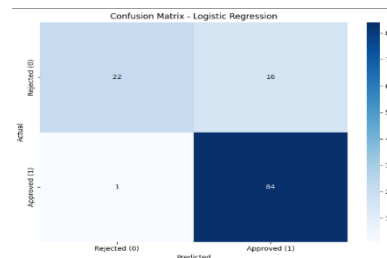
Compared models based on their performance metrics.

### Logistic Regression

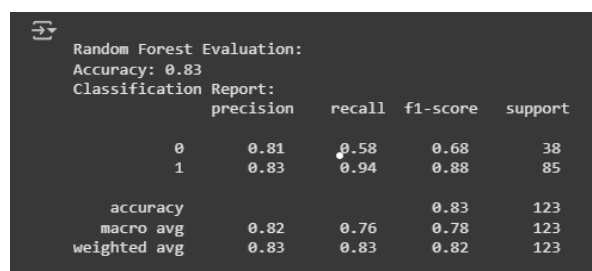


```
Logistic Regression Evaluation:  
Accuracy: 0.86  
Classification Report:
```

	precision	recall	f1-score	support
0	0.96	0.58	0.72	38
1	0.84	0.99	0.91	85
accuracy			0.86	123
macro avg	0.90	0.78	0.81	123
weighted avg	0.88	0.86	0.85	123

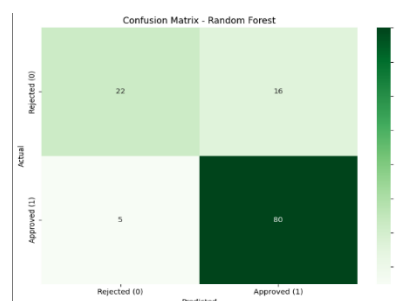


### Random Forest



```
Random Forest Evaluation:  
Accuracy: 0.83  
Classification Report:
```

	precision	recall	f1-score	support
0	0.81	0.58	0.68	38
1	0.83	0.94	0.88	85
accuracy			0.83	123
macro avg	0.82	0.76	0.78	123
weighted avg	0.83	0.83	0.82	123



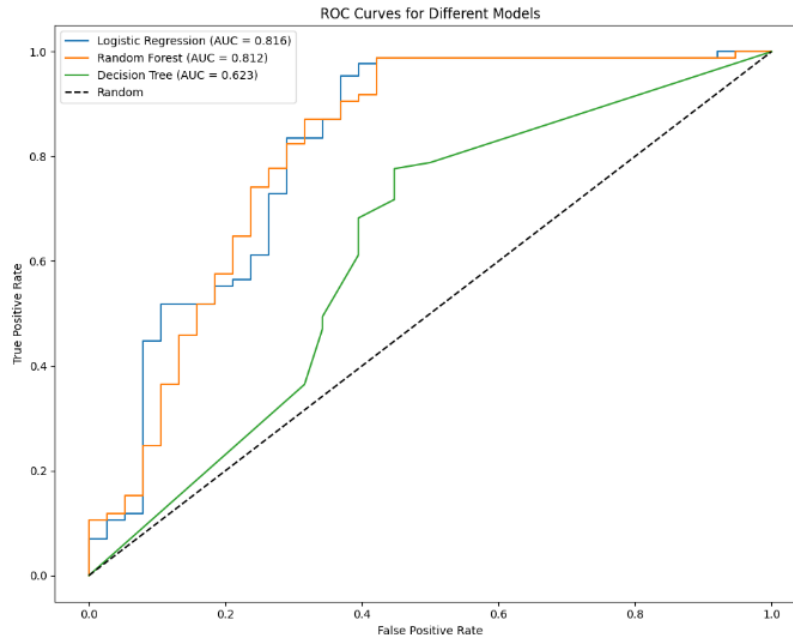
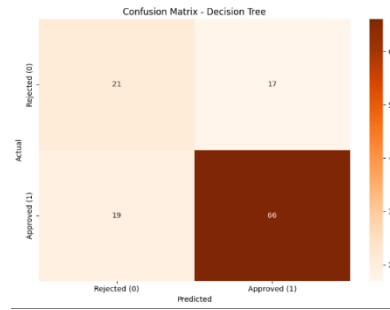
## Decision Tree

```

Decision Tree Evaluation:
Accuracy: 0.71
Classification Report:

```

	precision	recall	f1-score	support
0	0.53	0.55	0.54	38
1	0.80	0.78	0.79	85
accuracy			0.71	123
macro avg	0.66	0.66	0.66	123
weighted avg	0.71	0.71	0.71	123



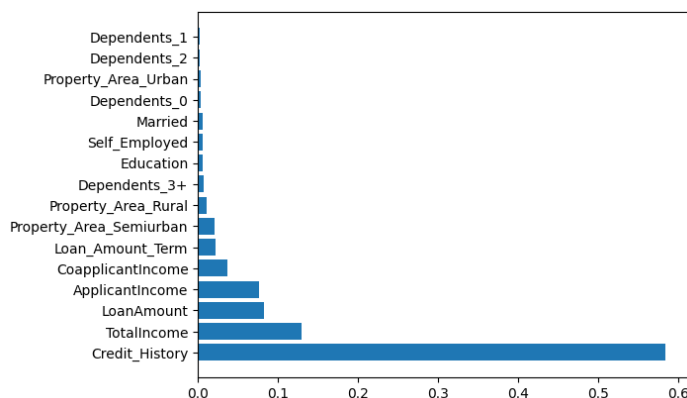
## 4. Model Selection and Training:

Choose appropriate machine learning algorithms (Random Forest, Decision Tree, Logistic Regression). Split the data into training and testing sets using ***train\_test\_split()***. Train the chosen models on the training data using ***model.fit()***.

- Tune hyperparameters to optimize model performance using techniques (***GridSearchCV***) or ***RandomizedSearchCV***.

## 6. Feature Importance Analysis:

- Analyze feature importance using ***model.feature\_importances\_*** to understand the model's decision-making process



## Task 2: AI Sales Agent (20 points)

### Tech Gadget Sales Agent

The Simple RAG Agent is a conversational AI sales agent designed to assist customers during their product journey. It leverages Retrieval-Augmented Generation (RAG) techniques to fetch relevant product information from a knowledge base and generate context-aware responses.

This project establishes a framework for building intelligent, responsive sales agents that enhance customer engagement through smart, data-driven conversations. It leverages the following components:

- **Retrieval-Augmented Generation (RAG):** The agent retrieves relevant information from a knowledge base (loaded and split into text chunks) using vector embeddings and similarity search (via FAISS) to provide context-aware responses.
- **LangChain Integration:** The chain is built using LangChain's modular components (like prompt templating, runnable chains, and output parsing) to process customer inputs and generate responses.
- **Sales-Focused Guidelines:** The system prompt is set up to guide the agent with a friendly and engaging tone, tailored to address common sales scenarios, product inquiries, and handle pricing or availability questions.
- **Conversation Memory:** It uses conversation history to maintain a seamless dialogue and context throughout the interaction.
- **Terminal Interface:** A simple command-line interface lets users engage with the agent interactively, making it easy to test and demo its functionality.

```
Agent: Hello! Welcome to our store! How can I assist you today?

You: what are the product do you have.?

Agent: We have a variety of products available in our store. Here are some of the categories we offer:

Feel free to let me know which category interests you, and I can provide you with more details on the specific products within that category.

You: i want to buy a laptop for gaming with above 8GB RAM and 15 inch above display

Agent: Great choice! For gaming with above 8GB RAM and a display size of 15 inches or above, here are a few laptop options you might consider:

1. Dell XPS 15
2. HP Spectre x360 14
3. Lenovo Legion Pro 7i
4. ASUS ROG Zephyrus G14
5. Acer Predator Helios 300
6. MSI Stealth 15
7. Alienware m18

These laptops offer a good balance of performance and display size for gaming. Do any of these options catch your interest, or would you like more details on a specific model?

You: yes i want details about MSI Stealth 15

Agent: The MSI Stealth 15 is a great choice for gaming enthusiasts looking for a powerful and portable gaming laptop. Here are the key specifications for the MSI Stealth 15:

- Brand: MSI
- Operating System: Windows 11
- Processor: Intel Core i7
- RAM: 16GB
- Storage: 1TB SSD
- Display Size: 15.6 inches
- Graphics Card: NVIDIA GeForce RTX 4060

This laptop offers a good balance of performance, storage, and graphics capabilities to enhance your gaming experience. The 15.6-inch display provides an immersive gaming experience.
```

## System Architecture

### 1. Knowledge Base Loading:

- Uses `TextLoader` to load the contents of the knowledge base text file.
- Applies a `CharacterTextSplitter` to break the document into manageable chunks.

### 2. Vector Store Creation:

- Generates document embeddings using `OpenAIEmbeddings`.
- Stores embeddings within a FAISS vector store to enable efficient similarity search.

### 3. Conversation Chain:

- Constructs a prompt using a system message outlining the role and guidelines for the sales agent.
- Utilizes a chain of operations where the context is retrieved from the vector store, applied to the prompt template and then processed by the ChatOpenAI language model.
- Integrates output parsing to produce a user-friendly answer.

### 4. Memory Management:

- Uses **`ConversationBufferMemory`** to retain conversation history, providing a seamless and coherent dialogue for repeated interactions.

## Scenario 1: Inquiring About a Specific Product

```
You: i want details about MSI Stealth 15

Agent: Certainly! Here are the details for the MSI Stealth 15 laptop:

- Brand: MSI
- Operating System: Windows 11
- Processor: Intel Core i7
- RAM: 16GB
- Storage: 1TB SSD
- Display Size: 15.6 inches
- Graphics Card: NVIDIA GeForce RTX 4060

The MSI Stealth 15 is a powerful gaming laptop with a sleek design and high-performance components. It offers smooth gaming experiences, fast loading times, and good multitasking capabilities with its 16GB RAM and 1TB SSD storage. The NVIDIA GeForce RTX 4060 graphics card ensures excellent graphics performance for gaming and content creation.

If you have any specific questions about the MSI Stealth 15 or need more details, feel free to ask!

You: 
```

## Scenario 2: Requesting Personalized Recommendations

```
You: I'm not sure what smartphone to buy. Can you recommend something that offers good performance and battery life for daily use?

Agent: For good performance and battery life for daily use, I recommend considering the following smartphones:

1. Google Pixel 8a: Offers a balance of performance and battery life with 8GB RAM and a 4500mAh battery.
2. Samsung Galaxy A54: A reliable choice with good performance and a 5000mAh battery for long-lasting use.
3. OnePlus Nord 3: Known for its smooth performance and 5000mAh battery capacity.
4. iPhone SE (3rd Gen): Provides excellent performance and optimized battery life for daily tasks.
5. Xiaomi Redmi Note 12: Offers a good balance of performance and battery life with a 5000mAh battery.

These options cater to different preferences and price ranges. Feel free to let me know if you have any specific requirements or preferences so I can provide a more tailored recommendation.

You: 
```

### Scenario 3: Addressing Price Objections

Agent: The Apple iPhone 15 Pro Max is a premium flagship smartphone known for its exceptional features and performance. Here are a few reasons why it's worth the extra investment compared to other models:

1. **Advanced Technology**: The iPhone 15 Pro Max is equipped with the latest technology and features, ensuring top-of-the-line performance and user experience.
2. **High-Quality Camera**: The iPhone 15 Pro Max offers a sophisticated camera system with advanced photography and videography capabilities, allowing you to capture stunning images and videos.
3. **Superior Build Quality**: Apple devices are renowned for their premium build quality and durability, ensuring longevity and a premium feel in hand.
4. **Long-Term Software Support**: Apple provides long-term software support for its devices, including regular updates and new features, ensuring a smooth and secure user experience for years to come.
5. **Resale Value**: Apple devices generally retain their value well, making them a good long-term investment if you plan to upgrade your phone in the future.

While the initial investment may be higher, the Apple iPhone 15 Pro Max offers a premium experience with top-notch features, performance, and reliability. If you value quality, performance, and the Apple ecosystem, it could be a worthwhile investment.

You:

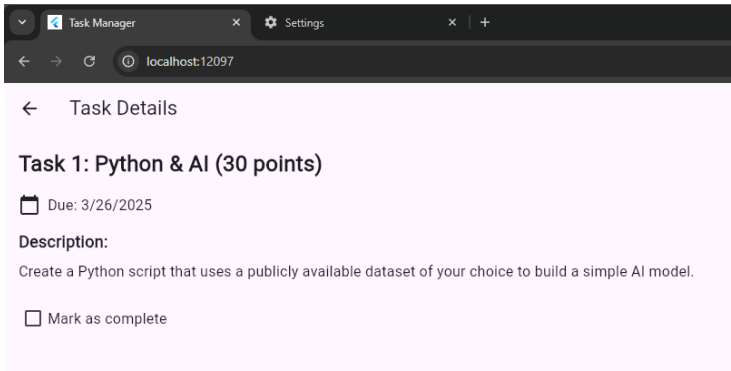
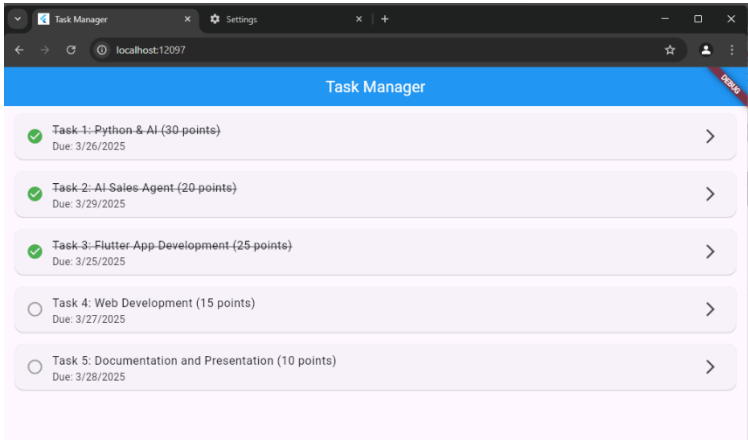
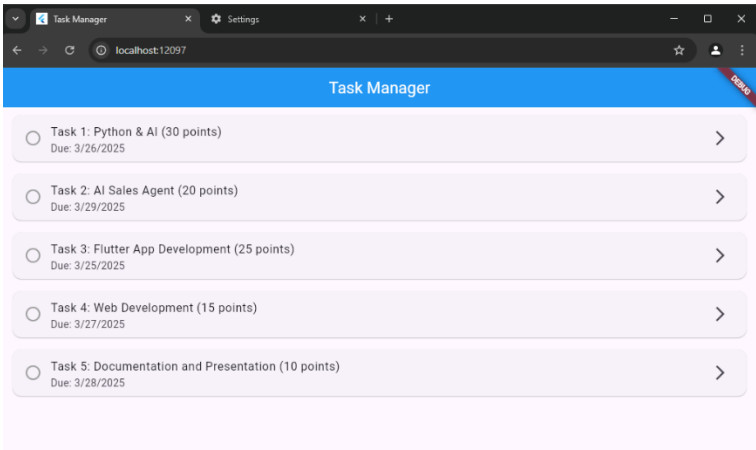
### Task 3: Flutter App Development (25 points)

#### Task Manager

This Flutter application implements a task manager with a clear separation of concerns, adhering to standard project architecture. The code is organized into distinct widgets, screens, and utility modules, facilitating efficient development and future updates.

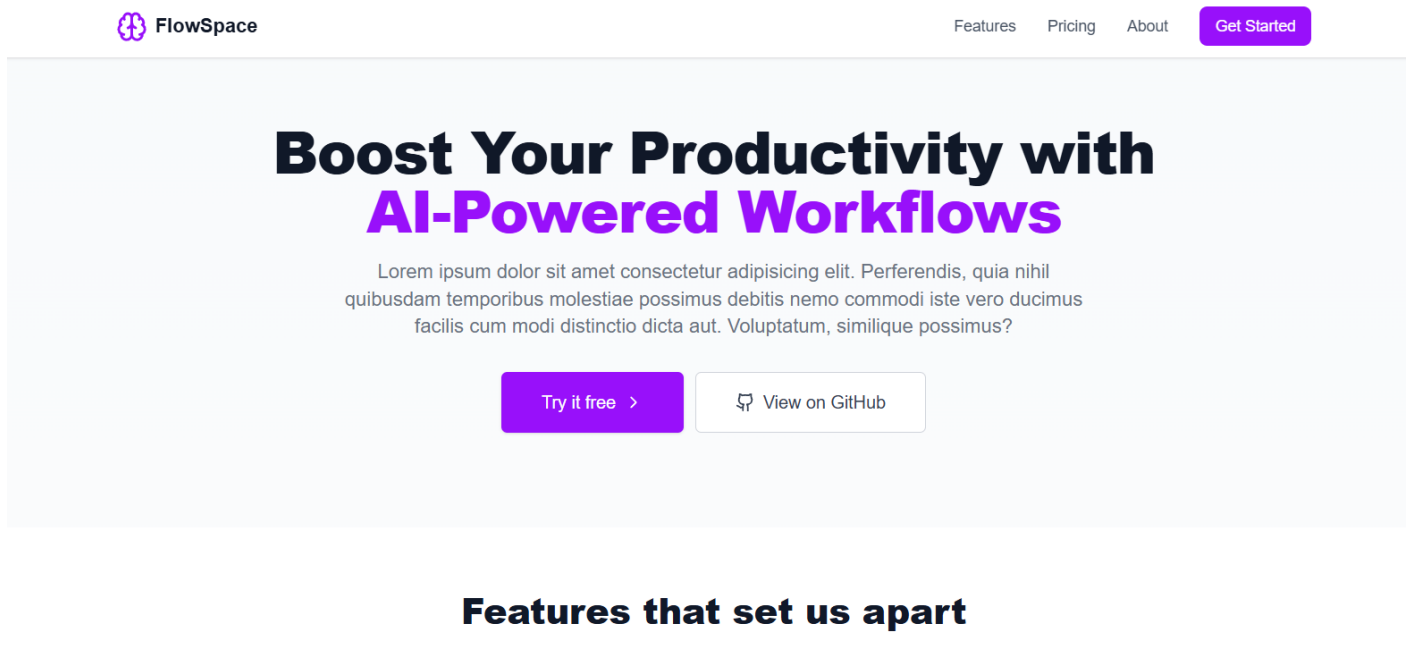
##### Core Functionality:

- Task Management:**  
The app allows users to view a list of tasks. Each task has attributes like title, description, due date, and completion status. Users can mark tasks as completed. Users can view detailed information of each task.
- State Management:**  
The application's state management is handled by the Provider package, with the TaskProvider class serving as the central repository for the task list. This class provides methods to access and modify task data, ensuring that any changes made to the list are consistently reflected across all screens within the application.
- Navigation**  
The application features two primary screens: HomeScreen, which presents a list of tasks, and DetailsScreen, which displays detailed information for a selected task. Navigation between these screens is facilitated using Navigator.push, enabling a seamless user experience.
- User Interface:**  
The application offers a clean and user-friendly interface, presenting tasks in a straightforward list format with clear visual cues to indicate completed items. Detailed task information is then displayed in an organized manner, ensuring ease of use and comprehension
- Date formatting:** The application utilizes a date formatter utility to format the date in a readable way.



## Task 4: Web Development (15 points)

### responsive landing page



This web page is a modern, responsive landing page built with React and styled using Tailwind CSS. Here's an overview of its key sections:

- **Navigation Bar:**  
A fixed header at the top features the brand logo (using the "Brain" icon from lucide-react) and links for "Features," "Pricing," and "About." It adapts to different screen sizes by displaying a mobile menu icon (toggle between Menu and X icons) on smaller viewports.
- **Hero Section:**  
Positioned below the navigation, this section introduces the product with a bold headline, "Boost Your Productivity with AI-Powered Workflows." A supporting paragraph provides a short description, and prominent call-to-action buttons ("Try it free" and "View on GitHub") invite users to engage further.
- **Features Section:**  
This part displays key product benefits in a grid layout. Each feature card includes an icon (such as Rocket, Clock, and Sparkles), a title, and a descriptive text that highlights how the product enhances productivity.
- **Call-to-Action (CTA) Section:**  
Featuring a vibrant background (purple), this section encourages visitors to start a free trial with a clear message and a "Get started" button.
- **Footer:**  
The footer organizes additional navigation into categories (Product, Company, Resources, Legal) with links to related pages and a copyright notice.



# Features that set us apart

Everything you need to maximize your productivity



## AI-Powered Automation

Lorem ipsum dolor sit amet consectetur adipiscing elit. Sequi quam voluptas praesentium, reprehenderit nemo suscipit odit tempore ex nam consequatur, doloribus corrupti nulla excepturi alias vero! Labore nam veritatis laborum!



## Time Tracking

Lorem ipsum dolor sit amet consectetur adipiscing elit. Expedita dignissimos beatae harum quis, labore dicta consequuntur earum ratione provident libero aperiam. Aperiam aliquam, eligendi ipsa voluptatum facere harum tempore rerum!



## Smart Suggestions

Lorem, ipsum dolor sit amet consectetur adipiscing elit. Aliquam officis quos veniam, similique porro accusantium ratione dicta impedit et natus nam quisquam aperiam voluptatem eos odit sapiente at. Ratione, dolorearque.

Ready to boost your productivity?  
Start your free trial today.

[Get started >](#)

PRODUCT

- [Features](#)
- [Pricing](#)
- [Updates](#)

COMPANY

- [About](#)
- [Blog](#)
- [Careers](#)

RESOURCES

- [Documentation](#)
- [Help Center](#)
- [API](#)

LEGAL

- [Privacy](#)
- [Terms](#)
- [Security](#)