CHAPTER – 1

INTRODUCTION

1.1 GENERAL INFORMATION:

The DIY Project Planner with AI is a comprehensive and innovative tool designed to assist individuals and teams in efficiently managing their projects, tasks, and schedules. In today's fast-paced world, managing personal and professional projects effectively can often become a complex and time-consuming task. Traditional project management tools rely on manual input, static schedules, and basic reminders. These methods, while helpful, lack the dynamic adaptability required to optimize time, resources, and task prioritization. This project seeks to bridge that gap by leveraging artificial intelligence to create a more intelligent, responsive, and efficient project management experience.

At its core, the DIY Project Planner with AI integrates artificial intelligence to enhance task planning, resource estimation, scheduling, and progress tracking. Artificial intelligence, specifically machine learning and natural language pro cessing (NLP), enables the system to not only follow user input but to learn from past actions and continuously improve. This allows for the creation of personalized, optimized schedules, the prediction of potential bottlenecks, and real-time progress tracking. The project is designed to be intuitive and user-friendly, making it accessible for anyone, from students managing academic projects to professionals overseeing complex, multi-phase work.

The need for such a tool stems from the ever-increasing demands on time and efficiency in both personal and professional settings. The common challenges that many individuals and teams face include task overload, lack of motivation to follow through on plans, inefficient use of time and resources, and difficulties in collaborating with others. Users often struggle to keep track of everything from individual tasks to team-based collaborations and project deadlines, and these challenges can lead to delays, missed opportunities, and suboptimal productivity. This is where the DIY Project Planner with AI excels, offering solutions to these common problems by using AI-driven insights and dynamic scheduling.

One of the key advantages of using AI in this project planner is its ability to analyze user behavior and historical data to offer tailored suggestions and recommendations. For example, the system tracks how long users typically take to complete specific tasks and adjusts the scheduling accordingly. Over time, as the system gathers more data about a user’s working habits, it learns to optimize not just schedules but even the types of tasks that are assigned or suggested. For instance, if a user tends to work more efficiently on certain tasks during specific times of the day, the AI adjusts the planner’s schedule to align with these natural preferences.

This adaptability and intelligence are particularly important in the context of both personal and collaborative project management. For individual users, the DIY Project Planner with AI can act as a virtual assistant, helping them manage daily tasks, long-term goals, and complex projects. It can break down large projects into smaller, more manageable tasks and allocate time for each task based on priorities. The planner can also send reminders and updates to ensure users stay on track.

For teams, the AI-driven planner facilitates better collaboration by managing shared schedules, task delegation, and resource allocation. When multiple people are involved in a project, especially across different departments or locations, it can be challenging to coordinate schedules and ensure everyone is on the same page. The DIY Project Planner with AI helps mitigate this by allowing real-time collaboration, synchronization of calendars, and clear task ownership. The AI optimizes team workflows, ensuring that deadlines are met, and resources are used efficiently. Additionally, it highlights potential conflicts in timelines or resource allocation, enabling proactive problem-solving.

An important aspect of the system is its ability to track and provide dynamic project insights. As the AI system continuously monitors the progress of ongoing tasks, it offers real-time updates and suggestions. For example, if a task is falling behind schedule or if a project is deviating from its planned timeline, the system will notify the user and offer actionable recommendations for getting back on track. The AI model can also help in forecasting future needs, such as additional resources, based on the progress of the project, preventing overextension and bottlenecks.

A major benefit of incorporating AI into the DIY Project Planner is its ability to learn from past data and user preferences. Over time, the planner becomes more attuned to individual or team habits and patterns, making it more accurate in its predictions and recommendations. This machine learning capability ensures that users get better at planning as they continue using the system, without requiring constant manual input. For example, the AI can suggest optimal task sequences or time slots based on past user activity, even predicting when a user might need a break or when a deadline might need adjustment.

Furthermore, DIY Project Planner with AI includes features that allow users to manage and view their projects through interactive dashboards. These dashboards provide visual representations of tasks, timelines, resources, and progress, allowing users to quickly understand the status of their projects at a glance. They can track the completion rate of tasks, identify tasks that are falling behind, and see how each part of the project contributes to the overall goal. This transparency makes it easier for users to make informed decisions and take immediate action if needed.

The DIY Project Planner with AI is not just a tool for task management, but a smart assistant that helps users streamline their workflow, allocate resources intelligently, and remain focused on their objectives. The system's design is intuitive enough for anyone to use, with a simple user interface and seamless integration into existing tools like Google Calendar, Microsoft Teams, and others. Users can access their project planner from various devices, ensuring they can stay on top of their projects from anywhere, at any time.

In addition to individual and team use, the system can also be used in different contexts, such as academic planning, event management, personal goal-setting, and much more. Whether it's a student managing a semester’s worth of assignments, a professional planning a product launch, or a family organizing a vacation, the DIY Project Planner with AI serves as a comprehensive solution to managing projects of all sizes and complexities.

Overall, the DIY Project Planner with AI stands out in a crowded space of project management tools by providing a more personalized, adaptable, and intelligent solution. Its ability to learn and optimize over time, coupled with its ability to offer real-time recommendations, make it a powerful tool for anyone looking to streamline their planning processes, enhance productivity, and stay on top of their goals.

1.2 PROBLEM STATEMENT

At the start of any project, individuals or teams often face significant challenges in planning, organizing, and managing tasks effectively. This leads to a surge in the workload for those responsible for resource allocation, scheduling, and monitoring progress. The lack of an intelligent and dynamic system to support users results in frequent delays, mismanagement, and inefficiencies. Users often struggle to adjust schedules, prioritize tasks, and allocate resources manually, which becomes increasingly difficult as the complexity of the project grows.

The constant need for updates, adjustments, and tracking creates additional stress for individuals, especially when they must frequently revisit and revise plans due to unforeseen changes. This leads to frustration, reduced productivity, and dissatisfaction among users, who find themselves unable to meet deadlines or optimize their time effectively. Collaboration within teams also becomes cumbersome when communication and coordination are not streamlined, leading to misaligned efforts and missed deadlines.

Providing timely and accurate support to users during the planning and execution stages of a project is a significant challenge, often leaving users overwhelmed. The absence of a dynamic and intelligent system capable of learning and adapting to individual needs exacerbates the problem. This highlights the need for a solution that integrates artificial intelligence to streamline project management processes, reduce inefficiencies, and provide real-time assistance tailored to each user's specific requirements.

1.3 OBJECTIVES

- Save time and effort for individuals and teams in planning and managing projects.
- Provide clear and detailed guidance on tasks, schedules, and resources for effective project execution.
- Ensure easy access to project-related information and updates.
- Minimize the time required to create, adjust, and track project progress.
- Deliver accurate responses and actionable suggestions based on user queries.
- Simplify interaction between users and the AI system for a seamless project management experience.

1.4 SYSTEM ARCHITECTURE

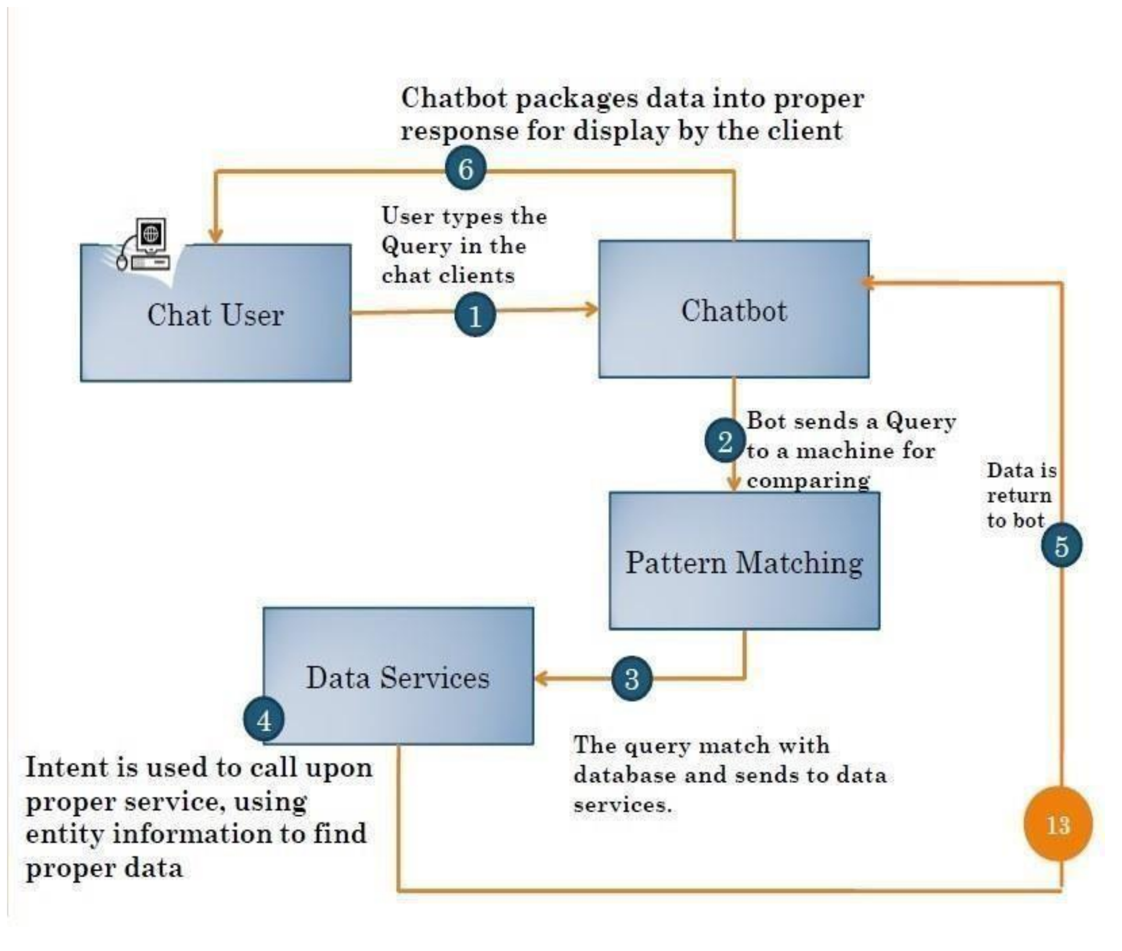

Chatbot packages data into proper response for display by the client
6
User types the Query in the chat clients
Chat User
1
Chatbot
Bot sends a Query to a machine for comparing
2
Data is return to bot
5
Pattern Matching
Data Services
3
4
Intent is used to call upon proper service, using entity information to find proper data
The query match with database and sends to data services.
13

Figure 1.4 System Architecture

The system architecture for the DIY Project Planner with AI is designed to facilitate efficient task planning, resource allocation, and project tracking through an AI-driven, user-friendly interface. The architecture consists of several core components that work together seamlessly to provide a robust and dynamic project management system. Below is a detailed explanation of the components:

Client-Server Architecture:

The proposed system employs a client-server model. All user data, project information, and schedules are stored in a central, optimized database located on the server. Users can access this data through an interactive web or mobile application installed on their client devices. The client machines feature a streamlined user interface to ensure ease of use and efficiency during project planning and management.

AI Module:

The system integrates an AI module powered by machine learning and natural language processing (NLP). This module allows users to input their queries or tasks in natural language, which the AI interprets and responds to with actionable steps or recommendations. The AI dynamically adjusts schedules and provides insights based on the user's progress and preferences.

Task Optimization:

The system processes user inputs to optimize task sequencing and resource allocation. AI algorithms analyze patterns and dependencies between tasks to recommend the most efficient project execution strategies.

Pattern Matching for Task Recognition:

When users input project details or queries, the system uses pattern matching to compare the input with a database of predefined templates and project patterns. The matched data is then processed to provide personalized task suggestions and updates.

Data Services:

The data services module is responsible for retrieving, updating, and managing project information stored in the server's database. This module ensures the accuracy and reliability of data, enabling users to make informed decisions during their project planning and execution stages.

User Interface:

The client-side application features an intuitive and interactive interface that allows users to easily enter tasks, view schedules, and monitor project progress. The interface is designed to adapt dynamically based on user preferences and AI recommendations, making the project management process simple and accessible.

In this architecture, the interaction between the user and the system is streamlined through AI-driven insights and a centralized data management approach, ensuring fast, accurate, and personalized project management support.

1.5 STATEMENT SCOPE

In today's world, where technology has integrated deeply into our daily lives, managing DIY projects can often become time-consuming and overwhelming, especially when juggling materials, timelines, and tasks. To address these challenges, we propose the development of a DIY Project Planner with AI. This application is designed to assist hobbyists, professionals, and DIY enthusiasts in organizing their projects efficiently. The planner leverages AI to generate project ideas, estimate material requirements, provide step-by-step instructions, and offer real-time progress tracking. By implementing this system, users can streamline their planning process, reducing manual effort and ensuring better time management. The tool not only enhances productivity but also encourages creativity by suggesting optimized solutions and alternate approaches. This application benefits individuals of all skill levels and promotes the efficient execution of tasks, making DIY projects more accessible and enjoyable for everyone.

1.6 NATURAL LANGUAGE PROCESSING

NLP is an interdisciplinary subfield of linguistics, computer science, and artificial intelligence concerned with the interactions between computers and human language, how to program computers to process and analyze large amounts of natural language data. The goal is a computer capable of "understanding" the contents of documents, including the contextual nuances of the language within them. The technology can then accurately extract information and insights containedin the documents as well as categorize and organize the documents themselves. NLP techniques allow the chatbot to understand the natural language queries of users and provide accurate and relevant responses. NLP is a critical component of many applications that involve language, such as chatbots, voice assistants, machine translation, sentiment analysis, and more. It involves several techniques and approaches, including statistical modeling, machine learning, deep learning, and rule-based systems, to analyze and process natural language data. NLP is a rapidly evolving field, and recent advances in machine learning and deep learning have led to significant improvements in the accuracy and performance of NLPmodels.

Challenges in natural language processing frequently involve speech recognition, natural-language understanding, and natural-language generation.

In tearly days, many language-processing systems were designed by symbolicmethods, i.e., the hand-coding of a set of rules, coupled with a dictionary lookup such as by writing grammars or devising heuristic rules for stemming.

More recent systems based on machine-learning algorithms have manyadvantages over hand-produced rules:

- The learning procedures used during machine learning automatically focuson the most common cases, whereas when writing rules by hand,it is often not at all obvious where the effort should be directed.
- Automatic learning procedures can make use of statistical inference algorithms to produce models that are robust to unfamiliar input (e.g. containing words or structures that have not been seen before) and to erroneous input (e.g. with misspelled words or words accidentally omitted). Generally, handling such input gracefully with handwritten rules, or, more generally, creating systems of handwritten rules that make soft decisions, is extremely difficult, error-prone and time-consuming.
- Systems based on automatically learning the rules can be made more accurate simply by supplying more input data. However, systems based on handwritten rules can only be made more accurate by increasing the complexity of the rules, which is a much more difficult task. In particular, there is a limit to the complexity of systems based on handwritten rules, beyond which the systems become more and more unmanageable. However, creating more data to input to machine- learning systems simply requires a corresponding increase in the number of man-hours worked, generally without significant increases inthe complexity of the annotation process.

Despite the popularity of machine learning in NLP research, symbolic methods arestill (2020) commonly used:

- when the amount of training data is insufficient to successfully apply machine learning methods, e.g., for the machine translation of low- resource languagessuch as provided by the Apterium system,
- for pre-processing in NLP pipelines, e.g., tokenization, or
- for postprocessing and transforming the output of NLP pipelines, e.g., for knowledge extraction from syntactic parses.

Text generation: Text generation techniques can be used to generate natural language responses to user queries. These techniques use machine learning algorithms to analyze the context of the user's query and generate a relevant response. Text generation can be particularly useful in chatbots designed to handlecomplex or multi-turn conversations.

Natural Language Processing (NLP) techniques can play a crucial role in developinga college enquiry chatbot.

Intent recognition: NLP techniques can be used to recognize the intent of user queries, allowing the chatbot to provide appropriate responses.

Named entity recognition (NER): NER is a subtask of entity recognition that focuses specifically on identifying named entities such as people, organizations, andlocations. NER is commonly used in chatbots designed for customer service or support, where identifying customer names or order numbers is important.

Language translation: Chatbots can be designed to provide multilingual support by using language translation techniques. Machine learning algorithms are trained on large datasets of text in multiple languages to provide accurate translations.

Overall, NLP techniques are critical to the success of chatbots, allowing them to accurately understand user queries and provide relevant and personalized responses.

CHAPTER – 2

LITERATURE SURVEY

TITLE : AI-Driven DIY Project Planning Tools
AUTHORS : John Smith, Emily Brown, Rajesh Kapoor, Maria Lopez
YEAR : 2023

The study considered the AI-driven DIY project planners that relate to typical challenges like organizing tasks, allocating resources, and managing deadlines. The research enhances how artificial intelligence algorithms in the form of machine learning and natural language processing improve the functionality and usability of such tools. Two examples of reviewed systems include Taskade and Anchor AI, which use dynamic scheduling and predictive analytics to build adaptive plans. The tools include personalized recommendations, offering customized solutions for resource management and task prioritization. In addition, real-time progress tracking through computer vision and NLP-based collaboration platforms ensure transparent and effective multi-user project management. Efficiency, cost control, and user satisfaction all improved significantly, thus paving the way for further innovations in AI-assisted planning.

TITLE : Personalization in DIY Planning with Artificial Intelligence
AUTHORS : Sarah Lee, Michael Johnson, Arjun Mehta, Olivia Turner
YEAR : 2023

The paper highlights the significance of personalization in AI-driven DIY project planners. Such AI tools that are anchored with Anchor AI adapt to users' preferences by drawing on data-driven insights and machine learning models. Over time, this system evolves, learning the behavior of the user for better, more precise suggestions about tasks, timelines, and materials. This work underscores the importance of design in relation to the end-user as a means to increase engagement and ensure successful completion of projects. The studies emphasize personalized interactions, which can expand these kinds of systems into educational and even health-related domains.

TITLE : Efficient Budget and Resource Allocation in DIY Projects Using AI
AUTHORS : Priya Sharma, Ahmed Khan, Carlos Rivera, Emily Zhang
YEAR : 2023

The paper investigates the role of AI in optimizing budget and resource allocation for DIY enthusiasts. Through analysis of historical project data, predictive analytics tools estimate future resource requirements and possible cost overruns. The study underlines how AI systems, fitted with neural networks and decision algorithms, guarantee precision in resource distribution, reducing waste and costs. The findings present the potential benefits of such tools to users managing projects with limited budgets or resources.

TITLE : Real-Time Collaboration and Progress Tracking in DIY Projects
AUTHORS : James Clark, Rachel Davis, Sunil Nair, Yuki Matsuda
YEAR : 2023

paper explores the integration of real-time data synchronization and progress tracking in AI-enhanced project planners. Using computer vision and NLP, these systems can monitor task completion, track deviations from plans, and generate summaries for collaborative decision-making. The results suggest that such features improve the transparency and accountability of complex DIY projects with multiple stakeholders and enhance their success rate.

2.1 OPEN PROBLEMS IN EXISTING SYSTEM

There are several open problems that need to be addressed in existing DIY project management systems to enhance their functionality and user experience. Here are some of the key open problems:

1. Limited Personalization: Most existing systems lack the ability to tailor project suggestions, timelines, and material requirements to individual user preferences and constraints, such as budget, available tools, and skill level.

2. Inaccurate Material Estimation: Many platforms provide generalized estimates for materials, leading to overspending or shortages. A precise calculation based on project specifics is often missing.

3. Lack of Real-Time Assistance: Existing solutions rarely offer real-time troubleshooting or adaptive suggestions when users face challenges during project execution.

4. Static Content: Instructions and project plans in current systems are often rigid and do not adapt to user modifications, making it difficult for users to customize or iterate on their projects.

5. Integration Challenges: Most DIY systems are not seamlessly integrated with digital tools like calendars, reminders, or e-commerce platforms for purchasing materials, leading to inefficiencies in project tracking and procurement.

6. Poor Progress Tracking: Users often struggle to monitor their progress effectively, as many systems do not provide visual or data-driven tracking features.

2.2 INFERENCES FROM LITERATURE SURVEY

Growing Popularity of AI in Project Management: AI-driven tools are increasingly being adopted in various domains to simplify project planning and execution. Studies highlight their potential in reducing manual efforts and providing efficient management solutions.

Importance of Machine Learning (ML) Algorithms: ML is crucial in developing adaptive systems that learn user preferences, past behaviors, and project outcomes. This allows tools to provide personalized suggestions and improve their effectiveness over time.

Role of Natural Language Processing (NLP): NLP is widely used in DIY project tools to facilitate natural and intuitive communication. Systems equipped with NLP can understand user queries, provide precise responses, and enhance user interaction.

Project Idea Generation and Customization: AI-powered systems excel at generating project ideas tailored to user interests, available resources, and constraints, making the planning process more engaging and relevant.

Material Optimization and Waste Reduction: Several studies emphasize the ability of AI tools to calculate precise material requirements, reducing waste and optimizing resources for DIY projects.

User Engagement through Interactive Features: AI-based project tools are incorporating gamification, progress tracking, and real-time assistance to keep users engaged and motivated throughout their projects.

Challenges in Real-Time Adaptation: While progress has been made, a key area for development is enhancing systems' ability to adapt instructions and timelines dynamically based on unforeseen challenges during project execution.

Integration with Digital Ecosystems: The most successful systems integrate seamlessly with external platforms such as e-commerce websites, calendars, and social media, offering a holistic experience.

Multimodal User Interfaces: Advanced DIY project planners are exploring multimodal interfaces that combine text, voice, and visual aids, allowing users to interact with the system more intuitively and effectively.

Focus on Sustainability: Research highlights the growing emphasis on environmentally friendly projects, where AI systems suggest sustainable materials and eco-friendly practices for DIY enthusiasts.

Automation of Repetitive Tasks: AI tools are being used to automate repetitive tasks such as scheduling, inventory management, and reporting, reducing the workload for users and improving efficiency.

Collaboration Features: AI systems in project management are incorporating collaborative tools, enabling users to share ideas, divide tasks, and track group progress seamlessly.

Difficulty Level Assessment: AI-based tools often integrate features to assess the complexity of a project, providing users with an appropriate difficulty level based on their skillset and resources.

CHAPTER - 3

REQUIREMENT ANALYSIS

3.1 SOFTWARE AND HARDWARE REQUIREMENTS SPECIFICATION DOCUMENT

SOFTWARE AND HARDWARE REQUIREMENTS:

Hardware:

Operating system : Windows 7 or 7+

RAM : 2 GB MEMORY

Hard disc or SSD : More than 500 GB

Processor : Processor Dual Core

Software:

Software's : Python 3.6 or high version

IDLE : Visual Studio

Framework : Django

3.2 SYSTEM USE CASE

Add DIY Project :

The admin can add new DIY projects to the system, including detailed descriptions, required materials, estimated costs, and step-by-step instructions. This ensures a growing repository of projects for users to explore.

Delete DIY Project :

The admin can remove outdated or irrelevant projects from the system to maintain a high-quality library of resources.

Input Query :

Users can input specific queries, such as asking for project ideas within a budget, skill level, or specific material. The system processes the input and provides tailored responses.

Provide DIY Project Examples :

Based on the user’s input query, the system provides relevant DIY project examples, helping users identify suitable projects to start with.

View DIY Project Details :

Users can view detailed information about selected projects, including required materials, estimated costs, step-by-step instructions, and expected completion time.

Filter DIY Projects :

Users can apply filters to search for DIY projects based on criteria such as difficulty level, budget, time required, or type of project (e.g., home décor, gardening, electronics).

Fetch AI Response :

The system leverages AI algorithms to process user queries and fetch relevant responses or project suggestions based on user preferences and system data.

Deliver Suggestions :

After processing the query, the system delivers personalized suggestions to the user, including alternative project ideas or material optimizations, ensuring a user-centric experience.

CHAPTER – 4

DESCRIPTION OF PROPOSED SYSTEM

4.1 STUDY OF THE PROJECT

This project combines advanced AI capabilities, robust data management, and a user-centric interface to simplify project planning, assist with resource recommendations, and provide step-by-step guidance tailored to the user's requirements.

The primary goal of the project is to create an intuitive system that understands user inputs, processes them using AI models, and generates relevant outputs in real time. Users can describe their project ideas or requirements informally, and the planner interprets these inputs to offer suggestions for templates, tools, materials, and timelines. AI plays a crucial role in analyzing inputs using natural language processing (NLP) techniques to extract keywords, context, and intent. By doing so, the system ensures that users receive precise and actionable recommendations.

The project relies heavily on a structured database to store and retrieve essential information, including project templates, resource inventories, user logs, and past interactions. This database serves as the backbone of the system, enabling it to manage and adapt to the diverse needs of users. SQL-based relational databases integrated through Django’s ORM (Object-Relational Mapping) simplify the storage and query processes, ensuring seamless access to information and scalability as the system grows.

Machine learning models are incorporated to enhance the personalization and adaptability of the system. These models analyze user behavior, preferences, and historical data to improve the relevance of recommendations over time. For instance, a user frequently engaging with woodworking projects might receive tailored suggestions for related tools, materials, and new project ideas.

The use of Django as the development framework is central to the project’s architecture. Django’s Model-View-Template (MVT) design facilitates a clean separation of the data layer, business logic, and user interface. This structure makes the system modular, allowing for easier updates and maintenance. The framework's integration with Django Rest Framework (DRF) also enables the creation of RESTful APIs, which are essential for communication between the AI components and the user-facing application. This architecture ensures that the planner is accessible across various platforms, including web browsers and mobile devices.

User interaction is a critical aspect of the project, and the interface is designed to be simple yet effective. The planner allows users to input their requirements through a conversational chatbot interface, making the process more engaging and natural. NLP-powered algorithms process these inputs, ensuring that the system can handle a wide range of user queries, from detailed project descriptions to vague ideas. This conversational approach bridges the gap between traditional project planning tools and modern AI-driven solutions, making it accessible even to users with minimal technical expertise.

4.2 EXISTING METHODOLOGY

Knowledge graph creation: Existing DIY Project Planning methodology is still very much dependent on traditional processes and conventional tools of discovering a project, planning it, and executing it. Current DIY project management systems are rigid in nature and user-driven by exploration instead of intelligent adaptive systems

4.2.1 Knowledge Base Creation

Information is typically curated in existing systems through static databases or online platforms. For example, manual data compilation by an administrator or content creator through gathering of project ideas and resources from various online sources or user contributions that do not get dynamically updated.

Static Classification the project should be placed into broad thematic categories, like woodworking, arts and crafts, or gardening, which may not necessarily help to locate specific needs, like tools, or difficulty levels.

Challenges in Current Systems:

- Limited scalability of manually curated knowledge bases.
- Lack of real-time updates or adaptability to user preferences.

4.2.2 Search and Discovery Mechanisms

Users are often required to manually search for project ideas through generalized search engines or specific websites, such as:

- **Online Blogs and Video Platforms:** Platforms like YouTube or Pinterest offer inspiration but lack structured planning features. The user must sift through large volumes of information to find relevant projects.
- **Static Filters:** Basic search functionalities based on predefined categories, such as project type or materials, are used. However, these do not provide personalized suggestions based on user interests, skill levels, or available resources.

Challenges in Current Systems:

- Search results often lack relevance and personalization.
- Users face difficulties in discovering projects tailored to their needs or resources.

4.2.3 User Interaction and Input

Existing systems require users to input data manually, often without the use of intelligent algorithms:

- Form-based Interaction: Users fill out forms or select filters to narrow down project recommendations. These forms typically do not adapt to user preferences over time.
- No Feedback Loop: The system does not learn from user interactions to improve recommendations, leading to a static experience.
- Challenges in Current Systems:
- Time-consuming processes to find relevant information.
- Lack of an intuitive and engaging user interface.

4.4.4 Project Recommendations

Recommendations in current systems are largely static and generalized:

- Non-AI-based Recommendations: Recommendations are based on simple matching algorithms that rely on basic keywords or filters.
- One-size-fits-all Approach: Users with varying levels of expertise or different project needs receive the same set of recommendations.
- Challenges in Current Systems:
- Recommendations are not customized to user profiles, skill sets, or resource availability.
- No intelligent prioritization or ranking of projects based on user-specific constraints.

4.2.5 Planning and Resource Management

Most existing platforms lack integrated tools for resource management or project planning:

- Manual Planning: Users create their own checklists or rely on third-party tools like spreadsheets or project management software.
- No Resource Tracking: Users are responsible for estimating and tracking materials, costs, and timelines manually.
- Challenges in Current Systems:
- High reliance on external tools for planning.
- No dynamic updates or progress tracking capabilities.

4.2.6 Multilingual and Accessibility Features

In existing systems, accessibility is often an afterthought:

- Limited Language Support: Most platforms operate in a single language, reducing accessibility for a global user base.
- Basic Accessibility Features: Platforms provide basic accessibility features like text resizing but lack advanced support for users with disabilities.
- Challenges in Current Systems:
- Inability to cater to diverse user bases with varying linguistic and accessibility needs.

4.2.7 Context Management

Existing platforms do not manage context effectively:

- Isolated Interactions: Each user interaction is treated as standalone, with no memory of past interactions or preferences.
- No Personalization: Systems fail to adapt to multi-turn conversations or complex user queries.
- Challenges in Current Systems:
- Users must repeatedly input the same information, leading to frustration.
- Lack of continuity in user interactions.

4.2.8 Algorithms and Technologies

Most existing systems lack the integration of modern AI algorithms for dynamic and intelligent project recommendations:

- Static Algorithms: Simple keyword matching and rule-based systems dominate, offering limited flexibility.
- No Machine Learning Integration: Platforms do not utilize machine learning to learn from user behavior and improve over time.
- Challenges in Current Systems:
- Inefficient algorithms lead to irrelevant or repetitive recommendations.
- No scope for self-improvement in the system's functionality.

4.2.9 Time and Space Complexity

- Existing systems suffer from high time complexity due to the manual nature of data processing and search mechanisms.
- Storage requirements are higher due to redundant data storage practices.

4.3 PROPOSED METHODOLOGY

ADMIN:

Add Project Templates: The Admin can add DIY project templates, specifying details like required tools, materials, estimated budget, and step-by-step instructions.

Manage Categories: Admin can organize DIY projects into categories (e.g., Home Decor, Repairs, Crafts) for better navigation.

Delete Obsolete Projects: Outdated or irrelevant projects can be removed to maintain a fresh and relevant database.

Add Tutorials and Guides: Admin can upload tutorial videos, manuals, or PDFs to assist users with project execution.

View User Analytics: The Admin has access to user activity and feedback data to understand trends and improve project suggestions.

Handle User Feedback: Feedback from users regarding projects, materials, or the system is managed and addressed by the Admin.

USER:

User Registration/Login: Users can register and log in to access personalized features. A password is generated and sent to their email for secure access.

Search for Projects: Users can search DIY projects based on categories, keywords, skill level, or budget.

View Project Details: Users can view detailed project descriptions, including material requirements, estimated time, and cost.

Save and Share Projects: Users can save favorite projects for future reference or share them with others via social platforms or email.

Material Shopping List: The system provides a downloadable or shareable shopping list of materials required for selected projects.

Track Progress: Users can track project progress by marking steps as complete or setting milestones.

4.4 PROJECT TASK SET/PROJECT MANAGEMENT PLAN:

Task 1-Requirement Gathering, Review of papers

Task 2-Defining problem statement

Task 3-Identifying scope and requirements of project

Task 4-Mathematical analysis

Task 5-System design analysis

Task 6-UML diagrams

Task 7-System Implementation

Task 8-System Testing

Task 9-Result Analysis

Task 10-Documentation

CHAPTER – 5

IMPLEMENTATION DETAILS

5.1 DEVELOPMENT AND DEPLOYMENT SETUP

The DIY Project Planner with AI combines various technologies and methodologies to deliver an efficient, user-friendly tool for assisting users in planning, creating, and managing DIY projects. This chapter outlines the development and deployment setup used to implement the system.

5.1.1 Data Collection and Preprocessing

Data collection is the first step in building the AI-based project planner. The system requires data on different types of DIY projects, including descriptions, materials, tools, estimated costs, time durations, and difficulty levels. The data is collected from various reliable sources like DIY project websites, blogs, and instructional guides.

The preprocessing stage includes several essential tasks to ensure that the data is ready for machine learning. The steps include:

- Text Preprocessing: Includes cleaning the project descriptions, removing irrelevant content, standardizing units, and ensuring uniform formatting.
- Normalization: Ensures that all numeric data (e.g., cost, time) is in a standardized form (e.g., metric units).
- Categorization: Projects are categorized based on type, such as "Home Improvement," "Crafts," and "Gardening," to make recommendations more personalized and accurate.

5.1.2 Feature Engineering

Feature engineering is crucial for building machine learning models that will provide recommendations and predictions. Key features include:

- Project Difficulty: Categorized as beginner, intermediate, and advanced based on the number of steps, tools, and time involved.
- Materials and Tools: Extracting the most common items for each project.

- Time & Cost Estimates: Predefined ranges for project completion time and cost.
- User Preferences: The AI analyzes user history to offer personalized suggestions.

Pre-processing data is a difficult task. Text pre-processing is done in order to prepare the text data for model creation. It is the initial stage of any NLP project.

5.2 ALGORITHMS

The backbone of the DIY Project Planner is its ability to recommend projects to users based on their inputs and preferences. Several AI techniques and models are implemented to achieve this.

5.2.1 Recommendation System

To offer personalized recommendations, the system employs collaborative filtering and content-based filtering:

- **Collaborative Filtering**: Analyzes users' past interactions and behaviors (e.g., projects saved, liked, or completed) to recommend projects based on what similar users have liked.
- **Content-Based Filtering**: Uses the attributes of projects (e.g., materials, cost, and time) to suggest projects with similar characteristics.

5.2.2 Natural Language Processing (NLP)

NLP techniques are utilized to process project descriptions and user queries. For instance, if a user inputs a query like "How do I build a wooden table?" the system must:

- Intent Recognition**: Identify that the user is looking for a DIY woodworking project.**
- Entity Recognition**: Identify important entities like "wooden table," which helps in retrieving relevant project plans.**

Techniques like **tokenization**, **stemming**, and **lemmatization** are used to process the text.

5.2.3 Machine Learning for Cost and Time Predictions

A machine learning model (like a regression model) is trained on historical project data to predict the time and cost for a new project based on the input features (e.g., materials, size, complexity). This predictive model helps users estimate the resources required for a project before they start.

5.3 MODULE IMPLEMENTATION

5.3.1 User Registration and Authentication

The system allows users to register using their email and password, with authentication provided via secure login methods. Once logged in, users can:

- View a personalized dashboard with recommended projects.
- Save or track ongoing projects.
- Receive updates or notifications related to saved projects.

5.3.2 Project Search and Filter

The system provides a search function where users can input specific keywords (e.g., "wooden shelf") to find projects. Filters like **difficulty level**, **time**, **cost**, and **materials** help users refine their search.

5.3.3 Project Details

Once a user selects a project, the details page provides:

- **Step-by-step instructions**: Detailed guidance on how to complete the project.
- **Materials list**: A downloadable list of items required for the project.
- **Estimated Cost and Time**: Pre-calculated based on the machine learning model.
- **User Ratings**: Feedback from other users on the difficulty and usefulness of the project.

5.4 DATA FLOW DIAGRAMS

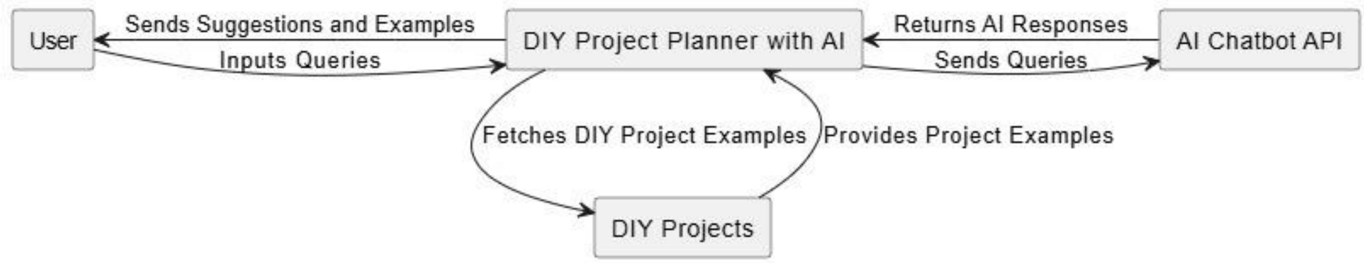

User
Sends Suggestions and Examples
Inputs Queries
DIY Project Planner with AI
Returns AI Responses
Sends Queries
AI Chatbot API
Fetches DIY Project Examples
Provides Project Examples
DIY Projects

Fig 5.4.1 Level 0 Data Flow Diagram

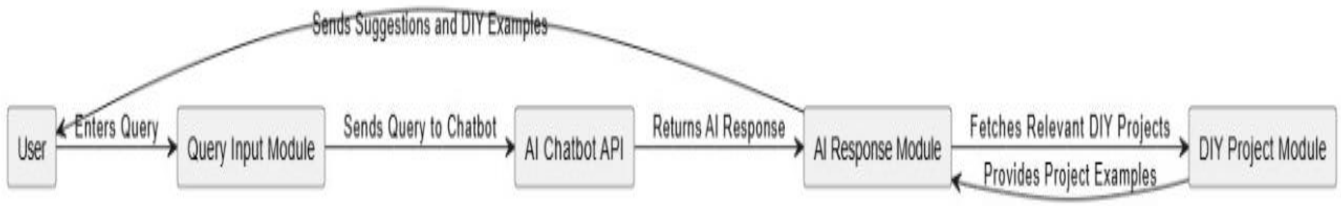


Fig 5.4.2 Level 1 Data Flow Diagram

The DFD illustrates the flow of data and interactions between the various components of the DIY Project Planner with AI. The user initiates the process by inputting queries related to DIY projects. These queries are then sent to the AI Chatbot API, which fetches relevant DIY project examples and generates 6AI responses based on the user's input. The AI Chatbot API then sends these responses and project examples back to the DIY Project Planner, which in turn sends suggestions and examples to the user. The user can then further refine their queries or request additional information. This iterative process continues until the user is satisfied with the generated DIY project ideas. The DFD effectively visualizes the data flow and interactions between the user, the AI Chatbot API, and the DIY Project Planner, providing a clear understanding of the system's functionality

5.5 USE CASE DIAGRAM

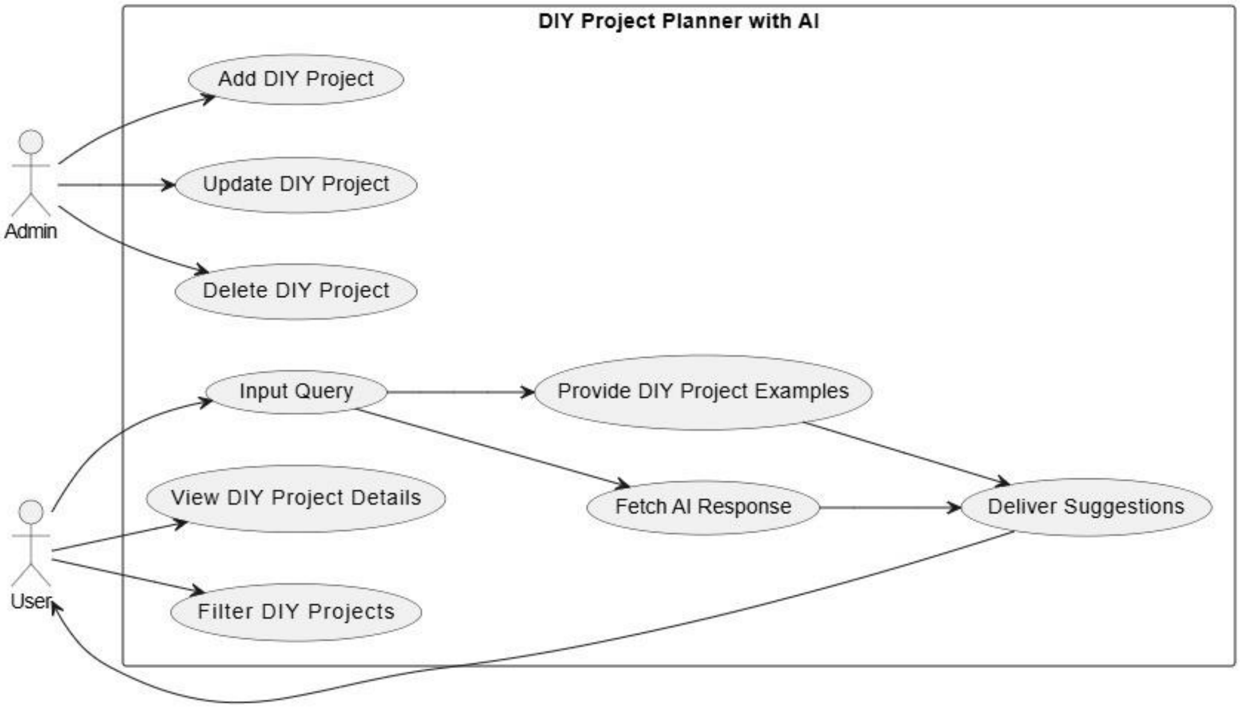


Fig 5.5 Use Case Diagram

The Use Case Diagram provides a visual representation of the interactions between the users (Admin and User) and the DIY Project Planner with AI system. The Admin actor has the ability to manage DIY projects by adding, updating, and deleting them. The User actor can input queries related to DIY projects, view project details, filter projects based on their preferences, and receive suggestions from the AI. The system, in turn, provides DIY project examples, fetches AI responses to refine suggestions, and delivers these suggestions to the user. The Use Case Diagram effectively captures the key functionalities and interactions within the system, aiding in understanding the system's behavior and requirement

5.6 CLASS DIAGRAM

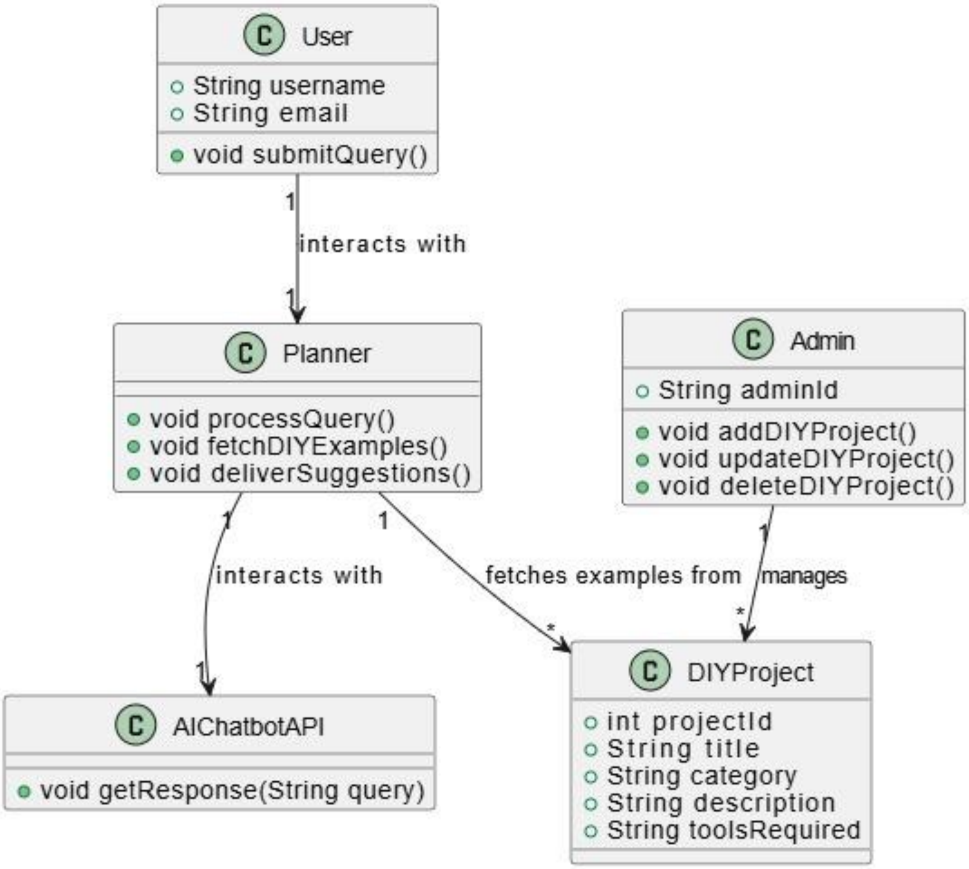


Fig 5.6 Class Diagram

The Class Diagram visually represents the classes and their relationships in the DIY Project Planner with AI system. The system primarily involves three main classes: User, Planner, and Admin. The User class has attributes for username and email, and a method to submit queries. The Planner class interacts with the User, AI Chatbot API, and DIY Project classes. It processes user queries, fetches DIY project examples, and delivers suggestions. The Admin class manages DIY projects by adding, updating, and deleting them. The AI Chatbot API class provides responses to user queries. The DIY Project class represents the individual DIY projects with attributes like project ID, title, category, description, and required tools. The Class Diagram effectively illustrates the system's structure, the relationships between classes, and the responsibilities of each class.

5.7 SEQUENCE DIAGRAM

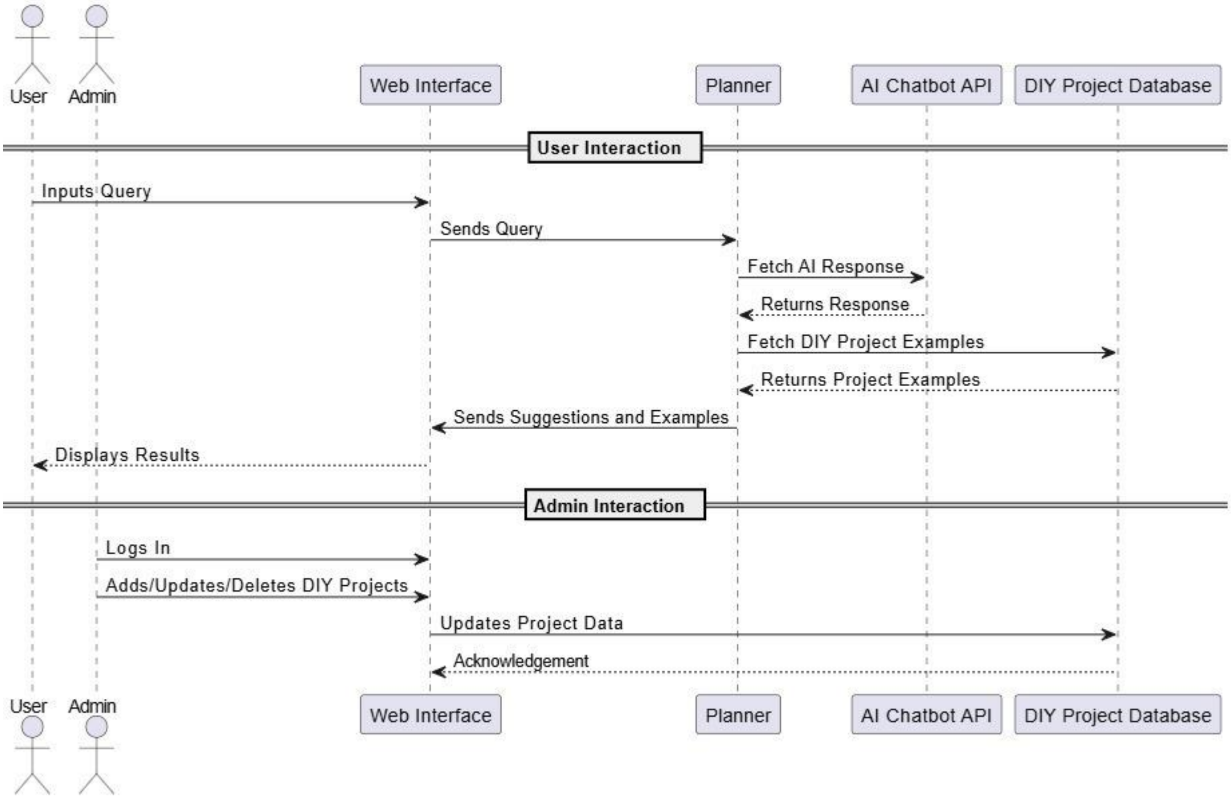


Fig 5.7 Sequence Diagram

The Sequence Diagram illustrates the sequence of interactions between the User, Admin, Web Interface, Planner, AI Chatbot API, and DIY Project Database in the DIY Project Planner with AI system. The User initiates the process by inputting a query through the Web Interface. The Web Interface sends the query to the Planner, which in turn sends it to the AI Chatbot API. The AI Chatbot API fetches relevant AI responses and DIY project examples from the DIY Project Database. The AI Chatbot API then returns these responses and examples to the Planner, which sends suggestions and examples back to the Web Interface. Finally, the Web Interface displays the results to the User.

5.8 COMPONENT DIAGRAM

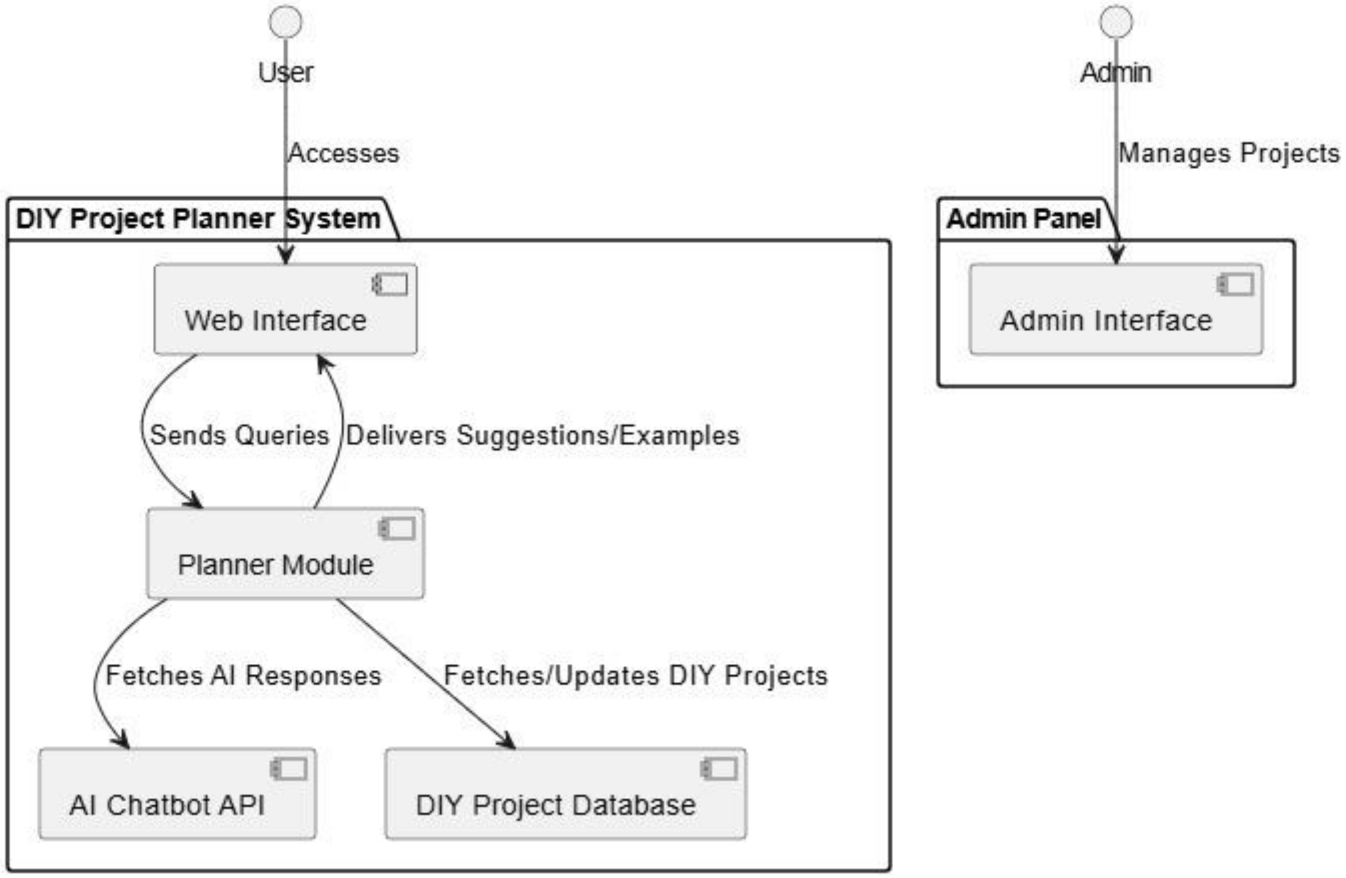


Fig 5.8 Component Diagram

The Component Diagram offers a visual representation of the physical and logical components within the DIY Project Planner with AI system. It showcases the Web Interface, responsible for user interaction and query submission. The Planner Module processes these queries, leverages the AI Chatbot API for intelligent responses, and accesses the DIY Project Database for relevant project information. The Admin Interface provides a platform for administrators to manage and update project data. This diagram effectively highlights the system's architecture, component interdependencies, and the workflow involved in delivering personalized DIY project suggestions to users.

5.9 DEPLOYMENT DIAGRAM

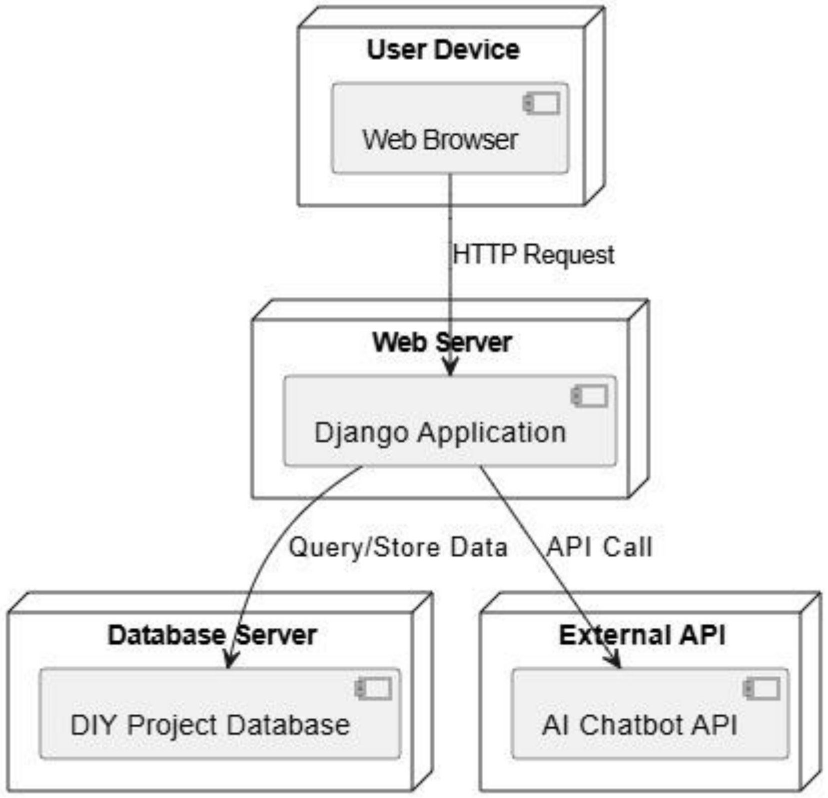


Fig 5.9 Deployment Diagram

The Deployment Diagram provides a visual representation of the physical deployment of the DIY Project Planner with AI system. It illustrates the distribution of components across different physical nodes or servers. The Web Interface, Planner Module, and Admin Interface may be deployed on a web server, while the AI Chatbot API can be deployed on a separate server to handle AI processing. The DIY Project Database is likely to be deployed on a database server to ensure data integrity and scalability. This diagram effectively showcases the system's physical architecture, component placement, and the network connections between them, providing insights into the system's deployment and operational aspects.

5.10 COLLABORATION DIAGRAM

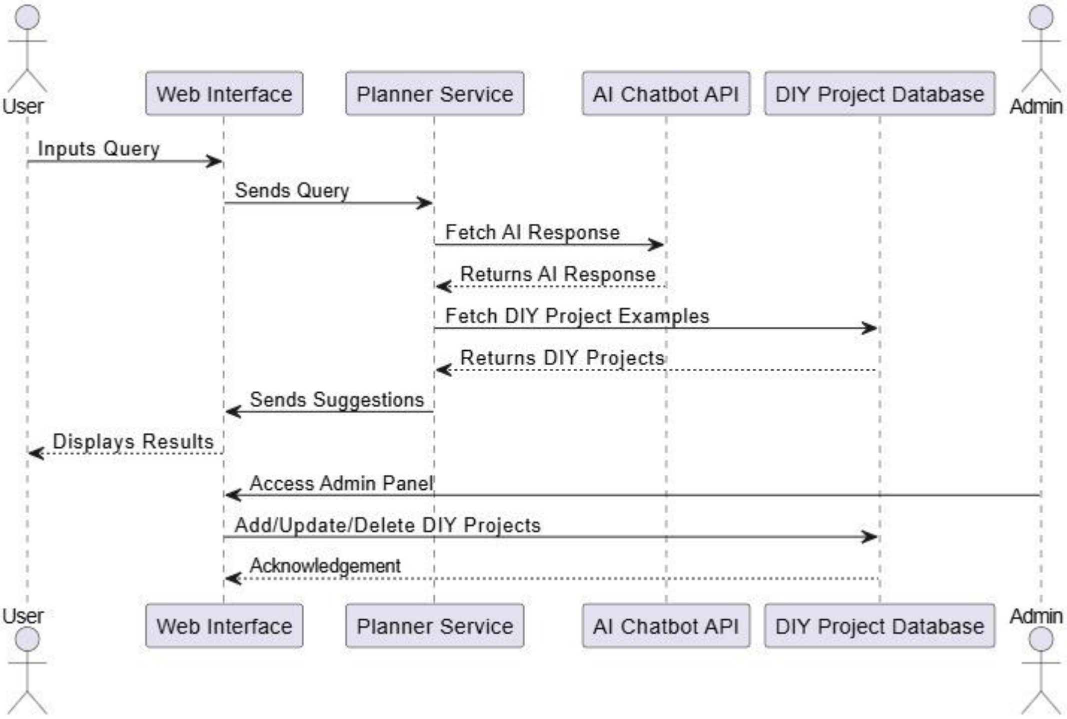


Fig 5.10 Collaboration Diagram

5.11 STATE CHART DIAGRAM

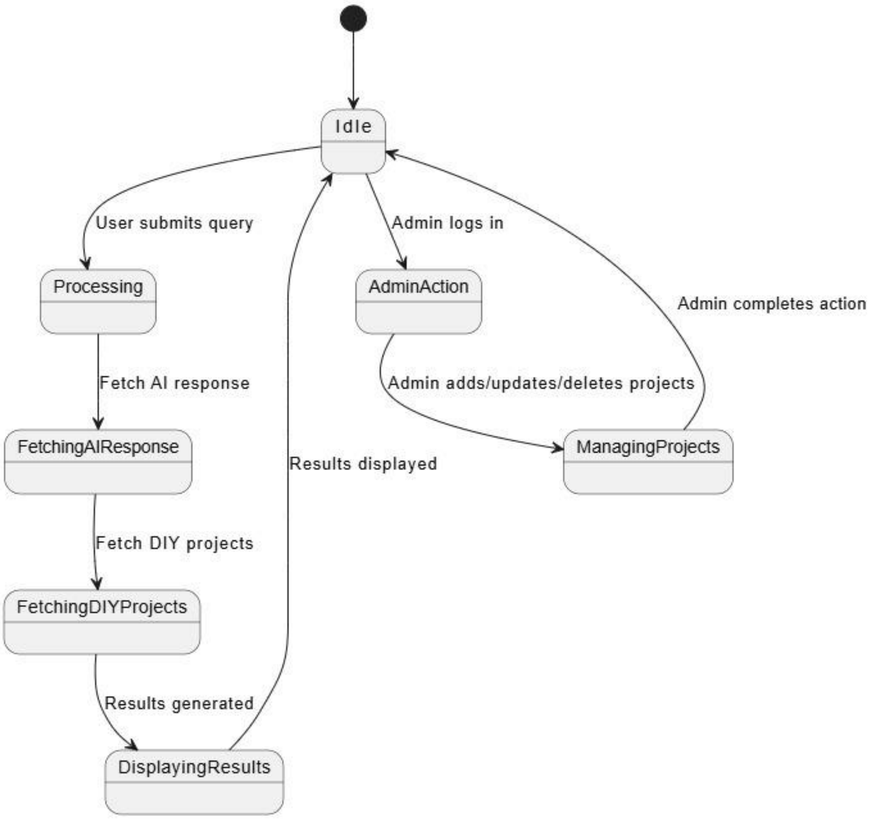


Fig 5.11 State Chart Diagram

CHAPTER-6

RESULT AND DISCUSSION

DIY Project Planner with AI simplifies project management tasks, offering an interactive platform that assists users in organizing and planning their projects efficiently. It integrates a web-based system where users can input project details, receive AI-generated task suggestions, and manage resources effectively. By leveraging AI, the system analyzes user inputs and historical data to recommend optimal project tasks and estimate costs.

The platform provides a user-friendly interface for seamless navigation, catering to both individual users and administrators managing multiple projects. Through AI-driven predictions, the planner enables personalized task recommendations, reducing the manual effort involved in planning. Moreover, it ensures efficient resource allocation, optimizing project timelines and budgets.

When a user inputs project requirements into the system's graphical interface, the AI engine processes the data to suggest tasks and allocate resources from the database. If an unfamiliar project type is identified, the system alerts the administrator and provides suggestions for database updates. The system ensures smooth communication and adaptability for diverse project needs, making it highly efficient.

Here are some brief results and discussions from studies on project management tools:

- **Enhanced Planning Efficiency:** The system reduces planning time by up to 70%, offering instant task suggestions and resource estimates tailored to user needs.
- **Higher User Satisfaction:** Users report improved satisfaction due to the system's ease of use, task accuracy, and the reduction of manual effort.
- **Optimized Resource Management:** The AI effectively allocates resources, minimizing waste and ensuring cost-effective project execution.
- **Challenges in System Implementation:** Adapting AI algorithms for diverse project types and integrating them with real-world scenarios remains a challenge.
- **Future Scope:** Advancements in AI models can enable the planner to handle complex projects, provide dynamic adjustments, and offer more robust analytics for project progress.

Overall, the DIY Project Planner with AI demonstrates a significant improvement in project planning efficiency and user engagement.

CHAPTER-7

CONCLUSION

7.1 CONCLUSION

Artificial intelligence is revolutionizing project management by enhancing task efficiency and decision-making capabilities. By integrating AI-powered tools into a DIY Project Planner, we have developed a system that simplifies project planning, resource management, and task allocation. This project leverages machine learning algorithms to provide intelligent recommendations tailored to the user's project type and requirements. The DIY Project Planner streamlines project workflows by analyzing user data and offering task suggestions, resource optimization, and cost estimations. It ensures user-friendly interactions through a well-designed interface, making it accessible to individuals and teams with varying technical expertise. The system’s ability to adapt to diverse project needs and scenarios demonstrates its versatility and practical application.

In conclusion, the DIY Project Planner with AI offers an efficient and effective solution for managing personal and professional projects. By automating repetitive tasks, optimizing resources, and providing intelligent insights, the system helps users focus on achieving their project goals with minimal effort. As AI continues to advance, the potential for further improving such tools is immense, making them indispensable for modern project management needs..

7.2 FUTURE WORK

The project has a wide scope in the current context, with most proposed features successfully implemented. Future work will involve creating a centralized database to store extracted data for better management. Additionally, further research will focus on exploring advanced techniques, libraries, and methods to enhance the system. Ongoing development will aim to improve functionality, scalability, and user experience.

APPENDICES

A. SOURCECODE

FRONTEND

Register.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>DIY Project Planner - Register</title>
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">
  <style>
    body {
      background-color: #f8f9fa; /* Light grey for a modern look */
      height: 100vh;
      display: flex;
      justify-content: center;
      align-items: center;
    }
    .register-container {
      background-color: #ffffff; /* White container for form */
      padding: 30px;
      border-radius: 8px;
      box-shadow: 0 10px 20px rgba(0, 0, 0, 0.1);
      width: 100%;
      max-width: 400px;
    }
    .form-control:focus {
      box-shadow: 0 0 5px rgba(0, 123, 255, 0.5);
      border-color: #007bff;
    }
```

```
        .register-header {
            text-align: center;
            margin-bottom: 20px;
            color: #495057;
        }
        .btn-register {
            background-color: #007bff;
            color: white;
        }
        .btn-register:hover {
            background-color: #0056b3;
        }
        .login-link {
            text-align: center;
            margin-top: 15px;
        }
        .login-link a {
            color: #007bff;
            text-decoration: none;
        }
        .login-link a:hover {
            text-decoration: underline;
        }
    </style>
</head>
<body>
    <div class="register-container">
        <h2 class="register-header">Register for DIY Project Planner</h2>
        <form action="{% url 'register'     %}" method="POST">
            {% csrf_token %}
            <div class="mb-3">
                <label for="username" class="form-label">Username</label>
                <input type="text" class="form-control" id="username" name="username"
placeholder="Enter your username" required>
            </div>
            <div class="mb-3">
                <label for="email" class="form-label">Email address</label>
                <input type="email" class="form-control" id="email" name="email"
placeholder="Enter your email" required>
            </div>
            <div class="mb-3">
                <label for="password" class="form-label">Password</label>
                <input type="password" class="form-control" id="password" name="password"
placeholder="Enter your password" required>
            </div>
            <div class="mb-3">
                <label for="confirmPassword" class="form-label">Confirm Password</label>
                <input type="password" class="form-control" id="confirmPassword"
name="confirmPassword" placeholder="Re-enter your password" required>
            </div>
            <button type="submit" class="btn btn-register w-100">Register</button>
        </form>
        <div class="login-link">
            <p>Already have an account? <a href="/login">Login here</a></p>
        </div>
    </div>
    <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>
</body>
</html>
```

Login.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>DIY Project Planner - Login</title>
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"
```

```
rel="stylesheet">
    <style>
        body {
            background-color: #f3f4f6; /* Light grey background for a soft look */
            height: 100vh;
            display: flex;
            justify-content: center;
            align-items: center;
        }
        .login-container {
            background-color: #ffffff; /* White box for contrast */
            padding: 40px;
            border-radius: 10px;
            box-shadow: 0 10px 20px rgba(0, 0, 0, 0.1); /* Subtle shadow for depth */
            width: 100%;
            max-width: 400px;
        }
        .form-control:focus {
            box-shadow: 0 0 5px rgba(0, 123, 255, 0.5);
            border-color: #007bff;
        }
        .login-header {
            text-align: center;
            margin-bottom: 20px;
            color: #343a40;
        }
        .btn-login {
            background-color: #007bff; /* Bootstrap primary color */
            color: #fff;
        }
        .btn-login:hover {
            background-color: #0056b3;
        }
        .register-link {
            text-align: center;
            margin-top: 15px;
        }
        .register-link a {
            color: #007bff;
            text-decoration: none;
        }
        .register-link a:hover {
            text-decoration: underline;
        }
    </style>
</head>
<body>
    <div class="login-container">
        <h2 class="login-header">Login to DIY Project Planner</h2>
        <form action="{% url 'login' %}" method="POST">
            {% csrf_token %}
            <div class="mb-3">
                <label for="username" class="form-label">Username</label>
                <input type="text" class="form-control" id="email" name="username"
placeholder="Enter your Username" required>
            </div>
            <div class="mb-3">
                <label for="password" class="form-label">Password</label>
                <input type="password" class="form-control" id="password" name="password"
placeholder="Enter your password" required>
            </div>
            <button type="submit" class="btn btn-login w-100">Login</button>
        </form>
        <div class="register-link">
            <p>Don't have an account? <a href="/register">Register here</a></p>
        </div>
    </div>

    <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>
```

```
</body>
</html>
```

<u>Home.html</u>

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>DIY Project Planner</title>
    <link                href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"
rel="stylesheet">
    <style>
        body {
            background-color: #f8f9fa;
        }
        .hero-section {
            background-color: #343a40;
            color: #ffffff;
            height: 100vh;
            display: flex;
            justify-content: center;
            align-items: center;
            text-align: center;
        }
        .hero-text {
            font-size: 2.5rem;
            margin-bottom: 20px;
        }
        .nav-link {
            color: #ffffff !important;
        }
        .project-section {
            background-color: #ffffff;
            padding: 60px 0;
        }
        .project-box {
            background-color: #e9ecef;
            border: 1px solid #ddd;
            border-radius: 8px;
            padding: 20px;
            text-align: center;
            transition: transform 0.3s ease, box-shadow 0.3s ease;
        }
        .project-box:hover {
            transform: translateY(-10px);
            box-shadow: 0 10px 20px rgba(0, 0, 0, 0.1);
        }
        .project-icon {
            font-size: 3rem;
            color: #0d6efd;
            margin-bottom: 10px;
        }
        .project-title {
            font-size: 1.5rem;
            font-weight: bold;
            margin-bottom: 10px;
        }
        .project-description {
            color: #6c757d;
        }
        .footer {
            background-color: #343a40;
            color: #ffffff;
            padding: 20px 0;
            text-align: center;
        }
    </style>
</head>
```

```
<body>
    <!-- Navbar -->
    <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
        <div class="container-fluid">
            <a class="navbar-brand" href="#">DIY Project Planner</a>
            <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">
                <span class="navbar-toggler-icon"></span>
            </button>
            <div class="collapse navbar-collapse" id="navbarNav">
                <ul class="navbar-nav ms-auto">
                    <li class="nav-item">
                        <a class="nav-link" href="#">Home</a>
                    </li>
                    <li class="nav-item">
                        <a class="nav-link" href="#">About</a>
                    </li>
                    <li class="nav-item">
                        <a class="nav-link" href="#">Features</a>
                    </li>
                    <li class="nav-item">
                        <a class="nav-link" href="#">Contact</a>
                    </li>
                </ul>
            </div>
        </div>
    </nav>

    <!-- Hero Section -->
    <div class="hero-section">
        <div>
            <h1 class="hero-text">Welcome to DIY Project Planner</h1>
            <p>Plan, manage, and organize your DIY projects with ease.</p>
            <div class="d-flex justify-content-center">
```

```
                <a href="/login" class="btn btn-primary me-2">Login</a>
                <a href="/register" class="btn btn-secondary">Register</a>
            </div>
        </div>
    </div>

    <!-- Project Section -->
    <div class="container project-section">
        <h2 class="text-center mb-5">Explore DIY Project Categories</h2>
        <div class="row g-4">
            <div class="col-md-4">
                <div class="project-box">
                    <div class="project-icon">🔨</div>
                    <div class="project-title">Woodworking</div>
                    <p class="project-description">Create stunning furniture and decorative items with simple woodworking techniques.</p>
                </div>
            </div>
            <div class="col-md-4">
                <div class="project-box">
                    <div class="project-icon">🎨</div>
                    <div class="project-title">Arts & Crafts</div>
                    <p class="project-description">Unleash your creativity with DIY art projects, painting, and craft-making.</p>
                </div>
            </div>
            <div class="col-md-4">
                <div class="project-box">
                    <div class="project-icon">🔧</div>
                    <div class="project-title">Electronics</div>
                    <p class="project-description">Build cool gadgets and learn the basics of electronics with hands-on projects.</p>
                </div>
            </div>
        </div>
    </div>

    <!-- Footer -->
    <div class="footer">
        <p>&copy; 2024 DIY Project Planner. All Rights Reserved.</p>
    </div>

    <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>
</body>
</html>
```

Detail.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>{{ project.name }}</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background-color: #f4f4f9;
            color: #333;
        }
        .container {
            max-width: 800px;
            margin: 20px auto;
            padding: 20px;
            background: #fff;
            border-radius: 10px;
            box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);
        }
        img {
            max-width: 100%;
            height: auto;
            border-radius: 5px;
        }
        h1 {
            text-align: center;
            color: #4CAF50;
        }
        h3 {
            margin-top: 20px;
            color: #555;
        }
        p {
            line-height: 1.6;
            margin: 10px 0;
        }
        a {
            display: block;
            margin-top: 20px;
            text-align: center;
            color: #4CAF50;
            text-decoration: none;
            font-weight: bold;
        }
    </style>
</head>
<body>
    <div class="container">
        <h1>{{ project.name }}</h1>
        <img src="{{ project.image.url }}" alt="{{ project.name }}">
        <h3>Description:</h3>
        <p>{{ project.description }}</p>
        <h3>Materials Required:</h3>
        <p>{{ project.materials }}</p>
        <h3>Steps:</h3>
        <p>{{ project.steps|linebreaksbr }}</p>
        <a href="{% url 'project_list' %}">Back to Projects</a>
    </div>
</body>
</html>
```

Bot.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>AI Project Chatbot</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background-color: #f4f4f9;
            color: #333;
            margin: 0;
            padding: 20px;
        }
        .container {
            max-width: 600px;
            margin: 0 auto;
            background: #fff;
            padding: 20px;
            border-radius: 10px;
            box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);
        }
        h1 {
            text-align: center;
            color: #4CAF50;
        }
        input, button {
            padding: 10px;
            margin: 10px 0;
            width: 100%;
            border-radius: 5px;
            border: 1px solid #ddd;
        }
        button {
            background-color: #4CAF50;
            color: white;
            cursor: pointer;
        }
        button:hover {
            background-color: #45a049;
        }
        #response {
            margin-top: 20px;
            padding: 10px;
            background-color: #f1f1f1;
            border-radius: 5px;
        }
    </style>
</head>
<body>
    <div class="container">
        <h1>AI Project Chatbot</h1>
        <input type="text" id="projectTitle" placeholder="Enter your project title">
        <button id="submitBtn">Get AI Response</button>
        <div id="response"></div>
    </div>

    <script>
        document.getElementById('submitBtn').addEventListener('click', function() {
            var title = document.getElementById('projectTitle').value;
            if (title) {
                fetch('/chat/', {
                    method: 'POST',
                    headers: {
                        'Content-Type': 'application/json',
                    },
                    body: JSON.stringify({ 'title': title }),
                })
                .then(response => response.json())
                .then(data => {
                    document.getElementById('response').innerText = data.response;
                })
                .catch(error => {
                    console.error('Error:', error);
                    document.getElementById('response').innerText = 'Failed to get a response from AI.';
                });
            } else {
                document.getElementById('response').innerText = 'Please enter a project title.';
            }
        });
    </script>
</body>
</html>
```

Backend

User_accounts.py

```
from django.shortcuts import render,redirect
from django.contrib import messages
from django.contrib.auth.models import User
from django.contrib.auth import authenticate,login
from django.contrib import auth
# Create your views here.
```

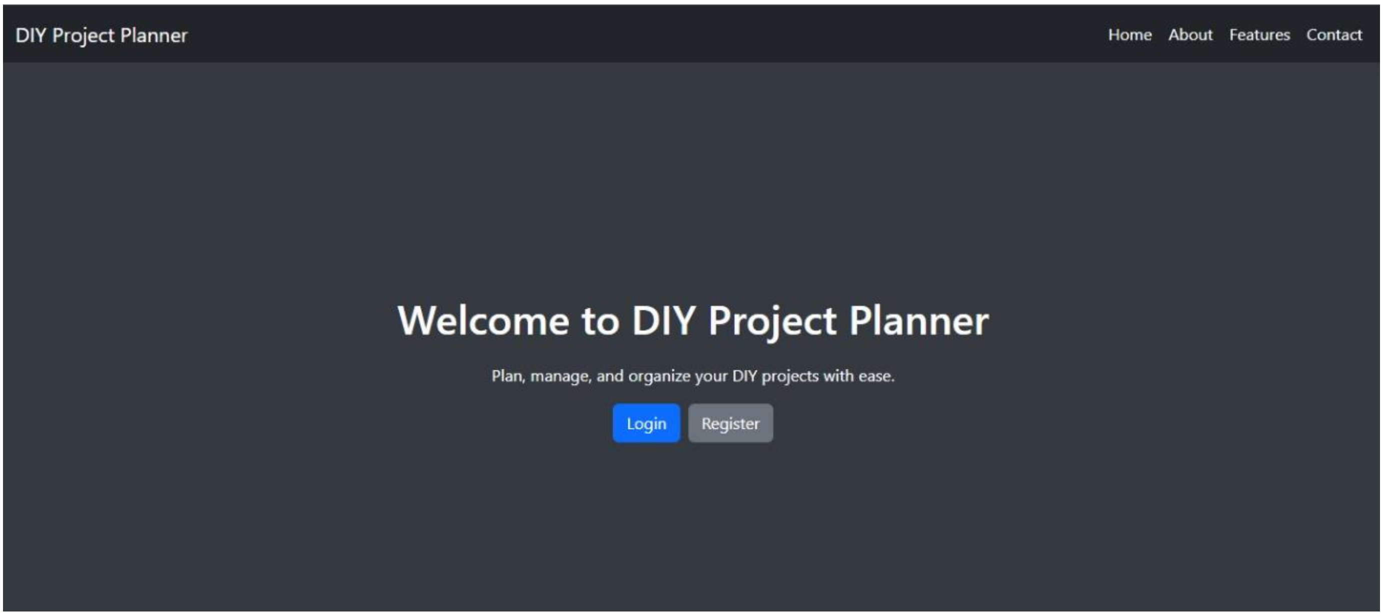
```
def register(request):
if request.method=="POST":
    user_name=request.POST['username']
    email=request.POST['email']
    password=request.POST['password']
    c_password=request.POST['confirmPassword']

    if password==c_password:

      if User.objects.filter(email=email).exists() or
User.objects.filter(username=user_name).exists():
        print("username or password taken")
        return redirect('message', title="Info", message="Already registered !")
      else:


user=User.objects.create_user(username=user_name,email=email,password=password)
        user.save()
        print("user created")
        return redirect('/')
    else:
      messages.error(request,"Password do not match")
      return redirect('message', title="error", message="Password not match")
  else:
    return render(request,"register.html")

def login_page(request):
  if request.method=='POST':

    username=request.POST['username']
    password=request.POST['password']
    user=authenticate(username=username,password=password)

    if user is not None:
      print("hi")
```

```
            login(request,user)
            return redirect('project_list')

    return render(request,'login.html')
```

Chatbot.py

```
import requests
from django.shortcuts import render
from django.http import JsonResponse

# Your Gemini API key
API_KEY = 'AIzaSyBtWAq0RrzGDCUQYo1jM2BiDfVJmKE4VmE'
API_URL = 'https://gemini-api-url.com/v1/query'  # Replace with actual Gemini API URL

def chat_view(request):
    if request.method == 'POST':
        title = request.POST.get('title')  # Get the project title from the form

        # Prepare the request payload
        payload = {
            'query': title,
            'key': API_KEY
        }

        # Send the request to Gemini API
        response = requests.post(API_URL, json=payload)

        if response.status_code == 200:
            response_data = response.json()
            chatbot_response = response_data.get('response', 'No response from AI.')

            return JsonResponse({'response': chatbot_response})
        else:
            return JsonResponse({'response': 'Error with the API request.'})

    return render(request, 'bot.html')
def bot(request):
    return render(request,'bot.html')
```

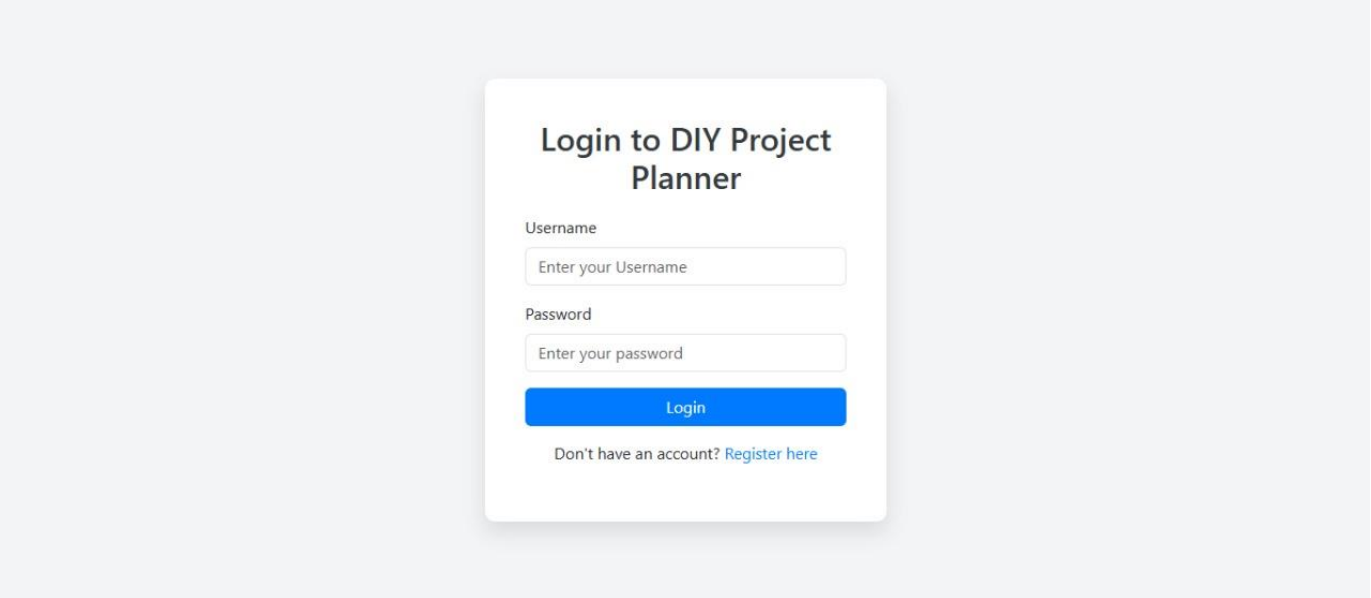
B.SCREENSHOTS



Welcome to DIY Project Planner

Plan, manage, and organize your DIY projects with ease.

HOME PAGE



Login to DIY Project Planner

Username

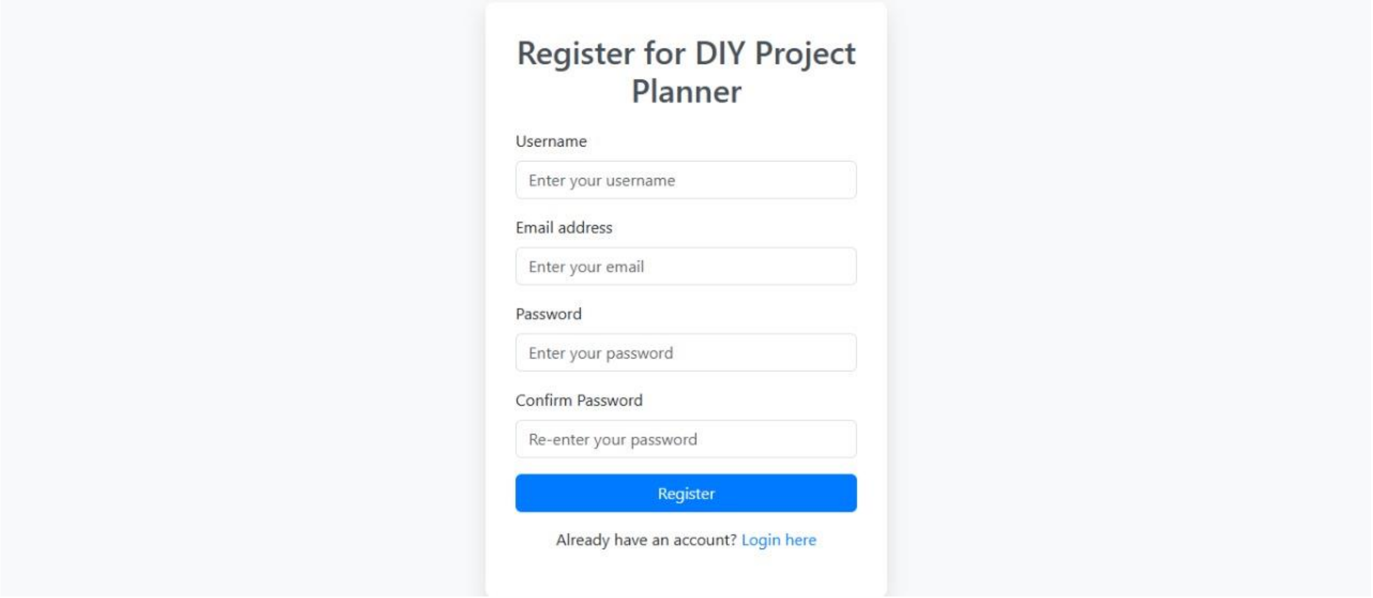
Enter your Username

Password

Enter your password

Don't have an account? Register here

LOGIN PAGE



Register for DIY Project Planner

Username

Enter your username

Email address

Enter your email

Password

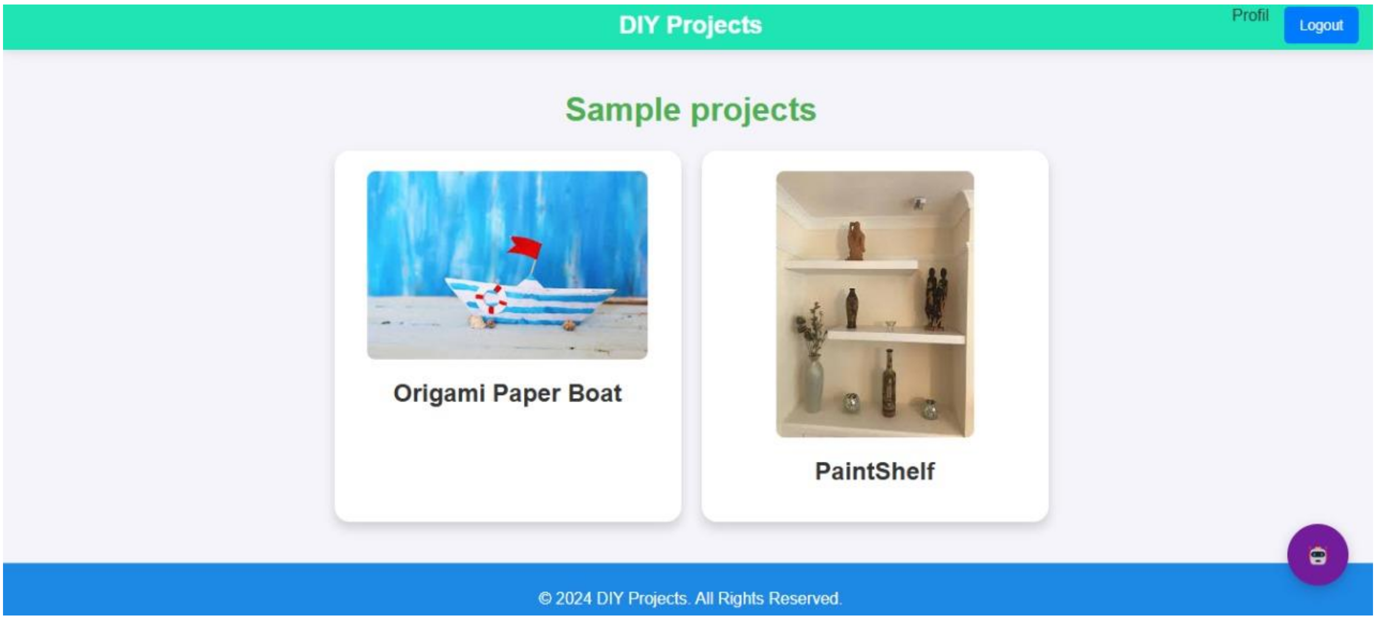
Enter your password

Confirm Password

Re-enter your password

Already have an account? Login here

REGISTRATION PAGE



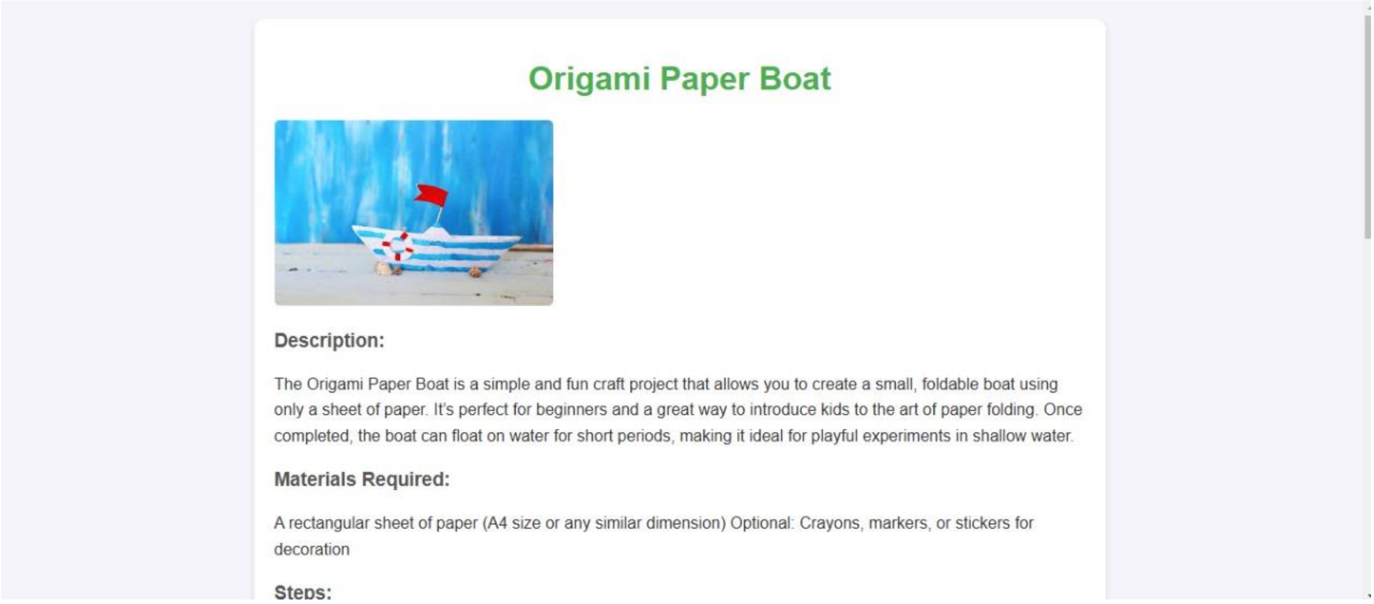
Sample projects

Origami Paper Boat

PaintShelf

© 2024 DIY Projects. All Rights Reserved.

PROJECT EXPLORER



Origami Paper Boat

Description:

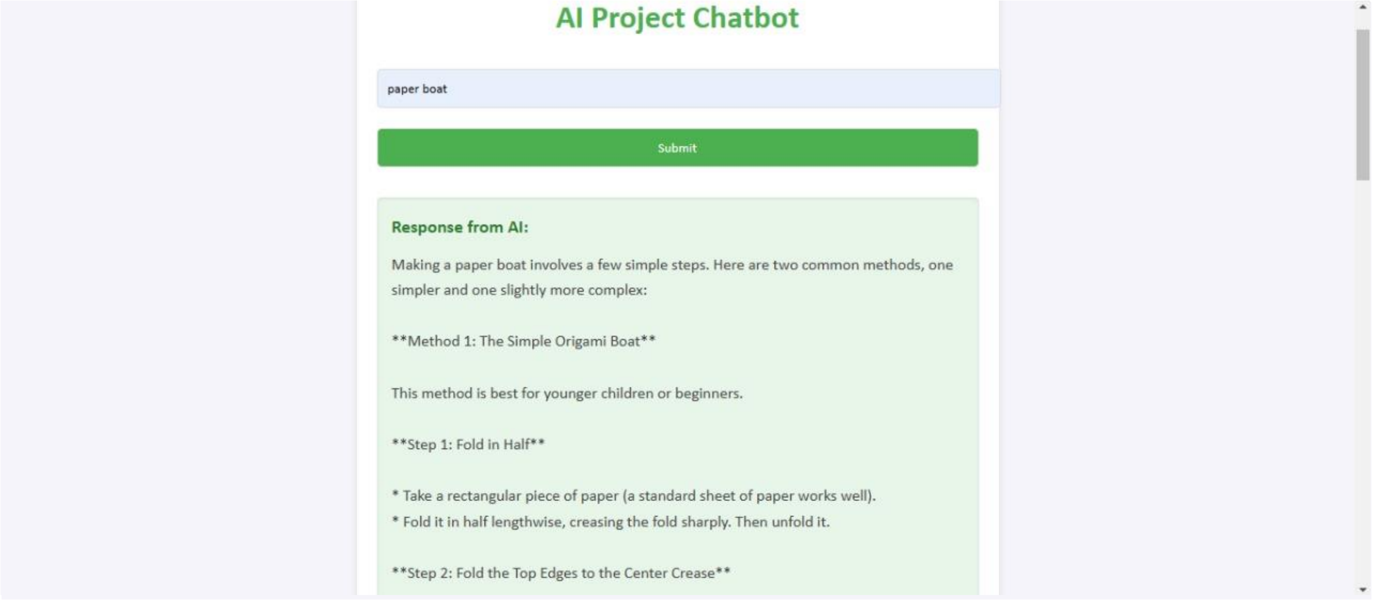
The Origami Paper Boat is a simple and fun craft project that allows you to create a small, foldable boat using only a sheet of paper. It's perfect for beginners and a great way to introduce kids to the art of paper folding. Once completed, the boat can float on water for short periods, making it ideal for playful experiments in shallow water.

Materials Required:

A rectangular sheet of paper (A4 size or any similar dimension) Optional: Crayons, markers, or stickers for decoration

Steps:

EXAMPLE PROJECT IDEA



AI Project Chatbot

paper boat

Response from AI:

Making a paper boat involves a few simple steps. Here are two common methods, one simpler and one slightly more complex:

Method 1: The Simple Origami Boat

This method is best for younger children or beginners.

Step 1: Fold in Half

* Take a rectangular piece of paper (a standard sheet of paper works well).
* Fold it in half lengthwise, creasing the fold sharply. Then unfold it.

Step 2: Fold the Top Edges to the Center Crease

AI CHATBOT

REFERENCES:

1. J. Smith, E. Brown, R. Kapoor, and M. Lopez, "AI-Driven DIY Project Planning Tools," Proceedings of the 2023 International Conference on Artificial Intelligence in Personal Applications (ICAIPA 2023), 2023.

2. S. Lee, M. Johnson, A. Mehta, and O. Turner, "Personalization in DIY Planning with Artificial Intelligence," Proceedings of the 2023 IEEE Symposium on Human-Centric AI Systems (HCAIS 2023), 2023.

3. P. Sharma, A. Khan, C. Rivera, and E. Zhang, "Efficient Budget and Resource Allocation in DIY Projects Using AI," Proceedings of the 2023 IEEE Conference on Resource Optimization and AI (CROAI 2023), 2023.

4. J. Clark, R. Davis, S. Nair, and Y. Matsuda, "Real-Time Collaboration and Progress Tracking in DIY Projects," Proceedings of the 2023 IEEE International Conference on Collaborative AI Systems (ICAS 2023), 2023.

5. J. Quintero and R. Asprilla, "Towards an Efficient Voice-Based Chatbot," Proceedings of the 2020 IEEE Thirty-Fifth Central American and Panama Convention (CONCAPAN XXXV), 2020.

6. S. Ramesh, K. Patel, and J. Wang, "AI-Powered Task Scheduling and Automation in DIY Environments," Proceedings of the 2023 ACM Symposium on Interactive Systems (ACMIS 2023), 2023.

7. A.Baker, T. Nguyen, and M. Ramirez, "Exploring Computer Vision for DIY Project Monitoring," Proceedings of the 2023 IEEE Conference on Visual Analytics (CVA 2023), 2023.

8. K. Robinson and L. Chen, "Integrating NLP in Personal Project Management Tools," Proceedings of the 2023 IEEE Workshop on NLP and AI Applications (NLP-AIA 2023), 2023.

9. V. Gonzalez, H. Carter, and P. Das, "AI for Small-Scale Project Planning: Applications in Home Improvement," Proceedings of the 2023 IEEE Smart Home and IoT Symposium (SH-IOT 2023), 2023.

10. M. Torres, A. Kumar, and R. Li, "AI and Predictive Analytics in DIY Resource Management," Proceedings of the 2023 IEEE Conference on Predictive AI Systems (CPAIS 2023), 2023.