INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

Impact Factor: 8.379

Task Ease
(Automated Task Manager)

Monthri Mrudhula, Mellacheruvu Venkata Gaayathri, Boddupally Sanjana, R. Sirisha

B. E. Student, Department of Artificial Intelligence & Data Science, Stanley College of Engineering & Technology for Women, Hyderabad, India

Assistant Professor, Department of Artificial Intelligence & Data Science, Stanley College of Engineering & Technology for Women, Hyderabad, India

ABSTRACT: In the modern digital landscape, efficient task management is crucial for individuals. The Integrated Task Manager Application is a cutting-edge solution designed to streamline and optimize task handling through the integration of advanced AI capabilities. This application revolutionizes task management using AI-driven language understanding and automation, making task handling more efficient and user-friendly through Natural Language Processing (NLP) technologies. It prioritizes tasks based on deadlines, importance, and user preferences. Additionally, it incorporates speech-to-text capabilities, enabling users to effortlessly input tasks by speaking, further enhancing convenience and accessibility. This capability enables precise classification, aiding in better organization and prioritization of tasks. The application employs NLP algorithms to interpret natural language input, allowing users to input tasks in conversational language through chatting. The application's ability to comprehend and extract task details from unstructured text enhances user convenience and efficiency. It has an intuitive and user-friendly interface, ensuring a smooth and enjoyable user experience. Some of the core features of this application include NLP-Powered Task Input and Understanding, Contextual Task Analysis and Categorization, Smart Reminders and Notifications. APIs are vital in the Integrated Task Manager Application, enabling task management functions like adding, updating, deleting, and retrieving tasks from the database. They integrate speech-to-text for verbal task input, use NLP for text interpretation, handle user authentication, and provide notification services for reminders and alerts.

KEYWORDS: Task Management; Speech – to – Text; NLP; Smart Reminders; APIs.

I. INTRODUCTION

In today's fast-paced digital landscape, task management is increasingly becoming a challenge for individuals striving to stay organized and efficient. To address this need, a new and innovative application has emerged, aiming to revolutionize the way tasks are managed. By leveraging cutting-edge technologies such as Natural Language Processing (NLP), this application seeks to transform the efficiency and effectiveness of task management for users.

At the core of this application is a novel approach to task prioritization and organization. Using AI-driven language understanding and automation, the application intelligently sorts tasks based on various factors including deadlines, importance, and user preferences. This not only helps in prioritizing tasks effectively but also ensures that each task is analyzed in its specific context, leading to a more organized and streamlined approach to task management.

One of the standout features of this application is its seamless integration of speech-to-text capabilities. This allows users to input tasks effortlessly by simply speaking, significantly enhancing accessibility and ease of use. Furthermore, the application's ability to interpret natural language input through NLP algorithms enables users to interact with the task manager using conversational language, making the process more intuitive and user-friendly.

In addition to its advanced task management capabilities, the application also includes smart reminders and notifications. These features help users stay on top of their tasks without the need for constant manual oversight, ensuring that important deadlines are not missed.

APIs are integral to the Integrated Task Manager Application, facilitating essential functions and connectivity between its components. They enable the system to manage tasks efficiently by performing operations like adding, updating, deleting, and retrieving task details from the underlying database. Additionally, APIs integrate speech-to-text functionality, allowing users to input tasks verbally, and leverage Natural Language Processing (NLP) to interpret text input for categorization and organization. Furthermore, APIs handle user authentication, ensuring secure access, and enable notification services for timely reminders and alerts, enhancing the user experience.

Overall, this application represents a fusion of advanced AI technologies and user-centric design principles, setting a new standard for efficiency and convenience in task management. It aims to transform the way individuals manage their tasks, making the process more efficient, organized, and enjoyable.

II. OBJECTVES

- ✓ Revolutionize task management
- ✓ Transform efficiency and effectiveness
- ✓ Intelligently prioritize tasks
- ✓ Enable effortless task input through speech-to-text
- ✓ Interpret natural language input through chat
- ✓ Provide smart reminders and notifications
- ✓ Set a new standard for efficiency and convenience

III. LITERATURE SURVEY

Table 1: Analysis of Techniques used in Literature Survey

Sl. No	Title & Author	Contribution	Technology Stack
1	Kenneth Conley, James Carpenter, "Towel: Towards an Intelligent To-Do List"	Proposed a personal task manager named Towel (Towards an Intelligent To-Do List), designed to assist users with task management and delegation.	Artificial Intelligence, Natural Language Processing, Integration with CALO system, communication protocols.
2	Android Java, XML, HTML, CSS for the front end, and PHP for the back end, it ensures a robust and user-friendly interface.	Android Java, XML, HTML, CSS for the front end, and PHP for the back end, it ensures a robust and user-friendly interface.	Task management algorithms, scheduling algorithms, Data structures for efficient storage and retrieval, Decision support algorithms, possibly AI-driven assistance, Task prioritization algorithms.
3	Grishma Hedaoo, Priyanka Thoke, Raksha Tabhane, Shubham Meshram, Swapnil Kumbalkar, "Online Task Management System (OTMS)"	Proposed the Online Task Management System (OTMS) that streamlines task management for HODs, students, teachers, and assistants in a college setting.	Android Java, XML, HTML, CSS for the front end, and PHP for the back end, it ensures a robust and user-friendly interface.
4	Sameera A. Abdul-Kader and Dr. John Woods, "Survey on Chatbot Design Techniques in Speech Conversation Systems"	Discussed about Human-Computer Speech, a growing method for computer interaction.	NLTK in Python aids chatbot design, which includes Responder, Classifier, and Graphmaster components. Performance evaluation relies on metrics like the Turing Test and Loebner Prize.

5	Dominic Weber, Alireza Sahami Shiraz, Niels Henze, "Towards Smart Notifications using Research in the Large"	Introduced a comprehensive system designed for sharing notifications seamlessly across a variety of devices, including smartphones, tablets, PCs, smart TVs, smartwatches, and smart glasses.	OAuth for user authentication. Bi-directional communication channel between client applications and the server application. Extensions for the Mozilla Firefox and Google Chrome web browsers. Push message services for pushing messages to mobile devices.
6	Konstantinos I. Roumeliotis and Nikolaos D. Tselikas, "ChatGPT and Open-AI Models: A Preliminary Review"	Proposed a study assessing ChatGPT's limitations in paraphrasing while highlighting its potential as a research tool, especially in creating a vaccine effectiveness article.	Unsupervised pre-training techniques.
7	Hao Wen, Yuanchun Li, Guohong Liu, Shanhui Zhao, Tao Yu, Toby Jia-Jun Li, Shiqi Jiang, Yunhao Liu, Yaqin Zhang, Yunxin Liu, "Empowering LLM to use Smartphone for Intelligent Task Automation"	Proposed AutoDroid which emerges as a pioneering system that addresses the limitations of existing mobile task automation approaches.	Large Language Models (LLMs), Smartphones, Android Applications, AutoDroid System.

Table 2: List of Reviewed Applications

Application	Description	Platforms	Pros/ Cons
Google Keep	Designed primarily for personal use, excels in simplicity and accessibility for uncomplicated to-do lists.	iOS, Mac, Android, Windows, Web	• Simple, accessible • Mobile & Voice Reminders • Flexible drawing and typing options • Direct integration into Google Docs • Falls short in rich-text formatting • Lacks the ability to share groups of notes
Wunderlist	Personal to-do list app with a clean interface, resembling a traditional list.	iOS, Mac, Android, Windows, Web, Kindle Fire	• Clean and intuitive interface • Seamless collaboration for sharing lists and coordinating tasks • Cross-platform functionality • Efficient task organization • Local data backup options • Absence of a calendar view • Lack of advanced collaboration tools
Microsoft To-Do	Designed as a successor to Wunderlist, is a personal to-do list app tailored for dedicated Outlook users.	Web, Windows, iOS, Mac, Android	• Integration with Outlook Tasks • Personalized daily planners • Cross-device functionality • Absence of subtasks • Limited collaboration tools

Todoist	A task management app for small teams or personal use, with a notepad-style task list. Features include due dates, recurring dates, and priorities.	Web, Windows, iOS, Mac, Android	• Seamless integration with productivity tools • Features like creating tasks from email, task feedback, and comments • Library of project templates • Lack of advanced task tracking features • Subpar calendar view
TickTick	A renowned collaborative to-do app designed for fans of the GTD (Get Things Done) method. The app facilitates quick idea capture through its inbox and features a three-panel interface for seamless task management.	Web, Windows, iOS, Mac, Android	•Simple & structured project organization • Integrates with calendar • Can edit tasks on a daily basis • Can organize tasks using lists, tags & due dates • Built in pomodoro timer in the user-friendly interface

IV. PROPOSED SYSTEM

✓ NLP-Powered Task Input: Enables task creation through natural language, enhancing user-friendliness and intuitiveness.
✓ Speech-to-Text Capabilities: Allows task input by speech, increasing convenience and reducing manual effort.
✓ Smart Reminders and Notifications: Provides timely alerts for tasks, improving task management and completion.
✓ Task Prioritization: Enables users to prioritize tasks based on various factors, enhancing time management.
✓ Intuitive User Interface: Designed for easy understanding and navigation, improving user satisfaction.
✓ Wrapping Up of Tasks: Allows visual marking of completed tasks, providing a sense of accomplishment and progress tracking.

REQUIREMENTS

Software Requirements	Hardware Requirements
➢ Windows / Mac OS ➢ HTML ➢ CSS ➢ Bootstrap ➢ Python ➢ Render ➢ FastAPI ➢ MongoDB ➢ Editor: Visual Studio Code	➢ 2 GHz processor ➢ 4 GB RAM

SOFTWARE DESCRIPTION

FRONT END:

HTML
HTML (Hypertext Markup Language) is the standard markup language used to create and design web pages. It provides a structure for web content by using a system of tags and attributes to define the various elements within a page, such as headings, paragraphs, links, images, and more. HTML documents consist of a series of elements, each enclosed in opening and closing tags, which describe the content and its formatting.

CSS

CSS (Cascading Style Sheets) is a style sheet language used to define the presentation and layout of HTML documents. It allows web developers to control the appearance of elements on a web page, such as fonts, colors, spacing, and positioning. CSS works by selecting HTML elements and applying styles to them using selectors and declarations.

BOOTSTRAP

Bootstrap is a CSS framework for building responsive and mobile-first websites. It provides pre-designed templates and components, like buttons and forms, that can be easily customized. Bootstrap's grid system helps create responsive layouts, and its mobile-first approach ensures a good user experience on mobile devices. Overall, Bootstrap simplifies web development by providing a consistent and modern design without the need for extensive custom styling.

BACK END:

PYTHON

Python is a high-level, interpreted programming language known for its simplicity and readability. It supports multiple programming paradigms and has a vast standard library and ecosystem of third-party libraries. Python's readability is achieved through the use of indentation to define code blocks. It is widely used in web development, data science, artificial intelligence, and scientific computing, thanks to its versatility and active community.

RENDER

Render is a unified cloud platform for building and running apps and websites. It includes free TLS certificates, a global CDN, private networks, and automated deployments from Git. With Render, developers can easily create and manage their projects without worrying about infrastructure management.

FastAPI

FastAPI is a Python web framework for building APIs that is easy to use, fast, and efficient. It leverages Python 3.6+ features, such as type hints, and is built on top of Starlette and Pydantic. FastAPI automatically generates interactive API documentation and supports asynchronous programming for more efficient request handling. Overall, FastAPI simplifies API development in Python, making it a popular choice for high-performance APIs.

MongoDB

MongoDB is a popular NoSQL database known for its document-oriented data model, which stores data in JSON-like documents with dynamic schemas, providing flexibility and scalability. It offers high performance, scalability, and ease of use, making it suitable for various applications. MongoDB can scale horizontally by distributing data across multiple servers and supports replication for data availability and fault tolerance. Its query language is designed for querying JSON-like documents, simplifying data manipulation for developers.

V. MODULES & METHODOLOGY USED

MODULES

- Task Input through Chat Window: Allows users to input tasks using natural language, leveraging NLP and LLMs.
- Speech-to-Text Input: Enables task input by speaking, converting speech to text.
- Smart Reminders and Notifications: Provides intelligent alerts for upcoming tasks and deadlines.
- View and Delete Tasks: Allows users to view and delete tasks from the list.
- Update Tasks: Enables users to modify and save updated tasks.
- Prioritization: Organize tasks based on importance or urgency, facilitating focused task completion by assigning priority levels.

Functionality Overview of APIs

The functionalities of each API endpoint, the underlying implementation details, and the technologies used in the development process are described below. The application leverages MongoDB for data storage, FastAPI for building APIs, and various Python libraries for tasks such as date parsing, email sending, and password hashing.

- ✓ Registration: Users can register by providing a username, password, and email address. The application checks for existing usernames or emails and stores the hashed password in the database.
- ✓ Login: Users can log in with their username and password. The application validates the credentials and returns a success message if the login is successful.

✓ Task Management: Users can add, update, and delete tasks. The application extracts date and time information from task text using 'datefinder' package and prevents duplicate tasks with the same date and time.

✓ Password Reset: Users can request a password reset token, which is sent to their email address. The token is stored in the database with a TTL index for security.

✓ Email Notifications: Users receive email notifications for password resets and other important updates. The application uses the 'smtplib' library to send emails securely.

✓ Chatbot Interaction: Users can interact with a chatbot to get task-related information and assistance. The chatbot response is generated using the OpenAI GPT-3 API.

METHODOLOGY USED

➢ User Interaction: Users interact with the application through a UI built with HTML, including forms and buttons for task management. Tasks can be added using chat or speech-to-text.

➢ JavaScript Functions: Functions in the HTML handle user interactions, making API calls for CRUD operations and managing task reminders. Users can check off tasks once completed.

➢ FastAPI API Endpoints: FastAPI builds API endpoints in Python for CRUD operations, defining HTTP methods and URL paths.

➢ MongoDB Storage: MongoDB stores tasks as documents, managed by API endpoints interacting with the database.

Flow Summary:

• The user interacts with the HTML UI to add tasks seamlessly, using either chat or speech- to-text functionality.

• JavaScript functions in the HTML make API calls to FastAPI endpoints.

• FastAPI endpoints handle the API requests and interact with MongoDB to perform CRUD operations on tasks.

• MongoDB stores and manages the tasks data.

• JavaScript functions handle displaying task reminders or notifications in popups and allow the user to check off checkboxes once a task is completed.

This flow allows for a dynamic and interactive user experience, with the HTML UI handling user interactions and the FastAPI backend managing the data and business logic.

VI. CONCLUSION

TaskEase, the GenAI Integrated Task Manager Application emerges as a useful solution in the contemporary digital landscape, amalgamating contemporary technologies to redefine task management efficiency and user interaction. Through the integration of Artificial Intelligence (AI) capabilities, specifically Large Language Models (LLMs) and Natural Language Processing (NLP), this application signifies a paradigm shift in how individuals navigate and handle their tasks.

A primary distinguishing feature of GenAI lies in its adept utilization of machine learning algorithms to judiciously prioritize tasks. These algorithms take into account various parameters such as deadlines, task importance, and user preferences, empowering users to concentrate on critical aspects while the application organizes and categorizes tasks. The incorporation of LLMs empowers the application to comprehend the nuanced context and intent underlying each task, setting a new standard for comprehension and organization in the realm of task management.

The inclusion of speech-to-text functionality elevates user convenience to unprecedented levels, allowing for the effortless input of tasks through spoken commands. GenAI's utilization of NLP algorithms enables users to input tasks in natural language, transforming the dynamics of task communication and interpretation. The application's capacity to extract details from unstructured text ensures that users can articulate tasks in a conversational manner, obviating the need for rigid and structured inputs. GenAI's user interface is meticulously crafted to be intuitive and user-friendly, prioritizing a seamless and enjoyable user experience.

In essence, the GenAI Integrated Task Manager Application transcends the conventional definition of a task management tool. It stands as a transformative solution, leveraging AI capabilities to augment efficiency, organizational efficacy, and accessibility. With its NLP-powered task input, contextual task analysis, categorization, and smart reminders, GenAI establishes a new benchmark for task management in the digital era, offering a holistic and user-centric approach to navigating the intricacies of modern life.

REFERENCES

1. Kenneth Conley, James Carpenter, "Towel: Towards an Intelligent To-Do List"
2. Victoria Bellotti, Brinda Dalal, Nathaniel Good, Peter Flynn, Daniel G. Bobrow and Nicolas Ducheneaut, "What a To-Do: Studies of Task Management Towards the Design of a Personal Task List Manager"
3. Suyash Kejriwal, Vaibhav Vishal, Aastha Gulati, Gaurav Gambhir, "A Review of Daily Productivity Growth Using To-Do Manager"
4. Grishma Hedaoo, Priyanka Thoke, Raksha Tabhane, Shubham Meshram, Swapnil Kumbalkar, "Online Task Management System (OTMS)"
5. Sameera A. Abdul-Kader and Dr. John Woods, "Survey on Chatbot Design Techniques in Speech Conversation Systems"
6. Asifur Rahman, Ferdouse Hassan Nowrin, Tasnimul Haque Nahin, Mahajabin Kabir Arin, Delwar Hossain Roman, "Project Report of Calendar App with Task Planner"
7. David Golding, "Creating a To-Do List Application"
8. Di Zhu, Dahua Wang, Ruonan Huang, Yuchen Jing, Li Qiao and Wei Liu, "User Interface (UI) Design and User Experience Questionnaire (UEQ) Evaluation of a To-Do List Mobile Application to Support Day-To-Day Life of Older Adults"
9. Oguzhan Topsakal, T. Cetin Akinci, "Creating Large Language Model Applications Utilizing LangChain: A Primer on Developing LLM Apps Fast"
10. Hao Wen, Yuanchun Li, Guohong Liu, Shanhui Zhao, Tao Yu, Toby Jia-Jun Li, Shiqi Jiang, Yunhao Liu, Yaqin Zhang, Yunxin Liu, "Empowering LLM to use Smartphone for Intelligent Task Automation"

LINKS

1. https://cdn.aaai.org/Symposia/Spring/2007/SS-07-04/SS07-04-005.pdf
2. https://www.researchgate.net/publication/221518959_What_a_to-do_studies_of_task_management_towards_the_design_of_a_personal_task_list_manager
3. https://www.irjmets.com/uploadedfiles/paper/volume2/issue_12._december_2020/5480/1628083219.pdf
4. https://www.mdpi.com/1999-5903/15/6/192
5. https://journals.mesopotamian.press/index.php/BJAI/article/view/201/192
6. https://quire.io/compare/best-to-do-list-apps

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462   6381 907 438   ijircce@gmail.com