International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

# Development of an personal AI assistant for Unified Task Management

Bharat Kumar
Computer Science and Engineering
Galgotias University

Vinay Kumar
Computer Science and Engineering
Galgotias University

Vimal Singh
Assitant Professor
Computer Science and Engineering
Galgotias University

*Abstract* - In today's digital environment, people often find themselves using several different applications to communicate, create time schedules, organize notes, keep track of things to do, etc. Although these devices are often beneficial, they tend to promote disjointed working environments and increased brain effort. The present-day AI assistants (Siri, Alexa, Google Assistant, etc.) primarily function with defined responses and reveal neither optimum multi-platform comprehension nor a strong ability to adapt dynamically to users' changing rituals. The present research deals with the design and development of a Personal AI Assistant that brings various digital utilities to bear on one intelligent device. The digital assistant suggested integrates data from calendars, emails, notes, and to-do lists, and uses natural language processing and machine learning procedures to ascertain user intent, automate repetitive chores, and suggest context-enlightening recommendations. The design of such a device centers about anticipating user requirement and automating task coordination, so as to reduce information load and improve efficiency in daily life. The outcome of this study demonstrates the strength inherent in an adaptive, integrated digital assistant to effectuate improvement in the efficiency of workflow and life overall.

*Index Terms* - Voice-based Interaction, Web Accessibility, Speech Recognition,

## I. INTRODUCTION

People in today's digital society depend on multiple applications to manage their communication needs and stay organized while working productively. Users must split their attention between separate platforms because these tools function independently despite offering convenience. The way different systems operate creates inefficient workflows which require users to perform additional mental work. Siri and Alexa and Google Assistant serve as basic digital assistants which execute simple commands yet they fail to detect patterns and link different system operations. The research develops a Personal AI Assistant which merges multiple productivity platforms into an intelligent unified system to solve current system limitations. The system unites calendar information with email content and note data and task schedules through natural language processing and machine learning to execute user commands and understand behavior patterns and perform scheduled operations. The system actively handles scheduling and work assignments to decrease digital information overload which results in better work performance. The research results demonstrate how adaptive assistants with context-sensitive capabilities can boost operational performance and digital health and concentration abilities.

### *A. Problem Statement*

1) *Fragmented Digital Environments:* Because individuals rely on, and thus engage, multiple digital units, such as messaging, calendaring, and note-taking platforms, they constantly transfer from one platform to another in order to accomplish even simple tasks. This fragmentation disrupts the continuity of workflow and wastes time.

2) *Cognitive Overload in Task Management:* The mental exertion required to manage deadlines, prioritize tasks, and keep schedules in conformance requires continuous effort. The refluctuating attention given to different platforms and the recalling of tasks not accomplished mentally apply cognitive pressure and affect concentration.

3) *Time Savings through Automation :* There are many chores, such as reminders, drafting repetitive messages and suggesting sales, which are capable of being automated. The accomplishment of repetitive chores by automation frees the user for other purposes to direct energy and time in the creative and strategic phases of work.

## II. RELATED WORK

Historically, how we organize our various job and life responsibilities (daily) were spread between multiple forms of low-tech/analog methods and many small individual pieces of software on our computers. The modern context (many devices) has brought a need for a Unified Task Management (UTM) approach to be created by both researchers and software designers. The idea is to take the many different types of information that require an individual's attention (email, calendar invites, IM's, PM tickets, and what we see in the ”real world”) and combine/develop a single workflow for performing those tasks. Even though PIM's (Personal Information Managers) were the first application to try to bring together all of the above forms of information into one solution, they typically end up creating ”data silos,” where data from separate proprietary PIM's is either unavailable to a central processing or the proprietary systems only give limited access to the aggregated data.

The present report contains a comprehensive study of the literature regarding the creation of an automated way to unify task management using Personal AI Assistants. The theoretical background on Personal Information Management (PIM) and the psychological difficulty associated with the fragmentation of

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

tasks is reviewed. The current status of agentic architectures, including process automation using the ProAgent framework [14] and context-aware proactive systems using the ContextAgent framework [15], is reviewed. Technologies that enable this are reviewed in depth, including the Model Context Protocol (MCP) [6] and the ReQAP method for heterogeneous data searching. The human-centric problems of developing and implementing these advanced systems, such as trust, cognitive load, and security, are evaluated.

Fragmentation, the lack of cohesion in the knowledge workers' creative processes, is a known cognitive burden on all knowledge workers. For example, when organizing a quarterly review meeting, the process will consist of multiple digital sources: emails containing discussions of the agenda, calendar invites for coordinating the meeting time, folders in the cloud for storing reports and files, and instant messaging streams for discussing attendees. [2]

This fragmentation problem, as identified by PIM literature, creates the need for the end user to serve as the integration layer of all of the information sources. The user is then required to mentally map the semantic connections between an email from "John" and a calendar entry titled "Q3 Sync." The UPI framework addresses some of this issue by stating that one of the core requirements of next-generation systems is to unify or aggregate the metadata across personal devices and cloud-based services, closing the gaps between all three of the time, space, and socialization dimensions of memory. Failure to provide for these integrated sources means the user will continue to have scattered or disconnected "islands of information" that cannot be managed comprehensively. [9]

## III. SYSTEM ARCHITECTURE

The research approach embraced in this work brings together analytical study and user-centred system design. Analytical study is concerned with an investigation of existing personal assistant technologies with examinations of their advantages and disadvantages, and an understanding of the practical problems which people faced in dealing with digital tasks. This user-centred design approach ensured that the system which developed as a result of this work produced a tool which satisfied human needs, and were not instrumental in providing a demonstrative theoretically efficient solution or a technical tool. The system was iteratively developed based upon a users' reactions, based on interaction patterns, evaluation of usability, and functional testing. In this way, the dual mode of research had the outcome of theory and practical value.

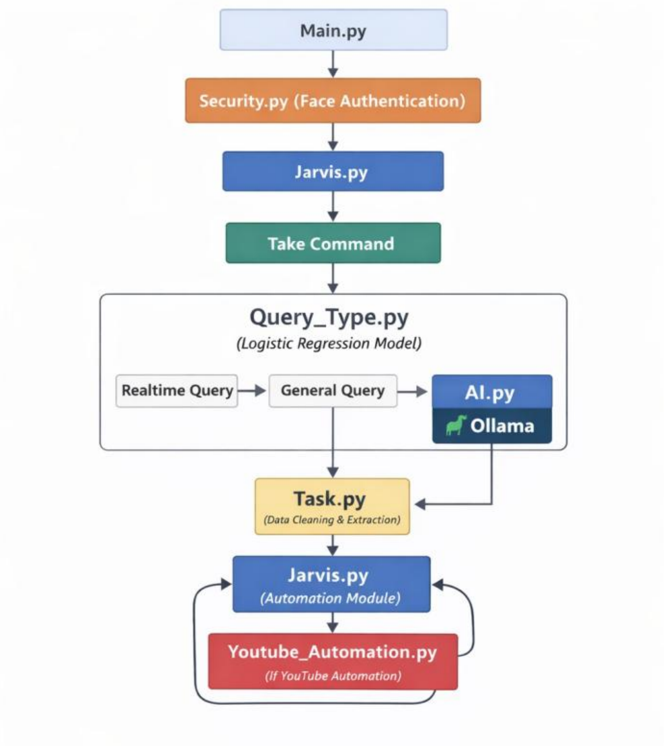


Fig. 1. System Architecture

### *A. Training of Machine Learning Systems*

For the assistant to be able to interpret the user intention and adapt itself to the behavioural patterns of the subjects, machine learning techniques are necessary.

*1) Data Acquisition:* All training data will be collated from known public conversation databases, given commands, and training stimuli compiled from real user interactions used.

*2) Pre-processing:* Pre-processing of the text which is the response to the user is necessary in order to provide clean and meaningful input for the model, by normalisation of the text data interpretation, breaking into text pieces, and omitting any components not deemed necessary.

*3) Training Process :* Transformer-based NLP architectures were utilized for intent recognition. The model was fine-tuned using collected user interaction patterns, enabling improved personalization over time.

### *B. The General Query Processing through Logistic Regression and LLM (Ollama)*

A General Query is an information request that does not require the live retrieval of real-time data, incoming or outgoing calls to an API, or the ability to automate system functions at a system level. In general, this type of inquiry is heavily knowledge-based. A General Query is essentially an inquiry that can be answered using the large amount of existing knowledge already built into the training set for a Large Language Model (LLM). In this proposed solution, a General Query will first be run through the intent classification module, which analyses the user's input and then sends the appropriate classification to the LLM through the Ollama framework. Once classified, the General Query will be sent to the AI that will allow for the AI system to interact with the locally hosted Large Language Model

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

(LLM) framework that is Ollama. The LLM framework of Ollama applies the existing learnt language model to a General Query and generates a response using the LLM framework's existing knowledge. Once the output has been generated, the Core Controller is responsible for sending it back to the User via the display or speech output.

General Queries do not impose real-time or automated response expectations on the LLM, but only expect the LLM to draw upon its internal reasoning capabilities and knowledge representation methods to produce a reasonable answer. Examples of General Queries include what is Artificial Intelligence?, Explain Logistic Regression, Who is APJ Abdul Kalam? and What is Python used For?

Logistic Regression is a good classifier for Intent Classification because it has low computational overhead, can provide fast predictions and provides a high level of accuracy for multiclass Intent detection (i.e., General Queries, Real Time Queries and Automation). The combined Hybrid Architecture of using traditional ML methods for Identifying Intent Classifiers with LLM's for producing a Natural Language Understanding of the request and generating responses allows for both Effective, Scalable, Robust and Reliable solutions to implement for Intelligent Personal Assistant Applications and Research Applications.

### *C. Frontend Architecture*

The frontend is the primary means by which users will interact with this system. Users will be able to utilize voice and text commands to communicate with the system using an intuitive interface.

The frontend captures the user's input and then sends it to the backend for further processing after detecting hotwords.

The frontend gives users real-time feedback by displaying and/or speaking back to them the responses that the system creates.

The modular design approach that has been adopted allows for better responsiveness, usability, and ease of integration with backend services.

### *D. Backend Architecture*

As the central processing layer of a back-end system, the modules that make up that system coordinate one another when processing data input, queries, responses, and completing tasks automatically.

To facilitate the classification of queries based upon type (General Query, Real-time Query, or Automated), backend modules utilize a logistic regression algorithm to classify them into one of these three types.

General Queries will produce AI-generated human-like responses utilizing the natural language processing capabilities of an LLM supported by Ollama.

Workflow task execution is also handled by backend modules that will route any automated request made by the user to a module that is specifically designed for that type of automation, such as a YouTube automation module.

### *E. Security Architecture*

Initially, system security is achieved through facial authentication, restricting access to authorized users only.

System security is not formally established until user identity verification is performed against the user before activating any core functionalities of the assistant; therefore, unauthorized usage will not occur.

All automated tasks or sensitive operations will only be executed after successfully passing authentication and verification.

The layered security design of the system has been created to ensure the reliability of user data, system resources, and automated actions from being misused.

## IV. IMPLEMENTATION ENVIRONMENT

Figure Labels: The implementation is carried out using the following environment and tools:

## V. METHODOLOGIES

### *A. Research Approach*

This research uses an analytical study combined with a user-centered system design (UCD) where we first analyzed existing personal assistant technologies to identify what their benefits or shortcomings were and what difficulties the users encounter on a daily basis when managing their digital tasks on multiple platforms. Based on this analytical study, we identified areas where there were gaps in the current task management area and in the personal assistant space. [10]

| Component | Technologies Used |
|---|---|
| Programming Language | Python |
| Speech Recognition | speech recognition + Google Web Speech |
| AI Model | OpenAI GPT (text-davinci-003 / updated model recommended) |
| Backed Logic | Python-based task automation and file handling |
| Platform Integration | webbrowser,OS command execution , app specific APIs |
| Operating System Used | Windows |

The UCD approach means that the system that we built prioritizes people, not the theoretical efficiency of the system or the technical optimization of the system. We used a process in the development of this system where we constantly received feedback from users, studied their interaction patterns, conducted usability evaluations, and performed functional testing. When done correctly, the iterative design process leads to higher usability and satisfaction among users (ISO, 2008). [11] Through a combination of analytical studies and practical development, we were able to gain both theoretical and practical knowledge to increase the overall contribution of this research [12]

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

*B. Collecting Requirements Using Different Methods*

Several different ways of collecting requirements were used when gathering information from users about their needs or expectations following established practices in software engineering (Sommerville, 2016). [13]

Interviews (Individual): To discover the main productivityrelated issues and problems with multiple digital tools, semistructured interviews were conducted with students, office workers and remote workers. Interviews are a powerful way to learn about users' pain points and provide additional context (Creswell ,Poth, 2018). [3]

Surveys (Structured Surveys): Quantitative data on daily digital habits, task management strategies, and the types of features that users prefer were gathered via structured surveys. Structured Surveys allow for large scale collection of data, and provide validation for the patterns observed during the interviews (Fowler, 2014). [4]

Workflow Analysis: A workflow analysis was completed to better understand how users accomplish their daily tasks, including the frequency of application switching, behaviour while scheduling, and productivity habits. Workflow analysis provides an opportunity to identify the inefficiencies in users' productivity processes and also offers possibilities for automation.

Utilizing both qualitative and quantitative data guarantees that the resulting functional requirements accurately reflect real user needs and the challenges they face with an automated system, thereby enhancing the relevance and usefulness of an automated system (Kitchenham ,Pfleeger, 2002). [7]

*C. Model of System Development*

An Agile methodology was chosen for this project due to its flexibility and ability to adapt to change. Agile is a methodology that enables the incremental delivery of a system by creating and executing short cycles (called sprints), which allow continuous testing, feedback integration, and enhancement of the system features (Beck et al., 2001). [8]

The Agile methodology allowed for concurrent development of fundamental system components (the natural language understanding component, the automation engine, the user interaction component, et cetera), a key advantage of utilizing Agile methodologies (Highsmith, 2009). Expressly, Agile provides the following major advantages for this type of project: [5]

1. Early identification of design and implementation problems and their subsequent resolution; 2. The use of user feedback on an ongoing basis in all aspects of the development of the system; 3. The ability to quickly respond to changes in requirements; and 4. Rapidly enhancing features incrementally.

These advantages make Agile methodologies particularly suited for the development of AI-based systems, where iterative improvement is necessary for optimum performance and usability (Amershi et al., 2019). [1]

## VI. LIMITATION

Not with standing the positive functioning of the system in the basic task management functions, some limitations were noted: Dependent on Speech Recognition: Performance is dependent on quality of background noise as well as microphone employed in device. Requires Accurate Personalization, (Behavioural Adaptation): The assistant requires significant interaction for accurate recommendations based on known behaviours to take place.Commercial/externally available Automation Limitation: Due to the nature of 3rd Party applications, there are limitations to automation possibilities due to permissions or API limitations.Focus on Individual Use: The system presently is designed to be used by an individual and presently does not support group sharing or collaborative tasks.

## VII. CONCLUSION

The project has demonstrated the design and development of a Personal AI Assistant meant to improve task management in Digital Productivity through the use of consolidated applications into a single intelligent system. The modern user uses many platforms and tools for communication, scheduling, note-taking and productivity tools that may result in degraded workflows and increased cognitive overhead. The introduced assistant seeks to rectify these issues with the use of natural language processing, machine learning and integration frameworks in order to interpret user commands, automate repetitive tasks as well as provide appropriate recommendations. The architecture and workflows of the system were laid out to provide efficient task management, data syncing and adaptability to user interaction. Testing and evaluation of the system demonstrated the ability of the assistant to perform basic task management functions that require less manual input. As the assistant learns user habits it is able to be increasingly more effective in its responses, thereby fostering increased efficiency and better task management. The project therefore demonstrates the capacity for intelligent integrated personal assistants to help people manage greater workloads digitally, thus providing a more useful, cohesive and user-centric digital experience.

## REFERENCES

[1] Saleema Amershi, Andrew Begel, Christian Bird, Robert DeLine, Harald Gall, Ece Kamar, Nachiappan Nagappan, Besmira Nushi, and Thomas Zimmermann. Software engineering for machine learning: A case study. In *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP)*, pages 291–300. IEEE, 2019.

[2] Philipp Christmann and Gerhard Weikum. Recursive question understanding for complex question answering over heterogeneous personal data. *arXiv preprint arXiv:2505.11900*, 2025.

[3] John W Creswell and Cheryl N Poth. *Qualitative inquiry and research design: Choosing among five approaches*. Sage publications, 2016.

[4] Floyd J Fowler Jr. *Survey research methods*. Sage publications, 2013.

[5] Jim Highsmith. *Agile project management: creating innovative products*. Pearson education, 2009.

[6] Xinyi Hou, Yanjie Zhao, Shenao Wang, and Haoyu Wang. Model context protocol (mcp): Landscape, security threats, and future research directions. *arXiv preprint arXiv:2503.23278*, 2025.

[7] Barbara Kitchenham and Shari Lawrence Pfleeger. Principles of survey research: part 5: populations and samples. *ACM SIGSOFT Software Engineering Notes*, 27(5):17–20, 2002.

[8] Agile Manifesto. Manifesto for agile software development, 2001.

[9] William Anthony Mason. *Indaleko: the unified personal index*. PhD thesis, University of British Columbia, 2025.

[10] Karen Myers, Pauline Berry, Jim Blythe, Ken Conley, Melinda Gervasio, Deborah L McGuinness, David Morley, Avi Pfeffer, Martha Pollack, and Milind Tambe. An intelligent personal assistant for task and time management. *AI Magazine*, 28(2):47–47, 2007.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

[11] Luisa Maria Ortiz-Escobar, Mario Andres Chavarria, Klaus Schonenberger, Samia Hurst, Michael Ashley Stein, Anthony Mugeere,¨ and Minerva Rivas Velarde. Assessing the implementation of usercentred design standards on assistive technology for persons with visual impairments: a systematic review. *Frontiers in Rehabilitation Sciences*, 4:1238158, 2023.

[12] Jennifer Preece. Citizen science: New research challenges for human–computer interaction. *International Journal of Human-Computer Interaction*, 32(8):585–612, 2016.

[13] Ian Sommerville. *YAZILIM MUHEND¨ ISL˙ I˙G˘I-Software Engineering˙* . Nobel Akademik Yayıncılık.

[14] Bufang Yang, Lilin Xu, Liekang Zeng, Yunqi Guo, Siyang Jiang, Wenrui Lu, Kaiwei Liu, Hancheng Xiang, Xiaofan Jiang, Guoliang Xing, et al. Proagent: Harnessing on-demand sensory contexts for proactive llm agent systems. *arXiv preprint arXiv:2512.06721*, 2025.

[15] Bufang Yang, Lilin Xu, Liekang Zeng, Kaiwei Liu, Siyang Jiang, Wenrui Lu, Hongkai Chen, Xiaofan Jiang, Guoliang Xing, and Zhenyu Yan. Contextagent: Context-aware proactive llm agents with open-world sensory perceptions. *arXiv preprint arXiv:2505.14668*, 2025.