iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** IV **Month of publication:** April 2026

DOI: https://doi.org/10.22214/ijraset.2026.79692

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

Action-Oriented Personal AI Assistant Using MCP Servers for Intelligent Task Execution

Dr. M. Sharada Varalakshmi[1], Sravan Chiluka[2], Srikanth Borkar[3], Srinidhi Jalla[4]
[1]*Professor, Dept. of Computer Science and Engineering, Methodist College of Engineering and Technology, Abids, Hyderabad, India*
[2, 3, 4]*Dept. Artificial Intelligence and Data Science, Methodist college of Engineering and Technology, Abids, Hyderabad, India*

Abstract: ***However, today's applications do not offer users any form of integration. Therefore, most people must switch between four to five apps when carrying out routine activities, such as scheduling, note-taking, searching for information, or sending reminders. Moreover, most of these apps cannot communicate with each other. In this paper, we present a solution to address this problem. Specifically, this paper seeks to describe an intelligent personal AI assistant with the capacity to integrate multiple functionalities, which currently exist in dozens of separate apps, in one application. This paper focuses on the Model Context Protocol (MCP), the architecture of which allows the AI to interact with external applications and services. This paper aims to highlight how the proposed solution works and how it can be implemented.***

The AI assistant would be capable of understanding the users' needs and picking the most relevant app to execute a task while leaving the users idle thanks to MCP servers. On the back end, Firebase would handle the management of user preferences and context transfer across sessions. Moreover, the AI assistant would support both text-based and voice commands, thus making it easier for users to schedule appointments, search for information, and send reminders.

Keywords: Artificial Intelligence, Intelligent Assistants, Model Context Protocol, Task Automation, Productivity Systems.

I. INTRODUCTION

For instance, most people need just a couple of applications in their everyday work, such as scheduling, note-taking, task managing, and searching for information online. Despite the fact that all these applications have improved our lives significantly, the reality is that they do not communicate with each other very often since they lack interconnectivity. The question is: why does it happen? The reason is that most users find themselves having to jump from platform to platform even for performing some of the simplest tasks. Such a process is extremely tiring for both users and decreases their productivity dramatically.

This fragmentation is a significant issue that should be addressed since people end up with numerous applications for different functions. Thus, users spend much time and, which is even more important, mental efforts on moving between applications, keeping up with all that is going on, and concentrating on what needs to be done. Several digital assistants have been designed by developers to make our lives easier, but most of these assistants can only handle the simplest tasks, such as making a call, turning off the lights, or asking about weather. That being said, however, change is coming. With the help of new developments in AI technology, particularly large language models and intelligent agent systems, there have been several opportunities created that could allow the technology to be utilized successfully. This includes not only allowing AI to comprehend natural language and determine the meaning behind what a person is saying but also to appropriately react to the situation. The biggest challenge, however, lies in creating safe and effective integration with the tools and services in the real world. It is precisely the MCP framework that comes to mind here. By providing for AI models a standardized interface for communicating with other systems, MCP enables them to exchange information, perform actions, and otherwise behave predictably. Indeed, MCP bridges the gap between reasoning-based AI and action-oriented AI, creating a powerful duo. As such, the current study builds upon that framework, putting forward its ideas for an Action-Oriented Personal AI Assistant that will employ MCP principles for the purpose of consolidating several different productivity-related tools within one framework. In particular, it is designed to take into account individual user goals, keep track of the context of the interaction, and provide actions like planning, reminders, document summaries, and information searching, without switching between applications. On top of that, it will be able to learn the preferences of individual users, thereby becoming more personalized over time. From the standpoint of this research, the proposed solution is particularly relevant because it combines excellent tool integration and reasoning abilities of AI. As a result, it promises to create genuinely effective digital assistants, rather than demonstration prototypes. At the same time, it moves closer to designing a fully-fledged productivity system based on the actual way people think.

II. LITERATURE REVIEW

Hou et al. (2024), paying particular attention to the architecture of the protocol and interactions of AI systems with other tools via the MCP. After reviewing the relevant literature and analyzing the architecture, the researchers concluded that the protocol performs well in terms of allowing modular and secure communication between different tools; however, one major deficiency was identified, namely, the absence of adequate governance structures within the ecosystem. The conclusion made by the researchers was that better authentication and security protocols should be implemented first [1].

Hasan et al. (2024) took up the task of analyzing existing deployments of MCP servers, in order to see how effective they perform. Conducting their study using empirical techniques as well as code-level static analysis of open-source programs, the authors found out several notable flaws, such as problems of tool poisoning vulnerability and maintainability, among others. In sum, according to the authors, what is needed is some kind of automated vulnerability detection systems and coding guidelines [2].

Zhao et al. attempt to analyze security risks associated with MCP toolchains specifically those that are able to propagate undetected until it is too late. Using data flow analysis, they were able to determine how these attacks were exploiting weak tool isolation. To prevent this from happening, the authors advise adhering to least privilege access and validation of input/output as minimum security precautions [3].

Singh et al. (2024) involved a comprehensive review of the MCP architecture, its interoperability, and unresolved issues. Through comparative analysis among various protocols for integration, they discovered that MCP significantly simplifies the complex nature associated with integrating several technologies, which is one of its evident strengths compared to previous models [4].

Snowflake (2025), there exists a framework which is centered around MCP servers to help integrate enterprise data within AI systems without compromising security. The method is based on using MCP connectors in terms of data integration and data governance tools to control access, thus ensuring that AI systems can access enterprise data without compromising security [5].

Jones (2018) analyzed the influence of AI voice assistants on marketing approaches and how people interact with technology. Through qualitative case studies, Jones discovered that the change is happening since people no longer want to use screens for communication but are rather interested in talking using their voices. It became evident that personalization is what helps in engaging people, despite data privacy issues still being a problem [6].

Todericiu (2025) A broad overview of the architectures of virtual assistants and the interaction models based on them. As a result of literature analysis, there has been an apparent shift towards the usage of large language models as the core architecture of contemporary virtual assistants.

It was shown that hybrid models work better than monolithic models; however, the study highlighted the absence of a standard evaluation metric within the field [7].

Malodia et al. (2021), researchers attempted to understand the real reasons behind the adoption of voice assistants powered by AI in our everyday lives. Conducting a mixed-methods analysis of the problem, scientists identified several key motivations. Convenience and usefulness of voice assistants were among the strongest motivational factors; yet, less obvious drivers of users' behavior were also discovered, such as social and affective aspects, which play an important role when dealing with voice assistants [8].

Mageira et al., AI chatbots can be used in an educational setting for the purposes of learning languages. In their research, these authors conducted a number of field experiments in actual educational settings and established that it is possible to use chatbots to enhance interactive learning and offer feedback to students. However, they also emphasized that NLP technology has significant constraints that may affect its application within an education setting [9].

Saklamaeva and Pavlič (2024) analyzed the integration of AI assistants into agile software development practices, especially in relation to the scaling of development teams. The systematic literature review conducted by the authors revealed some positive results, such as the use of AI assistants in planning, process automation, and predicting the results of projects. However, the authors did not ignore the challenges associated with the increased complexity of development and the dependence of teams on automated decision-making [10].

Lavanya and Sharada Varalakshmi (2019), a comparative study on text classification algorithms was done , who pitted classical machine learning techniques against the deep learning-based ones. From their findings, they favored the use of the latter since they proved to be superior compared to the former, especially when considering their ability to extract feature representations without the need for any form of human involvement [10].

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

III. METHODOLOGY

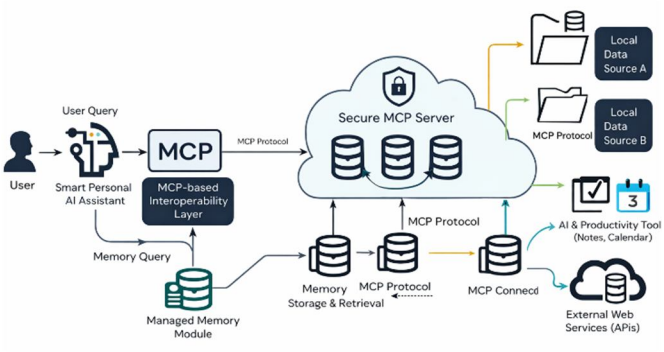


Fig.1: System Architecture

Fig.1, an application running on the host communicates to many data sources through the use of MCP Servers via a consistent protocol. The design does not depend on any predefined integration process; therefore, it will be able to connect to both locally stored systems and external APIs just as seamlessly, hence creating an architecture that is adaptable as well as scalable in the face of varying data requirements. The AI system being developed is called Action Oriented Personal AI Assistant which unifies many useful productivity features into a single application with MCP at the center of all its connections. The primary principle involved is simple: allow the AI system access to consistent channels of interaction with the outside world, then teach it how to interpret the user's requirements and carry them out.

The conversation begins with the user itself. The mode of interaction can be textual or vocal, in which case the vocal inputs are converted into text by the system. After the user inputs have been obtained, the artificial intelligence system analyzes them through natural language processing techniques to determine their intentions. Once the intentions of the user are understood, the next step is to determine what information and actions need to be taken in order to fulfill the requests.

For the sake of making the replies context-based and relevant, there is a context-aware memory unit in the system. This module takes into consideration all the past conversations with the user and their preferences in order to optimize the replies given to them and ensure that the same approach is used to complete tasks in the future, without having to start everything from scratch again. To execute the user's request, the system uses the MCP server and selects the appropriate tool for the job.

IV. IMPLEMENTATION

The core of the Action-Oriented Personal AI Assistant consists of a modular architecture where frontend, backend, AI computing, and tools communication are all integrated via the use of MCP. The modularity approach helps to ensure scalability, high performance, and independence of components from one another.

1) Frontend (User Interface): The front end is created using the React.js framework, and HTML, JavaScript, and Tailwind CSS take care of the remaining presentation layer. This choice of technologies simplifies creating a user interface that is not only mobile-friendly but also highly interactive. The input can be typed or dictated verbally by the users, while outputs, confirmation prompts, or notifications generated by the system are displayed instantly, without refreshing the web page.

2) Backend Framework: On the backend, the framework operates on the Node.js or Flask (depending on the specific use case) with RESTful APIs facilitating interaction among the components. In other words, the back end is responsible for taking the users' requests from the front end, processing the data, and maintaining consistency between the artificial intelligence model and the third-party services. In particular, the back end is the layer that ensures the system stability in multitasking environments.

3) Integration of Artificial Intelligence Model: A key part of the AI model in the chatbot is based on a language model by OpenAI with GPT. Thus, whenever a user submits a message to the system, the AI model processes it, deciphers the intention behind it, and decides what action to take. Moreover, another role of the model is generating replies in natural language.

4) Voice Input Processor: The module is able to take voice inputs by converting them into text via speech-to-text software such as the Web Speech API or the Whisper API created by OpenAI. Regardless of whether Web Speech API or Whisper API is utilized, both processes produce the same end result; the voice is converted into text that the remainder of the program can read just as if it were a regular text entry.

5) MCP for Tool Coordination: The MCP enables the AI model to extend its functionality beyond producing text alone. The MCP acts as the bridge between the AI and the outside world (e.g., external tools). It dictates when and which external tools will be invoked in order to achieve the user's objectives.
6) Task Management Tools: In terms of task management for simple productivity such as taking notes, setting reminders, managing the calendar, the assistant will be connected to a suite of integrated tools that can be accessed via the MCP platform. This suite covers the common functionalities that users might require for organizing themselves for the day.
7) External Tool Integration: In addition to connecting the user to a suite of tools, the assistant will be able to access web search and document processing APIs, allowing the assistant to access relevant data and assist users in working with documents. This includes tasks like summarization, retrieval, and generation of content using these documents.
8) Database and Context Management: All user information, interaction records, and context of the session are kept in the database hosted by Firebase either using Realtime Database or Firestore according to the need. It is precisely due to this persistence that the AI remembers certain things and maintains continuity of communication between the user and the assistant. Firebase's ability to synchronize data in real time means there will never be a need to manually refresh anything.
9) Authentication and Security: The user access authentication is provided by Firebase Authentication and JWT Token Validation. In combination, this guarantees that the interaction is done only by an authorized user. Ensuring security of any sort was considered essential as this system involves working with sensitive information such as the user's schedule or notes.

V. RESULTS

The development and evaluation of the Action-Oriented Personal AI Assistant proved that the idea was sustainable in real-world scenarios. In particular, in regard to automation of daily productivity-related activities, the solution did not excel because of its ability to perform a single task better, but rather due to the integration of several tools within a unified workflow and coherent design. This is illustrated below.

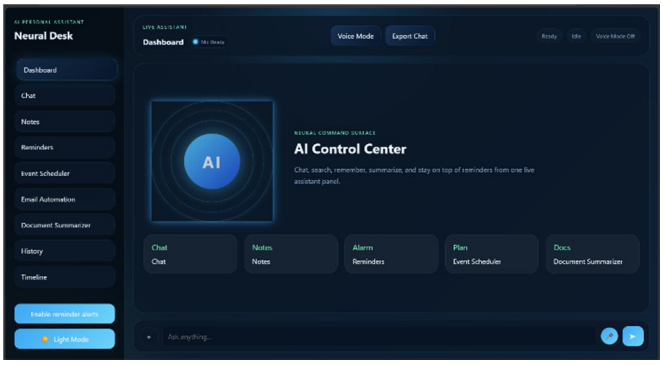


Fig.2: system dashboard overview

Fig.2 illustrates the primary control panel of the AI assistant, which is an integrated user interface for such functions as messaging, note-taking, scheduling, reminding, and document editing on a single screen.

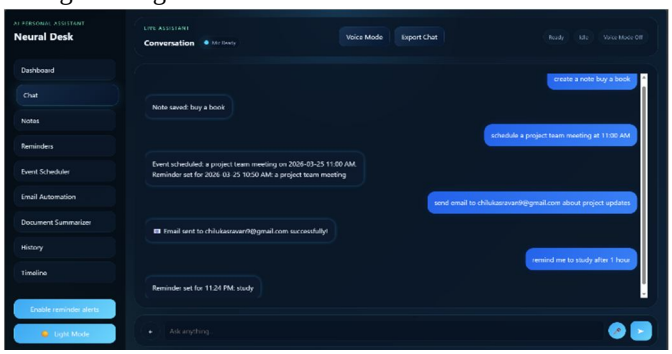


Fig.3: Neural Desk Chat Interface

Fig.3 presents the AI conversation interface, where users can send their requests, and the AI assistant executes the corresponding actions, including creating emails, scheduling events, and adding things to the calender.

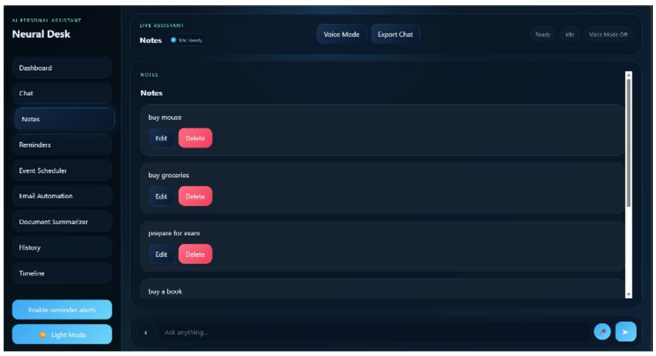


Fig.4: Notes Management Module

Fig.4 depicts the Notes feature available within the Neural Desk AI assistant. It is easy to use because notes are presented in a neat list and any item from the list can be edited or deleted by the user at any time. If people have been compiling their grocery lists or taking notes during classes, everything will be kept in one place and not create unnecessary clutter.

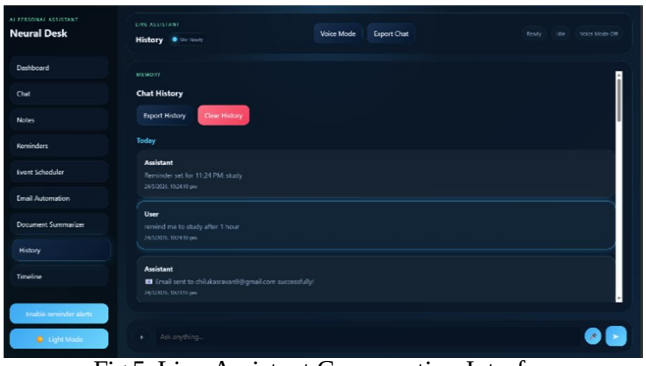


Fig.5: Live Assistant Conversation Interface

Fig.5 depicts the History feature available within the Neural Desk AI assistant, and it is used to present all past interactions in a simple chronological order. By using this feature, people can browse through previous chats, get information about sent messages or add reminders, and eventually save history as an additional file or remove it from the device entirely.

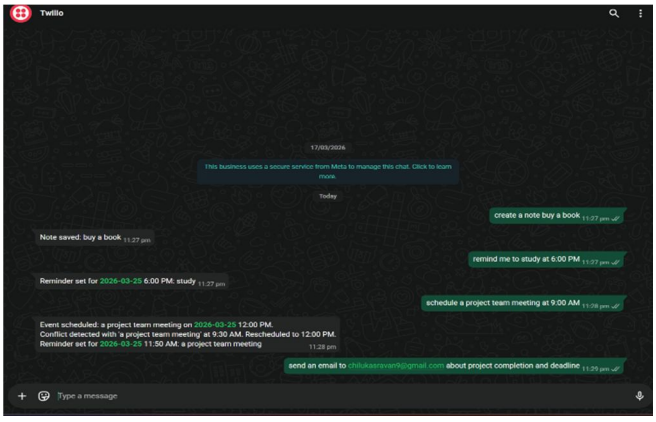


Fig.6: System WhatsApp Integration

Fig.6 Figure 6 depicts the messaging functionality of the assistant which is integrated via Twilio. If a user requests for example, a reminder or a new message to be sent, a corresponding action will be performed and the assistant will inform them about successful completion of that process.

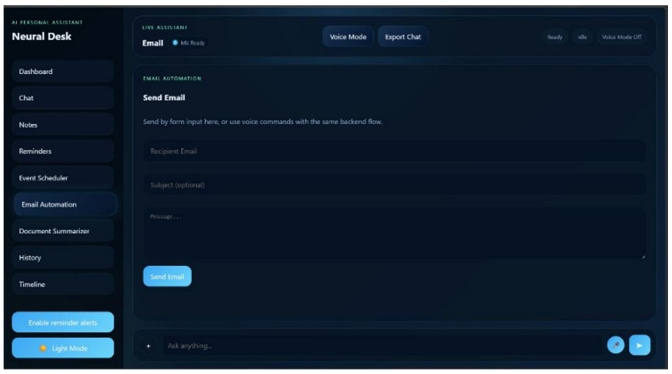


Fig.7: Email Automation Module

Fig.7 depicts the Email Automation module of the AI assistant, which allows for emails to be composed and sent either by typing or by using voice commands. The module has spaces to fill up with the name of the addressee, subject, and text of the email.

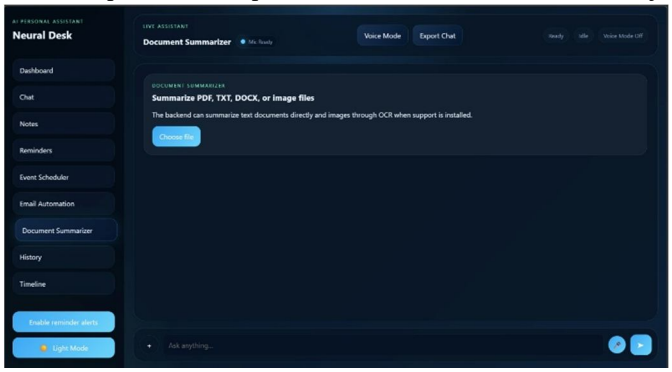


Fig.8: Document Summarizer Module

Fig.8 illustrates the Document Summarizer module, in which users can upload various documents including PDFs, TXT, DOCX, images and other files in order to have them automatically summarized. The use of AI processing technologies enables extracting the gist of documents, making the reading more efficient.

VI. CONCLUSION

In summary, this paper aimed to address an issue most people encounter in their everyday activities but pay little attention to it. Namely, the inconvenience that stems from the constant switching between various applications to accomplish routine tasks. The action-oriented approach of the proposed personal AI assistant merges several tools in one system that enables users to complete a variety of tasks in one window. It utilizes MCP as the mechanism to allow secure communication between the AI and external services such as Google Calendar, Weather Underground, Reminder List, etc.

Furthermore, it provides users with the option to control their work environment in a more effective manner through text and voice messages without adjusting to the peculiarities of particular applications. Finally, the fact that MCP is employed in the creation of this application creates grounds for further development.

There is potential for improvement in this field, including a greater diversity of tools being included, increasing the system's security, and making it even more personalized for every user. In conclusion, what this study shows is that the combination of robust AI and efficient tooling can lead to successful and highly practical assistants.

VII. FUTURE SCOPE

At present, the Action-Oriented Personal AI Assistant is quite sufficient in terms of its potential to automate daily processes. However, numerous modifications may be applied to improve the performance of the software. First and foremost, one should extend the number of available applications and services to be managed within the framework of a single interface. At the moment, the software concentrates mainly on enhancing productivity; however, other spheres such as finance, health, and business could greatly benefit from the development of an efficient and intelligent system that integrates different processes.

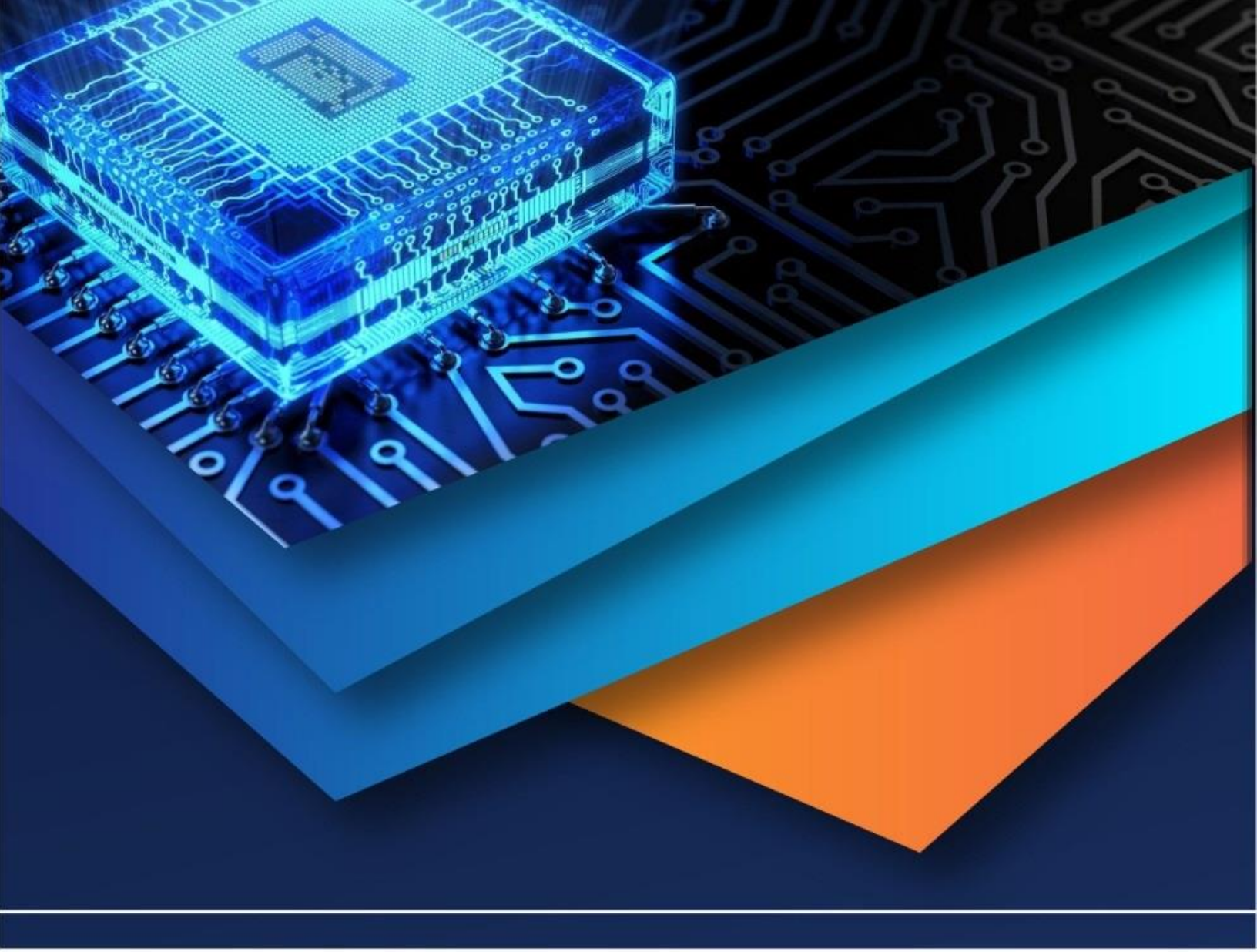
Furthermore, one can invest additional resources into the development of a new generation of the memory module. The currently existing module proves to be effective enough; however, the introduction of contextual analysis and anticipation features will enhance the level of personalized service. Thus, the software will become able not only to analyze the user's past experiences and actions but also to anticipate his or her future desires.

Security will always remain an issue that needs to be considered, especially as the integration of the assistant with MCP becomes increasingly complicated. Better authentication methods, stricter access restrictions, and end-to-end encryption are some of the features that should definitely be included in the development plan if the assistant deals with personal and confidential information in different external applications. But most importantly, these modifications will contribute to establishing trust relationships with users who will start using the system not only for entertainment but also for other critical purposes.

Lastly, accessibility may serve as another promising aspect for future research. Making the assistant available for people speaking different languages and installing it on various operating systems will attract many more users. With the development of IoT and wearables, it seems logical to explore how the assistant can be integrated into their ecosystem.

REFERENCES

[1] Hou et al., “Model Context Protocol: Landscape and Security Challenges,” 2024.
[2] Hasan et al., “MCP at First Glance: Security and Maintainability Study,” 2024.
[3] Zhao et al., “Parasitic Toolchain Attacks in MCP,” 2025.
[4] A. Singh, “Survey of Model Context Protocol,” 2024.
[5] Snowflake, “Managed MCP Servers for Secure Data Agents,” 2025.
[6] V. K. Jones, “Voice-Activated Change in AI Assistants,” 2018.
[7] I. A. Todericiu, “Virtual Assistants: A Review,” 2025.
[8] Malodia et al., “Adoption of AI Voice Assistants,” 2021.
[9] Mageira et al., “Educational AI Chatbots,” 2022.
[10] Saklamaeva & Pavlič, “AI Assistants in Agile Development,” 2024.
[11] M. P. Lavanya and S. Varalakshmi, “A Comprehensive Survey on Text Classification Using Machine and Deep Learning Mechanisms,” International Journal of Research and Analytical Reviews (IJRAR), vol. 6, no. 2, pp. 178–184, 2019.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY