robotics

MDPI

*Article*

# Revolutionizing Social Robotics: A Cloud-Based Framework for Enhancing the Intelligence and Autonomy of Social Robots †

**Abdelrahman Osman Elfaki [1], Mohammed Abduljabbar [2], Luqman Ali [2], Fady Alnajjar [2,*], Dua'a Mehiar [2], Ashraf M. Marei [1,3], Tareq Alhmiedat [1,3] and Adel Al-Jumaily [4,5]**

[1] Faculty of Computers & Information Technology, University of Tabuk, Tabuk 71491, Saudi Arabia
[2] Department of Computer Science and Software Engineering, College of Information Technology, United Arab Emirates University (UAEU), Alain 15551, United Arab Emirates
[3] Artificial Intelligence and Sensing Technologies (AIST) Research Centre, University of Tabuk, Tabuk 71491, Saudi Arabia
[4] Faculty of Engineering, University of Technology Brunei, Bandar Seri Begawan BE1410, Brunei
[5] École Nationale Supérieure de Techniques Avancées (ENSTA) Bretagne, French State Graduate, 29200 Brest, France
* Correspondence: fady.alnajjar@uaeu.ac.ae

† This paper is an extended version of our paper published in Abduljabbar, M. and Alnajjar, F. Web Platform for General Robot Controlling system. In Proceedings of the 2022 International Conference on Electrical and Computing Technologies and Applications (ICECTA), Ras Al Khaimah, United Arab Emirates, 23–25 November 2022, doi:10.1109/ICECTA57148.2022.9990192.



**Citation:** Elfaki, A.O.; Abduljabbar, M.; Ali, L.; Alnajjar, F.; Mehiar, D.; Marei, A.M.; Alhmiedat, T.; Al-Jumaily, A. Revolutionizing Social Robotics: A Cloud-Based Framework for Enhancing the Intelligence and Autonomy of Social Robots. *Robotics* **2023**, *12*, 48. https://doi.org/10.3390/robotics12020048

Academic Editor: Thierry Chaminade

Received: 27 February 2023
Revised: 17 March 2023
Accepted: 22 March 2023
Published: 24 March 2023



**Abstract:** Social robots have the potential to revolutionize the way we interact with technology, providing a wide range of services and applications in various domains, such as healthcare, education, and entertainment. However, most existing social robotics platforms are operated based on embedded computers, which limits the robot's capabilities to access advanced AI-based platforms available online and which are required for sophisticated physical human–robot interactions (such as Google Cloud AI, Microsoft Azure Machine Learning, IBM Watson, ChatGPT, etc.). In this research project, we introduce a cloud-based framework that utilizes the benefits of cloud computing and clustering to enhance the capabilities of social robots and overcome the limitations of current embedded platforms. The proposed framework was tested in different robots to assess the general feasibility of the solution, including a customized robot, "BuSaif", and commercialized robots, "Husky", "NAO", and "Pepper". Our findings suggest that the implementation of the proposed platform will result in more intelligent and autonomous social robots that can be utilized by a broader range of users, including those with less expertise. The present study introduces a novel methodology for augmenting the functionality of social robots, concurrently simplifying their utilization for non-experts. This approach has the potential to open up novel possibilities within the domain of social robotics.

**Keywords:** social robots; cloud-based; teleoperation; remote-control; intelligence; autonomy

## 1. Introduction

Social robots are increasingly being used in various domains, such as healthcare [1], education [2], business [3], telepresence [4], entertainment [5], and assistance [6]. They have the potential to provide social, behavioral, emotional, and cognitive support to people with diverse characteristics [7]. Furthermore, they can function as autonomous tools to support psychological health interventions [8]. The goal of social robots is to communicate and interact with humans in a manner that is socially acceptable and easily perceptible by humans [9]. In order to achieve their goals, social robots typically interact with several users for long periods of time [10]. In most applications of social robots, two primary constraints limit their ability to handle sophisticated human–robot interactions: one is related to the robot's hardware and software capabilities, and the other is related to its usability by non-experts. In the former case, all knowledge and control of the robot are housed within its onboard computers, which have limited computational memory and processing capacity. These limitations impede the ability of certain social robots to execute advanced AI algorithms, such as facial recognition [11], navigation [12], Natural Language Processing (NLP) [13], behavior analysis [14], etc. [15]. For the latter case, constraints arise from the complexity of the programming, which may make it difficult for non-experts to interact and operate the robots [16]. This can limit their potential use cases and make them less accessible to a broader range of users [17]. Programming complexity can also make it challenging to update and maintain the robot for extended use, leading to additional costs and a lack of flexibility. To overcome these challenges, there is a growing need for more user-friendly programming interfaces and practical tools to interact with social robots.

Cloud robotics (CR) is a field of robotics that attempts to invoke cloud technologies such as cloud computing, cloud storage, and other Internet technologies, to enhance the capabilities of robots [18,19]. This means that instead of all the necessary information and resources being stored and operated on the robot itself, they can be accessed through the internet using cloud computing. CR integrates robotic systems with cloud-based services such as storage, computation, and data processing to improve performance and functionality. This can enable robots to process more data, perform more complex tasks, be tailored to individual user needs, and make more informed and personalized decisions. Cloud resources allow for more powerful computing resources, access to big data, and collective learning. Additionally, cloud robotics enables remote monitoring, control, and maintenance of robots and makes deployment in different locations possible. Therefore, CR makes it possible for a single robot with limited computational capabilities to utilize sophisticated AI models directly from the cloud. In short, cloud robotics allows robots to become more intelligent, flexible, scalable, and autonomous by leveraging the power of cloud computing and IoT [20–22]. Web-based robotics (WBR), on the other hand, can be considered as a subfield of cloud robotics. In WBR, robots can be controlled and monitored through web interfaces and can receive instructions and transmit data through web protocols such as HTTP and WebSocket. WBR allows for remote monitoring and control of the robot and allows multiple users to interact with the robot simultaneously. In this study, we combine both CR and WBR technologies to design a dynamic platform to control a variety of robots.

The article is structured as follows: related works are discussed in Section 2, the proposed platform is described in Section 3, experimental results are shown in Section 4, and, finally, the discussion and the conclusion are in Sections 5 and 6, respectively. The key contributions of this study are the proposed platform and the demonstration of its effectiveness in enhancing the capabilities of social robots.

## 2. Related Works

The field of cloud robotics has emerged as a way to enhance the capabilities of robots by utilizing the resources provided by cloud computing. This integration allows robots to access data, services, and other resources from remote locations, thus overcoming the limitations imposed by their onboard resources. Since the introduction of the "cloud robotics" term, various models for connecting robots to the cloud have been developed and investigated [23–28]. For instance, in [29], a customized monitoring interface for robots with a high degree of freedom was introduced. Those robots can be accessed through desktop and mobile devices. Their proposed interface utilizes standard web technologies and provides a 3D visualization of the robot's position and sensor data. The platform is user-friendly, with no software installation required, and it can be deployed on a cloud for scalability to support multiple viewers. The introduced platform has been successfully tested on multiple robots and browsers. Similarly, in [30], the author proposed a cloud robotics architecture for the Simultaneous Localization and Mapping (SLAM) problem by distributing computational tasks and data among remote servers, freeing robots from computational loads using the Hadoop framework and FastSLAM2.0. The proposed framework resulted in an accurate real-time performance for single-robot navigation.

Authors in [31] have introduced XBotCloud, a component of the XBot framework that allows robots to access cloud resources and perform tasks with real-time execution and communication performance. The proposed platform addresses security challenges and has been tested on multiple robotic platforms for performance and flexibility. In [32], the authors demonstrated the advantages of utilizing cloud-based applications in robotics, specifically for the purposes of face detection and identification. They introduced an autonomous platform-switching algorithm that combines a Parallax Arlo Robot System with cloud computing, leveraging the cloud as the primary source of computational power and energy. To optimize performance and enhance endurance while minimizing hardware requirements, they designed an execution controller scheme. The system features computer vision for human-target navigation and identification, as well as an interactive procedure for face-to-face interaction with the target user. The switching algorithm allows the robot to switch between available resources (robot and cloud VM) based on the network and battery statuses. In practical implementation, the introduced switching algorithm and cloud-aided system improve system performance and longevity. The RC-Cloud Robot System, as proposed in [33], is designed for cloud-based multitasking, such as real-time face detection and recognition, by connecting to cloud computing resources, big data, and sharing information among clients. The proposed framework also utilizes ROS to support distributed computation on two local computers. The experimental results demonstrate the efficient execution of computation-intensive tasks for robot clients. In [34], a low-cost robot system with an embedded Android phone in a ROS environment was proposed to reduce hardware complexity and improve reproducibility in robotics development, tested using two Android phones, a laptop, and a robot with an Arduino microcontroller. The results showed advantages in incorporating smartphone processing and sensing as well as heavy external processing.

Several research studies have also investigated modular frameworks for social robots to simplify the programming of high-level modules, reducing the workload for non-expert programmers [35]. AMIRO, introduced by [36,37], is a modular architecture consisting of independent components connected through a planning unit. The modularity of the system allows each module to be run individually, as well as easily be replaced. Each component must implement a specific set of methods corresponding to the tasks it can execute. The framework includes modules for navigation, vision, communication, and planning, and it can be expanded to include dialogue capabilities, ambient intelligence sensors, and a Beliefs Desire Intent architecture implemented over ROS for improved planning and acting capabilities. The framework was validated in lab settings using a Pepper robot. Coronado et al. [38] proposed a modular framework that moves the robotics architecture of social robots from easy-to-use to easy-to-develop, enabling novice users to create complex intelligent robot systems without coding. The framework was successfully validated on a NAO robot. The Robot Interfaces from Zero Experience (RIZE) framework [39] is a cross-platform system that allows for the generation of intelligent authoring behaviors for various commercialized robots. It is built on NodE Primitive NEP, which supports distributed and modular systems using different middlewares and communication patterns. RIZE uses a modular approach based on independent behaviors that can be reused in other RIZE-based programs.

Microsoft Azure has been utilized to power autonomous robots and AI applications, such as natural language processing, robot-powered process automation, and machine learning [40–42]. Therefore, the global cloud robotics market is predicted to grow significantly, reaching $157.8 billion by 2030 [43]. Furthermore, collaborations between companies such as Ansys and Microsoft have been established to integrate simulation tools, Azure cloud, HPC, and digital twin technologies in cloud engineering initiatives [44]. Based on those facts, the adoption of Azure offers several advantages for cloud robotics, including cost optimization, flexibility, scalability, enhanced security, simplified management and maintenance, high availability, and quality services with a unique design and a high level of privacy [45–48].

Examples of such implementation include "robotic quarterback", Davidovich Group's use of Azure 3D and AI in the cloud for inventory management, and ABB's implementation of Azure cloud technology to replace traditionally human-led roles [49–51]. In [52], for instance, the authors tested data linking the NAO robot and Microsoft Azure as IoRT network servers to enable two-way communication and found it to be feasible. On the other hand, in [53], the authors explored the concept of cloud-edge hybrid robotic systems for physical human–robot interaction using MS Azur.

The field of cloud robotics is still developing, and there is a lack of consensus on the tools and technologies used in creating these systems. Additionally, a user-friendly system remains a challenge, as many proposed platforms utilize advanced ROS systems, which may require a high degree of technical expertise for use and maintenance [52]. Moreover, there may be limitations in terms of scalability and compatibility with a wide range of robot types and their hardware and software configurations [53]. The present investigation proposes a simple cloud robotics framework that harnesses the advantages of cloud computing to augment the functionality of social robots. This framework aims to surmount the inherent limitations imposed by these robots' hardware and software while enhancing their ease of use.

## 3. Proposed Platform

In the development of our platform, we have established four primary tenets to guide its implementation. These include (1) ease of use for the operator; (2) modularity to facilitate parallel and incremental advancement of capabilities; (3) adaptability to integrate with a range of robots and cloud systems; and (4) interactivity and versatility. In the following sections, we provide in-depth each of these principles and how they were integrated into the system's design.

### 3.1. Ease of Use of Operators

Our platform uses a simple graphical user interface (GUI) to facilitate effective system operation by operators with limited programming knowledge, Figure 1. Our user-friendly GUI can reduce the system's complexity and provide an intuitive interface. The following highlights the functionality of the main keys on the GUI.

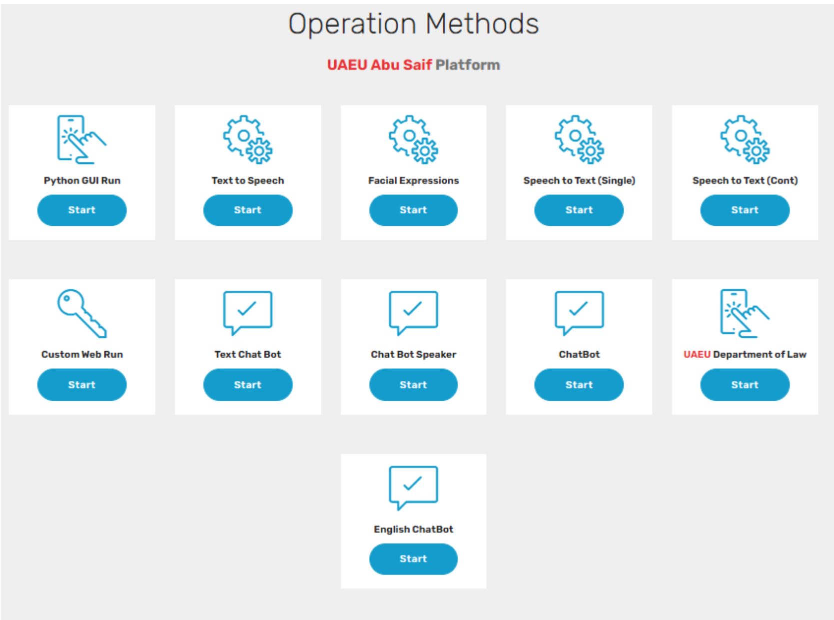


**Figure 1.** GUI of the proposed platform.

- Text to speech: A feature allowing for the input of Arabic text from the operator, which is then synthesized into speech by the AI agent speaker.
- Facial Expression: The capability to send commands to the facial expression AI agent to display emotions on the robot such as Happy, Sad, and Angry.
- Speech to text (continuous): A testing tab that led to the concurrent and continuous speech recognition and dynamic display on the web for the "Chatbot tab".
- Custom web run: A testing tab to ensure web operation.
- Text chatbot: A testing tab for the chatbot, which takes operator text input and displays the chatbot response on the interface.
- Chatbot speaker: A testing tab for the speaker AI agent and chatbot.
- Chatbot: A tab that continuously recognizes speech from the microphone AI agent and synthesizes the chatbot response using the speaker AI agent in Arabic.
- UAEU department of law: A test case for the robot to deliver information on the degree programs and other details for the UAEU Department of Law. The operator selects the desired information, and the robot then describes and clarifies it.
- English chatbot: A chatbot tab that continuously recognizes speech from the microphone AI agent and synthesizes the chatbot response using the speaker AI agent in English.

The backend of the GUI is crucial for guaranteeing the robustness and stability of the system, as it supports the various components and processes that make up the user interface, Figure 2. In our framework, the GUI's backend was powered by a suite of programming languages, including:

- PHP, which enabled server-side programming and facilitated functionality such as operator login/logout, admin sessions, access control, and task-loading screens.
- MySQL, which enabled database querying and manipulation for the robot chatbot and the admin's user credentials.
- Python, which provided standalone packages and AI agents specifically for the robot.
- JavaScript (client-side), which enabled browser-side programming and dynamic browser display.
- HTML, which was used for designing the user interface and theme.
- Microsoft Windows Bat Script, which allowed for the execution of operating system commands and the concurrent launching of processes across different programming languages.
- NODEJS, a JavaScript server that supported web sockets and the dynamic sending of data from other agents.
- GO Lang, which was used as an experimental system-side concurrent processing language for the purpose of achieving faster execution speeds and improved performance.

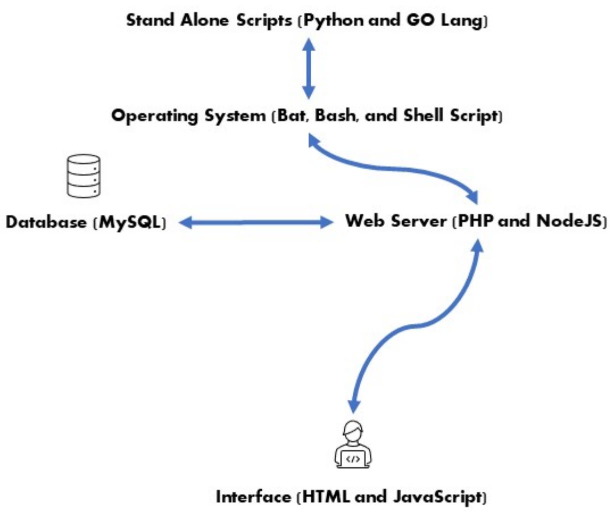


**Figure 2.** The backend of the GUI.

### 3.2. Modularity for Parallel, Incremental Capability Growth

To build a robust framework, we have utilized the principles of the design science research process (DSRP) model [54]. The DSRP model was selected for its ability to offer a structured methodology for the design development and evaluation of design artifacts, focusing on creating practical solutions to real-world issues. We have chosen the spiral development approach (SDP) for software development with a modular design for each prototype. The SDP involves iteratively creating prototypes with modular components to create flexible, scalable, and easily maintainable software. The offered framework aims to manage the modular functional cloud for robotics operations by treating each sensor as an individual AI agent and generating sub-AI agents as required for each task. The AI agents will communicate with the central "Meta AI agent," which will prioritize and make operational decisions, Figure 3. The framework design will mainly comprise the following components: Meta AI agent, Camera AI agent, GPS AI agent, Battery AI agent, Microphone AI agent, Speaker AI agent, and Navigation AI agent. Additional AI agents can be added as necessary to accommodate the specific needs of individual robots. For example, in the case of the BuSaif robot, two additional agents were required: A Facial Expression agent and a Conversation agent.

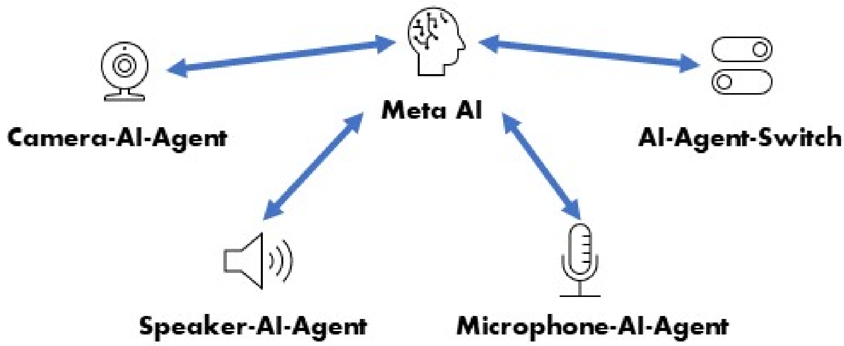


**Figure 3.** Meta AI Design.

At the initial stage, our framework was designed to be operating-system agnostic; thus, it can be used with both Windows and Linux systems. This is achieved by using general operating system commands that can execute programming language scripts, as shown in Figure 4. The commands will run Windows Batch Script on Windows, shell or bash script on Linux, and Android terminal on Android. Each web platform will have a homepage for the robot and a login button for the robot's web control panel, allowing the operator to access and control the robot.

The evolved platform design involves constructing a web platform on a cloud device that is equipped with a public IP and has the ability to control public ports. This platform connects directly to a remote robot using socket technologies such as WebSockets and HTTP POST/GET protocols, facilitating the operations of the robot.

At the final stage, we outlined the development of a prototype for cloud and general design platforms for robots in a laboratory setting, with a specific focus on the cognitive assessment robot BuSaif. The spiral development life cycle was employed to guide the platform's development, and the following stages were undertaken. During the first stage, a different approach was employed for the general AI agent, with a unique design for each sensor. The operating system was utilized to launch processes concurrently and await commands through sockets or WebSockets. In the second stage, improvements were made to the cache memory and security. Measures such as encrypting credentials, hiding all executable and operating system codes from the web server, and adding a "secret" to executable codes that would only execute if the correct secret were taken. The third stage involved the integration of the server-side framework MDD (Model, Database, and Display). Procedural code was transformed into functional and object-oriented code. During the fourth stage, all robot operational tabs were transformed into the Goal design of multi-AI agents and a meta-AI. The Go language was adopted, and Go routine concurrent channels were utilized for the AI agent, instead of solely relying on the operating system concurrent and asynchronous background processes. Once the multi-AI agent design and Meta AI were in place, most of the processing was transferred to a Kubernetes cluster (K3S/K8S), a mini cloud with 32 ARM CPU Cores and 64 GB RAM. This cluster was used to scale up the robot's operational capacity while processing the AI agent actuation on the robot.

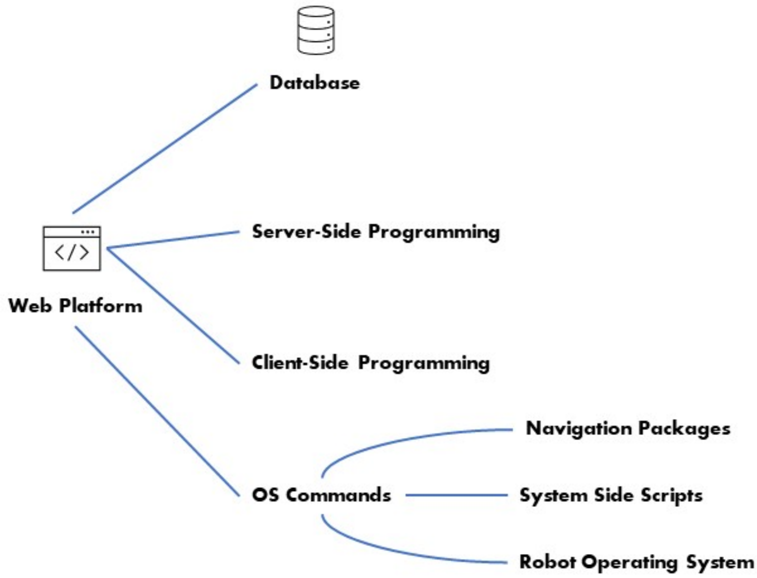


**Figure 4.** General Design.

### *3.3. Adaptability to Robots and Cloud Systems*

The introduced framework enables remote access to shared and dispersed computing resources over the internet. The platform is designed to be compatible with commonly used commercial cloud services providers such as Google Compute Engine and Microsoft Azure. Alongside the cloud infrastructure, the system utilizes a low-level component that can accommodate various robots, including the custom-made robot "BuSaif" as well as commercially available robots such as "Pepper," "NAO," and "Husky," all of which have undergone successful testing. The cloud infrastructure comprises high-performance servers, proxy servers, and databases, providing enhanced processing and storage capabilities. A diagram illustrating the architecture of cloud robotics systems and highlighting some of their applications is provided in Figure 5.

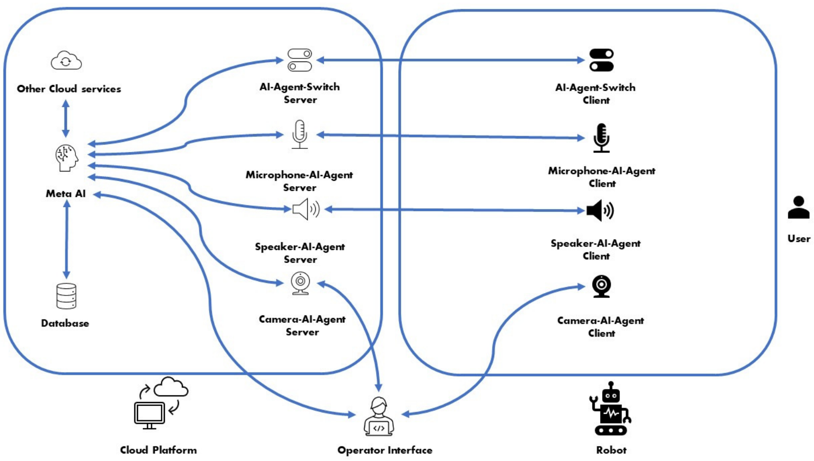


**Figure 5.** An overview of the proposed platform.

### 3.4. Interactivity and Versatility

The platform has been employed in three distinct scenarios. The first application involved the use of the BuSaif Robot in a cognitive assessment research project. The robot was equipped with various AI agents to detect and recognize individuals' faces and emotions, while also being capable of conducting communication sessions with both healthy individuals and those with cognitive impairments. Despite possessing only average programming skills, the operators were able to create the experimental scenario with ease. The second occasion involved the robot delivering opening speeches for three local conferences and interacting with guests by answering their questions about the event and program. The robot operators had no programming knowledge and were able to develop the scenario after a brief two-day training session. The last application involved the robot welcoming school students during their visit to the lab. In all of these instances, the BuSaif robot, owing to the developed platform, was able to leverage a broad spectrum of AI features, enabling it to effectively communicate with the public and exhibit advanced capabilities.

In order to ensure safe interactions with the robots, several standard safety features were considered. These features include the provision of an emergency stop button, collision detection and avoidance sensors, a safe movement speed, and secure data-handling mechanisms. By incorporating these safety features, operators and users can reduce the risk of accidents or injuries and ensure safe interactions with our social robots.

Our proposed cloud robotic platform incorporates all of the primary tenets discussed above and includes the following elements:

1. Robots: Our platform supports a wide range of robots, including BuSaif, Pepper, NAO, and Husky.
2. Cloud computing infrastructure: We utilize Google and Microsoft Azure as well as a database for our cloud computing needs.
3. Communication networks: Our platform supports various communication networks, including sockets, web sockets, HTTP post and get, as well as peer-to-peer media streaming through AI agents such as Microphone and Camera.
4. Control and management software: We use Meta-AI as our control and management software, as depicted in Figure 2.
5. User interfaces: Our platform provides user interfaces, as illustrated in Figures 2 and 3.

## 4. Results

### 4.1. Performance Analysis of a Cloud-Based Platform vs. a Local Platform for Robot Control

In this section, we present an analysis of the performance of the proposed cloud-based platform versus the local platform for controlling the BuSaif robot. The analysis includes the assessment of various aspects such as memory, speed, processors, and functionality. The originally used local PC, embedded with the BuSaif robot, served as the baseline for comparison. At stage 1 of the proposed design, the web platform was designed with a modular approach that divides each operation into a tab on the web platform control panel. Each operational tab is unique, and each tab launches multi-lingual programs concurrently and asynchronously via the operating system in the background. These operations communicate through sockets, web sockets, HTTP POST, and HTTP GET. The modular design of tabs simplifies the implementation, allowing for the ease of integrating new tabs for additional operations. The multi-processing design of the web platform enables the robot to scale up for heavy operations and ultimately communicate with the robot AI agents while processing all functions in a K3S/K8S cluster on the cloud.

Table 1 demonstrates an overview of the structure comparison between the proposed and local platforms. Table 2 demonstrates an overview of the comparison results between the cloud-based and local platforms. The comparison revealed that the cloud-based platform outperformed the local platform in terms of speed and memory usage. The modular design of the proposed platform and the use of multi-lingual programs led to improved functionality and scalability, as evidenced by the ability to process complex operations and communicate with the robot AI agents. The proposed platform's multi-processing design also allows for distributed cloud computing, which is a significant advantage over the local platform's single-process approach.

**Table 1.** Structure Comparison of Conventional, Web-based, and Cloud-based platforms.

| Components | Robot Application | Database | Networking |
|---|---|---|---|
| Conventional | Robot Computer | Filesystem | Stand Alone GUI |
| Web-Based | Intranet | MySQL database | HTTP, Sockets, Web-Sockets |
| Cloud-Based | Internet | Clustered MySQL database | HTTP, Sockets, Web-Sockets |

**Table 2.** Performance Comparison of Conventional, Web-based, and Cloud-based Platforms.

| Index | Python GUI | General Design | Goal Design (Meta AI) |
|---|---|---|---|
| Type | Stand-Alone | Web-Platform | Cloud-Platform |
| Process | Single | Concurrent | Concurrent |
| Start-Time | 9 s | 1 to 4 s | Depends on cluster specification |
| Mic | Recognize-Once | Continuous | Continuous |
| Chatbot | Folder | Database | Cluster-Database |
| Speaker | Synthesis | Socket-Synthesis | Compiled Language |
| Camera | Tensorflow-Caffe | Go-Tensorflow JS | Depends on cluster specification |

This study provides evidence that a cloud-based robotics platform can be a viable alternative to a local platform for robot control, particularly in terms of speed, memory usage, functionality, and scalability. The modular design of the proposed platform and its multi-processing capabilities offer significant advantages over the local platform's single process approach. Future work should focus on optimizing the cloud platform further to improve its performance and usability.

The cloud-based framework has undergone testing on commercially available robots, namely the "Husky", "NAO", and "Pepper". Although a comparative analysis similar to Table 2 could not be conducted due to insufficient specification information, which depends on the robot version, it is worth noting that these robots can leverage the cloud to gain access to advanced features such as updated object detection models, facial expression analysis, cognitive capabilities [55], and the new chatGPT API for advanced natural language processing capabilities in both English and Arabic [56].

### 4.2. Responding Time

The initialization duration of the conventional Python graphical user interface (GUI), i.e., local PC, is approximately 9 s, with the initial 3 s utilized for the loading of the Python interpreter and its dependent classes, including Chatter, Face, Communicator, Eye, and Tracker. The remaining 6 s are mainly attributed to loading the TensorFlow and Caffee models. On the other hand, our cloud-based platform features a modular design that enables the authentication of administrators to the control panel in less than a second. The platform launches multi-lingual processes concurrently in the background per tab, ranging between 1 to 4 s, depending on the tab. The design objective is not solely to rely on multiprocessing and concurrency but also to avoid the use of interpreted languages by employing compiled languages such as C++ and Go, which provide faster-compiled machine code execution speeds instead of a slower interpretation during run time. Furthermore, Go, which is designed with concurrency in mind, is a suitable candidate for this web platform, given its support for goroutines and communication channels.

### 4.3. Microphone AI Agent

The implementation of the microphone AI agent involves utilizing Microsoft Azure cognitive services' speech SDK for speech recognition. The algorithm employed entails utilizing a loop to invoke the SDK to recognize speech from the microphone and transcribe it into Arabic text. The loop allows for up to 11 s for phrase recognition, with the process continuing until a phrase is recognized. Upon detection, speech recognition ceases, and the identified words are transmitted to the chatbot component of the system. To enhance the efficacy of the Microphone AI agent, the cloud platform launches a concurrent background process that leverages Microsoft cognitive services' continuous recognition function to attempt speech recognition every 0.5 s. The process is replicated four to six times, following which the predicted phrase recognition is relayed to the subsequent AI agent, i.e., the language and chatbot agent.

This algorithm operates continuously without halting, and it persistently recognizes and predicts phrases as an independent concurrent process.

### *4.4. AI Based Chatbot Agent*

When utilizing the local PC on the robot, the classical Python GUI operates as a singular process that scans an intent folder containing labelled text files with phrases. It searches each labelled text file individually and calculates the probability of the phrase's occurrence for every file. The highest probability label is subsequently extracted from a response folder that also comprises labelled text files. From this folder, a phrase is randomly selected from a labelled text file and transmitted to the speaker agent. The presented platform follows the same principle but adopts a MySQL database for intent and response, which is populated with chatbot tables. The phrase in the cloud platform is conveyed from the microphone AI agent as an HTTP post to the web server, which searches the intent database for the phrase. If the phrase is detected, a random row is selected from the labelled response database and transmitted to the speaker AI agent for speech synthesis.

The updated platform features a database in a K3S/K8S Kubernetes cluster, with the phrase transmitted to the master node. The master node initiates a concurrent Go language routine, which divides the search. For instance, if there are 1000 tables in the intent database, the master node creates 20 concurrent Go routines, with each responsible for searching 50 tables. The master node then sends these requests to the chatbot service node port, distributing the load between the cloud processing units. The phrase is subsequently conveyed via socket and port to the next AI agent, i.e., the speaker's AI agent.

### *4.5. Speaker AI Agent*

The speaker AI agent employs the Microsoft cognitive services speech SDK to convert Arabic text phrases into sound for the speaker, utilizing the text-to-speech technology. Both the Python GUI and the web platform leverage the Microsoft speech synthesis function. The sole distinction is that the web platform already launches the speaker AI agent as a concurrent background process that listens on a socket and port to synthesize any message into speech. The advanced design follows the same principles as the cloud platform, but it utilizes a compiled language to boost the response speed.

### *4.6. Camera AI Agent*

The classical Camera AI agent employs the Python TensorFlow framework and Caffee model for object detection. Nevertheless, it is linked directly to the robot's operations owing to its high processing requirements. In contrast, the cloud platform's Camera AI agent is reliant on peer-to-peer media stream connection between the robot and operator's browser. It also utilizes TensorFlow JavaScript to load the model, which provides superior image processing speed.

## 5. Discussion

The current limitations of social robots can be attributed to two primary factors: limited computational power and difficulty to setup and operate for non-expert programmers. Due to their limited computational power, most existing robots are unable to perform complex algorithms, respond to multiple inputs, or understand and process complex human language or emotions. These limitations also impact the robot's ability to learn and adapt to new situations. For example, the NAO robot has a maximum of 2 GB of internal memory and a 1.6 GHz Intel Atom processor, which limits its ability to recognize advanced facial and speech expressions in real-time or perform trust inference based on unstructured data. To address these limitations, the utilization of cloud resources has been suggested as a potential solution, which facilitates the delegation of computational responsibilities to external servers by the robots. Consequently, this strategy can augment the robots' cognitive abilities, enabling them to conduct customized interactions in accordance with the individual users and environmental conditions. The cloud can enable robots to perform more challenging tasks such as object recognition and manipulation, social navigation, and advanced human–robot interactions. In addition to limited computational power, setting up and programming existing robots can also be challenging for non-expert programmers. The Robot Operating System (ROS), which is widely used to control robots, can be difficult to learn and use for novice users with little to no experience [57]. Real-time performance and setting up most robotic manipulation systems are also challenging that need to be addressed.

As technology continues to evolve, the limitations of social robots are expected to reduce, and a cloud robotic platform is proposed in this study that aims to increase the acceptance of robotic services by non-expert programmers. The proposed platform has the potential to facilitate the development of socially intelligent and user-friendly robots that can overcome the limitations associated with traditional stand-alone robots, which are difficult to program and have limited computational power. By leveraging cloud resources, the proposed platform aims to enable the creation of robots that are not only smarter but also more accessible to the public.

The cloud-robotics paradigm represents a promising approach for enhancing the efficiency, cost-effectiveness, and intelligence of robots via cloud computing capabilities [57]. However, the absence of a platform that can guarantee compatibility with a diverse range of robotics platforms and provide ease-of-use for non-expert users remains a challenge. Despite the emergence of various open-source and proprietary options, the research field of cloud robotics is still evolving, leaving ample opportunities for further exploration. In this study, we introduce a Meta AI cloud platform that is readily adoptable by many common social robots and is user-friendly for non-experts. Nevertheless, the platform's simplicity may limit its ability to handle complex robots, a trade-off that we deem acceptable given our target user base. Achieving both the handling of complex robots and non-expert usability is a challenging task that needs more explorations.

## 6. Conclusions

Social robots have the potential to revolutionize the way we interact with technology, providing a wide range of services and applications in various domains. However, most of the social robotics platforms are limited by their reliance on embedded computers, which limits their access to advanced AI-based services and online platforms. This study aims to enhance the capabilities of social robots by proposing a platform that utilizes the benefits of cloud computing and clustering. Our proposed platform was tested on a customized and commercialized robot, and the results indicate that the platform can handle both kinds of robots and allows for a user-friendly interface for general users. The implementation of our proposed platform has the potential to result in more intelligent and autonomous social robots that can be utilized by a wider range of users, including those with less expertise.

**Author Contributions:** Conceptualization, All authors; methodology, A.O.E. and M.A.; software, A.O.E. and M.A.; validation, M.A., L.A. and F.A.; formal analysis, A.O.E. and M.A.; investigation, M.A.; writing—original draft preparation, A.O.E., M.A. and F.A.; writing—review and editing, All authors; visualization, M.A.; supervision, F.A. and T.A.; project administration, A.O.E., F.A. and T.A.; funding acquisition, T.A. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by Deanship of Scientific Research (DSR) at the University of Tabuk, grant number 1441-105.

**Data Availability Statement:** Data sharing not applicable.

**Acknowledgments:** The authors would also like to acknowledge the financial support for this work from the Deanship of Scientific Research (DSR) at the University of Tabuk, Tabuk, Saudi Arabia, under grant no. 1441–105.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Guemghar, I.; de Oliverira Padilha, P.P.; Abdel-Baki, A.; Jutras-Aswad, D.; Paquette, J.; Pomey, M.-P. Social Robot Interventions in Mental Health Care and Their Outcomes, Barriers, and Facilitators: Scoping Review. *JMIR Ment. Health* **2022**, *9*, e36094. [CrossRef] [PubMed]
2. Alam, A. Social Robots in Education for Long-Term Human-Robot Interaction: Socially Supportive Behaviour of Robotic Tutor for Creating Robo-Tangible Learning Environment in a Guided Discovery Learning Interaction. *ECS Trans.* **2022**, *107*, 12389. [CrossRef]
3. Saari, U.A.; Tossavainen, A.; Kaipainen, K.; Mäkinen, S.J. Exploring factors influencing the acceptance of social robots among early adopters and mass market representatives. *Robot. Auton. Syst.* **2022**, *151*, 104033. [CrossRef]
4. Almeida, L.; Menezes, P.; Dias, J. Telepresence Social Robotics towards Co-Presence: A Review. *Appl. Sci.* **2022**, *12*, 5557. [CrossRef]
5. Lytridis, C.; Bazinas, C.; Kaburlasos, V.G.; Vassileva-Aleksandrova, V.; Youssfi, M.; Mestari, M.; Ferelis, V.; Jaki, A. Social Robots as Cyber-Physical Actors in Entertainment and Education. In Proceedings of the 2019 International Conference on Software, Telecommunications and Computer Networks (SoftCOM), Split, Croatia, 19–21 September 2019; pp. 1–6.
6. Giansanti, D. The Social Robot in Rehabilitation and Assistance: What Is the Future? *Healthcare* **2021**, *9*, 244. [CrossRef] [PubMed]
7. Park, C.H.; Ros, R.; Kwak, S.S.; Huang, C.-M.; Lemaignan, S. Editorial: Towards Real World Impacts: Design, Development, and Deployment of Social Robots in the Wild. *Front. Robot. AI* **2020**, *7*, 600830. [CrossRef] [PubMed]
8. Henschel, A.; Laban, G.; Cross, E.S. What Makes a Robot Social? A Review of Social Robots from Science Fiction to a Home or Hospital Near You. *Curr. Robot Rep.* **2021**, *2*, 9–19. [CrossRef]
9. Social Robot—An Overview | ScienceDirect Topics. Available online: https://www.sciencedirect.com/topics/computer-science/social-robot (accessed on 18 February 2023).
10. Kaptein, F.; Kiefer, B.; Cully, A.; Celiktutan, O.; Bierman, B.; Rijgersberg-peters, R.; Broekens, J.; Van Vught, W.; Van Bekkum, M.; Demiris, Y.; et al. A Cloud-Based Robot System for Long-Term Interaction: Principles, Implementation, Lessons Learned. *J. Hum.-Robot Interact.* **2021**, *11*, 1–27. [CrossRef]
11. Yan, H.; Ang, M.H.; Poo, A.N. A Survey on Perception Methods for Human–Robot Interaction in Social Robots. *Int. J. Soc. Robot.* **2014**, *6*, 85–119. [CrossRef]
12. Mavrogiannis, C.; Baldini, F.; Wang, A.; Zhao, D.; Trautman, P.; Steinfeld, A.; Oh, J. Core Challenges of Social Robot Navigation: A Survey 2021. *arXiv* **2021**, arXiv:2103.05668. [CrossRef]
13. Graterol, W.; Diaz-Amado, J.; Cardinale, Y.; Dongo, I.; Lopes-Silva, E.; Santos-Libarino, C. Emotion Detection for Social Robots Based on NLP Transformers and an Emotion Ontology. *Sensors* **2021**, *21*, 1322. [CrossRef] [PubMed]
14. Nocentini, O.; Fiorini, L.; Acerbi, G.; Sorrentino, A.; Mancioppi, G.; Cavallo, F. A Survey of Behavioral Models for Social Robots. *Robotics* **2019**, *8*, 54. [CrossRef]
15. Prescott, T.J.; Camilleri, D.; Martinez-Hernandez, U.; Damianou, A.; Lawrence, N.D. Memory and mental time travel in humans and social robots. *Philos. Trans. R. Soc. B Biol. Sci.* **2019**, *374*, 20180025. [CrossRef] [PubMed]
16. van Waveren, S.; Carter, E.J.; Örnberg, O.; Leite, I. Exploring Non-Expert Robot Programming through Crowdsourcing. *Front. Robot. AI* **2021**, *8*, 646002. [CrossRef] [PubMed]
17. Ajaykumar, G.; Steele, M.; Huang, C.-M. A Survey on End-User Robot Programming. *ACM Comput. Surv.* **2021**, *54*, 1–36. [CrossRef]
18. Bhattacharyya, S. What Is Cloud Robotics? Importance and Challenges | Analytics Steps. Available online: https://www.analyticssteps.com/blogs/what-cloud-robotics-importance-and-challenges (accessed on 18 February 2023).
19. Hu, G.; Tay, W.P.; Wen, Y. Cloud robotics: Architecture, challenges and applications. *IEEE Netw.* **2012**, *26*, 21–28. [CrossRef]
20. Lampropoulos, G.; Siakas, K.; Anastasiadis, T. Internet of Things in the Context of Industry 4.0: An Overview. *Int. J. Entrep. Knowl.* **2019**, *7*, 4–19. [CrossRef]
21. Wang, X.; Alhusin Alkhdur, A.; Wang, L. Cloud-Based Robotic System: Architecture Framework and Deployment Models. In Proceedings of the 25th International Conference on Flexible Automation and Intelligent Manufacturing, Wolverhampton, UK, 23–26 June 2015.
22. Turnbull, L.; Samanta, B. Cloud robotics: Formation control of a multi robot system utilizing cloud infrastructure. In Proceedings of the 2013 Proceedings of IEEE Southeastcon, Jacksonville, FL, USA, 4–7 April 2013; pp. 1–4.

23. Kehoe, B.; Patil, S.; Abbeel, P.; Goldberg, K. A Survey of Research on Cloud Robotics and Automation. *IEEE Trans. Autom. Sci. Eng.* **2015**, *12*, 1–12. [CrossRef]
24. Ichnowski, J.; Prins, J.; Alterovitz, R. Cloud-based Motion Plan Computation for Power-Constrained Robots. In *Algorithmic Foundations of Robotics XII: Proceedings of the Twelfth Workshop on the Algorithmic Foundations of Robotics*; Goldberg, K., Abbeel, P., Bekris, K., Miller, L., Eds.; Springer Proceedings in Advanced Robotics; Springer International Publishing: Cham, Switzerland, 2020; pp. 96–111. ISBN 978-3-030-43089-4.
25. Kappler, D.; Bohg, J.; Schaal, S. Leveraging big data for grasp planning. In Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 26–30 May 2015; pp. 4304–4311.
26. Ermacora, G.; Rosa, S.; Toma, A. Fly4SmartCity: A cloud robotics service for smart city applications. *J. Ambient. Intell. Smart Environ.* **2016**, *8*, 347–358. [CrossRef]
27. Russo, L.; Rosa, S.; Maggiora, M.; Bona, B. A Novel Cloud-Based Service Robotics Application to Data Center Environmental Monitoring. *Sensors* **2016**, *16*, 1255. [CrossRef]
28. Tang, S.; Wan, J.; Cai, H.; Chen, F. Cloud Robotics: Insight and Outlook. In Proceedings of the Industrial IoT Technologies and Applications, Guangzhou, China, 25–26 March 2016; Wan, J., Humar, I., Zhang, D., Eds.; Springer International Publishing: Cham, Switzerland, 2016; pp. 94–103.
29. Hilton, W.; Lofaro, D.M.; Kim, Y. A lightweight, cross-platform, multiuser robot visualization using the cloud. In Proceedings of the 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, Chicago, IL, USA, 14–18 September 2014; pp. 1570–1575.
30. Hammad, A.; Ali, S.S.; Eldien, A.S.T. A novel implementation for FastSLAM 2.0 algorithm based on cloud robotics. In Proceedings of the 2017 13th International Computer Engineering Conference (ICENCO), Cairo, Egypt, 27–28 December 2017; pp. 184–189.
31. Muratore, L.; Lennox, B.; Tsagarakis, N.G. XBotCloud: A Scalable Cloud Computing Infrastructure for XBot Powered Robots. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 1–9.
32. Rahman, A.; Jin, J.; Wong, Y.W.; Lam, K.S. Development of a cloud-enhanced investigative mobile robot. In Proceedings of the 2016 International Conference on Advanced Mechatronic Systems (ICAMechS) Melbourne, VIC, Australia, 30 November–3 December 2016; pp. 104–109.
33. Tian, S.; Lee, S.G. An implementation of cloud robotic platform for real time face recognition. In Proceedings of the 2015 IEEE International Conference on Information and Automation, Lijiang, China, 8–10 August 2015; pp. 1509–1514.
34. de A. Barbosa, J.P.; do P. de C. Lima, F.; dos S. Coutinho, L.; R. Rodrigues Leite, J.P.; Barbosa Machado, J.; Henrique Valerio, C.; Sousa Bastos, G. ROS, Android and cloud robotics: How to make a powerful low cost robot. In Proceedings of the 2015 International Conference on Advanced Robotics (ICAR), Istanbul, Turkey, 27–31 July 2015; pp. 158–163.
35. Coronado, E.; Mastrogiovanni, F.; Indurkhya, B.; Venture, G. Visual Programming Environments for End-User Development of intelligent and social robots, a systematic review. *J. Comput. Lang.* **2020**, *58*, 100970. [CrossRef]
36. Ghiță, A.S.; Gavril, A.F.; Nan, M.; Hoteit, B.; Awada, I.A.; Sorici, A.; Mocanu, I.G.; Florea, A.M. The AMIRO Social Robotics Framework: Deployment and Evaluation on the Pepper Robot. *Sensors* **2020**, *20*, 7271. [CrossRef]
37. Gavril, A.-F.; Ghita, A.S.; Sorici, A.; Florea, A.M. Towards a Modular Framework for Human-Robot Interaction and Collaboration. In Proceedings of the 2019 22nd International Conference on Control Systems and Computer Science (CSCS), Bucharest, Romania, 28–30 May 2019; pp. 667–674.
38. Coronado, E.; Mastrogiovanni, F.; Venture, G. Development of Intelligent Behaviors for Social Robots via User-Friendly and Modular Programming Tools. In Proceedings of the 2018 IEEE Workshop on Advanced Robotics and its Social Impacts (ARSO), Genova, Italy, 27–29 September 2018; pp. 62–68.
39. Coronado, E.; Deuff, D.; Carreno, P.; Tian, L.; Kulic, D.; Sumartojo, S.; Mastrogiovanni, F.; Venture, G. Towards a Modular and Distributed End-User Development Framework for Human-Robot Interaction. *IEEE Access* **2021**, *9*, 12675–12692. [CrossRef]
40. Mcnulty-Kowal, S. Microsoft's Azure Cloud Platform will Be the Brain for Their Future Autonomous Robots!—Yanko Design. Available online: https://www.yankodesign.com/2020/11/06/microsofts-azure-cloud-platform-will-be-the-brain-for-their-future-autonomous-robots/ (accessed on 18 February 2023).
41. AI and the Need for Purpose-Built Cloud Infrastructure. Available online: https://www.microsoft.com/azure/partners/news/article/ai-and-the-need-for-purpose-built-cloud-infrastructure (accessed on 18 February 2023).
42. Microsoft: We're Bringing ChatGPT to the Azure Cloud-Computing Service. Available online: https://www.zdnet.com/article/microsoft-were-bringing-chatgpt-to-the-azure-openai-cloud-computing-service/ (accessed on 18 February 2023).
43. Cloud Robotics Market Predicted to Grow to $157.8 Billion by 2030. Available online: https://roboticsandautomationnews.com/2020/07/09/cloud-robotics-market-predicted-to-grow-to-157-8-billion-by-2030/33909/ (accessed on 18 February 2023).
44. Robotics 24/7 News and Resources. Available online: https://www.robotics247.com/topic/tag/Microsoft_Azure (accessed on 18 February 2023).
45. Benefits of Cloud Migration | Microsoft Azure. Available online: https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/benefits-of-cloud-migration/ (accessed on 18 February 2023).
46. 4 Key Advantages of Microsoft's Azure Cloud. Available online: https://www.dyntek.com/blog/4-key-advantages-of-microsofts-azure-cloud (accessed on 18 February 2023).

47. Carutasu, G.; Botezatu, M.A.; Botezatu, C.; Pirnau, M. Cloud computing and windows azure. In Proceedings of the 2016 8th International Conference on Electronics, Computers and Artificial Intelligence (ECAI), Ploiesti, Romania, 30 June–2 July 2016; pp. 1–6.
48. Cepero, R. 7 Advantages of Choosing Microsoft Azure. Available online: https://bleuwire.com/advantages-choosing-microsoft-azure/ (accessed on 18 February 2023).
49. Pioneering "Robotic Quarterback" Uses Windows, Azure Ecosystems to Solve Developer Dilemmas—And Cut a Year Off Time to Market. Available online: https://customers.microsoft.com/en-gb/story/1496605524271652247-monarc-windowsiot-azure (accessed on 18 February 2023).
50. Robotics. Available online: https://devblogs.microsoft.com/azure-depth-platform/category/robotics/ (accessed on 18 February 2023).
51. ABB Case Study. Available online: https://partner.microsoft.com/ru-kz/case-studies/abb (accessed on 18 February 2023).
52. Rosly, M.A.; Miskam, M.; Shamsuddin, S.; Yussof, H.; Zahari, N. Data Linking Testing between Humanoid Robot and IoRT Network Server for Autism Telerehabilitation System Development. In *RITA 2018, Proceedings of the 6th International Conference on Robot Intelligence Technology and Applications, Kuala Lumpur, Malaysia, 16–18 December 2018*; Springer: Singapore, 2020; pp. 161–169, ISBN 9789811383229.
53. Tian, N. Cloud-Edge Hybrid Robotic Systems for Physical Human Robot Interactions. Ph.D. Thesis, University of California, Berkeley, Berkeley, CA, USA, 2020.
54. Peffers, K.; Tuunanen, T.; Gengler, C.E.; Rossi, M.; Hui, W.; Virtanen, V.; Bragge, J. Design Science Research Process: A Model for Producing and Presenting Information Systems Research 2020. *arXiv* **2020**, arXiv:2006.02763. [CrossRef]
55. Di Nuovo, A.; Varrasi, S.; Lucas, A.; Conti, D.; McNamara, J.; Soranzo, A. Assessment of Cognitive Skills via Human-Robot Interaction and Cloud Computing. *J. Bionic. Eng.* **2019**, *16*, 526–539. [CrossRef]
56. Microsoft Wants to Use ChatGPT to Control Robots through Language. Available online: https://www.infoq.com/news/2023/02/chagpt-robotics-microsoft/ (accessed on 11 March 2023).
57. Dawarka, V.; Bekaroo, G. Cloud Robotics Platforms: Review and Comparative Analysis. In Proceedings of the 2018 International Conference on Intelligent and Innovative Computing Applications (ICONIC), Mon Tresor, Mauritius, 6–7 December 2018; pp. 1–6.