การใช้งานคลาวด์คอมพิวติ้งและไอโอที

โดยการประเมินประสิทธิภาพ

IMPLEMENTATION OF CLOUD COMPUTING AND INTERNET OF THINGS

(lot) BY PERFORMANCE EVALUATION

จิรัณย์ สิทธิโยภาสกุล

JIRAN SITHIYOPASAKUL

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2567

KMITL-2024-EN-M-027-227

IMPLEMENTATION OF CLOUD COMPUTING AND INTERNET OF THINGS (IoT) BY PERFORMANCE EVALUATION

JIRAN SITHIYOPASAKUL

A THESIS SUMMITED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

MASTER OF ENGINEERING IN ELECTRICAL AND COMPUTER ENGINEERING

SCHOOL OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2024

KMITL-2024-EN-M-027-227

หัวข้อวิทยานิพนธ์	การใช้งานคลาวด์คอมพิวติ้งและไอโอทีโดย การประเมินประสิทธิภาพ
นักศึกษา	นาย จิรัณย์ สิทธิโยภาสกุล
รหัสประจำตัว	66016018
ปริญญา	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
พ.ศ.	2567
อาจารย์ที่ปรึกษาวิทยานิพนธ์	รศ.ดร. ชวลิต เบญจางคประเสริฐ
อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม	รศ.ดร. บุณย์ชนะ ภู่ระหงษ์

บทคัดย่อ

งานวิจัยนี้มีวัตถุประสงค์เพื่อสำรวจโอกาสในการเปลี่ยนแปลงที่นำเสนอโดยการการผสานรวมของการประมวลผลแบบคลาวด์และ Internet of Things (IoT) มีศักยภาพในการปฏิวัติอุตสาหกรรมและแอปพลิเคชัน ต่าง ๆ เช่น ระบบการดูแลสุขภาพ การขนส่ง เมืองอัจฉริยะ และ อื่น ๆ อย่างไรก็ตาม การดำเนินการให้ประสบความสำเร็จนั้น จำเป็นต้องมีการประเมินประสิทธิภาพอย่างรอบคอบ การประเมินประสิทธิภาพนี้จะเกี่ยวข้องกับการประเมินคุณภาพและประสิทธิภาพของคลาวด์คอมพิวติ้งและระบบ IoT วิทยานิพนธ์นี้นำเสนอการใช้งานเชิงลึกและการประเมินประสิทธิภาพของการประมวลผลบนคลาวด์และระบบ IoT โดยเน้นไปที่การวัดประสิทธิภาพบนแพลตฟอร์มคลาวด์หลักสามแพลตฟอร์ม: Amazon Web Services (AWS), แพลตฟอร์ม Google Cloud (GCP) และ Microsoft Azure วัตถุประสงค์ของการนำระบบคลาวด์คอมพิวติ้งและระบบ IoT ไปใช้ให้ประสบความสำเร็จนั้น จำเป็นต้องมีการประเมินประสิทธิภาพอย่างรอบคอบ เพื่อประเมินคุณภาพและประสิทธิภาพ วิทยานิพนธ์นี้นำเสนอภาพรวมของการประมวลผลแบบคลาวด์และ IoT รวมถึงแนวคิดพื้นฐานและสถาปัตยกรรม การประเมินระบบเหล่านี้จะมีการวัดเมตริกประสิทธิภาพหลัก เช่น เวลาตอบสนอง ปริมาณงาน และการใช้ทรัพยากร การประเมินนี้สามารถช่วยระบุจุดที่ต้องปรับปรุง ประสบการณ์ผู้ใช้ที่ได้รับการปรับปรุง ซึ่งจะนำไปสู่ประสิทธิภาพของระบบที่ดีขึ้น

Thesis	Implementation of Cloud computing and Internet of Things (IoT) by Performance Evaluation
Student	Mr. Jiran Sithiyopasakul

Student ID. 66016018

Degree Master of Engineering

Program Electrical and Computer Engineering

Year 2024

Thesis Advisor Assoc.Prof.Dr. Chawalit Benjangkaprasert

Co-Thesis Advisor Assoc.Prof.Dr. Boonchana Purahong

ABSTRACT

The purpose of this research is to explore the transformative opportunity presented by the integration of cloud computing and the Internet of Things (IoT) across various industries, from healthcare to transportation and smart cities. However, realizing this potential demands a rigorous evaluation of system performance. This thesis offers an in-depth implementation and performance evaluation of cloud computing and IoT systems, with a particular focus on measuring their effectiveness across three major cloud platforms: Amazon Web Services (AWS), Google Cloud Platform (GCP), and Microsoft Azure. The purpose of the successful implementation of cloud computing and IoT systems requires careful evaluation of their performance to assess quality and efficiency. This thesis provides an overview of cloud computing and IoT, including their fundamental concepts and architecture, and explores their potential applications in different sectors. The purpose of our evaluation of these systems encompasses measuring key performance metrics such as response time, throughput, and resource utilization. This assessment is crucial for identifying areas for improvement, which ultimately leads to better system performance, enhanced user experiences, and improved system efficiency. The purpose of focusing on AWS, GCP, and Azure is to provide insights into how different cloud platforms affect the performance of integrated cloud computing and IoT systems, enabling stakeholders to make informed decisions regarding platform selection and optimization strategies.

ACKNOWLEDGEMENT

I would like to express my heartfelt appreciation and extend my sincerest thanks to my supervisor, Associate Professor Dr. Chawalit Benjangkaprasert and Associate Professor Dr. Boonchana Purahong, for the educational opportunities that my committee has afforded me. I am immensely grateful for his invaluable guidance and support throughout the duration of my project. Without his assistance, I would not have been able to successfully complete my thesis, which serves as a significant source of motivation for me to conclude my work. The outstanding achievements in this research would not have been attainable without their mentorship. I genuinely value and gladly recognize their contributions. Lastly, I wish to convey my gratitude to my family for their unwavering support in my educational pursuits. Their influence has profoundly impacted me and broadened my perspective.

TABLE OF CONTENTS

Page

LIST OF TABLES

Table

Page

LIST OF FIGURES

Figure

Page

CHAPTER 1

INTRODUCTION

1.1 BACKGROUND AND MOTIVATION

The integration of cloud computing and the Internet of Things (IoT) has emerged as a pivotal paradigm with transformative potential across diverse industries. Cloud computing provides on-demand access to computing resources over the internet, offering scalability, flexibility, and cost-effectiveness. On the other hand, IoT technology facilitates seamless communication among physical devices, enabling data collection, exchange, and analysis [1]. The convergence of these two technologies opens up new avenues for innovation, automation, and efficiency enhancement in various domains, including smart cities, healthcare, agriculture, and industrial automation.

While the promise of cloud-enabled IoT systems is substantial, several challenges hinder their widespread adoption and optimization. These challenges encompass aspects such as security, privacy, scalability, and performance. Ensuring the robustness and efficiency of these integrated systems is crucial for realizing their full potential and overcoming barriers to adoption. Thus, there is a pressing need for comprehensive performance evaluation methodologies that can gauge the quality, reliability, and efficiency of cloud computing and IoT systems under various conditions and scenarios.

This study aims to provide valuable insights into the strengths and limitations of each cloud platform in supporting IoT applications. Furthermore, the research seeks to identify areas for improvement and optimization, guiding practitioners, researchers, and decision-makers in selecting the most suitable cloud platform for their specific use cases and optimizing system performance.

The purpose of this section is to provide an overview of the background and motivation behind the study, the primary objective of this research is to conduct a thorough performance evaluation of cloud computing and IoT systems, with a specific focus on assessing the performance of major cloud platforms (AWS, GCP, Azure) under different operational conditions by systematically analyzing performance metrics such as response time, throughput, latency, and reliability. Ultimately, the research endeavors to contribute to the advancement and refinement of cloud-enabled IoT systems, fostering their wider adoption and fostering innovation across industries.

1.2 CLOUD COMPUTING AND CLOUD INFRASTRUCTURE OVERVIEW

Cloud computing refers to the delivery of computing services over the internet, enabling users to access and utilize computing resources such as servers, storage, databases, networking, software, and more, without the need for on-premises infrastructure. These services are typically provided by cloud service providers on a pay-as-you-go basis, allowing organizations to scale resources up or down as needed without the overhead of maintaining physical hardware. The main cloud computing deployment models are public, private, and hybrid clouds and these offer different services.

Cloud infrastructure, on the other hand, encompasses the physical components and software systems that make up the cloud computing environment. This includes data centers, servers, networking equipment, storage devices, virtualization software, and management tools. Cloud infrastructure provides the foundation for delivering cloud computing services and supports the dynamic allocation of resources to meet changing demand as Fig. 1.1 below.

In essence, cloud computing is the concept of delivering computing services over the internet, while cloud infrastructure refers to the underlying hardware and software components that enable this delivery.

This research primarily focuses on the performance evaluation of cloud computing and Internet of Things (IoT) systems, particularly in the context of their integration. The study aims to comprehensively assess the quality, efficiency, and reliability of cloud-enabled IoT systems by conducting experiments and analyses under various operational conditions. Specifically, the research delves into evaluating the performance of three major cloud platforms—Amazon Web Services (AWS), Google Cloud Platform (GCP), and Microsoft Azure—across different scenarios. The study examines how these cloud platforms perform under normal operating conditions, heavy load scenarios, and variations in IoT applications, including both low-latency and high-throughput scenarios.

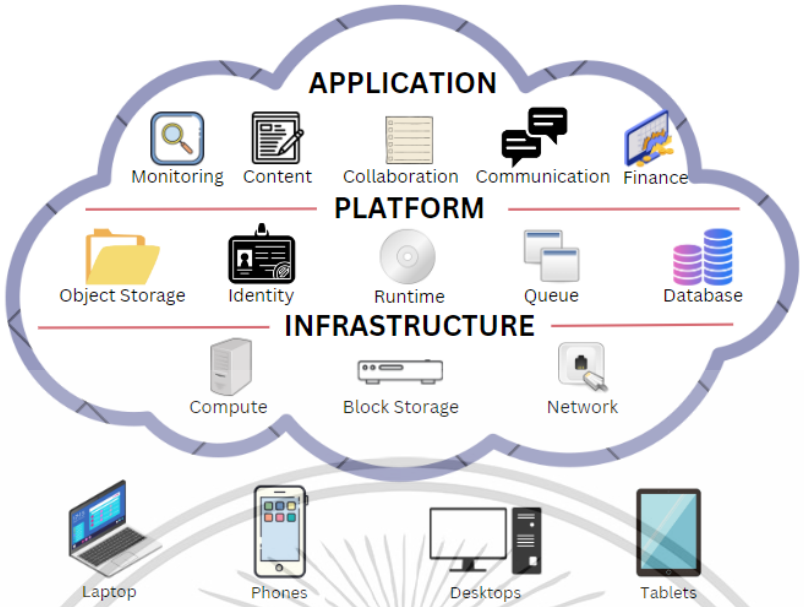

Fig. 1.1 Cloud Infrastructure

In the realm of cloud computing, the infrastructure, application, and platform layers represent distinct components that together enable the delivery of services over the internet.

Infrastructure as a Service (IaaS):

Definition: Infrastructure as a Service provides virtualized computing resources over the internet, including virtual machines, storage, and networking components.

Application: Organizations can utilize IaaS to outsource their entire IT infrastructure, eliminating the need for physical hardware maintenance and management. This enables scalability and flexibility, allowing businesses to scale resources up or down based on demand.

Platform: IaaS serves as the foundational layer upon which higher-level cloud services are built. Platforms like AWS EC2 (Elastic Compute Cloud), Microsoft Azure Virtual Machines, and Google Compute Engine provide scalable and on-demand access to virtualized computing resources.

Platform as a Service (PaaS):

Definition: Platform as a Service offers a development and deployment environment hosted in the cloud, allowing developers to build, deploy, and manage applications without worrying about infrastructure management.

Application: PaaS platforms streamline the development process by providing tools, libraries, and frameworks for application development, testing, and deployment. Developers can focus on writing code and building applications without dealing with underlying infrastructure complexities.

Platform: PaaS platforms abstract away the underlying infrastructure and provide a runtime environment for applications. Examples include AWS Elastic Beanstalk, Microsoft Azure App Service, and Google App Engine.

Software as a Service (SaaS):

Definition: Software as a Service delivers software applications over the internet on a subscription basis, eliminating the need for users to install, maintain, and update software locally.

Application: SaaS applications cover a broad range of services, including email, customer relationship management (CRM), enterprise resource planning (ERP), and collaboration tools. Users access these applications through web browsers or APIs, with the underlying infrastructure and maintenance handled by the service provider.

Platform: While SaaS applications are typically built on top of PaaS or IaaS infrastructures, they represent the highest level of abstraction for end-users, offering ready-to-use applications accessible via the internet. Examples include Salesforce, Microsoft Office 365, and Google Workspace.

In summary, cloud infrastructure provides the foundational computing resources (IaaS), which are then abstracted and extended through platform services (PaaS) to enable streamlined application development and deployment. Finally, software applications are delivered directly to end-users via the internet as Software as a Service (SaaS), completing the cloud service delivery model. Each layer offers unique benefits and caters to different use cases, enabling organizations to leverage cloud computing resources based on their specific requirements and preferences.

Cloud infrastructure forms the foundational layer of cloud computing, providing the essential computing resources necessary for the delivery of cloud services. It encompasses a diverse array of hardware and software components, including servers, storage devices, networking equipment, virtualization software, and management tools. At its core, cloud infrastructure enables the abstraction and pooling of resources, allowing users to access and utilize computing resources over the internet on an on-demand basis. This infrastructure is

typically hosted and managed by cloud service providers, who offer various service models and deployment options to cater to the needs of different users and organizations.

One of the defining features of cloud infrastructure is its scalability and elasticity. Cloud providers can dynamically allocate and reallocate resources based on demand, enabling users to scale their infrastructure up or down as needed without the constraints of physical hardware limitations. This scalability ensures optimal resource utilization and cost-efficiency, as users only pay for the resources they consume, leading to greater agility and flexibility in managing workloads.

Another key aspect of cloud infrastructure is its virtualized nature. Through virtualization technologies, physical hardware resources are abstracted and partitioned into virtual instances, allowing multiple virtual machines (VMs) or containers to run on a single physical server. This enables efficient resource utilization and consolidation, leading to higher levels of performance and density.

Moreover, cloud infrastructure is designed to be highly resilient and fault-tolerant, with redundant components and automated failover mechanisms to ensure uptime and availability. Data redundancy, backup, and disaster recovery capabilities are integral parts of cloud infrastructure, providing users with robust data protection and business continuity solutions.

Overall, cloud infrastructure plays a critical role in enabling the delivery of cloud computing services, empowering organizations to harness the benefits of scalability, flexibility, and reliability in meeting their IT and business needs. Whether it's deploying virtual servers, storing massive amounts of data, or running complex applications, cloud infrastructure provides the foundational framework for modern digital operations.

1.3 INTERNET OF THINGS (IOT)

The Internet of Things (IoT) is a vast network of physical objects, or "things," equipped with sensors, software, and connectivity, allowing them to gather and exchange data via the Internet. The core vision of IoT is to facilitate seamless communication among devices, automate processes, and provide intelligent insights for improved decision-making in various domains.

IoT technology spans applications from home automation and smart cities to healthcare, and environmental monitoring. In a smart home scenario, devices like

thermostats, smart lighting, and security cameras are integrated into a central system accessible remotely through smartphones. This empowers users to manage energy consumption, security, and other systems from anywhere.

In the realm of industrial automation, IoT technology assumes a pivotal role in monitoring and managing factory equipment. It enables the optimization of production processes while minimizing downtime. By outfitting machinery with sensors and connectivity, real-time data on factors like machine performance and energy utilization can be continuously gathered. This data serves as a foundation for insightful analysis, unveiling opportunities for process enhancement and efficiency.

A key advantage of IoT is its capacity to gather and analyze large volumes of data from diverse sources. This data-driven approach offers insights into consumer behavior, allowing businesses to streamline operations and innovate products and services. For example, in healthcare, wearable IoT sensors collect vital signs and activity data, enabling remote monitoring and timely intervention for patient well-being.

In summary, IoT technology is a transformative force in our daily lives and workplaces, driving automation, optimization, and intelligence across diverse applications as Fig. 1.2 below. It promises a future characterized by unprecedented connectivity and efficiency.

Fig. 1.2 Internet of Things (IoT)

1.4 CLOUD-BASED IOT APPLICATIONS

Cloud-based IoT applications refer to Internet of Things (IoT) solutions and services that leverage cloud computing resources to enhance and expand their capabilities. These applications rely on cloud platforms to store, process, and analyze the data generated by IoT devices, enabling more robust and scalable IoT solutions. Cloud-based IoT applications offer several advantages, including easier data management, enhanced security, and the ability to scale resources as needed as shown in Fig. 1.3.

Smart Cities: IoT technology enhances urban management by optimizing transportation, energy, waste management, and public safety through real-time sensor data. It improves traffic flow, reduces energy consumption, and enhances public safety measures.

Precision Agriculture: IoT enables precision farming by monitoring soil conditions, weather, and crop growth. This data-driven approach informs decisions on irrigation, fertilization, and pest control, leading to resource efficiency and increased yields.

Industrial IoT (IIoT): IIoT improves industrial efficiency and safety by enabling real-time equipment monitoring, remote machinery control, and data-driven improvements.

Smart Homes: IoT devices in smart homes elevate comfort, convenience, and security by enabling remote device control, security monitoring, and energy-saving measures.

Healthcare Monitoring: IoT facilitates remote patient monitoring, real-time vital signs tracking, and timely alerts to healthcare providers, reducing the need for in-person visits and improving patient care.

Traffic Management: IoT sensors monitor traffic flow, optimize signals, reroute vehicles, and provide real-time information to drivers, improving traffic management and reducing delays.

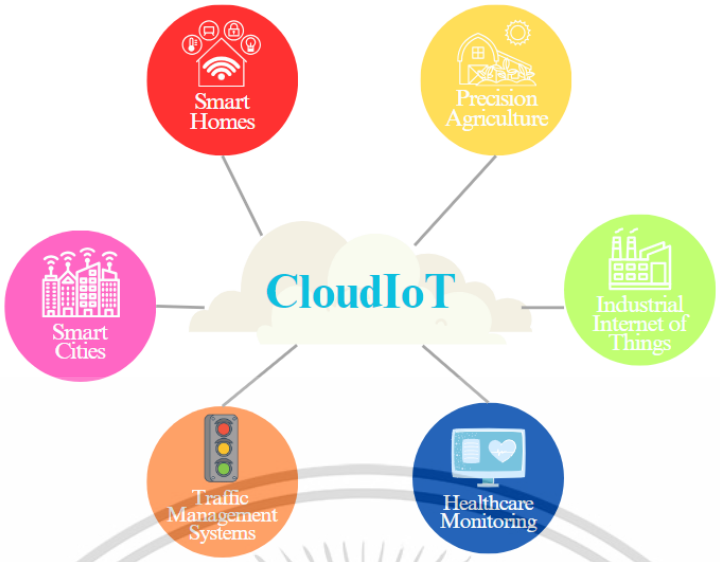

Fig. 1.3 The integration of cloud and IoT technologies

1.5 PERFORMANCE METRICS AND EVALUATION

Performance evaluation is a critical aspect of both cloud computing and IoT systems, and it involves the assessment of various key metrics to ensure optimal functionality.

Response time, which measures the system's speed in responding to requests, is essential for gauging performance. To evaluate this, a simple web application is deployed on virtual machines within each cloud provider's infrastructure, and Apache Benchmark (ab) is used to measure response times, recording statistics such as minimum, maximum, and average response times.

Throughput, determining the system's data processing capacity, is tested by initiating network throughput tests between a Raspberry Pi compute module 4 and each cloud provider's virtual machines using iperf3. Lower latency, which signifies quicker response to requests, is measured using the ping command from the Raspberry Pi compute module 4 to the cloud provider's virtual machines, recording average and maximum latency values.

Reliability is assessed by subjecting the web application to increased load, adjusting request numbers and concurrent connections to stress the system, and monitoring error rates to ensure system uptime and availability.

Overall, comprehensive evaluation using these metrics helps organizations optimize system performance, ensure reliability, and maximize the value of cloud computing and IoT solutions.

CHAPTER 2

RELATED THEORY

2.1 MESSAGE QUEING TELEMETRY TRANSPORT (MQTT)

MQTT (Message Queuing Telemetry Transport) is a lightweight messaging protocol widely used in IoT (Internet of Things) and M2M (Machine to Machine) communication scenarios. It operates on top of the TCP/IP or other network protocols, providing a publish/subscribe messaging model as shown in Fig. 2.1.

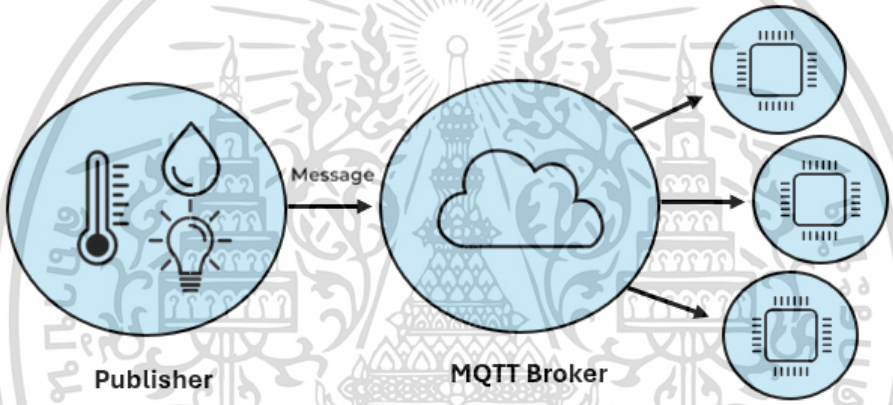

Fig. 2.1 Message Queuing Telemetry Transport Process

MQTT enables communication between devices and applications in a decentralized manner. It consists of clients that publish messages to topics and clients that subscribe to topics to receive messages. The publish/subscribe model allows for efficient and scalable message distribution across distributed systems.

MQTT supports Quality of Service (QoS) levels to ensure message delivery reliability. QoS levels include:

QoS 0 (At most once): Messages are delivered at most once, without acknowledgment or retransmission.

QoS 1 (At least once): Messages are guaranteed to be delivered at least once, with acknowledgment and potential duplication.

QoS 2 (Exactly once): Messages are guaranteed to be delivered exactly once, with acknowledgment and no duplication.

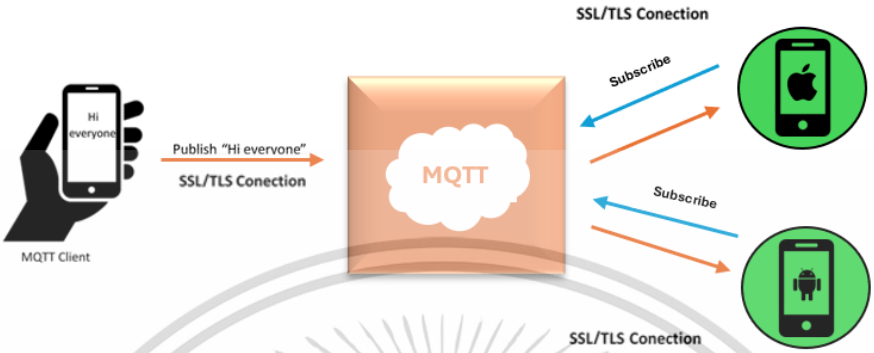

Fig. 2.2 MQTT connection

One of the key features of MQTT is its lightweight nature, making it suitable for resource-constrained devices and low-bandwidth networks. MQTT minimizes bandwidth usage and power consumption, making it ideal for IoT deployments.

Security in MQTT can be implemented through mechanisms such as TLS (Transport Layer Security) or SSL (Secure Sockets Layer) to encrypt communication between clients and brokers. This ensures confidentiality and integrity of data transmitted over MQTT connections as shown in Fig. 2.2.

MQTT messages consist of a topic, payload, QoS level, and optional properties. The topic is used to categorize messages and determine which clients receive them. The payload contains the actual data being transmitted. QoS level specifies the delivery guarantee for the message. Optional properties provide additional metadata for message processing.

Overall, MQTT provides a scalable, efficient, and lightweight messaging protocol for IoT and M2M communication, facilitating seamless integration and data exchange between devices and applications.

2.2 HYPERTEXT TRANSFER PROTOCOL (HTTP)

HTTP (Hypertext Transfer Protocol) is a foundational protocol widely utilized for communication between clients and servers, prevalent in web browsing, IoT, and various machine-to-machine (M2M) scenarios. It operates atop TCP/IP or other network protocols, presenting a request-response model.

Fig. 2.3 HTTP Request-Response Flow

HTTP facilitates decentralized communication between diverse devices and applications. It encompasses clients, which issue requests, and servers, which respond with requested resources or status updates. This model ensures effective and scalable exchange across distributed systems.

HTTP primarily operates over port 80 for unencrypted communication and port 443 for encrypted communication using HTTPS (HTTP Secure). HTTPS employs mechanisms like TLS or SSL to encrypt client-server communications, safeguarding data confidentiality and integrity, akin to MQTT's encryption support.

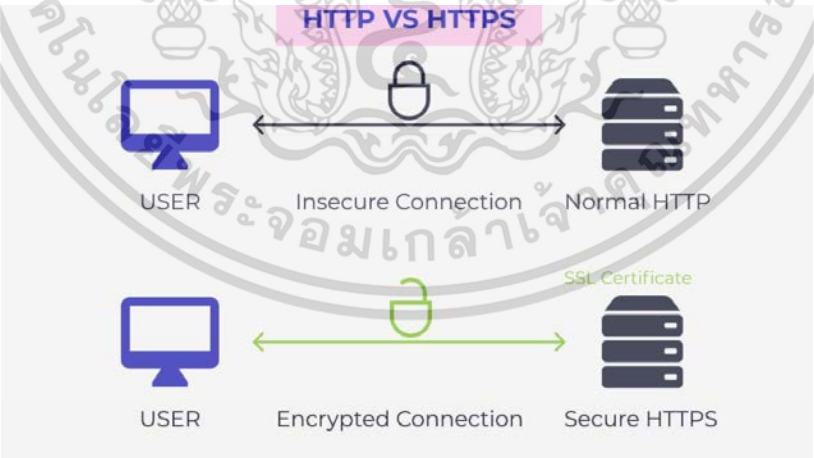

Fig. 2.4 HTTP and HTTPS Connection

One of HTTP's core attributes is its versatility, allowing deployment across resource-constrained devices and low-bandwidth networks, albeit not as lightweight as MQTT.

Through techniques like HTTP/2 multiplexing and content compression, HTTP optimizes bandwidth usage and power consumption.

Security in HTTP is further enhanced through HTTPS (HTTP Secure), ensuring secure communication over the web. This is particularly crucial for transmitting sensitive data, such as personal information or financial transactions.

HTTP messages feature headers and an optional body, akin to MQTT's topic, payload, and optional properties. Headers convey metadata, while the body carries transmitted data.

Overall, HTTP offers a scalable, efficient, and versatile protocol for IoT and M2M communication, fostering seamless integration and data exchange across devices and applications.

2.3 VIRTUAL MACHINE

A virtual machine (VM) is a software emulation of a physical computer system that enables multiple operating systems (OS) to operate on a single physical machine, depicted in Fig. 2.4. Each virtual machine operates independently, possessing its own virtual hardware and OS, thereby facilitating the simultaneous operation of multiple applications and services on the same physical hardware.

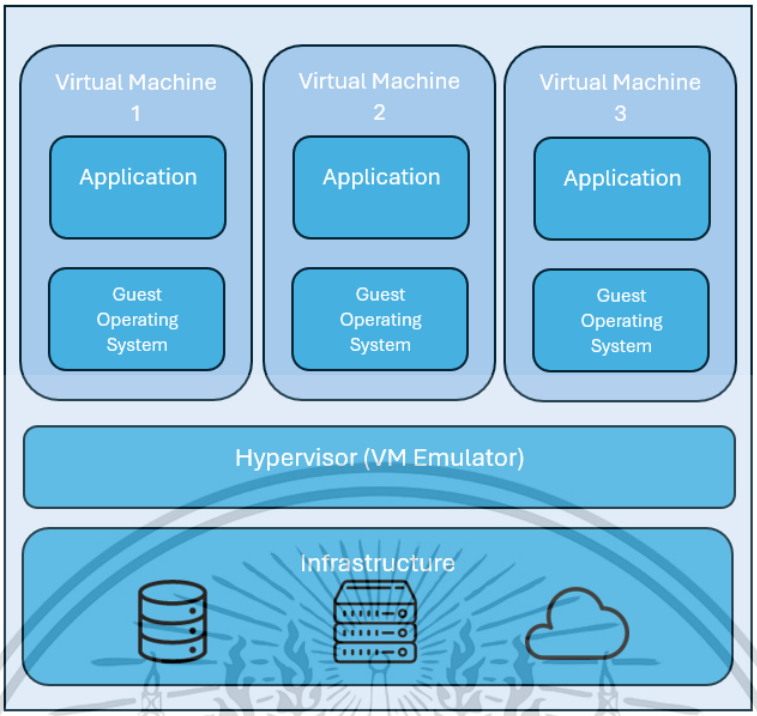

Fig. 2.5 Virtual machine contained operating system

One of the main benefits of using a virtual machine is improved efficiency and resource utilization. By allowing multiple operating systems and applications to run on a single physical machine, virtual machines can help organizations make better use of their hardware resources, reducing the need for additional hardware and associated costs. Another benefit of virtual machines is improved security. By isolating each virtual machine from the others, virtual machines can help prevent the spread of malware and other security threats between different systems. In addition, virtual machines can be easily backed up and restored, allowing organizations to quickly recover from security incidents or other disruptions.

Virtual machines also offer significant flexibility and scalability. Being software-based, they can be effortlessly created, duplicated, and migrated to alternative physical hardware, allowing organizations to promptly deploy new systems and scale their infrastructure to meet evolving demands.

Nevertheless, there are challenges associated with virtual machines. They can consume substantial resources, particularly memory and CPU cycles, potentially impacting overall system performance. Additionally, managing a large number of virtual machines may be intricate and necessitate specialized skills and expertise.

Compatibility issues also arise with virtual machines. Some applications may not be fully compatible or may require additional configuration to operate efficiently. Furthermore, certain applications may demand direct access to hardware resources, which may not be accessible in a virtualized environment.

Despite these challenges, virtual machines offer a valuable tool for organizations seeking to enhance efficiency, scalability, and security in their IT infrastructure. While managing virtual machines may be complex and compatibility issues may arise with certain applications, they provide a flexible and cost-effective solution for optimizing hardware resources and enhancing system performance.

2.4 SECURE SHELL

Secure Shell (SSH) is an encryption protocol operating on port 22, primarily utilized for administering Unix/Linux servers, Internet routers, and a significant portion of cloud computing infrastructure [4]. It serves as a means for remote access to servers and systems, enabling users to securely log in to a remote system and execute commands as if they were locally present.

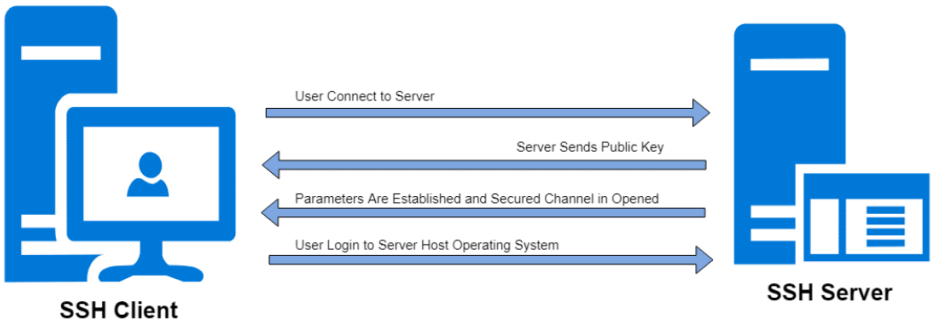

Fig. 2.6 Secured Socket Layer (SSH) Process

SSH works by establishing an encrypted connection between the client and the server as shown in Fig. 2.3, which provides protection against eavesdropping, tampering, and other forms of network-based attacks. It uses public keys for authenticating servers and users [4] ensuring that only authorized users are able to access the system.

SSH operates by establishing an encrypted connection between the client and the server, illustrated in Fig. 2.3, thereby safeguarding against eavesdropping, tampering, and other network-based attacks. Utilizing public keys for authenticating servers and users [4], SSH ensures that only authorized users can access the system.

A key advantage of SSH lies in its heightened security. Through encrypting all transmitted data between client and server, SSH thwarts attackers from intercepting sensitive information like passwords, usernames, and other credentials, significantly bolstering system security. Moreover, SSH offers versatility, supporting a broad spectrum of tasks such as remote access, file transfers, and tunneling of various protocols. This versatility renders it an indispensable tool for system administrators and developers, enabling them to manage and secure diverse systems and applications effectively.

Additionally, SSH supports various authentication methods, encompassing password-based authentication, public-key authentication, and multi-factor authentication. This affords users the flexibility to select the most suitable authentication method based on security requirements and system complexity.

Nonetheless, there are certain drawbacks associated with SSH usage. Configuring and setting up SSH can be more intricate compared to other remote access protocols. Furthermore, SSH sessions may consume substantial network bandwidth, especially during large file transfers or when employing graphical applications over remote connections.

In essence, SSH serves as a secure network protocol facilitating remote access to systems and applications. Its encryption and authentication mechanisms safeguard sensitive data during transmission between client and server, while its support for diverse authentication methods enhances overall security. Despite potential complexity during setup and configuration, SSH remains a versatile and secure solution for managing and securing a wide array of systems and applications.

2.5 CLOUD SERVICES PROVIDERS

Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform (GCP) are three of the most popular cloud providers in the world as shown in Fig. 4. Each of them has its strengths and advantages. AWS offers a comprehensive suite of IoT services, global scalability, and seamless integration with other AWS services. Microsoft Azure stands out for its compatibility with Windows and strong integration with Microsoft products, making it a preferred choice for organizations within the Microsoft ecosystem. It also provides robust analytics and AI capabilities. Google Cloud Platform excels in data processing and analytics, supports edge computing, and offers machine learning tools, with a global network of data centers. The choice among these providers depends on specific project requirements and existing technology stacks, so it's essential to verify the latest developments to make an informed decision.

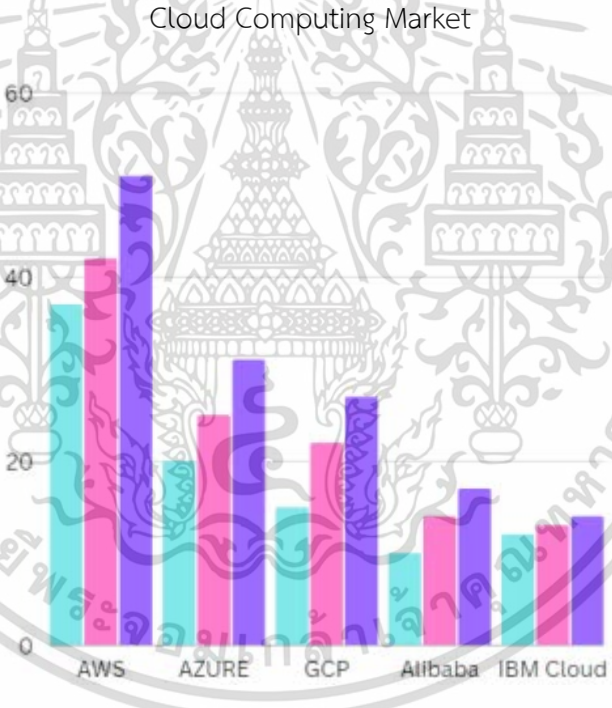

Fig. 2.7 Example of Cloud services used for IoT

2.6 LOAD BALANCER

A load balancer is a method used to efficiently distribute the workload across various nodes within a collaborative system, enhancing resource utilization and boosting job response times. This redistribution ensures that no single node becomes overwhelmed,

thereby preventing performance degradation. Load balancing considers factors such as CPU load, memory usage, latency, and network load when assigning tasks to nodes.

The primary advantage of load balancers is performance enhancement. By evenly distributing incoming network traffic among multiple servers or resources, load balancers prevent any single server from becoming overloaded, ensuring users experience fast response times and efficient resource access. Additionally, load balancers optimize resource usage by directing traffic to the most suitable server based on factors like capacity and health, preventing resource wastage.

Another benefit is increased availability. Load balancers distribute traffic across multiple servers, ensuring that even if one server fails, users can still access the application or service through other servers, minimizing downtime. Furthermore, load balancers offer security benefits by acting as a single-entry point into the network, allowing for effective traffic monitoring, threat detection, and additional security layers.

Despite these advantages, there are potential drawbacks. Load balancers can introduce complexity into the network architecture and require careful configuration and management to operate effectively. Additionally, they can be costly, with hardware-based load balancers requiring significant upfront investment and ongoing maintenance expenses, while cloud-based options may accrue usage fees over time.

In summary, load balancers are valuable tools for optimizing performance, resource utilization, availability, and security within a network. However, they require thorough management to ensure alignment with the organization's specific needs and to mitigate potential challenges.

2.7 AUTOSCALING

Autoscaling, a pivotal approach in cloud computing, dynamically adjusts computing resource allocation for applications based on demand fluctuations. It enables applications to automatically scale up or down the number of virtual machines, containers, or other resources, aligning with current workload requirements. This adaptive feature enhances performance and availability significantly. By provisioning additional resources during peak demand, applications maintain swift response times. Conversely, during reduced demand, autoscaling efficiently reduces resource usage, optimizing costs and operational efficiency.

Moreover, autoscaling enhances application resilience by mitigating hardware failures through redundancy and failover capabilities offered by multiple virtual machines or containers. This proactive approach minimizes downtime and data loss risks. Cost efficiency is another key benefit, as autoscaling ensures optimal resource utilization, preventing overprovisioning and reducing expenses related to unused resources.

Additionally, autoscaling fosters organizational agility by swiftly adapting to evolving demand scenarios, facilitating rapid deployment of new services and responsiveness to changing business needs. However, implementing autoscaling presents challenges. Configuring autoscaling algorithms accurately is intricate and requires optimization to avoid resource underutilization or overprovisioning. Additionally, managing the increased complexity in IT infrastructure can pose difficulties.

Cost implications also warrant consideration, as while autoscaling optimizes resource costs, expenses associated with virtual machines or containers remain significant, especially for large-scale applications or high-demand environments. In essence, while autoscaling empowers organizations to enhance performance, availability, cost efficiency, and flexibility, it necessitates meticulous planning and management to align with organizational requirements effectively.

CHAPTER 3

OVERVIEW OF CLOUD SERVICE PROVIDERS

3.1 AMAZON WEB SERVICES

Amazon Web Services (AWS) stands as a paramount cloud computing platform in the market, offered by Amazon.com. It presents an extensive suite of cloud-based solutions encompassing computing power, storage, databases, analytics, networking, machine learning, artificial intelligence, Internet of Things (IoT), security, and more. With its global network of data centers distributed across various geographic regions, AWS enables businesses to deploy applications and services in proximity to their end-users, enhancing performance and reducing latency. This expansive infrastructure serves as a cornerstone for many digital endeavors worldwide.

AWS offers compute services like Elastic Compute Cloud (EC2) and serverless computing with Lambda, catering to diverse computational needs. It also provides storage solutions such as Simple Storage Service (S3) and Glacier, ensuring scalability and cost-effectiveness.

Furthermore, AWS provides robust storage solutions, such as Simple Storage Service (S3) for object storage, Elastic Block Store (EBS) for block storage, and Glacier for archival purposes. These offerings are designed to meet the scalability, durability, and cost-efficiency requirements of modern businesses. Security and compliance are prioritized with IAM, encryption, firewall, monitoring, and compliance programs. AWS empowers organizations with advanced analytics and machine learning capabilities, driving innovation and competitive advantage.

In addition to computing and storage, AWS delivers managed database services like Amazon Relational Database Service (RDS), Amazon DynamoDB, and Amazon Redshift. These services relieve users of administrative burdens, allowing them to focus on application development and innovation.

AWS also addresses networking requirements through services like Virtual Private Cloud (VPC), AWS Direct Connect, and Amazon Route 53. These offerings enable organizations to establish secure and scalable network architectures, connecting their on-premises infrastructure with AWS cloud resources seamlessly.

Lastly, AWS IoT enables organizations to connect devices to the cloud, manage IoT devices, and process IoT data, facilitating digital transformation initiatives. Overall, AWS stands as a reliable, scalable, and flexible cloud computing platform, committed to innovation, security, and customer success.

3.2 GOOGLE CLOUD PLATFORM

Google Cloud Platform (GCP) stands as a leading cloud computing platform offered by Google. With a comprehensive suite of services spanning computing, storage, databases, machine learning, networking, and security, GCP caters to diverse business needs. Its global network of data centers ensures high performance and low latency for applications deployed on the platform.

GCP provides a range of compute services, including Compute Engine for virtual machines and Google Kubernetes Engine for containerized applications. These offerings enable organizations to scale their compute resources flexibly to meet demand while maintaining high availability and reliability.

In terms of storage, GCP offers solutions like Cloud Storage for object storage and Persistent Disk for block storage. These services provide scalable and durable storage options, ensuring data availability and integrity for businesses of all sizes.

Managed database services such as Cloud SQL, Cloud Spanner, and Firestore simplify database management tasks, allowing organizations to focus on building applications rather than managing infrastructure. Additionally, GCP's networking services like Virtual Private Cloud (VPC) and Cloud Load Balancing enable organizations to create secure and scalable network architectures to support their applications.

Security is a top priority for GCP, with features like Identity and Access Management (IAM), encryption, firewall, and advanced threat detection built into the platform. These security measures help protect data and applications hosted on GCP from unauthorized access and cyber threats.

Overall, Google Cloud Platform provides a reliable, scalable, and secure cloud computing environment, equipped with a comprehensive suite of services to meet the diverse needs of businesses across industries. Its commitment to innovation, performance, and security makes it a preferred choice for organizations looking to leverage cloud technology to drive digital transformation and achieve business objectives.

3.3 MICROSOFT AZURE

Microsoft Azure stands as a leading cloud computing platform offered by Microsoft. With a comprehensive suite of services spanning computing, storage, databases, AI, machine learning, networking, and security, Azure caters to diverse business requirements. Its global network of data centers ensures high availability and low latency, facilitating efficient and reliable access to applications and services worldwide.

Azure offers a variety of compute services, including Virtual Machines, Azure Kubernetes Service (AKS) for containerized applications, and Azure Functions for serverless computing. These services empower organizations to run their applications efficiently and scale resources as needed to meet demand.

In terms of storage, Azure provides options such as Azure Blob Storage for object storage, Azure Disk Storage for block storage, and Azure Files for file storage. These solutions offer scalability, durability, and reliability, ensuring organizations can securely store and access their data in the cloud.

Managed database services like Azure SQL Database, Azure Cosmos DB, and Azure Database for PostgreSQL offer fully managed database solutions, freeing organizations from the complexities of database management and enabling them to focus on application development and innovation.

Security is a paramount concern for Azure, with features such as Azure Active Directory (AD), encryption, network security groups, and Azure Security Center. These built-in security measures help protect data, applications, and infrastructure from threats and unauthorized access, ensuring a secure cloud computing environment for businesses of all sizes.

In summary, Microsoft Azure provides a robust, scalable, and secure cloud computing platform equipped with a comprehensive suite of services to meet the evolving needs of modern businesses. Its global reach, extensive offerings, and strong focus on security make it a preferred choice for organizations embarking on their cloud journey and seeking to drive innovation and achieve business objectives.

3.4 DIFFERENTIATION AMONG CLOUD SERVICE PROVIDERS

The realm of cloud computing is undergoing rapid evolution, fundamentally altering organizational approaches to IT infrastructure management. As demand for cloud services escalates, an abundance of providers emerges, each presenting a diverse spectrum of services. Consequently, discerning the optimal provider becomes a formidable task for organizations due to the unique attributes and deficiencies exhibited by each. In the ensuing section, we shall conduct a comparative examination of the three principal cloud service providers Google Cloud Platform (GCP), Microsoft Azure, and Amazon Web Services (AWS) with the aim of elucidating both their shared characteristics and points of differentiation.

Feature	aws	Google Cloud	Azure
Computing Services	EC2, Lambda	Compute Engine, GKE	Virtual Machines, AKS, Functions
Storage Solutions	S3, Glacier	Cloud Storage, Persistent Disk	Blob Storage, Disk Storage, Files
Managed Databases	RDS, DynamoDB, Redshift	Cloud SQL, Spanner, Firestore	SQL Database, Cosmos DB, PostgreSQL
Networking Services	VPC, Direct Connect	VPC, Cloud Load Balancing	VNet, Load Balancer, Traffic Manager
Security Features	IAM, Encryption, Firewall	IAM, Encryption, Firewall	Active Directory, Encryption, Security Center
Analytics and AI	Various, SageMaker	ML Engine, AI Platform	Azure AI, Machine Learning
Internet of Things (IoT)	IoT Core, Device Management	Cloud IoT Core, Device Management	IoT Hub, IoT Edge
Global Reach	Global network of data centers	Global network of data centers	Global network of data centers
Innovation Focus	Regular introduction of new services and features	Emphasis on innovation and new offerings	Strong focus on innovation and new offerings

Fig. 3.1 Comparison between cloud providers

Amazon Web Services (AWS), Google Cloud Platform (GCP), and Microsoft Azure are the top contenders in the cloud computing market, each offering a robust suite of services

tailored to meet the diverse needs of businesses worldwide. Fig. 3.1 provides a general overview of the products offered by these three providers.

AWS, as the pioneer in cloud computing, boasts a vast ecosystem of services, including Elastic Compute Cloud (EC2) for scalable virtual machines and Lambda for serverless computing. Its storage solutions like Simple Storage Service (S3) and Glacier provide reliable and cost-effective options for data storage and archival. AWS also leads in analytics and AI with services like Amazon SageMaker, empowering businesses with advanced machine learning capabilities.

On the other hand, GCP distinguishes itself with a strong emphasis on cutting-edge technologies such as machine learning and containerization. Google Kubernetes Engine (GKE) enables efficient management of containerized applications, while offerings like Cloud Storage and Persistent Disk ensure scalable and durable storage options. GCP's focus on innovation is evident in services like Machine Learning Engine and AI Platform, which leverage Google's expertise in artificial intelligence to drive business insights and innovation.

Microsoft Azure stands out for its seamless integration with existing Microsoft technologies, making it an attractive option for organizations already invested in the Microsoft ecosystem. Azure Virtual Machines and Azure Kubernetes Service (AKS) provide flexible compute options, while Azure Blob Storage and Disk Storage offer robust storage solutions. Azure's comprehensive suite of managed database services, including Azure SQL Database and Azure Cosmos DB, relieve organizations of database management complexities. Furthermore, Azure's strong focus on security, demonstrated through features like Azure Active Directory and Azure Security Center, instills confidence in businesses seeking a secure cloud environment.

Choosing the right cloud provider can indeed be challenging, given the wealth of options and considerations involved. However, AWS, GCP, and Azure stand out as excellent options. AWS leads with its extensive service offerings and market dominance, while GCP excels in innovation and advanced technologies. Azure's seamless integration with Microsoft products offers familiarity and interoperability, particularly for enterprises already entrenched in the Microsoft ecosystem. When selecting a cloud provider, organizations must carefully evaluate their specific requirements, such as scalability, security, compliance, and budget constraints.

Factors like geographic presence, pricing models, and customer support also play crucial roles in the decision-making process. Additionally, considering long-term strategic

objectives and future scalability needs is essential to ensure the chosen provider can accommodate growth and evolving business demands.

Furthermore, businesses should assess each provider's ecosystem and ecosystem compatibility with their existing infrastructure and applications. Integration capabilities, migration support, and ecosystem partnerships can significantly impact the ease of adoption and overall success of cloud initiatives. Ultimately, the right cloud provider will align closely with an organization's goals and objectives for digital transformation, providing the necessary tools, support, and expertise to drive innovation and competitive advantage in today's dynamic business landscape. Whether it's AWS, GCP, or Azure, making an informed decision based on thorough evaluation and strategic alignment will pave the way for a successful cloud journey.

CHAPTER 4

RESEARCH METHODOLOGY

4.1 RESEARCH DESIGN AND APPROACH

The research design and approach of this study were carefully crafted to address the complexities of evaluating the performance dynamics and operational characteristics of cloud computing and Internet of Things (IoT) integration. Recognizing the multifaceted nature of the research problem, a structured methodology was developed, drawing upon both quantitative and qualitative research methods. This approach aimed to provide a comprehensive understanding of the interplay between cloud platforms and IoT systems, particularly focusing on performance evaluation.

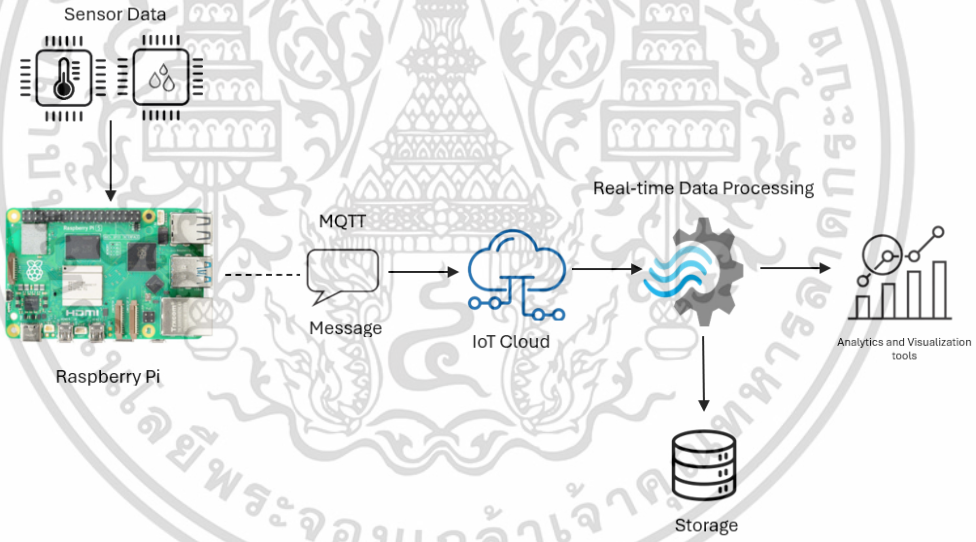

Fig. 4.1 Hardware and Software Overview

Fig. 4.1 shows designing of the research approach, particular attention was paid to aligning the methodology with the research objectives. The goal was to fill the existing gap in literature by conducting systematic evaluations of three major cloud platforms - Amazon Web Services (AWS), Google Cloud Platform (GCP), and Microsoft Azure - under various operational scenarios. By adopting a structured approach, the research team sought to ensure the reliability and validity of the findings, thereby enhancing the credibility of the study outcomes.

The system architecture of our cloud-based IoT solution comprises several key components that work together to collect, process, store, and analyze data, as well as to provide user interaction and control capabilities. The following flowchart illustrates the overall workflow of the system as shown in Fig. 4.2.

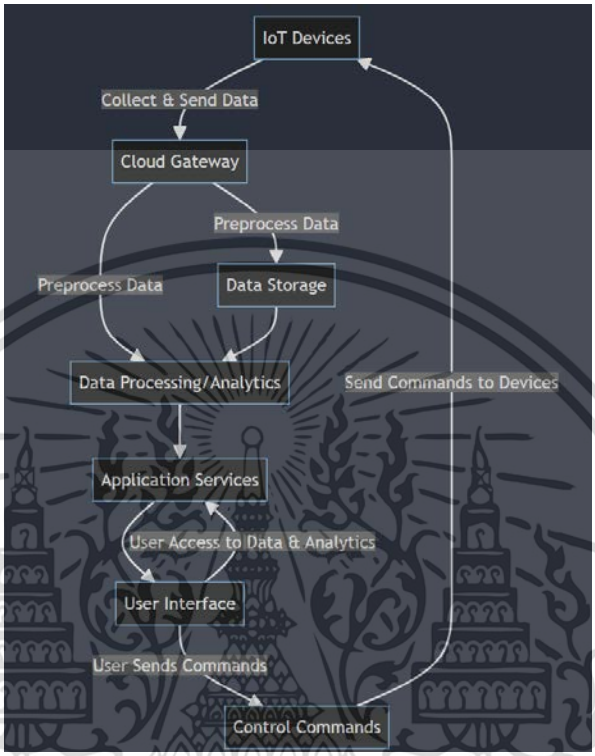

Fig. 4.2 System Architecture and Workflow

Table 1. Integration of Cloud and IoT

Cloud	IoT
Provides computing power, storage, and network for IoT data.	Offloads IoT computation and storage to the cloud
Enables real-time data processing and analysis.	Collects and transmits data for storage and analysis in the cloud.
Provides secure and reliable data storage and sharing.	Integrates with cloud security services to enhance security.
Offers scalable and flexible infrastructure for IoT services.	Can be remotely managed and updated from the cloud.
Enables IoT ecosystem integration.	Integrates with cloud machine learning and AI services.

Table 1 refers to how cloud computing and the Internet of Things (IoT) work together to provide a more comprehensive and effective solution for various use cases. Complementarity refers to how these two technologies complement each other by filling in each other's gaps and strengths. For example, cloud computing provides the computing power, storage, and network infrastructure that IoT devices need to transmit and process

their data, while IoT devices provide real-time data streams and physical interactions that enable the creation of intelligent, autonomous systems.

Integration, on the other hand, refers to how cloud and IoT technologies can be seamlessly integrated and work together as a unified system. For example, IoT devices can be connected to the cloud to leverage cloud-based services, such as analytics, security, and machine learning, while cloud services can be customized and optimized for IoT-specific use cases, such as smart homes, smart cities, or industrial IoT. Overall, the complementarity and integration of cloud and IoT technologies represent a powerful combination that can enable innovative solutions and transform various industries and domains.

.

4.2 EXPERIMENTAL SETUP

The experimental setup involves a network of physical devices acting as simulated IoT endpoints. We propose utilizing Raspberry Pi Compute Module 4 units for this purpose. These devices will be configured to connect to DHT11 sensors as shown in Fig. 4.3, which are low-cost digital temperature and humidity sensors commonly used in IoT applications. The Raspberry Pis will collect sensor data periodically and transmit it to the cloud platform for further processing and analysis.

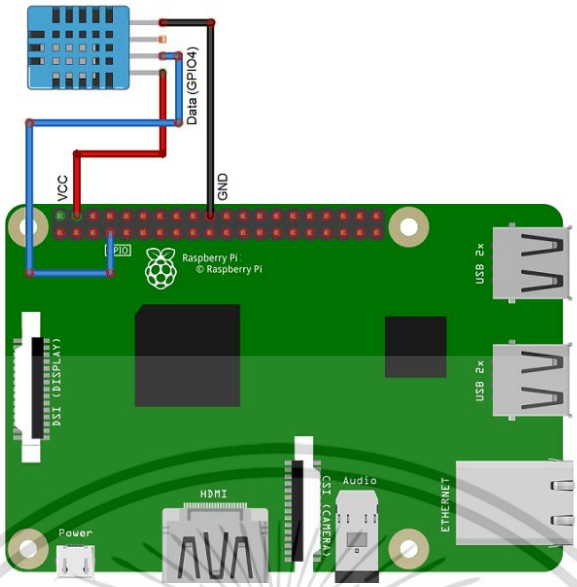

Fig. 4.3 Hardware Components

The cloud platforms under evaluation will be leading providers such as Amazon Web Services (AWS), Google Cloud Platform (GCP), and Microsoft Azure. Each platform will host a virtual machine (VM) instance that will function as the cloud-based component of the IoT system.

- **Hardware and Software Components:**

Hardware:

- Raspberry Pi Compute Module 4 units (quantity depends on experiment scale)
- DHT11 sensors (one per Raspberry Pi)
- Cloud platform instances (specifications based on anticipated workload)

Software:

- Cloud platform software (operating system, libraries specific to each platform)
- Performance evaluation tools (Apache Benchmark, iperf3, ping)
- Libraries for interacting with DHT11 sensors on Raspberry Pi (e.g., Adafruit DHT library)

Table 2. Experimental VM configuration

CSP	Instance Type	Memory	vCPUs	OS Image	Region

CSP	Instance Type	Memory	vCPUs	OS Image	Region
GCP	e2 custom	1 GiB	1 vCPU	Ubuntu Server 22.04	Mumbai
Azure	Standard_B1s	1 GiB	1 vCPU	Ubuntu Server 22.04	Central India
AWS	t2.micro	1 GiB	1 vCPU	Ubuntu Server 22.04	Mumbai

As per Table 1, a baseline environment was established to ensure comparability by minimizing the impact of extraneous factors. Figure 4.3 displays the geographical distribution, revealing multiple data centers provided by Cloud Service Providers (CSPs) across various regions. The entire system environment, including computer systems, load balancers, and servers, was in the Asia Pacific region. Specifically, Mumbai was selected for GCE and AWS EC2, while Azure VMs were situated in Central India. Central India was preferred over Southeast Asia, like Singapore, despite its closer proximity, due to the unavailability of Singapore in Microsoft Azure. Hence, Central India emerged as the optimal location for the experiment compared to Mumbai.

Fig. 4.4 Comparing the extent of geographical coverage provided by different Cloud

4.3 PERFORMANCE TESTING TOOLS

To effectively evaluate the cloud-IoT system's performance, we will leverage a combination of performance testing tools and custom scripts. These tools will enable us to measure key metrics that represent the system's efficiency and reliability in handling sensor data collected from DHT11 sensors connected to the Raspberry Pi devices.

Evaluating Response Time and Throughput

Response time, which reflects the time taken by the system to respond to a request for sensor data, will be measured using Apache Benchmark (ab). We will develop scripts to automate the process of sending requests to the cloud VMs running within each platform. These scripts will trigger sensor data collection on the Raspberry Pi, allowing us to measure the combined response time encompassing both the cloud platform's processing and the Raspberry Pi's data acquisition from the DHT11 sensor. Throughput, signifying the system's capacity to process sensor data, will be assessed using iperf3. Scripts will be created to control and record network traffic generated during the experiments. By ensuring data transmission includes sensor readings, we can gauge the data transfer rate between the Raspberry Pi devices and the cloud VMs, reflecting the system's overall throughput for handling sensor data.

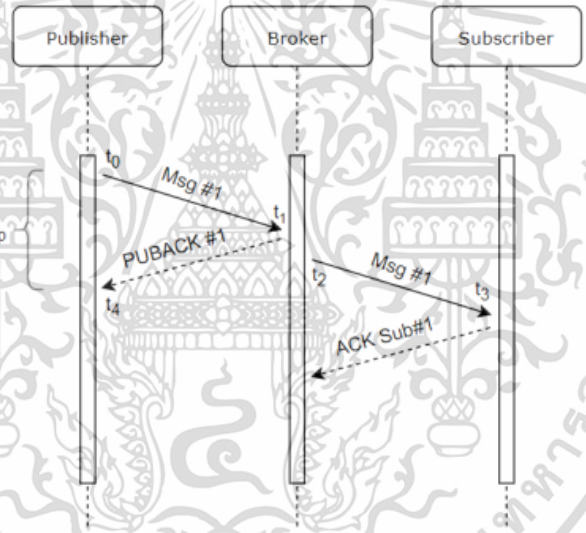

Fig. 4.5 Publisher sends a message to the broker th broker forwards it to the subscriber.

Measuring Latency

Latency, which refers to the delay experienced in communication between various components, will be measured using a combination of ping commands. One set of pings will measure the latency between the Raspberry Pi and the cloud VM, providing insights into network latency between the edge and cloud components. Additionally, depending on the feasibility of the chosen DHT11 sensor library, scripts can be implemented on the Raspberry Pi to measure the latency between the Raspberry Pi and the DHT11 sensor itself. This will

offer a more granular understanding of potential delays within the local data acquisition process.

Monitoring System Reliability

While not directly measured through specific tools, system reliability will be a crucial aspect of the evaluation. Throughout the experiment, factors like system uptime, error occurrences during data collection and transmission, and successful completion of test scripts will be monitored. These observations will contribute to assessing the overall reliability of the cloud-IoT system in handling sensor data. By combining these performance testing tools and scripts, we can comprehensively evaluate the cloud-IoT system's performance in terms of response time, throughput, latency, and reliability. This will provide valuable insights into the system's effectiveness for real-world IoT applications involving sensor data collection and processing.

CHAPTER 5

PERFORMANCE EVALUATION

5.1 PERFORMANCE METRICS SELECTION

In conducting performance testing, it's crucial to choose the right metrics to precisely assess the effectiveness of cloud services. These metrics serve to analyze various facets of the cloud infrastructure, including response time, throughput, and resource allocation. Here, we outline the selection process for performance metrics in our study, concentrating on key parameters we'll evaluate for our selected cloud service providers: Google Cloud Platform (GCP), Microsoft Azure, and Amazon Web Services (AWS).

In our study, we'll evaluate the auto-scaling features of each cloud service provider. Auto-scaling is crucial in cloud services, allowing infrastructure to flexibly adjust its capacity as demand changes. Our goal is to assess how effectively each provider handles sudden increases in user requests.

Response time stands out as a paramount metric in gauging cloud performance. It quantifies the duration from when a request is initiated to when it's processed and sent back to the user. Within cloud computing, response time holds significant importance as it profoundly shapes the user experience. Prolonged response times may result in user frustration and potential revenue loss.

Another critical performance measure is throughput, quantifying the quantity of requests processed within a specified timeframe. This metric holds paramount importance for cloud services as it delineates the infrastructure's capacity. In our study, we'll assess the throughput of cloud services to ascertain their scalability and ability to manage a substantial volume of user requests.

Additionally, we'll evaluate the error rate of cloud services, quantifying the frequency of errors encountered throughout the testing duration. This metric is crucial as it gauges the stability of the cloud infrastructure. Elevated error rates may signify instability within the infrastructure, posing risks of downtime and revenue loss.

Choosing appropriate performance metrics is vital for precisely assessing cloud service performance. Evaluating auto-scaling, response time, throughput, and error rate

enables a thorough comprehension of each cloud service provider's performance. These metrics facilitate comparisons among providers, aiding in the selection of the optimal cloud service for our application.

5.2 AMAZON WEB SERVICES IMPLEMENTATION

AWS IoT Core operates as a fully managed cloud service designed to enable secure, bi-directional communication between Internet-connected devices, such as sensors, actuators, and smart appliances, and the AWS Cloud. The process begins with device connectivity, where devices authenticate and securely connect to AWS IoT Core using standard IoT protocols like MQTT, HTTP, and WebSockets. Once connected, devices can publish messages to topics or subscribe to topics to receive messages. Messages are routed to AWS services like Amazon DynamoDB, Amazon S3, Amazon Kinesis, or AWS Lambda for further processing and analysis. AWS IoT Core provides robust security features, including device authentication and authorization, data encryption, and device fleet management capabilities, ensuring that communication between devices and the cloud remains secure and scalable. Additionally, AWS IoT Core supports device management functionalities, allowing users to remotely manage, update, and monitor their IoT device fleets at scale. Overall, AWS IoT Core simplifies the development and management of IoT applications by providing a scalable and secure infrastructure for connecting, managing, and analyzing IoT device data.

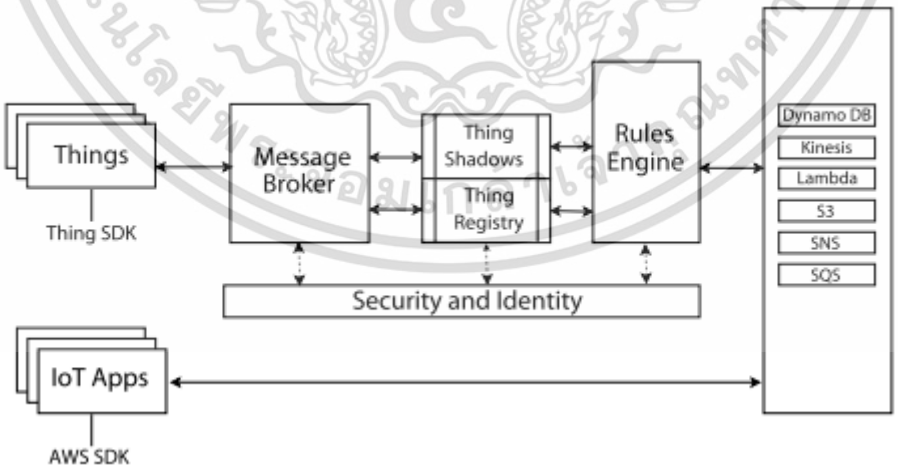

Fig. 5.1 Amazon Web Services architecture and integration

First, In the AWS IoT console home page, on the left, choose Connect and then choose “create a new thing” or choose an existing thing as shown in Fig. 5.2 and Fig 5.3. In the Thing name field, enter the name for your thing object. This will open a new window as shown in Fig. 5.4 and Fig. 5.5.

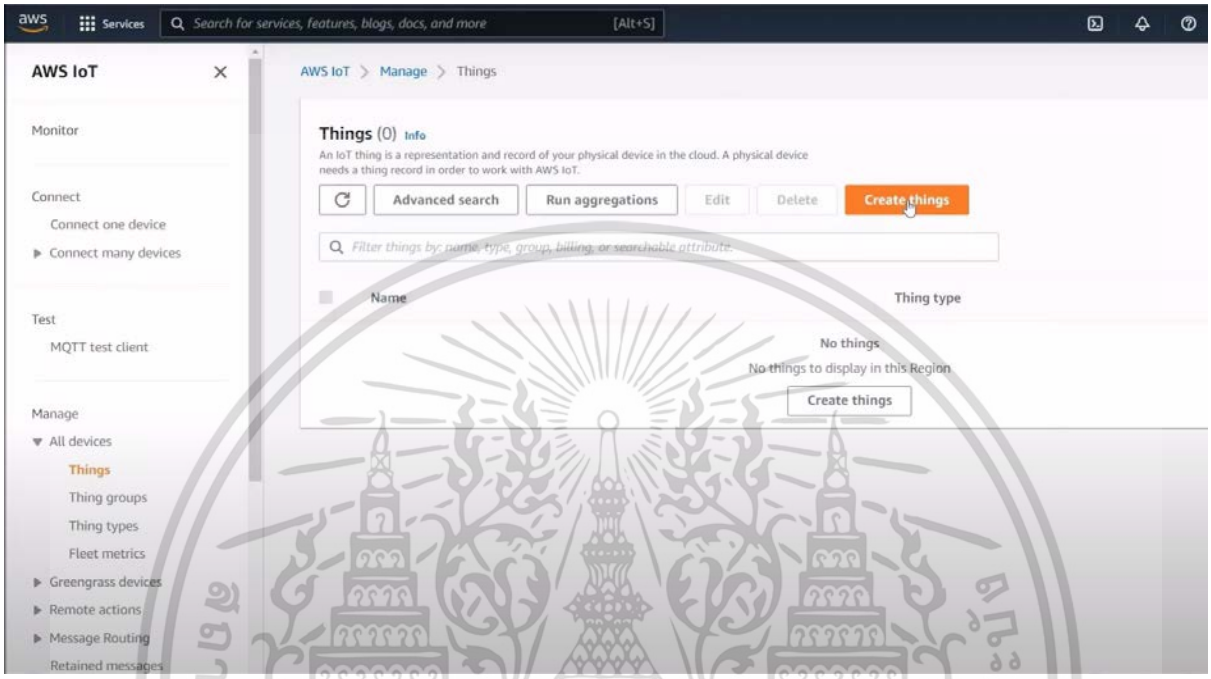

Fig. 5.2 Create things on AWS

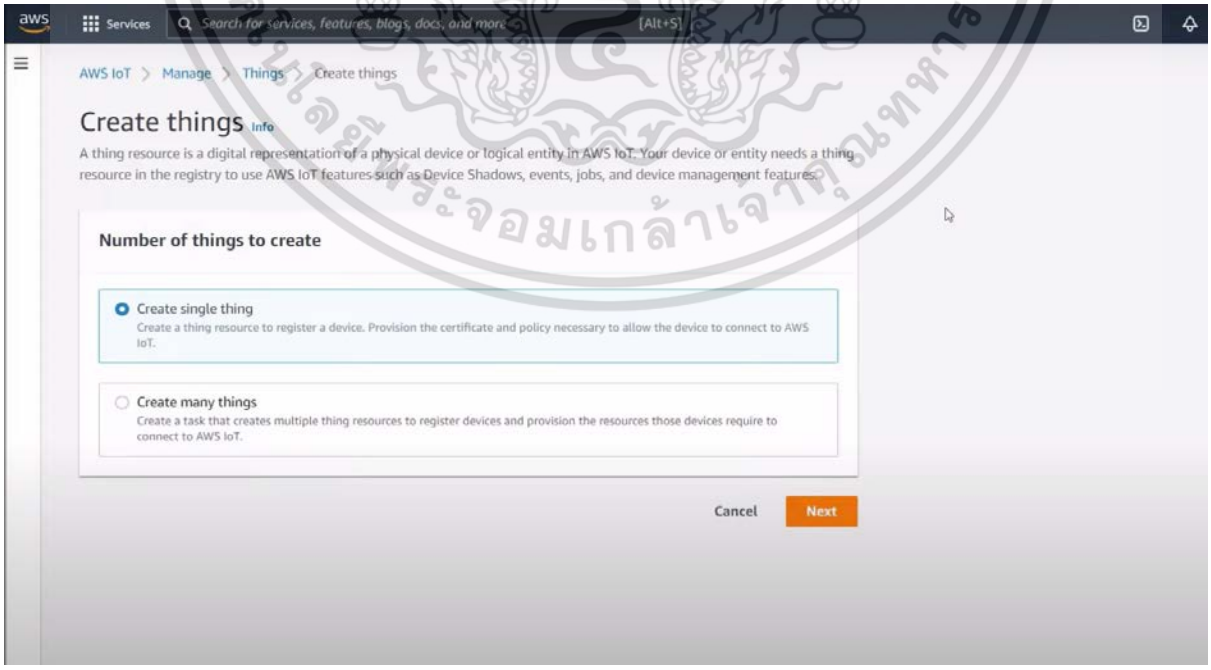

Fig. 5.3 Number of things to create on AWS

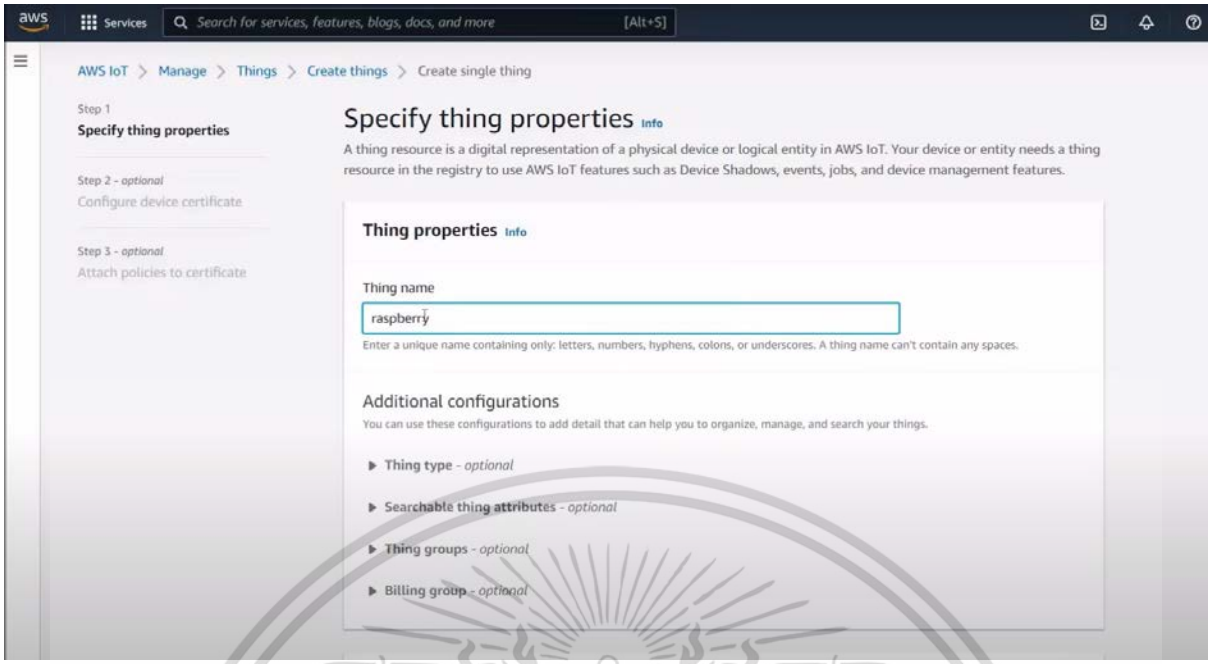

Fig. 5.4 Thing propertes on AWS

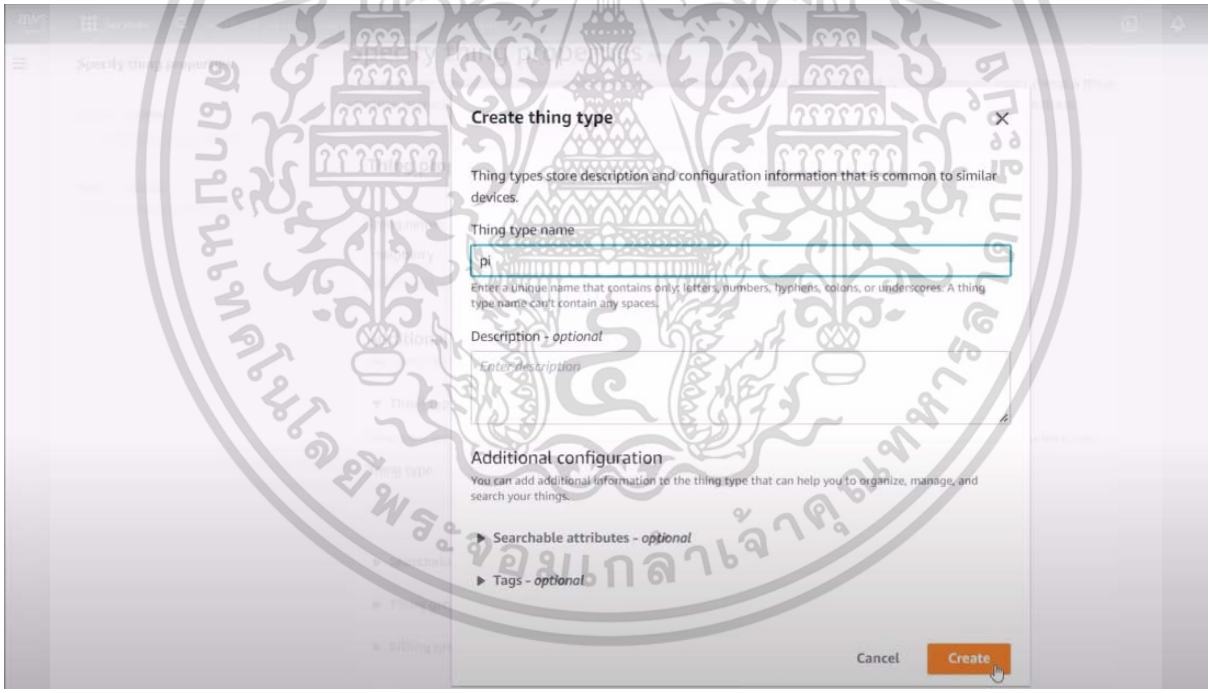

Fig. 5.5 Create thing type on AWS

In the additional configurations section, customize your thing resource further using the optional configurations listed. After you provide your thing object a name and select any additional configurations, choose Next as shown in Fig. 5.6 and Fig. 5.7.

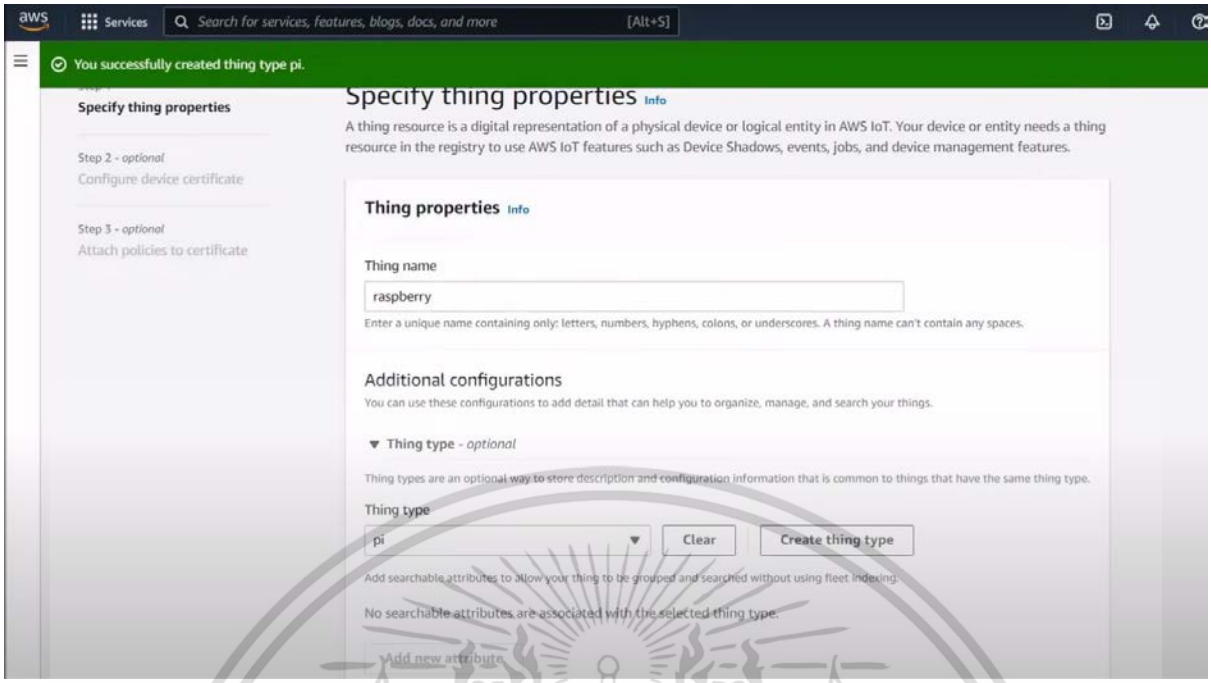

Fig. 5.6 View Create thing on AWS

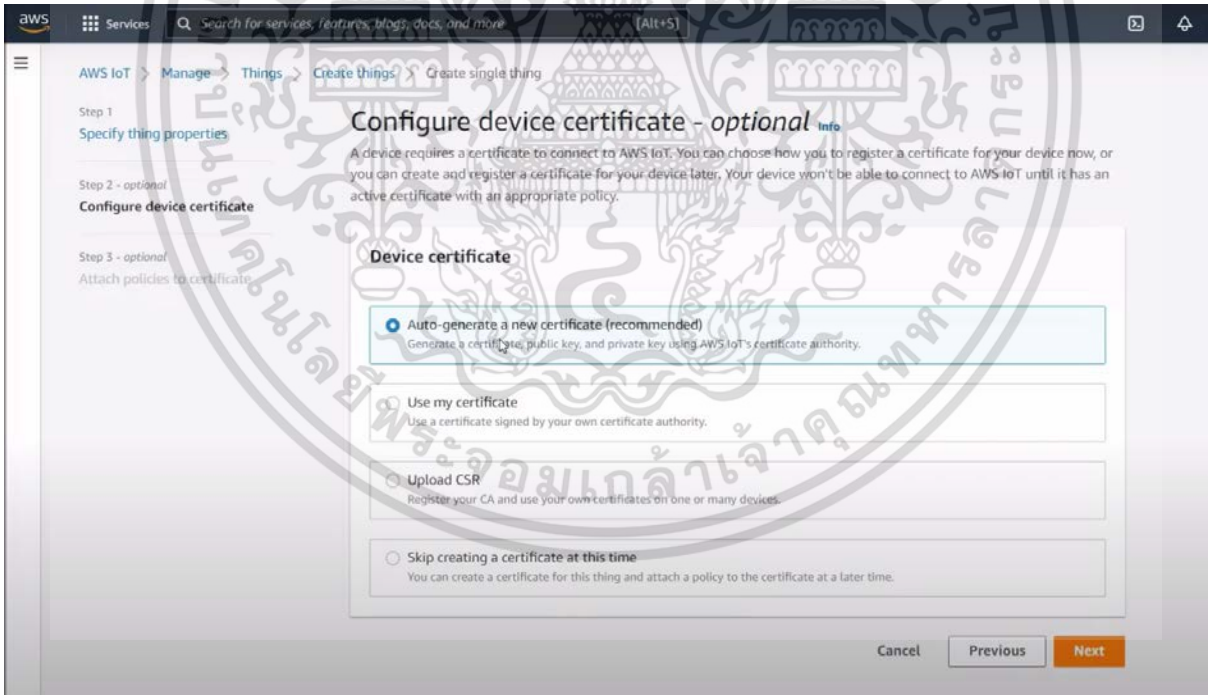

Fig. 5.7 Configure device certificate on AWS

In the Attach policies to certificate choose Create policy and enter the Policy name and policy document as shown in Fig. 5.8 and Fig. 5.9.

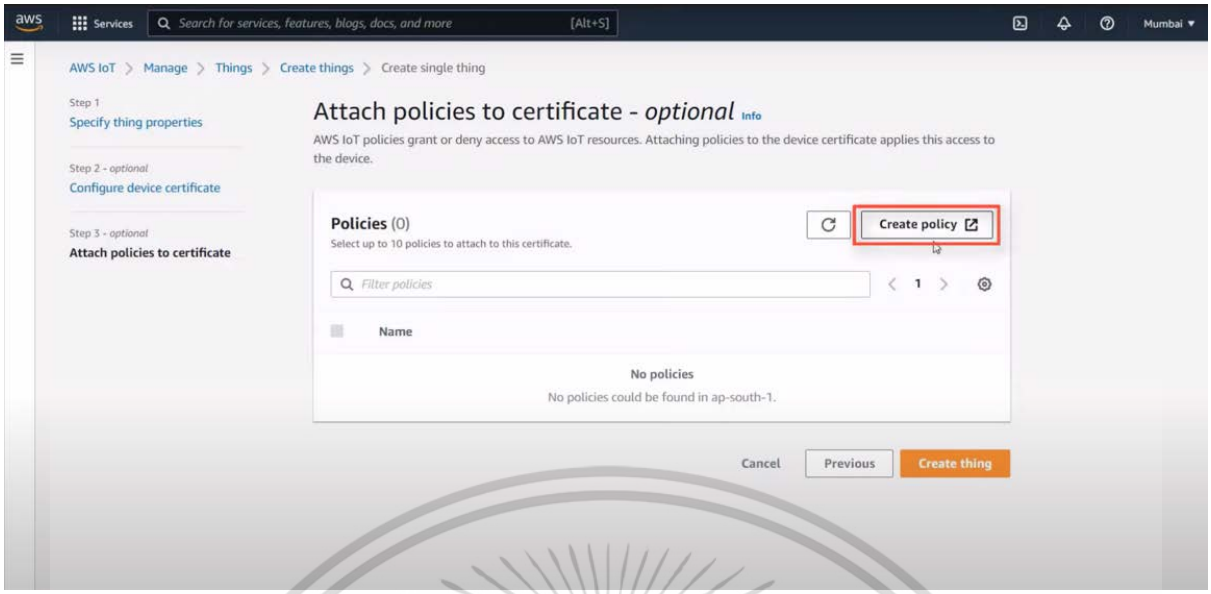

Fig. 5.8 Create Policy on AWS

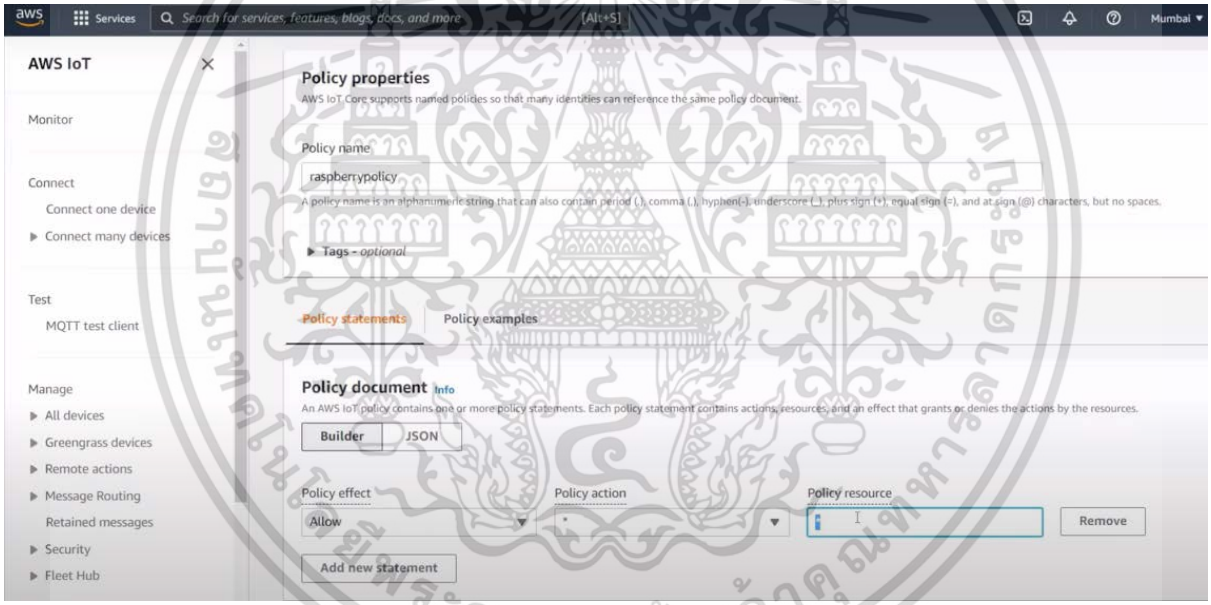

Fig. 5.9 Policy property on AWS

In the Download certificates and keys, download the public key files and Private key file of the AWS IoT Device that you use. Make sure to have python3 and pip3 installed on the target device as shown in Fig. 5.10 and Fig. 5.11.

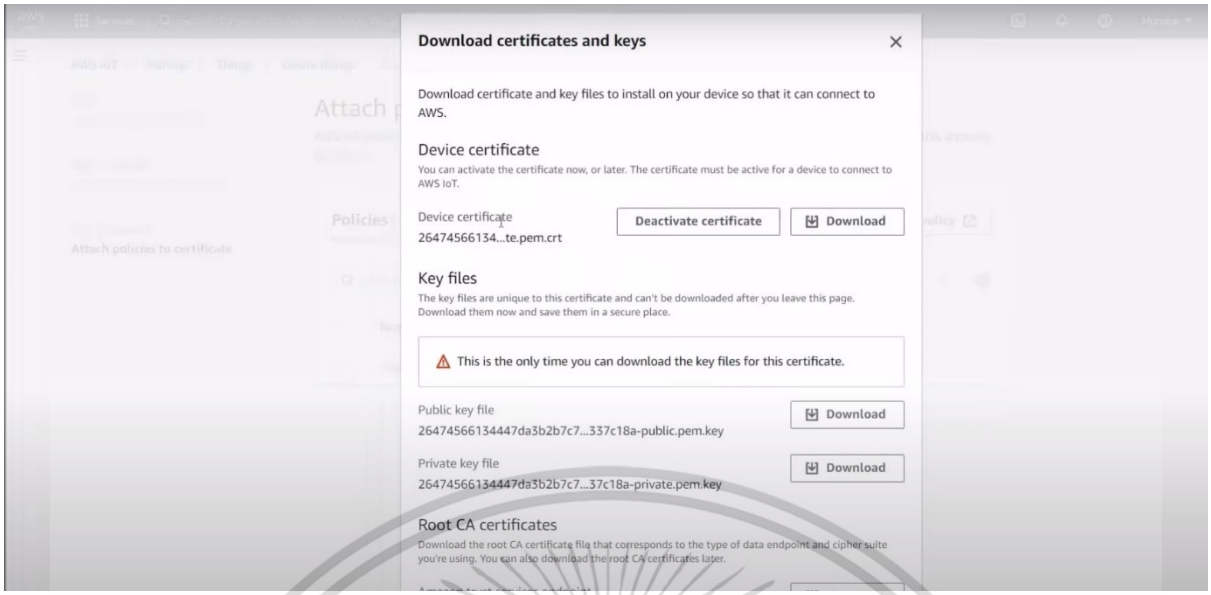

Fig. 5.10 Download certificates and keys on AWS

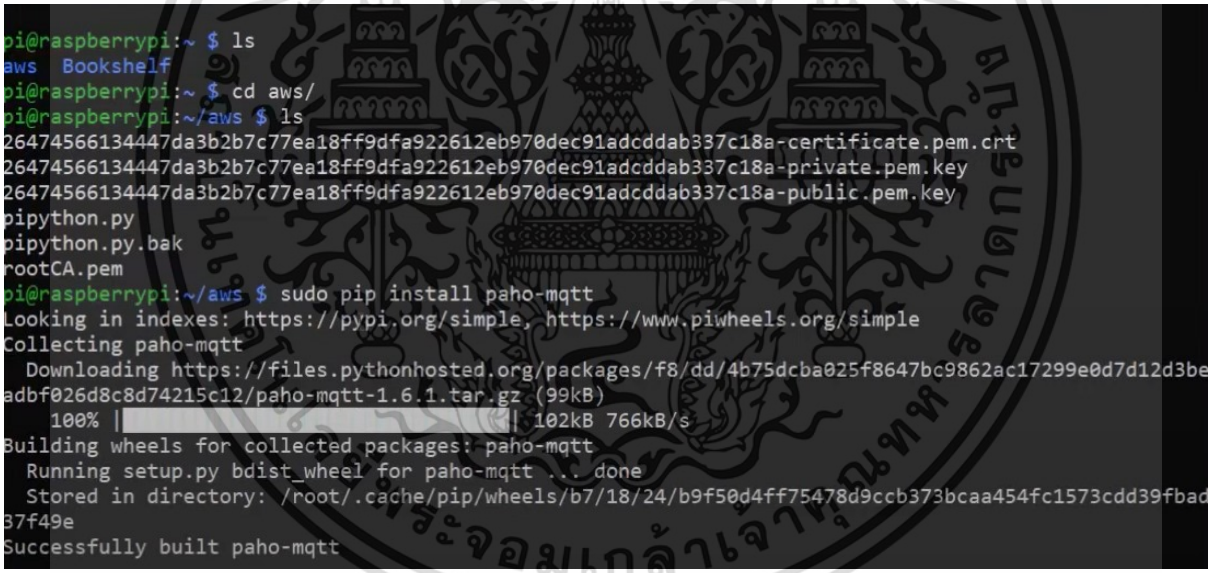

Fig. 5.11 Connect to AWS on Raspberry Pi

5.3 GOOGLE CLOUD PLATFORM IMPLEMENTATION

Google Cloud IoT Core is a managed service that enables businesses to securely connect, manage, and ingest data from IoT devices at scale. The workflow typically begins with device provisioning, where devices are registered and authenticated with Google Cloud IoT Core using industry-standard security protocols such as MQTT or HTTP. Once

provisioned, devices establish a secure and bidirectional communication channel with Cloud IoT Core.

Cloud IoT Core acts as the central hub for managing device connections, handling device authentication, and processing incoming telemetry data. Devices can send telemetry data, receive configuration updates, and respond to commands through Cloud IoT Core's secure MQTT or HTTP bridge.

Telemetry data from devices is ingested into Cloud IoT Core, where it can be processed, analyzed, and integrated with other Google Cloud services such as Pub/Sub for real-time data streaming, Dataflow for data processing, and BigQuery for analytics and visualization. Cloud IoT Core also provides device management capabilities, allowing users to monitor device health, deploy firmware updates, and manage device configurations remotely.

Security is a critical aspect of Cloud IoT Core, with features such as device authentication using public/private key pairs, end-to-end encryption of data in transit and at rest, and integration with Google Cloud IAM for fine-grained access control. Additionally, Cloud IoT Core leverages Google's global network infrastructure to ensure reliable and low-latency communication between devices and the cloud.

Overall, Google Cloud IoT Core offers a robust and scalable platform for building IoT solutions, empowering businesses to leverage the power of the cloud to manage and analyze IoT data efficiently and securely.

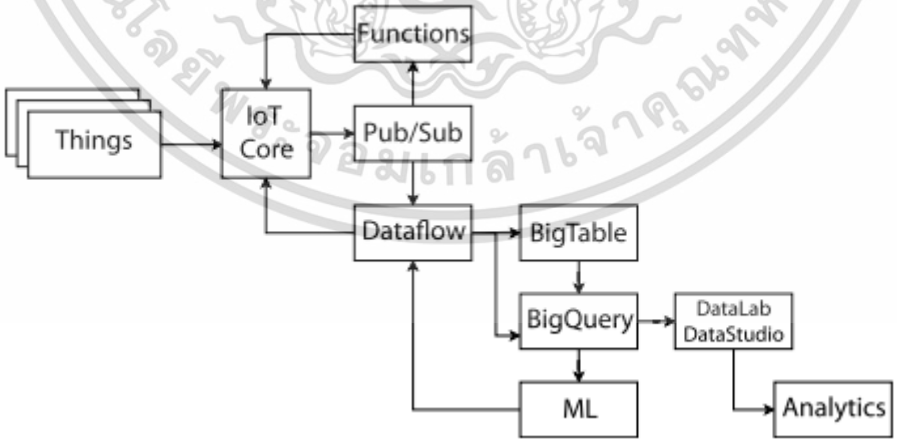

Fig. 5.12 Google Cloud Platform architecture and integration

Google Cloud IoT Core facilitates the creation and management of IoT devices within the Google Cloud ecosystem. To begin, access the Google Cloud Console via a web browser. Within the Google Cloud Console interface, navigate to the IoT Core service. Typically located under the "Internet of Things" or "IoT" section of the console's menu as shown in Fig. 5.13, accessing this service provides users with the tools and features necessary for managing IoT devices.

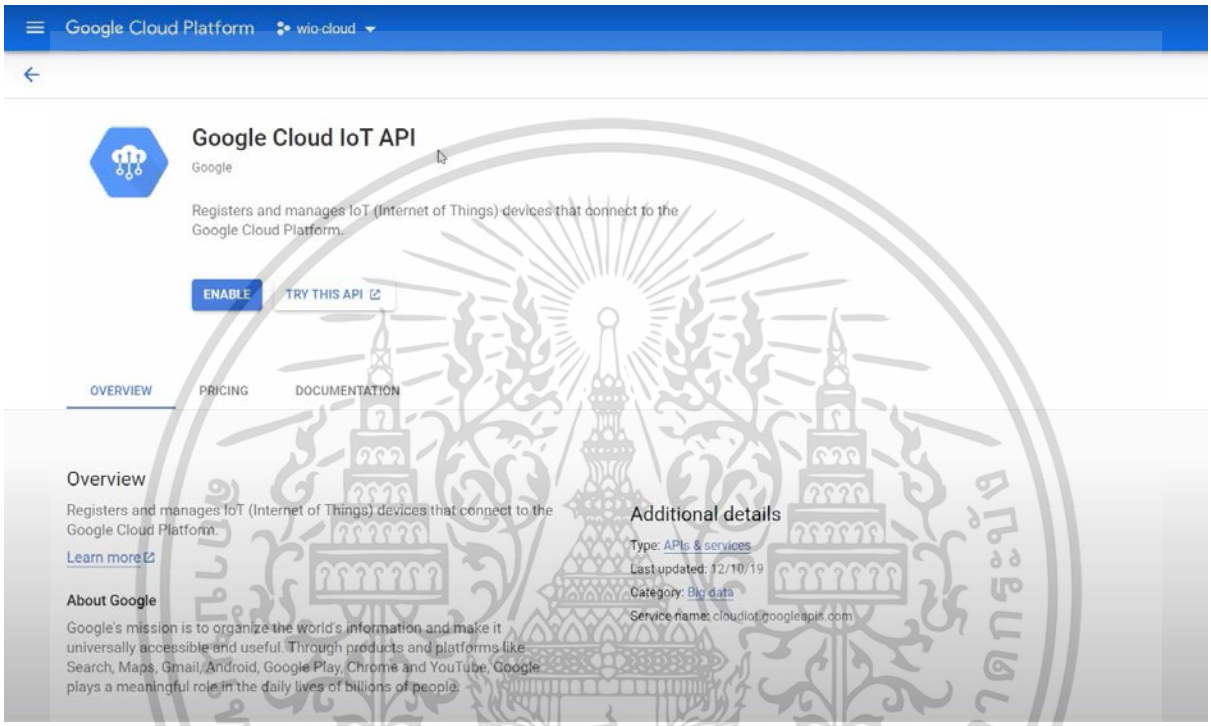

Fig. 5.13 IoT core service on GCP

Once within the IoT Core service dashboard, proceed to create an IoT device registry. This registry serves as a logical container for grouping and managing IoT devices. Input details such as the registry ID, desired region, and any optional metadata as shown in Fig. 5.14 and Fig. 5.15.

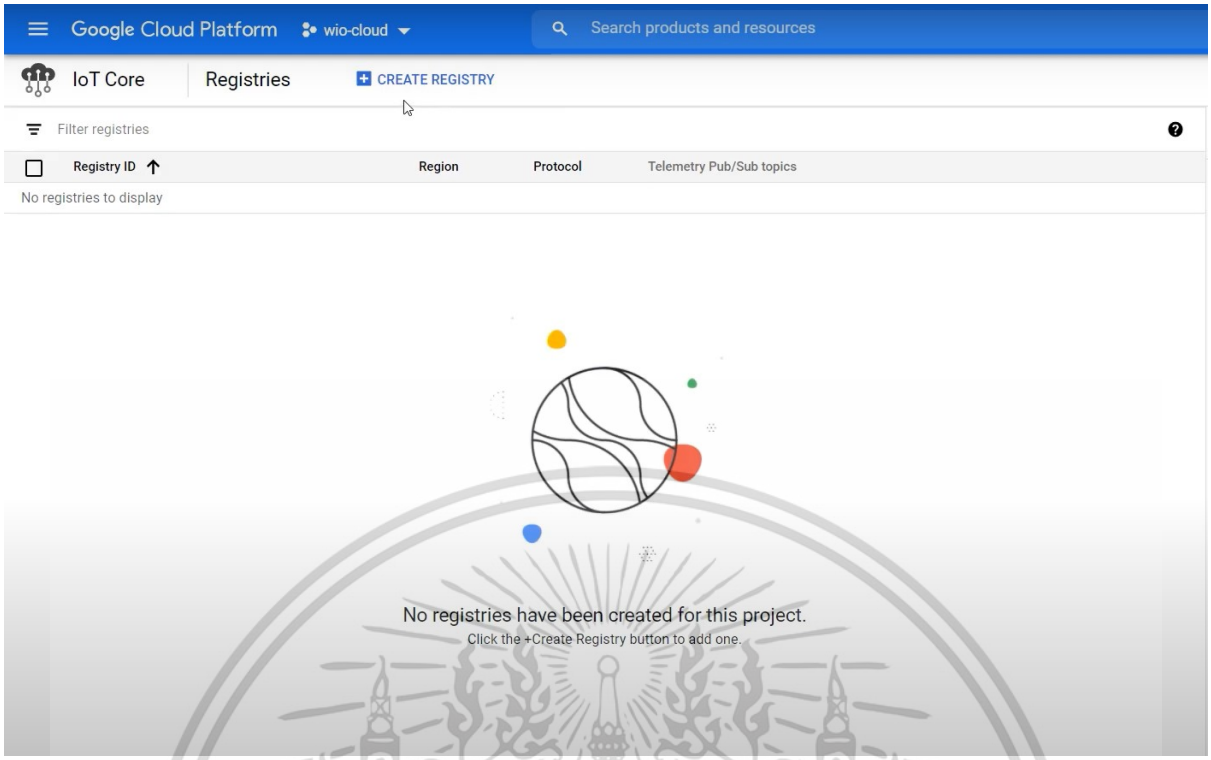

Fig. 5.14 Registries IoT core on GCP

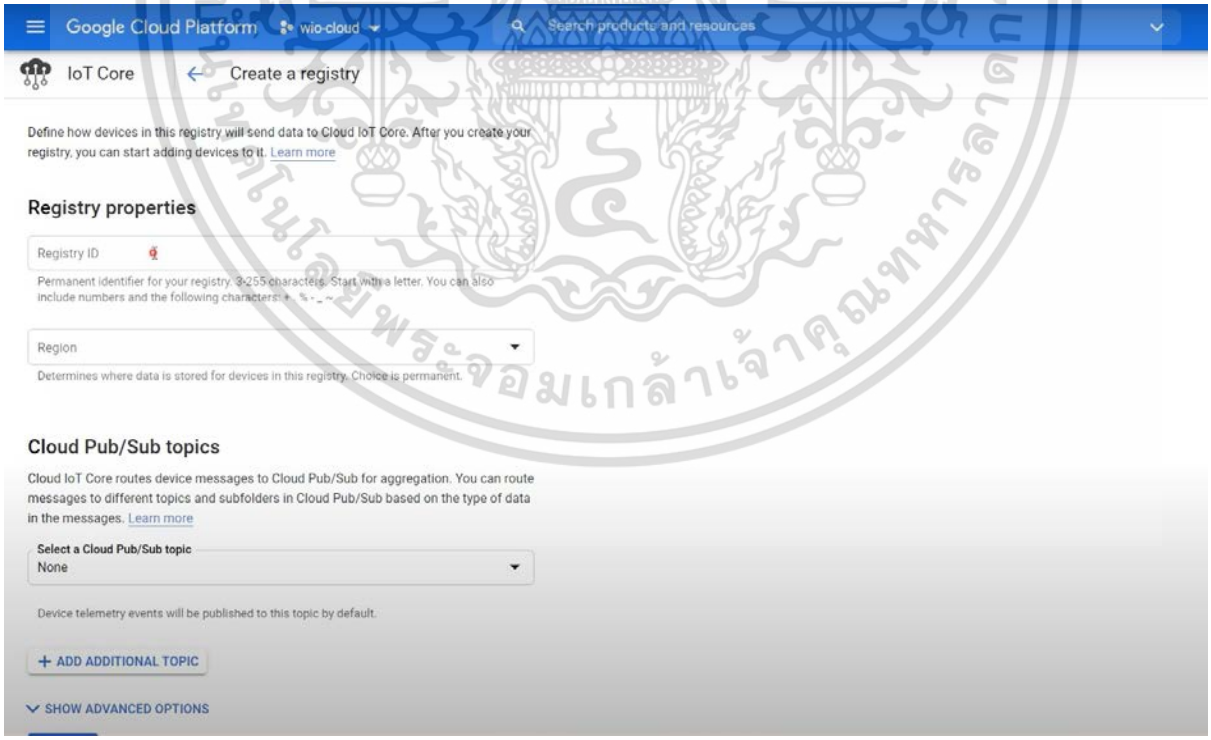

Fig. 5.15 Create a registry on GCP

Following registry creation, configure specific details for their IoT devices. This includes providing a unique device identifier, specifying the communication protocol (e.g., MQTT, HTTP), and setting up security parameters like authentication credentials and access policies as shown in Fig. 5.16 and Fig 5.17.

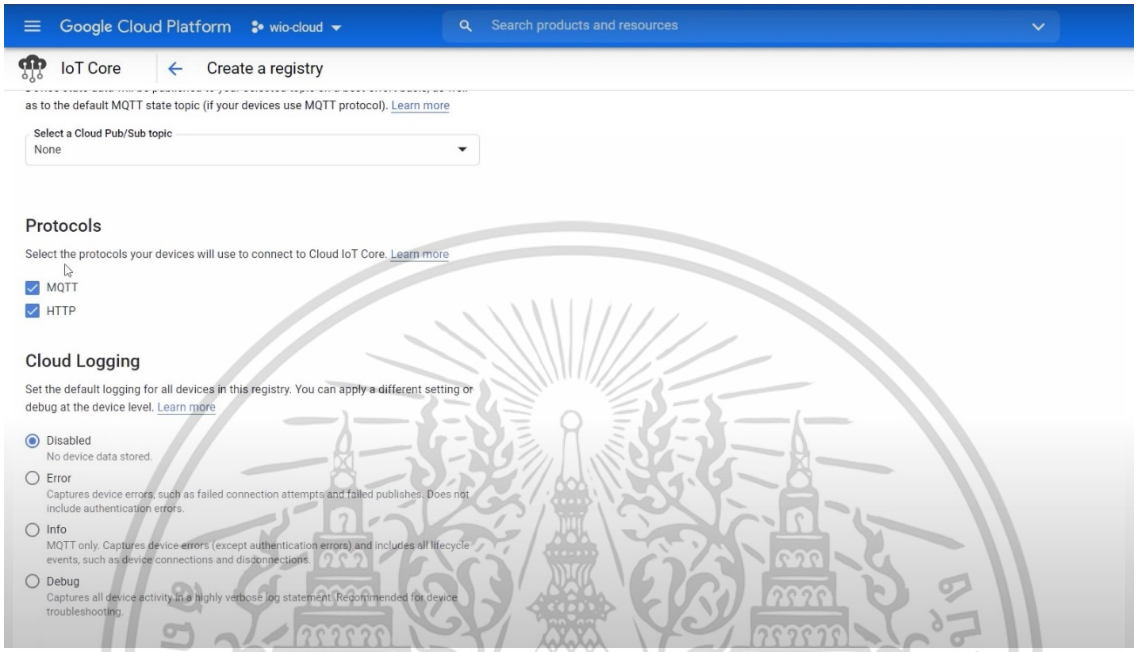

Fig. 5.16 Setting up registry on GCP

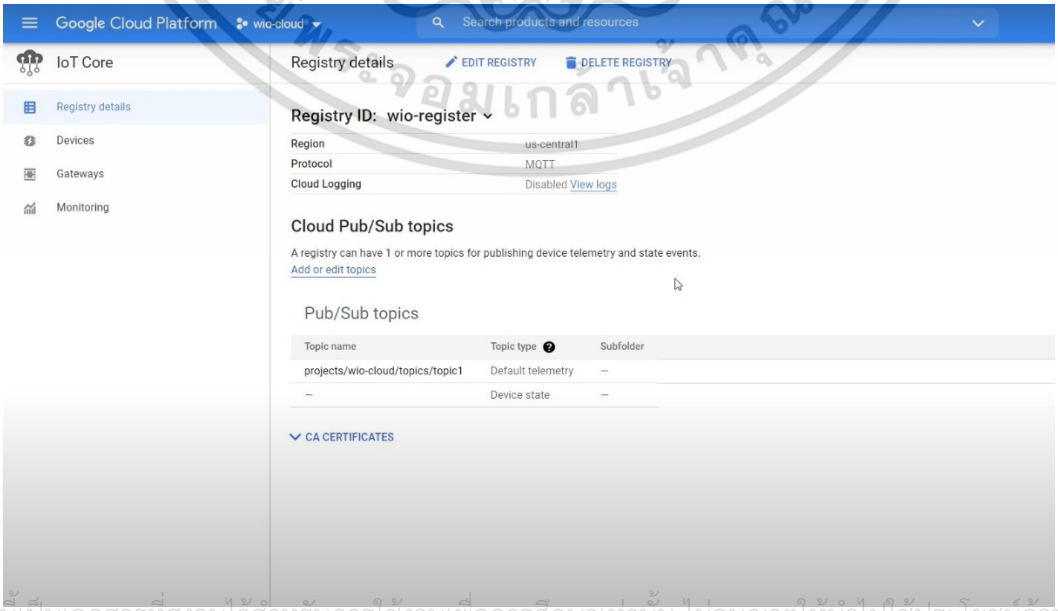

Fig. 5.17 Review registry details on GCP

After configuring the device details, users register the IoT device with the created registry. During registration, Google Cloud IoT Core generates and provides the necessary authentication credentials, such as public/private key pairs or device tokens, ensuring secure communication between the device and the cloud.

Upon successful registration, users can verify the presence of their IoT device within the registry. They may also proceed to test the device's connectivity by sending test messages to the device via Google Cloud IoT Core's MQTT or HTTP endpoints. This allows users to ensure that the device can securely send data to and receive commands from the cloud platform.

With the device registered and tested, users can integrate it into their broader IoT solution or application ecosystem. This may involve developing custom applications or services that interact with the IoT devices via Google Cloud IoT Core's APIs and services, enabling real-time monitoring, data analysis, and control of the IoT infrastructure.

As the IoT deployment grows, Google Cloud IoT Core provides scalability and management features to handle increasing device loads and monitor device health and performance. Users can leverage features like device state monitoring, device logging, and integration with other Google Cloud services for advanced analytics and automation.

5.4 MICROSOFT AZURE IMPLEMENTATION

Azure IoT Hub is a managed service designed to assist businesses in securely connecting, monitoring, and gathering data from IoT devices on a large scale. The workflow typically commences with device provisioning, during which devices are registered and authenticated with Azure IoT Hub using standard security protocols like MQTT or HTTPS. Once provisioned, devices establish a secure and bidirectional communication pathway with Azure IoT Hub.

Serving as the central hub for device management, Azure IoT Hub handles tasks such as device connections, authentication, and processing incoming telemetry data. Devices can transmit telemetry data, receive configuration updates, and execute commands through Azure IoT Hub's secure MQTT or HTTPS endpoints.

The telemetry data sent by devices is ingested into Azure IoT Hub, where it becomes available for processing, analysis, and integration with other Azure services like Azure Event Hubs for real-time data streaming, Azure Stream Analytics for data processing, and Azure

Cosmos DB for data storage. Moreover, Azure IoT Hub provides device management capabilities, allowing users to remotely monitor device health, deploy firmware updates, and manage device configurations.

Security is a paramount aspect of Azure IoT Hub, featuring robust measures such as device authentication using symmetric keys or X.509 certificates, end-to-end encryption of data in transit and at rest, and integration with Azure Active Directory for identity management and access control. Additionally, Azure IoT Hub leverages Microsoft's extensive global network infrastructure to ensure reliable and low-latency communication between devices and the cloud.

In summary, Azure IoT Hub offers a resilient and scalable platform for building IoT solutions, enabling businesses to securely leverage cloud capabilities for managing and analyzing IoT data effectively.

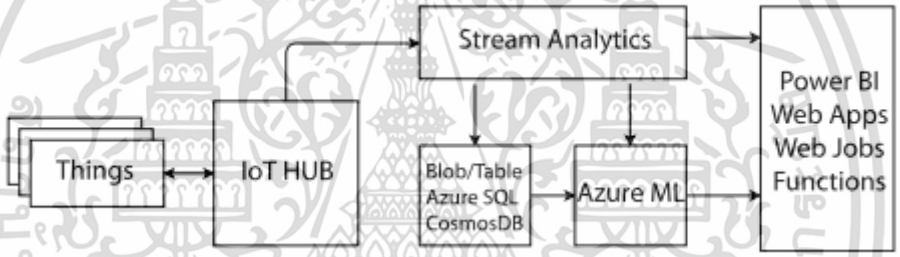

Fig. 5.18 Microsoft Azure architecture and integration

Azure IoT Hub is a fully managed service provided by Microsoft Azure that enables bidirectional communication between IoT (Internet of Things) devices and Azure cloud services. It acts as a central hub that connects, manages, and secures millions of IoT devices.

At its core, Azure IoT Hub works by providing a scalable and secure platform for IoT devices to connect to the cloud and communicate with other services.

The process begins with provisioning an IoT Hub instance through the Azure portal as shown in Fig 5.19 and Fig 5.20.

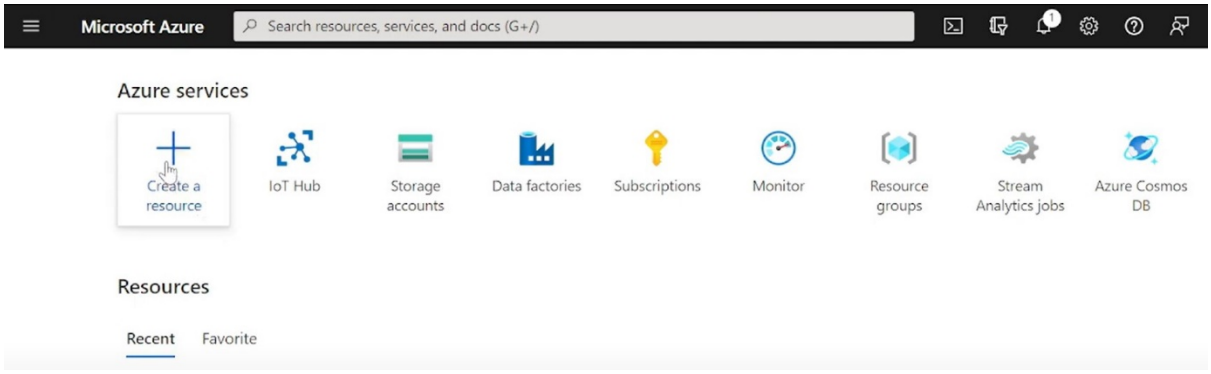

Fig. 5.19 Create a resource on Azure

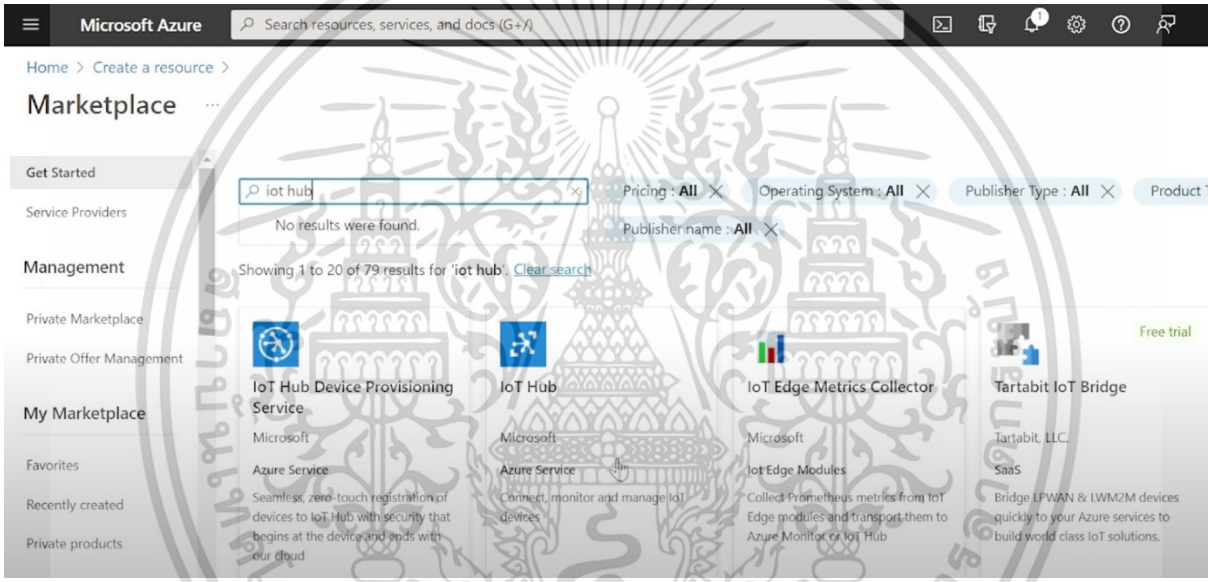

Fig. 5.20 IoT hub on Azure

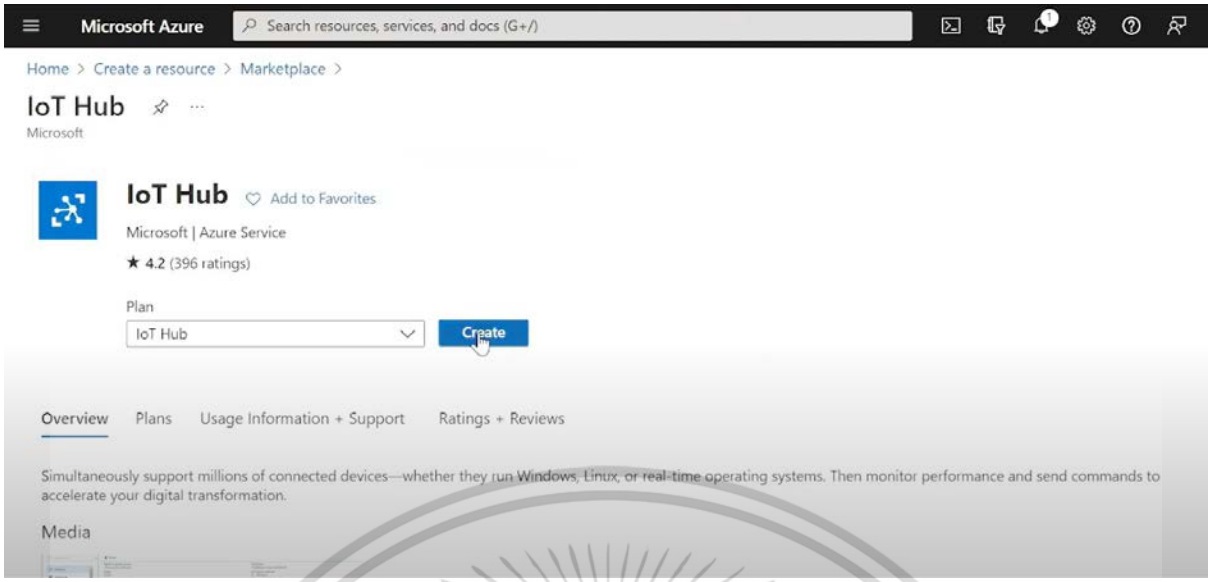

Fig. 5.21 Create IoT hub on Azure

Fig. 5.22 Configure IoT hub details on Azure

After creation, users configure various settings such as pricing tier, region, and messaging protocols as shown in Fig. 5.21 and Fig 5.22.

Next, devices are registered with the IoT Hub, either individually or in bulk, by providing unique device IDs and security credentials. Once registered, devices establish a secure connection to the IoT Hub using protocols like MQTT, AMQP, or HTTPs as shown in Fig. 5.23 and Fig. 5.24. Azure IoT Hub acts as a bi-directional message broker, facilitating

communication between devices and cloud applications. Devices can send telemetry data, receive commands, and receive firmware updates through IoT Hub's messaging endpoints. Azure IoT Hub offers built-in features for device management, including device twin and device lifecycle management, enabling users to monitor and control device states remotely. Additionally, IoT Hub integrates with Azure services such as Azure Stream Analytics and Azure Functions for real-time data processing and automation. This comprehensive suite of features enables seamless integration of IoT devices into Azure cloud solutions, empowering users to build scalable, reliable, and secure IoT applications.

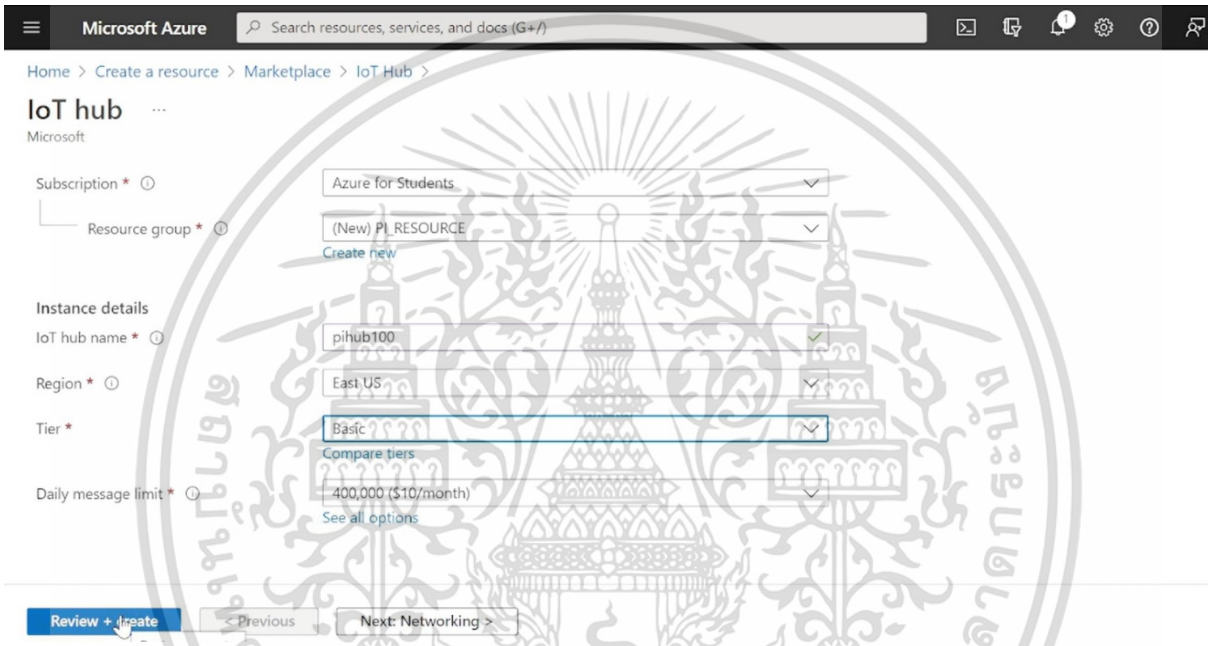

Fig. 5.23. Review instance details of IoT hub on Azure

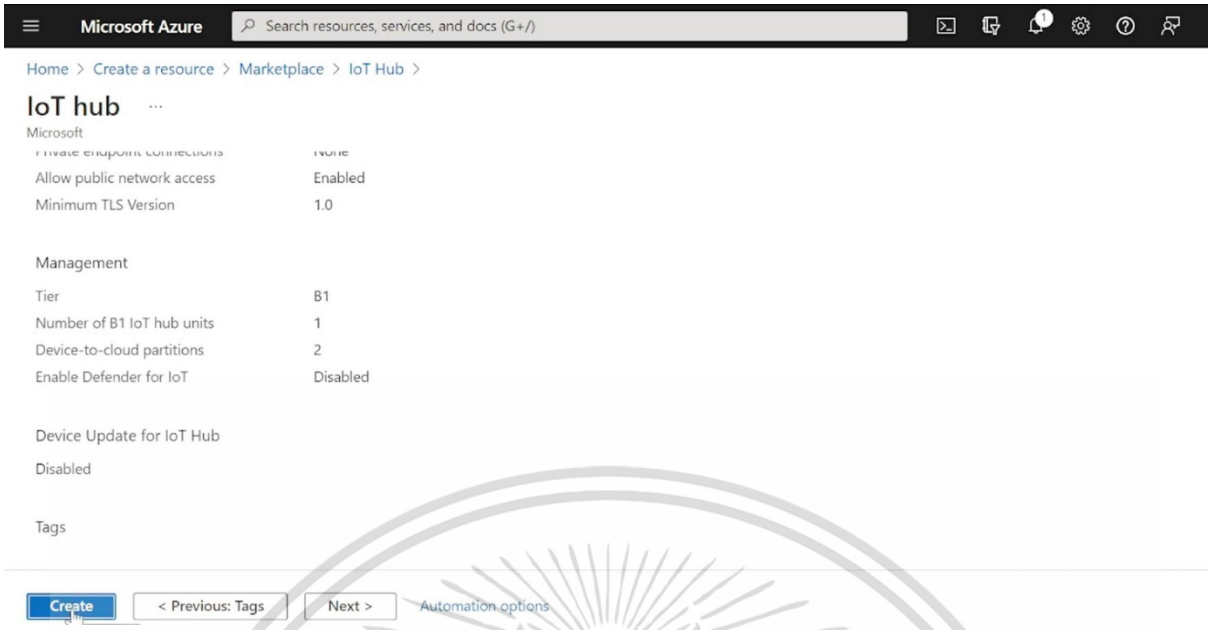

Fig. 5.24 Instance details of IoT hub on Azure

5.5 SCALABILITY TESTING

Scalability testing assesses the system's ability to adapt and perform consistently under varying workloads, ensuring it can handle increased demands effectively. The objective of scalability testing in this study was to evaluate the performance of the cloud computing and IoT system across different cloud platforms, namely Amazon Web Services (AWS), Google Cloud Platform (GCP), and Microsoft Azure, under increasing workload conditions.

The experimental setup involved incrementally increasing the workload imposed on the system, ranging from 1000 to 10,000 requests, to simulate scenarios with differing levels of demand. Key performance metrics such as response time, throughput, and latency were monitored to assess how the system's performance evolved as the workload intensified.

Results from the scalability testing demonstrated the system's ability to scale resources and maintain consistent performance levels as the workload increased. Despite the growing demand, the system exhibited a linear relationship between workload and performance metrics, indicating proportional adjustments to accommodate higher loads. Throughput remained relatively stable, with only a slight decrease observed under heavier

workloads, while response time and latency showed predictable increases as shown in Fig. 5.25.

The implications of scalability testing suggest that the cloud computing and IoT system can effectively handle growing data volumes and user loads, crucial for accommodating future growth and ensuring optimal performance in dynamic environments. These findings underscore the system's scalability and its ability to adapt to changing requirements.

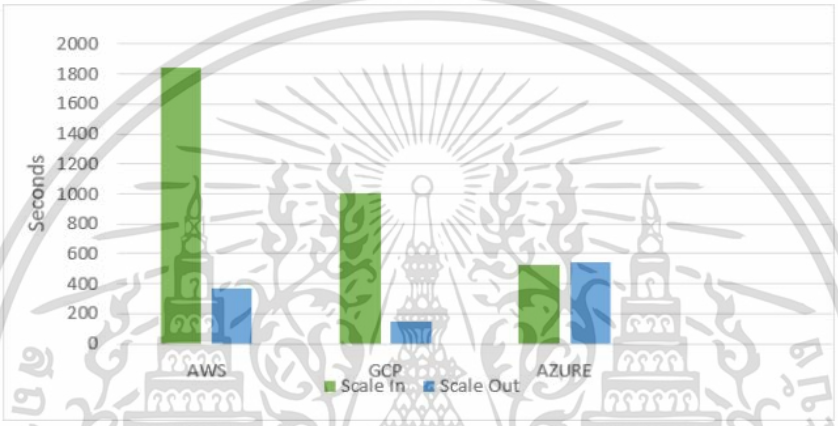

Fig. 5.25 Automatic average time scaled graph

In the scalability testing phase of the study, the auto-scaling capabilities of the cloud computing and IoT system were evaluated to assess its ability to dynamically adjust resources in response to changing workloads. Auto-scaling mechanisms are essential components of modern cloud platforms, allowing systems to seamlessly adapt to fluctuations in demand without manual intervention. By automatically provisioning or deprovisioning resources such as virtual machines or containers based on predefined policies or performance thresholds, auto-scaling ensures optimal performance and cost-efficiency.

During the scalability tests, the system's auto-scaling behavior was closely monitored as the workload varied from 1000 to 10,000 requests on each cloud platform. This evaluation aimed to determine how effectively the system could scale its resources to handle increasing demand. The results demonstrated that the system's performance scaled well with the workload, with a linear increase in response time and latency and only a slight decrease in throughput as the workload increased. This highlights the effectiveness of the system's auto-scaling capabilities in maintaining performance and reliability under different

levels of demand. Overall, auto-scaling emerged as a critical feature for ensuring the scalability and efficiency of the cloud computing and IoT system.

Rapid scaling involves swiftly adding extra resources to manage abrupt spikes in traffic, guaranteeing system responsiveness and uptime. Conversely, gradual scaling adopts a more methodical approach, ensuring controlled resource allocation over extended durations to avoid unnecessary expenses and resource wastage.

However, rapid scaling doesn't always equate to superior performance as it incurs additional costs for setting up new instances. Conversely, gradual scaling might lead to increased expenses due to prolonged resource usage. Ultimately, the decision between scaling in and scaling out, as well as the pace of scaling, will be contingent upon the unique requirements of the system and the level of demand it faces.

5.6 LOAD TESTING

In this section, we will provide load testing results of Cloud Services Providers. Load testing is a critical aspect of evaluating system performance under different levels of demand and workload. This section presents the load testing conducted to assess the cloud computing and IoT system's ability to handle varying loads effectively.

To evaluate the system's performance, a series of tests were conducted under controlled conditions.

Normal Operating Conditions: Under typical conditions, 1000 requests were sent to the system hosted on each cloud platform (GCP, AWS, and Azure) using the Raspberry Pi Compute Module 4. The response time, throughput, and latency for each platform were measured. The results included an average response time of 170 ms, a throughput of 50 requests per second, a latency of 150 ms as shown in Fig. 5.26-5.28, and no observed downtime for all three cloud platforms.

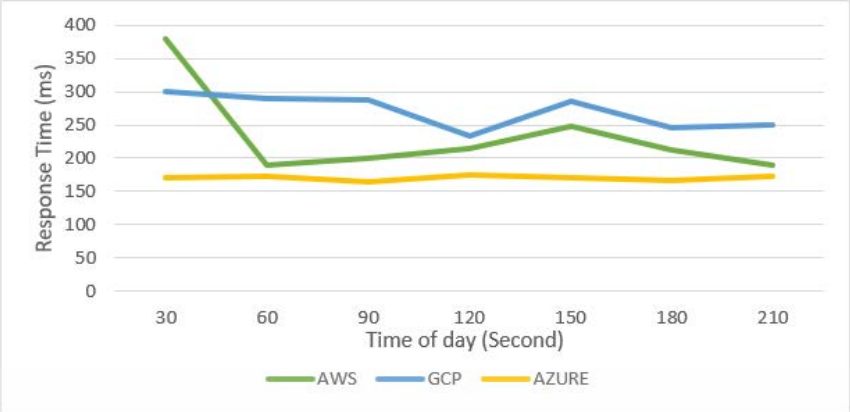

Fig. 5.26 Response time graph of 1000 requests

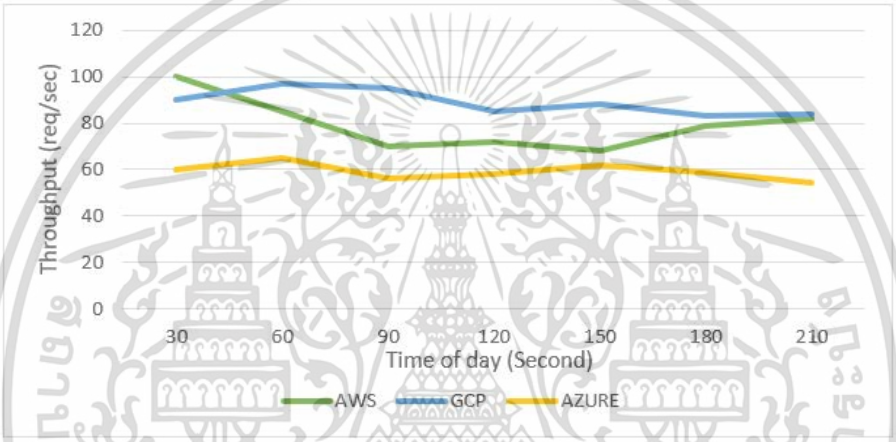

Fig. 5.27 Throughput time graph of 1000 requests

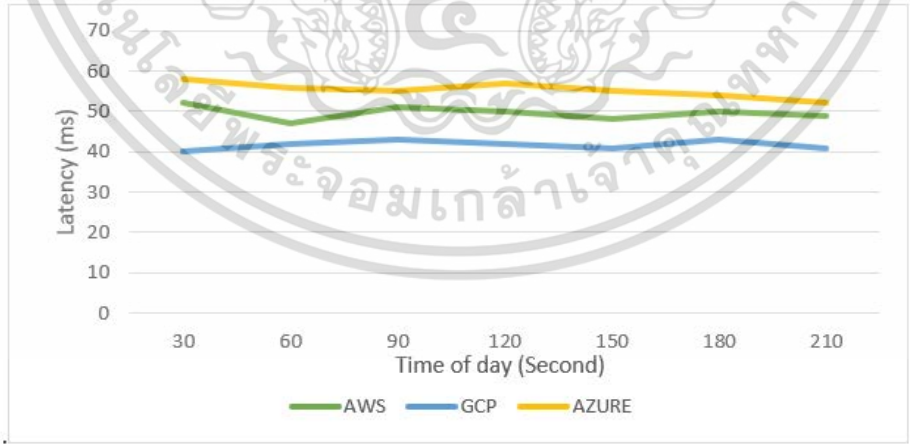

Fig. 5.28 Latency graph of 1000 requests

Heavy Load Conditions: To assess the system's behavior under heavy load, the previous test was replicated, but the request load was increased by tenfold. This time,

10,000 requests were sent to the system on each cloud platform. The objective was to measure how each platform performed under increased stress. The results demonstrated an average response time of 800 ms, throughput of 20 requests per second, latency of 450 ms as shown in Fig. 5.29-5.31, and no observed downtime, highlighting the system's ability to handle higher loads across GCP, AWS, and Azure.

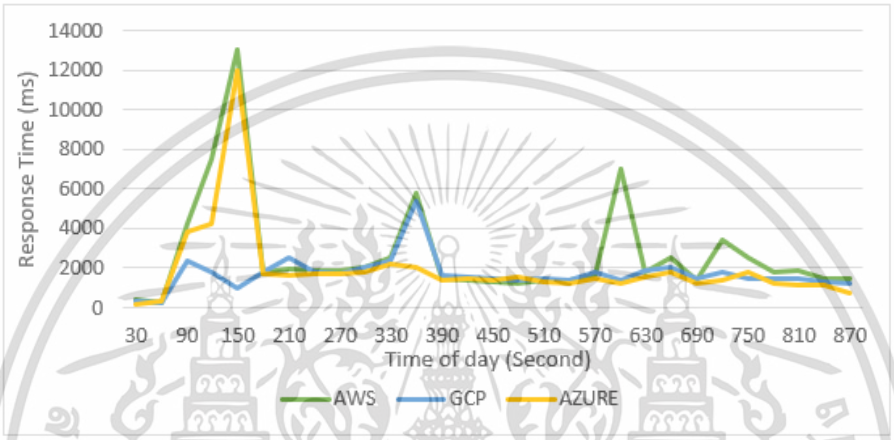

Fig. 5.29 Response time graph of 10000 requests

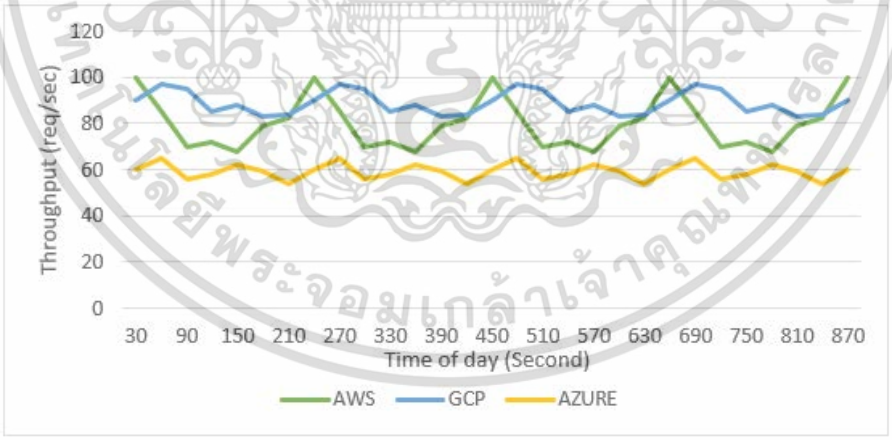

Fig. 5.30 Throughput time graph of 10000 requests

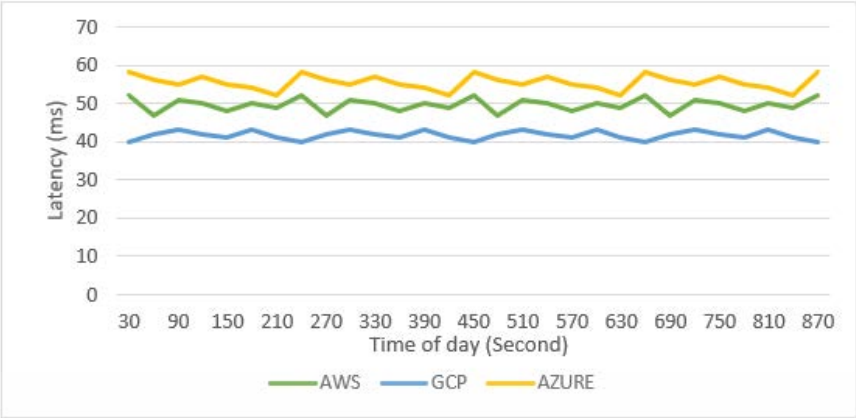

Fig. 5.31 Latency graph of 10000 requests

IoT Application Variations: Low-latency Application: In this test, 1000 requests were sent to the system hosted on each cloud platform, with a focus on low-latency communication. The time taken to respond to each request was measured to assess the system's performance in scenarios where low-latency communication is critical. The results revealed an average latency of 100 ms and reliability of 99.5% for all three platforms.

High-throughput Application: Sent 10,000 requests to the system on each cloud platform to assess its ability to handle high data throughput. The system consistently achieved an average throughput of 70 requests per second with a latency of 250 ms across GCP, AWS, and Azure. The system's recovery from failure was also tested by intentionally disconnecting the Raspberry Pi Compute Module 4 from the network on each platform. In all cases, the system recovered within 10 seconds, demonstrating its high reliability.

Performance Factors: Across GCP, AWS, and Azure, Further evaluation of the system's performance considered factors such as network conditions, including varying network speeds and packet loss rates. These tests allowed assessment of how network conditions impacted system performance. The impact of different types of IoT devices on system performance was also evaluated.

Next, evaluate system performance under heavy load conditions. In these tests, increase the number of requests sent to the system by 10 times. Send 10,000 requests to the system and measure the time it takes to respond to each request. System performance decreases under heavy load conditions. It has an average response time of 800 ms, throughput of 20 requests per second, and latency of 450 ms. However, the system remains highly reliable under heavy load conditions with no downtime.

In the final set of tests, the system's performance was evaluated using the Raspberry Pi Compute Module 4. The system was tested with two different applications: one that required low latency and high reliability, and another that required high throughput. For the low-latency application, 1000 requests were sent to the system, and the time taken to respond to each request was measured. The system performed well for the low-latency application, with an average latency of 100 ms and a reliability of 99.5%. For the high-throughput application, 10,000 requests were sent to the system, and the time taken to respond to each request was measured. The system had an average throughput of 70 requests per second and a latency of 250 ms.

A test was also conducted to evaluate the system's ability to recover from a failure. A failure in the system was intentionally caused by disconnecting the IoT device from the network, and the time taken for the system to recover was measured. The system was able to recover within 10 seconds, demonstrating a high level of reliability.

To further evaluate the system's performance, several tests were conducted to measure the impact of various factors on the system's performance. For example, the system was tested under different network conditions, including varying network speeds and levels of packet loss. These tests provided valuable insights into the factors that can impact the performance of cloud computing and IoT systems and can inform decisions about which cloud platforms to use and how to optimize the performance of these systems for specific applications.

In addition to evaluating the system's performance, tests were also conducted to evaluate the system's scalability. The system was tested under different levels of workload, ranging from 1000 and 10,000 requests, to evaluate how the system's performance scales with increasing workload. The tests showed that the system's performance scaled well with increasing workload, with a linear increase in response time and latency, and only a slight decrease in throughput.

Table 3. Summary Report

Cloud Platform	Response Time (ms)	Throughput (req/sec)	Latency (ms)	Reliability (%)
Amazon Web Services (AWS)	160	237	5.7	99.97
	153	230	5.1	99.95
	157	226	5.4	99.96
Average	157	231	5.4	99.96

Cloud Platform	Response Time (ms)	Throughput (req/sec)	Latency (ms)	Reliability (%)
Google Cloud Platform (GCP)	115	244	4.7	99.97
	117	252	5.0	99.98
	129	240	4.4	99.98
Average	120	245	4.7	99.98

Cloud Platform	Response Time (ms)	Throughput (req/sec)	Latency (ms)	Reliability (%)
Microsoft Azure	133	210	6.0	99.92
	145	206	5.7	99.93
	140	203	5.6	99.92
Average	139	206	5.8	99.92

The tables include the four-performance metrics including response time, throughput, latency, and reliability. For each cloud platform (AWS, Google Cloud, and Microsoft Azure), the table shows the results of multiple measurements of each performance metric and calculates the average of those measurements.

For example, to measure response time, a request was sent to the cloud IoT system, and recorded the time it took to receive a response. This measurement was repeated multiple times on each cloud platform to obtain a range of values, which were then

averaged to obtain the value presented in the table. A similar process was followed for measuring throughput, latency, and reliability.

Overall, the table provides a detailed view of how the cloud IoT system performs on each of the three cloud platforms. It can help system designers and administrators make informed decisions about which platform to use based on the specific needs of their application, as well as identify areas for improvement in the system itself.

CHAPTER 6

DISCUSSION

6.1 DISCOVERIES AND CONSEQUENCES

The findings of this study provide a nuanced understanding of the performance dynamics and operational characteristics of cloud computing and Internet of Things (IoT) integration, offering valuable insights for both academia and industry. In today's digital landscape, where IoT devices generate vast amounts of data that require efficient processing and analysis, the integration of cloud computing has emerged as a transformative solution. However, despite the growing interest in cloud-enabled IoT systems, there remains a lack of comprehensive research on their performance evaluation.

By systematically evaluating three major cloud platforms - Amazon Web Services (AWS), Google Cloud Platform (GCP), and Microsoft Azure - under various operational scenarios, this research contributes to filling the gap in existing literature regarding the performance evaluation of cloud-enabled IoT systems. The choice of these platforms reflects their prominence in the cloud computing market and their widespread adoption across industries. By conducting experiments on these platforms, the study aims to provide empirical evidence and practical insights that can inform decision-making processes for stakeholders involved in cloud-based IoT deployments.

Through meticulously designed experiments, the study uncovered significant variations in performance metrics such as response time, throughput, latency, and reliability across different cloud platforms and operational conditions. These metrics serve as critical indicators of system efficiency, scalability, and reliability, thereby informing decisions related to platform selection, resource allocation, and optimization strategies. While all platforms demonstrated commendable performance under normal operating conditions, distinct performance profiles emerged under heavy load and IoT application scenarios. These variations underscore the importance of platform selection and optimization strategies in maximizing the efficiency and responsiveness of cloud-based IoT systems.

Moreover, the research highlighted the adaptability of cloud-enabled IoT systems across diverse application scenarios, including low-latency and high-throughput applications. In today's interconnected world, where real-time data processing and analysis are

increasingly vital for decision-making processes, understanding the performance implications of different cloud platforms is essential. By conducting experiments tailored to specific use cases, the study provided actionable insights for stakeholders seeking to deploy IoT solutions in real-world environments. These findings have profound implications for system designers, developers, and decision-makers, guiding their choices in platform selection, resource allocation, and scalability strategies.

Finally, the study emphasized the critical role of performance evaluation in ensuring the robustness and reliability of cloud-based IoT systems. By meticulously measuring and analyzing performance metrics, stakeholders can gain deeper insights into system behavior and identify opportunities for optimization and improvement. Effective resource management, scaling infrastructure, and proactive monitoring are essential for maintaining optimal performance and meeting user expectations in dynamic IoT environments. Therefore, the findings of this study not only contribute to academic knowledge but also have practical implications for industry professionals involved in cloud-enabled IoT deployments.

6.2 CONSTRAINTS AND PROSPECTS

While this study has made significant strides in understanding the performance evaluation of cloud-enabled IoT systems, it is imperative to acknowledge the inherent limitations that may affect the generalizability and applicability of the findings. One such limitation lies in the controlled nature of the experiments conducted within a laboratory setting. Real-world deployments often encounter diverse and dynamic conditions, including fluctuating network connectivity, environmental influences, and user behavior patterns. Consequently, future research endeavors should aim to validate the study's findings in more

varied and realistic deployment scenarios to ensure the robustness and relevance of the conclusions.

Another limitation pertains to the focus primarily on performance evaluation metrics, which, while crucial, do not provide a comprehensive view of the multifaceted challenges and considerations inherent in cloud-enabled IoT deployments. Beyond performance metrics such as response time and throughput, factors such as data privacy, security vulnerabilities, and total cost of ownership are equally vital for the successful implementation and operation of IoT systems. Thus, future research efforts should endeavor to conduct more comprehensive evaluations that encompass a broader range of factors to provide a holistic understanding of the complexities involved.

Moreover, while the scalability of cloud-based IoT systems was explored to some extent in this study, scalability remains a multifaceted challenge that extends beyond mere throughput and response time. Scalability encompasses considerations such as data partitioning, load balancing, and auto-scaling mechanisms, all of which play pivotal roles in ensuring the system's ability to handle increasing workloads efficiently. Future research endeavors could delve deeper into these scalability implications, exploring novel architectural designs and optimization strategies to address the evolving demands of IoT deployments at scale.

Furthermore, the integration of emerging technologies such as 5G, edge computing, and blockchain presents both opportunities and challenges for cloud-enabled IoT deployments. While these technologies hold the promise of enhancing system performance, reliability, and security, their integration requires careful consideration and evaluation. Future research could focus on investigating the synergies between cloud computing, IoT, and emerging technologies, exploring innovative architectures, protocols, and algorithms that leverage the strengths of each to deliver more efficient and resilient IoT solutions.

Additionally, while the experiments conducted in this study provided valuable insights into the performance characteristics of cloud-based IoT systems, they were limited in scope and duration. Long-term monitoring and evaluation of system performance in real-world deployments could offer deeper insights into the system's behavior over time, including potential performance degradation, optimization opportunities, and evolving user requirements. Therefore, future research endeavors should aim to conduct more extensive field trials and longitudinal studies to validate and refine the findings of this study.

In terms of future work, there is an urgent need to evaluate the security of different CSPs. Cloud computing has become a critical component of modern computing, and it is increasingly being used to store and process sensitive data, such as personal health information, financial data, and intellectual property. As a result, security has become a major concern for organizations that use cloud services. Future studies could examine the security features of different cloud service providers to determine their strengths and weaknesses.

The impact of regulatory compliance and data governance on cloud-enabled IoT deployments cannot be overstated. Data privacy regulations such as the General Data Protection Regulation (GDPR) and the California Consumer Privacy Act (CCPA) impose strict requirements on the collection, storage, and processing of personal data, posing significant challenges for IoT deployments. Future research could explore strategies and techniques for ensuring compliance with these regulations while maximizing the utility and value of IoT data for stakeholders.

Moreover, the environmental sustainability of cloud-enabled IoT deployments is an increasingly pressing concern in light of growing energy consumption and carbon emissions associated with data centers and IoT devices. Future research endeavors could focus on developing energy-efficient architectures, resource management strategies, and renewable energy integration techniques to minimize the environmental footprint of cloud-based IoT deployments while maintaining performance and reliability.

Furthermore, the evolving threat landscape and cybersecurity challenges facing cloud-enabled IoT systems necessitate ongoing research and innovation in security mechanisms and protocols. As IoT deployments become increasingly interconnected and pervasive, they become more susceptible to cyber attacks and malicious activities. Future research could explore advanced encryption techniques, anomaly detection algorithms, and secure communication protocols to mitigate security risks and safeguard IoT deployments against potential threats.

Additionally, the economic implications of cloud-enabled IoT deployments warrant further investigation, particularly regarding cost-effectiveness, return on investment (ROI), and business models. While cloud computing offers scalability and flexibility, it also entails recurring operational costs that may vary depending on usage patterns, service-level agreements (SLAs), and pricing models. Future research could explore optimization strategies for minimizing costs, maximizing ROI, and ensuring the long-term sustainability of cloud-based IoT deployments.

Finally, interdisciplinary collaboration and knowledge exchange are essential for advancing research in cloud enabled IoT systems. By fostering collaboration between researchers, practitioners, policymakers, and industry stakeholders, future research endeavors can leverage diverse expertise and perspectives to address complex challenges and drive innovation in the field. Initiatives such as collaborative research projects, industry-academic partnerships, and knowledge-sharing platforms can facilitate the exchange of ideas, insights, and best practices, ultimately leading to the development of more robust, resilient, and sustainable IoT solutions.

CHAPTER 7

CONCLUSION

7.1 SUMMARY OF THE STUDY

This study embarked on a comprehensive exploration of the performance evaluation of cloud computing and Internet of Things (IoT) systems, with a particular emphasis on three major cloud platforms: Amazon Web Services (AWS), Google Cloud Platform (GCP), and Microsoft Azure. The primary objective was to conduct a systematic analysis of various performance metrics under diverse operational scenarios, encompassing normal operation, heavy load conditions, and variations in IoT applications. Through a series of meticulously designed experiments, the study sought to illuminate the efficacy and reliability of cloud-enabled IoT systems, providing valuable insights into their operational characteristics and performance dynamics.

The study commenced by elucidating the fundamental principles and paradigms underlying cloud computing and the Internet of Things, delineating their symbiotic relationship and transformative potential across myriad industries. By establishing a conceptual framework for understanding the integration of cloud and IoT technologies, the study laid the groundwork for the subsequent performance evaluation.

Experimental testing was conducted across multiple dimensions to assess the performance of cloud-based IoT systems comprehensively. Under normal operating conditions, the study examined the response time, throughput, and latency of each cloud platform, providing quantitative metrics to gauge their efficiency and responsiveness. Subsequent experiments under heavy load conditions scrutinized the robustness and scalability of the systems, offering insights into their ability to withstand increased stress and maintain performance under duress. Variations in IoT application scenarios, including low-latency and high-throughput applications, further elucidated the adaptability and versatility of cloud-enabled IoT systems in diverse contexts.

Table 4. Score of functionalities

	aws	Google Cloud	Azure
Response Time	★★★	★★★★★	★★★★

Latency	★★★	★★★★	★★
Reliability	★★★★	★★★★★	★★★
Throughput	★★★★	★★★★★	★★

The findings of the study underscored the importance of performance evaluation in guiding decision-making processes for system designers and administrators. By elucidating the strengths and weaknesses of different cloud platforms and operational scenarios, the study empowered stakeholders to make informed choices regarding platform selection, optimization strategies, and resource allocation. Furthermore, the study highlighted the significance of ongoing research and development efforts to enhance the efficiency, reliability, and scalability of cloud-based IoT systems, ensuring their continued relevance and effectiveness in an ever-evolving technological landscape.

Based on the defined ranges for each metric, Amazon Web Services (AWS) achieves 3 stars for response time with an average of 157 ms, falling within the 141-160 ms range. It earns 4 stars for throughput at 231 req/sec, within the 230-244 req/sec range. For latency, AWS scores 3 stars at 5.4 ms, in the 5.1-5.5 ms range, and for reliability, it earns 4 stars at 99.96%, in the 99.95-99.97% range. Google Cloud Platform (GCP) excels with 5 stars for response time at 120 ms, within the ≤120 ms range. It also scores 5 stars for throughput at 245 req/sec, within the ≥245 req/sec range, and for reliability at 99.98%, in the ≥99.98% range. For latency, GCP earns 4 stars at 4.7 ms, in the 4.6-5.0 ms range. Microsoft Azure scores 4 stars for response time at 139 ms, within the 121-140 ms range. It earns 2 stars for throughput at 206 req/sec, in the 200-214 req/sec range, and 2 stars for latency at 5.8 ms, in the 5.6-6.0 ms range. For reliability, Azure scores 3 stars at 99.92%, within the 99.90-99.94% range. These ratings provide a clear view of each provider's performance in key areas, highlighting GCP as the top performer overall.

The choice between platforms ultimately depends on specific needs and priorities. If lightning-fast response times and top-notch processing power are paramount, GCP emerges as the frontrunner. However, if cost-effectiveness or a slightly more conservative approach is preferred, AWS might be a suitable alternative. If high reliability is the biggest concern, both GCP and AWS offer exceptional uptimes, while Azure might require further evaluation depending on the application's criticality.

In summary, this study constitutes a significant contribution to the field of cloud computing and IoT integration, providing valuable insights into the performance characteristics and operational dynamics of cloud enabled IoT systems. By bridging the gap

between theory and practice, the study advances our understanding of the intricacies involved in leveraging cloud and IoT technologies synergistically, paving the way for the development of innovative and efficient IoT solutions with tangible societal impact and economic benefits.

To evaluate cost-effectiveness for cloud services like AWS, GCP, and Azure, it's essential to consider several factors beyond just the basic instance costs. Each platform has unique strengths that can affect overall cost-effectiveness based on your specific needs:

Amazon Web Services (AWS):

Strengths: Extensive global reach, comprehensive feature set, robust security.

Cost: AWS offers a range of instance types, including cost-effective options like t2.micro for low-usage applications. Pricing competitiveness varies by region, with Mumbai being noted for cost-effectiveness.

Performance: Known for strong performance and reliability due to its broad service portfolio and infrastructure.

Google Cloud Platform (GCP):

Strengths: Competitive pricing, sustained use discounts, strong support for machine learning.

Cost: GCP provides flexibility with instances like e2 custom instances, allowing tailored resource allocation for specific workloads, enhancing cost-effectiveness.

Performance: Efficient and scalable infrastructure with strong performance benchmarks, suitable for advanced analytics and scalable applications.

Microsoft Azure:

Strengths: Deep integration with Microsoft products, extensive enterprise services, hybrid cloud capabilities.

Cost: Azure offers budget-friendly options like Standard_B1s instances, particularly suited for testing and low-scale applications.

Performance: Solid performance, especially in enterprise environments, although latency can vary by region.

In summary, AWS is favored for its extensive services and global reach, making it suitable for integrated solutions across different regions. GCP stands out with flexible pricing options and strong analytics capabilities, appealing for businesses needing tailored solutions.

Azure excels in enterprise integration and hybrid cloud scenarios, leveraging its deep Microsoft ecosystem ties.

For precise cost comparisons, it's crucial to consult the respective cloud providers' pricing pages to match instance types and regions relevant to your deployment needs. This detailed assessment ensures you choose the most cost-effective option aligned with your application requirements and budget considerations.

7.2 CONTRIBUTION AND SIGNIFICANCE OF THE STUDY

The study offers several notable contributions to the field of cloud computing and IoT integration:

Comprehensive Performance Evaluation: By systematically assessing key performance metrics across multiple cloud platforms, the study provides a holistic view of the performance of cloud enabled IoT systems. This analysis aids in understanding the strengths and weaknesses of different platforms and informs decision-making processes for system designers and administrators.

Identification of Performance Factors: Through rigorous experimentation, the study identifies critical performance factors such as response time, throughput, latency, and reliability. Understanding these factors is crucial for optimizing the design and deployment of cloud based IoT applications and ensuring their effectiveness in real-world scenarios.

Practical Implications for Industry: The findings of this study have practical implications for industry professionals involved in the development and management of cloud enabled IoT solutions. By highlighting best practices and potential areas for improvement, the study aims to enhance the performance and scalability of cloud based IoT systems, ultimately leading to more efficient and reliable deployments.

Guidance for Future Research: The study outlines directions for future research, including the exploration of advanced optimization techniques, integration with emerging technologies such as 5G and blockchain, and the development of novel applications for cloud enabled IoT systems. These avenues for further investigation hold the potential to advance the state-of-the-art in cloud computing and IoT integration and address evolving challenges in the field.

7.3 RECOMMENDATIONS FOR FUTURE RESEARCH

Building upon the findings of this study, several recommendations for future research emerge:

Optimization of Performance Metrics: Future research could focus on developing advanced optimization techniques to further improve performance metrics such as response time, throughput, and latency in cloud enabled IoT systems. This could involve leveraging machine learning algorithms or advanced networking protocols to enhance system efficiency.

Exploration of Emerging Technologies: With the rapid evolution of technologies such as 5G and blockchain, future research could explore their integration with cloud computing and IoT to enable new applications and enhance system capabilities. Investigating the impact of these technologies on performance and scalability is essential for staying at the forefront of innovation.

Security and Privacy Considerations: As security and privacy concerns continue to be paramount in IoT deployments, future research could focus on developing robust security mechanisms and privacy-preserving techniques for cloud-enabled IoT systems. This includes encryption methods, access control mechanisms, and secure data sharing protocols.

Scalability and Resource Management: With the exponential growth of IoT devices and data volumes, scalability and resource management become critical challenges. Future research could explore novel approaches for scaling cloud based IoT systems and efficiently managing resources to meet increasing demands while maintaining optimal performance.

Implementation with an Application on Raspberry Pi**:** Future work could include implementing the system with an application on a Raspberry Pi. This would allow for the development of a cost-effective and portable solution, making it easier to deploy and test in various real-world environments. The Raspberry Pi's flexibility and extensive community support can facilitate rapid prototyping and iterative improvements of IoT applications.

By addressing these research areas, future studies can build upon the foundation laid by this research and contribute to advancing the field of cloud computing and IoT integration, ultimately driving innovation and facilitating the development of more efficient and reliable IoT solutions.

REFERENCES

[1] M. Ansari, S. Arshad Ali, and M. Alam, “Internet of things (IoT) fusion with cloud computing: current research and future direction”, International Journal of Advanced Technology and Engineering Exploration, 2022, pp. 1812–1845.

[2] N. Kashyap, A. Rana, and V. Kansal, Himdweep Walia, “Improve Cloud Based IoT Architecture Layer Security - A Literature Review”, 2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), Greater Noida, India, 2021, pp. 112–115.

[3] S. Shahzadi, M. Iqbal, Z. Qayyum, and T. Dagiuklas, “Infrastructure as a Service (IaaS): A Comparative Performance Analysis of Open-Source Cloud Platforms”, 2017 IEEE 22nd International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD), Lund, Sweden, 2017, pp. 271–350.

[4] M. Humayun, “Role of Emerging IoT Big Data and Cloud Computing for Real Time Application”, International Journal of Advanced Computer Science and Applications (IJACSA), 11(4), 2020, pp. 494–506.

[5] H. Khazaei, J. Misic, V. Misic, “Modelling of Cloud Computing Centers Using M/G/m Queues”, 2011 31st International Conference on Distributed Computing Systems Workshops, Minneapolis, MN, 2011.

[6] Z. Ma, Y. Liu, X. Liu, J. Ma, and F. Li, “Privacy-Preserving Outsourced Speech Recognition for Smart IoT Devices”, IEEE Internet of Things Journal, 6(5), 2019.

[7] H. Li, X. Li, H. Wang, J. Zhang, and Z. Jiang, “Research on Cloud Performance Testing Model”, 2019 IEEE 19th International Symposium on High Assurance Systems Engineering (HASE), pp. 179-183, Hangzhou, China, 2019, DOI: 10.1109/HASE.2019.00035

[9] M. Eisa Suliman, “A Brief Analysis of Cloud Computing Infrastructure as a Service (IaaS)”, International Journal of Innovative Science and Research Technology, pp. 1325-1333, 2021, ISSN No:-2456-2165

APPENDIX

PUBLISHED RESEARCH ARTICLES

[1] J. Sithiyopasakul, T. Archevapanich, S. Sithiyopasakul, A. Lasakul, B. Purahong and C. Benjangkaprasert, " Implementation of Cloud Computing and Internet of Things (IoT) by Performance Evaluation," 2024 International Electrical Engineering Congress (iEECON), 6-8 March, 2024, pp. 650-655.

AUTHOR BIOGRAPHY

Name-Surname	Jiran Sithiyopasakul
Date of birth	4 July 2001
Address	130/144 Ramkhamhaeng 43/1 Plubpla Wangthonglang Bangkok 10310 Tel. 097-234-0683
Education History	2022 School of Engineering, Information Engineering, King Mongkut's Institute of Technology Ladkrabang