People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research
Dr. Yahia FARES University of Medea



Faculty of Sciences
Department of Mathematics and Computer Science

Master's Thesis

Field: Computer Science

Specialization: Systems Engineering and Web Technology

Title

Design and Development of a Predictive Digital Twin for Proactive Cloud Resource Management: Application to E-Government Services

Presented by:

- BELHADJ MOHAMED YOUNES

- BOUCHENAFA AKREM BELKHACEM

Supervised by:

- Dr. TOUBAL El Bahi

- Dr. Rahim Messaoud

Academic Year: 2025-2026

Acknowledgements

First and foremost, we express our deepest gratitude to Almighty Allah for granting us the strength, patience, and perseverance required to accomplish this work.

We would like to extend our sincere thanks to our supervisors, **Dr. TOUBAL El Bahi** and **Dr. Rahim Messaoud**, for their invaluable guidance, constructive feedback, and continuous support throughout the realization of this thesis. Their expertise and encouragement played a fundamental role in the completion of this work.

We are also grateful to all our professors and teachers whose knowledge, dedication, and guidance have greatly contributed to our academic formation and personal development. Our sincere appreciation is extended to the members of the jury for accepting to evaluate this work and for the time and effort devoted to its assessment.

Finally, we would like to thank our families, friends, and everyone who supported us, directly or indirectly, throughout this academic journey. Their encouragement and confidence in us have been a constant source of motivation.

Mohamed Younes,

Akram Bouchenafa

Dedication

In the name of Allah, the Most Gracious, the Most Merciful.

I dedicate this work first to my beloved parents, whose unconditional love, sacrifices, and unwavering support have been the foundation of every achievement in my life. Their trust in me has always been my greatest source of motivation.

To my brothers and sister, whose encouragement, understanding, and presence have accompanied me throughout this journey.

To my family, for their continuous support and belief in my abilities.

To my friends, who shared both the challenges and the memorable moments of this academic experience.

To all my teachers and professors, whose guidance and knowledge have contributed to my personal and academic growth.

Finally, I dedicate this work to everyone who strives for knowledge, perseverance, and self-improvement.

Mohamed Younes

Dedication

Above all, I thank Almighty God for giving me the strength, patience, and guidance to complete this modest work.

I dedicate my contribution to this work:

To **my dear parents**, for their endless love, sacrifices, encouragement, and unwavering support throughout my life and studies.

To **my beloved brothers**, **Ali** and **Salim**, for their support, motivation, and constant presence by my side.

To **my dear little sister**, for the joy, affection, and happiness she brings to our family.

To all my family members who have always encouraged and supported me.

To my friends **Bobaker** and **Khaled**, and to **all my friends** near and far, for their friendship, encouragement, and memorable moments.

To everyone who contributed, directly or indirectly, to the realization of this work.

To **my teachers and professors**, whose knowledge, guidance, and dedication have greatly contributed to my academic journey.

Akrem Bouchenafa

Abstract

Digital Twin technology enables the creation of virtual representations of physical systems for monitoring and management purposes. When combined with Artificial Intelligence, Digital Twins can provide predictive capabilities that support proactive decision-making.

This thesis proposes a Predictive Digital Twin framework for proactive cloud resource management in e-government services. The framework integrates a Digital Twin platform, predictive analytics, and explainable artificial intelligence to predict future scaling decisions based on historical telemetry data. Several machine learning and deep learning models, including Random Forest, XGBoost, BiLSTM, and Transformer, were evaluated using the Alibaba Cluster Trace 2018 dataset.

The experimental results demonstrate the feasibility of predicting scaling actions (scale_up, maintain, and scale_down) from cloud resource metrics. The proposed framework extends traditional Digital Twin monitoring by incorporating predictive intelligence and decision-support capabilities for cloud administrators.

Keywords : Digital Twin, Artificial Intelligence, Cloud Computing, E-Government Services, Predictive Analytics, Resource Management.

Résumé

La technologie du Jumeau Numérique permet de créer une représentation virtuelle d'un système physique afin d'en assurer la surveillance et la gestion. Associée à l'Intelligence Artificielle, elle peut fournir des capacités prédictives favorisant une gestion proactive des infrastructures.

Ce mémoire propose un cadre de Jumeau Numérique Prédictif pour la gestion proactive des ressources cloud dans les services d'e-gouvernement. La solution intègre une plateforme de Jumeau Numérique, des modèles d'analyse prédictive et des techniques d'intelligence artificielle explicable afin de prédire les futures décisions de mise à l'échelle à partir de données historiques de télémétrie. Plusieurs modèles d'apprentissage automatique et profond, notamment Random Forest, XGBoost, BiLSTM et Transformer, ont été évalués à l'aide du jeu de données Alibaba Cluster Trace 2018.

Les résultats obtenus démontrent la faisabilité de la prédiction des actions de mise à l'échelle (scale_up, maintain et scale_down) à partir des métriques des ressources cloud. Le cadre proposé enrichit le Jumeau Numérique traditionnel en y intégrant des capacités prédictives et d'aide à la décision.

Mots-clés : Jumeau Numérique, Intelligence Artificielle, Cloud Computing, Services d'E-Gouvernement, Analyse Prédictive, Gestion des Ressources.

الملخص

تتيح تقنية التوأم الرقمي إنشاء تمثيل افتراضي للأنظمة الفيزيائية بهدف مراقبتها وإدارتها. وعند دمجها مع تقنيات الذكاء الاصطناعي، تصبح قادرة على توفير قدرات تنبؤية تدعم اتخاذ القرارات بشكل استباقي.

يقدم هذا العمل إطارًا لتوأم رقمي تنبؤي مخصص للإدارة الاستباقية للموارد السحابية في خدمات الحكومة الإلكترونية. يعتمد الإطار المقترح على دمج منصة التوأم الرقمي مع تقنيات التحليل التنبؤي والذكاء الاصطناعي القابل للتفسير من أجل التنبؤ بقرارات التوسع المستقبلية اعتمادًا على بيانات التليمترية التاريخية. وقد تم تقييم عدة نماذج للتعلم الآلي والتعلم العميق, من بينها Random Forest و XGBoostو BiLSTMوTransformer, باستخدام مجموعة بيانات Alibaba Cluster Trace 2018.

أظهرت النتائج إمكانية التنبؤ بقرارات التوسع (Scale Up و Maintainو Scale Down) انطلاقًا من مؤشرات الموارد السحابية. ويساهم الإطار المقترح في تطوير التوأم الرقمي التقليدي من أداة مراقبة إلى نظام تنبؤي داعم لاتخاذ القرار.

الكلمات المفتاحية :التوأم الرقمي, الذكاء الاصطناعي, الحوسبة السحابية, خدمات الحكومة الإلكترونية, التحليل التنبؤي, إدارة الموارد.

Table of Cotents

Acknowledgements 2

Dedication 3

Dedication 4

Abstract 5

General Introduction 13

CHAPTER I : DIGITAL TWIN AND ARTIFICIAL INTELLIGENCE 15

I. INTRODUCTION 16

II. DIGITAL TWIN TECHNOLOGY 16

1. **Definition and Concepts** 16

2. **Components of a Digital Twin** 17

3. **Types of Digital Twins** 20

4. **Digital Twin Maturity Models** 21

5. **Application Areas of Digital Twins** 21

III. ARTIFICIAL INTELLIGENCE FUNDAMENTALS 22

1. **Machine learning Methods** 22

2. **Deep Learning** 22

IV. INTEGRATION OF AI WITH DIGITAL TWINS 23

1. **From Descriptive to Predictive Digital Twins** 23

2. **Predictive Maintenance of Cloud Resources** 23

3. **Prescriptive Analytics and Decision Support** 23

V. TECHNOLOGIES AND TOOLS 23

1. **Eclipse Ditto (IoT Middleware)** 23

2. **Node-RED (Integration Layer)** 24

3. **Artificial Intelligence Development Stack** 24

4. **Deployment Infrastructure (Docker)** 24

VI. Conclusion 25

CHAPTER II: CLOUD COMPUTING AND E-GOVERNMENT SERVICES 26

I. Introduction 27

II. Cloud Computing Fundamentals 27

1. Definition and Core Characteristics 27

2. Cloud Service Models 28

3. Cloud Deployment Models 29

III. Cloud Resource Metrics 29

1. CPU Utilisation 29

2. Memory Consumption 30

3. Network Traffic and Latency 30

4. Disk I/O and Storage Utilisation 30

5. Service Availability and Response Time 30

IV. E-Government Services: Requirements and Infrastructure Challenges 31

1. Definition and Scope of E-Government 31

2. Infrastructure Requirements for E-Government Services 32

3. Infrastructure Challenges 32

V. Case Studies: E-Government Infrastructure in Practice 33

1. Algeria: Digital Transformation of Public Administration 33

2. Finland: A Model for Advanced E-Government Infrastructure 34

3. Comparative Analysis 34

VI. Current Monitoring Approaches and Their Limitations 35

1. Threshold-Based Monitoring 35

2. Time-Series Monitoring Tools 35

3. The Gap: From Reactive to Predictive Supervision 35

VII. Conclusion 36

CHAPTER III: CONCEPTION AND MODELING 37

I. Introduction 38

II. Requirements Analysis 38
III. Proposed Predictive Digital Twin Architecture 39
IV. Digital Twin Design 40
1. Global System Functionalities 40
2. Digital Twin Conceptual Model 41
3. Digital Twin Synchronization Mechanism 42
4. Predictive Intelligence Extension 43
5. Administrative Supervision 45
V. Conclusion 46
CHAPTER IV: IMPLEMENTATION AND EVALUATION 47
I. Introduction 48
II. Development Environment 48
III. Dataset Preparation 49
IV. Feature Engineering and Label Generation 50
V. Predictive Models Implementation 52
VI. Predictive Digital Twin Implementation 53
VII. Experimental Setup 55
VIII. Results and Evaluation 55
IX. Explainability Analysis 57
X. Ablation Study 58
XI. Additional Experiment: Period-Based Prediction 59
XII. Comparison with Related Works 60
XIII. Limitations and Future Work 61
XIV. Conclusion 62
General Conclusion 63
BIBLIOGRAPHY 65

LIST OF FIGURES

Figure 1.1 Concept of a Digital Twin showing the interaction between the physical entity and its virtual representation **14**
Figure 1.2 Main components of a Digital Twin **15**
Figure 2.1 Cloud Service Models **26**
Figure 3.1 Global Architecture of the Proposed Predictive Digital Twin System **37**
Figure 3.2 Global System Use Case Diagram **39**
Figure 3.3 Class Diagram of the Predictive Digital Twin Framework **40**
Figure 3.4 Global System Sequence Diagram **40**
Figure 3.5 AI Prediction Service Sequence Diagram **41**
Figure 3.6 Administrative Dashboard Use Case Diagram **41**
Figure 4.1 Dataset Preparation Workflow **46**
Figure 4.2 Label Generation Process **48**
Figure 4.3 Sliding Window Strategy **48**
Figure 4.4 Prediction Pipeline **49**
Figure 4.5 Implementation Architecture and Data Flow **50**
Figure 4.6 Class Distribution **52**
Figure 4.7 Model Performance Comparison **54**
Figure 4.8 SHAP Feature Importance Analysis **55**
Figure 4.9 Ablation Study Results **55**
Figure 4.10 Period-Based Prediction Confusion Matrices **56**

LIST OF TABLES

Table 1.1 Digital Twin Components and Functions 17

Table 2.1 Comparison of Cloud Service Models ..... ……26

Table 2.2 Key Cloud Resource Metrics and Critical Thresholds ……..... 29

Table 2.3 Comparative Analysis of E-Government Infrastructure:Algeria vs Finland……..... 32

Table 4.1 Software Technologies and Development Environment 45

Table 4.2 Dataset Characteristics 46

Table 4.3 Experimental Parameters 51

Table 4.4 Performance Comparison of Predictive Models 53

Table 4.5 Ensemble Model Results 54

Table 4.6 Comparison with Related Works 57

LIST OF ACRONYMS

AI Artificial Intelligence
API Application Programming Interface
AWS Amazon Web Services
BiLSTM Bidirectional Long Short-Term Memory
CPU Central Processing Unit
DTA Digital Twin Aggregate
DTI Digital Twin Instance
DTP Digital Twin Prototype
DT Digital Twin
EC2 Elastic Compute Cloud
G2B Government-to-Business
G2C Government-to-Citizen
G2G Government-to-Government
HTTP HyperText Transfer Protocol
IaaS Infrastructure as a Service
IoT Internet of Things
JSON JavaScript Object Notation
MAE Mean Absolute Error
ML Machine Learning
MQTT Message Queuing Telemetry Transport
PaaS Platform as a Service
RAM Random Access Memory
REST Representational State Transfer
RF Random Forest
RMSE Root Mean Square Error
SaaS Software as a Service
SHAP SHapley Additive exPlanations
SLA Service Level Agreement
UI User Interface
VM Virtual Machine
XGBoost Extreme Gradient Boosting

General Introduction

The rapid expansion of cloud computing has transformed the way digital services are delivered and managed. In particular, e-government platforms increasingly rely on cloud infrastructures to provide scalable, reliable, and continuously available services to citizens. As the number of users and digital services grows, managing cloud resources efficiently becomes a critical challenge. Insufficient resources may lead to service degradation, while excessive resource allocation increases operational costs.

Traditional monitoring solutions primarily focus on observing the current state of cloud infrastructures through metrics such as CPU utilization, memory consumption, and disk activity. Although these approaches provide valuable information about system status, they remain largely reactive and offer limited support for anticipating future workload variations. Consequently, administrators often respond to performance issues only after they occur.

Digital Twin technology has emerged as a promising solution for creating virtual representations of physical systems and enabling continuous monitoring and synchronization. When combined with Artificial Intelligence techniques, Digital Twins can evolve from descriptive monitoring platforms into predictive decision-support systems capable of anticipating future system behavior and assisting operational management.

Despite the growing interest in Digital Twin technology and cloud resource prediction, limited research has explored the integration of Digital Twins, machine learning models, and explainable artificial intelligence for proactive cloud resource management. Furthermore, many existing studies focus on forecasting resource utilization values rather than directly supporting operational scaling decisions.

The main objective of this work is therefore to design and implement a Predictive Digital Twin framework capable of anticipating future cloud resource requirements and generating scaling recommendations for e-government infrastructures. To achieve this objective, telemetry data collected from cloud resources are processed and analyzed using machine learning and deep learning techniques in order to predict future scaling actions classified as scale_up, maintain, or scale_down.

The proposed framework combines Digital Twin technology, predictive analytics, and explainable artificial intelligence within a unified architecture. Historical telemetry data are transformed into supervised learning datasets using a sliding-window strategy, and multiple predictive models are evaluated, including Random Forest, XGBoost, BiLSTM, and

Transformer architectures. In addition, explainability techniques are applied to identify the most influential factors affecting prediction outcomes.

The principal contributions of this work can be summarized as follows:

- Design of a Predictive Digital Twin architecture for proactive cloud resource management.
- Development of an AI-based prediction service capable of generating future scaling recommendations.
- Comparative evaluation of machine learning and deep learning models using real cloud telemetry data.
- Integration of explainability techniques to improve the interpretability of prediction results.
- Development of an administrative dashboard supporting monitoring and predictive decision-making.

This thesis is organized into four chapters:

- **Chapter 1: Digital Twin and Artificial Intelligence**

  Presents the theoretical foundations of Digital Twin technology, Artificial Intelligence, and their integration for predictive supervision and decision support.

- **Chapter 2: Cloud Computing and E-Government Services**

  Introduces cloud computing concepts, key resource metrics, e-government infrastructures, and the challenges associated with monitoring and resource management.

- **Chapter 3: Conception and Modeling**

  Describes the design of the proposed Predictive Digital Twin framework, including the system architecture, Digital Twin design, and UML modeling.

- **Chapter 4: Implementation and Evaluation**

  Presents the implementation of the framework, dataset preparation, predictive model development, experimental evaluation, explainability analysis, and comparison with related works.

Finally, the General Conclusion summarizes the main contributions of this research, discusses its limitations, and outlines future research directions.

CHAPTER I : DIGITAL TWIN AND ARTIFICIAL INTELLIGENCE

I. INTRODUCTION

Digital Twin technology is an innovative approach that enables the creation of a virtual representation of a physical system for real-time monitoring and analysis. Initially adopted in industrial environments, it has gradually expanded to various domains, including healthcare, smart cities, and e-government services [2]. At the same time, advances in Artificial Intelligence, particularly in machine learning and deep learning, have significantly improved the ability to generate predictions from historical data [3]. The integration of AI into Digital Twin platforms enables the transition from reactive supervision to proactive supervision, allowing potential issues to be anticipated before they occur. This evolution is especially important for e-government platforms, which provide critical services to citizens and require high levels of reliability and availability [4]. Service disruptions can negatively impact service quality and reduce citizens' trust in digital public services [5]. Building upon an existing Digital Twin platform, this thesis enhances its capabilities by integrating predictive intelligence. Therefore, this chapter presents the theoretical foundations of Digital Twin technology, Artificial Intelligence, and their convergence for predictive management of digital infrastructures.

II. DIGITAL TWIN TECHNOLOGY

1. Definition and Concepts

A Digital Twin is a virtual representation of a physical object, process, or system that mirrors its real-world counterpart in real-time [1]. This virtual replica is continuously synchronized with the physical entity through data exchange, enabling monitoring, analysis, simulation, and optimization throughout the lifecycle of the physical asset [2].

The concept of the Digital Twin was first introduced by Dr. Michael Grieves at the University of Michigan in 2002, initially referred to as the "Mirrored Spaces Model" [3]. The term "Digital Twin" gained prominence through NASA's work on Apollo missions, where ground-based replicas of space vehicles were used for simulation and problem-solving [4]. Today, the concept has evolved significantly, driven by advances in the Internet of Things (IoT), cloud computing, and artificial intelligence.

According to the literature, a Digital Twin is characterized by three main components: the physical entity, the virtual representation, and the data connection that links them [5]. This bi-directional connection allows data to flow from the physical entity to the virtual twin for monitoring, and from the virtual twin to the physical entity for control and optimization. Several definitions of Digital Twins exist in the academic literature. Tao et al define a Digital Twin as

"a set of virtual information constructs that mimics the structure, context, and behavior of a physical twin." The authors emphasize that the Digital Twin is not merely a 3D model but a dynamic representation that evolves with its physical counterpart [1]. **Figure 1.1** illustrates the fundamental concept of a Digital Twin and the continuous interaction between the physical entity and its virtual representation through real-time data exchange.

The key characteristics of a Digital Twin include:

1. **Real-time synchronization:** The digital twin continuously receives data from its physical counterpart, ensuring that the virtual representation reflects the current state of the physical asset [6].
2. **Bidirectional data flow:** Data moves both from physical to virtual (for monitoring) and from virtual to physical (for control and optimization) [7].
3. **Lifecycle coverage:** Digital twins span the entire lifecycle of an asset, from design and manufacturing to operation and maintenance [8].
4. **Simulation capability:** Digital twins can simulate future states and scenarios, enabling what-if analysis and predictive capabilities [9].
5. **Data integration:** Digital twins integrate data from multiple sources, including sensors, historical records, and external systems [10].

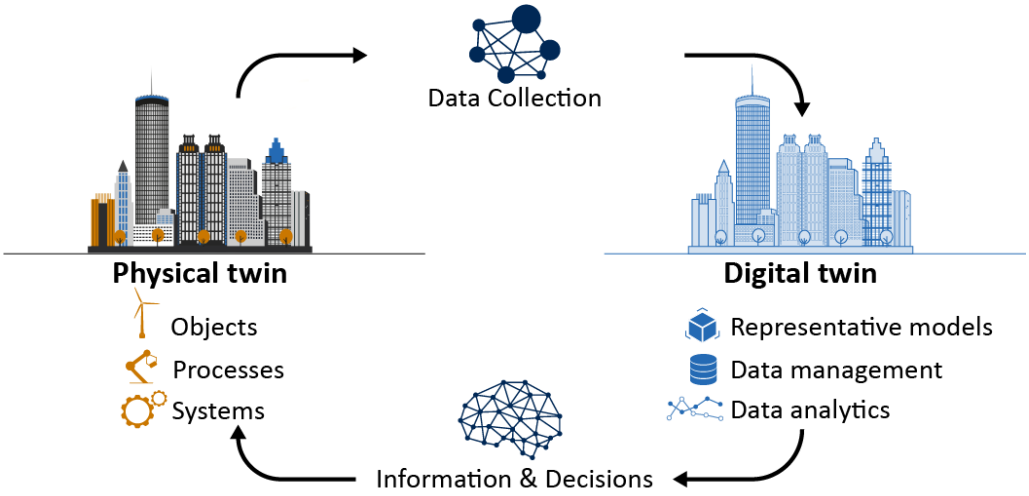


Figure 1.1: Concept of a Digital Twin showing the interaction between the physical entity and its virtual representation.

2. Components of a Digital Twin

A Digital Twin system consists of several interconnected components that work together to create and maintain the virtual representation of a physical entity [11]. These components can be categorized into three main layers: hardware, middleware, and software [12]. **Figure 1.2**

presents the main components of a Digital Twin system and illustrates the interaction between the hardware, middleware, and software layers that collectively enable real-time synchronization and intelligent system management.

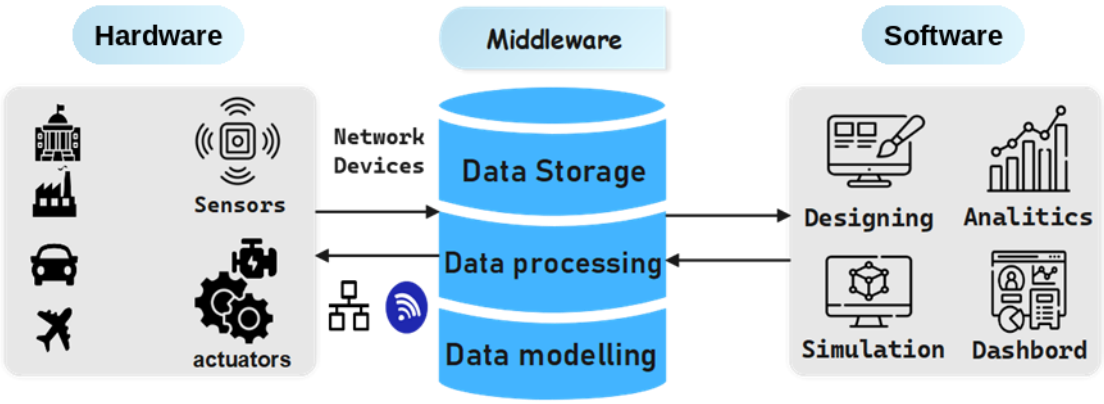


Figure 1.2 illustrates the main components of a Digital Twin.

2.1 Hardware Components

The hardware layer forms the physical infrastructure that supports data acquisition and communication [13]. Key hardware components include:

- **Sensors and actuators:** Sensors collect operational data from physical assets, while actuators enable control actions to be applied to the physical environment based on decisions generated by the Digital Twin platform.[14].
- **IoT devices:** Internet of Things devices serve as the interface between the physical world and the digital twin environment, facilitating data acquisition, processing, and transmission [15].
- **Edge computing devices:** These devices perform initial data processing close to the data source, reducing latency and bandwidth requirements [16].
- **Communication networks:** Robust network infrastructure enables reliable data transfer between physical assets and the digital twin platform [17].

2.2 Middleware Components

The middleware layer acts as a bridge between hardware and software, facilitating data integration, communication, and interoperability [18]. Middleware components include:

- **IoT platforms:** These platforms provide standardized interfaces for connecting heterogeneous devices and managing device lifecycles [19].

- **Message brokers:** Systems such as Apache Kafka and RabbitMQ handle real-time data streaming and event distribution between components [20].
- **Protocol converters:** These components translate data between different communication protocols (MQTT, AMQP, HTTP, WebSocket) to ensure interoperability [21].
- **API gateways:** RESTful APIs provide standardized endpoints for applications to interact with the digital twin [22].

2.3 Software Components

The software layer forms the core of the digital twin framework, encompassing applications, algorithms, and platforms for data processing, analytics, modeling, and visualization [23]. Key software components include:

- **Digital twin platforms:** Platforms such as Eclipse Ditto, Azure Digital Twins, and AWS IoT TwinMaker provide the core infrastructure for creating and managing digital twins [24].
- **Data processing engines:** These components handle data aggregation, filtering, and transformation before analysis [25].
- **Analytics engines:** Machine learning and statistical algorithms derive insights from collected data, enabling predictive and prescriptive capabilities [26].
- **Simulation engines**: These engines execute virtual models that replicate the behavior of physical systems, enabling what-if analysis and scenario testing [27].
- **Visualization tools:** Dashboards and user interfaces provide intuitive ways to monitor digital twin data and interact with the system [28].

To summarize the organization of a Digital Twin architecture, **Table 1.1** presents the three principal layers: hardware, middleware, and software along with their main components and functions. This layered architecture ensures effective communication between physical assets and their virtual representations while supporting real-time monitoring, data processing, analytics, and decision-making capabilities.

Layer	Components	Primary Functions
HARDWARE LAYER	Sensors (temperature, pressure, humidity, vibration, utilization); Actuators; IoT devices; Edge computing devices; Communication networks (WiFi, 5G, Ethernet)	Data collection from physical assets; Physical control; Initial data processing; Data transmission
MIDDLEWARE LAYER	IoT platforms; Message brokers (Kafka, RabbitMQ); Protocol converters (MQTT, AMQP, HTTP, WebSocket); API gateways; Data storage systems (MongoDB, InfluxDB)	Device management; Data streaming; Protocol conversion; Interoperability; Data persistence
SOFTWARE LAYER	Digital twin platforms (Eclipse Ditto, Azure Digital Twins); Data processing engines; Analytics engines (ML, AI); Simulation engines; Visualization tools (Dashboards, charts); Alert management systems	Twin creation and management; Data aggregation; Predictive analytics; What-if simulation; User interaction; Notification

Table 1.1 summarizes the three layers of Digital Twin components.

This hierarchical structure of hardware, middleware, and software components provides a comprehensive framework for building robust digital twin systems capable of real-time monitoring, analysis, and control of physical assets [29].

3. Types of Digital Twins

Digital Twins are classified based on their application hierarchy and their position within the product or system lifecycle. According to the foundational classification formalized

by Grieves and Vickers, three primary types exist: the Prototype, the Instance, and the Aggregate [30].

- **Digital Twin Prototype (DTP):** Describes the prototypical physical asset. It contains the informational sets necessary to describe and produce a physical version that duplicates the virtual version. In this thesis, the DTP includes the architectural designs and initial AI model parameters before deployment.
- **Digital Twin Instance (DTI):** Linked to a specific, unique physical entity throughout its entire lifecycle. The DTI is continuously updated with real-time data from its specific counterpart (e.g., a specific e-government server) to reflect its current state and performance metrics.
- **Digital Twin Aggregate (DTA):** A computing construct that aggregates data from multiple DTIs. It allows for querying and identifying patterns across an entire fleet of services to correlate previous state changes with subsequent behavioral outcomes.

4. Digital Twin Maturity Models

To assess the sophistication of a system, it must be positioned within a maturity framework. The model proposed by Yung Woon Kim [31] identifies five distinct stages of progression:

1. **Look-alike:** A one-time rendering or visual model lacking real-time integration.
2. **Static:** Incorporates historical data but lacks continuous synchronization.
3. **Dynamic:** Features real-time sensor data for active monitoring—the current state of many existing platforms.
4. **Interactive:** Uses machine learning and simulation to forecast future states and suggest decisions. This is the primary goal of this thesis research.
5. **Autonomous:** The highest level, where the system is fully self-aware and can execute optimization actions without human intervention.

5. Application Areas of Digital Twins

Digital Twin technology has transcended its origins in manufacturing and aerospace to impact a variety of complex industries. In manufacturing, it optimizes production lines and enables predictive maintenance. In healthcare, it facilitates personalized patient monitoring and surgical planning. For smart cities, DTs are used for urban planning and traffic management. More recently, the application of DTs in e-government services has emerged as a critical tool for improving the efficiency and reliability of public administration [32].

III. ARTIFICIAL INTELLIGENCE FUNDAMENTALS

Artificial Intelligence (AI) refers to the ability of computer systems to perform tasks that normally require human intelligence, such as learning, reasoning, pattern recognition, and decision-making. In recent years, AI has become a key technology for analyzing large volumes of data and supporting intelligent decision-making processes. Within Digital Twin environments, AI enables predictive analytics, anomaly detection, and optimization capabilities that extend beyond traditional monitoring functionalities [33].

1. Machine learning Methods

Machine Learning (ML) is a branch of Artificial Intelligence that allows systems to learn patterns from historical data and make predictions without being explicitly programmed. By analyzing previous observations, machine learning algorithms can identify relationships between variables and generate predictions for unseen situations. In the context of cloud resource management, machine learning techniques can be used to predict future system behavior and support proactive resource allocation decisions [33].

Among the most widely used machine learning algorithms are decision-tree-based methods such as Random Forest and XGBoost. These models are particularly effective for structured tabular data and have demonstrated strong performance in predictive analytics applications.

2. Deep Learning

Deep Learning is a subset of Machine Learning based on artificial neural networks composed of multiple layers. These models are capable of automatically learning complex patterns from large datasets and have achieved remarkable results in fields such as computer vision, natural language processing, and time-series analysis.

For sequential and temporal data, architectures such as Bidirectional Long Short-Term Memory (BiLSTM) networks and Transformers have become widely adopted [35]. BiLSTM networks capture temporal dependencies in both forward and backward directions, while Transformer architectures leverage attention mechanisms to model long-range relationships within data sequences. These capabilities make them suitable for predictive analytics tasks involving cloud telemetry data.

IV. INTEGRATION OF AI WITH DIGITAL TWINS

1. From Descriptive to Predictive Digital Twins

Traditional Digital Twins primarily focus on monitoring, visualization, and real-time synchronization between physical and virtual systems. While these capabilities provide valuable operational awareness, they remain largely descriptive and reactive. Recent advances in Artificial Intelligence have enabled the development of Predictive Digital Twins capable of anticipating future system behavior through data-driven analytics. By integrating machine learning models into the Digital Twin environment, organizations can move from reactive supervision toward proactive decision-making and resource management. This evolution transforms the Digital Twin from a monitoring platform into a predictive decision-support system. [39].

2. Predictive Maintenance of Cloud Resources

In cloud-based infrastructures, the predictive capability of a digital twin is primarily applied through predictive maintenance strategies. Unlike reactive maintenance, which occurs after a failure, or preventive maintenance, which is scheduled based on time, predictive maintenance uses trend analysis to initiate actions only when resource exhaustion is imminent. By analyzing historical metrics, the AI layer detects patterns of CPU and memory degradation, providing a "forecast horizon" that allows for "just-in-time" resource scaling [40].

3. Prescriptive Analytics and Decision Support

Prescriptive analytics represents the pinnacle of the digital twin maturity model, addressing the critical question: "What should we do about the predicted outcome?". While predictive models forecast a CPU spike, the prescriptive engine within the digital twin analyzes these forecasts against operational constraints to recommend specific actions, such as scaling up specific server instances or migrating workloads. This transformation turns the digital twin into a decision-support system that minimizes human intervention and ensures Service Level Agreement (SLA) compliance [42].

V. TECHNOLOGIES AND TOOLS

1. Eclipse Ditto (IoT Middleware)

Eclipse Ditto is an open-source framework specifically designed to manage the state of digital twins by abstracting physical devices into "Things" with associated "Features". It provides a unified resource-based API that allows the AI layer to synchronize with the twin's state in real-time. In this thesis, Ditto acts as the central state repository, differentiating between

the "reported" state (current metrics from e-government servers) and the "desired" state (resource allocation targets) [49].

2. Node-RED (Integration Layer)

Node-RED is a flow-based development tool for visual programming, used primarily for wiring together hardware devices, APIs, and online services [50]. In the proposed architecture, Node-RED serves as the "junction layer." It handles the ingestion of real-time metrics from e-government service APIs, transforms the data into valid JSON payloads, and pushes updates to the Eclipse Ditto platform via MQTT or HTTP protocols.

3. Artificial Intelligence Development Stack

The predictive component of the proposed framework is developed using Python due to its extensive ecosystem for data science and artificial intelligence. Several libraries are employed throughout the implementation process:

- **Scikit-Learn:** Used for the development and evaluation of machine learning models, particularly Random Forest.
- **XGBoost:** Used for gradient boosting-based predictive modeling.
- **TensorFlow:** Used to implement and train deep learning models, including the BiLSTM architecture.
- **PyTorch:** Used for the implementation of the Transformer model.
- **Pandas and NumPy:** Used for data preprocessing, feature engineering, and numerical computations.
- **SHAP:** Used to provide explainability capabilities by identifying the most influential features affecting prediction outcomes.

4. Deployment Infrastructure (Docker)

The entire system stack—including Eclipse Ditto, MongoDB, Node-RED, and the AI API is containerized using Docker and orchestrated via Docker Compose [52]. This containerization ensures environmental consistency across the development and validation phases, allowing for scalable deployment of the predictive twin across different server environments.

VI. Conclusion

This chapter presented the theoretical foundations of Digital Twin technology and Artificial Intelligence that support the development of the proposed framework. The concepts of Digital Twins, their components, types, maturity levels, and application domains were first introduced. Subsequently, the main Artificial Intelligence concepts relevant to this work were discussed, including machine learning, deep learning, and explainable artificial intelligence.

The chapter also highlighted the evolution from traditional monitoring-oriented Digital Twins toward Predictive Digital Twins capable of anticipating future system behavior and supporting proactive decision-making. This convergence between Digital Twin technology and Artificial Intelligence constitutes the foundation of the proposed Predictive Digital Twin framework for cloud resource management.

Finally, the principal technologies supporting the implementation of the framework were presented, including Eclipse Ditto, Node-RED, Python-based AI frameworks, and Docker. The next chapter introduces cloud computing concepts and discusses the challenges associated with e-government infrastructures and cloud resource management.

CHAPTER II: CLOUD COMPUTING AND E-GOVERNMENT SERVICES

I. Introduction

Cloud computing has fundamentally transformed the way organisations design, deploy, and manage digital services. By offering on-demand access to scalable computing resources over the internet, cloud infrastructure eliminates the need for large upfront capital investments while providing the flexibility to adapt to fluctuating workloads. In the public sector, these capabilities have enabled governments worldwide to accelerate the digitalisation of administrative processes, giving rise to a new generation of e-government platforms.

E-government services encompass a broad range of citizen-facing and back-office functions, from online identity verification and tax filing to public health management and social benefit administration. These services are characterised by high availability requirements, strict data security obligations, and the need to serve large, diverse populations. As a result, the infrastructure underpinning e-government platforms must be robust, elastic, and continuously monitored to ensure uninterrupted service delivery.

This chapter establishes the theoretical and contextual foundations necessary for understanding the resource supervision challenges addressed in this thesis. Section 2.2 introduces the fundamental concepts and service models of cloud computing. Section 2.3 examines the key resource metrics used to monitor cloud infrastructure. Section 2.4 analyses the specific requirements and infrastructure challenges inherent to e-government services. Section 2.5 presents relevant case studies from Algeria and Finland. Section 2.6 discusses current monitoring approaches and their limitations. Finally, Section 2.7 concludes the chapter.

II. Cloud Computing Fundamentals

1. Definition and Core Characteristics

The National Institute of Standards and Technology (NIST) defines cloud computing as a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources including networks, servers, storage, applications, and services that can be rapidly provisioned and released with minimal management effort or service provider interaction [53]. According to this foundational definition, cloud computing is characterised by five essential attributes:

- **On-demand self-service:** Users can provision computing resources automatically as needed, without requiring human interaction with the service provider.
- **Broad network access:** Resources are accessible over the network through standard mechanisms, supporting a wide variety of client platforms.

- **Resource pooling:** Provider resources are pooled to serve multiple consumers using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to demand.
- **Rapid elasticity:** Capabilities can be elastically provisioned and released to scale rapidly outward and inward in accordance with demand.
- **Measured service:** Cloud systems automatically control and optimise resource use by leveraging a metering capability appropriate to the type of service.

2. Cloud Service Models

Cloud computing is delivered through three primary service models, each offering a different level of abstraction and control [54]: These models allow organizations to select the level of control and operational complexity that best suits their requirements, ranging from complete infrastructure management to fully managed software solutions.

The three main service models are Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). Each model defines how responsibilities are shared between the cloud provider and the customer. **Table 2.1** summarizes the characteristics, examples, and management responsibilities associated with each service model.

Service Model	Description	Examples	Responsibility
Infrastructure as a Service (IaaS)	Provides virtualised computing infrastructure including virtual machines, storage, and networking.	AWS EC2, Azure VMs, Google Compute Engine	User manages OS, applications; provider manages hardware
Platform as a Service (PaaS)	Delivers a managed runtime environment for developing, testing, and deploying applications.	Google App Engine, Azure App Service, Heroku	User manages applications; provider manages OS and runtime
Software as a Service (SaaS)	Offers complete software applications over the internet on a subscription basis.	Microsoft 365, Salesforce, Google Workspace	Provider manages everything; user only interacts with the application

Table 2.1: Cloud service models comparison

As the level of abstraction increases from IaaS to SaaS, the responsibility of the cloud provider also increases while the management burden on the user decreases. Organizations can

therefore select the service model that best balances flexibility, control, and operational simplicity according to their business requirements.

3. Cloud Deployment Models

Cloud environments are deployed in four main configurations, each suited to different organizational requirements [55]:

1 **Public Cloud:** Resources are owned and operated by a third-party provider and shared among multiple tenants. Highly scalable and cost-effective, but may raise data sovereignty concerns for government use.

2 **Private Cloud:** Infrastructure is operated exclusively for a single organisation, either on-premises or hosted by a provider. Offers greater control and security, preferred for sensitive government data.

3 **Hybrid Cloud:** Combines public and private cloud environments connected through standardised interfaces. Allows governments to keep sensitive data private while leveraging public cloud scalability for high-demand services.

4 **Community Cloud:** Shared infrastructure provisioned for a specific community of organisations with common concerns (e.g., multiple government agencies sharing a secure platform).

Among these deployment models, hybrid cloud architectures are particularly relevant for e-government services because they provide a balance between scalability and data sovereignty requirements. Sensitive citizen data can remain within private infrastructures, while public cloud resources can be leveraged to handle periods of increased demand.

III. Cloud Resource Metrics

1. CPU Utilisation

CPU utilisation measures the percentage of processor capacity actively used at a given moment. It is one of the most critical indicators of server health and application performance. In e-government environments, CPU usage fluctuates significantly based on citizen activity patterns, peaking during business hours, tax deadlines, or national registration campaigns [56].

Sustained CPU utilisation above 85% typically signals the need for resource scaling. Conversely, consistently low CPU usage may indicate over-provisioning and unnecessary expenditure. Effective supervision requires not only tracking current utilisation but also predicting future CPU demand to enable proactive resource management.

2. Memory Consumption

Memory (RAM) consumption tracks the proportion of available random-access memory in use. Unlike CPU, which fluctuates rapidly, memory usage tends to grow progressively and may exhibit gradual "memory leak" patterns in applications that fail to release allocated resources [57]. For e-government platforms handling concurrent citizen sessions and large datasets, memory exhaustion can lead to application crashes and complete service unavailability.

Monitoring memory consumption alongside CPU enables a more complete picture of resource health. The combination of high CPU and rising memory usage is a strong precursor to imminent service degradation.

3. Network Traffic and Latency

Network throughput and latency metrics capture the volume of data transmitted and the delay experienced in communication between components. High network traffic can saturate bandwidth, causing response time degradation even when CPU and memory resources remain adequate. For e-government services that transmit sensitive documents, certificates, and large administrative datasets, network performance is a primary quality-of-service indicator [58].

4. Disk I/O and Storage Utilisation

Disk input/output operations per second (IOPS) and storage capacity utilisation reflect the health of persistent data infrastructure. Database-heavy e-government applications — such as civil registry systems or tax administration platforms — are particularly sensitive to storage bottlenecks. Slow disk access directly increases response times for citizens and administrators alike [59].

5. Service Availability and Response Time

Service availability measures the proportion of time a service is operational and accessible. Service Level Agreements (SLAs) commonly target availability levels such as 99.9% uptime. In this thesis, however, service availability below 99.5% is considered the critical threshold indicating service degradation. Response time captures the duration required to process and return a citizen request. Together, these metrics define the quality of service experience and are the primary indicators tracked in Service Level Agreements (SLAs) between government agencies and their cloud providers [60].

In addition to availability and response time, cloud infrastructures must continuously monitor resource-related metrics such as CPU utilization, memory consumption, network latency, and disk I/O activity. Exceeding critical thresholds for these metrics may negatively affect system performance, service reliability, and user satisfaction. **Table 2.2** summarizes the principal cloud resource metrics, their associated thresholds, and their potential impact on e-government services.

Metric	Unit	Critical Threshold	Impact if Exceeded
CPU Utilisation	Percentage (%)	> 85%	Processing delays, request queuing, service slowdown
Memory Consumption	Percentage (%)	> 90%	Application crashes, out-of-memory errors
Network Latency	Milliseconds (ms)	> 200ms	Poor user experience, session timeouts
Disk I/O	IOPS	Saturation point	Database query failures, slow file processing
Service Availability	Percentage (%)	< 99.5%	SLA violations, citizen dissatisfaction
Response Time	Milliseconds (ms)	> 2000ms	Citizen abandonment, service perception failure

Table 2.2: Key cloud resource metrics and critical thresholds for e-government services

These metrics constitute the primary indicators used throughout this thesis to characterize the state of cloud resources and support predictive scaling decisions. Their continuous monitoring provides the information required by the proposed Predictive Digital Twin framework to anticipate future resource requirements and assist administrators in proactive resource management.

IV. E-Government Services: Requirements and Infrastructure Challenges

1. Definition and Scope of E-Government

E-Government refers to the use of Information and Communication Technologies (ICT) by public administrations to deliver services, exchange information, and improve interactions with citizens, businesses, and other government entities. Its main objectives are to improve service accessibility, increase administrative efficiency, and enhance transparency.

According to the United Nations, e-government systems evolve through five maturity stages: Emerging, Enhanced, Interactive, Transactional, and Connected, with higher stages providing greater service integration and citizen participation [61].

E-government services are generally classified into three main interaction domains:

- **Government-to-Citizen (G2C):** Digital services provided directly to citizens, such as identity documents, tax declarations, and administrative procedures.
- **Government-to-Business (G2B):** Services supporting businesses, including registration, procurement, and regulatory compliance.
- **Government-to-Government (G2G):** Systems enabling information sharing and coordination between government agencies.

These interaction models collectively support the delivery of efficient and accessible digital public services.

2. Infrastructure Requirements for E-Government Services

E-government platforms impose stringent infrastructure requirements that distinguish them from typical commercial cloud applications [62]:

- **High Availability:** Government services must maintain near-continuous operation, with downtime windows often contractually limited to minutes per month. Citizens and businesses depend on these platforms for legally binding operations.
- **Data Sovereignty and Security:** National regulations frequently mandate that citizen data be stored and processed within national borders. Strong encryption, access control, and audit logging are non-negotiable requirements.
- **Scalability under Predictable Peaks:** Many government services experience highly predictable load spikes tax filing deadlines, election registration periods, national examination results requiring infrastructure capable of handling sudden surges.
- **Interoperability:** Government platforms must integrate with legacy systems, national identity databases, and third-party service providers through standardized APIs and data exchange protocols.
- **Regulatory Compliance:** Adherence to national e-government standards, data protection regulations (such as GDPR in Europe or its equivalents), and international cybersecurity frameworks is mandatory.

3. Infrastructure Challenges

Despite the advantages of cloud adoption, e-government services face several persistent infrastructure challenges that motivate the development of advanced supervision systems [63].

1 **Reactive Incident Management:** Most existing monitoring systems detect failures only after they have already disrupted service. This reactive approach results in citizen-visible outages, administrative delays, and reputational damage to public institutions.

2 **Resource Over-provisioning:** To guard against unexpected demand peaks, government IT departments often over-provision resources, leading to significant waste of public funds during off-peak periods.

3 **Heterogeneous Infrastructure:** Government IT environments typically combine on-premises legacy systems with modern cloud deployments, creating monitoring complexity and integration challenges.

4 **Lack of Predictive Insight:** Traditional monitoring tools provide dashboards of current metrics but lack forecasting capabilities. Administrators cannot anticipate resource exhaustion or performance degradation before it affects citizens.

5 **Skilled Resource Constraints:** Public sector IT teams may lack the specialized expertise required to implement advanced monitoring and optimization strategies, making automated and intelligent supervision tools especially valuable.

V. Case Studies: E-Government Infrastructure in Practice

1. Algeria: Digital Transformation of Public Administration

Algeria has embarked on a comprehensive e-government transformation programme under the national strategy for digital economy development. Key initiatives include the e-ALGERIA portal, which aggregates citizen-facing services across multiple ministries, and the digital civil status system that enables citizens to request official documents online [64].

The Algerian e-government infrastructure faces several notable challenges relevant to this thesis. Service platforms hosted on national data centre infrastructure experience significant performance variability during peak periods such as university enrolment seasons and tax declaration deadlines. The monitoring infrastructure in place relies primarily on threshold-based alerting, which can only react to failures rather than anticipating them. Several reports and studies have highlighted challenges related to service availability and performance during periods of high demand, emphasizing the need for improved monitoring and resource management strategies [65].

Furthermore, the Algerian context emphasises the importance of domestic cloud infrastructure, with national data sovereignty regulations requiring that sensitive citizen data remain within national territory. This constraint makes efficient resource utilisation particularly important, as scaling options may be more limited compared to global public cloud environments.

2. Finland: A Model for Advanced E-Government Infrastructure

Finland is consistently ranked among the leading countries in e-government development according to the United Nations E-Government Survey. The Finnish public sector has implemented advanced digital services through platforms such as **Suomi.fi**, which provides citizens and organizations with a unified access point to a wide range of government services using a common authentication and authorization framework [66].

To ensure high service availability and performance, Finnish e-government infrastructures rely on modern cloud technologies, continuous monitoring, and data-driven resource management practices. These approaches enable public services to remain reliable, scalable, and accessible even during periods of high demand [67].

The Finnish experience demonstrates the importance of proactive infrastructure management in maintaining service quality and supporting the digital transformation of public administration. These practices align with the objective of the proposed Predictive Digital Twin framework, which seeks to support proactive resource management through predictive analytics.

3. Comparative Analysis

Algeria and Finland represent two different levels of e-government maturity and cloud infrastructure adoption. While Algeria is still progressing toward large-scale digital transformation, Finland is recognized as one of the leading countries in digital public services. **Table 2.3** summarizes the main differences between the two environments and highlights their relevance to this research.

Dimension	Algeria	Finland
UN E-Gov Index (2022)	Developing stage	Top 5 globally
Cloud Adoption	Hybrid (national DC + limited public cloud)	Full cloud-native, multi-provider
Monitoring Approach	Threshold-based, reactive	Predictive, data-driven observability
Data Sovereignty	Strict national residency requirement	EU GDPR compliant, flexible
Key Challenges	Reactive incidents, peak-period failures	Cost optimisation, cross-agency integration
Relevance to Thesis	Direct context: need for proactive supervision	Reference model for predictive management

Table 2.3: Comparative analysis of e-government infrastructure: Algeria vs Finland

The comparison highlights the growing importance of predictive and data-driven infrastructure management. While Algeria continues to face challenges related to service

availability and reactive monitoring, Finland demonstrates how advanced monitoring and predictive approaches can support more reliable and efficient e-government services. These observations further motivate the development of the Predictive Digital Twin framework proposed in this thesis.

VI. Current Monitoring Approaches and Their Limitations

1. Threshold-Based Monitoring

Threshold-based monitoring is one of the most widely used approaches for supervising cloud infrastructures. In this method, predefined thresholds are assigned to resource metrics such as CPU utilization, memory consumption, network latency, or disk activity. When a metric exceeds its predefined threshold, an alert is generated and sent to system administrators for corrective action [68].

While simple to implement and effective for detecting obvious failures, this approach suffers from several limitations:

1. **Reactivity:** Alerts are generated only after the problematic condition has already manifested, meaning that some degree of service degradation has already occurred by the time an administrator can respond.
2. **False positives:** Static thresholds do not account for normal operational variability, leading to alert fatigue among operations staff.
3. **Lack of context:** Threshold alerts provide no indication of whether an elevated metric will return to normal or continue escalating toward failure.
4. **No lead time:** Administrators receive no advance warning, making it impossible to take preventive action before a service disruption occurs.

5. Time-Series Monitoring Tools

Modern monitoring platforms such as Prometheus, Grafana, and InfluxDB have significantly improved visibility into cloud resource behaviour by enabling continuous collection, storage, and visualisation of time-series metrics [69]. These tools provide operations teams with rich historical dashboards and correlation capabilities. However, despite their sophistication, they remain fundamentally descriptive tools: they show what has happened and what is happening now, but do not predict what will happen next.

6. The Gap: From Reactive to Predictive Supervision

The limitations of current monitoring approaches define the central research gap addressed in this thesis. E-government services require a supervision paradigm that goes

beyond reactive alerting and descriptive dashboards. What is needed is a predictive intelligence layer capable of forecasting resource states in advance, generating proactive warnings before problems occur, and recommending specific preventive actions to administrators [70].

This gap is particularly acute in the context of Digital Twin platforms, which are well-suited to integrate predictive AI capabilities but have not yet been fully exploited for proactive e-government service supervision. This limitation motivates the integration of predictive analytics within Digital Twin environments. By combining real-time monitoring with machine learning models, it becomes possible to anticipate future resource states and support proactive decision-making before service degradation occurs. The following chapters address this gap by proposing and implementing an AI-enhanced Digital Twin framework for proactive resource management.

VII. Conclusion

This chapter has provided the essential background for understanding the cloud computing infrastructure that underpins modern e-government services and the challenges inherent in its supervision. We have examined the core characteristics of cloud computing, the key resource metrics that determine service health, and the specific requirements and constraints of government digital platforms.

The comparative case studies of Algeria and Finland have illustrated both the current state of e-government infrastructure in different national contexts and the opportunities for improvement through more intelligent supervision approaches. The analysis of existing monitoring methods has confirmed the critical limitation of current reactive systems: by the time a problem is detected, service disruption has already begun.

The limitations identified in current monitoring approaches highlight the need for more intelligent and proactive supervision mechanisms. These observations motivate the integration of predictive analytics within a Digital Twin environment to support resource management and scaling decisions before service degradation occurs. The next chapter presents the conception and modeling of the proposed Predictive Digital Twin framework designed to address these challenges.

CHAPTER III: CONCEPTION AND MODELING

I. Introduction

Following the theoretical foundations presented in the previous chapters, this chapter describes the conception and modeling of the proposed Predictive Digital Twin framework for proactive cloud resource management. The proposed solution integrates Digital Twin technology, cloud telemetry monitoring, and Artificial Intelligence within a unified architecture capable of monitoring infrastructure resources, maintaining a synchronized virtual representation, and generating predictive scaling recommendations for e-government cloud services.

II. Requirements Analysis

To achieve the objectives of the proposed framework, a set of functional and non-functional requirements was identified. These requirements define the expected system behavior, performance, and operational constraints.

a. Functional Requirements

- The system must continuously collect telemetry data from the monitored cloud infrastructure, including CPU utilization, memory utilization, and disk activity.
- The Digital Twin must maintain an up-to-date virtual representation of the monitored resources and synchronize their operational state.
- The AI prediction service must analyze historical telemetry data and generate future scaling recommendations classified as scale_up, maintain, or scale_down.
- The system must provide a visualization interface allowing administrators to monitor resource utilization, consult prediction results, and observe the current state of the Digital Twin.
- The system must support communication between all components through standardized REST-based interfaces.

b. Non-Functional Requirements

- The system must ensure scalability to support multiple monitored cloud resources and e-government services simultaneously.
- The architecture must be modular to facilitate maintenance, integration, and future extensions.
- The prediction service must provide scaling recommendations with low latency, while telemetry data are collected every 10 seconds and the administrative dashboard is refreshed every 4 seconds to support near real-time decision-making.

- The system must ensure reliability by maintaining continuous synchronization between the physical infrastructure and its Digital Twin representation.
- The system must ensure security by protecting communications between components and restricting access to administrative functionalities through appropriate authentication and authorization mechanisms.

III. Proposed Predictive Digital Twin Architecture

The proposed architecture was designed to support proactive supervision of cloud resources hosting e-government services. It combines Digital Twin technology and Artificial Intelligence within a unified framework capable of monitoring infrastructure resources, analyzing historical telemetry, and generating predictive scaling recommendations.

To achieve these objectives, the framework is organized into five interconnected layers responsible for data collection, synchronization, prediction, and supervision. **Figure 3.1** presents the overall architecture of the proposed Predictive Digital Twin system and illustrates the interactions between its main components.

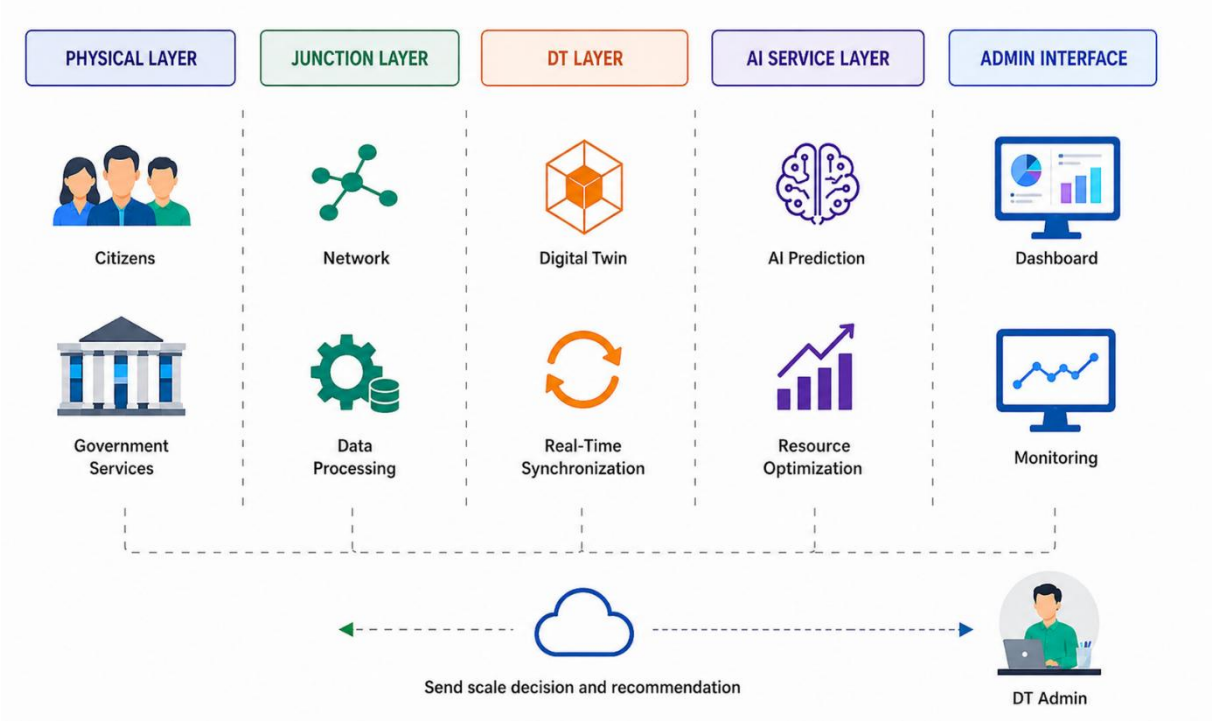


Figure 3.1: Global Architecture of the Proposed Predictive Digital Twin System

As illustrated in **Figure 3.1**, the architecture is organized into **five interconnected layers**. The **Physical Layer** represents the cloud infrastructure and e-government services that generate operational telemetry data. These data are collected and processed through the **Junction Layer**, which acts as a communication bridge between the physical environment and the Digital Twin. The **Digital Twin Layer** maintains a synchronized virtual representation of the monitored

resources and provides a centralized view of their current state. Building upon this representation, the **AI Service Layer** analyzes telemetry information and generates predictive scaling recommendations based on historical resource behavior. Finally, the **Admin Interface Layer** provides monitoring and visualization capabilities, enabling administrators to access Digital Twin information and prediction results through a unified dashboard.

This layered organization ensures a clear separation of responsibilities while enabling continuous data flow between monitoring, prediction, and decision-support components.

IV. Digital Twin Design

The proposed Digital Twin Layer was designed to provide a synchronized virtual representation of the monitored cloud infrastructure and serve as the central component connecting telemetry collection, predictive analytics, and administrative supervision.

1. Global System Functionalities

The proposed framework integrates monitoring, Digital Twin management, predictive analytics, and administrative supervision within a unified environment. Administrators interact with the platform through the dashboard, while the underlying layers continuously collect, synchronize, and analyze telemetry information.

Figure 3.2 presents a global view of the functionalities provided by the framework and the interactions between the administrator and the different system services.

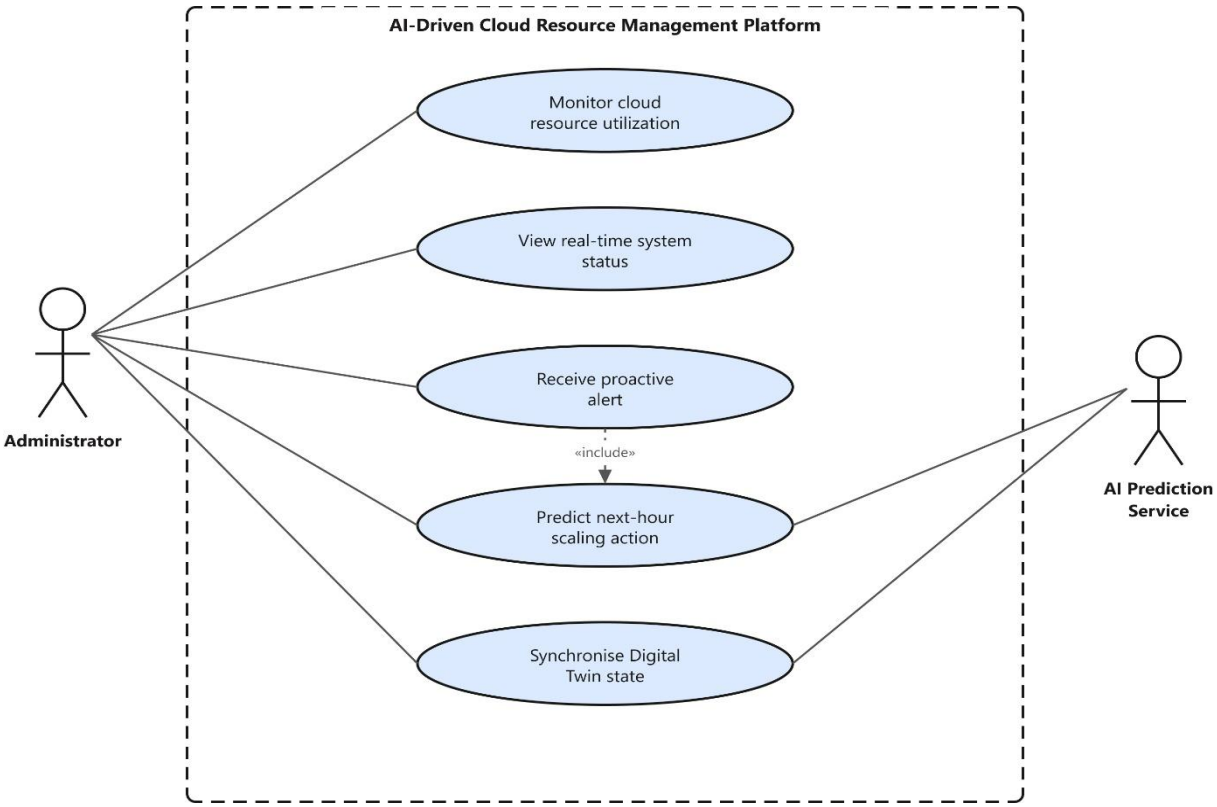


Figure 3.2: Global System Use Case Diagram

This diagram provides a high-level representation of the services available within the Predictive Digital Twin Layer and the interactions between the administrator and the system. The Administrator can monitor cloud resources, view real-time system status, and receive proactive alerts. Each alert includes a next-hour scaling prediction generated by the AI Prediction Service, shown here as a secondary actor that also keeps the Digital Twin synchronized with the physical infrastructure.

2. Digital Twin Conceptual Model

The Digital Twin Layer serves as the central component of the proposed framework by maintaining a virtual representation of the monitored cloud infrastructure. It stores telemetry information collected from physical resources and provides a unified view of the current system state for monitoring and prediction purposes.

The conceptual model is composed of several principal entities, including the monitored machine, telemetry metrics, Digital Twin Layer representation, prediction service, prediction results, and administrative dashboard. These entities collaborate to support data collection, synchronization, predictive analysis, and decision support.

Figure 3.3 presents the class diagram of the proposed framework and illustrates the relationships between the main system entities.

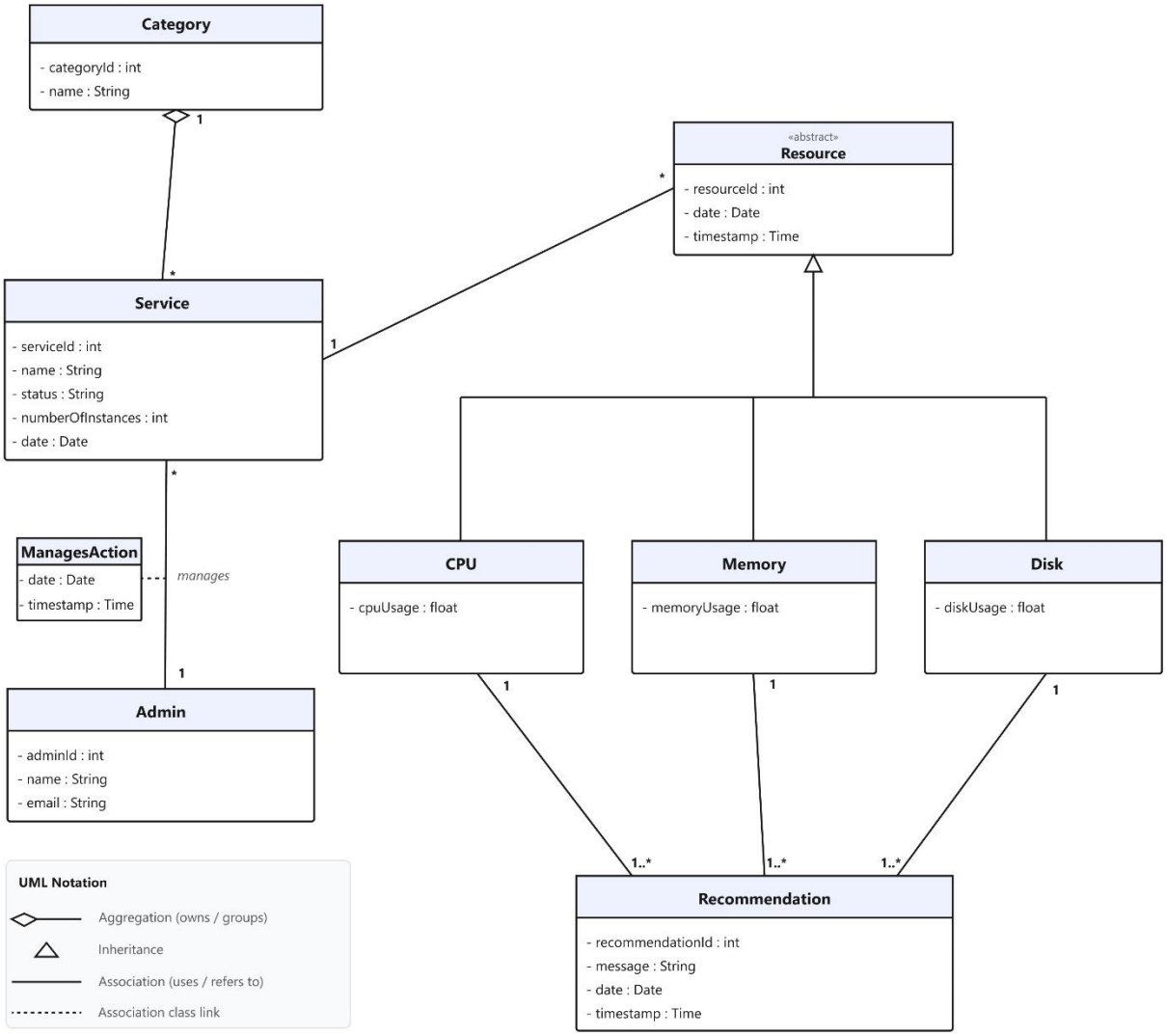


Figure 3.3: Class Diagram

This diagram provides a structural view of the framework's principal entities. A Category aggregates multiple Service instances, each associated with one or more Resource instances and managed by an Admin. Each management action is timestamped through the ManagesAction association class, recording the date and time an administrator acted on a given service. Resource is an abstract class specialized into CPU, Memory, and Disk, each carrying its own usage metric. Recommendation entities reference one instance of each resource subtype, expressing the scaling action generated by the prediction service at a given moment.

3. Digital Twin Synchronization Mechanism

A fundamental characteristic of a Digital Twin is the continuous synchronization between the physical system and its virtual counterpart. Within the proposed framework, telemetry data generated by the cloud infrastructure are transmitted through the Junction Layer and used to update the Digital Twin representation.

This synchronization process ensures that the Digital Twin Layer reflects the current state of monitored resources and provides accurate information to the prediction service and administrative dashboard.

Figure 3.4 illustrates the overall sequence of interactions between the different architectural layers, from telemetry acquisition to dashboard visualization.

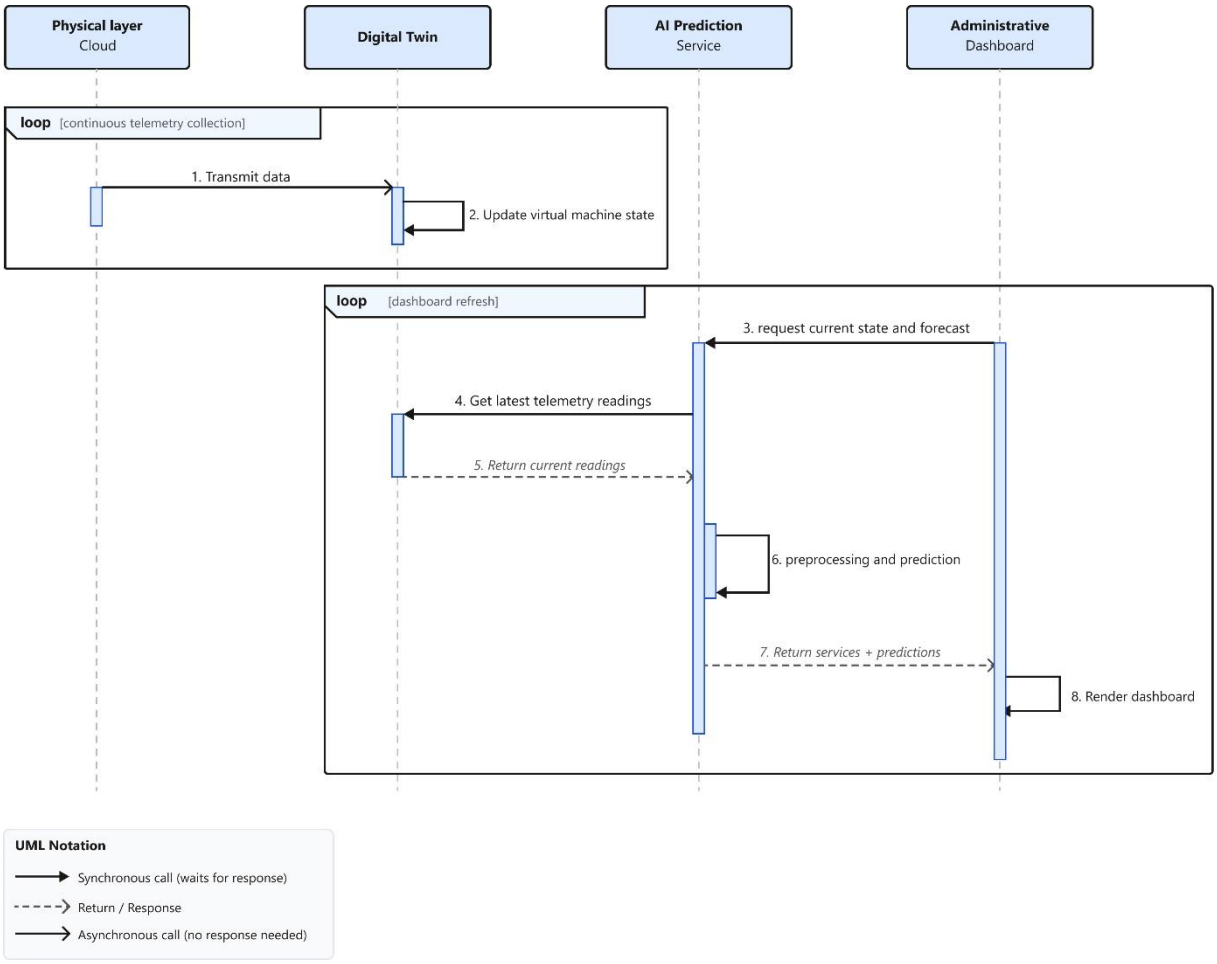


Figure 3.4: Global System Sequence Diagram

This diagram illustrates the overall sequence of interactions between the physical infrastructure, the Digital Twin, the AI Prediction Service, and the Administrative Dashboard. Telemetry data are continuously collected from the monitored infrastructure and used to update the Digital Twin Layer representation. The synchronized information is subsequently made available to both the prediction service and the administrative dashboard, ensuring a consistent view of the system state across all components.

4. Predictive Intelligence Extension

Traditional Digital Twins primarily focus on monitoring and visualization functionalities. In the proposed framework, predictive intelligence is integrated through the AI Service Layer,

enabling the system to anticipate future resource requirements and support proactive decision-making.

Historical telemetry data stored within the Digital Twin Layer are processed by predictive models that generate scaling recommendations classified as scale_up, maintain, or scale_down. These predictions are subsequently transmitted to the administrative dashboard and made available to administrators.

Figure 3.5 presents the prediction workflow implemented within the AI Service Layer and illustrates how telemetry data are transformed into predictive recommendations.

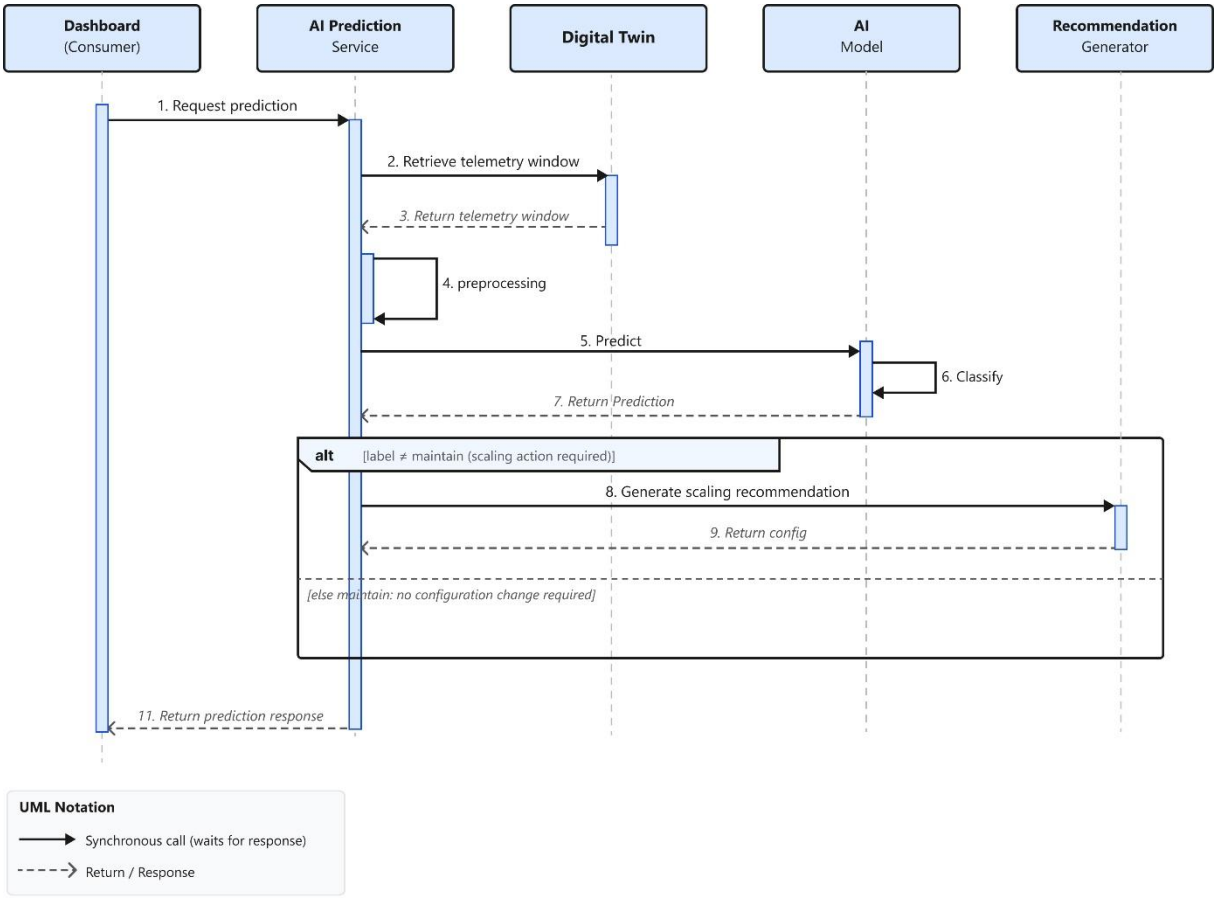


Figure 3.5: AI Prediction Service Sequence Diagram

This diagram illustrates the prediction workflow implemented within the AI Service Layer. Telemetry data stored within the Digital Twin Layer are retrieved and processed by the prediction service. After feature extraction, the predictive model generates a scaling recommendation. The resulting prediction is then transmitted to the administrative dashboard to support proactive resource management.

5. Administrative Supervision

The Admin Interface Layer provides a centralized environment for monitoring infrastructure resources and consulting prediction results. Through the dashboard, administrators can visualize resource utilization, monitor the Digital Twin state, request predictions, and analyze future scaling recommendations.

By combining real-time monitoring with predictive insights, the dashboard supports proactive resource management and facilitates operational decision-making.

Figure 3.6 illustrates the principal functionalities available to administrators through the supervision interface.

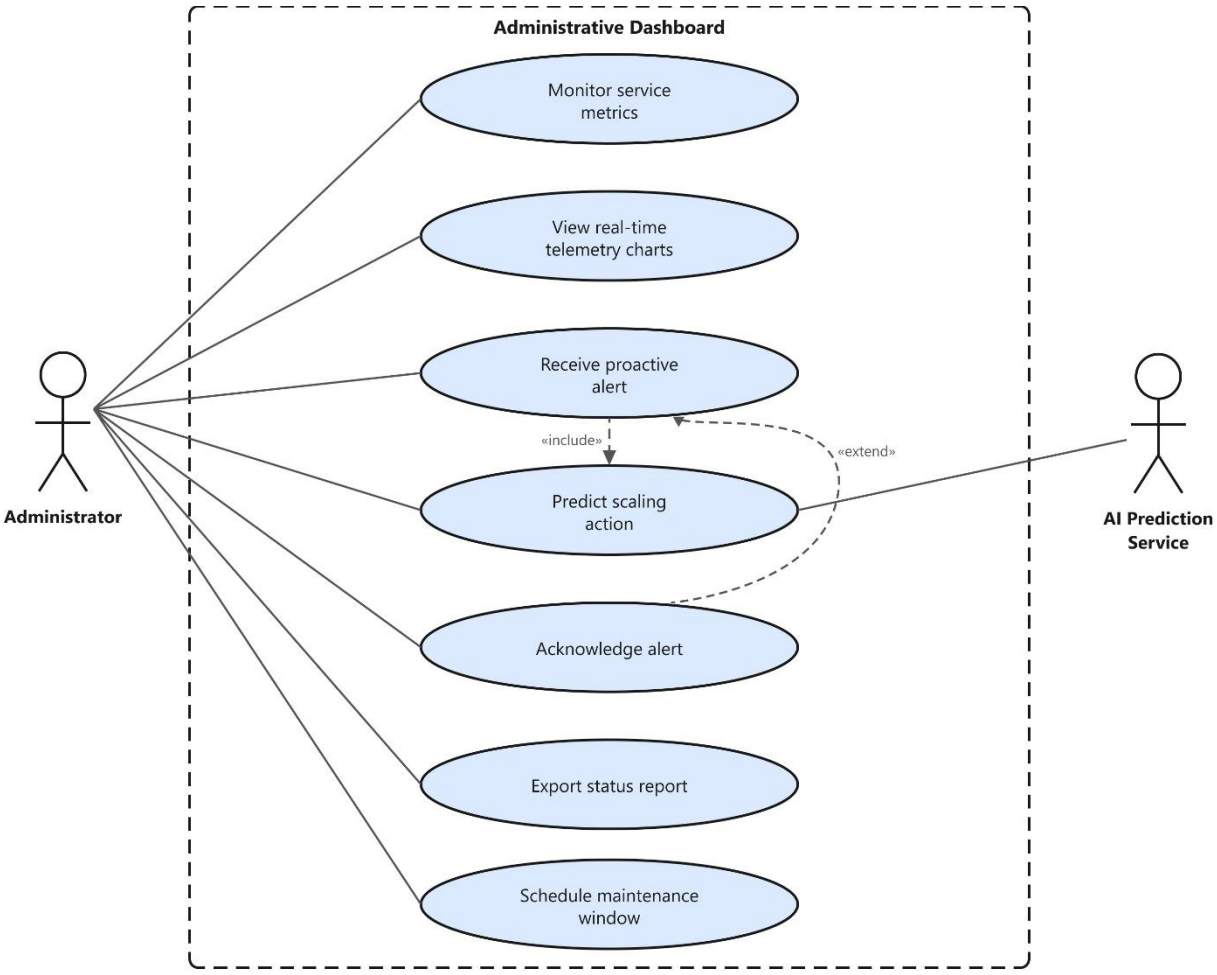


Figure 3.6: Administrative Dashboard Use Case Diagram

This diagram focuses on the functionalities available through the supervision interface. Administrators can monitor service metrics, view real-time telemetry charts, and receive proactive alerts, each of which includes a scaling prediction from the AI Prediction Service. Acknowledging an alert is shown as an extension, since it is an optional follow-up action the administrator may take once an alert is received. The dashboard also supports exporting status reports and scheduling maintenance windows.

V. Conclusion

This chapter presented the conception and modeling of the proposed Predictive Digital Twin framework for proactive cloud resource management. The functional and non-functional requirements of the system were identified, followed by the presentation of the five-layer architecture composed of the Physical Layer, Junction Layer, Digital Twin Layer, AI Service Layer, and Admin Interface Layer.

The chapter also described the Digital Twin design and the UML models used to represent the system functionalities, interactions, and structure. Particular attention was given to the AI Service Layer and the Admin Interface Layer, which constitute the principal contributions of this work by introducing predictive intelligence and decision-support capabilities into the Digital Twin environment.

The next chapter presents the implementation and evaluation of the proposed framework, including dataset preparation, model development, system integration, and experimental validation.

CHAPTER IV:
IMPLEMENTATION AND EVALUATION

I. Introduction

This chapter presents the implementation and evaluation of the proposed Predictive Digital Twin framework. The implementation phase covers dataset preparation, feature engineering, predictive model development, and integration of the Digital Twin environment with the AI prediction service.

The evaluation phase assesses the predictive performance of the implemented models using the Alibaba Cluster Trace 2018 dataset [75]. Experimental results, explainability analyses, and feature ablation studies are presented to evaluate the effectiveness of the proposed approach and identify the most influential factors affecting cloud resource scaling decisions.

II. Development Environment

The implementation of the proposed Predictive Digital Twin framework relies on several technologies for data processing, predictive analytics, system integration, and deployment.

- **Python 3.11:** Used as the primary programming language for data processing, feature engineering, machine learning, and system integration. Several libraries, including Scikit-Learn, XGBoost, TensorFlow, PyTorch, Pandas, NumPy, and SHAP, were used to support model development, data analysis, and explainability.
- **Eclipse Ditto:** Used as the Digital Twin platform responsible for creating and managing the virtual representation of monitored cloud resources.
- **Node-RED:** Served as the integration layer, enabling communication and data exchange between the Digital Twin, prediction services, and dashboard components.
- **Flask:** Used to develop the REST API responsible for exposing prediction services and handling interactions between system components.
- HTML, CSS, and JavaScript: Used to implement the administrative dashboard for visualization and monitoring purposes.
- **Docker:** Used to containerize and deploy the different components of the framework within a unified environment.
- **Google Colab and Visual Studio Code:** Used for model training, experimentation, and application development.

III. Dataset Preparation

The evaluation of the proposed framework was conducted using the Alibaba Cluster Trace 2018 dataset [75], a large-scale public cloud workload dataset containing telemetry measurements collected from production servers. The dataset provides resource utilization information that can be used to analyze workload behavior and develop predictive resource management solutions.

Due to the large size of the original dataset, a subset of ten machines was selected for experimentation. The selected machines contained sufficient telemetry records to support sliding-window generation while maintaining manageable computational requirements for model training and experimentation.

Three infrastructure metrics were retained for the study:

- CPU utilization
- Memory utilization
- Disk I/O activity

These metrics represent the principal resources involved in cloud scaling decisions and provide a comprehensive view of workload behavior.

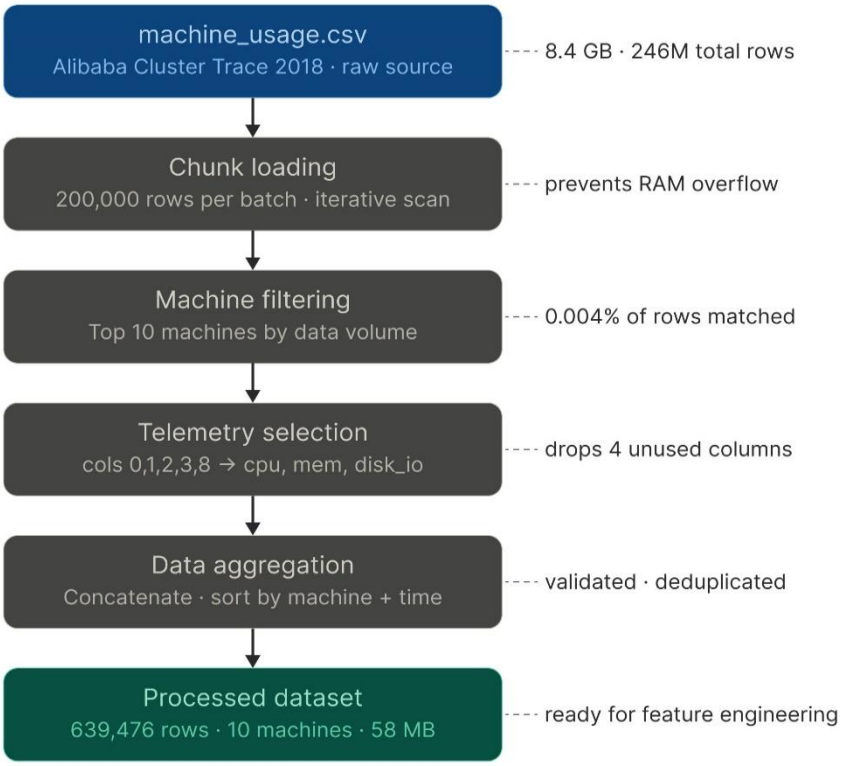


Figure 4.1: Chunk-Based Dataset Extraction Process

The selected telemetry streams were aggregated and merged into a unified dataset. Data from all machines were chronologically ordered to preserve temporal relationships and prepare the dataset for feature extraction and predictive modeling.

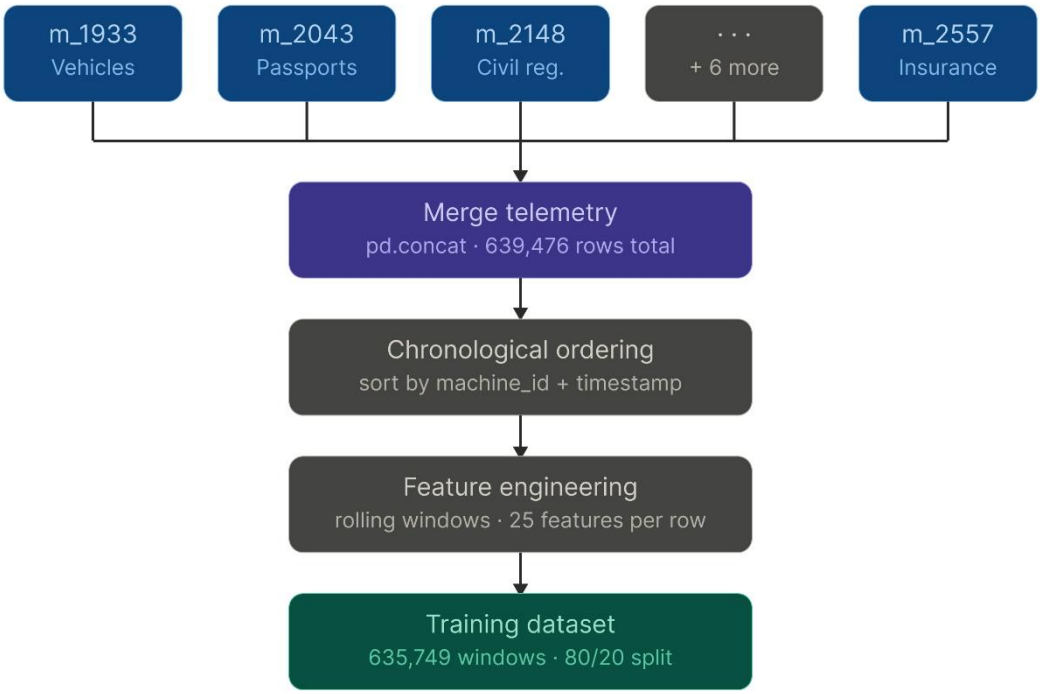


Figure 4.2: Multi-Machine Dataset Construction Workflow

The resulting dataset contains telemetry observations collected from ten cloud machines and constitutes the basis for all experiments performed in this work.

IV. Feature Engineering and Label Generation

To transform the raw telemetry streams into a format suitable for machine learning, a feature engineering process was applied. The objective was to summarize historical resource behavior and generate representative inputs for predictive models.

The telemetry data were first organized chronologically and divided into fixed-length sliding windows. Each window represents one hour of historical observations and serves as the input for future workload prediction.

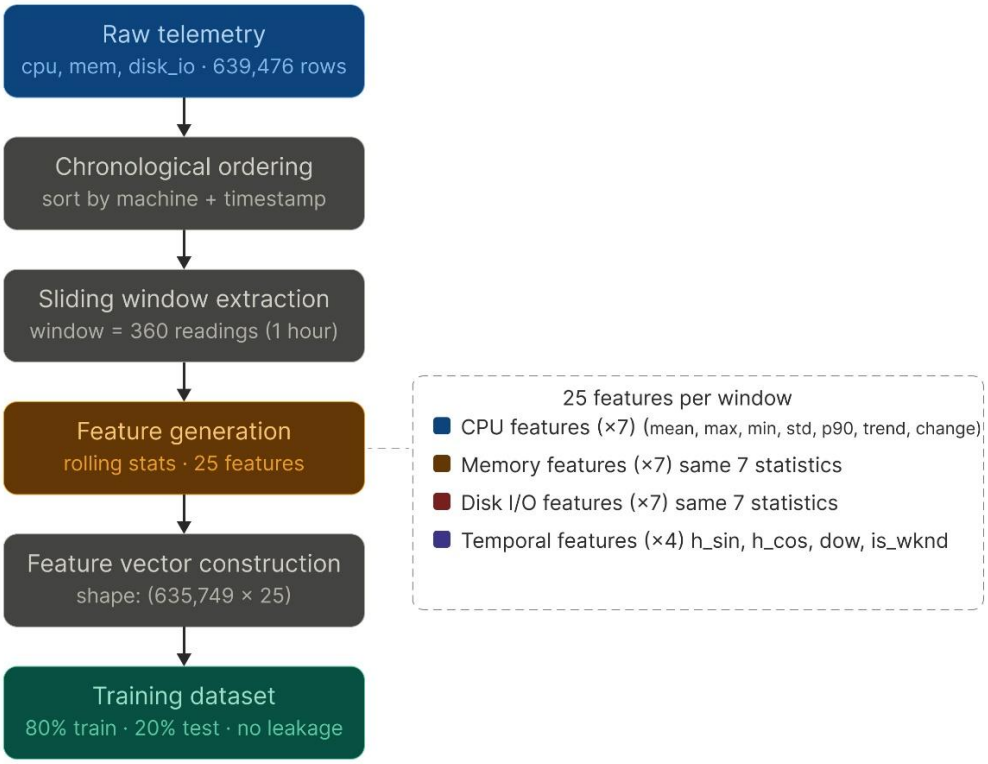


Figure 4.3: Data Preprocessing and Feature Engineering Pipeline

For each window, statistical indicators describing CPU utilization, memory utilization, and disk activity were extracted. Additional temporal and trend-based features were generated to capture workload evolution over time.

A total of twenty-five features were produced for each observation window. These features include statistical indicators such as mean, minimum, maximum, standard deviation, trend, and variation measures extracted from CPU, memory, and disk telemetry streams.

Future scaling decisions were generated using an adaptive threshold strategy based on future resource utilization levels. Each sample was assigned one of three labels :

- scale_up
- maintain
- scale_down

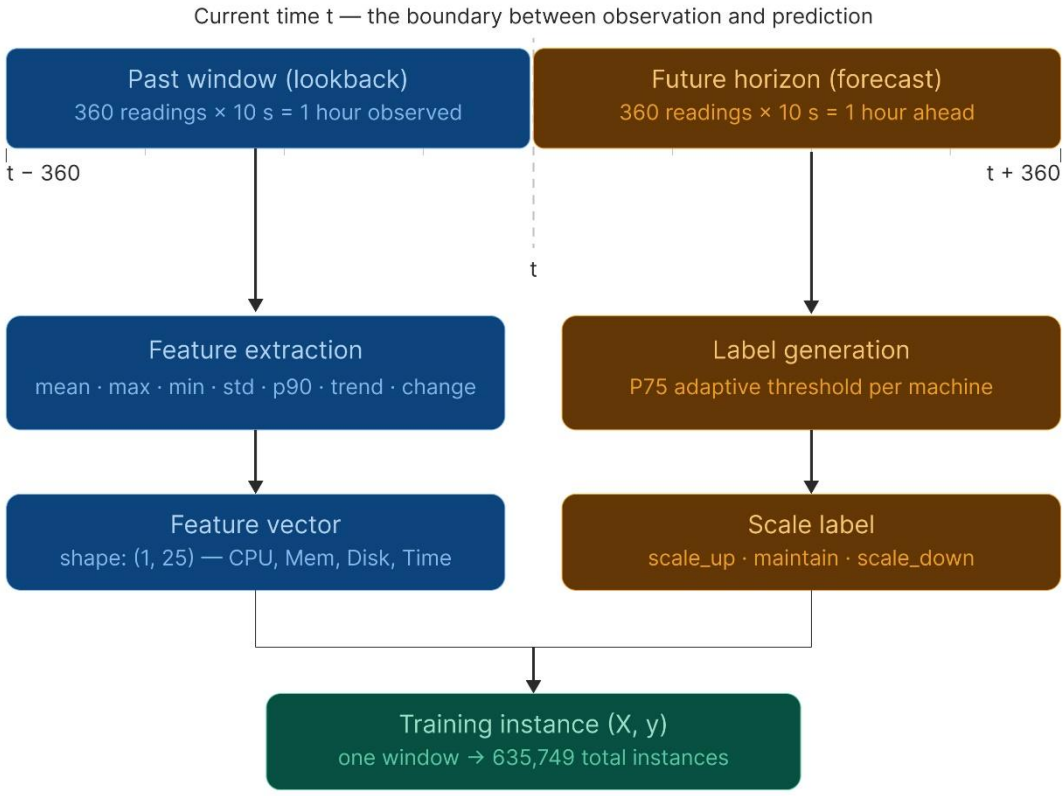


Figure 4.4: Sliding Window Generation Process

The resulting labeled dataset was subsequently used for training and evaluating the predictive models developed in this work.

V. Predictive Models Implementation

To evaluate the effectiveness of the proposed Predictive Digital Twin framework, four predictive models were implemented and compared. The selected models include both machine learning and deep learning approaches in order to investigate different strategies for cloud resource prediction.

a. Random Forest:

Random Forest was selected as a baseline machine learning model due to its robustness and ability to handle high-dimensional feature spaces. The model was trained using the generated feature vectors and used to predict future scaling decisions based on historical resource utilization patterns.

b. XGBoost:

XGBoost was implemented as an advanced ensemble learning technique capable of improving predictive performance through gradient boosting. The model was trained using the same feature set and was evaluated alongside the other approaches.

c. BiLSTM:

The Bidirectional Long Short-Term Memory (BiLSTM) network was implemented to capture temporal dependencies within the telemetry data. By processing historical observations in both forward and backward directions, the model is able to learn complex workload patterns and resource utilization trends.

d. Transformer:

A Transformer-based architecture was implemented to investigate the effectiveness of attention mechanisms for cloud workload prediction. The model analyzes relationships between telemetry features and identifies the most relevant information when generating future scaling decisions.

VI. Predictive Digital Twin Implementation

The Predictive Digital Twin framework was implemented by integrating Digital Twin services, telemetry processing components, predictive models, and visualization tools within a unified environment. The implementation architecture of the proposed framework is illustrated in **Figure 4.5.** Telemetry data generated by the monitored cloud infrastructure are first collected through the Junction Layer, where they are processed and transmitted to the Digital Twin environment. The Digital Twin maintains a synchronized virtual representation of the monitored resources and provides telemetry information to the AI prediction service. Based on historical resource behavior, the prediction service generates future scaling recommendations, which are then made available through the administrative dashboard. This architecture enables continuous monitoring, prediction, and decision support within a unified environment.

1. Digital Twin Layer

The Digital Twin layer was implemented using Eclipse Ditto. This component maintains a virtual representation of the monitored cloud resources and provides a centralized view of the infrastructure state.

2. Junction Layer

The Junction Layer was implemented using Node-RED. Its primary role is to collect telemetry information, manage data flows, and facilitate communication between the monitored infrastructure and the Digital Twin environment.

3. AI Prediction Service

The AI Prediction Service was implemented using Flask and serves as the predictive intelligence component of the framework. It receives telemetry information from the Digital Twin, applies the trained prediction models, and generates future scaling recommendations

classified as scale_up, maintain, or scale_down. The resulting predictions are then transmitted to the administrative dashboard to support proactive resource management and decision-making.

4. Administrative Dashboard

A web-based dashboard was developed to provide visualization and decision-support functionalities. Through this interface, administrators can monitor resource utilization, inspect Digital Twin information, and consult AI-generated scaling recommendations.

5. Containerized Deployment

The different system components were deployed using Docker containers. Containerization simplifies system deployment, improves portability, and facilitates communication between the various services composing the Predictive Digital Twin framework.

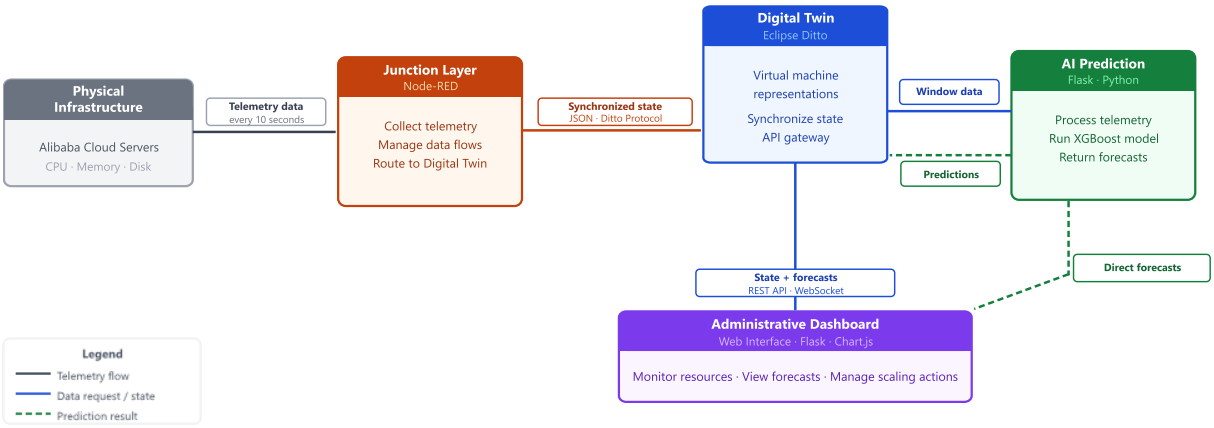


Figure 4.5: Implementation Architecture & Data Flow

The implemented architecture demonstrates the integration of Digital Twin technology and Artificial Intelligence within a single framework. By combining telemetry collection, Digital Twin synchronization, predictive analytics, and visualization services, the system provides administrators with both real-time infrastructure monitoring and proactive scaling recommendations.

VII. Experimental Setup

The predictive models were evaluated using the dataset generated from the Alibaba Cluster Trace 2018 telemetry records. To preserve temporal consistency and prevent information leakage, the dataset was divided using a chronological split strategy.

The first 80% of the generated samples were used for training, while the remaining 20% were reserved for testing. This approach ensures that future observations are never used during model training.

The evaluation was performed using four metrics:

- Accuracy
- Precision
- Recall
- Macro F1-score

Among these metrics, the Macro F1-score was selected as the primary evaluation criterion because it provides a balanced assessment of model performance across all prediction classes.

In addition to the standalone models, an ensemble combining Random Forest and BiLSTM was evaluated to investigate the benefits of integrating machine learning and deep learning approaches.

VIII. Results and Evaluation

Before model evaluation, the distribution of the generated scaling labels was analyzed in order to understand the characteristics of the prediction problem.

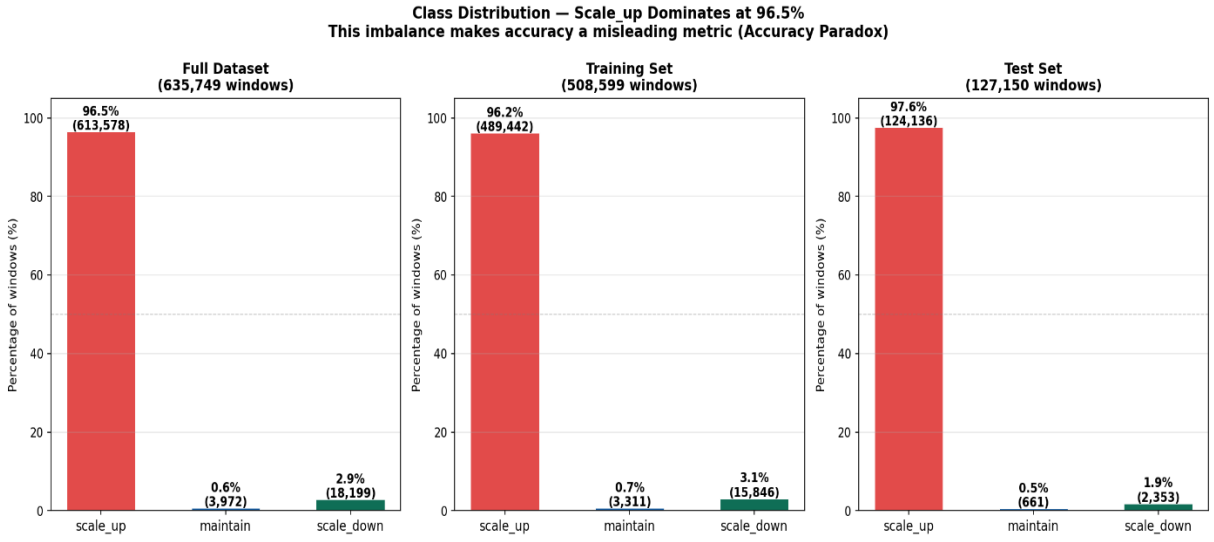


Figure 4.6: Class Distribution

The distribution reveals a significant dominance of scale_up events, while maintain and scale_down events occur less frequently. This imbalance increases the difficulty of accurately predicting minority scaling actions.

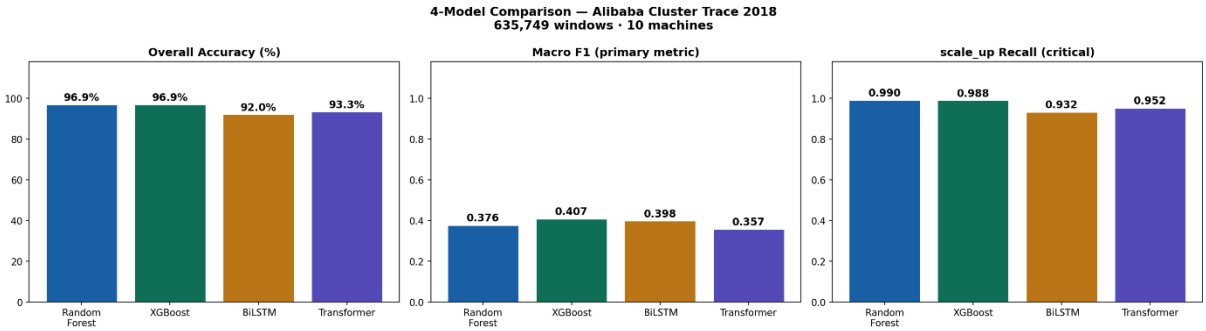


Figure 4.7: Model Performance Comparison

The results indicate that XGBoost achieved the highest standalone Macro F1-score (0.4075), making it the most effective individual model for the considered prediction task. The BiLSTM model achieved competitive performance and demonstrated stronger detection of scale-down events, highlighting its ability to capture temporal workload patterns.

The Transformer model achieved satisfactory performance but did not outperform the other approaches. Random Forest provided stable results and served as a strong baseline throughout the evaluation.

IX. Explainability Analysis

Beyond predictive performance, explainability techniques were applied to better understand the factors influencing scaling decisions generated by the models.

- **SHAP Feature Importance**

SHAP analysis was performed on the Random Forest model to quantify the contribution of each feature to the prediction process and improve the interpretability of the model's scaling decisions [76].

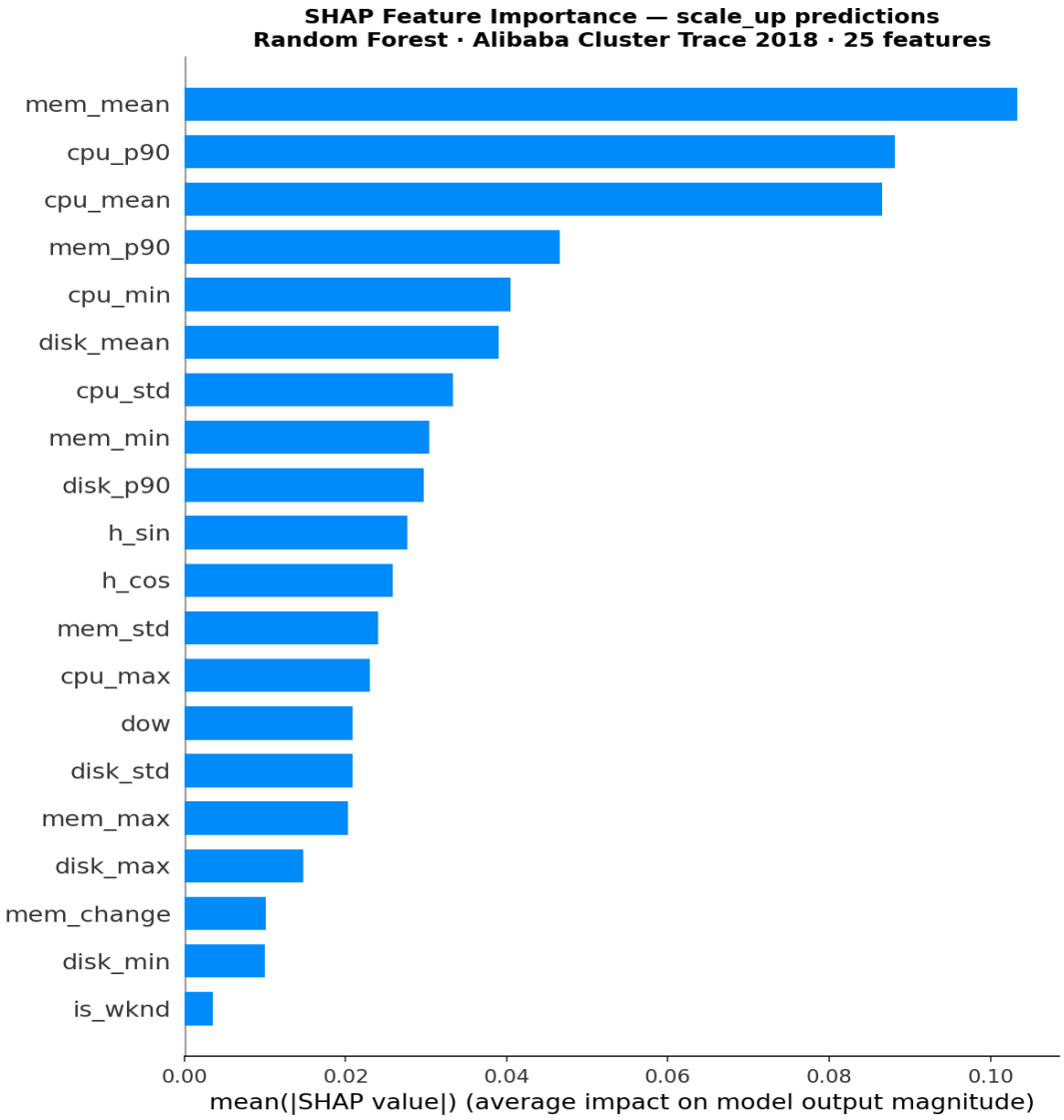


Figure 4.8: SHAP Feature Importance

The SHAP results indicate that memory-related features are the most influential variables affecting future scaling decisions. In particular, mem_mean (average memory utilization) is identified as the most important feature, followed by cpu_p90, cpu_mean, and mem_p90. This indicates that both sustained memory utilization and high CPU usage significantly influence the prediction of future scaling actions.

In addition, disk-related features, including disk_mean, disk_p90, and disk_std, also contribute to the prediction process, although their influence is lower than that of CPU and memory features. Temporal features such as hour of day (h_sin, h_cos) and day of week (dow) exhibit a smaller but non-negligible contribution, suggesting that periodic workload patterns also help the model distinguish between different scaling scenarios.

Overall, the explainability analysis demonstrates that memory utilization is the dominant factor, while CPU behavior and disk activity provide complementary information that improves the prediction of future cloud resource requirements.

X. Ablation Study

To evaluate the contribution of different resource categories, an ablation study was conducted using seven feature subsets: CPU only, Memory only, Disk only, CPU+Memory, CPU+Disk, Memory+Disk, and All Features.

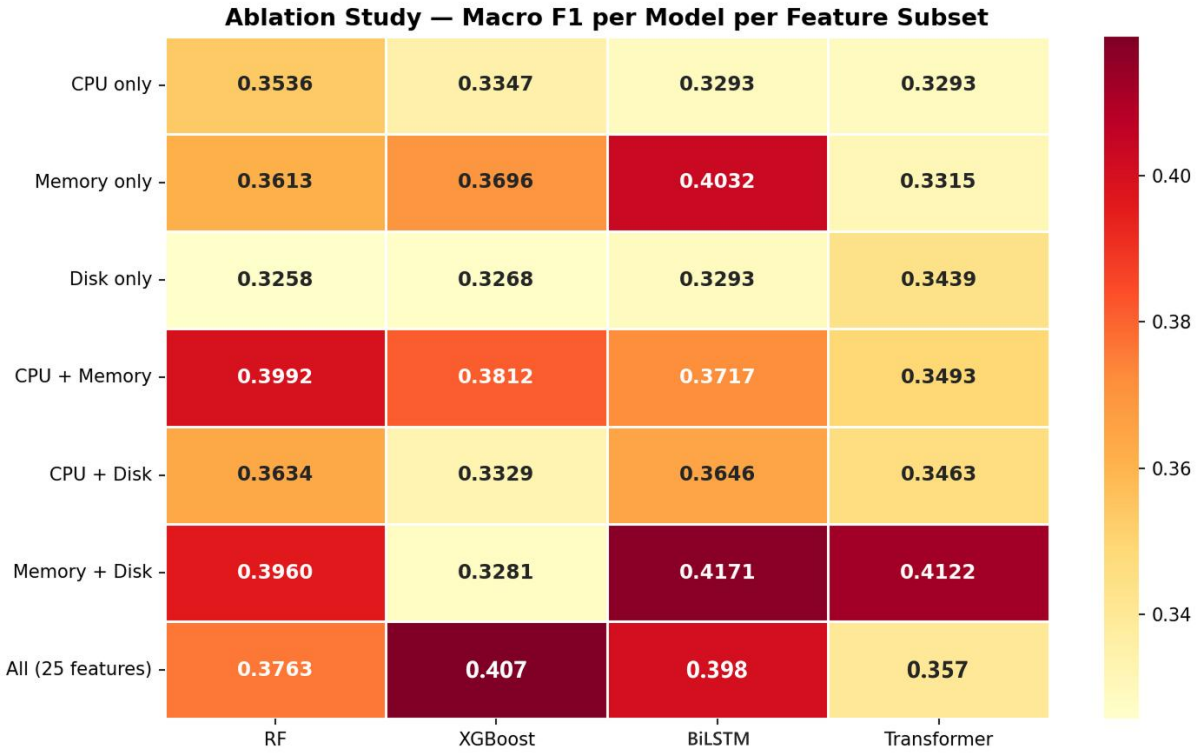


Figure 4.9: Ablation Heatmap

The results indicate that memory-related features consistently produce the strongest predictive performance across most models. The Memory+Disk subset achieved competitive performance, while CPU-only configurations generally produced lower results.

These observations confirm the importance of memory utilization for future scaling decisions and support the findings obtained from SHAP analysis.

XI. Additional Experiment: Period-Based Prediction

In addition to the primary sliding-window methodology, a secondary period-based prediction experiment was conducted. The objective of this experiment was to evaluate whether aggregated workload periods could provide a simpler alternative to sliding-window forecasting.

The period-based dataset contains approximately 430 aggregated observations, making it substantially smaller than the sliding-window dataset containing more than 635,000 prediction windows.

Random Forest and XGBoost models were trained and evaluated using this alternative representation. The resulting performance was significantly lower than that obtained using the sliding-window methodology, indicating that the reduced dataset size limits the ability of predictive models to learn representative workload patterns.

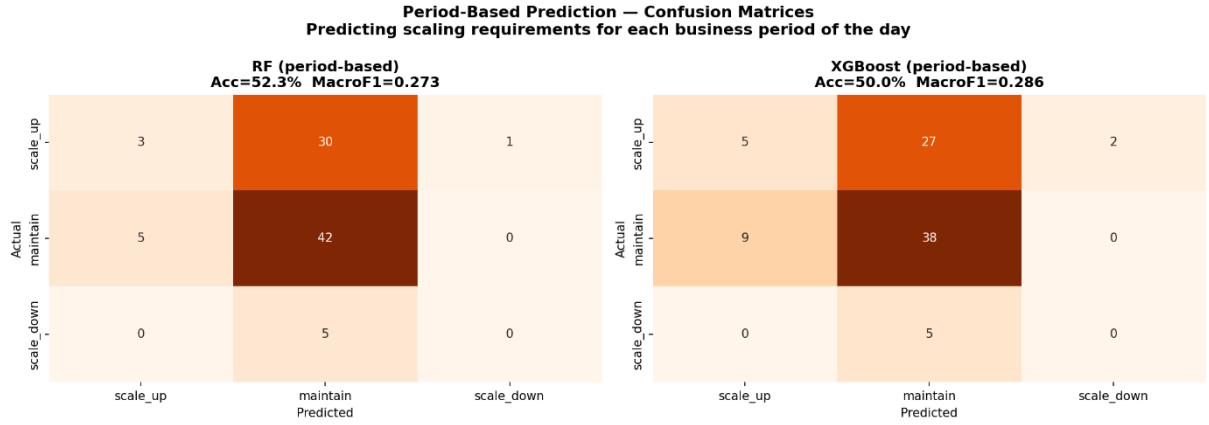


Figure 4.10 – Period-Based Model Confusion Matrices

Figure 4.10 compares the confusion matrices obtained using the period-based prediction methodology. Both Random Forest and XGBoost achieve substantially lower Macro F1-scores than their sliding-window counterparts. These results suggest that fine-grained telemetry observations provide richer predictive information than coarse workload aggregation. The experiment therefore supports the adoption of the sliding-window forecasting methodology as the primary prediction framework used throughout this research.

XII. Comparison with Related Works

To position the proposed framework within the existing literature, the obtained results were compared with representative studies addressing cloud workload prediction and resource management.

Study	Dataset	Resources	Model	Objective	Evaluation Metric	Main Result
Gupta et al. (2018) [77]	Alibaba, Google, Bitbrains	CPU, Memory	Deep BiLSTM	Resource Usage Prediction	RMSE	BiLSTM outperformed conventional LSTM models
Zhu et al. (2019) [78]	Alibaba Cluster Trace	CPU	Attention-LSTM	Workload Forecasting	RMSE	Attention improved forecasting accuracy
Patel et al. (2023)[79]	Google Cluster Trace	CPU, Memory	Attention-GRU	Multi-Step Forecasting	MAE	Improved long-term workload prediction
PRPOS Framework [80]	Alibaba Cluster Trace 2018	CPU, Memory	Periodicity-Based Prediction	Resource Forecasting	MAE	Improved periodic workload prediction
Proposed Work	Alibaba Cluster Trace 2018	CPU, Memory, Disk I/O	RF, XGBoost, BiLSTM, Transformer	Scaling Decision Prediction	Macro F1	XGBoost achieved Macro F1 = 0.4075

Table 4.2: Comparison with Representative Cloud Prediction Studies

The comparison reveals that most existing studies focus on forecasting future resource utilization values such as CPU usage, memory consumption, or workload intensity. Consequently, these works are typically evaluated using regression metrics such as RMSE, MAE, or MAPE.

In contrast, the proposed framework addresses a different problem by directly predicting operational scaling decisions classified as scale_up, maintain, and scale_down. As a result, classification-oriented metrics such as Macro F1-score are more appropriate for evaluating performance. Therefore, Direct numerical comparison is difficult because previous studies

focused on forecasting resource utilization values using regression metrics such as RMSE and MAE, whereas the proposed framework predicts scaling decisions and is evaluated using Macro F1-score. Therefore, the comparison focuses on objectives, datasets, and predictive capabilities rather than raw metric values.

Nevertheless, the comparison highlights a common objective: enabling proactive cloud resource management through predictive analytics. Unlike previous forecasting approaches, the proposed framework integrates prediction capabilities within a Digital Twin environment and combines machine learning, deep learning, explainability techniques, and decision-support functionalities.

XIII. Limitations and Future Work

Although the proposed framework demonstrated promising results, several limitations remain. First, the experiments were conducted using telemetry data collected from ten machines extracted from the Alibaba Cluster Trace 2018 dataset [75]. A larger number of machines and longer observation periods could improve the generalization of the predictive models.

Second, the dataset exhibits a significant class imbalance, resulting in a limited number of maintain and scale_down samples. This imbalance increases the difficulty of accurately predicting minority scaling events.

Third, a secondary period-based prediction experiment was conducted using periodic workload observations. The obtained results were considerably lower than those achieved using the sliding-window approach, mainly due to the limited number of available periodic instances.

Future work may investigate larger cloud datasets, additional prediction models, advanced ensemble techniques, and real-time deployment within operational cloud environments. Particular attention could be given to forecasting-oriented approaches such as Facebook Prophet and NeuralProphet, which may be better suited for capturing long-term workload seasonality and periodic resource consumption patterns.

Future research may also explore real-time deployment within production cloud environments in order to evaluate the framework under operational conditions.

XIV. Conclusion

This chapter presented the implementation and evaluation of the proposed Predictive Digital Twin framework. The implementation process included dataset preparation, feature engineering, predictive model development, and integration of the Digital Twin environment with the AI prediction service.

Experimental results demonstrated the feasibility of predicting future cloud scaling decisions using machine learning and deep learning techniques. Among the evaluated standalone models, XGBoost achieved the highest Macro F1-score, while BiLSTM demonstrated competitive performance and stronger detection of minority scaling events.

The explainability analyses confirmed the dominant influence of memory utilization on scaling decisions, while the comparison with related works highlighted the contribution of integrating predictive analytics, Digital Twin technology, and explainable artificial intelligence within a unified framework.

The next section presents the General Conclusion, summarizing the principal contributions, limitations, and future perspectives of this research.

General Conclusion

The rapid growth of cloud-based e-government services has increased the importance of efficient resource management in order to guarantee service availability, performance, and scalability. Traditional monitoring approaches primarily focus on observing the current state of infrastructure resources, often providing limited support for anticipating future workload variations. This limitation motivated the development of a Predictive Digital Twin framework capable of combining infrastructure monitoring with predictive intelligence.

The objective of this research was to design and implement a Digital Twin-based framework capable of predicting future cloud resource scaling requirements using machine learning and deep learning techniques. To achieve this objective, a multi-layer architecture was proposed, integrating telemetry collection, Digital Twin synchronization, predictive analytics, and administrative decision support within a unified environment.

The implementation and evaluation of the framework were conducted using the Alibaba Cluster Trace 2018 dataset. A complete data processing pipeline was developed, including dataset extraction, feature engineering, sliding-window generation, and adaptive label creation. Four predictive models were implemented and evaluated: Random Forest, XGBoost, BiLSTM, and Transformer.

The experimental results demonstrated the feasibility of predicting future scaling decisions using historical telemetry information. Among the standalone models, XGBoost achieved the highest Macro F1-score (0.4075), while BiLSTM demonstrated competitive performance and superior detection of minority scaling events. Furthermore, the RF+BiLSTM ensemble achieved the best overall predictive performance with a Macro F1-score of 0.4325, confirming the benefits of combining machine learning and deep learning approaches.

The explainability analyses provided additional insights into the prediction process. SHAP feature importance and ablation experiments consistently revealed that memory utilization is the most influential factor affecting future scaling decisions. These findings were consistently confirmed by the explainability and ablation analyses, highlighting the importance of memory-related features in scaling decision prediction. which highlighted the importance of memory and disk-related features.

Beyond predictive performance, this work contributes by integrating Digital Twin technology, predictive analytics, and explainable artificial intelligence within a single framework. Unlike traditional workload forecasting studies that focus on predicting resource

utilization values, the proposed approach directly predicts operational scaling decisions and presents them through an administrative decision-support interface.

Despite the encouraging results, several limitations remain. The experiments were conducted using a subset of machines extracted from the Alibaba Cluster Trace 2018 dataset, and the generated dataset exhibits a significant class imbalance. In addition, the period-based prediction experiment produced lower performance than the sliding-window approach due to the limited number of available training instances.

Future research may investigate larger cloud datasets, advanced ensemble strategies, and forecasting-oriented models such as Facebook Prophet and NeuralProphet for long-term workload prediction. Additional studies may also explore real-time deployment within production cloud environments and evaluate the framework under operational conditions.

In conclusion, this research demonstrates that the integration of Digital Twin technology and artificial intelligence can provide an effective foundation for proactive cloud resource management. By transforming traditional monitoring systems into predictive decision-support platforms, the proposed framework contributes toward the development of more intelligent, adaptive, and efficient cloud infrastructures for future e-government services.

BIBLIOGRAPHY

[1] Tao, F., Zhang, H., Liu, A., & Nee, A. Y. C. (2019). Digital twin in industry: State-of-the-art. IEEE Transactions on Industrial Informatics, 15(4), 2405-2415.

[2] Liu, M., Fang, S., Dong, H., & Xu, C. (2021). Review of digital twin about concepts, technologies, and industrial applications. Journal of Manufacturing Systems, 58, 346-361.

[3] Grieves, M. (2014). Digital twin: Manufacturing excellence through virtual factory replication. White paper, Florida Institute of Technology.

[4] Glaessgen, E. H., & Stargel, D. S. (2012). The digital twin paradigm for future NASA and US Air Force vehicles. 53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference.

[5] Kritzinger, W., Karner, M., Traar, G., Henjes, J., & Sihn, W. (2018). Digital twin in manufacturing: A categorical literature review and classification. IFAC-PapersOnLine, 51(11), 1016-1022.

[6] Qi, Q., Tao, F., Hu, T., Anwer, N., Liu, A., Wei, Y., ... & Nee, A. Y. C. (2021). Enabling technologies and tools for digital twin. Journal of Manufacturing Systems, 58, 3-21.

[7] Jones, D., Snider, C., Nassehi, A., Yon, J., & Hicks, B. (2020). Characterising the digital twin: A systematic literature review. CIRP Journal of Manufacturing Science and Technology, 29, 36-52.

[8] Madni, A. M., Madni, C. C., & Lucero, S. D. (2019). Leveraging digital twin technology in model-based systems engineering. Systems, 7(1), 7.

[9] Rasheed, A., San, O., & Kvamsdal, T. (2020). Digital twin: Values, challenges and enablers from a modeling perspective. IEEE Access, 8, 21980-22012.

[10] Fuller, A., Fan, Z., Day, C., & Barlow, C. (2020). Digital twin: Enabling technologies, challenges and open research. IEEE Access, 8, 108952-108971.

[11] Alam, K. M., & El Saddik, A. (2017). C2PS: A digital twin architecture reference model for the cloud-based cyber-physical systems. IEEE Access, 5, 2050-2062.

[12] Redelinghuys, A. J., Basson, A. H., & Kruger, K. (2020). A six-layer digital twin architecture for a manufacturing cell. In Service Oriented, Holonic and Multi-agent Manufacturing Systems for Industry 4.0 (pp.193-210). Springer.

[13] Aheleroff, S., Xu, X., Zhong, R. Y., & Lu, Y. (2021). Digital twin as a service (DTaaS) in industry 4.0: An architecture reference model. Advanced Engineering Informatics, 47, 101225.

[14] Barricelli, B. R., Casiraghi, E., & Fogli, D. (2019). A survey on digital twin: Definitions, characteristics, applications, and design implications. IEEE Access, 7, 167653-167671.

[15] Boyes, H., Hallaq, B., Cunningham, J., & Watson, T. (2018). The industrial internet of things (IIoT): An analysis framework. Computers in Industry, 101, 1-12.

[16] Sittón-Candanedo, I., Alonso, R. S., Corchado, J. M., Rodríguez-González, S., & Casado-Vara, R. (2019). A review of edge computing reference architectures and a new global edge proposal. Future Generation Computer Systems, 99, 278-293.

[17] Kherbache, M., Maimour, M., & Rondeau, E. (2022). Digital twin network for the IIoT using eclipse ditto and hono. IFAC-PapersOnLine, 55(8), 37-42.

[18] Qi, Q., & Tao, F. (2018). Digital twin and big data towards smart manufacturing and industry 4.0: 360 degree comparison. IEEE Access, 6, 3585-3593.

[19] Minerva, R., Lee, G. M., & Crespi, N. (2020). Digital twin in the IoT context: A survey on technical features, scenarios, and archi tectural models. Proceedings of the IEEE, 108(10), 1785-1824.

[20] Wang, K., Wang, Y., Li, Y., Fan, X., Xiao, S., & Hu, L. (2020). A review of the edge-cloud collaboration architectures for the industrial internet of things. IEEE Internet of Things Journal, 8(6), 4147-4164.

[21] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural Computation, 9(8), 1735-1780.

[22] Joshi, P. R., & Islam, S. (2018). E-government maturity model for sustainable e-government services. Sustainability, 10(6), 1-18.

[23] Toubal, E. B., & Rahim, M. (2024). Conception and Development of a Digital Twin for the Control and Supervision of E-Government Services [Master's thesis]. University of Medea.

[24] Zhang, M., Tao, F., Huang, B., Liu, A., Wang, L., Anwer, N., & Nee, A. Y. C. (2021). Digital twin data: methods and key technologies. Digital Twin, 1(2), 1-15.

[25] Dalenogare, L. S., Benitez, G. B., Ayala, N. F., & Frank, A. G. (2018). The expected contribution of industry 4.0 technologies for industrial performance. International Journal of Production Economics, 204, 383-394.

[26] Singh, M., Srivastava, R., & Fuenmayor, E. (2022). Applications of digital twin across industries: A review. Sensors, 22(15), 5727.

[27] Fuller, A., Fan, Z., Day, C., & Barlow, C. (2020). Digital twin: Enabling technologies, challenges and open research. IEEE Access, 8, 108952-108971.

[28] Attaran, M., & Celik, B. G. (2023). Digital twin: Benefits, use cases, challenges, and opportunities. Decision Analytics Journal, 6, 100165.

[29] Semeraro, C., Lezoche, M., Panetto, H., & Dassisti, M. (2021). Digital twin paradigm: A systematic literature review for smart manufacturing. Advanced Engineering Informatics, 47, 101229.

[30] Grieves, M., & Vickers, J. (2016). Digital Twin: Mitigating Clouded Vision. In Transdisciplinary Perspectives on Complex Systems (pp. 73-92). Springer.

[31] IBM. (2025). Digital Twin Maturity Model: Yung Woon Kim. [Online]. Available: https://www.ibm.com/think/topics/digital-twin-maturity-model.

[32] Singh, M., Srivastava, R., & Fuenmayor, E. (2022). Applications of digital twin across industries: A review. *Sensors*, 22(15), 5727.

[33] Russell, S. J., & Norvig, P. (2016). *Artificial Intelligence: A Modern Approach*. Global Edition. Pearson.

[34] Kumar, P. (2018). *Forecasting Cloud Resource Utilization Using Time Series Methods*.

[35] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735-1780.

[36] Taylor, S. J., & Letham, B. (2018). Forecasting at scale. *The American Statistician*, 72(1), 37-45.

[37] Hyndman, R. J., & Athanasopoulos, G. (2018). *Forecasting: Principles and Practice*. OTexts.

[38] Géron, A. (2019). *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow*. O'Reilly Media.

[39] Du, X., et al. (2026) AI-driven digital twins: A unified four-stage framework, arXiv preprint https://arxiv.org/html/2601.01321v1

[40] Costa, M., & Del Papa, G. (2025) AI-Enhanced Digital Twins: A Unified Predictive Framework, ResearchGate / Applied Sciences. https://www.researchgate.net/publication/396638516

[41] JavaCodeGeeks. (2025) Digital Twins with Eclipse Ditto for Predictive Maintenance, JavaCodeGeeks Technical Blog. https://www.javacodegeeks.com/2025/10/digital-twins-with-eclipse-ditto-modeling-physical-assets-for-predictive-maintenance.html

[42] Julius AI / Glossary. (2025), Prescriptive Analytics: The Pinnacle of Analytics Maturity. Julius.ai URL: https://julius.ai/glossary/prescriptive-analytics

[47] Kitchin, R. (2022). Adaptive Management of Behavioural Development for Smart Cities: A Review of Cognitive Digital Twins for Urban Governance.

[48] Yan, et al. (2021). HANSEL: Adaptive horizontal scaling for microservices using Bi-LSTM. Frontiers in Computer Science.

[49] Eclipse Foundation. (2026). Eclipse Ditto: Architecture and Concepts Overview. Retrieved from https://eclipse.dev/ditto/architecture-overview.html.

[50] Node-RED. (2026). IoT Automation with Node-RED and Docker. Retrieved from https://nodered.org/docs/getting-started/docker.

[51] Géron, A. (2019). Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow. O'Reilly Media.

[52] Pranaya, L. (2025). Getting Started with Eclipse Ditto Using Docker. Medium Technical Publication. https://medium.com/@gudiya6277/getting-started-with-eclipse-ditto-using-docker-c61a7bae9ea0

[53] Mell, P., & Grance, T. (2011). The NIST definition of cloud computing. National Institute of Standards and Technology, Special Publication 800-145.

[54] Armbrust, M., Fox, A., Griffith, R., et al. (2010). A view of cloud computing. Communications of the ACM, 53(4), 50-58.

[55] Zhang, Q., Cheng, L., & Boutaba, R. (2010). Cloud computing: State-of-the-art and research challenges. Journal of Internet Services and Applications, 1(1), 7-18.

[56] Calheiros, R. N., Ranjan, R., Beloglazov, A., De Rose, C. A. F., & Buyya, R. (2011). CloudSim: A toolkit for modeling and simulation of cloud computing environments. Software: Practice and Experience, 41(1), 23-50.

[57] Lorido-Botran, T., Miguel-Alonso, J., & Lozano, J. A. (2014). A review of auto-scaling techniques for elastic applications in cloud environments. Journal of Grid Computing, 12(4), 559-592.

[58] Ghosh, R., Longo, F., Naik, V. K., & Trivedi, K. S. (2010). Modeling and performance analysis of large scale IaaS clouds. Future Generation Computer Systems, 29(5), 1216-1234.

[59] Buyya, R., Yeo, C. S., Venugopal, S., Broberg, J., & Brandic, I. (2009). Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. Future Generation Computer Systems, 25(6), 599-616.

[60] Emeakaroha, V. C., Netto, M. A. S., Calheiros, R. N., Brandic, I., Buyya, R., & De Rose, C. A. F. (2012). Towards autonomic detection of SLA violations in cloud infrastructures. Future Generation Computer Systems, 28(7), 1017-1029.

[61] United Nations. (2022). UN E-Government Survey 2022: The Future of Digital Government. United Nations Department of Economic and Social Affairs.

[62] Joshi, P. R., & Islam, S. (2018). E-government maturity model for sustainable e-government services from the perspective of developing countries. Sustainability, 10(6), 1882.

[63] Alrawabdeh, W. (2014). Internet and e-government: Barriers affecting successful implementation. International Journal of Emerging Technologies in Learning, 9(3), 7-11.

[64] Benmoussa, R., & Hamdini, M. (2020). Towards an efficient digital administration in Algeria: Challenges and perspectives. International Journal of E-Government Research, 16(2), 45-62.

[65] Meziani, N., & Benabdelhafid, A. (2021). E-government adoption challenges in Algeria: An empirical study of public services digitalization. Algerian Journal of Computer Science and ICT, 3(1), 1-18.

[66] Finnish Ministry of Finance. (2021). Government's Digitalisation Programme 2023-2027. Finnish Government Publications.

[67] Valtori. (2022). Finnish Government Cloud Services: Architecture and Operations Manual. State IT Service Centre of Finland.

[68] Aceto, G., Botta, A., de Donato, W., & Pescape, A. (2013). Cloud monitoring: A survey. Computer Networks, 57(9), 2093-2115.

[69] Chowdhury, N. M. K., Rahman, M. R., & Boutaba, R. (2009). Virtual network embedding with coordinated node and link mapping. IEEE INFOCOM 2009, 783-791.

[70] Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., & Ayyash, M. (2015). Internet of things: A survey on enabling technologies, protocols, and applications. IEEE Communications Surveys & Tutorials, 17(4), 2347-2376.

[71] Object Management Group (OMG), *Unified Modeling Language (UML) Specification, Version 2.5.1*, 2017. Available: https://www.omg.org/spec/UML/2.5.1/

[72] B. Dobing and J. Parsons, "How UML is Used," *Communications of the ACM*, vol. 49, no. 5, pp. 109–113, 2006. Available: https://doi.org/10.1145/1125944.1125949

[73] M. Bell, *UML Basics: The Sequence Diagram*, IBM DeveloperWorks, IBM Corporation. Available: https://www.ibm.com/docs/

[74] Object Management Group (OMG), *UML Class Diagram Specification*, UML Version 2.5.1, 2017. Available: https://www.omg.org/spec/UML/

[75] Alibaba Group, *Alibaba Cluster Trace Program (Cluster Trace 2018)*, 2018. Available: https://github.com/alibaba/clusterdata

[76] S. M. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017. Available: https://doi.org/10.48550/arXiv.1705.07874

[77] S. Gupta and D. A. Dinesh, "Resource Usage Prediction of Cloud Workloads Using Deep Bidirectional Long Short-Term Memory Networks," *IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, 2017. Available: https://doi.org/10.1109/ANTS.2017.8384098

[78] Y. Zhu, X. Wang, and Y. Liu, "A Novel Approach to Workload Prediction Using Attention-Based LSTM Encoder–Decoder Network in Cloud Environment," *EURASIP Journal on Wireless Communications and Networking*, 2019. Available: https://doi.org/10.1186/s13638-019-1605-z

[79] Bi, J., Ma, H., Yuan, H., & Zhang, J. (2023). Accurate Prediction of Workloads and Resources With Multi-Head Attention and Hybrid LSTM for Cloud Data Centers. IEEE Transactions on Sustainable Computing. **Availbale:** https://s2.smu.edu/~jiazhang/Papers/JiaZhang-Multihead.pdf

[80] Patel, Y. S. et al. (2023). MAG-D: A Multivariate Attention and Gated Recurrent Unit Based Deep Learning Approach for Cloud Workload Forecasting in Data Centers. Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC9855517/