University of Padua

Department of Information Engineering

Master Degree in ICT for Internet and Multimedia

Leveraging cloud and machine learning technologies for the development of a knowledge IOT database

Master thesis

Relator

Prof. Lorenzo Vangelista

Master Candidate

Alessandro Discalzi

ID 2088235

Academic Year 2023-2024

Summary

This document describes the work done during the 750-hours final project at 221e S.r.l. The project's goal is to architect and develop a cloud-based system capable of ingesting and processing data from heterogeneous IoT sensors so that a knowledge database can be built.
The system must be designed to be scalable and fault-tolerant, and it must be platform-agnostic.
This document is going to describe the company, the idea behind the project, the work done and an assessment of what I developed and learned during my internship.

"If the past is just dust
Then the future could be our dream"

— Lorna Shore

Acknowledgements

Prof. Lorenzo Vangelista, my thesis supervisor, deserves my deepest gratitude for his exceptional support and guidance throughout the completion of this research.
My family, for their encouragement and understanding throughout this academic endeavour, has my heartfelt thanks.
I am truly grateful to Luca Perosa, Bledar Gogaj, Marco Lionello, and all my peers at SCAI ITEC, for their unwavering support when I made the decision to pursue a Master's degree.
I extend my sincere appreciation to PhD. Roberto Bortoletto, my company tutor, and all my colleagues in 221e for their invaluable support and guidance throughout my final project.
Last but not least, I want to give a shoutout to all my friends for having my back and just being there through thick and thin. Your friendship means a lot to me, and I appreciate the support and good times we've shared.

Padova, October 2024 Alessandro Discalzi

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 The Company

221e S.r.l.[1] is an Italian company based in Abano Terme (PD). The company, founded in 2012, is a leading supplier of IoT solution providing a wide range of products, both hardware and software, for industrial and wereable applications. The great experience and know how of the company's team, allows to provide clients with the best hardware solution for their needs, which can be enhached via the company's software platform or via a custom solution.

Figure 1.1: 221e's logo

1.2 Idea

The idea behind the project is to create a new cloud-based software platform for the company's IoT devices. The platform needs to ingest data from a variety of IoT devices, process it and create a knowledge database that can be used to provide insights and predictions to the end user.

1.3 Thesis outline

The second chapter describes the high level requirements.

The fourth chapter describes how the solution has been implemented.

The fifth chapter assess the results of the project.

[1] *221e S.r.l.* URL: `https://www.221e.com/`.

The last chapter describes the conclusion of the project and the future work.

Chapter 2

Objectives and requirements

2.1 Data collection

2.2 Security

2.3 Cloud infrastructure

2.4 Scalability

Chapter 3

Technologies

In this chapter it's reported the study made on the various technologies taken into account to develop the project.

3.1 Cloud

This section describes the services offered by Amazon Web Services[1] and Microsoft Azure[2], exploring their features, Pros: and Cons:. Services features are better in the respective offical AWS[3] and Azure[4] documentation, while Pros: and Cons: are based on the author's and the company experience as well as other users reviews that can be found in TrustRadius[5] website.
These services are the most used cloud services in the world and they offer a wide range of services that can be used to develop the project. It's also important to mention that these two providers were chosen right away because of the already developed experience with them, both in the company and in the author of this thesis.

3.1.1 Amazon Web Services

Batch

AWS Batch is a fully managed service that enables developers to easily and efficiently run thousands of batch and machine learning computing jobs on AWS.
Pros:

- Fully managed
- Scalable
- Cost effective
- Supports different batch processing scenarios
- Supports machine learning

[1] *Amazon Web Services.* URL: `https://aws.amazon.com/`.
[2] *Microsoft Azure.* URL: `https://azure.microsoft.com/`.
[3] *AWS Documentation.* URL: `https://docs.aws.amazon.com/`.
[4] *Azure Documentation.* URL: `https://docs.microsoft.com/en-us/azure/`.
[5] *Trust Radius.* URL: `https://trustradius.com/`.

- Easy to use
- Versatile

Cons:

- Not well documented

DynamoDB

AWS DynamoDB is a fully managed NoSQL database service that provides fast and predictable performance with seamless scalability. Tables can store and retrieve virtually any amount of data, serving any level of request traffic. It automatically spreads the data and traffic for the table over a sufficient number of servers to handle the request capacity specified by the customer and the amount of data stored, while maintaining Cons:istent and fast performance.
Pros:

- Fully managed
- Fast and predictable performance
- Scalable
- Highly available
- NoSQL

Cons:

- Hard to make changes against bulks of records
- Need to know at prior which queries will be made

Elastic map reduce (EMR)

AWS EMR is a big data platform that simplifies the deployment and management of big data frameworks, like Apache Hadoop and Apache Spark, on AWS.
Pros:

- Fully managed
- Scalable
- Petabyte scale data processing
- Easy resources provisioning
- Reconfigurable

Cons:

- Complexity
- Costly

Glue

AWS Glue is a fully managed ETL service that enables efficient data integration on a large scale.
Pros:

- Fully managed
- Pay per use
- Scalable
- Provides a centralize metadata repository
- Supports different data sources and formats
- Can automatically discover and catalog data from various sources
- Allow for job scheduling
- Data encryption

Cons:

- Costly for high workloads
- Performance issues with large datasets
- Complexity

Greengrass

AWS Greengrass is an open source edge runtime and cloud service used to build, deploy, and manage device software. It enables the devices to process the data locally, while still using the cloud for management, analytics, and durable storage.
It also enables encryption at rest and in transit and it can also extend device functionality with AWS Lambda functions.
Pros:

- Edge computing
- Encryption at rest and in transit
- Extend device functionality with AWS Lambda functions
- ML models deployment

Cons:

- Restrained to AWS services
- Not platform agnostic
- Resource intensive for small devices
- Need a connection for the initial setup

IoT Core

AWS IoT Core is a fully managed cloud service that lets connected devices easily and securely interact with cloud applications and other devices. It is composed of multiple services like Device Management, Device Defender, Device Advisor, and IoT Analytics and only some of them can be used during the development.
Pros:

- Composed of multiple services so only the necessary ones can be used
- Encription at rest and in transit
- Supports MQTT, HTTP, and WebSockets
- Allows for device management
- Allows for machine learning at edge
- Can trigger events thanks to custom rules

Cons:

- Not platform agnostic if installed on devices
- Lacks of integration for some devices

Kendra

AWS Kendra is a fully managed enterprise search service that allows developers to add search capabilities across various content repositories leveraging on built in connectors.
Pros:

- Fully managed
- Scalable
- Supports multiple data sources
- Easy to use and set up
- Accurate search results

Cons:

- Costly

Kinesis Data Firehose

AWS Kinesis Data Firehose is a fully managed service that simplifies the process of capturing, transforming and loading streaming data. It acts as an ETL service that can capture, transform, and load streaming data into a variety of AWS services. Additionaly it can transform raw data in column oriented data formats like Apache Parquet[6]
Pros:

- Fully managed

[6] *Apache Parquet.* URL: `https://parquet.apache.org/`.

- Can read data from IoT core and Kinesis Data Streams
- Scalable
- Can transform data
- Can load data into different AWS services
- Supports batching based on time or size

Cons:

- Not always cost effective
- Limited transformation capabilities
- Does not support batching based on more complex rules

Kinesis Data Streams

AWS Kinesis Data Stream is a fully managed service that simplify the capture, processing and loading of streaming data in real time at any scale thus enabling real-time data analytics with ease.
Pros:

- Fully managed
- Scalable
- Real-time and fast data processing
- Keeps data for 24 hours by default

Cons:

- Not always cost effective
- Limited data retention
- Limited data transformation
- Not useful for certain batch processing scenarios

Lake Formation

AWS Lake Formation is a fully managed service that simplifies the creation, security and management of data lakes. It allows for cleaning and transforming the data using machine learning.
Pros:

- Fully managed
- Scalable
- Secure
- Simplifies lake creation
- Simplifies ingestion management

- Simplifies permission management
- Provides data auditing
- Supports machine learning
- Supports data cataloging

Cons:

- Complexity
- Costly
- Not native support for all data sources

Lambda

AWS Lambda is an event driven serverless compute service that automatically manages the underlying compute resources. AWS Lambda can be used to extend other AWS services with custom logic, and to create new back-end services that can operate at AWS scale, performance, and security.
Pros:

- Fully managed
- Serverless
- Pay per use
- Scalable
- Easy to integrate with other AWS services
- Supports multiple programming languages
- Easy to deploy and maintain
- Can run parallel executions
- Low time to market
- Supports custom libraries

Cons:

- Limited execution time (15 mins)
- Limited memory
- Limited environment variables
- Maximum 1000 concurrent executions
- Cold start
- Not cost effective for high workloads

Managed Service for Apache Flink (MSF)

AWS MSF is a fully managed service that simplifies the creation and the execution of real time application using Apache Flink[7] .
Pros:

- Fully managed
- Scalable
- Supports batch and stream processing
- Real-time processing
- Large-scale data processing

Cons:

- Complexity

Managed Streaming for Kafka (MSK)

AWS MSK is a fully managed service that simplifies the setup, the scaling and the management of Apache Kafka[8] clusters.
Pros:

- Fully managed
- Scalable
- Cost effective
- Secure
- High availability
- Easy to integrate with other AWS services

Cons:

- Local testing challenges: hard to replicate the same environment in production and locally
- Not suitable for high traffic scenarios
- Complexity

Sage Maker

AWS Sage maker is a cloud-based machine learning platform that allows developer to build, train and deploy machine learning models.
Pros:

- Fully managed
- Scalable

[7] *Apache Flink*. URL: `https://flink.apache.org/`.
[8] *Apache Kafka*. URL: `https://kafka.apache.org/`.

- Supports multiple machine learning frameworks
- Supports multiple programming languages
- Allow for easy model deployment

Cons:

- Cannot schedule training jobs
- Costly for high workloads

Simple Storage Service (S3)

AWS S3 is an object storage service offering scalability, data availability, security, and performance. With S3, any amount of data can be stored and retrieved from anywhere on the web. Data is stored as objects in buckets, with each object representing a file and its metadata.
Pros:

- Scalable
- Highly available
- Secure
- Durable
- Cost effective
- No bucket size limit
- No limit to the number of objects that can be stored in a bucket
- Has different storage classes to fit frequent access, infrequent access, and long-term storage

Cons:

- Not suitable for small files
- Object size limit (5TB)
- Maximum 100 buckets per account
- Max 5GB per file upload via PUT operation

3.1.2 Microsoft Azure

Blob Storage

Azure Blob Storage is a fully managed object storage service that is highly scalable and available. It can store large amounts of unstructured data, making it suitable for a wide range of workloads.
Pros:

- Fully managed

- Scalable
- Highly available
- Secure
- Cost effective
- No limit to the number of objects that can be stored in a container
- Has different storage tiers to fit frequent access, infrequent access, and long-term storage
- Different storage options (Blob, archive, queue, file and disk)

Cons:

- Not suitable for small files
- Object size limit (4TB)
- Maximum 2PB per account in US and Europe
- Maximum 500TB per account in other regions

Cosmos DB

Azure Cosmos DB is a fully managed NO-SQL database service supporting multiple data models. It supports multiple NoSQL databases like PostgreSQL, MongoDB, and Cassandra.
Pros:

- Fully managed
- Scalable
- Multi model
- Global distribution
- Cons:istency levels based on the application needs
- Easy to use

Cons:

- Expensive
- Slow for complex queries

DataBricks

Azure Databricks is a fully managed Apache Spark-based analytics platform supporting a variety of libraries and languages.
Pros:

- Fully managed
- Scalable
- Supports multiple programming languages (Python, R, Scala, SQL, Java)
- Supports multiple libraries (Tensorflow, PyTorch, Scikit-learn, etc.)
- Open data lakehouse

Cons:

- Costly
- Complexity
- Hard to configure

Data Explorer

Azure data explorer is a fully managed, real-time and high volume data analytics service. It offers speed and low latency, being able to get quick insights from raw data.
Pros:

- Fully managed
- Scalable
- Real-time data processing
- Low latency
- Supports multiple data sources
- Supports structured, semi-structured and unstructured data
- Fast data ingestion
- Can use batch processing

Cons:

- Complexity
- Costly
- Limited capabilities for data transformation
- Hard configuration

Data Factory

Azure data factory is a fully managed cloud-based data integration service. It provides tools to orchestrate data workflows while monitoring executions. **Pros:**

- Fully managed
- Scalable
- Can perfom data Analytics using Synapse

Cons:

- Complexity
- Limited transformation capabilities

Data Lake Storage

Azure Data Lake Storage is a secure and scalable data lake platform. It provides a single place to store structured and unstructured data, making it easy to perform big data analytics.
Pros:

- Fully managed
- Scalable
- Secure
- Cost effective
- Compatible with Hadoop
- Supports Python for data analytics

Cons:

- Data governance challenges

Event Hubs

Azure Event Hubs is a fully managed, real-time data ingestion service that is simple, secure, and scalable. It can be used to stream millions of events per second with low latency, from any source to any destination. It offers also native support for Apache Kafka, allowing user to run existing Kafka applications.
Pros:

- Fully managed
- Scalable
- Secure
- Low latency
- Supports Apache Kafka

- Schema registry: centralize repository for schema management
- Real time data processing

Cons:

- Costly
- Complexity
- Limitation in event storage
- Cons:umers need to manage their state of processing

Functions

Azure functions is a serverless compute service that enables developers tu run code in response to events without the need to manage the infrastructure.
Pros:

- Fully managed
- Pay per use
- Scalable
- Supports multiple programming languages
- Easy to deploy and maintain
- Can run parallel executions
- Low time to market
- Supports custom libraries

Cons:

- Limited execution time (10 mins)
- Cold start
- Not cost effective for high workloads

IoT Hub

Azure IoT Hub is a cloud service that serves as the bridge between IoT devices and solutions in the cloud, facilitating reliable and secure communication. It can handle and manage a large number of devices making it suitable both for small-scale and enterprise-level solutions.
Pros:

- Secure
- Supports MQTT, AMQP, and HTTP
- Allows for device management
- Allows for machine learning at edge

- Can trigger events thanks to custom rules
- Can extend device functionality with Azure Functions

Cons:

- Not platform agnostic if installed on devices
- Lacks of integration for some devices
- Not well documented
- Costly

Machine Learning

Azure Machine Learning is a fully managed service that allows developers to build, train, and deploy machine learning models.
Pros:

- Fully managed
- Scalable
- Supports multiple machine learning frameworks
- Supports multiple programming languages
- Allow for easy model deployment
- Cost effective
- Has MLOps capabilities
- Pay as you go

Cons:

- Cost raises when training big models
- Hard to optimize

3.2 Present Solutions

In this section are presented the solutions that are currently available on the market.

3.2.1 Alleantia IoT Edge Hub

IoT Edge Hub is Alleantia[9]'s plug and play solution for the industrial IoT. It offers a wide range of features like device management, alarms and events, log management and report generation. It also supports the integration with Microsoft Azure[10].
Pros:

- Plug and play

[9] *Alleantia.* URL: `https://www.alleantia.com/`.
[10] *Microsoft Azure.*

- Device management
- Alarms and events
- Log management
- Report generation
- Integration with Microsoft Azure

Cons:

- Not platform agnostic
- Does not support Amazon Web Services[11]

3.2.2 Eclipse Kura

Eclipse Kura[12] is an open source IoT Edge Framework that serves as a platform for building IoT gateways. It's based on Java/OSGi and it provides API access to the hardware interfaces of IoT Gateways.
Pros:

- Open source
- Platform agnostic
- Allows for flexible and modular development
- API access to hardware interfaces
- Introduces AI capabilities at the edge

Cons:

- Computational complexity for small devices
- Not well documented
- No native support for cloud services

3.2.3 Eurotech Everyware Cloud

Eurotech[13] Everyware Cloud is a cloud-based IoT Integration Platform with a microservices architecture that allows to connect, configure and manage IoT gateways and devices.
Pros:

- Cloud-based
- Microservices architecture
- Allows to connect, configure and manage IoT gateways and devices

[11] *Amazon Web Services.*
[12] *Eclipse Kura.* URL: `https://eclipse.dev/kura/`.
[13] *Eurotech.* URL: `https://www.eurotech.com/`.

- Supports multiple protocols
- Supports multiple cloud services

Cons:

- Last update in 2019

3.2.4 STMicrelectronics X-Cube Cloud

STMicrelectronics[14] X-Cube Cloud is a software package that enables the connection of STM32 microcontrollers to the cloud.
Pros:

- Supports multiple cloud services
- Offers generic and secure connection to the cloud
- Supports multiple protocols

Cons:

- Specific for STM32 microcontrollers
- Specific packages for each cloud service if you want to use the full potential
- The generic solution only runs on a subset of STM32 microcontrollers

3.3 Machine Learning at edge

This section describes the technologies that can be used to build and deploy machine learning models at the edge and the Federated Learning approach.

3.3.1 Tensorflow Lite

3.3.2 Tiny Engine

3.3.3 Federated Learning and Transfer Learning

Federated Learning is a machine learning approach that trains an algorithm across multiple decentralized edge devices or servers holding local data samples, without exchanging them. This approach allows for privacy preservation and data security, as well as for the possibility to train models on devices with low computational power as described in "EdgeFed: Optimized Federated Learning Based on Edge Computing"[15]. Another important approach that has been taken into account is Transfer Learning, a technique that transfer the knowledge of a model trained on a specific task to a new task, improving the performance of the model and reducing the time and resources needed to train it, as described in "Federated learning for IoT devices: Enhancing TinyML with on-board training"[16].
Pros:

[14] **site:st**.

[15] Yunfan Ye et al. "EdgeFed: Optimized Federated Learning Based on Edge Computing". In: *IEEE Access* 8 (2020), pp. 209191–209198. DOI: 10.1109/ACCESS.2020.3038287.

[16] M. Ficco et al. "Federated learning for IoT devices: Enhancing TinyML with on-board training". In: *Information Fusion* 104 (2024), p. 102189. ISSN: 1566-2535. DOI: https://doi.org/10.1016/j.inffus.2023.102189. URL: https://www.sciencedirect.com/science/article/pii/S1566253523005055.

- Privacy preservation
- Data security
- No need for data centralization
- Low computational power needed
- Low latency

Cons:

- Complex to implement
- Limited capabilities

Chapter 4

Methodology

Introduction

4.1 Data Collection

4.2 Architecture

Chapter 5

Results

Chapter intro

5.1 Tests

Chapter 6

Conclusion

6.1 Objectives achieved

6.2 Future developments

6.3 What I learned

6.4 Final considerations

Chapter 7

Bibliography

Article references

Ficco, M. et al. "Federated learning for IoT devices: Enhancing TinyML with on-board training". In: *Information Fusion* 104 (2024), p. 102189. ISSN: 1566-2535. DOI: https://doi.org/10.1016/j.inffus.2023.102189. URL: https://www.sciencedirect.com/science/article/pii/S1566253523005055 (cit. on p. 19).

Ye, Yunfan et al. "EdgeFed: Optimized Federated Learning Based on Edge Computing". In: *IEEE Access* 8 (2020), pp. 209191–209198. DOI: 10.1109/ACCESS.2020.3038287 (cit. on p. 19).

Website references

221e S.r.l. URL: https://www.221e.com/ (cit. on p. 1).

Alleantia. URL: https://www.alleantia.com/ (cit. on p. 17).

Amazon Web Services. URL: https://aws.amazon.com/ (cit. on pp. 5, 18).

Apache Flink. URL: https://flink.apache.org/ (cit. on p. 11).

Apache Kafka. URL: https://kafka.apache.org/ (cit. on p. 11).

Apache Parquet. URL: https://parquet.apache.org/ (cit. on p. 8).

AWS Documentation. URL: https://docs.aws.amazon.com/ (cit. on p. 5).

Azure Documentation. URL: https://docs.microsoft.com/en-us/azure/ (cit. on p. 5).

Eclipse Kura. URL: https://eclipse.dev/kura/ (cit. on p. 18).

Eurotech. URL: https://www.eurotech.com/ (cit. on p. 18).

Microsoft Azure. URL: https://azure.microsoft.com/ (cit. on pp. 5, 17).

STMicroelectronics. URL: https://www.st.com/content/st_com/en.html.

Trust Radius. URL: https://trustradius.com/ (cit. on p. 5).