



University of Manouba
National School of Computer Science



INTERNSHIP REPORT OF ENTERPRISE IMMERSION

Enhancing recommendation systems based on analysis of user's behavior

Host organisation :



Author:

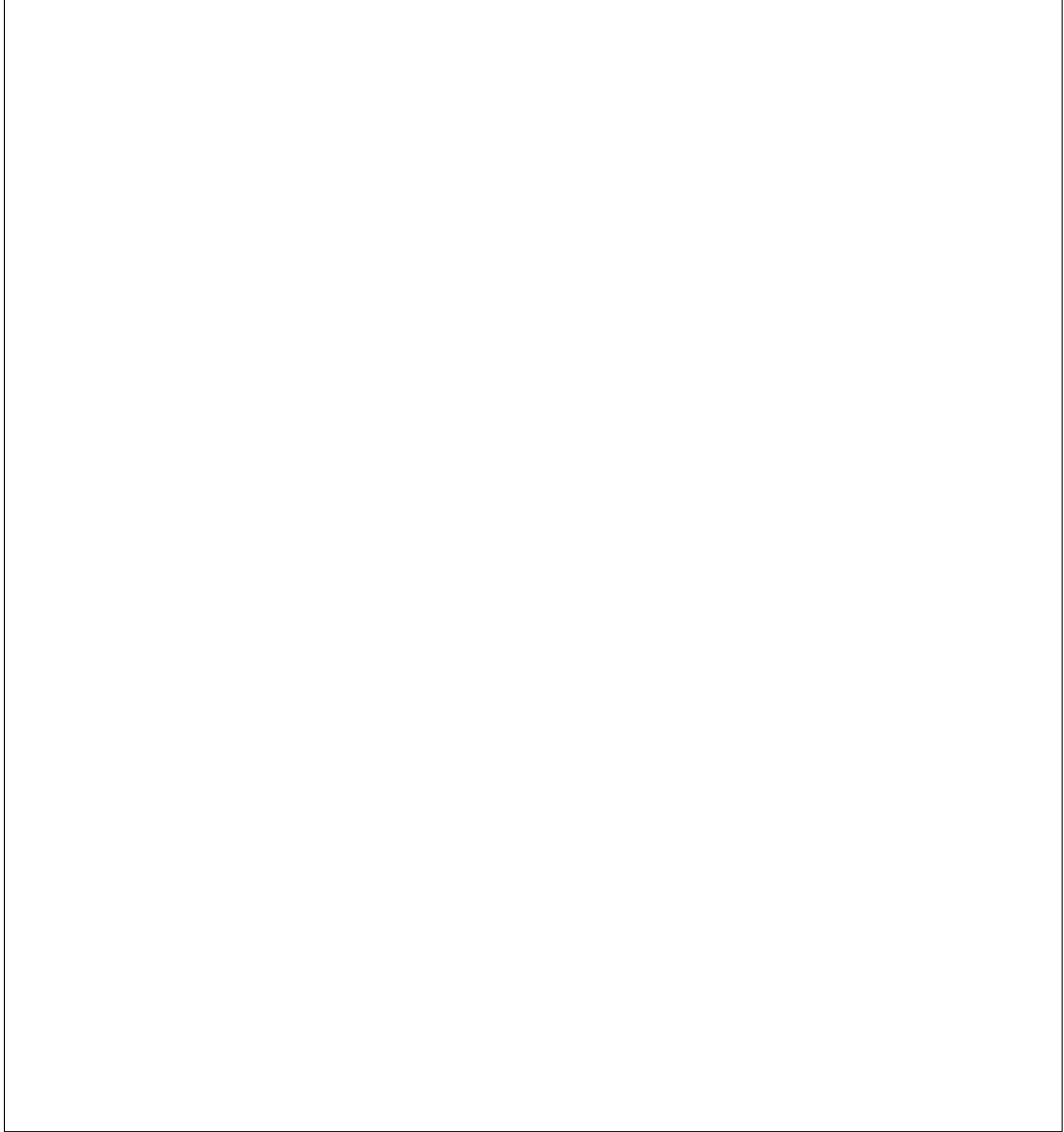
Mrs. Maryem BEN ALI

Professional supervisor :

Mr. Ahmed BEN ALI

Academic year : 2022 / 2023

Supervisor's feedback and signature

A large, empty rectangular box with a thin black border, intended for a supervisor to provide feedback and a signature. The box is currently blank.

Acknowledgements

I am pleased to reserve this page to express my gratitude and deep appreciation to all those who provided essential assistance for the successful completion of this work.

First and foremost, I extend my sincere gratitude to my professional supervisor, **Mr. Ahmed Ben Ali**, for his unwavering support, guidance, and invaluable insights throughout my internship.

I also wish to express my sincere thanks to the **Ip-Label team**, for their collaboration, mentorship, and trust in my ability to contribute effectively to the project.

Lastly, I extend my deep respect to **the jury members** for their time, expertise, and commitment in evaluating and examining my work.

Contents

Introduction	1
1 Navigating Research Horizons	2
1.1 Introduction	2
1.2 General Context	2
1.2.1 Host organisation	2
1.2.2 Overview on Real User Monitoring (RUM)	3
1.2.3 Academic context	4
1.3 Research Area Identification and Alignment with Company Vision	5
1.3.1 Research Area Identification	5
1.3.2 Alignment with Company Vision	7
1.4 Conclusion	8
2 Project context	9
2.1 Introduction	9
2.2 Problem statement	9
2.3 Objective	9
2.4 Data	10
2.5 Requirements analysis and specification	11
2.5.1 Identifying actors	11
2.5.2 Requirements analysis	11
2.5.3 Requirements specification	12
2.6 Conclusion	13
3 Crafting Insights: Work Undertaken and Challenges Faced	14
3.1 Introduction	14
3.2 Work environment	14
3.2.1 Hardware environment	14
3.2.2 Software environment	15
3.3 Data collection	15
3.4 Data understanding	16
3.4.1 Database	16
3.4.2 Log Files	16

3.5	Data preparation	18
3.5.1	Data cleaning	19
3.5.2	Feature engineering	19
3.6	Challenges faced	22
3.7	Conclusion	22
4	Forward Momentum: Proposing Strategies for Future Success	23
4.1	Introduction	23
4.2	The Potential of Data Lake	23
4.2.1	Exploring Data Lake Layers	23
4.3	Conclusion	27
	Conclusion	27
	Bibiliography	28

List of Figures

1.1	Ip-Label logo	2
1.2	E-kara logo	3
1.3	Evolution of real user monitoring over the time	5
2.1	Example of a received log file	10
2.2	Sample Snapshot of the userlog Database	11
2.3	Use case diagram	12
2.4	Sequence diagram	13
3.1	An example of a log file type API-Restit-RUM-Web-error	21
3.2	An example of a Json file containing information on feature 1 extracted from the example log file	21
4.1	Data lake structure	24
4.2	Microsoft Azure logo	26
4.3	Amazon web services logo	27

List of Tables

3.1	Hardware Environment	14
-----	--------------------------------	----

General Introduction

In the dynamic and ever-evolving landscape of modern digital experiences, the fundamental importance of achieving seamless application performance has risen to an indisputable status. As individuals, businesses, and entire industries harness the capabilities of cutting-edge technology to enhance efficiency, connectivity, and accessibility, any lag, glitch, or delay in performance can have far-reaching consequences, impacting not only productivity and profitability but also user satisfaction, trust, and engagement. Therefore, the quest for flawless application performance has evolved from a mere technical requirement into a critical business and user experience imperative.

This context, shaped by our reliance on digital solutions, serves as the foundation for a remarkable and collaborative 8-week internship project hosted within the innovative confines of ip-label's esteemed Research and Development (R&D) Department. The project's core ambition was to elevate the company's Real User Monitoring (RUM) solution undertaken with a visionary perspective aimed at the next five decades.

This document encapsulates the entire journey of the internship project. It is structured into three main chapters, each addressing a distinct phase of the endeavor.

The first chapter presents the host organisation, the academic context of the project as well as the journey that led to choose the internship subject.

The second chapter details the project context, its problem statement, and objectives. It also showcases the acquired data and the requirements analysis.

The third chapter provides an in-depth account of the work accomplished during this phase, while also highlighting the challenges encountered.

Finally, the final chapter concludes the narrative by presenting a proposed approach to address the challenges identified throughout the project.

Chapter 1

Navigating Research Horizons

1.1 Introduction

In the upcoming section, we explore the process of aligning our project with the broader vision of "ip-label." We'll start by introducing the host organization, followed by an overview of real user monitoring. Then, we'll introduce into the academic context of the project. Lastly, we'll discuss the process of Research Area Identification and Alignment with Company Vision.

1.2 General Context

1.2.1 Host organisation

ip-label, founded in 2001 by Philippe Borfiga and Pierre Montcel, stands as a global leader in the domain of Real User Monitoring (RUM) solutions and performance optimization. The organization's dedication to driving superior digital experiences has solidified its position as a trailblazing force in the field, with a significant impact on the global stage.



Figure 1.1: Ip-Label logo



Figure 1.2: E-kara logo

Mission: At ip-label, the unwavering mission is to empower organizations across the world to deliver unparalleled digital experiences. Recognizing that the key to success lies in user satisfaction, ip-label is committed to equipping businesses with the tools and insights needed to achieve flawless application performance. By merging cutting-edge technology with an innovative outlook, ip-label spearheads a more interconnected and responsive digital era.

Key Offerings: Central to its offerings is e-kara, an industry-disrupting Real User Monitoring (RUM) solution that has revolutionized the landscape of application performance. With e-kara, organizations gain an unparalleled window into user interactions, enabling them to proactively address issues and significantly enhance user satisfaction.

e-kara empowers businesses with real-time visibility into user interactions, facilitating on-the-fly optimization of application performance. Placing the user at the core, e-kara enables organizations to tailor strategies to user preferences and requirements. Harnessing advanced analytics, e-kara forecasts potential bottlenecks and concerns, allowing preemptive measures for uninterrupted user experiences. From web applications to mobile devices, e-kara supports monitoring across diverse platforms, ensuring consistent performance throughout the digital ecosystem.

Global Impact: With a substantial global footprint, ip-label's solutions have left a mark on the digital landscape. The success story is punctuated by impressive numbers. It has a presence in several major cities, including Paris, Madrid, Shanghai and Tunis resonating with businesses worldwide. The client base spans diverse industries, comprising more than 400 clients who benefit from tailored solutions. ip-label's solutions have enhanced digital experiences for millions of users globally. With over a decade of dedicated service, ip-label has amassed a wealth of expertise and insights, making it a pioneer in the industry.

1.2.2 Overview on Real User Monitoring (RUM)

In the realm of digital experiences, the invisible intricacies that shape user interactions often remain unexplored. Real User Monitoring (RUM) emerges as an illuminating technology, enabling us to pierce through this veil of obscurity and gain profound insights into the actual user experience. At its core, RUM is a passive approach that captures real-time data about user interactions with websites and applications, providing a comprehensive view into performance and uncovering hidden opportunities for optimization.

Defining RUM: Real User Monitoring (RUM) is a sophisticated methodology that facilitates the passive collection of data regarding how users interact with digital platforms. This encompasses a wide array of actions, from initial page load times to the navigation between different sections. RUM operates silently, akin to an attentive observer, meticulously tracking every click, scroll, and pause without disrupting the user's natural flow.

The Inner Workings: At the heart of RUM lies its capacity to capture critical performance metrics without intruding on the user's experience. As users navigate through websites and apps, RUM discreetly measures parameters such as response times, page rendering, and interaction delays. This Delicate data forms a Comprehensive representation of the user's journey, offering valuable insights into the challenges they face and the quality of the provided experience.

From Data to Optimization: RUM doesn't simply provide raw data; it transforms it into actionable insights. By analyzing the collected information, we unveil potential pitfalls that hinder optimal user experiences. Whether it's a sluggish page load, a non-responsive button, or an error-prone process, RUM exposes these issues, allowing us to strategically refine the platform for heightened user satisfaction.

Predicting and Elevating Experiences: RUM's predictive prowess sets it apart. By tracking trends and patterns over time, it foresees potential bottlenecks and discrepancies before they manifest. This foresight empowers us to proactively address performance challenges and maintain a seamless user experience, mitigating the negative impact on user satisfaction and business success.

Towards the Horizon: As the digital landscape evolves, RUM continues to evolve with it. Cloud-based and Software as a Service (SaaS) implementations amplify its potential, offering scalability and deeper insights into global user behavior. The fusion of RUM with artificial intelligence and machine learning promises to refine its capabilities further, enabling even more accurate predictions and proactive optimizations.

In Conclusion: In today's digital world, where user experiences are incredibly important, RUM acts like a guiding light. It dispels any confusion and gives us the insights we need to make digital interactions smooth and effortless. By using RUM, we match what users expect, making sure that every click, tap, and action works perfectly, providing the best possible experience.

1.2.3 Academic context

This project is a component of the summer internship program designed for second-year students at the National School of Computer Science. It is conducted in partnership with the company "Ip-Label" over a span of 8 weeks, starting on June 12th. The project is under the guidance and supervision of Mr. Ahmed Ben Ali.

1.3. RESEARCH AREA IDENTIFICATION AND ALIGNMENT WITH COMPANY VISION

1.3 Research Area Identification and Alignment with Company Vision

This section explores the initial phase of our project journey, highlighting the process of identifying potential research areas and aligning them effortlessly with the overarching vision of "Ip-Label". This groundwork was pivotal in shaping the direction of our internship, ultimately leading us to select a subject that held significant promise and relevance within the field of Real User Monitoring .

1.3.1 Research Area Identification

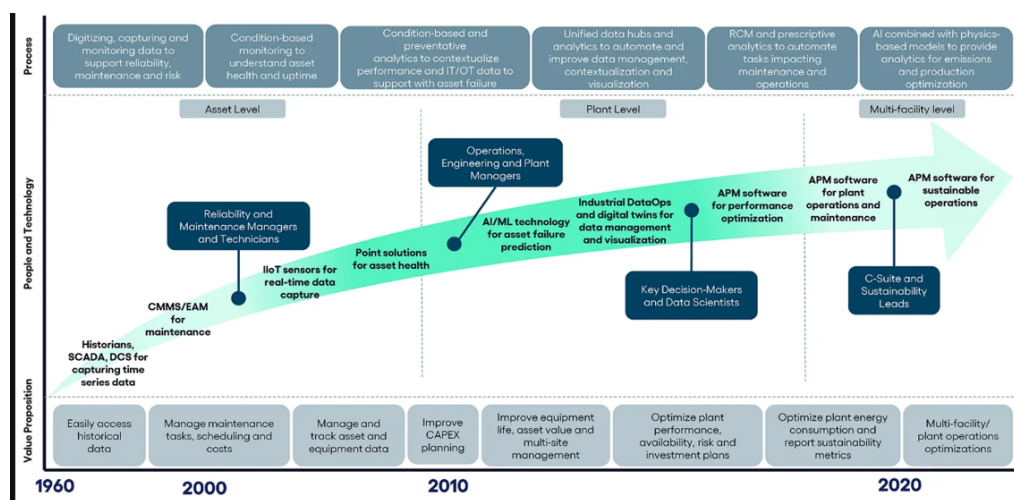


Figure 1.3: Evolution of real user monitoring over the time

My approach to envisioning the future of RUM involved a forward-thinking perspective. I recognized that to forecast the future of RUM, we must first anticipate the future of applications and the metrics that will drive data collection. By doing so, we can effectively evaluate and enhance application performance in a synoptic manner.

In the course of our research, I engaged in a constructive survey, employing a Google Form to gather insights and ideas from peers and potential users of RUM solutions. This invaluable input enriched our understanding and infused practical perspectives into our exploration.

The Future of Applications

- **Neuro-Interface Applications:** Brain-Computer Interfaces (BCIs) will pave the way for applications controlled by the power of thought, providing unprecedented levels of accessibility and functionality.

1.3. RESEARCH AREA IDENTIFICATION AND ALIGNMENT WITH COMPANY VISION

- **Emotionally Intelligent Applications:** Applications will be equipped with Emotional AI, enabling them to perceive and respond to users' emotions, leading to more personalized and empathetic experiences.
- **Sustainable Development:** Applications will increasingly focus on sustainability, emphasizing eco-friendly practices and reducing environmental impact. This approach aligns with global efforts towards the balance of economic, social, and environmental concerns .
- **AR/VR/MR Technologies:** Augmented Reality (AR), Virtual Reality (VR), and Mixed Reality (MR) applications will become mainstream, transforming how we interact with digital content and the physical world.
- **Cloud-Based Apps:** Cloud-native applications will dominate, providing scalable, flexible, and cost-effective solutions for various industries and user needs.
- **Applications Fully Developed by AI:** AI-powered application development will be more prevalent, where AI systems autonomously create and optimize applications.

To keep up with these future advancements in applications, RUM solutions will need to evolve and adapt. Here are the new features that can be added to RUM solutions to meet the demands of the future:

Possible enhancements in RUM Solutions:

- **Neuro-Interface RUM:** Develop specialized RUM modules to capture, interpret, and analyze real-time brainwave data from Neuro-Interface applications, gaining insights into user mental states during interactions. Utilize AI algorithms to correlate brainwave patterns with application performance, enabling personalized adjustments based on the user's cognitive responses.
- **Emotion Monitoring and User Sentiment Analysis:** Integrate sentiment analysis and emotion recognition tools into RUM solutions, enabling real-time assessment of user emotions and reactions during application interactions.
- **Sustainability Metrics:** Enhance RUM solutions to include sustainability metrics, allowing businesses to assess and optimize the environmental impact of their applications.
- **AR/VR/MR Performance Metrics:** Enhance RUM capabilities to capture and analyze specific performance metrics unique to AR, VR, and MR applications, ensuring smooth experiences in these immersive environments.
- **Cloud Performance Optimization:** Implement cloud-specific monitoring to analyze application performance across various cloud providers, enabling businesses to

1.3. RESEARCH AREA IDENTIFICATION AND ALIGNMENT WITH COMPANY VISION

optimize performance and cost-efficiency. RUM solutions can also facilitate application migration to the cloud, analyzing compatibility, monitoring performance during and after migration, and providing recommendations to optimize configurations in the cloud environment.

- **Auto-Recovery for RUM Applications:** Develop auto-recovery mechanisms within RUM solutions to proactively detect and rectify performance issues, ensuring uninterrupted user experiences. This feature will contribute to higher reliability and reduced downtime.

1.3.2 Alignment with Company Vision

In this section, we focus on aligning our research with the company’s vision by selecting an idea from the explored areas of interest. This ensures that our chosen project harmonizes with the organization’s strategic goals.

Possible generated ideas are as follows:

- **Brain-Computer Interface Integration:** Investigate the integration of brain-computer interface technology with RUM, leveraging brain signals for user experience analysis.
- **Emotion Recognition in RUM:** Explore the incorporation of emotion recognition for real-time monitoring and analysis of user emotions during application usage.
- **Intelligent Alert Prioritization:** Implement intelligent algorithms within the RUM solution to prioritize and classify alerts based on their severity and impact on the user experience.
- **Monitoring Ethical Compliance:** Develop a RUM solution equipped with capabilities to monitor and identify potential violations of ethics, cultural norms, or religious beliefs specific to different contexts.
- **Monitoring Child-Friendly Content:** Implement mechanisms within the RUM solution to ensure that content and applications intended for children adhere to appropriate standards and guidelines.
- **Wise Energy Consumption Evaluation:** Devise a rating system to assess application energy efficiency, supporting sustainable practices and resource optimization.
- **Sustainable Cloud Resource Allocation:** Integrate the RUM solution with Cloud infrastructure providers and develop algorithms to optimize the allocation of Cloud resources based on sustainability criteria.
- **Real-time Carbon Emissions Dashboard:** Develop a dashboard that presents real-time data on carbon emissions alongside application performance indicators.

1.4. CONCLUSION

As ideas were evaluated, it became clear that some concepts were too specific or divergent from ip-label's client base. The brain-computer interface idea was eliminated due to its limited applicability in the current client ecosystem. Similarly, the child-friendly content monitoring and real-time carbon emissions dashboard concepts were deemed too specialized. The intelligent alert system faced the challenge of customization for each client's unique requirements.

The final choice was made: Emotion Recognition and Analysis in RUM, which later evolved into the concept of a Recommendation System designed to enhance user experiences based on the analysis of their behavior. As we delved into the project, it became evident that extracting different client patterns would be crucial for diverse recommendation scenarios. These patterns form the bedrock for predicting users' subsequent actions, suggesting relevant features or content, and even tailoring a user interface that aligns with their unique profile.

The process of extracting and analyzing these patterns from user interactions and behaviors is at the heart of our endeavor. By doing so, we are committed to elevating the overall user experience and providing personalized recommendations that cater to individual preferences and needs.

1.4 Conclusion

In this chapter, we've laid the foundation for our project's direction. We introduced "ip-label and" explored Real User Monitoring. Through Research Area Identification and Alignment with Company Vision, we selected the Recommendation System, driven by behavior analysis, as our focal point. This chapter inaugurates our project work, poised to revolutionize digital interactions through advanced technology. As we conclude, we look forward to diving into the practical implementation in the next chapter.

Chapter 2

Project context

2.1 Introduction

Within this chapter, we will explore the multifaceted context surrounding our project. We will dig into the problem statement, objectives, and the valuable data at our disposal. Additionally, we will perform an in-depth analysis of the project's requirements, identifying both functional and non-functional aspects that shape our development path.

2.2 Problem statement

In the dynamic landscape of research and development, challenges often present themselves as opportunities for innovation. Within this context, the R&D Department at "Ip-Label" recognized a unique avenue for advancement, improving their Real User Monitoring (RUM) solution. The pursuit of a forward-looking, futuristic product prompted the inception of this internship project. Unlike conventional problem statements, this project emerges as a proactive endeavor to anticipate and address potential areas of enhancement. Rather than grappling with issues, the focus lies on envisioning the next phase of RUM technology.

2.3 Objective

The objective of this project is to design and implement an advanced machine learning model capable of extracting diverse client patterns from user interactions within the Real User Monitoring (RUM) solution. These patterns will form the foundation for developing distinct recommendation systems that will be utilized in subsequent steps.

These recommendation systems will serve various purposes, including suggesting relevant features and content, as well as anticipating and suggesting the next move of the user and tailoring user interfaces to align with individual profiles .

2.4. DATA

Through comprehensive analysis of user interactions and behavior, our primary aim is to enhance the overall user experience by delivering personalized recommendations that cater to unique preferences and needs.

2.4 Data

In the data understanding phase, our preliminary dataset encompasses 661 log files, each uniquely labeled with titles such as "API-Restit-RUM-Web-error__2023-06-08_00-00-00" and "DFY-active-results-api-out__2023-06-08_00-00-00." These log files document specific events and actions tied to diverse APIs, services, and components within the application ecosystem. The filenames, often indicative of the logged content's nature, bear timestamps reflecting the moment of log entry recording.

```
[-----[info][2023-06-12T00:00:29.759Z] at [/tools/error.js:20:18]-----]
Error: timeout of 5000ms exceeded
    at createError (/opt/iplabel/dfy/apps/dfy-active-results-api/node_modules/axios/lib/core/createError.js:16:15)
    at RedirectableRequest.handleRequestTimeout (/opt/iplabel/dfy/apps/dfy-active-results-api/node_modules/axios/lib/adapters/http.js:280:16)
    at Object.onceWrapper (node:events:509:28)
    at RedirectableRequest.emit (node:events:390:28)
    at RedirectableRequest.emit (node:domain:475:12)
    at Timeout._onTimeout (/opt/iplabel/dfy/apps/dfy-active-results-api/node_modules/follow-redirects/index.js:166:13)
    at listOnTimeout (node:internal/timers:557:17)
    at processTimers (node:internal/timers:500:7)

[-----[info][2023-06-12T00:00:33.727Z] at [/tools/error.js:20:18]-----]
Error: timeout of 5000ms exceeded
    at createError (/opt/iplabel/dfy/apps/dfy-active-results-api/node_modules/axios/lib/core/createError.js:16:15)
    at RedirectableRequest.handleRequestTimeout (/opt/iplabel/dfy/apps/dfy-active-results-api/node_modules/axios/lib/adapters/http.js:280:16)
    at Object.onceWrapper (node:events:509:28)
    at RedirectableRequest.emit (node:events:390:28)
    at RedirectableRequest.emit (node:domain:475:12)
    at Timeout._onTimeout (/opt/iplabel/dfy/apps/dfy-active-results-api/node_modules/follow-redirects/index.js:166:13)
    at listOnTimeout (node:internal/timers:557:17)
    at processTimers (node:internal/timers:500:7)

[-----[info][2023-06-12T07:18:25.131Z] at [/tools/error.js:20:18]-----]
error: select json_agg(DISTINCT (elem.value -> 's_measurementId')) AS "metrics", json_agg(DISTINCT jsonb_build_object('index', rsl_stepid::integer + 1, 'name', rsl_st
    at Parser.parseErrorMessage (/opt/iplabel/dfy/apps/dfy-active-results-api/node_modules/pg-protocol/dist/parser.js:287:98)
    at Parser.handlePacket (/opt/iplabel/dfy/apps/dfy-active-results-api/node_modules/pg-protocol/dist/parser.js:126:29)
    at Parser.parse (/opt/iplabel/dfy/apps/dfy-active-results-api/node_modules/pg-protocol/dist/parser.js:39:38)
    at TLSSocket.<anonymous> (/opt/iplabel/dfy/apps/dfy-active-results-api/node_modules/pg-protocol/dist/index.js:11:42)
    at TLSSocket.emit (node:events:390:28)
    at TLSSocket.emit (node:domain:475:12)
    at addChunk (node:internal/streams/readable:315:12)
    at readableAddChunk (node:internal/streams/readable:289:9)
    at TLSSocket.Readable.push (node:internal/streams/readable:228:10)
    at TLSSocket.onStreamRead (node:internal/stream_base_commons:199:23)
```

Figure 2.1: Example of a received log file

Additionally, we've received a database containing essential information about user interactions. This database documents a comprehensive range of actions performed through the application by both human and robot users.

2.5. REQUIREMENTS ANALYSIS AND SPECIFICATION

cli_id	usr_id	ual_objecttype	ual_objectname	ual_source	ual_action	ual_timestamp	ual_content	ual_objectid
1592	164040	auth	Ekara	api	login	2023-06-02 00:15:53.451		164040
1592	164040	auth	Ekara	api	login	2023-06-02 00:15:55.278		164040
1592	164040	auth	Ekara	api	login	2023-06-02 00:19:21.786		164040
1592	164040	auth	Ekara	api	login	2023-06-02 00:20:08.493		164040
1592	164040	auth	Ekara	api	login	2023-06-02 00:20:24.064		164040
1592	164040	auth	Ekara	api	login	2023-06-02 00:25:33.328		164040
1592	164040	auth	Ekara	api	login	2023-06-02 00:25:38.044		164040
1592	164040	auth	Ekara	api	login	2023-06-02 00:27:17.611		164040
1592	164040	auth	Ekara	api	login	2023-06-02 00:27:26.990		164040
1592	164040	auth	Ekara	api	login	2023-06-02 00:28:39.618		164040
1592	164040	auth	Ekara	api	login	2023-06-02 00:28:43.609		164040
1592	164040	auth	Ekara	api	login	2023-06-02 00:30:15.941		164040

Figure 2.2: Sample Snapshot of the userlog Database

This data resource, coupled with the log files, holds immense promise for driving insights into user behavior and engagement patterns, laying the foundation for our subsequent analytical and machine learning endeavors.

2.5 Requirements analysis and specification

In this section, we will present and analyze in detail both functional and non-functional requirement.

2.5.1 Identifying actors

The primary actor in this project, the "**Data Team**" at "Ip-Label", collaborates closely to orchestrate data collection, processing, analysis, and the development of recommendation systems.

2.5.2 Requirements analysis

Functional requirements

The implemented model will enable its user to analyze and classify distinct client patterns from collected data.

Non-functional requirements

- **Scalability:** The model should efficiently handle increasing amounts of user data while maintaining performance.
- **Privacy and Security:** User data must be securely stored and processed to prevent unauthorized access.
- **Real-time Responsiveness:** The model should provide quick recommendations in response to user interactions.

2.5.3 Requirements specification

2.5.3.1 Use case diagram

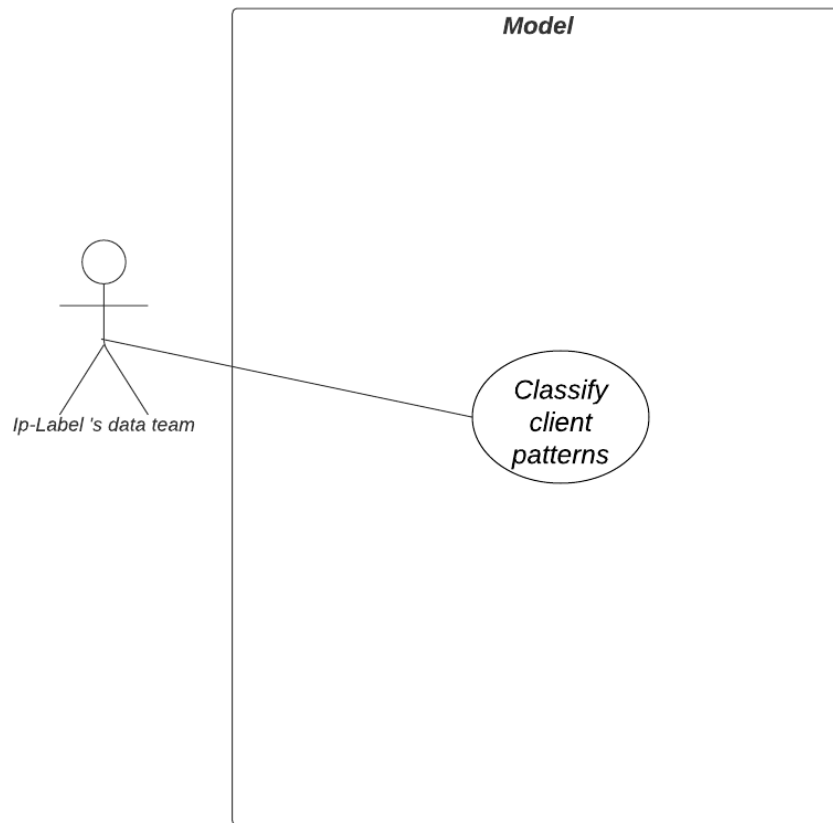


Figure 2.3: Use case diagram

2.6. CONCLUSION

2.5.3.2 Sequence diagram

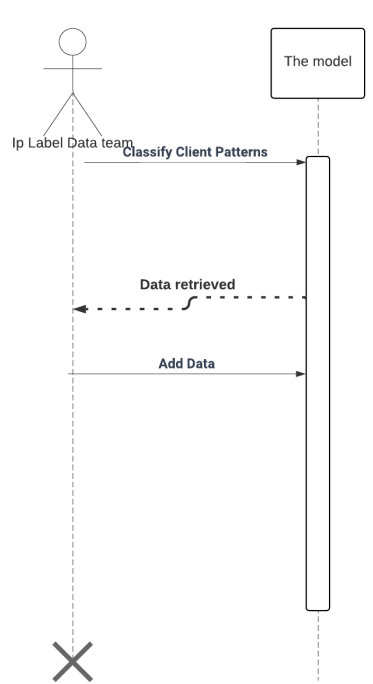


Figure 2.4: Sequence diagram

2.6 Conclusion

This chapter's journey has illuminated the various facets of our project's context. From understanding the problem statement and setting clear objectives to exploring the richness of the available data, we have laid a robust foundation. Furthermore, the comprehensive requirements analysis showcases our commitment to ensuring a scalable, secure, and responsive solution. In the upcoming chapter, we will transition from planning to the implementation phase of the project.

Chapter 3

Crafting Insights: Work Undertaken and Challenges Faced

3.1 Introduction

In this chapter, we explore the practical aspects of our project, shedding light on the work environment, hardware and software setups, data collection methods, and the detailed steps taken for data preparation.

3.2 Work environment

In this section, we will present the hardware and software environments that were used for the research and development of this project as well as the documentation phase. We will explain in detail the libraries that were used and the reason for using them.

3.2.1 Hardware environment

The hardware environment used for data collection and preparation consisted of a computer with the following specifications:

Table 3.1: Hardware Environment

Component	Specifications
Processeur	Intel Core i7 7500U
RAM	8 GB
Stockage	512 GB SSD
Processeur graphique	Nvidia 920 MX
Système d'exploitation	Windows 10

3.2.2 Software environment

The software environment utilized for this project includes a combination of powerful tools and libraries that support the development of data-driven applications:

- **Python:** Python, an interpreted high-level programming language, has emerged as one of the most popular languages in data science. Its versatility and extensive library ecosystem make it a natural choice for our project. Python’s adaptability aligns with the diverse domains we address, while its updated libraries provide cutting-edge solutions.
- **Anaconda:** Anaconda, an open-source distribution for Python and R, facilitates the management of virtual environments and libraries. Particularly beneficial for data scientists, Anaconda comes with pre-installed libraries such as Numpy, Pandas, TensorFlow, and scikit-learn.
- **Jupyter Notebook:** Jupyter Notebook, an open-source web application, enables the creation and sharing of interactive documents containing code, data visualizations, and explanatory text. Its user-friendly interface enhances the documentation and communication of findings.
- **MySQL Workbench:** MySQL Workbench provides a comprehensive visual tool for database design, development, and administration. Its intuitive interface supports the management and exploration of data stored in databases.

Additionally, the project employs several libraries and frameworks to facilitate various tasks within the software environment:

- **Natural Language Toolkit (NLTK):** The NLTK library is used for natural language processing tasks. It provides functionalities for text preprocessing, including tokenization, stop-word removal, and stemming. These features enable the analysis of textual data in a structured manner.
- **JSON Library:** The JSON library allows the project to handle data in JSON format efficiently. It supports the serialization and deserialization of JSON data, which is widely used for data interchange.

3.3 Data collection

The data collection phase was approached with a strategic outlook, aiming to gather insights from both user log files and a complementary database to ensure a comprehensive understanding of user behavior.

To commence the process, an extensive collection of user log files was undertaken, spanning

3.4. DATA UNDERSTANDING

the timeframe from June 16th to July 16th.

Simultaneously, an additional layer of context was sought through a database dump. This supplementary dataset enables a more profound understanding of the relative significance of log files to their corresponding users and actions.

This thoughtful approach to data collection underscores our commitment to harnessing a diverse range of information sources. By strategically combining user log files and the contextual database, we aimed to extract a holistic view of user behavior, setting the stage for subsequent analysis and modeling.

3.4 Data understanding

3.4.1 Database

The database is a foundational component in our data understanding phase, offering a comprehensive view of user interactions within the application. This relational dataset contains a total of 9,777 lines and nine essential columns, as described below:

- **cli_id:** Unique identifier for clients.
- **usr_id:** Unique identifier for users.
- **ual_objecttype:** Type of object associated with the action (e.g., "auth").
- **ual_objectname:** Object's name linked to the action (e.g., "Ekara").
- **ual_source:** Source of the action (e.g., "api", "ui").
- **ual_action:** Specific action performed (e.g., "login").
- **ual_timestamp:** Date and time of the action.
- **ual_content:** Additional content related to the action.
- **ual_objectid:** Identifier for the object involved in the action.

3.4.2 Log Files

During the data understanding phase, a crucial dataset under analysis is comprised of various log files, each containing valuable insights into user interactions and application activities. These log files serve as a fundamental source for capturing user behaviors and uncovering potential anomalies within the system. The log files are organized based on their specific purposes and service activities, each differentiated by its name and timestamp.

3.4. DATA UNDERSTANDING

The log files are named in accordance with the services they correspond to, encapsulating both errors and diverse log entries.

Here is an overview of the different log files we have gathered:

- **API-Restit-RUM-Web-error:** This file, present in 31 instances, captures error-related information involving Restitution and Web interactions.
- **API-Restit-RUM-Web-out:** Comprising 30 log files, this collection similarly pertains to Restitution and Web interactions but focuses on log entries rather than errors.
- **DFY-active-results-api-error:** With 30 instances, this file documents errors associated with Active Results API activities.
- **DFY-active-results-api-out:** Consisting of 30 log files, this group provides log entries pertaining to Active Results API interactions.
- **DFY-administration-api-error:** This collection, with 30 instances, records errors linked to Administration API activities.
- **DFY-administration-api-out:** Similarly, containing 30 log files, this group offers log entries related to Administration API interactions.
- **DFY-Auth-API-error:** These log files, numbering 30, highlight errors arising from Auth API interactions.
- **DFY-Auth-API-out:** This collection of 30 log files presents log entries associated with Auth API activities.
- **DFY-Bo-RumWeb-API-error:** Involving 30 instances, these log files cover errors connected to Back Office and Restitution Web API interactions.
- **DFY-Bo-RumWeb-API-out:** This group of 30 log files contains log entries for Back Office and Restitution Web API activities.
- **DFYConfigService-error:** With 30 instances, this collection records errors concerning the Config Service.
- **DFYConfigService-out:** Just like the previous group, this collection of 30 log files captures log entries associated with the Config Service.
- **DFY-Script-API-error:** This set of 30 log files records errors pertaining to Script API activities.
- **DFY-Script-API-out:** Comprising 30 log files, this collection captures log entries related to Script API interactions.

- **DFYWebSocketService-error:** Although this collection consists of 30 log files documenting WebSocket Service errors, the content focuses on internal network operations, and hence was excluded from our analysis.
- **DFYWebSocketService-out:** This group of 30 log files contains log entries associated with WebSocket Service activities.
- **pm2-logrotate-out:** This set of log files includes entries related to log rotation and management processes, offering insights into the operational aspects of the system.

These various log files encapsulate a plethora of insights into user interactions and system activities. Throughout our analysis, we will examine the content of these log files to uncover patterns, anomalies, and valuable information that will contribute to the enhancement of our recommendation system.

After a thorough analysis of the provided log files, our focus will be directed towards log file groups that contain information particularly relevant to our project. Specifically, we will concentrate our efforts on the log file groups:

- **API-Restit-RUM-Web-error**
- **DFY-active-results-api-error**
- **DFY-active-results-api-out**
- **DFY-administration-api-error**
- **DFY-administration-api-out**
- **DFY-Auth-API-error**

These log file groups offer insights into user interactions, system activities, and error occurrences that directly align with the goals and objectives of our recommendation system project.

3.5 Data preparation

In the data preparation phase, we laid the groundwork for building an effective recommendation system by carefully selecting features and extracting relevant information from the log files. The goal was to create a foundation for generating metrics that would later aid in identifying valuable client patterns. The data preparation process encompassed the following key steps:

3.5.1 Data cleaning

Throughout the data collection and preparation process, data cleaning played a pivotal role in ensuring the quality and reliability of our dataset. This vital step was a recurring routine that we diligently performed whenever we encountered unclear or inconsistent data.

3.5.2 Feature engineering

Extracted Features

After conducting a comprehensive analysis of the provided log files, our attention will be directed to individual log files, each of which will be followed by the extracted features particularly relevant to our project. We will concentrate our efforts on the following log file groups:

- **API-Restit-RUM-Web-error** log files:
 - **Feature 1:** Device Types - This feature captures different device types encountered in the logs, including "probe," "na," "smarttv," and others.
 - **Feature 2:** SQL Errors - This feature identifies SQL errors encountered within the logs.
- **DFY-active-results-api-error** log files:
 - **Feature 1:** "Not Enough Data" Errors - This feature indicates instances where the log files report a lack of sufficient data, potentially affecting the analysis and results.
 - **Feature 3:** Data Retrieval Failures - This feature points to cases where the system fails to retrieve the required data.
 - **Feature 4:** SQL Errors - This feature highlights SQL errors giving insights into database interactions.
- **DFY-active-results-api-out** log files:
 - **Feature 1:** SQL Errors - This feature identifies SQL errors within the logs, contributing to our understanding of database-related issues.
- **DFY-administration-api-error** log files:
 - **Feature 1:** Non-Accessible Client Errors - This feature captures instances where users attempt to access non-accessible clients due to constraints.
- **DFY-administration-api-out** log files:
 - **Feature 1:** Log Action Details - This feature includes log action IDs, timestamps, and updates, allowing us to generate metrics such as access frequency and usage patterns for the application.

3.5. DATA PREPARATION

- **Feature 2:** database updates
- **DFY-Auth-API-error** log files:
 - **Feature 1:** Authentication Errors - This feature captures authentication-related errors within SQL queries,.

Generated Features

Based on the extracted features, we further generated insights into user interaction, behavior, emotion, and application state. The following features were derived from the previously mentioned features:

- Time Between Error Occurrences - Calculated using error timestamps to understand the emotional or performance state of users or their devices.
- User Engagement Metrics - Derived from login and interaction timestamps to measure engagement levels and usage patterns.
- User Interaction Frequency - Analyzed from interaction timestamps to understand how often users interact with the application.
- User Access Frequency to Specific Features - Calculated from feature-specific interactions to gauge the popularity of different application sections.
- User Update Frequency - Derived from update timestamps to identify how often users modify their profiles or settings.

Feature Generation Process

The features were generated using Python and the Natural Language Toolkit (NLTK) library. The process involved extracting relevant information from log files and performing necessary calculations. Each generated piece of information was stored in JSON files associated with their respective features.

Finally, our intention was to employ a SQL database to correlate each JSON file with its corresponding user ID, setting the stage for the subsequent step in our data preparation pipeline. This next phase involves comprehensive Date and Time Engineering, alongside essential feature scaling and additional data filtering. However, in practice, we encountered challenges that impeded matching user IDs with the generated JSON files. This issue will be elaborated upon in the following paragraph.

3.5. DATA PREPARATION

```
[-----[error][2023-07-07T03:20:59.841Z] at [/routes/results/overview.js:125:18]-----]
ReduceWeeks fx - tracker 1064: Unknown device type probe

[-----[error][2023-07-07T03:20:59.842Z] at [/routes/results/overview.js:125:18]-----]
ReduceWeeks fx - tracker 1064: Unknown device type probe

[-----[error][2023-07-07T04:48:23.187Z] at [/routes/results/overview.js:125:18]-----]
ReduceWeeks fx - tracker 1167: Unknown device type na

[-----[error][2023-07-07T04:48:23.188Z] at [/routes/results/overview.js:125:18]-----]
ReduceWeeks fx - tracker 1167: Unknown device type na

[-----[error][2023-07-07T05:03:00.132Z] at [/routes/results/overview.js:125:18]-----]
ReduceWeeks fx - tracker 970: Unknown device type mplayer

[-----[error][2023-07-07T05:03:00.132Z] at [/routes/results/overview.js:125:18]-----]
ReduceWeeks fx - tracker 970: Unknown device type smarttv

[-----[error][2023-07-07T05:03:00.133Z] at [/routes/results/overview.js:125:18]-----]
ReduceWeeks fx - tracker 970: Unknown device type probe

[-----[error][2023-07-07T05:03:00.133Z] at [/routes/results/overview.js:125:18]-----]
ReduceWeeks fx - tracker 970: Unknown device type na

[-----[error][2023-07-07T05:03:00.134Z] at [/routes/results/overview.js:125:18]-----]
ReduceWeeks fx - tracker 970: Unknown device type probe
```

Figure 3.1: An example of a log file type API-Restit-RUM-Web-error

```
[
  {
    "date_time": "2023-06-07T01:00:55.767Z",
    "tracker_number": 1074,
    "device_type": "probe"
  },
  {
    "date_time": "2023-06-07T01:00:55.769Z",
    "tracker_number": 1074,
    "device_type": "na"
  },
  {
    "date_time": "2023-06-07T01:00:55.769Z",
    "tracker_number": 1074,
    "device_type": "probe"
  },
  {
    "date_time": "2023-06-07T02:47:02.684Z",
    "tracker_number": 1064,
    "device_type": "probe"
  },
  {
    "date_time": "2023-06-07T06:04:53.567Z",
    "tracker_number": 1189,
    "device_type": "na"
  },
]
```

Figure 3.2: An example of a Json file containing information on feature 1 extracted from the example log file

3.6 Challenges faced

Following an extensive examination of both the log files and the database, achieving the intended correlation between extracted feature information (represented as actions) and their respective user IDs turned out to be more complex than initially anticipated. Several significant challenges have arisen:

1. **Correlation Difficulties:** The primary challenge lies in establishing a reliable connection between the extracted feature information from log files and the corresponding user IDs within the database. Infact, Discrepancies have been identified between the timestamp records present in the log files and those within the `useractionlog` table.
2. **Automatic Log Purging:** Unfortunately, due to a recurring automated log purging mechanism, only logs from the most recent 45 days are retained. This temporal limitation severely constrains our ability to effectively retrieve and match historical data for analysis at the end of the internship period.
3. **Database Limitations:** Despite rigorous research, it has become evident that the current structure of the database system does not facilitate the correlation between user IDs and feature information due to inherent limitations. This unforeseen barrier impedes our progress in achieving a comprehensive and accurate integration of insights.

Given these complexities, it is imperative to explore alternative strategies for extracting and integrating the required data. Relying solely on the log files for this purpose seems unfeasible due to the aforementioned challenges. A strategic and innovative approach is essential to overcome these obstacles and ensure the successful integration of insights into our recommendation system. The forthcoming and final chapter will unveil the strategy we have explored to address these challenges and achieve our project's goals.

3.7 Conclusion

This chapter has provided a close look at the practical dimensions of our project. We explored the work environment, hardware and software setups, data collection strategies, and the essential process of data preparation. As we transition to the next chapter, we'll focus on the challenges encountered and the solutions devised.

Chapter 4

Forward Momentum: Proposing Strategies for Future Success

4.1 Introduction

Ip-Label's success hinges on a profound understanding of user interactions, underscoring the significance of projects like "Leveraging Recommendation Systems Based on User Interactions and Traces." Our project uncovered the challenges obstructing the realization of such endeavors. These challenges primarily revolve around data collection and the associated issues of data availability. A significant challenge was the intricate task of correlating user IDs with actions extracted from log files. In this chapter, we will explore potential solutions, specifically the establishment of a data lake within the company, and explore the various approaches to achieving this objective

4.2 The Potential of Data Lake

To address these challenges, we propose a solution: the establishment of a data lake within the organization. This strategic move aims to facilitate more efficient data usage and enable similar projects in the future.

4.2.1 Exploring Data Lake Layers

A data lake is a comprehensive storage repository designed to accommodate and manage extensive volumes of raw, unstructured, and structured data. This repository offers the versatility necessary for modern data management, enabling consistent storage, analysis, and utilization of diverse data types.

4.2. THE POTENTIAL OF DATA LAKE

4.2.1.1 Data lake structure

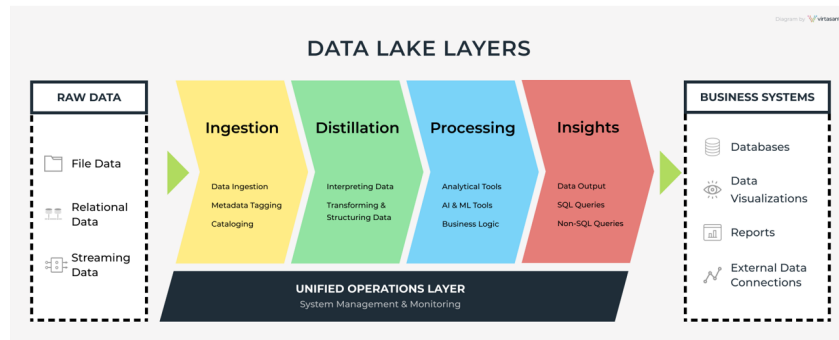


Figure 4.1: Data lake structure

- **Ingestion Layer:** In this layer, data from diverse sources is gathered and channeled into the data lake. Specifically, within our context, this phase marks the initial step where we capture user interactions, logs, and traces, ensuring their integration into the data lake for further processing and analysis.
- **Distillation Layer:** The distillation layer converts raw data into structured datasets suitable for analysis. It interprets, transforms, cleanses, denormalizes, and derives structured data from raw sources.
- **Processing Layer:** Structured data from the distillation layer undergoes processing through user queries and analytical tools. Business logic is applied, and data is consumed by analytical applications.
- **Insights Layer:** The insights layer serves as the output interface, allowing data to be queried to generate reports or dashboards. Insights are derived from structured data related to user interactions and behavior.
- **Unified Operations Layer:** This layer focuses on system monitoring, workflow management, auditing, and proficiency management, ensuring smooth data lake ecosystem functioning.

4.2.1.2 Key Considerations for Data Lake Architecture

Choosing the right data lake architecture requires careful consideration of various factors to ensure alignment with Ip-Label's goals, needs, and capabilities. Among these factors, we can name a few key ones, such as:

- **Data Types and Sources:** Determine the types of data and their sources, ensuring that the chosen architecture can effectively handle diverse data varieties.
- **Scalability and Performance:** Assess the architecture's scalability and performance to accommodate the expanding volume of data.

4.2. THE POTENTIAL OF DATA LAKE

- **Storage Requirements:** Consider whether the architecture provides sufficient storage capacity to meet the anticipated data storage needs.
- **Cost Considerations:** Evaluate the cost implications associated with the chosen data lake architecture.
- **Data Governance and Metadata Management:** Consider the architecture's support for data governance practices, metadata management, data lineage, and cataloging.
- **Integration with Existing Systems:** Analyze how the architecture integrates with the company's existing systems, applications, and tools.
- **Data Processing Capabilities:** Evaluate the architecture's ability to handle various data processing requirements.
- **Security and Compliance:** Ensure that the chosen architecture aligns with the company's security and compliance standards.

4.2.1.3 Exploring Data Lake Establishment

Upon thorough exploration of data lake architecture, it has become clear that the process is considerably complicated and time-consuming. Establishing an effective data lake solution demands a collaborative effort involving expertise from various teams, including development and data quality assurance. Given the complexity and scope of this task, it is evident that tackling it single-handedly is not feasible

Data Lake Architect: Strategic Approach

As the complexities of establishing a data lake become increasingly evident, it becomes indisputable that navigating this process requires a profound understanding of data architecture, integration, and operational workflows.

In this context, the significance of enlisting the expertise of an experienced data lake architect cannot be overstated. With their adeptness in both design and execution, these professionals play a pivotal role in crafting a data ecosystem that aligns perfectly with the organization's objectives. Their role extends beyond mere design, encompassing the orchestration of integration processes and the successful resolution of complexities.

In essence, a data lake architect brings the assurance of a well-structured and efficient system that optimizes data management and empowers strategic decision-making.

Cloud Data Platforms: A Streamlined Approach



Figure 4.2: Microsoft Azure logo

Cloud Data Platforms offer a simplified data management solution, reducing complexity and possibly align with the challenges of Ip-Label's solution. Here are examples of cloud data platforms offered by leading providers:

- **Microsoft Azure Cloud Data Platform:**

- **Azure Stream Analytics:** A real-time data ingestion solution that can handle data from various sources, Azure Stream Analytics enables processing of streaming data for timely insights.
- **Azure Event Hubs:** Designed for scalable event-driven data streaming, Azure Event Hubs is suitable for handling large amounts of incoming data streams.
- **Azure Data Factory:** This service supports data transformation and ETL processes, ensuring that data is refined and prepared for analysis.
- **Azure Data Lake Storage:** Designed for big data analytics, Azure Data Lake Storage offers hierarchical storage to accommodate the storage needs of a data lake environment.
- **Azure Synapse Analytics:** Combining data warehousing and big data analytics, Azure Synapse Analytics empowers integrated analytics and data exploration.
- **Azure Data Lake Analytics:** By offering on-demand data processing using U-SQL or .NET, Azure Data Lake Analytics enhances data processing capabilities.
- **Power BI:** With advanced data visualization and analytics, Power BI empowers users to create insightful reports and dashboards.
- **Azure Monitor:** Providing comprehensive monitoring and diagnostics, Azure Monitor ensures the health and performance of the data lake ecosystem.
- **Azure Resource Manager:** Enabling consistent resource management, Azure Resource Manager aids in automating and managing infrastructure provisioning.

- **Amazon Web Services (AWS) Cloud Data Platform:**

- **Amazon Kinesis:** A real-time data streaming solution tailored to accommodate the diverse data streams generated by IoT devices and applications. It enables efficient data ingestion for immediate processing and analysis.



Figure 4.3: Amazon web services logo

- **Amazon CloudWatch:** This service acts as a central hub for collecting and monitoring log data in real time, facilitating timely insights and enabling effective performance monitoring.
- **AWS Glue:** Equipped with ETL capabilities, AWS Glue supports automated schema inference and data cataloging, making it a powerful tool for distilling raw data into structured datasets suitable for analysis.
- **Amazon S3:** As a scalable storage solution, Amazon S3 is well-suited for storing various data types, both raw and transformed. Its versatility supports the diverse needs of a data lake ecosystem.
- **Amazon Redshift:** Designed for data warehousing, Amazon Redshift empowers analytical queries at scale, enabling efficient data processing and exploration.
- **Amazon Athena:** This service offers serverless SQL querying capabilities on data stored in Amazon S3, facilitating on-demand data exploration and analysis.
- **Amazon QuickSight:** With interactive dashboards and visualizations, Amazon QuickSight transforms data into actionable insights, aiding in informed decision-making.
- **AWS CloudWatch:** AWS CloudWatch extends beyond log monitoring to comprehensive resource and application monitoring, ensuring the smooth functioning of the data lake ecosystem.
- **AWS CloudFormation:** This service automates infrastructure provisioning, contributing to efficient and consistent resource management within the data lake environment.

4.3 Conclusion

In closing, this chapter provided a comprehensive overview of data lakes as a solution to the challenges encountered in data collection and integration. We explored their layered architecture and essential considerations for effective implementation. Additionally, we discussed two potential approaches: hiring a data lake architect and adopting streamlined cloud data platforms.

Conclusion

In the broader context of Real User Monitoring (RUM), this project emerges as a testament to the Research and Development Department's commitment to advancing technological frontiers. The driving force behind this initiative was the pursuit of innovation, aiming to enhance the RUM solution and pioneer new horizons in user experience optimization.

This journey led us through the intricate landscape of data collection, preparation, and analysis, where the complexities of integrating heterogeneous information sources were unveiled. As we delved into log files, database entries, and feature synthesis, a comprehensive understanding of user patterns and system dynamics emerged.

Central to this endeavor was the exploration of data lake solutions as a strategic response to the challenges encountered. This insight broadens the horizon for future data management, offering the potential to derive even deeper insights from the interplay of diverse data sources.

Throughout my internship, I had the privilege of gaining hands-on experience in real-world applications of machine learning, bridging the gap between theory and practice. This project exemplified the importance of holistic problem-solving, teamwork, and the dynamic nature of technological challenges.

As this chapter concludes, the door to future possibilities remains wide open. The insights gained and the solutions devised serve as foundations for continued explorations in recommendation systems, user experience enhancement, and strategic decision-making. The path forward involves refining models, optimizing processes, and embracing the potential of data lakes to unleash new realms of knowledge. The journey of innovation continues, shaping the evolution of technology at Ip-Label and beyond.

Bibliography

- [1] <https://www.chaossearch.io/blog/data-lake-architecture>
- [2] <https://static.googleusercontent.com/media/research.google.com/fr/pubs/archive/45530.pdf>
- [3] <https://www.researchgate.net/>
- [4] Source<https://www.baeldung.com/cs/amazon-recommendation-system#hybrid-approaches>
- [5] <https://jupyter.org/>
- [6] <https://www.cognite.com/en/blog/>
- [7] https://webpages.charlotte.edu/aatzache/Papers/2021_EmotionClassificationUsingRecurrentNeuralNetworkandScalablePatternMining.pdf
- [8] <https://estuary.dev/real-time-data-lake/>