Strategies for Avoiding Vendor Lock-In in Public Cloud Web Applications

A Practical Case Study

Sara Molina Delgado

Techniques of Software Application Development
Cloud Computing

Project supervisor
Raul Gomis Hidalgo
Coordinating professor
José Luis Soler

January 2025

Universitat Oberta de Catalunya

Index

List of figures

1. Introduction

1.1. Problem definition: what is vendor lock-in and what are their implications

This project has the aim to investigate and propose some guides that can help to avoid the problem of vendor lock-in in web applications that are hosted in the public cloud. Kumar et all defines the term vendor lock-in as “a scenario where the cost of switching to another vendor is mainly high and for this reason, the consumer is mainly struck with a single cloud vendor” [1].

When this situation appears in a project, it impacts in a lot of key areas of a company or a project:

- Economical: there is an obvious consequence of basing part of your architecture on third party technology, which is that they have a power on your business. This power is directly proportional to the criticality of the infrastructure that relies on the external services because the vendor can increase its prices without much capacity for reaction on the part of the affected company. There are other costs related to this situation like the salary of specialists in the vendor ecosystem, the formation in new features that can be released by the third company or for the new talent incorporations, the costs of a future migration if it is needed and the opportunity cost for paths that cannot be taken because of the limitations of the technology provided.

- Technological: the project or company are tied to the vendor solutions, and this will lead their technical path. New scenarios, new technical talent or new technologies are no longer possible to incorporate because there is a blocked due to the dependency on other company technologies. Contemplate these possibilities is directly related to the previous point, economical resources.

- Staff: the members of the team of a project must work with the technology of the project. It is always necessary to hire people that are specialized on it or to form people to acquire the knowledge. In some situations, this is especially difficult, when the technology that causes the problem is ancient and there are few people who domains or that are keen on learning it.

- Innovation: the fact of being trapped in a technology that is extremely expensive to migrate causes situations where the innovation is not possible. There can be new solutions in the market for the implicated company or project that cannot be implemented because it is completely coupled with the vendor solutions. Therefore, some opportunities to improve the company performance can be lost.

Undoubtedly, it generates a situation which is not desirable for any company. In fact, this is a problem which has been faced over the years and the consequences of not giving it the proper importance is still more than uncomfortable situation for a great number of corporations.

Vendor lock-in problem has never disappeared, however, nowadays it poses a threat that may be stronger than in the past because of the development of Cloud Computing. In the paper “Critical analysis of vendor lock-in and its impact on cloud computing migration: a business perspective” a survey is carried out and the results show that, for the 114 participants, as computing resources migrate from on-premises to the cloud, the vendor lock-in problem is exacerbated. According to the authors, their findings “exemplify the importance of interoperability, portability and standards in cloud computing [2].”

1.2. Context and motivation

1.2.1 The Cloud: definition, pros and cons

Cloud Computing can be defined as “a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction” [3].

The Cloud has certainly been a revolution and its impact in our lives is sometimes compared with the one the internet has in the past. Before the cloud, transfer a big content was impossible without a hardware device as an intermediary. Nowadays, gigabytes can be shared using temporal storage in the cloud and only a link is needed for the other person to recover the data. As it is also pointed out in *Cloud Computing Technology* book, years ago, if there should be a meeting of people of different geographies all the members invited should be had to move to an agreed location to attend. With the Cloud, all documents, schemas, data, voice and video can be shared across the world, and everyone remains at his location [4].

These are only some examples of how the cloud is changing our daily lives, but it has also had a great impact in enterprises all over the world. The cloud provides some characteristics that are very attractive to companies [5].

- On-demand self-service: the user of the service can automatically provision computing capabilities, storage or network systems without the mediation of any employer of the supplier company. This changes completely how the resources are supplied because, for example, you can have more storage in the database or a new computing resource in seconds without having to contact, negotiate, make a contract with any person or waiting for the delivering of new hardware.
- Broad network access: the resources provided are available through internet and they can be accessed from any device in any part of the world.
- Resource Pooling: the resources are assigned to a variety of clients dynamically. The clients can choose some characteristics of the physical space their data occupy, for example, the geography, but they do not know much about in which exact server it is placed. Also, generally they must share these resources with other clients and the location is flexible for all of them inside the same data centre. That allows that the clients can be benefited from economy of scale because the cloud providers can offer better prices that the

ones obtained in on-premises infrastructure. Moreover, it is more efficient because the infrastructure is assigned to the clients that needs it and freed when it is no longer needed avoiding an extra of infrastructure which is unused, and there are solutions for different data accesses needs.
- Rapid elasticity. Resources in the cloud can be expanded even automatically given some conditions, this implies a rapid reaction when an application need more resources.
- Measured service. The cloud system monitors all the services used. There are also tools that order and display the data letting the organizations to make decisions based on these reports.

However, there are also cons to all these benefits of the cloud:
- The user is the one that decides which services will be used, which implies that a very restrictive policy about resources provisioning must be stablished. It is a common situation, especially in a big company, that the resources demanded are grater that the one used. Then, incur in extra costs is a risk in a system deployed in the Cloud.
- The resources can be accessed through internet, with the cybersecurity risks and compliance challenges that it entails.
- Share the data and flexibility in the location also can have compliance problems. For example, some laws related to sensible data specifies strict requirements about the location of the information. That is the case of Esquema National de Seguridad (ENS) and Ley Orgárnica de Protección de Datos Personales (LODP) which stablish some measures related to data treatment and storage which should be considered before picking a Cloud infrastructure in some cases [6]. There are also other possible problems like data filtrations between clients that shares the physical space, even attacks from “neighbours” clients. Another challenge is the concentration of important clients in the same installations that, in case of attack, can suppose a heavy hit to multiple important companies and cause a great economic impact.
- Elasticity is usually based on automatic rules that provides more resources when they are needed, but they also increment the consumption and, thus, the invoice.

Therefore, it can be stablished that there are some the drawbacks that must been considered before creating or migrating an application to the Cloud environment. However, as it happened years ago with the mainframe, more and more companies are adopting Cloud infrastructure and Cloud computing is growing at a frantic pace.

And the cyphers shown it. Eurostat has developed a graphic that shows the increment of the companies that buys Cloud computing services in the EU:

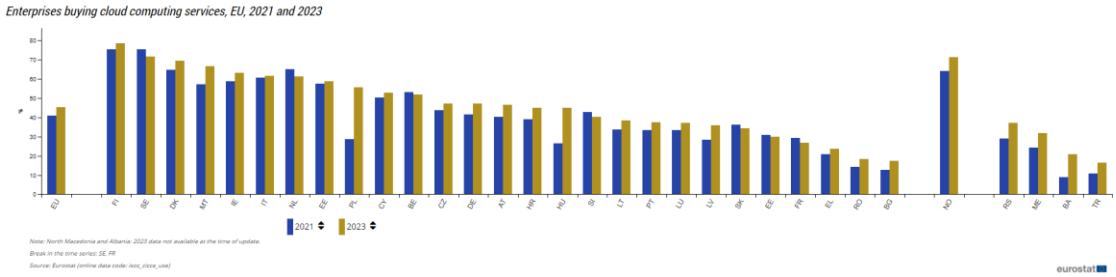


Figure 1: Enterprises buying cloud computing services in EU 2021 and 2023.

The main conclusion of this study is that 45.2 % of EU enterprises bought cloud computing services in 2023, mostly for e-mail, storage of files and office software and that the increment from 2021 has been a 4.2% [7].

From a Global point of view, according to Silicon newsletter, during the second trimester of 2024 the Cloud infrastructure spending was 78.200 million dollars all over the world. This means that the interannual increment hast been 19%. The 63% of the benefits has been registered by the main three cloud providers: Amazon AWS, Microsoft Azure and Google Cloud [8].

1.2.2 The adoption of the Cloud infrastructure for SMBs web applications

The Cloud has been a revolution also for the SMBs. The adoption of Cloud technologies is growing in multiple fields. SaaS solutions are one of the most popular cloud technologies between the small and medium companies, but also there are a growing interest in migrating or developing web applications in the Cloud infrastructure. According to the Flexera State of The Cloud Report of 2024, which is based on a survey of 753 cloud decision-makers and users from around the world, 56% of enterprises are outsourcing at least some public cloud work. Also, SMB usage of MSPs has increased to 36% from 26% YoY [9].

Regarding to the Cloud providers, there is currently a clear leader between the SMBs: AWS. For this reason, this will be the Cloud provider used for the prototype application.

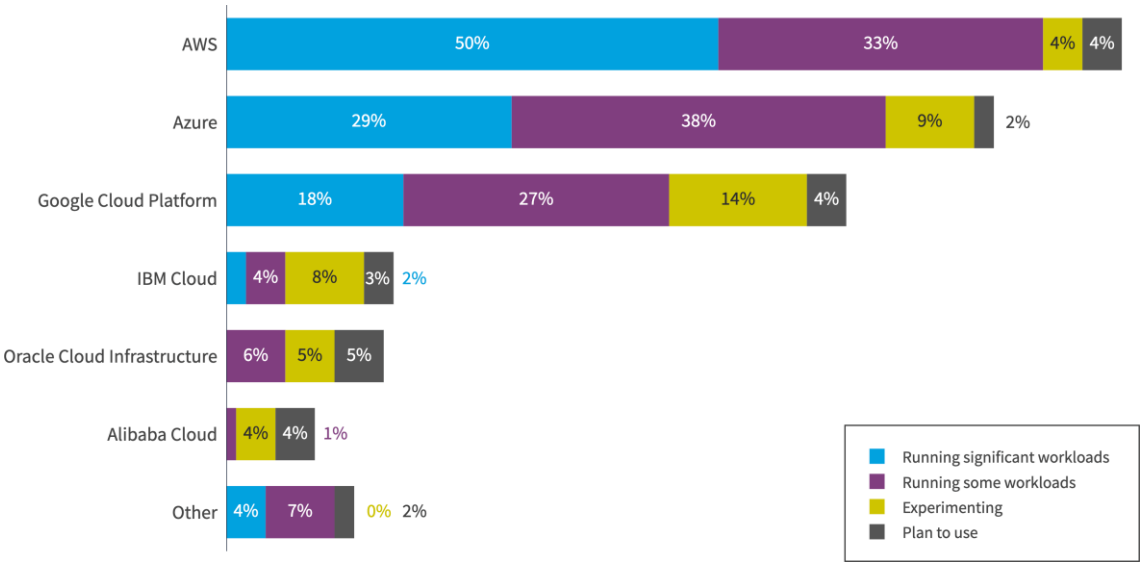


Figure 2: Cloud providers used by SMBs

In addition, the survey results point out that “Migrating more workloads to cloud” occupies the second position of the ranking of the Cloud initiatives for European organizations. It implies that that the topic of this project covers one of the most significant initiatives for the Europe companies regarding to the Cloud environment.

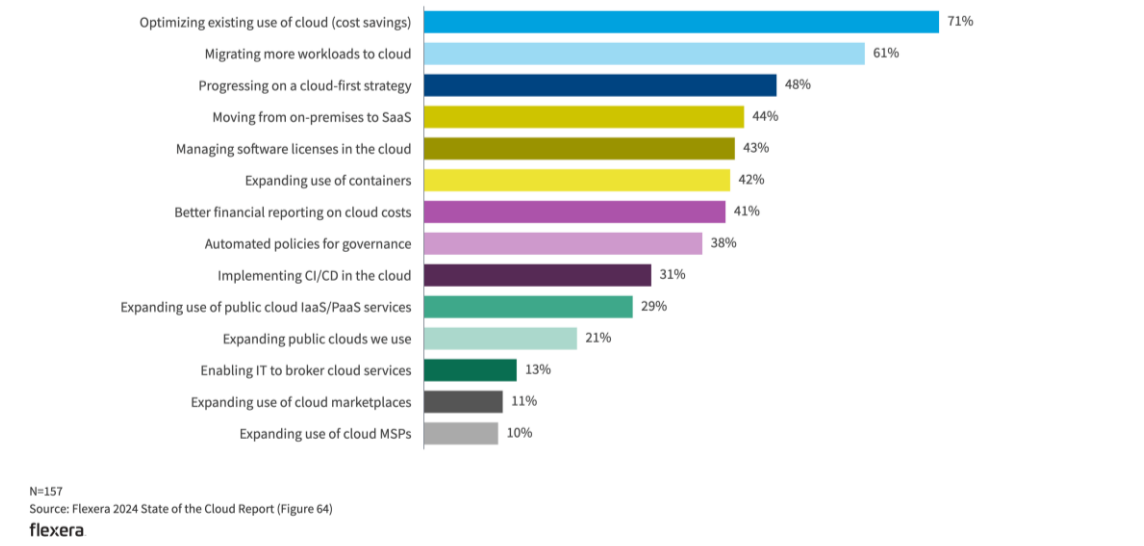


Figure 3: Cloud initiatives for European organizations

More concretely, in Spain, following the global tendency, the spending in Cloud solutions is continuously increasing. Channel Partner published a report in 2024 which stated that in the previous year in this country, 4.767 million of euros were destined to the Cloud services. The ciphers were expected to grow until 5.825 million in 2024, more than 7.000 million in 2025 and it is expected that the spend

will reach 8.500 million in 2026. Following the global tendency, SaaS congregates the main part of the expenditure; but PaaS and IaaS are the segments that are growing faster. This suggest that developing applications in the cloud is more common year by year [10].

1.2.3 Historical approach to vendor lock-in

Vendor lock-in is a problem that has historically tormented TIC professionals. One of the most famous cases, the Mainframe, has not been overcome yet for most banks, public organizations and insurance companies. Cobol provided a robust language that was massively used in this kind of industries that should deal with great amounts of data. This solution has been adopted and lines and lines of code has been written in COBOL and run in the mainframe machines from then on. Some companies bought the machines, another still rents the machines and the rest that are still trapped pay per use.

This is traduced in cyphers that may cause surprise nowadays, especially for a technology that this year has turned 60. According to IBM: “Two thirds of Fortune 100 companies, 45 of the world’s top 50 banks, 8 of the top 10 insurers, 7 of the top 10 global retailers, and 8 out of the top 10 telcos use IBM zSystems for mission critical workloads” [5]. This has marked the software development strategies from years, it has also positioned IBM as one of the most important companies in the TIC marked and it has also cost a huge amount to another businesses. The mainframe is only one example of this issue, but it is also very representative and convenient to show the motivations of this project.

Although the impressive figures that IBM reports about the mainframe, it is known that most of the companies “trapped” by IBM wants to abandon it. As in the 60s when Cobol appears, we are facing today a new technology that can be even more revolutionary: the Cloud. That is why some of them are looking to it as the alternative.

That is the case of Santander bank and its own new Platform Gravity. Gravity is the solution that Santander has found to modernize its core baking. Gravity will help Santander to migrate its core banking from the mainframe to the Cloud. That can be done because Gravity executes the workloads simultaneously in the mainframe and in the cloud [11]. In the beginning of this year, they had migrated the 90% of its core banking to the Cloud [12]. This is a great achievement; however, it brings up the weight that Cloud computing is representing nowadays due to, apart from its other multiple uses, it is being also proposed as the alternative to the mainframe. It is remarkable that Santander ensures that they are using a hybrid model, which means that their private cloud is combined with the solutions of the public cloud providers [13]. Yet, in terms of vendor lock-in, there is an opportunity of managing the situation better.

Because the migration of the mainframe, even when it is a milestone for the bank industry (it worths Santander bank the “most innovative bank” award [14], it is

only an example. The Cloud development has gone much further, and this is only a grain of sand in the mountain that the Cloud has become.

Private corporations like the mentioned banks (Santander, Openbank, BBVA, BCP), Insurance companies (Allianz), e-commerce (Pharmashop, Mercado Libre), etc; has their infrastructure currently implanted in the Cloud. But also, public entities like Spanish Government are planning to migrate part of its IT infrastructure to the Cloud adopting the hybrid model [15].

This suppose a great concentration in the cloud market, and consequently, is a hint that shows the importance of considering the vendor lock-in situation, not only from an individual perspective, but also from a social point of view. It is clear that the explosive rise of Cloud Computing leads to the fact that the cloud is positioning itself as a potential cause of vendor lock-in.

1.2.4 State of the art: vendor lock-in challenge in the Cloud

As we have seen across all the introduction, there are multiple investigations about all fields related to the Cloud: its adoption by segments, the amount spent every year on its solutions, the perception of its clients about its benefits, its impact in the climate change, etc. Also, the vendor lock in has been explained theoretically in the introduction of this project. Therefore, this subsection will cover the explanation of the relationship between vendor lock in and the cloud in detail and the strategies that have been developed until now divided in three layers: infrastructure, architecture and code development.

Opara-Martins et all explains clearly when vendor lock appears in the Cloud: “the lock-in issue arises when a company, for instance, decides to change cloud providers (or perhaps integrate services from different providers), but is unable to move applications or data across different cloud services because the semantics of resources and services of cloud providers do not match with each other” [2].

Infrastructure

This study, as many other studies and IT media, remark the importance of retain the flexibility to change between Cloud providers focusing in the infrastructure creation, as it leads to a situation where the market is more competitive for providers as customers.

The interoperability and portability arise as the key aspects to address this challenge. Ensures them can help to increment the reliability on the Cloud solutions and a there is a consensus about the importance of the adoption of standards in the Cloud.

These standards are commonly focused on the infrastructure part of a Cloud Application and rely on of Infrastructure as code (IaC). IaC is a widely used practice which is based on managing and providing infrastructure resources using code instead of manual procedures. These is a powerful tool for many reasons: it allows to have a documentation about this part of the process, which is the code itself, it can be replicated, reduces the human errors and also decouples the infrastructure creation from the provider.

Cloud standards can be classified according to the categories of cloud services: PaaS standards and IaaS standards. Some of the most famous of these standards are CAMP and TOSCA respectively.

Karmakar provides a concise definition of CAMP: “is a language-, framework-, and platform independent API to manage application in a PaaS cloud. It provides HTTP based REST API with JSON serialization, portable packaging (zip, tar, and gzip) with YAML metadata”. CAMP can be used to create a Cloud infrastructure that can be easily migrated to another cloud provider that supports the standard [16]. The drawback of this standard is that it has not been widely embraced by public Cloud providers.
.
Conversely, TOSCA has more popularity between the cloud providers.
However, none of the main Cloud providers are especially keen on standards, “premature standardization considered harmful” is a famous statement from Werner Vogels, CTO of Amazon [17].

Another strategy that has been detected that can be useful to avoid vendor lock in is implementing DevOps. Technologies like Docker and containerization in general, orchestrators, deployment tools like Jenkins, etc. automatizes process that can be used after a migration to another provider.

Operating system virtualization can be used to deploy the application code in any public or private cloud service provider due to its lightweightness [14]. The reason is these tools does not have a complete Operating System, they rely on the host operating system and only provides by itself the necessary infrastructure to run the application. Kratzke indicates that this implies some coupling with the OS of the host, but tools like Docker runs in Linux Operating Systems which are the most common to deploy web applications and the Cloud providers support it. So, even when operating system virtualization is not as flexible as hypervisor-based virtualization, this author states that it provides a “natural” and immanent cloud infrastructure abstraction layer. Therefore, containerization is a better option in general that hypervisor virtualization.

From the architectural point of view, the most common solution related to the Cloud Computing is microservices. In a migration, microservices can be moved part by part. But some studies point to FaaS as a strategy to facilitate the migration between cloud providers, but, as Mo et all suggest, there are sometimes dependencies with the Cloud provider through the BaaS utilities. That implies that, for example, developing a lambda in AWS would be common to use

a utility to connect to S3 and this complicates migrations. However, these authors offer Cloud service abstraction libraries (like apache JCloud) as a possible solution to this problem [18].

Another recommendation is to use popular industry standards like HTTP, JSON and OAuth.

Although is a key aspect that can determine the difficulty in the migration process, code is an aspect that is underestimated in the consulted investigations. However, some recommendations can be found in forums or IT webpages, more focused on the programmer work [19]: use of interfaces for loosely coupling code, use of the strictly necessary third-party dependencies, avoid hardcoding information related to external providers, void proprietary formatting, create documentation, etc.

Increase application portability is a key aspect to escape from vendor lock in, and it should be kept in mind while coding. From a wider point of view, microservices is the architecture solution more appropriate; but there are useful strategies that can be used inside them. DDD approach is confirmed as one of them. Maintain the domain logic isolated from the logic related to the cloud providers is a good strategy that prevents from coupling.

In conclusion, in the three layers of a web application project there are several known strategies that prevent vendor lock-in. Nevertheless, studies are more concentrated in the infrastructure topic. It is relevant to mention that, out of these three perspectives, there are other initiatives that should be mentioned: multi-cloud strategy, alignment with business strategies,

1.2.4.2 Research gaps

Even when there are plenty of documentation, web page information, forums, etc that address the problem of the vendor lock-in there are areas for further exploration. That is the case of coding strategies, because most of the research are focused on infrastructure and architecture; due to it seems that these are they key aspects of the Cloud Computing but the way the programmers code influences determinedly in the migration difficulty. That is the reason why, even when this project will cover the other two disciplines, it will also be developed from the practical point of view of developing and application.

There is another aspect which is not fully covered in other investigations and inspires this work. The research consulted are commonly focused on one of the three presented layers, but I think that for a more practical approach that intent to be useful for the creator of a small/medium web app, research on best practices in all these fields should be collected, classified, and presented together in a guide that outlines fundamental aspects.

1.3. Goals

1.3.1 Main goal

This project has the aim of study how to address vendor lock-in, an issue that has caused problems in web applications development historically and that is currently a challenge in the context of Cloud Computing expansion. The strategy used will be study the relationship between vendor lock-in and cloud development, then investigate and propose strategies that can be applied to all the phases of the software development cycle and finally propose a prototype which exemplifies this approach.

1.3.2 Academic goals

Apply the knowledge acquired in the Techniques of Software Application Development degree to provide solutions for an issue that is currently a milestone in the industry. In the scope of this project, I will be able to apply the theory learned in the different subjects of the degree but also the practical knowledge in the code prototype which will consist in an SBM's web application example developed for the cloud environment which should be easy to migrate to another infrastructure.

1.3.3 Professional goals

As it has been mentioned before, Cloud Computing is an area that is thriving sector in computer programming. There is a growing demand for professional profiles related to Cloud technology and it is of my interest to delve in this field.
My profile is currently oriented to cloud technologies, but this is an opportunity to make an investigation in a field that is familiar to me but in which there are still much to learn. The good practices proposed by this investigation which will the ones that I will use in the projects that I will address in my professional career. Problems that I am facing today and that I see the companies I have worked for are facing too.

1.3.4 Personal goals

This final project is the only opportunity that many students have to dedicate a great amount of time to investigate a topic of its interest. Investigate about vendor lock-in is a personal challenge that I had since I become studying and working with cloud environments. Also, I was requested by a familiar to develop a web application for his business in the cloud and I realized that SMBs also develop

their solutions in the cloud and that it was my duty to ensure that I could decouple the project from the provider as much as possible so in the future this small company is not tied to any of them .

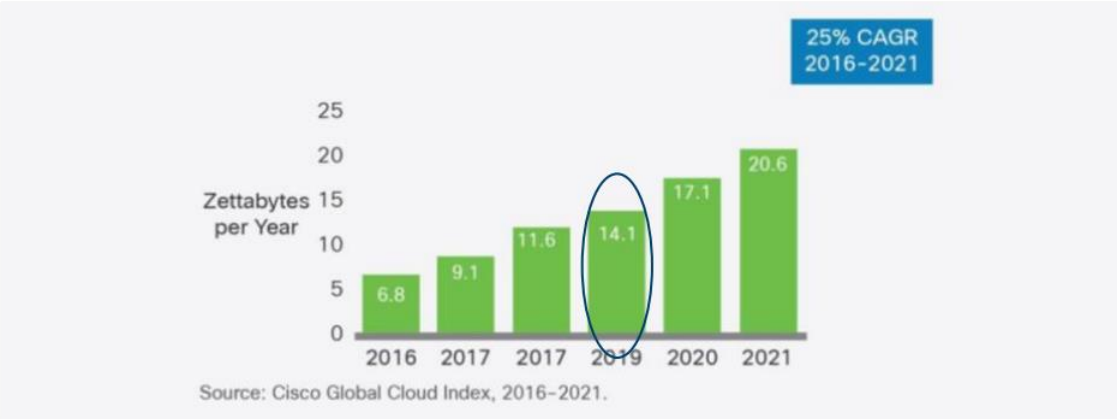
1.4. Sustainability and ethical/social challenges

Sustainability

As it was said in section 1.2.1, Could infrastructure have been adopted by many different companies and its use is growing fast during the last years. This implies that the data centers energy consumption has become a critical issue regarding to the environment impact.

As this work purpose incites to use Cloud Computing, it is important to point out the importance of the energy consumption that this type of infrastructure generates. According to the European Union, “The topic of energy-efficient cloud computing has become a priority on the EU political agenda and the European Digital Strategy”. Their findings stablish that data centers in the EU accounted for the 2.7% of the total energy consumption and this cypher is expected to reach 3.21% in 2030 [20].

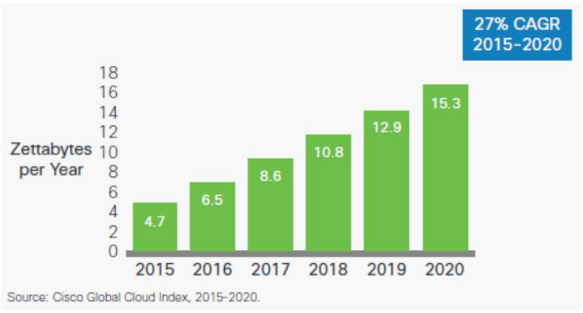
Therefore, the way this energy is obtained is fundamental. According to the Shift Project, a Paris investigation of the environmental impact of the digital world, the forecast of the data center traffic and network was the following in 2017:



[Source: (Cisco, Visual Networking Index: Global - 2021 Forecast Highlights, 2017)]

Figure 4: Forecast of data center network traffic in 2017

However, the prediction figures they were working with in 2016 and 2015 show that their development is growing way faster than in their predictions.



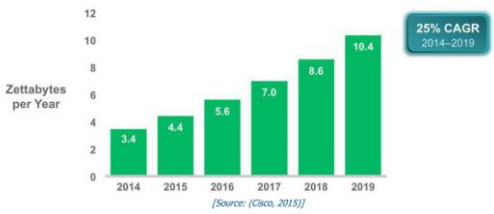


Figure 5: Forecast of data center network traffic in 2016 **Figure 4: Forecast of data center network traffic in 2015**

In their report presented in 2019, they already highlighted the magnitude of the data centers traffic which has been increased a 35% per year and also a similar situation is given by the data centers storage which grows more than a 40% per year [21].

This can be traduced in “a 8-10% of our energy consumption and 2-4% of our greenhouse gas emissions” [22] according to EU reports. As these projects propose some guidelines to create a web application using the Cloud infrastructure, it is important to point out that this impact in the environment affects its scope.

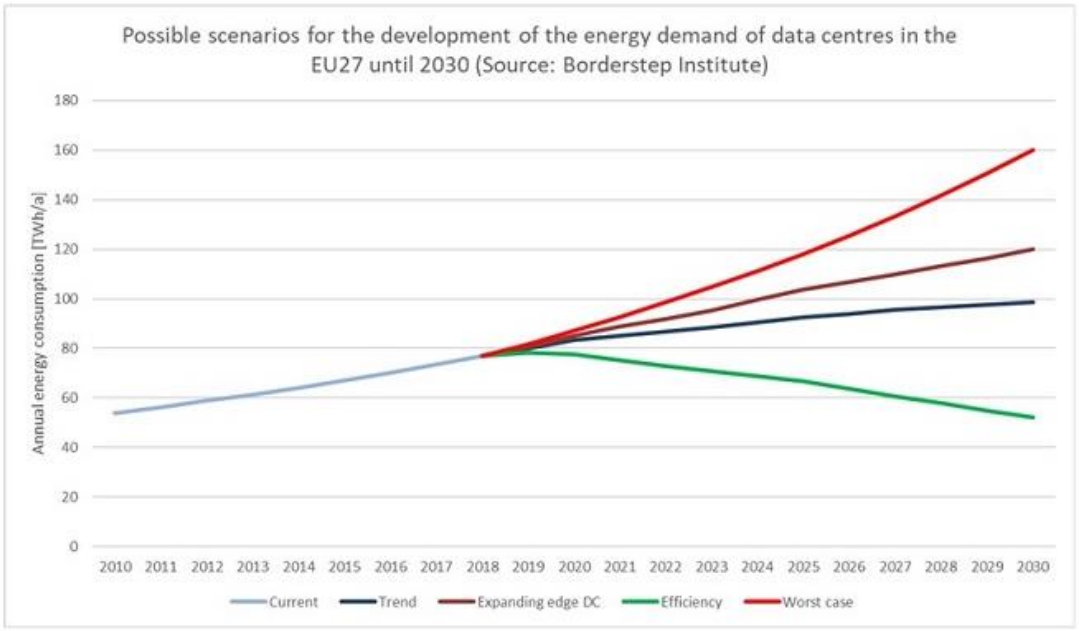


Figure 6: Forecast of the energy consumption of data centers until 2030

However, public organizations like EU are working to provide guidelines to walk into a greener Cloud, so the benefits that the cloud provides are not compromised by the drawbacks that their impact suppose. Also, even when the data centers are proven to be a point of contamination, that does not mean that the private

hardware acquisitions do not produce greenhouse gas emissions and a huge carbon footprint.

If there is a positive aspect of this concentration is the control. Cloud providers can be easily forced to reduce its emissions and the measure of the climate impact of the digital services is easier.

Ethical behaviour and social responsibility

It is a great responsibility to have a critical knowledge about the Cloud providers to know which one to choose. And also, there is a great responsibility, especially for projects related to SMBs that cannot afford a big migration (possibly neither a small one) to ensure that in all the stages of the project development the application is as decoupled from the cloud provider as possible.

There is a social perspective of this problem. The concentration of projects of all kinds of industries: banks, e-commerce, public organizations, SMBs, etc suppose a great deal that must be addressed. Some organizations can have resources like specialists or even teams specifically dedicated to decoupling its applications from the cloud providers, but this is not usually the case of small and medium companies that does not have the same budget.

Therefore, they rely on the knowledge that the technicians of this companies have, and this implies that every developer has a responsibility to help their clients to decouple its code and supply clean code, infrastructure and architectural practices so the resulting product of its work is oriented to avoid as much as possible vendor lock in.

The problem is that the good practices, not only but also those that avoid vendor lock in, are not usually a key aspect of software development and a main concern for the software companies. But this project points out sufficient reasons why awareness of this problem is important.

The social aim of this project is to provide a concrete guideline of practices that decouple a project for the cloud. It is intended to do it in a way that will be easy to implement them and also to explain why it is important to implement them. And their practical approach to a simple web application shows how they can be applied to a SMB’s company webpage. Another key point regarding to this aspect is that the methodology objective is to be decoupled from the cloud providers so the concentration is the same but the power this companies can have over the ones that uses its infrastructure is less.

Also, regarding to Cloud providers, another key point of this project is that many companies deny using its solutions precisely because they are afraid of vendor lock-in. Solutions that avoid them can calm possible clients that are afraid of this issue.

As a conclusion, understanding the importance of avoiding vendor lock-in can help all companies, but especially small companies to enjoy the benefits of the cloud but maintaining its dependency and capacity of negotiating with cloud providers.

1.5. Approach and methodology

The methodology proposed intends to fulfil the main objective of creating a web application with the intention of avoiding as much as possible cloud vendor lock in. To do that, the main purpose of this project is to study the vendor lock-in and strategies to avoid it in IT problems, extract existing good practices and propose new ones and elaborate a concrete guide with them and finally develop a small application for a small company web development that applies this guideline.

According to this approach, there are few steps that must be taken to complete this project:

- Provide a historical overview of the vendor lock-in problem to develop the context of the project.
- Determine lines of action to prevent vendor lock in and the strategies that have been historically implemented to avoid it. This will be done in the first step of the project in order to have a better understanding of it and of the state of the art.
- Combine existing ideas and approaches to avoid vendor lock in with new ones that can appear after the investigation in each actuation lines.
- Investigate about bad practices regarding to vendor lock-in and tendencies in companies that are favourable to the extension of this problem and expose causes of the resistance to apply good practises.
- Expose a guideline with a composition of good practices that in the future will be applied to the web application prototype.
- Develop a web app for a small company that exemplifies the use of the good practices proposed.
- Create a migration plan to another cloud provider.

1.6. Schedule

1.6.1 Submissions

This subject supposes 12 ECTS which means 300 hours of student work. This work will be divided following the structure of the next table, which suppose approximately between 2 and 3 hours per day, with a compensation of temporarily more hours during the weekends.

PHASE	DATES	HOURS	SUBMISSION
1	29/09 – 13/10	50	CAT 1. Project plan
2	14/10 – 10/11	90	CAT 2. Design

3	11/11 – 01/12	100	CAT 3. Implementation
4	02/12 – 12/01	60	CAT 4. Final submission

1.6.2 Planning

In this schedule the steps mentioned in section 1.5 will be related to the different sections proposed in this work and the time dedicated to each section will be detailed.

TASK	DURATION	START	END	SUBMISSION
PROJECT PLAN	**50**	**29/09**	**13/10**	**CAT 1**
Read and planification of CATS	5	29/09	30/09	CAT 1
Definition of goals, ethical implications and planification	15	01/10	04/10	CAT 1
Historical overview and context	20	05/10	09/10	CAT 1
Introduction. Problem definition	5	05/10	06/10	CAT 1
Introduction. The cloud: definition, pros and cons	5	06/10	07/10	CAT 1
Introduction. The adoption of the Cloud infrastructure for SMB's web applications	5	07/10	08/10	CAT 1
Introduction. Historical approach to vendor lock-in	5	08/10	09/10	CAT 1
Investigate known prevention strategies	10	09/10	13/10	CAT 1
Introduction. State of the art: the vendor lock-in challenge in the Cloud	10	09/10	13/10	CAT 1
DESIGN	**80**	**14/10**	**10/11**	**CAT 2**
Combine existing ideas with new propositions to prevent vendor lock-in	10	14/10	16/10	CAT 2
Lines of action to prevent vendor lock-in		14/10	16/10	CAT 2
Investigate about bad practices	15	16/10	21/10	CAT 2
Bad practices vs good practices	10	16/10	18/10	CAT 2
Resistance to bad practice applications	5	19/10	21/10	CAT 2
Guideline creation	5	21/10	22/10	CAT 2
App development	135	23/10	10/11	CAT 2
Functional explanation of the prototype	10	23/10	26/10	CAT 2
Use cases definition	10	26/10	29/10	CAT 2
Technical explanation of the prototype – prototype design	30	29/10	06/11	CAT 2
Database design	5	06/11	08/11	CAT 2
Class and architectural diagram	5	08/11	10/11	CAT 2
IMPLEMENTATION	**90**	**11/11**	**01/12**	**CAT 3**
Prototype development Sprint 1	25	11/11	15/11	CAT 3
Prototype development Sprint 2	25	16/11	20/11	CAT 3
Prototype development Sprint 3	25	21/11	25/11	CAT 3

Explanation of the application of the guideline to the prototype	5	26/11	27/11	CAT 3
Migration plan	10	28/11	29/11	CAT 3
Creation of migration plan	5	28/11	29/11	CAT 3
Explanation of prototype after migration	5	30/11	01/12	CAT 3
FINAL SUBMISSION	**70**	**02/12**	**12/01**	**CAT 4**
Report review	45	02/12	26/12	CAT 4
Presentation	25	26/12	12/01	CAT 4

1.6.3 Gantt Diagram

General project Gantt.

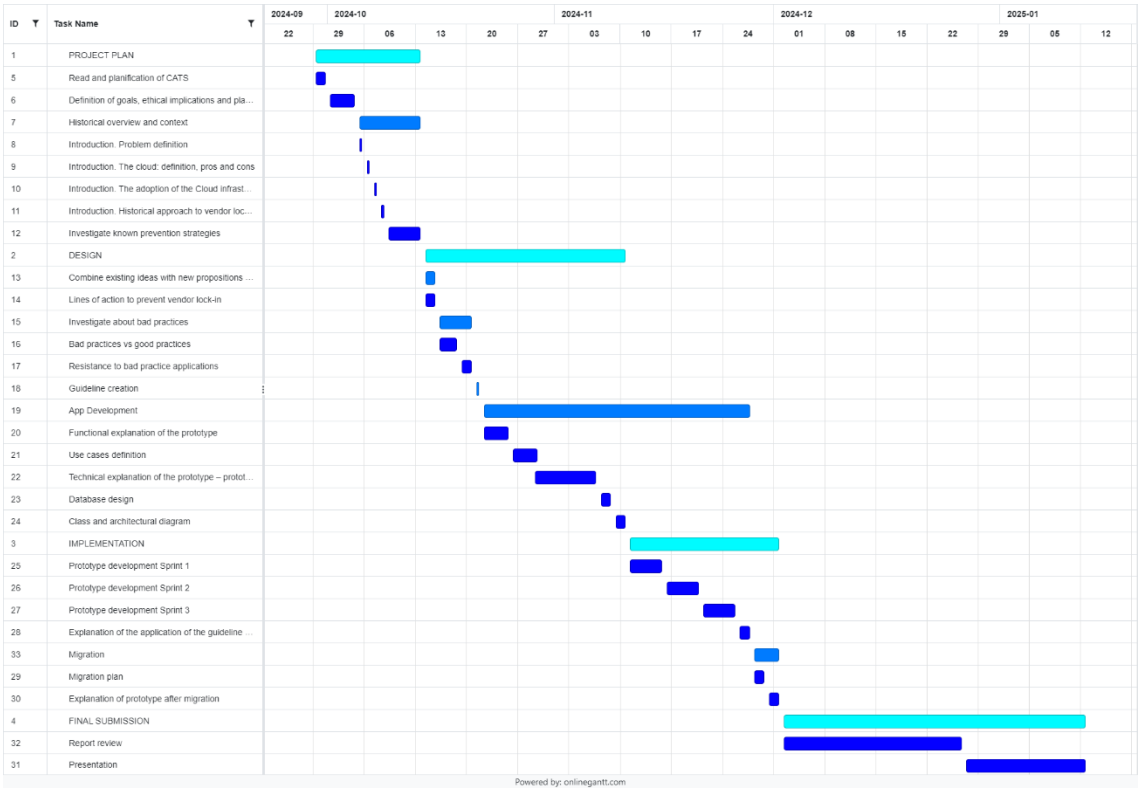


Figure 7: Gantt Diagram for the general project development

Gantt diagram by phases:

Project plan:

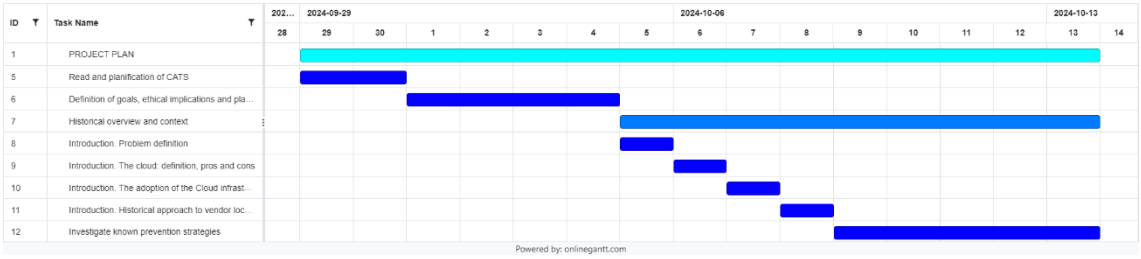


Figure 8: Gantt Diagram for the phase 1 of the project

Design:

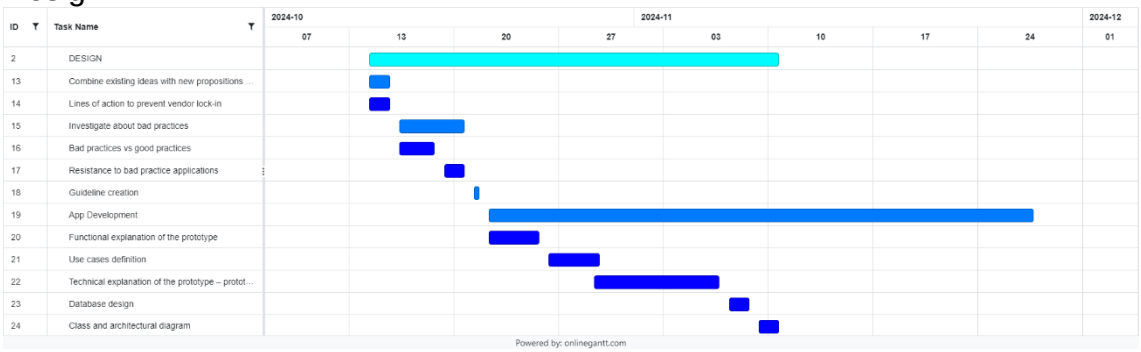


Figure 9 Gantt Diagram for the phase 2 of the project

Implementation:

Figure 10: Gantt Diagram for the phase 3 of the project

Final submission:

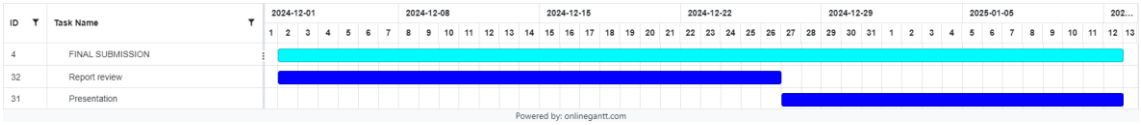


Figure 11: Gantt Diagram for the phase 4 of the project

1.7. Summary of the outputs of the project

There are three main outputs of this project:

- Report: the report is the current document in which the investigation is conducted. Inside the report, another product can be found which is the guideline of good practice for developing Cloud Computing web applications.

- Presentation: as a result of this project, a presentation will be created with the main points and results of the report.
- Product: the product presented is a development of a web application for a small company.

1.8. Brief description of the remaining chapters of the report

The following chapters will be subdivided into the following subsections:

Section 2. Methods and resources

In a first block of this chapter, the first point will be dedicated to classifying the lines of action: at coding level, at architectural level and at infrastructure level. Then, an investigation of bad practices will be conducted and a comparation with good practice to expose the benefits of this last. After this, the guideline with good practises will be created.

Then the prototype design is created, first exposing it from the functional point of view showing the functional requirements and how will be the application main behaviours. Then use cases will be explained.

The next subsection is dedicated to the technical explanation. It will illustrate the reason of the architecture, language, database and coding decisions. It also indicates the technical resources needed, the cloud solutions used, etc.

The last parts of the technical design are focused on the database design and the class and architectural diagram of the web application.

Section 3. Results

In this section the results of the investigation are presented and used to explain the application decision. In this phase the prototype is developed considering the conclusions of the previous chapters and using the good practice guideline.

Then the technical decisions will be shown, and the application created will be used as an example.

The last part of the results section is the migration planification and the explanation of the status of the application in case to a migration to another cloud provider.

Section 4. Conclusions and future work

In this last step the conclusions are stated and a reflection about them will be done. Also, the lines of investigation that has not been explored and the future work paths will be also indicated.

2. Glossary

Vendor lock-in. A scenario where the cost of switching to another vendor is mainly high and for this reason, the consumer is mainly struck with a single cloud vendor.

Cloud Computing. A model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.

Cloud provider. Is an organization that offers Cloud computing services. Amazon Web Services, Microsoft Azure or Google Cloud are the most famous examples of cloud providers.

Scalability. Capacity of adaptation that the cloud must be adapted to the client necessities.

Economy of scale. Cost reductions because of the volume of the production.

Cloud. Remote servers connected to the internet that allow users to store, manage, and process data, as well as access software and networking services without the need for on-premises infrastructure

Cobol. Language developed in the second half of the twenties, creates for efficient data processing with a friendly grammar.

Public Cloud. Service model in the cloud in which the resources are provided by an external company and available for the public.

Private Cloud. Service model in the cloud in which the cloud services are used only by a single company.

Hybrid Cloud. Service model in the cloud in which the cloud services are provided as a combination of external resources and private resource.

Software Development. Process of design, create and implement a applications.

Web Application. Software executed in a server that is accessed through a browser.

Microservices. Architecture in which an application is divided in small services that communicates with each other through API, queues or events.

3. Bibliography

[1] Hari Kumar, P., & Mala, G. A. (2022). H2RUN: An efficient vendor lock-in solution for multi-cloud environment using horse herd Runge Kutta based data placement optimization. *Transactions on Emerging Telecommunications Technologies, 33*(9), e4541. https://doi.org/10.1002/ett.4541

[2] Opara-Martins, J., Sahandi, R., & Tian, F. (2016). Critical analysis of vendor lock-in and its impact on cloud computing migration: a business perspective. *Journal of Cloud Computing*, *5*, 1-18.

[3] National Institute of Standards and Technology. (2011). *NIST special publication 800-145: The NIST definition of cloud computing*. U.S. Department of Commerce.
https://nvlpubs.nist.gov/nistpubs/legacy/sp/nistspecialpublication800-145.pdf

[4] Huawei Technologies Co., L. (2023). *Cloud Computing Technology / by Huawei Technologies Co., Ltd.* (1st ed. 2023.). Springer Nature Singapore. https://doi.org/10.1007/978-981-19-3026-3.

[5] IBM. (2022, June 28). Modern mainframe: A strategic imperative for government. IBM Policy. Available in: [https://www.ibm.com/policy/modern-mainframe-government/]. Consulted on: 04/10/2024.

[6] Centro Criptológico Nacional. (2014). *Seguridad en entornos cloud: Esquema nacional de seguridad* (CCN-STIC 823). Available in: [https://www.ccn-cert.cni.es/publico/seriesCCN-STIC/series/800-Esquema_Nacional_de_Seguridad/823-Seguridad-en-entornos-cloud/823-Cloud_Computing_ENS.pdf]. Consulted on: 02/10/2024.

[7] Eurostat. (2023). *Cloud computing - statistics on the use by enterprises*. European Commission. Available in: [https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Cloud_computing_-_statistics_on_the_use_by_enterprises]. Consulted on: 03/10/2024

[8] Pérez, M. (2024, August 15). *El gasto en servicios de infraestructura en la nube evoluciona a doble dígito*. Silicon. Available in: [https://www.silicon.es/el-gasto-en-servicios-de-infraestructura-en-la-nube-evoluciona-a-doble-digito-2556265]. Consulted on: 03/10/2024.

[9] Flexera. (2024). State of the Cloud 2024: Cloud initiatives and metrics. Flexera. Available at: [https://info.flexera.com/CM-REPORT-State-of-the-Cloud-2024-Thanks#cloud-initiatives-metrics]. Consulted on: 05/10/2024.

[10] Channel Partner. (2024). La evolución del negocio del cloud en 2024. Channel Partner. Available at: [https://www.channelpartner.es/cloud/la-evolucion-del-negocio-del-cloud-en-2024/#Las_cifras_de_la_nube_publica_en_Espana]. Consulted on: 05/10/2024.

[11] Santander. (2024, April 18). *7 preguntas sobre cómo Gravity está transformando Santander*. Santander. Available in : [https://www.santander.com/es/stories/7-preguntas-sobre-como-gravity-esta-transformando-santander]. Consulted on: 05/10/2024

[12] UE STUDIO. (2024, 8 de enero). *Gravity, la gran apuesta digital de Banco Santander*. *El Mundo*. Available in: [https://www.elmundo.es/uestudio/2024/01/08/659beca8e4d4d883708b457a.html]. Consulted on: 05/10/2024

[13] Banco Santander. (2024, 26 de julio). *Santander Cloud: innovación y revolución al servicio de nuestros clientes*. Available in: [https://www.santander.com/es/stories/santander-cloud-innovacion-y-revolucion-al-servicio-de-nuestros-clientes]. Consulted on: 05/10/2024.

[14] Muñoz, J. (2023, 29 de septiembre). *Gravity: el viaje de Santander hacia la nube corona a la entidad como la más innovadora*. *La Vanguardia*. Available in: [https://www.lavanguardia.com/economia/20230927/9237544/gravity-viaje-santander-nube-corona-entidad-mas-innovadora-brl.html]. Consulted on: 05/10/2024.

[15] Gobierno de España. (2022, 24 de noviembre). *Estrategia de servicios en la nube para las AAPP.* Available in: [https://www.lamoncloa.gob.es/serviciosdeprensa/notasprensa/asuntos-economicos/Paginas/2022/241122-estrategia-servicios-nube-aapp.aspx]. Consulted on: 05/10/2024.

[16] Karmarkar, A. (2014, October). CAMP: a standard for managing applications on a PaaS cloud. In *Proceedings of the 2014 Workshop on Eclipse Technology eXchange* (pp. 1-2).

[17] Kratzke, N. (2014). Lightweight virtualization cluster how to overcome cloud vendor lock-in. *Journal of Computer and Communications*, *2*(12), 1.

[18] Mo, D., Cordingly, R., Chinn, D., & Lloyd, W. (2023, December). Addressing Serverless Computing Vendor Lock-In through Cloud Service Abstraction. In *2023 IEEE International Conference on Cloud Computing Technology and Science (CloudCom)* (pp. 193-199). IEEE.].

[19] Coralogix. (2023). 11 tips for avoiding cloud vendor lock-in. Coralogix. Available at: [https://coralogix.com/blog/11-tips-for-avoiding-cloud-vendor-lock-in/]. Consulted on: 05/10/2024.

[20] Comisión Europea. (2020, 9 de noviembre). *Energy-efficient cloud computing technologies and policies for an eco-friendly cloud market*. Available in: [https://digital-strategy.ec.europa.eu/en/library/energy-efficient-cloud-

computing-technologies-and-policies-eco-friendly-cloud-market]. Consulted on: 05/10/2024.

[21] The Shift Project. (2019). *Lean ICT: Towards a more sustainable digital economy*. Available in: [https://theshiftproject.org/wp-content/uploads/2019/03/Lean-ICT-Report_The-Shift-Project_2019.pdf]. Consulted on: 05/10/2024

[22] Comisión Europea. (n.d.). *Green cloud*. Available in: [https://digital-strategy.ec.europa.eu/en/policies/green-cloud]. Consulted on: 05/10/2024.

4. Appendices

List of sections that are too long to be included in the body of the report and that are self-contained (for instance, user manuals, installation instructions, …)

Depending on the type of project, this part of the report may be unnecessary.

,