TÉCNICO LISBOA

CCalc: Cloud Calculator for the Public Administration

Eduardo André Fernandes Cidres

Thesis to obtain the Master of Science Degree in

Information Systems and Computer Engineering

Supervisor: André Ferreira Ferrão Couto e Vasconcelos

Examination Committee

Chairperson: Paolo Romano
Supervisor: André Ferreira Ferrão Couto e Vasconcelos
Member of the Committee: Miguel Filipe Leitão Pardal

January 2021

This is dedicated to my grandfather João Cides. For him, knowledge meant everything and that is something that I share with him, for it was what allowed me to get where I am today. If he saw me today, I know he would be proud. May you rest in piece grandfather.

Acknowledgments

I want to start by thanking my loved ones and friends, for giving me emotional support as well as motivation and belief in my ability to carry out this challenge. I am also thankful for Instituto Superior Técnico for providing me all the tools and personal growth to be able to face this masters thesis challenge. I want to thank professor André Vasconcelos for providing me the guidance and needed information to succeed and reach the end of this journey. A thanking note to Filipe Leitão as well, where his cooperation and provided information helped me understand and enrich the developed work.

Resumo

À medida que a adoção de cloud se vai tornando cada vez mais comum enquanto forma de alojar serviços e impulsionar negócios, o seu processo de adoção e migração permanece até hoje pouco claro ou definido. O objetivo deste trabalho consiste em criar uma ferramenta de simples utilização que permita determinar a viabilidade de migração ou adoção de cloud de um dado sistema. Para isso um conjunto de critérios baseado numa anterior revisão da literatura é criado de forma a representar a generalidade de aspetos tecnológicos a ter em consideração numa análise de viabilidade de cloud de um sistema. A partir deste conjunto de critérios é também criado um questionário baseado nos mesmos, de forma a providenciar um melhor entendimento sobre os critérios em causa. Isto torna a ferramenta mais acessível a utilizadores que não possuam um nível avançado de conhecimento sobre cloud. Este trabalho é depois anexado a um algoritmo de decisão multi-critério de forma a produzir uma pontuação, representado o nível de recomendação para sistema (se este deve ou não ir para cloud). Após isto, a ferramenta é testada com base num caso real de adoção de cloud utilizado na administração pública de forma a avaliar a sua aplicabilidade.

Palavras-chave: Adoção cloud, Ferramenta de migração, Avaliação de viabilidade, Decisão Multi-critério

Abstract

As cloud adoption becomes increasingly popular as a way to host services and improve businesses, its adoption and migration still remain to be a clearly defined process. Specifically, when the adoption is to be made within the public administration, where additional constraints apply when compared to the private sector. Legislation needs to be created, standards need to be developed, and public organizations need to be in sync with their cloud goals and approaches. This work aims to create a simple to use tool that allows determining the viability of cloud migration or adoption of a system. To do this, a set of criteria based on a previously conducted literature review is created, aiming to represent the spectrum of technological aspects to take into consideration in a cloud feasibility analysis of a given system. From this set of criteria, a questionnaire is also created to ease the understanding of each criterion, to make the tool accessible to decision-makers who do not possess an expert level of cloud-related knowledge. This work is then married with a multi-criteria decision-making algorithm to produce a recommendation score for the evaluated system, where this translates to if the system should or should not go to a cloud environment. The tool is then tested against a real-life scenario of cloud feasibility analysis to evaluate its usability and applicability.

Keywords: Cloud adoption, Migration tool, Feasibility assessment, Multi-criteria decision-making

Contents

Acknowledgments . . . v
Resumo . . . vii
Abstract . . . ix
List of Tables . . . xiii
List of Figures . . . xv

1 Introduction 1
1.1 Motivation . . . 1
1.2 Context and Problem Overview . . . 1
1.3 Objectives . . . 2
1.4 Thesis Outline . . . 3

2 Background and Related Work 5
2.1 Cloud service models . . . 5
2.2 Deployment of cloud computing . . . 6
2.3 Cloud characteristics . . . 7
2.4 Cloud migration popular use cases . . . 7
2.5 Cloud migration challenges . . . 8
2.6 Multi-criteria decision-making . . . 9
2.7 Enterprise architecture applied to the cloud . . . 11
2.8 Current state of cloud in the Portuguese public administration . . . 12
2.9 Cloud adoption in public administrations . . . 12
2.10 Criteria used in cloud adoption evaluation . . . 14
2.11 Framework analysis . . . 17
2.12 Decision-making approaches to cloud adoption . . . 18

3 Cloud assessment tool 19
3.1 Solution description . . . 19
3.2 Use cases . . . 20
3.3 Data representation . . . 21
3.4 Determination of relevant criteria set to use and respective definitions . . . 21
3.5 Elaboration of question set to represent the criteria . . . 23

3.6 Answer scales and scoring system 25
3.7 Solution architecture 25

4 Evaluation and Results **29**
4.1 Evaluation methodology 29
4.2 Evaluation case 29
4.3 Use of the tool 31
4.4 Obtained results and analysis 32

5 Conclusions **33**
5.1 Achievements 33
5.2 Limitations 33
5.3 Future Work 34

Bibliography **35**

A Additional figures **A.1**
A.1 Tool screenshots A.1

List of Tables

2.1 Cloud service models . . . 6
2.2 Cloud deployment models . . . 7
2.3 Cloud characteristics . . . 8
2.4 Cloud migration challenges . . . 10
2.5 Cloud adoption benefits . . . 13
2.6 Cloud adoption drivers and objectives in other countries . . . 15

3.1 Questions used in the evaluation of criteria . . . 24
3.2 Scoring example of question to assess the Trialability criterion . . . 25

4.1 Hardware requirements for host machines of production environment . . . 31

List of Figures

2.1 Popular cloud use cases 9
2.2 General structure of an MCDM process 10

3.1 Use case diagram of the cloud assessment tool 21
3.2 UML representation of the data used in the tool 22
3.3 Example of question used to assess the Trialability criterion 25
3.4 Application layer of the cloud assessment tool 26
3.5 Business process diagram of the interaction between the user and the tool 26

4.1 Pre-production deployment architecture 30
4.2 Production deployment architecture 31

A.1 Front page of the tool A.1
A.2 End of the front page of the tool A.2
A.3 MCDM page of the tool A.2
A.4 Edit/Add page of the tool using Django Admin page A.3
A.5 Results of the application of the tool evaluation case A.3

Chapter 1

Introduction

1.1 Motivation

In the last twenty years, cloud computing has become an ever-increasing topic in research and development of new technologies that are able to leverage great benefits that come with it when properly applied. Some of these benefits are more apparent than others, such as upfront investment cost reduction, flexibility, and agility when deploying services in a cloud environment. Others possess a more indirect impact like environmental and energy efficiency or universal access, where cloud services can be accessed from anywhere and through any device that can to connect itself to the internet. This proves to be advantageous, since it allows remote work [1], something that has become increasingly relevant in the latest years.

1.2 Context and Problem Overview

However, cloud adoption does not come without its challenges, one of the biggest being security since it brings new possible vulnerabilities and threats along with all of the benefits it brings [2]. Another challenge involves the architecture of the software product, where a loosely coupled architecture is required, which is something that generally does not exist in legacy applications, these being the subjects to cloud migration frequently [3].

Despite all the advances made in the latest years, cloud migration remains a process with many uncertainties, specifically when it comes to governmental organizations since these are often subject to additional constraints when tackling the adoption of cloud computing. This is mostly due to the lack of legislation and definition of standards on how to approach a sizable transformation as the adoption of cloud. Even when looking at the development of cloud frameworks, it can be observed that to this day there is not a unified approach to take when facing the possibility of cloud adoption [3–7].

There have been efforts to produce tools to aid in the process of decision making for cloud computing adoption, but usually, these are built with fixed criteria in mind and lack when it comes to customization [7–10]. It is also worth mentioning that none of these tools allow for the measurement of cloud readiness of a given system, which is something that can be useful, particularly when the subject of evaluation is a legacy system where different aspects need to be assessed and evaluated to conclude if the required changes to adapt the system to the cloud environment are proven to be worth doing as to benefit from the best that the cloud has to offer.

In essence, there is a lack of tools and mechanisms that enable an efficient and yet simplified cloud feasibility analysis while trying to converge on a standardized approach. An approach that takes the full spectrum of technological aspects of the evaluated system into account and combines them with a decision-making support system, while simultaneously simplifying the process. This makes it easier to use for decision makers who do not possess as high level of knowledge about the cloud environment as an expert of the field. Convergence on a standardized approach would not only be beneficial for cloud adoption or migration in general, but it would have considerably more impact from a public administration point of view.

The importance of creating standards and procedures to follow across departments and institutions can be considered greater when comparing to the private sector, creating demand for tools and consolidation of knowledge to be used and standardized across all governmental entities and institutions that may require the adoption of cloud services. With this work we intend to develop a tool that addresses the issues stated and contributes to the overall process of cloud adoption within the public administration, by making it a usable yet effective instrument to aid in the decision-making process of cloud adoption and migration.

1.3 Objectives

The objective of this thesis consists in the development of a tool to aid the public administration institutions in the decision-making process of cloud adoption for a given system. By taking advantage of multi-criteria decision-making algorithms as well as a system that allows the simplification of score attribution, we intend to offer a simpler process to assess the cloud feasibility of a system for decision makers that do not possess an expert level of knowledge when it comes to cloud. Not only this, but the tool is also to be developed with flexibility and future changes in mind, allowing for customization for almost everything it contains.

By taking advantage of previously developed work, where we were able to gather enough information, we created a set of criteria to use for the general case of cloud migration. With this new information, we now use it as a basis for the development of the tool and expand upon it as to create a simpler process.

With this work we intend to bring the following contributions:

- Consolidation of knowledge and criteria required for the technological assessment of cloud feasibility analysis
- Study and application of multi-criteria decision-making algorithms that use the previously mentioned criteria set to produce a tool to aid decision-makers
- A tool built with simplification and ease of use in mind, making it accessible to decision-makers

who do not possess an expert level of knowledge while still requiring some baseline knowledge about the cloud environment

1.4 Thesis Outline

In chapter 2 a complete background is given and the related work is laid out, ranging from cloud nomenclatures and aspects to government applications of cloud. In addition topics such as existing cloud adoption/migration frameworks are also explored as well as the world of multi-criteria decision making and its most known algorithms.

In chapter 3 an extensive explanation is given about the developed solution, going from its description to the several components used to build it. These components are the criteria used in the tool, the respective questions, and the multi-criteria decision making (MCDM) algorithm used.

In chapter 4 we present the used evaluation methodology for the developed work as well as the description of the evaluation case used to test the developed tool.

In chapter 5 we display the results of our work, going from a full walkthrough of the tool itself and its capabilities to the results obtained from testing the tool and its respective analysis.

In chapter 6 we draw conclusions from the developed work and the respective achievements, reflecting on what could have been done differently, and what can be improved in future developments.

Chapter 2

Background and Related Work

This chapter starts with section 2.1 where we provide a brief explanation of the cloud service models that can be encountered today, in section 2.2 we explain the deployment models available when adopting cloud and their respective characteristics as well as benefits. We then go to section 2.3, where we briefly explain the characteristics of cloud which can also be seen as the reasoning behind choosing to adopt cloud. Within section 2.4 we describe the most popular use cases where the capabilities of cloud can be leveraged to its full potential. In section 2.5 we present the most frequent challenges that are faced when considering the adoption of cloud, followed then by section 2.6 where we explain succinctly what multi-criteria decision making is, and how it can help us in finding the best solution possible. Not only this, but we also go further and present some MCDM algorithms and the differences between them. Following this, we have section 2.7 where we broach the considerations to take into account when considering the adoption of cloud from an enterprise architecture point of view. In section 2.8 we provide an overview of the current state of cloud adoption within the Portuguese public administration, and in section 2.9 we analyze the state and efforts made for cloud adoption in other countries in the same field of public administration. Throughout sections 2.10 and 2.11 we analyze results obtained from a conducted systematic literature review while providing some background on currently existing cloud adoption frameworks 2.11. This chapter ends with section 2.12, where a comparison between multi-criteria decision-making approaches to cloud adoption is established.

2.1 Cloud service models

The existing cloud service models are Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) as explained in [11] and shown in table 2.1.

In the IaaS service model, the customer can take advantage of all infrastructural resources supplied by the cloud provider from servers and networks to operating systems and storage solutions. These on-demand resources rely primarily on virtualization.With this model, the customer only needs to be worried about what is running in the virtual machines, since the responsibilities of managing and controlling the underlying infrastructure as well as the hardware purchase and installation, are passed to the cloud provider.

In the PaaS service model, the customer gets access to resources such as programming languages, libraries, APIs, and tools given by the cloud provider. This service model allows the customer to develop custom applications while the responsibility of the installation and configuration of the development environment is held by the cloud provider.

In the SaaS service model, the cloud provider gives access to software that is hosted on the cloud, therefore eliminating the process of buying, maintaining, and installing the infrastructure as well as developing the software itself. Customers simply need to log in and use the provided application.

Service model	Characteristics
Infrastructure as a Service (IaaS)	Virtualization Infrastructure provided by the CSP Management and control responsibilities belong to CSP Hardware purchase and installation by the CSP
Platform as a Service (PaaS)	Access to programming languages, APIs and tools provided by the CSP Installation and configuration of development environment responsibilities belong to CSP
Software as a Service (SaaS)	CSP gives access to software that is hosted on the cloud Development, infrastructure maintenance, purchase and installation responsibilities belong to CSP

Table 2.1: Cloud service models

2.2 Deployment of cloud computing

When it comes to the deployment of our cloud solution, there are four different options depending on the customer requirements. These options include public, private, community or hybrid cloud [11] and are also represented in table 2.2.

In public cloud deployment the access to the infrastructure, whether it is computation, storage, etc. is shared by all clients and given by a single cloud service provider. All resources are operated and hosted in an external cloud and are made available through the internet.

In a private cloud, the provided resources are usually used exclusively by a company or an organization and can be hosted off-premises or on-premises, which is a better solution if the client is looking for a more secure way to store and manage data as well as running critical business operations.

A community cloud is based on the same idea as private cloud solutions, but it extends the access of the provided resources to more than one entity. This is a more suitable solution for a group or organizations that possess a common interest.

When it comes to the hybrid cloud, it can be defined as a combination of two or more cloud deployment models, where the infrastructure components are interconnected.

Deployment model	Benefits	Characteristics
Public	Easy setup Easy data access Flexibility in updating resources according to use Year-round provided maintenance	Availability to all clients Infrastructure is provided and managed by the cloud providers Resources provided in a pay-per-use model
Private	Control over hardware Control over information High level of security, privacy and reliability	Infrastructure belongs to a single organization allowing for a greater control of security mechanisms and information
Community	Easy data sharing and collaboration High level of security, privacy and reliability	Shares similarities with the private model but the resources are shared across similar organizations Best suited for cooperative projects between organizations
Hybrid	High level of security, privacy and reliability than private	Combination of two or more models allow the retainment of the benefits present in a singular model

Table 2.2: Cloud deployment models

2.3 Cloud characteristics

The cloud presents several characteristics described by the US National Institute of Standards and Technology (NIST) [11] and presented in table 2.3, where these also pose as some of the reasons for the adoption of cloud solutions by companies.

One of these is on-demand self-service, which means that cloud capabilities can be provided to clients at any place and time without the need of the intervention of a human operator.

Another present characteristic is the broad network access due to the ease of access to cloud capabilities over the network in several types of devices. The cloud also offers resource pooling, by aggregating computational resources that can provide services to multiple clients. These resources are made available according to the clients' needs, in the shortest time possible [12].

Rapid elasticity is another characteristic, referring to the fact that scaling is possible according to the need of resources of the clients, this can happen in an automatic fashion and to a great amount, at any given time.

The final cloud characteristic pertains to the monitoring of the utilized resources while allowing for greater management and transparency between the cloud service provider (CSP) and the client.

2.4 Cloud migration popular use cases

The characteristics previously mentioned make a cloud solution look very appealing for many businesses looking to make the transition. Although not all cases justify a migration of services to the cloud, there are a couple of common cloud use cases as mentioned in [13] and represented in figure 2.1.

Cloud characteristic	Description
On-demand self-service	Cloud capabilities can be provided at any time and place automatically
Broad network access	Ease of access to cloud services through any device over the network
Resource pooling	Aggregation of computational resources in order to provide services to multiple clients
Elasticity	Scaling is done according to client needs automatically at any given time
Monitoring of resources	Provided tools that allow careful monitorization of utilized resources and respective costs

Table 2.3: Cloud characteristics

One of the most common is cloud bursting, where the cloud capabilities allow a business to handle peaks of traffic. This is achieved by redirecting the excess traffic from their data centers to a CSP, making the business's traffic capacity more flexible.

Another popular use case is archiving/storage since the cost of storage in the cloud has become drastically cheaper than local storage. It becomes cheaper by making efficient use of storage capacity allocated services as well as its scale capabilities. By taking advantage of storage in the cloud, also makes the process of data retrieval a less complex process.

The cloud is a great environment for processing large amounts of data on-demand as well, for data mining and analytics. Since in the public cloud resources can be provisioned upon request, it makes for a great cost-saving solution not only in physical infrastructure but in the management of the systems as well, by deploying an on-demand cloud model.

When it comes to software development, the cloud can prove to be very helpful regarding testing. With test environments being able to be run on the cloud, it allows companies to speed up the development cycle, taking advantage of its flexibility and scalability to stress test their software on a scale that would never be possible with an on-premises solution.

2.5 Cloud migration challenges

Despite all the benefits that come with migrating to the cloud, they can only be achieved if the applications that are migrated are designed and developed with the cloud mindset and architecture in consideration [13].

This makes cloud transitioning challenging for a lot of companies who have a lot of legacy applications that are based on different principles from the ones used in the cloud. To make legacy applications better candidates to move into the cloud, most of the times it requires a rebuild of the underlying structure of the applications [3]. Since cloud computing requires a loosely coupled architecture, it is ideal for stateless services and applications since they do not have infrastructure dependencies. Most of legacy applications rely on stateful services, and the cost of making a transition into stateless can make the transition to cloud unfeasible.

One of the major concerns and challenges presented by the cloud is security [2]. But this can be

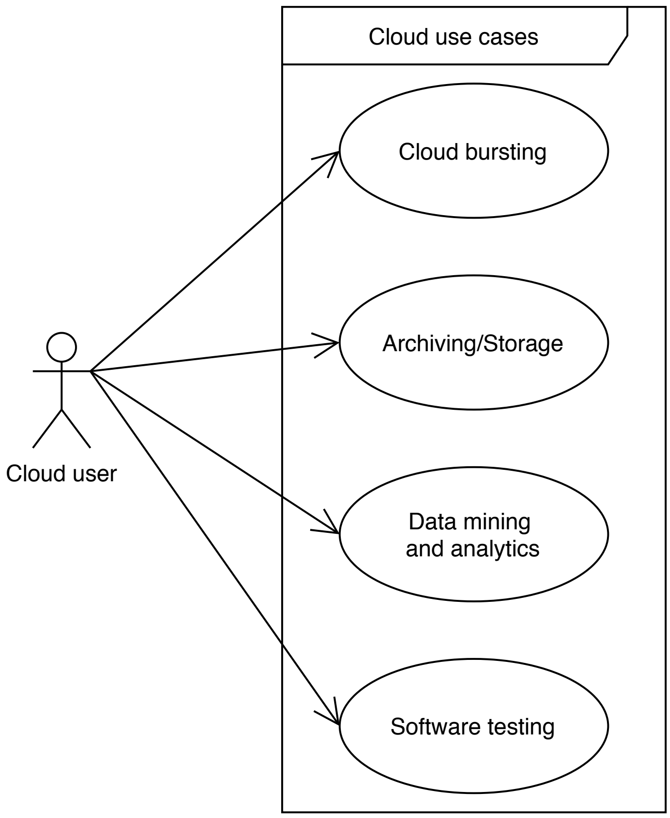


Figure 2.1: Popular cloud use cases

minimized and even prove to be a better security solution if the application is designed from the ground up with the cloud security architecture in mind.

Another challenge is the impact made in the organization when transitioning to cloud. Managing change and the people that work inside the company is difficult and poses as a risk, although this can be mitigated by starting with smaller and lower-risk initiatives as early candidates for cloud computing projects. This allows a paced adaptation process to the new business process, by giving time to new cloud specialized workers to teach and support the ones that are making the transition to the cloud model.

2.6 Multi-criteria decision-making

Multi-criteria decision-making consists of finding the best solution from a pool of possible candidates according to selected criteria, by basing the decision on mathematical and programming tools [14]. The process of applying MCDM techniques is made of several phases such as establishing the criteria to use in the evaluation, the number of alternative solutions one wants to decide upon, applying weights to the previously defined criteria, and then apply the MCD method itself. This whole process is more comprehensible in figure 2.2 [15].

There are two types of MCDM approaches that can be used: there is Multi-Attribute Utility Theory (MAUT) and outranking approaches [14]. MAUT consists in obtaining a function that represents the utility

Challenge	Description
On-demand self-service	Use of cloud computing requires a loosely coupled architecture where most legacy applications are not, therefore requiring major changes to be fit for the cloud
Security	Adopting cloud means giving up some of the controls and security mechanisms to the cloud provider, posing considerable risks
Organization impact	Changing to cloud also means managing change within the people that work in the organization, for the success of cloud adoption hinges on their adaptation to the new technology

Table 2.4: Cloud migration challenges

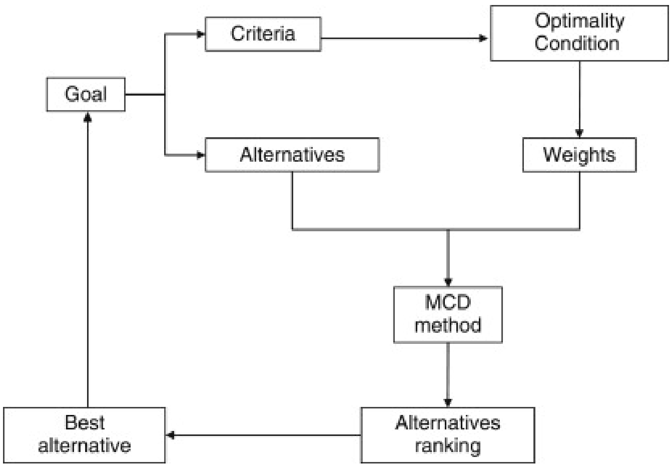


Figure 2.2: General structure of an MCDM process

of an alternative, where the alternative then attributes a marginal utility, with a number representing the preferability of the alternative. The total utility score results from the sum of all marginal utilities.

On the other hand, outranking approaches are based on pairwise comparisons between the aspects of the considered solutions to establish which solution is better than the other.

Within these two approaches, there are commonly used techniques such as AHP, ELECTRE, PROMETHEE, TOPSIS, and VIKOR.

- AHP is a technique that takes into account the mutual relationship between criteria, since it is not always possible to keep them all independent from each other in real-world problems [16]. It is based on pairwise comparisons of the criteria, where these are organized in a hierarchical relationship.

- ELECTRE is a technique that falls in the class of outranking methods, which consists of performing pairwise comparisons between alternatives and creates an outranking relationship among them [16]. The created relationship is then used to identify and exclude the alternatives that are dominated by other alternatives, reducing the total number of possible alternatives.

- PROMETHEE stands for preference ranking organization method of enrichment evaluation and is an outranking method just like ELECTRE, but unlike it, it stands as an improved technique over ELECTRE. The difference between the two lies in the pairwise comparison stage [16]. Instead of only verifying if an alternative is better than another, it also checks the degree to which an alternative is better or worse than the other.

- TOPSIS tries to select an alternative with the following two conditions: it needs to be the closest to the ideal solution while simultaneously being the farthest from the anti-ideal solution [16].

- VIKOR focuses on the ranking and selection from a set of alternatives in the existence of conflicting criteria [17]. It is similar to the TOPSIS method, the difference being that VIKOR looks to find the alternative that is closest to the ideal solution.

2.7 Enterprise architecture applied to the cloud

When developing an enterprise architecture for the cloud, there are significant considerations to be taken. The Open Group defined a reference model for the development of cloud-based architectures, providing guidance on how to apply them according to other existing standards such as the Open Group Architecture Framework (TOGAF) and ArchiMate [18].

One of the considerations is cloud deployment models, which were described in section 2.2, and affect the development of the architecture depending on which one(s) is/are adopted.

Another consideration is the specialized organizational roles that come with the adoption of a cloud architecture. One of these is the Cloud Service Administrator, responsible for administering cloud systems. Another one is the Cloud Services Strategist, which can be a person or a strategic business unit that develops cloud services strategies and provides guidance on how to transform the business into cloud services. There is also the Cloud Service Manager who provides guidance and direction for cloud computing efforts. This person leads a team to ensure consistency in the cloud computing business and service delivery models of an enterprise. Adding to the previously mentioned roles there is also the End User, this being the person who interacts with and uses a cloud service.

There are also security considerations to bear in mind such as: assuring the safety of intellectual-property as well as capital assets, the adaptation of security measures and protocols for cloud deployment, the management of confidential information and application of regulatory policy requirements, and the definition of an approach to facilitate policy-based service delivery.

Following this are the business considerations which consist of: allowing the chief information officer (CIO) to concentrate on business information and applications that give business value to stakeholders, the reduction of investments in IT infrastructure, the management of business processes in the cloud ecosystem, the integration capabilities and collaboration with suppliers, partners, and back-office. Adding to these there is also the consideration of the standardization of business processes for cost-effective and consistent use.

Then there are technical considerations which can be briefly described as common framework applications, robust integration capabilities, network and bandwidth, and working with a distributed environment.

Ultimately there are operational considerations such as operational excellence, cloud services operational management, workforce management, problem, and error resolution management, service-level agreements, licensing and contract management, cloud service subscription and life cycle management, cloud strategies for adoption, migration and exit and subsequently the capacity and monitoring of services.

Aside from all of the aspects mentioned above, one thing to take into account is the impact on application architectures themselves when transitioning to the cloud model [19]. By not having the responsibility of assembling and maintaining hardware that comes with the adoption of cloud, there is a change of focus in the ArchiMate Technology Layer. The transition goes from focusing on Active Structure Elements (Devices) to the Behavior Elements (Services and Functions) and Passive Structure Elements such as the Artifacts that constitute the computing payload.

2.8 Current state of cloud in the Portuguese public administration

The purpose of cloud adoption in the Portuguese public administration is to improve the quality of public services provided while achieving reduced costs for both citizens and companies and reducing public expenditure, particularly in Information and Communication Technologies (ICT) [1].

The Portuguese government identified two strategic sectors that focus precisely on cost reductions and in the implementation of common IT solutions across the public administration. An initiative was launched to rationalize the already existing data centers with the intent to fulfill the following objectives:

- Increase the profitability of the investments made in public infrastructure and data centers, resorting to the private cloud whenever it proves to be a more competitive solution
- Unification of existing data centers
- Provide guidance to promote the adoption of cloud computing in the public administration
- Assure the adoption of more agile and transparent solutions, as well as its management, with improved security, lower costs, and efficient use of resources

The results of the initiative showed the fragmentation of acquisition and management of infrastructure, with a lot of redundant and replicated systems. It was also noted that the use of open-source software and norms was lacking, and cloud adoption benefits were identified (see table 2.5).

2.9 Cloud adoption in public administrations

Other countries (some being part of the EU) have already made great progress in cloud adoption for their respective governments and public administration. Each one with a slightly different approach since the

Benefit	Description
Cost reduction	Low cost of both hardware and software that can be bought and managed according to the client's needs, allowing for a cheaper implementation of applications and services
Flexibility and agility	Infrastructure requirements can be scaled and adapted according to demand, which is specifically useful in certain times of the year where peaks of traffic are predicted to happen
Upfront investment cost reduction	Initial investments in hardware, software licensing and maintenance are no longer required
Updated software	Cloud service providers ensure the provided software is always up to date
Universal access	Applications can be accessed from any place and time, and from any device, allowing for remote work
Choice of applications	Applications can be tested according to the client's needs, allowing for a more informed decision, consequently avoiding costly upfront investments
Environmental and energetic efficiency	The cost of computational resources as well as energy is reduced when comparing to on-site solutions

Table 2.5: Cloud adoption benefits

existing conditions are specific to each country such as legislation, geopolitical status, previously built infrastructure, willfulness to change, and others. Additional information about each of the countries' cloud adoption drivers and objectives can be observed in table 2.6.

When it comes to Australia, particularly Western Australia, a certain emphasis was made on the reasons to adopt a public cloud model [20]. Some of these are disaster recovery, support of special events requiring large capabilities of infrastructure, and the cheaper upfront cost of the development of proofs of concept. It is also worth noting that the Australian government opted to keep the control and responsibility of ICT delivery agency independent, favoring a more decentralized approach to ICT in the public administration.

In the USA the development of a roadmap for cloud adoption among governmental agencies was carried out by NIST [21], covering all topics ranging from the establishment of priority requirements that need to be satisfied to transition further into cloud adoption, to the definition of standards, guidelines, and technology required for portability, security, and interoperability. This served as a basis for further development in both the implementation of cloud solutions in institutions and maturity of models and adopted frameworks.

The UK made it clear in its Cloud First policy to prioritize the adoption of public cloud solutions rather than other cloud deployment models [22]. Although there are situations where other deployment models would be more appropriate, the UK government believes that the main benefits of cloud adoption are achieved with the use of a public cloud model. The UK created the Government Cloud (G-Cloud) initiative, to make the process of acquiring cloud computing solutions easier for government departments. Another relevant project is a private network (Government Secure Intranet), interconnecting government departments and agencies up to a total of 600 organizations as of 2015.

Italy's process of cloud adoption in its public administration started with the merging and consolidation of existing government-owned data centers [23], a similar situation to the one of Portugal. After this process was concluded, it allowed for the use of cloud services whether they relied on the governmental cloud for services that manipulate more sensitive data or other non-critical data applications hosted on third-party cloud providers.

Spain's cloud adoption is interesting due to the already existing communication line infrastructure Red SARA (the Telecommunication Network of the Spanish Public Administration), that provided interconnectivity among public administration entities as well as connectivity with EU institutions [12].

2.10 Criteria used in cloud adoption evaluation

To establish the criteria for assessing cloud adoption feasibility within public administration institutions, a systematic literature review of existing cloud migration frameworks and their respective utilized criteria was conducted to analyze what solutions and parameters are the most popular and comprehensive.

The articles used in the literature review are present in journals and conferences obtained using search engines of publishers such as Springer, IEEE, ResearchGate, and Elsevier. Others were obtained using the Google Scholar search engine. It is also worth noting that only articles published

Country	Drivers	Objectives
Australia	Faster deployment of services More agile services Less maintenance	Increase adoption of cloud computing and value generated by organizations while maintaining adequate security
USA	Cost savings AI applications	Migrate public agencies to secure cloud infrastructure, using a revision of Cloud-First policy (Cloud Smart policy)
UK	Cost savings Simplicity in development and deployment of applications	Prioritization of the adoption of public cloud solutions with a Cloud-First policy
Italy	Cost savings	Rationalization of data centers
Spain	Cost savings Economic impact for the private sector Faster deployment of public services	Expansion and improvement of the SARA network in order to provide cloud services

Table 2.6: Cloud adoption drivers and objectives in other countries

starting in the year 2015 were considered, to constrain the number of results further as well as having the most recent research material on cloud computing adoption. The keywords used to conduct the search for relevant articles were a mix of the following: Cloud, Adoption, Migration, Framework, Criteria, Assessment, Tool, Feasibility.

After collecting around 30 possibly relevant articles using the method described above, a manual review was conducted to filter them further. Importance was given to those who presented literature reviews about cloud adoption criteria and frameworks as well as those that propose their own frameworks and decision support tools to be used or that extend already existing tools and frameworks. This resulted in a total of 13 relevant articles to be studied.

The analysis of the 13 articles that either specifically mention criteria used in the phases of cloud feasibility analysis/assessment or mention cloud migration goals that can be translated into criteria, was followed by the consolidation of the criteria and respective definitions. This was achieved by merging similar characteristics with similar definitions present in several articles. Some less popular criteria were also included due to their considered relevance to the work to be developed.

The first iteration of knowledge consolidation related to the criteria resulted in the following criteria set and definitions:

- The **complexity** of current systems was first derived from analyzing [4] (also appearing in [3, 7]), where this aspect allows to assess the degree of technical complexity of the current system (in the case of legacy migration) or of the new system that is to be implemented directly in the cloud.

In the case of legacy migration, this can be related to the concept of interoperability present in [3, 6, 24–26] referring to the level of application integration across multiple platforms, contributing to the level of complexity present/required in the application.

- **Compatibility** with current systems according to [4] is the level of modularity present in the current system and the ability to seamlessly interface with other applications that may not be in the cloud and that are already in use. This concept is also mentioned in [3, 24–29] even though in most of them it is not as clearly defined.

- **Availability and accessibility** as mentioned in [24, 30] refer to their respective levels of requirement by the system and the existence of redundancy measures within it, as well as the extent up to which the cloud providers can support these requirements. It also refers to the ease of access to the system from different devices [27]. These two aspects are sometimes combined and used as one [4, 6]. Both characteristics are related to what the cloud providers present as service level agreements (SLAs), since the requirements of availability of service and accessibility are some of the aspects covered in these agreements to give reassurance to cloud service providers' (CSP) clients while giving them a realistic view of what level of service quality to expect.

- **Security** as defined in [4] represents the security measures that are necessary in the system: if they need to be location-based, if the cloud provider is able to replicate the same measures such as protecting organization data and maintaining the same level of privacy and confidentiality of the data [24, 29]. Not only this but it is also important to assess the existing level of expertise in dealing with security threats whether it is from the current IT staff or expertise given by the cloud provider. This represents the major risk in cloud adoption [26], having only two articles that do not mention it directly. The article [30] also underlines the importance of having control mechanisms able to respond to security threats and incidents, including the existence of security policies as well as compliance with rules and regulations. Although the latter mentioned are not directly part of the technology that is being considered for migration, they are fundamental aspects to consider before the appliance of a system migration.

- **Portability** as mentioned in [6] refers to the disruption level that the system will suffer when migrated to cloud, or even between cloud providers. This criterion is related to the ease of decoupling the system from the underlying infrastructure of the cloud provider, therefore avoiding vendor lock-in.

- **Scalability** as defined in [4] is the ability to keep up with an increasing workload by incrementally increasing a proportionate amount of computational resources [27].

- **Trialability** can be portrayed as the adequacy and availability given by the cloud providers to try out their services before the actual use [24, 29].

- **Testability** is relevant due to the advantageous merging between agile development and cloud computing. Therefore, the level of ease or improvement in deployment speed when testing and developing a system [6] before putting it fully operational in the cloud can be considered as important factor when weighing the decision of cloud adoption/migration. Although not specifically mentioned, it can be included in the agility category mentioned in [28].

- Depending on the system and organization needs, **backup and recovery** are important factors when opting for cloud migration as seen in [26, 27].
- **Performance** as described in [6], is the throughput speed and the existing computational power, where its assessment consists in the validation according to the system requirements.
- **Elasticity** referenced in [27] is the ability to increase and decrease computational power in a simple and instantaneous manner according to the needs of the system.
- **Continuous monitoring** in only defined in [3] as the level of service provided when it comes to the ability to monitor the system in the cloud and cloud resources to assure SLA compliance.

2.11 Framework analysis

Several frameworks for conducting cloud adoption have been developed, each with its own approaches for the assessment of cloud feasibility. This section will be used to conduct a deeper analysis of a previous systematic literature review, to obtain a better comparison between what this work intends to achieve and the already existing methodologies, therefore highlighting similarities and differences between them.

One of these frameworks relies on a risk management approach to tackle cloud migration [30] by encompassing all dimensions of sustainability such as economic, environmental, social, and technology. These dimensions then possess a set of controls used to evaluate the risk in conjunction with the use of the Dempster Shafer Theory of Evidence belief functions.

Another framework that appeared on the review results suggests a brokerage model [31] for automating both the cloud service selection and deployment, focusing more on the first phase of the model which corresponds to the assessment phase. This phase then consists of five steps, starting with an analysis of the existing information system of which results in a document with all the current technologies used in the functioning of the system, ranging from software to hardware including operating systems and services. The next step in this phase involves the identification of the operational requirements from an organizational point of view. Following this is the security assessment, showing the importance of security issues and concerns that need to be taken into consideration before migrating to cloud. It is worth noting the magnitude of security of the system is so considerable to the point where an entire step of the assessment phase is dedicated to this topic. The assessment phase ends with the consideration of management costs to establish a base of comparison for the system when it is migrated to cloud, this step then being followed by the creation of a proof of concept to present visibility over problems that may or will arise before the transition.

The next framework taken into consideration derives from systematic research on existing work on cloud migration of legacy systems [7], taking particular focus on creating a secure migration model.

The result of this is a 5-Phase Cloud Migration Model, where the phases consist of: feasibility analysis, requirement analysis and migration planning, migration execution, testing and migration validation, monitoring and maintenance. The phase worth giving particular consideration is the feasibility analysis, being one of the phases that had the least amount of studies dedicated to it which shows an opportunity for the development of models and mechanisms to go into greater depth into the feasibility analysis.

A similar study on already existing frameworks and cloud adoption models is present in [25], resulting in a migration framework consisting of six steps: initiation, adoption, decision making and selection, migration, adaptation and control, and routinization and maintenance. Both phases of adoption and decision making are based on multiple articles of work developed where it mentions the importance of understanding the applications being subjected to migration and evaluating the respective requirements. Although these aspects are covered in a broad manner, there are no mentions of existing tools or mechanisms that help with the required analysis during these phases, something that possesses opportunities for expansion.

Unlike the aforementioned frameworks, we intend to introduce our tool as a simpler mechanism to aid in phases where an extensive analysis of the current system is required and where multiple technological aspects need to be taken into consideration before deciding whether or not the system in question is suitable for cloud migration.

2.12 Decision-making approaches to cloud adoption

One of the already existing decision-making approaches to cloud adoption uses multi-layer cognitive maps, using the FCM (fuzzy c-means) technique which combines neural networks with fuzzy logic [27]. The article in question presents a complete and tested model, having been submitted to several synthetic and real-life case scenarios to evaluate the effectiveness and usability of the model.

Another approach relies on BOCR (benefit, opportunity, cost, risk) analysis as the criteria base for the application of the AHP (analytic hierarchy process) multi-criteria decision-making algorithm [28]. The use of BOCR allows the use of a criteria set that represents the general ecosystem affected by the decision of cloud adoption, while on the other hand, it does not provide a more granular representation of the technological aspects of the system being submitted to cloud adoption decision.

Both of the approaches mentioned above consist only in model form, not providing a simple and usable tool implementation of the models.

One approach that was implemented as a tool is TradeCIS, using a trade-off based decision system [32]. This decision system utilizes the TOPSIS algorithm from a technical perspective of decision-making, and the ANP algorithm (a generalization of the analytic hierarchy process algorithm) for the business, economical, and organizational perspectives. Although the article mentions a developed and tested prototype of the tool, it does not present a general set of criteria that can be used as a basis for the cloud adoption decision process.

Chapter 3

Cloud assessment tool

This chapter starts with an extensive description of the developed solution in section 3.1, followed by a description of the use cases of the tool provided in section 3.2. In section 3.3 the representation of data is provided as well as how it interacts with the tool. The explanation of the final criteria set used as a basis for the tool along with their respective definitions is presented in section 3.4. After this, the question set that was created based on the criteria set to evaluate the criteria is presented in section 3.5. The explanation in detail of how the tool scoring system works is given in section 3.6. The chapter ends with section 3.7 where the solution architecture is presented as well as the underlying development infrastructure chosen to develop our work.

3.1 Solution description

To fulfill the objectives shown in section 1.3 and tackle the problem that is the cloud feasibility assessment of a system in a public administration environment, this work suggests the creation of a tool that simplifies the process. In doing this, it also makes the process more accessible to decision-makers that do not possess an expert level of cloud knowledge, while still requiring some basic grasp of the cloud ecosystem. This tool is comprised of two parts: a questionnaire and the calculation of the final recommendation score by resorting to an MCDM approach. It is important to note that this tool is focused on the IaaS public cloud service model, compared to a private on-site hosting approach.

The questionnaire takes as basis a refined criteria set obtained in previously developed work. Not only this allows us to display the criteria in a simpler manner, but we are also able to derive guide questions from their respective definitions, which allows us to build the questionnaire. This questionnaire in conjunction with the definition of scales to be used in its answers described in section 3.6, allows us to build a scoring system that is then used to obtain the values for the comparison between alternatives (on-premises or cloud hosting) for each of the criteria. A visual example of the questionnaire (in this particular case for the Trialability criterion) can be observed in figure 3.3.

The second part of the tool involves the use of MCDM to calculate a recommendation score for each of the alternatives (on-premises or cloud hosting). The MCDM algorithm used in this case is TOPSIS, an algorithm that was explained in section 2.6. This algorithm was chosen due to the way it works, in trying to find the best alternative that is both the closest to the ideal solution and the farthest from the anti-ideal solution. Not only this but it is also a relatively easy algorithm to implement, which in this case proved to be a plus. By applying the obtained information in the questionnaire in conjunction with the criteria weights required to be input, the tool is then able to feed all of this data to the MCDM algorithm, retrieving then the recommendation score for each of the specified alternatives.

The questionnaire can be submitted empty in the case of the user possessing a more detailed level of the system and the cloud environment as to feel confident enough to attribute the scores of each criterion for each of the alternatives manually. Whether the questionnaire was submitted empty or not, the user is then redirected to the calculation page. Here the interface shows the table already pre-filled (in the case of answering the questionnaire), where in the next step the user is asked to determine the importance of each criteria comparing them between one another. To aid in this weighting process, the criteria definitions are also provided to inform the user as much as possible as to obtain the most exact and valuable data possible. After the previous steps are complete, the user then submits all of the data and the tool then calculates and shows the recommendation scores for each of the alternatives.

3.2 Use cases

With this tool, we not only intend to provide a simpler process to evaluate the cloud feasibility of a system but also allow it to be customizable to the problem that the user might be trying to solve. This can be done whether by adding/deleting or editing the currently present criteria set and respective definitions, or by adding new possibly more complete or relevant questions for a criterion, or even edit the current ones or deleting ones that may not be relevant. With this level of customization, we allow the user to shape the tool around the problem to better evaluate the system at hand so that the results may be the most accurate and create the most value.

The user can also generate the numerical scores for each pair of alternatives and criteria by answering a questionnaire. This is made to facilitate the quantitative measurement of each criterion for the alternatives at hand through a qualitative process. In doing so we provide additional context and description to each criterion through questions, allowing for a clearer understanding of each criterion and what particular aspects of the system are being evaluated.

If the tool is being used by someone more knowledgeable about the cloud ecosystem, where the answering of the questionnaire may not be that relevant anymore, we allow the user to skip it entirely if needed. Whether the questionnaire is skipped or not, the user still needs to determine the weight of each criterion in the final table (by assessing which of the criteria has more or less priority in the determination of cloud feasibility).

After all the inserted data, the user can then press a button to submit the values mentioned before and obtain the calculated recommendation score for each of the alternatives that comes from the application of the MCDM algorithm.

All of the use cases mentioned above can be observed in figure 3.1.

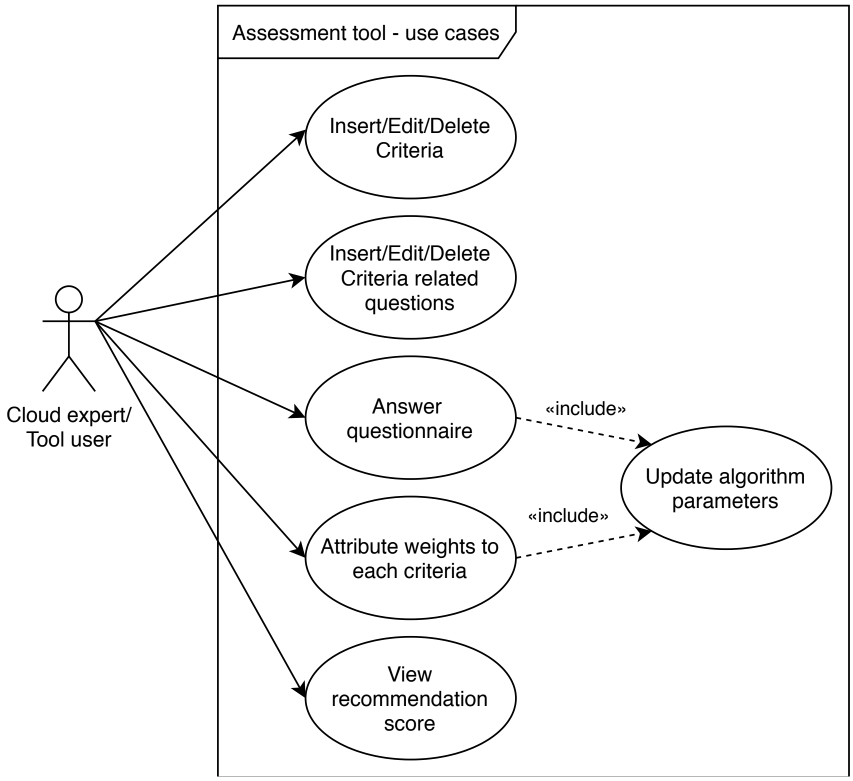


Figure 3.1: Use case diagram of the cloud assessment tool

3.3 Data representation

The data used to store in the tool database to work refers mainly to the criteria and questions used at the beginning of the tool, which can be visualized in the UML representation in figure 3.2

Each criterion has a name, its respective definition/description, and a weight and impact. The impact of a criterion refers to the effect it has on the evaluation of a system, in a way that essentially identifies if the criterion represents either a benefit or a cost towards the adoption of cloud. The weight is only attributed when the recommendation calculation is performed, therefore having its own "setting" function.

Each question has the question text which identifies the question and the respective answer options. As these are only currently accepted in the form of a yes/no answer or a five-point Likert scale, the additional validation method is used to ensure these constraints.

3.4 Determination of relevant criteria set to use and respective definitions

We took advantage of the previously conducted systematic literature review in section 2.10, taking it a step further by refining the criteria definitions and merging some of them where it made sense to do so. The end product ended up being the following criteria:

- **Complexity** represents the overall technical complexity of the current system (in the case of legacy migration) or of the new system that is to be implemented directly in the cloud. This complexity

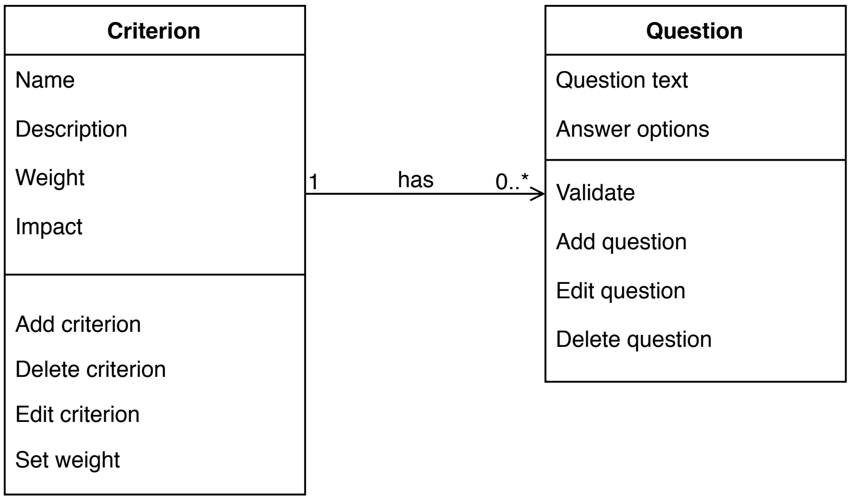


Figure 3.2: UML representation of the data used in the tool

can include both underlying code as well as infrastructure. In the case of legacy migration, it must also be considered the level of application integration across multiple platforms, which contributes to the level of complexity present/required in the application.

- **Compatibility** represents the level of modularity present in the system and the ability to seamlessly interface with other applications that may not be in the cloud and that are already in use.

- **Availability** represents both the availability and accessibility requirements by the system, the existence of redundancy measures within it, as well as the extent up to which the cloud providers can support these requirements. It also refers to the ease of access to the system from different devices. Since these characteristics are usually present in the SLA (service level agreement), this is also to be included in the consideration of this criterion.

- **Security** represents the security measures that are necessary for the system as well as the following characteristics: if the measures need to be geography-based (due to data sovereignty) if the cloud provider is able to replicate the same measures such as protecting organization data and maintaining the same level of privacy and confidentiality of the data. Not only this but it is also the existing level of expertise in dealing with security threats whether it is from the current IT staff or expertise given by the cloud provider. Another aspect to include in the consideration is the importance of having control mechanisms able to respond to security threats and incidents, including the existence of security policies as well as compliance with rules and regulations. Although the latter mentioned are not directly part of the technology that is being considered for migration, they are fundamental aspects to consider before the appliance of system migration. Another aspect to consider is backup and recovery. Continuous monitoring can also be included in security since it is an important function and a great tool to have to detect threats or unusual behavior within a system. One more aspect is the ability to monitor the system in the cloud and cloud resources to assure SLA compliance.

- **Portability** represents the disruption level that the system will suffer when migrated to cloud, or even between cloud providers. This criterion is related to the ease of decoupling the system from the underlying infrastructure of the cloud provider, therefore avoiding vendor lock-in.

- **Elasticity and Scalability** represents the ability to increase and decrease computational power in a simply and instantaneously according to the needs of the system. This criterion also represents the ability to keep up with an increasing workload by incrementally increasing a proportionate amount of computational resources.

- **Trialability** represents the adequacy and availability given by the providers to try out their services before the actual use.

- **Testability** represents how easy and quick it is to deploy a product or application when testing and developing a system before making it fully operational in the cloud.

- **Performance** represents the throughput speed and the existing computational power, where its assessment consists in the validation according to the system requirements.

We considered best to merge the security criteria with both continuous monitoring and backup and recovery since these are parts that directly influence the security of a system, by both being able to effectively and simply monitoring all the resources and current states of the system as well as the communications with other systems. Both backup and recovery work more as measures to attenuate consequences when something goes wrong (whether its a security breach or hardware failure), but are directly connected with the ability to respond to threats if they ever emerge,

The accessibility and availability criteria were both merged to create a more solid criterion while incorporating both definitions in its definition, something similar to where the SLA component was inserted in this criteria, for it encompasses both of these aspects.

Both criteria of elasticity and scalability were merged since these two aspects always go hand in hand in a system, despite having different definitions, we thought they were better placed as part of a more general criterion, while still evaluating both the components it is made out of.

3.5 Elaboration of question set to represent the criteria

To simplify the process of cloud adoption assessment as to make it available for decision-makers that do not possess an expert level of cloud knowledge, we took the criteria set present in the previous section and used its definitions to create a set of questions to be used in the elaboration of a questionnaire which can be observed in 3.1. This allows for a simple and representative approach when evaluating each of the criteria against the system. This questionnaire intends to achieve a balance between covering as much of the definition as possible of each of the criteria, while still being able to be applied to the majority of systems and cases that the tool may be subjected to.

Criteria	Questions
Complexity	What is the dependency of the system's code/functionality on the current underlying infrastructure? How difficult is it to integrate the system with additional platforms? What is the level of comfort/knowledge of your staff to work on a cloud environment?
Compatibility	What is the current level of modularity in the system? How important is it for the system to able to seamlessly interface with other existing applications that may or may not be in the cloud, and that are being currently in use? Is the system a legacy system (i.e. an outdated system that is still in use)?
Availability	How important is it for the system to be able to be accessed from multiple different devices? How important is it for the system to be consistently available to be used? How important is it for the system to be continually available to be used? How important is it to have technical support constantly available (99,9%)? How important is it to have remote access to the system?
Security	How important is it to have full control and oversight over all the data used by the system? How important is it to have control over all the required security mechanisms for the system? How important is it to perform a consistent and periodical backup of the system? (and easily accessible and available) How important is it to have the ability to constantly monitor the system and its consumed resources? How important is it to have the system run on always up-to-date underlying software?
Portability	How much disruption would the system suffer if required to migrate to another platform/cloud? How difficult is it to decouple the current system from the underlying infrastructure?
Elasticity & Scalability	How important is it for the system to be able to keep up with varying workload? How important is it for the system to efficiently adapt the used computational resources according to its needs?
Trialability	How important is it to try out provider services and capabilities before actual use?
Testability	How important is it to easily be able to test the system in both development and production environments? How important is it to be able to quickly test and deploy the system? How important is it to test the system with a high amount of computational and network resources?
Performance	How important is the access to high-quality and modern hardware to have the best performance possible for the system? How important is it to have access to a high-quality and high-performance network/internet connection for the system? How important is it to have updated hardware to maintain or improve performance?

Table 3.1: Questions used in the evaluation of criteria

Answer option	Attributed score for: Cloud hosting	On-site hosting
Not at all important	0	5
Slightly important	0	2.5
Moderately important	0	0
Very important	2.5	0
Extremely important	5	0

Table 3.2: Scoring example of question to assess the Trialability criterion

3.6 Answer scales and scoring system

To be able to translate and build a numerical score from the answers to the questionnaire, we opted for the most simple and straightforward solution. By resorting to a five-point Likert scale as well as a uniform question structure, we are able to deduce the score to add for each alternative (on-premises or cloud hosting). However, since the number of questions for each criterion can be different we end up calculating the average score for the questions for each criterion to obtain the value for the alternatives' score. In this case, we chose a maximum score for each combination of criteria and alternative of five.

For example in figure 3.3, assuming we are only using that question to evaluate the trialability criterion, we have five answer options. Depending on the answer given, a score is added to either the cloud adoption alternative or the hosting on-site alternative in a cumulative manner. The attributed scores for the five-point Likert answers can be visualized in table 3.2. When the answers are submitted the tool then normalizes the trialability score by dividing the cumulative added score by the maximum score defined previously (in this case five).

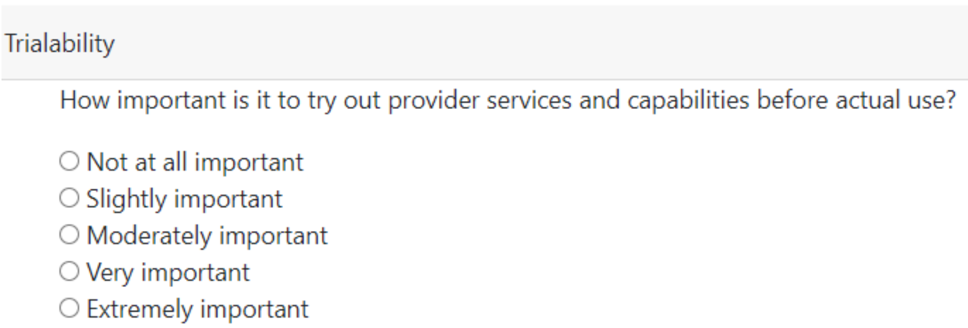


Figure 3.3: Example of question used to assess the Trialability criterion

3.7 Solution architecture

As it was mentioned in section 3.1 the cloud assessment tool consists of two parts: a questionnaire and the multi-criteria decision-making algorithm responsible for the calculation of the cloud assessment recommendation scores for each of the alternatives (on-site hosting or public cloud). A more detailed representation of the interaction between the tool components can be observed in the application layer diagram in figure 3.4. The calculation of the recommendation scores takes as input the data provided by the user, this being done in two ways: the answering of the questionnaire mentioned in section 3.5 and the attribution of the comparative weights for each of the criteria. With both inputs obtained, it is then possible to feed this data to the multi-criteria decision-making algorithm. The algorithm then produces the recommendation scores which will be displayed to the user. A representation of the interaction between the components of the tool can also be observed in figure 3.5.

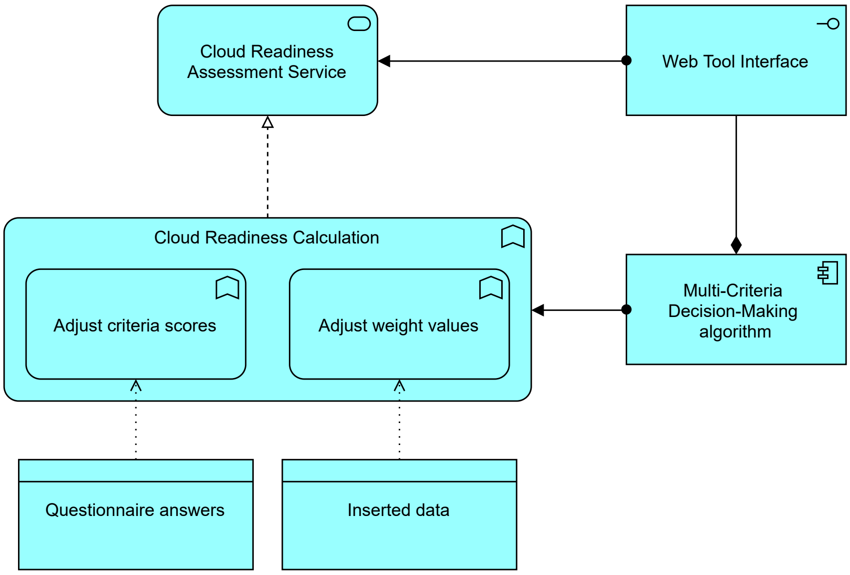


Figure 3.4: Application layer of the cloud assessment tool

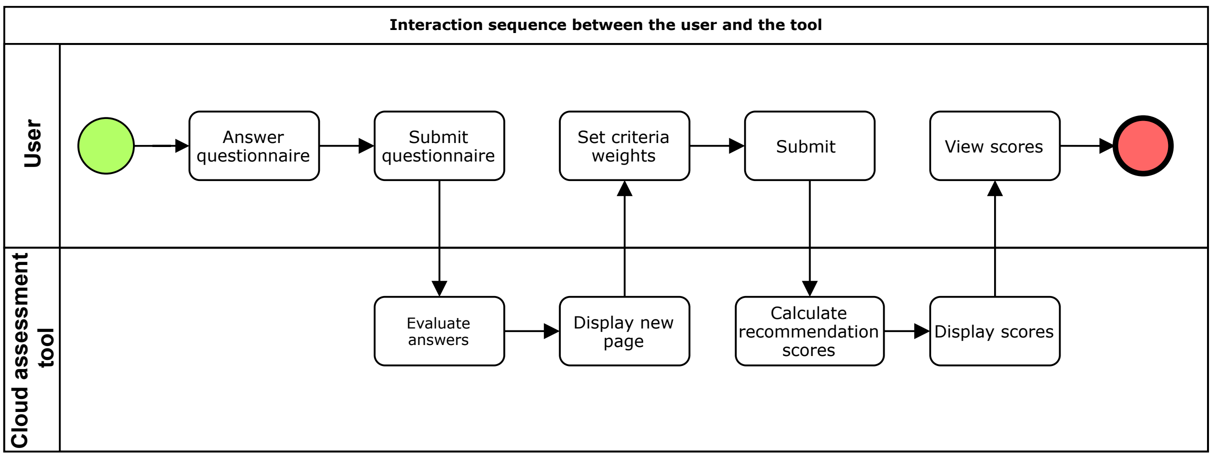


Figure 3.5: Business process diagram of the interaction between the user and the tool

The chosen technology in which to develop the tool was the Django framework in conjunction with the Python programming language. The reasoning behind this decision relies in the advantages that Django brings into web-development such as simplicity, scalability, flexibility and reliability. In having as its base the Python language, it allows for an easier understating and use of the syntax. Another great feature of Django is what its called a CRUD interface, where CRUD stands for Create, Read, Update and Delete. This not only facilitates communication between the back-end and database systems, but it also provides a back office right out of the gate, even if very simple in order to establish a baseline user experience without the need of extra coding. All of the aspects mentioned above make for a compelling reason to use Django to develop our tool, whether if it’s to create a proof of concept or a more completely developed end product.

Chapter 4

Evaluation and Results

This chapter starts with a detailed explanation of the evaluation process of this work in section 4.1. With this laid out, we then use the next sections 4.2 to present the case used to evaluate our tool. In section 4.3 we provide a walkthrough of the developed tool and how to navigate it, which is followed by the evaluation results obtained and the respective analysis in section 4.4.

4.1 Evaluation methodology

The evaluation process for this work will be primarily based on the creation of representative cases on which the tool would be used in a real-world application. These cases will describe a system with some technical aspects and also provide some context of their use and purpose. These created evaluation cases will then be distributed to a/some cloud experts who will conduct a manual cloud assessment approach (like they would normally do without resorting to the tool), and will then apply the same process but by using the tool in order to establish a base of comparison between the process that is currently used to assess the cloud feasibility of a system and our tool. After this, a survey is distributed in order to collect some feedback on the tool as well as additional information that may be useful to make the comparison.

4.2 Evaluation case

For the evaluation of our tool, we used a real cloud migration tool from AMA (the Portuguese agency for administrative modernization). This case consists of a new system, which already has the cloud paradigm in mind, with the goal of creating a unique platform to provide support to citizenship and participatory procedures within local and municipal governments. This application is based on blockchain technology as the underlying system for the attribution of votes to citizens. This project is created with the intent of having both production and pre-production environments hosted in the cloud, each with its own hardware requirements. This new system is also to be integrated with the existing governmental authentication application.

For the pre-production environment the solution is based on Apache HTTP Server, Tomcat, MariaDB, and MongoDB, which make use of Docker containers in order to allow for easier expansion to cloud-based environments. This makes it so that the application possesses both high scalability as well as a deployment with continuous delivery and development. To facilitate the administration of the MariaDB database, there are also Adminar containers. This brings it up to a total of five Docker images, which will be running on the same host. A visualization of the system setup is provided in figure 4.1. When it comes to hardware requirements, there is a baseline minimum to support the application, consisting of:

- CPU: with at least 8 cores
- RAM: with at least 16 GB
- Storage: at least 120GB of SSD storage space
- Operating System: CentOS 7

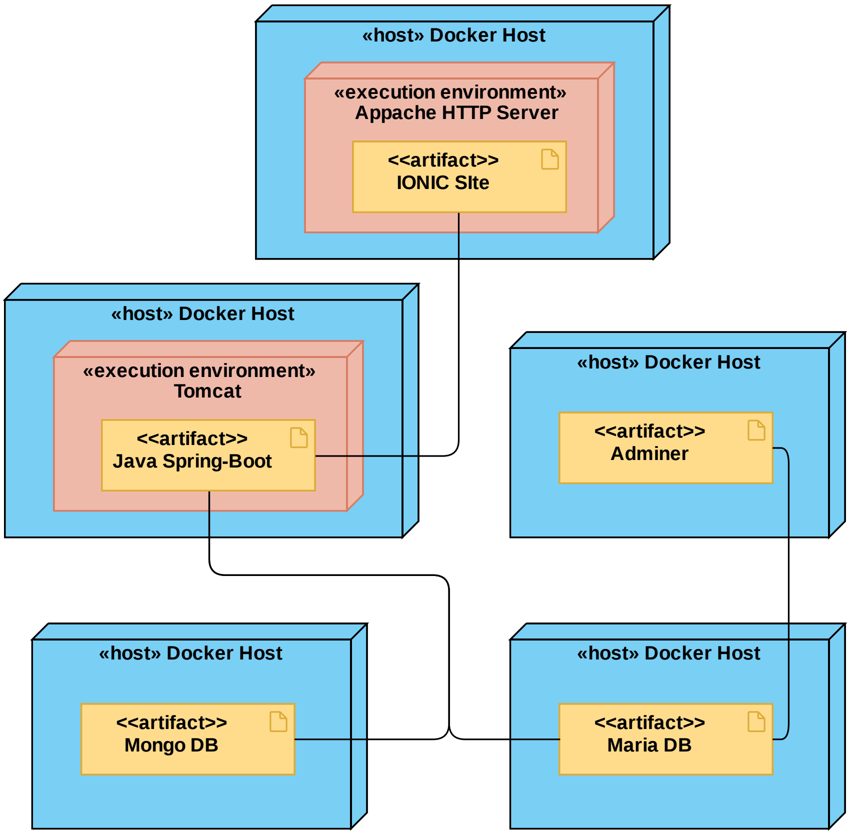


Figure 4.1: Pre-production deployment architecture

As for the production environment, the base of the solution is similar to the pre-production environment, by using Apache HTTP Server, Tomcat, MariaDB, and MongoDB. There is the addition of a load balancer prior to entering the application, where the amount of requests is distributed by the machines in the environment. This load balancing system is based on the Round-robin algorithm. Instead of a single host machine used in the pre-production environment, here we have a total of six host machines:

- Two Frontends

Host machine(s)	Hardware requirements
Frontend (1 and 2)	CPU: at least 8 cores RAM: at least 16GB Storage: at least 40 GB of SSD storage OS: CentOS 7
DB1 and DB2	CPU: at least 8 cores RAM: at least 16GB Storage: at least 100 GB of SSD storage OS: CentOS 7
DB3 and DB4	CPU: at least 8 cores RAM: at least 16GB Storage: at least 300 GB of SSD storage OS: CentOS 7

Table 4.1: Hardware requirements for host machines of production environment

- Two Relational Database Systems
- Two Document-oriented Database Systems

These host machines will work in two sets of three (one frontend, one relational database, one document-oriented database) to attain load distribution and service availability. A visualization of the system setup is provided in figure 4.2. For the production environment, the hardware requirements differ slightly over the pre-production one, as can be observed in the following table 4.1.

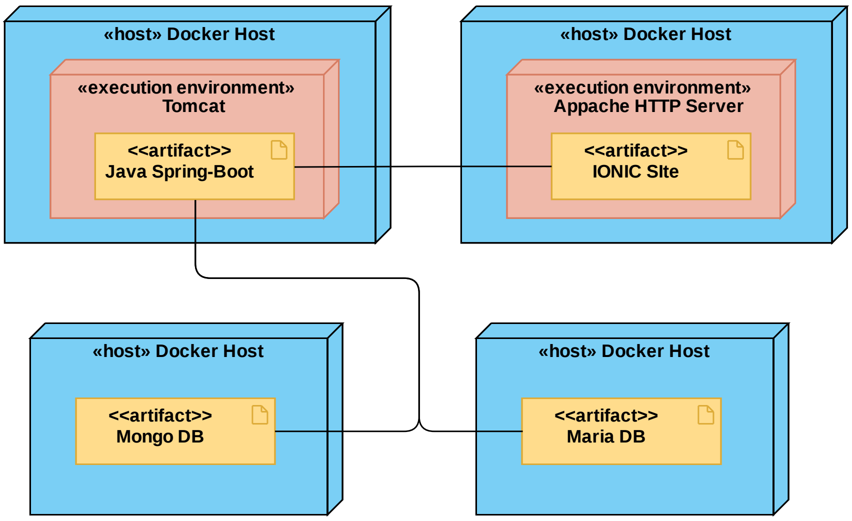


Figure 4.2: Production deployment architecture

4.3 Use of the tool

With all of the tool components explained previously in chapter 3 and the respective development being done, we will use this section to display the end result before going over the result of the testing and application of the tool on a real case.

At the beginning of the front page of the tool, we make a brief introduction to the tool as well as provide some guidance to start using it as can be seen in A.1. We then display the questionnaire, showing the set of questions pertaining to each of the criteria, which in this case starts with availability. At the end of the front page, we then give the ability to whether add or edit criteria and respective questions, which will take you into the Django admin page that allows you to make the aforementioned changes. This page can be seen in figure A.4. At the end of the front page, there is also the option to submit the questionnaire with the "Submit" button. After pressing this button we are taken into the multi-criteria decision-making page, which can be observed in figure A.3. In here, further guidance is given on how to proceed to the calculation of the results and additional input needed to achieve it. If the questionnaire of the previous page was not submitted empty, the "Alternative" columns of the MCDM page should be prefilled according to the submitted answers. The remaining input to be provided is on the "Weights" column, where the user uses the sliders to attribute the importance of each weight for the analysis when making a comparison between them. After this, all that is needed is to click on the "Submit" button and the row of scores will be filled with the scoring of each alternative, where a higher score means a higher level of recommendation for that alternative. In addition to this, an informative message is also displayed to make the result clearer which can be seen in figure A.5. We also present the definitions of each criterion further down the page to provide information to produce the most informed input possible.

4.4 Obtained results and analysis

After sending the tool to be tested to a cloud expert, the end results produced by the tool can be seen in figure A.5. Although the result coincided with the one produced from the usual approach taken by the expert to assess the cloud feasibility of a system, in this case, the application at hand ended not being approved for cloud migration due to budgeting constraints. The tester of the tool reported that both the interface and information used were relatively easy to use and understand.

By analyzing figure A.5 we can observe that the criteria that were given the most relevance, in this case, are availability and security. These aspects seem to prove most crucial when considering cloud migration: availability for the ability to have the system continuously available and in working condition, and security to assure the data security and confidentiality that goes through the system. It can be also noted the least relevant criteria in comparison, these being trialability, complexity, and testability as a close third. In the end, we observe that the cloud-based alternative has a score of 0.564 while the on-site alternative has a score of 0.436, resulting then in the recommendation of cloud-based hosting as the suggested alternative for the evaluated system.

Chapter 5

Conclusions

In this chapter, we will go over the achievements we were able to obtain with this work in section 5.1. After this, we go over the limitations and shortcomings in our work in section 5.2. We end with some insights of future development opportunities that can be expanded beyond this work in section 5.3.

5.1 Achievements

One of the achievements reached with this work was the consolidation of knowledge that allowed us to produce a generally applicable set of criteria that can be used to assess the cloud feasibility of a system when it comes to technological factors. This criteria set allows to cover the whole spectrum of technological aspects that a system can present and require the respective evaluation.

The other achievement would be the consolidation of the criteria set with the multi-criteria decision-making approach, enabling then the creation of a tool that can be applied in the early stages of cloud adoption/migration decision. Whether it is used by cloud experts to simplify the process and confirm their own analysis or by decision-makers that do not possess such a high level of knowledge of the cloud environment but are still able to grasp key concepts so that the information contained in the tool is enough to allow them to produce an evaluation over the feasibility of cloud of a given system/application.

5.2 Limitations

Although there were some insightful achievements obtained with this work, there are also some aspects that could have gone better or could have come out improved. The fact that there was only one multi-criteria decision-making algorithm used, leaves no comprehensible basis to compare to what may have been different results produced by other algorithms. This would have enriched our work significantly. Another limitation to point out is the usability of the tool, where by being tested over a single case (albeit a real life scenario) it is still not enough to conclude the true value that the tool can bring to the cloud adoption/migration process.

5.3 Future Work

Envisioning what future directions further development of the tool, there are many possibilities to take into consideration. Starting with testing of additional multi-criteria decision-making algorithms to verify differences and maybe determine which ones are better suited for a cloud feasibility analysis or for what type of system being subjected to evaluation in said analysis. Implementation with an AI component that would not only allow for the tool to learn from previous systems that were subjected to cloud migration and cloud feasibility, but it would also allow for another factoring component to be weighed in the evaluation.

Bibliography

[1] A. A. para a Modernização Administrativa. *Sumário da Estratégia de Racionalização dos centros de dados e computação em nuvem na Administração Pública*, 2012 (accessed October 9, 2019). URL https://www.ama.gov.pt/documents/24077/29210/Estrategia_Racionalizacao_Comunicacoes.pdf/001b4c63-f24a-4462-a9f4-1ff457082252.

[2] T. Halabi and M. Bellaiche. Evaluation and selection of cloud security services based on multi-criteria analysis mca. In *2017 International Conference on Computing, Networking and Communications (ICNC)*, pages 706–710. IEEE, 2017.

[3] M. F. Gholami, F. Daneshgar, G. Low, and G. Beydoun. Cloud migration process—a survey, evaluation framework, and open challenges. *Journal of Systems and Software*, 120:31–69, 2016.

[4] D. Ray. Cloud adoption decisions: Benefitting from an integrated perspective. *Electronic Journal Information Systems Evaluation Volume*, 19(1), 2016.

[5] M. M. Rajaeian, A. Cater-Steel, and M. Lane. A systematic literature review and critical assessment of model-driven decision support for it outsourcing. *Decision Support Systems*, 102:42–56, 2017.

[6] M. Fahmideh and G. Beydoun. Reusing empirical knowledge during cloud computing adoption. *Journal of Systems and Software*, 138:124–157, 2018.

[7] R. Rai, G. Sahoo, and S. Mehfuz. Exploring the factors influencing the cloud computing adoption: a systematic study on cloud migration. *SpringerPlus*, 4(1):197, 2015.

[8] A. Khajeh-Hosseini, I. Sommerville, J. Bogaerts, and P. Teregowda. Decision support tools for cloud migration in the enterprise. In *2011 IEEE 4th International Conference on Cloud Computing*, pages 541–548. IEEE, 2011.

[9] R. Chard, K. Chard, B. Ng, K. Bubendorfer, A. Rodriguez, R. Madduri, and I. Foster. An automated tool profiling service for the cloud. In *2016 16th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid)*, pages 223–232. IEEE, 2016.

[10] K. Ramchand, M. B. Chhetri, and R. Kowalczyk. Towards a flexible cloud architectural decision framework for diverse application architectures. In *The 28th Australasian Conference on Information Systems,(ACIS2017), WestPoint Tasmania, Hobart, Tasmania*, pages 4–6, 2017.

[11] P. Mell and T. Grance. The nist-national institute of standards and technology-definition of cloud computing. *NIST Special Publication*, (800-145), 2011.

[12] A.-M. Zaharia-Rădulescu and I. Radu. Cloud computing and public administration: approaches in several european countries. In *Proceedings of the International Conference on Business Excellence*, volume 11, pages 739–749. Sciendo, 2017.

[13] A. Mazrekaj, I. Shabani, and B. Sejdiu. Pricing schemes in cloud computing: an overview. *International Journal of Advanced Computer Science and Applications*, 7(2):80–86, 2016.

[14] M. Abdel-Basset, M. Mohamed, and V. Chang. Nmcda: A framework for evaluating cloud computing services. *Future Generation Computer Systems*, 86:12–29, 2018.

[15] S. Brown, R. Tauler, and B. Walczak. *Comprehensive chemometrics: chemical and biochemical data analysis*. Elsevier, 2020.

[16] Z. ur Rehman, O. K. Hussain, and F. K. Hussain. Iaas cloud selection using mcdm methods. In *2012 IEEE Ninth international conference on e-business engineering*, pages 246–251. IEEE, 2012.

[17] S. Opricovic and G.-H. Tzeng. Compromise solution by mcdm methods: A comparative analysis of vikor and topsis. *European journal of operational research*, 156(2):445–455, 2004.

[18] TOGAF. *The Open Group Cloud Ecosystem Reference Model*, (accessed October 16, 2019). URL http://www.opengroup.org/cloud/cloud_ecosystem_rm/index.htm.

[19] T. Karttunen. *Presenting a Decentralized Cloud Architecture with ArchiMate*, 2019 (accessed December 29, 2019). URL http://www.opengroup.org/cloud/cloud_ecosystem_rm/index.htm.

[20] AUG. *Digital WA : information and communications technology (ICT) strateg 2016-2020*, 2018 (accessed October 23, 2019). URL https://www.wa.gov.au/sites/default/files/2018-06/DigitalWAStateICTStrategy.pdf.

[21] U. Further. Us government cloud computing technology roadmap.

[22] UKG. *Government cloud first policy*, 2017 (accessed October 25, 2019). URL https://www.gov.uk/guidance/government-cloud-first-policy#public-cloud-first.

[23] ITG. *Three-Year Plan for ICT*, 2019 (accessed October 23, 2019). URL https://docs.italia.it/media/pdf/pianotriennale-ict-doc-en/stabile/pianotriennale-ict-doc-en.pdf.

[24] A. Alkhalil, R. Sahandi, and D. John. An exploration of the determinants for decision to migrate existing resources to cloud computing using an integrated toe-doi model. *Journal of Cloud Computing*, 6(1):2, 2017.

[25] H. reza Bazi, A. Hassanzadeh, and A. Moeini. A comprehensive framework for cloud computing migration using meta-synthesis approach. *Journal of Systems and Software*, 128:87–105, 2017.

[26] A. Mahajani, V. Pandya, I. Maria, and D. Sharma. A comprehensive survey on extractive and abstractive techniques for text summarization. In *Ambient Communications and Computer Systems*, pages 339–351. Springer, 2019.

[27] A. Christoforou and A. S. Andreou. A multilayer fuzzy cognitive maps approach to the cloud adoption decision support problem. In *2015 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–8. IEEE, 2015.

[28] S. Khan, M. S. A. Khan, and C. S. Kumar. Multi-criteria decision in the adoption of cloud computing services for sme's based on bocr analysis. *Asian Journal of Management Research*, 5(4):606–619, 2015.

[29] F. Mohammed, O. Ibrahim, M. Nilashi, and E. Alzurqa. Cloud computing adoption model for e-government implementation. *Information Development*, 33(3):303–323, 2017.

[30] A. A. L. Abdul Rahman, S. Islam, C. Kalloniatis, and S. Gritzalis. A risk management approach for a sustainable cloud migration. *Journal of Risk and Financial Management*, 10(4):20, 2017.

[31] H. Malouche, Y. B. Halima, and H. B. Ghezala. Enterprise information system migration to the cloud: Assessment phase. In *2016 IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA)*, pages 1–6. IEEE, 2016.

[32] R. Garg, M. Heimgartner, and B. Stiller. Trade-offs based decision system for adoption of cloud-based services. In *International Conference on Cloud Computing and Services Science*, pages 65–87. Springer, 2016.

Appendix A

Additional figures

A.1 Tool screenshots

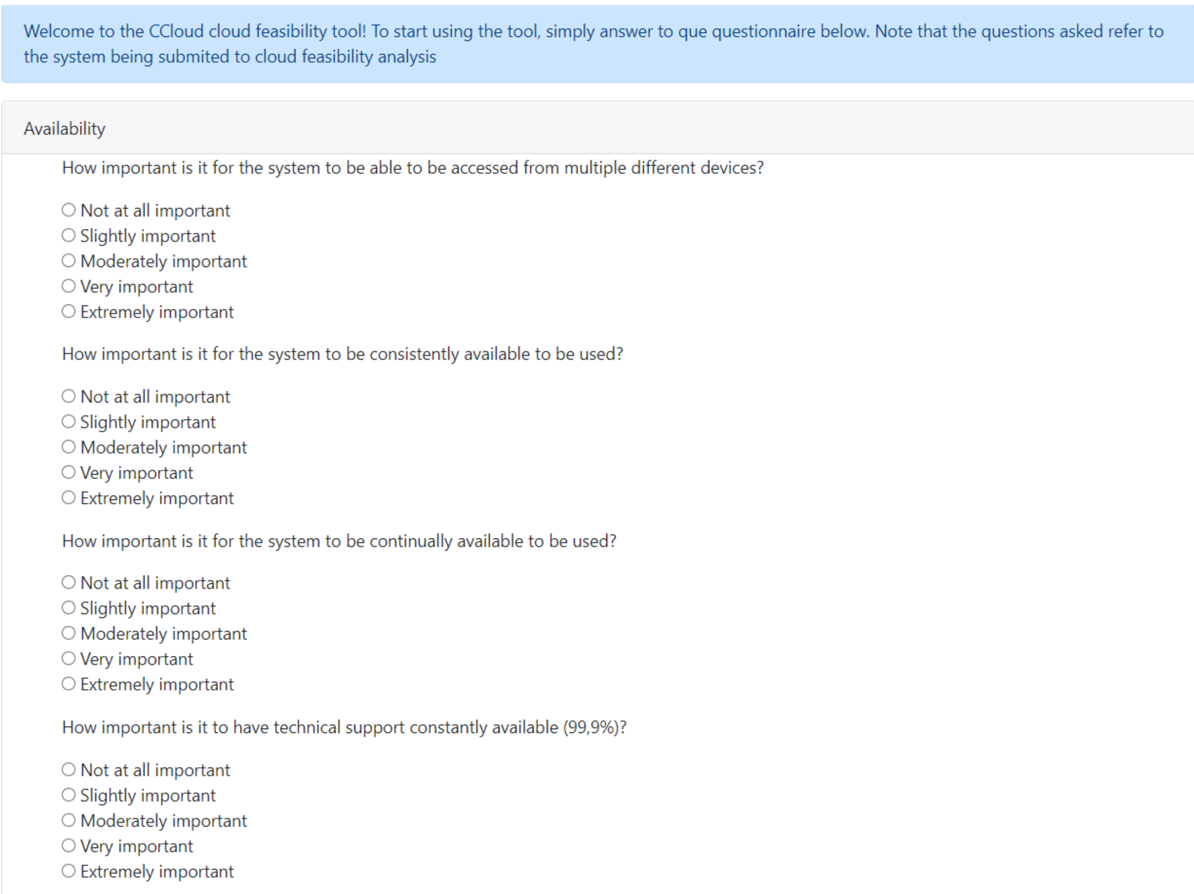


Figure A.1: Front page of the tool

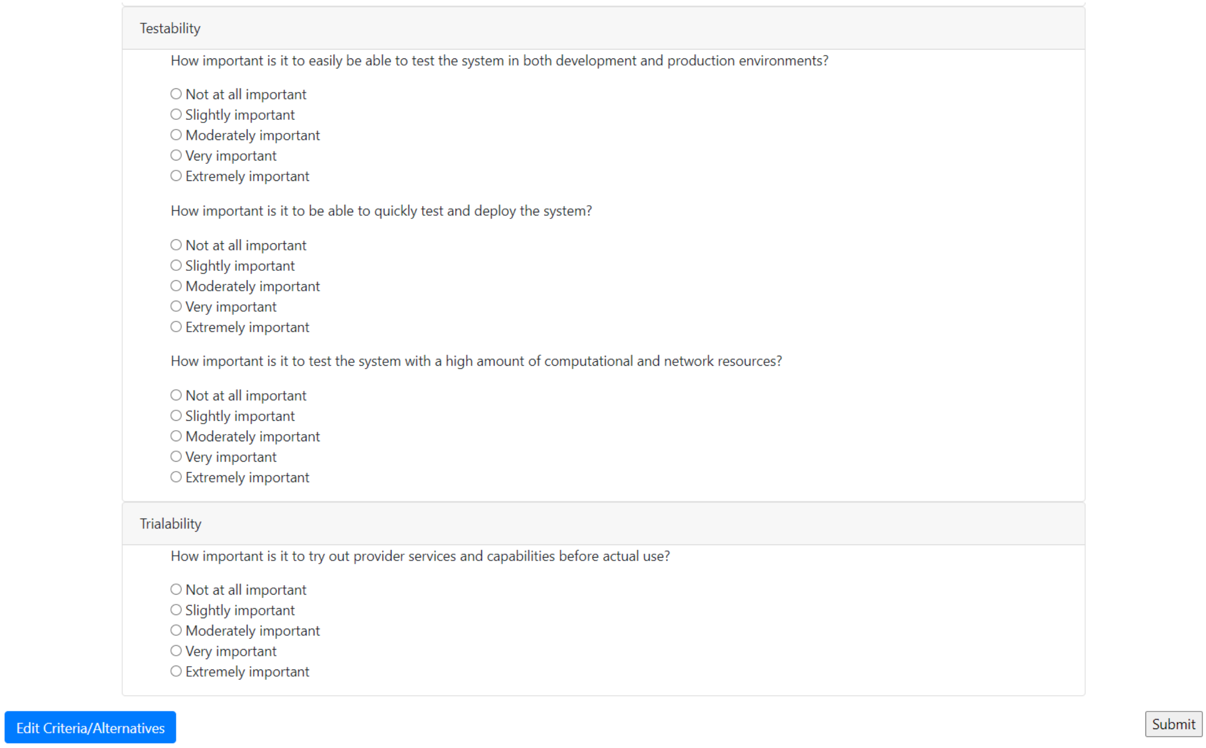


Figure A.2: End of the front page of the tool

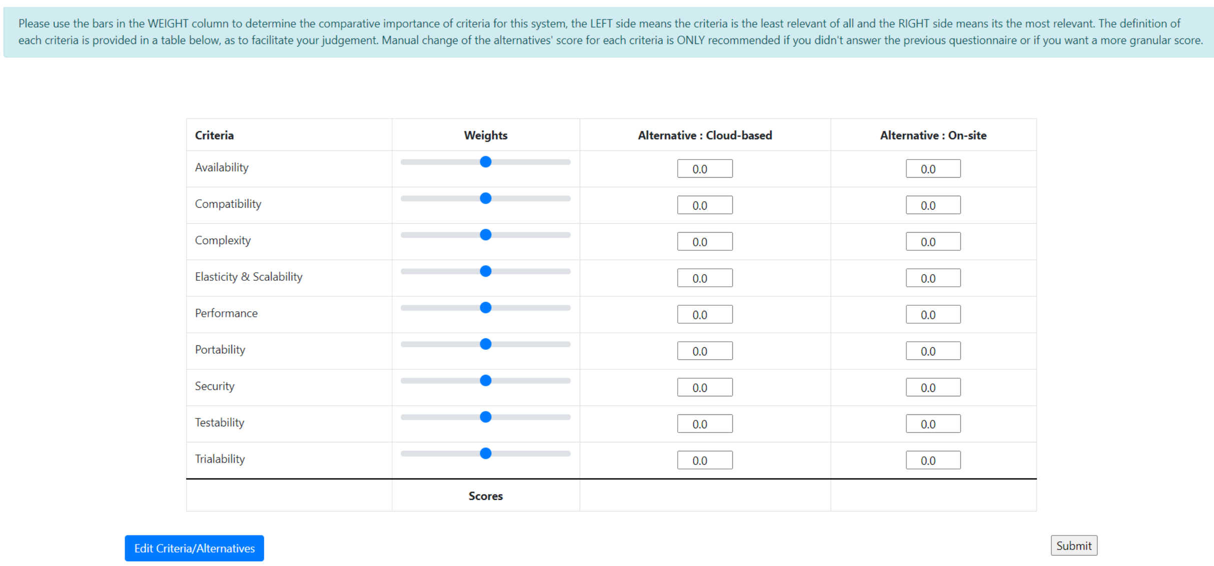


Figure A.3: MCDM page of the tool

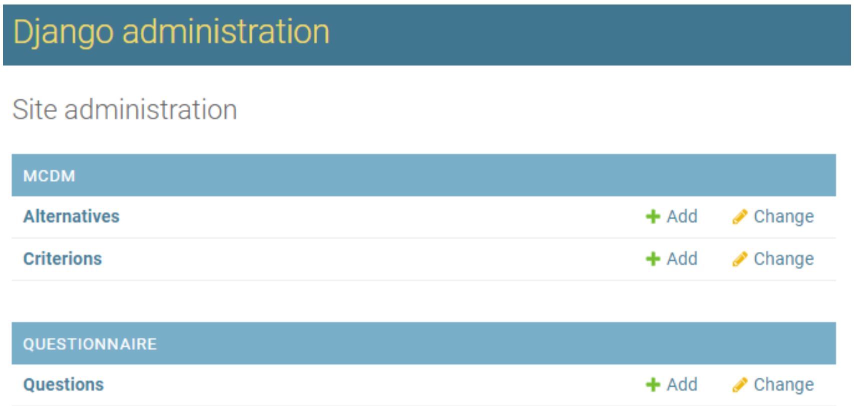
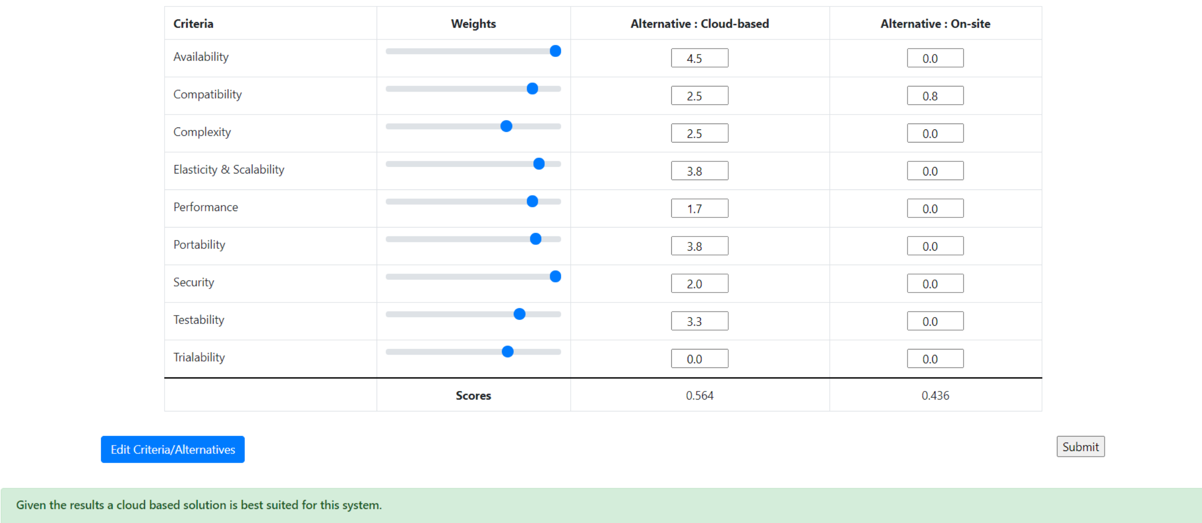


Figure A.4: Edit/Add page of the tool using Django Admin page



Criteria	Weights	Alternative : Cloud-based	Alternative : On-site
Availability		4.5	0.0
Compatibility		2.5	0.8
Complexity		2.5	0.0
Elasticity & Scalability		3.8	0.0
Performance		1.7	0.0
Portability		3.8	0.0
Security		2.0	0.0
Testability		3.3	0.0
Trialability		0.0	0.0
	Scores	0.564	0.436

Figure A.5: Results of the application of the tool evaluation case