VAASAN AMMATTIKORKEAKOULU
UNIVERSITY OF APPLIED SCIENCES

Faith Adekunle Omojola

OPTIMISATION OF CLOUD SERVICE PROVIDER SELECTION FOR COST -EFFECTIVE SOFTWARE DEVELOPMENT

Technology
2024

VAASAN AMMATTIKORKEAKOULU
UNIVERSITY OF APPLIED SCIENCES
Cloud-Based Software Engineering

ABSTRACT

Author	Faith Omojola
Title	Optimisation of Cloud Service Provider Selection for Cost-Effective Software Development
Year	2024
Language	English
Pages	55
Name of Supervisor	Tommi Rintala

This project presents the development and evaluation of a software tool designed to assist small and medium-sized enterprises in choosing a cost-beneficial cloud service provider. The tool uses artificial intelligence to analyse and recommend optimal cloud services based on the SME's specific requirements. The core objectives of the project were to simplify the decision-making process related to cloud adoption and optimise the use of cloud resources by SMEs.

The project used a comprehensive development approach, utilising user-centric design methodologies to ensure an intuitive and accessible interface integrated with OpenAI's GPT-4 model for dynamic cloud service recommendations. Extensive testing, including unit, integration, and system tests, achieved 80% coverage, demonstrating the robustness and functionality of the tool.

Overall, the decision-support tool significantly contributes to the SME sector, potentially improving how SMEs approach cloud service adoption, with implications for cost efficiency, scalability, and operational flexibility.

Keywords	Cloud Service Providers, Software Development, Small and Medium-sized Enterprises, Cloud Computing

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

Cloud computing, mobile technology, and advanced data analytics have surpassed how small and medium enterprises (SMEs) interact with customers and offer competitive opportunities. Many SMEs regularly discover that their internal IT cannot keep up with the market's rapid growth and increase. This is well articulated by Al-Mutawa (2024). They must then outsource certain essential services from Cloud Service Providers (CSPs) like Amazon Web Services, Google Cloud Platform, etc., using a variety of products such as Software as a Service (SaaS), Database Management Service (DBMS), Platform as a Service (PaaS), Infrastructure as a Service (IaaS), and so forth. Adopting CSP is a simple and cost-effective method of rapidly acquiring infrastructure that can efficiently manage the large volume of traffic or resources created by the SME's activities. This migration provides a cost-efficient alternative to conventional on-premises infrastructure arrangements, which can become progressively costly as the SME operations grow. Therefore, selecting the appropriate Cloud Service Provider (CSP) is essential for current and future business requirements.

The purpose of this study is to undertake a detailed review of the literature on choosing CSP critical factors, assess how researchers, business units, and stakeholders from Payhippo prioritise the development in a case study, design a data-supporting system to choose one, assess its practicality, and seek input from the Payhippo team.

1.1 Background and Motivation

Payhippo is a high-growth FinTech start-up in Lagos that is shaping the Nigerian ecosystem. The company was founded in 2019 with the vision of being the operating system for SMEs across Nigeria and, eventually, other markets in the rest of Africa. To achieve this, Payhippo is offering or plans to offer working capital loans, green energy financing, inventory financing, asset financing, business banking, inventory management, bookkeeping, and APIs for risk assessment, bank statement analysis, and credit profile look-up. Less than a year after going live, Payhippo has served a wide range of businesses. It was primarily focused on serving SMEs operating in the following industries: mobile money, FMCGs, oil and gas, and agriculture. On average, Payhippo's customers had an annual net turnover of €1,000,000 to €2,000,000.

Historically, Payhippo has operated on a hybrid cloud approach, using the infrastructure of leading CSPs, Amazon Web Services (AWS), and Google Cloud Platform (GCP) to deliver a wide variety of services. When the company started, Google and Amazon provided free or heavily discounted credits through their incubation programs. As the company scaled, it burned these credits at a high rate, quickly running out and driving significant costs. The move from start-up to scaling company required a multimillion-dollar investment in cloud infrastructure. In Figure 1, Payhippo's cloud service provider billing from March 2022 to March 2024 is illustrated, revealing a high annual expenditure exceeding $90,000 per year.

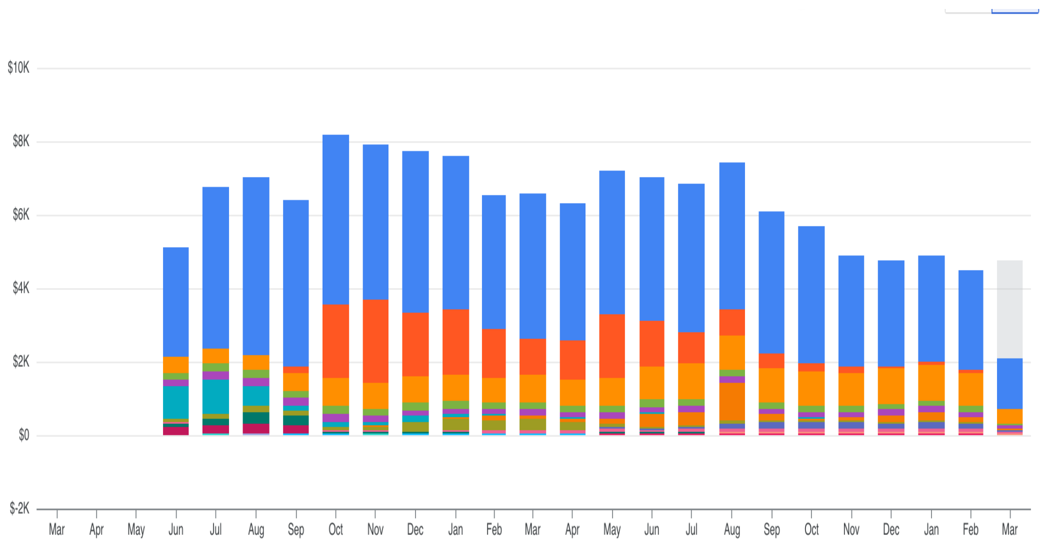

Figure 1. Payhippo GCP Billing Cost trend from March 2022 till March 2024. (Payhippo, 2024)

Due to significant financial constraints and high CSP costs, as seen in Figure 1 above, Payhippo is exploring more cost-effective options among Cloud Service Providers. Recognising the critical nature of this issue, Payhippo is developing a decision-support tool to help navigate this complex landscape. This tool will aid in making informed CSP selections for Payhippo, as well as its direct customers and partners involved in offering cloud-based solutions.

1.2 Statement of Problem and Research Questions

The research will discuss how cloud service providers can be selected in the most effective way for SMEs. The central problem or issue this study aims to solve is the efficient selection of a CSP for SMEs specifically. Since Payhippo’s decision to expand services across Nigeria and beyond, choosing a CSP has had a significant impact on operating expenses, performance, and business scalability. Hence, the research will present factors that determine which Cloud Services Provider is more relevant to an SME’s growth and consideration. The research formulates the following questions:

1. What are the factors that determine which Cloud Provider is more relevant for SMEs and Growth?
2. How do Payhippo's engineers and stakeholders perceive and prioritise the importance of various factors?
3. How effective is the developed tool in assisting engineers at Payhippo in their CSP decision-making processes?

1.3 Objectives of the Study

The first objective of this research was to conduct a comprehensive literature review, delving into the adoption of cloud service providers by SMEs. This review helped identify the factors influencing the adoption process and provided the foundation for understanding how these businesses could optimise their selection of cloud service providers.

Next, the study explored the ranking of these factors from an SME perspective by closely examining how engineers and other staff at Payhippo prioritise different aspects. Payhippo served as a valuable case study for analysing the decision-making process and determining which criteria play a significant role in selecting cloud providers. The insights gathered from this analysis were crucial for aligning technical principles with business administration needs.

The research also aimed to develop a decision-support tool to guide the selection of cloud providers. This tool was created to assist businesses in understanding, from technical and administrative perspectives, the best approaches to make such decisions. The engineering staff at Payhippo provided input on the tool's efficacy, which allowed for iterative changes to enhance its general relevance and use. In the end, the study sought to enable businesses to develop practical plans for selecting cloud providers who best suit their particular requirements and goals.

2 LITERATURE REVIEW AND THEORETICAL FRAMEWORK

This chapter considers the entire research and literature review to summarise the subject and thoroughly investigate CSP selection. It also reviews research papers and selects relevant theories and models for the research. Finally, a theoretical model will be produced after the literature review. A theoretical framework can be defined as a roadmap comprising the theories and amalgamated research concepts that try to assist us in understanding a concrete CSP selection for SMEs. This roadmap will serve as a guide to understanding the selection of CSPs for SMEs by breaking down the background and trends of cloud computing, analysing CSPs by classification and market shares, and identifying the specific factors influencing SMEs' decision-making process. This roadmap structure will underpin the development of the decision-support tool introduced later. In this case, the technical and business implications of the problem are covered, and we will determine our support for the solution creation.

2.1 Cloud Computing: Background and Historic Trends

The history of the cloud covers several decades. Understanding the historical context of cloud computing reveals how the evolution of cloud computing has helped shape technological advancements. It provides insight into why adopting cloud technologies is significant for SMEs today and emphasises the importance of selecting the appropriate Cloud Service Provider. This section describes cloud computing background development and key milestones.

Mainframe computers were the leading technological solution in the early days of computing. Large, centralised devices enabled multiple operators to access a single system using the time-sharing method. According to Rice's (2023) detailed roadmap of cloud computing development, during the mainframe era, the concept of cloud computing was first discussed and developed with it the idea of linking computers together over great distances for a mixture of scientific and military purposes. This idea further shaped the development of cloud computing.

A critical paradigm shift in computing that was brought about by the client-server architecture entailed decentralising operations and tearing apart clients (end-user devices) from servers (centralised machines). This model made organisations take responsibility for their servers, yet the requirement for scalable, cost-effective computing solutions began to push them toward fostering cloud computing (Varghese, 2019).

The 1990s saw the emergence of the World Wide Web for streamlined communication and information exchange. Thus, with the appearance of web services and web applications, it became possible to enable cloud-based solutions whereby users would access resources from remote locations. That would actually constitute the nature of cloud services (Arutyunov, 2012).

In fact, the very concept of today's cloud computing began taking shape sometime around the early 2000s. The prototype of the real Infrastructure as a Service model was pioneered by none other than AWS in the form of the on-demand renting of computing resources for businesses. Prompted by these technological advances, there was a movement to more scalable and flexible cloud environments through increased virtualisation. This only came after AWS, a move that has been followed by big players such as GCP and Microsoft Azure (Bigelow, 2022).

The term "cloud" has been very popular these last few decades; hence, they called it "cloud," like a virtual space where an application runs. Services designed in the model of "infrastructure as a service" make an abstraction of physical hardware so that users concentrate on functionality rather than the totality of the underlying hardware. This abstraction has fundamentally led to improved efficiency and flexibility in cloud computing.

Today, its tentacles have reached far beyond a small and select circle of technology enthusiasts to a larger public; cloud computing is now a mainstream foundation across nearly every sector. It has become more democratic in this regard and more accessible to small businesses and regular folks. A better appreciation of the effect of cloud computing on modern technology can only be enhanced through an understanding of the history of cloud computing (Mathenge, 2021).

2.2 Analysis of Cloud Service Providers

Cloud Service Providers play an integral role in the IT landscape of contemporary businesses by offering an array of infrastructure and software solutions over the Internet (Google Cloud, n.d.). These services allow for easy and flexible access to applications, resources, and services, with CSPs taking on full management responsibilities. This setup reduces the need for on-premises hardware, software, and maintenance personnel to oversee everything (Shetty & Panda, 2021).

This section will examine CSP classifications, market shares, and pricing dynamics using current research and analysis.

2.2.1 Classification of CSPs

CSPs are generally divided into two main categories based on the scope of services they provide: Major CSPs and Niche Providers, which include Infrastructure and Platform Providers.

Major CSPs include providers like Amazon Web Services, Microsoft Azure, and Google Cloud Platform that offer an extensive portfolio of services across Infrastructure as a Service, Platform as a Service, and Software as a Service. They are known for their vast global infrastructure, broad service offerings, and substantial market presence (Richter, 2024).

Niche Providers include CSPs that cater to specific market segments or industries with tailored services that meet unique business needs or regulatory requirements. According to Almishal and Youssef (2014), “They offer specialised services tailored to unique business requirements or regulatory environments”. The major advantage of niche providers is their simplicity and ease of use, as they usually focus on only one segment of cloud computing product offerings.

Among these niche providers are Infrastructure Providers like Digital Ocean, Linode, and Vultr primarily offer IaaS solutions. They are praised for their simplicity and ease of use and for targeting developers and SMEs.

Another segment includes **Platform Providers** such as Heroku, specialising in PaaS solutions, offering platforms that simplify much of the infrastructure management, allowing developers to concentrate on building applications.

2.2.2 Market Shares and Pricing

AWS, Azure, and GCP dominate the cloud services market and have established significant market shares due to their comprehensive service offerings and global reach. AWS leads the market, followed by Azure, which has capitalised on its enterprise relationships, and GCP, known for its analytics and machine learning services (Patel, Tiwari, & Khureshi, 2022).

Pricing among CSPs varies based on services provided, geographic location of data centres, and customer needs. Major CSPs offer pricing calculators and models, including pay-as-you-go, reserved instances, and spot pricing. Infrastructure and platform providers often compete on price and simplicity, appealing to SMEs and developers with straightforward pricing models and packages.

2.3 Adoption of Cloud Services by SMEs

The adoption of cloud services by Small and Medium Enterprises is a topic gaining traction in academic and business communities. SMEs represent a crucial market for cloud-based offerings. The digital environment provides numerous opportunities for entrepreneurs who start and run SMEs to impact and grow their businesses. To survive in a turbulent competitive environment, SMEs have had to adapt to the digital environment to adjust to customer needs globally, particularly in a post-COVID-19 world. Therefore, there is a strong need for digital technologies to improve business performance, cost advantages, and competitive advantages with a restricted budget and limited resources (Kumar, Naveen, & Savitha, 2022). As a representative of emerging digital technology, cloud computing has become an important information technology (IT) development trend in recent years.

Cloud computing sidesteps the need for capital investments in hardware and expensive information technology (IT) infrastructure. The provider hosts products and services from a far-off location through his energy-disbursing servers, and the consumers begin to use the network without fitting or installing anything. Some of the promised benefits from cloud computing can be very appealing for SMEs, which need to maximise the return on their investment and remain competitive in an ever-demanding business environment (Iskandar, 2021).

This section of the literature review discusses the main factors affecting SMEs' decision-making regarding adopting cloud services. Among the topics to be discussed are the benefits, challenges, cost efficiency, and other dynamics surrounding the adoption process.

2.3.1 Benefits of Cloud Adoption for SMEs

One of the primary benefits to SMEs of cloud services adoption is the significant reduction in costs for the IT infrastructure. According to Shetty and Panda (2021), generally, cloud computing eliminates substantial upfront investments in hardware and software through pay-as-you-go models suitable for business requirements. This financial gain, in turn, allows SMEs to allocate resources more efficiently, enabling them to focus on strategic growth areas instead of being burdened by heavy IT expenses.

Cloud computing also gives SMEs the capability to tailor the use of cloud resources either upwards or downwards in response to their capability, hence giving them flexibility. The former includes the use of cloud services to upscale or downscale according to demand, hence allowing flexibility to SMEs, therefore making SMEs able to grow and adapt to service workloads without expensive infrastructure changes (Mathur, 2023). As a result, SMEs can better manage their resources and remain agile in a rapidly evolving business landscape.

Furthermore, Cloud computing improves collaboration in and across organisations. It allows workers to work on required documents and applications through access and sharing from various places, hence improving their productivity and innovation. This seamless access fosters an environment where teams can work more efficiently, ultimately driving the growth and success of SMEs.

2.3.2 Challenges in Cloud Adoption for SMEs

When SMEs consider using cloud service providers to support their IT infrastructure, one of their main concerns is data protection. The main topics of discussion in this case are fears about data breaches, giving up control over private information, and following data protection laws (Kumar, Naveen, & Savitha, 202). Most SMEs put protecting their mission-critical data first, so the issues often lead to reluctance or unwillingness to fully embrace cloud services. It is imperative to address these issues by putting in place extensive security measures and compliance procedures to provide confidence in the use of cloud services.

In addition, several small and medium-sized businesses lack the IT expertise required to find, implement, and manage most cloud services effectively. According to Kumar, Naveen, and Savitha (2022), this flaw might result in wasted money, increased operational risks, and ineffective use of cloud resources. When SMEs lack qualified IT personnel, it may be challenging to maximise their use of cloud services, which can result in missed opportunities for possible benefits and unintentionally expose vulnerabilities that could jeopardise their operations.

Among the challenges is vendor lock-in. SMEs that get overly reliant on a single cloud service provider struggle to change and negotiate better conditions. These limitations could make it harder for them to adjust to shifting company needs or bargain for better service conditions with competing suppliers. To reduce the negative effects of vendor lock-in on their ability to expand and maintain operational flexibility, they must thus carefully and strategically select cloud services.

2.4 Factors Determining SMEs' Selection of CSPs

According to Zhang, Wang, and Liang (2021), selection of CSP by SMEs is a complex process influenced by many factors. These factors, which Hsu and Lin (2016) researched and summarised in Table 1, align with their operational, technical, and financial goals, focusing on optimising costs and ensuring effective service efficiency.

Table 1. Factors SMEs consider while selecting CSPs, according to Hsu and Lin (2016).

FACTOR GROUP	FACTORS
FINANCIAL CONSIDERATIONS	- Budget
	- Total Cost of Ownership (TCO)
TECHNICAL TEAM AND EXPERTISE	- Team Size
	- DevOps Capabilities
SERVICE REQUIREMENTS	- Scalability and Flexibility
	- Reliability and Uptime
SECURITY AND COMPLIANCE	- Data Security Measures
	- Compliance with Regulations
SUPPORT AND SERVICES	- Customer Support
	- Technical Support
VENDOR CHARACTERISTICS	- Market Reputation and Stability
	- Vendor Lock-in and Flexibility
OPERATIONAL AND STRATEGIC CONSIDERATIONS	- Data Governance and Management
	- Partnerships and Ecosystem
	- Technological Capabilities

2.4.1 Financial Considerations

Financial considerations are crucial when selecting a cloud service provider because they directly impact the feasibility and sustainability of cloud adoption for SMEs. The two primary factors under this category are:

1. Budget Allocation**:** The SME should decide on the budget allocation to the project. According to Hsu and Lin (2016), the budget for the project determines the type and level of cloud services a given SME can acquire and is, therefore, an important factor in the decision-making process of the CSP.
2. Total Cost of Ownership (TCO): The TCO framework also includes total costs that SMEs have to incur for running some activity in the cloud environment, as they cover both indirect ones related to data transfer and

costs incurred for the use of additional functionalities and integration with already existing systems.

2.4.2 Technical Team and Expertise

The technical team's size and expertise are pivotal in leveraging cloud services effectively. SMEs must evaluate the team's capacity to manage the selected provider's services, ensuring alignment with the team's skills and resources. This category includes the following:

1. The size of the technical team influences the selection of the CSP because all the various providers offer different kinds of support and levels of services that have to be matched with the team's ability to administer the cloud services (Hsu & Lin, 2016).
2. The presence of a DevOps team may influence the CSP choice. While some certain providers provide state-of-the-art services and tools clearly crafted to meet CI/CD practices, all of them come with an explicit requirement for DevOps skills in order to be managed.

2.4.3 Usage and Traffic Projections

Usage and traffic projections are essential for selecting a cloud service provider because they help estimate future demands and ensure that the chosen provider can accommodate these requirements.

SMEs should be able to estimate the quantity of data generated and the number of individual users needing access to the services offered by the CSPs. They will have to consider a CSP whose offering is within the line of their anticipated workload at reasonable costs (Charles, 2018).
The Project Reach may be determined based on whether the SME is initiating a local or international project. This will, therefore, determine the number of CSPs' data centres to use, the area in which they are used, and the maximum number of

utilisations. There may also be a need for multi-region and multi-zone services to ensure low latency.

SMEs need to ascertain their current and projected usage/traffic over the next 3, 6, and 12 months and must ensure that the selected CSP has the capacity to scale resources accordingly and affordably meet dynamic demands (Charles, 2018).

2.4.4 Service Scalability and Reliability

Service scalability and reliability are vital to ensuring that SMEs can grow and maintain their business operations smoothly.

Scalability is the ability to scale services in line with business needs or contraction, which is crucial for maintaining cost efficiency and operational flexibility. Reliability and uptime for the services needs to be very high so that it does not hamper the business of the SMEs. The preferred ones are CSPs who have shown strong reliability (Naser, 2014).

2.4.5 Security and Compliance

Security and compliance are fundamental factors when selecting a CSP, ensuring that SME data is secure and adheres to regulatory standards. An important consideration by SMEs is the security features and adherence to regulation. They should offer well-advertised security measures and demonstrate compliance with, for example, standards established by HIPAA, PCI DSS, or GDPR (Charles, 2018).

SMEs must ensure that their data is managed securely and in compliance with privacy laws, which in turn affects their choice of CSP (Wong, 2021).

2.4.6 Support and Service Offerings

Support and service offerings are critical because they determine how effectively SMEs can access help and meet their business needs. One of the most important offerings of a CSP is the level of Customer Support and Technical Assistance that they provide. This is of huge significance, particularly for SMEs that do not have a robust technical team. The specific services a CSP offers and their ability to meet an SME's unique needs are crucial in the selection process (Waferwire, 2020).

2.4.7 Additional Considerations

In addition to the core considerations, several other factors play a role in selecting the right CSP. SMEs should evaluate the ease of migrating away from a CSP and the presence of exit fees or other barriers to changing providers (Cloudindustryforum.org, n.d.). The availability of a global community of partners, such as ISVs, SIs, Solution Providers, and Resellers, can provide additional support and expertise to SMEs without strong IT support in-house. Access to a CSP's virtual marketplace can help SMEs keep third-party software spending economical and introduce them to new tools to keep their business efficient.

SMEs' selection of a CSP is a complex process influenced by various factors. These factors are not only related to the cost but also to the technical expertise of the team, the projected usage and traffic, the scalability and reliability of the service, and security and compliance requirements. Additional considerations such as vendor lock-in, partner ecosystem, and access to a virtual marketplace also play a crucial role in the selection process.

2.5 Review and Ranking by Payhippo Colleagues

To verify and prioritise the above-explained influencing factors that determine the choice of Cloud Service Providers, members of the Engineering department at Payhippo, the focal company of this thesis, were involved in the review exercise. The engineering department comprises 12 colleagues: one chief technology officer, one engineering manager, and one product manager. Subdivided into nine engineers, six hold senior positions in software engineer capacity, while one is a

Quality Assurance Engineer and two are mid-level Software Engineers. This review uses the practical experience and insights offered by colleagues who are fully aware of the company’s current business requisites and the methods through which it progresses its daily operations.

Colleagues were asked to ascertain the factors explained to be the most critical factors for the organisation. Determinations were based on each colleague’s perception of their daily operational challenges and the employers’ current priorities and needs. In Table 2, the findings from the rank results developed by the colleagues at Payhippo are shown.

Table 2. Factors Ranking by Colleagues at Payhippo.

Factors	Importance Ranking
Cost and Budget Alignment	1
Security and Compliance	2
Technical and Customer Support	3
Service Scalability	4
Reliability and Uptime	5
Data Governance and Management	6
Vendor Lock-in and Flexibility	7
Technological Capabilities	8

The insights obtained from this exercise, in conjuncture with the literature review findings, will form the basis of the discussions. The findings will subsequently be integrated to develop a tool upon which SMEs can rely to select the best CSP for their enterprise. The formulated tool will use the ranked factors to develop a decision-making framework that prioritises revenues first and cost minimisation while considering other crucial factors such as security, support, vendor reputation, and scalability. Notably, the tool will demonstrate the practical foundation of the theory covered in the literature review. This discourse seeks to link the research element and the business application requirements. The tool attempts to assist SMEs such as Payhippo with a clear methodology for making the CSP selection process. As such, this exercise exposes the importance of blending the research and the empirical observations to solve the common business problem. Further discussions in the subsequent chapters will expand on this tool.

2.6 Theoretical Framework

The theoretical framework for this thesis is constructed around the factors influencing SMEs' decision-making process when selecting a Cloud Service Provider. This framework synthesises prior research and practical insights to form the foundation upon which the decision-support tool was developed. It integrates the comprehensive factors that affect the selection process.

These factors were ranked by the engineering team at Payhippo, whose practical experience provided valuable insights into how they impact the decision-making process. The team emphasised the importance of Cost and Budget Alignment, Security and Compliance, and Technical and Customer Support, prioritising these factors to reflect the real-world operational challenges and current business priorities of the organisation.

This theoretical framework underpins the decision-support tool by translating these factors into an interactive and user-centric recommendation engine. Through the framework, the tool ensures that SMEs can make informed decisions that align with their business strategies while navigating the complexities of CSP selection.

3 DECISION TOOL DEVELOPMENT

This chapter will describe the development process for decision-support tools to help SMEs choose the most suitable cloud service providers. It will present the design of the architecture and the rationale behind the user interface and discuss how these choices enabled effective performance. It intends to focus on the workflow of this system to show how user input is entirely integrated with algorithm input and decision outputting. From this perspective, it will be possible to demonstrate that the recommendation system can be used reliably in everyday practice.

It will also explain the tool software development process, highlighting the technologies/tools and advanced API solutions used, such as OpenAI's API for intelligent question processing, Blasmiq for sketches, and Figma for high-fidelity wireframing. It would also explain the UI development aspect of this project to highlight the increased interaction and intuitive user experience. This chapter connects the usability of the system with the functional aspect, where SMEs will be able to use this tool effectively in practice.

3.1 User Interface Design Decisions

The User Interface (UI) bridges technology with the user with its aesthetic design principles and functionality. Careful consideration was given to UI decisions during the development of the CSP decision support software to offer an intuitive and efficient user experience (UX). The main goal was to make a design that would be easily navigable, and at the same time, this simplicity would also showcase the sophistication of the tool.

3.1.1 Initiating with Wireframes (Low-Fidelity)

The first step of designing the UI was the wireframing, and this was done with the use of a specialised wireframing tool called Blasmiq, which helps in the creation of low-fidelity wireframes, in other words, sketches. As a result of the evaluation of the generalised diagramming tool Miro and the use of the tool designated explicitly for wireframing tool Blasmiq, Balsamiq was chosen for its wireframing capabilities because it has fewer distractions, is easy to use without having to learn anything special, allows for iterations and immediate suggestions that helped enhance the appearance and navigational characteristics of the UI. Most importantly, the tool allowed for skipping more generalised features of diagramming tools and facilitated real-time wireframing improvement. Figure 2 illustrates a sketch of the UI using Blasmiq.

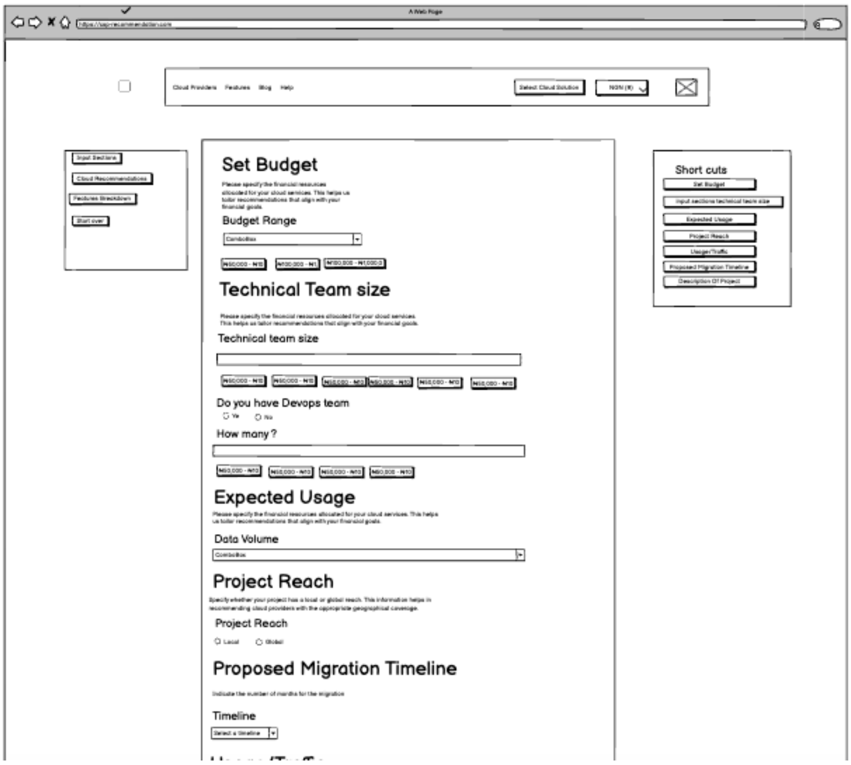

Figure 2. Blasmiq Wireframe: Depicting the User Input User Interface Sketch.

3.1.2 High-Fidelity Wireframing

After the layout ideas from the low-fidelity sketches were completed, the design process shifted focus to developing high-fidelity wireframing. For the high-fidelity wireframing, Figma was used, a tool notorious for its detailed and complex design capabilities and collaborative ability. High-fidelity wireframes are fundamental as they provide a more accurate representation of the final product, utilising real colours, text, and other elements imitating the real product.

In Figma, the rough outlines of Balsamiq were converted into visually elaborate and interactive prototypes. This process allowed the designed layout to match the expectations of the user and functional requirements. Moreover, Figma supported real-time collaboration, ensuring that members provided instant input and feedback for validation. Figure 3 shows the high-fidelity version of the same UI in low-fidelity mock-up.

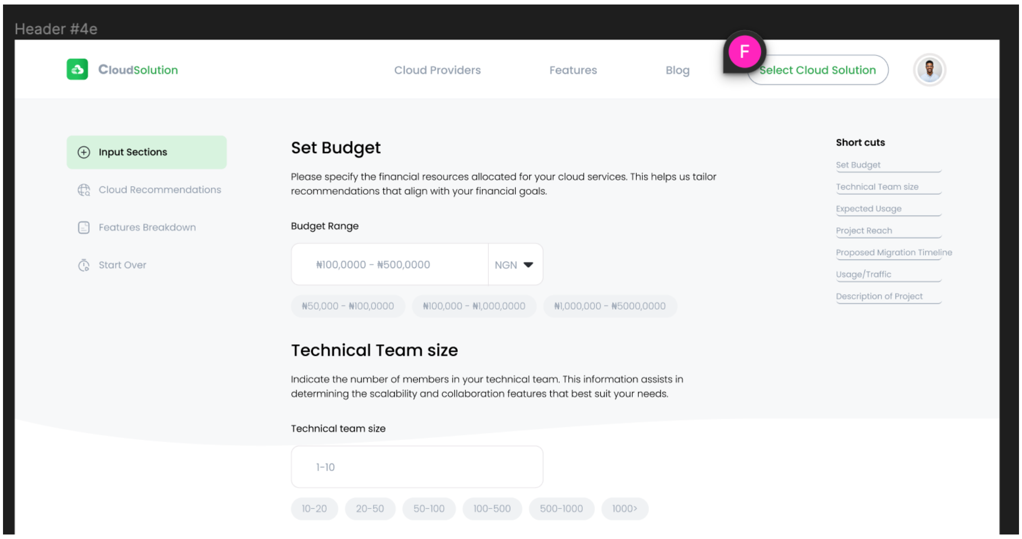

Figure 3. Screenshot of UI from Figma's high-fidelity mock-ups.

3.1.3 Commitment to Core Design Tenets

In developing the user interface of the decision-support tool, strong emphasis was placed on clarity, ensuring information is presented in a friendly and easily digestible manner without overwhelming users with excessive jargon or data. This principle is essential for promoting user involvement and facilitating intuitive navigation of the instrument. Figure 4 demonstrates the transition from Balsamiq wireframes to Figma high-fidelity mock-ups, showcasing the interface's development while ensuring clarity and coherence in the design. Developing a unified visual language required maintaining consistency, ensuring that buttons, cards, and other design elements adhered to a structured and uniform style. This systematic methodology fosters familiarity, establishing a predictable environment that boosts user confidence and minimises the time required to learn.

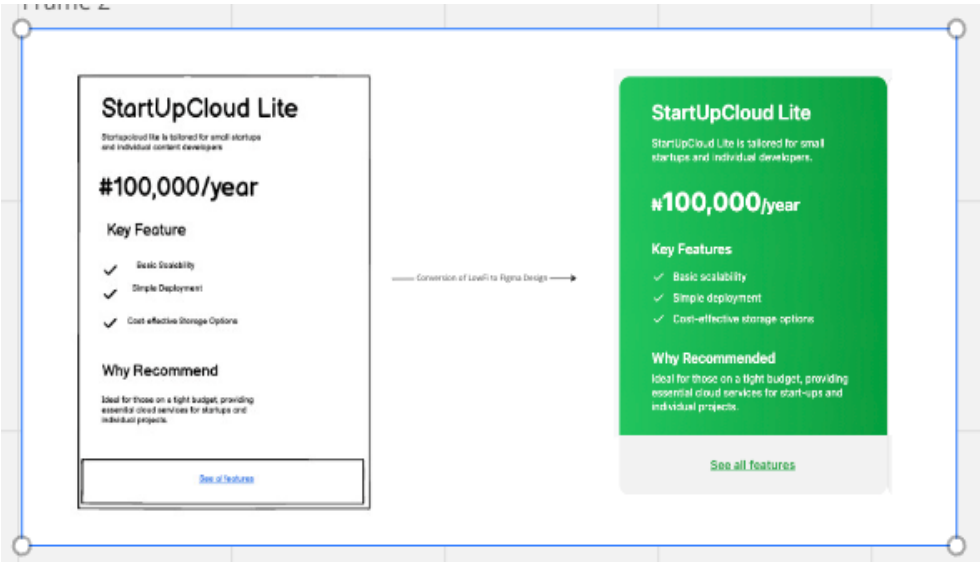

Figure 4. User interface evolution from high-fidelity mock-ups produced by Figma to wireframes created by Balsamiq.

Carefully considered to satisfy accessibility requirements, the design guaranteed that it could support inclusivity and accommodate those with disabilities. Features including keyboard navigation, high-contrast mode, and customisable font size improve accessibility and broaden the tool's applicability to people with different abilities. By using intuitive navigation and well-placed components, inclusiveness is improved, and all users may easily find and understand the tool's features.

To guarantee easy use independent of the platform, the design was also thoroughly tested to assure compatibility across a wide range of devices and screen sizes. All users may interact with the tool comfortably on their preferred devices because it has a responsive and adaptive design that provides an ideal user experience whether accessed from a desktop computer, tablet, or smartphone. The combination of these design principles ensures that the user interface remains functional, accessible, and enjoyable, providing SMEs with a supportive environment to explore and implement cloud service recommendations.

3.1.4 User Engagement and Feedback Integration

Following the prototype development, live demonstrations were done for a curated audience, including engineers and managers at Payhippo and the thesis academic supervisor, Tommi Rintala. Observing user interactions with the prototype provided concrete data crucial for refining the UI. Figure 5 captures the feedback sessions, depicting the interactive elements of the UI and user engagement metrics.

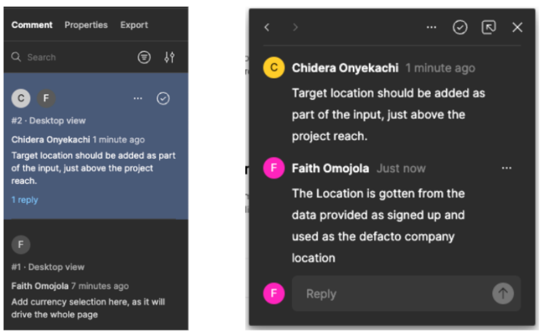

Figure 5. Feedback Snippet Screenshot.

3.2 Architecture Decisions

The architecture crafted for the decision-support tool for selecting a CSP has been built to break down the various components of the system in a systematic fashion.

This was aimed to ensure that the tool's framework would provide the user with a transparent view of the underlying structure that supports the recommendation processes.

3.2.1 High-Level System Architecture

At the highest system level, the system is designed as a two-tiered architecture that further segregates the user interface as one tier and the processing logic, data management and external integration as the second tier.

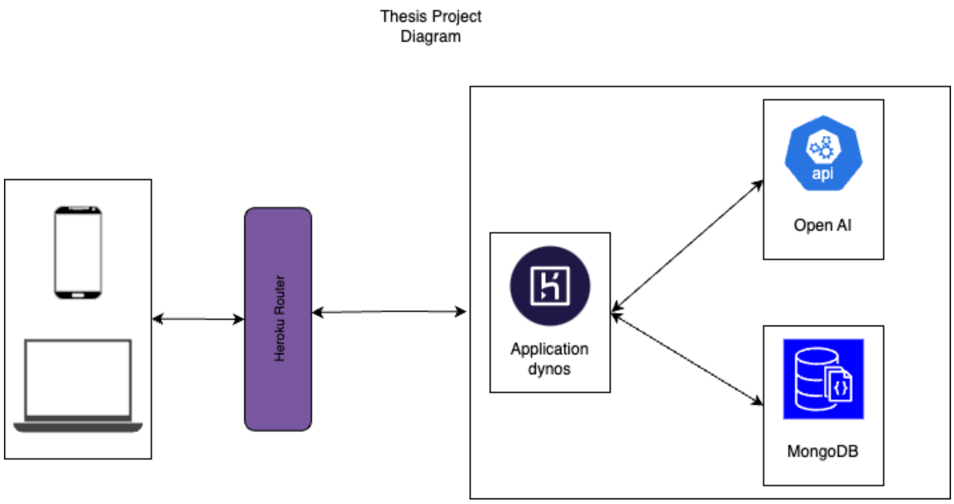

Figure 6. High-level architecture Diagram.

Figure 6 illustrates how requests from the user interface are routed through the application server hosted on Heroku and then processed by the logic layer, where OpenAI's API is utilised for intelligent querying. Data persistence is managed by MongoDB, ensuring data consistency and speed of retrieval.

3.2.2 User Interaction Workflow

An understanding of how the users interact with the system is essential in understanding the architectural model. The user interaction workflow gives the user steps on how they handle the system, from account setup to receiving CSP recommendations.

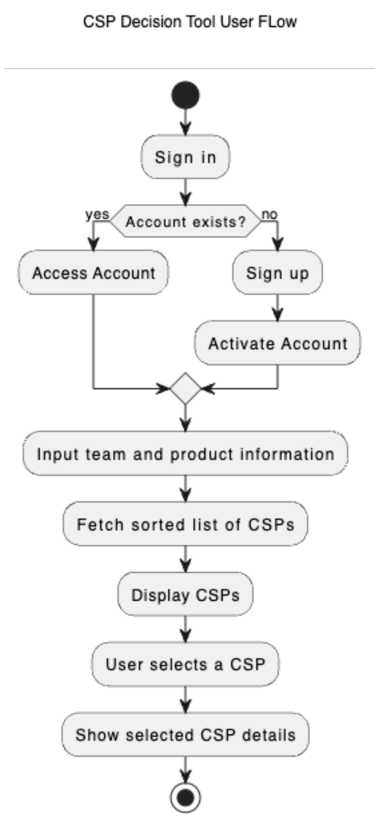

Figure 7. User Interaction Workflow UML.

In Figure 7, the decision tool UML diagram shows the steps a user must undertake to obtain a CSP recommendation. The user makes critical decisions such as creating an account, data entry phases, getting CSP recommendations, and selecting their CSP.

3.2.3 Data Structure and Relationships

The entity relationship diagram (ERD) shows the how the various data entities in the project are related.

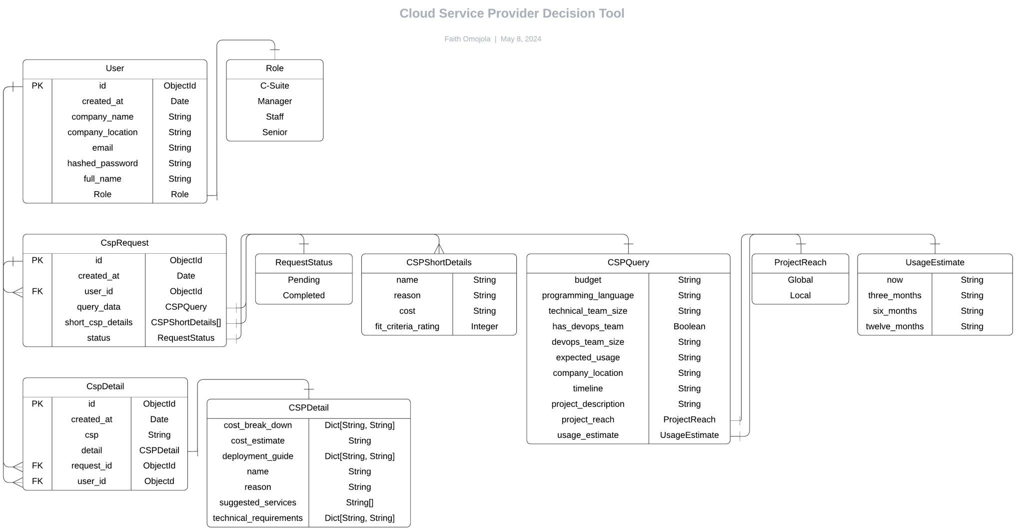

Figure 8. Entity Relationship Diagram.

In Figure 8, the ERD relationship diagram illustrates the relationships among the User, CSPRequest, CSPDetail, and related models, providing insight into the database schema and critical relationships. This aligns with the MongoDB model definition.

3.2.4 Cloud Service Provider Selection and Integration

When choosing the CSP responsible for managing the system, some variables were put into consideration, the major ones being how easy it was to integrate, cost, scaling capabilities, and well the provider is known to be reliable. The CSP to be selected needs to offer a suite of services required by the decision-support tool, like being capable of handling complex computations and data management essential for generating accurate recommendations.

The decision was made to use Heroku as the primary CSP, mainly due to its affordability at approximately 60 Euro per year and its user-friendly platform that simplifies the deployment process, particularly for applications shipped using docker containers. Also, Heroku's seamless integration with GitHub's CI/CD pipeline offers an efficient flow from code commits to deployment, covering the entire process with minimal setup.

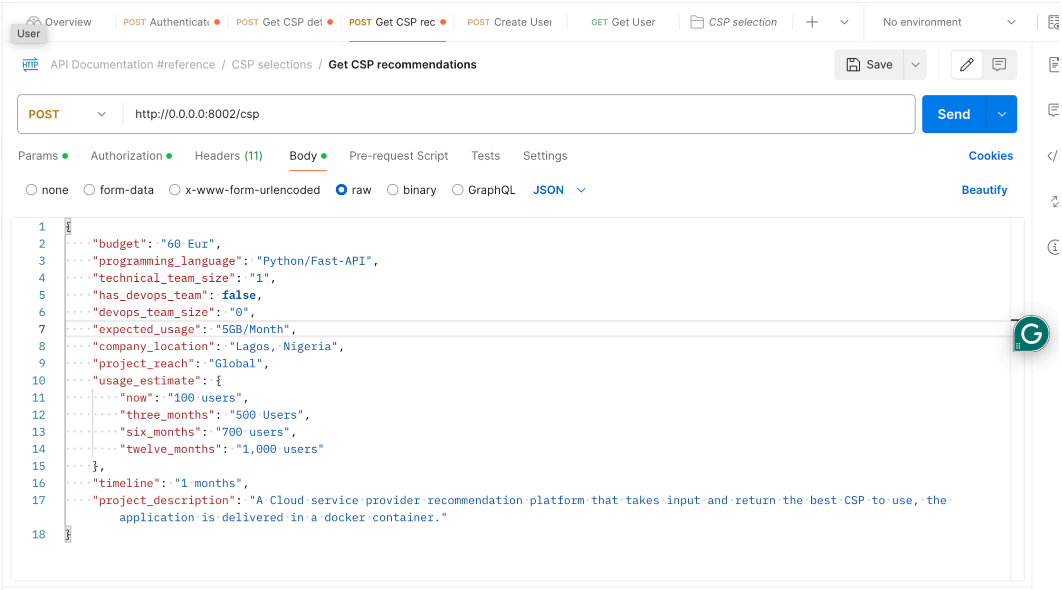

Figure 9. Screenshot of CSP recommendation API call from Postman.

In Figure 9, the Postman screenshot shows the structure of the API calls and the input parameters that drive the selection of the CSP. It provides a demonstration of how the CSP decision tool can be used.

In Table 3, the selection criteria, including cost-effectiveness, ease of setup, support for Docker, and integration with GitHub CI/CD, are shown. Each CSP is rated on these criteria, revealing Heroku's suitability with the highest score based on the tool's recommendation.

Table 3. CSPs recommendation comparison generated by the decision tool powered by OpenAI's API.

Name	**Reason**	**Cost (Euro)**	**Rating**
Heroku	Heroku is incredibly user-friendly and ideal for developers who want to focus on application development without worrying about infrastructure. It supports easy deployment of containerised applications.	60	5
DigitalOcean	DigitalOcean offers simplicity and cost-effectiveness, making it ideal for start-up's and small teams without a dedicated DevOps team. Their Droplets and App Platform are user-friendly for deploying containerised applications.	72	4
Linode	Linode provides straightforward pricing and powerful compute instances that are suitable for small to medium-sized projects. Their support and documentation are excellent for teams without a DevOps background.	70	4
Vultr	Vultr offers competitive pricing and a wide range of cloud services that can scale with the project's growth. Their global footprint ensures low latency for users worldwide.	720	4
Alibaba Cloud	Alibaba Cloud offers competitive pricing and a strong presence in Asia, making it a good choice for projects targeting users in that region. They provide services that are scalable and have a wide range of cloud computing solutions.	720	4
AWS LightSail	AWS LightSail is designed for simplicity and cost predictability, making it a good choice for small projects and start-up's. It provides a managed environment that simplifies the deployment process.	750	3
Google Cloud Platform (GCP)	GCP offers a robust and scalable infrastructure with a strong emphasis on AI and machine learning services. Their always-free tier and sustained use discounts can be beneficial for cost management.	800	3

Microsoft Azure	Azure provides a comprehensive set of services that cater to all aspects of cloud computing. They heavily support hybrid cloud environments and the integration with Microsoft products make them a versatile choice.	850	3
IBM Cloud	IBM Cloud offers a wide range of services with a focus on AI and machine learning through Watson. Their cloud solutions are scalable and secure, suitable for enterprises and projects with specific compliance needs.	780	3
Oracle Cloud	Oracle Cloud is known for its high performance and strong emphasis on database and enterprise applications. Their cloud infrastructure is designed to support demanding workloads and is a good fit for projects requiring robust database management.	800	3

3.2.5 CI/CD Pipeline for Agile Development

One of the fundamentals of any modern software development is continuous integration (CI)/continuous delivery (CD)—a relatively simple way for teams to better schedule and provide changes to their code base quicker and more accurately. To drive this pipeline, there are tools that save time and eventually fully automate the entire SDLC workflow. Figure 10 presents how everything is orchestrated from a code commit.

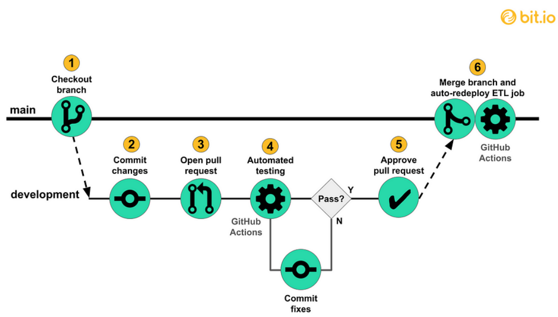

Figure 10. GitHub CI/CD workflow (Doss, 2021).

GitHub Actions implements a wide range of automation of the essential workflows, such as pulling saved secrets and environment variables in a secure manner after code commit, ensuring that solely those builds that have passed the testing phase are deployed. This is important to ensure that unanticipated behaviours are not experienced due to changes in the code. Figure 10 shows the CI/CD flow showing how code moves from commit to deployment, all powered by GitHub Action.

Figure 11 shows a code snippet that presents the GitHub Actions configuration. It provides a compact view of the command output and parameters that define the jobs within the CI/CD pipeline.

```
name: Deploy

on:
  push:
    branches:
      - master

jobs:
  build:
    runs-on: ubuntu-latest
    environment: Prod_secrets

    steps:
    - uses: actions/checkout@v2

    - name: Create .env file
      run: |
        echo ACCESS_TOKEN_EXPIRE_MINUTES=${{ secrets.ACCESS_TOKEN_EXPIRE_MINUTES }} >> .env
        echo ALGORITHM=${{ secrets.ALGORITHM }} >> .env
        echo DATABASE_NAME=${{ secrets.DATABASE_NAME }} >> .env
        echo DATABASE_URL=${{ secrets.DATABASE_URL }} >> .env
        echo OPEN_AI_API_KEY=${{ secrets.OPEN_AI_API_KEY }} >> .env
        echo SECRET_KEY=${{ secrets.SECRET_KEY }} >> .env

    - name: Install Heroku CLI
      run: |
        curl https://cli-assets.heroku.com/install.sh | sh

    - name: Login to Heroku
      uses: akhileshns/heroku-deploy@v3.12.12
      with:
        heroku_api_key: ${{ secrets.HEROKU_API_KEY }}
        heroku_app_name: "csp-app"
        heroku_email: ${{ secrets.HEROKU_EMAIL }}

    - name: Build and push
      run: |
        heroku container:login
        docker build -t registry.heroku.com/csp-app/web .
        docker push registry.heroku.com/csp-app/web

    - name: Release
      run: heroku container:release web -a csp-app
```

Figure 11. Build YAML Snippet For CI/CD Build.

3.2.6 Product Research and Design

The development was kicked off with two weeks of constant and deep research into the product, meant to understand the unique issues, challenges, and needs facing small and medium-sized enterprises as they were selecting CSPs. This stage included in-depth market research with the intent of identifying current gaps in market solutions and attaining general information on industries. At this point, user research was done through surveys and interviews with the company's employees at Payhippo and other start-up operators in Nigeria. Thereafter, the collected feedback was instrumental in identifying the essential features desired by the tool. Leading from this foundational data, a low-fidelity mock-up was created, clearly outlining the primary user flow and tool layout. This prototype was sketched using Balsamiq, keeping in mind practicality and giving detail precedence over detail and ease of use.

Following the sketches, the project moved into the more detailed design phase with high-fidelity prototyping completed over two-and-a-half weeks using Figma. This phase led to an elaboration of the user interface, developed from a raw concept into a professional and interactive experience that would be testable by stakeholders. High-fidelity designs developed the ideas of the first mock-ups to represent an aesthetically pleasing and usable user interface. Concurrently with the design phase, an ERD was drawn extremely carefully in Lucid to very minor details of drawing to detail the schema and to be functional in the database with accurate mapping. This phase was crucial in closing the chasm between the conceptual groundwork and its practical implementation.

3.2.7 Project Setup and Development

Core development took place over five weeks, where the tool was developed from scratch, including frontend and backend development. For the backend, FastAPI was decided to be the technology used, as it has good performance and is simple for developing RESTful APIs. Since FastAPI's operations are asynchronous, this is a good deal that allows the service to handle multiple users' requests at the same time. With this framework, solid API endpoints were developed to enable dynamic interactions of the frontend with the database and external integration like Open AI. On the front end, the choice fell on Vue.js. It was chosen due to the friendliness of the framework to integration with other technologies and very high flexibility with regard to adaptivity. Vue.js enabled building responsive, flexible user interface applications that could dynamically change from the backend without reloading the page, hence enhancing the interaction experience of users.

During this phase, the Agile development methodologies provide guidance, and a series of coding sprints allow for the continuous integration and deployment of new features. Testing at the end of each sprint should be vigorous, ensuring that the development meets predefined standards and user needs. It was an iterative process that allowed quick, successive changes and improvements in the tool such that it could fully implement the initial requirements and also be flexible enough to accommodate the changes that could arise from evolving feedback and technical requirements from users. In this development phase, the selected methodologies were then carefully merged to ensure that the pace of the project was continued, and the high usability standards set for the project beginning were kept.

3.2.8 Testing, Feedback, and Refinement

The final stage was focused on user testing, feedback, and revision, which took a period of two weeks to conduct. The focus at this stage was on the elimination of all the bugs that had been identified from the primary output. As such, it should ensure that the tool runs as it is intended to be quickly running without interference. The UI was designed to be more user-friendly in such a way that the user would not find it hard to surf through the sections required.

The implementation of a feedback section was a strategic plan to boost the usability of the tool. Participation brought about an open dialogue whereby the developer and the tool received a measure of feedback that would enable them to be able to align the capability of the tool according to satisfaction. These sensitive attention phases contributed to developing a sustainable product through testing and direct participation. The feedback used enabled timely adjustments that reflected cantering on the user of the developed project.

3.3 Target Users of the Decision-Support Tool

The tool has been initially deployed with the strategic planning for usage within Payhippo, a dynamic SME where fast decision-making and proper usage of resources are of the essence. The key users here are the team members of the company, mostly senior team engineers and management, and staff directly related to operations that need cloud solutions. These are normally arduous tasks, like selecting the most cost-effective and scalable CSP that supports their increasing technological needs. The tool's design and functionality were meant to provide swift, reliable, and data-driven suggestions aimed at improving these users' decision-making processes.

Other SMEs in the Payhippo community will also be brought on board in the alpha test and beta test internal phases of the tool. Those firms operate in industries ranging from fintech to e-commerce, which may need the right technical background to make informed choices about getting the most efficient cloud services. The tool makes this easy for them through an easy-to-understand user interface and simple guidance, which can be understood even by those with little technical background. The core needs of these users include the following:

- Cost-effectiveness: Ensuring that the selected cloud services indeed offer value for the investment.

- Scalability**:** This will require small businesses to consider CSPs with the ability to scale with the business.
- Reliability: Access to reliable cloud Services that guarantee continuous running.
- Ease of integration: Choosing CSPs who could easily become a part of their current workflow and systems.

To address these needs, the tool incorporates features such as:

- Automated CSP Evaluation**:** As per user input of needs, this tool will provide an already sorted list of CSPs along predefined criteria, which include cost, scalability, and reliability.
- Detailed CSP Profiles: Each recommendation is supported with a detailed overview of the CSP's offerings, pricing models, and user ratings to make an informed decision.
- Feedback Mechanism: Provides a mechanism for users to give feedback on their experiences so that recommendations can be improved, and the tool evolves with their needs.

4 TOOL TESTING AND FEEDBACK

This chapter describes the testing methodologies used to ensure that the CSP selection tool developed functions with high reliability and efficiency. All components and interactions of the tool were carefully tested using unit tests, integration tests, and system tests. The purpose of this testing was not only to identify and correct mistakes but also to ensure that the software works according to the conditions established at the beginning of the project. This process ensured that the software could reliably support SMEs in cloud service decisions.

Testing is a fundamental component for developing robust software, particularly for applications like this decision-support software, where accurate and reliable outputs are crucial for its users. Thus, comprehensive testing, being a shared responsibility, becomes vital since it plays a pivotal role in checking the performance and reliability of the developed tool. In fact, to a large extent, this process was significantly improved with feedback from team members at Payhippo and the thesis supervisor, Tommi Rintala. The frequency of this input-shaped relationship was based on our shared interest, where the feedback helped us identify more critical test cases and refine the developed tool to be user-friendly and accurate as initially intended.

This chapter details the structured testing and tools used and how the feedback was integrated into the testing. With high-test coverage and continuous integration and deployment processes, the assurance was of the project quality as per requirements and that the tool was dependable and user satisfaction orientated. The discussions will cover the testing stages, challenges faced, and solutions implemented to maintain the high-quality benchmarks sought for this project.

4.1 Testing Methodology

The methodology for testing the developed CSP decision-support tool was designed to include several test categories, each aiming at a different architecture level of the application. Such multi-layered testing guaranteed comprehensive tool checks from an individual code unit up to integrated system functioning.

The base of this methodology was unit testing, where, isolatedly, individual components of an application are tested to ensure that they perform exactly as expected. This phase was essential for catching and addressing granular issues within the most atomic functional blocks of the system, ensuring that small mistakes do not bubble up to big, blown-out-of-proportion system-wide issues.

This was in turn followed by integration testing to ensure that these individual components can work together effectively. This phase mostly focused on module interaction, the controllers, and integration with data sources or external APIs to guarantee the flow of data across the system without loss or corruption of its attributes.

Finally, system testing was used in testing the entire integrated application against the overall system specifications with a view of verifying that it meets all user requirements and operates correctly under simulated real-life conditions. This stage is critical in assuring that all parts of the application work in unison to meet the desired outcomes, ensuring functionality and performance up to standards expected by the stakeholders.

These testing phases are well defined and detailed; they allow for a thorough check of every aspect of the application for robustness and reliability before going live. This approach allowed for identifying and fixing possible issues in several development stages; it was important for reaching the optimisation of system performance and user experience.

4.2 Test Environments and Tools

To ensure the effectiveness of this testing methodology, several test environments were set up. This was custom-made to model different stages in application development so that every test conducted would give results as close to real tests as possible.

Unit testing and initial development took place in the development environment, which was configured to allow quick iterations and instant feedback on code changes.

The staging environment was used in all the system development and engineering. This environment was an instance of a production environment as close as possible, with no changes to production environment data. It was the primary environment for integration and system testing, providing one with good grounds to test the interaction between various components.

The production environment was set up for the final stage of system testing, load, and stress testing. It helped to pinpoint precisely how the system would behave under real-world operational loads.

4.3 Tools Used

As discussed in the previous section, it is necessary to have rigorous testing of the development and maintenance of any software solution to prove the functionality and reliability of the product. So, to support this face, a suite of tools has been chosen to support the management and automation of testing processes over the development of the cloud service provider decision support tool. Each tool for specific capabilities was chosen to support different aspects of testing from the unit level to the system level with comprehensive coverage and efficient workflows for testing. Each of the tools applied and what they represent in the testing framework is summarised as follows:

The use of Pytest tool was based on excellent performance when it comes to the handling of different test cases, from simple unit testing to very complex functional tests and even integration testing. Pytest has a simple syntax and many plugins that bring in an extra feature, thus establishing it as the top tool for creating and executing unit tests in Python.

During the integration testing phase, Postman turned out to be a helpful tool in simulating requests and responses for APIs being used. It allowed us to check if the outside interfaces of the application are right—that is to say, all the interactions with the API should work in the right way at different conditions.

In the process of running system tests, Selenium was employed to simulate the operation of a user inside the web application. This is a very helpful tool, mostly for testing a graphical user interface and user experiences, ensuring that they reach an operational and intuitive interface.

For database integration testing, MongoDB in-memory servers were used to run tests involving data operations quickly and effectively without any change to the persistent data. This structure was to execute the test several times in a controlled environment against the integrity and performance of database operations.

GitHub Actions allowed the CI/CD pipeline for automation, where the tests ran every time the code was committed. It validates all the changes in real-time, which significantly helps identify integration problems early, reducing much of the manual testing efforts.

All these tools together had a strong enough testing regimen to validate each individual functionality as part of the overall system, with the tools working together to make the sum of the parts operate coherently as a unit. This set of tools helped the team avoid falling below a reasonable level of quality and reliability at every stage of the development process.

4.4 Feedback Integration and Iterative Testing

Collection and incorporation of feedback are essential items in the testing phase since from it comes some of the most essential insights, which eventually significantly impact the iterative software development and improvement process. For example, the development of the tool for CSP decision support is based on the structured feedback of important stakeholders like Payhippo peers and the thesis supervisor Tommi Rintala on both functionality and ease of use.

The whole process of testing was, in turn, made feedback-loop oriented in a meticulous manner. After each central testing phase, it was ascertained that the members of the team give their feedback via demos at unit, integration, and system testing levels. Then, this was analysed into actionable improvements, which further led to these improvements being prioritised for the next set of sprints in development. After each phase of the development and testing, demos were held to showcase the state of the tool at that point in time. This greatly helped affirm the work in progress, collect spontaneous reactions with great spontaneity, and provide user-focused detailed feedback. Feedback was analysed to look for patterns or recurring issues that might affect the system significantly regarding user satisfaction or performance. These were based on the findings' urgency and the impact they may have on the project objectives.

Prioritised changes were quickly implemented in the development pipeline, ensuring that improvements were made continuously and efficiently. The iterative process refined features and functionalities that would make the tool meet or go beyond stakeholder expectations.

Integrating the stakeholders' feedback into the testing cycle ensured the tool was of high technical quality. This approach to development was more user-centric, making the tool more usable and valuable in real-life fields. Feedback and testing from the end-users and SMEs contributed much to refining the final product, hence ensuring that the tool developed was robust and user-friendly.

4.5 Conclusion of Testing Efforts

The comprehensive testing strategies deployed during the development of the CSP decision-support software were sufficient in ensuring it is ready for use and functions reliably. In fact, unit, integration, and system testing confirmed that it was ready and adhered to the critical requirements.

The process of testing has exposed a few things. The first and most critical is the need for early and continuous testing. It helps find potential problems before they become critical and start rearing their heads. In addition, different testing tools and methodologies have made it possible to understand the architecture and functioning of the tool widely. This helps in understanding the interaction between processes involving multiple independent components and understanding the bottlenecks, which if not identified, could result in improper operation.

Further enhance its capabilities, ensuring that it remains adaptable to new technologies and the evolving needs of users. Regular improvement contains iterative testing focusing on scalability, security improvements, and user-interface features. Besides, it shall look further at the possibilities of integrating other cloud service providers with their advanced analytical features to give users more personalised and accurate recommendations. This will involve further engagement with end-users and stakeholders in shaping these updates so that the tool can not only meet the needs of its users but also exceed their expectations.

This entire testing framework strengthened the tool's operational reliability and built a solid foundation for future evolution. With thorough testing and a keen eye on feedback, the tool is well positioned to assist with its purpose effectively within the Payhippo community, and in so doing, helping SMEs make informed decisions on their choice of cloud service provider.

5 CONCLUSION AND FUTURE WORK

This chapter provides an analysis of the development and deployment of a CSP Decision Support Tool specific to SMEs. It sums up the insights drawn from the thesis, synthesising the findings into a coherent narrative that underscores the tool's impact and utility. The subsequent sections then try to summarise the conclusions of this study, describe a few limitations, and suggest strategic areas for further improvements.

The following sections summarize the research journey, offering a conclusion and setting the stage for future developments in the realm of cloud service adoption by SMEs.

5.1 Summary of Findings

This thesis has outlined the development and deployment of a decision-support tool that would help SMEs in making a choice of CSPs that would be most suitable for their specific needs. The tool has used the state-of-the-art OpenAI's model through an API and a simple interface to make the very critical task of choosing a cloud service provider easy for users.

In the second chapter, we explore the key factors affecting cloud service provider selection based on a thorough literature review. These factors were considered under cost and budget, security and compliance, technical and customer support, scalability, reliability, vendor lock-in flexibility, and technological capabilities. By grouping these factors into financial, technical, security, support, and strategic categories, it will offer a more detailed assessment framework of these factors against the operational and growth aspirations of SMEs.

This was done in a collaborative interaction with the engineering team at Payhippo to further understand and rank these factors, ensuring the algorithm that powers the tool reflects practical operational considerations such as cost management, reliability, and service scalability perfectly. According to the ranking provided by the team, the top three factors influencing cloud service provider selection were cost and budget alignment, security and compliance, and technical and customer support. This greatly improved the decision process in such a way that it now truly reflects the daily operational challenges and business needs of the engineers.

The deployment of the tool within Payhippo greatly validated its effectiveness in streamlining the CSP selection process. Engineers reported huge time savings in the evaluation process and an increased confidence level in decisions made that were better aligned with the strategic direction of the business. This is an obvious statement reflecting the value of the tool for cloud service selection in a simplistic manner while adding value to the decision support system of the SME.

This research basically laid the groundwork for the incorporation of advanced decision support tools within the operations of an SME in a structured manner that could be aligned with the strategic needs of the SME. The findings of this research will be a sound base on which new cloud service provider selection tools will be developed.

5.2 Limitations

Although the CSP decision-support tool has been successful and has a strong capability, it has several limitations that were discovered throughout its development and deployment stages. Recognising these constraints is essential for comprehending the tool's range and identifying new avenues for enhancement.

1. Limited Input Scope**:** The tool presently functions within a limited range of inputs, which may not encompass the complete range of variables that influence cloud service provider decisions. This constraint may limit the

tool's usefulness in situations when it is essential to consider additional, industry-specific factors to make well-informed recommendations.

2. Dependency on OpenAI’s API**:** There are latency and customisation limitations when CSP suggestions are generated using OpenAI’s API. Its flexibility and response time are reduced by this reliance, which would reduce its usefulness and efficiency for making decisions in real-time.
3. Limited Customisation Options**:** The current version of the tool offers a basic degree of personalisation when it comes to choosing CSPs according to present standards. Deeper personalisation options are missing, though, which could be essential for companies with unique or specialised needs.
4. Integration with Existing Systems**:** Although the tool functions well as a stand-alone program, it is not very capable of integrating with current IT systems that SMEs employ. This restriction may prevent companies wishing to seamlessly integrate the product into their current technical environment from using it.

Addressing these limitations will require targeted enhancements and a re-evaluation of some of the foundational aspects of the tool's design and operational framework. By tackling these issues, the next iterations of the tool will greatly increase its usefulness and efficacy.

5.3 Recommendations for Future Work

Recommendations for further development of the Cloud Service Provider Decision Support Tool to promote enhanced functionality and user experience are presented in this section. These suggestions seek to lessen the noted drawbacks and expand the tool's perspective in a way that improves its ability to satisfy the evolving demands of SMEs.

1. Expanding Input Parameters: Increasing the number of input parameters in future versions of the decision tool would enable a more thorough

review of cloud service providers. This improvement would make the tool more versatile and able to meet industry-specific needs.

2. Development of a Custom AI Model: A custom AI model made especially for this decision-support tool is advised to get over the drawbacks related to using OpenAI's API. A model of this kind will not only lessen reliance on outside APIs but also maybe improve the accuracy and processing speed of the recommendations catered to particular user needs.
3. Improved API Robustness**:** To prevent making API users totally dependent on external APIs, it is imperative to develop fallback mechanisms or other solutions that will function independently if the external services are compromised. In the event that an external API service fails, it also improves dependability. Developing alliances with a few suppliers of the same API is another way to put this strategy into practice and guarantee uninterrupted service provision.
4. Easy Integration: The CSP tool can be converted into a Software as a Service with plugins and public APIs to facilitate the system's simple integration into current company systems, such as Customer Relationship Management (CRM) and Enterprise Resource Planning (ERP); this would increase the tool's acceptance rate. Now, SMEs will be able to use the tool with ease inside their present technological frameworks.
5. Continuous User Feedback Loop: To spearhead the next round of improvements, the tool should be enhanced with a continuous feedback loop that allows direct input on the CSP recommendation from end users. This loop might be automated even more inside the tool so that users can instantly report issues or propose features.

By addressing these recommendations, the decision-support tool can develop into an even more useful tool for SMEs, providing improved functionality, security, and a more customised user experience. These improvements will guarantee that the product stays current and keeps offering a lot of value in the quickly evolving cloud services market.

5.4 Conclusion

Building the unique CSP decision-support tool to optimise cost for cloud-deployed software was a journey filled with marked accomplishments and priceless insights. Unlike any other, this tool is designed to enhance the decision-making process for SMEs seeking optimal cloud services, contributing to the advancement of cloud computing utilisation. The scientific contribution of this work lies in developing a unique tool that simplifies and personalises the process for SMEs to select the best cloud service provider based on their specific needs.

The tool has shown to be useful by providing exact suggestions depending on certain SME demands, such as scalability, technical requirements, nd budget. Good comments from SMEs, particularly those in the Payhippo community, indicating a notable improvement in their Cloud Provider selection process have been received during its development and subsequent testing. The application successfully simplified CSP selection decisions and saved time and maybe money, according to the overwhelming favourable feedback from the testers during the testing period.

Technically sophisticated APIs, such as the GPT models from OpenAI's, were included into the solution to improve query processing intelligence and produce the most precise advice. The tool was built on a strong, scalable architecture that was easily modified for any future integration or growth. It made quick and responsive application development possible with maintainability and security on top of a modern technological stack (using FastAPI to construct the backend and Vue.js for the UI).

This tool provides a more personalised experience by considering a wide range of SME-specific factors compared to existing market solutions mostly built by individual CSPs, primarily focusing on service recommendations in their ecosystem. Unique from generic decision support tools that usually follow rigid algorithms, this tool matches evolving user preferences and market conditions to give up-to-date recommendations.

Feedback from initial users, including Payhippo team members and academic supervisors, was quite positive regarding the practical applicability and user-friendliness of the tool. Through the testing stage, users really liked the possibility to make changes in the input parameters and quick enough have them appearing in a recommendation for cloud services. This underscores interactions with the tool that will ensure a reasonably high chance of adequately meeting the varied SME needs.

During the exercise test, 80% accuracy has been reached in associating SMEs with their respective cloud providers. It is quite promising. The integration and unit tests cover more than 80% of the codebase and are proof of the strength and reliability of the tool.

The development process was full of learning opportunities and challenges. Major hurdles included the assurance of high standards in data security throughout the processes of API integrations and many others. However, this was met with innovative solutions and adaptive strategies—very key to the success of the project.

REFERENCES

Al-Mutawa, B. and Saeed Al Mubarak, M.M. (2024). Impact of cloud computing as a digital technology on SMEs sustainability. Competitiveness Review, Vol. 34 No. 1, pp. 72-91.

Almishal, A., & Youssef, A. (2014). Cloud Service Providers: A Comparative Study. International Journal of Computer Applications & Information Technology, 5, 2278-7720.

Arutyunov, V. V. (2012). Cloud computing: Its history of development, modern state, and future considerations. Scientific and Technical Information Processing, 39, 173–178.

Bigelow, S. J. (2022, November 15). The history of cloud computing explained. Retrieved 19 February 2024 from https://www.techtarget.com/whatis/feature/The-history-of-cloud-computing-explained

Charles, P. (2018, July 17). 7 of the Most Important Criteria for Selecting a CSP. Retrieved 21 February 2024 from https://charlesphillips.me/7-of- the-most-important-criteria-for-selecting-a-csp/

Cloudindustryforum.org. (n.d.). 8 Criteria to Ensure You Select The Right Cloud Service Provider. Retrieved 22 February 2024 from https://cloudindustryforum.org/8-criteria-to-ensure-you-select-the-right-cloud-service-provider/

Doss, A. (2021, October 5). Making a simple data pipeline, Part 4: CI/CD with GitHub Actions. Retrieved 13 April 2024 from https://innerjoin.bit.io/mak- ing-a-simple-data-pipeline-part-4-ci-cd-with-github-actions-733251f211a6.

Google Cloud. (n.d.). What is a cloud service provider? Retrieved 20 February 2024 from https://cloud.google.com/learn/what-is-a-cloud- service-provider

Hsu, C.-L., & Lin, J. C.-C. (2016). Factors affecting the adoption of cloud services in enterprises. Information Systems and e-Business Management, 14, 791–822.

Iskandar, M. (2021). Cloud accounting adoption in SMEs: An overview. International Journal of Multidisciplinary Research and Growth Evaluation, 2(1), 26-30.

Kumar, K. H., Naveen, B. R., & Savitha, J. (2022). Business Factors Challenging SMEs for Adopting Cloud-Based Solutions. International Journal of Global Business and Competitiveness, 17, 203-214.

Mathenge, J. (2021, August 4). Cloud Service Providers (CSPs) Explained. Retrieved 20 February 2024 from https://www.bmc.com/blogs/csp-cloud- service-providers/

Mathur, S. (2023). Benefits of cloud adoption by SMEs. Retrieved 20 February 2024 fromhttps://www.globallinker.com/bizforum/article/benefits-of-cloud-adoption-by-smes/50190

Naser, C. (2014). Factors Influencing the Adoption of Cloud Computing by Small and Medium Size Enterprises (SMEs). In Lecture Notes in Computer Science. Springer International Publishing.

Patel, A. R., Tiwari, R. V., & Khureshi, R. A. (2022). Comparative Study of Top Cloud Providers on basis of Service Availability and Cost. International Journal for Multidisciplinary Research (IJFMR), 4.

Payhippo (2024) GCP billing dashboard. Retrieved 12. March 2024.

Rice, I. (2023, June 3). Timeline of cloud computing. Retrieved 19 February 2024 from https://timelines.issarice.com/wiki/Timeline_of_cloud_computing

Richter, F. (2024, February 5). Worldwide market share of leading cloud infrastructure service providers. Retrieved 20 February 2024 from https://www.statista.com/chart/18819/worldwide-market-share-of-leading-cloud-infrastructure-service-providers/

Shetty, J. P., & Panda, R. (2021). An overview of cloud computing in SMEs. Journal of Global Entrepreneurship Research, 11, 175–188.

Waferwire. (2020, June 10). How to Select the Right Cloud Service Provider? Retrieved 22 February 2024 from https://www.waferwire.com/blog/how-to-select-the-right-cloud-service-provider/

Wong, M. (2021, May 12). How CSPs Can Make the Security and Compliance Evaluation Process Easier for Financial Institutions. Retrieved 20 February 2024 from https://blogs.oracle.com/cloudsecurity/post/how-csps-can-make-the-security-and-compliance-evaluation-process-easier-for-financial-institutions-v1

Zhang, G., Wang, W., & Liang, Y. (2021). Understanding the Complex Adoption Behaviour of Cloud Services by SMEs Based on Complexity Theory: A Fuzzy Sets Qualitative Comparative Analysis (fsQCA). Complexity, 2021, Article ID 5591446.