Using Artificial Intelligence (AI) to Make Cloud Computing More Sustainable

Introduction

The integration of cloud computing into internet technology revolutionised web services as we now know them. Cloud computing, as defined by the National Institute of Standards and Technology (NIST), is 'a model for the enabling of ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction'. In simpler terms, cloud computing allows organisations to share infrastructure and resources over the internet.

In a survey conducted by PwC, 78% of business leaders said they have adopted cloud technology into most areas in their organisation, but more than half of these responses said they cannot yet see the outcome of their investment (PricewaterhouseCoopers, 2023). Cloud computing is not being utilised to its full potential. Organisations are not to blame for this – cloud deployment has many different aspects to consider, and specialists are not always available to guide them. Hence, an Artificial Intelligence (AI) agent that could compute the technical decisions on behalf of an organisation would be extremely beneficial.

Additionally, the cloud consumes massive amounts of resources. Global water consumption of cloud data centres exceeds 560 billion litres annually (Kenny, 2025). However, cloud computing has the potential to reduce the environmental impacts of the technology industry. Moving commonly used software applications to the cloud could decrease energy usage by 87% (Beardmore, 2020). With the environment becoming an increasingly high priority for many organisations, sustainable development is key. In this project, the specific focus will be limiting water and energy usage. While these are separate issues to be tackled, they also complement each other, as reducing water waste also reduces energy usage in most cases.

When looking at energy conservation, virtualisation is a technique that has the potential to create a huge impact. Virtualisation lowers energy use by reducing the number of active servers, cuts cooling requirements by generating less heat and improves hardware utilisation with tools including dynamic resource allocation (Ambros, 2025). A quantitative measure of energy use that can be used is Power Usage Effectiveness (PUE). PUE is a metric used to determine the energy efficiency of a data centre and is calculated by dividing the total amount of power entering a data centre by the power used to run the IT equipment within it (Kirvan, 2025). Another quantitative figure of energy efficiency is server utilisation, which is a measure of how much a server's processing capacity is being used over a period of time (Wilson, 2024). A widespread problem amongst data centres is that assets are powered on but are doing little to no useful work, known as "ghost servers" (Wilson, 2024). If organisations could share server space dynamically, this problem will be minimised as any redundant space could be used by a different organisation. Maximising utilisation in this sense would save both energy, cost and e-waste, another large environmental concern from cloud computing. An additional energy saving technique for organisations is simply turning off resources at night and on the weekends, which can save up to 65% of power usage (Kuss, 2021). These resources are wasted from being left idle, and solving this problem using automation can save both economic and environmental costs.

Regarding water waste, cooling systems have the largest contribution. Large data centres can consume up to 5 million gallons per day, equivalent to the water use of a town populated by 10,000 to 50,000 people (Yañex-Barnuevo, 2025). Water Usage Effectiveness (WUE) is a metric for measuring the water efficiency in data centres by comparing water used for cooling with energy consumed by IT equipment (Higgins, 2024). This works in the same way as PUE, providing a quantitative measure of water waste. There are two ways to reduce water usage – increase the efficiency of the system or recycle the water. Reducing overall water consumption is important to prevent areas with large data centres experiencing regional water stress.

Therefore, this project will investigate whether a weighted scoring system combined with a large language model call can produce reliable, sustainability-focused cloud deployment plans for startups.

Background Research

Prior to implementation, the specifics and logic of the decision-making assistant must be established. The organisation will input their details, and the script will produce a score for each cloud feature based on the organisation's specific needs. It will then use these scores to propose the ideal cloud model for the company. Scoring will work using a weighted criterion, with organisations deciding which factors they would like to give more weight to. For example, a company with sustainability as their top priority will be able to give a higher weighting to reducing energy and water consumption with the PUE and WUE metrics as outlined in the introduction.

Using a scoring matrix in the decision-making process has many advantages. When working with large language models, multi-criteria decision matrices have been identified as an established approach for cloud service selection (Grgurević and Kordić, 2017). This justifies the effectiveness of this approach for this project.

This tool has been scoped towards small organisations, such as startups, that do not have the time nor the funding to hire cloud consultants but are looking to deploy cloud technologies to improve the efficiency and scalability of their business. This is due to the assumption that large companies already use cloud technology, or in the rare case they do not, would outsource their cloud deployment.

Startups tend to avoid large upfront costs and look for on-demand scalability (Northrop, 2023). If we look at startup cloud usage patterns, there is a clear problem of the lack of cost visibility, caused by the dynamic and complex charges of cloud services (CloudSyntrix, 2025). Therefore, the AI agent must understand the priorities as a startup to keep costs low while also dynamically monitoring the changing costs of cloud services.

Comparing the various deployment models, the first stage would be to decide between a private, public or hybrid cloud, or even the multicloud approach where appropriate. In a private cloud, the organisation controls the infrastructure to deliver the IT resources whereas in a public cloud, it is provided to them by a third-party company as a fully managed service (Amazon Web Services, 2025). A hybrid cloud, as defined by Google Cloud (n.d.), is 'a mixed computing environment where applications are run using a combination of computing, storage and services in different environments – public clouds and private clouds, including on-premises data centres or "edge" locations'. In short, hybrid clouds are exactly as the name suggests, finding a middle ground between private and public grounds. Having the advantages from both cloud infrastructures often makes hybrid clouds the preferred deployment model. An upcoming infrastructure is the idea of a multicloud, where organisations use services from at least two cloud providers to run their applications, either public clouds, private clouds or both. This allows them to select the capabilities that best suit the specific needs of the business and therefore provide the greatest flexibility (Google Cloud, n.d.). However, this is the most complex to set up and requires in depth understanding of cloud computing. Hence, this AI agent will need to interpret when a multicloud approach is desirable and when it would be better to stick to one of the more traditional methods.

Cloud computing has three main service models, Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS) and Software-as-a-Service (SaaS). In Nayan Ruparelia's 'Cloud Computing' (2022), he adds two further levels known as Information-as-a-Service (INaaS) and Business Process-as-a-Service (BPaaS). These can be thought of as different levels of abstraction that cloud services can offer. IaaS and PaaS are priced on a utility basis, whereas the rest of the service models are priced per use. This is important to consider when evaluating financial factors, particularly because startups will likely have lower budgets. For a startup, the more abstract features that come with the outer models of Figure 1 won't be necessary, so the agent will only need to recommend one of the three core service models. Beginning with IaaS, it provides all the hardware, servers and storage, setting up the frameworks for cloud computing, whereas PaaS also includes the operating system and any other core applications. Finally, Ruparelia explains that SaaS adds the customisable aspect which will serve particularly useful to businesses. It 'includes hosted applications that fulfil a function' which means the cost of installing and maintaining the application and its supporting hardware infrastructure are placed on the cloud service provider (Ruparelia, 2022). The requirements of a business will likely to fall into the SaaS abstraction level, because without a team of technology specialists, it is much easier to give the responsibility of software management to a third-party. While this raises cloud costs, for a small business, it is more financially viable than hiring full-time technology experts.

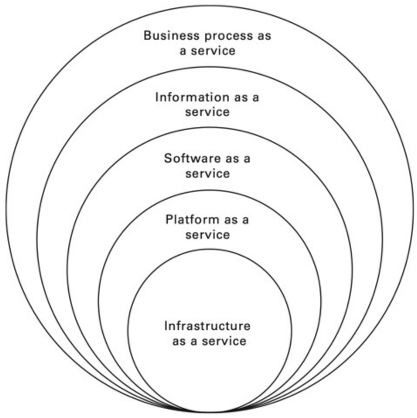

Figure 1: Levels of Abstraction (Ruparelia, 2022)

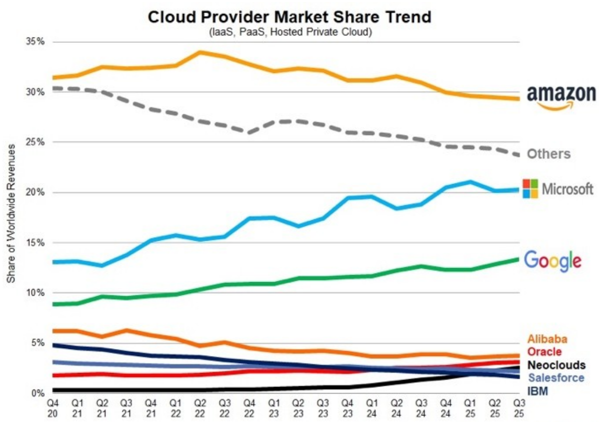

Figure 2: Cloud Provider Market Share Trend (Dinsdale, 2025)

Next, we need to evaluate existing cloud services. As shown in Figure 2, the three largest cloud providers are Amazon Web Services, Microsoft Azure and Google Cloud. Together, they accounted for 63% of enterprise spending on cloud infrastructure in Q3 of 2025 (Dinsdale, 2025). The most convenient visual representation to compare these services is a table which has been created below using information found in the official documentation of each provider. This information will be directly transferred into the decision matrix in the next section, to be used by the AI agent as part of its decision-making process.

	Amazon Web Services - S3 (AWS, n.d.)	Microsoft Azure - Blob Storage (Microsoft Learn, n.d.)	Google Cloud - Storage (Google Cloud, n.d.)
Costs	Pay as you go: up to 50TB at $0.023/GB per month Flat rate plans: roughly between $200 to $1000 a month for startups	Pay as you go: up to 100TB $0.036-$0.15/GB per month Flat rate plans: roughly between $676 to $1545 a month for startups	Pay as you go: $0.004-$0.02/GB per month Multiple startup offerings using a credit-based model
Scalability	Anything from personal use to large businesses	Massively scalable blob storage	Scale quickly from zero to high request volumes
Geographic Coverage	Isolated local zones: North America, South America, Africa, Asia Pacific, Europe and the Middle East	Works across the Americas, Europe, the Middle East, Africa and Asia Pacific	130 zones across the Americas, Europe, Asia Pacific, the Middle East and Africa Multi-region tools available
Sustainability goals	100% renewable energy, water positive by 2030, PUE of 1.15 (Amazon, 2025)	Net zero by 2030, 24/7 carbon-free energy, 64% water replenishment rate (Google, 2025)	Carbon negative by 2030, remove all historical carbon by 2050, 100% renewable energy, zero waste by 2030, water positive by 2030 (Microsoft, 2025)

Figure 1: Provider Comparison Table

When looking at existing AI cloud advisory agents, there are two products which stood out. The first is ‘AI Cloud Advisor’ (n.d.) which is designed to assist with ‘cloud best practices, troubleshooting, and architecture recommendations’. However, this tool fails to consider sustainability, which is a pillar of this project. The second existing AI cloud assistant is a product of Alibaba Cloud known as ‘Intelligent Advisor’ (n.d.) and is specifically targeted at Alibaba cloud users, which is less than 5% of the market share according to Figure 2. These products show there is a clear gap for an AI agent that advises specifically on sustainable cloud deployment, as well as the usual factors of cost, scalability and the like.

The initial plan was to create this tool using the development platform Vertex AI Agent Builder by Google Cloud. It is an ‘open and comprehensive platform that empowers enterprises to rapidly build, scale and govern enterprise-grade agents grounded in your enterprise data’ (Google Cloud, n.d.). This decision was made based on the following combination of factors. Firstly, the tool is extremely well documented which makes it easy to follow and develop with. Secondly, as shown above, Google Cloud services are amongst the top 3 in the world, adding credibility to Vertex AI. It also has the capability to scale AI agents, providing the infrastructure needed for this tool to be used in practice. Thirdly, targeted at developers, Vertex AI has comprehensive technical functionalities which allow the attention for detail that is crucial to create reliable agentic systems. Within the top three cloud providers also stand Amazon and Microsoft, with their competitor tools, Amazon SageMaker and Azure ML respectively. When evaluating Vertex AI with its competitors, it had the advantage of being tailored to create agentic AI rather than machine learning models. Overall, it appeared that Vertex AI stood out as the platform to develop this tool. However, this was re-evaluated during the development phase which is explained in the following section.

When handling business data, regulatory and compliance context is also crucial. General Data Protection Regulation (GDPR) is one of the main legal frameworks concerning data privacy. Under Article 5, GDPR requires that any data that is collected is done so for a specified purpose (European Parliament and Council of the European Union, 2016). The most relevant section of GDPR for this project is Article 22, which looks at the involvement of data in automated decision-making. To ensure a fully GDPR compliant end-product, all of the data collected is used to consult on cloud deployment solutions and no data is stored in the program.

Development Phase

Phase One: Designing the Decision Matrix

When planning a program, the primary task is defining a list of inputs and outputs. The table below details all the inputs that the agent will take in from the user, and these will be used to form the output.

These factors were selected based on the features that cloud providers offer and how quantifiable they are. Budget was the most obvious factor to consider, because this will vary greatly for smaller firms depending on grants, investment and their initial revenue success. The next factor is storage requirements – a resource-heavy AI startup will have very different needs to a web-based platform that only stores limited data. The example of the multicloud model demonstrates the necessity of considering technical expertise. As previously explained, this model allows for the greater flexibility because different features can be selected depending on the one that is most appropriate for each task. However, high technical understanding is required to successfully manage a multicloud, else it could be less effective than an alternative approach. Then, there are two optional inputs, security requirements and growth expectation. This is to give the agent contextual understanding of the company, and while not necessary, the user may receive a higher quality response if they provide these additional details. Finally, the sustainability weighting is requested; the unique aspect of this project, and the angle which the agent diverts maximum attention to. Out of these factors, the user will also inform the program of what their top priority is, so it can be weighted higher during the calculation stage.

Input Name	Data Type	Range of Values	Required (Y/N)
Monthly budget	Float – needs to be converted to USD before comparison	0 < X	Y
Data storage requirements	Integer – rough value measured in terabytes	0 < X	Y
Technical expertise level	String – one of the listed options	None, Basic, Moderate, High, Expert	Y
Security requirements	Integer – reflects the required strength of data security	0 <= X <= 10	N
Sustainability weighting	Integer – converts to a percentage	0 <= X <= 100	Y
Growth expectation (5 year time frame)	Integer – estimated growth multiplier	0 < X	N
Top priority	String – one of the listed options	Cost, Security, Sustainability, Scalability	Y

Figure 2: Input Table

Criterion	AWS	Azure	Google Cloud
Cost	8	5	10
Scalability	9	9	9
Sustainability	7	7	8
Geographic Coverage	9	9	10
Security	9	9	9

Figure 3: Provider Scores

The next part of the decision matrix was scoring each of the three main providers using the information in the first table. Each provider was scored against cost, scalability, sustainability, geographic coverage and security. Each category is scored between 1-10 and has been assigned weights based on the importance of each category to a startup. These are multiplied by weights derived from the user's inputs to calculate the final scores. This is much more effective than hard coding the weights because it makes the tool versatile and will allow different businesses' needs to be catered to. Then, these weights are multiplied by the score for each supplier to calculate the final weighted score.

For the next two sheets of the matrix, the different models of deployment and different service models were allocated ratings based on the qualitative findings in the Background Research section. Below are the final tables:

Criterion	Public	Private	Hybrid	Multicloud
Cost	8	3	5	4
Technical Complexity	9	4	5	2
Scalability	9	5	8	10
Sustainability	7	5	7	6
Security	5	10	8	7

Figure 6: Deployment Scores

Criterion	IaaS	PaaS	SaaS
Cost	7	6	4
Technical Complexity	3	6	9
Sustainability	6	7	7
Customisability	10	7	5
Management Overhead	3	6	9

Figure 7: Service Scores

Please note that for all three tables above, cost has been compared monthly per GB. The scoring matrix has been created using Excel, for better GitHub compatibility as opposed to its competitor, Google Sheets. This file can be viewed in the GitHub repository by A. Gupte linked in the bibliography.

Phase Two: Training the Agent

The agent has been programmed in Python using the Google Cloud developer tools. First, the user is prompted for the required inputs. The comparison matrix created in Phase One is then read, and finally a decision is presented by the agent. The code written in this section can also be found in the GitHub repository by A. Gupte linked in the bibliography.

The primary source of information for Vertex AI is the official documentation titled 'Training a model using Vertex AI and the Python SDK'. As shown in Figure 8, the documentation details all the necessary steps for this process, which was one of the factors for choosing to use Vertex AI over its competitors. For the prerequisite step, a Google Cloud project was created and the Vertex AI API was enabled. However, when beginning development of the agent using Vertex AI (Figure 9), the concept of the decision-matrix appeared to be more code-heavy than originally thought, meaning a Python script and API call would be better suited to its requirements. This would mean the code is making calculated decisions rather than speculation, which reduces the likelihood of hallucinations and therefore errors. Hence, the plan was adapted to use the Gemini API key to generate the natural-language side of the agent.

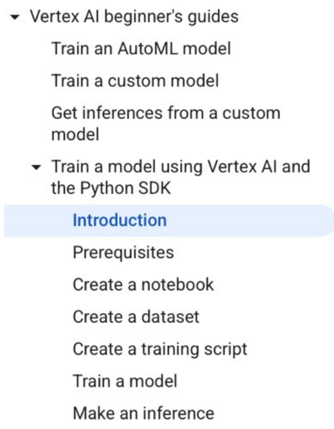

Figure 8: Vertex AI Documentation (Google Cloud n.d.)

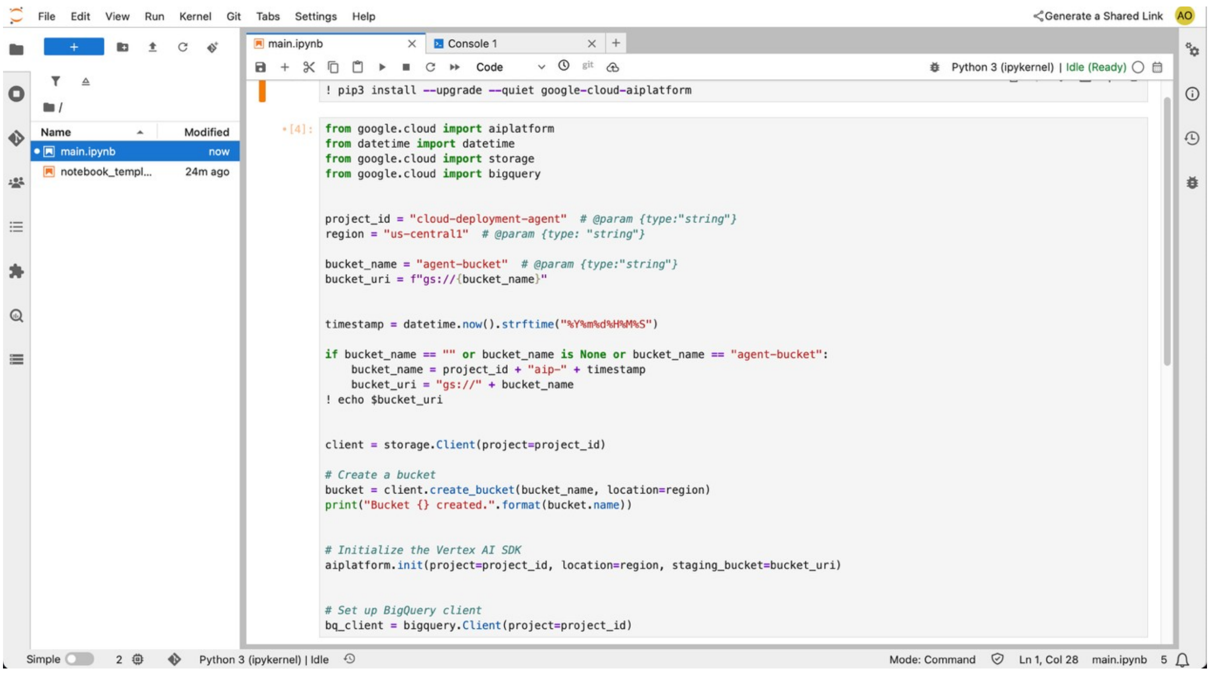

Figure 9: Vertex AI Code

All the information for the Gemini API was found from the official documentation, which included a full walkthrough in Python and JavaScript (Google, 2026). Python was selected for this project because it dominates the AI and machine learning space for its intuitive, clean layout (Cisco DevNet, 2026). To create the agent, two extra packages are required as shown in Figure 11. The first package is for reading the data from Excel, known as openpyxl. The second package is specifically for using the Google API. Initially, google-generativeai was downloaded as per the documentation. However, when testing the code, the newer library google-genai was found to be more suitable because it supports Vertex AI authentication rather than standard API calls, which means Gemini can be called free of charge.

Figure 11: Python Package Installations

The Python script has been written as six core subroutines, as well as a main() procedure to call the rest of the code. First, the user is prompted to input the required criteria and given options to select from where possible. The program carries out appropriate verification and validation of inputs so it does not crash further down the line. Using these inputs, weights are created based on the business' priorities. These are multiplied by the data that has been read from Excel to create scores for each cloud feature. The feature that scores the highest 'wins', and this data, alongside the original inputs, is added to the prompt for the agent.

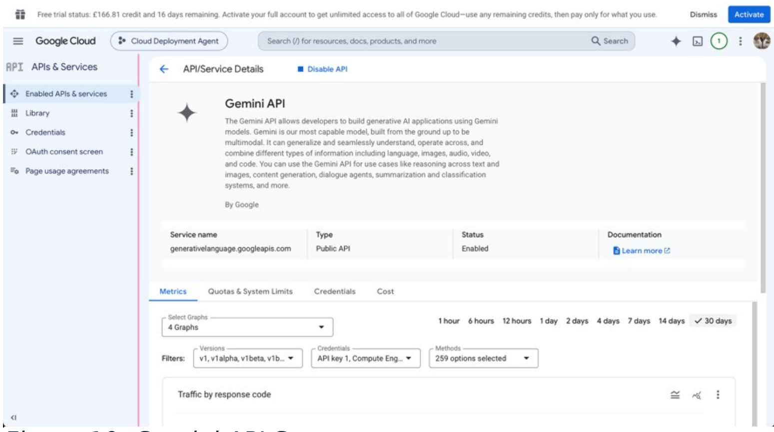

Figure 10: Gemini API Setup

In the code below, the prompt is shown as a string encompassing all the research and calculations that have already been carried out. Formulating a well-structured prompt is increasingly important when working with large language models, because what is produced will directly impact the strength of advice from the agent (White et al, 2023). To provide context, the agent has been told it is a specialised cloud advisor, as seen on line 206. This constitutes both a role-based and contextual prompt because the agent is asked to assume a viewpoint but is also given additional information about the scenario. Specific details are then provided for the next six lines, focusing on how the output should be structured and which factors to take into consideration to increase clarity. In this version, there isn't capability to build on the conversation because it uses a single prompt, however, this is a strategy that

```
199 def get_recommendation(winners, inputs):
200     """
201     create prompt and call Gemini API
202     :param winners: dictionary of strings
203     :param inputs: dictionary of strings
204     """
205 
206     prompt = "You are an experienced, specialised cloud deployment advisor for startups."
207     prompt += f" Give them a detailed explanation of why the chosen stategy is best suited for them and produce a step by step plan for deployment."
208     prompt += f" The plan should be extremely detailed, and also include further tips for management of the cloud infrastructure."
209     prompt += f" Reference their specific inputs including priorities and constraints."
210     prompt += f" Specifically reference sustainability from water and energy usage - the aim should be to save these resources."
211     prompt += f" Consider techniques such as virtualisation and simply switching off power at night."
212     prompt += f" Give advice on how to consistently maintain a reduction in energy/water usage."
213 
214     prompt += f" Monthly budget: {inputs['budget']} USD."
215     prompt += f" Storage requirements: {inputs['storage']} TB."
216     prompt += f" Technical expertise: {inputs['expertise']}."
217     if inputs['security'] is not None:
218         prompt += f" Security requirements (between 0 and 1): {inputs['security']}."
219     prompt += f" Sustainability requirements (between 0 and 1): {inputs['sustainability']}."
220     if inputs['growth'] is not None:
221         prompt += f" Growth multiplier expectation for the next five years: {inputs['growth']}."
222     prompt += f" Their top priority is {inputs['top_priority']}."
223 
224     prompt += f" Scores and weights have been calculated and the winners as follows. Use this to structure the advice."
225     prompt += f" Provider: {winners['Provider']}."
226     prompt += f" Provider: {winners['Deployment']}."
227     prompt += f" Provider: {winners['Service']}."
228 
229     client = genai.Client(vertexai=True, project="cloud-deployment-agent", location="us-central1")
230     gemini_call = client.models.generate_content(model="gemini-2.5-flash", contents=prompt)
231     return gemini_call.text
```

Figure 12: The get_recommendation() Function

can be looked at for further versions of the software.

Testing

When writing the code, the test case that was used was written for a business with a $200 monthly budget, 3 TB storage requirements, 5/10 security, 75/100 sustainability and scalability as their top priority. The results were outstanding – the agent produced a structured, detailed (over 4000 words) plan for both the short-term and long-term of the business which can be viewed under tests/initial_text.docx in the same GitHub repository. If we analyse the results in more detail, the agent successfully encompasses the narrative of a cloud advisor. It breaks its feedback into three sections, why the chosen strategy is the most appropriate for the business, a deeper dive into sustainability practices and a step-by-step plan for deployment. The level of detail for all three sections is immaculate – the justification explores the provider, in this case Google Cloud, from a sustainability, scalability, cost-effectiveness, technical expertise and security perspective. It then presents its conclusion for the service model, namely SaaS, and explains why this aligns with the startups profile and priorities. In the second part of the plan, it explains Google's intrinsic sustainability advantages and then goes further into techniques that the startup can incorporate into their practices. The plan covers consistency and long-term sustainability management. Finally, the third section is a step-by-step cloud deployment plan. It begins day by day and then progresses into

Your startup's journey into the cloud, guided by Google Cloud, a Public Cloud model, and a SaaS-first approach, is strategically sound. By rigorously adhering to this detailed plan, embracing sustainable practices like serverless architectures and automated shutdowns, and diligently managing your budget and resources, you will build a scalable, resilient, and environmentally responsible foundation for your business.

This strategy empowers your team to focus on innovation, knowing that your infrastructure is optimized for your top priority (scalability) while meeting your critical sustainability goals and tight budget. I wish you the very best in your deployment and am confident this approach will lead to success.

weeks and even months later, covering database strategy, networking and security configuration. The report concludes with a short summary and an encouraging note to the start up, reiterating its narrative of a cloud advisor (Figure 13). Overall, at this occasion, the agent has exceeded expectations producing a full report with a high level of detail and clarity. To further test the agent, the following real-life test cases were then trialled, and the results are described below:

Test Case 1: Airbnb

Airbnb began with two hosts welcoming three guests to their home in San Francisco and grew to over 2.5 billion guest arrivals as of 2025 (Statista Research Department, 2026). As an early-stage startup, Airbnb had rapid user growth and chose to prioritise scalability, selecting an AWS public cloud with a mix of IaaS/PaaS (Ceresnak, 2024). This is going to be the first test case for the agent, and the advice will be compared to the route that the business actually took. The inputs replicating Airbnb's initial profile are shown in Figure 14.

```
please enter your monthly budget (in USD): 500
please enter your rough storage requirements (in terabytes): 1
please enter your level of technical expertise (none / basic / moderate / high / expert): basic
please enter your security requirements between 1 and 10, or hit ENTER to skip: 8
please enter your sustainability_requirements between 1 and 100: 25
please enter your growth multiplier expectation for the next five years, or hit ENTER to skip: 10
please enter your top priority (cost / security / sustainability / scalability): scalability
```

Figure 14: Airbnb Test Case Inputs

The agent outputted a plan detailing a recommended approach as well as justification for selecting it. The similarity between the agent's advice and the route Airbnb actually took is the public cloud deployment type. However, the agent recommended Google Cloud as the provider rather than AWS. While this is a subjective decision, it's important to note reasons for this variation. Firstly, Airbnb was founded in 2007 (Statista Research Department, 2026) whereas this advice was given in 2026 – both companies have evolved greatly in this time and while one may have been objectively better at the time, it may not be anymore. Another possible reason is the Gemini API bias which is explored in the conclusion. For service models, the agent recommended SaaS as opposed to the IaaS/PaaS route that Airbnb took. To strengthen this response, it could be useful to provide the agent with more contextual information because choosing a service model is extremely specific to individual requirements of a business.

Test Case 2: Spotify

In 2016, Spotify swapped from AWS to Google Cloud to "remove the operational complexity from our ecosystem" in the words of Tyson Singer, vice president of technology and platform (Google Cloud, n.d.). Another contributing factor to the migration were their carbon neutral goals, suggesting sustainability was a high priority for the company. Therefore, this will be a particularly interesting case to see if the agent recommends the AWS or the Google Cloud route. Spotify used a public cloud with a PaaS model (Google Cloud, n.d.).

```
please enter your monthly budget (in USD): 50000
please enter your rough storage requirements (in terabytes): 100
please enter your level of technical expertise (none / basic / moderate / high / expert): expert
please enter your security requirements between 1 and 10, or hit ENTER to skip: 9
please enter your sustainability_requirements between 1 and 100: 80
please enter your growth multiplier expectation for the next five years, or hit ENTER to skip: 25
please enter your top priority (cost / security / sustainability / scalability): scalability
```

Figure 15: Spotify Test Case Inputs

The results recommend a public Google Cloud, exactly the route the Spotify ended up taking. This shows credibility of the agent's results, especially because Spotify made an active decision to migrate their infrastructure. It does however recommend the IaaS model rather than PaaS, but these are only one level of abstraction apart and have mostly overlapping features. Therefore, despite Spotify being larger than a startup when it switched provider, the deployment plan written by the agent is very accurate and aligns with their real growth path.

Test Case 3: Duolingo

In 2025, Duolingo surpassed 50 million daily active users, revenue by 40% in the first nine months of the year compared to same period the year before (Duolingo, 2025). They used an AWS public cloud with a PaaS model (AWS, 2017). Figure 16 shows the selected inputs for Duolingo in its early stages.

```
please enter your monthly budget (in USD): 5000
please enter your rough storage requirements (in terabytes): 10
please enter your level of technical expertise (none / basic / moderate / high / expert): expert
please enter your security requirements between 1 and 10, or hit ENTER to skip: 8
please enter your sustainability_requirements between 1 and 100: 20
please enter your growth multiplier expectation for the next five years, or hit ENTER to skip: 20
please enter your top priority (cost / security / sustainability / scalability): cost
```

Figure 16: Duolingo Test Case Inputs

For this result, the agent has recommended a public cloud, however using a different provider; Google Cloud. The recommended service model is IaaS, which once again incorporates most of the features from the PaaS model used by Duolingo. Therefore, the agent's advice aligns relatively well with the route that Duolingo took in real-life.

Conclusion

This project investigated whether a weighted scoring system combined with a large language model call could produce reliable, sustainability-focused cloud deployment plans for startups. The agent produces an extremely detailed (3000-4000 word) plan for minimal (5-7 word) input. These are subjective results, therefore the crucial part is the justification for which it has explained for all three major decisions, as well as suggesting complementary strategies for the long-term. Its encouraging tone is professional yet meaningful, encompassing the narrative of a professional cloud advisor.

While the agent performed well in most aspects, throughout the results there is a clear trend in the recommendation of Google Cloud. A reason for this could be because the code uses a Gemini API call, which is a Google service. Hence, it may be unconsciously biased (or even hard-coded by the organisation) towards recommending its own products. A future improvement could be using multiple API calls to different agents and combining the results to create a final plan. Nevertheless, these results show clear promise, and the developed product has market potential, especially with further fine-tuning and reinforcement learning of the AI agent.

Particularly focusing on sustainability, the ideas proposed in the test plans reduce both energy and water usage using the PUE and WUE metrics. Using this tool has a clear environmental advantage, which will successfully improv the sustainability of cloud computing practices if carried out.

While the results have been summarised as best as possible, it is highly recommended to view these documents directly from the GitHub repository (A. Gupte) to understand the level of detail and clarity produced by the agent. The code is entirely open-source so can be tested locally if wished.

Bibliography

AI Cloud Advisor (n.d.). *Custom AI-Powered Cloud & Microservices Advisor.* Available at: https://aicloudadvisor.net. Last accessed at 22/03/26.

Alibaba Cloud (n.d.). *Intelligent Advisor.* Available at: https://www.alibabacloud.com/en/Product/advisor?_p_lc=1. Last accessed at 22/03/26.

Ambros (2025). *Energy Efficiency in Data Centres: Virtualization Benefits.* Available at: https://www.serverion.com/uncategorized/energy-efficiency-in-data-centers-virtualization-benefits/. Last accessed at 02/02/2026.

Amazon (2025). *Amazon Sustainability Report – AWS Summary. Available at:* https://sustainability.aboutamazon.com/2024-amazon-sustainability-report-aws-summary.pdf. Last accessed at 06/06/2026.

Amazon Web Services (2025). *What's the Difference Between Public Cloud and Private Cloud?* Available at: https://aws.amazon.com/compare/the-difference-between-public-cloud-and-private-cloud/. Last accessed at 02/02/2026.

AWS (2017). *Using AI to Teach 300 Million People.* Available at: https://aws.amazon.com/machine-learning/customers/innovators/duolingo/. Last accessed at 08/06/2026.

AWS (n.d.). *Welcome to AWS Documentation.* Available at: https://docs.aws.amazon.com. Last accessed at 22/03/26.

Beardmore, Adele (2020). *Uncovering the Environmental Impact of Cloud Computing.* Available at: https://earth.org/environmental-impact-of-cloud-computing/. Last accessed at 25/02/2026.

Ceresnak, Roman (2024). *AWS for Startups: Scaling Your Business with Cloud Technology.* Available at: https://medium.com/codex/aws-for-startups-scaling-your-business-with-cloud-technology-824c8b675f5b. Last accessed at 07/06/2026.

Cisco DevNet (2026). *What is the Best Programming Language for AI? - Best Programming Language for AI - Cisco DevNet.* Available at: https://developer.cisco.com/articles/best-programming-language-for-ai/what-is-the-best-programming-language-for-ai/#what-is-the-best-programming-language-for-ai. Last accessed at 01/06/2026.

CloudSyntrix (2025). *The Hidden Cloud Cost Crisis: How Startups Are Bleeding Money in 2025.* Available at: https://www.cloudsyntrix.com/blogs/the-hidden-cloud-cost-crisis-how-startups-are-bleeding-money-in-2025/. Last accessed at 02/02/2026.

Dinsdale, John (2025). *Cloud Market Share Trends – Big Three Together Hold 63% while Oracle and the Neoclouds Inch Higher.* Available at: https://www.srgresearch.com/articles/cloud-market-share-trends-big-three-together-hold-63-while-oracle-and-the-neoclouds-inch-higher. Last accessed at 12/03/2026.

Duolingo, Inc (2025). *Duolingo Surpasses 50 Million Daily Active Users, Grows DAU 36% and Revenue 41% in Third Quarter 2025 Year over Year.* Available at: https://investors.duolingo.com/newsreleases/news-release-details/duolingo-surpasses-50-million-daily-active-users-grows-dau-36. Last accessed at 08/06/2026.

European Parliament and Council of the European Union (2016). *Regulation (EU) 2016/679 — General Data Protection Regulation*. Available at: https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:32016R0679. Last accessed at 22/03/2026.

Gupte, Anika (2026). *cloud-deployment-agent*. Available at: https://github.com/anikagupte/cloud-deployment-agent. Last accessed at 07/06/2026.

Google (2025). *2025 Environmental Report*. Available at: https://sustainability.google/reports/google-2025-environmental-report/. Last accessed at 07/06/2026.

Google (2026). *Gemini API quickstart*. Google AI for Developers. Available at: https://ai.google.dev/gemini-api/docs/quickstart?lang=python. Last accessed at 01/06/2026.

Google Cloud (n.d.). *Build, scale, and govern your enterprise grade AI agents*. Available at: https://cloud.google.com/products/agent-builder. Last accessed at 11/03/2026.

Google Cloud (n.d.). *Generative AI beginner's guide*. Available at: https://docs.cloud.google.com/gemini-enterprise-agent-platform/models/beginners-guide. Last accessed at 07/06/2026.

Google Cloud (n.d.). *Google Cloud Documentation*. Available at: https://docs.cloud.google.com/docs. Last accessed at 22/03/2026.

Google Cloud (n.d.). *How Application Default Credentials works*. Available at: https://docs.cloud.google.com/docs/authentication/application-default-credentials. Last accessed at 07/06/2026.

Google Cloud (n.d.). *Spotify: The future of audio. Putting data to work, one listener at a time.* Available at: https://cloud.google.com/customers/spotify. Last accessed at 08/06/2026.

Google Cloud (n.d.). *Train a model using Vertex AI and the Python SDK*. Available at: https://docs.cloud.google.com/vertex-ai/docs/tutorials/tabular-bq-prediction. Last accessed at 23/03/2026.

Google Cloud (n.d.). *What is a Hybrid Cloud?*. Available at: https://cloud.google.com/learn/what-is-hybrid-cloud. Last accessed at 11/03/2026.

Google Cloud (n.d.). *What is multicloud?*. Available at: https://cloud.google.com/learn/what-is-multicloud. Last accessed at 11/03/2026.

Google Cloud (n.d.). *You are now authenticated with the gcloud CLI!* Available at: https://docs.cloud.google.com/sdk/auth_success. Last accessed at 07/06/2026.

Grgurević, Ivan and Kordić, Gordana (2017). *Multi-criteria Decision-making in Cloud Service Selection and Adoption*. Available at: https://www.academia.edu/51255599/Multi_criteria_Decision_making_in_Cloud_Service_Selection_and_Adoption. Last accessed at 09/06/2026.

Higgins, Andrew (2024). *What Is Water Usage Effectiveness (WUE) in Data Centers?* Available at: https://blog.equinix.com/blog/2024/11/13/what-is-water-usage-effectiveness-wue-in-data-centers/. Last accessed at 02/02/2026.

Kenny, Rich (2025). *Report: Water Use in AI and Data Centres*. Available at: https://assets.publishing.service.gov.uk/media/688cb407dc6688ed50878367/Water_use_in_data_centre_and_AI_report.pdf. Last accessed at 25/02/2026.

Kirvan, Paul (2025). *What is PUE (power usage effectiveness)?*. Available at: https://www.techtarget.com/searchdatacenter/definition/power-usage-effectiveness-PUE. Last accessed at 02/02/2026.

Kuss, Edwin (2021). *How to stop paying for idle cloud instances and services*. Available at: https://hystax.com/how-to-stop-paying-for-idle-cloud-instances-and-services/. Last accessed at 02/02/2026.

Mell, Peter and Grace, Timothy (2011). *The NIST Definition of Cloud Computing*. Available at: https://nvlpubs.nist.gov/nistpubs/legacy/sp/nistspecialpublication800-145.pdf. Last accessed at 02/02/2026.

Microsoft (2025). *2025 Environmental Sustainability Report*. Available at: https://www.microsoft.com/en-us/corporate-responsibility/sustainability/report/. Last accessed at 06/06/2026.

Microsoft Learn (n.d.). *Azure documentation*. Available at: https://learn.microsoft.com/en-us/azure/?product=popular. Last accessed 22/03/26.

Northrop, Jett (2023). *The Impact of Cloud Computing on Startup Ecosystems and Entrepreneurial Innovation*. Available at: https://www.researchgate.net/publication/391162451_The_Impact_of_Cloud_Computing_on_Startup_Ecosystems_and_Entrepreneurial_Innovation. Last accessed at 02/02/2026.

PricewaterhouseCoopers (2023). *The Cloud – most businesses use it, but are they using it to its full potential?*. Available at: https://www.pwc.co.uk/services/value-creation/insights/unlocking-value-of-cloud-investments.html. Last accessed at 25/01/2026.

Python Software Foundation (n.d.). *Python 3.14.5 documentation*. Available at: https://docs.python.org/3/. Last accessed at 01/06/2026.

Ruparelia, Nayan (2022). *Cloud computing*. Cambridge, Massachusetts ; London, England: The Mit Press.

Statista Research Department (2026). *Airbnb – statistics & facts*. Available at: https://www.statista.com/topics/2273/airbnb/?srsltid=AfmBOor2ab9WwcPa9JCOMqlA7GtF7fyAitUq34auK0NDAw7a2LloZtXj. Last accessed at 08/06/2026.

White, Jules et al. (2023). *A prompt pattern catalog to enhance prompt engineering with ChatGPT.* Available at: https://arxiv.org/abs/2302.11382. Last accessed at 09/06/2026.

Wilson, Michael (2024). *The Criticality of Measuring Server Utilization in Delivering Sustainable Data Centres*. Available at: https://www.nlyte.com/blog/the-criticality-of-measuring-server-utilization-in-delivering-sustainable-data-centers/. Last accessed at 02/02/2026.

Yañez-Barnuevo, Miquel (2025). *Data Centers and Water Consumption*. Available at: https://www.eesi.org/articles/view/data-centers-and-water-consumption. Last accessed at 02/02/2026.