

Homework #1

This report examines and compares the Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) models provided by three major cloud vendors: Amazon Web Services (AWS), Google Cloud, and IBM Cloud.

1. Amazon Web Services (AWS):

- IaaS (EC2): The process began with setting up an AWS account, followed by locating the EC2 service for instance creation. After filling the parameters configuration, it took just about 3-5 seconds for the web page to show that instance has been initialized, which is quite impressive. Nonetheless, the UI is not very user friendly, especially for novices like me. Number of options on the screen seems to be a little too much.
- PaaS (Lambda): I used this service when building a web app. After setting up the AWS configuration needed in AWS CLI (enter credentials and verify), I wrote a few lines of code and selected the most basic version of this lambda service and created a lambda serverless function to run a chunk of code without the need to worry about the server. Each invocation of the function engaged AWS's backend infrastructure to handle the request and return/give me the respective result.
- SaaS (WorkMail): Using this service includes two options: either creating a new organization or creating a new organization. After joining an organization, this just functions similarly to an email app like outlook or gmail that allows users to send and receive emails and calendar information. Everything took place on a webpage so friendly to the general public.

2. Google Cloud Platform:

- IaaS (Compute Engine): I tried this service during the lecture. The creation took 5-7 seconds and could be done in just the "create a new VM instance" page where all the configuration parameters are available. I tested uploading and downloading using the corresponding built-in "upload" and "download" buttons and tested a few commands like "ls" in the cloud shell terminal.

- PaaS (App Engine): I followed a youtube tutorial to run this service. I first logged into google cloud and created an app engine project. Then, I opened the cloud shell terminal to clone a simple Python application from GitHub and run the project and through the process it automatically downloaded all the dependencies and finally I could just ctrl + click the "localhost:8000", which redirected me to the correct url where the app is running and showing "hello world" on the web browser.
- SaaS (Gmail): Simply logging in with a google account and using it like a regular mail app (not much to say for this service).

3. IBM Cloud:

- IaaS (IBM Cloud Virtual Server for VPC): Generating this instance took about 5-7 seconds to generate the instance. Initially I didn't understand why it didn't let me create an instance, later I found out it requires me to specify an ssh key (either create new or use existing but must have one).
- PaaS (Code Engine): After accessing IBM Cloud and navigating to Code Engine, I initiated a project with the default settings. The only thing that I want to mention is the component type selection—application, job, or function—each with its configuration nuances. While this differs from AWS and Google Cloud's approach, the rest just are the same with the other vendors.
- SaaS (IBM Watson x Assistant): I simply went to this url [\[https://www.ibm.com/products/watsonx-assistant\]](https://www.ibm.com/products/watsonx-assistant) and tried this chatbot. It works similarly with any other conversational AI.

Comparison:

- IaaS Comparison: My preference leans towards Google Cloud for its user-friendly interface. While Amazon AWS may have the best performance in terms of speed and performance as well as best reputation in this realm, its design is a little burdensome and less intuitive for new users like me. IBM Cloud ranks as my second choice; its design is commendable, yet the instance creation process was slightly marred by the SSH key requirement, which wasn't made clear upfront. Overall, in terms of capability, all of the three vendors demonstrated good results.
- PaaS Comparison: When evaluating the speed and capabilities, all of the three vendors execute tasks rapidly— within seconds. Since I'm not doing any serious work at this stage, I look more into the design and details. Both IBM Code Engine and Google App Engine stand out for their intuitive design. Google Cloud's shell

terminal and sidebar with step-by-step tutorials significantly aid in my learning process, and its buttons to upload and download also added to its convenience. For IBM, I liked its design to have three different components and customizable configurations for each. However, for AWS, setting up the CLI is not that straightforward and has a steeper learning curve for me. Therefore, in terms of usability (or from a beginner's perspective on user-friendliness) I rank AWS as the 3rd.

- SaaS Comparison: All of them are great products, but I like Gmail the most not only because I use it on a daily basis but also because the IBM and AWS's offerings are more to-company products that I am less familiar with.

Executive Summary/Takeaways:

Based on my experience, engaging with any of the cloud services provided by IBM, Google, and Amazon is straightforward at the current stage: simply login in and we could create these services and access the related functionalities. I think what's more difficult is to integrate these cloud services with development tools, or to deploy and continuously develop large-scale projects, or training complex ML models on the cloud.

References:

AWS:

[Types of Cloud Computing - SaaS vs PaaS vs IaaS - AWS](#)

▶ Amazon/AWS EC2 (Elastic Compute Cloud) Basics | Create an EC2 Instance Tutori...

Google Cloud:

▶ Google App Engine Project for Beginners [Hands on Lab]

IBM:

▶ Create Virtual Server for VPC in IBM Cloud #1