ASSIGNMENT 2 – Migrating a traditional On-Premise E-Commerce website Shop Cart to AWS(Enhancement) Cloud (Due by: 05-11-2023)

Akilandeshwari Srinivasan (451036)

CLCM3102: Cloud Application Requirements and Specifications

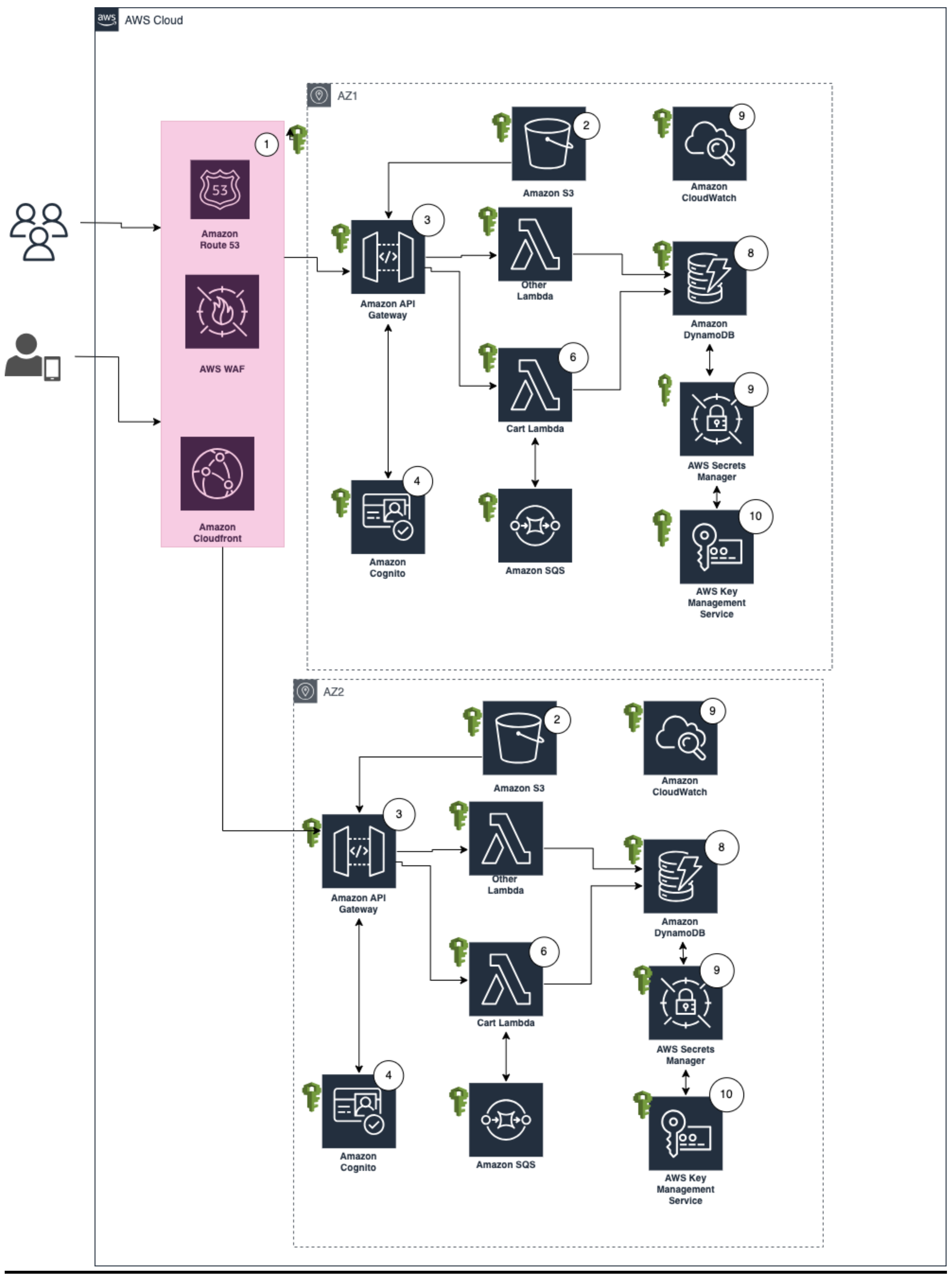
Preparation:

Gone through the AWS documentation and previous assignment to enhance it.

Observations:

- I have learnt many services and its features in deeper while compared to before. In order to complete assignment, I have surfed a lot in the internet which helps me to know new services in real time.
- As this is the enhancement of the previous assignment, I have added the availability zone in order to achieve reliability and AWS KMS for security purpose.

Architectural Diagram:



1. Amazon Route53:

Route 53 lies under the category Networking in aws services. It is highly available and scalable domain name system (DNS). It helps to perform any combination of following functionalities:

- *domain registration* which is the name for your domain like www.akilashop.com, Here "akilashop" is the domain name.
- *DNS Routing*, which is it*,* helps to route the request of user to navigate to the domain name which is mentioned in the request.
- *Health checking* which is sends automated requests over the internet to a resource, such as a web server, to verify that it's reachable, available, and functional.

Justification for why I used this service:
Here I have used the Route53 (for DNS Routing) to route the request to shopCart website whenever user search the domain name.

Amazon WAF (Web Application Firewall):

AWS WAF is a web application firewall that lets you monitor the HTTP(S) requests that are forwarded to your protected web application resources. It let you to access control to the content.

Justification for why I used this service:

Here I have used the WAF to act as a firewall to my website so that no hackers cannot access my website. Moreover, it checks the criteria which I have configured to access the site if the request satisfies then it allows the request else block it.

Amazon CloudFront:
Amazon CloudFront is a content delivery network (CDN) that accelerates delivery of static and dynamic web content to end users. CloudFront delivers content through a worldwide network of data centres called edge locations.

Justification for why I used this service:

Here I have used the CloudFront to act as a cache to my static website content so that the images of the product or frequently used or searched item in the site which is easily to load which leads to less latency and high performance.

2. Amazon S3 (Simple Storage Service):
Amazon S3 comes under AWS storage service. It is an object storage service that helps to store all form of data. The object data stored in s3 container called bucket. It is scalable and has version control and data consistency.

Justification for why I used this service:

Here for my small e-commerce website, it is easy to store all type of data in the bucket. Moreover in the future if the website grows S3 supports scalability. In addition to this, it helps to host shopcart website's static content, such as HTML, CSS, and JavaScript files. It can act as a content delivery network (CDN) to accelerate content delivery, resulting in faster load times for the users.

3. <u>Amazon API Gateway:</u>

AWS API Gateway is like a receptionist for the web services. It receives requests, ensures they're safe, and directs them to the right place. It's a way to manage, secure, and monitor access to your data and applications.

<u>Justification for why I used this service:</u>
As a small e-commerce website, Amazon API Gateway is a crucial tool for me. It helps secure my API endpoints, integrates seamlessly with services like Amazon S3 to fetch product data, ensures scalability during traffic spikes, and provides caching, rate limiting, and monitoring for optimal performance. It's an asset for enhancing the shopping experience for my customers while maintaining control and security.

4. <u>Amazon Cognito:</u>

AWS API Gateway is like a receptionist for the web services. It receives requests, ensures they're safe, and directs them to the right place. It's a way to manage, secure, and monitor access to the data and applications.

<u>Justification for why I used this service:</u>
For a small e-commerce website, Amazon API Gateway is a crucial tool for it. It helps secure to API endpoints, integrates seamlessly with services like Amazon S3 to fetch product data, ensures scalability during traffic, and provides caching, rate limiting, and monitoring for optimal performance.

5. <u>Amazon Lambda:</u>

AWS Lambda is coming under compute service provided by AWS. And it is a serverless computing. It executes the function once we upload the executable file to the server without worrying about the infrastructure and environment maintenance to run the file.

<u>Justification for why I used this service:</u>
For a small e-commerce website, AWS Lambda is a game-changer. I use Lambda to automate tasks like order processing and sending confirmation emails. It scales automatically, so I don't have to worry about server management. This serverless service helps me focus on my business while providing a cost-effective and efficient way to handle various functions in the shopping process.

6. Amazon Lambda (cart lambda):

I have assigned a lambda server for the cart. As the cart has lot of functionality and actions such as add items to the cart, calculate amount according to the cart items, shipping details, etc., so added a lambda server for this section.

7. Amazon SQS (Simple Queue Service):
SQS offers a secure, durable, and available hosted queue that lets you integrate and decouple distributed software systems and components. Amazon SQS offers common constructs such as dead-letter queues and cost allocation tags. It provides a generic web services API that you can access using any programming language that the AWS SDK supports.

   Justification for why I used this service:
   I used SQS to manage the processing of orders and notifications, ensuring no requests are lost and all tasks are executed in a reliable and orderly manner. It provides a buffer between different parts of my system, helping me maintain responsiveness during high-demand periods while decoupling and organizing my services. SQS ensures a smooth shopping experience for my customers and simplifies the orchestration of essential tasks behind the scenes.

8. Amazon DynamoDB:
DynamoDB comes under database service in AWS. It is a fully managed NoSQL database service that provides fast and predictable performance with seamless scalability. With DynamoDB, you can create database tables that can store and retrieve any amount of data and serve any level of request traffic. You can scale up or scale down your tables throughput capacity without downtime or performance degradation.

   Justification for why I used this service:

   AWS DynamoDB is a crucial part of my operations. It's the database I use to store customer information, order history, and product details. DynamoDB's flexibility and scalability allow me to adapt to changing demands and handle large volumes of data while maintaining high performance. It ensures a seamless shopping experience by enabling fast and reliable access to customer information and order tracking, contributing to the success of the website.

9. Amazon CloudWatch:
Cloud watch is helping to monitor the services in the cloud which you have deployed. To put in other ways, its like a sensor which set in cloud if something happens to the infrastructure it alerts you with the mail or message when it is appended with AWS SQS. Without SQS, it normally contains the log which we can go through. It always keeps an eye on the resources in the cloud. So we no need to worry about the cloud.

   Justification for why I used this service:
   I have used the cloud watch to monitor the health and performance of my website and the server in the cloud. For the e-commerce website cloud watch is essential as it alerts me with the notification if something bad happens.

10. Amazon Secrets Manager:

AWS Secrets Manager is an AWS service that securely stores and manages sensitive information such as passwords, API keys, and database credentials. It helps to protect this data, allows for automated rotation of secrets, and integrates seamlessly with other AWS services, enhancing overall security and secret management.

Justification for why I used this service:

I have used AWS Secrets Manager to securely store sensitive information like API keys and database credentials for my shopcart app. AWS KMS ensures data encryption, while DynamoDB handles cart data. This combination simplifies management, enhances security, and streamlines integration as my app scales.

11. Cloud IAM (Identity Access Management):

AWS Key Management Service (KMS) is a fully managed service that enables you to create and control encryption keys used to protect your data. It integrates with various AWS services and offers hardware security modules (HSMs) to secure your keys and encrypt sensitive information stored in AWS resources.

Justification for why I used this service:

I have used AWS Key Management Service (KMS) in my AWS architecture to ensure the encryption and security of sensitive data. KMS plays a crucial role in protecting secrets stored in AWS Secrets Manager and securing data in DynamoDB. It simplifies key management, enhances data protection, and helps me meet compliance requirements while integrating seamlessly with other AWS services.

12. Cloud IAM (Identity Access Management):

It comes under the security and privacy services of AWS. It is a web service that helps you securely control access to AWS resources. Only the authorized person can access the resources in the cloud. We can grant other people permission to administer and use resources in your AWS account without having to share your password or access key.

Justification for why I used this service:

I have used the IAM for the security purpose. I used this to secure all the service in the cloud so that only the authorized person can access and do the modification if needed.

References:

AWS Official Documentation, www.google.com