

Charles Malon Silva Rocha - 2021376

**Cloud Services**

**Designing and Implementing a Proof-of-Concept Cloud Solution Based on a Client’s Needs**

**CA2**

**Dublin**

**2024**

**CCT College Dublin**

**Assessment Cover Page**

*To be provided separately as a word doc for students to include with every submission.*

|  |  |
| --- | --- |
| **Module Title:** | Cloud Services |
| **Assessment Title:** | Designing and Implementing a Proof-of-Concept Cloud Solution  Based on a Client’s Needs |
| **Lecturer Name:** | Michael Weiss |
| **Student Full Name:** | Charles Malon Silva Rocha |
| **Student Number:** | 2021376 |
| **Assessment Due Date:** | 22nd November2024 |
| **Date of Submission:** | 08th December 2024 |

Below you can access the progress of this assignment:

**<https://github.com/CharlesMalonRocha/Could-Services-CA2>**

**Declaration**

|  |
| --- |
| By submitting this assessment, I confirm that I have read the CCT policy on Academic Misconduct and understand the implications of submitting work that is not my own or does not appropriately reference material taken from a third party or other source. I declare it to be my own work and that all material from third parties has been appropriately referenced. I further confirm that this work has not previously been submitted for assessment by myself or someone else in CCT College Dublin or any other higher education institution. |

Contents

[Introduction 4](#_Toc9358)

[TASK 1: Translating Business Requirements into an AWS Cloud Solution 5](#_Toc8356)

[TASK 2: Designing the Cloud Architecture for High Availability 6](#_Toc27587)

[TASK 3a: Identifying and Resolving Existing Anti-Patterns 7](#_Toc4147)

[TASK 3b: Enhancing Cost Efficiency and Resilience Using AWS Best Practices 8](#_Toc1445)

[TASK 4a: Understanding VPC and Its Components 9](#_Toc24698)

[TASK 4b: Hosting the Medi-Advice Website in a Custom VPC 10](#_Toc13358)

[TASK 5a: Accelerating Content Delivery with AWS CloudFront 11](#_Toc2922)

[Challenge Task 1: Demonstrating Content Delivery with Edge Caching 12](#_Toc20961)

[Challenge Task 2: Implementing a Highly Available Auto-Scaling Web Solution 13](#_Toc4203)

[Challenge Task 3: Automating Infrastructure Deployment with CloudFormation 14](#_Toc31662)

[Conclusion 15](#_Toc28465)

[References 16](#_Toc14070)

Introduction

Cloud computing has changed how businesses manage their IT systems, making it easier, cheaper, and more reliable to handle growing demands. This report focuses on creating a cloud solution for Medi-Advice, a medical startup in Dublin. Medi-Advice helps patients and doctors connect through online consultations, prescription transfers, and payment services for users in Ireland and North America. Right now, their on-premises setup struggles with issues like slow performance, limited scalability, and downtime, making it hard to keep up with growing user needs.

To fix these problems, this report suggests moving to Amazon Web Services (AWS). The solution will use services like EC2, S3, CloudFront, and VPC to create a system that is fast, scalable, and cost-effective. This report explains how these AWS tools can solve the company’s issues, improve performance, and support global users better.

By addressing Medi-Advice’s current challenges, this project will show how AWS can help build a stronger, more efficient system for the company’s needs. It also includes steps for implementation, diagrams, and tips on saving costs while improving performance.

TASK 1: Translating Business Requirements into an AWS Cloud Solution

The proposed technical solution was designed to address Medi-Advice’s current infrastructure challenges while meeting their specific needs for scalability, global accessibility, reliability, and cost efficiency. As a growing medical startup, Medi-Advice faces issues such as slow performance during high traffic, regional latency affecting U.S. users, and limited resilience in case of server failures. These problems can lead to poor user experiences and disruptions in critical medical services.

By leveraging AWS services, the solution provides a flexible and modern infrastructure capable of handling these demands. Services like Amazon EC2, Auto Scaling, and Elastic Load Balancer ensure that the system can scale dynamically with user demand, maintaining high availability. AWS CloudFront addresses global latency by caching content closer to users, while Amazon RDS and VPC enhance reliability, security, and disaster recovery capabilities. Importantly, this solution incorporates AWS’s cost-effective tools, allowing Medi-Advice to optimize expenses without compromising performance (Amazon.com, 2024, Padhy and Patra, 2013, AWS, 2019).

This solution was chosen because it not only solves Medi-Advice’s current issues but also prepares the company for future growth, ensuring a stable, secure, and scalable platform for their services.

#### ****Key AWS Services and Their Roles****

#### ****Amazon EC2 (Elastic Compute Cloud):****

* **Purpose:** Host scalable web, app, and database servers.
* **Benefits**: Elasticity for demand changes, ability to automate scaling using Autoscaling Groups.

#### ****Amazon S3 (Simple Storage Service):****

* **Purpose:** Store brochures and other static files.
* **Benefits**: Durable storage, cost-effective for large files, and integration with CloudFront for global delivery.

#### ****AWS CloudFront (Content Delivery Network):****

* **Purpose:** Improve performance for global users by caching content at edge locations.
* **Benefits**: Reduced latency, improved access speeds for brochures.

#### ****AWS VPC (Virtual Private Cloud):****

* **Purpose:** Securely isolate Medi-Advice’s network with public and private subnets.
* **Components:** Internet Gateway, NAT Gateway, Security Groups, and NACLs.
* **Benefits**: Enhanced security and network management.

#### ****Amazon RDS (Relational Database Service):****

* **Purpose:** Host the SQL Server database.
* **Benefits**: Automatic backups, cross-region replication for disaster recovery.

#### ****Elastic Load Balancer (ELB)****:

* **Purpose:** Distribute traffic across servers in multiple Availability Zones.
* **Benefits**: Enhanced fault tolerance, improved performance.

#### ****AWS Auto Scaling****:

* **Purpose:** Automatically scale EC2 instances based on traffic.
* **Benefits**: Handles spontaneous changes in demand.

#### ****AWS IAM (Identity and Access Management)****:

* **Purpose:** Manage user permissions.
* **Benefits**: Ensures secure access to AWS resources.

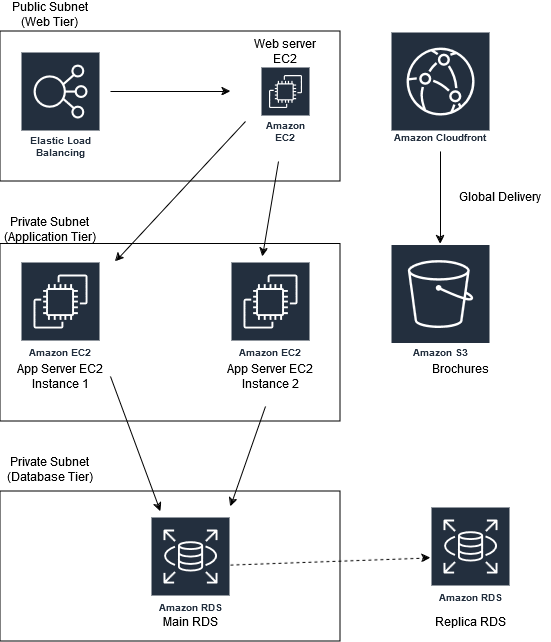
#### ****How Services Address Challenges****

* **Scalability**: EC2 with Auto Scaling dynamically adjusts resources.
* **Global Accessibility**: CloudFront reduces latency.
* **Disaster Recovery**: RDS with cross-region replication ensures uptime.
* **Cost Optimization**: Use reserved/free-tier resources for low-demand periods (Amazon.com, 2024, AWS, 2019).

TASK 2: Designing the Cloud Architecture for High Availability

The proposed cloud architecture design ensures that Medi-Advice’s infrastructure is highly available, scalable, and resilient. Their current on-premises setup struggles with single points of failure, manual resource adjustments, and poor user experiences due to server downtimes and traffic surges.

The new architecture addresses these challenges by leveraging AWS services such as Auto Scaling, Elastic Load Balancers, and multiple Availability Zones. These components enable the infrastructure to dynamically respond to traffic spikes, balance load effectively, and remain operational even in the event of regional failures. The design also ensures the infrastructure aligns with AWS best practices, offering a robust foundation for Medi-Advice’s continued growth and improved user experience (Amazon.com, 2024, AWS, 2019).

****

(Draw.io, 2024)

### ****What Happens After the Implementation of This Architecture****

Once the proposed architecture is implemented, Medi-Advice will experience a significant improvement in the availability, scalability, and resilience of their infrastructure. Each tier of the system has been carefully designed to address their specific requirements for reliability and performance:

1. **Web Tier (Public Subnets with Load Balancer):**

* The **Elastic Load Balancer (ELB)** distributes incoming traffic across multiple EC2 instances, ensuring that if one instance fails, traffic is automatically routed to healthy instances.
* This redundancy minimizes downtime and ensures uninterrupted access for users, aligning with the five nines availability requirement.

1. **Application Tier (Private Subnets with EC2 Instances):**

* The application layer, hosted on EC2 instances within private subnets, benefits from Auto Scaling. This ensures that additional instances are automatically provisioned during traffic spikes, while unused instances are terminated during low-demand periods.
* This dynamic scaling prevents service disruptions caused by overloading and contributes to consistent uptime.

1. **Database Tier (Private Subnet with Amazon RDS):**

* T**he RDS instance is configured for multi-AZ deployment, providing automatic failover to a standby database in case the primary instance becomes unavailable.**
* **Cross-region replication further ensures that if one region experiences a failure, a backup database in another region can take over, preserving data integrity and availability.**

1. **Global Delivery (S3 with CloudFront):**

* Static assets like brochures are stored in Amazon S3 and delivered through CloudFront, which caches content at edge locations. This reduces latency for global users and prevents resource overload on backend servers.
* Content delivery remains uninterrupted even during regional outages, as CloudFront automatically serves cached data (Amazon.com, 2024, AWS, 2019).

### ****How This Architecture Supports the Five Nines Availability****

The design incorporates multiple layers of redundancy and failover mechanisms, which are essential to achieving 99.999% uptime:

* **Redundancy at Every Tier**: From load-balanced web servers to multi-AZ databases, the architecture ensures no single point of failure.
* **Failover and Disaster Recovery**: Multi-AZ RDS and cross-region replication provide robust failover options, ensuring continuous operation during outages.
* **Scalability and Elasticity**: Auto Scaling adapts to traffic changes, preventing performance degradation or downtime during high demand.
* **Global Accessibility**: CloudFront ensures that static content is always available, even during infrastructure disruptions.

By implementing this architecture, Medi-Advice will meet the five nines availability goal, reducing downtime to under 5.26 minutes per year. This reliability is critical for the medical services they provide, ensuring that patients and doctors can access the platform without interruptions, regardless of location or traffic conditions (Amazon.com, 2024, AWS, 2019).

TASK 3a: Identifying and Resolving Existing Anti-Patterns

Medi-Advice’s current infrastructure relies on outdated practices, such as manual scaling and single-region deployment, which limit its ability to adapt to growing demands and ensure reliability. These anti-patterns lead to inefficiencies, higher operational costs, and reduced user satisfaction during peak traffic or regional failures.

By identifying these inefficiencies, the proposed solutions introduce automated scaling and multi-region deployment. AWS services such as Auto Scaling and cross-region disaster recovery mechanisms ensure the infrastructure is efficient, reliable, and capable of handling Medi-Advice’s user demands while eliminating operational bottlenecks (Amazon.com, 2024, AWS, 2019).

#### ****Anti-Pattern 1: Manual Scaling****

* **Problem**: Current setup requires manual intervention to scale servers during high demand.
* **Solution**: Use AWS Auto Scaling for EC2 instances to dynamically adjust capacity.

#### ****Anti-Pattern 2: Single Region Deployment****

* **Problem**: Hosting only in Dublin increases latency for U.S. users and risks single points of failure.
* **Solution**: Deploy cross-region with failover using RDS and multi-region Load Balancers.

TASK 3b: Enhancing Cost Efficiency and Resilience Using AWS Best Practices

Cost efficiency and resilience are critical for Medi-Advice’s success as a startup. Their current infrastructure lacks optimization, resulting in unnecessary expenses and vulnerabilities. This task leverages the AWS Well-Architected Framework’s six pillars—Security, Performance, Reliability, Operational Excellence, Cost Optimization, and Sustainability—to enhance the overall efficiency and stability of the system.

By incorporating these best practices, the solution ensures Medi-Advice can effectively manage costs, maintain high availability, and secure their data while preparing for future scalability and operational challenges (Amazon.com, 2024, AWS, 2019).

1. **Operational Excellence**:

* Automate deployments using CloudFormation or AWS CLI.

1. **Security:**

* Implement IAM roles, encrypt data in S3 and RDS.

1. **Reliability:**

* Use multi-AZ deployment for RDS and ELB.

1. **Performance Efficiency:**

* Enable Elastic Load Balancing for traffic distribution.

1. **Cost Optimization:**

* Use S3 lifecycle policies to archive old files.

1. **Sustainability:**

* Scale down unused resources during off-peak hours.

TASK 4a: Understanding VPC and Its Components

To secure and isolate Medi-Advice’s infrastructure, the proposed solution introduces a Virtual Private Cloud (VPC). A VPC allows the creation of a private, logically isolated network within AWS, ensuring that sensitive resources like databases are protected from unauthorized access.

This task focuses on explaining the components of a VPC, such as subnets, Internet Gateways, NAT Gateways, and security rules, which together form the backbone of a secure and scalable cloud environment. Implementing these components allows Medi-Advice to maintain strict control over their network and enhance overall system security (Rajasoundaran et al., 2021).

#### ****VPC Components****:

1. **Public and Private Subnets**:

* **Public subnet:** Hosts resources like Load Balancers accessible from the internet.
* **Private subnet:** Secures resources like databases and app servers.

1. **CIDR Blocks:**

* Define IP address ranges for subnets.

1. **Internet Gateway:**

* Connects the VPC to the internet.

1. **NAT Gateway:**

* Enables instances in private subnets to access the internet securely.

1. **Security Groups:**

* Firewall rules at the instance level.

1. **NACLs:**

* Firewall rules at the subnet level.

1. **VPC Peering:**

* Connects multiple VPCs securely.

TASK 4b: Hosting the Medi-Advice Website in a Custom VPC

Medi-Advice’s current setup does not leverage the security and flexibility of a custom network. This task involves creating a custom VPC with public and private subnets to securely host the Medi-Advice website. Public subnets allow internet-facing components, while private subnets protect sensitive resources like databases.

Hosting the website on an EC2 instance within this VPC demonstrates how secure networking combined with AWS services can improve website availability, performance, and user experience (Rajasoundaran et al., 2021, Amazon.com, 2024, AWS, 2019).

Practical

TASK 5a: Accelerating Content Delivery Network with AWS CloudFront

CloudFront reduces latency by caching Medi-Advice’s content at global edge locations. This task explains how CloudFront improves performance for users accessing brochures and other resources.

#### ****What is Edge Computing and a CDN?****

Edge computing involves storing and processing data closer to the end users at "edge" locations rather than relying on a central server. A **Content Delivery Network (CDN)** like AWS CloudFront uses this principle by caching resources such as files, images, or brochures at servers distributed globally. This reduces the distance between users and the server delivering the content, significantly improving speed and availability (Helle Sjøvaag, Ragnhild Kr. Olsen and Ferrer-Conill, 2024).

#### ****How AWS CloudFront Speeds Up Content Delivery****

AWS CloudFront caches content at over 450 edge locations worldwide. Here’s how it works for Medi-Advice:

1. **Caching at Edge Locations**: When a user requests a resource, CloudFront serves it from the nearest edge location instead of the central server. For instance, a customer in New York will receive Medi-Advice's brochure from a nearby U.S. edge location.
2. **Reduced Latency**: By minimizing the distance between users and the content, load times are significantly reduced.
3. **Load Reduction on Origin Servers**: The origin server (e.g., S3 bucket) only handles uncached requests, reducing its workload.

#### ****Example:****

* Medi-Advice uploads a brochure to an S3 bucket in Dublin. Without CloudFront, customers in the U.S. face delays accessing the content due to physical distance and network hops.
* With CloudFront, the brochure is cached at multiple U.S. edge locations. U.S. users now access the cached version, enjoying faster load times.

#### ****Benefits for Medi-Advice:****

1. **Improved User Experience**: Faster access ensures users worldwide can quickly view brochures, enhancing satisfaction.
2. **Global Reach**: CloudFront ensures the same performance for users regardless of their location.
3. **Cost Savings**: CloudFront reduces data transfer costs from the origin server by serving cached content.
4. **Resilience**: Cached content remains accessible even if the origin server is temporarily unavailable.

By implementing CloudFront, Medi-Advice can deliver its sales brochures faster and more reliably to its global customers, aligning with their business growth and user experience goals (Amazon.com, 2024, AWS, 2019).

Challenge Task 1: Demonstrating Content Delivery with Edge Caching

Practical

To showcase the benefits of edge caching, this task involves creating a sample sales brochure, uploading it to S3, and configuring CloudFront for global distribution. Users will access cached content through edge locations, reducing latency and improving load times.

This demonstration highlights how Medi-Advice can leverage CloudFront to enhance user satisfaction by ensuring fast and reliable access to their materials, no matter where users are located.

Challenge Task 2: Implementing a Highly Available Auto-Scaling Web Solution

Practical

Medi-Advice needs a web solution that can handle fluctuating traffic while maintaining high availability. This task involves setting up an Auto Scaling Group with an Elastic Load Balancer within a custom VPC spanning multiple Availability Zones. The system automatically adjusts the number of servers based on traffic demand, ensuring cost efficiency and performance.

This solution demonstrates how integrating Auto Scaling with load balancing can eliminate downtime, support peak traffic periods, and ensure a seamless experience for users.

Challenge Task 3: Automating Infrastructure Deployment with CloudFormation

Practical

Manual infrastructure deployment is time-consuming and prone to errors. This task introduces AWS CloudFormation, which uses Infrastructure-as-Code to automate the deployment of Medi-Advice’s resources. By creating a custom VPC and launching an EC2 instance using a CloudFormation template, this task shows how automation improves efficiency.

CloudFormation simplifies future updates and ensures consistency across deployments, making it an essential tool for Medi-Advice as they scale their operations.

Conclusion

‌

‌

References

docs.aws.amazon.com. (n.d.). *What is Amazon S3? - Amazon Simple Storage Service*. [online] Available at: <https://docs.aws.amazon.com/AmazonS3/latest/userguide/Welcome.html#S3Features.>

(docs.aws.amazon.com, n.d.)

‌

Amazon Web Services (2019). What Is Amazon EC2? - Amazon Elastic Compute Cloud. [online] Amazon.com. Available at: <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/concepts.html.>

(Amazon Web Services, n.d.)

AWS (2019). *What Is an Application Load Balancer? - Elastic Load Balancing*. [online] Amazon.com. Available at: <https://docs.aws.amazon.com/elasticloadbalancing/latest/application/introduction.html.>

(AWS, 2019)

‌

AWS (2019). *What Is Amazon EC2 Auto Scaling? - Amazon EC2 Auto Scaling*. [online] Amazon.com. Available at: <https://docs.aws.amazon.com/autoscaling/ec2/userguide/what-is-amazon-ec2-auto-scaling.html.>

(AWS, 2019)

Draw.io (2024). Flowchart Maker & Online Diagram Software. [online] app.diagrams.net. Available at: <https://app.diagrams.net.>

Helle Sjøvaag, Ragnhild Kr. Olsen and Ferrer-Conill, R. (2024). Delivering content: Modular broadcasting technology and the role of content delivery networks. Telecommunications Policy, pp.102738–102738. doi:https://doi.org/10.1016/j.telpol.2024.102738. (Helle Sjøvaag, Ragnhild Kr. Olsen and Ferrer-Conill, 2024)

‌

Rimal, B.P., Jukan, A., Katsaros, D. and Goeleven, Y. (2010). Architectural Requirements for Cloud Computing Systems: An Enterprise Cloud Approach. Journal of Grid Computing, 9(1), pp.3–26. doi:https://doi.org/10.1007/s10723-010-9171-y. (Rimal et al., 2010)

Padhy, R.P. and Patra, M.R. (2013). Architecture & Design of Affordable and Highly Available Enterprise Cloud Service. International Journal of Cloud Computing and Services Science (IJ-CLOSER), 2(2). doi:https://doi.org/10.11591/closer.v2i2.1527. (Padhy and Patra, 2013)

Rajasoundaran, S., Prabu, A.V., Routray, S., Kumar, S.V.N.S., Malla, P.P., Maloji, S., Mukherjee, A. and Ghosh, U. (2021). Machine learning based deep job exploration and secure transactions in virtual private cloud systems. Computers & Security, 109, p.102379. doi:https://doi.org/10.1016/j.cose.2021.102379. (Rajasoundaran et al., 2021)

‌‌

‌‌

‌‌

‌‌‌‌

‌