



# **University of East London**

**SCHOOL OF ARCHITECTURE, COMPUTING &  
ENGINEERING**

## **INDIVIDUAL CW REPORT**

Student Name: Rizwan Ullah Ahmed

Student Number: -----

Module Leader: GAURAV MALIK

Module Code: CN7026

## Table of Contents

<i>Title: Hospital.....</i>	<b>4</b>
<i>Introduction.....</i>	<b>4</b>
<i>Project Plan.....</i>	<b>4</b>
<i>Project Overview.....</i>	<b>4</b>
<i>Project Goals.....</i>	<b>4</b>
<i>Project Timeline.....</i>	<b>5</b>
<i>Project Team.....</i>	<b>5</b>
<i>Requirement Gathering.....</i>	<b>6</b>
<i>Functional Requirements.....</i>	<b>6</b>
<i>Non-Functional Requirements.....</i>	<b>6</b>
<i>Choice of Data Centre and Standards.....</i>	<b>6</b>
<i>Location:.....</i>	<b>6</b>
<i>Managed Data Center.....</i>	<b>6</b>
<i>Advantages of Managed Data Center.....</i>	<b>7</b>
<i>Standards.....</i>	<b>7</b>
<i>Cloud Architecture.....</i>	<b>8</b>
<i>Choice of type of cloud platform.....</i>	<b>8</b>
<i>Public Cloud.....</i>	<b>8</b>
<i>Advantages of Public Cloud.....</i>	<b>8</b>
<i>Cloud System Architecture Diagram.....</i>	<b>9</b>
<i>Brief explanation on services I selected for my architecture.....</i>	<b>9</b>
<i>Implementation using the AWS cloud platform.....</i>	<b>11</b>
1. <i>IAM Implementation.....</i>	<b>11</b>
2. <i>VPC, Availability Zone, Subnet, Nat Gateway, Route, and Internet Gateway Implementation....</i>	<b>12</b>
3. <i>Security Group Implementation.....</i>	<b>13</b>
4. <i>EC2 Server Implementation.....</i>	<b>14</b>
5. <i>EC2 Backup Server Creation.....</i>	<b>16</b>
6. <i>RDS Implementation.....</i>	<b>17</b>
7. <i>Two S3 Bucket with ACLs Implementation.....</i>	<b>20</b>
8. <i>Cloud Watch Usage Alarm and SNS Notification Implementation.....</i>	<b>21</b>
<i>Costing.....</i>	<b>22</b>
<i>These services chosen for the system.....</i>	<b>22</b>
<i>Analysis and Reflection.....</i>	<b>23</b>
<i>System maintenance.....</i>	<b>23</b>

<b>Manage its evolution.....</b>	<b>23</b>
<b>Future upgradation.....</b>	<b>23</b>
<b>New rule and regulations.....</b>	<b>23</b>
<b><i>References.....</i></b>	<b>24</b>



# **Title: Hospital**

## **Introduction**

In today's healthcare environment, hospitals are under increasing pressure to improve efficiency, reduce costs, and improve patient care. Cloud computing can help hospitals achieve these goals by providing a scalable, reliable, and secure platform for storing and managing patient data. Cloud-based systems can help hospitals automate many of their administrative tasks, such as scheduling appointments, and staff and patients' data. This can free up staff time to focus on patient care. Cloud-based systems can help hospitals reduce their IT costs by eliminating the need to purchase and maintain their hardware and software. Cloud-based systems can provide patients with access to their medical records from anywhere, which can help improve communication between patients and providers. Cloud-based systems can also be used to provide remote monitoring and care, which can help improve patient outcomes.

## **Project Plan**

### **Project Overview**

The purpose of this project is to implement a cloud-based hospital system in 4 months. The system will be used to store and manage patient records, generate reports on patient care, and integrate with other hospital systems. The system will be compliant with HIPAA regulations and will be easy to use for both patients and staff.

### **Project Goals**

1. Implement a cloud-based hospital system that is secure, compliant, and easy to use.
2. Migrate patient data to the cloud securely and efficiently.
3. Train hospital staff on how to use the cloud-based system.
4. Ensure that the cloud-based system meets the needs of the hospital.



## Project Timeline

	L	AUG	SEP	OCT
HS-1 Planning and requirements gathering	[ ]			
HS-2 System design		[ ]		
HS-3 System development		[ ]		
HS-4 System testing			[ ]	
HS-5 Deployment on AWS cloud			[ ]	
HS-6 Create IAM users	[ ]			
HS-7 Create VPC and Availability Zones	[ ]			
HS-8 Create Subnets and set NAT Gai...	[ ]			
HS-9 Create Security Groups	[ ]			
HS-10 Create two EC2 Servers	[ ]			
HS-11 Create Two RDS Databases	[ ]			
HS-12 Create two S3 Buckets	[ ]			
HS-13 Create API Gateway	[ ]			
HS-14 Create Elastic Load Balance	[ ]			
HS-15 Create Cloud Front	[ ]			
HS-16 Create Cloud Watch	[ ]			
HS-17 Create SNS Service	[ ]			
HS-18 Post-deployment cloud support			[ ]	
HS-19 Project closure				[ ]

## Project Team

- Project Manager
- System Architect
- Software Developers
- Quality Assurance Engineers
- Technical Support Engineers

# **Requirement Gathering**

## **Functional Requirements**

- The system must be easy to use for staff and patients.
- The system must be able to handle a large number of patients.
- The system must be able to store track records of patients.
- The system must be able to generate reports on patient care.
- The system must be able to keep track of staff attendance and payroll.
- The system must be connected to NHS and other hospital systems.
- The system must be handling appointment system.
- The system must be certified by HIPAA regulations.
- The system must be well-connected to health insurance companies.
- The system must have a billing system as well.

## **Non-Functional Requirements**

- The system must be easy to update and maintain.
- The system must be available 24/7.
- The system must be scalable to handle a large number of users.
- The system must be secure and protected from unauthorized access.
- The system must be reliable and should have a low latency rate.
- The system must be available around the globe.

# **Choice of Data Centre and Standards**

**Location:** The location I selected for my data center is US East (N. Virginia us-east-1)

## **Managed Data Center**

A managed data center involves outsourcing the data center infrastructure and management to a third-party provider. This option can be beneficial for hospitals that want to focus on their core healthcare services



without dealing with the complexities of maintaining a data center. Managed data centers offer services such as hardware provisioning, maintenance, security, and network management. They provide scalability and expertise in data center operations. This option is suitable for hospitals with limited IT resources or expertise.

## Advantages of Managed Data Center

- Expertise and Support
- Scalability and Flexibility
- Cost Savings
- Enhanced Security and Compliance
- High Reliability and Uptime
- Focus on Core Business

## Standards

- **HIPAA Compliance:** The Health Insurance Portability and Accountability Act (HIPAA) sets standards for protecting patient health information. It is crucial to ensure that the managed data center is HIPAA compliant, even if the hospital is using a public cloud architecture. This certification demonstrates the data center's commitment to safeguarding sensitive healthcare data.
- **SOC 2 Type II:** Service Organization Control 2 (SOC 2) is a widely recognized certification that validates a data center's controls and security measures. It focuses on security, availability, processing integrity, confidentiality, and privacy. SOC 2 Type II certification assures that the managed data center meets stringent security and operational standards.
- **ISO 27001:** This certificate demonstrates that the managed data center has implemented robust security practices and controls.
- **Cloud Security Alliance (CSA) STAR Certification:** CSA STAR Certification provides an independent assessment of a managed data center's security posture and adherence to best practices. It evaluates the data center's security controls, data privacy, and transparency.
- **Data Center Tier Certification:** Although not specific to the healthcare industry, considering a managed data center with Uptime Institute Tier Certification can ensure the facility meets stringent reliability and

availability standards. Tier certifications range from Tier I to Tier IV, with Tier IV being the highest level of fault tolerance and uptime.



# Cloud Architecture

## Choice of type of cloud platform

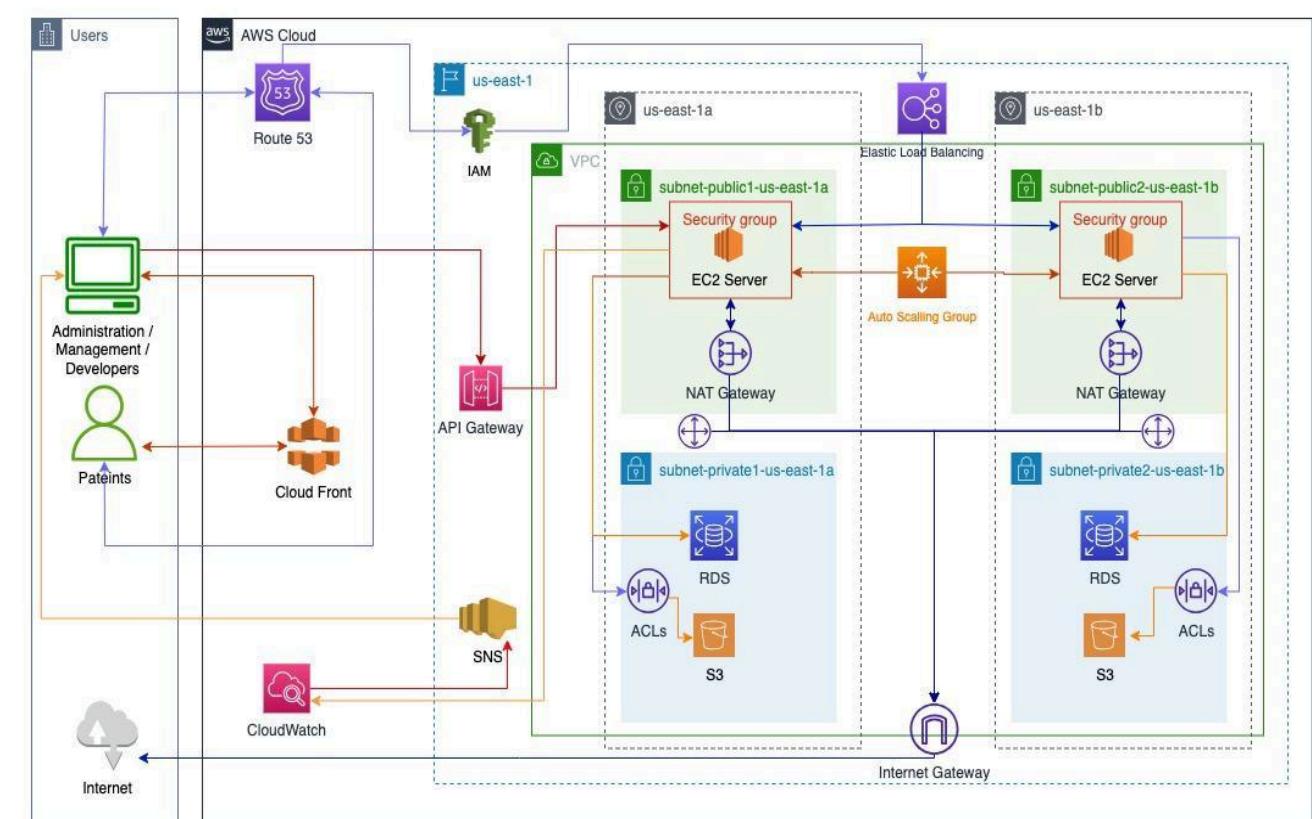
**Public Cloud:** A public cloud architecture involves utilizing cloud services and infrastructure provided by third-party vendors, such as Amazon Web Services (AWS), Microsoft Azure, or Google Cloud Platform. In this model, hospitals can leverage the vendor's resources, including virtual servers, storage, databases, and various other services, on a pay-as-you-go basis. Public clouds offer scalability, flexibility, and a wide range of services, allowing hospitals to quickly provision resources and adapt to changing needs. It can be suitable for non-sensitive data and applications that don't have strict compliance requirements.

### Advantages of Public Cloud

- Scalable and flexible
- Cost efficient
- Accessible and global reach
- Reliable and highly available
- Secure and Compliance
- Disaster recovery and business continuity
- Technical support and maintenance available
- Innovation

# Cloud System Architecture Diagram

## Hospital Cloud Architecture



## Brief explanation on services I selected for my architecture

**IAM:** AWS Identity and Access Management (IAM) is a web service that helps you securely control access to AWS resources. With IAM, we can centrally manage permissions that control which AWS resources users can access.

**VPC:** With Amazon Virtual Private Cloud (Amazon VPC), we can launch AWS resources in a logically isolated virtual network that we've defined. This virtual network closely resembles a traditional network that you'd operate in your own data center, with the benefits of using the scalable infrastructure of AWS.

**Regions and Availability Zones:** AWS Regions are large and widely dispersed into separate geographic locations. Availability Zones are distinct locations within an AWS Region that are engineered to be isolated from failures in other Availability Zones.

**Subnet:** A subnet is a range of IP addresses in VPC

**NAT gateway:** A NAT gateway is a Network Address Translation (NAT) service. You can use a NAT gateway so that instances in a private subnet can connect to services outside VPC but external services cannot initiate a connection with those instances.

**Internet Gateway:** It is a highly available VPC component that allows communication between VPC and the internet.

**Security group:** It controls the traffic that is allowed to reach and leave the resources that it is associated with. For example, after we associate a security group with an EC2 instance, it controls the inbound and outbound traffic for the instance.

**Amazon EC2:** It provides on-demand, scalable computing capacity in the AWS Cloud. Using Amazon EC2 reduces hardware costs so we can develop and deploy applications faster. We can use Amazon EC2 to launch as many or as few virtual servers as you need, configure security and networking, and manage storage.

**Amazon RDS:** It is a web service that makes it easier to set up, operate, and scale a relational database in the AWS Cloud. It provides cost-efficient, resizable capacity for an industry-standard relational database and manages common database administration tasks.

**Amazon S3:** It is an object storage service that offers industry-leading scalability, data availability, security, and performance.

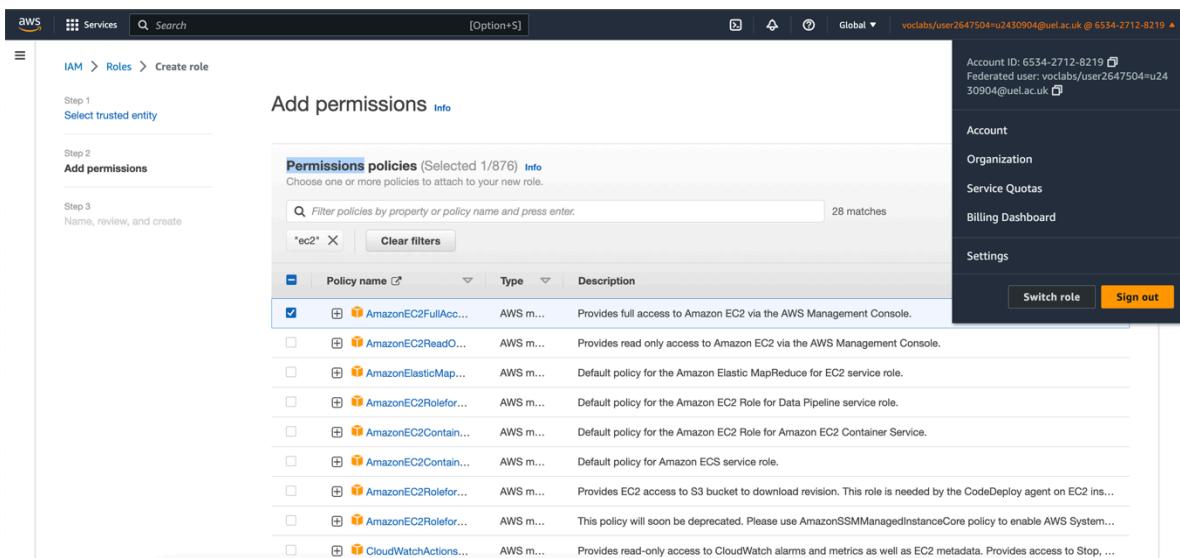
**Amazon CloudWatch:** It monitors your Amazon Web Services (AWS) resources and the applications you run on AWS in real-time. You can use CloudWatch to collect and track metrics, which are variables you can measure for your resources and applications.

**Amazon SNS:** It is a managed service that provides message delivery from publishers to subscribers. Publishers communicate asynchronously with subscribers by sending messages to a topic, which is a logical access point and communication channel.

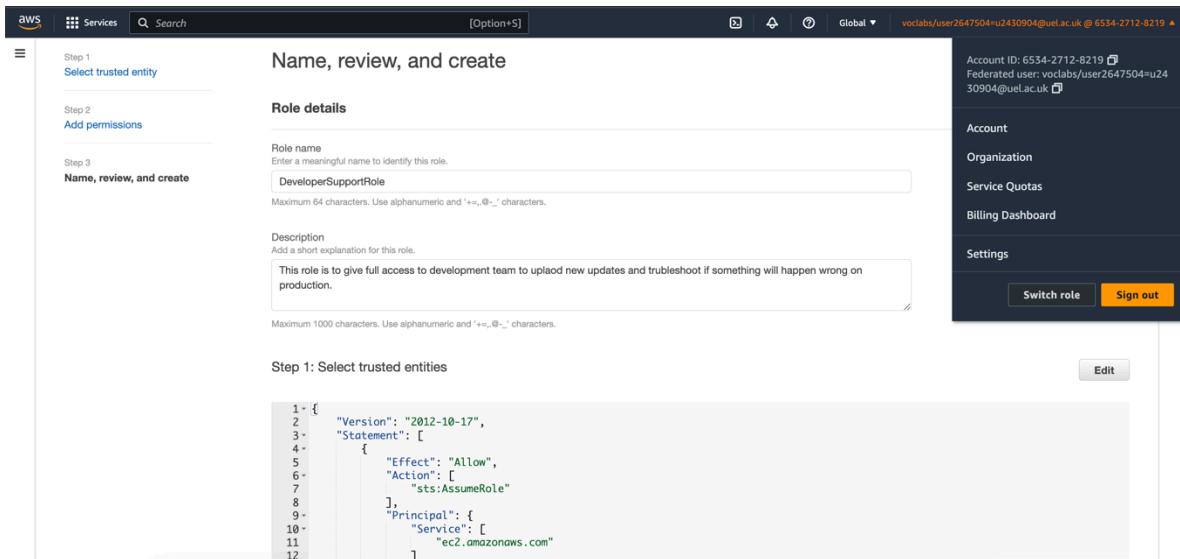
# Implementation using the AWS cloud platform

## 1. IAM Implementation

**Details:** Here I tried to create an IAM for the development and management team with full access to all services in my architecture like EC2, RDS, S3, etc. but couldn't succeed because the sandbox account did not allow me. For that, I have selected default policies, but we can also create custom policies as well in which we can give permissions of our own choice.



The screenshot shows the 'Add permissions' step of creating a new IAM role. In the search bar, 'ec2\*' is typed. The 'AmazonEC2FullAccess' policy is selected and highlighted with a blue checkmark. Other policies listed include 'AmazonEC2ReadOnlyAccess', 'AmazonElasticMapReduceRole', 'AmazonEC2RoleforDataPipeline', 'AmazonEC2ContainerServiceRole', 'AmazonEC2ContainerServiceRole', 'AmazonEC2RoleforCodeDeploy', 'AmazonEC2RoleforCloudWatchLogs', and 'CloudWatchActions'. The sidebar on the left shows steps: Step 1 (Select trusted entity), Step 2 (Add permissions - currently active), and Step 3 (Name, review, and create). The top right corner displays account information: Account ID: 6534-2712-8219, Federated user: vclabs/user2647504=u24309094@uel.ac.uk.



The screenshot shows the 'Name, review, and create' step of creating a new IAM role. The role name is set to 'DeveloperSupportRole'. The description field contains the text: 'This role is to give full access to development team to upload new updates and troubleshoot if something will happen wrong on production.' The sidebar on the left shows steps: Step 1 (Select trusted entity), Step 2 (Add permissions), and Step 3 (Name, review, and create - currently active). The bottom of the screen shows the JSON code for the role's trust policy:

```
1- {
2-   "Version": "2012-10-17",
3-   "Statement": [
4-     {
5-       "Effect": "Allow",
6-       "Action": [
7-         "sts:AssumeRole"
8-       ],
9-       "Principal": [
10-         "Service": [
11-           "ec2.amazonaws.com"
12-         ]
13-       ]
14-     }
15-   ]
16- }
```

## 2. VPC, Availability Zone, Subnet, Nat Gateway, Route, and Internet Gateway Implementation

**Details:** Here I created a VPC in the us-east-1 region with two availability zones us-east-1a and us-east-1b, two public and two private subnets, NAT Gateway 1 per AZ, and Internet Gateway. I created this architecture for the reliability, availability, and security of my system.

The screenshot shows the AWS VPC creation wizard. On the left, there are several configuration sections:

- Number of Availability Zones (AZs)**: Set to 2.
- Number of public subnets**: Set to 2.
- Number of private subnets**: Set to 4.
- NAT gateways (\$)**: Set to 1 per AZ.
- VPC endpoints**: Set to S3 Gateway.

On the right, the **Preview** section shows the resulting VPC structure:

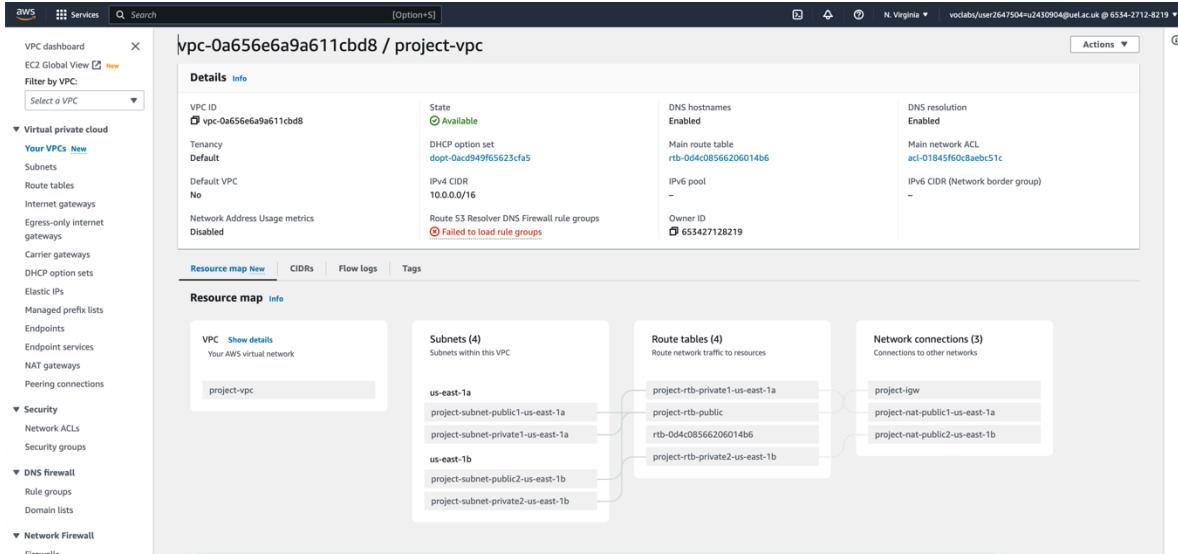
- VPC**: project-vpc
- Subnets (4)**:
  - us-east-1a
    - project-subnet-public1-us-east-1a
    - project-subnet-private1-us-east-1a
  - us-east-1b
    - project-subnet-public2-us-east-1b
    - project-subnet-private2-us-east-1b

The top right corner shows the AWS account information: Account ID: 6534-2712-8219, Federated user: voclabs/user2647504=u2430904@uel.ac.uk @ 6534-2712-8219.

The screenshot shows the success summary of the VPC creation workflow:

- Success**
- Details**
- Success steps:
  - Create VPC: vpc-0a656e6a9a611cd88
  - Enable DNS hostnames
  - Enable DNS resolution
  - Verifying VPC creation: vpc-0a656e6a9a611cd88
  - Create subnet: subnet-05334c3e05acd12936
  - Create subnet: subnet-05334c0770e44b3fe
  - Create subnet: subnet-0e0dd8f16321f211a
  - Create subnet: subnet-050696e1b9d5384a5
  - Create internet gateway: igw-09f643bba8e66305
  - Attach internet gateway to the VPC
  - Create route table: rtb-04372fb8068f9d9e
  - Create route
  - Associate route table
  - Associate route table
  - Allocate elastic IP: eipalloc-010f21a5f158211fc
  - Allocate elastic IP: eipalloc-09f2e09948c664d8
  - Create NAT gateway: nat-067bbcced16186089e
  - Create NAT gateway: nat-0d1108ade81cd56a
  - Wait for NAT Gateways to activate
  - Create route table: rtb-01def4d40886c349828
  - Create route
  - Associate route table
  - Verify route table creation

At the bottom, there is a **View VPC** button.



### 3. Security Group Implementation

**Details:** Here I created a security group with the Hospital Security Group so that I can connect my EC2 with it afterward.

The screenshot shows the 'Create security group' form. The security group name is `Hospital Security Group`. The inbound rules section contains a single rule allowing `HTTP` traffic from `Anywhere...` on port `80` to the `0.0.0.0/0` source. The right sidebar provides account and organization details.

The screenshot shows the AWS VPC Security Groups console. A success message at the top states: "Security group (sg-02bc10067b99c3065 | Hospital Security Group) was created successfully". The main page displays the details of the newly created security group, including its name, ID, owner, and rule counts. The "Inbound rules" tab is selected, showing one rule: "hospital security..." (sgr-08e6bc5f4aab68a93) allowing HTTP traffic from port 80 to 0.0.0.0/0. The sidebar on the left provides navigation for VPC management, including Virtual private cloud, Security groups, DNS firewall, Network ACLs, and more.

## 4. EC2 Server Implementation

**Details:** Here I created two EC2 servers in VPC first one in the us-east-1a availability zone with the name Hospital Server in the public1 subnet and the second in the us-east-1b availability zone with the name Hospital Backup Server in the public2 subnet to make my system available if any disasters occur. I attached both servers with the above security group and chose Amazon Linux AMI for them. I created this to deploy my system on them and use them as servers.

The screenshot shows the AWS EC2 Instances console. The user is launching a new instance named "Hospital Server". In the "Summary" section, it shows 1 instance. The "Software image (AMI)" field is set to "Amazon Linux 2023 AMI 2023.1.2...". The "Virtual server type (instance type)" is "t2.micro". The "Firewall (security group)" is set to "Hospital Security Group". Under "Storage (volumes)", it shows 1 volume(s) - 8 GiB. A tooltip for the free tier of t2.micro instances is displayed. At the bottom, there are "Cancel", "Launch instance", and "Review commands" buttons.

**AWS Services** | Search [Option+S] N. Virginia voulabs/user2647504@u2430904@uel.ac.uk @ 6534-2712-8219 ▲

**Key pair (login) Info**

**Network settings Info**

VPC - required Info  
vpc-0a656e6a9a611cb8 (project-vpc)  
10.0.0.0/16

Subnet Info  
subnet-033ec3e05acd120f6 project-subnet-public1-us-east-1a  
VPC: vpc-0a656e6a9a611cb8 Owner: 65342728219 Availability Zone: us-east-1a  
IP addresses available: 4090 CIDR: 10.0.0.0/20

Auto-assign public IP Info  
Enable

Firewall (security groups) Info  
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Create security group Select existing security group

Common security groups Info  
Select security groups

Hospital Security Group sg-02bc10067b99c3065 X  
VPC: vpc-0a656e6a9a611cb8

Security groups that you add or remove here will be added to or removed from all your network interfaces.

Advanced network configuration

**Summary**

Number of instances Info  
1

Software Image (AMI)  
Amazon Linux 2023 AMI 2023.1.2...read more  
ami-0ba52ed0321b52alc

Virtual server type (instance type)  
t2.micro

Firewall (security group)  
Hospital Security Group

Storage (volumes)  
1 volume(s) - 8 GiB

**Free tier: In your first year includes**  
750 hours of t2.micro in the Regions in which t2.micro is unavailable, instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million I/Os, 1 GiB of snapshots, and 100 GiB of bandwidth to the internet.

Cancel Launch instance Review commands

**AWS Services** | Search [Option+S] N. Virginia voulabs/user2647504@u2430904@uel.ac.uk @ 6534-2712-8219 ▲

**EC2 > Instances > Launch an instance**

**Success**  
Successfully initiated launch of instance (i-019d2eb9bac02fa09)

Launch log

**Next Steps**

What would you like to do next with this instance, for example "create alarm" or "create backup"

<b>Create billing and free tier usage alerts</b> To manage costs and avoid surprise bills, set up email notifications for billing and free tier usage thresholds. <a href="#">Create billing alerts</a>	<b>Connect to your instance</b> Once your instance is running, log into it from your local computer. <a href="#">Connect to Instance</a> <a href="#">Learn more</a>	<b>Connect an RDS database</b> Configure the connection between an EC2 instance and a database to allow traffic flow between them. <a href="#">Connect an RDS database</a> <a href="#">Create a new RDS database</a> <a href="#">Learn more</a>	<b>Create EBS snapshot policy</b> Create a policy that automates the creation, retention, and deletion of EBS snapshots. <a href="#">Create EBS snapshot policy</a>
<b>Manage detailed monitoring</b> Enable or disable detailed monitoring for the instance. If you enable detailed monitoring, the Amazon EC2 console displays monitoring graphs with a 1-minute period. <a href="#">Manage detailed monitoring</a>	<b>Create Load Balancer</b> Create a application, network gateway or classic Elastic Load Balancer. <a href="#">Create Load Balancer</a>	<b>Create AWS budget</b> AWS Budgets allows you to create budgets, forecast spend, and take action on your costs and usage from a single location. <a href="#">Create AWS budget</a>	<b>Manage CloudWatch alarms</b> Create or update Amazon CloudWatch alarms for the instance. <a href="#">Manage CloudWatch alarms</a>

## 5. EC2 Backup Server Creation

The screenshot shows the AWS EC2 'Launch an instance' wizard. The 'Summary' step is displayed, showing the following details:

- Number of instances:** 1
- Software image (AMI):** Amazon Linux 2023 AMI 2023.1.2... (ami-08a52ddb321b32ac)
- Virtual server type (instance type):** t2.micro
- Firewall (security group):** Hospital Security Group
- Storage (volumes):** 1 volume(s) - 8 GiB

A callout box highlights the 'Free tier' information: "Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million I/Os, 1 GiB of snapshots, and 100 GiB of bandwidth to the internet."

At the bottom right of the summary step, there are 'Cancel', 'Launch instance', and 'Review commands' buttons.

The screenshot shows the 'Key pair (login)' step of the EC2 instance creation wizard. The 'Summary' step is visible above it. The 'Key pair (login)' section includes:

- VPC - required:** project-vpc (project-vpc)
- Subnet:** subnet-0535ce770e44b3fe (project-subnet-public2-us-east-1b)
- Auto-assign public IP:** Enable
- Firewall (security groups):** Hospital Security Group (sg-02bc10067b99c3065)
- Common security groups:** Hospital Security Group

A callout box highlights the 'Free tier' information: "Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million I/Os, 1 GiB of snapshots, and 100 GiB of bandwidth to the internet."

At the bottom right of the key pair step, there are 'Cancel', 'Launch instance', and 'Review commands' buttons.

**Instances (2/3) Info**

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic...
Hospital Back...	i-04953a659caf47156	Running	t2.micro	2/2 checks passed	No alarms	us-east-1b	ec2-52-91-158-238.co...	52.91.158.238	-
Hospital Server	i-019d2eb9bac02fa09	Running	t2.micro	2/2 checks passed	No alarms	us-east-1a	ec2-3-89-131-124.com...	5.89.131.124	-
Bastion Host	i-065853aa4f0d49133	Running	t2.micro	2/2 checks passed	No alarms	us-east-1a	ec2-44-202-28-207.co...	44.202.28.207	-

**Instances: i-04953a659caf47156 (Hospital Backup Server), i-019d2eb9bac02fa09 (Hospital Server)**

**Monitoring**

1h 3h 12h 1d 3d 1w Custom Add to dashboard

CPU utilization (%) Status check failed (any) (count) Status check failed (instance) (count) Status check failed (system) (count)

Percent Various units Various units Various units

83.7 1 1 1

41.8 0.5 0.5 0.5

0 0 0 0

12:15 12:30 12:45 13:00 12:15 12:30 12:45 13:00 12:15 12:30 12:45 13:00

## 6. RDS Implementation

**Details:** Here I created two RDS one in the private1 subnet and the second in the private2 subnet for backup purposes. I created these databases to store patient and hospital data with MySQL engine option and a minimum allocated storage of 20GB. I enabled auto scaling so that it can scale up and down for price optimization and choose General Purpose SSD drive for these databases.

Successfully created DB-Subnet-Group. View subnet group

RDS > Subnet groups

Subnet groups (1)

Name	Description	Status	VPC
db-subnet-group	DB Subnet Group	Complete	vpc-0a656e6a9a611cb

Account Organization Service Quotas Billing Dashboard Settings

Switch role Sign out

**Amazon RDS**

- Dashboard
- Databases
- Query Editor
- Performance insights
- Snapshots
- Exports in Amazon S3
- Automated backups
- Reserved instances
- Proxies
- Subnet groups**
  - Parameter groups
  - Option groups
  - Custom engine versions
  - Zero-ETL Integrations: Preview
- Events
- Event subscriptions
- Recommendations (0)
- Certificate update

**Create database**

**Choose a database creation method** [Info](#)

**Standard create**  
You set all of the configuration options, including ones for availability, security, backups, and maintenance.

**Easy create**  
Use recommended best-practice configurations. Some configuration options can be changed after the database is created.

**Engine options**

**Engine type** [Info](#)

Aurora (MySQL Compatible)

Aurora (PostgreSQL Compatible)

MySQL

MariaDB

PostgreSQL

Oracle

Microsoft SQL Server

**Templates**  
Choose a sample template to meet your use case.

**Production**  
Use defaults for high availability and fast, consistent performance.

**Dev/Test**  
This instance is intended for development use outside of a production environment.

**Free tier**  
Use RDS Free Tier to develop new applications, test existing applications, or gain hands-on experience with Amazon RDS. [Info](#)

**Availability and durability**

**Deployment options** [Info](#)  
The deployment options below are limited to those supported by the engine you selected above.

**Multi-AZ DB Cluster - new**  
Creates a DB cluster with a primary DB instance and two readable standby DB instances, with each DB instance in a different Availability Zone (AZ). Provides high availability, data redundancy and increases capacity to serve read workloads.

**Multi-AZ DB instance**  
Creates a primary DB instance and a standby DB instance in a different AZ. Provides high availability and data redundancy, but the standby DB instance doesn't support connections for read workloads.

**Single DB instance**  
Creates a single DB instance with no standby DB instances.

**Settings**

**DB instance identifier** [Info](#)  
Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.

**Storage**

**Storage type** [Info](#)  
General Purpose SSD (gp3)  
Performance scales independently from storage

**Allocated storage** [Info](#)  
 GiB  
Minimum: 20 GiB. Maximum: 6,144 GiB

**After you modify the storage for a DB instance, the status of the DB instance will be in storage-optimization. Your instance will remain available as the storage-optimization operation completes.** [Learn more](#)

**Advanced settings**  
Baseline IOPS of 3,000 IOPS and storage throughput of 125 MiBps are included for allocated storage less than 400 GiB.

**Storage autoscaling** [Info](#)  
Provides dynamic scaling support for your database's storage based on your application's needs.

**Enable storage autoscaling**  
Enabling this feature will allow the storage to increase after the specified threshold is exceeded.

**Maximum storage threshold** [Info](#)  
Charges will apply when your database autoscales to the specified threshold  
 GiB  
The minimum value is 22 GiB and the maximum value is 6,144 GiB

**Connectivity** [Info](#)

AWS Services Search [Option+S] N. Virginia vocabs/user2647504+u2430904@uel.ac.uk @ 6534-2712-8219 ▾

**Amazon RDS** X

Dashboard Databases Query Editor Performance insights Snapshots Exports in Amazon S3 Automated backups Reserved instances Proxies Submit review

RDS > Databases

Consider creating a Blue/Green Deployment to minimize downtime during upgrades You may want to consider using Amazon RDS Blue/Green Deployments and minimize your downtime during upgrades. A Blue/Green Deployment provides a staging environment for changes to production databases. [RDS User Guide](#) [Aurora User Guide](#)

Databases (1)

DB identifier Status Role Engine Region & AZ Size Actions CPU Current activity Maintenance VPC

DB identifier	Status	Role	Engine	Region & AZ	Size	Actions	CPU	Current activity	Maintenance	VPC
hospitaldb-db	Available	Instance	MySQL Community	us-east-1a	db.t3.micro	3 Actions	2.67%	0 Connections	none	vpc

AWS Services Search [Option+S] N. Virginia vocabs/user2647504+u2430904@uel.ac.uk @ 6534-2712-8219 ▾

**Amazon RDS** X

Dashboard Databases Query Editor Performance insights Snapshots Exports in Amazon S3 Automated backups Reserved instances Proxies Subnet groups Parameter groups Option groups Custom engine versions Zero-ETL Integrations [Preview](#) Events Event subscriptions Recommendations [3](#) Certificate update

**hospitaldb-db**

**Summary**

DB identifier hospitaldb-db CPU 2.60% Status Available Class db.t3.micro Account Account ID: 6534-2712-8219 Federated user: vocabs/user2647504+u2430904@uel.ac.uk Organization Organization Service Quotas Service Quotas Billing Dashboard Billing Dashboard

Role Instance Current activity 0 Connections Engine MySQL Community Region & AZ us-east-1a

Connectivity & security Monitoring Logs & events Configuration Maintenance & backups Tags

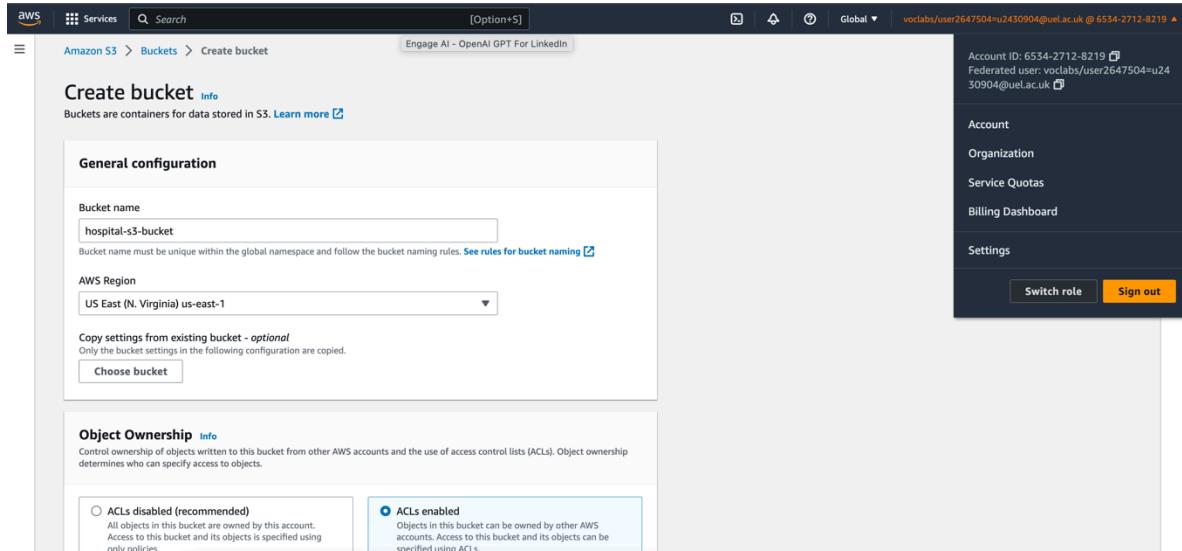
**Connectivity & security**

Endpoint & port	Networking	Security
Endpoint hospitaldb-db.csfhryguf0.us-east-1.rds.amazonaws.com	Availability Zone us-east-1a VPC project-vpc (vpc-0a656e6a9a611cb8)	VPC security groups Hospital Security Group (sg-02bc10067b99c3065) Active
Port 3306	Subnet group db-subnet-group	Publicly accessible No Certificate authority Info rds-ca-2019
	Subnets subnet-050e96e1b9d58e45 subnet-0e0ad8f16521f211a	Certificate authority date August 22, 2024, 18:08 (UTC+01:00) DB instance certificate expiration date
	Network type	

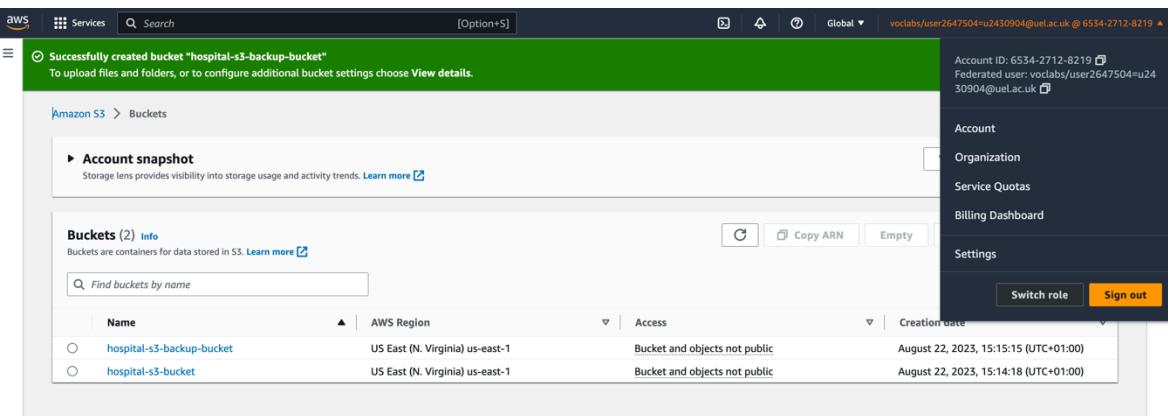
Switch role Sign out

## 7. Two S3 Bucket with ACLs Implementation

**Details:** Here I created two S3 Buckets, one in the private1 subnet and the second in the private2 subnet for backup purposes. I created these buckets to store patients' image data and data that acquire more space like prescriptions, test reports, and other documents. I have also enabled the ACLs access control list to ensure that only authorized users can access this data because of patient data privacy.



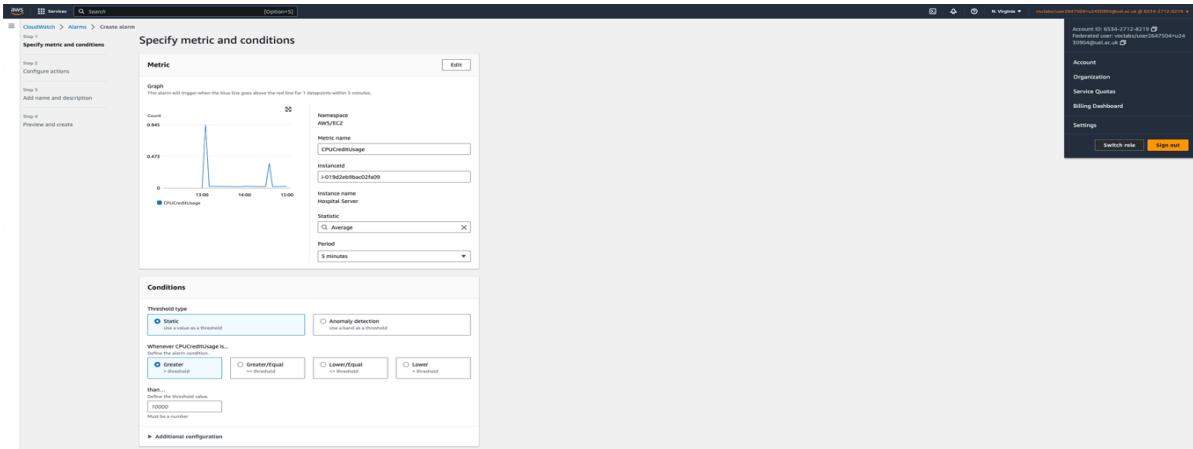
The screenshot shows the 'Create bucket' page in the AWS Management Console. The 'General configuration' section includes a 'Bucket name' field containing 'hospital-s3-bucket', an 'AWS Region' dropdown set to 'US East (N. Virginia) us-east-1', and a 'Copy settings from existing bucket - optional' section with a 'Choose bucket' button. The 'Object Ownership' section shows that 'ACLs enabled' is selected. The right sidebar displays account information and navigation links.



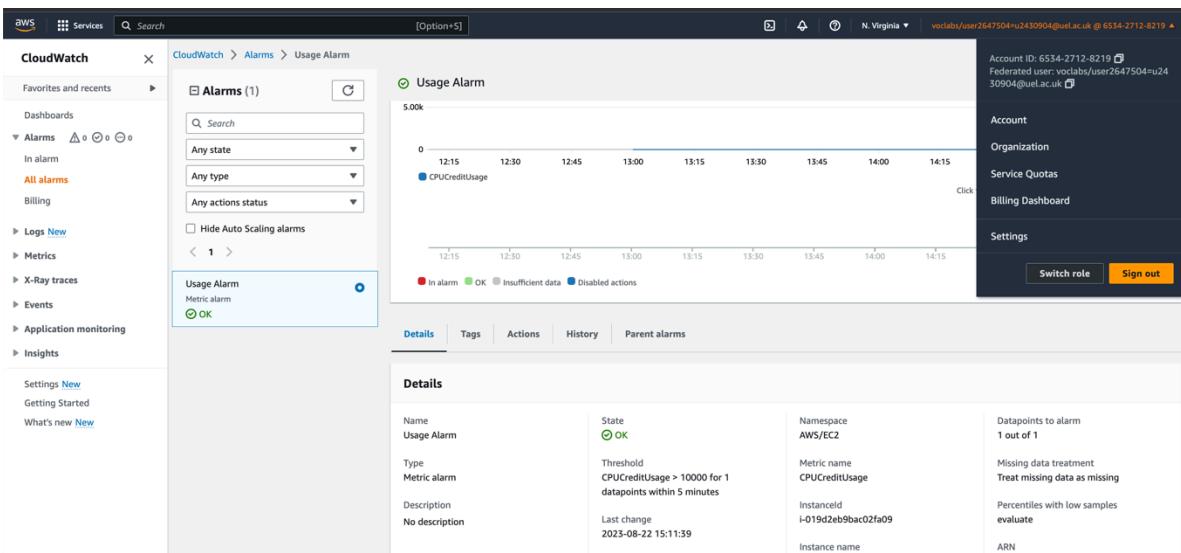
The screenshot shows the 'Buckets' page in the AWS Management Console. A green success message at the top states 'Successfully created bucket "hospital-s3-backup-bucket"'. Below it, there's an 'Account snapshot' summary and a table listing two buckets: 'hospital-s3-backup-bucket' and 'hospital-s3-bucket'. Both buckets are in the 'US East (N. Virginia) us-east-1' region and have 'Bucket and objects not public' access. The table includes columns for Name, AWS Region, Access, and Creation date. The right sidebar shows account details and navigation links.

## 8. Cloud Watch Usage Alarm and SNS Notification Implementation

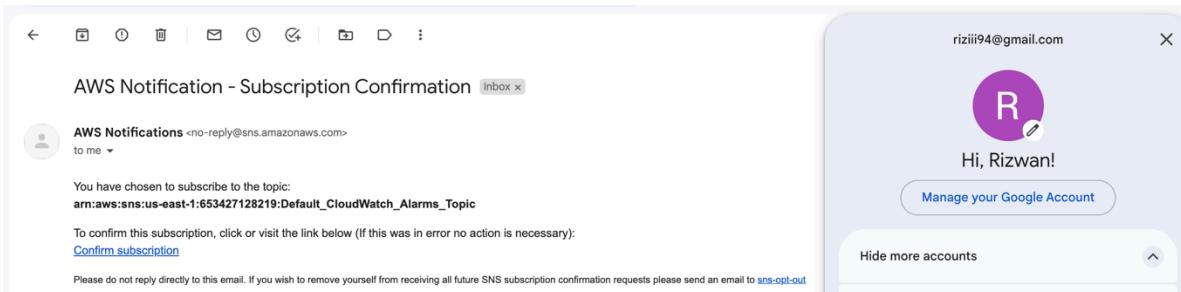
**Details:** Here I created an alarm on my EC2 server CPU usage and set the threshold so if usage increases, this alarm will trigger and send me a notification via email. For notification, I used the Amazon SNS service.



The screenshot shows the 'Specify metric and conditions' step of creating a new alarm. The 'Metric' section displays a graph of CPU Credit Usage over a 5-minute period. The 'Conditions' section shows a static threshold type where the metric value must be greater than 10000. The alarm is named 'Usage Alarm' and is associated with the AWS/EC2 namespace.



The screenshot shows the 'Usage Alarm' listed under the 'Alarms' section. The alarm is currently in an 'OK' state. The 'Details' tab provides a summary of the alarm configuration, including its name, state, namespace, metric name, threshold, last change, instance name, and ARN.



The screenshot shows an email from AWS Notifications confirming a subscription to the topic 'arn:aws:sns:us-east-1:653427128219:Default\_CloudWatch\_Alerts\_Topic'. The email includes a link to 'Confirm subscription'. To the right, a Google Account sign-in overlay is visible, displaying a purple 'R' icon and the message 'Hi, Rizwan! Manage your Google Account'.

# Costing

The screenshot shows the AWS Pricing Calculator interface. At the top, it says "aws pricing calculator". Below that, the path "AWS Pricing Calculator > Hospital System" is shown. The main title is "Hospital System" with an "Edit" link. On the right, there are "Export" and "Share" buttons. The "Estimate summary" section shows the following costs:

Upfront cost	Monthly cost	Total 12 months cost
0.00 USD	4,325.65 USD	51,907.80 USD

It also includes the note "Includes upfront cost". To the right, under "Getting Started with AWS", there are buttons for "Get started for free" and "Contact Sales".

Note : In future we can reduce the cost by auto scaling of services and by reducing the usage of services if note needed or needed on specific times.

## These services chosen for the system

Note: I created price estimation on monthly basis with the selection of medium usage of each service.

The screenshot shows the AWS Pricing Calculator interface with the title "Hospital System". The table lists the following services and their details:

Service Name	Status	Upfront cost	Monthly cost	Description	Region	Config Summary
Amazon API Gateway	-	0.00 USD	930.00 USD	Need this for devel...	US East (N. Virginia)	REST API request units (millions), Cache memory size (GB) (None), WebSoc...
Amazon CloudFront	-	0.00 USD	175.20 USD	To available my syst...	US East (N. Virginia)	Data transfer out to internet (1000 GB per month), Data transfer out to inter...
Amazon CloudWatch	-	0.00 USD	300.00 USD	-	US East (N. Virginia)	Number of Metrics (includes detailed and custom metrics) (1000)
Amazon Simple Not...	-	0.00 USD	299.48 USD	-	US East (N. Virginia)	Requests (200 million per month), EMAIL/EMAIL-JSON Notifications (10 milli...
Amazon Virtual Priv...	-	0.00 USD	1,476.20 USD	-	US East (N. Virginia)	Working days per month (22), Number of Site-to-Site VPN Connections (10) ...
Amazon EC2	-	0.00 USD	581.08 USD	For server purpose	US East (N. Virginia)	Tenancy (Shared Instances), Operating system (Linux), Workload (Consistent,...
Amazon RDS Custo...	-	0.00 USD	328.69 USD	to store patients data	US East (N. Virginia)	Storage for each RDS Custom for SQL Server instance (General Purpose SSD ...
Amazon Simple Sto...	-	0.00 USD	230.00 USD	-	US East (N. Virginia)	S3 Standard storage (10000 GB per month)
Amazon Route 53	-	0.00 USD	5.00 USD	for routing purpose	US East (N. Virginia)	Hosted Zones (10)

# **Analysis and Reflection**

## **System maintenance**

- The system in the AWS cloud should be patched and updated regularly to ensure that it is up to date with the latest security fixes.
- The system should also be monitored for performance and availability issues.
- A disaster recovery plan should be in place to ensure that the system can be restored in the event of a failure.

## **Manage its evolution**

- The system in the AWS cloud can be evolved to meet the changing needs of the hospital by adding new features, improving performance, or making the system more secure.
- For example, the system could be updated to include a new patient portal that allows patients to access their medical records online.
- The system could also be made more secure by implementing two-factor authentication.

## **Future upgradation**

- The system in the AWS cloud is constantly being upgraded with new features and capabilities.
- For example, the system is currently being upgraded to support the use of artificial intelligence.
- This will allow the system to provide more personalized care to patients.

## **New rule and regulations**

- The system in the AWS cloud will be compliant with new rules and regulations by implementing the appropriate security controls.
- For example, the system will be required to implement encryption to protect patient data.
- The system will also be required to have a privacy policy that outlines how patient data will be collected and used.



## References

*What is IAM? - AWS Identity and Access Management* (no date). Available at: <https://docs.aws.amazon.com/IAM/latest/UserGuide/introduction.html>.

*What is Amazon VPC? - Amazon Virtual Private Cloud* (no date). Available at:

<https://docs.aws.amazon.com/vpc/latest/userguide/what-is-amazon-vpc.html>.

*Choosing regions and availability zones - Amazon ElastiCache* (no date). Available at:

[https://docs.aws.amazon.com/AmazonElastiCache/latest/mem-ug/RegionsAndAZs.h  
tml](https://docs.aws.amazon.com/AmazonElastiCache/latest/mem-ug/RegionsAndAZs.html).

Malik, F. (2022) “What is AWS VPC and Subnets? - FinTechExplained - Medium,”

*Medium*, 14 December. Available at:

<https://medium.com/fintechexplained/what-is-aws-vpc-and-subnets-43eb67bea492>.

Wolmarans, B. (2023) “AWS Security Groups Guide,” *Sysdig* [Preprint]. Available

at: <https://sysdig.com/blog/aws-security-groups-guide/>.

Wright, N. (2022) *AWS databases: how to choose the best storage option* | Jefferson Frank. Available at: <https://www.jeffersonfrank.com/insights/choosing-an-aws-database>.

*What is Amazon S3? - Amazon Simple Storage Service* (no date). Available at:

<https://docs.aws.amazon.com/AmazonS3/latest/userguide>Welcome.html>.

Jain, A. (2023) “Amazon Simple Notification Service (AWS SNS),” *Cloud Training*

*Program*, 7 August. Available at:

<https://k21academy.com/amazon-web-services/amazon-sns/>.