AWS re:Start

LAB

Optimize Utilization

WEEK 11

aws re/start

Overview

Optimizing resource utilization in AWS is crucial for reducing costs and enhancing efficiency. By carefully managing and optimizing Amazon EC2 instances, you can significantly lower your cloud expenditures. This involves right-sizing instances based on actual usage, taking advantage of reserved instances or spot instances for cost savings, and ensuring that idle instances are terminated or scaled down. Regularly reviewing and adjusting your EC2 configurations ensures that you are not over-provisioning and are using your resources in the most cost-effective manner.

Additionally, utilizing tools like the AWS Pricing Calculator allows you to estimate the costs of various AWS services and plan your budget accordingly. By inputting your specific requirements and configurations, you can get a clear picture of your potential expenses and make informed decisions to optimize costs. This proactive approach to cost management helps in maintaining a balance between performance and expenditure, ensuring that your AWS environment is both efficient and economical.

Topics covered

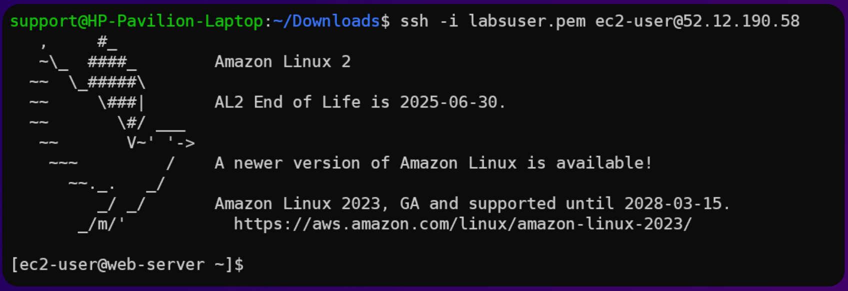
- Optimize an Amazon Elastic Compute Cloud (Amazon EC2) instance to reduce costs.
- Use the AWS Pricing Calculator to estimate AWS service costs.

Task 1

Optimize the website to reduce costs

Step 1: Connect to the CafeInstance

Connect to the **CafeInstance** by using SSH.

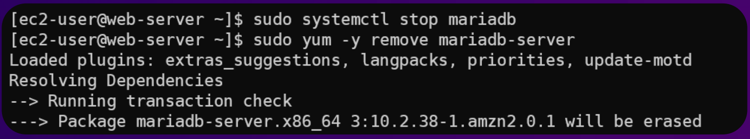


```
support@HP-Pavilion-Laptop:~/Downloads$ ssh -i labsuser.pem ec2-user@52.12.190.58
   ,     #_
   ~\_  ####_        Amazon Linux 2
  ~~  \_#####\
  ~~     \###|       AL2 End of Life is 2025-06-30.
  ~~       \#/ ___
   ~~       V~' '->
    ~~~         /    A newer version of Amazon Linux is available!
      ~~._.   _/
         _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
       _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/

[ec2-user@web-server ~]$
```

Step 2: Uninstall MariaDB

Stop the local database and uninstall it from the **CafeInstance**. In the **CafeInstance** SSH window, enter the following commands. Then, close the SSH window for the **CafeInstance**.



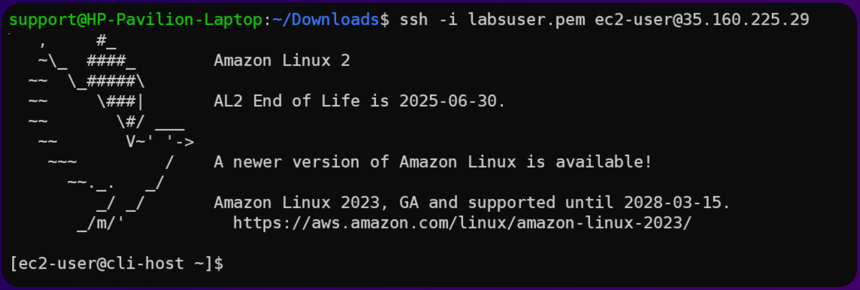
```
[ec2-user@web-server ~]$ sudo systemctl stop mariadb
[ec2-user@web-server ~]$ sudo yum -y remove mariadb-server
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
---> Package mariadb-server.x86_64 3:10.2.38-1.amzn2.0.1 will be erased
```

Task 1

Optimize the website to reduce costs

Step 3: Connect to the CLI Host instance

Connect to the **CLI Host** instance by using SSH.

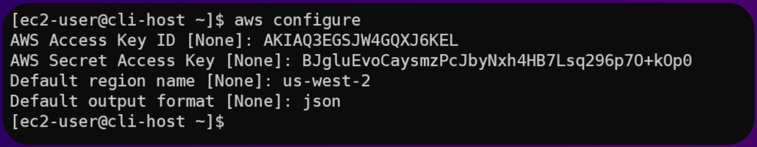


```
support@HP-Pavilion-Laptop:~/Downloads$ ssh -i labsuser.pem ec2-user@35.160.225.29
   ,     #_
   ~\_  ####_        Amazon Linux 2
  ~~  \_#####\
  ~~     \###|       AL2 End of Life is 2025-06-30.
  ~~       \#/ ___
   ~~       V~' '->
    ~~~         /    A newer version of Amazon Linux is available!
      ~~._.   _/
         _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
       _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/

[ec2-user@cli-host ~]$
```

Step 4: Configure the AWS CLI

Discover the region in which the **CLI Host** instance is running, and update the AWS CLI software with the credentials.



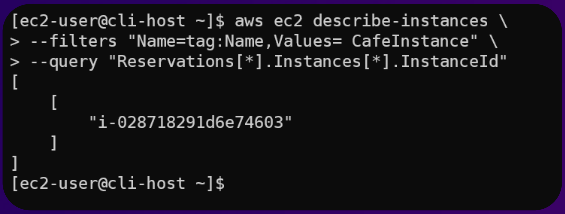
```
[ec2-user@cli-host ~]$ aws configure
AWS Access Key ID [None]: AKIAQ3EGSJW4GQXJ6KEL
AWS Secret Access Key [None]: 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
Default region name [None]: us-west-2
Default output format [None]: json
[ec2-user@cli-host ~]$
```

Task 1

Optimize the website to reduce costs

Step 5: Determine the Instance ID

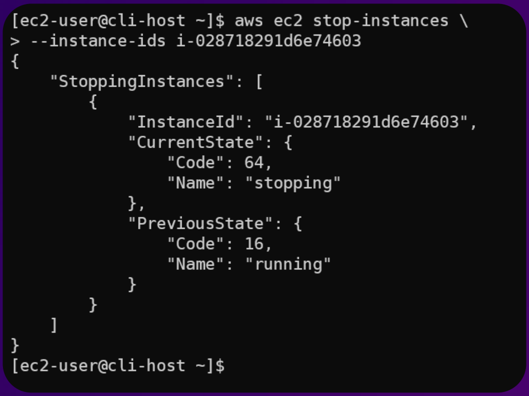
To determine the Instance ID of the **CafeInstance**, enter the following aws ec2 describe-instances command.



```
[ec2-user@cli-host ~]$ aws ec2 describe-instances \
> --filters "Name=tag:Name,Values= CafeInstance" \
> --query "Reservations[*].Instances[*].InstanceId"
[
    [
        "i-028718291d6e74603"
    ]
]
[ec2-user@cli-host ~]$
```

Step 6: Stop the CafeInstance

To stop the **CafeInstance**, enter the following aws ec2 stop-instances command.



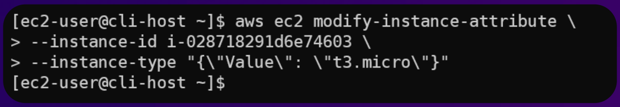
```
[ec2-user@cli-host ~]$ aws ec2 stop-instances \
> --instance-ids i-028718291d6e74603
{
    "StoppingInstances": [
        {
            "InstanceId": "i-028718291d6e74603",
            "CurrentState": {
                "Code": 64,
                "Name": "stopping"
            },
            "PreviousState": {
                "Code": 16,
                "Name": "running"
            }
        }
    ]
}
[ec2-user@cli-host ~]$
```

Task 1

Optimize the website to reduce costs

Step 7: Change the instance type

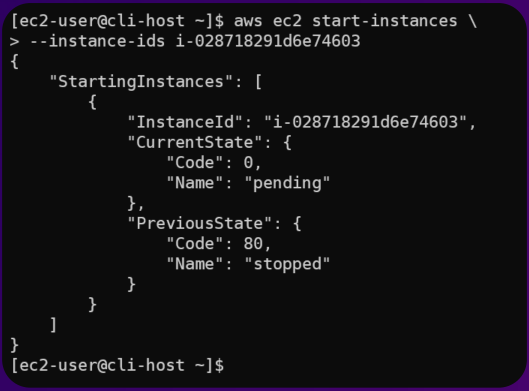
To change the instance type to **t3.micro**, enter the following aws ec2 modify-instance-attribute command.



```
[ec2-user@cli-host ~]$ aws ec2 modify-instance-attribute \
> --instance-id i-028718291d6e74603 \
> --instance-type "{\"Value\": \"t3.micro\"}"
[ec2-user@cli-host ~]$
```

Step 8: Start the CafeInstance

To start the **CafeInstance**, enter the following aws ec2 start-instances command.



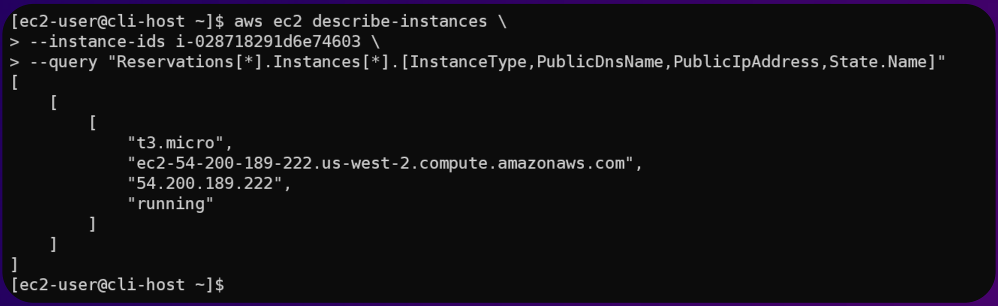
```
[ec2-user@cli-host ~]$ aws ec2 start-instances \
> --instance-ids i-028718291d6e74603
{
    "StartingInstances": [
        {
            "InstanceId": "i-028718291d6e74603",
            "CurrentState": {
                "Code": 0,
                "Name": "pending"
            },
            "PreviousState": {
                "Code": 80,
                "Name": "stopped"
            }
        }
    ]
}
[ec2-user@cli-host ~]$
```

Task 1

Optimize the website to reduce costs

Step 9: Check the state of the instance

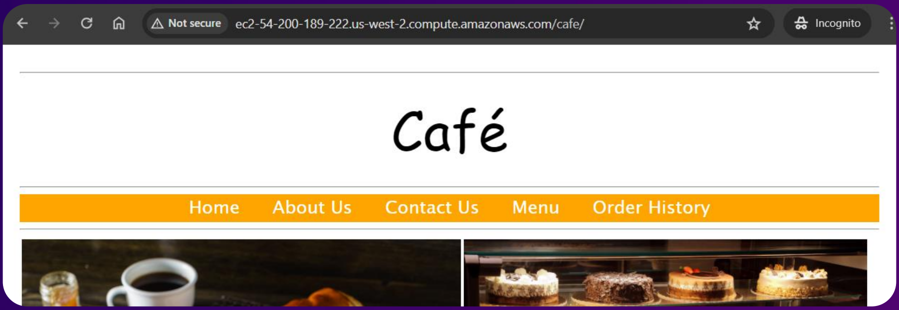
To check the current state of the instance, enter the following aws ec2 describe-instances command. Wait for the instance to reach the running state. Also, record the **PublicDnsName** and **PublicIPAddress** values that are returned by the command.



```
[ec2-user@cli-host ~]$ aws ec2 describe-instances \
> --instance-ids i-028718291d6e74603 \
> --query "Reservations[*].Instances[*].[InstanceType,PublicDnsName,PublicIpAddress,State.Name]"
[
    [
        [
            "t3.micro",
            "ec2-54-200-189-222.us-west-2.compute.amazonaws.com",
            "54.200.189.222",
            "running"
        ]
    ]
]
[ec2-user@cli-host ~]$
```

Step 10: Test the Café website

Access the instance PublicDnsName or PublicIPAddress, and test the Café website to make sure that it is functional.

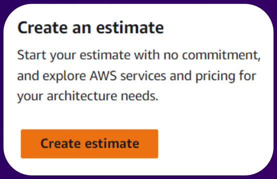


Task 2

Use the AWS Pricing Calculator to estimate AWS service costs

Step 1: Create estimate

Open the AWS Pricing Calculator, and click on Create estimate.

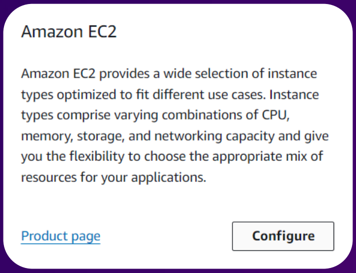


Create an estimate

Start your estimate with no commitment, and explore AWS services and pricing for your architecture needs.

Step 2: Configure Amazon EC2 estimate

Choose Configure in the Amazon EC2 service panel.



Amazon EC2

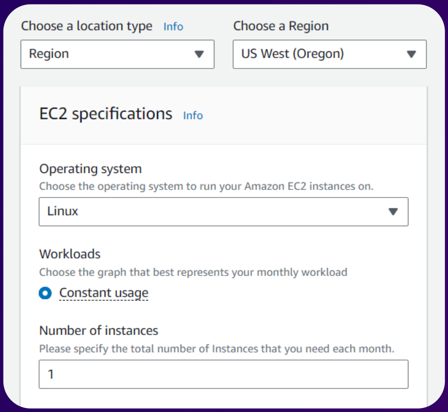
Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instance types comprise varying combinations of CPU, memory, storage, and networking capacity and give you the flexibility to choose the appropriate mix of resources for your applications.

Task 2

Use the AWS Pricing Calculator to estimate AWS service costs

Step 3: EC2 specifications

Configure the following EC2 specifications parameters.



Choose a location type Info: Region

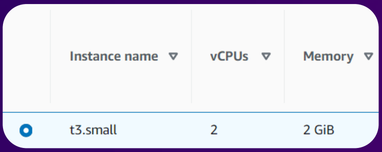
Choose a Region: US West (Oregon)

EC2 specifications Info

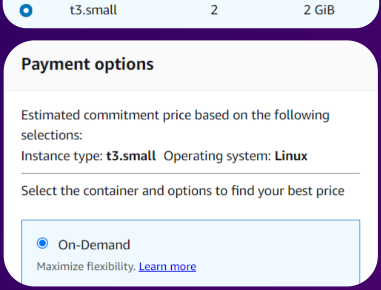
Operating system
Choose the operating system to run your Amazon EC2 instances on.
Linux

Workloads
Choose the graph that best represents your monthly workload
● Constant usage

Number of instances
Please specify the total number of Instances that you need each month.
1



	Instance name	vCPUs	Memory
●	t3.small	2	2 GiB



Payment options

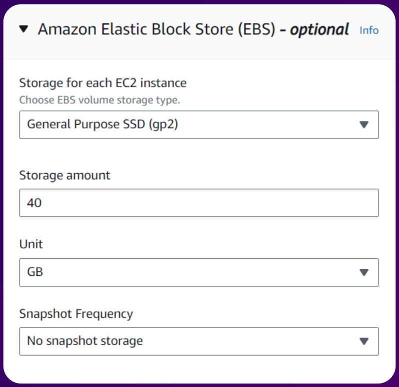
Estimated commitment price based on the following selections:
Instance type: **t3.small** Operating system: **Linux**

Select the container and options to find your best price

● On-Demand
Maximize flexibility. Learn more

Step 4: EBS specifications

Configure the following EBS specifications parameters.



▾ Amazon Elastic Block Store (EBS) - *optional* Info

Storage for each EC2 instance
Choose EBS volume storage type.
General Purpose SSD (gp2)

Storage amount
40

Unit
GB

Snapshot Frequency
No snapshot storage

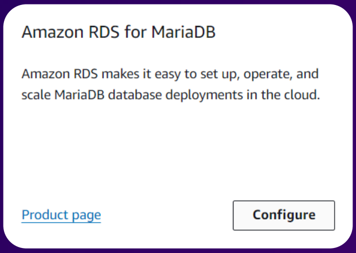
aws re/start

Task 2

Use the AWS Pricing Calculator to estimate AWS service costs

Step 5: Configure Amazon RDS estimate

Choose Configure in the Amazon RDS for MariaDB service panel.

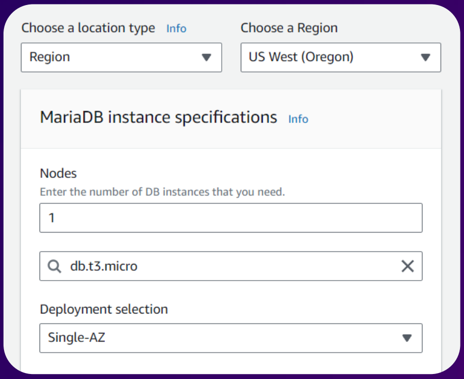


Amazon RDS for MariaDB

Amazon RDS makes it easy to set up, operate, and scale MariaDB database deployments in the cloud.

Step 6: MariaDB specifications

Configure the following MariaDB specifications parameters.



Choose a location type: Region

Choose a Region: US West (Oregon)

MariaDB instance specifications

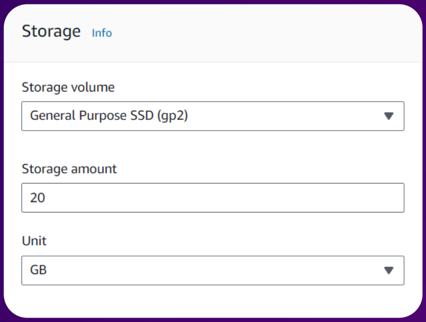
Nodes

Enter the number of DB instances that you need.

1

db.t3.micro

Deployment selection: Single-AZ



Storage

Storage volume: General Purpose SSD (gp2)

Storage amount: 20

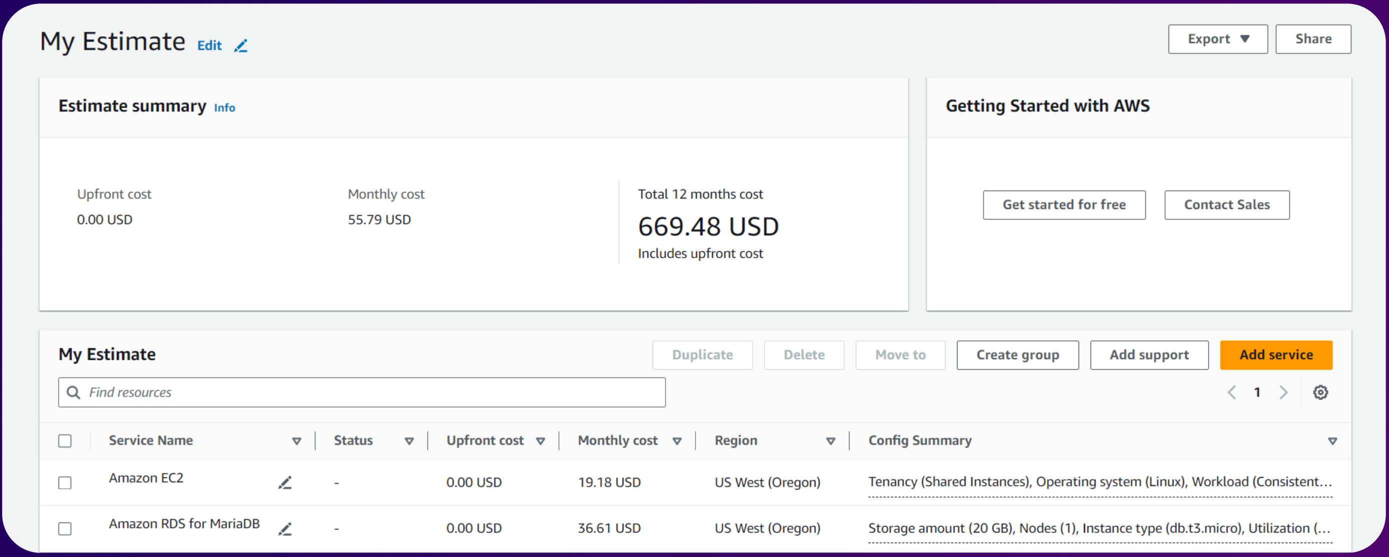
Unit: GB

Task 2

Use the AWS Pricing Calculator to estimate AWS service costs

Step 7: Review Estimate before optimization

This is the estimated cost before optimization.



My Estimate Edit

Export ▼ Share

Estimate summary Info

Upfront cost
0.00 USD

Monthly cost
55.79 USD

Total 12 months cost
669.48 USD
Includes upfront cost

Getting Started with AWS

Get started for free | Contact Sales

My Estimate

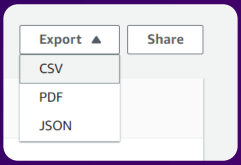
Duplicate | Delete | Move to | Create group | Add support | Add service

Find resources

☐	Service Name	Status	Upfront cost	Monthly cost	Region	Config Summary
☐	Amazon EC2	-	0.00 USD	19.18 USD	US West (Oregon)	Tenancy (Shared Instances), Operating system (Linux), Workload (Consistent…
☐	Amazon RDS for MariaDB	-	0.00 USD	36.61 USD	US West (Oregon)	Storage amount (20 GB), Nodes (1), Instance type (db.t3.micro), Utilization (…

Step 8: Export the estimate

Export the estimate to a comma-separated values (CSV) file.



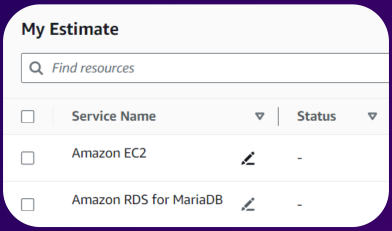
aws re/start

Task 2

Use the AWS Pricing Calculator to estimate AWS service costs

Step 9: Edit the Amazon EC2 estimate

Click Edit next to the Amazon EC2 entry.



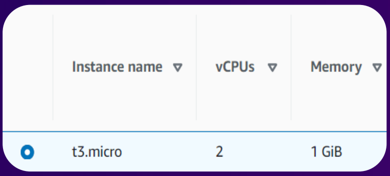
My Estimate

Find resources

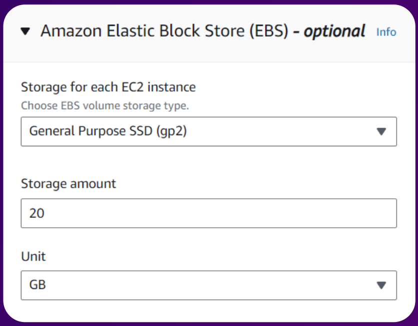
☐	Service Name		Status
☐	Amazon EC2	✎	-
☐	Amazon RDS for MariaDB	✎	-

Step 10: EC2 and EBS specifications

Configure the following EC2 and EBS specifications parameters.



	Instance name	vCPUs	Memory
●	t3.micro	2	1 GiB



▼ Amazon Elastic Block Store (EBS) - *optional* Info

Storage for each EC2 instance
Choose EBS volume storage type.
General Purpose SSD (gp2) ▼

Storage amount
20

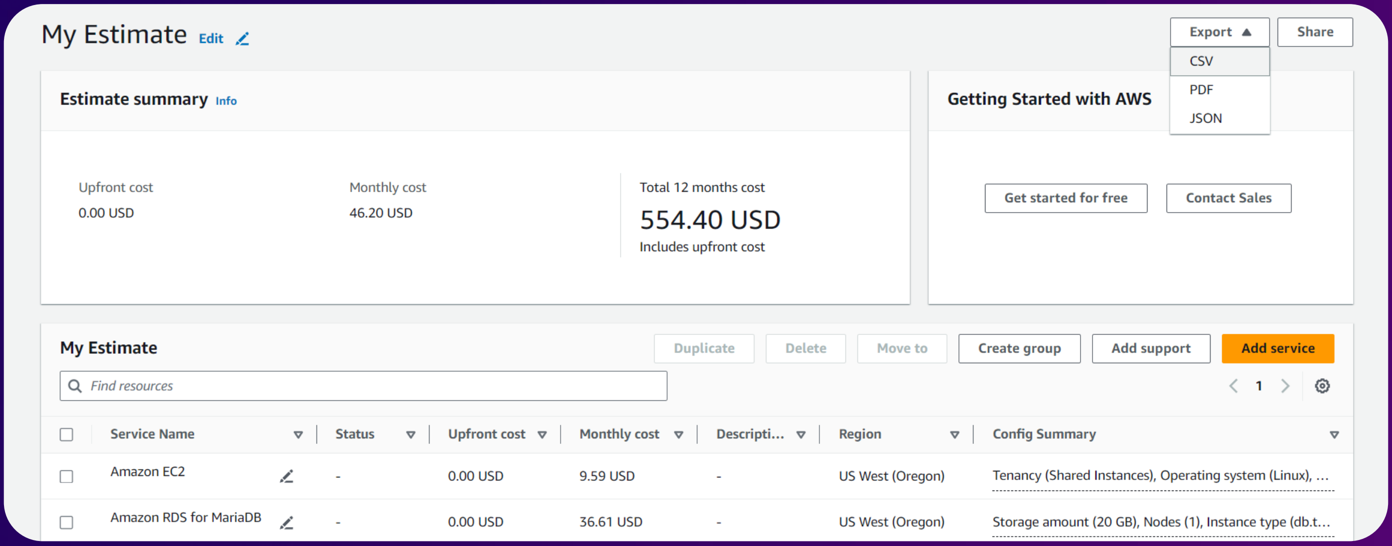
Unit
GB ▼

Task 2

Use the AWS Pricing Calculator to estimate AWS service costs

Step 11: Review Estimate after optimization

This is the estimated cost after optimization. Export the estimate to a comma-separated values (CSV) file.



Upfront cost	Monthly cost	Total 12 months cost
0.00 USD	46.20 USD	554.40 USD Includes upfront cost

☐	Service Name	Status	Upfront cost	Monthly cost	Descripti...	Region	Config Summary
☐	Amazon EC2	-	0.00 USD	9.59 USD	-	US West (Oregon)	Tenancy (Shared Instances), Operating system (Linux), ...
☐	Amazon RDS for MariaDB	-	0.00 USD	36.61 USD	-	US West (Oregon)	Storage amount (20 GB), Nodes (1), Instance type (db.t...

Step 12: Estimate the projected cost savings

Estimate the overall projected cost savings as follows.

```
AWS Services Before Optimization Estimated Monthly Cost: $55.79
AWS Services After Optimization Estimated Monthly Cost: $46.20

Overall monthly cost savings    $9.59
```

aws re/start

Conclusions

The AWS Pricing Calculator

The AWS Pricing Calculator helps estimate service costs, enabling effective budgeting and optimization of AWS resource utilization.

Creating Estimates

Creating estimates with the AWS Pricing Calculator allows for accurate cost planning, ensuring optimized utilization and minimized expenses.

Resizing instances

Resizing instances to match workload requirements optimizes performance and reduces unnecessary costs, enhancing overall resource efficiency.

Uninstalling applications and databases

Uninstalling unused applications and databases frees up resources, contributing to cost savings and more efficient utilization of AWS infrastructure.

Calculating monthly cost savings

Calculating monthly cost savings from optimized utilization practices provides insights into the financial benefits of efficient resource management.

aws re/start

Cristhian Becerra

cristhian-becerra-espinoza

+51 951 634 354

cristhianbecerra99@gmail.com

Lima, Peru

aws re/start