COSC349 Assignment 2 Report

Samuel Ng 2955262
Matt Mouat 7552560

For this assignment, we opted to utilize our first one, building upon it to generate a database that can both save and retrieve data. How this application is intended to be used is that a user can upload data of various kinds, which we have currently kept to basic numbers and letters, and retrieve it via a separate website, arranged in an orderly manner. We utilized three virtual machines in our first assignment to demonstrate our application, but as the AWS (Amazon Web Service) service provides storage, we changed our number of machines to two. For our storage choice, we opted for the RDS (Relational Database Service) database, as it had support for MySQL with little change to the code. We have also utilised the security benefits of Amazons VPC (virtual private cloud) services to be able to limit the traffic to and between our VMS as to increase security and maximize the resources we rent.

Our virtual machines are deployed through Vagrant, using the AWS plugin. With some alterations to the Vagrant file, our application would be easily deployed through a simple git bash terminal, and thus we opted for this route. By providing the Vagrant file with our AWS credentials, our virtual machines would be uploaded as EC2 instances, and be automatically setup without much hassle. The database itself is set up manually through AWS, and thus the only subject of worry was the errors thrown by any scripts or lines of code.

For accessing our database query and retrieval, there will be two websites up, and thus two links. These are provided in the GitHub https://github.com/matt53211/Cosc349asgn2, and can be used to enter data or retrieve data. Provided in the GitHub are also instructions if you wish to set this up for yourself, either on your local computer or on a cloud server. There are in-depth instructions for both, including a disclaimer on the operating system an individual may be using.

Although there were multiple options for storage such as Aurora and MySQL, we persisted with the RDS database as they provided multiple types of SQL servers. RDS databases have an added benefit of deploying MySQL servers within a short period, in a cost-efficient manner, which are all priorities of a simple database. The data is stored in a simple table, which can then be retrieved and displayed via another website. The way the data is stored is relatively insecure, due to our style of handling.

Difficulties

There were multiple issues going forward from our development. The first issue we had to tackle was altering the Vagrant file. As the AWS plugin had specific requirements, we had to first figure these out before adding them into the Vagrant file, ensuring we had input the right credentials. The second issue we had to tackle was the development and alteration of the PHP files. These files, as they were based upon our first assignment, were considerably underdeveloped and thus needed work. On top of that, we were caught off guard by the need to specify a webpage within the PHP files themselves, which led to more issues. Finally, we had multiple errors, mostly to do with the credentials, and permissions in the shell scripts. Even after fixing credential errors, there were many self-wrought errors that lead to problems down the line, such as invalid syntaxes and wrong comments used. Even after rectifying these errors, we had some name mismatches that went unnoticed for a long time, as the website technically loaded. These were quickly rectified once noticed.

Listed below are our considerations and how we set up our files.

Database Possibilities

We use a RDS for our database as it is the cheaper option for basic data storage, and it comes with a wide range of useful tools for organisation and finding data.

RDS Pros and Cons

RDS is made for information that is unlikely to change much if at all, so our contact information recording program is well within the bounds. This also means that we can utilise the RDS's ability to sort and index the information quickly to allow for massive scalability, but due to its strict regulations of data that is rarely changed, it makes for a poor database to choose when considering repurposing as it severely limits the changes that can be made.

Pros

- RDS can roll back an operation if something goes wrong whereas S3 needs a separate program to go back a step.
- RDS may be slightly more expensive than S3 for storage, but it comes with a wide range of sorting, searching and other useful tools which would have to be acquired separately if S3 is used instead. These wide range of tools makes it cheaper in the long run.

Cons

- RDS has limited flexibility when it comes to altering the program to needs other than text or numeral based recording software

Prices (USD)

S3

- 0.025$ per GB up to 50T.
- Retrieving and insertion costs are 0.005$ per 1000 requests.
- Free downloads up to 1GB a month, then 0.009$ per GB for next 10T.

RDS

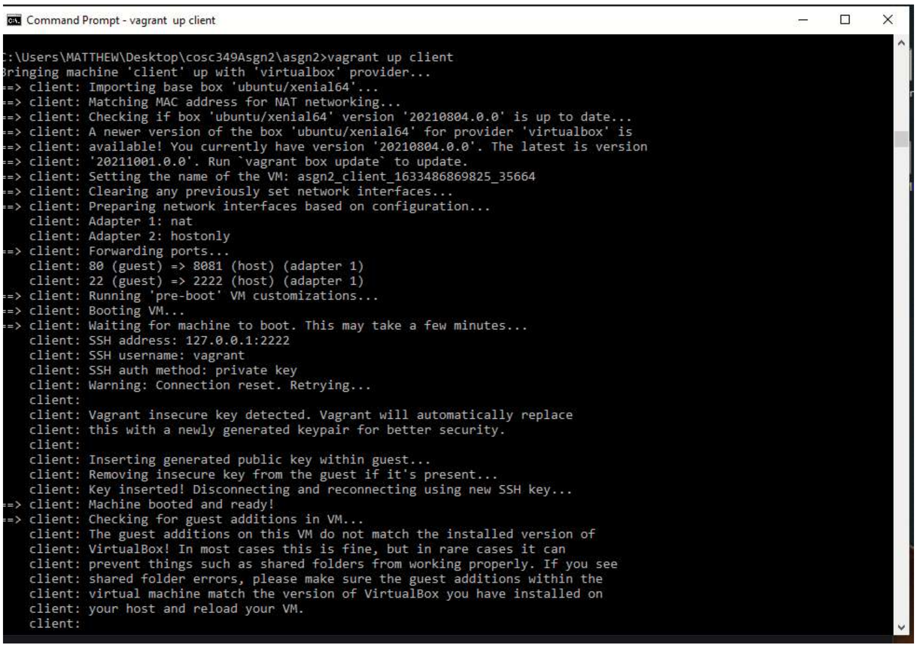
- Varies depending on engine, currently using Aurora's prices.
- 0.1$ per GB of storage per month.
- 0.2$ per million requests at s3 rates that's 5$ per million.
- Low storage usage regardless as its just set text data.

Cost estimation

- Cost of 2 EC2 VM's is $13.72 USD per month. We add this to the monthly cost of low usage RDS ($2.48 per month) and the low estimate running cost for idle is $16.20 USD per month.
- During light usage (40% or less) the cost estimation is around $30.57 USD a month.
- Cost estimation was calculated using Amazon's calculator.
  https://calculator.aws/#/createCalculator/EC2

Setup

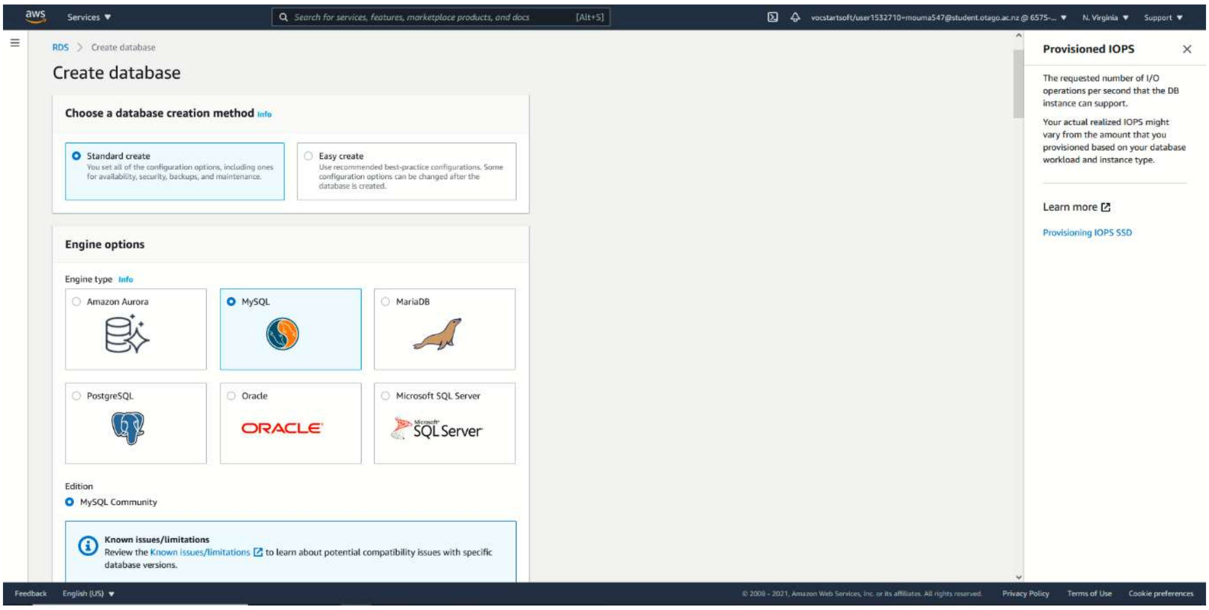
Running Vagrant up



```
:\Users\MATTHEW\Desktop\cosc349Asgn2\asgn2>vagrant up client
Bringing machine 'client' up with 'virtualbox' provider...
==> client: Importing base box 'ubuntu/xenial64'...
==> client: Matching MAC address for NAT networking...
==> client: Checking if box 'ubuntu/xenial64' version '20210804.0.0' is up to date...
==> client: A newer version of the box 'ubuntu/xenial64' for provider 'virtualbox' is
==> client: available! You currently have version '20210804.0.0'. The latest is version
==> client: '20211001.0.0'. Run `vagrant box update` to update.
==> client: Setting the name of the VM: asgn2_client_1633486869825_35664
==> client: Clearing any previously set network interfaces...
==> client: Preparing network interfaces based on configuration...
    client: Adapter 1: nat
    client: Adapter 2: hostonly
==> client: Forwarding ports...
    client: 80 (guest) => 8081 (host) (adapter 1)
    client: 22 (guest) => 2222 (host) (adapter 1)
==> client: Running 'pre-boot' VM customizations...
==> client: Booting VM...
==> client: Waiting for machine to boot. This may take a few minutes...
    client: SSH address: 127.0.0.1:2222
    client: SSH username: vagrant
    client: SSH auth method: private key
    client: Warning: Connection reset. Retrying...
    client:
    client: Vagrant insecure key detected. Vagrant will automatically replace
    client: this with a newly generated keypair for better security.
    client:
    client: Inserting generated public key within guest...
    client: Removing insecure key from the guest if it's present...
    client: Key inserted! Disconnecting and reconnecting using new SSH key...
==> client: Machine booted and ready!
==> client: Checking for guest additions in VM...
    client: The guest additions on this VM do not match the installed version of
    client: VirtualBox! In most cases this is fine, but in rare cases it can
    client: prevent things such as shared folders from working properly. If you see
    client: shared folder errors, please make sure the guest additions within the
    client: virtual machine match the version of VirtualBox you have installed on
    client: your host and reload your VM.
    client:
```

Setting up the RDS

Using MySQL as it has the most compatibility with the original database setup.



RDS > Create database

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

○ Amazon Aurora
● MySQL
○ MariaDB
○ PostgreSQL
○ Oracle
○ Microsoft SQL Server

Edition
● MySQL Community

Known issues/limitations
Review the Known issues/limitations to learn about potential compatibility issues with specific database versions.

Provisioned IOPS

The requested number of I/O operations per second that the DB instance can support.

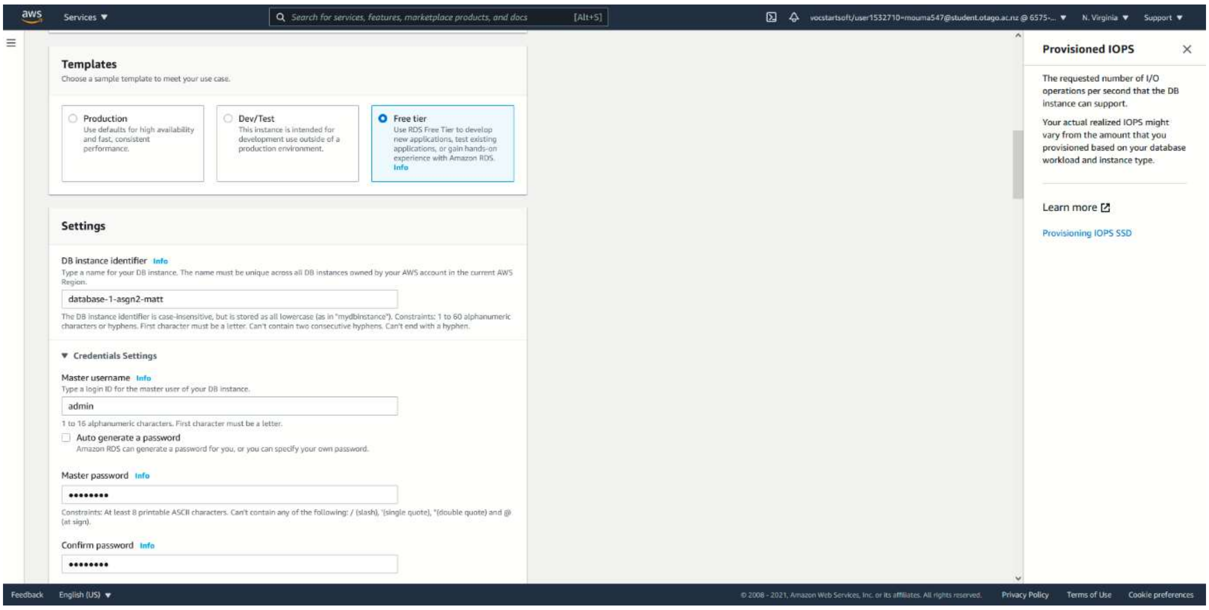
Your actual realized IOPS might vary from the amount that you provisioned based on your database workload and instance type.

Learn more

Provisioning IOPS SSD

aws
Services ▼
Search for services, features, marketplace products, and docs [Alt+S]
vocstartsoft/user1532710=mouma547@student.otago.ac.nz @ 6575-... ▼
N. Virginia ▼
Support ▼
Feedback
English (US) ▼
© 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use
Cookie preferences

Choosing the free tier as we want to use as little money as possible for presenting our product online.



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

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.

database-1-asgn2-matt

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 60 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

▼ Credentials Settings

Master username Info
Type a login ID for the master user of your DB instance.

admin

1 to 16 alphanumeric characters. First character must be a letter.

□ Auto generate a password
Amazon RDS can generate a password for you, or you can specify your own password.

Master password Info

••••••••

Constraints: At least 8 printable ASCII characters. Can't contain any of the following: / (slash), '(single quote), "(double quote) and @ (at sign).

Confirm password Info

••••••••

Provisioned IOPS

The requested number of I/O operations per second that the DB instance can support.

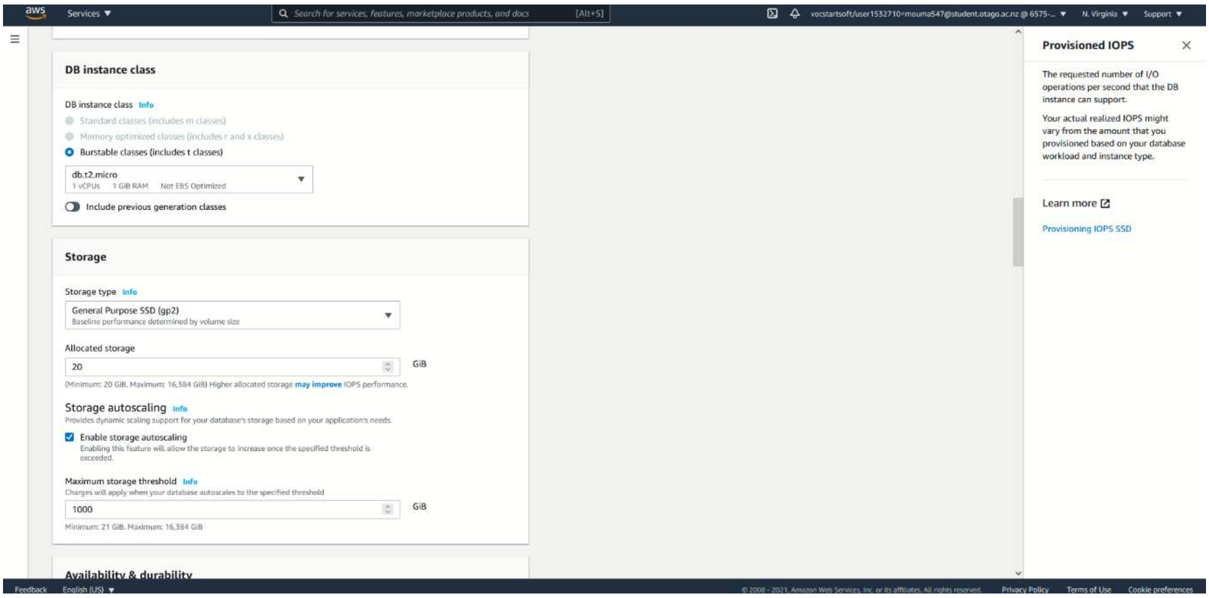
Your actual realized IOPS might vary from the amount that you provisioned based on your database workload and instance type.

Learn more

Provisioning IOPS SSD

aws
Services ▼
Search for services, features, marketplace products, and docs [Alt+S]
vocstartsoft/user1532710=mouma547@student.otago.ac.nz @ 6575-... ▼
N. Virginia ▼
Support ▼
Feedback
English (US) ▼
© 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use
Cookie preferences

We are using a “db.t2.micro”. For our current usage expectations, it is the most efficient cost to capacity ratio that we can attain. Bigger companies will need to go for amazon’s larger scale servers which unfortunately cost more.



DB instance class

DB instance class Info

○ Standard classes (includes m classes)

○ Memory optimized classes (includes r and x classes)

● Burstable classes (includes t classes)

db.t2.micro
1 vCPUs 1 GiB RAM Not EBS Optimized

Include previous generation classes

Storage

Storage type Info

General Purpose SSD (gp2)
Baseline performance determined by volume size

Allocated storage

20 GiB

(Minimum: 20 GiB. Maximum: 16,384 GiB) Higher allocated storage **may improve** IOPS performance.

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 once the specified threshold is exceeded.

Maximum storage threshold Info

Charges will apply when your database autoscales to the specified threshold

1000 GiB

Minimum: 21 GiB. Maximum: 16,384 GiB

Availability & durability

Provisioned IOPS

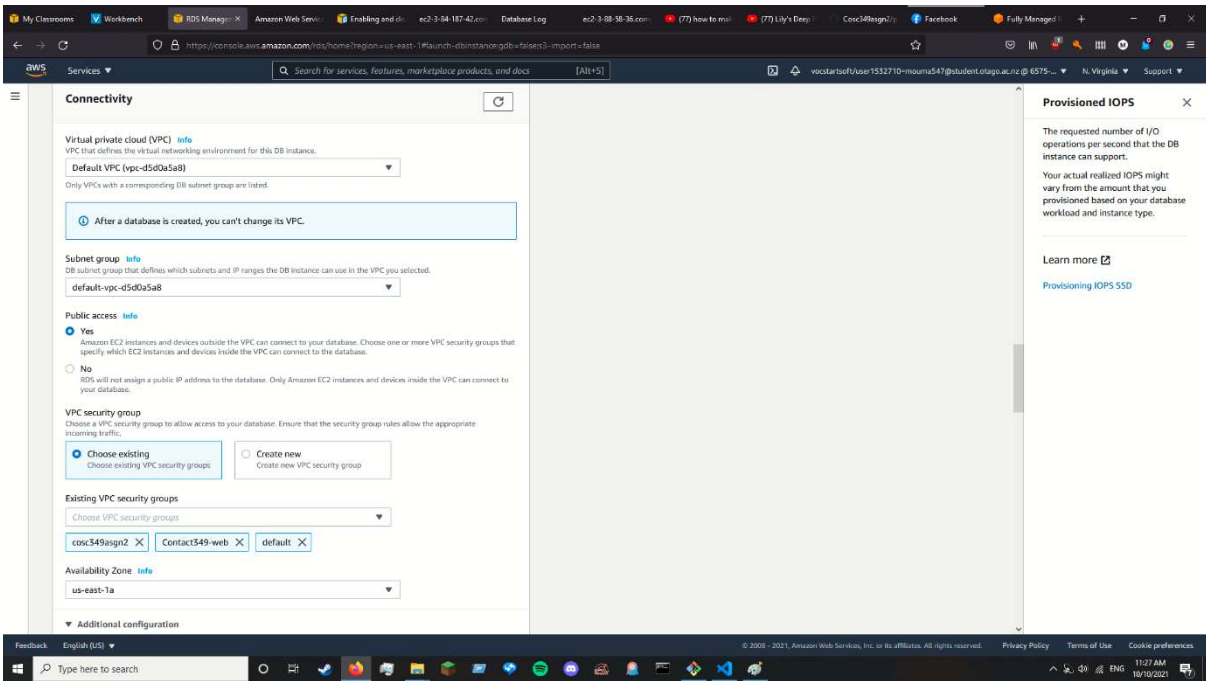
The requested number of I/O operations per second that the DB instance can support.

Your actual realized IOPS might vary from the amount that you provisioned based on your database workload and instance type.

Learn more

Provisioning IOPS SSD

We have selected the 3 security groups to allow access from the webpages to the database and set the region to the same one we used for the EC2 instances.



Connectivity

Virtual private cloud (VPC) Info

VPC that defines the virtual networking environment for this DB instance.

Default VPC (vpc-d5d0a5a8)

Only VPCs with a corresponding DB subnet group are listed.

After a database is created, you can't change its VPC.

Subnet group Info

DB subnet group that defines which subnets and IP ranges the DB instance can use in the VPC you selected.

default-vpc-d5d0a5a8

Public access Info

● Yes
Amazon EC2 instances and devices outside the VPC can connect to your database. Choose one or more VPC security groups that specify which EC2 instances and devices inside the VPC can connect to the database.

○ No
RDS will not assign a public IP address to the database. Only Amazon EC2 instances and devices inside the VPC can connect to your database.

VPC security group

Choose a VPC security group to allow access to your database. Ensure that the security group rules allow the appropriate incoming traffic.

● Choose existing
Choose existing VPC security groups

○ Create new
Create new VPC security group

Existing VPC security groups

cosc349asgn2 ✕ Contact349-web ✕ default ✕

Availability Zone Info

us-east-1a

▼ Additional configuration

Provisioned IOPS

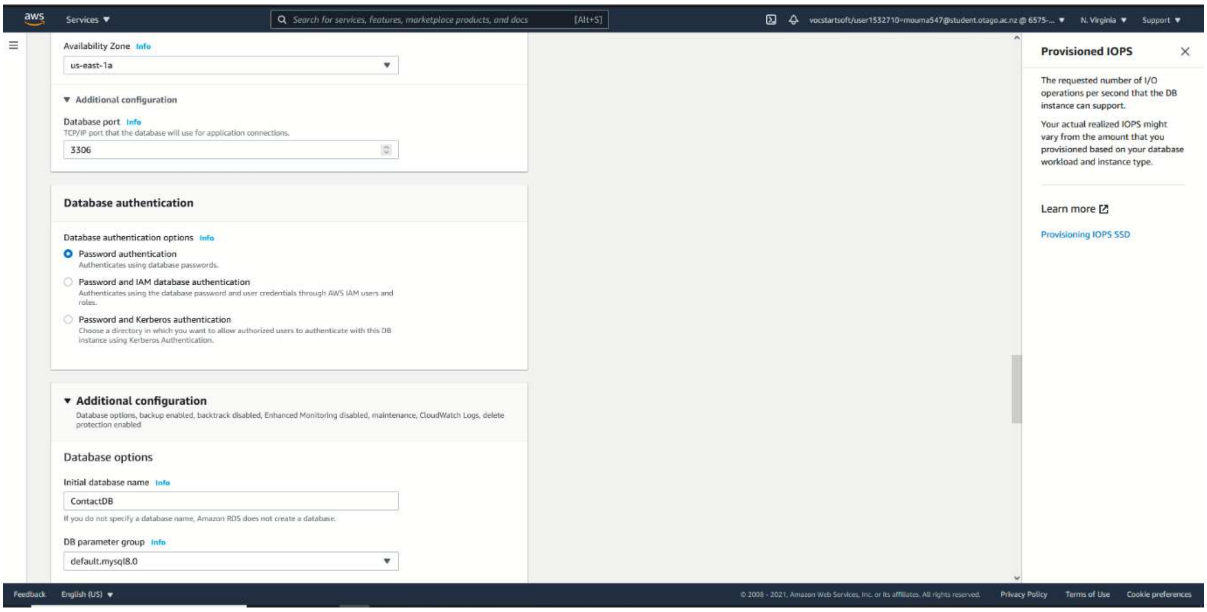
The requested number of I/O operations per second that the DB instance can support.

Your actual realized IOPS might vary from the amount that you provisioned based on your database workload and instance type.

Learn more

Provisioning IOPS SSD

Only having password authentication as this service does not require a high degree of security. Thus, we prioritise ease of access.


aws
Services ▼
Search for services, features, marketplace products, and docs [Alt+S]
vocstartsoft/user1532710=mouma547@student.otago.ac.nz @ 6575-... ▼
N. Virginia ▼
Support ▼
Feedback
English (US) ▼
© 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use
Cookie preferences

Availability Zone Info

us-east-1a ▼

▼ Additional configuration

Database port Info
TCP/IP port that the database will use for application connections.

3306

Database authentication

Database authentication options Info

● Password authentication
Authenticates using database passwords.

○ Password and IAM database authentication
Authenticates using the database password and user credentials through AWS IAM users and roles.

○ Password and Kerberos authentication
Choose a directory in which you want to allow authorized users to authenticate with this DB instance using Kerberos Authentication.

▼ **Additional configuration**
Database options, backup enabled, backtrack disabled, Enhanced Monitoring disabled, maintenance, CloudWatch Logs, delete protection enabled

Database options

Initial database name Info

ContactDB

If you do not specify a database name, Amazon RDS does not create a database.

DB parameter group Info

default.mysql8.0 ▼

Provisioned IOPS ✕

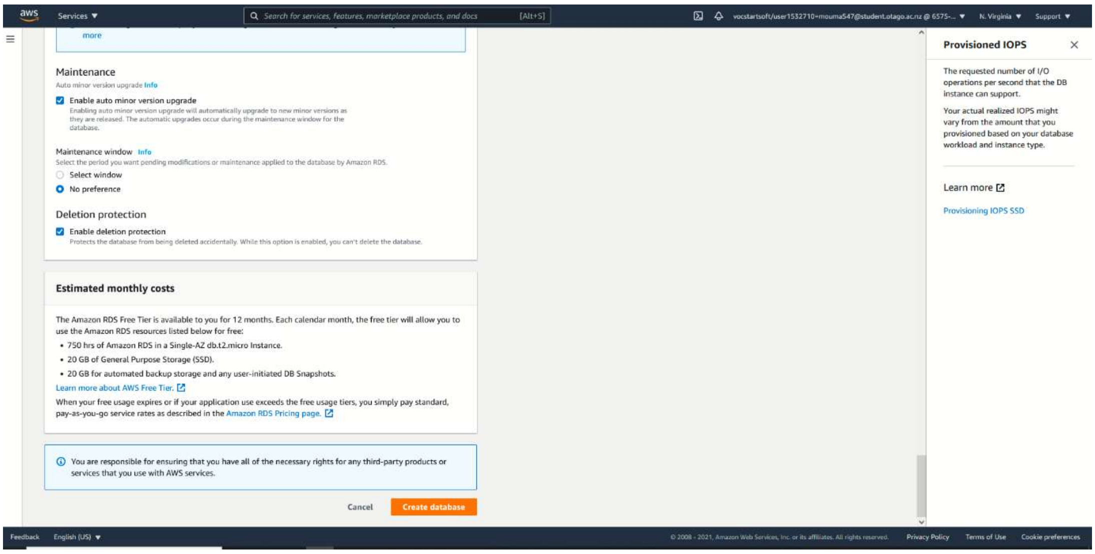
The requested number of I/O operations per second that the DB instance can support.

Your actual realized IOPS might vary from the amount that you provisioned based on your database workload and instance type.

Learn more

Provisioning IOPS SSD

Enabling deletion protection to prevent an intern or new employee destroying the database and leaving minor version upgrades on, ensuring long term stability. Both are important for a long-term database.


aws
Services ▼
Search for services, features, marketplace products, and docs [Alt+S]
vocstartsoft/user1532710=mouma547@student.otago.ac.nz @ 6575-... ▼
N. Virginia ▼
Support ▼
Cancel
Create database
Feedback
English (US) ▼
© 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use
Cookie preferences

more

Maintenance
Auto minor version upgrade Info

☑ Enable auto minor version upgrade
Enabling auto minor version upgrade will automatically upgrade to new minor versions as they are released. The automatic upgrades occur during the maintenance window for the database.

Maintenance window Info
Select the period you want pending modifications or maintenance applied to the database by Amazon RDS.

○ Select window

● No preference

Deletion protection

☑ Enable deletion protection
Protects the database from being deleted accidentally. While this option is enabled, you can't delete the database.

Estimated monthly costs

The Amazon RDS Free Tier is available to you for 12 months. Each calendar month, the free tier will allow you to use the Amazon RDS resources listed below for free:

- 750 hrs of Amazon RDS in a Single-AZ db.t2.micro Instance.
- 20 GB of General Purpose Storage (SSD).
- 20 GB for automated backup storage and any user-initiated DB Snapshots.

Learn more about AWS Free Tier.

When your free usage expires or if your application use exceeds the free usage tiers, you simply pay standard, pay-as-you-go service rates as described in the Amazon RDS Pricing page.

ⓘ You are responsible for ensuring that you have all of the necessary rights for any third-party products or services that you use with AWS services.

Provisioned IOPS ✕

The requested number of I/O operations per second that the DB instance can support.

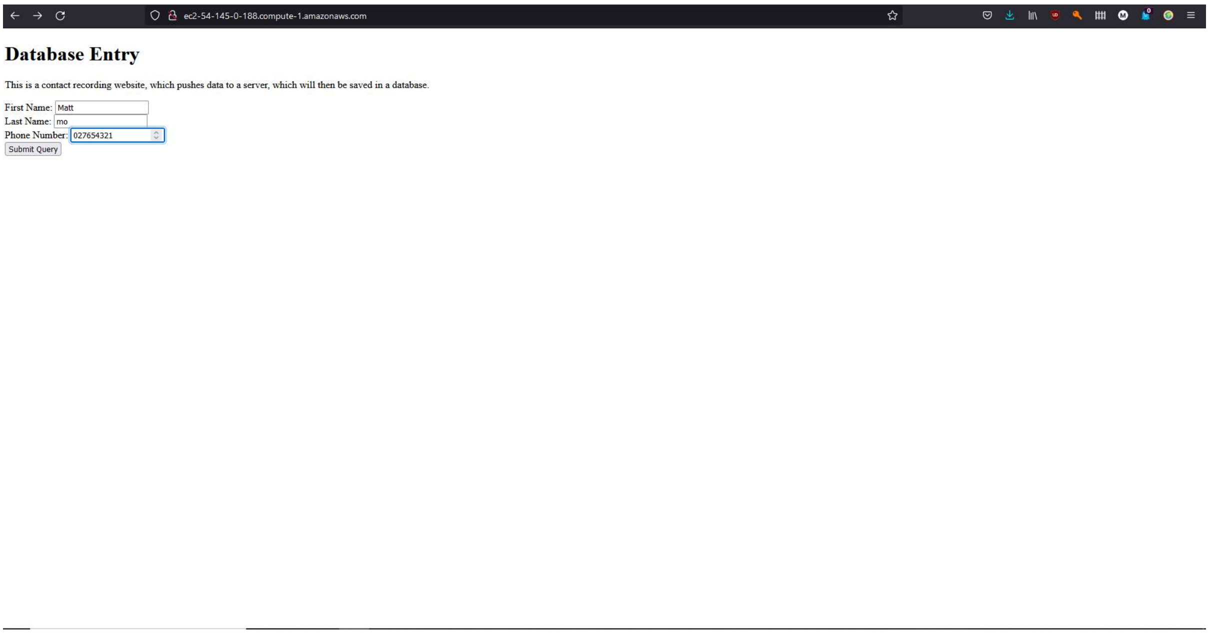
Your actual realized IOPS might vary from the amount that you provisioned based on your database workload and instance type.

Learn more

Provisioning IOPS SSD

How to use our service

First input the contact info you want to record at http://ec2-3-88-224-165.compute-1.amazonaws.com/, then click submit query.



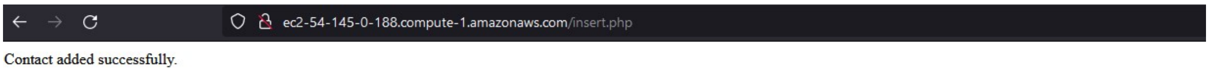
ec2-54-145-0-188.compute-1.amazonaws.com

Database Entry

This is a contact recording website, which pushes data to a server, which will then be saved in a database.

First Name: Matt
Last Name: mo
Phone Number: 027654321

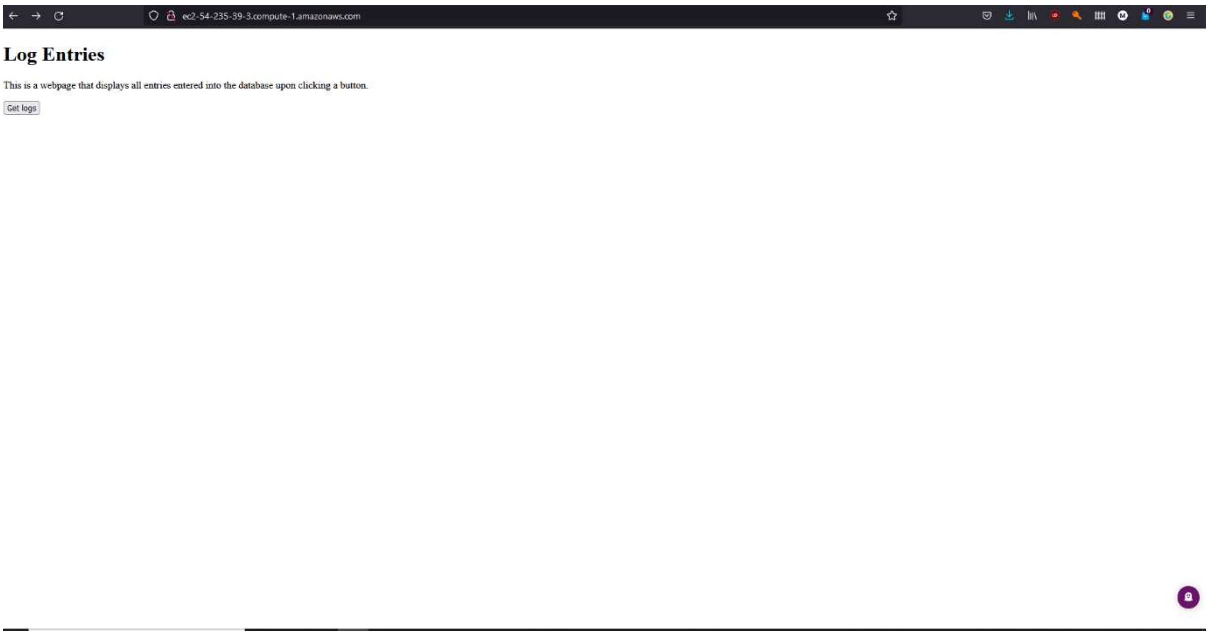
If your data has been added, you will see the below screen.



ec2-54-145-0-188.compute-1.amazonaws.com/insert.php

Contact added successfully.

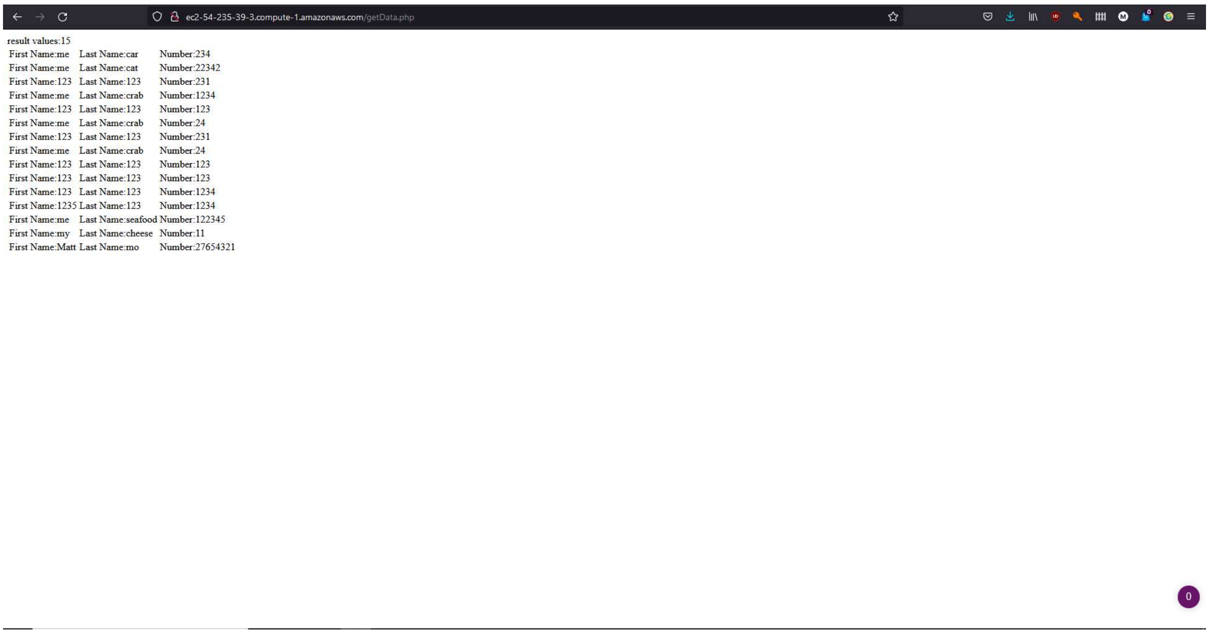
Once you have some data, you can visit the next webpage at http://ec2-18-209-11-239.compute-1.amazonaws.com/. When you have finished your data inputs, click Get Logs.



Log Entries

This is a webpage that displays all entries entered into the database upon clicking a button.

It will display the data in a neat table.



```
result values:15
 First Name:me    Last Name:car      Number:234
 First Name:me    Last Name:cat      Number:22342
 First Name:123   Last Name:123      Number:231
 First Name:me    Last Name:crab     Number:1234
 First Name:123   Last Name:123      Number:123
 First Name:me    Last Name:crab     Number:24
 First Name:123   Last Name:123      Number:231
 First Name:me    Last Name:crab     Number:24
 First Name:123   Last Name:123      Number:123
 First Name:123   Last Name:123      Number:123
 First Name:123   Last Name:123      Number:1234
 First Name:1235 Last Name:123      Number:1234
 First Name:me    Last Name:seafood Number:122345
 First Name:my    Last Name:cheese  Number:11
 First Name:Matt Last Name:mo       Number:27654321
```