**AWS Service Manager (ASM) APP User Manual**

**Introduction**

**Description**

The ASM command-line app facilitates the efficient management of AWS cloud services. Upon successful authentication, users can interact with AWS services and infrastructure through a user-friendly menu system.

**NOTE: I am not on AWS Free Tier anymore! But I don’t mind you running the test on my account.**

**File Structure**

* **root Folder - "Solution":** Contains the entire app structure.
* **core\_app folder**: Houses essential modules for the app's functionality.
* **asm\_main.py file:** The entry point for running the application.
* **data Folder:** Stores data to be uploaded to or retrieved from AWS.
* **password.txt file**: It contains the username and password used to login into the system and it resides inside the **core\_app** folder.

**Dependencies**

* **boto3:** The sole package required to operate this app efficiently.

**ASM Application Usage**

The ASM application features a menu-driven system designed to manage five key AWS services: **EC2, EBS, S3, CloudWatch, and RDS**. Access to these services requires successful authentication using credentials stored in the **'password.txt'** file.

Below, I will provide instructions on authenticating within the ASM app and showcasing the outcomes of the methods I've implemented for the mentioned AWS services.

**Login Process**

To log in and utilize the ASM app's functionalities, follow these steps:

1. Place your **username, password, aws\_accesskey, and aws\_secretkey in the 'password.txt' file** located within the 'core\_app' folder.
2. Access the terminal and navigate to the 'core\_app' directory. Execute **'python asm\_main.py**' to launch the application.
3. Upon launching, the app prompts for user details, including whether you're a returning user, username, password, and AWS region. Provide accurate information to proceed.
4. Upon successful login, you are provided access to the AWS service menu, which allows interaction and management of various AWS services as shown in the below image.

A screenshot of a computer

Description automatically generated

**EC2 Menu**

This section outlines the methods accessible within the EC2 menu, coupled with the anticipated outcomes that demonstrate expected behavior upon successful execution of each method.

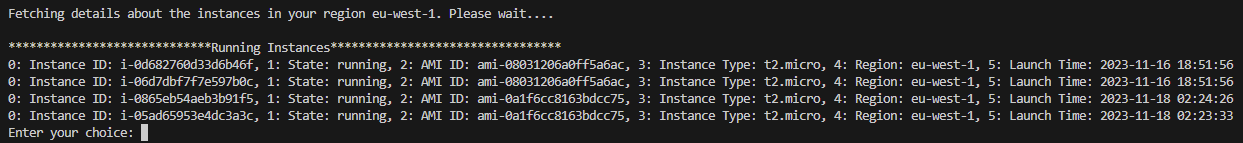
* **List instances:** This function displays all the EC2 instances within either a user’s AWS account or in the logged in region.

1. Display of all the instances within the AWS account

A screen shot of a computer

Description automatically generated

b. Display of the instances within a user logged in region which is eu-west-1.



* **Start instance:** This function starts up an instance specified by the user. When you execute this method, you will be provided with a lisof running and stopped instances within your region. When you select any of the stopped instances, it will start up that instance. A computer screen shot of a black screen

  Description automatically generated
* **Stop instance**: This function stops an instance specified by the user. Upon execution, it displays a list comprising of running and stopped

Instances within your AWS region. When an instance in a running state is chosen, it stops that instance.

A computer screen with white text

Description automatically generated

* **Launch instance:** This function launches an EC2 instance based on the type of ami selected by the user. The two available options are: Linux and Windows. Note that only a free tier AMI is launched and in eu-west-1 region

A screen shot of a computer

Description automatically generated

* **Terminate instance:** This method terminates an EC2 instance chosen by the user. When you execute this method, you will be provided with a list of running and stopped instances within your region. When you select any of the running or stopped instances, they will be terminated.

A computer screen with white text

Description automatically generated

**EBS Menu**

This section outlines the methods accessible within the EBS menu, coupled with the anticipated outcomes that demonstrate expected behavior upon successful execution of each method.

* **List volume:** It lists all the EBS within the user’s AWS account as shown below.

A screen shot of a computer

Description automatically generated

* **Create volume:** It creates an EBS volume based on the parameter provided by the user. Three parameters that must be supplied by the user are: Availability Zone, volume size and volume type.   
  A computer screen with white text

  Description automatically generated
* **Attach volume:** It attaches an existing volume to an EC2 instance specified by the user. Upon executing the code, you are presented with a list of available volumes within the AZ where the EC2 instance you specified is located. If there is none, you will be provided with an option of creating one that then gets attached to the EC2 instance.   
  **Note:** If you intend to attach the volume as the root volume, choose "Yes" when prompted "Do you want to attach a root volume?". Otherwise, choose "No"

A black screen with white text

Description automatically generated

* **Detach volume:** It detaches an EBS volume from an EC2 instance specified by the user. Upon executing this function, you will be prompted for the instance ID that its volume needs to be detached. Also, a list of all volumes attached to the chosen instance will be presented. You need to provide the Volume ID of one of those volumes when prompted for Volume ID.

A computer screen shot of a black background

Description automatically generated

* **Modify volume:** It modifies the size of an EBS volume. If the target size is larger than the current EBS volume size, the volume is automatically increased to the target size. However, if the target size is smaller, it will first generate a snapshot and then create a new volume from that snapshot with the target size.

A screenshot of a computer program

Description automatically generated

* **List snapshot:** It lists all the snapshot in the user logged in region.

A black screen with white text

Description automatically generated

* **Take snapshots:** It takes a snapshot of the volume provided by the user.

A screen shot of a computer

Description automatically generated

* **Create volume from snapshot:** It creates a volume from the snapshot ID provided by the user. The user will be required to enter the parameters of the new volume that will be created from the snapshot.

A screenshot of a computer error

Description automatically generated

**S3 Menu**

This section outlines the methods accessible within the S3 menu, coupled with the anticipated outcomes that demonstrate expected behavior upon successful execution of each method.

* **Lists buckets:** It lists all the S3 buckets present in the users AWS account.

A screenshot of a computer program

Description automatically generated

* A screenshot of a computer program

  Description automatically generated**List bucket objects:** It lists all the objects within an S3 bucket. The user will specify the name of the bucket that its object should be listed.
* **Upload object:** It uploads data to an S3 bucket specified by the user. For uploading data to an S3 bucket, place it within the "uploads" subfolder under the data directory.

A computer screen shot of a computer

Description automatically generated

* **Download object:** It downloads an object from s3 bucket and stores it within the ‘download’ subfolder under the data directory. When executed, this command fetches and displays a list of all available objects within the specified bucket. The user is prompted to select one of the displayed objects for download.

A computer screen shot of a building

Description automatically generated

* **Delete bucket:** It deletes a bucket specified by the user**.**

A black screen with white text

Description automatically generated

**CloudWatch Menu**

This section outlines the methods accessible within the CloudWatch menu, coupled with the anticipated outcomes that demonstrate expected behavior upon successful execution of each method.

* **Display metric value:** It displays the DisckReadOps, CPUCreditUsage and DiskWriteBytes metric of an EC2 instance specified by the user.

A screenshot of a computer

Description automatically generated

* **Set Alarm:** It sets alarm for DiskWriteBytes metric. The user supplies the EC2 instance ID from which the metric value will be extracted to assess the alarm condition.A screen shot of a computer

  Description automatically generated

**RDS Menu**

This section outlines the methods accessible within the RDS menu, coupled with the anticipated result that demonstrate expected behavior upon successful execution of each method.

* **List rds instances:** It lists all the rds instance in the user’s logged in region.

A screen shot of a computer

Description automatically generated

* **Start rds instance:** It starts up an rds instance specified by the user.

A screen shot of a computer

Description automatically generated

* **Stop rds instance:** It stops an rds instance specified by the user

A screen shot of a computer

Description automatically generated

* **Delete rds instance** deletes rds instance specified by the user.

A screen shot of a computer

Description automatically generated