Project Report

Mitul Kabutarwala

The project was to parse Time log file in comforted programming language and deploy it on Free Cloud Hosting Services.

Time Log file – A text file which contains date, time and description of time taken. There are 5-time log files TimeLogCarbon.txt, TimeLogEnergy.txt, TimeLogNitrogen.txt, TimeLogWater.txt, and TimeLogWatershed.txt. Apart from these files, there is a time log file that has been created by me during this course CPL-2783593.txt.

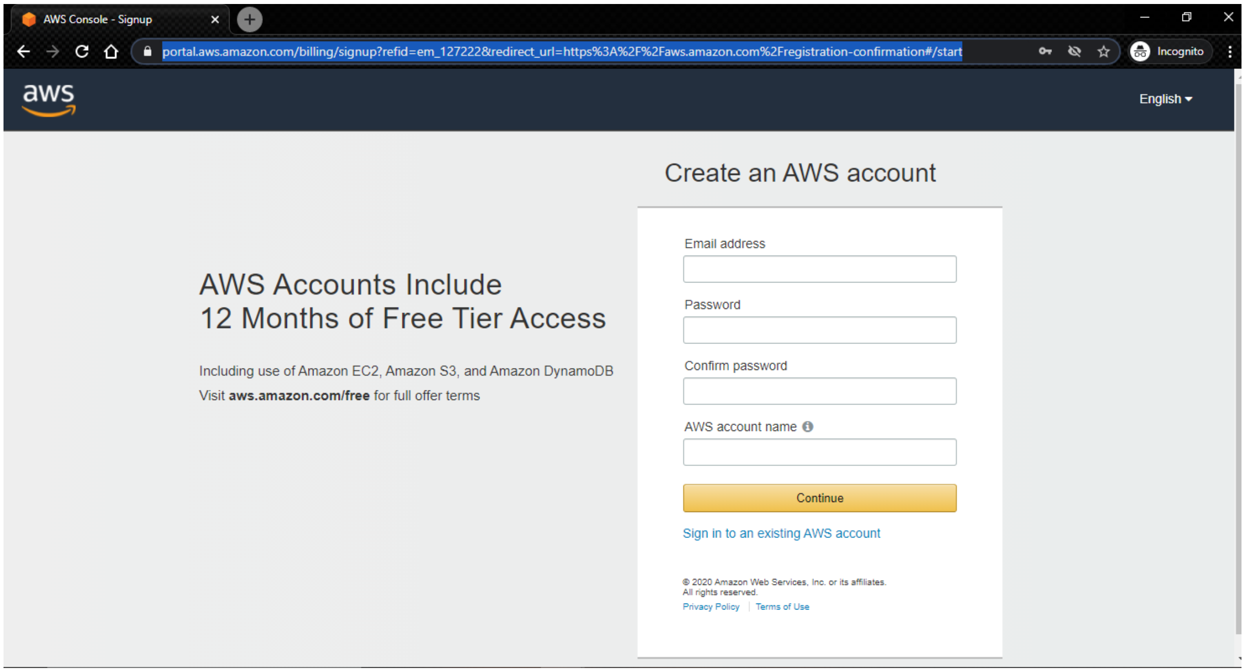
Cloud hosting services - Cloud hosting refers primarily to the use of virtual hardware, network, storage and composite solutions from a cloud vendor. It is enabled through virtualization, whereby the entire computing capacity of an infrastructure or data center is distributed and delivered to multiple users simultaneously. The user uses underlying infrastructure to host its own applications, services and data [1].

Amazon Web Services Elastic Beanstalk is cloud hosting services that has been used to host the web application using **python flask**. Flask is a web framework for python, meaning that it provides a simple interface for dynamically generating responses to web requests [2].

Install flask by terminal command

- ✓ **pip install flask** (windows 10)

Firstly, create Amazon Web Services account
https://portal.aws.amazon.com/billing/signup?refid=em_127222&redirect_url=https%3A%2F%2Faws.amazon.com%2Fregistration-confirmation#/start


AWS Console - Signup
portal.aws.amazon.com/billing/signup?refid=em_127222&redirect_url=https%3A%2F%2Faws.amazon.com%2Fregistration-confirmation#/start
Incognito
aws
English

Create an AWS account

AWS Accounts Include
12 Months of Free Tier Access

Including use of Amazon EC2, Amazon S3, and Amazon DynamoDB
Visit **aws.amazon.com/free** for full offer terms

Email address

Password

Confirm password

AWS account name

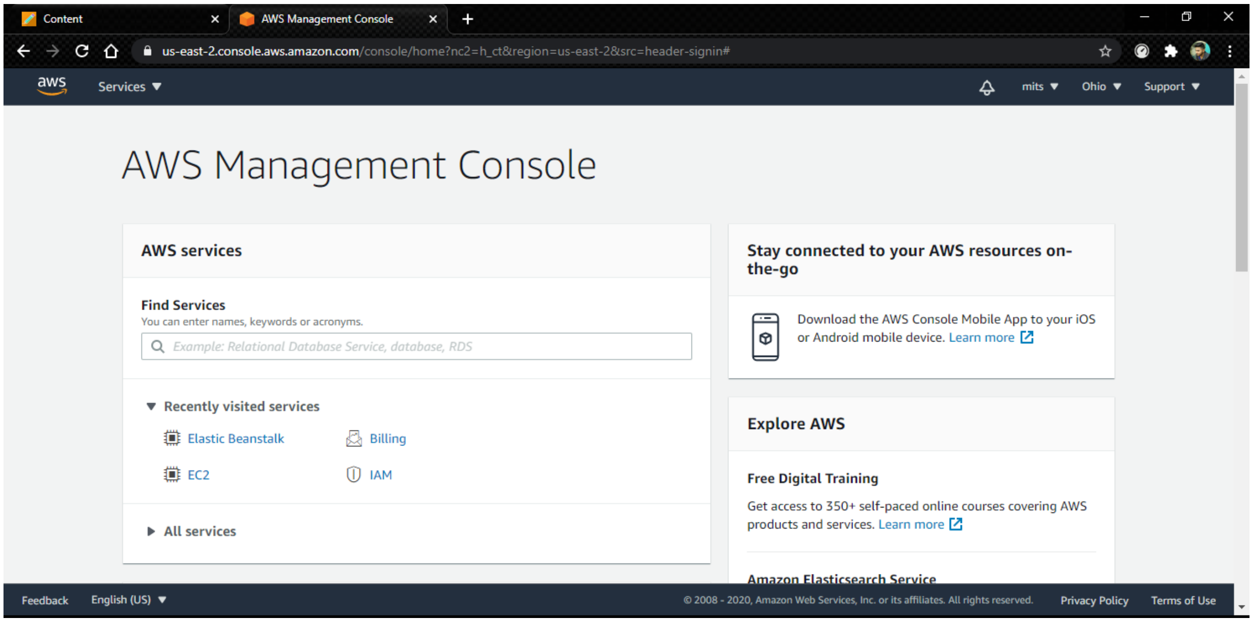
Continue

Sign in to an existing AWS account

© 2020 Amazon Web Services, Inc. or its affiliates.
All rights reserved.
Privacy Policy | Terms of Use

It does ask for credit/debit card details, but it do not charge till 1 year for any free tier access.

After creating the account, you can see the AWS console.


Content
AWS Management Console
us-east-2.console.aws.amazon.com/console/home?nc2=h_ct&region=us-east-2&src=header-signin#
aws
Services
mits
Ohio
Support

AWS Management Console

AWS services

Find Services
You can enter names, keywords or acronyms.

Example: Relational Database Service, database, RDS

Recently visited services

Elastic Beanstalk
Billing
EC2
IAM

All services

Stay connected to your AWS resources on-the-go

Download the AWS Console Mobile App to your iOS or Android mobile device. Learn more

Explore AWS

Free Digital Training

Get access to 350+ self-paced online courses covering AWS products and services. Learn more

Amazon Elasticsearch Service

Feedback English (US)
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

As signing-in in AWS we must deploy the code. So, code for parse time log file along with flask which is kind of gateway to deploy web application.

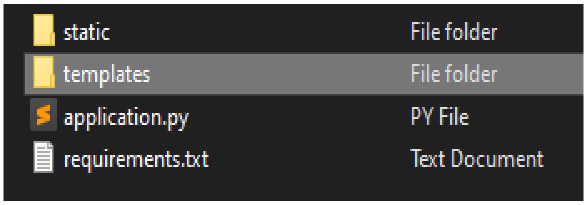
- ✓ Python and flask code:- application.py

As there is web application, write HTML and CSS code to fetch input from system to HTML

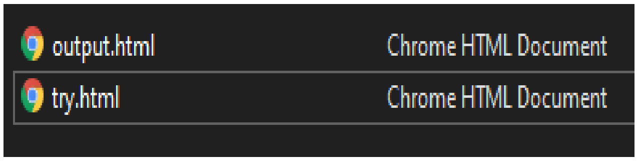
- ✓ HTML for input:- templates/try.html
- ✓ CSS for input:- static/trycss.css

This HTML code will pass the file to python so python parsing code will execute, and appropriate output should generate. This output will redirect to another HTML file that will display the result.

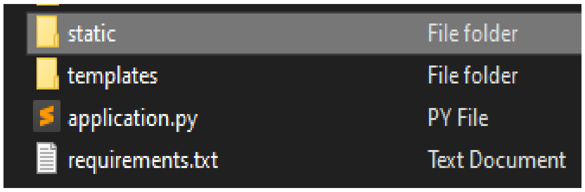
- ✓ HTML for output:- templates/output.html
- ✓ CSS for output:- static/outputcss.css



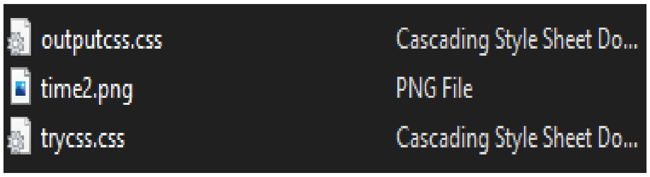
static	File folder
templates	File folder
application.py	PY File
requirements.txt	Text Document



output.html	Chrome HTML Document
try.html	Chrome HTML Document



static	File folder
templates	File folder
application.py	PY File
requirements.txt	Text Document



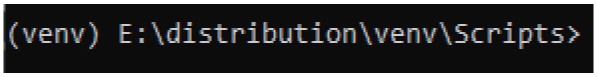
outputcss.css	Cascading Style Sheet Do...
time2.png	PNG File
trycss.css	Cascading Style Sheet Do...

Now to setup the server of aws we need to install all library of python on server. So, create an enviroment through terminal in the curent folder path or the python file(aplication.py) path with

- ✓ python -m venv venv

Above command will create an enviroment so to active type

- ✓ cd venv/Scripts
- ✓ activate



```
(venv) E:\distribution\venv\Scripts>
```

To deactivate type

- ✓ deactivate

Now to install flask type

- ✓ pip install flask
- ✓ pip install freeze

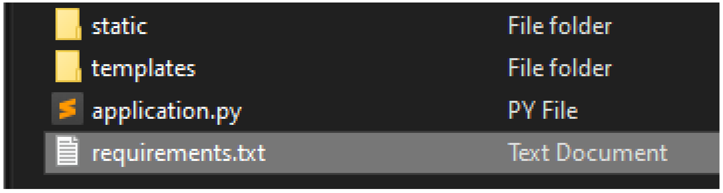
To check the library used in this environment type

- ✓ pip freeze

this command will show the output like this

```
click==7.1.2
Flask==1.1.2
itsdangerous==1.1.0
Jinja2==2.11.2
MarkupSafe==1.1.1
Werkzeug==1.0.1
```

Save this library in requirements.txt in main folder path or python file(aplication.py) path

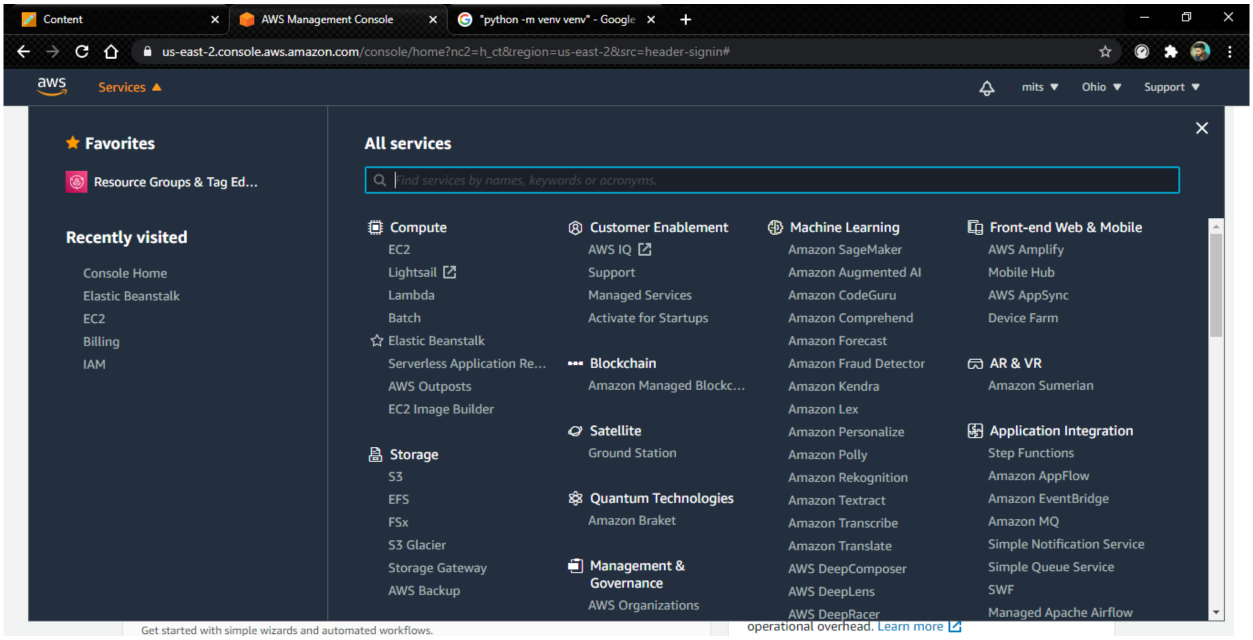


static	File folder
templates	File folder
application.py	PY File
requirements.txt	Text Document

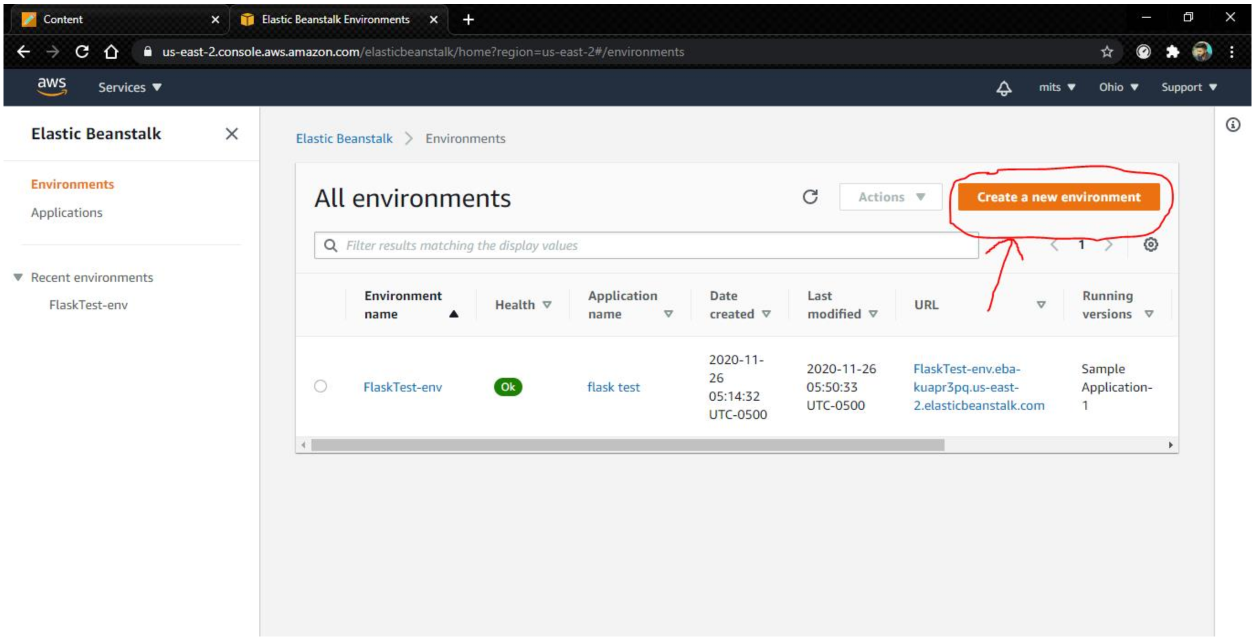
Now zip 4 file together as alpha1.zip

1. Application.py
2. static
3. templates
4. requirements.txt

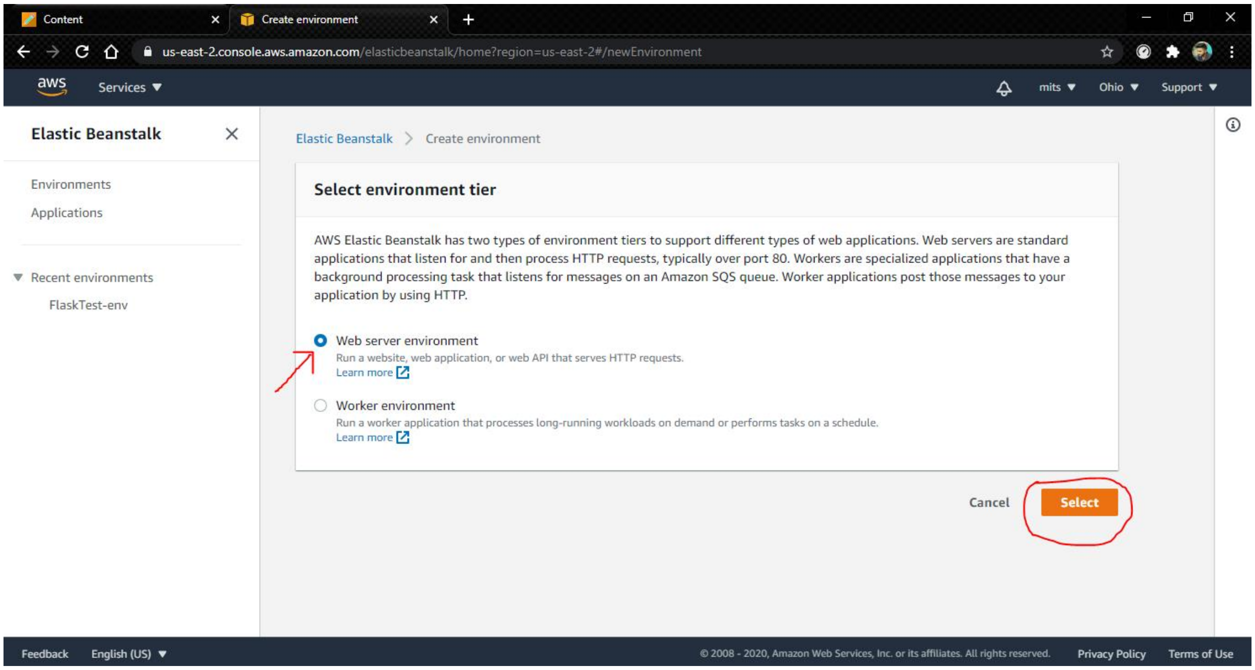
Now to host it on AWS, click on Elastic Beanstalk on your AWS console.



Create a new environment by clicking button at top right.


Content
Elastic Beanstalk Environments
us-east-2.console.aws.amazon.com/elasticbeanstalk/home?region=us-east-2#/environments
aws
Services
mits
Ohio
Support
Elastic Beanstalk
Environments
Applications
Recent environments
FlaskTest-env
Elastic Beanstalk > Environments
All environments
Actions
Create a new environment
Filter results matching the display values
1

Environment name	Health	Application name	Date created	Last modified	URL	Running versions
○ FlaskTest-env	Ok	flask test	2020-11-26 05:14:32 UTC-0500	2020-11-26 05:50:33 UTC-0500	FlaskTest-env.eba-kuapr3pq.us-east-2.elasticbeanstalk.com	Sample Application-1



Content
Create environment
us-east-2.console.aws.amazon.com/elasticbeanstalk/home?region=us-east-2#/newEnvironment
aws
Services ▼
mits ▼
Ohio ▼
Support ▼
Elastic Beanstalk
Environments
Applications
▼ Recent environments
FlaskTest-env
Elastic Beanstalk > Create environment
Feedback
English (US) ▼
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

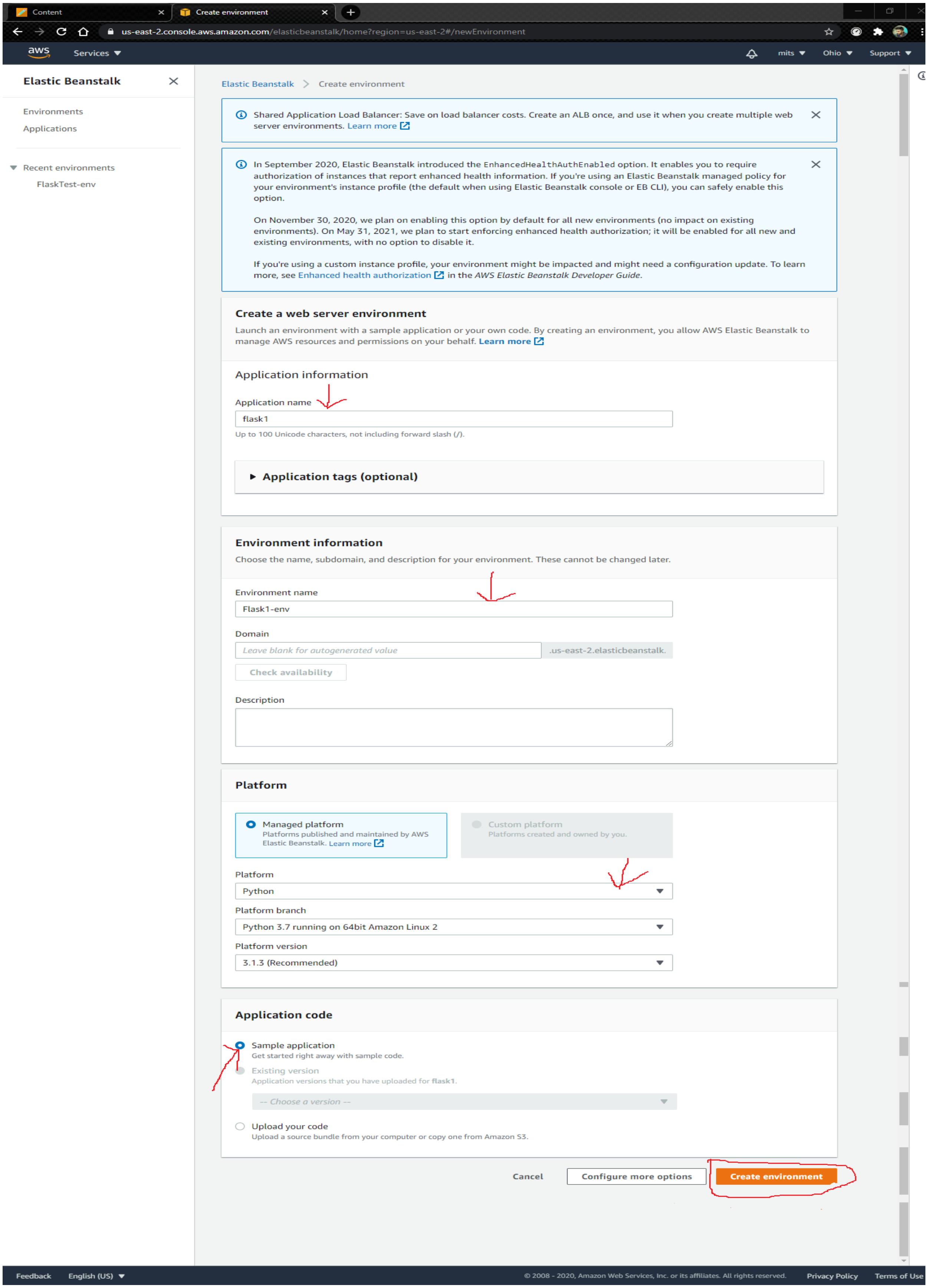
Select environment tier

AWS Elastic Beanstalk has two types of environment tiers to support different types of web applications. Web servers are standard applications that listen for and then process HTTP requests, typically over port 80. Workers are specialized applications that have a background processing task that listens for messages on an Amazon SQS queue. Worker applications post those messages to your application by using HTTP.

● Web server environment
Run a website, web application, or web API that serves HTTP requests.
Learn more

○ Worker environment
Run a worker application that processes long-running workloads on demand or performs tasks on a schedule.
Learn more

Cancel [✓ Select]



Shared Application Load Balancer: Save on load balancer costs. Create an ALB once, and use it when you create multiple web server environments. Learn more

In September 2020, Elastic Beanstalk introduced the `EnhancedHealthAuthEnabled` option. It enables you to require authorization of instances that report enhanced health information. If you're using an Elastic Beanstalk managed policy for your environment's instance profile (the default when using Elastic Beanstalk console or EB CLI), you can safely enable this option.

On November 30, 2020, we plan on enabling this option by default for all new environments (no impact on existing environments). On May 31, 2021, we plan to start enforcing enhanced health authorization; it will be enabled for all new and existing environments, with no option to disable it.

If you're using a custom instance profile, your environment might be impacted and might need a configuration update. To learn more, see Enhanced health authorization in the *AWS Elastic Beanstalk Developer Guide*.

Create a web server environment

Launch an environment with a sample application or your own code. By creating an environment, you allow AWS Elastic Beanstalk to manage AWS resources and permissions on your behalf. **Learn more**

Application information

Application name: flask1

Up to 100 Unicode characters, not including forward slash (/).

▶ **Application tags (optional)**

Environment information

Choose the name, subdomain, and description for your environment. These cannot be changed later.

Environment name: Flask1-env

Domain: *Leave blank for autogenerated value* .us-east-2.elasticbeanstalk.

Check availability

Description:

Platform

● Managed platform
Platforms published and maintained by AWS Elastic Beanstalk. Learn more

○ Custom platform
Platforms created and owned by you.

Platform: Python

Platform branch: Python 3.7 running on 64bit Amazon Linux 2

Platform version: 3.1.3 (Recommended)

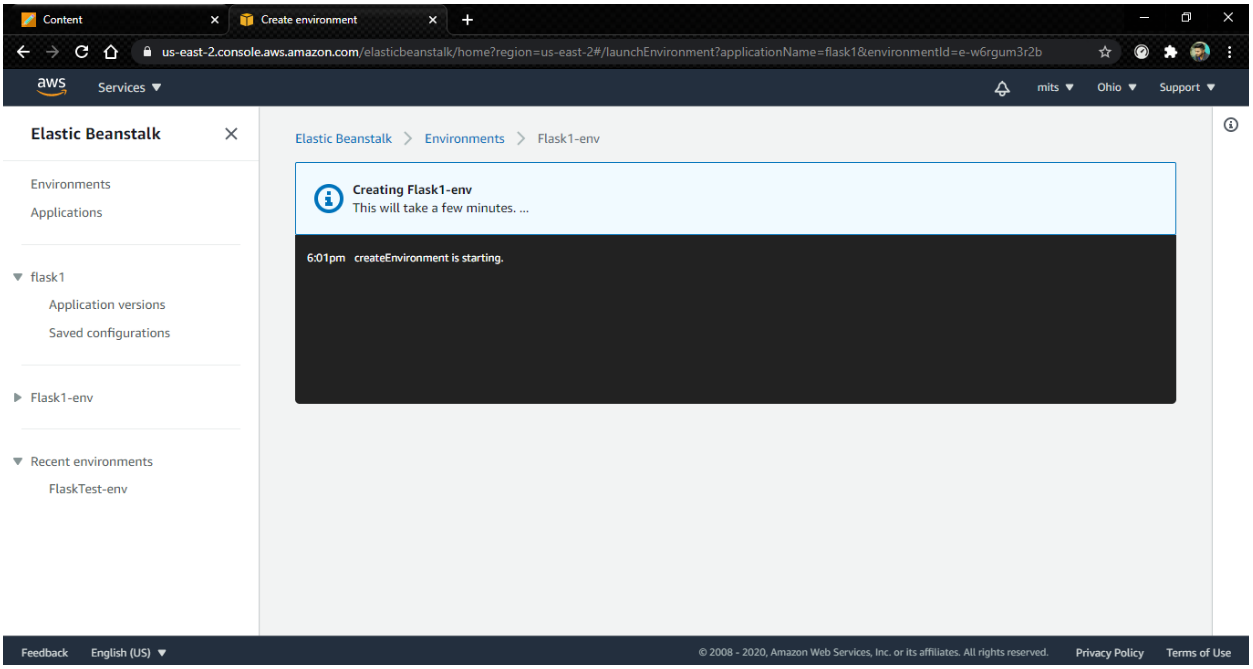
Application code

● Sample application
Get started right away with sample code.

○ Existing version
Application versions that you have uploaded for **flask1**.

-- Choose a version --

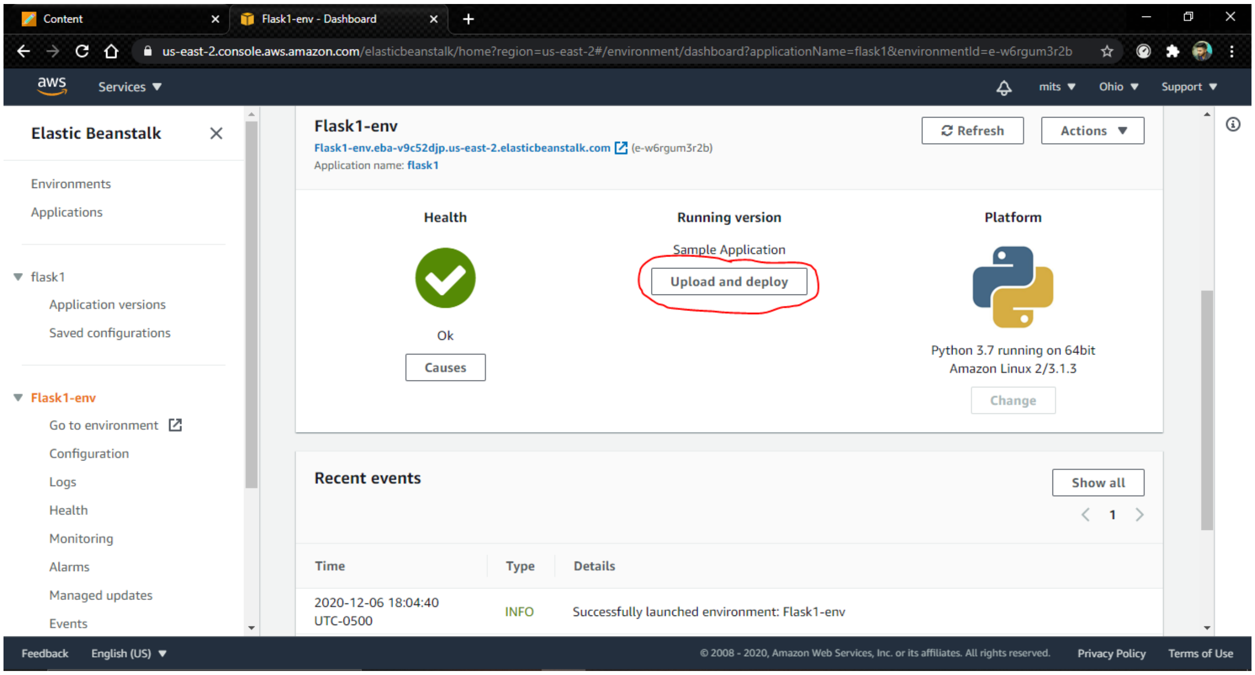
○ Upload your code
Upload a source bundle from your computer or copy one from Amazon S3.



Content
Create environment
us-east-2.console.aws.amazon.com/elasticbeanstalk/home?region=us-east-2#/launchEnvironment?applicationName=flask1&environmentId=e-w6rgum3r2b
aws
Services ▼
mits ▼
Ohio ▼
Support ▼
Elastic Beanstalk
Environments
Applications
▼ flask1
Application versions
Saved configurations
▶ Flask1-env
▼ Recent environments
FlaskTest-env
Elastic Beanstalk > Environments > Flask1-env
Feedback
English (US) ▼
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

Creating Flask1-env
This will take a few minutes. ...

```
6:01pm   createEnvironment is starting.
```



Flask1-env
Flask1-env.eba-v9c52djp.us-east-2.elasticbeanstalk.com (e-w6rgum3r2b)
Application name: flask1

Health
Ok

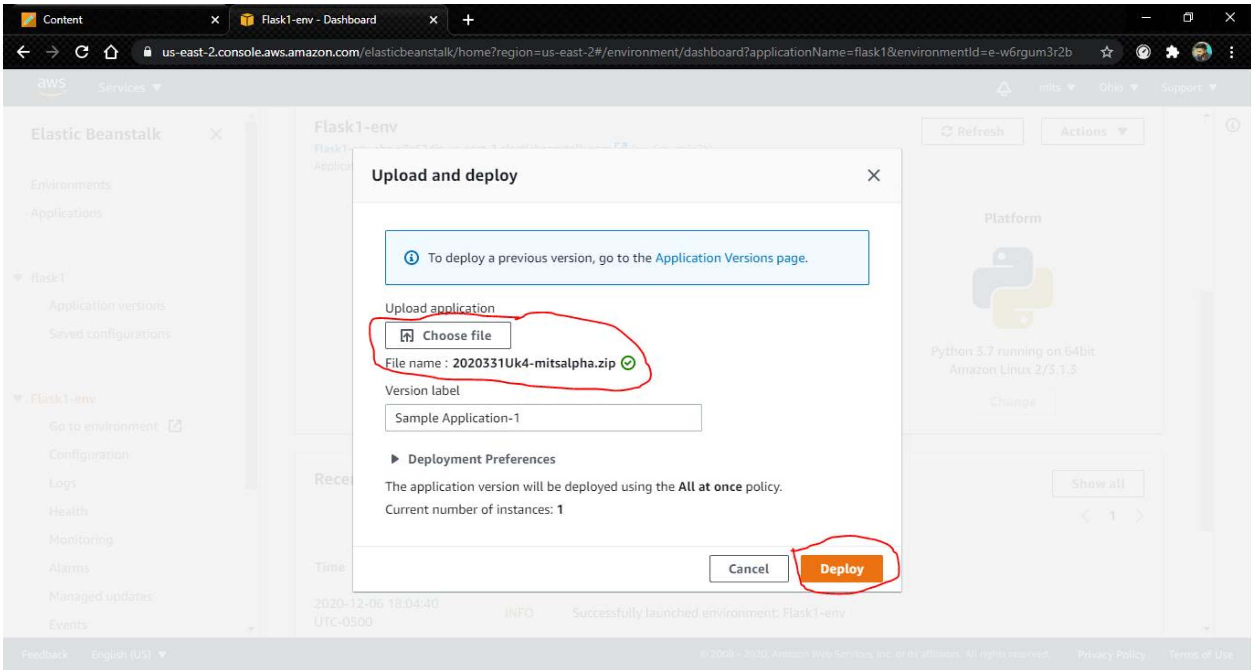
Running version
Sample Application

Platform
Python 3.7 running on 64bit Amazon Linux 2/3.1.3

Recent events

Time	Type	Details
2020-12-06 18:04:40 UTC-0500	INFO	Successfully launched environment: Flask1-env

Select the zip file alpha1.zip


Content
Flask1-env - Dashboard
us-east-2.console.aws.amazon.com/elasticbeanstalk/home?region=us-east-2#/environment/dashboard?applicationName=flask1&environmentId=e-w6rgum3r2b
Upload and deploy
Choose file
Cancel
Deploy

To deploy a previous version, go to the Application Versions page.

Upload application

File name : 2020331Uk4-mitsalpha.zip

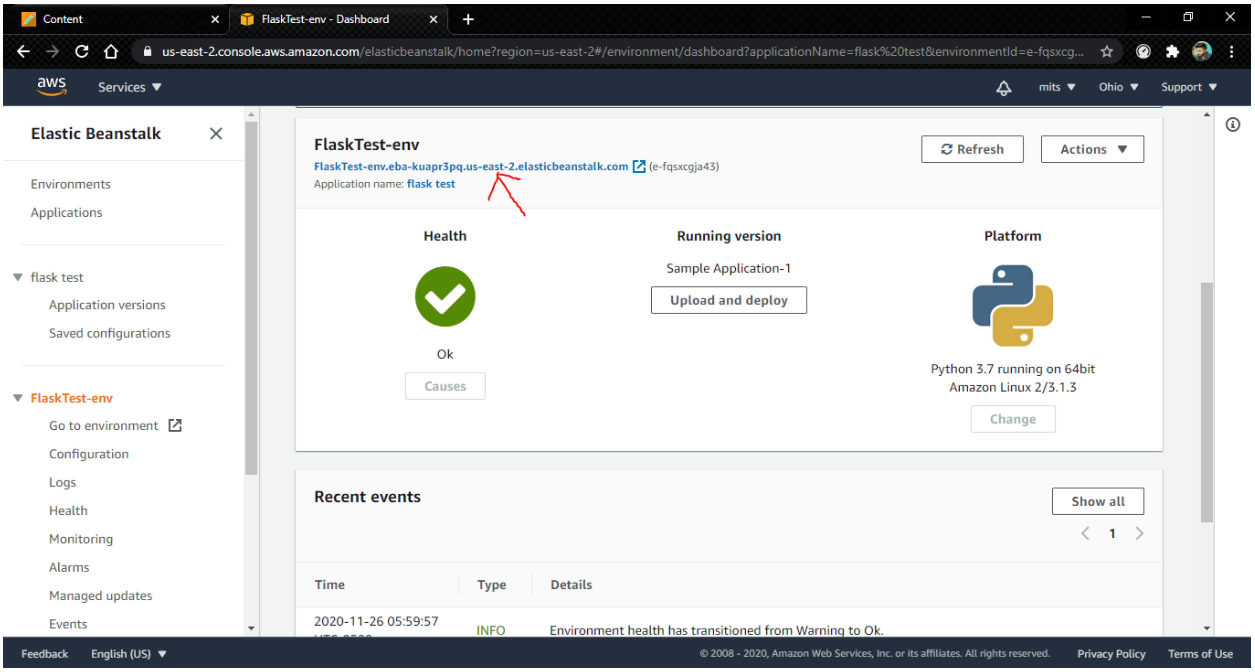
Version label: Sample Application-1

Deployment Preferences

The application version will be deployed using the **All at once** policy.

Current number of instances: **1**

After successfully deploying click the URL



FlaskTest-env

FlaskTest-env.eba-kuapr3pq.us-east-2.elasticbeanstalk.com (e-fqsxcgja43)

Application name: flask test

Health: Ok

Running version: Sample Application-1

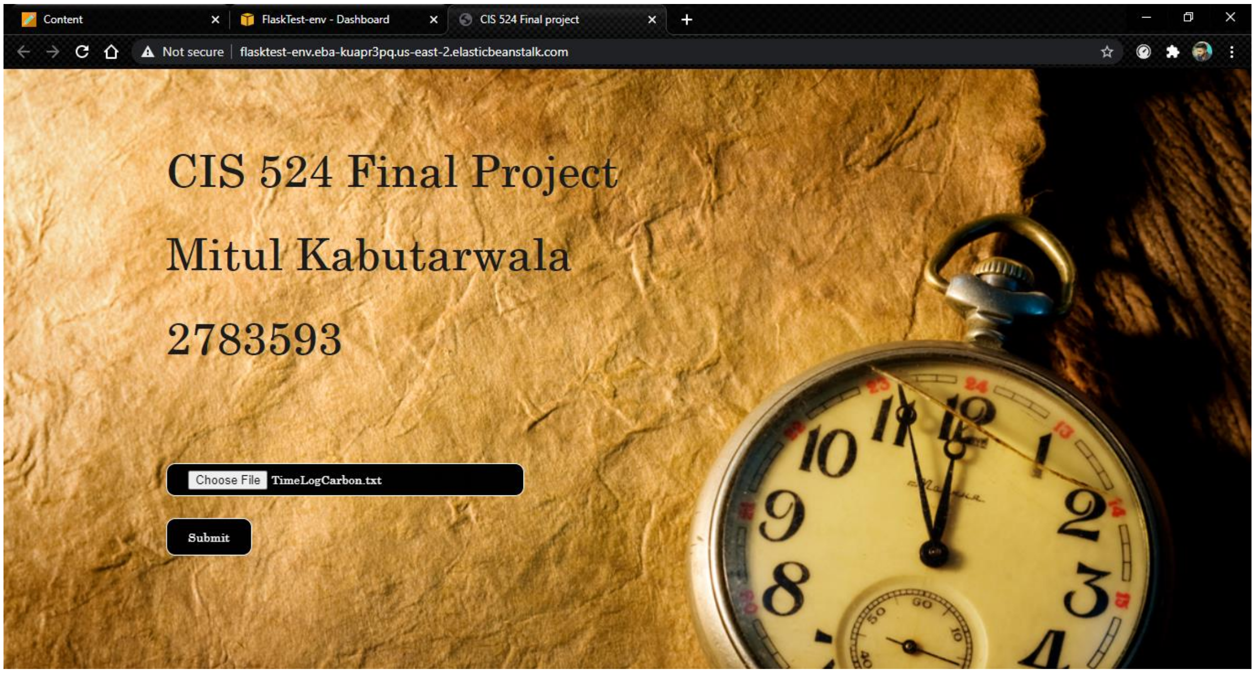
Platform: Python 3.7 running on 64bit Amazon Linux 2/3.1.3

Recent events

Time	Type	Details
2020-11-26 05:59:57	INFO	Environment health has transitioned from Warning to Ok.

Link:- http://flasktest-env.eba-kuapr3pq.us-east-2.elasticbeanstalk.com/

Select the file by clicking Choose file which must be Time Log File and then Submit



Content
FlaskTest-env - Dashboard
CIS 524 Final project
Not secure
flasktest-env.eba-kuapr3pq.us-east-2.elasticbeanstalk.com

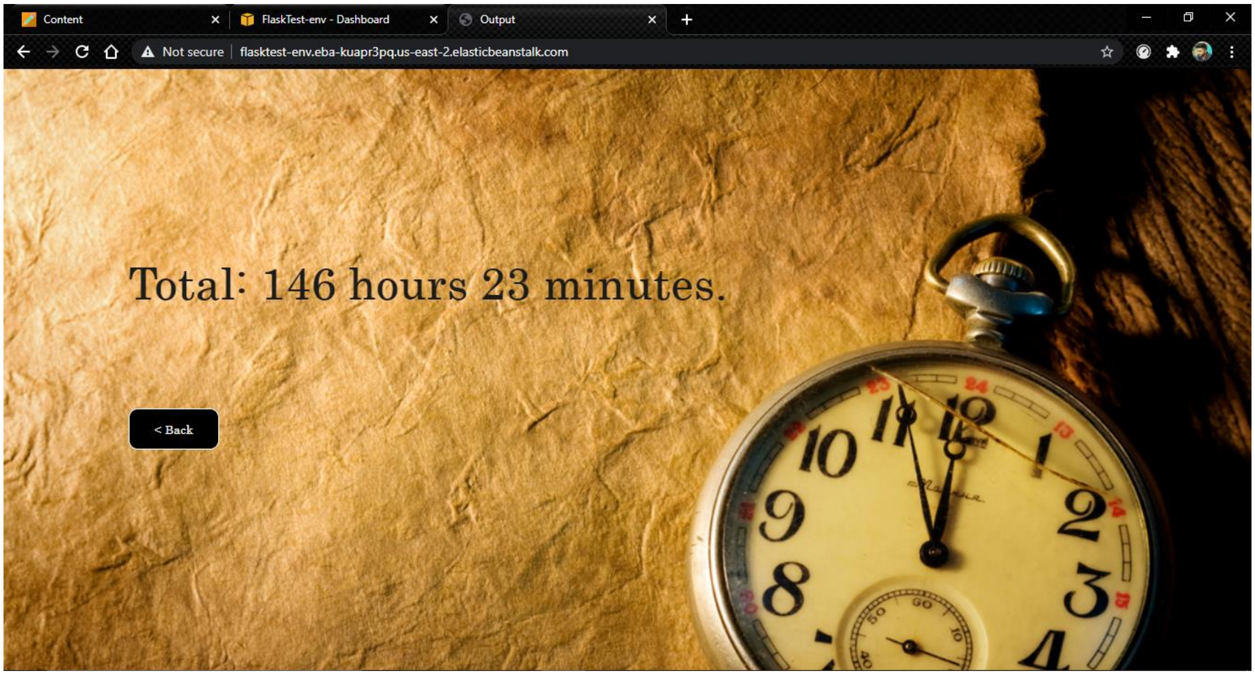
CIS 524 Final Project

Mitul Kabutarwala

2783593

Choose File
TimeLogCarbon.txt
Submit

Output



Content
FlaskTest-env - Dashboard
Output
Not secure
flasktest-env.eba-kuapr3pq.us-east-2.elasticbeanstalk.com

Total: 146 hours 23 minutes.

The zip file is submitted with this report please check the code in it.

Python and flask file:- application.py

Html files:- templates/try.html and templates/output.html

CSS files:- static/trycss.css and static/outputcss.css

Link of project:- http://flasktest-env.eba-kuapr3pq.us-east-2.elasticbeanstalk.com/

Reference:

1. Cloud hosting services, https://www.techopedia.com/definition/29018/cloud-hosting
2. Flask, https://www.codementor.io/@jqn/deploy-a-flask-app-on-aws-ec2-13hp1ilqy2