Xiaomei Xie
CPSC5200 WIN 2021

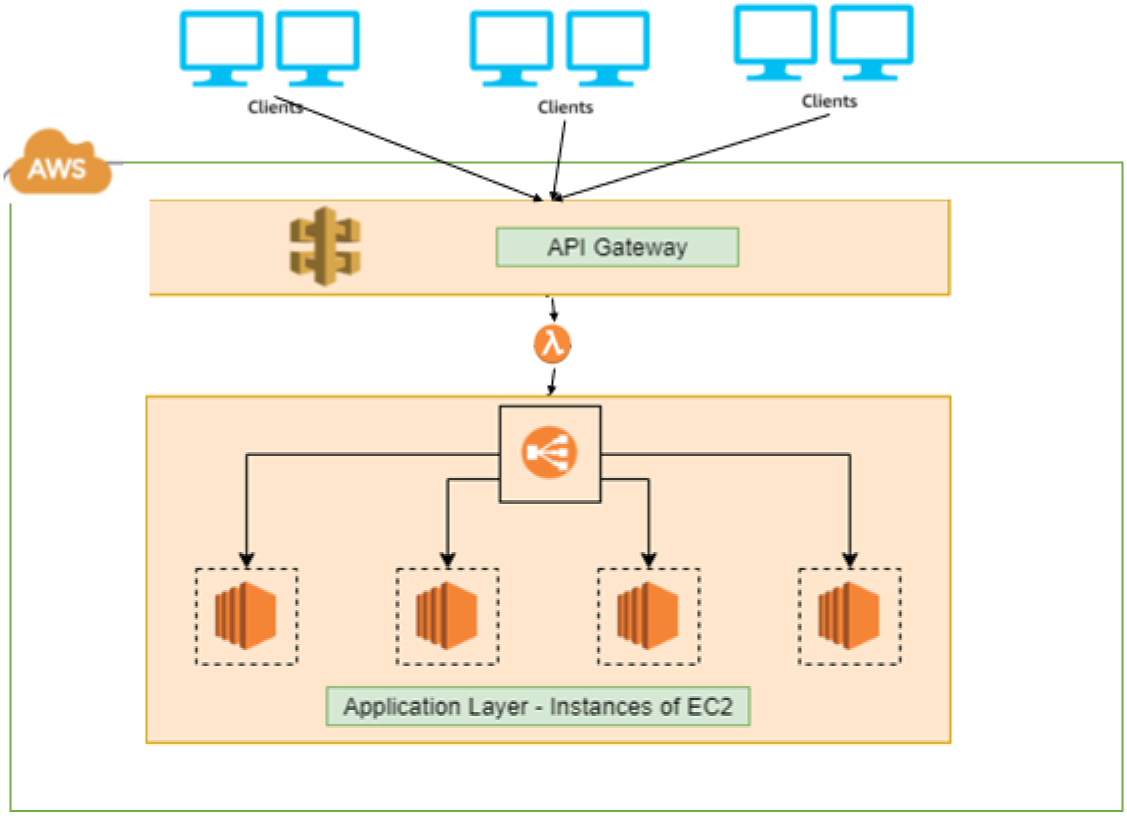
Image Processor

Introduction:

Image Processor application is a cloud hosted, distributed system. It allows user to apply transformations on top of image in order. Transformations include grayscale, flip horizontally, flip vertically, resize, thumbnail, rotate -/+ n degrees, Black and white, blend image. User will upload the image from their hard drive and send request for transformations along with an image. Application on server side will apply transformations, process the image, then returns result image.

Architecture:

I am using Amazon AWS cloud platform services to host application.



Application Layer: core application code lies here. Its deployed on EC2 with Auto scale enabled.

API Gateway: its an entry point for clients to talk to application layer. Request are sent to Load Balancer, which will route requests to EC2 servers.

Language/Design choices: Python3, Flask and Pillow from python liberary

1. API accepts and produces application; charset=UTF-8 content type.
2. Python as primary language. Python3 version is used to develop application.
3. To create REST API's and manage application, Flask is used to implement API.
4. CORS is enabled for all domains for all paths.
5. Image is stored on application memory.
6. Import Pillow from python library to process image. Pillow as external libraries to process image need be installed before starting the sever.
7. API works on HTTP protocol.
8. Image to be processed is sent to API .
9. Client must provide at least one transformation, in valid format, when using API to process image.
10. Application will validate the user's input before post to server.
11. For the front-end, use html, css to create dynamic web form by using "POST and GET" to interact with servers.

CI/CD:

AWS Code Pipeline is used to automate delivery and deployment process.

GitHub is using for project repository.