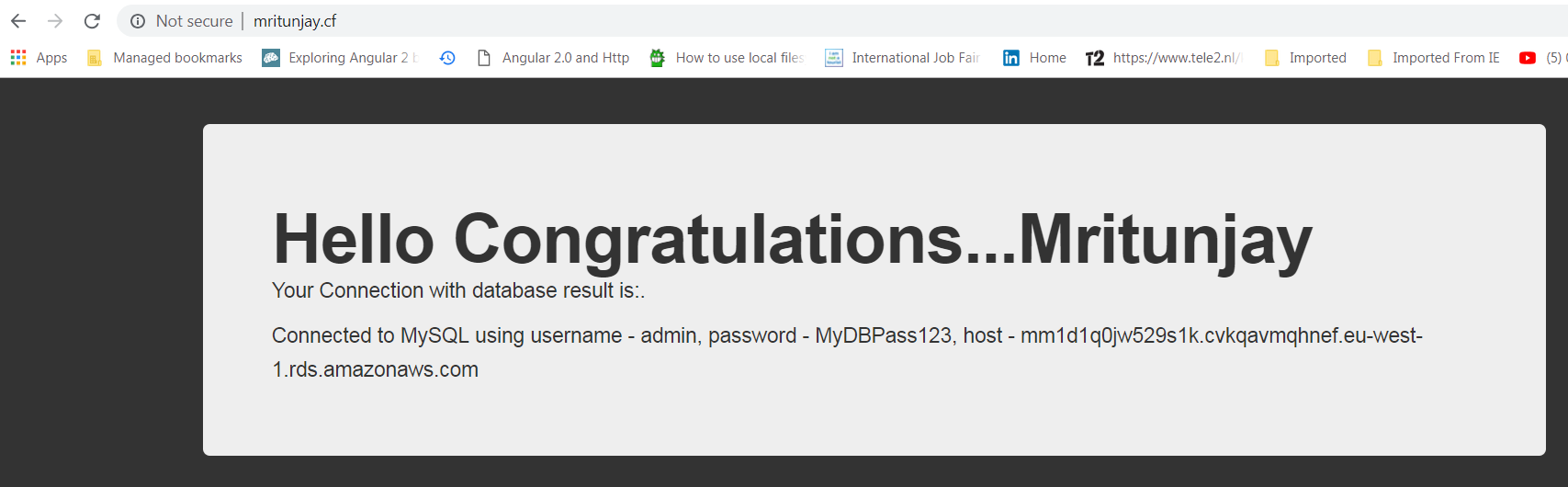
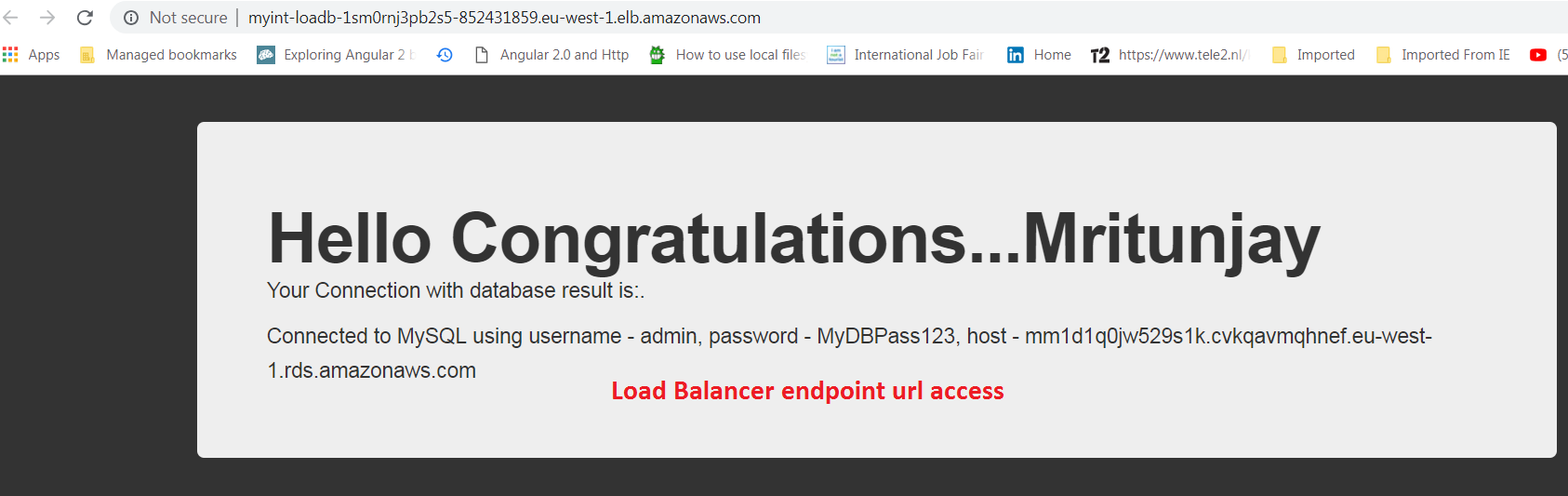
|  |  |
| --- | --- |
| Endpoint | URL |
| Route 53 Endpoint | http://mritunjay.cf/ |
| Load Balancer Endpoint | http://myint-loadb-1sm0rnj3pb2s5-852431859.eu-west-1.elb.amazonaws.com/ |





Application code is divided into 2 parts.

1. Application source code.
2. Cloud formation template for Amazon web services (AWS)

# Application Source Code:

Source code of application are located on my GitHub repository.

Url: <https://github.com/mritunjayemail/code-base-github-interview>

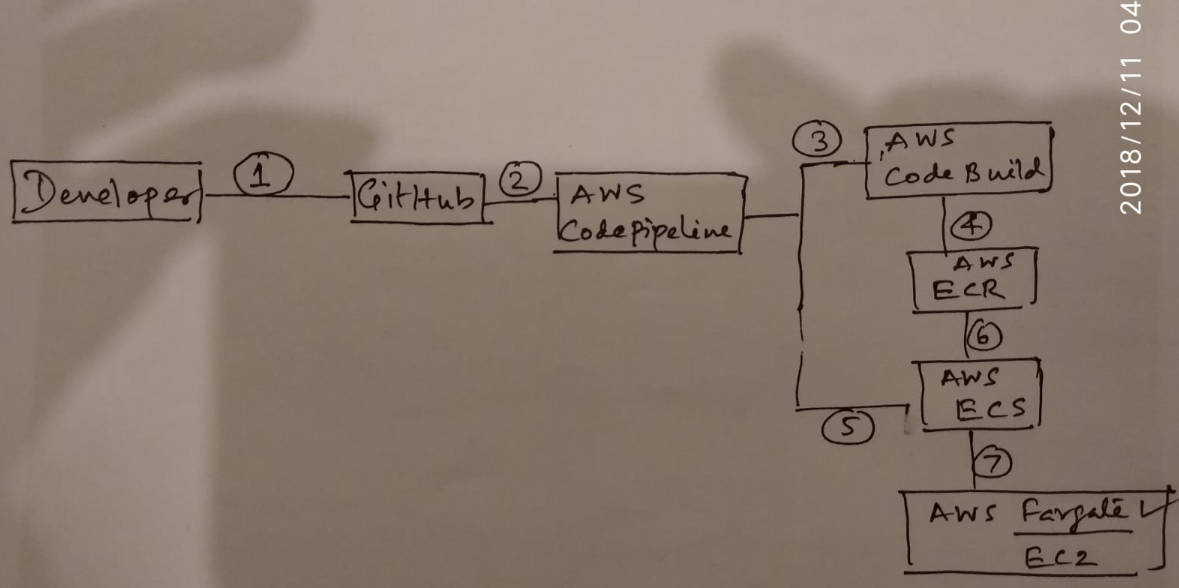
Git Repository URL for clone : <https://github.com/mritunjayemail/code-base-github-interview.git>

# Cloud formation template for Amazon web services (AWS)

Url: <https://github.com/mritunjayemail/CloudFormation-templates-continuous-deployment>

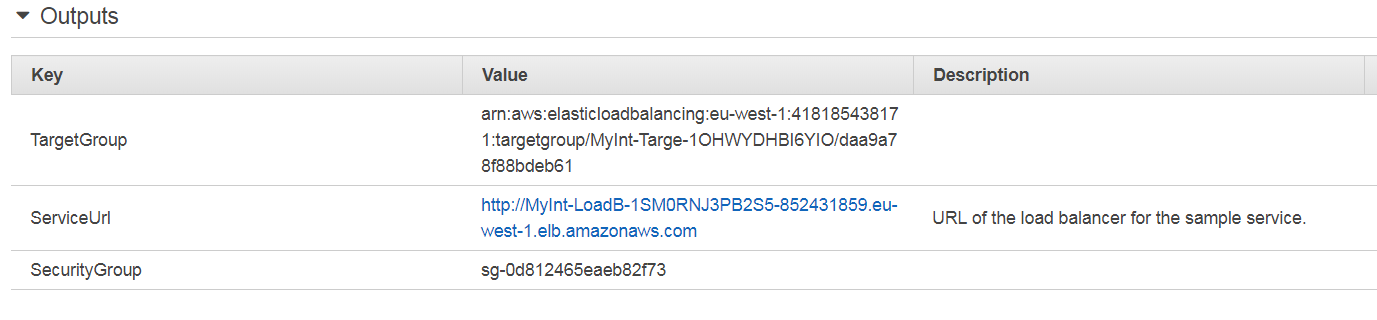
Git Repo: <https://github.com/mritunjayemail/CloudFormation-templates-continuous-deployment.git>

Overview:

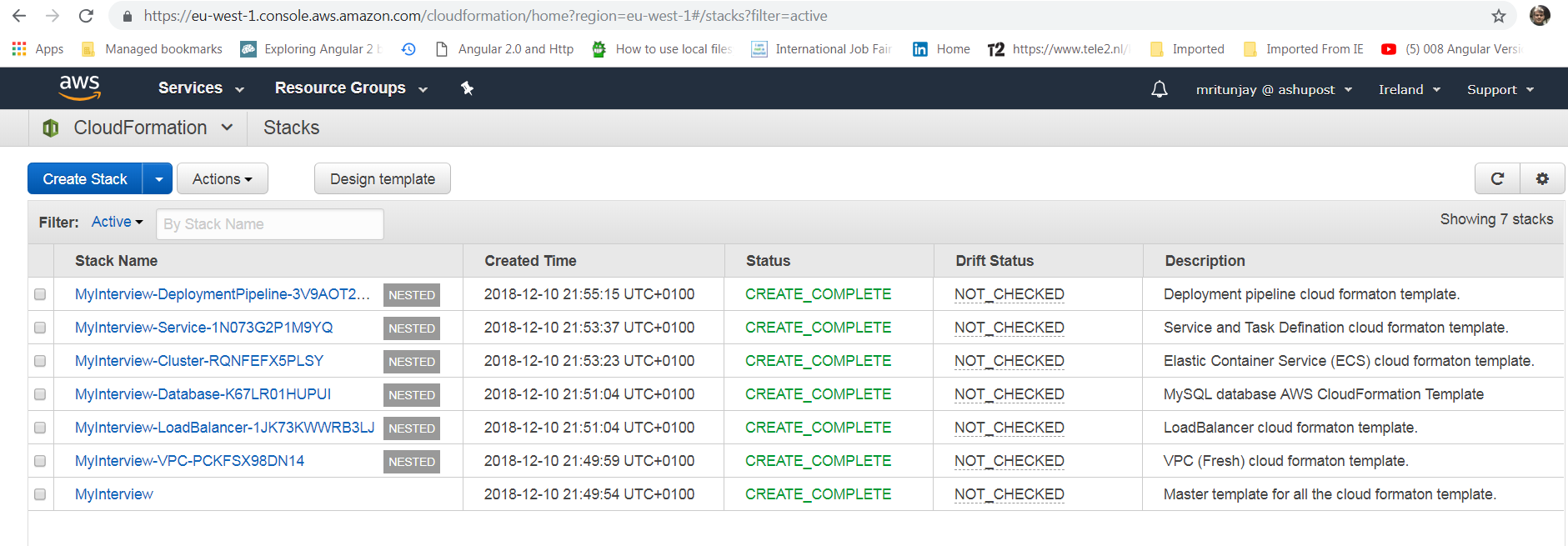


* Developer continuously integrate their changes into main branch hosted within main source code repository system such as GitHub.
* AWS code pipeline continuously polls the source code repository and trigger an execution of the pipeline when the new revision is found.
* AWS code pipeline executes a build of the new revision in AWS code build which create a docker container image from the source code.
* AWS code build pushes a newly build Docker container image tagged with the revision id to an ECR repository.
* AWS code pipeline an initiate an update of the Amazon ECS stack definition and service with the new image location.
* AWS ECS fetches the new container from Amazon ECR and replicate the old task with a new one on the cluster.
* The new revision of the service now running on the cluster using the specific launch type Fargate.

# Load balancer Cloud formation stack gives us output of load balancer Service URL.



# Complete stack overview on AWS console:



AWS CodePipeline is a fully managed continuous delivery service. CodePipeline automates the build, test, and deploy phases of your release process every time there is a code change.

## GitHub Configuration

Repo: The repo name of the sample service.

Branch: The branch of the repo to deploy continuously.

User: Your username on GitHub.

Personal Access Token: Token for the access of github.

