

## Automating the CSR Infrastructure and CSR Code Deployment

**Note:** All the Automation is achieved through the Terraform scripts and finally integrating it into one click deployment through Jenkins Job.

1. **Automation server creation** :Created an EC2 instance and configured the same to be used as our Automation server on which we have installed Terraform, Jenkins and other required tools.  
**Status:** Done.
2. **Creating the S3** : Creating the S3 bucket as per the infrastructure needed in a scriptable fashion.  
**Status:** Done.
3. **Code Deployment to the S3 Bucket:** Uploading folder and files in the bucket in a scriptable fashion directly from the Github repository.  
**Status:** Done.  
Note: Right now the code is deployed from the local path not Github. I will map it to Github by the end of the sprint.
4. **Creating the Jenkins Job for the whole S3 bucket:** One click deployment to deploy the infrastructure and code (Data) into the S3 bucket.  
**Status:** Done.
5. **Creating the API Gateway:** Creating our present API Gateway in a scriptable fashion.  
**Status:** Done.
6. **Importing the API configuration:** Mapping the swagger File into the API Gateway.  
**Status:** Done.
7. **Jenkins Job for the API Gateway:** Creating jenkins job and achieves one click deploy for the API Gateway with the swagger file integrated.  
**Status:** Pending (Jenkins job will be created after achieving the API integration with Lambdas)
8. **Creation of Lambda Layers:** Creating the present lambda layers along with importing the zip packages from the Github repository in a scriptable fashion.  
**Status:** Done.
9. **Creation of Lambda Functions:** Creating the present lambda Functions in a scriptable fashion.

**Status:** Done.

10. **Code deploy to lambda Functions:** Importing the python code into the lambda function.

**Status:** Done.

11. **Code deploy the ACG-ETL to functions:** Importing the python code for ACG-ETL into Lambdas.

**Status :** Need to get clarity and verify from Arun (Will be doing today)

Questions: what is this ETL lambda Function for..?

12. **Mapping Layers to Lambda Functions:** Integrating the lambda layers to the Lambda Functions.

Status : In progress

**ETA:** Monday 18th july, 2022

13. **Integrating the Lambdas to the API Gateway:** Mapping the layered lambda functions to the API gateway.

**Status:** Pending

**ETA:** Wednesday 20th july, 2022

14. **Jenkins Job for Lambdas and Api gateway:** Jenkins job to deploy layered Lambda functions and integrating it to the API Gateway which achieves both infra and Code deployment for both lambdas and API Gateway.

**Status:** Pending

**ETA:** Thursday 21th july, 2022

15. **Unify the S3, lambda and API :** one click deployment which deploys S3, lambda and API with both infrastructure and code integrated via jenkins.

**Status:** Pending

**ETA:** Friday 22th july, 2022

16. **Creating the DB instances:** creation of the DB instance in a scriptable fashion.

**ETA:** Still to do R&D

17. **Integration of Postgres DB:** Integrating the postgres Db into the DB instance

**ETA:** Still to do R&D

18. **pg\_admin tool:** Deploying the pg\_admin tool that is installing pg\_admin tool in the ec2 box

**ETA:** Still to do R&D

19. **DataBase Creation:** Creating the database server and mapping that to the RDS instance endpoint.

**ETA:** Still to do R&D

20. **Restoring the Dump File:** Restoring the dump file into the created Databases to achieve restoration to see AWS\_DDL schema under the DB with 4 different tables. (Basically dump file is available in Git repo)

Alternative : Use terraform to create empty tables to achieve the structure (Column names and table names )

**ETA:** Still to do R&D

Note: We are not automating the R side