AWS Group Project Chapter 8 Submission Document

Fill out each of the areas highlighted in yellow.

Who

Who contributed to this project submission? (Leave off any group members who did not substantially contribute to the project)

John Doe, Jane Doe, Jeremiah Doe, Janessa Doe

Rubric

Fill out the grading rubric below, indicating what your target score is for this project based on the work you completed.

Milestone 3 Features	Target	Possible
1. Create a code pipeline for your application	X	23.25
2. Application runs on infrastructure from IaC, including an RDS instance	X	1
3. Application is NOT based on Node	X	1
4. EC2 instances are launched into an auto scaling group	X	.5
5. EC2 instances in auto scaling group are behind a load balancer using IaC for all parts.	X	.5
6. Alternative but equivalent technology to CodePipeline/CodeDeploy is used	X	1
7. Other enhancements that support the principles of using a code pipeline	X	TBD
Total	X	**25 (max)**

For example, if you tried to earn the minimal points for an A or 93% (23.25 points), the table would look like this:

Milestone 3 Features	Target	Possible
1. Create a code pipeline for your application	23.25	23.25
2. Application runs on infrastructure from IaC, including an RDS instance	0	1
3. Application is NOT based on Node	0	1
4. EC2 instances are launched into an auto scaling group	0	.5
5. EC2 instances in auto scaling group are behind a load balancer	0	.5
6. Alternative but equivalent technology to CodePipeline/CodeDeploy is used	0	1
7. Other enhancements that support the principles of using a code pipeline	0	TBD
Total	**23.25**	**25 (max)**

Proof of Feature Completion

For each feature you completed, explain WHAT you did and provide PROOF that you did it. You can either write out these items and use screenshots below, or create a short video and go over each feature in the video.

Feature 1. Create a code pipeline for your application

WHAT YOU DID
PROOF THAT YOU DID IT

EXAMPLE:

WHAT: We created a code pipeline using CodeCommit, CodePipeLine, and CodeDeploy. For infrastructure we used RDS, For the application we used Node as the programming language

PROOF THAT YOU DID IT:
Screenshots of our deployment code including appspec.yml and start/stop scripts [IMGS]

Screenshot of the running website page[IMG]

Screenshot of CodePipeline progress page showing a successful release [IMG]

Feature 2. Application runs on infrastructure from IaC, including an RDS instance

WHAT: We updated our IaC code to create the infrastructure, properly tagged EC2 instances, and an RDS instance.

PROOF THAT YOU DID IT:
Screenshot of Stack information related to the running infracture
CloudFormation Template pasted here (or a link to it)

Feature 3. Application is NOT based on Node

WHAT: We created our application using a Django stack

PROOF THAT YOU DID IT:
Screenshot of Representative Sample Code
Screenshot of start/stop scripts from Feature 2

Feature 4. EC2 instances are launched into an auto scaling group

WHAT YOU DID
PROOF THAT YOU DID IT

Feature 5. EC2 instances in auto scaling group are behind a load balancer

WHAT YOU DID
PROOF THAT YOU DID IT

Feature 6. Alternative but equivalent technology to CodePipeline/CodeDeploy is used

WHAT YOU DID
PROOF THAT YOU DID IT

Feature 7. Other enhancements that support the principles of using a code pipeline

WHAT YOU DID
PROOF THAT YOU DID IT