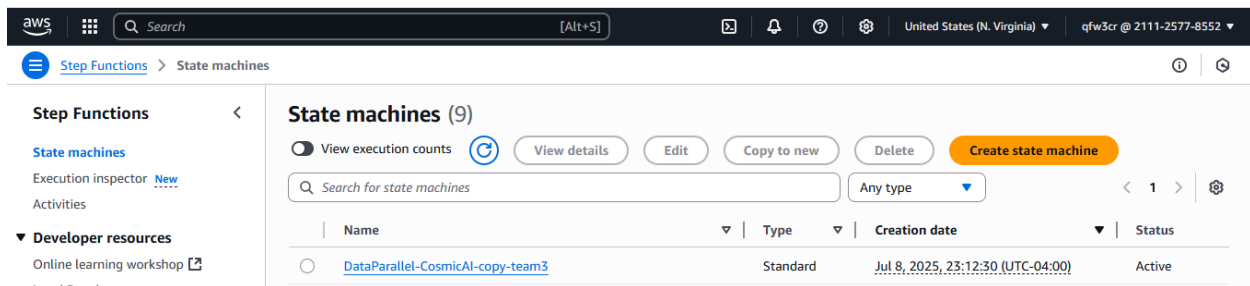


TEAM 3

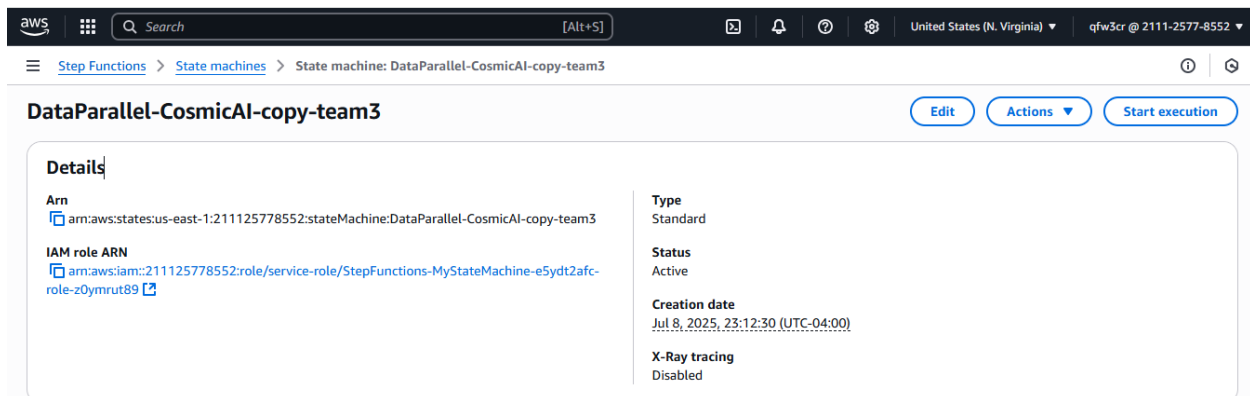
Nikpour Bardia

Victor Ontiveros

- Screenshots of your designed state machine in AWS Step Functions



- Copy of IAM role configurations used for S3 access



- Sample JSON payloads used for parameter passing

The screenshot shows the AWS Step Functions console in the 'Design' view. The state machine 'DataParallel-CosmicAI-copy-tea...' is displayed with a linear workflow. The 'Map JSON Payload Distributed Test' step is selected, and its payload is shown in the right-hand pane. The payload is a JSON object with the following structure:

```

1 {
2   "bucket": "cosmicai-data",
3   "file_limit": "41",
4   "batch_size": 512,
5   "object_type": "folder",
6   "s3_object_name": "Anomaly Detection",
7   "script": "/tmp/Anomaly
8     Detection/Inference/inference.py",
9   "result_path": "scaling/result-partition-
10     100MB/8GB/2",
11   "data_bucket": "cosmicai-data",
12   "data_prefix": "100MB"
13 }

```

- Screenshots of execution results showing successful workflow completion

The screenshot shows the AWS Step Functions console in the 'Execution' view for the state machine 'DataParallel-CosmicAI-copy-team3'. The execution ID is '24f02170-27c5-42bd-b7a6-5761c8b675f4'. The execution status is 'Succeeded'. The execution input and output are shown in the right-hand pane. The input is a JSON object with the following structure:

```

1 {
2   "bucket": "cosmicai-data",
3   "file_limit": "41",
4   "batch_size": 512,
5   "object_type": "folder",
6   "s3_object_name": "Anomaly Detection",
7   "script": "/tmp/Anomaly
8     Detection/Inference/inference.py",
9   "result_path": "scaling/result-partition-
10     100MB/8GB/2",
11   "data_bucket": "cosmicai-data",
12   "data_prefix": "100MB"
13 }

```

The output is a JSON object with the following structure:

```

1 {
2   "bucket": "cosmicai-data",
3   "file_limit": "41",
4   "batch_size": 512,
5   "object_type": "folder",
6   "s3_object_name": "Anomaly Detection",
7   "script": "/tmp/Anomaly
8     Detection/Inference/inference.py",
9   "result_path": "scaling/result-partition-
10     100MB/8GB/2",
11   "data_bucket": "cosmicai-data",
12   "data_prefix": "100MB"
13 }

```

- Performance measurement table showing memory usage, duration, and cost metrics

Table view

	Name	Type	Status	Duration	Timeline	Started After	Timestamp
<input type="radio"/>	Initialize	Task	✓ Succeeded	00:00:01.453	<div><div></div></div>	00:00:00.027	Jul 8, 2025, 23:23:31.533 (UTC-04:00)
<input type="radio"/>	Distributed...	Map	✓ Succeeded	00:00:30.377	<div><div></div></div>	00:00:01.480	Jul 8, 2025, 23:23:32.986 (UTC-04:00)
<input type="radio"/>	Summarize	Task	✓ Succeeded	00:00:07.711	<div><div></div></div>	00:00:31.857	Jul 8, 2025, 23:24:03.363 (UTC-04:00)

Brief explanation (1-2 paragraphs) of your implementation approach and any challenges encountered

- We followed the class instructions without any problem. It was easier to clone a previous Step Function to use as reference.
- The most challenging part was the execution...we got errors related with the access to S3 bucket. After we gave the correct access everything was working fine.