

SAM AI Explanation

Your AWS SAM template defines a **serverless application** that includes:

1. A **DynamoDB Table** (MomentsTable)

- Table name: SAM-Moments
- Primary key: `_id` (String)
- Billing mode: **PAY_PER_REQUEST** (on-demand pricing)

2. Four **AWS Lambda Functions** for CRUD operations:

- `CreateMomentFunction` → Handles **POST /moments**
- `GetAllMomentsFunction` → Handles **GET /moments**
- `GetMomentFunction` → Handles **GET /moments/{id}**
- `DeleteMomentFunction` → Handles **DELETE /moments/{id}**
- Each Lambda function has a fixed IAM role (`arn:aws:iam::123456789012:role/LabRole`)

- Environment variable TABLE_NAME is injected globally

3. An API Gateway (MomentsApi)

- Manages the API routes
- Uses a dynamic StageName parameter (default: dev)

4. Output (ApiUrl)

- Constructs the API endpoint dynamically using AWS::Region

Notes:

- Your template is **modular** (supports different deployment stages via StageName).
- It follows **best practices** (environment variables, separation of concerns).
- If needed, consider adding **IAM permissions** for better security.