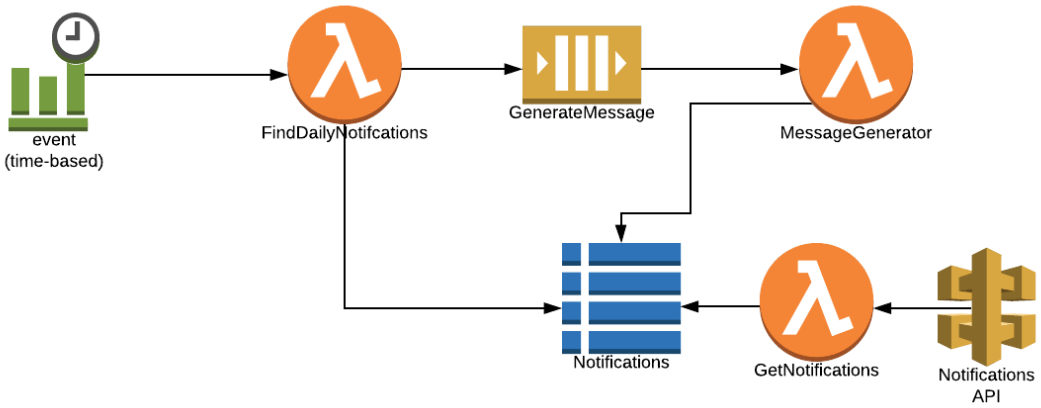
DevOps Engineering Take Home

Deploy your solution to GitHub or another code repository of your choice. Please provide a link to the completed solution within 4 days.

Consider the following AWS architecture for a notification system:



This system runs daily, finds all notifications for that particular day, generates messages for all applicable users, and saves for later use by an API.

Name	Service	Notes
Time Based Event	Cloud Watch	Scheduled event that runs once a day
FindDailyNotifications	Lambda	Queries the DB, to find all notifications for the day. Each notification to be generated results in a queued message
GenerateMessage	SQS	
MessageGenerator	Lambda	Finds all user that need to receive the notifications and saves records into the DB
Notifications Table	DynamoDB	Storage for user generated notifications
GetNotifications	Lambda	Backend logic to fetch

		notifications for a user
Notifications API	API Gateway	Endpoint to fetch notifications on demand

- **Build a Delivery PIpeline:** Create a delivery pipeline to deploy the architecture detailed above. Your solution should contain:
    a. Scripts to deploy the infrastructure across dev, qa, and production environments including configuration for each environment
    b. Infrastructure defined as code
    c. Permissions that follow least privilege principle
    d. Take networking into consideration, making sure systems can be as private as possible and minimize traffic over the public internet
    e. Tests that validate your deployment worked as expected
    f. Any assumptions made should be detailed with what the assumption is, why it is reasonable to make it, and how your solution may change if your assumption is incorrect
- **Data Pipeline:** Create a data pipeline that takes the attached csv, and see it from an s3 bucket to database technology of your choosing. Your solution should:
    a. Put a file into an s3 bucket
    b. Manage s3 bucket and bucket policies
    c. Create a secure a database
    d. And copy data into it
- **Monitoring and Alerting**: Describe how you would monitor the two systems above once once they are deployed. Your answer should include:
    a. Key metrics/SLAs for each service that you would look at to determine system health
    b. Events that you would use to alert the team of errors.
    c. Discuss what tooling you would use to facilitate the alerting and monitoring