DevOps Project Breakdown

Project Introduction:

Students will use infrastructure-as-code (IaC) constructs to build and operate a web application running across multiple AWS Regions. They will write and deploy a Python application on AWS, store logs in CloudWatch. They will then create alarms on the metrics to identify performance. Finally, they will log the alarm information in a no-SQL database.

Week-wise Project Plan

Week	Project	Learning Objectives
3	Use AWS CDK to build a canary in a Lambda function. This canary runs in one AWS Region and measures metrics of a web resource for monitoring. Push the code to versioning control repo. Manage README files in markdown on GitHub.	• Introduction to the DevOps Engineer Role and Infrastructure-as-Code (IaC) • Introduction to AWS: Regions/AZs/Edge Services, Foundational services (EC2, S3, CloudFront), Microservice architecture • Learn AWS Services: IAM, Lambda • Learn GitHub • Start writing code on AWS
4	Extend the canary Lambda function into a web crawler -- crawl a custom list (json file) of websites. Run the crawler periodically on a 5 min cadence and write <availability, latency> metrics for each website and each run to CloudWatch using CloudWatch's API. Create a CloudWatch Dashboard to monitor website health. Manage README files and runbooks in markdown on GitHub. Update the Github dashboard with new user stories and feature breakdowns.	• Introduction to the Art of Monitoring Web Applications • Learn AWS Services: CloudWatch • Introduce scalability in web application
5	Set up alarms when availability or latency falls below prescribed thresholds. Alarms are also published to SNS notifications with tags that can be used to filter by metric type. The alarms are also logged in a no-SQL database. Manage README files and runbooks in markdown on GitHub.	• Learn AWS Services: SNS, DynamoDB
6	Backlog Week	