AWS CLOUD INSTITUTE

Level Up: Developer Intermediate Syllabus

Session Overview

In Level Up: Developer Intermediate, you will embark on a comprehensive journey through the service-design process. This immersive experience will equip you with the skills that are necessary to effectively tackle business problems and develop robust solutions. Through hands-on activities and practical exercises, you will learn to create use cases that accurately capture the requirements of a given business problem. By building on these use cases, you will develop a comprehensive testing plan to ensure the quality and functionality of your solution.

Additionally, you will delve into the art of frontend design by crafting intuitive and user-friendly interfaces for your applications. You will also explore the intricacies of using API design to enable seamless communication and data exchange between different systems and components.

Instructor-Led Training (ILT) Sessions

Attendance is strongly encouraged to continue to build your developer skills. These sessions will allow you to dive deep into specific topics and ask questions. The sessions will be recorded, allowing you to view each session on demand to continue to build your knowledge. Additionally, office hours are available for you to schedule one-to-one time with an AWS instructor, getting further support in preparing and meeting your goals.

Prerequisites

Enrollment in, or completion of, any of the Developer Intermediate courses is required.

Week 1: Create a Design Document from Business Requirements

Goal:

Understand the design document, construct functional requirements based on a scenario, and create functional requirements and use cases for a problem.

Learning Objectives:

- Learn how a design document supports collaborative work by describing what needs to be done to solve a problem
- Write a design document to communicate plans to solve a problem
- Construct clear and testable use cases in a *given-when-then* format
- Identify a set of use cases that fulfills customer needs for a software system
- Create use cases to cover both the happy and alternate cases for a software system

Week 2: Create a Test Plan

Goal:

Create a test plan to validate that the end-to-end functionality of a service fulfills a given set of use cases.

Learning objectives

- Construct a test plan and write a set of tests to validate that the end-to-end functionality of a service fulfills a given set of use cases.

Week 3: Frontend Design

Goal:

Discuss user interface (UI) design and create mock-ups of a UI for a web application.

Learning Objectives:

- Review how a system's frontend is responsible for presenting an interface to the user, while its backend is responsible for manipulating data
- Write a test plan that describes the functionality of a frontend
- Create UI mock-ups for a web application

Week 4: API Design

Goal:

Discuss create, retrieve, update, and delete (CRUD) API design, and define a set of APIs to support a set of use cases.

Learning Objectives:

- Explain the concept of CRUD operations
- Recommend a set of API endpoints that support a set of use cases
- Define a service endpoint that uses inputs and outputs to support a use case

Week 5: Service Design – Composition

Goal:

Discuss the composition of classes and objects needed to solve a business problem.

Learning Objectives:

- Determine the core objects needed for a set of use cases
- Design classes and objects to implement a set of API endpoints
- Create a class diagram that represents the core objects for a service

Week 6: Service Design – Business Logic

Goal:

Learn how to write the business logic for an API endpoint.

Learning objectives

- Write a sequence diagram for an API endpoint.

Week 7: Database Table Design

Goal:
Learn how to design tables and items using Amazon DynamoDB and create an entity relationship diagram for a simple business problem.

Learning Objectives:

- Design database items to represent core objects from a service design
- Define a key structure for database items for a simple business problem
- Create an entity relationship diagram to represent a data model for a simple business problem

Week 8: Database Query Design

Goal:
Learn how to use database queries to solve business problems and design a DynamoDB query to retrieve data for a simple business problem.

Learning Objectives:

- Plan CRUD operations for a simple business problem
- Define a key structure for database items for a simple business problem
- Design queries to retrieve data for a simple business problem

Week 9: Architecture Design

Goal:
Discuss microservice architecture and use AWS services to plan a microservice architecture.

Learning Objectives:

- Explain the advantages of a microservice architecture
- Use AWS services to design a microservice architecture
- Create an architecture diagram to solve a simple business problem

Week 10: Development Operations

Goal:
Discuss development operations processes, including source control, code reviews, and deployment processes.

Learning Objectives:

- Explain the importance of source control
- Understand the importance of code-review processes
- Learn how to use different deployment stages to help separate test data from production data

Week 11: Project Tracking

Goal:
Learn how to decompose a project into tasks and milestones and offer clear communication to collaborators.

Learning Objectives:

- Divide a project into a set of tasks and milestones
- Learn how to make goals and plans clear to all collaborators