Module 2: Learn Kafka Basics

User Story 1: Setup Kafka Environment Locally

Description: Set up Confluent Kafka locally, practice using Kafka Topic, Producer, and Consumer CLI commands.

Acceptance Criteria: Kafka is installed, and Kafka Topic, Producer, and Consumer commands are executed successfully.

Story Points: 5

User Story 2: Set up Spanner DB Emulator (GCP) and Java IDE

Description: Install and configure Spanner DB Emulator and Java IDE for development.

Acceptance Criteria: Spanner DB Emulator is functional, and Java development environment is set up.

Story Points: 3

Module 3: Learn Kafka Consumer and Producer API

User Story 3: Create a Java Application with Kafka Producer/Consumer APIs (Part 1)

Description: Build a Java application using Kafka Producer and Consumer APIs. Consume events from a source Kafka topic, filter them, and produce the filtered events to a target Kafka topic.

Acceptance Criteria: Events are successfully consumed, filtered, and produced to the target Kafka topic.

Story Points: 8

User Story 4: Rewrite Application Using KStreams API (Part 2)

Description: Convert the Java Producer/Consumer application into a stateless streaming application using Kafka Streams API. Integrate with Schema Registry for managing schema.

Acceptance Criteria: Events are streamed and processed statelessly, schemas are registered in Schema Registry.

Story Points: 8

User Story 5: Enhance Application with KTable for Stateful Streaming (Part 3)

Description: Convert the stateless Kafka Streams application into a stateful streaming application by using KTable as a state store. Enrich the stream by joining it with customer data and produce the enriched events to the target Kafka topic.

Acceptance Criteria: Events are enriched and processed statefully using KTable, and the enriched data is produced to the target topic.

Story Points: 8

Module 4: Learn Kafka Connect API/Configuration

User Story 6: Create a No-Code/Low-Code Kafka Connect Data Pipeline

Description: Use Kafka Connect APIs to create a data ingestion pipeline that reads data from a curated Kafka topic and ingests it into Spanner DB.

Acceptance Criteria: Kafka Connect pipeline is set up, and data is successfully ingested into Spanner DB.

Story Points: 8

Module 5: Performance Optimization

User Story 7: Conduct Kafka Performance Testing and Create Report

Description: Run performance tests on the Kafka pipeline, measure latency and throughput, and optimize Kafka configuration for better performance.

Acceptance Criteria: Performance report generated, and any bottlenecks are resolved.

Story Points: 8

Module 6: GCP Solution Design

User Story 8: Revise Design Document with Monitoring, Alerting, and Security Enhancements

Description: Revise the original GCP design to include monitoring, alerting, and data security for Kafka in a cloud environment.

Acceptance Criteria: Design document is updated with appropriate GCP tools for monitoring, alerting, and securing Kafka data.

Story Points: 5

Final Demo

User Story 9: Conduct Final Review and Demonstration

Description: Review the entire system, run a final demo, and make any required changes based on feedback.

Acceptance Criteria: Final demo is completed successfully, and feedback is incorporated into the solution.

Story Points: 3

Summary of Story Points by Module:

Module 2: 8 Story Points

Module 3: 24 Story Points

Module 4: 8 Story Points

Module 5: 8 Story Points

Module 6: 5 Story Points

Final Demo: 3 Story Points

Total: 56 Story Points across 9 weeks, with 5-8 points per sprint.