

NoSQL and Distributed SQL Databases - GD

Course Objective

After completing this course, students will understand NoSQL (DynamoDB) and Distributed SQL (YDB) databases, design and optimize data models for them, and deploy, secure, and monitor database-backed applications on the cloud.

Prerequisites

- Backend development in one language, with Git and command-line basics
- Relational DB and SQL fundamentals (keys, joins, ACID, indexing)
- HTTP/API basics (curl/Postman, auth tokens)

Technologies

Amazon DynamoDB, YDB (YQL), AWS SDK, Terraform/IaC, CloudWatch, Git

Course Topics

Introduction to Modern Databases

- Modern problem statement: high availability, scale-out, fault tolerance
- Why NoSQL and Distributed SQL exist
- Technology market overview

NoSQL and Distributed SQL Fundamentals

- NoSQL: basic components, use cases, limitations
- Distributed SQL: basic components, use cases, limitations

Designing Your Schema

- NoSQL schema design: access patterns, partition/sort keys, sample schemas
- Distributed SQL schema design: basic considerations, sample schemas
- Practical: design a schema for an assigned application scenario (team exercise)

Testing Your Schema

- Peer review of student schema designs
- Step-by-step plan to load-test a schema (consistency, failures, scalability)

Schema and Query Optimization

- Review of common schema-design errors and reported problems
- Finding suboptimal queries: query plans, PK usage, index usage, optimistic locking
- Practical: analyze and optimize queries in your project

Deploying to the Cloud

- Serverless vs. dedicated deployment
- Infrastructure planning (IaC)
- Security: service accounts, least privilege, dev/test/prod separation
- Database deployment

- Backup and restore testing
- Practical: deploy your project to the cloud

Production Monitoring

- Golden signals and where to source them
- Application and database SDK signals
- Alerting design and common problem diagnostics
- Practical: set up monitoring and alerting for your project, in IaC

Final Project Presentations

Demo requirements:

- live dashboard
- induced failures
- scalability demonstration

Literature and sources

- Alex DeBrie: *The DynamoDB Book*, self-published, 2020. <https://www.dynamodbbook.com/>
- Martin Kleppmann: *Designing Data-Intensive Applications*, O'Reilly, 2017.
- Amazon DynamoDB Developer Guide: <https://docs.aws.amazon.com/dynamodb>
- YDB Documentation: <https://ydb.tech/docs/en/>
- AWS Database Blog: <https://aws.amazon.com/blogs/database/>
- Alex DeBrie's blog: <https://www.alexdebrie.com/>
- Marc Brooker's blog: <https://brooker.co.za/blog/>
- CMU 15-445/645 Intro to Database Systems (Andy Pavlo): <https://15445.courses.cs.cmu.edu/>
- Jepsen analyses: <https://jepsen.io/analyses>
- Google: Site Reliability Engineering, <https://sre.google/books/>