Independent Study Proposal

Independent Study Advisor: Senior Teaching Faculty Professor Timothy Richards
Independent Study Advisee: Undergraduate student Yongye Tan in his junior year

Statement Objectives: The main objective of this proposal is to delve into cloud migration and containerization within the framework of a microservice architecture using the Javascript or Typescript programming language, but not limited to these languages. Another important objective is to focus on API version management and explore GraphQL, as well as acquiring skills to manage software workflows through CI/CD methodologies.

Learning Outcomes:

- Define the concepts and application of JSON Web Tokens (JWT) & session management.
- Explain procedures of creating, managing and deploying Docker containers in the cloud
- Explain the architecture and importance of Kubernetes in orchestrating containers.
- Use existing Amazon Web Service (AWS) services such as Lambda, API Gateway, EC2, ECS, and S3, explaining their integration and utilization in distributed web systems.
- Explain the principle of Continuous Integration and Continuous Deployment (CI/CD) and implement pipelines using tools like GitHub Actions and Jenkins.
- Analyze and compare the pros and cons between GraphQL and traditional REST API.
- Understand the basics of query language, and learn about overfetching and underfetching.
- Use and adopt multiple API versioning strategies and their importance in scalability.
- Apply languages, libraries, and tools that are important for scalability.

Schedule (Subject to Change):

Week	Topic	Details
Week 1	JWT & Session Management	Introduction to JWT (JSON Web Tokens), session management concepts, and best practices in web security.
Week 2-3	Cloud Deployment	Learning Docker basics, containerization concepts, Kubernetes orchestration, and deploying simple applications.
Week 4	Homework 1	Develop & deploy microservices to the cloud using Docker containers and Kubernetes
Week 5-6	Exploring AWS Services	Overview of key AWS services, such as EC2, S3, ECS, and API Gateway, and Lambda, and understand their application in microservices architecture in real life.

Week 6-7	CI/CD Methodology	Introduction to CI/CD pipelines, hands-on with GitHub Actions, and a comparison with Jenkins for automation.
Week 8	GraphQL with API	Introduction to GraphQL, benefits over REST, and building a simple GraphQL API.
Week 9	Homework 2	1. Design & Implement microservices with at least two AWS service 2. Apply GitHub Actions by writing YAML file for the GitHub Repository 3. Design a server with a large amount of data and compare fetching them using GraphQL queries
Week 10-11	API Versioning	Understanding API versioning strategies, implementing version control in APIs, and best practices.
Week 11-13	Final Project	1. Develop a full-stack MERN web application that utilizes all the tools mentioned above 2. Incorporate the Jest testing framework and ESLint for code formatting

What does a week-to-week schedule look like?

The plan involves conducting extensive research using a variety of resources, including online materials, books, and videos, not limited to websites like Udemy and LinkedIn Learning. This process will focus on acquiring theoretical knowledge and applying it practically. A detailed page on the subject will be written, supplemented by additional modules that outline steps and instructions, such as creating to-do lists in Notion or Google Sheets. If time allows, a video describing the technology being worked on will be produced. Weekly reports will be made to Professor Timothy Richards about the activities and progress, with feedback received to ensure Yongye is on the right track. Professor Richards will assess the quality of Yongye's work and provide guidance and potential resources. It is Yongye's responsibility to ensure that he meets the expectations that he sets and follows his non-traditional way of learning. He will create a repository exclusively for this independent study and consistently push commits of his work.