TCSS 562: SOFTWARE ENGINEERING FOR CLOUD COMPUTING

Tutorial Demos

Wes J. Lloyd
School of Engineering and Technology
University of Washington – Tacoma

1

OFFICE HOURS – FALL 2024

- **THIS WEEK**
- **Tuesdays:**
  - 2:30 to 3:30 pm - CP 229
- **Thursday*:**
  - 6:00 pm to 7:00 pm – CP 229 and via Zoom*
- Or email for appointment

> Office Hours set based on Student Demographics survey feedback
** - tentative – waiting on confirmation of Friday faculty meeting schedule*

2

THIS WEEK

- Tuesday November 12th
- The class will meet and begin with the usual review, but instead of new lecture, we will focus on tutorial demonstrations and questions to provide a catch-up day
- Thursday November 14th
- Quiz 1 Update results discussion – Quiz 1 Update to be returned
  - *Quiz 1 Update can be used as notes for quiz 2*
- Lecture 15 on containerization

3

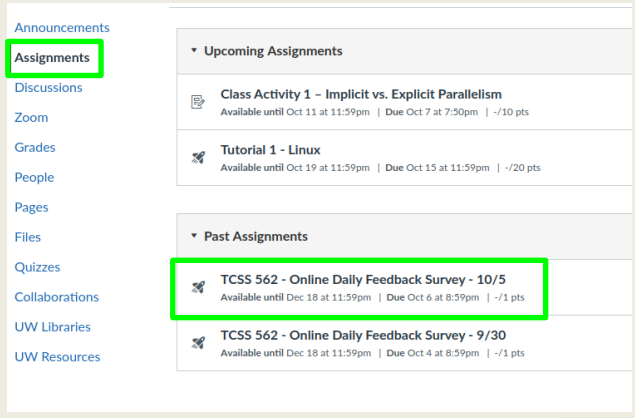
OBJECTIVES – 11/12

- Questions from 11/7
- Tutorials Questions
- Class Presentations:
  Cloud Technology or Research Paper Review
- Quiz 1 Update
- GraphQL
- Tutorial 5 Demo
- Tutorial 6 Demo

4

ONLINE DAILY FEEDBACK SURVEY

- Daily Feedback Quiz in Canvas – Take After Each Class
- Extra Credit for completing



5

TCSS 562 - Online Daily Feedback Survey - 10/5

Started: Oct 7 at 1:13am

Quiz Instructions

Question 1 0.5 pts

On a scale of 1 to 10, please classify your perspective on material covered in today's class:

1	2	3	4	5	6	7	8	9	10
Mostly Review To Me				Equal New and Review					Mostly New to Me

Question 2 0.5 pts

Please rate the pace of today's class:

1	2	3	4	5	6	7	8	9	10
Slow				Just Right					Fast

6

MATERIAL / PACE

- Please classify your perspective on material covered in today's class (**45** respondents):
- 1-mostly review, 5-equal new/review, 10-mostly new
- **Average – 5.84 (↑ - *previous* 5.42)**

- Please rate the pace of today's class:
- 1-slow, 5-just right, 10-fast
- **Average – 5.13 (↑ - previous 5.39)**

- **Response rates:**
- TCSS 462: 29/42 – 69.05%
- TCSS 562: 16/20 – 80.00%

7

FEEDBACK FROM 11/7

8

AWS CLOUD CREDITS UPDATE

- AWS CLOUD CREDITS ARE NOW AVAILABLE FOR TCSS 462/562
- Credit codes must be securely exchanged
- Request codes by sending an email with the subject "**AWS CREDIT REQUEST**" to **wlloyd@uw.edu**
- Codes can also be obtained in person (or zoom), in the class, during the breaks, after class, during office hours, by appt
  - ~56 credit requests fulfilled as of Nov 11 @ 11:59p
- Codes not provided using discord

9

OBJECTIVES – 11/12

- Questions from 11/7
- **Tutorials Questions**
- Class Presentations:
  Cloud Technology or Research Paper Review
- Quiz 1 Update
- GraphQL
- Tutorial 5 Demo
- Tutorial 6 Demo

10

Don't Forget to Terminate (Shutdown) all EC2 instances for Tutorials 3

Spot instances:
c5d.large instance @ ~3.2 cents / hour

$0.78 / day
$5.48 / week
$23.78 / month
$285.42 / year

AWS CREDITS → → → → → → → →

11

TUTORIAL 5 – DUE NOV 14

- Introduction to Lambda II: Working with Files in S3 and CloudWatch Events
- https://faculty.washington.edu/wlloyd/courses/tcss562/tutorials/TCSS462_562_f2024_tutorial_5.pdf
- Customize the Request object (add getters/setters)
  - Why do this instead of HashMap ?
- Import dependencies (jar files) into project for AWS S3
- Create an S3 Bucket
- Give your Lambda function(s) permission to work with S3
- Write to the CloudWatch logs
- Use of CloudTrail to generate S3 events
- Creating CloudWatch rule to capture events from CloudTrail
- Have the CloudWatch rule trigger a target Lambda function with a static JSON input object (hard-coded filename)
- **Optional**: for the S3 PutObject event, dynamically extract the name of the file put to the S3 bucket for processing

12

TUTORIAL 6 – NOV 23

- Introduction to Lambda III: Serverless Databases
- https://faculty.washington.edu/wlloyd/courses/tcss562/tutorials/TCSS462_562_f2024_tutorial_6.pdf

- Create and use Sqlite databases using sqlite3 tool
- Deploy Lambda function with Sqlite3 database under /tmp
- Compare in-memory vs. file-based Sqlite DBs on Lambda
- Create an Amazon Aurora "Serverless" v2 MySQL database
- Using an ec2 instance in the same VPC (Region + availability zone) connect and interact with the database using the mysql CLI app
- Deploy an AWS Lambda function that uses the MySQL "serverless" database

13

TUTORIAL 7 – TO BE POSTED

- Introduction to Docker
- (to be posted)

- Must complete using Ubuntu 24.04 (for cgroups v2)
- Use DOCX file for copying and pasting Docker install commands
- Topics:
  - Installing Docker
  - Creating a container using a Dockerfile
  - Using cgroups virtual filesystem to monitor CPU utilization of a container
  - Persisting container images to Docker Hub image repository
  - Container vertical scaling of CPU/memory resources
  - Testing container CPU and memory isolation

14

TUTORIAL COVERAGE

- **Docker CLI → Docker Engine (dockerd) → containerd → runc**
- **Working with the docker CLI:**
- **docker run** create a container
- **docker ps -a** list containers, find CONTAINER ID
- **docker exec --it** run a process in an existing container
- **docker stop** stop a container
- **docker kill** kill a container
- **docker help** list available commands
- **man docker** Docker Linux manual pages

15

Docker CLI

```
Commands:
  attach      Attach local standard input, output, and error streams to a running container
  build       Build an image from a Dockerfile
  commit      Create a new image from a container's changes
  cp          Copy files/folders between a container and the local filesystem
  create      Create a new container
  deploy      Deploy a new stack or update an existing stack
  diff        Inspect changes to files or directories on a container's filesystem
  events      Get real time events from the server
  exec        Run a command in a running container
  export      Export a container's filesystem as a tar archive
  history     Show the history of an image
  images      List images
  import      Import the contents from a tarball to create a filesystem image
  info        Display system-wide information
  inspect     Return low-level information on Docker objects
  kill        Kill one or more running containers
  load        Load an image from a tar archive or STDIN
  login       Log in to a Docker registry
  logout      Log out from a Docker registry
  logs        Fetch the logs of a container
  pause       Pause all processes within one or more containers
  port        List port mappings or a specific mapping for the container
  ps          List containers
  pull        Pull an image or a repository from a registry
  push        Push an image or a repository to a registry
  rename      Rename a container
  restart     Restart one or more containers
  rm          Remove one or more containers
  rmi         Remove one or more images
  run         Run a command in a new container
  save        Save one or more images to a tar archive (streamed to STDOUT by default)
  search      Search the Docker Hub for images
  start       Start one or more stopped containers
  stats       Display a live stream of container(s) resource usage statistics
  stop        Stop one or more running containers
  tag         Create a tag TARGET_IMAGE that refers to SOURCE_IMAGE
  top         Display the running processes of a container
  unpause     Unpause all processes within one or more containers
  update      Update configuration of one or more containers
  version     Show the Docker version information
  wait        Block until one or more containers stop, then print their exit codes
```

16

TUTORIAL 7

- Tutorial introduces use of two common Linux performance benchmark applications
- stress-ng
- 100s of CPU, memory, disk, network stress tests
- Sysbench
- Used in tutorial for memory stress test

17

OBJECTIVES – 11/12

- Questions from 11/7
- Tutorials Questions
- **Class Presentations: Cloud Technology or Research Paper Review**
- Quiz 1 Update
- GraphQL
- Tutorial 5 Demo
- Tutorial 6 Demo

18

GROUP PRESENTATION

- TWO OPTIONS:
- ***Cloud technology presentation***
- ***Cloud research paper presentation***
  - Recent & suggested papers will be posted at: http://faculty.washington.edu/wlloyd/courses/tcss562/papers/
- Submit presentation type and topics (paper or technology) with desired dates of presentation via Canvas by: ***Sunday November 17th @ 11:59pm***
- Presentation dates:
  - Tuesday November 26
  - Tuesday December 3*, Thursday December 5
    - ** - day of quiz 2. only 1 presentation slot*

19

OBJECTIVES – 11/12

- Questions from 11/7
- Tutorials Questions
- Class Presentations: Cloud Technology or Research Paper Review
- Quiz 1 Update
- GraphQL
- Tutorial 5 Demo
- Tutorial 6 Demo

20

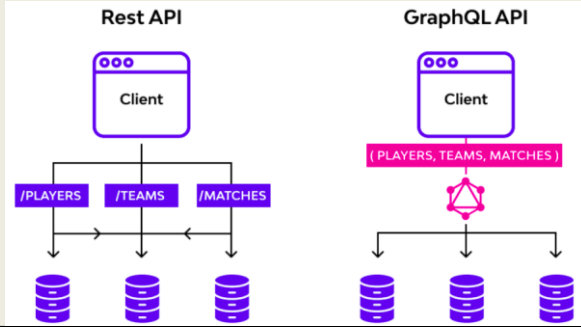
OBJECTIVES – 11/12

- Questions from 11/7
- Tutorials Questions
- Class Presentations:
  Cloud Technology or Research Paper Review
- Quiz 1 Update
- **GraphQL**
- Tutorial 5 Demo
- Tutorial 6 Demo

21

GRAPHQL

- **GraphQL** provides an alternative to RESTful APIs
- Instead of requiring a remote client to call multiple endpoints to obtain separate data, a GraphQL server provides a single aggregated (combined) endpoint and responds with precisely the data a client asks for.



22

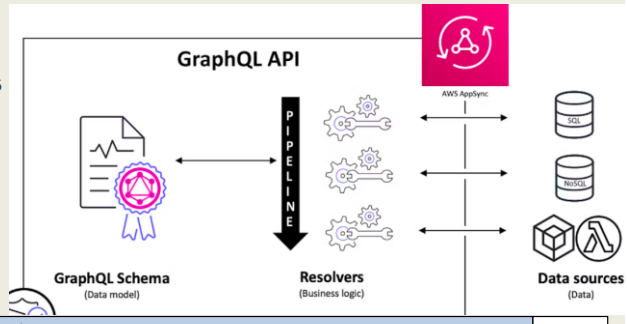
GRAPHQL - 2

- GraphQL is responsible for fetching data that a client requests from multiple databases, storage engines, or services (including FaaS AWS Lambda functions)
- For remote mobile clients, reduces the # of client service calls (round-trips) by aggregating them together
  - Only pay for the cost of network latency between the client and server one time
- Initially created by Facebook in 2012, and released as open source in 2015
- Supports reading and writing data, and also subscribing to updates

23

GRAPHQL - 3

- GraphQL service consists of **types** with **fields** then provides functions to resolve data for each field
- The collection of types and fields is the **schema** definition
- Functions that retrieve and map data are called **resolvers**
- Data sources: SQL, NoSQL, services, Lambdas



24

GRAPHQL - 4

- GraphQL requires a server to implement schemas using resolvers
- The server is typically hosted in the cloud near the databases and services
- Open source:
  - Apollo Server – build and run GraphQL APIs w/ Node.js
  - Express GraphQL – also Node.js based
  - Hot Chocolate – create GraphQL APIs for .NET
- Managed solutions:
  - AWS AppSync – ***Amazon's managed GraphQL service***
  - Google Apigee
  - Azure API Management
  - IBM API Connect

GRAPHQL - 5

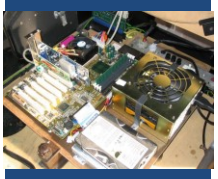
- Managed GraphQL services or Open Source GraphQL servers would be good topics for the Cloud Technology Presentation
- For the standard TLQ term project, a GraphQL could be developed against the database
- Using 5 separate 'Q'uery Lambda functions different data could be fetched from the sales or medical records datasets
- A schema could be defined to fetch all 5 elements as a single query
- Client performance could be compared for fetching the data using separate REST/Lambda calls vs. an aggregated GraphQL API

OBJECTIVES – 11/12

- Questions from 11/7
- Tutorials Questions
- Class Presentations:
  Cloud Technology or Research Paper Review
- Quiz 1 Update
- GraphQL
- **Tutorial 5 Demo**
- Tutorial 6 Demo

27

WE WILL RETURN AT ~4:50 PM



28

OBJECTIVES – 11/12

- Questions from 11/7
- Tutorials Questions
- Class Presentations:
  Cloud Technology or Research Paper Review
- Quiz 1 Update
- GraphQL
- Tutorial 5 Demo
- **Tutorial 6 Demo**

29

QUESTIONS

30