TCSS 462/562: (SOFTWARE ENGINEERING FOR) CLOUD COMPUTING

Introduction

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

1

OBJECTIVES – 9/29

- **Syllabus**
- Course Introduction
- Demographics Survey
- AWS Cloud Credits Survey
- Tutorial 1 – Intro to Linux
- Cloud Computing – How did we get here? (10/4)
  Chapter 4 Marinescu 2nd edition:
  Introduction to parallel and distributed systems

2

TCSS562 – SOFTWARE ENGINEERING FOR CLOUD COMPUTING

- Course webpage is embedded into Canvas
  - In CANVAS to access links:
    RIGHT-CLICK – Open in new window
- Syllabus online at:
  http://faculty.washington.edu/wlloyd/courses/tcss562/
- Grading
- Schedule
- Assignments

3

OBJECTIVES – 9/29

- Syllabus
- **Course Introduction**
- Demographics Survey
- AWS Cloud Credits Survey
- Tutorial 1 – Intro to Linux
- Cloud Computing – How did we get here?
  Chapter 4 Marinescu 2nd edition:
  Introduction to parallel and distributed systems

4

TCSS 462/562 – Fall 2022

- ***In-Person (UWT BHS 104)***
  Live Streamed on Zoom
- *Class sessions are streamed LIVE and recorded for 24/7 availability*
  - *UW deletes content after ~90 days*
- 20 class meetings
  - 1 Holiday: No Class on Nov 24
- This course will not have exams
- This course helps with preparation for TCSS 558 – Applied Distributed Computing

TCSS 462/
TCSS 562
FALL 2022

L1.5

5

REFERENCES

- [1] Cloud Computing: Concepts, Technology and Architecture*
  Thomas Erl, Prentice Hall 2013
- [2] Cloud Computing - Theory and Practice
  Dan Marinescu, First Edition 2013 *, Second Edition 2018
- [3] Cloud Computing: A Hands-On Approach
  Arshdeep Bahga 2013

* *- available online via UW library*

6

REFERENCES - 2

- [4] Systems Performance: Enterprise and the Cloud *
- Brendan Gregg, First Edition 2013
- [5] AWS Administration – The Definitive Guide *
- Yohan Wadia, First Edition 2016
- Research papers

*- available online via UW library

7

TCS462/562 COURSE WORK

- **Project Proposal**
- **Project Status Reports / Activities**
  - ~ 2-4 total items (??)
  - Variety of formats: in class, online, reading, activity
- **Quizzes**
  - Open book, note, etc.
- **Class Presentation (TCSS 562)**
- **Class Presentation Summaries (TCSS 462)**
- **Term Project / Paper / Presentation**

8

TERM PROJECT

- Project description to be posted
- Teams of ~4, self formed, one project leader
- Project scope can vary based on team size and background w/ instructor approval
- Proposal due: Tuesday October 18, 11:59pm (tentative)
- Approach:
  - Build a "cloud native" serverless application
    - App will consist of multiple FaaS functions (services)
    - Objective is to compare outcomes of design trade-offs
      - Performance (runtime)
      - Cost ($)
  - How does application design impact cost and performance?

9

TERM PROJECT - 2

- GOAL: Compare implementations with alternate:
  - Different service compositions / services
  - Different external services (e.g. database, key-value store)
  - Application control flow - AWS Step Functions, laptop client, etc.
- A & B Testing
  - As developers it is common to implement a system or algorithm multiple ways
  - But which implementation is more effective for a given set of goals, objectives, metrics?
- WHAT are some metrics that would be interesting to compare?

10

TERM PROJECT - 3

- Deliverables
  - Short Demo in class at end of quarter (< 5 min)
  - Project report paper (4-6 pgs IEEE format, template provided)
  - GitHub (project source)
  - How-To document (via GitHub markdown)
- Standard project(s) will be suggested or propose your own:
  - (Example) Extract-Transform-Load (ETL) style serverless data processing pipeline combing AWS Lambda, S3, and Amazon Aurora Serverless DB

11

COMPARING DESIGN TRADE-OFFS

- **What design trade-offs can be compared?**
- Compare and contrast alternative designs using various cloud services, languages, platforms, etc.
- Examples – Compare different:
- Cloud storage services: Object/blob storage services
  - Amazon S3, Google blobstore, Azure blobstore, vs. self-hosted
- Cloud relational database services:
  - Amazon Relational Database Service (RDS), Aurora, Self-Hosted DB
- Platform-as-a-Service (PaaS) alternatives:
  - Amazon Elastic Beanstalk, Heroku, others
- Open source FaaS platforms
  - Apache OpenWhisk, OpenFaaS, Fn, others…

12

COMPARING DESIGN TRADE-OFFS - 2

- Serverless storage alternatives
  - From AWS Lambda: Amazon EFS, S3, Containers, others
- Container platforms
  - Amazon ECS/Fargate, AKS, Azure Kubernetes, Self-hosted Kubernetes cluster on cloud VMs
- Contrasting queueing service alternatives
  - Amazon SQS, Amazon MQ, Apache Kafka, RabbitMQ, 0mq, others
- NoSQL database services
  - DynamoDB, Google BigTable, MongoDB, Cassandra
- CPU architectures
  - Intel (x86_64), AMD (x86_64), ARM (Graviton), MAC (M1)
- Service designs or compositions

13

TERM PROJECT: BIG PICTURE

1. BUILD A MULTI-FUNCTION SERVERLESS APPLICATION
   - *Typically consisting of AWS Lambda Functions or Google Cloud Functions, etc. (e.g. FaaS platfrom)*
2. CONTRAST THE USE OF ALTERNATIVE CLOUD SERVICES TO INSTRUMENT SOME OR MULTIPLE ASPECTS OF THE APPLICATION
3. CONDUCT A PERFORMANCE EVALUATION, REPORT ON YOUR FINDINGS IN A LIGHTNING TALK (5-minutes) AND TERM PAPER

14

TERM PROJECT - KEY REQUIREMENTS

- Application should involve multiple processing steps
- Implementation does not have to be Function-as-a-Service (FaaS)
- Implementation leverges external services (e.g. databases, object stores, queues)
- Projects will contrast alternate designs
- Define your comparison metrics:
- Which designs offer the ***fastest performance (runtime)***?
- ***Lowest cost ($)***?
- ***Best maintainability***?
  Consider size, lines of code (LOC), smaller programs are generally considered to be easier to maintain

15

TERM PROJECT: RESEARCH

- **Alternative I**: conduct a cloud-related research project on any topic focused on specific research goals / questions
  - Can be used to help spur MS Capstone/Thesis projects or honors thesis
  - If you're interested in this option, please talk with the instructor
  - First step is to identify 1 – 2 research questions
- **Alternative II**: conduct a gap-analysis literature survey of cloud computing research papers, produce a report which identifies open problems for future research in cloud computing that have tractable next steps
  - Suitable for 1-person teams and students interested in research
- Instructor will help guide projects throughout the quarter

16

PROJECT SUPPORT

- Project cloud infrastructure support:
- **Standard AWS Account (RECOMMENDED)**
  - Create standard AWS account with UW email
  - Credit card required
  - Instructor provides students with $50 credit vouchers from AWS
  - When voucher is used up, request another voucher from instructor
  - Credits provided throughout Fall quarter (within reason)
- **Instructor provided IAM AWS Account**
  - No Credit Card required
  - Instructor creates and manages account security and permissions
  - More restricted

17

PROJECT SUPPORT - 2

- **Other Support :**
- **Github Student Developer Pack:**
  - https://education.github.com/pack
  - Formerly offered AWS credits, but Microsoft bought GitHub
  - Includes up to $200 in Digital Ocean Credits
  - Includes up to $100 in Microsoft Azure Credits
  - Unlimited private git repositories
  - Several other benefits
- **Microsoft Azure for Students**
  - $100 free credit per account valid for 1 year – no credit card (?)
  - https://azure.microsoft.com/en-us/free/students/
- **Google Cloud**
  - $300 free credit for 1 year
  - https://cloud.google.com/free/
- **Chameleon / CloudLab**
  - Bare metal NSF cloud - free

18

TCSS562 TERM PROJECT OPPORTUNITIES

- Projects can lead to papers or posters presented at ACM/IEEE/USENIX conferences, workshops
  - Networking and research opportunity
    - *… travel ???*
  - Conference participation (posters, papers) helps differentiate your resume/CV from others
- Project can support preliminary work for: UWT - MS capstone/thesis project proposals
- Research projects provide valuable practicum experience with cloud systems analysis, prototyping
- Publications are key for building your resume/CV, Also very important for applying to PhD programs

19

TCSS562 TERM PROJECT - 3

- Project status report / term project check-ins
  - Written status report
  - ~2 reports during the quarter
  - Part of: ***"Project Status Reports / Activities / Quizzes"*** category
  - 10% of grade
- Project meetings with instructor
  - After class, end half of class, office hours

20

TCSS 562: CLASS PRESENTATION

- TCSS 562 students will give a team presentation **teams of ~3**
- **Technology sharing presentation**
  - PPT Slides, demonstration
  - Provide technology overview of one cloud service offering
  - Present overview of features, performance, etc.
- **Cloud Research Paper Presentation**
  - PPT slides, identify research contributions, strengths and weaknesses of paper, possible areas for future work

21

CLASS PRESENTATION PEER REVIEWS

- Students will submit reviews of class presentations using rubric worksheet (~ 1-page)
- Students will review a minimum of one presentation for each presentation day, for a minimum of 4 reviews
  - *Optionally additional reviews can be submitted (Extra Credit)*
- In addition to the reviews, students will write two questions about content in the presentation. These can be questions to help clarify content from the presentation that was not clear, or any related questions inspired by the presentation.
- To ensure intellectual depth of questions, questions should not have yes-no answers.
- Peer reviews will be shared with presentation groups to provide feedback but **will not** factor into the grading of class presentations

22

CLASS PRESENTATION PEER REVIEWS – EXTRA CREDIT

- Students submitting more than 4 peer reviews of presentations will be eligible for extra credit at the end of the quarter
  - Extra credit will be: (#-of-extra-reviews / (num-of-presentations – 4)) * 2% (up to 2% added to the overall course grade)
- For TCSS 462 – the peer reviews will count for the entire presentation score
- For TCSS 562 – the peer reviews will count as ~20% of the entire presentation score

23

OBJECTIVES – 9/29

- Syllabus
- Course Introduction
- **Demographics Survey**
- AWS Cloud Credits Survey
- Tutorial 1 – Intro to Linux
- Cloud Computing – How did we get here? Chapter 4 Marinescu 2nd edition: Introduction to parallel and distributed systems

24

DEMOGRAPHICS SURVEY

- Please complete the ONLINE demographics survey:
- https://forms.gle/XAhBRUR8wsm7CqSs5
- Linked from course webpage in Canvas:
- http://faculty.washington.edu/wlloyd/courses/tcss562/announcements.html

25

OBJECTIVES – 9/29

- Syllabus
- Course Introduction
- Demographics Survey
- **AWS Cloud Credits Survey**
- Tutorial 1 – Intro to Linux
- Cloud Computing – How did we get here?
  Chapter 4 Marinescu 2nd edition:
  Introduction to parallel and distributed systems

26

AWS CLOUD CREDITS SURVEY

- Please complete the ONLINE demographics survey:
- https://forms.gle/yz8yrqB7yGD5iHSh9
- Linked from course webpage in Canvas:
- http://faculty.washington.edu/wlloyd/courses/tcss562/announcements.html

27

OBJECTIVES – 9/29

- Course Introduction
- Syllabus
- Demographics Survey
- AWS Cloud Credits Survey
- **Tutorial 1 – Intro to Linux**
- Cloud Computing – How did we get here?
  Chapter 4 Marinescu 2nd edition:
  Introduction to parallel and distributed systems

28

QUESTIONS

29

SEE YOU TUESDAY!

OCTOBER 4TH
5:50 PM
BHS 104

30