



CS4740 CLOUD COMPUTING

Course Introduction

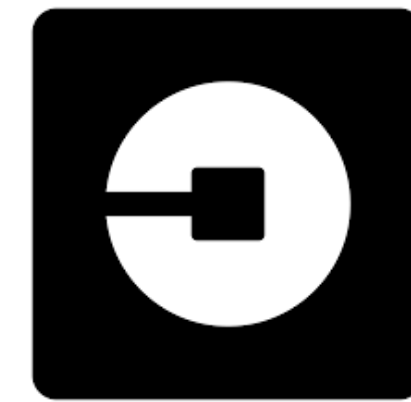
Prof. Chang Lou, UVA CS, Fall 2026

AGENDA

- **What this course is about**
 - what to expect from this course
 - **How to get the most from this course**
 - .. and what we expect from you
 - **But first, tell us about yourself**
 - who you are, and why you chose this course
-

WHY STUDY CLOUD COMPUTING

— Cloud services are the foundation for modern society.



WHY STUDY CLOUD COMPUTING

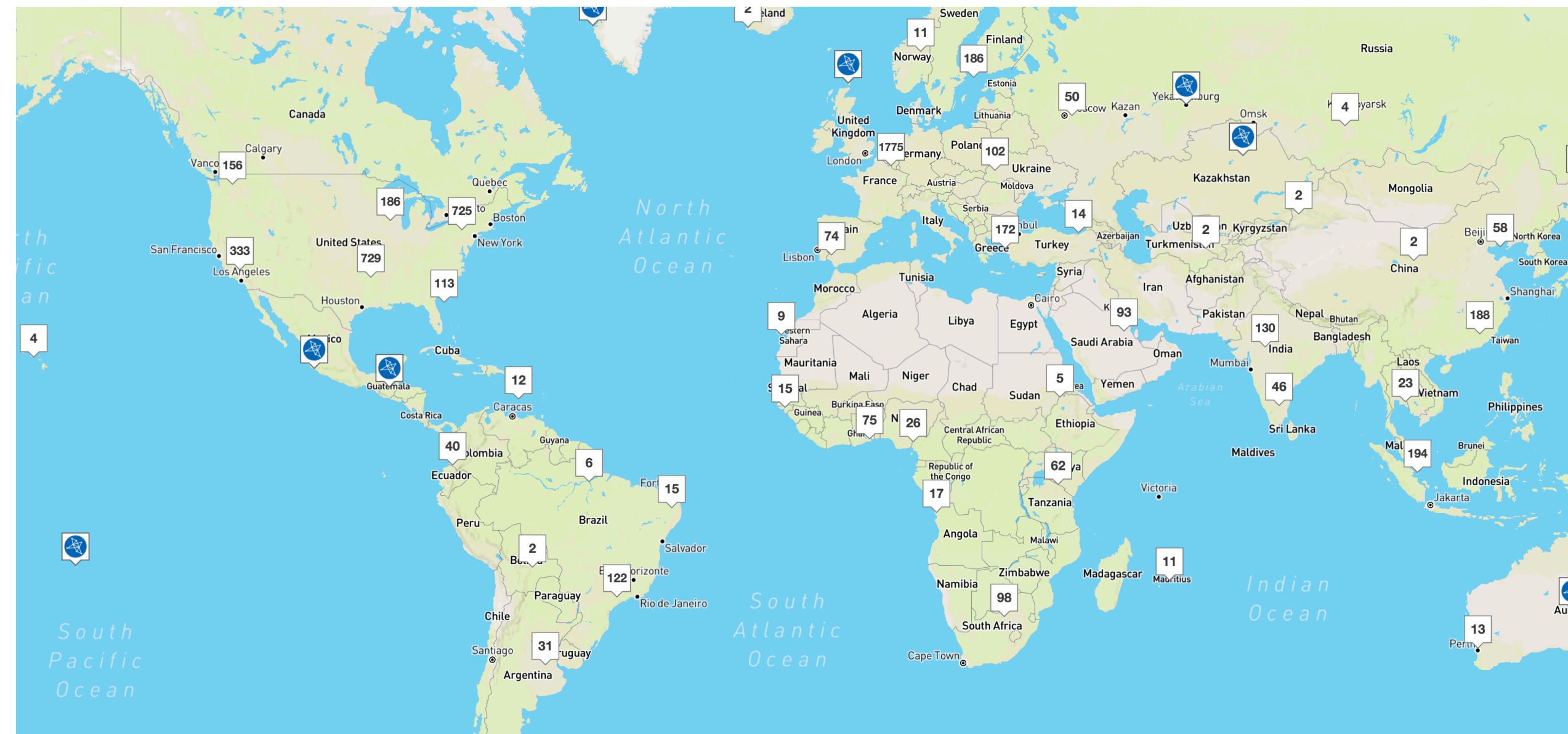
— A day without cloud?

In 2026, 94% of enterprises use cloud services!



WHY STUDY CLOUD COMPUTING

- **Rising needs for cloud computing**
 - Today: cloud market is \$676B (expected to reach \$2.2T by 2032)
 - There is a high demand for talents on the market

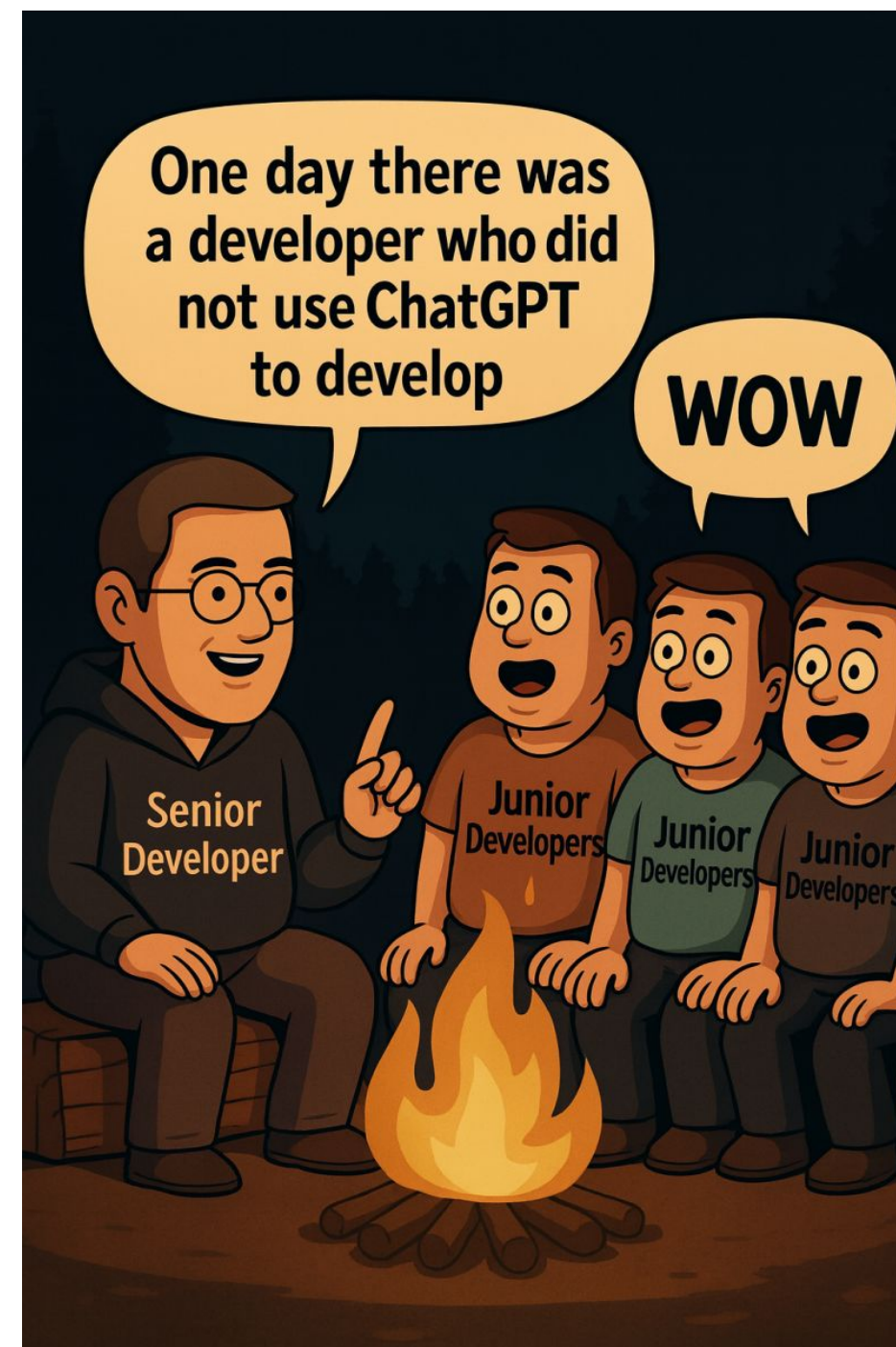


AMAZON HQ2 IN ARLINGTON, VIRGINIA



WHY STUDY CLOUD COMPUTING

— AI can now easily generate millions of lines of code...



WHY STUDY CLOUD COMPUTING

— From code to infrastructure

- Scalability: How do we turn programs into infrastructure that serves the world?

How to scale systems to handle requests from millions of users?

How to make thousands of servers behave like one single machine?

— From implementation to trust

- Reliability: How do we know that infrastructure actually works?

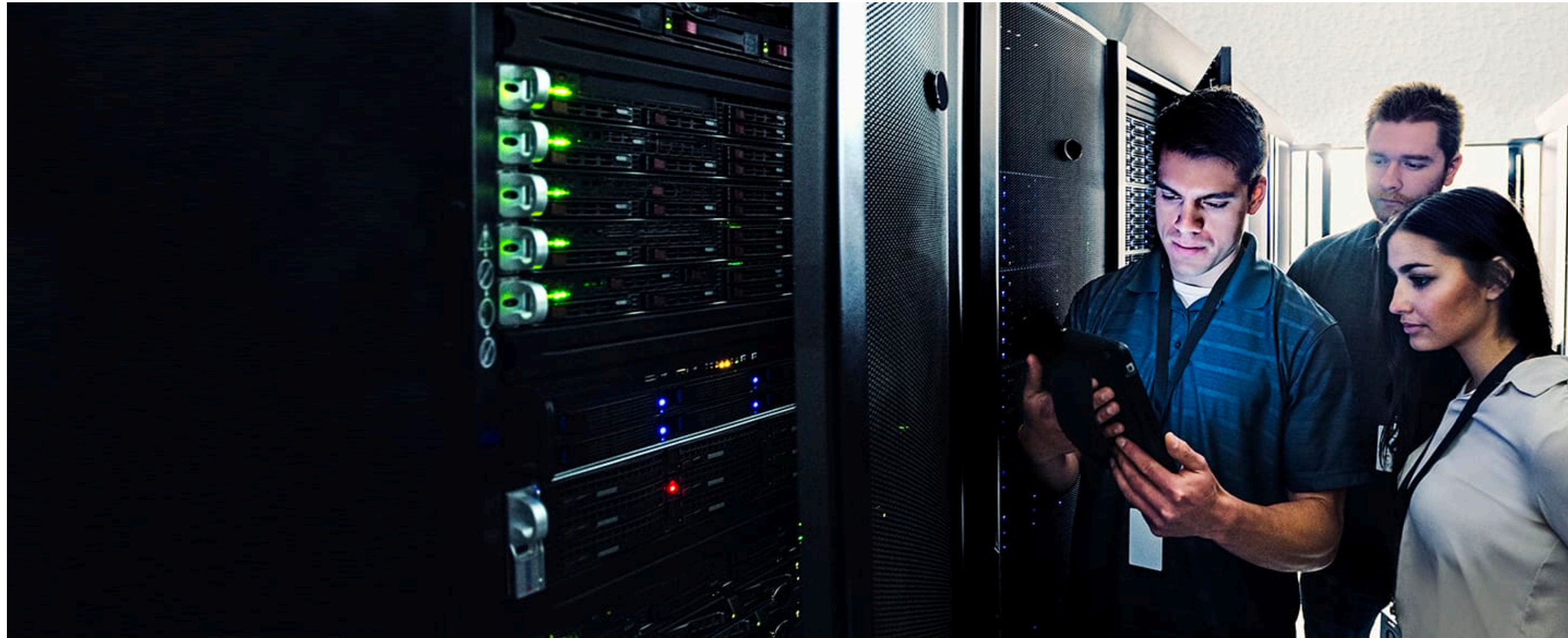
How to keep systems running despite constant failures?

How to ensure systems behave correctly?

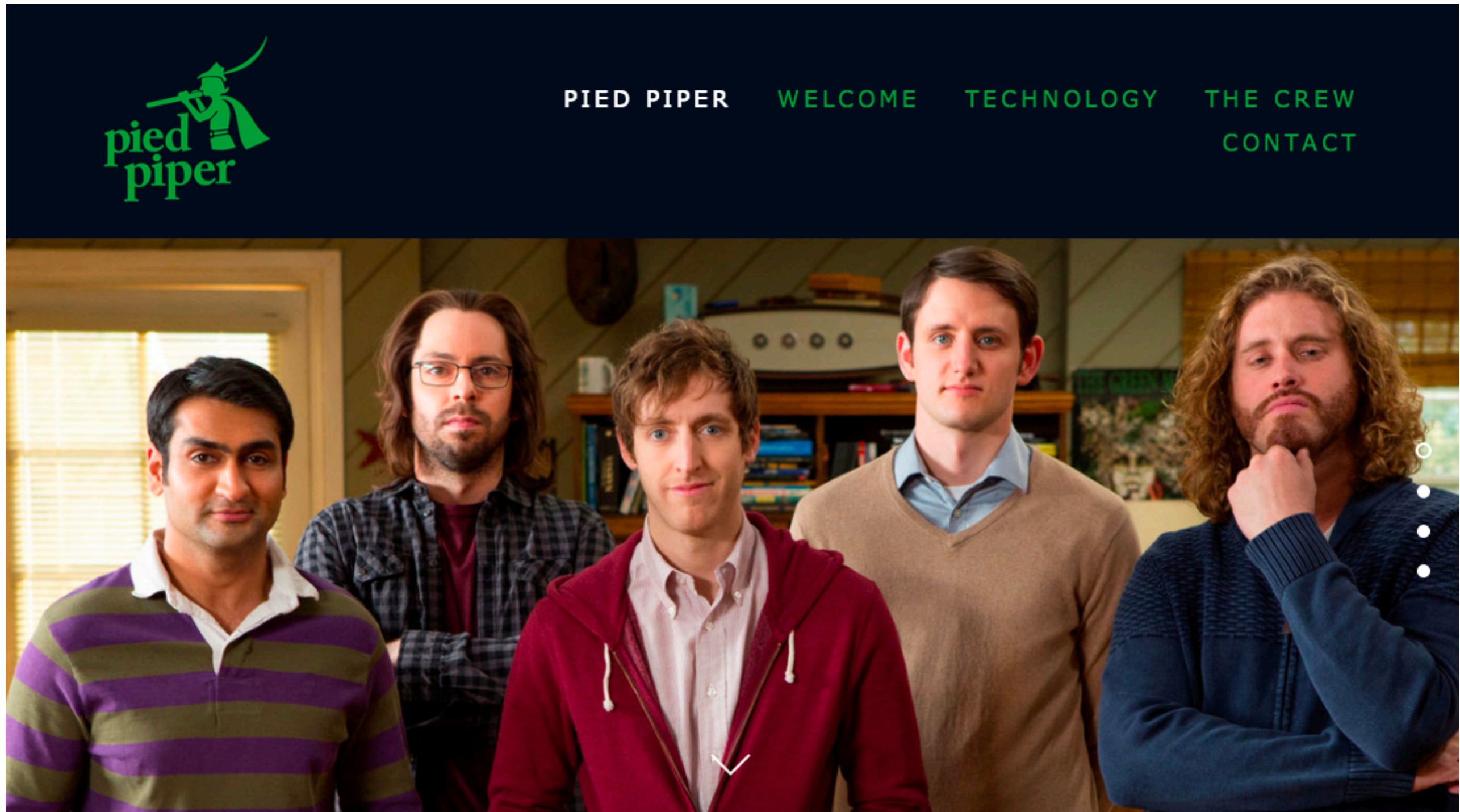
WHAT WE FOCUS IN THIS COURSE

- Learn AWS toolkit?
 - Design cloud apps?
 - Important to know how to **use** cloud services
 - many related courses provided in UVA
 - **In this course we focus on how to **build** cloud systems**
-

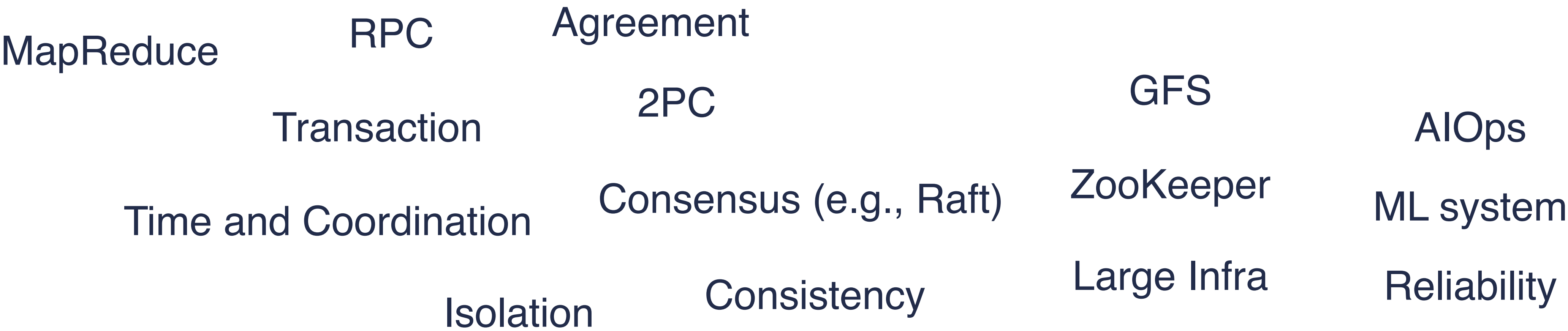
WHAT WE FOCUS IN THIS COURSE



WHAT WE FOCUS IN THIS COURSE



TOPICS (TENTATIVE)



Cloud and Distributed System Fundamentals

Real-world Cloud

Special Topics

WHAT WE COVER IN THIS COURSE

- Foundational concepts of cloud systems: **distributed systems**
 - challenges, algorithms, techniques, abstractions
 - The inner-workings of large **cloud infrastructure**.
 - e.g., Google's protobuf/Spanner/MapReduce, Yahoo's Hadoop, Amazon's Dynamo, etc.
 - specifically, play and hack one of the most popular systems, ZooKeeper, on Lab Day
 - Building small cloud systems yourself.
 - a MapReduce system that scales computing workloads
 - a Raft-based system that tolerates failures on any single node
-

Course Information

LECTURER INFO

— Prof. Chang Lou

- Ph.D., Johns Hopkins
- Experience with Cloud Infrastructure in Production

— Research Interests

- Cloud/Distributed Systems, Software Reliability

— Contact

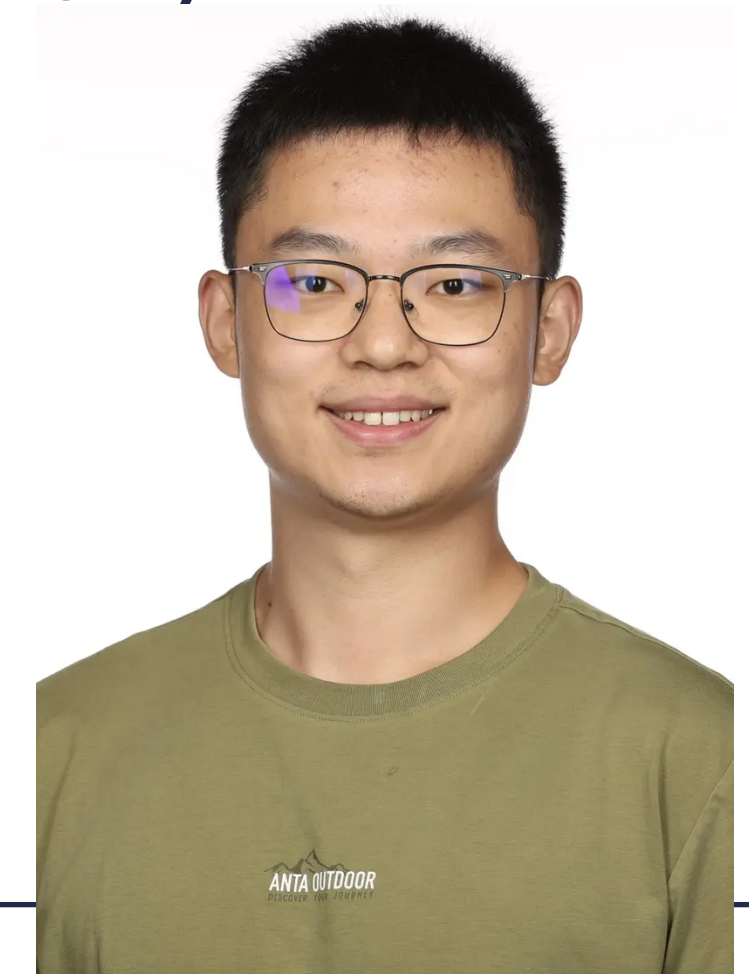
- Office: Rice Hall 304
- Office Hours: MoWe 3:30-4:30pm (drop-ins welcome; heads-up email appreciated)
- Email address: chlou@virginia.edu



TEACHING STAFF

— TAs

- Kahfi Soobhan Zulkifli (kwf3wv@virginia.edu), office hour: Tuesdays 3-4pm, Rice 442
- Dimas Parikesit (vqx2dc@virginia.edu), office hour: Thursday 3-4 pm, Rice 442
- Jinhao Liang (njs4nu@virginia.edu), office hour: Friday 4–5pm, Rice 442
- **cs4740staff@virginia.edu** (preferred email channel for the course staff)



IMPORTANT LINKS

- Website: <https://chenglousys.github.io/CS4740/fall26/about/>
 - Questions & Discussions:
 - Lecture related questions: Piazza, Lecturer office hours
 - Lab related questions: Piazza, TA office hours
 - Important links (available on course website, including this slide):
 - Piazza (discussion): <https://piazza.com/virginia/fall2026/cs4740>
 - Canvas (course announcement): <https://canvas.its.virginia.edu/courses/190112>
 - Github Repo Link for Project: <https://forms.gle/zbtvrKtpmYmETWoo7>
-

LAB

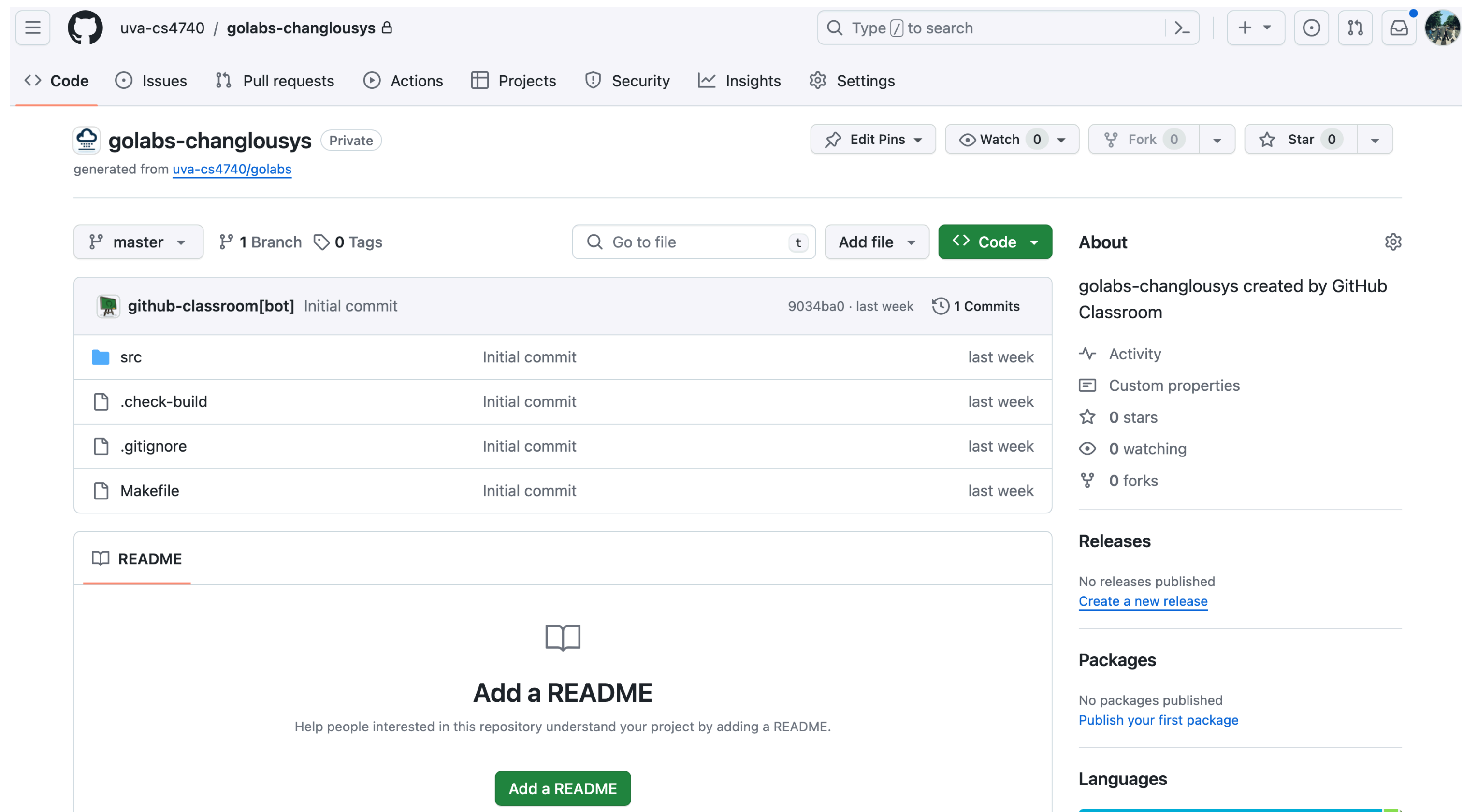
- Sad news: Github classroom is decommissioned
- Instead, submit your GitHub userid to this form and we will create a managed repo for you: <https://forms.gle/zbtvrKtpmYmETWoo7>
- Our TAs pull the list on a daily basis, if you haven't gotten your repo created, email the team

The screenshot shows a GitHub repository page for 'uva-cs4740-fall2025-golabs-golabs'. The repository is marked as 'Private' and was generated from 'uva-cs4740/golabs'. It has 1 branch (master) and 0 tags. The commit history shows an initial commit by 'github-classroom[bot]' 3 minutes ago, with a commit hash of 0e4c19a. The commit includes three files: 'src', '.check-build', and '.gitignore', all marked as 'Initial commit'.

File	Commit Message	Time
src	Initial commit	3 minutes ago
.check-build	Initial commit	3 minutes ago
.gitignore	Initial commit	3 minutes ago

LAB

- We automatically collect your codes by taking a snapshot of your repo before deadline (unless late tokens are used)
- all you need is to push your local commits



PREREQUISITES

- **Solid programming experience** (C, C++, Java), preferably system-level programming experience.
 - Do you enjoy debugging? :)
 - UVA courses:
 - CS 2130 (Computer Systems and Organization)
 - If you lack these prerequisites, taking this class will be very difficult, because heavy **coding** accounts for vast portion of grade.
 - Use Lab 0 to determine if you have sufficient experience.
-

LATE POLICIES

- What if I have emergencies?
 - Everyone has 72 hour late tokens on projects
 - If you decide to use the late hour tokens, by the deadline send an email to `cs4740staff@virginia.edu` with the subject “[Late Request]: \$GitHub_Repo_Name\$” (empty content is fine) so we won’t be collecting and grading your solution immediately.
 - When you finish (within the token limit), send another email to `cs4740staff@virginia.edu` with the subject “[Late Finish]: \$GitHub_Repo_Name\$”.
-

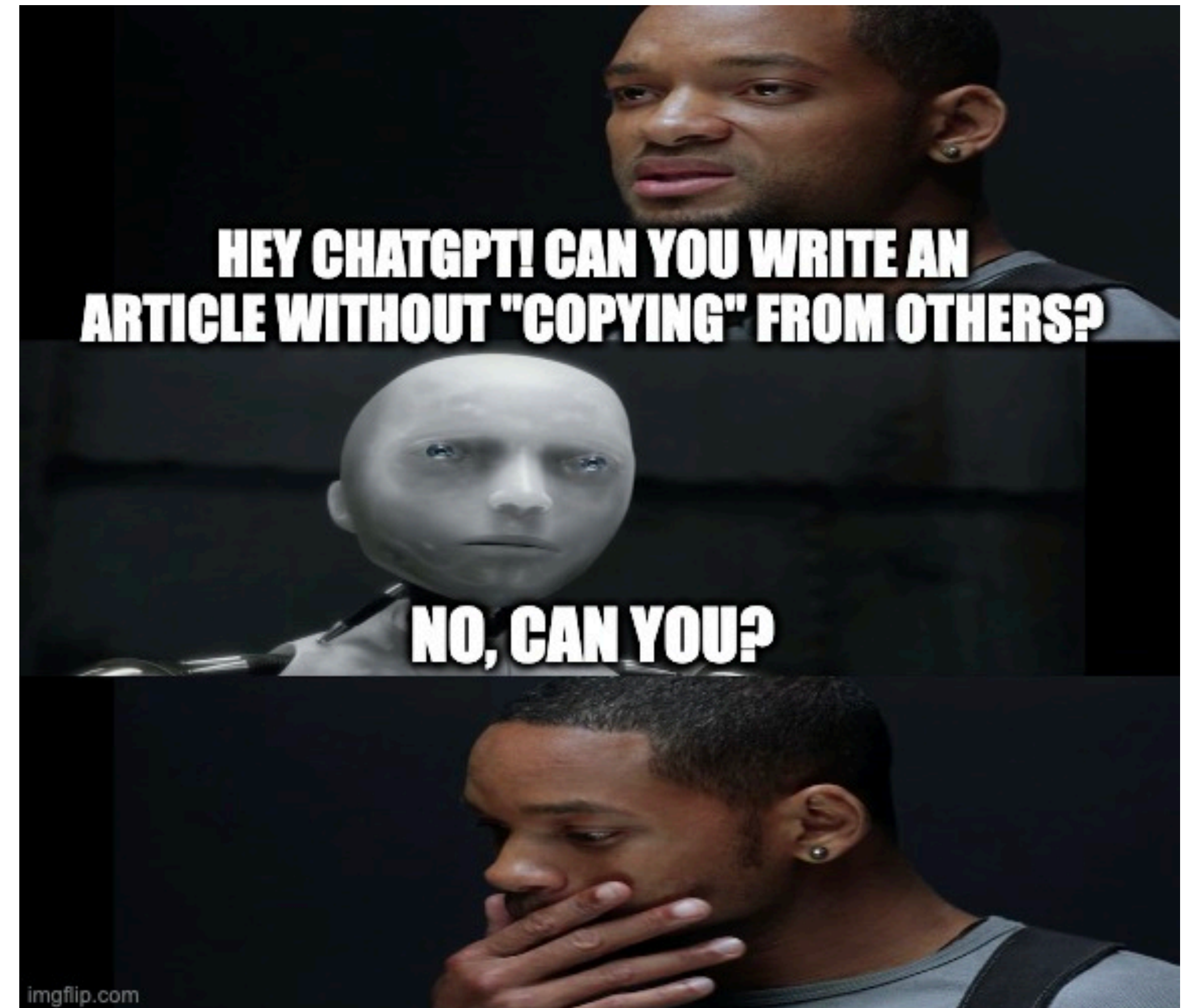
CHEATING POLICIES

- You can discuss, preferably openly on Piazza, but NOT look at or copy someone else's code!
 - “Someone else's code” could be: from a present colleague, from a past student (at UVA or somewhere else), from the Internet.
 - We will be checking all code for cheating and may do oral examinations for individuals.
-

AI TOOL POLICY

- **How to best use AI tools**
 - understand error messages
 - brainstorm design/debugging tips
- **You should not use AI to**
 - directly generate codes for you

It is a violation of academic integrity as well.



GRADING

- Midterm exam: 20% + 10% **(project question)**
 - Final exam: 20% + 10% **(project question)**
 - Projects: 40%
 - Two labs: based on MIT's 6.584 series
 - Assignments are written in Go
 - Use Github Classroom to create your repo
-

LAB 0

- Not graded but mandatory and important:
 - Sets up github, which you'll use for all homeworks.
 - Complete this homework even if you are not yet enrolled!
 - Test if you like the style of this course :)
 - Deadline: 9/4/2026 23:59:59 EST
 - Please finish it as soon as possible to weed out any issues so you can start on Lab 1 on time.
-

HOW NOT TO PASS CS 4740?

- Do not come to lecture
 - The slides and labs are online anyway
 - Lecture walks you through difficult materials and tells you the context
 - Do not ask questions in lecture, office hours or online
 - It's scary, I don't want to embarrass myself
 - Asking questions is the best way to clarify lecture material
 - Office hours and email will help with homework, projects
-

HOW NOT TO PASS CS 4740?

- Wait until the last couple of days to start a project
 - We'll have to do the crunch anyways, why do it early?
 - The projects cannot be done in the last few days
 - **Repeat:** The projects cannot be done in the last few days
 - (p.s. The projects cannot be done in the last few days)
-

Questions?



NEXT STEPS

- Start working on **Lab 0** today!
 - If you run into issues, we are happy to help!
- Next class: Introduction to cloud



ACKNOWLEDGEMENT

THIS COURSE IS DEVELOPED HEAVILY BASED ON COURSE MATERIALS SHARED BY PROF. INDRANIL GUPTA, PROF. ROBERT MORRIS AND PROF. ROXANA GEAMBASU. MANY APPRECIATIONS FOR GENEROUSLY SHARING THEIR MATERIALS AND TEACHING INSIGHTS.
