Deployment

17-316/616 Fall 2025

AI Tools for Software Development

https://ai-developer-tools.github.io

Austin Henley and Andrew Begel

Administrivia

- P5: Testing due tonight 11:59pm
- P6: Deployment released today, due Wed. Nov 18, 11:59pm
- Reflection Signup for Week 12 will be released later today.

Topics for Deployment

- Cloud Computing
- AWS – Amazon Web Services
  - AWS Amplify (frontend)
  - AWS Lambda (backend)
  - Continuous Deployment
- Monitoring the Deployment
- Risks to Using Cloud Services

Scenario: You started a company in 2005
(as a baby)

- You created an online store for your parents' welding supply business.

- What does your tech stack look like?

Scenario: You started a company in 2005
(as a baby)

- Hook up Internet to your house. Put a server in your garage.

- Use the LAMP stack:
  - Linux
  - Apache web server (serves directories and files)
  - MySQL database (holds welding supply info)
  - All code written in Perl (e.g., for updating inventory)

What’s nice about this?

- You can set up and run everything by yourself
- Everything is relatively simple.

What's tough about this?

- You must set up and run everything by yourself!
- How can you restart the server if you're not home?
- Who rebuilds the system when the hard drive dies?
- What if a hacker gets in and hijacks your site?
- How do you scale?
  - What if you needed more storage? bandwidth?

Enter Cloud Computing

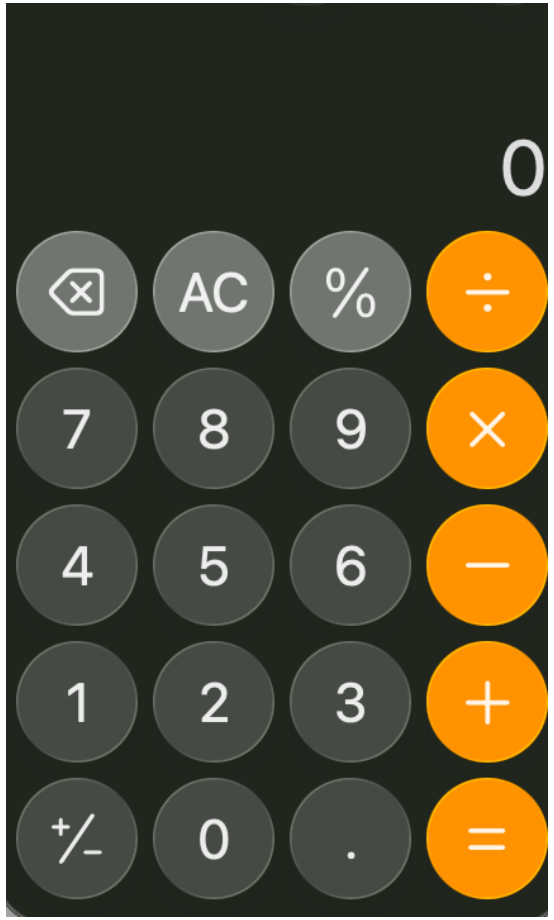
- Let's make datacenters with thousands of computers
- Someone else pays for hardware, cooling, networking infrastructure
- Someone else maintains the physical infrastructure

Selling Services

- Google offers Gmail
  - Software as a Service (SaaS)
    - They offer an application that does mail. You log in and use the app.
- Amazon offers EC2: Elastic Compute Cloud
  - Infrastructure as a Service (IaaS)
    - They offer virtual servers, networking, and storage. You do the OS and apps.
- Microsoft offers Azure
  - Platform as a Service (PaaS)
    - They offer an OS, scaling, deployment. You do apps.

Activity: Create a Calculator App

- Using Cursor, create a Node.js React app for a basic calculator.
- The frontend should contain the following buttons:
  - Numbers: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, . (decimal)
  - Functions: +, -, x, ÷, %, =, +/-, Clear, Backspace
  - It should create a string with the current expression as the user presses buttons and display it at the top.
  - Handle Clear and Backspace on the frontend.
- The backend should compute the value of the expression.
  - It has one public function: string calculate(string)
- When the user presses =, the frontend should call calculate() on the backend with the current expression and display the return value at the top of the calculator.
- Test out your calculator on localhost to make sure it works.



Activity: Deploy Calculator Frontend to Amplify

- Check your calculator app into a GitHub repo
- Connect to AWS Console (https://console.aws.amazon.com)
  - Create an AWS account if you don't have one
  - Go to AWS Amplify and connect it to your GitHub
    - Authorize AWS Amplify to use your GitHub account
    - Pick your calculator repo and branch. Ask us if you don't see it in the pulldown.
    - Check the box "My app is a monorepo."
      - Fill in the root directory of your frontend app
    - Save and deploy
- Deploy and visit the deployed URL