CLASS OVERVIEW

CSE312

1. Introduction

- 4 Total LOs
- 28 AOs
- Recitation Attendance for 2 AOs
- Recitation will Go over Homework Topics

2. Course Topics

- HTTP
- Backend API server
- Authentication
- Web sockets

3. Homework

- Creating backend API server
- Frontend is provided
- LOs required
- Autograded
- Finished Demo Site

4. Group Project

- 4-5 students per team
- **Any** Libraries and Online resources allowed
- Present at end of semester

Introduction

CSE312 you will learn about

- Full stack web development
  - Mainly backend server-side code
- Learn about HTTP
- Databases
- Secure Authentication and Authorization
- How files are handled on the web
- Websockets
- Encryption
- Proxys and more

Introduction

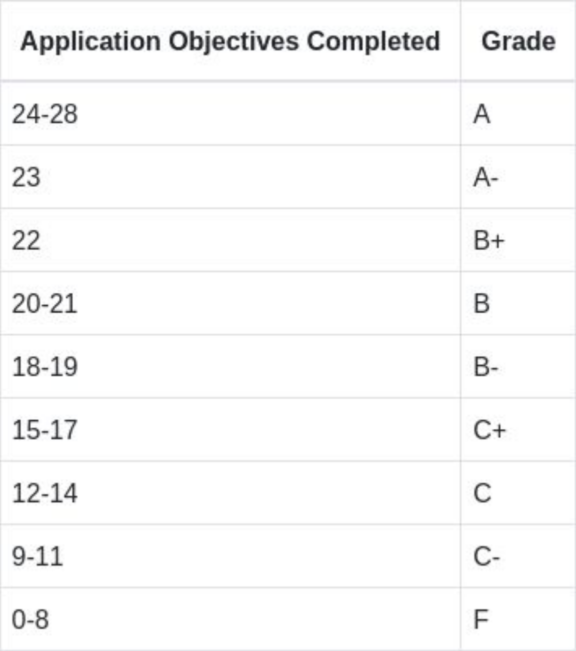
Learning Objectives

- 1 LO per HW = 4 Total LOs **(Required to pass)**

Application Objectives

- Homework
  - 16 Total Application Objectives
- Group Project
  - 8 Total Application Objectives
- Recitation Attendance
  - 2 Application Objectives
- Lecture Attendance
  - 2 Application Objectives

Total: **28 AOs**



Application Objectives Completed	Grade
24-28	A
23	A-
22	B+
20-21	B
18-19	B-
15-17	C+
12-14	C
9-11	C-
0-8	F

Introduction

Recitation

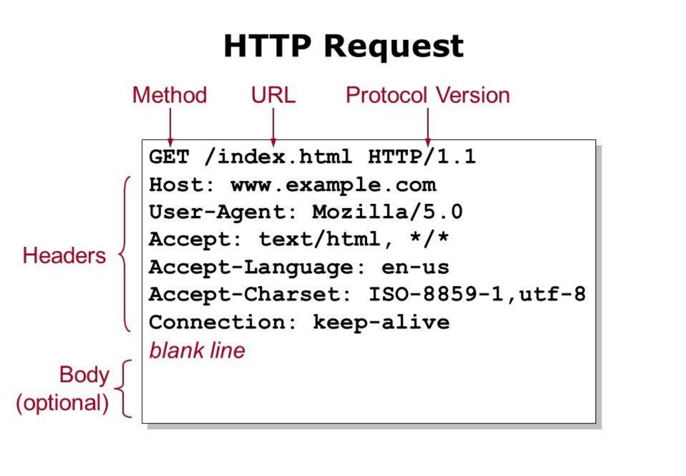
- Go over Homework / Group Project
- Explain common questions
- Attendance is required for 2 AOs
  - Taken by scanning UB card
  - Must be on time to receive attendance credit

# of recitations	**# of AOs**
10-12	**2 AOs**
8-9	**1 AOs**
0-7	**No AOs**

Course Topics

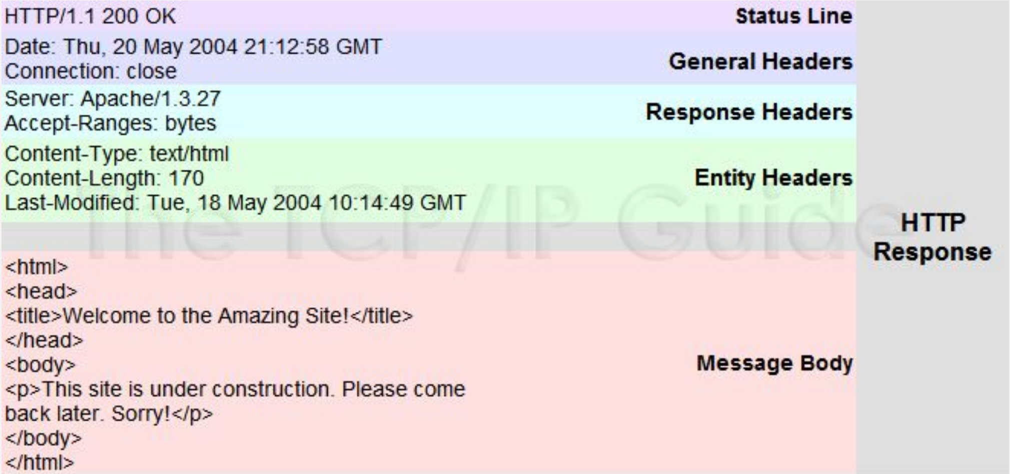
HTTP

- Backbone of the modern internet
- You will implement the protocol from scratch
- Works over TCP connection
- Stateless (except when you use cookies)
- Works via request → response (sent in plain text)



HTTP Request

```
GET /index.html HTTP/1.1
Host: www.example.com
User-Agent: Mozilla/5.0
Accept: text/html, */*
Accept-Language: en-us
Accept-Charset: ISO-8859-1,utf-8
Connection: keep-alive
blank line
```



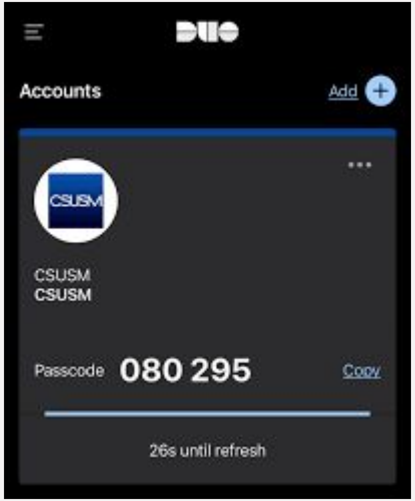
```
HTTP/1.1 200 OK
Date: Thu, 20 May 2004 21:12:58 GMT
Connection: close
Server: Apache/1.3.27
Accept-Ranges: bytes
Content-Type: text/html
Content-Length: 170
Last-Modified: Tue, 18 May 2004 10:14:49 GMT

<html>
<head>
<title>Welcome to the Amazing Site!</title>
</head>
<body>
<p>This site is under construction. Please come
back later. Sorry!</p>
</body>
</html>
```

Course Topics

Authentication

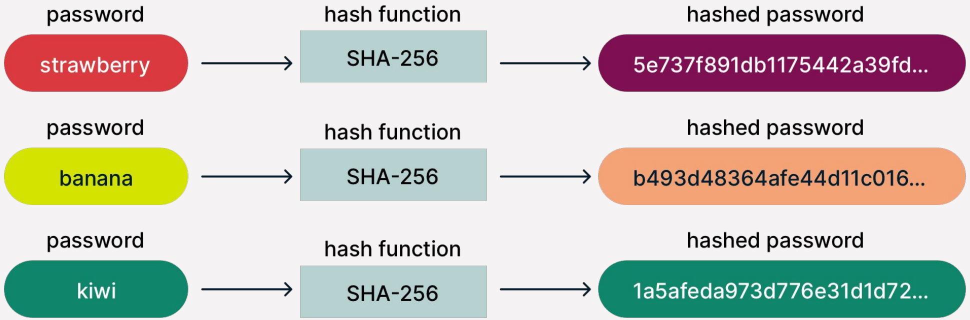
- How to prove a user is who they say they are
- Implementing proper cryptographically safe storage
  - Hashing passwords
- OAuth (google sign in)
- 2FA (TOTP)



CSUSM
CSUSM

Passcode 080 295

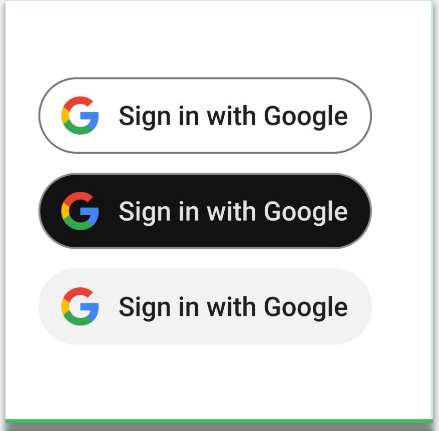
26s until refresh



password: strawberry → hash function: SHA-256 → hashed password: 5e737f891db1175442a39fd...

password: banana → hash function: SHA-256 → hashed password: b493d48364afe44d11c016...

password: kiwi → hash function: SHA-256 → hashed password: 1a5afeda973d776e31d1d72...



Course Topics

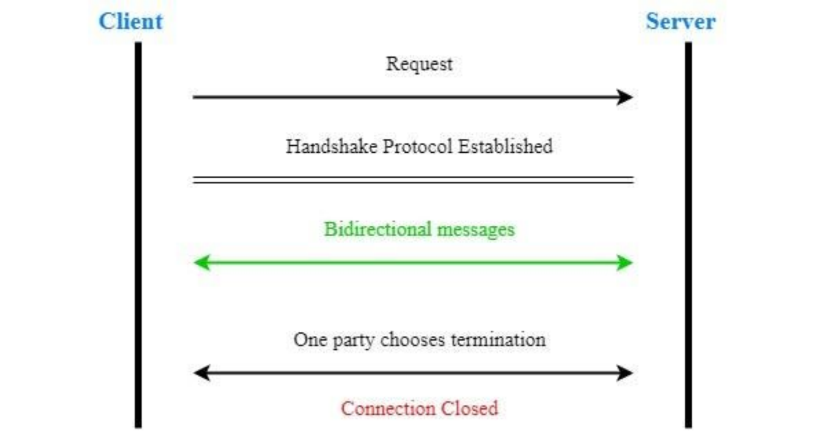
File handling

- How to properly handle file uploads
  - How to trust is file user says it is
  - How to store and distribute to other users
- Implementing file uploads
  - How to handle large uploads, store proper file type

Course Topics

Web Sockets

- Not limited to Request, Response like HTTP
  - Allows server to send messages to clients (without request)
- Where you want real-time updates without polling
  - Twitch chat, web based games
- Implementing protocol from scratch



Homework

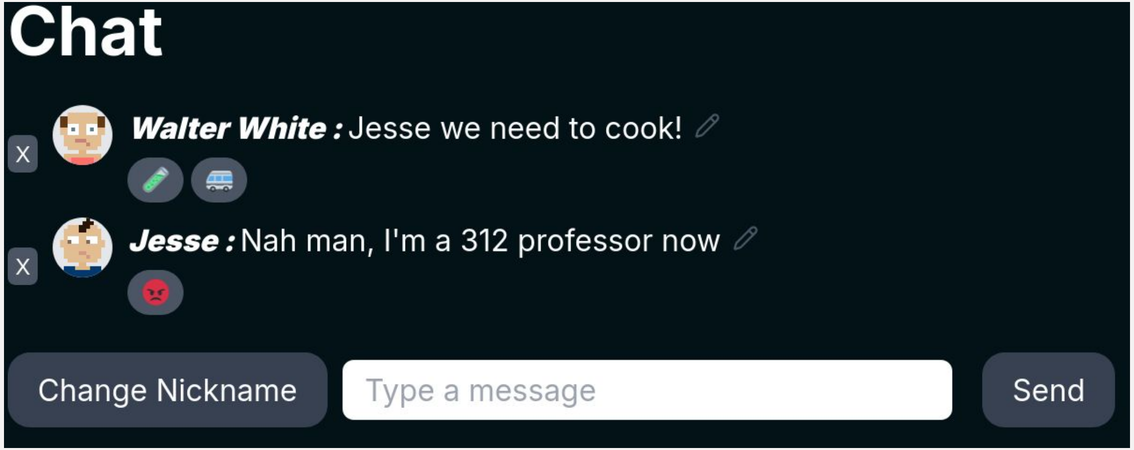
HW Structure

- Creating backend framework
  - Real world this is done for you
- Frontend is provided in GitHub repo
- Learning Objectives are required to pass **(1 per HW)**
  - Will be autograded
- Application Objectives **(3 per HW)**
  - Increase in difficulty (1st easier than 3rd)
  - Third AO, meant to be a challenge **(only 24 AOs needed for A)**
- No outside resources or libraries allowed
- Homework will be written in python
  - Any IDE can be used

Homework

HW #1 - Chat application and basic HTTP server

- Serving and handling HTTP requests
- Multiple chat users
- Reactions with emojis



Chat

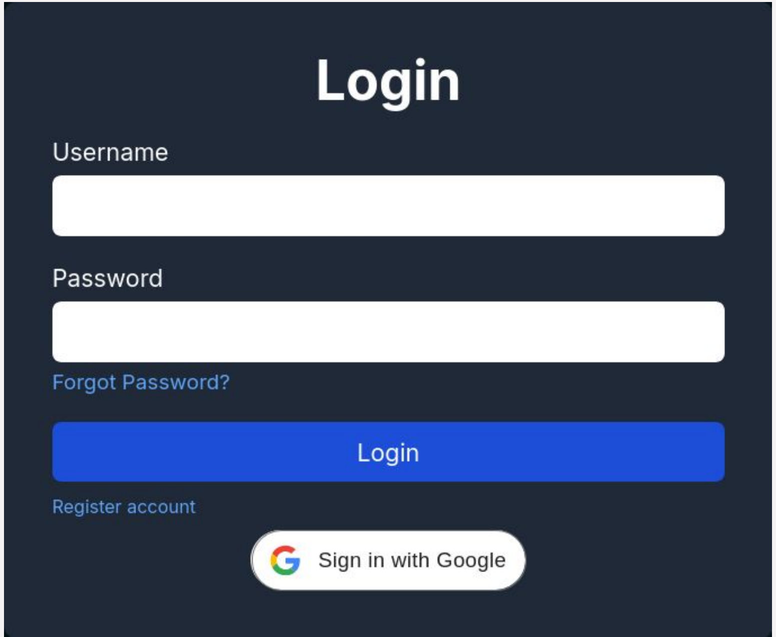
Walter White : Jesse we need to cook!

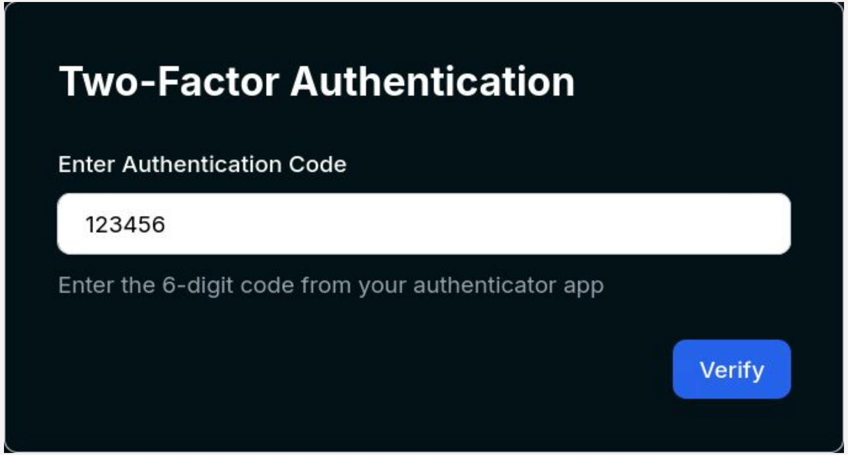
Jesse : Nah man, I'm a 312 professor now

Homework

HW #2 - Authentication and Authorization (Subject to change slightly)

- Login / Signup - Authenticated Chat
- Google Auth, 2FA TOTP





Two-Factor Authentication

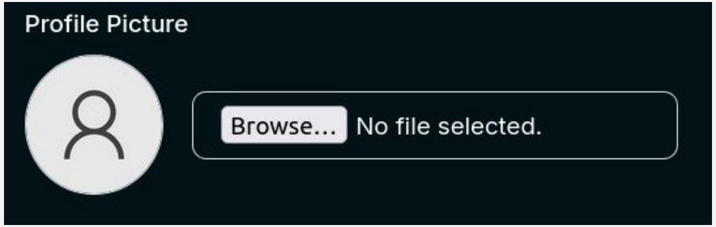
Enter Authentication Code: 123456

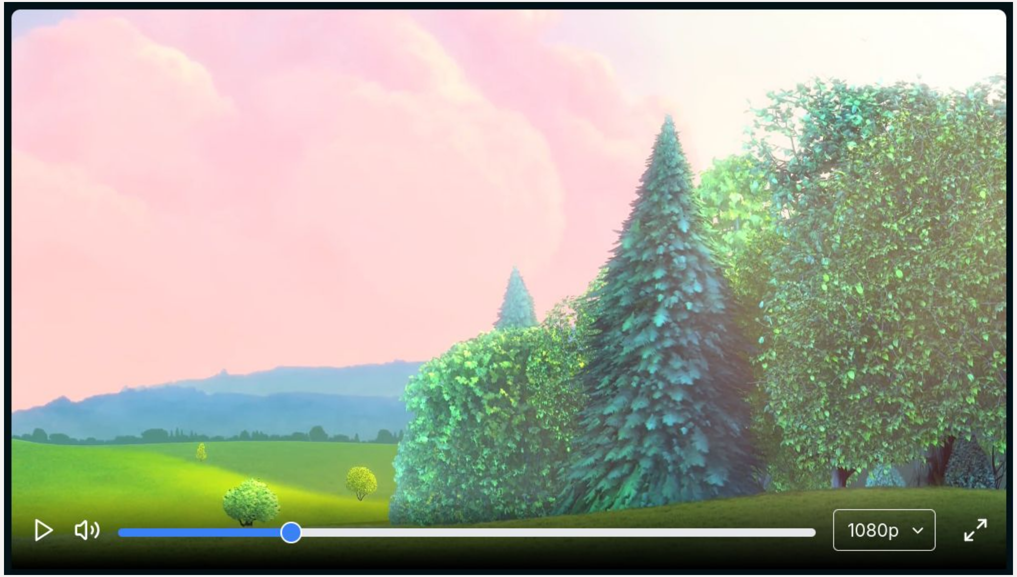
Enter the 6-digit code from your authenticator app

Homework

HW #3 - Handling File uploads (Subject to change slightly)

- Basic YouTube like clone
- Video / Images uploads PFP

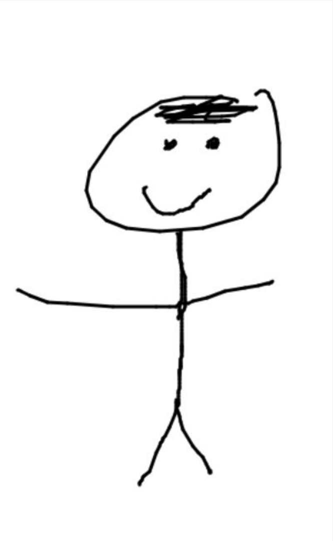


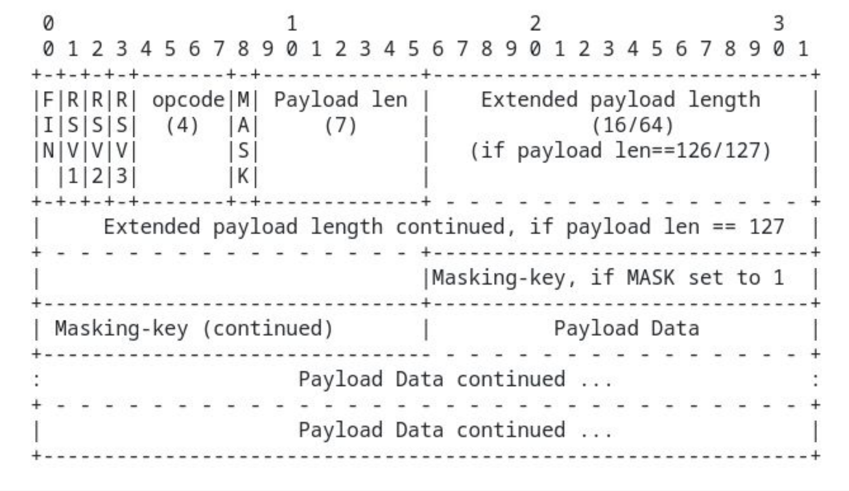


Homework

HW #4 - Web sockets (Subject to change)

- Implement web socket protocol
- Global Drawing board





```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-------+-+-------------+-------------------------------+
|F|R|R|R| opcode|M| Payload len |    Extended payload length    |
|I|S|S|S|  (4)  |A|     (7)     |             (16/64)           |
|N|V|V|V|       |S|             |   (if payload len==126/127)   |
| |1|2|3|       |K|             |                               |
+-+-+-+-+-------+-+-------------+ - - - - - - - - - - - - - - - +
|     Extended payload length continued, if payload len == 127  |
+ - - - - - - - - - - - - - - - +-------------------------------+
|                               |Masking-key, if MASK set to 1  |
+-------------------------------+-------------------------------+
| Masking-key (continued)       |          Payload Data         |
+-------------------------------- - - - - - - - - - - - - - - - +
:                     Payload Data continued ...                :
+ - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - +
|                     Payload Data continued ...                |
+---------------------------------------------------------------+
```

Homework

Show HW Demo site

Group Project

Group Project Foundation

- 4–5 students per team
  - Group Formation link will be on cse312.com
- Homework you build everything from scratch
- Group project you get to use tools / libraries
  - Meant to simulate real word
- Allowed to use AI and online resources
  - Gives you time to work on some cool original idea
- Project Exact Requirements / Structure **TBD**

Group Project

Group Project Structure

- Unlike HW, required to use framework
    - Flask / Python
    - Express / Node.js
    - Django / Python
    - gin / go
    - Play / Java;Scala
    - Koa / Node.js
    - FastAPI / Python
    - Elysia / Node.js
- If framework of choice not on this list, ask Jesse to add it

How to succeed in 312

Office Hours

- Go to them, investment of your time

Try to do all the AOs

- Give your self the ability to have a buffer
  - Don’t tell yourself I can just get the AOs on the next HW

Don’t wait till last day

- These project you can’t finish in a day
- You need time to think about the problem

Delete TikTok