CSE 331
Software Design & Implementation

Autumn 2023

Section 9 – Debugging

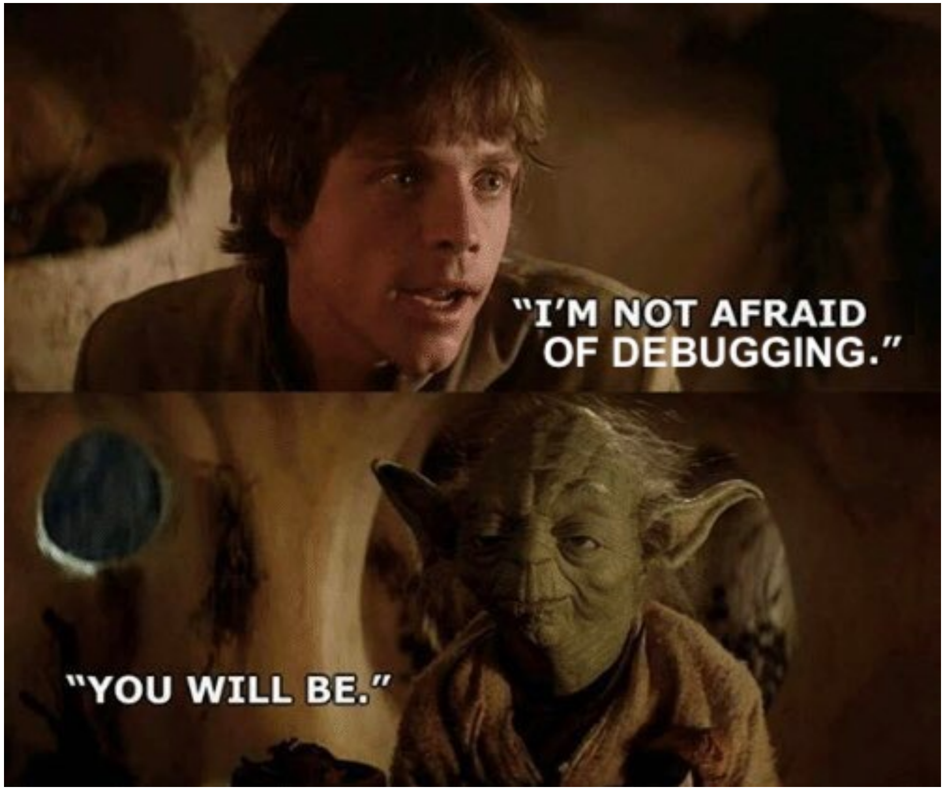
Administrivia

- **HW 9**
  - Due Friday 12/8, 11pm
  - Released this afternoon
- **Final**
  - Tuesday 12/12
  - Final review (likely) Monday 12/11 afternoon
  - Final review section next Thursday!

HW9

- Spec will just be screenshots and some brief explanation of the app you should create
  - Culmination of skills we've practiced in app building
  - (Don't forget about testing, documentation, error checking)
- We will manually test your app for *minimal, required functionality*
  - *Appearance* doesn't have to match
  - Additional app features (CSS styling is not a feature) may earn extra credit

Debugging!



- **Don't stay stuck on the same bug**
  - Continuing isn't helpful
  - Get help!
  - Take a break!
- **Turn bugs into test cases**
  - In case they come back again
- Utilize `console.log()` and the network tab

Client-Server Communication Debugging Steps

1. **Do you see the request in the Network tab?**
   - the client didn't make the request

2. **Does the request show a 404 status code?**
   - the URL is wrong (doesn't match any `app.get` / `app.post`) **or**
     the query parameters were not encoded properly

3. **Does the request show a 400 status code?**
   - *your* server rejected the request as invalid
   - look at the body of the response for the error message **or** add `console.log`'s in the server to see what happened
   - the request itself is shown in the Network tab

Client-Server Communication Debugging Steps

4. **Does the request show a 500 status code?**
   - the server crashed!
   - look in the terminal where you started the server for a stack trace

5. **Does the request say “pending” forever?**
   - your server forgot to call `res.send` to deliver a response

6. **Look for an error message in browser Console**
   - if 1-5 don’t apply, then the client got back a response
   - client should print an error message if it doesn’t like the response
   - client crashing will show a stack trace

sec-debug coding exercise

debugging practice !!

HW9 Tips

- **Design on paper first!**
  - Draw out UI & decide how to organize into components *before* coding anything

- **Don’t reinvent the wheel!**
  - The auction example from lecture will be **very** helpful
  - Stick to the patterns we’ve taught
    - making requests, type checking, error/success responses
    - Feel free to copy helper functions (e.g. `isRecord`, etc)

- If you’re running out of time, watch out for tips in spec for things you can skip and only lose a few points
  - (must complete entire app to get *full* points)

HW9 Tips

- **Recommended implementation order:**
  1. Write the client UI with local data
     - No client/server interaction yet
  2. Write the server
     - Official store of data
       - design appropriate data structures
     - Only provide operations needed by the client
       - look at where state is updated in UI only version to determine what is needed
  3. Connect the client to the server
     - Use fetch to update data on server *before* updating in client
       - Client *always* asks server for data to display in UI
       - Client *always* asks server to perform updates to data