

ECS 189 WEB PROGRAMMING

6/7

Administration

- Sign up for interactive grading
- We're not done grading Midterm 2; planning to finish tonight
- I should be able to give grades so far for those whose Photobooth 2's have been graded tomorrow
- If you have not signed up for interactive grading for Photobooth 2 please do so immediately!

Final

- This room, June 14, 6-8pm
- Will include material from both midterms, Weather and Photobooth
- I will email seats again
- Similar format
 - short answers
 - hard Javascript question
 - programming problem related to projects
- Open book, open notes

Identify the pieces

- Our apps put together pretty standard parts, that can be mixed and matched, like Mr/Ms Potato Head
- I want to know that you understand the parts, and how they fit together; exact syntax you can copy



Identify the pieces

- Suggestion: don't just print stuff out. Make your own notes that you can access quickly during the test
- Include an example, and notes, for each piece
 - Flexbox layout
 - Adding DOM elements
 - Adding button actions, styles from Javascript
 - JSONp request and callback
 - AJAX request and callback
 - Handling an AJAX request on server
 - AJAX request, and server handling, to upload a file
 - Database request and callback
 - SQL and Node Sqlite3 commands
 - Server API request and callback

Hard Javascript question

- There are really only three hard things
 - Global variables – forget the var statement and the variable becomes global.
 - Closure – inner function defined inside an outer function remembers all the local variable values from when the outer function exited. Handy for callbacks.
 - Anonymous functions – often used for callbacks. Handy for constructing a closure, and for adding extra arguments to “normal” external functions.
- Also objects, JSON

Things we did not learn

- There is a ton still to learn both for front (browser) and back (server) end
- Should be useful for senior projects, but you will need to go beyond
- Skills and concepts transfer to mobile frameworks
- Not on the final

Model-View-Controller

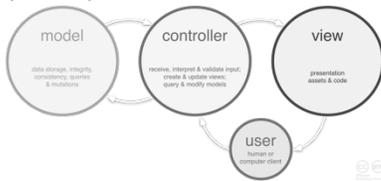
- Software pattern – a way of organizing code. So, mostly a thing people argue about....

This article has multiple issues.

- Many frameworks organized around MVC: Django, Rails, ASP.NET MVC on the server, and Angular, Ember, Backbone in the browser. Let you plug in content and build Web apps quickly (once you know them!)

MVC

- Model = state data and code that changes it
- View = presentation of data to user
- Controller = code that handles user interactions. Might change both model and view



(arguable) MVC examples

- Front end
 - ▣ Model = HTML
 - ▣ View = CSS
 - ▣ Controller = Javascript
- Photobooth?

(arguable) MVC examples

- Front end
 - ▣ Model = HTML
 - ▣ View = CSS
 - ▣ Controller = Javascript
- Photobooth
 - ▣ Model = database
 - ▣ View = front end
 - ▣ Controller = server files

Cookies

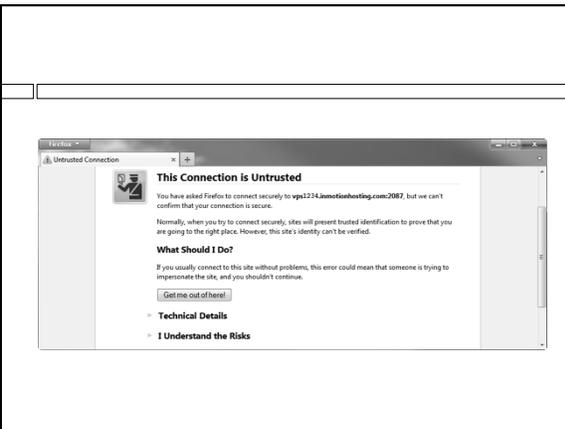
- The server does not remember browsers
- But it can ask browser to remember it!
- Browser stores a file – a cookie – with information about previous interactions
- Server asks for cookie in every interaction, maybe returns it with changes
- Can be used (with crypto) to remember that you're logged in
- Other examples of servers that seem to remember stuff? Sometimes they really do...

Logins

- This would be essential to a real photo sharing site
- For users to login, they need to send you a secret password
- Since the password should not travel unencrypted over the internet, it has to be sent using HTTPS
- You need to store passwords encrypted, in case you get hacked
- Things get messy quickly!

HTTPS

- HTTPS uses the SSL to transmit HTTP messages. Only the destination and port is visible in transit
- To set up an HTTPS server, you need to register with a certification authority and get a certificate - a cryptographic key that the authority knows is yours
- Browsers check your certificate with the authority before using it to encrypt HTTPS messages going to you
- While you can create your own certificates, browsers are deeply suspicious of this



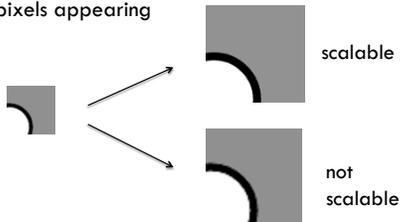
Third-party authentication



- Outsources this and other crypto issues
- But, users need a Google, Facebook, etc, account
- "Easy" to set up using Passport Node module
- You probably want both HTTPS and third-party authentication and some encrypting of private database data in a production app

Scalable Vector Graphics - SVG

- Zoom in and they still look sharp – you don't see pixels appearing



- Why is this good?

SVG

- Language to define 2D vector graphics, runs in the browser
- HTML-like syntax
- Can produce with, eg. Adobe Illustrator, Sketch, Figma, or just using text.

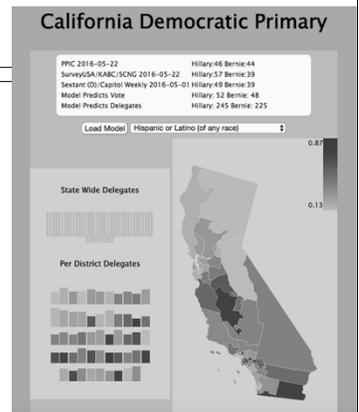
```
<path id="rightEnd"
  d="M 0 0
    // start position
    1 50 0
    // line move 50 in x direction, 0 y
    a 25 25 0 0 1 0 50
    // circular arc down 50
    1 -50 0
    // line back 50 in x
    z" />
// z means close up the shape
```

Animation using SVG

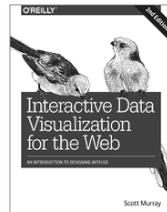
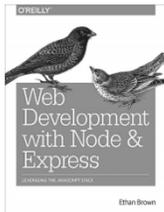
- We saw the command to sleep and force a callback:
`setTimeout(callback, 30);`
- Changing the SVG graphics every 30 milliseconds or so creates animations
- Adding some buttons makes a game!

D3

- Browser visualization library
- Maps, charts, graphics
- Very structured, powerful
- Non-trivial learning curve



Books!



Good luck!

- Some of you will end up knowing a lot more about Web development than I do!