

# ECS 162

## WEB PROGRAMMING

# Weather grade issue

☐

- ☐ Go to grades->Weather
- ☐ Add a comment
- ☐ TA's will review these

# Flashcards

- Due tomorrow, 10pm.
- If by Sunday, only 10% points off, not 20%.
- Questions:
  - ▣ How to decide if user should go to create or review page? One way: AJAX request to retrieve user's cards BEFORE you display anything.

# Schedule

- 10pm Thur 6/6 Flashcards due
- 10pm Sun 6/9 Late flashcards
  
- Possible review sessions?
- Lab hours: Monday 10-11, 3-4, Tues 9-10
  
- 10:30 AM Thurs 6/13 FINAL
- Grades on exam posted 6/15

# Final

- This room, June 13, 10:30 AM
- Will include material from the whole course, but emphasis on material from Flashcards program
- Similar format, maybe some short answers
- I will email seats again
- Bring a SCANTRON
- Open book, open notes; print stuff out

# Last year's final

- ❑ You can't really treat it as a practice final, since we did not use Express or JSX last year.
- ❑ Look it over to get a sense of what kind of questions to expect.
- ❑ Especially, don't worry about the React code in the BuildCollection app.
- ❑ Instead...

# Image Viewer

- App that I can ask you questions about, have you add code to.
- Will be on Canvas, you can download and run it.
- Understand it well before the exam.
- Print it out and bring it with you!
- Will use the Flickr API
  - ▣ User types keywords, app fetches photos from Flickr with those keywords
  - ▣ Lets user cycle through them, pick ones to add to own database

# Other things to bring

- Code examples, from lecture and from your own programs.
- A good set of notes, especially on login and sessions.
- Example for each of the basic parts we use to build up an app.





# Browser side

- HTML and CSS, inc. flexbox
- AJAX request and callback
- CORS request and callback
- React
  - ▣ props and state
  - ▣ render function, constructors
  - ▣ onclick functions and other methods
  - ▣ JSX
- Modules using import

# Server side

- Server construction with Node and Express
- Serving static files
- Answering an AJAX request
- Database requests and callbacks
  - ▣ SQL and Node Sqlite3 commands for submitting them
- API requests and callbacks
- Combining AJAX requests with database or API requests
- Login process
- Security
- Modules using require

# Javascript

- ❑ Scalar data types, automatic type conversion
- ❑ Arrays and objects
- ❑ Object classes and constructor functions
- ❑ Scope using let and const, globals
- ❑ Passing functions as arguments, assigning functions to variables
- ❑ Closure

# Browser or server?

```
url = "http://www.fishServer.org?fish=walleye";  
let xhr = new XMLHttpRequest();  
xhr.open("GET",url,true);  
xhr.onload = function () {}  
xhr.send();
```

Answer: Browser, since it uses an XMLHttpRequest.

# AJAX or CORS?

```
url = "http://www.fishServer.org?fish=walleye ";
```

```
let xhr = new XMLHttpRequest();
```

```
xhr.open("GET",url,true);
```

```
xhr.onload = function () {}
```

```
xhr.send();
```

Answer: CORS, since the URL includes the domain name.

# Browser or server?

```
sendQuery(query, callback) {  
    APlrequest(  
        { // HTTP header stuff  
          url: query,  
          method: "GET"  
        },  
        callback // for response  
    );  
}
```

Answer: Server, this is the using the APlrequest function.

# Using sendQuery(query, callback)

```
handler(req, res, next) {  
    url = "http://www.fishServer.org?"  
    query = url+req.query;  
    sendQuery(query, callback);  
}
```

```
function callback(err, obj, res) {  
    res.json(obj);  
}
```

□ Crashes. How to fix?

Answer: Crashes because the callback from an API request gets two arguments (error and data). The calling code (somewhere in Node) is only calling it with two arguments. But, we need to use res (the response object)! One fix:

```
sendQuery(url, function(err, obj) {  
    callback(err, obj, res);  
})
```

Notice the anonymous function has the response object in its closure.



# Database access

```
handleGetCollection(req, res, next) {  
  cmd = "SELECT * FROM photos";  
  _____ (cmd, dataCallback);  
  function dataCallback(err, data)  
  { res.json(data); }  
}
```

- fill in the blank - db.run, db.get, db.all?
- What ends up in the response?

Answer: db.all, since we expect many rows. data will contain an array of objects, one object for each row.

# React

```
function Tile(props) {  
  return (  
    <div className='imageTile'>  
      <img src={props.src} />  
    </div> ); }  

```

- How would you call this function using JSX? How would you pass in the argument?

Answer: `<Tile src="http://..." />`

# Pass in a function

```
function Tile(props) {  
  return (  
    <div className='imageTile'>  
      <img src={props.src}  
        onclick={props.pFun}/>  
    </div> ); }  

```

□ Why would you do this?

Answer: Clicks usually change state. This component has no state. So maybe it changes the state of its parent component.

# State in parent

```
class ImageDisplay extends React.Component {  
  constructor(props) {  
    super(props);  
    this.state = { photos: props.photos, index: 0 };  
    this.advance = this.advance.bind();  
  }  
  // pass to Tile as pFun  
  advance() { current=this.state.index;  
              this.setState( index: current+1 ); }  
}
```

--	--

Comment: Only components defined by classes can have state. The state object is typically initialized in the constructor function, and changed by interaction with the user or other events.

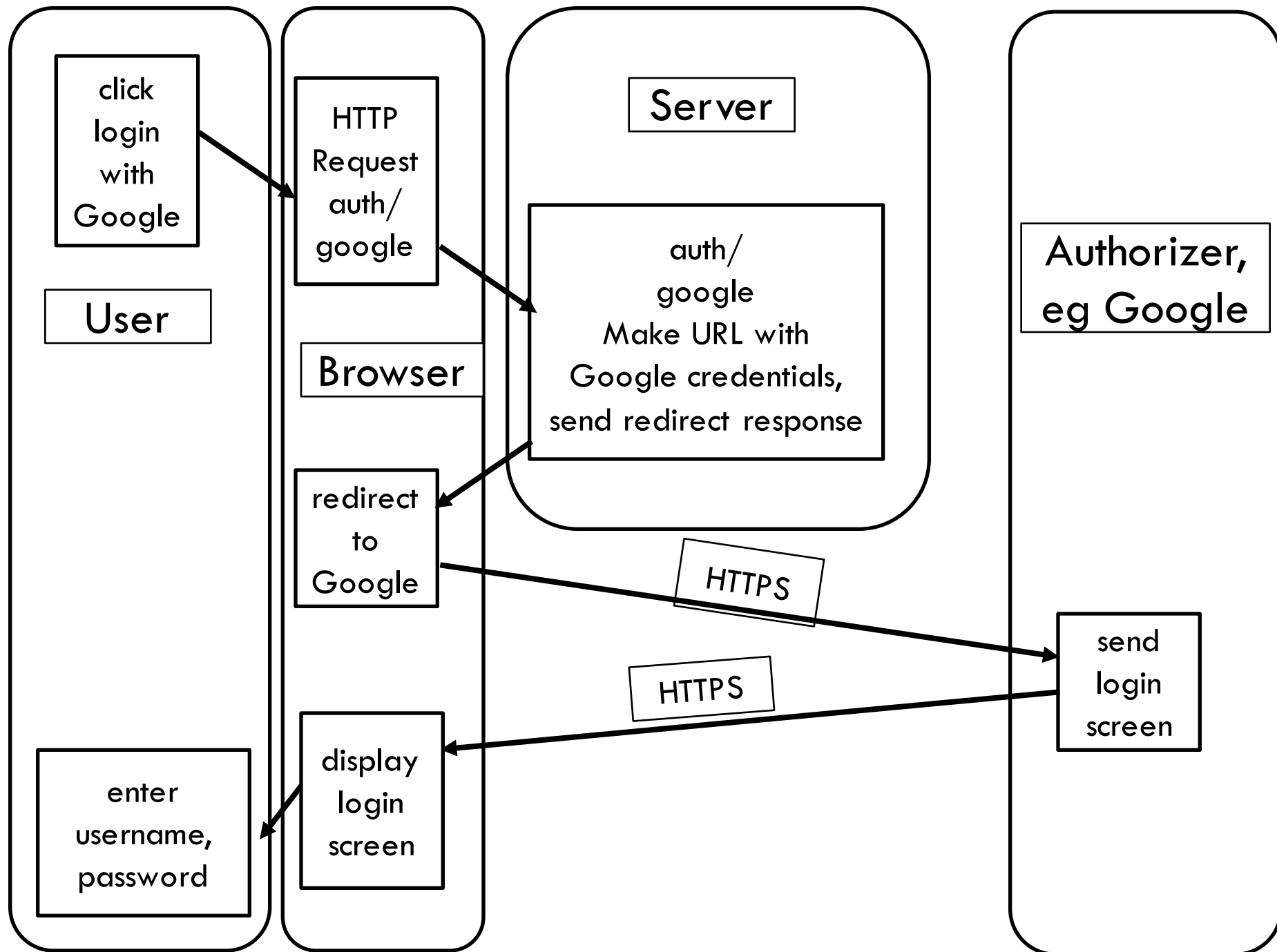
If this object had a Tile child, it could pass "advance" in as the onclick function.

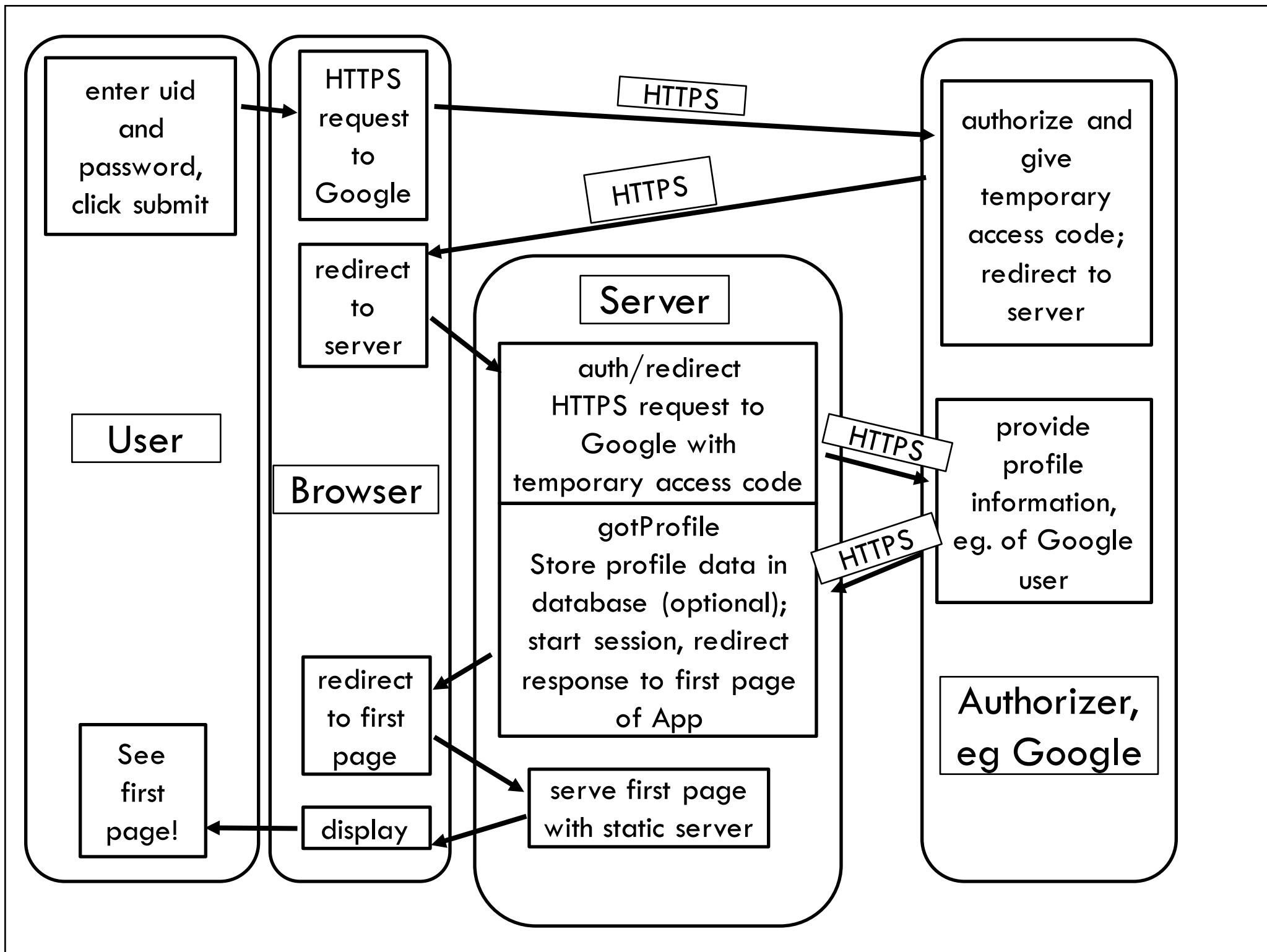
Behavior in `imageViewer.jsx` will be similar.

# Session Cookies

When the user logs in with Google, the browser receives a session cookie. Pick the best completion of this sentence: The browser includes the cookie in...

- a) the header of all HTTP requests, everywhere
- b) the header of all HTTP requests to the server
- c) the body of all HTTP requests, everywhere
- d) the body of all HTTP requests to the server
- e) the header of all AJAX requests
- f) the body of all AJAX requests







# Login "why?" or "what if?" question

- Let's say a user is already logged into Google when they go to lango.html. The splash screen comes up, they hit "login with Google". What happens?

Answer: they still have to login to lango.

- Why does the process start with the browser loading the page at server162.site/auth/google rather than doing an AJAX request to server162.site/auth/google?

Answer: you can't redirect an AJAX request outside the Server's domain.

# Good luck!

--	--

□ See you on the 13<sup>th</sup>