

USF CPH - Week 4

Zero to prototype in 35 days

Getting Set Up

1. Go to `github.com / cvburgess / usfcph-week4`
2. Fork the repository to your account
3. Clone the repository to your computer
4. Open with VSCode

GitHub interface showing the repository **cvburgess / usfcph-week1**.

Navigation tabs: Code, Issues (0), Pull requests (0), Actions, Projects (0), Wiki, More, Settings.

Repository statistics: 1 commit, 1 branch, 0 releases, 1 contributor.

Branch: master

Buttons: Create new file, Find file, Clone or download (highlighted with a red circle and the number 3).

Files list:

| File | Description | Time |
|--------------|------------------------|---------------|
| homework.md | create repo with files | 3 minutes ago |
| notes.md | create repo with files | 3 minutes ago |
| readme.md | create repo with files | 3 minutes ago |
| resources.md | create repo with files | 3 minutes ago |

Help people interested in this repository understand your project by adding a README. (Add a README button)

How to Fork and Clone with the GitHub UI

Week 3

Looking back before we move forward

Logic and JavaScript

- Logical languages
 - Used to express how things behave and interact
 - Conditional, iterative, and chained logic
- JavaScript
 - One of many languages to express logic
 - Not the easiest (Python ?) but the most versatile
- npm is a directory of open source work you can use for free
 - React is a popular package for building complex apps
 - There are packages for virtually everything

Data

From the developer perspective

Defining “data”

- Any information that is used or stored
- Text, numbers, images, videos, sound, etc
- Examples:
 - Contact info, payment data, app info, analytics, etc

1st Party vs 3rd Party

(Either way it's still a party)

Let's Make a Todo App

Or at least review the code for one

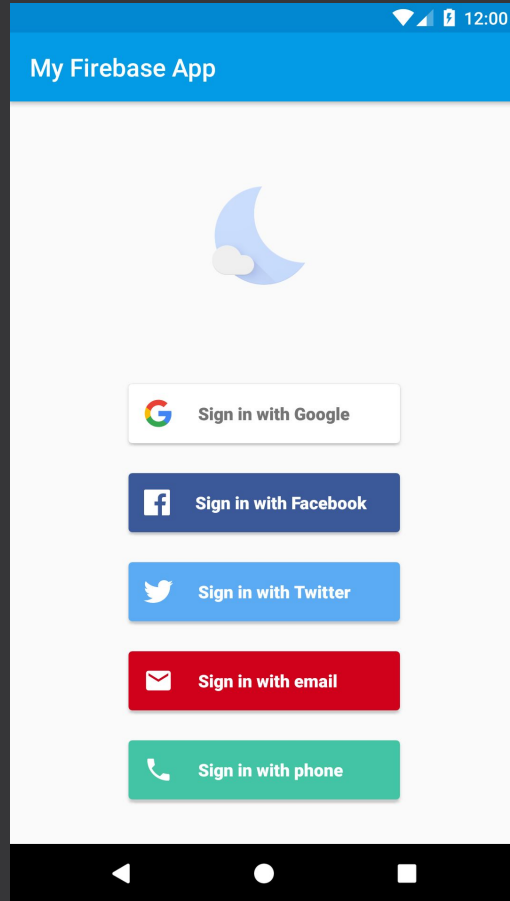


Who

Data access and security

Authentication

- Defines who can access your application and its data
 - What can a user see without logging in?
 - What roles and permissions are there?
 - What can different users create / view / change / delete ?
- Often outsourced to an identity provider (Facebook, Google, etc)
 - There are trade-offs that come with not controlling this data
- Consider leveraging an authentication platform to minimize risk
 - Auth0, Okta, AWS Cognito, etc



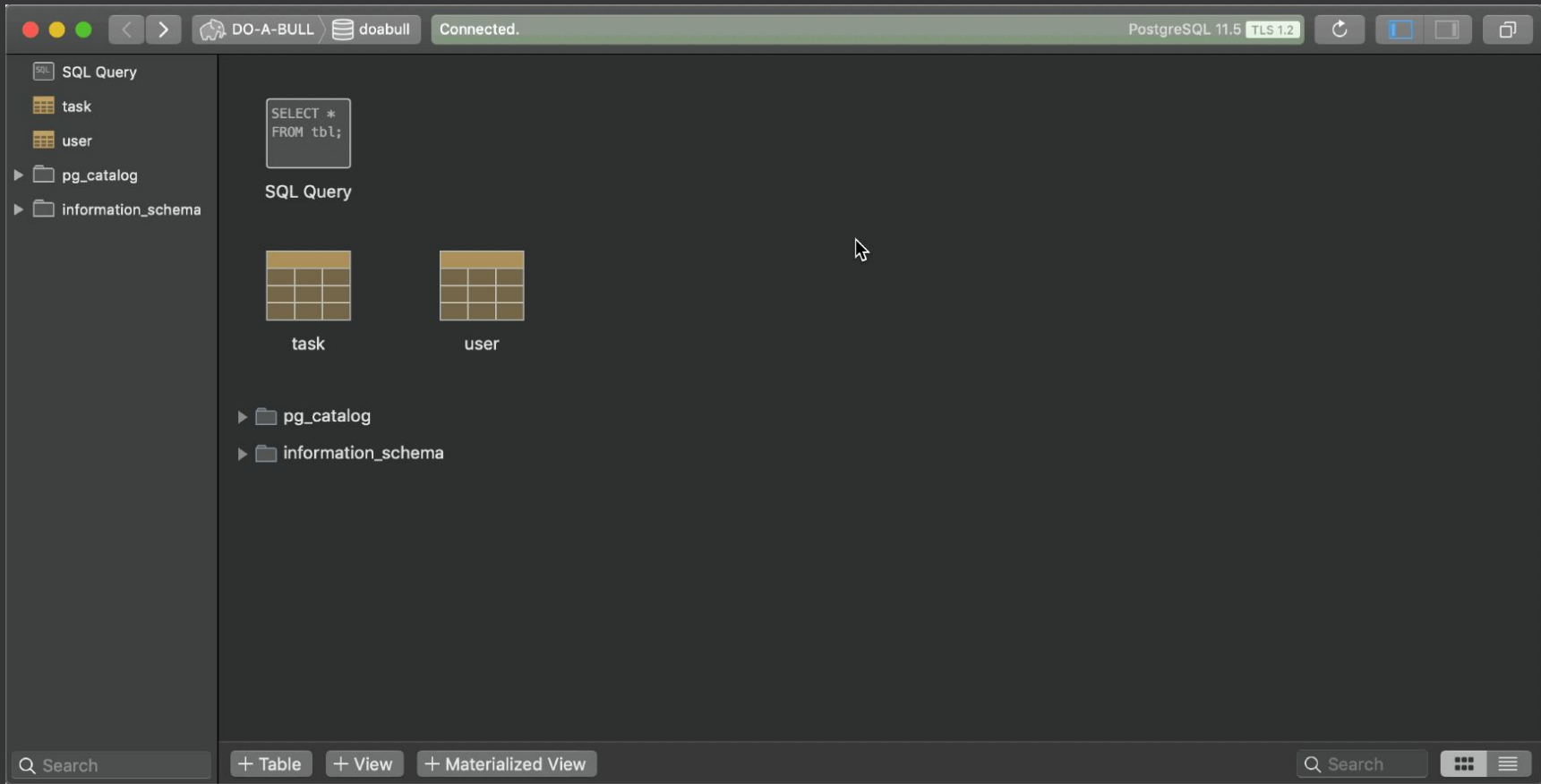
Sample app that supports multiple authentication strategies

Where

Because PODS does not store data

Storage Mechanisms

- Relational databases (RDBMS or DB)
 - Good for the vast majority of data and applications
 - Consists of tables, columns, indexes, and relationships - like excel on steroids
- Document store / data lake (NoSQL DB)
 - Good for massive data collections or unstructured data
 - Example: Library of Congress
- File storage / Content Delivery Network (CDN)
 - Used for fast, cheap file storage (images, videos, music)



Viewing a relational database with an app (Postico)



What

Data segmentation and filtering

Filtering

- “ All data “ is typically not useful
- Filter the data down to:
 - What is relevant to the task at hand
 - What is relevant to the current user
 - Allowing for keyword search and other conditions like date and time
- Done in different ways
 - Relational DBs use SQL (Query Language)
 - ORMs let you interact with a database without writing SQL
 - APIs use query parameters and arguments



```
SELECT -- The names of the fields you want to return
  title,
  description,
  is_urgent,
  due_on
FROM -- The name of the table that has the data
  task
WHERE -- Filters and conditions to apply
  created_by = 1
ORDER BY -- How the data should be sorted
  due_on
```

A sample SQL query for our TODO app

Exercise

Let's play with databases

Using a database

1. Go to [pgweb-demo . herokuapp . com](https://pgweb-demo.herokuapp.com)
2. Choose “Scheme”
3. Paste value from email “ postgresql://... ”



Why

Ask and you shall receive

Formatting data

- We store data for machines, but display it to humans
- Relationships
 - Grab related data and package it up for use
 - Example: Todos + User that made them + The list they are in
 - In SQL, these are called JOINS



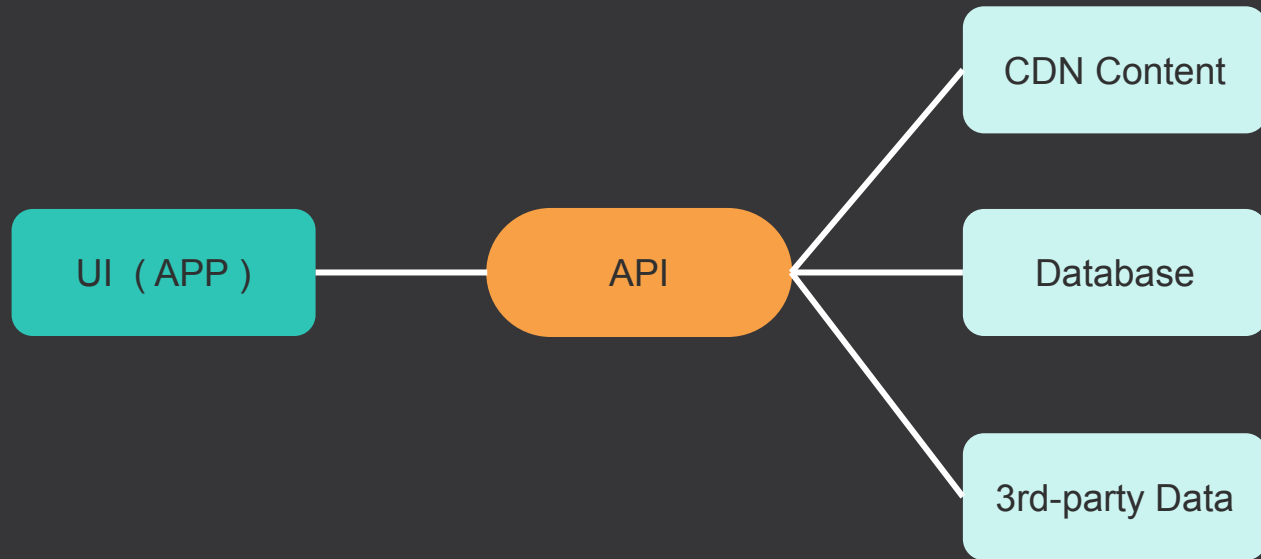
How

Get ready for more acronyms



APIs

- Server-side apps for data
- Handles most of the things we talked about:
 - Authentication and permissions
 - Fetching and formatting
 - Filtering and sorting
 - “Gluing “ multiple data sources together



How APIs act as middlemen

3rd Party Data

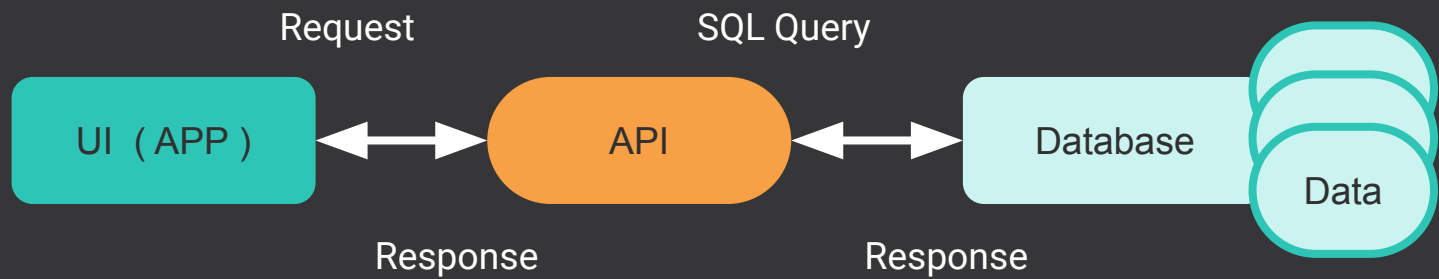
- 99% of the time, you will get access to an API
 - Usually with a set of credentials, limits, and a fee
- Public domain / government data
 - Large data sets that change infrequently
 - Strategy: download and put in your own database

Exercise

Using Chrome Dev Tools

The Network Tab

1. Go to [sad - easley - 2648f6 . netlify . com](https://sad-easley-2648f6.netlify.com)
2. Right-click > Inspect
3. Click the “ Network ” tab
4. (optional) Filter by “ XHR ”



Under the hood: flow of data

Exercise

Using Chrome Dev Tools

Data Privacy + Security

More than just logging in

Never Trust the Client

Under any circumstance. Ever.

Sensitive Data

Credit Cards, Socials, Passwords, etc

PII + GDPR + HIPAA

Credit Cards, Socials, Passwords, etc

Recap

- Specialized servers and tools for storing data
 - Most of the time, use a database
- APIs wrap data with logic and an easy-to-use interface
 - Especially true for 3rd-party data
- Sensitive data needs to be handled with care
 - Unless you enjoy being sued, of course

Homework

1. Review your notes
2. Play with the app we created
 - a. Make overdue tasks red
 - b. Use the API to let users complete tasks
3. Discuss what data your idea will need
 - a. Where will the data come from?
 - b. Does the data have special security or privacy concerns?
 - c. What is the simplest way to store the data?

USF CPH - Week 4

Zero to prototype in 35 days