

# JAVASCRIPT DEVELOPMENT

Sasha Vodnik, Instructor

# HELLO!

1. Pull changes from the `svodnik/JS-SF-7` repo to your computer
2. Open the `starter-code` folder in your code editor

---

**JAVASCRIPT DEVELOPMENT**

---

# **INTRO TO CRUD AND FIREBASE**

# LEARNING OBJECTIVES

At the end of this class, you will be able to

- Explain what CRUD is.
- Explain the HTTP methods associated with CRUD.
- Implement Firebase in an application.
- Build a full-stack app with create and read functionality.

# AGENDA

- CRUD
- Firebase intro and setup
- Create
- Read
- Update
- Delete

---

## INTRO TO CRUD AND FIREBASE

---

# WEEKLY OVERVIEW

### WEEK 8

Project 2 Lab / this & Module pattern

### WEEK 9

CRUD & Firebase / Deploying your app

### WEEK 10

(holiday) / Instructor/student choice

---

**INTRO TO CRUD AND FIREBASE**

---

# **HOMEWORK REVIEW**

# Checkin and questions

- The **most significant thing I learned** about this and the module pattern is \_\_\_\_\_.
- My **biggest outstanding question** about this and the module pattern is \_\_\_\_\_.

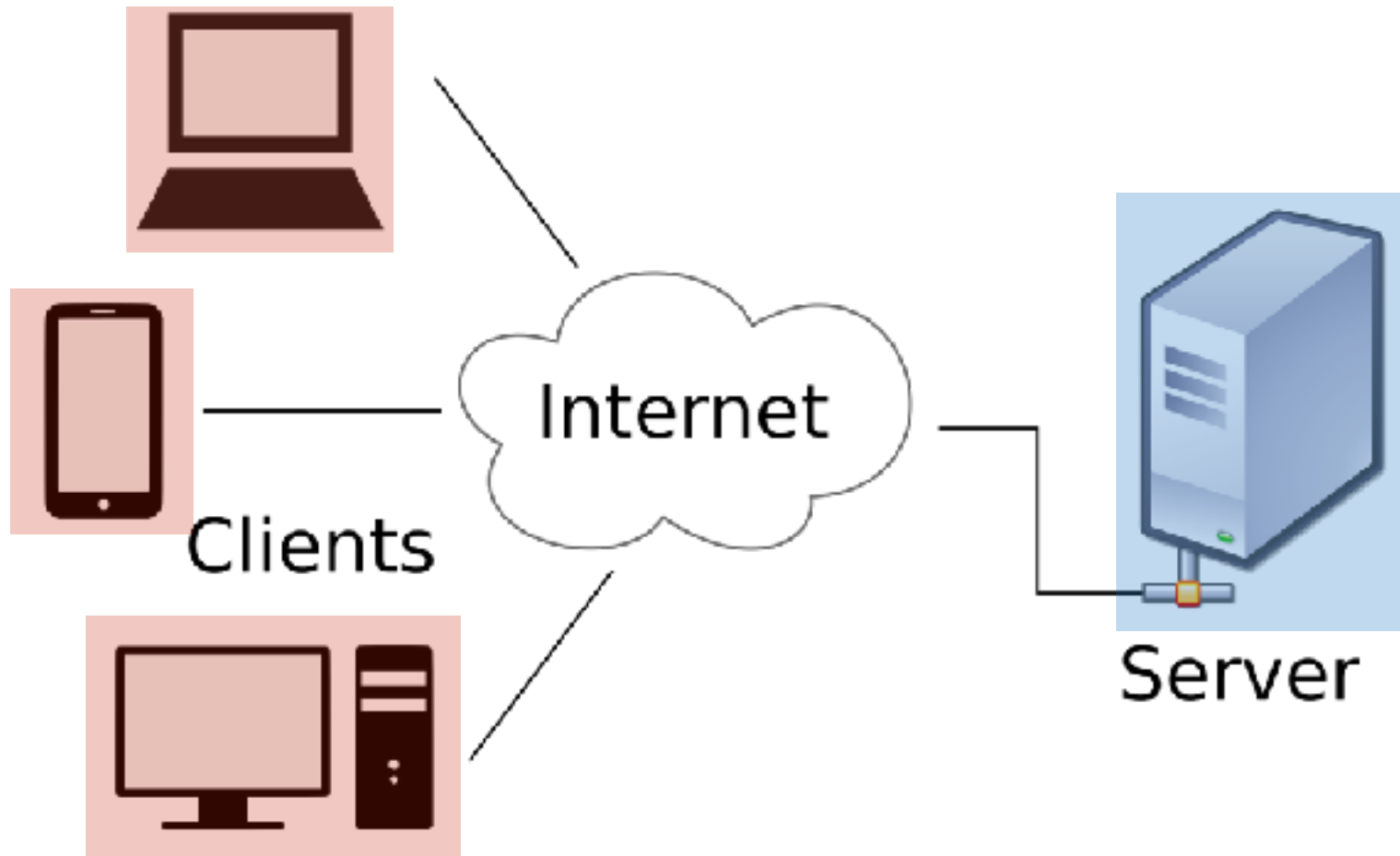


**What are some apps that allow you to create, read, update, and delete data?**

## Back-end review

### Front end

- HTML
- CSS
- JS



### Back end

- JS
- Python
- Ruby
- PHP
- ...

# CRUD

- Create
- Read
- Update
- Delete

# **CRUD and HTTP**

<b>CRUD action</b>	<b>HTTP verb</b>
Create	POST
Read	GET
Update	PATCH/PUT
Delete	DELETE

# EXERCISE — API METHODS

---



## EXERCISE

### KEY OBJECTIVE

---

- Identify API methods that let you implement CRUD functionality using a popular web service

### TYPE OF EXERCISE

---

- Groups of 3

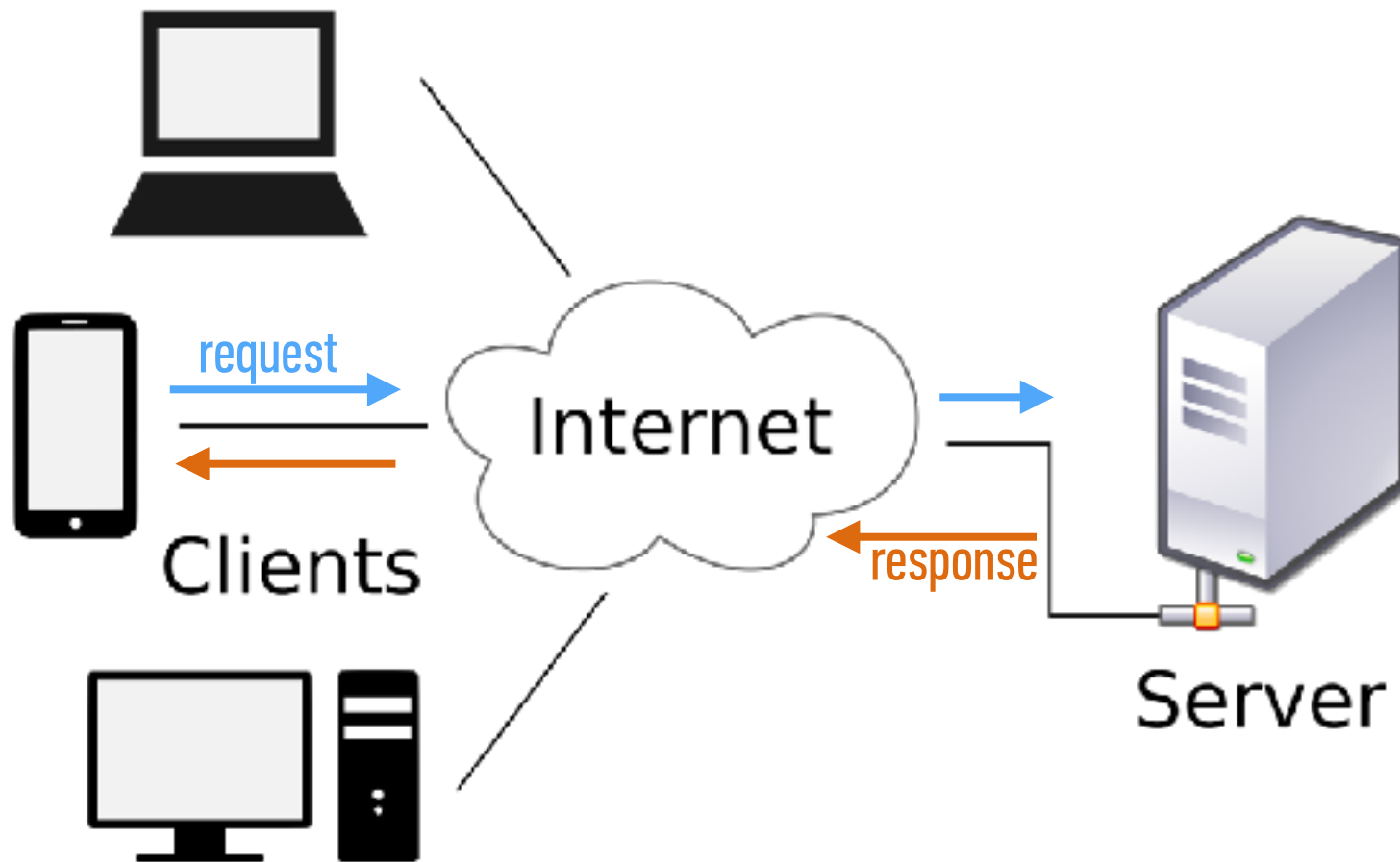
### TIMING

---

*5 min*

1. Research your assigned API to see what HTTP methods a developer must use to perform at least one instance of create, read, update and delete. (If your API doesn't fully support CRUD, note any limitations.)
2. Further, define what exactly is being created, read, updated or deleted. For example, for Facebook what HTTP method on what endpoint must you ping in order to create a post in a feed?

# THE CLIENT-SERVER MODEL WITH CRUD



**Stores HTML/CSS/JS code**

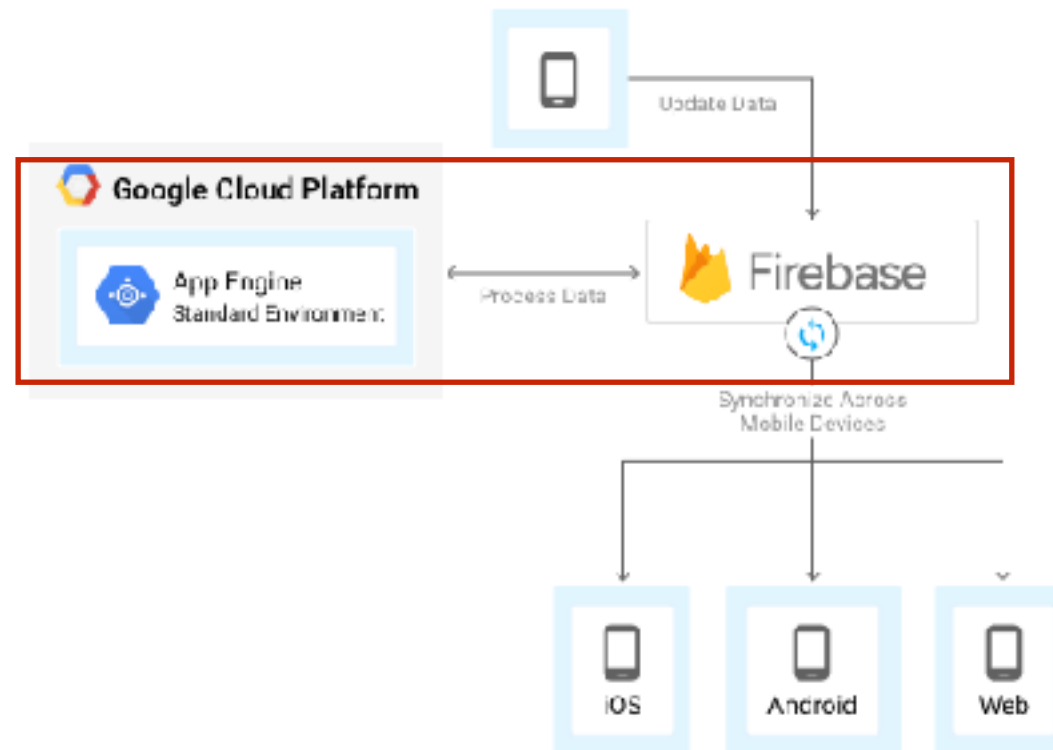
- Accepts HTTP requests
- Generates HTTP responses

**Stores database**

- Provides create access
- Provides read access
- Provides update access
- Provides delete access

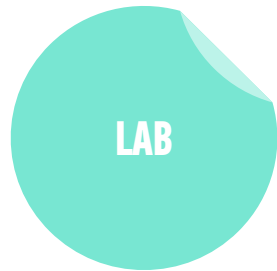
## FIREBASE

Back end



# LAB — PLAN A CRUD APP

---



## KEY OBJECTIVE

---

- › Plan a full-stack app with full CRUD functionality

## TYPE OF EXERCISE

---

- › Solo or in pairs

## TIMING

---

*20 min*

1. Come up with an idea for an app that implements CRUD. You'll build your app this week in class (this is not your final project). Your app must be able to Create, Read, Update and Delete data.
2. Build out your HTML, CSS, and JS files.
3. Use the Firebase dashboard to create your new app.
4. Get the URL key to initialize your app, and initialize your Firebase app utilizing the starter code.

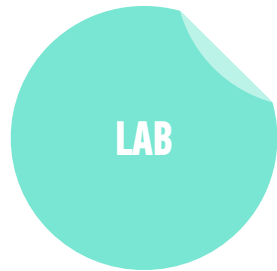


## **CRUD and HTTP**

<b>CRUD action</b>	<b>HTTP verb</b>	<b>Firestore method</b>
Create	POST	push()
Read	GET	ref()
Update	PATCH	update()
	PUT	set()
Delete	DELETE	remove()

# LAB — IMPLEMENT CREATE FUNCTIONALITY

---



## KEY OBJECTIVE

---

- Build the Create functionality of a full-stack app

## TYPE OF EXERCISE

---

- Solo or in pairs

## TIMING

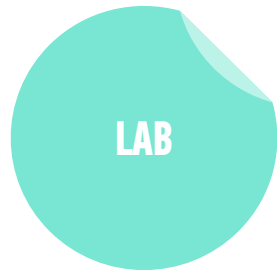
---

*20 min*

1. Create a form
2. Get user input
3. Create a section in your database for your data
4. Save your data to the database
5. Change security rules to allow access without authentication
6. View your data in the Firebase dashboard

# LAB — IMPLEMENT READ FUNCTIONALITY

---



## KEY OBJECTIVE

---

- Build the Read functionality of a full-stack app

## TYPE OF EXERCISE

---

- Solo or in pairs

## TIMING

---

*20 min*

1. Examine the API documentation at <https://firebase.google.com/docs/reference/js/firebase.database.Reference>
2. Listen for changes (use `.ref()` and `.on()`)
3. Add returned data to your front end using DOM manipulation

# **LEARNING OBJECTIVES – REVIEW**

- Explain what CRUD is.
- Explain the HTTP methods associated with CRUD.
- Implement Firebase in an application.
- Build a full-stack app with create and read functionality.

## **NEXT CLASS PREVIEW**

### **Deploying your app**

- Add update and delete functionality to a full-stack app.
- Understand what hosting is.
- Identify a program's needs in terms of host providers.
- Deploy to a web host.

**Exit Tickets!**

# Q&A