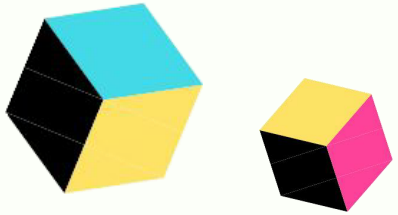
orlando math circle

OMC

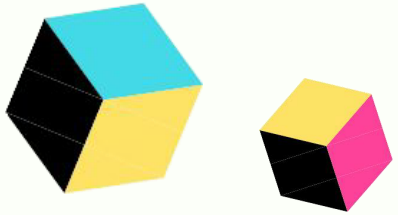
Introductions

- Richard Bent – Backend Testing, Backend Developer
- Luisa Cardona – Project Manager, Frontend Designer
- Samantha Perez – Lead Frontend Designer
- Christopher Polynice – Lead Tester, Backend Developer
- Cody Traywick – QA, Lead Backend Developer



Workshop Schedule – Day 2

- 9:00 – 9:15 Warm-Ups, Backend Preview
- 9:15 – 9:45 Backend Introduction – JavaScript Review
- 9:45 – 10:30 TypeScript Introduction – Concepts
- 10:30 – 11:00 Break/Free Time
- 11:00 – 11:30 Backend Exercises
- 11:30 – 11:45 Discussion
- 11:45 – 12:30 NestJS Introduction – Code Preview
- 12:30 – 1:00 Concluding Remarks



Warm-Ups

Exercise 1 – 10 minutes

- Create a simple HTML webpage that contains a heading and a paragraph.

```html
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Title of the document</title>
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

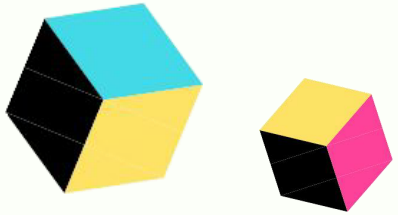
JavaScript Review

Review JavaScript Concepts

- Dynamic programming language normally used with webpages to add intractability
- Usually invoked by appending <script></script> tags within the page

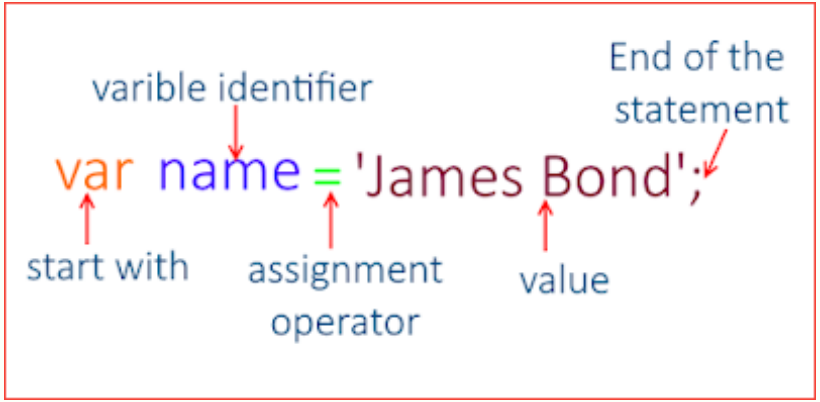
Review JavaScript Concepts

- Programming concepts supported such as:
  - Variables
  - Conditionals
  - Events
  - Functions



Variables

- const, let, var – variable declarations
- number, string, boolean – variable types
  - Ex: let number = 4;
  - Ex: const string = 'I love OMC!'



Conditionals

- Logic that helps to structure code and influence decisions
  - *if* condition, *then* statement, *else...*

```
if(condition01) {
    //condition01 = true (will execute)
} else if(condition02) {
    // condition02 = true (will execute
} else {
    // condition 02 = false (will execu
}
```

Events

- Code that handles actions invoked from user
- Generates dynamic behavior that separates JavaScript from HTML

Common HTML Events

Here is a list of some common HTML events:

Event	Description
onchange	An HTML element has been changed
onclick	The user clicks an HTML element
onmouseover	The user moves the mouse over an HTML element
onmouseout	The user moves the mouse away from an HTML element
onkeydown	The user pushes a keyboard key
onload	The browser has finished loading the page

Functions

- A block of code that containerizes all the previously-explained concepts
  - Maintains readability and promotes code modularity

```
function add(num1, num2) {
    // code
    return result;
}

let x = add(a, b);
// code
```

function call

Live Coding Session – 20 minutes

TypeScript Introduction

TypeScript, like JavaScript…

- Builds upon JS concepts learned to create a more strongly typed language
- TypeScript is known for type checking, validating code and logic *before* code is ran
  - More effective in catching ambiguous errors than JS
  - The OMC application is coded in TypeScript

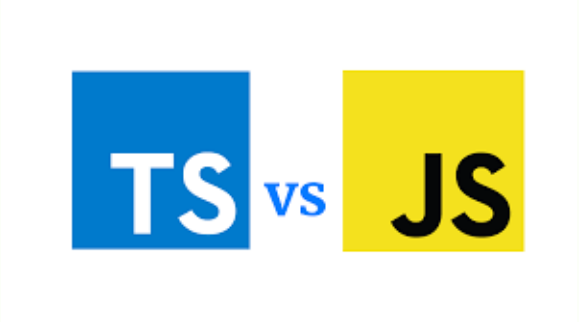
TypeScript YT Video – 5 ~ 10 min

Break / Free Time

Practice

Exercise 1 – 10 minutes

- List three differences between JavaScript and TypeScript.
- When TypeScript code is compiled, what is produced, assuming the code is free of errors?



Exercise 2 – 15 minutes

- From the HTML page created in the warm-up, write a function that, when the button is clicked after inputting a number, doubles it.

Discussion

Exercises – Answers

- List three differences between JavaScript and TypeScript.

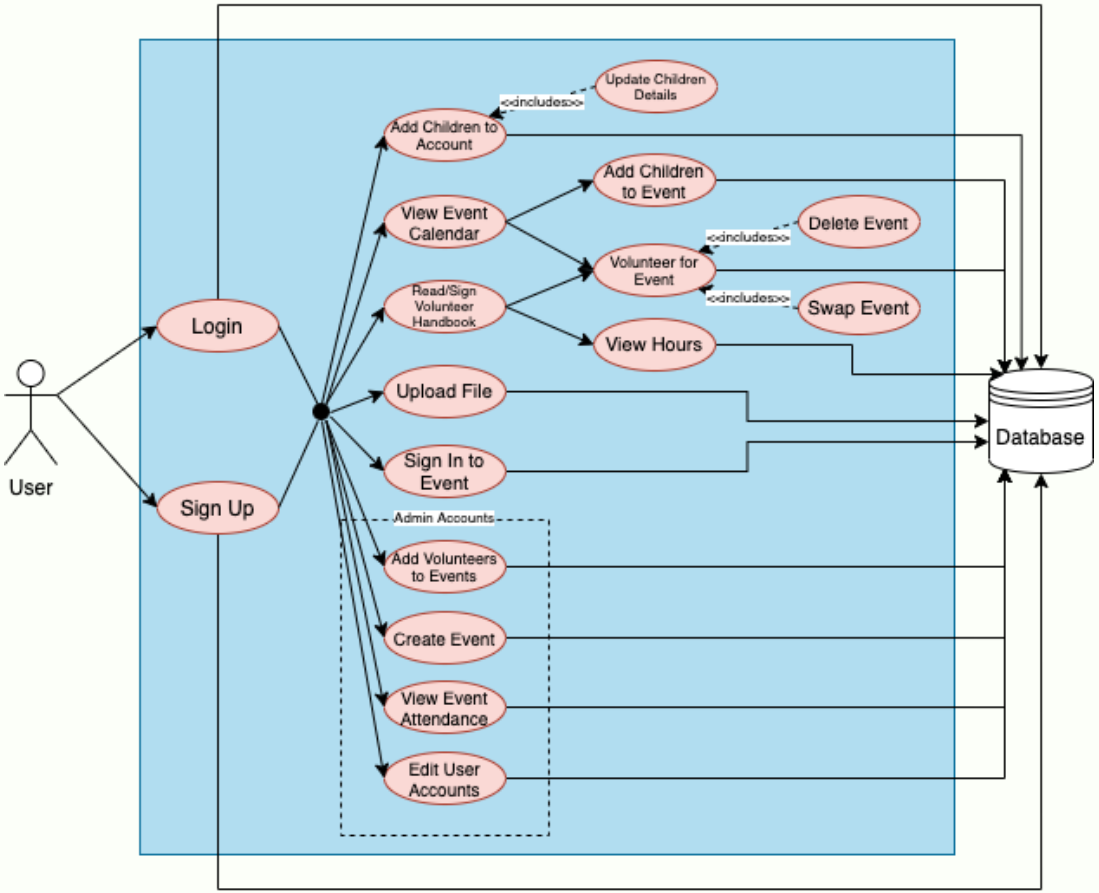
- When TypeScript code is compiled, what is produced, assuming the code is free of errors?

Exercise 2 – Answer

- Code live

Backend Processes

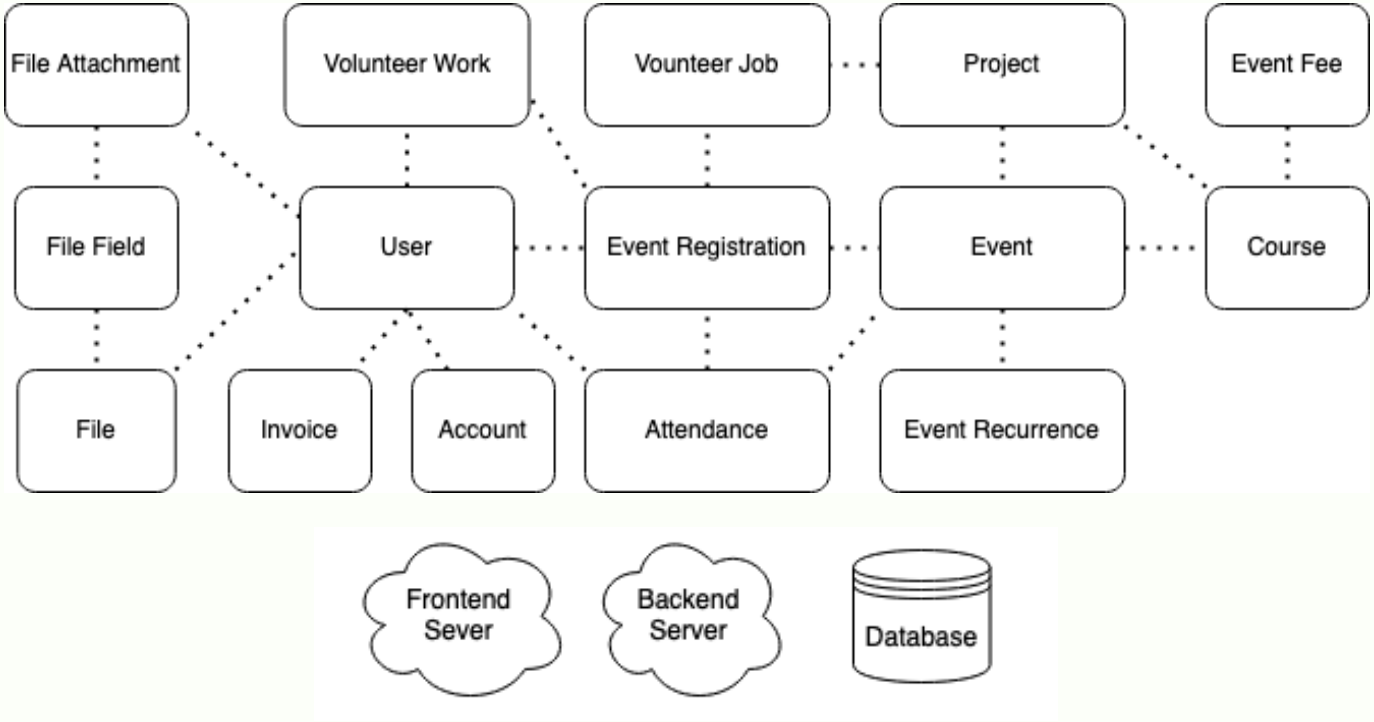
Use Case Diagram



Why PostgreSQL?

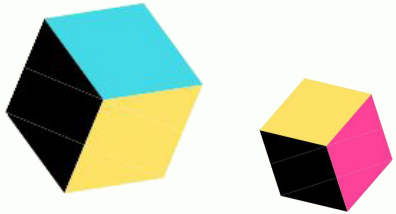
- Desire for joining data and reverse lookups eliminated NoSQL
- Constraints help enforce COPPA compliance and referential integrity
- PostgreSQL has a faster development schedule compared to MySQL
- JSON storage with query and index support through the `jsonb` type
- Automatic schema generation and TypeScript support with MikroORM

System Design



GitHub

- Centralized repository for project files
- Version control
- Forks, pull requests, and merges
- Documentation hosted in GitHub Pages
- https://github.com/orlando-math-circle/omc-app

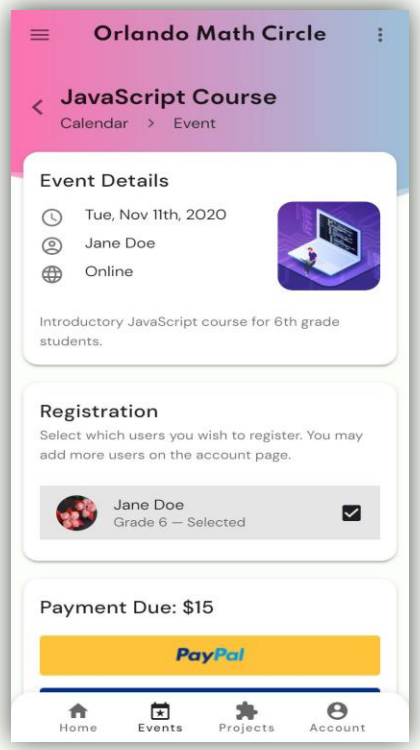


NestJS

NestJS...

What the OMC Application uses for Backend.

- Supports a combination of both TypeScript and JavaScript
- Module based



YouTube Video – 30 min

NestJS Part 1 - Basics - YouTube

Closing Remarks

THANK YOU!