Evolve Full Stack Developer

Introduction to ExpressJS

INCEPTIONU

Now about that spaceship

Agenda

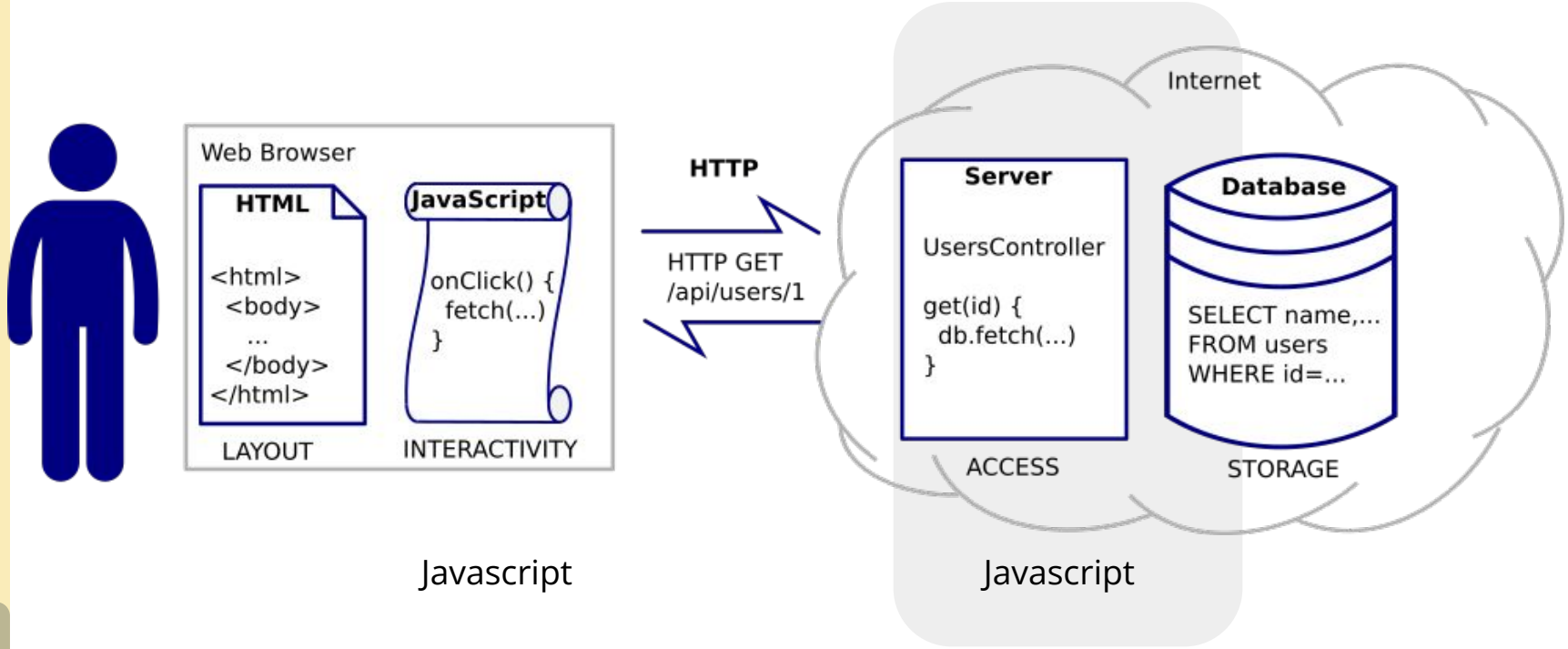
Building an ExpressJS HTTP server

- Intro to **ExpressJS**
- Activity: “You’re in a deep dark wood”
- Handling **URL**s
- Handling **GET** & **POST Requests**
- Intro to OpenAPI

By the end of today you be able to start an ExpressJS server to service GET and POST requests, as well as document and test it.

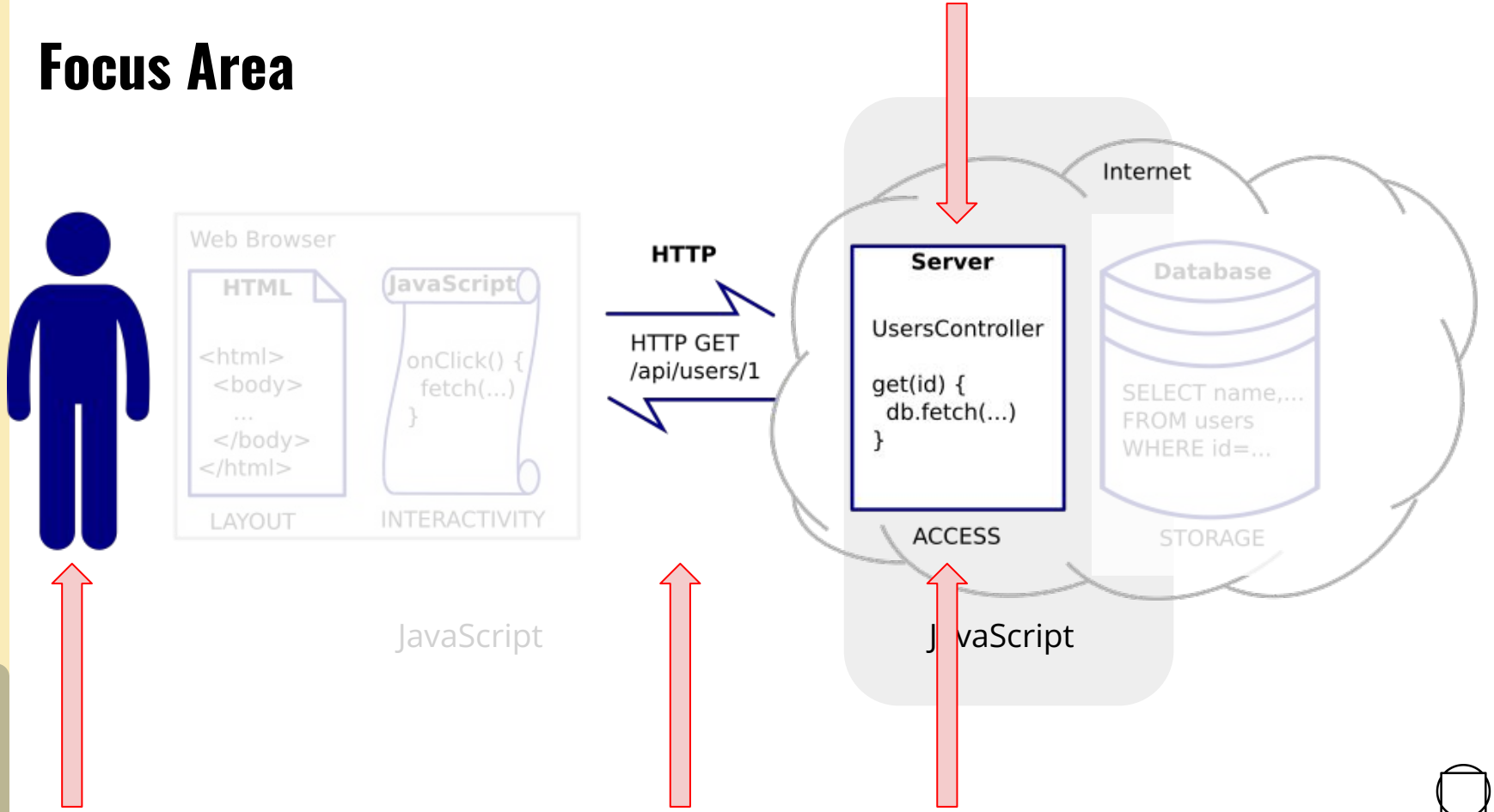
INCEPTIONU

Focus Area



INCEPTIONU

Focus Area



INCEPTIONU

Let’s Get Started

Creating a new NPM project

Review of *npm*

- NPM is a set of command-line tools that ships with “node” and it is a a repository of *modules* (code built by other developers) available from npmjs.com
- To initialize a new project:
  a. Make a new directory: **`mkdir intro-to-expressjs`**
  b. Change to that new directory: **`cd intro-to-expressjs`**
  c. Create a new NPM project: **`npm init`** (or **`npm init -y`**)
- Once created, start VS Code: **`code .`**
- Update package.json with “type”: “module”

Takeaways

- Create a new project with **`npm init`** (or other tools as we’ll see when we get to React)
- **`package.json`** is updated by the npm command or by editing the file manually (name, version, scripts, etc.)
- Add **`package.json`** and **`package-lock.json`** to your *git repository*.
- Add **`node_modules`** to your **`.gitignore`** file. This directory stores all your dependencies (often lots of files) which should not be committed to your repository.
- *Do not* manually edit **`package-lock.json`**.

ExpressJS - Getting started

Set Up ExpressJS

- Install the *express module* to your project: **`npm install express`**
- ExpressJS app example: start and console.log()
- ExpressJS app example: Add "Hello World!" API
- res.send() vs res.json() vs res.end()

Key Takeaways

- ExpressJS builds two handy objects for you: ***request*** and ***response*** that are passed to the “handler function” or the “callback function” or the “middleware”.
- If a route sends a response, the connection is closed and no further processing takes place.

Exercise - Create a GET handler in ExpressJS

Instructions

Extend the ***hello world*** ExpressJS app

- Mild: Add a new GET endpoints that returns the plain text: `"ExpressJS Rulez!!!"`
- Medium: Add a new GET endpoint that returns JSON: `{"hello": "world"}`
- Spicy: Add GET endpoints that returns JSON including the server's current Date:
  - `{"currentDate": "Friday, October 14, 2022 10:42AM"}`

ExpressJS - handling URL query parameters

Review

- Article: URL Anatomy

Code Example

- Mild: Handling URL query parameters
- Medium: Respond to the values of multiple parameters (e.g. **daylight** and **numberOfDragons**)

Key Takeaways

- Query parameters are the name/value pairs that come after the ? in a URL.
- Name/values are separated by =
- Different name/value pairs are separated by &
- Example query URL: `/forest?daylight=true&numberOfDragons=8`
- URL query parameters are automatically parsed by ExpressJS and are defined in the ***request.query*** object

INCEPTIONU

ExpressJS - Browser vs. Command Line

- You can interact with an ExpressJS server using the Browser:
  a. http://localhost:4000/forest?daylight=true&numberOfDragons=8
- A browser does “a lot of work” that the user cannot see
- Using a command-line tool bring you (the developer) “closer to what is actually happening”
- Run the command:

```
curl "http://localhost:4000/forest?daylight=true&numberOfDragons=8"
```

- See the “man page” for curl and its `-v, --verbose` option

INCEPTIONU

Express - handling POST requests

Code Example

- Handling POST Requests
- To send data via POST with:
  - curl -d '{"text":"Hello, World!"}' -H 'Content-Type: application/json' <url>
  - curl -d @<filename> <url>

Key Takeaways

- The **`express.urlencoded()`** middleware will look for form submission headers and add the data to **`request.body`**. This assumes data was submitted using a traditional submit button (i.e. not using Javascript to submit json) or a POST request sent using form data in an application such as Postman.
- There is a JSON version of this middleware if you're expecting a form to be submitted using that data structure: **`express.json()`**. It performs the same functionality and adds submitted data to **`request.body`**

OpenAPI

What is OpenAPI

- OpenAPI is a specification that helps developers design, document, and test APIs that are consistent, flexible, and easy to use.

How we can use it

- Sample OpenAPI doc
- Full specification documentation
- VSCode Extension

INCEPTIONU

Today we...

1. Created an Express web server
2. Created GET endpoint handlers
3. Accepted submitted data from URL query parameters using GET
4. Accepted submitted data from a POST request
5. Introduced OpenAPI to document and test our APIs