

A Smart Guide to Building Web Applications

João Costa Seco

January 26, 2024

Chapter 1

Introduction

This small booklet has X parts illustrating small aspects of the construction of a web server in JavaScript, a static web page using basic technologies like HTML and CSS. And then how to develop a dynamic web page with events and dynamic construction of HTML elements.

The goal of this booklet is to illustrate the different aspects of the construction of a web server in TypeScript. The booklet is not intended to be a complete reference, but rather a quick guide to get you started.

Part I

A Web Server in TypeScript

Chapter 2

Setting up your first node application

A Web server is a program that listens to requests from a browser and returns a response. Responses can be HTML pages, images, JSON data, etc. Content can be either static if provided by an existing file in the server filesystem, or dynamic if it is created on demand and based on the information from the request.

We are going to use Node.js to create our web server. Node.js is a JavaScript runtime environment that allows us to run JavaScript and TypeScript code outside of the browser.

1. To start with you need to install Node.js. You can download it from the Node.js website.

```
npm init
```

2. Next, you need to create a directory for your project and initialize it.

```
mkdir myproject  
cd myproject  
npm init -y
```

These commands will add a `package.json` file to your project. This file contains all the configuration information for your project.

3. Next, you need to install the TypeScript compiler and initialize it.

```
npm install -g typescript
tsc --init
npm install ts-node typescript --save-dev
```

These commands will add the file `tsconfig.json` to your project and install the necessary packages. This will contain the information for the TypeScript compiler to compile your code.

4. Next, you need to write your first TypeScript file. Create a file called `index.ts` in a folder called `src`.

```
console.log("Hello World");
```

5. Change your `package.json` file and modify the "start" script to run your program.

```
"scripts": {
  "start": "ts-node src/index.ts"
},
```

6. Finally, you can run your program with the following command:

```
npm start
```


Chapter 3

Setting up your first web server

1. Next, you need to install the Express framework

```
npm install express --save
```


Chapter 4

Loading files from the web server

Chapter 5

Returning dynamic content

Use a form to upload parameters.

Chapter 6

Webservices: returning JSON data

Part II

A Web page using HTML and CSS

Chapter 7

HTML: Writing your first web page

Chapter 8

HTML: The anatomy of an HTML element

Chapter 9

HTML: Basic Text Formatting

Chapter 10

HTML: Adding images

Chapter 11

HTML: Linking different pages

Chapter 12

HTML: Formatting with stylesheets

Chapter 13

HTML: Basic Layout with CSS

Chapter 14

HTML: Making requests with Forms

Part III

Reactive pages: Events and JavaScript

Part IV

Dynamic pages with JavaScript

