Web Development Basics

The Shrey Method Fitness Platform Guide

← Back to Index | Next: Tools Setup →

Before we dive into building your fitness platform, let's understand some basic web development concepts. This will help you make sense of what we're doing in later sections.

What is a Website vs. Web Application?

- A **website** is like a digital brochure - it displays information but doesn't do much else. Think of a restaurant's website that shows their menu and hours.

- A **web application** is interactive - users can log in, submit information, and get personalized experiences. Examples include Gmail, Facebook, and the fitness platform we're building.

We're building a web application because your fitness platform needs to be interactive, allowing clients to log in, view personalized content, and interact with your services.

Front-end vs. Back-end

Web applications have two main parts:

- **Front-end**: What users see and interact with (the visual part)
- **Back-end**: The "behind the scenes" part that stores data and handles logic

Think of it like a restaurant:

- Front-end = The dining area, menus, and waitstaff that customers interact with
- Back-end = The kitchen, inventory system, and staff that customers don't see

Basic Web Technologies

We'll be using these fundamental technologies:

HTML (HyperText Markup Language)

HTML is the structure of web pages (like the skeleton of a body). It defines elements like:

- Headings
- Paragraphs
- Images
- Links
- Forms

Example of HTML:

```
<h1>Welcome to The Shrey Method</h1>
<p>This is a paragraph about fitness coaching.</p>
<img src="workout.jpg" alt="Person working out">
```

CSS (Cascading Style Sheets)

CSS is the styling of web pages (like the clothing and appearance). It controls:

- Colors
- Fonts
- Spacing
- Layout
- Responsive design (how pages look on different devices)

Example of CSS:

```
h1 {
    color: green;
    font-size: 32px;
}

p {
    font-family: Arial, sans-serif;
    line-height: 1.6;
}
```

JavaScript

JavaScript makes pages interactive (like the muscles that create movement). It enables:

- User interactions
- Form validation
- Dynamic content updates
- Communication with servers

Example of JavaScript:

```
document.getElementById('login-button').addEventListener('click', function() {
    // Code that runs when the login button is clicked
    checkCredentials();
});
```

Databases

Databases store information (like your client data, workout plans, etc.). They:

- Organize data in a structured way
- Allow for efficient retrieval and updating
- Keep data secure and persistent

We'll be using AWS DynamoDB, a cloud-based database service.

How These Technologies Work Together

1. **HTML** provides the structure of your web pages
2. **CSS** makes them look good and work on different devices
3. **JavaScript** makes them interactive and communicates with the back-end
4. **Back-end code** (which we'll create using AWS services) processes requests and interacts with the database
5. **Database** stores all your application's data

Cloud Computing

Instead of running our own servers, we'll use **cloud computing** services from Amazon Web Services (AWS). This means:

- No need to buy or maintain physical servers
- Pay only for what you use
- Automatic scaling as your business grows
- Built-in security features

What We're Building

Our fitness platform will use all these technologies:

- HTML, CSS, and JavaScript for the user interface
- AWS Amplify to simplify back-end development
- AWS Cognito for user authentication
- AWS DynamoDB for the database
- Stripe for payment processing
- Calendly for scheduling

Don't worry if these concepts seem confusing now - we'll explain each one in more detail as we use it!

← Back to Index | Next: Tools Setup →