CHAPTER 24

HTTP: The Postal Service of the Internet

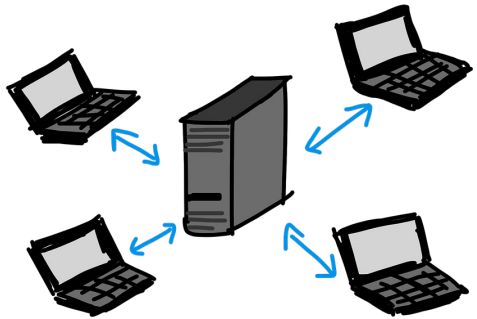
Chapter Breakdown

- 24.1 How the Internet Works
- 24.2 HTTP at a Glance
- 24.3 Requests
- 24.4 Responses
- 24.5 HTTP in the Browser

24.1 How the Internet Works

How the Internet Works

Servers and Clients
Protocols
Web Addresses



- Internet uses the client-server model
  - Servers provide the resources
  - Client requests resources from servers
- Protocols
  - HTTP - Hypertext Transfer Protocol
  - TCP/IP- Transmission Control Protocol / Internet Protocol
  - DNS- Domain Name Service
- Web Addresses
  - Uniform resource locator (URL)
  - URL Components
    - Scheme, Host, Port, Path, Query String
    - Scheme and Host are required
  - Scheme - http, https, ftp , etc.
  - Host - where the request should be sent
    - Either an IP address or domain name
  - Port - determines the specific application on the server
  - Path - tells server what is being requested
    - Seperated by /
  - Query String - provides additional data that may be needed to fulfill the request

24.2 HTTP at a Glance

HTTP at a Glance

Request and Responses
The Postal Service of the Internet

- Client (usually a web browser) makes a request to a web server.
- Server will send back a single response
- HTTP is similar to the Post Office
  - Letter is the request message
  - Envelope contains location and metadata
  - When you drop it off you know it's going to be delivered.

24.3 Requests

REQUESTS

Request Methods
Headers
Body

HTTP requests must conform to the structure outline by the W3C

Structure - Request Line, Request Header, Blank Line, Request Body

Request Methods: part of the Request Line. Specifies action to be carried out by the requested resource

- GET Method - wants to retrieve the resource
- POST Method - wants to create new data on the server

Headers - defines operating parameters of the HTTP transaction

Body - optional, often includes form data submitted via a POST request

24.4 Responses

```
HTTP/2.0 200 OK
Date: Wed, 22 May 2019 17:36:50 GMT
Content-Type: text/html; charset=UTF-8
Content-Length: 8050
Last-Modified: Wed, 22 May 2019 17:33:45 GMT

<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<!--Rest of HTML page -->
</html>
```

RESPONSES

Response Codes
Response Headers
Response Body

Response Codes: Standardized codes that servers use to convey the result of attempting to fulfill the client's request

- 200 - Success
- 301 - requested resource moved
- 404 - Not found
- 500 - server connection error

Similar to Requests, the response will have a Response header

Response Body: while requests body are optional, Responses always have a body.

24.5 HTTP in the Browser

HTTP in the Browser

Viewing Requests and Responses in Dev Tools
Browser Flow

- Network pane displays all HTTP requests and responses involved in loading a page
- Browser Flow
  - Ex loaded via GET request
  - Browser requests a page from server
  - Browser receives the HTML page and parces it
  - For each image - new HTTP request is sent for each file
  - Browser processes each additional response as received.

Chapter 25

User Input with Forms

Chapter Breakdown

- 25.1 Forms
- 25.2 Form Submission
- 25.3 POST Form Submission
- 25.4 Text Inputs
- 25.5 Specialized Text Inputs
- 25.6 Checkbox Input
- 25.7 Radio Input
- 25.8 Select Input
- 25.9 Validation With Javascript

25.1 Forms

Forms

Create a Form
Input Element
Labels
Value Attribute

- Form is created using the ‹form› tag with opening and closing tags
  - Serves as a container for various types of elements used to capture user input
  - Empty form won't appear unless it has something with an ‹input› tag
- Input Element is used to add interactive fields. Has two attributes
  - Name attribute - identify input value when submitted
  - Type attribute - type of value the input represents
- Labels are used to provide text label which informs the user the purpose of the field. The ‹label› tag wraps around the corresponding ‹input› tag.
- Value attribute can be used to set a default value of an input.

25.2 Form Submission

Form Submission

Trigger Form Submission
Key-Value Pairs

Form submission is triggered by clicking a button inside the form.

Either an input with type=”submit” or a button element will work.

When a form is submitted an HTTP request is sent to the location set in the action attribute of the ‹form›

If an action attribute is not present or is empty the form will submit to the URL of the current page

When a form is submitted a key-value pair is created for each named input.

- Key - values of the name attributes
- Values - what you entered into those inputs.

25.3 POST Form Submission

POST Form Submission

Form Submission Using Post
Send Form Submission to a Server

We add the form method to the ‹form› by adding method = “POST”

Data submitted by GET requests is less secure than POST

Action attributes tells us where to send the form

Form Handlers are web server actions that receive, inspect and process form requests. They then send a response to the client.

25.4 Text Inputs

Type	Syntax	Description	Demo
text	`<input type="text" name="username"/>`	A single line text field.	This is a test
textarea	`<textarea name="missionDescription"> </textarea>`	A larger, multi-line text box. Must have open and closing tags.	this is a test of multi line section!
password	`<input type="password" name="passCode"/>`	A text field that obscures the text typed by the user.	••••••••••••••••••••••

25.5 Specialized Text Input

Type	Syntax	Description	Demo
date	`<input type="date" name="flightDate"/>`	Browser validates the value is a valid date format. Some browsers provide a *date picker*.	
email	`<input type="email" name="emailAddress"/>`	Browser validates the value is a valid email address format.	
number	`<input type="number" name="fuelTemp"/>`	Browser validates the value is a valid number format.	

25.6-25.8 Checkbox, Radio, Select Input

Type	Syntax	Description	Demo
checkbox	`<input type="checkbox" name="signUp"/>`	A small box for marking form option as *checked*.	sign up☐

Type	Syntax	Description	Demo
radio	`<input type="radio" name="crewReady" value="yes"/>`	A small circle that allows selecting *one* of multiple values. Used in groups of two or more.	yes○no○

Type	Syntax	Description	Demo
select	`<select name="weather"> <option value="1">clear</option> <option value="2">cloudy</option> </select>`	A menu that allows selection of one option. Requires options to be in `<option>` tags.	clear ⌄

25.9 Validation with Javascript

Validation with Javascript

Form Inputs and the DOM
Steps to add Validation

‹input› tags can be selected and referenced like any other HTML element

To read the value of an input you have to check the value. We can assign a new value dynamically with input.value

Add an event listener for the form.

When the form is submitted get the current values of the inputs

Validate that the inputs have correct values

If they don't prevent the form from submitting utilizing event.preventDefault();