

Diploma in Web Development – Part I



Lesson 2

How Do Websites Work?

Presented by:

Julian Quirke

Web Development Educator



Lesson 1 Recap

- ✓ Introducing Shaw Academy Systems
- ✓ Course Agenda
- ✓ The Function of Websites
- ✓ Key Terms in the Industry
- ✓ Summary
- ✓ Q&A



How Do Websites Work?

- Websites & Web Applications
- Components of a Web Application
- Delivering Websites to a Browser

- Summary
- Q&A

AGENDA



WEB DEVELOPMENT TOOL KIT

INCLUDES:



Exclusive Starter Pack



Webinar Slides for every lesson



Summary Notes for every lesson



Bonus Videos

FREE

for completing
each lesson

or

€39.95

Enhance your learning experience!



1. *Setting Up Your Development Environment*

- Lesson 2
- Available at the end of today's lesson

2. *Creating Your First HTML Document*

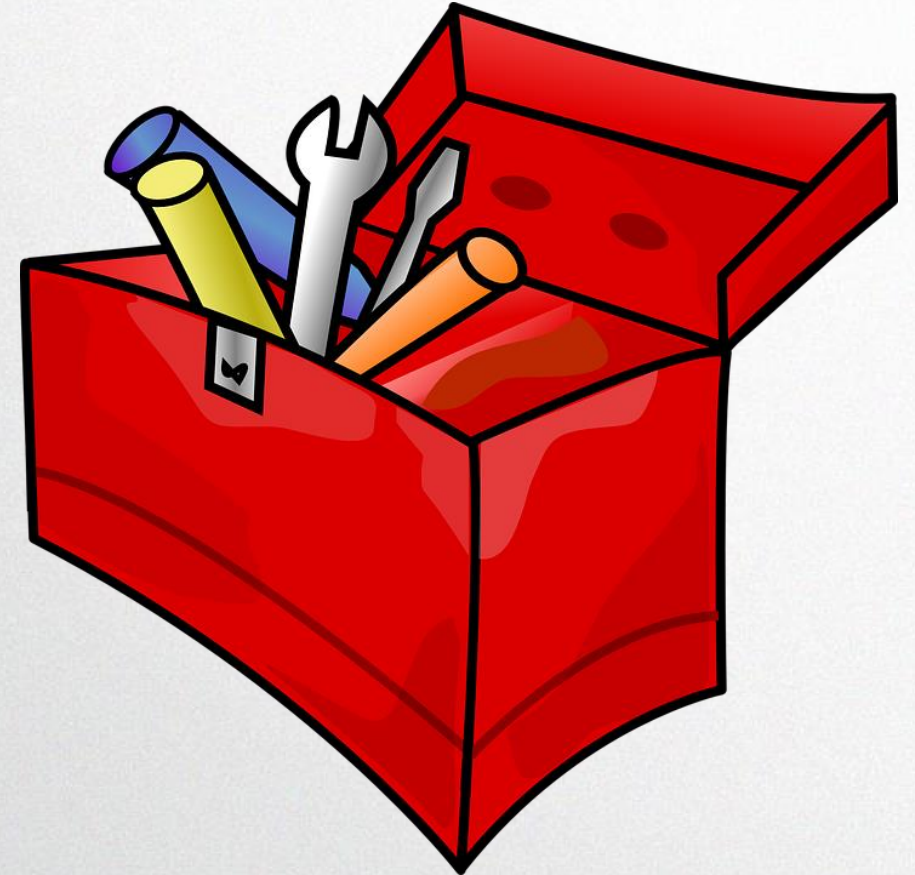
- Lesson 3

3. *Styling Your Webpage with CSS*

- Lesson 4

4. *Adding Interaction with JavaScript*

- Lesson 5



Let's Begin!



Website



Website

Collection of **web pages**, written in HTML, that are connected to the internet



Web Application



Web Application

Dynamic, server-driven website that can deliver personalised websites to users



Websites & Web Applications

Comparison

Website	Web Application
Static Informational, brochure Generally the same across users	Dynamic Dependant on User Interaction Changes based on User's Data



Websites & Web Applications

Makeup

Website	Web Application
Web Pages Web Server	Web Pages Web Server Dynamic Content Logic



Components of Web Applications



Components of Web Applications



Web Server

Web Pages



Reminder:

Web Servers & Web Pages

are enough to build a

Static Website



Components of Web Applications



Back End Code

Database Software



Components of Web Applications

Servers

- Computer connected to a network
- “The Cloud” – the Internet!
- Web Servers:
 - Special software installed
 - Contain the other components



Components of Web Applications

Servers



APACHE



Components of Web Applications

Web Pages

HTML:

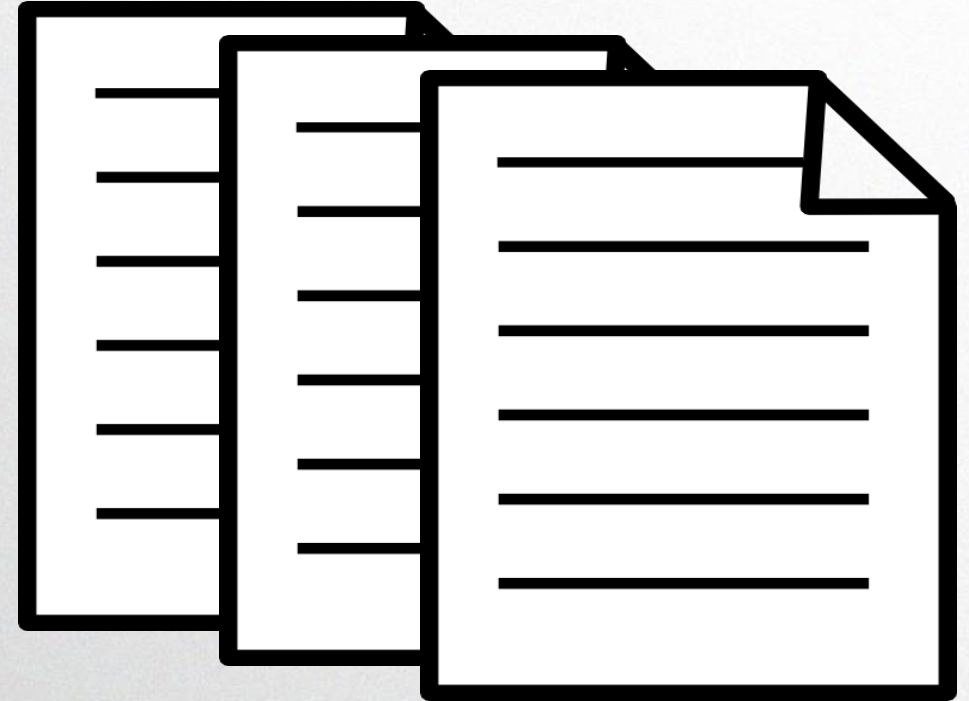
- Markup for content

CSS:

- Defines the visuals and layout

JavaScript:

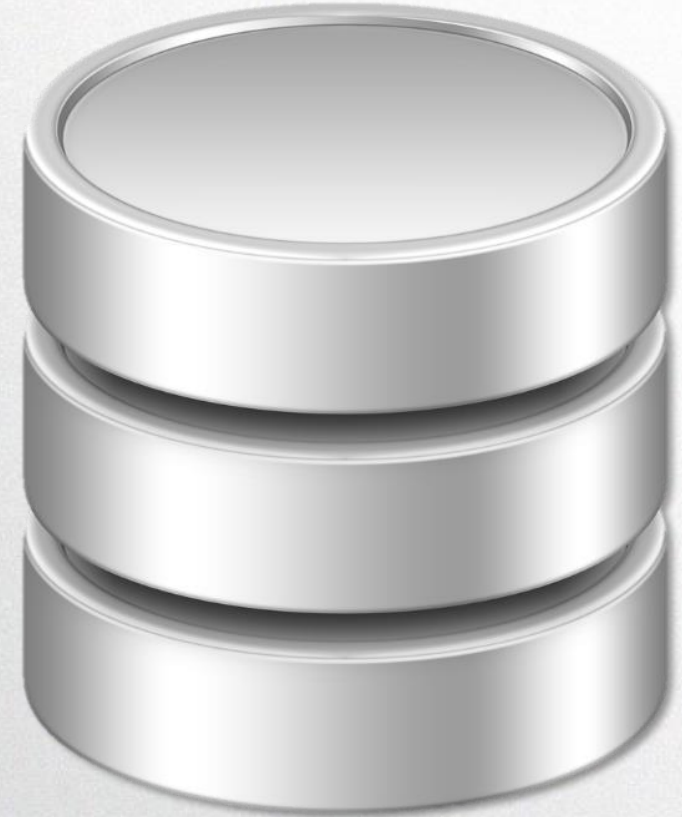
- Responds to user interaction & other events



Components of Web Applications

Database

- Handles long-term (persistent) storage of data
- Data is **organised**
- Easy to access, manage, and update



Components of Web Applications

Back-End Code



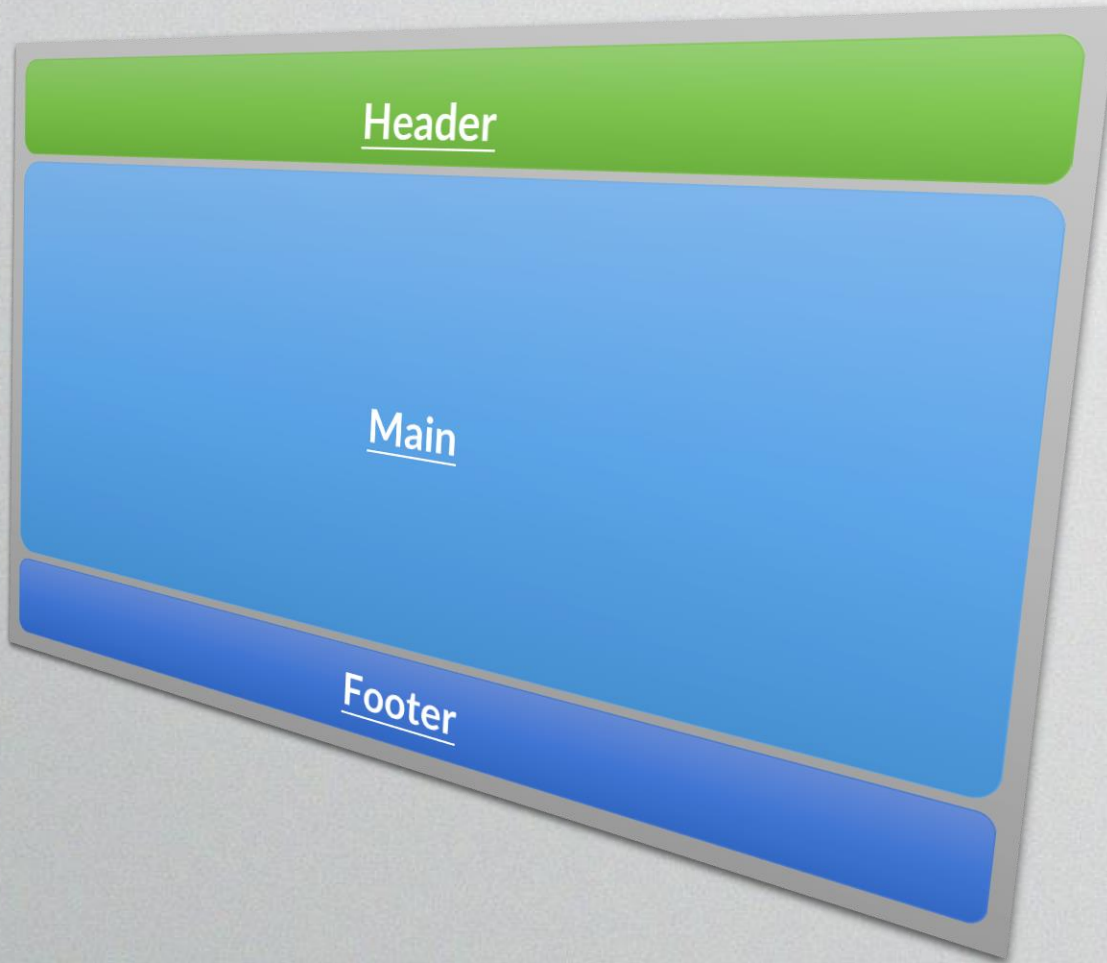
- Analyses User Requests
- Stores to and requests data from database
- Places dynamic content into static website
- *Much, much more!*



Components of Web Applications



Components of Web Applications



The Visible Front-End

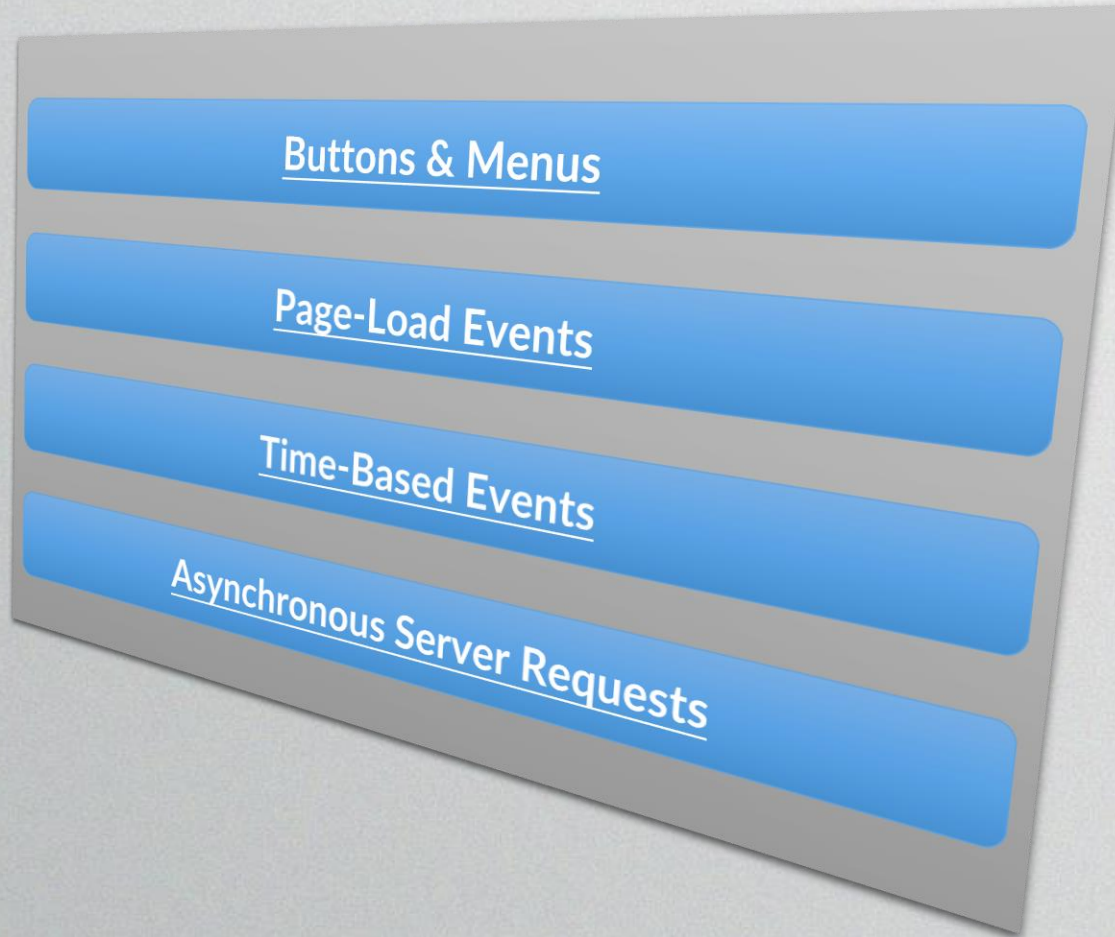
- What the user sees
- Content and Layout
- Built with HTML & CSS



Components of Web Applications



Components of Web Applications



The Invisible Front-End

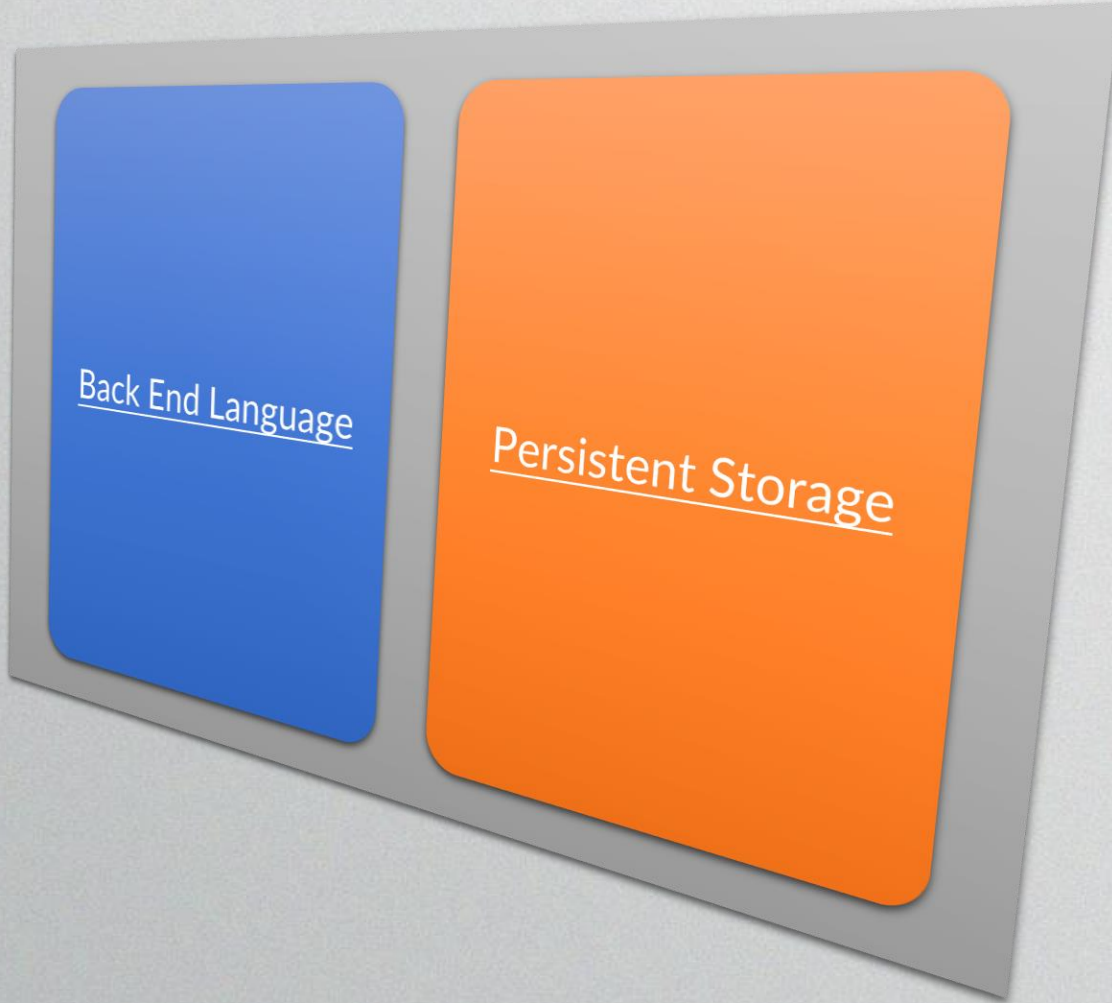
- Web page interaction
- Animations
- Built with JavaScript



Components of Web Applications



Components of Web Applications



The Invisible Back-End

- Dynamic, personalised content
- Validation & Security
- Built with Back-End Code (e.g. PHP) and Database



Delivering Websites to a Browser



Browser



Browser

is a software application that interprets **HTML**, **CSS**, & **JavaScript** on a client's computer

Also responsible for sending **requests** made by a client



Delivering Websites to a Browser

Browser



- Sends requests via HTTP(S)
- Retrieves data from servers via HTTP(S)
- **Interprets** HTML, CSS, & JavaScript
- Stores cookies
- Bunch of other stuff



Delivering Websites to a Browser

Popular Browsers



User's Computer

User's Computer

Browser
(e.g. Mozilla
Firefox)

User's Computer

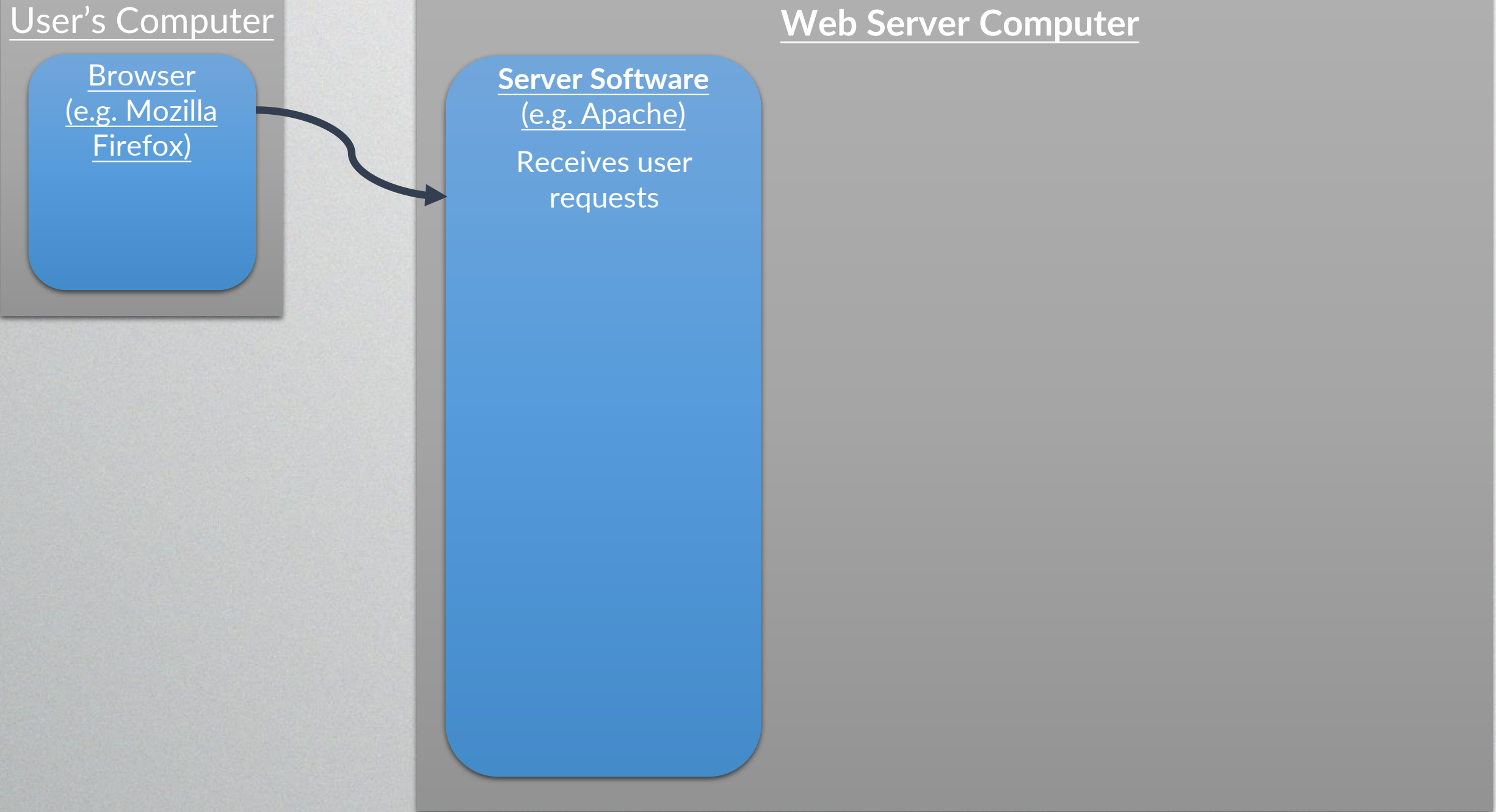
Browser
(e.g. Mozilla
Firefox)



Web Server Computer

User's Computer

Browser
(e.g. Mozilla
Firefox)



```
graph LR; subgraph UC [User's Computer]; B[Browser (e.g. Mozilla Firefox)]; end; subgraph WSC [Web Server Computer]; S[Server Software (e.g. Apache) Receives user requests]; end; B --> S;
```

The diagram illustrates the interaction between a user's computer and a web server computer. On the left, a box labeled 'User's Computer' contains a blue rounded rectangle representing a 'Browser (e.g. Mozilla Firefox)'. An arrow points from this browser to a larger blue rounded rectangle on the right, labeled 'Web Server Computer'. This rectangle contains the text 'Server Software (e.g. Apache) Receives user requests'.

Web Server Computer

Server Software
(e.g. Apache)
Receives user
requests

User's Computer

Browser
(e.g. Mozilla
Firefox)

```
graph LR; subgraph UserComputer [User's Computer]; Browser["Browser  
(e.g. Mozilla  
Firefox)"]; end; subgraph WebServerComputer [Web Server Computer]; subgraph ServerSoftware ["Server Software  
(e.g. Apache)"]; direction TB; S1["Receives user requests"]; S2["Sends to back end code"]; end; subgraph BackEndModule ["Back End Module  
(e.g. PHP)"]; direction TB; B1["Interprets user requests"]; end; Browser --> S1; S2 --> B1; end;
```

Web Server Computer

Server Software
(e.g. Apache)

Receives user requests

Sends to back end code

Back End Module
(e.g. PHP)

Interprets user requests

User's Computer

Browser
(e.g. Mozilla
Firefox)

```
graph LR; subgraph UserComputer [User's Computer]; Browser["Browser (e.g. Mozilla Firefox)"]; end; subgraph WebServerComputer [Web Server Computer]; subgraph ServerSoftware [Server Software (e.g. Apache)]; direction TB; S1[Receives user requests]; S2[Sends to back end code]; end; subgraph BackEndModule [Back End Module (e.g. PHP)]; direction TB; B1[Interprets user requests]; B2[Stores appropriate data in database]; end; Database["Database (e.g. MariaDB)"]; ServerSoftware --> BackEndModule; BackEndModule --> Database; end; Browser --> ServerSoftware;
```

Web Server Computer

Server Software
(e.g. Apache)

Receives user requests

Sends to back end code

Back End Module
(e.g. PHP)

Interprets user requests

Stores appropriate data in database

Database
(e.g. MariaDB)
Receives and stores data

User's Computer

Browser
(e.g. Mozilla
Firefox)

Web Server Computer

Server Software
(e.g. Apache)

Receives user requests
Sends to back end code

Back End Module
(e.g. PHP)

Interprets user requests
Stores appropriate data in database
Uses appropriate web page template

Database
(e.g. MariaDB)
Receives and stores data

Web Pages

Statically defined HTML, CSS, & JavaScript
Content placeholder for dynamic content addition using back end code

User's Computer

Browser
(e.g. Mozilla
Firefox)

Web Server Computer

Server Software
(e.g. Apache)

Receives user requests
Sends to back end code

Back End Module
(e.g. PHP)

Interprets user requests
Stores appropriate data in database
Uses appropriate web page template
Requests appropriate data from database

Database
(e.g. MariaDB)

Receives and stores data
Responds to data requests

Web Pages

Statically defined HTML, CSS, & JavaScript
Content placeholder for dynamic content addition using back end code

User's Computer

Browser
(e.g. Mozilla
Firefox)

Loads new
content

Web Server Computer

Server Software
(e.g. Apache)

Receives user
requests
Sends to back end
code
Sends web page
to user

Back End Module
(e.g. PHP)

Interprets user
requests
Stores appropriate data
in database
Uses appropriate web
page template
Requests appropriate
data from database
Adds dynamic content
to static web pages
Returns dynamic web
page to server
software

Database
(e.g. MariaDB)
Receives and stores
data
Responds to data
requests

Web Pages

Statically defined
HTML, CSS, &
JavaScript
Content placeholder
for dynamic content
addition using back
end code

Delivering Websites to a Browser



Delivering Websites to a Browser



How Do Websites Work?

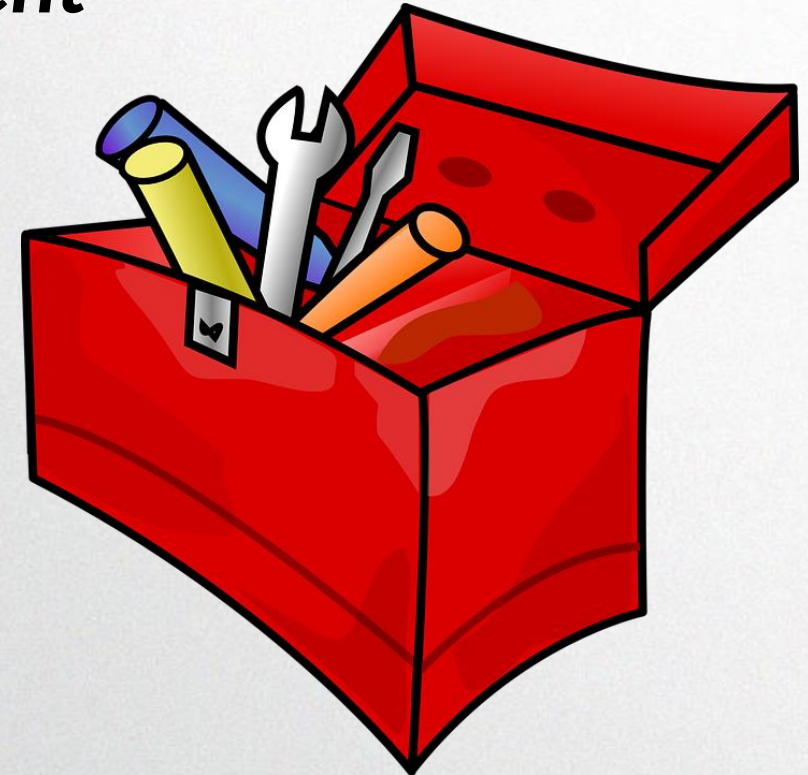
- ✓ Websites & Web Applications
- ✓ Components of a Web Application
- ✓ Delivering Websites to a Browser

- Summary
- Q&A



Setting Up Your Development Environment

- Installing a Text Editor
- A Computer's File System
- Navigating the Interface



Building the Website

- The next session is “**What is the Front-End?**”
 - HTML: A Container for Content
 - [CSS: The Language of Web Design](#)
 - JavaScript: For Dynamic Interactivity
- Toolkit 2: “**Creating Your First HTML Document**”
- Attend all lessons **LIVE** to grow your knowledge
- Shaw Academy [12-Month Membership](#) Prize during **Lesson 6**
- Recordings are available within 24 hours of the live webinar
 - Login to www.shawacademy.com using the link at the Top Right Corner – **Members Area**
(Use your email address and password)



Next Lesson is

What is the Front-End?

- An in-depth look at the technologies and languages used to build the visual end-user interface of your website
- You will understand the purpose and make-up of **HTML**, **CSS**, & **JavaScript**



www.shawacademy.com



www.facebook.com/shawacademy



www.twitter.com/shawacademy



support@shawacademy.com



See local numbers on website



