A picture containing logo

Description automatically generated

|  |
| --- |
| School Of Computing and Informatics |
| Website Design and Development |
| Assignment 1 |
|  |

|  |
| --- |
| Student Name: Obada Alhalaybeh  Student ID: 23110107  Section Number: 3  GitHub Link:  Website Link: <http://readrealm.online/> |

Contents

[1. DNS 2](#_Toc123593618)

[1.1 Types And Purpose Of DNS 2](#_Toc123593619)

[1.2 How Domain Names Are Organised and Managed 2](#_Toc123593620)

[2. Communication Protocols, Server Hardware, Operating System, And Web Server Software 2](#_Toc123593621)

[2.1 Communication Protocols Definition 2](#_Toc123593622)

[2.2 Relationship Between Protocols, Hardware, Operating System, Web Server Software 2](#_Toc123593623)

[2.2.1 Purpose And Relationship 2](#_Toc123593624)

[2.2.2 Relation To Designing, Accessing, and Publishing the Website 2](#_Toc123593625)

[3. Website Development Technologies, Tools, Techniques, And Frameworks 2](#_Toc123593626)

[3.1 Website Development Technologies 2](#_Toc123593627)

[3.1.1 Available Website Technologies 3](#_Toc123593628)

[3.1.2 Technologies Used to Design, Develop and Manage Your Website 3](#_Toc123593629)

[3.2 Website Development Tools 3](#_Toc123593630)

[3.2.1 Available Tools for Website Design and Development 3](#_Toc123593631)

[3.2.2 Tools Used to Design and Develop Your Website 3](#_Toc123593632)

[3.3 Website Development Techniques 3](#_Toc123593633)

[3.3.1 Available Techniques for Website Design and Development 3](#_Toc123593634)

[3.3.2 Techniques Used in Your Website 3](#_Toc123593635)

[3.4 Website Development Frameworks 3](#_Toc123593636)

[3.4.1 Available Frameworks for Website Design and Development 3](#_Toc123593637)

[3.4.2 Frameworks Used in Your Website. 3](#_Toc123593638)

[4. Search Engine Optimization 4](#_Toc123593639)

[4.1 Search Engine Optimization Definition and Purpose 4](#_Toc123593640)

[4.2 Indexing Definition 4](#_Toc123593641)

[4.3 Techniques For Improving the Ranking 4](#_Toc123593642)

[4.4 Evidence For the Used SEO Techniques in Your Website 4](#_Toc123593643)

[5. Front-End And Back-End Website Technologies 4](#_Toc123593644)

[5.1 Front-End And Back-End Definition and Related Technologies. 4](#_Toc123593645)

[5.2 Presentation And Application Layers Definition. 4](#_Toc123593646)

[5.3 Relationship. 4](#_Toc123593647)

[6. Online Website Creation Tools VS Custom-Built Websites 4](#_Toc123593648)

[6.1 Online Website Creation Tools 4](#_Toc123593649)

[6.2 Custom-Built Websites 4](#_Toc123593650)

[6.3 Comparison With Regards to Design Flexibility, Performance, Functionality, User Experience (UX) And User Interface (UI). 4](#_Toc123593651)

[7. Design And Development Process 5](#_Toc123593652)

[7.1 Website Vs. Design Document 5](#_Toc123593653)

[7.2 Evaluation and Analysis 5](#_Toc123593654)

[Hosting 5](#_Toc123593655)

[References 5](#_Toc123593656)

# DNS

## Types And Purpose Of DNS

The purpose of the DNS is to translate human-readable domain names like ([www.example.com](http://www.example.com)) into IP addresses like (200.10.0.25) that the computers understand and use to identify each other on the network.

Types:

* **Recursive resolver**: this is the first point of contact for user’s DNS query, it also act as the middle man between a client and DNS nameserver, after it receive the DNS query it either send it to the root then to the TLD finally it send a request for the authoritative nameserver after it get the response from the authoritative nameserver the recursive resolver send the response to the clint during the this process the recursive resolver will cache information received, and when the client request the same IP address for the same domain name the was visited recently the resolver will cut its way around the nameservers and just deliver the requested IP address from the cache.
* **Root**: There are 13 root server worldwide and they redirect queries to the appropriate TLD server based on the domain extension ([www.example.com](http://www.example.com)) the .com here.
* **TLD (top level domain)**: the TLD servers manage specific top level domains like (.org .com) and provide the address of the authoritative server for the domain.
* **Authoritative**: this server hold information specific to the domain name, and it will provide the IP address for the domain.

## How Domain Names Are Organised and Managed

Domain names are organized in a hierarchical structure, starting from the root and moving down to Top-Level Domains (TLDs), second-level domains, and subdomains. This format guarantees that domain names are distinctive, manageable, and available everywhere. Here’s how domain names are organized:The root directs queries to TLD servers, which then point to the appropriate second-level domains. Second-level domains are unique names registered by individuals or organizations, while subdomains allow for further organization of website content.

And here how is the DNS managed: Domain names are managed by registrars like hostinger and regulated by ICANN (Internet Corporation for Assigned Names and Numbers).

Example for the levels

Root 🡪[www.example.com](http://www.example.com). (the dot after com)

TLD 🡪[www.example.com](http://www.example.com) the com

Second level 🡪[www.example.com](http://www.example.com) the example

Subdomain 🡪blog.example.com the blog here

# Communication Protocols, Server Hardware, Operating System, And Web Server Software

## Communication Protocols Definition

Communication protocols are standardized rules and procedures that allow devices to communicate over a network. They define how data is formatted, transmitted, received, and acknowledged.

Here is some communication protocols:

* HTTP (Hypertext Transfer Protocol): Used for transferring web pages from a server to a client.
* HTTPS (HTTP Secure): A secure version of HTTP that encrypts data using SSL/TLS.
* FTP (File Transfer Protocol): Used for uploading and downloading files between a client and a server.
* SMTP (Simple Mail Transfer Protocol): Used for sending emails.
* TCP/IP (Transmission Control Protocol/Internet Protocol): The foundational protocol suite for internet communication.

## Relationship Between Protocols, Hardware, Operating System, Web Server Software

## Purpose And Relationship

The relationship between communication protocols, server hardware, operating systems, and web server software is essential for hosting, managing, and delivering websites. And they work together like that: communication protocols define how data is transmitted between the client and the server. For example, HTTP/S ensures that web pages are delivered securely and efficiently, hardware is the physical machine that host the website or the device you are trying to reach the website from, operating system is the software that manage the server hardware and you device and provide a platform to rum applications on, and webserver software is the software that handles HTTP/S requests.

For the relationship The operating system runs on the server hardware and provides the environment for web server software to operate. The web server software uses communication protocols like http/s to communicate with clients and deliver web content. Together, these components ensure that websites are accessible, functional, and secure.

## Relation To Designing, Accessing, and Publishing the Website

The interaction between communication protocols, hardware, operating systems, and web server software is critical for designing, accessing, and publishing a website.

Here is how they relate to each one of them:

* **Designing**: developers use communication protocols to test website interactions, while the choice of operating system and web server software influences development tools.
* **Accessing**: When a user accesses a website, the web server software processes HTTP/S requests, retrieves files, and sends them to the client, with the operating system managing resources and the hardware providing infrastructure.
* **Publishing**: Developers upload files to the server using protocols like FTP, and the web server software serves these files to users, supported by the operating system and server hardware.

# Website Development Technologies, Tools, Techniques, And Frameworks

## Website Development Technologies

## Available Website Technologies

* Front-End Technologies:
  + HTML: Used for structuring web content.
  + CSS: Used for styling and designing web pages.
  + JavaScript: Used for adding interactivity and dynamic behavior to websites.
  + React.js: A JavaScript library for building user interfaces.
  + Bootstrap: A CSS framework for responsive design.
* Back-End Technologies:
  + Python: Used with frameworks like Django and Flask for back-end development.
  + PHP: A server-side scripting language for web development.
  + Java: Used for building robust and scalable web applications.
  + C#: Used with frameworks like ASP.NET for web development.
* Databases:
  + MySQL: A relational database management system.
  + MongoDB: A NoSQL database for storing unstructured data.
  + PostgreSQL: An advanced relational database system.

## Technologies Used to Design, Develop and Manage Your Website

* Front-End:
  + HTML: For structuring the content of the website.
  + CSS: For styling and ensuring a visually appealing design.
  + JavaScript: For adding interactivity and dynamic features.
  + Bootstrap: For creating a responsive and mobile-friendly layout. (nav bar)
* Back-End:
  + PHP: For server-side logic and handling HTTP requests.
  + phpMyAdmin (MYSQL): For managing the MySQL database through a web-based interface.

## Website Development Tools

## Available Tools for Website Design and Development

* Code Editors:
  + Visual Studio Code.
  + Sublime Text.
  + Atom.
* Design Tools:
  + Figma.
  + Adobe XD.
  + Just in mind.
* Version Control:
  + Git: For tracking changes in code.
  + GitHub: For hosting and sharing code repositories.
* SEO Tools:
  + Google Search Console: For monitoring and optimizing website performance in search engines.
* Developer tools of the browser.

## Tools Used to Design and Develop Your Website

* Visual Studio Code: For writing and editing code.
* Just in mind: For creating wireframes.
* Chrome Developer Tools: For debugging and testing the front-end code.
* Xampp: to create a localhost to view the php changes.
* Xmind: to create a site map for my project.

## Website Development Techniques

## Available Techniques for Website Design and Development

* Responsive Design: Ensuring the website works well on all devices (desktop, tablet, mobile).
* Mobile-First Design: Designing for mobile devices first, then scaling up for larger screens.
* SEO Techniques: Using meta tags, alt text, and keyword optimization to improve search engine rankings.
* Good Navigation: Ensuring the website is easy to navigate with clear menus and links.
* Consistency Across Pages: Maintaining a consistent design and layout across all pages.
* Video/Audio Fallback: Providing fallback options for multimedia content.
* Using Relative Units: Using percentages, em, or rem instead of fixed units for better scalability.

## Techniques Used in Your Website

* Responsive Design: Ensuring the website works well on all devices (desktop, tablet, mobile).
* Good Navigation: Ensuring the website is easy to navigate with clear menus and links.
* Video/Audio Fallback: Providing fallback options for multimedia content.
* Using Relative Units: Using percentages, em, or rem instead of fixed units for better scalability.

## Website Development Frameworks

## Available Frameworks for Website Design and Development

* Front-End Frameworks:
  + Bootstrap: A CSS framework for responsive design.
  + React.js: A JavaScript library for building user interfaces.
  + Angular.js: A TypeScript-based framework for building dynamic web apps.
  + Vue.js: A progressive JavaScript framework for building UIs.
* Back-End Frameworks:
  + Laravel: A PHP framework for web application development.
  + Django: A Python framework for rapid development and clean design.
  + Flask: A lightweight Python framework for web development.
  + ASP.NET: A C# framework for building web applications.

## Frameworks Used in Your Website.

* Front-End:
  + Bootstrap: For creating a responsive and mobile-friendly layout.

# Search Engine Optimization

## Search Engine Optimization Definition and Purpose

* Search Engine Optimization (SEO) is the process of optimizing a website to improve its visibility and ranking on search engine results pages (SERPs). The primary purpose of SEO is to increase organic (non-paid) traffic to a website by making it more relevant and authoritative in the eyes of search engines like Google, Bing, and Yahoo.

SEO involves various techniques and strategies, including keyword optimization, content creation, technical improvements, and link building. By implementing SEO best practices, websites can attract more visitors, improve user experience, and ultimately achieve their business goals.

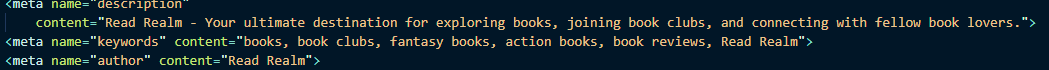
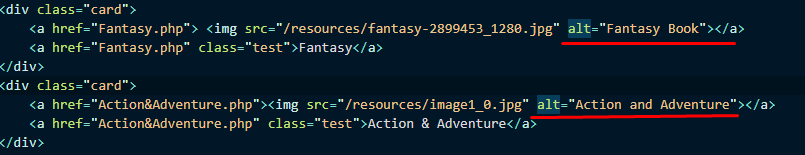
## Indexing Definition

* Indexing refers to the method by which search engines examine, sort, and save web pages within their databases. When a search engine indexes a site, it navigates through the site's content, gathers relevant data, and records it in an index. This index is subsequently utilized to fetch and present relevant results whenever users conduct a search query.
* For a website to show up in search engine results, it must be indexed first. If a website isn’t indexed, it won’t be visible to users looking for associated content. Thus, making sure that your website is correctly indexed is an essential part of the SEO process.

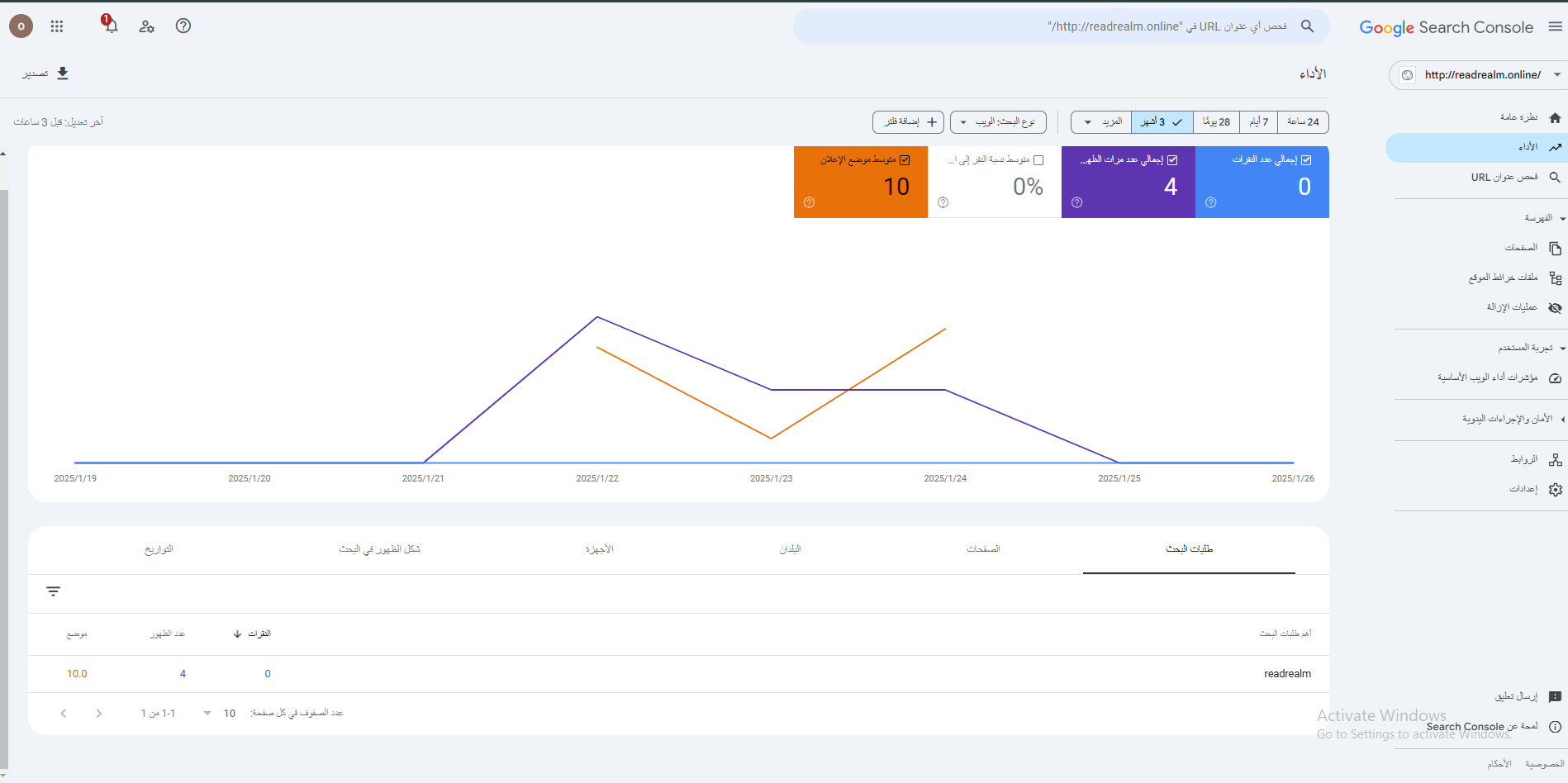
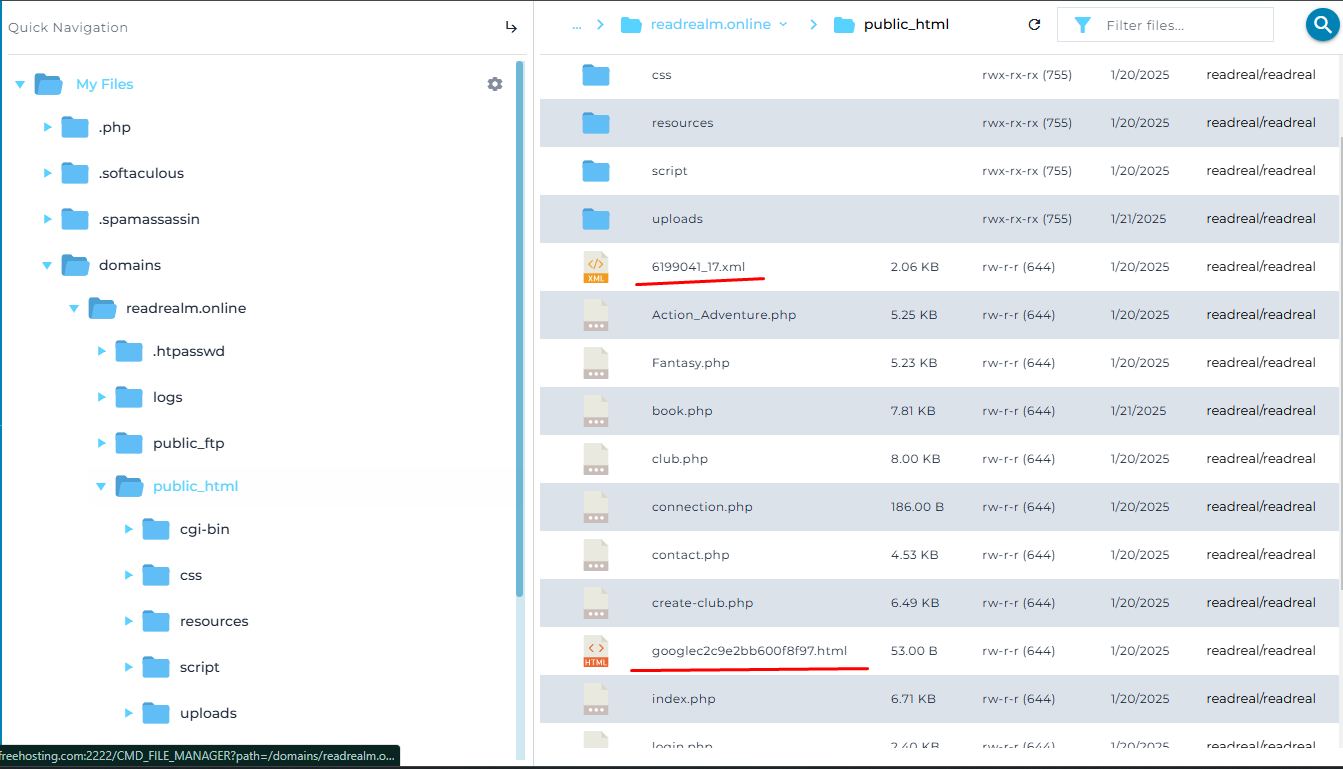
## Techniques For Improving the Ranking

* Keyword Research and Optimization:
  + Identify relevant keywords that users are searching for.
  + Incorporate these keywords naturally into the website's content, including titles, headings, meta descriptions, and body text.
* High-Quality Content Creation:
  + Create valuable, informative, and engaging content that addresses the needs and queries of your target audience.
  + Regularly update the website with fresh content to keep it relevant.
* On-Page SEO:
  + Optimize meta tags (title tags, meta descriptions, and header tags).
  + Use descriptive and keyword-rich URLs.
  + Add alt text to images for better accessibility and SEO.
* Technical SEO:
  + Ensure the website is mobile-friendly and responsive.
  + Improve website speed and performance.
* User Experience (UX) Optimization:
  + Ensure the website is easy to navigate and user-friendly.
  + Reduce bounce rates by improving page load times and providing a seamless browsing experience.
* Social Media Integration:
  + Promote content on social media platforms to increase visibility and drive traffic.
  + Encourage social sharing to improve the website's reach and engagement.

## Evidence For the Used SEO Techniques in Your Website

  
  
  
A computer code with text

Description automatically generated A screenshot of a computer program

Description automatically generated 

# Front-End And Back-End Website Technologies

## Front-End And Back-End Definition and Related Technologies.

* Front end
  + Front end development relates to the aspect of web development that emphasizes the user interface (UI) and user experience (UX). It includes all aspects that users engage with directly, including the arrangement, aesthetics, and interactive elements of a website. Front-end development mainly focuses on the appearance and user experience of the website.
  + Related Front-End Technologies:
    - HTML (Hypertext Markup Language): Used for structuring content on the web.
    - CSS (Cascading Style Sheets): Used for styling and designing web pages.
    - JavaScript: Used for adding interactivity and dynamic behavior to websites.
    - React.js: A JavaScript library for building user interfaces.
    - Bootstrap: A CSS framework for creating responsive and mobile-friendly designs.
* Back end
  + Back end development relates to the server aspect of website development. It encompasses all the activities that occur behind the scenes, including database interactions, server logic, and application features. Back-end development guarantees the proper functioning of the website and manages data processing.
  + Related Back-End Technologies:
    - PHP: A server-side scripting language used for web development.
    - Python: Often used with frameworks like Django and Flask for back-end development.
    - Node.js: A JavaScript runtime environment for building server-side applications.
    - MySQL: A relational database management system for storing and retrieving data.
    - MongoDB: A NoSQL database for storing unstructured data.

## Presentation And Application Layers Definition.

* Presentation Layer:
  + The presentation layer is the front-end part of a web application. It is responsible for displaying data to the user and handling user interactions. This layer includes the UI components, such as buttons, forms, and navigation menus, as well as the visual design of the website.
* Application Layer:
  + The application layer is the back-end part of a web application. It handles the business logic, data processing, and communication between the front-end and the database. This layer is responsible for processing user requests, performing calculations, and managing data storage.

## Relationship.

* The relationship between front-end and back-end technologies is fundamental to the functionality and user experience of a web application. These two layers work together seamlessly to ensure that users can interact with the application effectively and that data is processed and managed efficiently. The front-end, also known as the presentation layer, is responsible for everything the user sees and interacts with directly. This includes the layout, design, and interactive elements of the website, which are built using technologies like HTML, CSS, and JavaScript. When a user performs an action, such as submitting a form or clicking a button, the front-end captures this input and sends it to the back-end for processing.

The back-end, or application layer, handles the logic, data processing, and communication with the database. It receives the user input from the front-end, processes it, and performs necessary operations such as querying the database, performing calculations, or updating records. For example, when a user logs in, the back-end verifies the credentials against the database and determines whether to grant access. Once the back-end completes its processing, it sends the results back to the front-end, which then displays the information to the user in a visually appealing format. This continuous interaction between the front-end and back-end is facilitated through APIs (Application Programming Interfaces) or HTTP requests, ensuring smooth communication and data exchange.

The relationship between front-end and back-end technologies is fundamental to the functionality and user experience of a web application. These two layers work together seamlessly to ensure that users can interact with the application effectively and that data is processed and managed efficiently. The front-end, also known as the presentation layer, is responsible for everything the user sees and interacts with directly. This includes the layout, design, and interactive elements of the website, which are built using technologies like HTML, CSS, and JavaScript. When a user performs an action, such as submitting a form or clicking a button, the front-end captures this input and sends it to the back-end for processing.

The back-end, or application layer, handles the logic, data processing, and communication with the database. It receives the user input from the front-end, processes it, and performs necessary operations such as querying the database, performing calculations, or updating records. For example, when a user logs in, the back-end verifies the credentials against the database and determines whether to grant access. Once the back-end completes its processing, it sends the results back to the front-end, which then displays the information to the user in a visually appealing format. This continuous interaction between the front-end and back-end is facilitated through APIs (Application Programming Interfaces) or HTTP requests, ensuring smooth communication and data exchange.

# Online Website Creation Tools VS Custom-Built Websites

## Online Website Creation Tools

* Online website creation tools are platforms that allow users to build websites without needing to write code. These tools provide pre-designed templates, drag-and-drop editors, and built-in features that make it easy for non-technical users to create websites quickly.
* Examples:
  + Wix
  + WordPress
* Advantages:
  + Ease of Use: No coding knowledge is required, making it accessible to beginners.
  + Speed: Websites can be created and launched quickly using pre-designed templates.
  + Cost-Effective: Often cheaper than custom-built websites, especially for small projects.
  + Maintenance: The platform handles updates, security, and hosting, reducing the burden on the user.
* Disadvantages:
  + Limited Design Flexibility: Users are restricted to the templates and features provided by the platform.
  + Performance: Websites may load slower due to bloated code and unnecessary features.
  + Scalability: May not be suitable for large or complex websites.

## Custom-Built Websites

* Custom-built websites are developed from scratch using programming languages (like HTML, CSS, JavaScript, PHP) and frameworks (like React, Django, or Laravel).   
  These websites are tailored to meet specific requirements and offer full control over design, functionality, and performance.
* Examples:
  + Front-End: HTML, CSS, JavaScript, React.js, Bootstrap.
  + Back-End: PHP, Python, Node.js, MySQL,
* Advantages:
  + Design Flexibility: Complete control over the design and layout.
  + Performance: Optimized code and lightweight frameworks ensure faster loading times.
  + Functionality: Ability to implement custom features and integrations tailored to specific needs.
  + Scalability: Suitable for large and complex websites with high traffic and advanced functionality.
  + SEO Optimization: Full control over SEO practices, such as meta tags, structured data.
* Disadvantages:
  + Time-Consuming: Development takes longer compared to using online tools.
  + Cost: More expensive due to the need for skilled developers and ongoing maintenance.
  + Technical Knowledge Required: Requires expertise in coding and web development.
  + Maintenance: The developer or team is responsible for updates, security, and hosting.

## Comparison With Regards to Design Flexibility, Performance, Functionality, User Experience (UX) And User Interface (UI).

|  |  |  |
| --- | --- | --- |
| Aspect | Online Website Creation Tools | Custom-Built Websites |
| Design Flexibility | Limited to pre-designed templates and drag-and-drop editors. | Full control over design, allowing for unique and creative layouts. |
| Performance | Could experience longer loading times because of excessive code and unnecessary functionalities. | Optimized for performance with lightweight code and efficient frameworks. |
| Functionality | Limited to the features provided by the platform, difficult to implement custom functionality. | Fully customizable with the ability to add advanced features and integrations. |
| User Experience (UX) | Basic UX features are built-in, but customization is limited. | Tailored UX design to meet specific user needs and preferences. |
| User Interface (UI) | Pre-designed UI templates are available, but customization options are limited. | Complete control over UI design, ensuring a unique and branded look. |

# Design And Development Process

## Website Vs. Design Document

* The design document serves as the blueprint for your web application. It outlines the goals, scope, client and user requirements, sitemap, and wireframes. Comparing the final website with the design document helps ensure that the project aligns with the initial vision and meets the specified requirements.

|  |  |  |
| --- | --- | --- |
| Aspect | Design Document | Final Website |
| Goal and Scope | Create a responsive, fully functional web app with user account management. | Successfully implemented all goals and scope. |
| Client Requirements | User authentication, dynamic content, database interactions, responsive design. | All requirements met with additional enhancements for usability. |
| Sitemap | Home, About, Services, Contact, Sign-Up, Sign-In | Final website structure matches the sitemap exactly. |
| Wireframes | Visual layout of headers, navigation, content sections, and footers. | Final design closely follows wireframes with minor improvements. |
| Functionality | User authentication, dynamic content, database interactions. | All functionalities implemented successfully. |
| ****Design and Layout**** | Clean, modern, and user-friendly design with responsiveness. | Final website adheres to design principles and is fully responsive. |

## Evaluation and Analysis

* The process of design and development was organized and adhered to a methodical approach, beginning with the formation of the design document and concluding with the launch of the web application. The procedure encompassed these phases:
  + Planning: Defining the goals, scope, and requirements of the project.
  + Design: Creating the sitemap and wireframes to visualize the website's structure and layout.
  + Development: Implementing the front-end and back-end functionalities based on the design document.
  + Testing: Conducting thorough testing to ensure the website's functionality, performance and usability.
  + Deployment: Publishing the website on the internet and making it accessible to users.
* Technical challenges I faced:
  + Responsive Design: Ensuring the website was fully responsive across different devices (desktop, tablet, mobile) required careful testing and adjustments to the CSS and layout while using the media query.
  + Database Integration: Implementing database interactions the CRUD oprations (read, insert, update, delete) required debugging to ensure efficient data handling.
  + User Authentication: Developing a secure user authentication system (sign-up, sign-in, sign-out) involved implementing password hashing and session management to protect user data, the hashing was somewhat hard to fix when it broke.
* Overall Success:
  + The web application meet all the requirements and provide user friendly experience.
  + For the key achievements:
    - Functionality: All required features, such as user account management and dynamic content display, were implemented successfully.
    - Usability: The website is easy to navigate, with a clear and intuitive interface.
    - Responsiveness: The website is fully responsive and works seamlessly on all devices.
* For the future there is some enhancements that could be done like:
  + Adding more advanced features from the user’s feedback.
  + More SEO optimizations to make suer to rank higher each time.

## Analyse a range of tools (such as IDE’s, design tools, and other software) and techniques available to design and develop a custom-built web app. (Report).

Designing and developing a custom-built web application requires a combination of tools and techniques to ensure efficiency, functionality, and a high-quality user experience. Below is an analysis of the tools and techniques used in the development process:

1. Integrated Development Environments (IDEs)

IDEs are software applications that provide comprehensive facilities for software development. They typically include a code editor, debugger, and build automation tools.

Popular IDEs for Web Development:

* Visual Studio Code (VS Code):
  + Features: Lightweight, supports multiple programming languages, extensive extensions for web development (e.g., HTML, CSS, JavaScript, PHP).
  + Advantages: Free, cross-platform, excellent debugging tools, and Git integration.
  + Usage: Used for writing and editing code for both front-end and back-end development.
* WebStorm:
  + Features: Advanced JavaScript support, integration with frameworks like React and Angular, built-in terminal, and version control.
  + Advantages: Powerful for JavaScript-heavy projects, intelligent code completion.
  + Usage: Ideal for front-end development with modern JavaScript frameworks.
* PhpStorm:
  + Features: PHP-specific IDE with support for databases, version control, and debugging.
  + Advantages: Excellent for back-end development with PHP, integrates well with MySQL.
  + Usage: Used for back-end development in PHP-based web applications.

2. Design Tools

Design tools are essential for creating wireframes, prototypes, and visual designs before development begins.

Popular Design Tools:

* Figma:
  + Features: Collaborative design tool for creating wireframes, prototypes, and high-fidelity designs.
  + Advantages: Real-time collaboration, cloud-based, easy to share designs with stakeholders.
  + Usage: Used for creating wireframes and UI designs for the web application.
* Adobe XD:
  + Features: Design and prototyping tool for web and mobile applications.
  + Advantages: Seamless integration with other Adobe tools, responsive design features.
  + Usage: Used for designing the layout and user interface of the web application.
* Just in Mind:
  + Features: Wireframing and prototyping tool with drag-and-drop functionality.
  + Advantages: Easy to use, supports interactive prototypes.
  + Usage: Used for creating low-fidelity wireframes for the web application.

3. Front-End Development Tools

Front-end development tools are used to create the user interface and ensure a responsive design.

Popular Front-End Tools:

* HTML/CSS/JavaScript:
  + Features: Core technologies for building the structure, style, and interactivity of web pages.
  + Advantages: Standard for web development, supported by all browsers.
  + Usage: Used for creating the structure, design, and interactivity of the web application.
* Bootstrap:
  + Features: CSS framework for creating responsive and mobile-first websites.
  + Advantages: Pre-designed components, grid system, and responsive utilities.
  + Usage: Used for creating a responsive layout and styling the web application.
* React.js:
  + Features: JavaScript library for building user interfaces.
  + Advantages: Component-based architecture, reusable components, and efficient rendering.
  + Usage: Used for building dynamic and interactive user interfaces.

4. Back-End Development Tools

Back-end development tools are used to handle server-side logic, database interactions, and application functionality.

Popular Back-End Tools:

* PHP:
  + Features: Server-side scripting language for web development.
  + Advantages: Easy to learn, widely supported, integrates well with MySQL.
  + Usage: Used for server-side logic and database interactions.
* MySQL:
  + Features: Relational database management system for storing and retrieving data.
  + Advantages: Scalable, reliable, and widely used in web applications.
  + Usage: Used for managing the database and storing user data.
* phpMyAdmin:
  + Features: Web-based tool for managing MySQL databases.
  + Advantages: Easy to use, provides a graphical interface for database management.
  + Usage: Used for managing the MySQL database during development.

# Hosting (Add your domain and URL)

<http://readrealm.online/>

readrealm.online

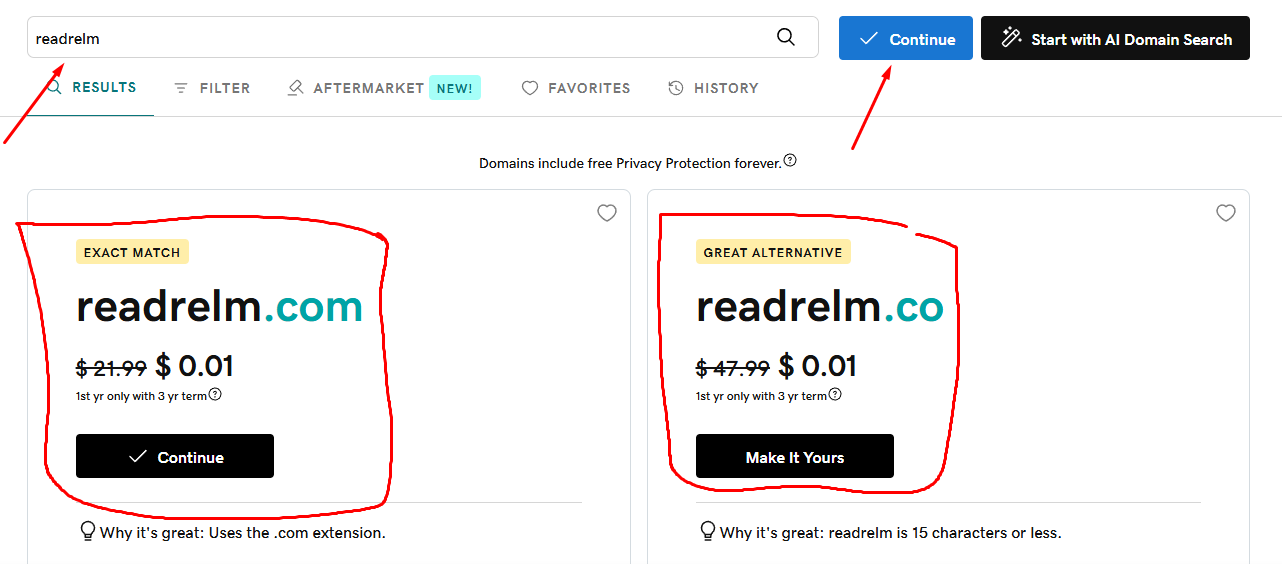
## Specify which hosting server you use

The web application is hosted on **FreeHosting.com**, a free hosting service that provides basic hosting features for small projects. FreeHosting.com offers the following features:

* Free Hosting: No cost for hosting, making it ideal for small projects or personal websites.
* 1 GB Disk Space: Sufficient storage for small to medium-sized websites.
* Unlimited Bandwidth: Ensures the website can handle traffic without additional costs.
* MySQL Database: Supports database-driven web applications.
* FTP Access: Allows for easy file uploads and management.

## Domain

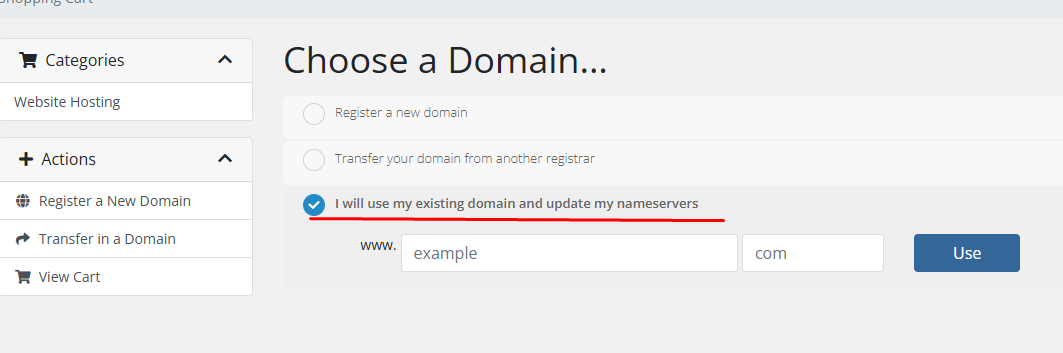
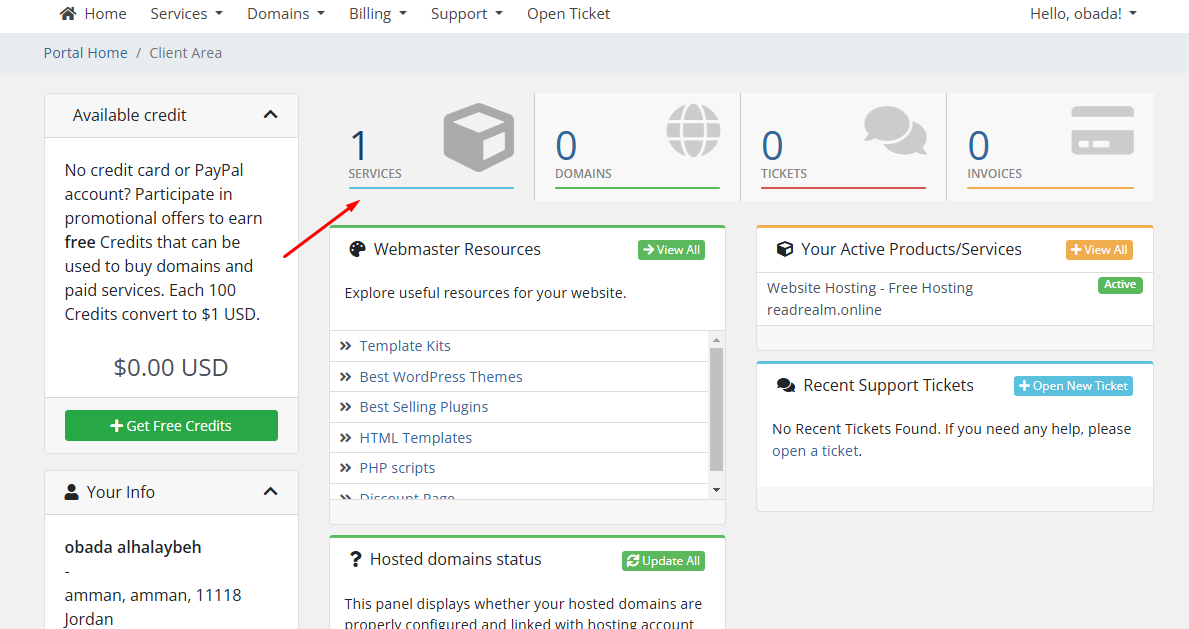
for the domain I went to **godaddy.com** to search if the name I picked is taken or not, the name I chose based on the type of my website and how is it easy to remember short and has a catching words in it so I went with readrealm.online.

here is the steps on how to buy a domain  
  
**since its for small project for the course** A screenshot of a computer

Description automatically generatedA screenshot of a computer screen

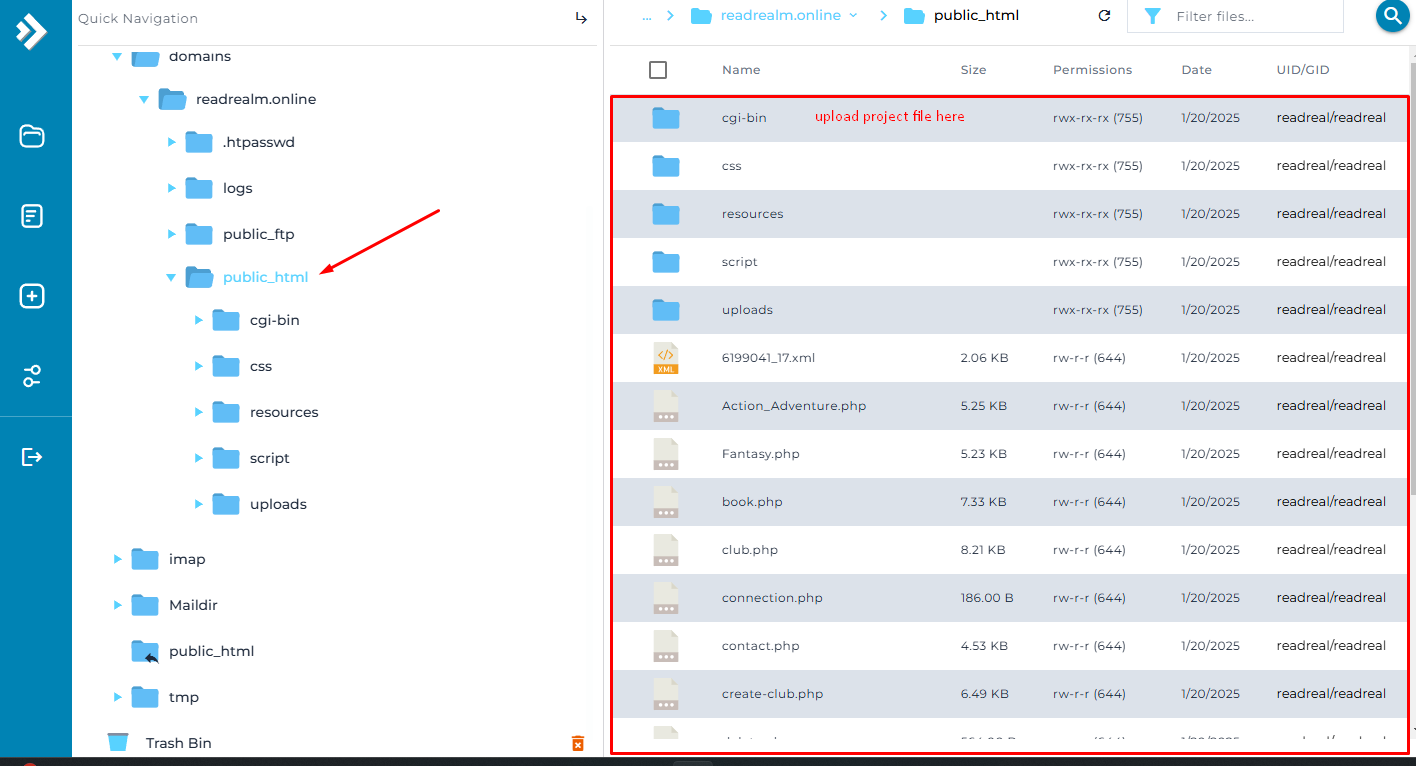
Description automatically generated  
then add payment method and you will have it.  
  
for the hosting:

first go to freeshoting.com then create an account

  
after that continue the process and fill in your information  
A screenshot of a computer

Description automatically generatedA screenshot of a web hosting

Description automatically generatedA screenshot of a computer

Description automatically generated  
you will add both the sitemap xml here and the google verification here too  
A screenshot of a computer

Description automatically generatedand change both of this to the hosting sever you are using

# References

Fortinet (2022). *What Is DNS? Definition & How DNS Works*. [online] Fortinet. Available at: <https://www.fortinet.com/resources/cyberglossary/what-is-dns>.

Cloudflare (n.d.). DNS Server Types | Cloudflare. *Cloudflare*. [online] Available at: <https://www.cloudflare.com/learning/dns/dns-server-types/>.

‌ www.emailhosting.com. (n.d.). *How Domain Names are Organized and Managed - Knowledgebase - EmailHosting.com*. [online] Available at: <https://www.emailhosting.com/service/index.php?rp=/knowledgebase/5/How-Domain-Names-are-Organized-and-Managed.html>.

‌ geeksforgeeks (2023). *Communication Protocols In System Design*. [online] GeeksforGeeks. Available at: <https://www.geeksforgeeks.org/communication-protocols-in-system-design/>.

Web Design, UI/UX, Branding, and App Development Blog. (2024). *What is Web Development? The Process, Technology, and Future | Ramotion Agency*. [online] Available at: <https://www.ramotion.com/blog/what-is-web-development/>.

‌ BrowserStack. (n.d.). *Top Web Development Tools in 2023*. [online] Available at: <https://www.browserstack.com/guide/web-development-tools>.

‌ Alokai.com. (2024). *Best 16 Frontend Frameworks for Web Development in 2024 | Alokai*. [online] Available at: <https://alokai.com/blog/top-frontend-frameworks>.

‌ Radixweb. (2022). *Best Backend Frameworks for Backend Development in 2023*. [online] Available at: <https://radixweb.com/blog/best-backend-frameworks>.

‌ Ellis, M. (2023). *What is SEO?* [online] MOZ. Available at: <https://moz.com/learn/seo/what-is-seo>.

growhackscale.com. (n.d.). *Indexing SEO: How To Optimise Your Website for Indexing - SEO Glossary*. [online] Available at: <https://growhackscale.com/glossary/indexing-seo>.

Business.gov.au (2024). *Improve your search engine rankings*. [online] business.gov.au. Available at: <https://business.gov.au/online-and-digital/business-website/improve-your-search-engine-rankings>.

‌ Backlinko. (2018). *21 Actionable SEO Techniques For 2020*. [online] Available at: <https://backlinko.com/seo-techniques>.

‌ MDN Web Docs. (2024). *Server-side website programming - Learn web development | MDN*. [online] Available at: <https://developer.mozilla.org/en-US/docs/Learn_web_development/Extensions/Server-side>.

‌

MDN Web Docs. (2024). *MDN Curriculum | MDN Curriculum*. [online] Available at: <https://developer.mozilla.org/en-US/curriculum/>.

‌

Networking. (n.d.). *What is presentation layer? | Definition from TechTarget*. [online] Available at: <https://www.techtarget.com/searchnetworking/definition/presentation-layer>.

‌

SearchNetworking. (n.d.). *What is Application layer? - Definition from WhatIs.com*. [online] Available at: <https://www.techtarget.com/searchnetworking/definition/Application-layer>.

‌

S, M. (2023). *Website Builder vs Custom Website: A Comprehensive Comparison*. [online] Hostinger Tutorials. Available at: <https://www.hostinger.com/tutorials/website-builder-vs-custom-website>.

M, B. (2020). *Advantages of Custom Built Sites vs Online Website Builder Tools*. [online] Icecube Digital. Available at: <https://www.icecubedigital.com/blog/advantages-of-custom-built-sites-vs-online-website-builder-tools/>.

‌

‌

‌

‌