MDN LEARNING PATH WEB DESIGN.

Set your goals, what you want to accomplish.

Decide what you want to do. Your ideas.

Lista de lo que quiero lograr con el site y ordenarlas por prioridades.

Por ejemplo: crear un site para PROMOCIONAR algo pero que a la vez use otros sites para ciertas tareas.

Ejemplo de un sitio musical:

Let people hear your music:

1. Record music
2. Prepare some audio files usable online (Could you do this with existing web services?)
3. Give people access to your music on some part of your website

Talk about your music:

1. Write a few articles to start the discussion
2. Define how articles should look
3. Publish those articles on the website (How to do this?)

Meet other musicians:

1. Provide ways for people to contact you (Email? Facebook? Phone? Mail?)
2. Define how people will find those contact channels from your website

SELL GOODIES:

1. Create the goodies
2. Store the goodies
3. Find a way to handle shipping
4. Find a way to handle payment
5. Make a mechanism on your site for people to place orders

TEACH MUSIC THROUGH VÍDEOS:

1. Record video lessons
2. Prepare video files viewable online (Again, could you do this with existing web services?)
3. Give people access to your videos on some part of your website

**IP ADDRESS**: es el numero único que tiene ‘cada computadora’ y se compone de 4 numers separados por un punto por ej: **192.0.2.172**  pero a su vez para no complicar, le podemos dar a ese numero un ‘alias’ o sea un nombre ‘propio’, eso se denomina **DOMAIN** (ej: google.com).

Internet es la infraestructura que permite usar Servicios como la WEB, email o IRC.

IRC(internet relay chat) es un servicio de chat que solo necesita una conexión a internet y un Cliente IRC.

IRCS server conecta usuarior via IRC channels (cada uno con su propio ID) usando el Protocolo TCP.

TCP (Transmission Control Protocol): para conectar e intercambiar data stream entre dos usuarios.

WEB SERVER:

A computer that hosts a WEB SITE on the Internet.

WEB SITE >>> WEB PAGES.

1. On the hardware side, a web server is a computer that stores web server software and a website's component files (for example, HTML documents, images, CSS stylesheets, and JavaScript files). A web server connects to the Internet and supports physical data interchange with other devices connected to the web.
2. On the software side, a web server includes several parts that control how web users access hosted files. At a minimum, this is an HTTP server. An HTTP server is software that understands URLs (web addresses) and HTTP (the protocol your browser uses to view webpages). An HTTP server can be accessed through the domain names of the websites it stores, and it delivers the content of these hosted websites to the end user's device.

At the most basic level, whenever a browser needs a file that is hosted on a web server, the browser requests the file via HTTP. When the request reaches the correct (hardware) web server, the (software) *HTTP server* accepts the request, finds the requested document, and sends it back to the browser, also through HTTP. (If the server doesn't find the requested document, it returns a [404](https://developer.mozilla.org/en-US/docs/Web/HTTP/Status/404) response instead.)

BASICAMENTE MANDAMOS DESDE NUESTRO BROWSER UN HTTP ‘REQUEST’ AL WEBSERVER Y ÉSTE NOS DEVUELVE UN HTTP ‘RESPONSE’ QUE SERIAN LOS FILES PARA PODER VISUALIZAR ESE DOCUMENTO. TODO ESTO SE HACE BAJO EL PROTOCOLO HTTP.

**To publish a website, you need either a static or a dynamic web server.**

A **static web server**, or stack, consists of a computer (hardware) with an HTTP server (software). We call it "static" because the server sends its hosted files as-is to your browser.

A **dynamic web server** consists of a static web server plus extra software, most commonly an *application server* and a *database*. We call it "dynamic" because the application server updates the hosted files before sending content to your browser via the HTTP server.

For example, to produce the final webpages you see in the browser, the application server might fill an HTML template with content from a database. Sites like MDN or Wikipedia have thousands of webpages. Typically, these kinds of sites are composed of only a few HTML templates and a giant database, rather than thousands of static HTML documents. This setup makes it easier to maintain and deliver the content.

BASICAMENTE: TU ELIJES UN WEB SERVER Y SUBES TUS ARCHIVOS AHI (HOSTING).

 A web server provides support for [HTTP](https://developer.mozilla.org/en-US/docs/Glossary/HTTP) (**H**yper**t**ext **T**ransfer **P**rotocol). As its name implies, HTTP specifies how to transfer hypertext (linked web documents) between two computers.

A [**Protocol**](https://developer.mozilla.org/en-US/docs/Glossary/Protocol) is a set of rules for communication between two computers. HTTP is a textual, stateless protocol.

[**Textual**](https://developer.mozilla.org/en-US/docs/Learn/Common_questions/Web_mechanics/What_is_a_web_server#textual)

All commands are **plain-text** and human-readable.

[**Stateless**](https://developer.mozilla.org/en-US/docs/Learn/Common_questions/Web_mechanics/What_is_a_web_server#stateless)

Neither the server nor the client remember previous communications. For example, relying on HTTP alone, a server can't remember a password you typed or remember your progress on an incomplete transaction. You need an **application server** for tasks like that.

* Usually only *clients* make HTTP requests, and only to *servers*. Servers *respond* to a *client*'s HTTP request. A server can also populate data into a client cache, in advance of it being requested, through a mechanism called [server push](https://en.wikipedia.org/wiki/HTTP/2_Server_Push).
* When requesting a file via HTTP, clients must provide the file's [URL](https://developer.mozilla.org/en-US/docs/Glossary/URL).
* The web server *must answer* every HTTP request, at least with an error message.

On a web server, the HTTP server is responsible for processing and answering incoming requests.

1. Upon receiving a request, an HTTP server checks if the requested URL matches an existing file.
2. If so, the web server sends the file content back to the browser. If not, the server will check if it should generate a file dynamically for the request (see [Static vs. dynamic content](https://developer.mozilla.org/en-US/docs/Learn/Common_questions/Web_mechanics/What_is_a_web_server#static_vs._dynamic_content)).
3. If neither of these options are possible, the web server returns an error message to the browser, most commonly [404 Not Found](https://developer.mozilla.org/en-US/docs/Web/HTTP/Status/404). The 404 error is so common that some web designers devote considerable time and effort to designing 404 error pages.

DOMAIN NAMES:

Alternatively, if you use a system with a built-in shell, type a whois command into it, as shown here for mozilla.org:

BASHCopy to Clipboard

whois mozilla.org

This will output the following:

Domain Name:MOZILLA.ORG

Domain ID: D1409563-LROR

Creation Date: 1998-01-24T05:00:00Z

Updated Date: 2013-12-08T01:16:57Z

Registry Expiry Date: 2015-01-23T05:00:00Z

Sponsoring Registrar:MarkMonitor Inc. (R37-LROR)

Sponsoring Registrar IANA ID: 292

WHOIS Server:

Referral URL:

Domain Status: clientDeleteProhibited

Domain Status: clientTransferProhibited

Domain Status: clientUpdateProhibited

Registrant ID:mmr-33684

Registrant Name:DNS Admin

Registrant Organization:Mozilla Foundation

Registrant Street: 650 Castro St Ste 300

Registrant City:Mountain View

Registrant State/Province:CA

Registrant Postal Code:94041

Registrant Country:US

Registrant Phone:+1.6509030800

SI DESPUES DE UMA CONSULTA whois APARECE ‘NOT FOUND’, EL DOMINIO ESTA AVAILABLE

**Getting a domain name**

The process is quite straightforward:

1. Go to a registrar's website.
2. Usually there is a prominent "Get a domain name" call to action. Click on it.
3. Fill out the form with all required details. Make sure, especially, that you have not misspelled your desired domain name. Once it's paid for, it's too late!
4. The registrar will let you know when the domain name is properly registered. Within a few hours, all DNS servers will have received your DNS information.

***What will your website look like?* discusses the planning and design work you have to do for your website before writing code, including "What information does my website offer?", "What fonts and colors do I want?", and "What does my site do?"**

A website consists of many files: text content, code, stylesheets, media content, and so on. When you're building a website, you need to assemble these files into a sensible structure on your local computer, make sure they can talk to one another, and get all your content looking right before you eventually [upload them to a server](https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/Publishing_your_website).

[**Getting hosting and a domain name**](https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/Publishing_your_website#getting_hosting_and_a_domain_name)

To have more control over content and website appearance, most people choose to buy web hosting and a domain name:

* Web hosting is rented file space on a hosting company's [web server](https://developer.mozilla.org/en-US/docs/Learn/Common_questions/Web_mechanics/What_is_a_web_server). You put website files on the web server. The web server provides website content to website visitors.
* A [domain name](https://developer.mozilla.org/en-US/docs/Learn/Common_questions/Web_mechanics/What_is_a_domain_name) is the unique address where people find your website, such as https://www.mozilla.org or https://www.bbc.co.uk. You can rent your domain name for as many years as you want from a **domain registrar**.

Many professional websites go online this way.

In addition, you will need a File Transfer Protocol (FTP) program (see How much does it cost: software for more details) to actually transfer the website files over to the server. FTP programs vary widely, but generally, you have to connect to your web server using details provided by your hosting company (typically username, password, hostname). Then the program shows you your local files and the web server's files in two windows, and provides a way for you to transfer files back and forth.