

# Requisitos Recursos Tecnológicos de la Asignatura

1. Microsoft Teams – Verificar pertenecer al equipo de la asignatura
2. Grupo de Whatsapp – Revisar comunicados o envío de documentos.
3. Cuenta en GitHub con Sitio web publicado con Portafolio
4. Laptop propia

## ➤ Responsabilidad del Estudiante

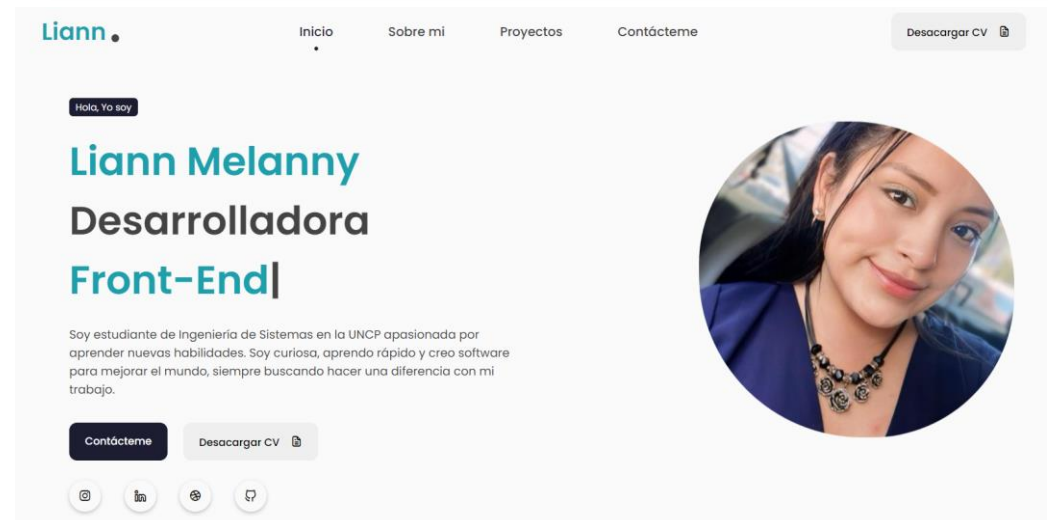
- Es responsabilidad del estudiante de pertenecer y contar con los recursos tecnológicos para los laboratorios y examen.
- Entregar sus laboratorios y examen a tiempo
- Pertenecer a un grupo de trabajo
- Estudiante que para el examen no tiene acceso nota o o

## Delegado de la asignatura.

1. Crear 1 y 2
2. Verificar toma de asistencia
3. Verificar puesta de materiales
4. Abierto aula y laboratorio / llaves de gabinete y encendido de proyector. / Disponibilidad de Internet
5. Atento a los bienes del Laboratorio / Verificar cierre de laboratorio
6. Comunicar sucesos inesperados

# Portafolio Electrónico

- ❖ Sitio web publicado en github -Documentar su *aprendizaje* en Git Hub Site Web (Conforme va aprendiendo va incorporando elementos personalizables)
- ❖ Estructura
  - ★ **Portada** – Título, bienvenida presentación, estudios y experiencia
  - ★ Información **Sobre mi**
  - ★ **Proyectos** Concluidos
  - ★ **Cuaderno**
    - 👉 Semana 1:
      - *Describir temas aprendidos con sus propias palabras* (Definiciones, procedimiento)
      - *Ejercicios de laboratorio - Resultados*
      - *Reflexión ¿Qué aprendió? y ¿Cómo aprendió?*
    - 👉 Semana 2: igual a1
    - 👉 Semana 3: igual a1
  - ★ **Reflexión Final**
    - 👉 Redactar un texto de reflexión metacognitiva
    - 👉 ¿Qué aprendió? ¿Cómo aprendió?
    - 👉 Por semana y Fin de curso.
  - ★ **Bibliografía**: Libros y Urls no videos





C  
INGENIERÍA  
DE  
SISTEMAS

## Fundamentos de la Tecnología Web

**IS093 – DESARROLLO DE APLICACIONES WEB**

**JAIME SUASNABAR TERREL**







- ❖ El evento contaba con 26 charlas en inglés y 9 en español.
- ❖ Luis Merino sobre inteligencia artificial (IA).
- ❖ Daniel Ramos implementar una funcionalidad sobre un pet project usando TDD.

- ❖ Fran Palacios conferencia Web3. Montó un ejemplo de cómo integrar Ethereum y smart contracts con React para tener un backend descentralizado.
- ❖ Adrián Ferrera charla "101. How to build Bad Software",

## Artículos Interesantes



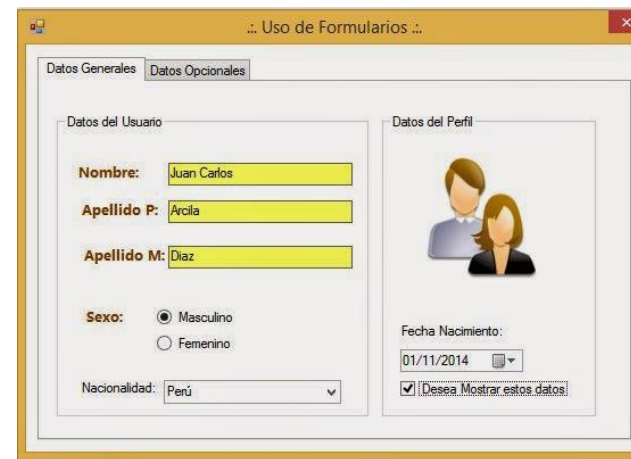
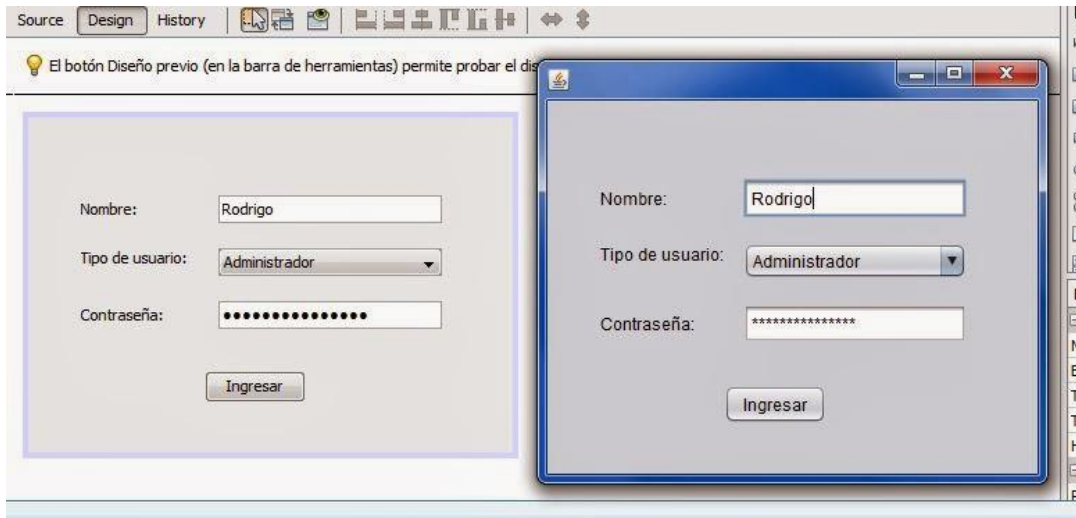
- ❖ Lectura de los siguientes artículos
- ❖ [Juegos](#)
- ❖ [Aprendizaje de Programación](#)
- ❖ [Bolsa de Valores](#)

# Temas del Curso

- ❖ Estrategia de aprendizaje
  - ★ Portafolio Electrónico - Git Hub Sitio Web
- ❖ Tecnologías de desarrollo de software
- ❖ Estándares Internacionales para el Desarrollo Web
- ❖ ¿Cómo funciona la web? / DNS
- ❖ Contexto de Aplicaciones de Internet
- ❖ FrontEnd
  - ★ Librerías y herramientas del desarrollo frontend
- ❖ BackEnd
  - ★ Librerías y herramientas del desarrollo backend

# Tecnologías de Desarrollo de Software

- ❖ La transformación en el tratamiento de la información y la evolución digital de los procesos empresariales, ha generado tecnologías para desarrollar software, orientadas a satisfacer las exigencias de los mercados.
- ❖ Actualmente tenemos dos tecnologías de desarrollo de software:
  - ★ La Tecnología Web
  - ★ La Tecnología de Software Nativo

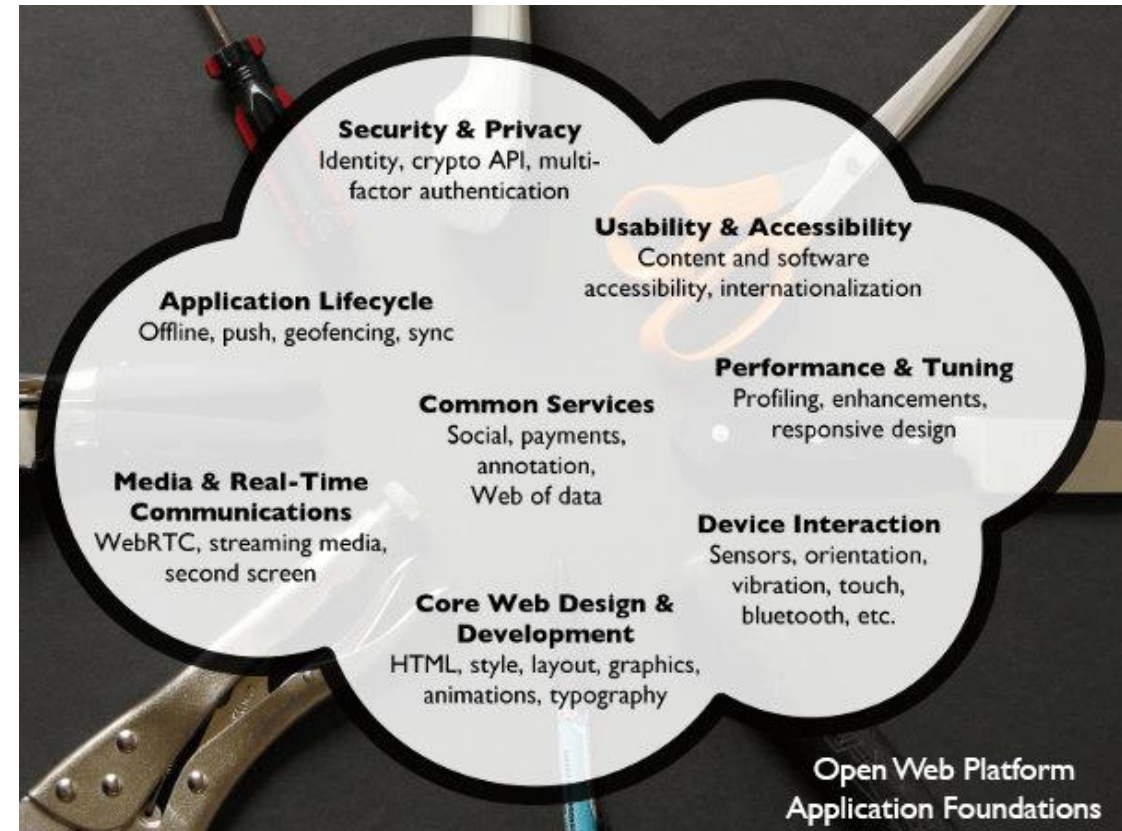




## ¿Qué es?

**Open Web Platform** es una colección de tecnologías abiertas y estándares desarrollados por organismos como W3C, Unicode Consortium, Internet Engineering Task Force, y ECMA International. Algunas de las tecnologías que cubre son HTML5, CSS, SVG, MathML, WAI-ARIA, ECMAScript, WebGL, etc..

- PRINCIPIOS:
- **Evita ser controlado** por ninguna empresa.
- **Aporta transparencia**, haciendo visibles todos los niveles desde el origen de documento hasta las URLs y las capas HTTP.
- **Capacidad de integración.** Fuentes externas.
- **Documentación y especificaciones abiertas.**
- **Libertad de Uso.**
- **Cadena de favores.** Ser integrante de la Open Web significa compartir lo aprendido con blogs, conferencias o el uso de tecnologías abiertas.





# Estándares Internacionales para el Desarrollo Web

## ❖ Básicas

- ❖ **HTTP/HTTPS** Hypertext Transfer Protocol (HTTP) o Protocolo de Transferencia de Hipertexto (HTTP) es un protocolo de la capa de aplicación para la transmisión de documentos hipermedia, como HTML. <https://httpwg.org/specs/>
- ❖ **HTML** HyperText Markup Language (HTML) es el lenguaje utilizado para describir y definir el contenido de una página web en un formato bien estructurado. <https://html.spec.whatwg.org/>
- ❖ **CSS** Cascading Style Sheets (CSS) u Hojas de estilo en cascada se utilizan para describir el aspecto o presentación del contenido en una página web. <https://www.w3.org/Style/CSS/Overview.en.html>
- ❖ **XML** o Lenguaje de Marcado Extensible es un estándar que se utiliza para almacenar e intercambiar información estructurada entre diferentes plataformas. <https://www.w3.org/TR/xml/>
- ❖ **JavaScript** JavaScript es el lenguaje de programación que se ejecuta en el navegador y que se utiliza para crear sitios web interactivos y aplicaciones avanzadas para que sean ejecutadas de modo seguro en el navegador. <https://www.w3.org/TR/DOM-Level-2-HTML/ecma-script-binding.html>

# Estándares Internacionales para el Desarrollo Web

## ❖ Complementarias

❖ **Web APIs** Web Application Programming Interfaces (Web APIs) Son utilizadas para desarrollar una gran variedad de tareas, como manipular el DOM, reproducir audio o video o generar gráficos 3D.

## ❖ Gráficos

★ **SVG** Scalable Vector Graphics o Gráficos vectoriales escalables permite describir las imágenes como conjuntos de vectores (líneas) y formas con el fin de permitir ajustarse a escala sin problemas independientemente del tamaño en el que están trazadas.

★ <https://www.youtube.com/watch?v=ZJSCI6XEdP8>

★ **WebGL** WebGL es una API Javascript que permite dibujar gráficos 3D o 2D utilizando elementos HTML5 <canvas>.

★ <https://www.youtube.com/watch?v=3GqUM4mEYKA&t=11s>

## ❖ Otros

★ **MathML** Mathematical Markup Language o Lenguaje de Marcado Matemático permite mostrar complicadas ecuaciones y sintaxis matemáticas. <https://www.youtube.com/watch?v=OKZ0hf3VUI4>

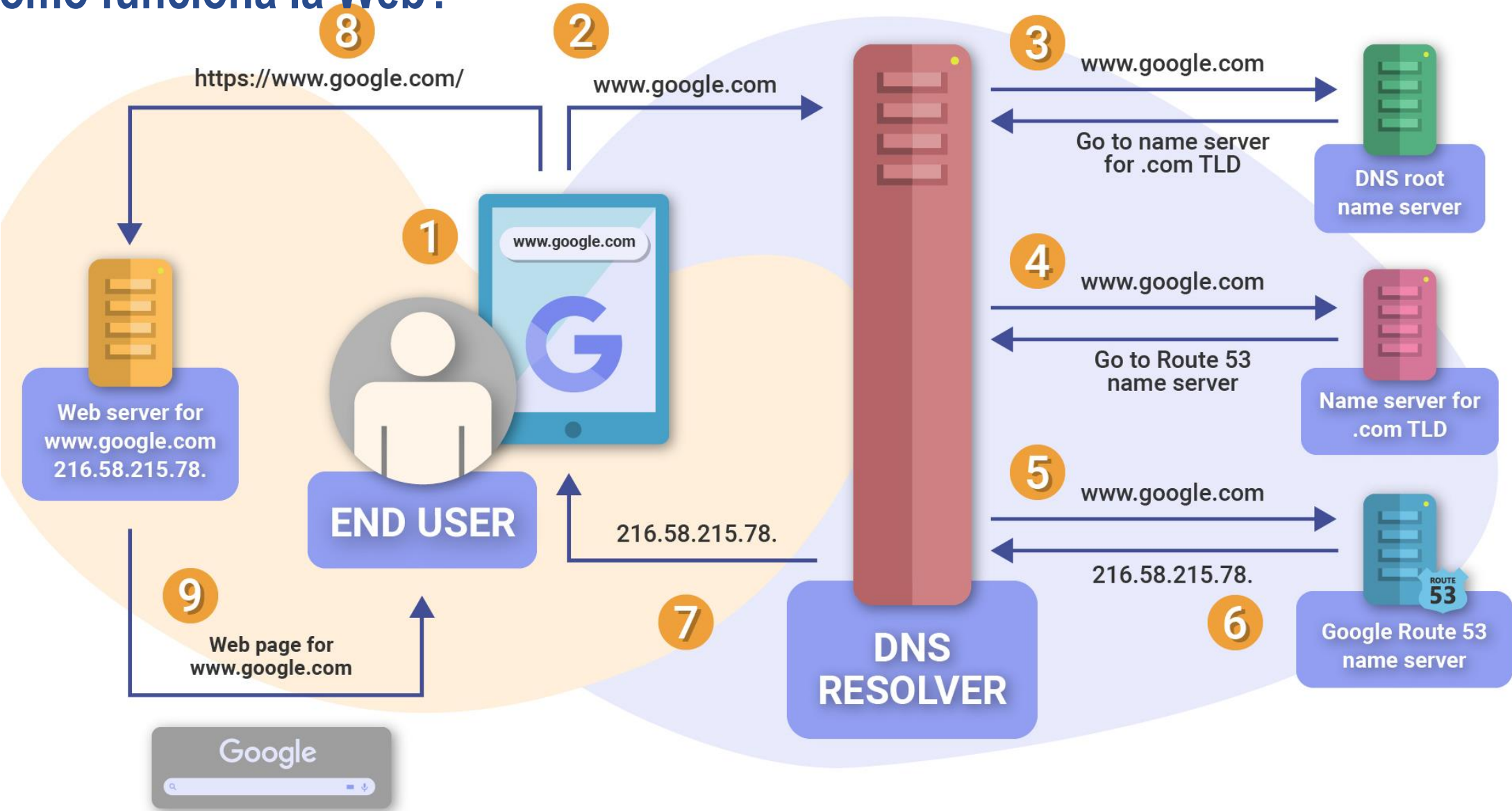
★ **Aria** Accessible Rich Internet Applications. Personas con discapacidad. <https://www.w3.org/TR/html-aria/>

## DNS - Sistema de Nombres de Dominio

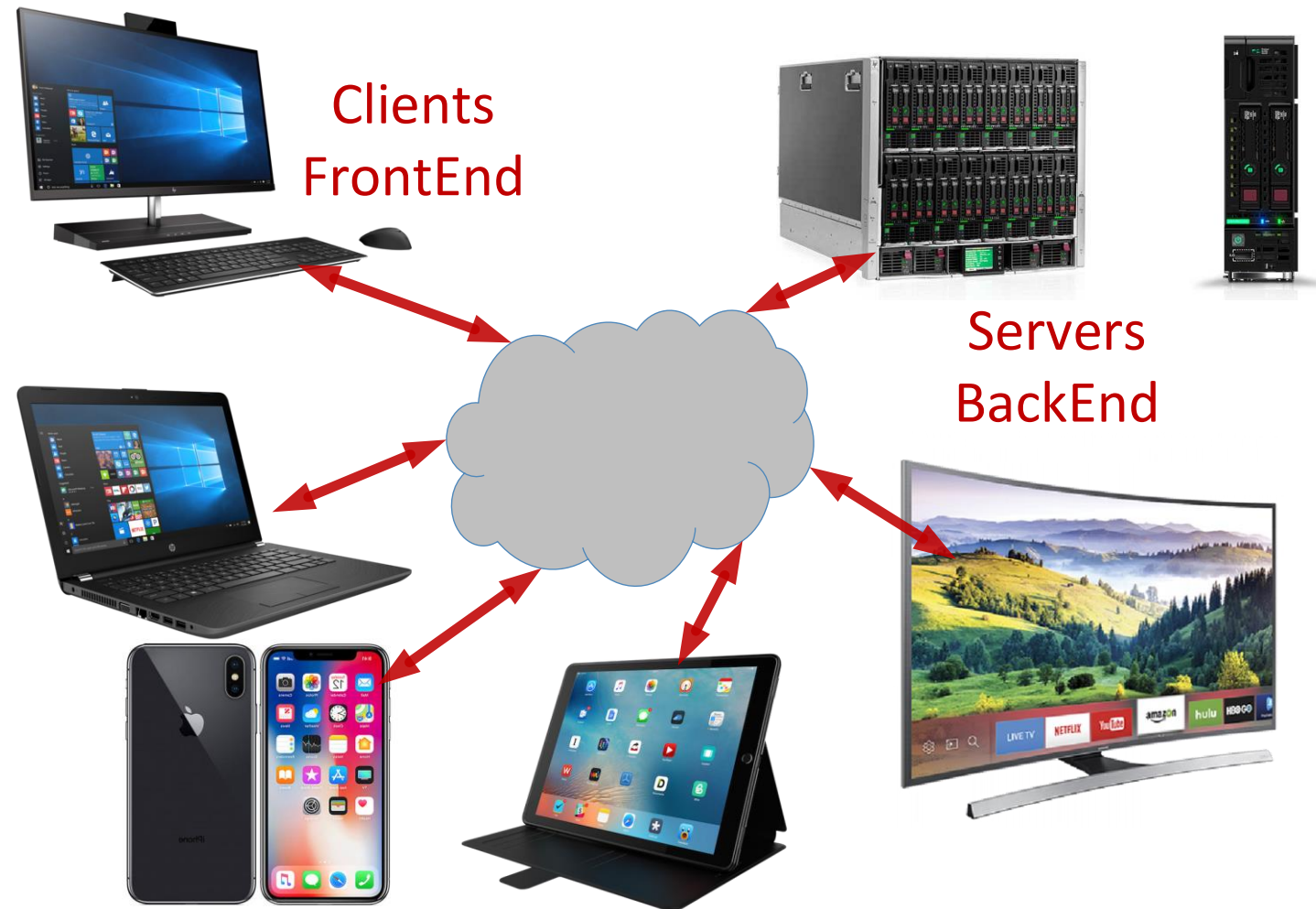
- ❖ DNS es el directorio telefónico de Internet. Las personas acceden a la información en línea a través de nombres de dominio como [www.uncp.edu.pe](http://www.uncp.edu.pe). Los navegadores web interactúan mediante direcciones de Protocolo de Internet (IP). El DNS traduce los nombres de dominio a direcciones IP para que los navegadores puedan cargar los recursos de Internet.



# ¿Cómo funciona la Web?



# Contexto de Aplicaciones para Internet



# Desarrollo de aplicaciones web

## ❖ Desarrollo frontend

- ★ El desarrollo **frontend** es la parte de la programación web que se enfoca en la interfaz y experiencia del usuario. En otras palabras, es todo lo que los usuarios ven y con lo que interactúan en una página web o aplicación.
- ★ HTML, CSS y Javascript

## ❖ Desarrollo backend

- ★ El desarrollo backend es la parte de la programación web que se encarga de la lógica, la base de datos y el funcionamiento interno de una aplicación o sitio web. Es, por así decirlo, el motor que hace posible que el frontend funcione correctamente.
- ★ Los desarrolladores backend trabajan con:
  - 👉 - Lenguajes de programación como Python, Java, PHP, Node.js
  - 👉 - Bases de datos como MySQL, PostgreSQL, MongoDB
  - 👉 - Servidores y APIs que permiten la comunicación entre el frontend





# Lado Cliente - Frontend



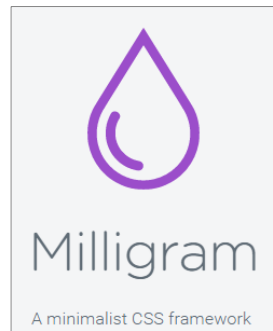
# Lenguajes y herramientas del Desarrollo del Frontend

## Desarrollo FrontEnd



# Librerías y Herramientas del Desarrollo del Frontend - Framework CSS

## Framework CSS



## Preprocesadores





# Librerías y Herramientas del Desarrollo del Frontend - Herramientas para JS

## Herramientas JS

UI Libraries / Frameworks



Language Spec



Build Tools



Package Managers



# Librerías del Desarrollo del Frontend- Frameworks JS

## Framework JS



Meteor JS



Ember JS



React JS



Vue JS



Polymer JS



JavaScript Frameworks



Angular JS

# Sistema de Control de Versiones

VCS (Version Control System)



# Desarrollo del Lado Servidor- Backend

## Desarrollo BackEnd

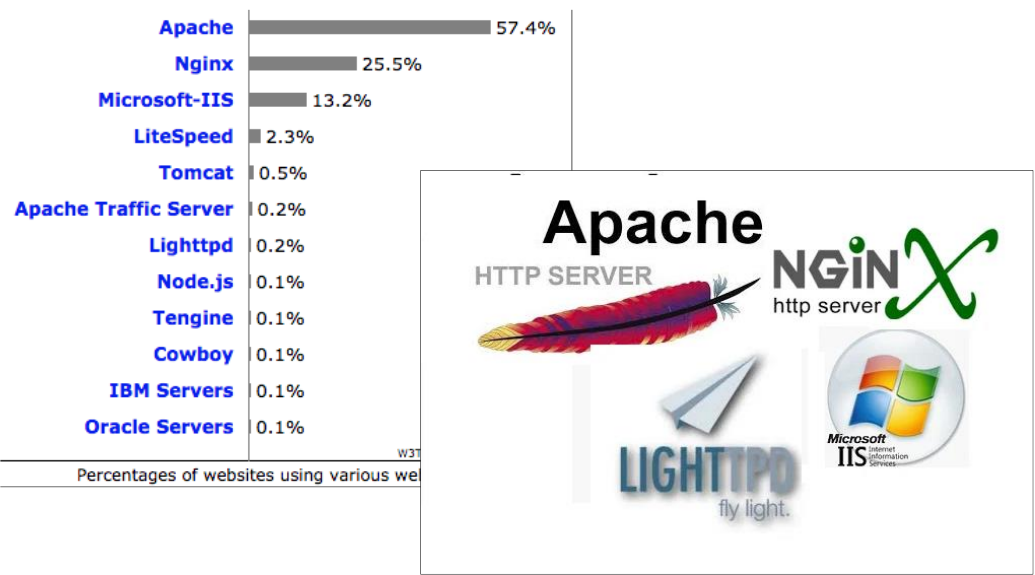
- Servers Group
- Hosting Service





# Servidores Web

## Web Servers



# Lenguajes del Lado Backed

## Language Backend



# Lenguajes del Lado Backed

Language Backend

Microsoft  
**ASP.net**

Microsoft®  
**.NET**  
Framework



# Gestores de Base de Datos

## Gestores de BD





# ¿Qué es el diseño UX/UI (UX and UI Design)?

## ❖ User interface UI – Mockups

- ★ Figma / Adobe XD
- ★ Gliffy / Balsamiq /Gravity

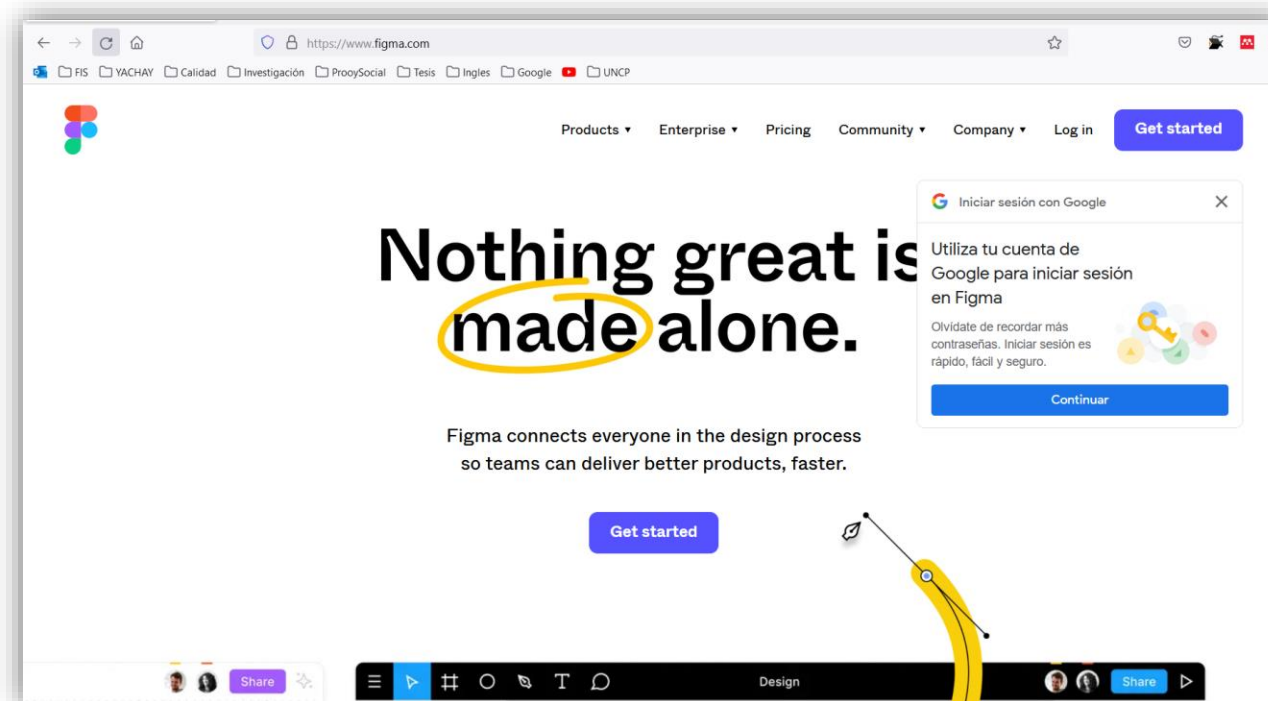
## ❖ User experience o UX se define como la experiencia que el público tiene al entrar en contacto con tus productos, servicios o cualquier ámbito de tu empresa.

## ❖ Su objetivo es entregar a los usuarios soluciones agradables, intuitivas, atractivas y eficientes; y que su experiencia con tu marca sea muy satisfactoria.



# Figma

- ❖ **Figma se trata de una herramienta de diseño enfocado en sitios web y aplicaciones.** A diferencia de otras herramientas de prototipado, Figma **se aloja en la web y permite la colaboración en vivo**: tienes la posibilidad de compartir el proyecto con los miembros de tu equipo y editarlo en la misma mesa de trabajo.
- ❖ <https://www.youtube.com/watch?v=uIDUYPHg3XA>



# Práctica de Laboratorio 01: Manejo de Visual Studio Code

## ❖ Ejercicio 01

- ★ Reconocer y utilizar la herramienta de desarrollo integrado Visual Studio Code, áreas de la Pantalla Principal, y Aplicaciones de la Barra de Menus.

## ❖ Ejercicio 02

- ★ Activación de la Paleta de comandos y ejemplos prácticos del uso de la paleta: Ejemplo de uso de GitHub Codespaces.

## ❖ Ejercicio 03

- ★ Uso de la Terminal de Visual Studio Code ejemplo práctico de git: git init, git add, git commit, git checkout, git push. Ramas y merge en github.

## ❖ Ejercicio 04

- ★ Uso de disparadores Emmet para generar código rápido en Visual Studio Code. Creación y Uso de Fragmentos de Código en VSC.

👉 <https://docs.emmet.io/cheat-sheet/>

## ❖ Ejercicio 05

- ★ Uso de diferentes Keyboard shortcuts en VSC

