GITHUB

Objetivos de la Clase:

1. Familiarizarse con GitHub y la gestión de proyectos colaborativos.
2. Aprender a subir proyectos a GitHub y utilizar **.gitignore** correctamente.
3. Comprender el flujo de trabajo con ramas y fusiones (branching, merging, pull requests).
4. Publicar un proyecto estático usando GitHub Pages.

GRUPO iDT

Teoría

¿Qué es GitHub?

GitHub es una plataforma para almacenar, gestionar y colaborar en proyectos de software usando Git. Permite trabajar en equipo, seguir cambios y desplegar proyectos.

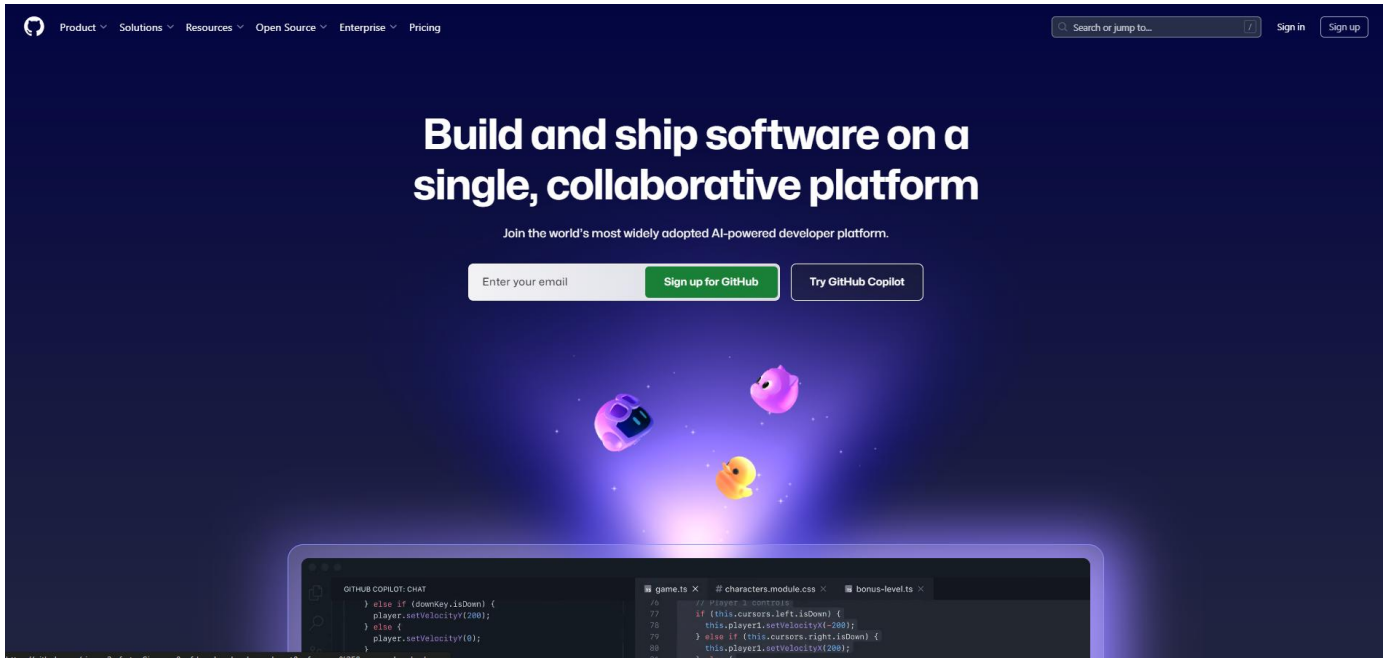
Teoría

Conceptos clave

Término	Descripción
Repositorio	Espacio donde se almacena un proyecto.
Commit	Registro de cambios en el código.
Branch (Rama)	Línea de desarrollo separada de la principal.
Pull Request	Propuesta de cambios para fusionar código.
Merge	Integración de ramas en un solo código.

Creación de una cuenta en GitHub

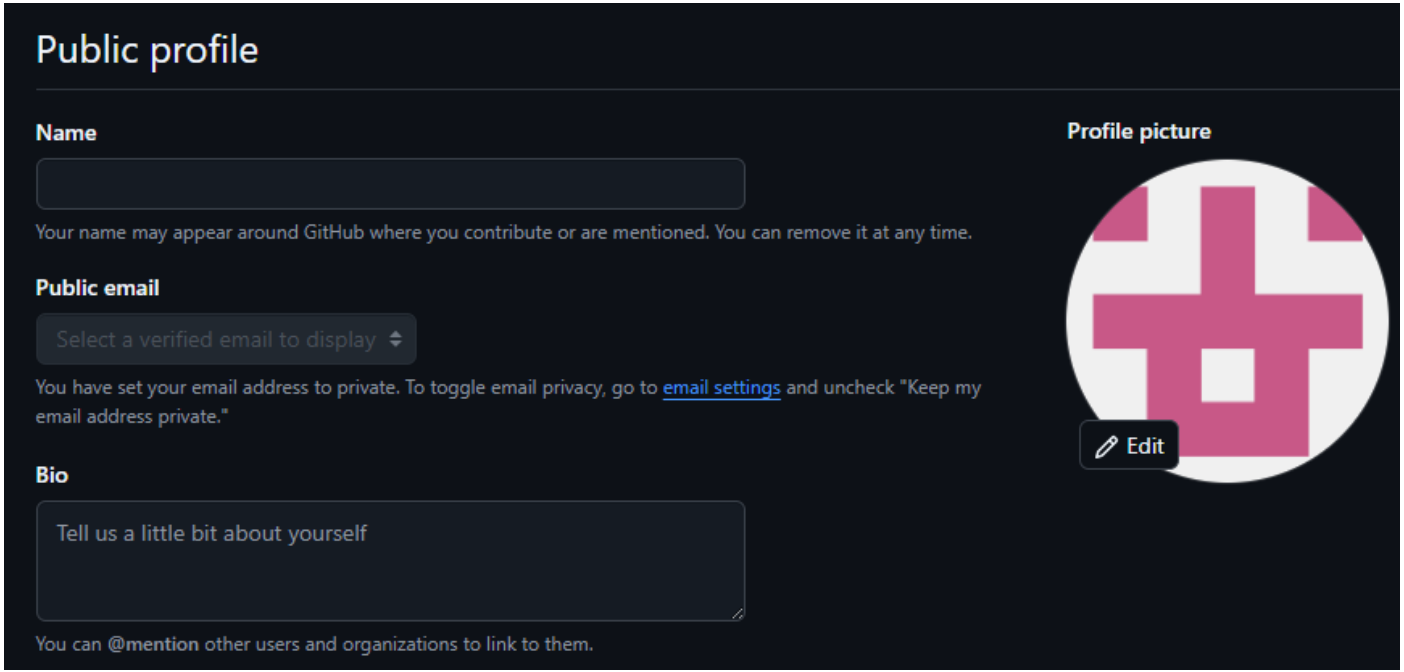
Ir a github.com y registrarse.



GRUPO iDT

Creación de una cuenta en GitHub

Configurar perfil y activar autenticación en dos pasos (opcional).



GRUPO iDT

¿Cómo ayuda GitHub en proyectos colaborativos?

Debido a que es una plataforma esencial para proyectos colaborativos, especialmente en el desarrollo de software. Te explico brevemente cómo nos puede ayudar:

- **Control de versiones:** Al usar git, permite rastrear y gestionar los cambios en código.

- **Colaboración y trabajo en equipo:** Los miembros del equipo pueden trabajar en manera remota y luego fusionar sus cambios en el mismo proyecto.

- **Seguimiento de problemas y tareas:** Ofrece forma de crear y gestionar tareas, organizando las incidencias y comunicación del equipo.

- **Revisión y mejora contínua:** Las pulls requests facilitan revisión por parte de todos los miembros, promoviendo la colaboración y aprendizaje mutuo.

¿Cómo ayuda GitHub en proyectos colaborativos?

- **Automatización de tareas:** Github Actions permite automatizar tareas como pruebas, compilación y despliegues, de esta forma ahorramos tiempo, mejoramos eficiencia y reducimos los errores.

- **Documentación centralizada:** Almacena y gestiona la documentación junto al código, facilitando la información y garantiza que la misma esté actualizada.

- **Comunidad y colaboración abierta:** Alberga una gran cantidad de desarrolladores, los proyectos de código abierto permiten la colaboración con personas de todo el mundo, fomentándo la innovación y el intercambio de conocimientos.

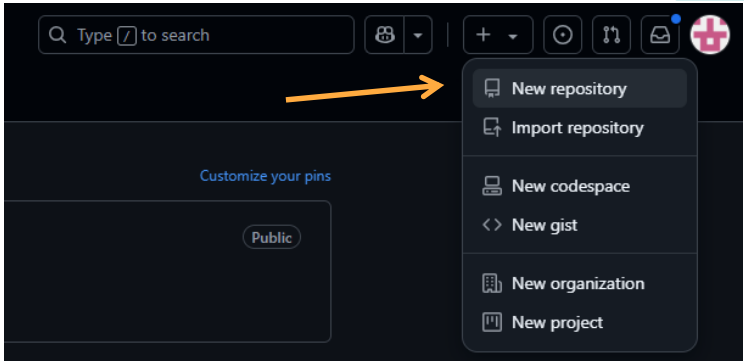
Subida y gestión de proyectos con GIT y .gitignore

GRUPO IDTi

Flujo básico

Creación de un repositorio en GitHub

1) Hacer clic en New Repository.
2) Elegir nombre y visibilidad (público o privado).
3) Clonar el repositorio en local:




```
git clone https://github.com/usuario/nombre-repositorio.git
```

GRUPO iDT

Flujo básico

Subir archivos a GitHub

```
git add .
git commit -m "Primer commit"
git push origin main
```

Flujo básico

Uso de .gitignore, es un archivo que evita subir archivos no deseados al repositorio.

GRUPO iDT

¿Por qué es importante el .gitignore?

El archivo .gitignore es una herramienta esencial para mantener los repositorios limpios, seguros y eficientes. Su correcto uso es fundamental para un flujo de trabajo de Git óptimo, especialmente en proyectos colaborativos.

Ejercicio

Creación de un repositorio en GitHub

1) Crear un repositorio nuevo en GitHub.
2) Subir archivos desde la terminal con git push.
3) Agregar un archivo en el .gitignore, y hacer otro commit.
4) Verificar archivos en el repositorio remoto y contrastar con el local

Ramas, Fusiones y Pull Requests

GRUPO IDT

¿Qué son las ramas?

Las ramas permiten trabajar en nuevas características sin afectar el código principal.

Comandos Básicos de Branching & Merging

Crear una nueva rama y cambiar a ella

```
git checkout -b nueva-rama
```

Hacer cambios y subirlos

```
git add .
git commit -m "Nueva funcionalidad"
git push origin nueva-rama
```

Fusionar la rama con “main”

```
git checkout main
git merge nueva-rama
```

Eliminar una rama después de fusionar

```
git branch -d nueva-rama
```

Pull Requests (PRs) en GitHub

Pasos a seguir

Creación de un repositorio en GitHub

1) Subir cambios a una rama remota.
2) Ir a GitHub → Pull Requests → New Pull Request.
3) Comparar ramas y solicitar revisión.
4) Mergear si no hay conflictos.

Despliegue básico con GitHub Pages

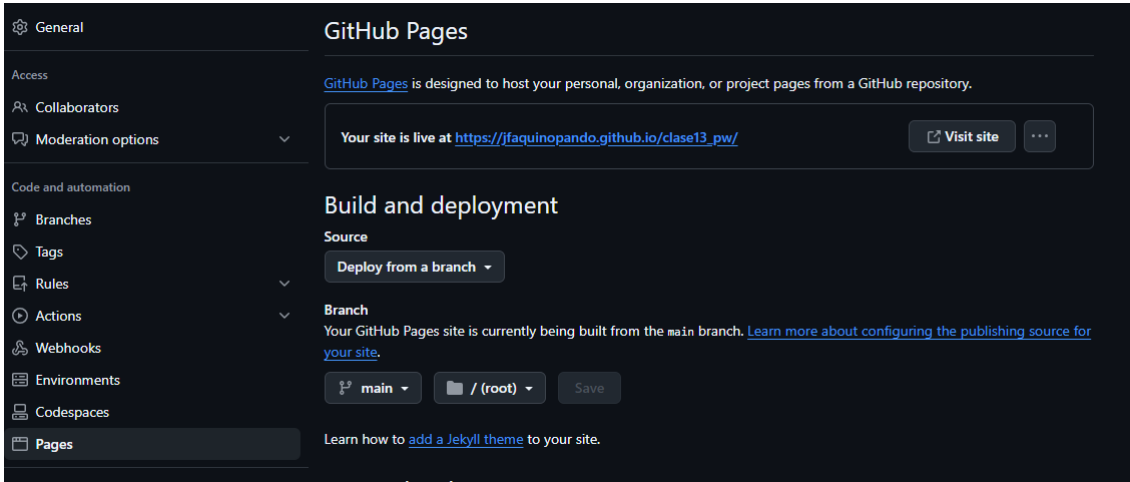
GRUPO IDT

¿Qué es GitHub Pages?

GitHub Pages permite publicar sitios estáticos directamente desde un repositorio.

¿Cómo activar GitHub Pages?

1) Ir a Settings → Pages en GitHub.
2) Seleccionar la rama de despliegue (main).
3) GitHub generará una URL pública:
   https://usuario.github.io/nombre-repositorio



Tipos de clases más usadas en Tailwind

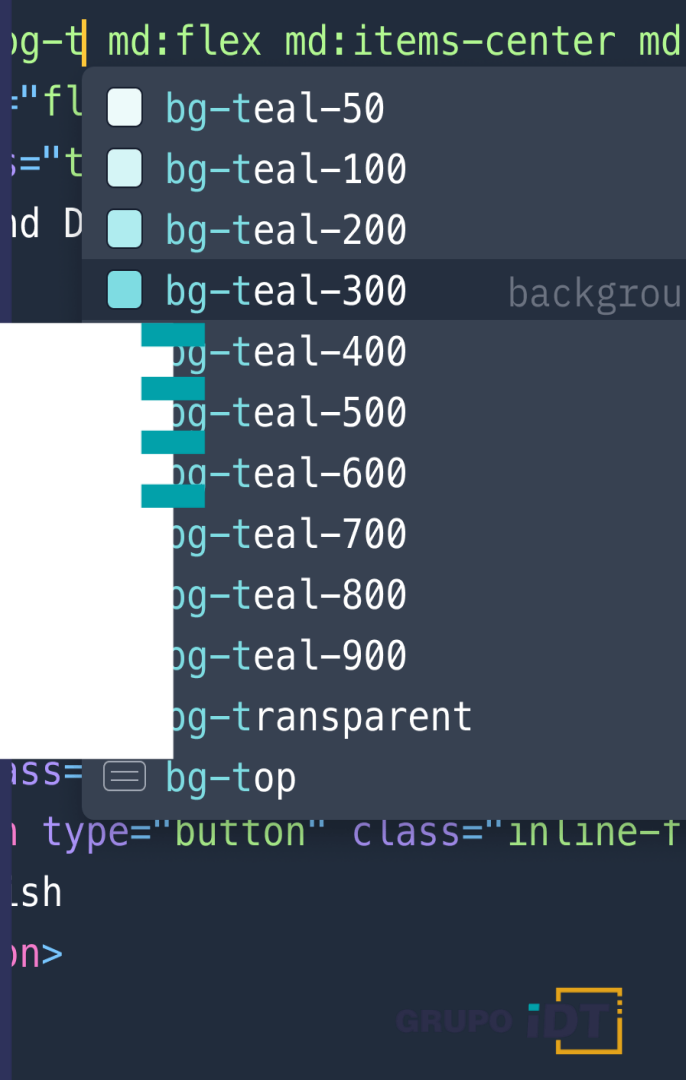
- Colores (bg, text, border)

archivo.html

```
<p class="text-red-500 bg-gray-100 border border-gray-300 p-2">Texto en rojo con fondo gris</p>
```

Texto en rojo con fondo gris

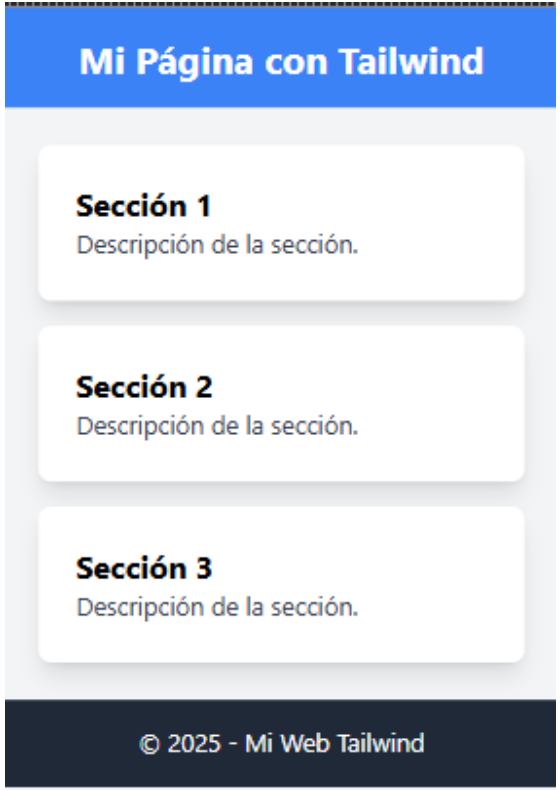
Veamos como aplicar a una web



```
<body class="bg-gray-100">
    <header class="bg-blue-500 text-white p-4 text-center text-2xl font-bold">
        Mi Página con Tailwind
    </header>
    <section class="container mx-auto p-6">
        <div class="grid grid-cols-1 md:grid-cols-3 gap-4">
            <div class="bg-white p-6 rounded-lg shadow-lg">
                <h2 class="text-xl font-bold">Sección 1</h2>
                <p class="text-gray-700">Descripción de la sección.</p>
            </div>
            <div class="bg-white p-6 rounded-lg shadow-lg">
                <h2 class="text-xl font-bold">Sección 2</h2>
                <p class="text-gray-700">Descripción de la sección.</p>
            </div>
            <div class="bg-white p-6 rounded-lg shadow-lg">
                <h2 class="text-xl font-bold">Sección 3</h2>
                <p class="text-gray-700">Descripción de la sección.</p>
            </div>
        </div>
    </section>
    <footer class="bg-gray-800 text-white text-center p-4">
        © 2025 - Mi Web Tailwind
    </footer>
</body>
```

En vista Desktop

En vista Mobile



GRUPO iDT

Explicación del Código

- Se utiliza container mx-auto para centrar el contenido.
- grid grid-cols-1 md:grid-cols-3 adapta el diseño a una columna en móvil y tres en escritorio.
- shadow-lg agrega sombra a las tarjetas para mejorar el diseño.

Animaciones y Hover Effects

- Aplicamos efecto hover a un botón

estilos.css

```css
.card:hover {
    transform: scale(1.05);
}
```

index.html

```html
<button class="card bg-green-500 text-white px-4 py-2
rounded-md hover:bg-green-700 transition">
    Comprar
</button>
```

GRUPO iDT