

SALESIANOS
INSTITUTO TECNICO RICALDONE



TEMA: “Proyecto, página web”

ESPECIALIDAD: 1° Desarrollo de Software.

NIVEL: PRIMER AÑO DE BACHILLERATO.

GRUPO: 2

SECCION: A

INTEGRANTES:

- Yacira Elisheba, Gutiérrez Castillo (20190142)
- López Huevo, Víctor René (20190189)
- Castillo Oliva, Susan Abigail (20190200)
- Rodrigo José, Díaz Hernández (20190380)
- Hernández Amaya, Rebeca Marcela (20200248)

Docente Guía: Daniel Wilfredo Granados Hernández.

San Salvador, marzo de 2022

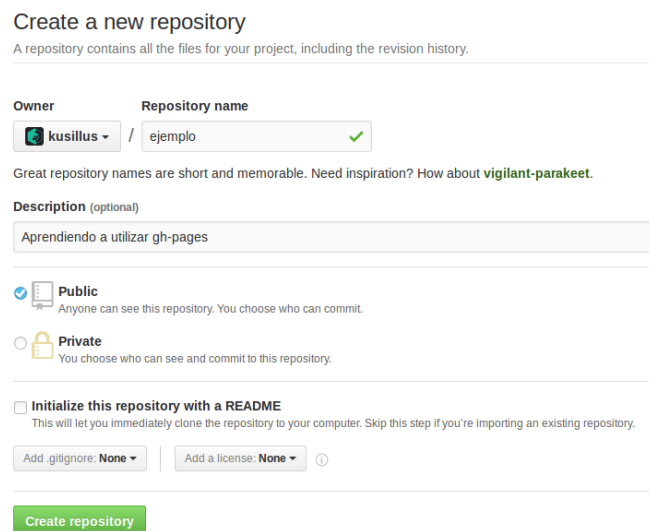
¿En qué consiste Travigo?

Travigo consiste en un sitio web informativo sobre sitios turísticos que hay en el país, con diferentes secciones que pueden ayudar a interactuar con el público, a través de dicha información que se muestran diferentes apartados, cada uno puede ayudar al público a conocer y descubrir nuevas opciones de viaje en los diferentes departamentos de El Salvador.

En el sitio web se publica información sobre los departamentos del país y sus respectivos lugares populares o conocidos del país, nos brinda información sobre actividades a realizarse en diferentes festividades del año, así como servicios adicionales, promociones, tarifas, etc. Y un último apartado que muestra información sobre nosotros.

¿Cómo subimos el documento a un servidor local?

Primero vamos a github.com y creamos un nuevo repositorio para nuestro código, le pondremos un nombre a nuestro repo, se recomienda que el nombre del repositorio sea igual al nombre de proyecto en nuestro ordenador.



The screenshot shows the 'Create a new repository' page on GitHub. At the top, it says 'Create a new repository' and 'A repository contains all the files for your project, including the revision history.' Below this, there are two input fields: 'Owner' with a dropdown menu showing 'kusillus' and 'Repository name' with a text input showing 'ejemplo'. A green checkmark is next to the repository name. Below these fields, there is a note: 'Great repository names are short and memorable. Need inspiration? How about [vigilant-parakeet](#).' Then, there is a 'Description (optional)' section with a text input containing 'Aprendiendo a utilizar gh-pages'. Below the description, there are two radio button options: 'Public' (selected) and 'Private'. The 'Public' option has a description: 'Anyone can see this repository. You choose who can commit.' The 'Private' option has a description: 'You choose who can see and commit to this repository.' Below the radio buttons, there is a checkbox labeled 'Initialize this repository with a README' which is unchecked. Below the checkbox, there is a note: 'This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.' At the bottom, there are two dropdown menus: 'Add .gitignore: None' and 'Add a license: None'. At the very bottom, there is a green button labeled 'Create repository'.

Una vez creado el repositorio, copiaremos la ruta que nos mostrará, aquí es donde subimos nuestro proyecto almacenado en el ordenador utilizando Git.



Mediante el terminal nos ubicamos dentro de la carpeta “index.html”, la cual contiene los archivos de la web que subiremos.

Inicializamos nuestro repositorio local y hacemos nuestro primer commit con los siguientes comandos.

```
git init
git add .
git commit -m “Mi primer commit”
```

Y los cambios quedarán registrados, ahora podemos subir estos cambios a nuestro repositorio en GitHub, para eso escribimos:

```
git remote add origin https://github.com/kusillus/ejemplo.git
git push origin master
```

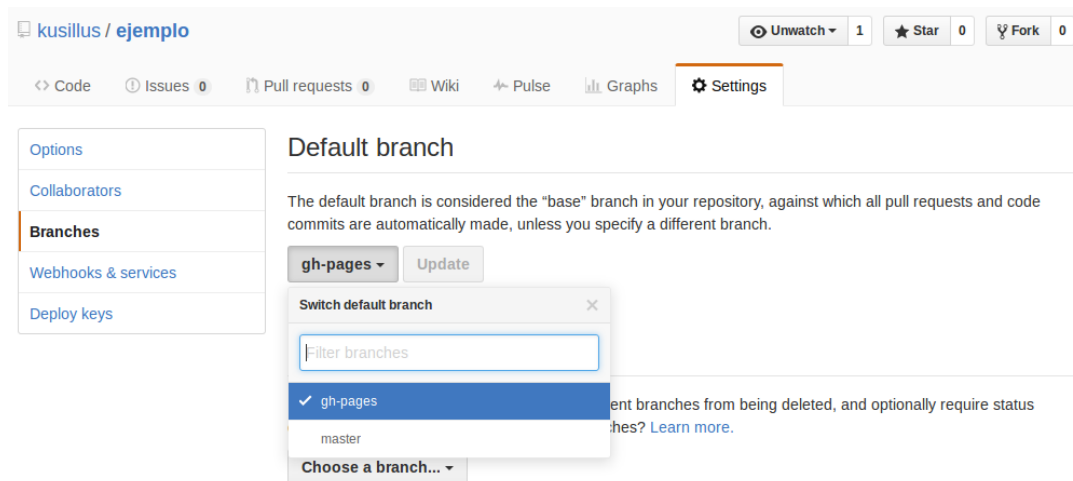
Nos pedirá nuestro usuario y contraseña y si el proceso se realizó correctamente debería mostrarte algo así:

```
kusillus@kusillus-Dev:~/Documents/ejemplo$ git push origin master
Username for 'https://github.com': kusillus
Password for 'https://kusillus@github.com':
Counting objects: 5, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (5/5), 880 bytes | 0 bytes/s, done.
Total 5 (delta 0), reused 0 (delta 0)
To https://github.com/kusillus/ejemplo.git
 * [new branch]      master -> master
kusillus@kusillus-Dev:~/Documents/ejemplo$ _
```

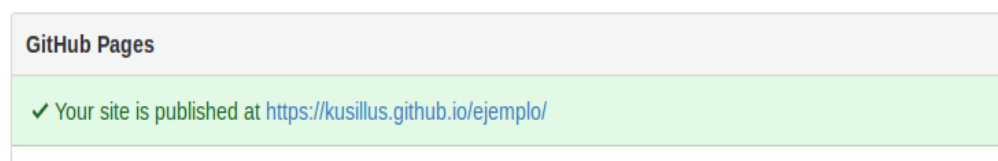
Ahora los archivos del proyecto aparecerán en nuestro repositorio en GitHub.

En la página de nuestro repositorio vamos a dar clic en el botón Branch: master se nos desplegará un cuadro y dentro de la caja de texto escribimos gh-pages, de esta manera crearemos una nueva rama.

A continuación accederemos a la opción de settings y luego a branches. Cambiaremos la rama por defecto “master” por “gh-pages”, luego damos clic en el botón “update” y aceptamos el mensaje que nos muestra.



Regresamos a options y bajamos hasta el cuadro llamado “GitHub Pages”, ahí abrimos el enlace que se nos muestra.



Y así tenemos nuestra página publicada en GitHub.