

INFORMACION REPOSITORIO GITHUB

Versión	Fecha	Revisado y Aprobado por	Descripción del Cambio
1.0.1	22/10/2022	Carlos Vasquez	Versión inicial

Alcance

- Manejo del repositorio para mayor control en el versionamiento en el proyecto asegurando un trabajo en equipo mas ordenado

Proyecto a realizar: Anexos proyectos N° 9 Retos científicos

TEAM

Cristian Esteban Álvarez Peña - cristian-012@hotmail.com

Carlos William Vasquez Vargas - willvasquez98@hotmail.com

Diana Catalina Vasquez Rendon - catonga9823@gmail.com

Yoraima Rincon Contreras - Yori.rincon@gmail.com

cristian-012@hotmail.com willvasquez98@hotmail.com catonga9823@gmail.com yor.rincon@gmail.com

Roles:

Gestor del proyecto - Catalina Vasquez

Gestor base de datos - Yoraima Rincon

Desarrollador frontend - Cristian Alvarez

Desarrollador backend -

Tester - William Vasquez

CONFIGURACION AMBIENTE DE DESARROLLO CON VISUAL STUDIOS CODE

1. Abrir carpeta en visual Code

1. File → Open folder → "Seleccionar carpeta (proyecto) en el repositorio clonado"

1. Deben aparecer las carpetas: public - src, y los archivos: .env - gulpfile.js, jsconfig.json - package.json

2. Terminal → New terminal:

1. Comando a ejecutar

1. npm install --force

2. npm audit fix --force (opcional)

3. npm start

En caso de aparecer el error opensslErrorStack: ['error:03000086:digital envelope routines::initialization error'], elimine la carpeta node_module que se encuentra dentro de la carpeta proyecto, desde tu visor de archivos "debe estar cerrado el VisualCode para no presentar fallos en la eliminacion".

Despues vuelves a ejecutar los comandos desde el terminal del visualCode: npm install --force y npm start

Dependencias extras a instalar

1. Ejecutar en terminal de VS Code

```
1. yarn add react-select
```

```
2. yarn add semantic-ui-react semantic-ui-css
```

- `npm install express`
- `npm install nodemon`
- `npm install mongoose`
- `npm install uniqid --force`
- `npm install axios --force`
- `npm install body-parser`
- `npm install/sweetalert2 --force`

Iniciar servidor

Se deben ubicar en la ruta "\\MisionTIC-Grupo56-Ciclo4\\Desarrollo" en su VS-Code y en el terminal ingresan este comando antes de arrancar el front

- `node server.js`

Para validar la conexion al back visiten la url

<http://localhost:5000/>

RESOURCES

Plantilla: <https://demos.creative-tim.com/now-ui-kit-react/#/index>

Archivo Plantilla: <https://drive.google.com/file/d/1n5Xwxro9n5vwDZjHG1CdonNgtq0y9xL0/view?usp=sharing>

Git: <https://github.com/ragnarok0498/MisionTIC-Grupo56-Ciclo4>

Trello: [Ciclo 4 Historias de usuario](#) | [Trello](#)

MongoDB: [Cloud: MongoDB Cloud](#)

Drive: [Grupo56 - Ciclo 4 MisionTIC UTP - Google Drive](#)

Compilar proyecto

En el visual Studios Code, Abriremos una nueva carpeta "seleccionamos la carpeta (proyecto) que esta en nuestro repositorio", despues en visual abrimos un terminal: en la barra superior de opciones buscamos terminal → nuevo terminar o (Ctrl + shift + ñ)

```
npm install → si sale algun error instamos con el siguiente comando  
npm install --force  
npm start → para ejecutar la aplicacion
```

Comandos

```
git init  
git commit -m "Nombre del commit (que identifique los cambios agg)"  
git branch -M main  
git remote add origin https://github.com/ragnarok0498/MisionTIC-UTP-Grupo-56.git  
git push
```

Antes de empezar a codificar valida que no hayan cambios en el repositorio remoto para evitar conflitos este comando actualizara los cambios remotos en la rama ubicada con tu repositorio local

```
git pull
```

Si no se hay cambios en el remoto valida que no hayas cambios locales para evitar errores:

```
git checkout .
```

para enviar tus cambios:

```
git add .  
git commit -m "titulo del commit"  
git push
```

NOTA: Para clonar el repositorio solo digital

```
git clone "url HTTPS repositorio"
```

Ramas [Para tener mas control en los cambios realizados]

Se dejara por defecto una rama por default con el codigo principal que usemos:

- **Main.**
- **desarrollo**

1. Para cambiar de ramas, con el comando

```
git checkout "nombre de la rama"
```

2. git checkout "nombre de la rama"

NOTA: En caso de presentarse un error al cambiar de rama, valide que en la rama donde actualmente se encuentra no tenga cambios que usted haya realizado en este caso ignorelos con el comando git checkout .

Si deseas cambiar de rama o actualizar el repositorio y tienes cambios no commitiados usa el stash para almacenamiento temporal, actualizar el repositorio y/o cambiar de rama realizando los siguientes pasos

```
git status → Validamos los cambios realizados
git add . → Agregamos todos los cambios
git stash → Guardamos los cambios en un espacio temporal (stash)
git status → Validamos que no quede ningun cambio
git pull / git checkout "nombre rama" → actualizamos o cambiamos de rama
git stash pop → Agregamos lo que tenemos en el stash a la rama nueva o la rama actual actualizada
```

ANEXOS

Copyright©Todos los derechos reservados.