

# La computación como herramienta de trabajo del profesional de ingeniería

Alejandro Pimentel  
[profpimentel9@gmail.com](mailto:profpimentel9@gmail.com)

Facultad de Ingeniería  
UNAM  
<https://qrگو.page.link/aeaVL>

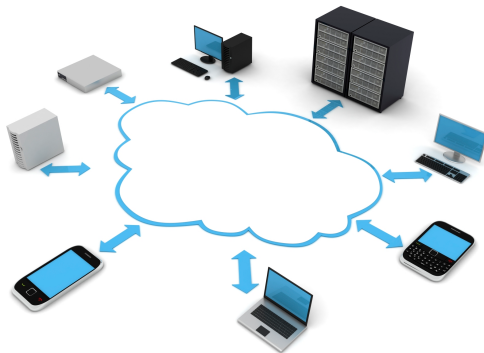


# Objetivo

Descubrir y utilizar herramientas de software que se ofrecen en Internet que permitan realizar actividades y trabajos académicos de forma organizada y profesional a lo largo de la vida escolar.

- 3 / 49

# La nube



- Servicios y almacenamiento accesibles mediante internet.
- Redes de servidores interconectados para operar de forma segura e ininterrumpida.





## Actividad:

Búsqueda y captura de todos los ejemplos, y al menos 3 combinaciones:

- "la Facultad de Ingeniería de la UNAM"
- noticias cch AND prepa
- noticias cch OR prepa
- noticias cch -prepa
- Facultad de \* UNAM
- define:ingeniería

## Actividad:

- **site:**ingenieria.unam.mx
- **filetype:**pdf (PDF,DOCX,TXT,PPT,etc)
- **intitle:**facultad ingenieria unam (**allintitle**)
- **inurl:**facultad ingenieria unam (**allinurl**)
- **intext:**facultad ingenieria unam (**allintext**)
- universidad **AROUND(2)** mexico





google scholar

 Books Images News

About 333,000,000 results (0.65 seconds)

Google Scholar

<https://scholar.google.com/> ▼

The system can't perform the operation now. Try again later.  
words, with the exact phrase, with at least one of the ...







# Google Imágenes



Koala.jpg x mi pc

Web

**Imágenes**

Videos

Más ▾

Herramientas de búsqueda

Cerca de 1,970,000 resultados (0.50 segundos)



Tamaño de la imagen:  
1024 × 768

Buscar esta imagen en otros tamaños:  
[Todos los tamaños](#) - [Grande](#)

Resultados de [mi pc](#)

[Mi PC Comunicaciones. Computadoras, Laptops, Impresoras y ...](#)  
[www.mipc.com.mx/](http://www.mipc.com.mx/)

En **MiPC.com.mx** contamos con los precios más bajos en Computadoras, Laptops y Accesorios para venta en línea en México. Contamos con 15 sucursales en ...

[Imágenes similares](#) - [Notificar imágenes](#)



## Actividad:

Buscar la imagen del escudo de la facultad, y otra imagen más que les interese

- 1 La nube
- 2 Comandos de búsqueda avanzada en Google
- 3 Control de versiones

Configuración inicial:

```
$ git config --global user.name "NOMBRE ALUMNO"
```

```
$ git config --global user.email nombre@correo.com
```

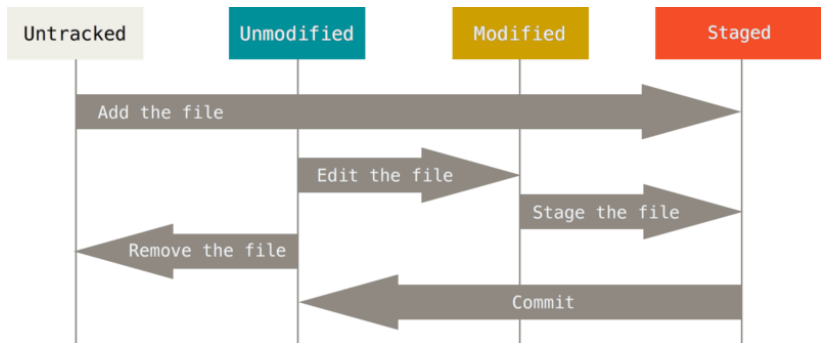
```
$ git config --global core.editor nano
```

Esto identifica al programador. Hay que recordar que estos sistemas están pensados para la colaboración. El SCV registra los cambios y el autor, la identificación es obligatoria. El editor es el editor por defecto que va a usar *git* cada que necesite una entrada de texto.





## git



```
$ git status
```



Si se tienen muchos archivos, se puede añadir la carpeta completa:

```
$ touch archivo1
```

```
$ touch archivo2
```

```
$ touch archivo3
```

```
$ touch archivo4
```

```
$ touch archivo5
```

```
$ git add .
```

```
$ git status
```

También es posible que quieran dar seguimiento a todo excepto a algunos archivos en particular, para eso se usa el archivo especial: *.gitignore*

```
$ echo "*.temp" > .gitignore
```

```
$ touch pruebas.temp
```

```
$ git add .
```

```
$ git status
```

`.gitignore` tiene algunas reglas:

- Líneas vacías y líneas que comiencen con '#' se ignoran.
- '\*' coincide con cualquier cosa.
- Se puede comenzar una línea con '/' para evitar recursividad. Es decir, para que solo se ignoren archivos en la carpeta raíz del proyecto.
- Se puede terminar una línea con '/' para especificar que se trata de un directorio completo el que debe ser ignorado.
- Se puede comenzar una línea con '!' para negarla. Es decir, se ignora todo lo que no coincida con esa línea.





El mensaje del *commit* se guarda junto con los cambios en el proyecto. Es posible consultar las entradas que se han hecho y usar al controlador de versiones como una bitácora:

```
$ git log
```

```
commit d49a6f135b018e9ca09045a5ac52dfc4ca9c0570
Author: pimentel <profpimentel9@gmail.com>
Date: Tue Jul 9 15:31:51 2019 -0500
```

Aquí narro los cambios que se hicieron

## git

Borrando y moviendo archivos:

```
$ rm archivo5
```

```
$ git status
```

```
$ git rm archivo4
```

```
$ git status
```

```
$ mv archivo3 archivo6
```

```
$ git status
```

```
$ git mv archivo2 archivo7
```

```
$ git status
```

```
$ git commit -m "Borrando y moviendo archivos 1"
```

```
$ git status
```

## git

```
$ git rm archivo3
```

```
$ git status
```

```
$ echo "Archivo temporal" > temp
```

```
$ git add .
```

```
$ git status
```

```
$ git rm --cached temp
```

```
$ git status
```

```
$ git commit -m "Borrando y moviendo archivos 2"
```

Gracias a la opción *cached*, el archivo *temp* se saca del seguimiento, pero no se elimina de la carpeta.

No es raro que uno cometa errores y deje *commits* incompletos:

```
$ mv temp no_temp
```

```
$ git add .
```

```
$ git commit -m "Siempre si quiero agregar el archivo temporal"
```

```
$ echo "Olvidé editar mi nuevo archivo!" >> no_temp
```

```
$ git add .
```

```
$ git commit --amend
```

Notarán que se abre una ventana de edición para que modifiquen, si gustan, el mensaje que ya se tenía en el *commit*. Ese comportamiento es el comportamiento por defecto de la instrucción *commit* si es que no le dan un mensaje con la opción *-m*.

Regresar a versiones anteriores:

- Hagamos algunos cambios primero

```
$ echo "Cambios al archivo1" >> archivo1
$ echo "Cambios al archivo6" >> archivo6
$ echo "Cambios al archivo7" >> archivo7
$ git add .
$ git commit -m "Cambios para hacer pruebas"

$ echo "Mas cambios al archivo1" >> archivo1
$ echo "Mas cambios al archivo6" >> archivo6
$ echo "Mas cambios al archivo7" >> archivo7
$ git add .
```

## git

```
$ git status
```

```
En la rama master
Cambios para hacer commit:
  (use «git reset HEAD <archivo>...» para sacar del stage)

    modificado:    archivo1
    modificado:    archivo6
    modificado:    archivo7
```

Si hemos cometido un error, lo primero que tenemos que hacer, es sacar al archivo del *stage* tal y como lo dice en la ayuda del *git status*

```
$ git reset HEAD archivo1
```

# git

```
$ git status
```

```
En la rama master
Cambios para hacer commit:
  (use «git reset HEAD <archivo>...» para sacar del stage)

    modificado:    archivo6
    modificado:    archivo7

Cambios no preparados para el commit:
  (use «git add <archivo>...» para actualizar lo que se con
  (use «git checkout -- <archivo>...» para descartar cambio
  rabajo)

    modificado:    archivo1
```

Luego, seguimos una vez mas la ayuda para descartar los cambios:

```
$ git checkout -- archivo1
```

Podemos quitar todos los archivos del *stage* al mismo tiempo:

```
$ git reset HEAD
```

Además, podemos descartar cambios hasta puntos específicos de nuestro *log*:

```
$ git log
```

Regresaremos el *archivo7* a como estaba cuando hicimos la entrada: "Borrando y moviendo archivos 1"



Cambios para hacer pruebas

```
commit 07c52588f6126544ec1d51c241e7352e2fd8066b
Author: pimentel <profpimentel9@gmail.com>
Date: Tue Jul 9 19:05:21 2019 -0500
```

Siempre si quiero agregar el archivo temporal

```
commit 80ec9c40bc70a0c8d7cd24a3546b85e28c5e1e4e
Author: pimentel <profpimentel9@gmail.com>
Date: Tue Jul 9 17:41:37 2019 -0500
```

Borrando y moviendo archivos 2

```
commit a65b9887da238822de89666fdf379b9dfdfbd4c2
Author: pimentel <profpimentel9@gmail.com>
Date: Tue Jul 9 17:35:15 2019 -0500
```

Borrando y moviendo archivos 1

## git

Ustedes deben usar su propio identificador para la siguiente instrucción:

```
$ git checkout a65b98 archivo7
```

```
$ git status
```

```
Cambios para hacer commit:
  (use «git reset HEAD <archivo>...» para sacar del stage)

    modificado:   archivo7

Cambios no preparados para el commit:
  (use «git add <archivo>...» para actualizar lo que se confirmará)
  (use «git checkout -- <archivo>...» para descartar cambios en el d
rabajo)

    modificado:   archivo6
```

El archivo fue "modificado" a como estaba en ese momento y añadido automáticamente al *stage*

NOTA IMPORTANTE: Usen la instrucción *checkout* siempre con un argumento que indique el archivo que quieren revertir. Si dejan ese espacio en blanco pueden ocasionar comportamientos inesperados.

Si quieren regresar el proyecto completo a una versión anterior, pueden poner como argumento la carpeta completa (recuerden resetear el *stage*):

```
$ git checkout a65b98 .
```

The world's leading software development platform - GitHub - Mozilla Firefox

The world's leading soft x +

← → ↺ 🏠 ⓘ GitHub, Inc. (US) | https://github.com ... 📄 ☆ 🔍 github → 📁 📄 📄 📄 📄 📄

🐙 Why GitHub? ▾ Enterprise Explore ▾ Marketplace Pricing ▾ Search GitHub [Z] Sign in Sign up

# Built for developers

GitHub is a development platform inspired by the way you work. From **open source** to **business**, you can host and review code, manage projects, and build software alongside 36 million developers.

**Username**

**Email**

**Password**

Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more.](#)

**Sign up for GitHub**

By clicking "Sign up for GitHub", you agree to our [Terms of Service](#) and [Privacy Statement](#). We'll occasionally send you account related emails.

https://github.com/join?source=header-home

Join GitHub · GitHub · Mozilla Firefox


Join GitHub · GitHub × +


GitHub, Inc. (US) https://github.com/join?source=github


github

Join GitHub

The best way to design, build, and ship software.

 **Step 1:**  
Set up your account

 **Step 2:**  
Choose your subscription

 **Step 3:**  
Tailor your experience

Create your personal account

**Username \***

This will be your username. You can add the name of your organization later.

**Email address \***

We'll occasionally send updates about your account to this inbox. We'll never share your email address with anyone.

**Password \***


Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more.](#)


Verify account


**You'll love GitHub**

**Unlimited** public repositories

**Unlimited** private repositories

 Limitless collaboration

 Frictionless development

 Open source community

GitHub - Where software is built - Mozilla Firefox

GitHub, Inc. (US) | https://github.com/join/plan | gmail

# \$0

per month

**Includes:**

- ∞ Unlimited public and private repositories
- ✓ 3 collaborators for private repositories
- ✓ Issues and bug tracking
- ✓ Project management

# \$7

per month

[\(view in MXN\)](#)

**Includes:**

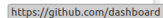
- ∞ Unlimited public and private repositories
- ∞ Unlimited collaborators
- ✓ Issues and bug tracking
- ✓ Project management
- ✓ [Advanced tools and insights](#)

Are you a [student](#)? Get access to the best developer tools for free with the [GitHub Student Developer Pack](#).

☐ **Help me set up an organization next**  
Organizations are separate from personal accounts and are best suited for businesses who need to manage permissions for many employees. [Learn more about organizations](#)

☐ **Send me updates on GitHub news, offers, and events**  
Unsubscribe anytime in your email preferences. [Learn more](#)

[Continue](#)



Your Actions - Mozilla Firefox

Your Actions × +

← → ↻ 🏠 ⓘ GitHub, Inc. (US) https://github.com/dashbo... ⋮ 📄 ☆ 🔍 gmail → 📁 📧 📺 📅 🌐 ☰

🐙 Search or jump to... Pull requests Issues Marketplace Explore 🔔 + 🌱

### Repositories

Your most active repositories will appear here. [Create a repository](#) or [explore repositories](#).

## Learn Git and GitHub without any code!

Using the Hello World guide, you'll create a repository, start a branch, write comments, and open a pull request.

[Read the guide](#)

[Start a project](#)

### Discover interesting projects and people to populate your personal news feed.

Your news feed helps you keep up with recent activity on

#### GitHub Sponsors Matching Fund

Ready to support open source? GitHub will match your contribution to developers during their first year in GitHub Sponsors.

#### Welcome to the new dashboard. Get closer to the stuff you care about most.

#### Discover repositories

**aspnet/AspNetCore**  
ASP.NET Core is a cross-platform .NET framework for building modern cloud-based web applications on Windows, Mac, or Linux.  
🟢 C# ★ 12.5k

**freeCodeCamp/freeCodeCamp**  
The <https://www.freecodecamp.org> open source codebase and curriculum. Learn to code for free together with millions of people.  
🟡 JavaScript ★ 304k

**openruse/kruise**  
Automate application workloads management on Kubernetes  
🟢 Go ★ 684

https://github.com/new



Create a New Repository - Mozilla Firefox

Create a New Repository x +

GitHub, Inc. (US) | https://github.com/new

gmail

## Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

---

**Owner** **Repository name \***

propfimentel9 / primer\_repositorio ✓

Great repository names are short and memorable. Need inspiration? How about **fantastic-spork**?

**Description** (optional)

---

☒ **Public**  
Anyone can see this repository. You choose who can commit.

☐ **Private**  
You choose who can see and commit to this repository.

---

Skip this step if you're importing an existing repository.

☒ **Initialize this repository with a README**  
This will let you immediately clone the repository to your computer.

Add .gitignore: **None** | Add a license: **None** ⓘ

**Create repository**

© 2019 GitHub, Inc. [Terms](#) [Privacy](#) [Security](#) [Status](#) [Help](#) [Contact GitHub](#) [Pricing](#) [API](#) [Training](#) [Blog](#) [About](#)

profimpintel9/primer\_repositorio - Mozilla Firefox

profimpintel9/primer\_ x +

GitHub, Inc. (US) | https://github.com/profimpintel9/primer\_ | gmail

Search or jump to... Pull requests Issues Marketplace Explore

profimpintel9 / primer\_repositorio

Watch 0 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Security Insights Settings

No description, website, or topics provided. Edit

Manage topics

1 commit 1 branch 0 releases 1 contributor

Branch: master New pull request Create new file Upload files Find File Clone or download

profimpintel9 Initial commit

README.md Initial commit

README.md

primer\_repositorio

Clone with HTTPS Use SSH

Use Git or checkout with SVN using the web URL.

https://github.com/profimpintel9/primer\_

Download ZIP

© 2019 GitHub, Inc. Terms Privacy Security Status Help Contact GitHub Pricing API Training Blog About

## git

Con la liga anterior, pueden hacer una copia de su proyecto de forma local. La siguiente instrucción crea una carpeta en la que se va a copiar el repositorio, y hace la configuración pertinente para que *github* funcione como su repositorio remoto:

```
$ git clone https://github.com/...
```

```
Clonar en «primer_repositorio»...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3),
remote: Total 3 (delta 0), reused 0 (d
Unpacking objects: 100% (3/3), done.
Comprobando la conectividad.. hecho.
```

Eso implica que pueden cargar los commits que hagan localmente en el repositorio:

```
$ cd primer_repositorio
$ echo "Creando un archivo de prueba" > ARCHIVO
$ git add .
$ git commit -m "Primer commit para github"
$ git push
```

El *push* les pedirá su usuario y contraseña de *github* para cargar sus commits al repositorio remoto.

profimpintel9/primer\_repositorio - Mozilla Firefox

profimpintel9/primer\_ x +

GitHub, Inc. (US) https://github.com/profimpintel9/assailant

Search or jump to... Pull requests Issues Marketplace Explore

profimpintel9 / primer\_repositorio

Watch 0 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Security Insights Settings

No description, website, or topics provided. Edit

Manage topics

2 commits 1 branch 0 releases 1 contributor

Branch: master New pull request Create new file Upload files Find File Clone or download

profimpintel9 Primer commit para github Latest commit c218942 2 minutes ago

ARCHIVO Primer commit para github 2 minutes ago

README ARCHIVO Initial commit 32 minutes ago

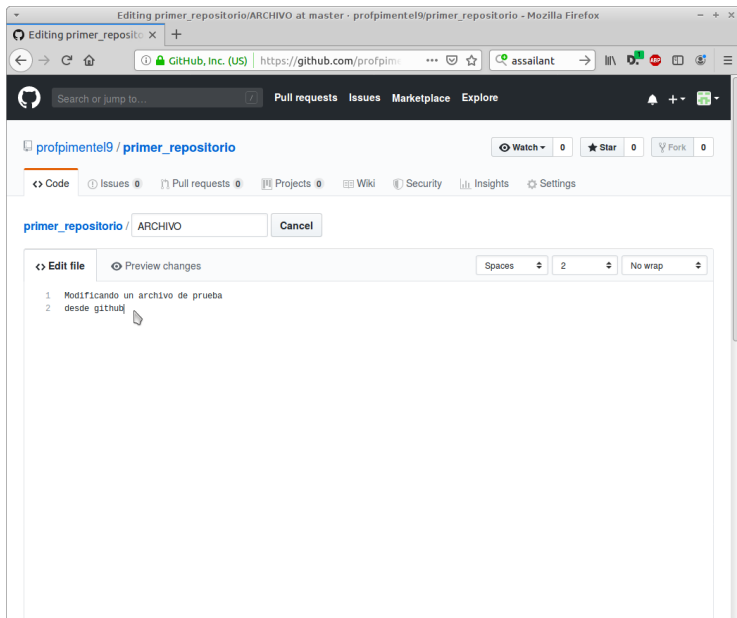
README.md

primer\_repositorio

© 2019 GitHub, Inc. Terms Privacy Security Status Help Contact GitHub Pricing API Training Blog About

https://github.com/profimpintel9/primer\_repositorio/blob/master/ARCHIVO

The screenshot displays a web browser window with the URL `https://github.com/propimintel9/primer_repositorio`. The page header includes the GitHub logo, a search bar, and navigation links: Pull requests, Issues, Marketplace, and Explore. The repository page for 'primer\_repositorio' by user 'propimintel9' is shown. It features a 'Watch' button with 0 notifications, a 'Star' button with 0 stars, and a 'Fork' button with 0 forks. The 'Code' tab is selected, showing a commit history table with one entry: 'propimintel9 Primer commit para github' (c218942, 4 minutes ago) by 1 contributor. The commit details show '2 lines (1 sloc)' and '29 Bytes'. The file content is '1 Creando un archivo de prueba'. A tooltip 'Edit this file' is visible over the 'History' button.



Editing primer\_repositorio ARCHIVO at master · profpime/primer\_repositorio - Mozilla Firefox

GitHub, Inc. (US) | https://github.com/profpime | ded message

## Commit changes

Título del commit

Detalles de los cambios que se hicieron en este commit

☒ Commit directly to the `master` branch.

☐ Create a new branch for this commit and start a pull request. [Learn more about pull requests.](#)

**Commit changes** Cancel

© 2019 GitHub, Inc. [Terms](#) [Privacy](#) [Security](#) [Status](#) [Help](#) [Contact GitHub](#) [Pricing](#) [API](#) [Training](#) [Blog](#) [About](#)



## git

Desde la terminal también pueden usar un "título" y un "mensaje largo" en sus *commits*:

```
$ git commit -m "Título" -m "Mensaje largo con detalles"
```

Su repositorio en *github* ahora está más adelantado que el local. Para actualizar el repositorio local con los nuevos cambios, se debe usar la instrucción:

```
$ git pull
```

Es recomendable usar la instrucción *pull* SIEMPRE antes de un *push*. Ya que si hay cambios en el repositorio remoto que no estén en su repositorio local, el sistema NO permitirá el *push*.