























# Sprint 3. Version control and databases access

## “Consultorio Online”

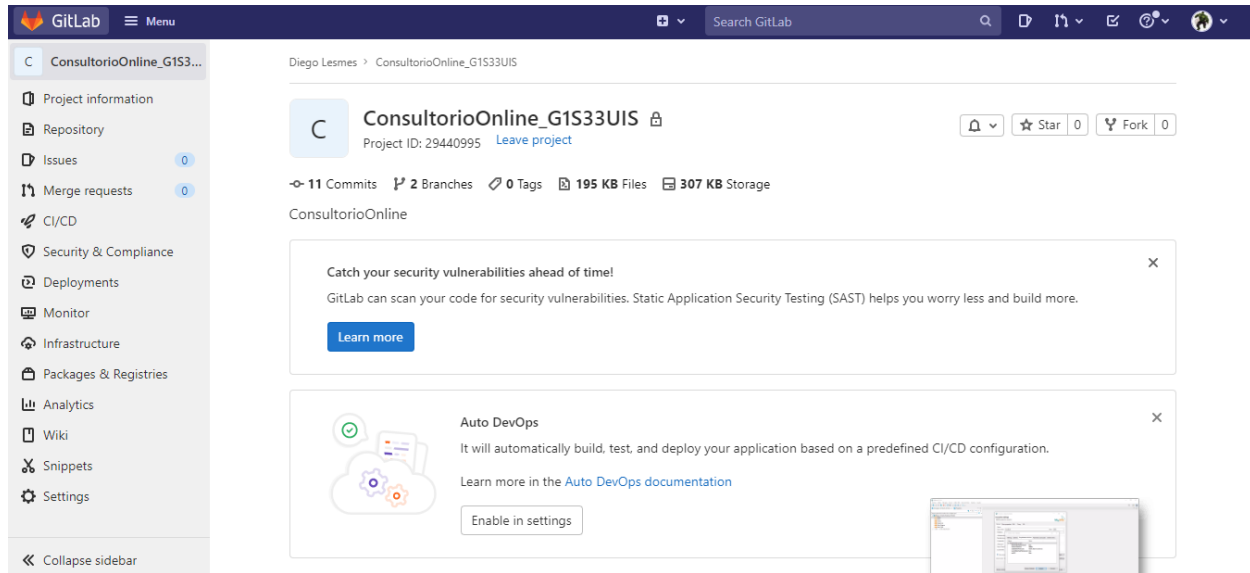
Group 1 S33 - September 16th, 2021

## Gitlab Members

Members 7					
Filter members					
Account	Source	Access granted	Access expires	Max role	Expiration
 <b>Diego Lesmes</b> <span>It's you</span> @DLesmes	Direct member	3 days ago by <a href="#">Diego Lesmes</a>	No expiration set	Maintainer	Expiration date 
 <b>Elkin Díaz</b> @ediazMintic	Direct member	3 days ago by <a href="#">Diego Lesmes</a>	No expiration set	Maintainer	Expiration date  
 <b>Miller Puentes</b> @millerpuentes	Direct member	3 days ago by <a href="#">Diego Lesmes</a>	No expiration set	Maintainer	Expiration date  
 <b>Nathalia Moreno Sánchez</b> @nathy92moreno	Direct member	3 days ago by <a href="#">Diego Lesmes</a>	No expiration set	Maintainer	Expiration date  
 <b>Nicolás Rojas</b> @Nico Rojas25	Direct member	3 days ago by <a href="#">Diego Lesmes</a>	No expiration set	Maintainer	Expiration date  
 <b>Pedro Ortiz</b> @danjack010892	Direct member	3 days ago by <a href="#">Diego Lesmes</a>	No expiration set	Maintainer	Expiration date  
 <b>Tutor 11</b> @tutor11	Direct member	3 days ago by <a href="#">Diego Lesmes</a>	No expiration set	Maintainer	Expiration date  

## Local clones

The connection between GitLab and Sourcetree, cloning the repo locally:



## Diego Lesmes' clone

```

asf@asf-SVF15A17CLB:~/Documents/ConsultorioOnline_G1S33UIS$ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
asf@asf-SVF15A17CLB:~/Documents/ConsultorioOnline_G1S33UIS$ ls -al
total 28
drwxrwxr-x 6 asf asf 4096 sep 10 20:22 .
drwxr-xr-x 27 asf asf 4096 sep  8 23:34 ..
drwxrwxr-x 2 asf asf 4096 sep 10 20:22 diseño
drwxrwxr-x 2 asf asf 4096 sep 10 20:22 documentación
drwxrwxr-x 8 asf asf 4096 sep 10 20:22 .git
-rw-rw-r-- 1 asf asf 173 sep  8 23:38 README.md
drwxrwxr-x 2 asf asf 4096 sep 10 20:22 software
asf@asf-SVF15A17CLB:~/Documents/ConsultorioOnline_G1S33UIS$ git config --list
user.email=ing.dlesmes@gmail.com
user.name=DLesmes
core.repositoryformatversion=0
core.filemode=true
core.bare=false
core.logallrefupdates=true
remote.origin.url=git@gitlab.com:DLesmes/ConsultorioOnline_G1S33UIS.git
remote.origin.fetch=+refs/heads/*:refs/remotes/origin/*
branch.main.remote=origin
branch.main.merge=refs/heads/main
asf@asf-SVF15A17CLB:~/Documents/ConsultorioOnline_G1S33UIS$

```



## Natalia Moreno's Clone

```

MINGW64/c/Users/nathalia.moreno/Documents/ConsultorioOnline_G1S33UIS
nathalia.moreno@NATHALIAP-MORENO MINGW64 ~/Documents/ConsultorioOnline_G1S33UIS
(nathalia_moreno_branch)
$ git branch
* main
  nathalia_moreno_branch

nathalia.moreno@NATHALIAP-MORENO MINGW64 ~/Documents/ConsultorioOnline_G1S33UIS
(nathalia_moreno_branch)
$ git config --list
diff.astextplain.textconv=astextplain
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
http.sslbackend=openssl
http.sslcainfo=C:/Program Files/Git/mingw64/ssl/certs/ca-bundle.crt
core.autocrlf=true
core.fscache=true
core.symlinks=false
pull.rebase=false
credential.helper=manager-core
credential.https://dev.azure.com.usehttppath=true
init.defaultbranch=master
user.name=Nathalia Moreno Sánchez
user.email=natha92moreno@gmail.com
difftool.sourcetree.cmd='' "$LOCAL" "$REMOTE"
mergetool.sourcetree.cmd=''
mergetool.sourcetree.trustexitcode=true
core.repositoryformatversion=0
core.filemode=false
core.bare=false
core.logallrefupdates=true
core.symlinks=false
core.ignorecase=true
remote.origin.url=https://gitlab.com/DLesmes/ConsultorioOnline_G1S33UIS.git
remote.origin.fetch=+refs/heads/*:refs/remotes/origin/*
branch.main.remote=origin
branch.main.merge=refs/heads/main

nathalia.moreno@NATHALIAP-MORENO MINGW64 ~/Documents/ConsultorioOnline_G1S33UIS (nathalia_moreno_branch)

```

## Miller Puentes' Clone

The screenshot shows the Git GUI interface with the repository **ConsultorioOnline\_G1S33UIS** open. The left sidebar displays the workspace structure, including branches (**main**, **millier\_puentes\_branch**), tags, remotes, and stashes. The main area shows a graph of commits and a table of commit history.

Commit	Date	Author	Commit
56896c9	13 sept. 2021 19:37	Miller Puentes <miller.puentes@gmail.com>	es un ejemplo
809eb48	13 sept. 2021 19:37	Miller Puentes <miller.puentes@gmail.com>	Se elimina el ejemplo
eccc727	10 sept. 2021 20:37	DLesmes <ing.dles@gmail.com>	first file in my own branch
8421e4a	9 sept. 2021 19:22	Miller Puentes <miller.puentes@gmail.com>	Estructura del repositorio creado
6a73c24	8 sept. 2021 23:38	DLesmes <ing.dles@gmail.com>	Readme edited
92e15b6	8 sept. 2021 19:26	Diego Lesmes <diego.lesmes@gmail.com>	Update README.md
5a1d522	8 sept. 2021 19:17	Miller Puentes <miller.puentes@gmail.com>	Update README.md
5211f6c	7 sept. 2021 19:33	Nathalia Moreno <nathalia.moreno@gmail.com>	Update README.md
0d0fb59	7 sept. 2021 19:33	Miller Puentes <miller.puentes@gmail.com>	Update README.md
b460b91	7 sept. 2021 19:19	Nicolas Rojas <nrojas@gmail.com>	Update README.md
91de9a8	7 sept. 2021 19:19	Nicolas Rojas <nrojas@gmail.com>	Update README.md
d14cbda	26 ago. 2021 21:43	Diego Lesmes <diego.lesmes@gmail.com>	Hi im Pedro
ace1795	26 ago. 2021 16:00	Jack Ortiz <dandjad@gmail.com>	Initial commit
c6be95c	25 ago. 2021 20:31	Diego Lesmes <diego.lesmes@gmail.com>	

The bottom section shows the file status for **ejemplo.txt**, indicating it has been modified. The commit message for the selected commit is: **es un ejemplo**.



## Pedro Ortiz's Clone

```
jack@beast-pc:~/Documents/Consultorio/ConsultorioOnline_G1S33UIS$ git status
On branch pedro_ortiz_branch
nothing to commit, working tree clean
jack@beast-pc:~/Documents/Consultorio/ConsultorioOnline_G1S33UIS$ ls -lh
total 108K
drwxr-xr-x 2 jack jack 4.0K Sep  9 19:32 diseño
drwxr-xr-x 2 jack jack 4.0K Sep  9 19:32 documentación
-rw-rw-r-- 1 jack jack  0 Sep 13 19:47 ejemplo.txt
-rw-r--r-- 1 jack jack 173 Sep  9 19:32 README.md
drwxr-xr-x 2 jack jack 4.0K Sep  9 19:32 software
-rw-r--r-- 1 jack jack 90K Sep  7 19:27 spidy2.png
jack@beast-pc:~/Documents/Consultorio/ConsultorioOnline_G1S33UIS$ git config --list
user.name=Jack Ortiz
user.email=danjack010892@gmail.com
core.repositoryformatversion=0
core.filemode=true
core.bare=false
core.logallrefupdates=true
remote.origin.url=https://gitlab.com/DLesmes/ConsultorioOnline_G1S33UIS.git
remote.origin.fetch=+refs/heads/*:refs/remotes/origin/*
branch.main.remote=origin
branch.main.merge=refs/heads/main
taggrouping.groups=
taggrouping.singles=
pull.rebase=true
jack@beast-pc:~/Documents/Consultorio/ConsultorioOnline_G1S33UIS$
```

## Nicolas Rojas' Clone

```
MINGW64~/c:/Users/Nicolás Rojas/Desktop/repositorio Misión TIC/ConsultorioOnline_G1S33UIS
Nicolás Rojas@DESKTOP-CJQ81VO MINGW64 ~/Desktop/repositorio Misión TIC/ConsultorioOnline_G1S33UIS (nicolas_rojas_branch)
$ ls -al
total 10
drwxr-xr-x 1 Nicolás Rojas 197121  0 Sep 13 21:22 ./
drwxr-xr-x 1 Nicolás Rojas 197121  0 Sep 13 20:40 ../
drwxr-xr-x 1 Nicolás Rojas 197121  0 Sep 13 21:22 .git/
-rw-r--r-- 1 Nicolás Rojas 197121 183 Sep 13 20:39 README.md
drwxr-xr-x 1 Nicolás Rojas 197121  0 Sep 13 20:39 diseño/
drwxr-xr-x 1 Nicolás Rojas 197121  0 Sep 13 20:39 documentación/
-rw-r--r-- 1 Nicolás Rojas 197121 24 Sep 13 21:22 'ejemplo Nicolás.txt'
-rw-r--r-- 1 Nicolás Rojas 197121  0 Sep 13 20:39 ejemplo.txt
drwxr-xr-x 1 Nicolás Rojas 197121  0 Sep 13 20:39 software/

Nicolás Rojas@DESKTOP-CJQ81VO MINGW64 ~/Desktop/repositorio Misión TIC/ConsultorioOnline_G1S33UIS (nicolas_rojas_branch)
$ git config --list
diff.astextplain.textconv=astextplain
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
http.sslbackend=openssl
http.sslcainfo=C:/Program Files/Git/mingw64/ssl/certs/ca-bundle.crt
core.autocrlf=true
core.fsckache=true
core.symbols=false
pull.rebase=false
credential.helper=manager-core
credential.https://dev.azure.com.usehttppath=true
init.defaultbranch=master
core.editor="C:/Users/Nicolás Rojas/AppData/Local/Programs/Microsoft VS Code/Code.exe" --wait
user.email=rojasnicolas25.nr@gmail.com
user.name=Nicolás Rojas
core.repositoryformatversion=0
core.filemode=false
core.bare=false
core.logallrefupdates=true
core.symbols=false
core.ignorecase=true
remote.origin.url=https://gitlab.com/DLesmes/ConsultorioOnline_G1S33UIS
remote.origin.fetch=+refs/heads/*:refs/remotes/origin/*
branch.main.remote=origin
branch.main.merge=refs/heads/main

Nicolás Rojas@DESKTOP-CJQ81VO MINGW64 ~/Desktop/repositorio Misión TIC/ConsultorioOnline_G1S33UIS (nicolas_rojas_branch)
$
```



## Source code

The screenshot shows the GitLab interface for a repository named 'ConsultorioOnline\_G1S33'. The left sidebar contains navigation links: Project information, Repository, Issues (0), Merge requests (0), CI/CD, Security & Compliance, Deployments, Monitor, Infrastructure, Packages & Registries, Analytics, Wiki, Snippets, and Settings.

The main content area displays the repository structure created by 'Miller Puentes' 3 days ago. It includes buttons for 'Upload File', 'README', 'Add LICENSE', 'Add CHANGELOG', 'Add CONTRIBUTING', 'Add Kubernetes cluster', 'Set up CI/CD', 'Configure Integrations', and 'Add Security Testing'.

Name	Last commit	Last update
diseño	Estructura del repositorio creado	3 days ago
documentación	Estructura del repositorio creado	3 days ago
software	Estructura del repositorio creado	3 days ago
README.md	Readme edited	4 days ago

Below the table, there is a section for 'README.md' with the title 'ConsultorioOnline\_G1S33'.

The screenshot shows a Windows File Explorer window with the address bar set to 'Este equipo > Documentos > ConsultorioOnline\_G1S33UIS'. The left sidebar shows the navigation pane with 'Documentos' selected.

Nombre	Fecha de modificación	Tipo	Tamaño
diseño	9/09/2021 7:14 p. m.	Carpeta de archivos	
documentación	9/09/2021 7:19 p. m.	Carpeta de archivos	
software	9/09/2021 7:16 p. m.	Carpeta de archivos	
README.md	9/09/2021 6:39 p. m.	Archivo de origen ...	1 KB



## Work branches

Protected branches can be managed in [project settings](#).

<b>main</b> <small>default</small> <small>protected</small> 5a85433a · Prueba · 2 minutos ago				
<b>nicolas_rojas_branch</b> <small>merged</small> 4c504663 · Hola, esto es una prueba · 29 minutos ago	1	0	Merge request	Compare
<b>pedro_ortiz_branch</b> 5c269200 · added peter profile pic · 1 hour ago	2	1	Merge request	Compare
<b>nathalia_moreno_branch</b> bf3b3302 · Hola · 1 hour ago	2	1	Merge request	Compare
<b>miller_puentes_branch</b> 809eb485 · Se elimina el ejemplo · 1 hour ago	2	1	Merge request	Compare
<b>diego_lesmes_branch</b> eccc7277 · first file in my own branch · 3 days ago	3	1	Merge request	Compare

The screenshot shows the Git GUI interface. The top menu bar includes File, Edit, View, Repository, Actions, Tools, and Help. The toolbar contains buttons for Commit, Pull, Push, Fetch, Branch, Merge, Stash, Discard, and Tag. The left sidebar shows the workspace with options like File Status, History, Search, and a list of branches (main, tags, remotes, stashes). The main area displays a graph of commits with a table of commit details. The table has columns for Date, Author, and Commit ID. The commit history shows a sequence of updates to README.md, followed by a merge pull request from Jackie0513/main, and then a new commit by Miller Puentes.

Date	Author	Commit
9 sept. 2021 19:22	Miller Puentes <miller.puentes@gmail.com>	8421e4a
8 sept. 2021 23:38	Diego Lesmes <ing.dies@gmail.com>	6a73c24
8 sept. 2021 19:26	Diego Lesmes <ing.dies@gmail.com>	92e15b6
8 sept. 2021 19:17	Miller Puentes <miller.puentes@gmail.com>	5a1d522
7 sept. 2021 19:33	Nathalia Moreno S	5211f6c
7 sept. 2021 19:33	Miller Puentes <miller.puentes@gmail.com>	0d0fb59
7 sept. 2021 19:19	Nicolás Rojas <roji@gmail.com>	b460b91
7 sept. 2021 19:19	Nicolás Rojas <roji@gmail.com>	91de9a8
26 ago. 2021 21:43	Diego Lesmes <ing.dies@gmail.com>	d14cbda
26 ago. 2021 16:00	Jack Ortiz <danjacl@gmail.com>	ace1795
25 ago. 2021 20:31	Diego Lesmes <ing.dies@gmail.com>	c6be95c

The bottom section shows the file explorer for 'diseño/readme.txt'. It displays the commit history for this file, including the commit by Miller Puentes (8421e4a) which added the file. The file content is shown as 'Carpeta creada para almacenar la documentación del software'.

## Conflicts resolution

During the life cycle of a team-driven software project, team members will probably have the need to make changes to the same source code file at the same time. A VCS monitors and assists in conflicts between multiple developers. These conflict resolution operations leave an audit trail that provides information about the history of a project.

["https://bitbucket.org/product/es/version-control-software"](https://bitbucket.org/product/es/version-control-software)

## Conflict



A conflict occurs when the system cannot adequately handle changes made by two or more users to the same file. For example, if this sequence of circumstances occurs:

1. The users X and Y deploy versions of the file A which lines n1 until n2 are common.
2. The user X sends changes between lines n1 and n2 on the file A.
3. The user Y doesn't update the file A later than the user X sends it.
4. The user Y makes changes between lines n1 and n2.
5. The user Y tries later to send these changes to the file A.

The system is unable to merge the changes. User Y must resolve the conflict by combining the changes, or choosing one of them to discard the other.

["https://es.wikipedia.org/wiki/Control\\_de\\_versiones"](https://es.wikipedia.org/wiki/Control_de_versiones)

In the software development area, having a version control tool that allows the registration of the modifications that have been made to the programs or documents, and that provides the previous versions of the software, can avoid conflicts when generating a final product.

A conflict occurs when two or more people try to make different changes to the same piece of code. All version control systems detect these conflicts automatically and notify at least one of the humans involved that their changes conflict with someone else's. It is then their job to resolve the conflict and communicate this resolution to the version control system. Git's ability to resolve conflicts is very useful, but conflict resolution takes time and effort, and can introduce errors if the conflicts are not solved correctly. If you find yourself resolving many conflicts in a project, keep these technical approaches in mind to reduce them:

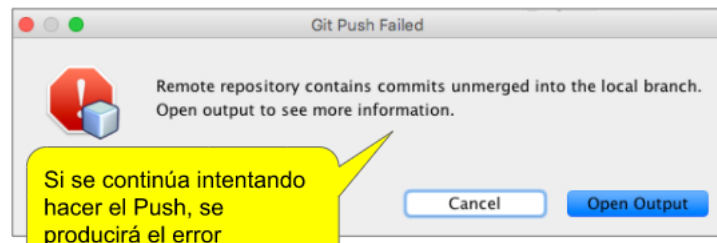
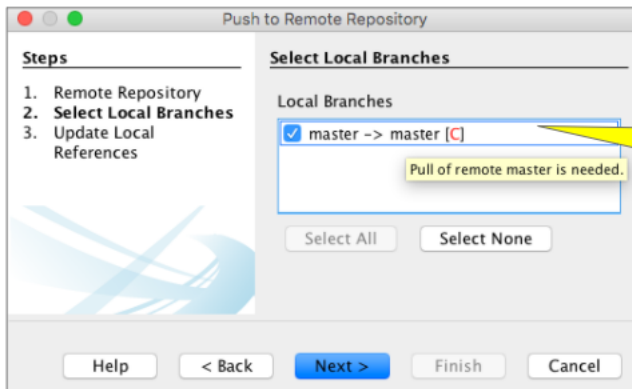
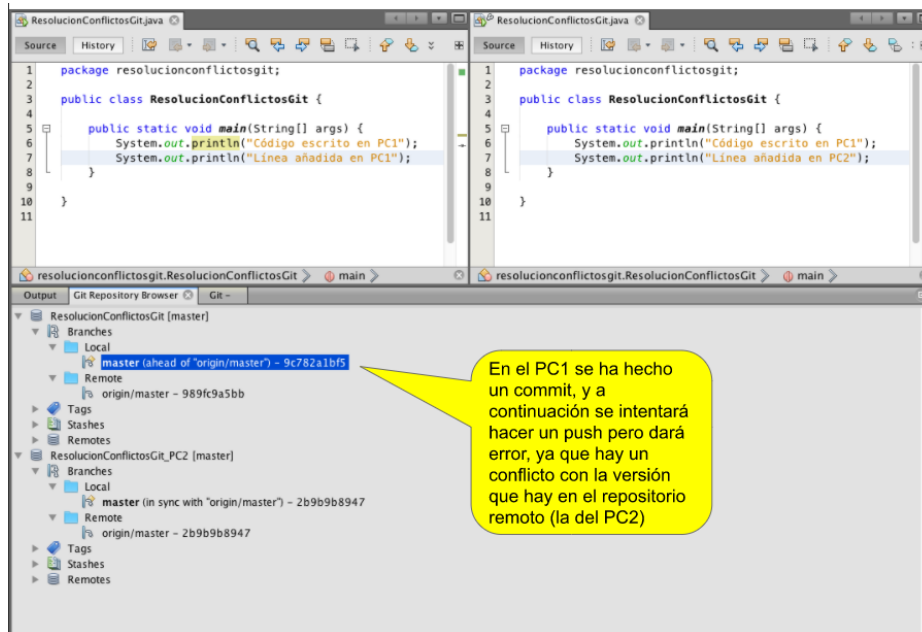
- Make a pull with a higher frequent, especially before start a new task
- Use thematic branches to separate work, joining the main branch - master- when they are complete
- Make short comments and concise
- When appropriate, split large files into several smaller ones so that two authors are less likely to alter the same file simultaneously

Conflicts can also be minimized with project management strategies:

- Clarify with your collaborators who's the responsible for each area
- Discuss with your collaborators in what order the tasks should be performed so that tasks that can change the same lines are not worked simultaneously.
- If the conflicts are in style (e.g. tabs vs. spaces), establish a convention that governs the project and use code style tools

Example:

GIT Conflict resolution from NETBEANS

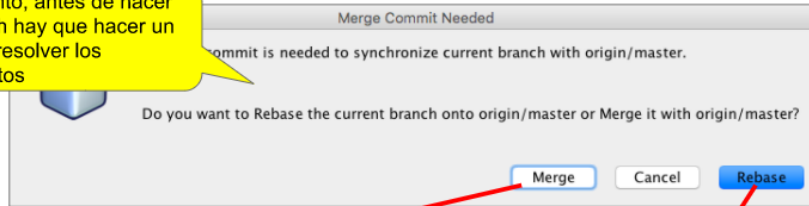


```
Remote Repository Updates
Branch Update : master
Old Id       : 2b9b9b89477565deefb72b0b1fb40122296a382d
New Id       : 9c782a1bf55fa1714228b28441dbd4ab0223eeb9
Result       : REJECTED_NONFASTFORWARD
```





Por tanto, antes de hacer el Push hay que hacer un Pull y resolver los conflictos



```
package resolucionconflictosgit;

public class ResolucionConflictosGit {

    public static void main(String[] args) {
        System.out.println("Código escrito en PC1");
        <<<<<< HEAD
        System.out.println("Línea añadida en PC1");
        <=====
        System.out.println("Línea añadida en PC2");
        >>>>>> origin/master
    }
}
```



```
package resolucionconflictosgit;

public class ResolucionConflictosGit {

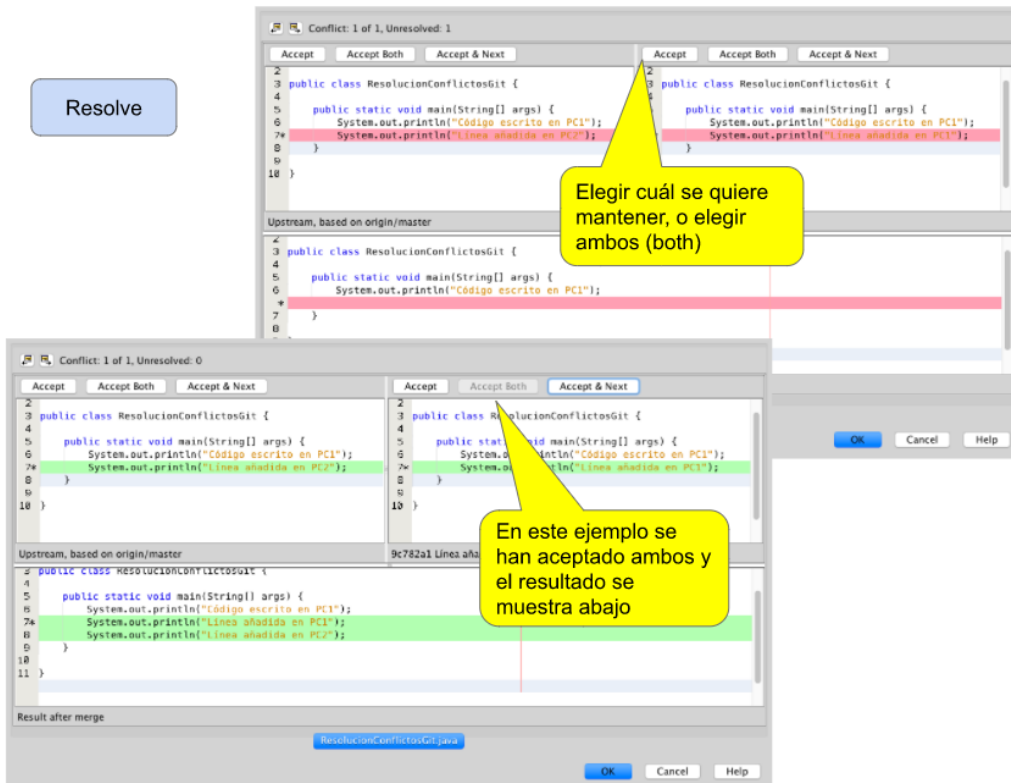
    public static void main(String[] args) {
        System.out.println("Código escrito en PC1");
        <<<<<< Upstream, based on origin/master
        System.out.println("Línea añadida en PC2");
        <=====
        System.out.println("Línea añadida en PC1");
        >>>>>> 9c782a1 Línea añadida en PC1
    }
}
```

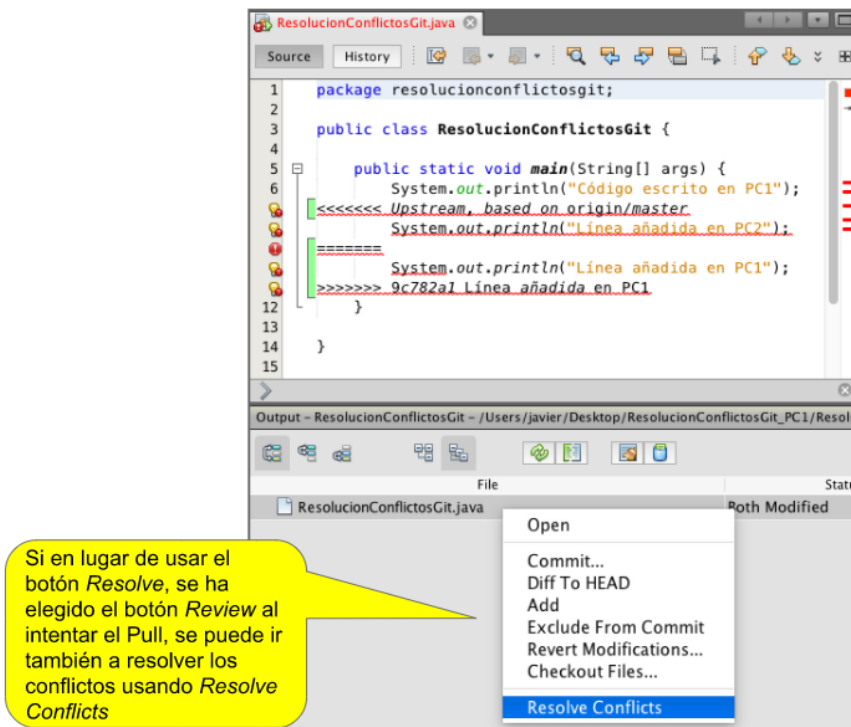


Resolve

Elegir cuál se quiere mantener, o elegir ambos (both)

En este ejemplo se han aceptado ambos y el resultado se muestra abajo





["https://javiergarciaescobedo.es/cvs-sistemas-de-control-de-versiones/99-git/413-resolucion-de-conflictos-en-git-desde-netbeans"](https://javiergarciaescobedo.es/cvs-sistemas-de-control-de-versiones/99-git/413-resolucion-de-conflictos-en-git-desde-netbeans)

Conflict management is a very important part of any VCS, not only because of its resolution capacity, but also because of the ability to detect them. If a VCS does not detect a conflict, it can introduce code corruption, the results of which are often problematic. On detecting a conflict that the VCS cannot resolve automatically, the user is informed and is expected to be resolved by hand or with one of the tools designed to assist in resolving these issues. Fortunately, most modern VCS are very focused on this problem and have very good conflict detection and resolution algorithms.

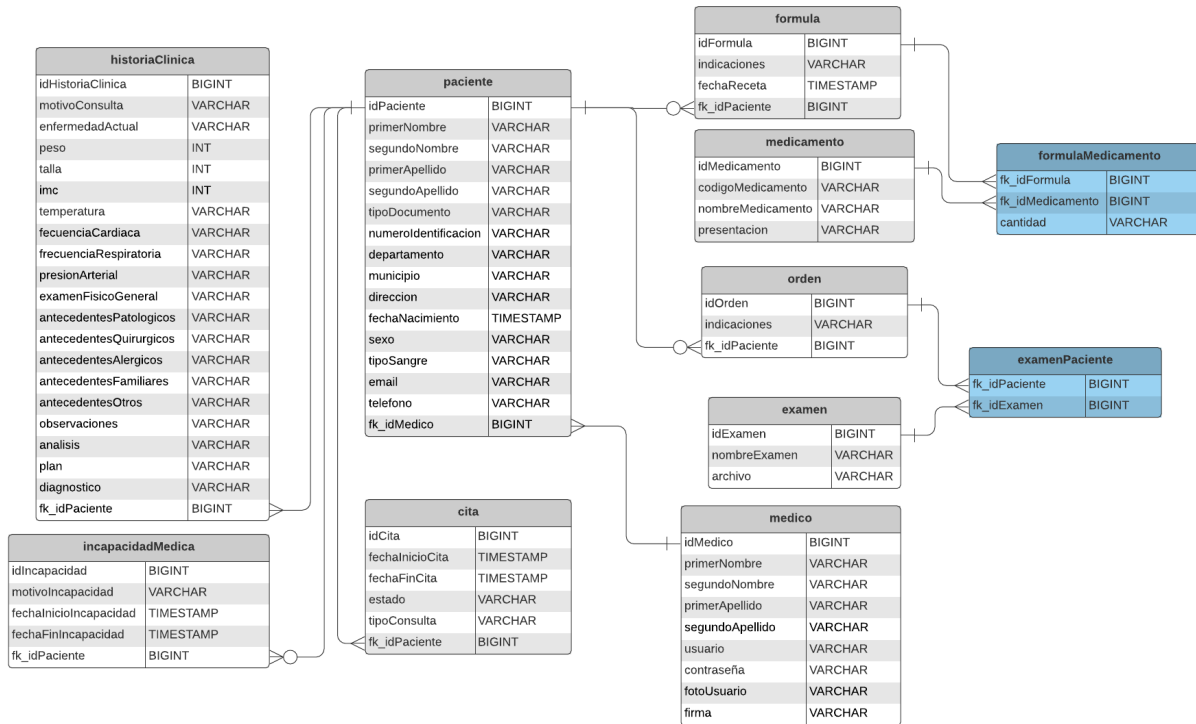
## Entity relational databases diagram

### Details



Entity Relation Diagram OnLine Consultor

Grupo1, S33 UIS MisiónTic2022



[ER Diagram - to edit](#)

Jira evidence

[JIRA's Project](#)

Setting Spring 3



Jira Software

Tu trabajo

Proyectos

Filtros

Paneles

Personas

Aplicaciones

Crear

S33Grupo1

Proyecto de software

Hoja de ruta

Backlog

Tablero

Informes

Incidencias

Código

Implementaciones

Versiones

De guardia

Páginas de proyectos

Añadir acceso rápido

Configuración del pro...

Proyectos / S33Grupo1

Sprint 3

Descripción del sprint: Cada integrante del grupo de trabajo debe crear una cuenta en GitLab; luego el líder del grupo, debe crear un proyecto en blanco y agregar a los integrantes del grupo, al formador y al tutor. Para ello, ver instrucciones en Instructivo GitLab y SourceTree diapositiva 1. Ahora, se usará el software SourceTree para manejar de forma gráfica el repositorio. Instalarlo desde el siguiente enlace: Sourcetree | Free Git GUI for Mac and Windows (sourcetreeapp.com). Posteriormente, lleve a cabo las siguientes acciones: Clonar el proyecto localmente. Ver instrucciones en Instructivo GitLab y SourceTree diapositiva 2. Copiar los códigos fuentes en la rama principal y subirlos al repositorio. Ver instrucciones en Instructivo GitLab y SourceTree diapositiva 3. Crear una rama de trabajo. Ver instrucciones en Instructivo GitLab y SourceTree diapositiva 4. Integrar la rama al proyecto principal. Descargar cambios de la rama principal a mi rama local. Investigar sobre resolución de conflictos.

Q

Etiqueta

TO DO 7 INCIDENCIAS

TR\_023\_Diagrama conceptual de la base de datos

☒ S33GROUP-51
 

8

TR\_024\_Subir Tareas al Backlog y colocar estimación de las HU

☒ S33GROUP-53
 

1

TR\_019\_UI/UX Vistas de usuario de la aplicación

FrontEnd

☒ S33GROUP-29
 

13

TR\_023\_Diagrama conceptual de la base de datos

☒ S33GROUP-52
 

8

TR\_025\_Revisar los últimos requerimientos de las ramas mencionadas en el sprint sobre Git

☒ S33GROUP-55
 

5

TR\_026\_Investigar sobre la resolución de conflictos en el control de versiones

☒ S33GROUP-56
 

3

IN PROGRESS 3 INCIDENCIAS

TR\_011\_Crear modelo Base de datos

BackEnd

☒ S33GROUP-20
 

5

TR\_025\_Revisar los últimos requerimientos de las ramas mencionadas en el sprint sobre Git

☒ S33GROUP-54
 

5

TR\_028\_Dejar listos los campos de evidencia en el documento Sprint 3

☒ S33GROUP-58
 

1

PEER REVIEW 1 INCIDENCIA

TR\_027\_Crear el documento del Sprint 3 y el documento de requerimientos

☒ S33GROUP-57
 

1

DONE

+

Estás en un proyecto gestionado por el equipo

Más información

Proyectos / S33Grupo1

## Sprint 3

Descripción del sprint: Cada integrante del grupo de trabajo debe crear una cuenta en GitLab; luego el líder del grupo, debe crear un proyecto en blanco y agregar a los integrantes del grupo, al formador y al tutor. Para ello, ver instrucciones en Instructivo GitLab y SourceTree diapositiva 1. Ahora, se usará el software SourceTree para manejar de forma gráfica el repositorio. Instalarlo desde el siguiente enlace: Sourcetree | Free Git GUI for Mac and Windows (sourcetreeapp.com). Posteriormente, lleve a cabo las siguientes acciones: Clonar el proyecto localmente. Ver instrucciones en Instructivo GitLab y SourceTree diapositiva 2. Copiar los códigos fuentes en la rama principal y subirlos al repositorio. Ver instrucciones en Instructivo GitLab y SourceTree diapositiva 3. Crear una rama de trabajo. Ver instrucciones en Instructivo GitLab y SourceTree diapositiva 4. Integrar la rama al proyecto principal. Descargar cambios de la rama principal a mi rama local. Investigar sobre resolución de conflictos.



Q

Etiqueta

TO DO

TR\_025\_Crear las carpetas del proyecto en el repositorio

☒ S33GROUP-55
 

5

TR\_025\_Revisar los últimos requerimientos de las ramas mencionadas en el sprint sobre Git

☒ S33GROUP-54
 

5

TR\_023\_Diagrama conceptual de la base de datos

☒ S33GROUP-51
 

8

IN PROGRESS 4 INCIDENCIAS

TR\_024\_Subir Tareas al Backlog y colocar estimación de las HU

☒ S33GROUP-53
 

1

TR\_027\_Crear el documento del Sprint 3 y el documento de requerimientos

☒ S33GROUP-57
 

1

TR\_019\_UI/UX Vistas de usuario de la aplicación

FrontEnd

☒ S33GROUP-29
 

13

PEER REVIEW 6 INCIDENCIAS

TR\_024\_Subir Tareas al Backlog y colocar estimación de las HU

☒ S33GROUP-53
 

1

TR\_027\_Crear el documento del Sprint 3 y el documento de requerimientos

☒ S33GROUP-57
 

1

TR\_019\_UI/UX Vistas de usuario de la aplicación

FrontEnd

☒ S33GROUP-29
 

13

DONE 2 INCIDENCIAS

TR\_029\_Actualizar Cronograma del proyecto

☒ S33GROUP-59
 

1

TR\_030\_Explicar la creación de ramas en el proyecto a los del equipo

☒ S33GROUP-60
 

1

+



## Burnup Report

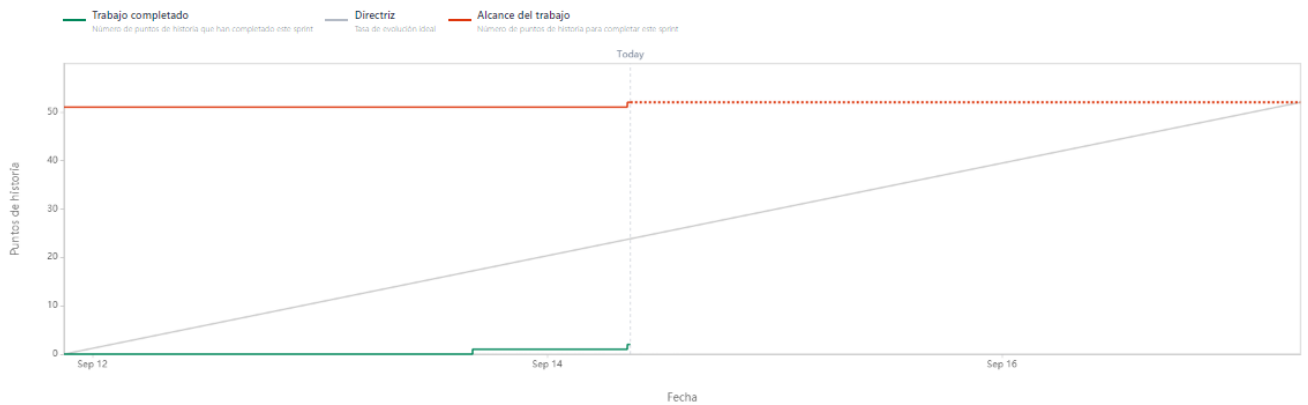
Proyectos / S33Grupo1 / Informes

### Informe de trabajo completado

[Cómo leer este informe](#)

Fecha - 11 de septiembre de 2021 - 17 de septiembre de 2021

**Objetivo del sprint** - Descripción del sprint: Cada integrante del grupo de trabajo debe crear una cuenta en GitLab; luego el líder del grupo, debe crear un proyecto en blanco y agregar a los integrantes del grupo, al formador y al tutor. Para ello, ver instrucciones en Instructivo GitLab y SourceTree diapositiva 1. Ahora, se usará el software SourceTree para manejar de forma gráfica el repositorio. Instalarlo desde el siguiente enlace: Sourcetree | Free Git GUI for Mac and Windows (sourcetreeapp.com). Posteriormente, lleve a cabo las siguientes acciones: Clonar el proyecto localmente. Ver instrucciones en Instructivo GitLab y SourceTree diapositiva 2. Copiar los códigos fuentes en la rama principal y subirlos al repositorio. Ver instrucciones en Instructivo GitLab y SourceTree diapositiva 3. Crear una rama de trabajo. Ver instrucciones en Instructivo GitLab y SourceTree diapositiva 4. Integrar la rama al proyecto principal. Descargar cambios de la rama principal a mi rama local. Investigar sobre resolución de conflictos.



## Sprint burndown chart

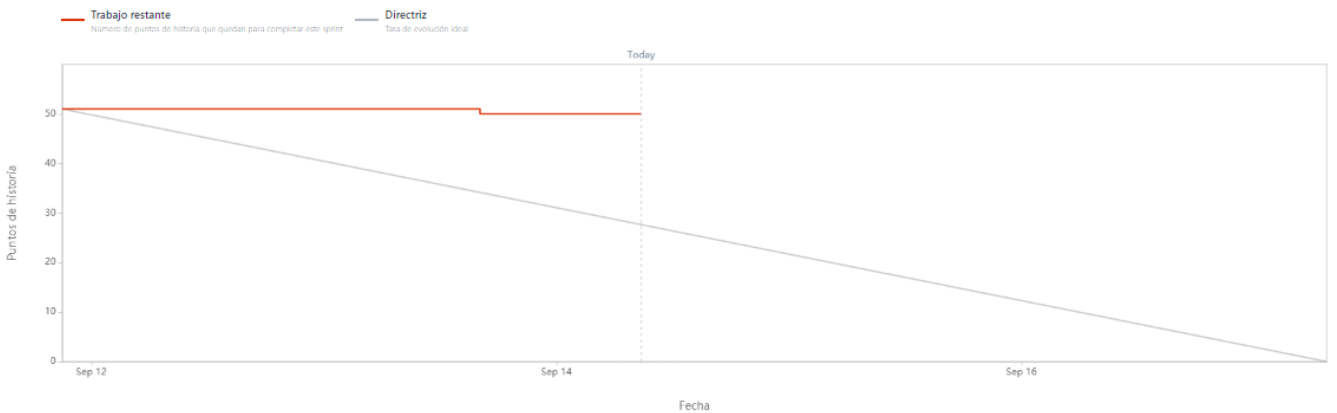
Proyectos / S33Grupo1 / Informes

### Gráfica de trabajo pendiente en el sprint

[Cómo leer este informe](#)

Fecha - 11 de septiembre de 2021 - 17 de septiembre de 2021

**Objetivo del sprint** - Descripción del sprint: Cada integrante del grupo de trabajo debe crear una cuenta en GitLab; luego el líder del grupo, debe crear un proyecto en blanco y agregar a los integrantes del grupo, al formador y al tutor. Para ello, ver instrucciones en Instructivo GitLab y SourceTree diapositiva 1. Ahora, se usará el software SourceTree para manejar de forma gráfica el repositorio. Instalarlo desde el siguiente enlace: Sourcetree | Free Git GUI for Mac and Windows (sourcetreeapp.com). Posteriormente, lleve a cabo las siguientes acciones: Clonar el proyecto localmente. Ver instrucciones en Instructivo GitLab y SourceTree diapositiva 2. Copiar los códigos fuentes en la rama principal y subirlos al repositorio. Ver instrucciones en Instructivo GitLab y SourceTree diapositiva 3. Crear una rama de trabajo. Ver instrucciones en Instructivo GitLab y SourceTree diapositiva 4. Integrar la rama al proyecto principal. Descargar cambios de la rama principal a mi rama local. Investigar sobre resolución de conflictos.

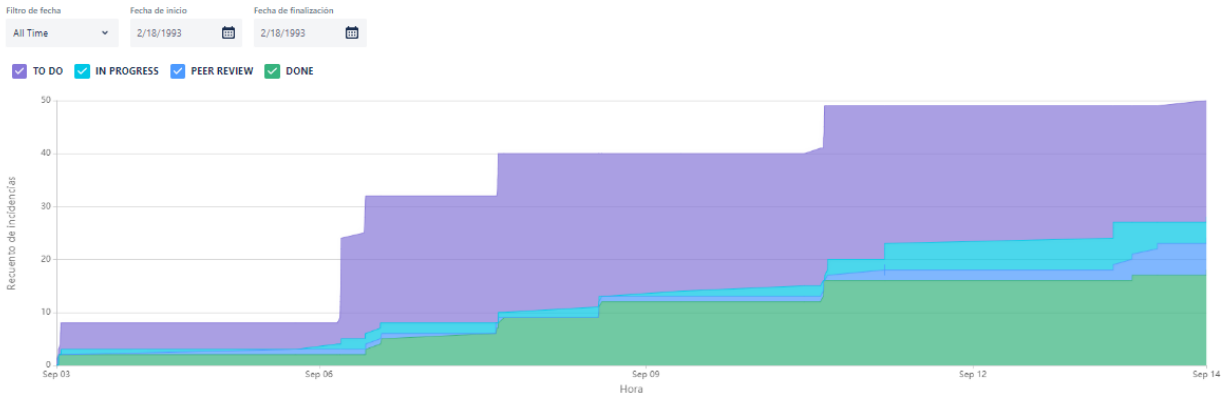


## Cumulative flow diagram

Proyectos / 533Grupo1 / Informes

Diagrama de flujo acumulado

[Cómo leer este informe](#)



## UI/UX Mockups

### Mockups - to edit

Mi unidad > PROYECTO MINTIC > Vistas de usuario

