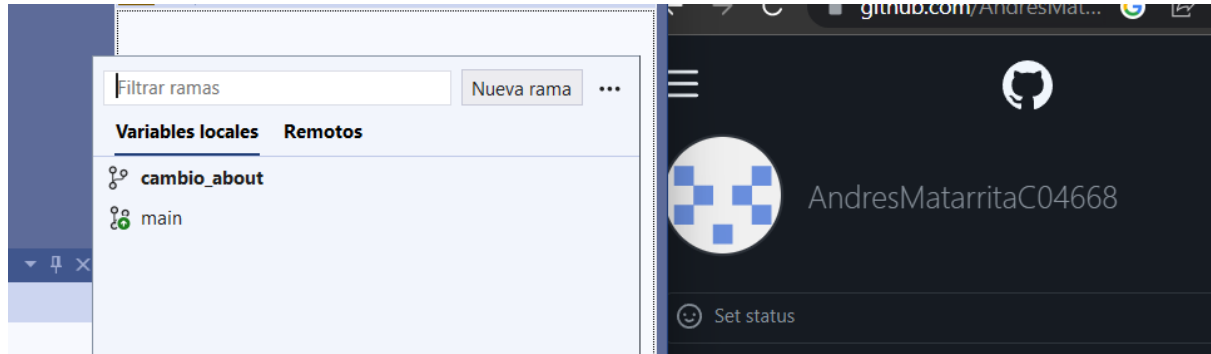
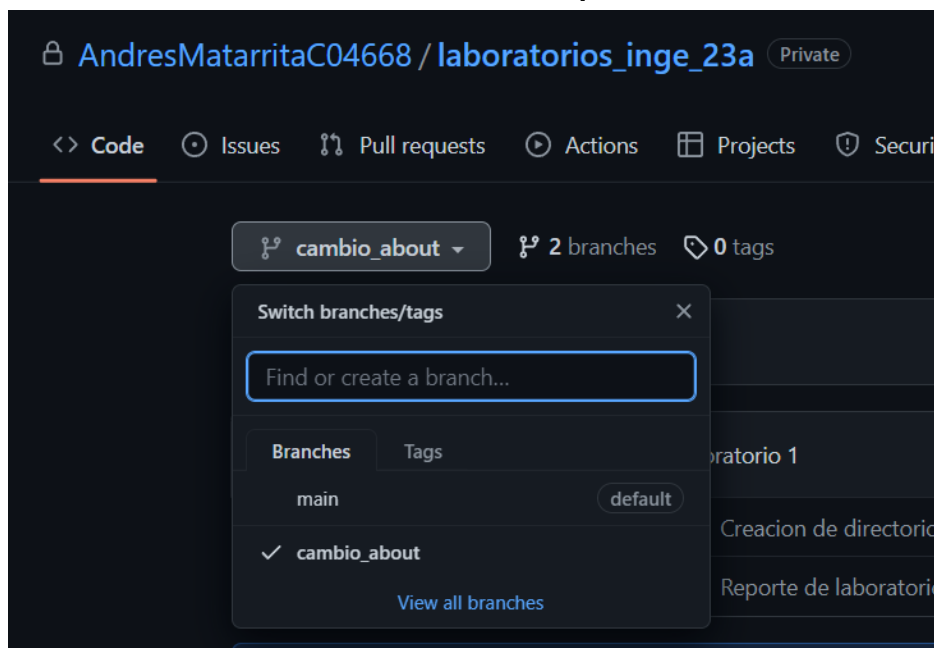


Alumno: José Andrés Matarrita Miranda
Carné: C04668
CI-0126 Ingeniería de Software
Prof. Dr. Allan Berrocal Rojas
Grupo 01 - I-2023

1. Creación de la nueva rama con Visual Studio

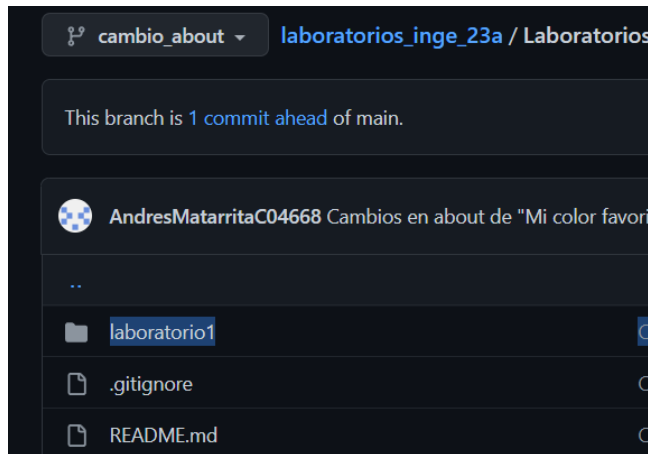


2. Visualización de la nueva rama en el repositorio en GitHub



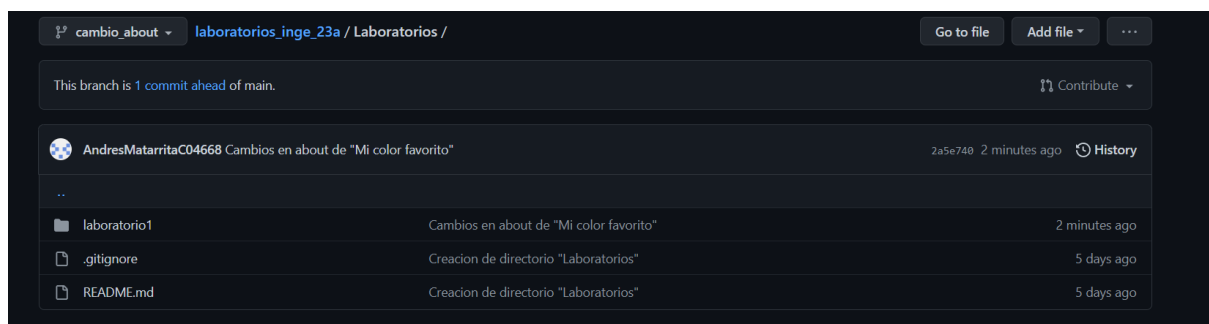
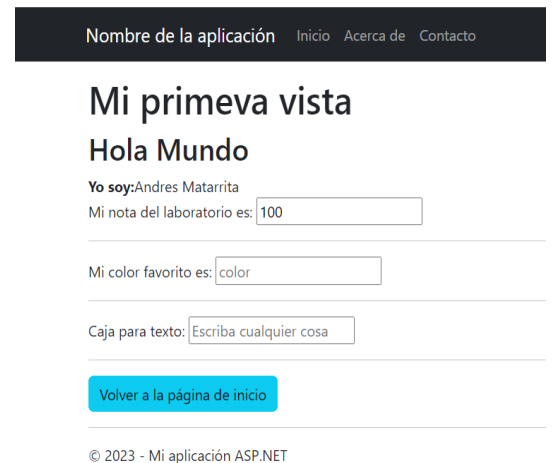
3. Actualización de la nueva rama con código fuente

Nota importante , hice el laboratorio 2 en el directorio del laboratorio 1 por equivocación , por tal razón el nombre del directorio es laboratorio 1 , no obstante , se crea un nuevo laboratorio llamado “laboratorio 2” , donde se movieron todos los archivos correspondientes al final de laboratorio.



4. Código agregado a la nueva rama del proyecto ASP.NET

Cambios en la rama cambio_about:



Cambios en la rama caja_en_about:

```
String nombreAutor = "Andres Matarrita";
int nota = 100;

<html>
<head>
<title>@ViewBag.Title</title>
</head>
<body>
<div>
<h1>Mi primera vista</h1>
<h2>Hola Mundo</h2>
<p>
<strong>Yo soy:</strong>@nombreAutor<br />
Mi nota del laboratorio es: <input type="number" value="@nota"
placeholder="Escriba su nota" />
</p>
<hr />
</div>
<div>
<label>Mi número de carné es:</label>
<input type="text" placeholder="Carné" />
<hr />
</div>
<div>
<label>Caja para texto:</label>
<input type="text" placeholder="Escriba cualquier cosa" />
<hr />
</div>
<div>
<a class="btn btn-info" href="/Home/Index">
Volver a la página de
inicio
</a><br />
</div>
</div>
```

Mi primera vista

Hola Mundo

Yo soy: Andres Matarrita

Mi nota del laboratorio es:

Mi número de carné es:

Caja para texto:

[Volver a la página de inicio](#)

© 2023 - Mi aplicación ASP.NET

caja_en_about ▾ laboratorios_inge_23a / Laboratorios /

Go to file Add file ...

This branch is 1 commit ahead of main. Contribute ▾

AndresMatarritaC04668 Cambio en about "Mi numero de carne es:" 88eec22 now History


laboratorio1	Cambio en about "Mi numero de carne es:"	now
.gitignore	Creacion de directorio "Laboratorios"	5 days ago
README.md	Creacion de directorio "Laboratorios"	5 days ago

5. Identificación del conflicto generado al hacer el merge

Repositorio G...rios_inge_23a About.cshtml

⚠ Fusión mediante combinación completada con conflictos en el repositorio laboratorios_inge_23a. Resuelva los conflictos y confirme los resultados.

Cuenta de personalización



Andres Matarrita
andresmm1919@hotmail.com

[Administrar perfil de Visual Studio](#)

[Opciones de la cuenta](#)

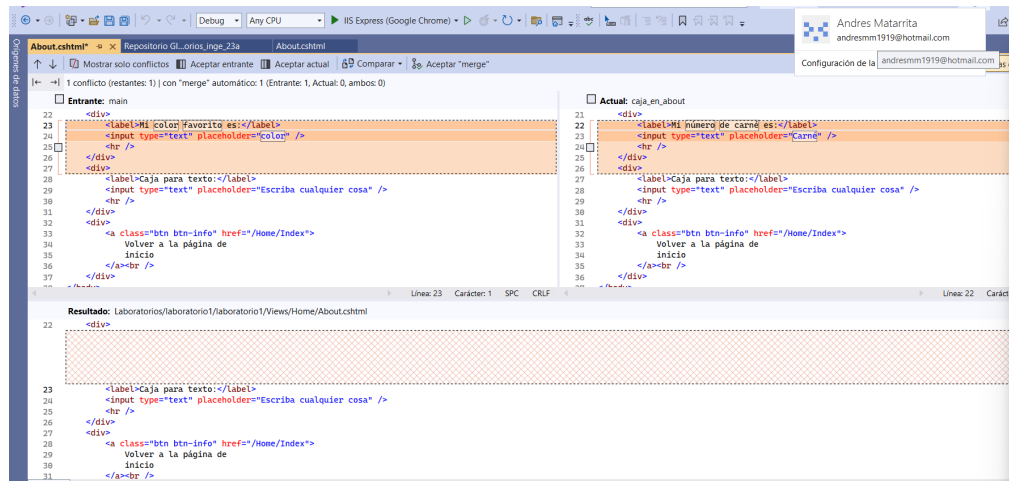
[Cerrar sesión](#)

Visual Studio

Community 2022

Licencia: Clave de producto aplicada

6. Evidencia de la resolución del conflicto



7. Página desplegada con la versión de código de la nueva rama

Mi primeva vista

Hola Mundo

Yo soy: Andres Matarrita

Mi nota del laboratorio es:

Mi número de carné es:

Mi color favorito es:

Caja para texto:

[Volver a la página de inicio](#)

© 2023 - Mi aplicación ASP.NET

8. Evidencia de la creación del pull request

The screenshot shows the GitHub web interface for a repository named 'AndresMatarritaC04668 / laboratorios_ing_23a'. The 'Pull requests' tab is active, displaying 'Open a pull request' with instructions to create a new pull request by comparing changes across two branches. A comparison is shown between 'base: main' and 'compare: caja_en_about', with a green checkmark indicating 'Able to merge'. Below this, the 'Caja en about' pull request is visible, with a 'Write' tab selected. A green banner at the bottom states 'This branch has no conflicts with the base branch' and includes a 'Merge pull request' button. A final notification at the bottom indicates 'Pull request successfully merged and closed', with a 'Delete branch' button.

9. Visualización de la actividad del proyecto en GitHub

The screenshot displays the 'Commits' section of the GitHub repository, filtered by date. The commits are listed in chronological order, showing the progression of the project. The most recent commit is a 'Merge pull request #2 from AndresMatarritaC04668/caja_en_about', followed by 'Resolucion de conflictos del merge', 'Merge pull request #1 from AndresMatarritaC04668/cambio_about', and several other commits related to 'Caja en about' and 'Cambios en about'. The commits are grouped by date, with the most recent group being 'Commits on Mar 22, 2023'.

Enlace de repositorio:

https://github.com/AndresMatarritaC04668/c04668_ci0126_23a.git

Resumen

Al realizar este laboratorio número 2, aprendí a resolver conflictos entre dos ramas que tratan de modificar la rama principal(main/master), anteriormente en los cursos de programación pasados, surgían problemas graves debido a el desconocimiento acerca de cómo resolver un conflicto de merge. Por otro lado, conocí por primera vez la existencia de un pull request , dicho proceso me pareció muy interesante e importante para evitar dañar la rama principal. Además, siempre es importante preguntar a los miembros del equipo si aprueban hacer un cambio a la rama principal.

Por otra parte, todas las funciones realizadas en Visual Studios fueron nuevas para mí debido al poco contacto que he tenido con Visual. Aprendí a unificar dos ramas directamente de un IDE, el proceso es delicado debido al gran problema que se puede causar si se comete un error al hacer merge de una rama secundaria con la rama principal. También adquirí conocimiento acerca de cómo crear ramas secundarias directamente de Visual Studio sin necesidad de recurrir manualmente al GitHub para crear ramas.

Es muy importante mencionar para la persona que revise este laboratorio, que tuve dificultades al realizar el laboratorio 2, debido a que inicié realizando los cambios y pasos correspondiente de este laboratorio 2 en el directorio del laboratorio 1, por lo que si observa que el directorio 1 posee los pasos y commits de este laboratorio es por dicha razón .No obstante , al terminar los pasos cree un directorio nuevo llamado “laboratorio2” donde están los archivos correspondientes junto con el reporte , para cumplir con las instrucciones de este laboratorio. Otro aspecto que resultó difícil fue hacer merge de una rama secundaria con la rama master, debido a que la versión que uso de Visual Studio tiene ligeras diferencias que la versión usada en los screenshots de las instrucciones del laboratorio.

Afortunadamente el proceso de generar un “pull request” se me resultó fácil debido a la sencillez de los pasos necesarios para crear dicha solicitud. Lo mismo sucedió con el proceso de crear nuevas ramas, nuevamente los pasos eran muy pocos y sencillos. Para este laboratorio me tomó alrededor de 4 horas y 30 minutos, pude haberlo hecho en menos tiempo, pero el error de no trabajar con un directorio llamado “laboratorio2” cometido al inicio del laboratorio. El tratar de mover los archivos a este nuevo directorio se me dificultó y al mismo tiempo cambiar el nombre del proyecto de Visual Studio de “laboratorio1” a “laboratorio2” me tomó más de una hora.

Investigue y agregue una pequeña explicación de los siguientes conceptos indicando al menos un caso en el cual es conveniente utilizarlo. Incluya las referencias o fuentes que utilizó para investigar :

1. Rebase and merge

De acuerdo con Sánchez A. (2015) tenemos que “rebase” es un comando que permite a los usuarios integrar cambios de una rama a otra, y los registros se modifican una vez que se completa la acción. El “rebase” se desarrolló para superar las deficiencias de la fusión, específicamente con respecto a los registros.

Por otro lado, según Sánchez A. (2015) “merge” es un comando que permite a los desarrolladores fusionar ramas de Git mientras los registros de confirmaciones en las ramas permanecen intactos. La redacción de fusionar puede ser confusa porque tenemos dos métodos para fusionar ramas, y una de esas formas se llama “merge”, aunque ambos procedimientos hacen esencialmente lo mismo. Git merge debería usarse para subir cambios y nuevas features a la rama principal y git rebase debería ser usado cuando se trata de integrar ramas secundarias.

2. Squash and merge

Según Gilroy, S. P., & Kaplan, B. A. (2019) tenemos que “squash y merge” se encargan de combinar los commits existentes en la rama Head para crear un solo commit para más tarde fusionar la rama Head con la rama Base. Es así, que el historial de commits de la rama Head queda unificado en un solo commit que se encontrará en la rama Base. Se podría usar en un caso en donde se necesite condensar el historial de Git de las ramas al terminar una solicitud de incorporar cambios.

3. Git reset

Es un comando poderoso para deshacer cambios locales en el estado de un repositorio de Git. El comando reset funciona en “los tres árboles de Git”. Estos árboles son el historial de ejecución (HEAD), el índice del entorno de ensayo y el directorio de trabajo. Se puede usar cuando quiero regresar mi estado actual de archivos a un estado específico anterior.

4. Git revert

Según Scillone G (2016) el comando git revert es un comando que se utiliza para eliminar de manera continua que posibilita una forma segura de deshacer los cambios en el historial de commits. En lugar de eliminar las confirmaciones o dejarlas solas en el historial de confirmaciones, la reversión creará una nueva confirmación que regresa los cambios especificados. Este comando se puede usar cuando hice un commit con errores y lo subí al repositorio es decir al historial de commits. Se puede usar este comando para deshacer ese commit del historial de cambios.

Bibliografía

Sánchez A. (2015). Introducción a GIT Sistema de control de versiones. En *Aprendeconalf.es*.

Gilroy, S. P., & Kaplan, B. A. (2019). Furthering open science in behavior analysis: An introduction and tutorial for using GitHub in research. *Perspectives on Behavior Science*, 42(3), 565–581.
<https://doi.org/10.1007/s40614-019-00202-5>

Scillone, G. (2016). *Control de Versiones - Git*.
http://cdr.ing.unlp.edu.ar/files/presentaciones/010_%20Introduccion%20a%20Git.pdf