

UNIVERSIDAD TECNOLÓGICA DE LA RIVIERA MAYA
TÉCNICO SUPERIOR UNIVERSITARIO EN DESARROLLO DE
SOFTWARE MULTIPLATAFORMA

“PROYECTO INTEGRADORA”

PRESENTA:

DIEGO BLANCO RODRÍGUEZ
MOISES DAVID CASTAÑEDA MAY
ERICK DANIEL VARGAS CORONA

2402027

2402046

2402037

GENERACIÓN 2024-2028

EMPRESA: DREAM TEAM INTERNATIONAL SYSTEM ENTERTAINMENT

ASESOR EMPRESARIAL:

Grado abreviado. Nombre Apellidos

Playa del Carmen, Quintana Roo

Septiembre

"Sprint 3 - Implementación de Control de Versiones"

1. Objetivo del Sprint

Desarrollar la capacidad de gestionar versiones y control de cambios en los diferentes artefactos del proyecto integrador "PrimerPaso" (base de datos, diseño UI/UX y desarrollo web). El objetivo es implementar un sistema de control de versiones robusto utilizando GitHub, aplicando los principios de la metodología SCRUM y un flujo de trabajo Git-flow para gestionar el avance incremental, el trabajo colaborativo y el registro de cambios.

2. Descripción de la Actividad

Para cumplir con los requerimientos de la actividad, el equipo configuró cuatro repositorios de control de versiones para reflejar el trabajo colaborativo en los componentes principales del proyecto:

1. **Repositorio de Base de Datos (PrimerPaso-db):** Este repositorio se configuró para contener el modelado lógico (diagrama E-R), el modelo físico (relacional) y los archivos SQL que componen la estructura de la base de datos implementada en el Sprint 1.
2. **Repositorio de Diseño (PrimerPaso-Frontend):** Dado que el prototipo de alta fidelidad se desarrolló directamente en código con Astro, este repositorio contiene dicho prototipo navegable, el cual representa la versión funcional del diseño UI/UX del Sprint 2.
3. **Repositorio de Desarrollo Web (PrimerPaso-Backend):** Se configuró un repositorio separado para aislar la lógica del servidor (API), las autenticaciones y las conexiones a la base de datos, separándolo del código del cliente (frontend).
4. **Repositorio de Diseño V2 (PrimerPaso-Design):** Este repositorio se configuró para la **migración del diseño UI/UX a un nuevo framework (Vue.js)**. Actualmente se encuentra en una fase "beta" y contiene el prototipo funcional en Vue.

El flujo de trabajo implicó un proceso avanzado de gestión de ramas. Inicialmente, el frontend y el backend coexistían en un solo repositorio (major-magnitude). El trabajo de este sprint consistió en **separar este monorepo** en dos repositorios distintos, limpiando la rama main para que sirva solo como documentación y migrando las ramas frontend y backend a sus propios repositorios (PrimerPaso-Frontend y PrimerPaso-Backend, respectivamente).

3. Entregables

A continuación, se presentan los entregables generados durante este sprint.

3.1. Enlaces a los Repositorios

- **Repositorio de Base de Datos:** <https://github.com/DiegoB-R/PrimerPaso-db.git>
- **Repositorio de Diseño (Frontend):** <https://github.com/DiegoB-R/PrimerPaso-Frontend.git>
- **Repositorio de Desarrollo (Backend):** <https://github.com/Eroroshy/PrimerPaso-Back.git>
- **Repositorio de Diseño V2 (Migración a Vue):** <https://github.com/DiegoB-R/PrimerPaso-Design.git>

3.2. Documento con Evidencia (Flujo de Trabajo)

Esta sección documenta la estructura de los repositorios, el flujo de trabajo, los commits descriptivos y la gestión de ramas. La evidencia gráfica se encuentra en los anexos de este sprint.

A. Repositorio de Base de Datos (Gestión Gráfica)

Para el repositorio de la base de datos (PrimerPaso-db), se utilizó la interfaz gráfica de GitHub para la carga inicial de los artefactos del Sprint 1.

Carga y Commit Inicial: Se prepararon los tres artefactos clave (script SQL, diagrama E-R y modelo relacional) y se subieron en un solo commit a la rama main para establecer la **Versión 1.0.0** de la base de datos. Se utilizó un título de commit descriptivo (feat(db): Carga inicial v1.0.0...) y una descripción detallada, estableciendo la línea base para el Sprint 1.

(Ver [Anexo A-Repositorio Base de Datos](#)).

Resultado del Repositorio: El repositorio final muestra los artefactos organizados en carpetas, el commit descriptivo y el archivo README.md documentando el proyecto, la versión y los últimos cambios, cumpliendo con los requisitos de la tarea. (Ver [Anexo A-Repositorio Base de Datos](#)).

B. Repositorio de Desarrollo (Gestión por Comandos y Ramas)

Para los repositorios de Frontend y Backend, se utilizó la terminal de comandos para un control de versiones más granular y avanzado, incluyendo la separación de un repositorio existente.

Paso 1: Limpieza de la Rama main El repositorio original contenía todo el código. El primer paso fue limpiar la rama main para que solo contuviera el archivo README.md de documentación. Se usó git rm para eliminar todos los archivos de código (.gitignore, .vscode, login.js, etc.) y se realizó un commit para registrar esta limpieza. (Ver [Anexo B- Repositorios de Diseño \(Frontend\) y Desarrollo Web \(Backend\)](#)).

Paso 2: Aislamiento de Ramas (frontend y backend) Se crearon y limpian las ramas específicas. Para la rama backend, se eliminaron todos los archivos pertenecientes al frontend (CSS, JS públicos, layouts de Astro, etc.) para aislar únicamente los archivos de la API (auth/login.js, auth/signup.js). (Ver [Anexo B- Repositorios de Diseño \(Frontend\) y Desarrollo Web \(Backend\)](#)).

Paso 3: Migración a Nuevos Repositorios (Push a Múltiples Remotes) Este fue el paso crucial. Se configuraron nuevos *remotes* (destinos de subida) para los repositorios de Frontend y Backend. Se actualizó el README.md de la rama backend y se subió (push) tanto al repositorio origin (ahora PrimerPaso-Frontend) como al nuevo remote eroroshy (PrimerPaso-Back). Esta acción demuestra la separación final del código. (Ver [Anexo B- Repositorios de Diseño \(Frontend\) y Desarrollo Web \(Backend\).](#)).

C. Repositorio de Diseño V2 (Gestión por Comandos - Migración a Vue.js)

Para el nuevo **Repositorio de Diseño**, se utilizó la línea de comandos para inicializar y subir el proyecto de migración a Vue.js.

Paso 1: Inicialización y Autenticación El miembro del equipo (Eroroshy) se autenticó usando la CLI de GitHub (gh auth login) e inicializó un nuevo repositorio local (git init). (Ver [Anexo C- Repositorio Diseño V2 \(VUE\).](#)).

Paso 2: Creación de Rama y Primer Commit Se creó una rama aislada (git checkout -b vuemigration) para no trabajar en main. Se añadieron todos los archivos del proyecto Vue.js (git add .) y se realizó el primer commit descriptivo (Commit Inicial, Conexión con Repo compartido). (Ver [Anexo C- Repositorio Diseño V2 \(VUE\).](#)).

Paso 3: Conexión y Subida al Repositorio Remoto El primer intento de subida (push) falló al no tener un "remote" configurado. Se añadió el repositorio PrimerPaso-Design como el origin (git remote add origin ...) y se subió exitosamente la rama vuemigration al repositorio de GitHub. (Ver [Anexo C- Repositorio Diseño V2 \(VUE\).](#)).

D. Archivos README.md y Documentación

Se crearon archivos README.md para cada repositorio, detallando su propósito, versión y fecha de actualización, como lo solicita la tarea.

- **README de Base de Datos:** Detalla el contenido del repositorio (SQL, diagramas) y la versión actual. (Ver [Anexo A-Repositorio Base de Datos](#)).

- **README de Frontend (Astro):** Se actualizó el README de la rama frontend para indicar que este repositorio contiene la aplicación de frontend construida con Astro. (Ver [Anexo B- Repositorios de Diseño \(Frontend\) y Desarrollo Web \(Backend\)](#)).
- **Commit de Documentación:** Se realizaron commits específicos solo para actualizar la documentación, como la actualización del README.md en la rama main para explicar la nueva estrategia de ramas (frontend y backend).

3.4. Reflexión del Sprint

La implementación del control de versiones ha sido un sprint técnico fundamental que, aunque no añade funcionalidades visibles al usuario como los Sprints 1 y 2, construye la columna vertebral para el trabajo colaborativo y la sostenibilidad del proyecto.

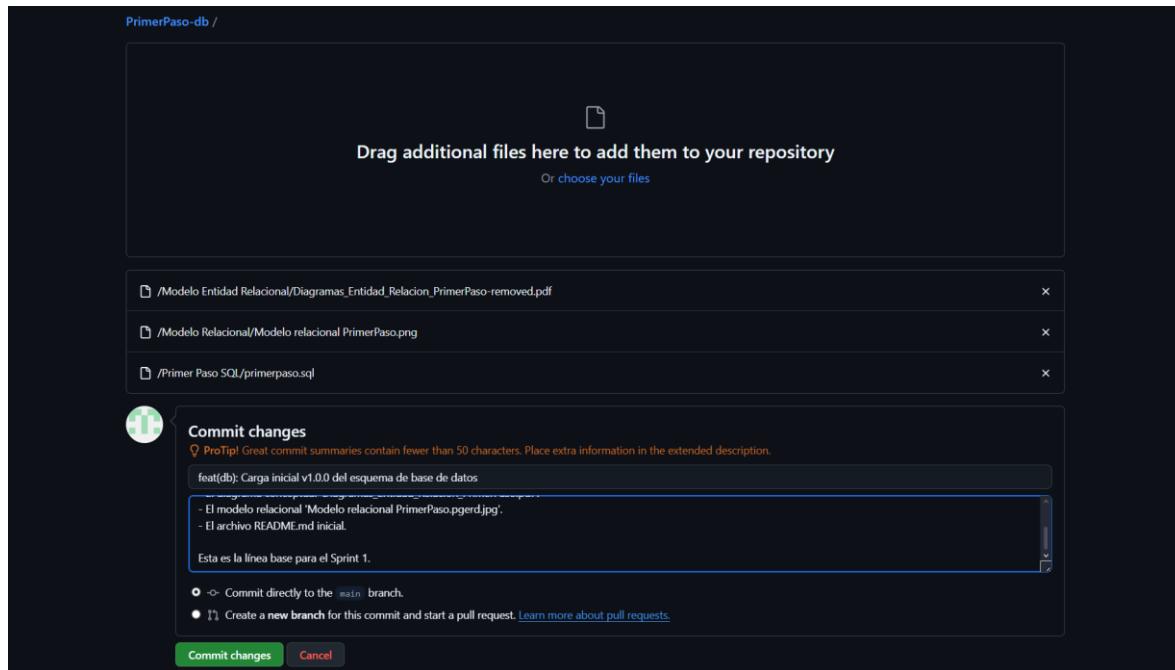
El aprendizaje más significativo fue la diferencia entre la gestión gráfica de repositorios (usada para la BD) y la gestión avanzada por la línea de comandos. La separación de un monorepo en dos repositorios (Frontend y Backend) fue un desafío técnico que requirió un uso preciso de los comandos de Git, especialmente en la gestión de *remotes* y la limpieza de ramas.

Aplicando SCRUM, este sprint establece el "Definition of Done" (Definición de Hecho) para todo el desarrollo futuro: ninguna característica se considerará "completa" si no está registrada en un commit descriptivo y fusionada a través de un *Pull Request* en su repositorio correspondiente. El uso de ramas separadas (feature/login, fix/db-connection, etc.) será ahora la norma, permitiendo a los desarrolladores (Diego, Erick y Moises) trabajar en paralelo sin conflictos.

Anexos 3.1

Anexo A-Repositorio Base de Datos

Creación de Repositorio en GitHub



Repositorio – PrimerPaso-db

The screenshot shows the GitHub repository page for 'PrimerPaso-db'. The repository is public and has 1 branch and 0 tags. The main commit is by 'DiegoB-R' titled 'feat(db): Carga inicial v1.0.0 del esquema de base de datos' (1 minute ago). The repository contains files: 'Modelo Entidad Relacional', 'Modelo Relacional', 'Primer Paso SQL', and 'README.md'. The 'README' tab is active, displaying the content: 'Proyecto Integrador: PrimerPaso (Base de Datos)'. It describes the repository as containing scripts for PostgreSQL database creation and conceptual entity-relationship diagrams. The 'About' section includes links to 'Readme', 'Activity', 'Stars' (0), 'Forks' (0), and 'Code'. The 'Releases' section indicates no releases have been published. The 'Packages' section also indicates no packages have been published.

README Repositorio

The screenshot shows a GitHub repository interface. The top navigation bar has 'README' selected. The main content area features a large title 'Proyecto Integrador: PrimerPaso (Base de Datos)' in bold. Below it is a descriptive paragraph: 'Repositorio del modelado y scripts de la base de datos para el proyecto PrimerPaso, una plataforma de conexión entre estudiantes y empresas.' A section titled 'Descripción' follows, containing a bullet point about the repository's contents. Another section, 'Versión Actual', lists the version, update date, and recent changes.

Proyecto Integrador: PrimerPaso (Base de Datos)

Repositorio del modelado y scripts de la base de datos para el proyecto PrimerPaso, una plataforma de conexión entre estudiantes y empresas.

Descripción

Este repositorio contiene:

- El script SQL para la creación de la base de datos en PostgreSQL.
- El diagrama Entidad-Relación (conceptual) del proyecto.
- El modelo relacional (físico) exportado desde pgAdmin.

Versión Actual

- **Versión:** 1.0.0
- **Fecha de Actualización:** 06/11/2025
- **Últimos cambios:**
 - Corrección de llaves foráneas.
 - Ajuste del tipo de dato `telephone_student` a `varchar(20)`.
 - Definición de todas las relaciones (`ALTER TABLE`).

Anexo B- Repositorios de Diseño (Frontend) y Desarrollo Web (Backend).

Creación de Rama - frontend

```
PS C:\Users\bymaz\Downloads\astro 2> cd "C:\Users\bymaz\Downloads\astro 2\major-magnitude"; git status --porcelain; git rev-parse --abbrev-ref HEAD
main
PS C:\Users\bymaz\Downloads\astro 2\major-magnitude> cd "C:\Users\bymaz\Downloads\astro 2\major-magnitude"; git checkout -b frontend; git push -u origin frontend
Switched to a new branch 'frontend'
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'frontend' on GitHub by visiting:
remote:     https://github.com/DiegoB-R/major-magnitude/pull/new/frontend
remote:
To https://github.com/DiegoB-R/major-magnitude.git
 * [new branch]      frontend -> frontend
branch 'frontend' set up to track 'origin/frontend'.
```

Verificación de Rama - main

```
PS C:\Users\bymaz\Downloads\astro 2\major-magnitude> cd "C:\Users\bymaz\Downloads\astro 2\major-magnitude"; git checkout main; git branch --show-current; git ls-files --st
age | sed -n '1,10p'
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
main
```

Validación de archivos en carpeta

```
PS C:\Users\bymaz\Downloads\astro 2\major-magnitude> cd "C:\Users\bymaz\Downloads\astro 2\major-magnitude"; git ls-files --stage
100644 e0e0ef4915aaaf45a90a757c11ad97a27440b68da 0          .gitignore
100644 22a15055d638988eb33bb6153d27f623df92e4a 0        .vscode/extensions.json
100644 d6422097621fd7c1b1ccc6daa670c46aed7ef5b7 0       .vscode/launch.json
100644 414a13a6195fc0ef0ac602f72b3b167333e9750 0      README.md
100644 8dd5b3fb379095f8e4733a85ae93a0d1e0c44955 0    astro.config.mjs
100644 e69de29bb2d1d6434b8b29ae775ad8c2e48c5391 0   login.js
100644 e69de29bb2d1d6434b8b29ae775ad8c2e48c5391 0   login.ts
100644 6d9151829ed3ec95ddbefbc5c77961583f9cbd5a 0  package-lock.json
100644 d94ff0adeb81be44e37786a8dac3ff281b54f803 0  package.json
100644 d198216c5a819a4dbbf256471df973b6beff302 0  public/css/styles.css
100644 f157bd1c5e287c70a508a98a13f538491aa4dafc 0  public/favicon.svg
100644 72574428e4bb77b8d9a5d6f953ac09024eaef772 0  public/js/app.js
100644 c928a992b37f181508562ecac9b93a8c6fd1230a 0  public/js/company-jobs.js
100644 4b5af1ae4bb8747d9b2506e4ac94465ed3894539a 0  public/js/jobs.js
100644 e69de29bb2d1d6434b8b29ae775ad8c2e48c5391 0  public/js/signup.ts
100644 a1e4855860a730d99232156f8832912efe1f49af 0  public/js/user-menu.js
100644 40e04969cdcb1f3c34e52338fc9215701ea8adcc 0  public/login/company-login.js
100644 8f5e6beb6e138622749b61fbe4177128464832a48 0  public/login/login.js
100644 666ec745a6565c05f501b2741f78c137b87c7a54 0  public/profile/perfiluser.js
100644 eb2a9d26e721dc6a3d0095b32e5c0e7203f4328e 0  public/signup/company-signup.js
100644 d3d5123a2f9ca3a28f94a4b22f0f71761c41231a 0  public/signup/signup.js
100644 8cb68a1e85d81f395cbc420cd9dc1826e530a2e9 0  scripts/check-db.cjs
```

Creación de Rama - Backend

```
PS C:\Users\bymaz\Downloads\astro 2\major-magnitude> cd "C:\Users\bymaz\Downloads\astro 2\major-magnitude"; git checkout -b backend
Switched to a new branch 'backend'
PS C:\Users\bymaz\Downloads\astro 2\major-magnitude> cd "C:\Users\bymaz\Downloads\astro 2\major-magnitude"; git ls-files | Select-String "src/pages" -NotMatch -SimpleMatch; git
ls-files | Select-String "src/pages" -NotMatch -SimpleMatch | Select-Object -First 5
src/pages/api/auth/login.js
src/pages/api/auth/signup.js
.gitignore
.vscode/extensions.json
.vscode/launch.json
README.md
astro.config.mjs
```

Se eliminan subcarpetas

```
PS C:\Users\bymaz\Downloads\astro 2\major-magnitude> cd "C:\Users\bymaz\Downloads\astro 2\major-magnitude"; git rm -r public src/components src/layouts src/assets src/page
s -ignore-unmatch
rm 'public/css/styles.css'
rm 'public/favicon.svg'
rm 'public/js/app.js'
rm 'public/js/company-jobs.js'
rm 'public/js/jobs.js'
rm 'public/js/signup.ts'
rm 'public/js/user-menu.js'
rm 'public/login/company-login.js'
rm 'public/login/login.js'
rm 'public/profile/perfiluser.js'
rm 'public/signup/company-signup.js'
rm 'public/signup/signup.js'
rm 'src/assets/astro.svg'
rm 'src/assets/background.svg'
rm 'src/components/Welcome.astro'
rm 'src/layouts/BaseLayout.astro'
rm 'src/layouts/Layout.astro'
rm 'src/layouts/user-menu.js'
rm 'src/pages/api/auth/login.js'
rm 'src/pages/api/auth/signup.js'
rm 'src/pages/company-dashboard.astro'
```

Se mantiene únicamente la carpeta api en la rama del Backend

```
PS C:\Users\bymaz\Downloads\astro 2\major-magnitude> cd "C:\Users\bymaz\Downloads\astro 2\major-magnitude"; git status --porcelain; git add src/pages/api -f; git commit -m
"Isolate backend APIs: keep only src/pages/api in backend branch"
D public/css/styles.css
D public/favicon.svg
D public/js/app.js
D public/js/company-jobs.js
D public/js/jobs.js
D public/js/signup.ts
D public/js/user-menu.js
D public/login/company-login.js
D public/login/login.js
D public/profile/perfiluser.js
D public/signup/company-signup.js
D public/signup/signup.js
D src/assets/astro.svg
D src/assets/background.svg
D src/components/Welcome.astro
D src/layouts/BaseLayout.astro
D src/layouts/Layout.astro
D src/layouts/user-menu.js
```

Se hace un commit y sube la Rama Backend a Github

```
PS C:\Users\bymaz\Downloads\astro 2\major-magnitude> cd "C:\Users\bymaz\Downloads\astro 2\major-magnitude"; git checkout frontend -- src/pages/api; git status --porcelain
git add src/pages/api -f; git commit -m "Restore API files into backend branch"; git push -u origin backend
A src/pages/api/auth/login.js
A src/pages/api/auth/signup.js
[backend d20ba21] Restore API files into backend branch
2 files changed, 192 insertions(+)
create mode 100644 src/pages/api/auth/login.js
create mode 100644 src/pages/api/auth/signup.js
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 12 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (7/7), 733 bytes | 733.00 KiB/s, done.
Total 7 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 1 local object.
remote:
remote: Create a pull request for 'backend' on GitHub by visiting:
remote:   https://github.com/DiegoB-R/major-magnitude/pull/new/backend
remote:
To https://github.com/DiegoB-R/major-magnitude.git
 * [new branch]      backend -> backend
branch 'backend' set up to track 'origin/backend'.
```

Cambio a Rama Main, se eliminan carpetas no deseadas, se hace commit y se sube a Github

```
PS C:\Users\bymaz\Downloads\astro 2\major-magnitude> cd "C:\Users\bymaz\Downloads\astro 2\major-magnitude"; git checkout main; git rm -r public src scripts package.json package-lock.json astro.config.mjs tsconfig.json --ignore-unmatch; git commit -m "Clean main: remove frontend/backend source; keep docs only"; git push origin main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
rm 'astro.config.mjs'
rm 'package-lock.json'
rm 'package.json'
rm 'public/css/styles.css'
rm 'public/favicon.svg'
rm 'public/js/app.js'
rm 'public/js/company-jobs.js'
rm 'public/js/jobs.js'
rm 'public/js/signup.ts'
rm 'public/js/user-menu.js'
rm 'public/login/company-login.js'
rm 'public/login/login.js'
rm 'public/profile/perfiluser.js'
rm 'public/signup/company-signup.js'
rm 'public/signup/signup.js'
rm 'scripts/check-db.cjs'
rm 'scripts/check-db.js'
rm 'scripts/check-new-tables.cjs'
```

Se agrega el archivo README.md a Github

```
PS C:\Users\bymaz\Downloads\astro 2\major-magnitude> cd "C:\Users\bymaz\Downloads\astro 2\major-magnitude"; git add README.md; git commit -m "Docs: add branch strategy and SCRUM notes to README"; git push origin main
warning: in the working copy of 'README.md', LF will be replaced by CRLF the next time Git touches it
[main d5de50e] Docs: add branch strategy and SCRUM notes to README
 1 file changed, 21 insertions(+)
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 1.12 KiB | 1.12 MiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/DiegoB-R/major-magnitude.git
  e5c9aac..d5de50e main -> main
```

Conexión a Repositorio de GitHub compartido

```
PS C:\Users\bymaz\Downloads\astro 2\major-magnitude> cd "C:\Users\bymaz\Downloads\astro 2\major-magnitude"; git remote remove eroroshy >& null; git remote add eroroshy https://github.com/Eroroshy/PrimerPaso-Back.git; git remote -v; git push eroroshy backend:initial-backend
eroroshy      https://github.com/Eroroshy/PrimerPaso-Back.git (push)
origin        https://github.com/DiegoB-R/major-magnitude.git (fetch)
origin        https://github.com/DiegoB-R/major-magnitude.git (push)
Enumerating objects: 114, done.
Counting objects: 100% (114/114), done.
Delta compression using up to 12 threads
Compressing objects: 100% (105/105), done.
Writing objects: 100% (114/114), 101.28 KiB | 2.41 MiB/s, done.
Total 114 (delta 40), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (40/40), done.
remote:
remote: Create a pull request for 'initial-backend' on GitHub by visiting:
remote:     https://github.com/Eroroshy/PrimerPaso-Back/pull/new/initial-backend
remote:
To https://github.com/Eroroshy/PrimerPaso-Back.git
 * [new branch]      backend -> initial-backend
```

Cambios finales en carpetas y Push a GitHub compartido para Frontend

```
PS C:\Users\bymaz\Downloads\astro 2\major-magnitude> cd "C:\Users\bymaz\Downloads\astro 2\major-magnitude"; git ls-files | Where-Object { $_ -ne 'README.md' } | ForEach-Object { Write-Output "rm -r $_"; git rm --ignore-unmatch "$_" }; git status --porcelain; if ((git status --porcelain) -ne '') { git commit -m "Cleanup main: keep only README.md"; git push origin main } else { Write-Output 'No changes to commit' }
rm -r .gitignore
rm -r .vscode/extensions.json
rm -r .vscode/extensions.json'
rm -r .vscode/launch.json
rm -r .vscode/launch.json'
rm -r login.js
rm -r 'login.js'
rm -r login.ts
PS C:\Users\bymaz\Downloads\astro 2\major-magnitude> cd "C:\Users\bymaz\Downloads\astro 2\major-magnitude"; echo '--- tracked files (after) ---'; git ls-files;
--- tracked files (after) ---
README.md
PS C:\Users\bymaz\Downloads\astro 2\major-magnitude> cd "C:\Users\bymaz\Downloads\astro 2\major-magnitude"; git add README.md; git commit -m "Docs(main): README en español indicando frontend en rama 'frontend' 2025-11-06"; git push origin main
warning: in the working copy of 'README.md', LF will be replaced by CRLF the next time Git touches it
[main b4a537d] Docs(main): README en español indicando frontend en rama 'frontend' 2025-11-06
1 file changed, 10 insertions(+), 10 deletions(-)
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 941 bytes | 941.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/DiegoB-R/major-magnitude.git
  380dfb..b4a537d main -> main
```

Cambios finales en carpetas y Push a GitHub compartido para Backend

```
PS C:\Users\bymaz\Downloads\astro 2\major-magnitude> cd "C:\Users\bymaz\Downloads\astro 2\major-magnitude"; Move-Item -Path .gitignore -Destination .gitignore.local -Force
; git checkout backend
Switched to branch 'backend'
Your branch is up to date with 'origin/backend'.
PS C:\Users\bymaz\Downloads\astro 2\major-magnitude> cd "C:\Users\bymaz\Downloads\astro 2\major-magnitude"; git add README.md; git commit -m "Docs(initial en español - backend v0.1.0 (06-11-2025))"; git push origin backend; git push eroroshy backend:initial-backend
[backend 60860a7] Docs(initial en español - backend v0.1.0 (06-11-2025))
1 file changed, 23 insertions(+)
Enumerating objects: 14, done.
Counting objects: 100% (14/14), done.
Delta compression using up to 12 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (10/10), 1.63 KiB | 334.00 KiB/s, done.
Total 10 (delta 4), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (4/4), completed with 2 local objects.
remote: This repository moved. Please use the new location:
remote:   https://github.com/DiegoB-R/PrimerPaso-Frontend.git
remote:
remote: Create a pull request for 'backend' on GitHub by visiting:
remote:   https://github.com/DiegoB-R/PrimerPaso-Frontend/pull/new/backend
remote:
To https://github.com/DiegoB-R/major-magnitude.git
 * [new branch]      backend -> backend
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 941 bytes | 470.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/Eroroshy/PrimerPaso-Back.git
 d20ba21..60860a7 backend -> initial-backend
```

README Repositorio (Frontend).

□ README

PrimerPaso — Frontend (Astro)

Este repositorio actúa como contenedor del frontend del proyecto PrimerPaso. El código fuente del frontend se mantiene en la rama `frontend`.

Estado actual (06-11-2025):

- El frontend está implementado con Astro y se encuentra en la rama `frontend` de este repositorio.
- Actualmente se están realizando modificaciones relacionadas con la migración de frameworks; por eso la rama `main` no contiene el código fuente.

Instrucciones rápidas

- Para trabajar con el frontend, cambia a la rama `frontend` :

```
git fetch origin
git checkout frontend
```

- La rama `main` contiene únicamente documentación y notas del proyecto. El backend se mantiene en un repositorio separado.

Fecha: 2025-11-06

Anexo C- Repositorio Diseño V2 (VUE).

Inicio de Sesión en Github en Visual Studio Code, Inicialización de Repositorio y Estado

```
PS C:\Users\eroro\Documents\TOKATECH\scorchingray> gh auth login
? Where do you use GitHub? GitHub.com
? What is your preferred protocol for Git operations on this host? HTTPS
? Authenticate Git with your GitHub credentials? Yes
? How would you like to authenticate GitHub CLI? Login with a web browser

! First copy your one-time code: 4C1B-CFDF
Press Enter to open https://github.com/login/device in your browser...
✓ Authentication complete.
- gh config set -h github.com git_protocol https
✓ Configured git protocol
✓ Logged in as Eroroshy
PS C:\Users\eroro\Documents\TOKATECH\scorchingray> git init
Initialized empty Git repository in C:/Users/eroro/Documents/TOKATECH/scorchingray/.git/
PS C:\Users\eroro\Documents\TOKATECH\scorchingray> git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    .gitignore
    .vscode/
    README.md
    env.d.ts
    index.html
    package-lock.json
    package.json
    public/
```

Creación de Rama (vuemigration) y adición de archivos a ésta

```
PS C:\Users\eroro\Documents\TOKATECH\scorchingray> git checkout -b vuemigration
Switched to a new branch 'vuemigration'
PS C:\Users\eroro\Documents\TOKATECH\scorchingray> git add .
warning: in the working copy of '.gitignore', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of '.vscode/extensions.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'README.md', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'env.d.ts', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'index.html', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'package-lock.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'package.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'src/App.vue', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'src/main.ts', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'src/router/index.ts', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'tsconfig.app.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'tsconfig.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'tsconfig.node.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'vite.config.ts', LF will be replaced by CRLF the next time Git touches it
PS C:\Users\eroro\Documents\TOKATECH\scorchingray> git status
On branch vuemigration
```

Validación de archivos por incluirse en el Commit.

```
Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   .gitignore
    new file:   .vscode/extensions.json
    new file:   README.md
    new file:   env.d.ts
    new file:   index.html
    new file:   package-lock.json
    new file:   package.json
    new file:   public/favicon.ico
    new file:   src/App.vue
    new file:   src/assets/styles/Header.css
    new file:   src/assets/styles/Landing.css
    new file:   src/components/CButton.vue
    new file:   src/components/Header.vue
    new file:   src/components/Landing.vue
    new file:   src/main.ts
    new file:   src/router/index.ts
    new file:   src/styles.css
    new file:   tsconfig.app.json
    new file:   tsconfig.json
    new file:   tsconfig.node.json
    new file:   vite.config.ts
```

Commit Inicial

```
PS C:\Users\eroro\Documents\TOKATECH\scorchingray> git commit -m "Commit Inicial, Conexión con Repo compartido"
[vuemigration (root-commit) c77556d] Commit Inicial, Conexión con Repo compartido
 21 files changed, 3563 insertions(+)
 create mode 100644 .gitignore
 create mode 100644 .vscode/extensions.json
 create mode 100644 README.md
 create mode 100644 env.d.ts
 create mode 100644 index.html
 create mode 100644 package-lock.json
 create mode 100644 package.json
 create mode 100644 public/favicon.ico
 create mode 100644 src/App.vue
 create mode 100644 src/assets/styles/Header.css
 create mode 100644 src/assets/styles/Landing.css
 create mode 100644 src/components/CButton.vue
 create mode 100644 src/components/Header.vue
 create mode 100644 src/components/Landing.vue
 create mode 100644 src/main.ts
 create mode 100644 src/router/index.ts
 create mode 100644 src/styles.css
 create mode 100644 tsconfig.app.json
 create mode 100644 tsconfig.json
 create mode 100644 tsconfig.node.json
 create mode 100644 vite.config.ts
```

Intento de Push a GitHub, conexión remota a repositorio compartido y subida exitosa.

```
PS C:\Users\eroro\Documents\TOKATECH\scorchingray> git push -u origin vuemigration
fatal: 'origin' does not appear to be a git repository
fatal: Could not read from remote repository.

Please make sure you have the correct access rights
and the repository exists.
PS C:\Users\eroro\Documents\TOKATECH\scorchingray> git remote -v
PS C:\Users\eroro\Documents\TOKATECH\scorchingray> git remote add origin https://github.com/DiegoB-R/PrimerPaso-Design
PS C:\Users\eroro\Documents\TOKATECH\scorchingray> git remote -v
origin https://github.com/DiegoB-R/PrimerPaso-Design (fetch)
origin https://github.com/DiegoB-R/PrimerPaso-Design (push)
PS C:\Users\eroro\Documents\TOKATECH\scorchingray> git push -u origin vuemigration
Enumerating objects: 30, done.
Counting objects: 100% (30/30), done.
Delta compression using up to 4 threads.
Compressing objects: 100% (24/24), done.
Writing objects: 100% (30/30), 32.06 Kib | 1.69 MiB/s, done.
Total 30 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'vuemigration' on GitHub by visiting:
remote:   https://github.com/DiegoB-R/PrimerPaso-Design/pull/new/vuemigration
remote:
To https://github.com/DiegoB-R/PrimerPaso-Design
 * [new branch]      vuemigration -> vuemigration
branch 'vuemigration' set up to track 'origin/vuemigration'.
```

D-Formato: Acta de Reunión

1. Datos Generales

- **Proyecto:** PrimerPaso.com
- **Fecha:** 06 de noviembre del 2025
- **Hora de inicio:** 11:10 a.m **Hora de término:** 12:00 p.m
- **Lugar / Medio (Presencial / Virtual):** Presencial
- **Tipo de reunión (Planificación / Seguimiento / Revisión / Cierre):** Revisión (Sprint 3) y Planificación (Sprint 4)
- **Responsable de la reunión:** Diego Blanco Rodriguez
- **Elaboró el acta:** Diego Blanco Rodriguez

2. Participantes

Nombre	Cargo / Rol	Asistencia (Sí/No)
Diego Blanco Rodriguez	Scrum Master/ Desarrollador	Sí
Erick Corona Vargas	Product Owner/Desarrollador	Sí
Moises David Castañeda May	Desarrollador	Sí

Exportar a Hojas de cálculo

3. Orden del Día

N.º	Tema / Punto a Tratar
1	Revisión y cierre de entregables del Sprint 3: "Implementación de Control de Versiones".
2	Verificación de la configuración de los 3 repositorios (BD, Frontend/Design, Backend).
3	Discusión de la migración a Vue.js (Repositorio PrimerPaso-Design).
4	Definición de objetivos para el Sprint 4: "Conexión de Backend (API) y Funcionalidad de Autenticación".
5	Asignación de tareas e historias de usuario para el Sprint 4 (HU-01, HU-02, HU-03).
6	Planificación de la conexión entre el backend y la base de datos PostgreSQL.

4. Desarrollo de la Reunión.

Durante la sesión se revisaron los resultados del Sprint 3, confirmando la exitosa creación y configuración de los tres repositorios en GitHub. Se validó la separación del monorepo en PrimerPaso-Frontend y PrimerPaso-Backend, así como la creación del repositorio PrimerPaso-db y el nuevo PrimerPaso-Design para la migración a Vue.js.

Con el control de versiones ya implementado, se establecieron los objetivos del Sprint 4, el cual será un sprint técnico enfocado en **desarrollar el backend (API) y conectar** todos los componentes (Frontend, Backend y Base de Datos) para lograr la funcionalidad de autenticación (Registro e Inicio de Sesión), cubriendo las historias de usuario HU-01, HU-02 y HU-03.

5. Acuerdos y Compromisos.

N.º	Acuerdo / Compromiso	Responsable	Fecha Compromiso	Estado
1	Cierre y documentación del Sprint 3 (Control de Versiones).	Scrum Master	06/11/2025	Completado
2	Desarrollar los <i>endpoints</i> de la API (Backend) para Registro de Estudiante (HU-01) y Empleador (HU-02).	Product Owner	10/11/2025	Completado
3	Desarrollar el <i>endpoint</i> de la API (Backend) para Inicio de Sesión (HU-03).	Product Owner	10/11/2025	Completado
4	Establecer la conexión entre el Backend (API) y la Base de Datos (PostgreSQL) del Sprint 1.	Scrum Master	08/11/2025	Completado
5	Conectar el Frontend (Login y Registro) con los <i>endpoints</i> de la API.	Desarrollador	12/11/2025	Completado
6	Continuar con la migración del prototipo de Astro a Vue.js (Repo PrimerPaso-Design).	Desarrollador	12/11/2025	Completado

N.º	Acuerdo / Compromiso	Responsable	Fecha Compromiso	Estado
7	Documentar los <i>endpoints</i> de la API en el README.md del repositorio Backend.	Product Owner	13/11/2025	Completado
8	Planificación del Sprint 5 (Módulos de Perfiles y Ofertas).	Scrum Master y Product Owner	13/11/2025	Completado

6. Observaciones Generales

Se define que el Sprint 4 es crítico, ya que unirá por primera vez los tres componentes desarrollados en los sprints anteriores (BD, UI y Git). Se da prioridad a la funcionalidad de autenticación para poder construir los perfiles (Estudiante y Empresa) en el siguiente sprint. La migración a Vue.js (PrimerPaso-Design) continuará en paralelo.

Actualización (Post-Sprint 3): El Sprint 3 ("Implementación de Control de Versiones") se completó con éxito. Los tres repositorios principales (Base de Datos, Frontend, Backend) y el repositorio de migración (Diseño) están configurados, documentados y listos para el desarrollo colaborativo.

7. Cierre de la Reunión

- **Hora de cierre:** 12:00 p.m
- **Próxima reunión:** 13 de noviembre del 2025 11:10 a.m (Fecha y hora)

8. Firmas de Conformidad

- **Scrum Master:** Diego Blanco Rodriguez **Firma:** _____
- **Product Owner:** Erick Corona Vargas **Firma:** _____
- **Representante del Equipo:** Diego Blanco Rodriguez **Firma:** _____