

**UNIVERSIDAD DON BOSCO  
FACULTAD DE INGENIERÍA  
ESCUELA DE COMPUTACIÓN  
CICLO I - 2022**



**PROYECTO DE CÁTEDRA  
SEGUNDA ENTREGA**

**INGENIERÍA**  
**DE**  
**SOFTWARE**

**DOCENTE:**

*Inga. Carmen Celia Morales.*

**INTEGRANTES:**

- *Irma Gabriela Hernández Martínez HM190181*
- *Jony Edenilson Morales López ML190272*
- *Carlos Moisés Pérez Cabrera PC190261*
- *Gissela Verenice Serrano López SL190836*
- *Victoria Margarita Sura Jimenez SJ190260*

## Índice

1.	Definición del Proyecto .....	2
1.1.	Descripción detallada del proyecto y planteamiento de objetivos .....	2
	Objetivo General .....	2
	Desarrollar una aplicación web, haciendo uso del concepto E-commerce y el patrón de arquitectura MVC (Modelo – Vista – Controlador) para la realización de compras automatizadas.....	2
	Objetivos Específicos.....	2
1.2.	Empresa seleccionada (rubro y área en la que será aplicada la propuesta del proyecto) .	2
1.3.	Procesos que serán realizados con la automatización del proyecto .....	2
	Alcances.....	2
	Limites .....	3
2.	Marco de desarrollo del proyecto.....	3
2.1.	Actores del Sistema.....	3
2.2.	Requerimientos para el manejo de software.....	3
3.	Metodología de trabajo sugerida.....	5
4.	Técnicas para la recolección de información .....	6
	La Observación .....	6
	La Encuesta.....	6
5.	Recursos Disponibles.....	6
6.	Sistema Actual .....	7
	¿Qué origino el cambio? .....	7
	Aspectos relevantes con respecto del sistema .....	7
7.	Procesos de proyecto desde iniciación hasta cierre .....	8
7.1.	Descripción detallada actividades desarrolladas .....	8
7.1.1.	Procesos de Iniciación .....	8
7.1.2.	Proceso de planificación.....	9
7.1.3.	Proceso de ejecución.....	10
7.1.4.	Proceso de cierre.....	10
7.2.	Responsables y distribución de trabajo .....	11
7.3.	Artefactos generados al final de cada proceso .....	11
8.	Herramientas de seguimiento de procesos .....	12
8.1.	Identificación de herramientas / evaluación .....	12

8.2. Aplicación de herramientas en el proyecto .....	12
9. Herramientas de gestión de configuración de software.....	15
10. Avance de Software .....	37
11. Fuentes de Información .....	50
12. Glosario de Términos .....	51
13. Anexos.....	52

Tabla 1.Interesados del Proyecto.....	3
Tabla 2.Requerimientos Windows .....	4
Tabla 3.Requerimientos Android .....	4
Tabla 4.Requisitos IOS.....	4
Tabla 5.Requisitos Mac OS .....	5
Tabla 6.Roles según Metodología Scrum .....	5
Tabla 7.Recursos Disponibles .....	7
Tabla 8.Roles Scrum .....	8
Tabla 9.Recursos .....	10
Tabla 10.Responsables.....	11
Tabla 11.Artefactos .....	12

## 1. Definición del Proyecto

### 1.1. Descripción detallada del proyecto y planteamiento de objetivos

Se creará un proyecto de programación web de “comercio electrónico” el cual consiste en el marketing y venta de productos o servicios a través de Internet, en nuestro caso se elaborara para una ferretería, en la cual tendrá el funcionamiento del conocido carrito de compras, el contáctanos e implementaremos con el método ágil Scrum en la cual los empleados de la empresa podrán ingresar los productos disponibles en cada sucursal a través del sistema ERP que actualmente han implementado, esto debido a que el sistema de base de datos se encuentra actualmente en “Azure” y esto nos permitirá generar un llenado automático de las existencias de los productos.

#### Objetivo General

Desarrollar una aplicación web, haciendo uso del concepto E-commerce y el patrón de arquitectura MVC (Modelo – Vista – Controlador) para la realización de compras automatizadas.

#### Objetivos Específicos

Hacer uso de la metodología ágil “SCRUM”, para el trabajo colaborativo con el fin de obtener mejores resultados.

Diseñar una interfaz gráfica amigable e intuitiva, haciendo uso de las hojas de estilo en cascada (CSS) y Bootstrap, permitiendo una mejor experiencia al usuario.

Permitir el control de ventas y facturas al administrador, para el manejo de información en la empresa.

Aplicar criterios de seguridad, a través del uso de la encriptación de credenciales y manejo de sesiones, para mantener la confidenciales e integridad de los datos.

### 1.2. Empresa seleccionada (rubro y área en la que será aplicada la propuesta del proyecto)

Ferretería Sumersa, es una ferretería salvadoreña que se ha convertido en un gran distribuidor a nivel nacional de materiales de construcción. El área para la cual se propone el proyecto es para el área de ventas, un eCommerces o mejor conocida como tienda online. Eso beneficiaría a generar un mayor ingreso ya que se llegaría a una mayor población.

### 1.3. Procesos que serán realizados con la automatización del proyecto

Con la implementación del sitio web la empresa automatizará el proceso de compra facilitando a sus consumidores la adquisición de los diversos productos disponibles en cada una de las seis sucursales a través de una tienda en línea, esta tienda en línea contara con las funciones de poder comprar un producto y recibirla en la puerta de tu casa o de poder pasar a recoger el producto en la sucursal de tu elección.

#### Alcances

Diseñar, y crear un prototipo funcional de un sistema enfocado en una idea de negocios que consiste en un servicio online enfocado en la compra de productos, recolectando data de

información respectivos de todo el mercado consumidor (clientes individuales) y que este sistema tenga la funcionalidad de vista empleados, para poder observar todos los pedidos realizados y que estarán alojados en la base de datos.

#### Límites

Ejecutar el sistema de servicio online enfocado a compras, sin tener la posibilidad de generar pruebas directas con el sistema RPG, esto debido a temas seguridad y protección que maneja la institución

## 2. Marco de desarrollo del proyecto

### 2.1. Actores del Sistema

Interesado	Rol	Interés	Expectativa	Influencia
Empresa ferreteria sumersa	Cliente	Contar con una tienda en línea para incrementar sus clientes e ingresos	Alta	Alta
Grupo de trabajo universidad Don Bosco	Desarrolladores e Investigadores	Aprender sobre Ingeniería de software y desarrollar el proyecto de cátedra de la materia	Alta	Alta
Trabajadores Internos	Usuarios del Software	Trabajar de mejor forma y de manera eficiente	Media	Alta
Ing Carmen Morales	Calificador del proyecto	Verificar la puesta en práctica de los conocimientos adquiridos en la materia de Ingeniería de Software	Alta	Alta
DUEÑ@	DUEÑ@	Interés en ahorrar lo que podría significar una inversión a largo plazo, y tener ganancias elevadas	Alta	Media

Tabla 1. Interesados del Proyecto

### 2.2. Requerimientos para el manejo de software

#### Requisitos Recomendados

Windows (de la versión 7 en adelante)

---

Navegadores Web    Google Chrome, Brave, Opera Mini, Mozilla Firefox, etc.

---

---

<b>Procesador</b>	32 o 64 bits Intel Pentium 4 en adelante
<b>Conexión a internet</b>	1 Mbps
<b>RAM</b>	2 GB
<b>Disco</b>	120 GB SSD

---

*Tabla 2.Requerimientos Windows*

Android (de la versión 6.0(Marshmallow) en adelante)

---

<b>Navegadores Web</b>	Google Chrome, Brave, Opera Mini, Mozilla Firefox, etc.
<b>Conexión a internet</b>	1 Mbps
<b>RAM</b>	2 GB
<b>Almacenamiento</b>	16 GB

---

*Tabla 3.Requerimientos Android*

IOS (de la versión 14.0 o posterior)

---

<b>Navegadores Web</b>	Google Chrome, Safari, etc.
<b>Conexión a internet</b>	1 Mbps
<b>RAM</b>	2 GB
<b>Almacenamiento</b>	16 GB

---

*Tabla 4.Requisitos IOS*

Mac Os (de la versión Mac OS X Leopard 10.5.7 o Mac OS X Tiger 10.4.11 en adelante)

<b>Navegadores Web</b>	Google Chrome, Safari, etc.
<b>Procesador</b>	Intel o Power PC G5, G4 o G3
<b>Conexión a internet</b>	1 Mbps
<b>RAM</b>	2 GB
<b>Disco</b>	120 GB SSD

*Tabla 5.Requisitos Mac OS*

### 3. Metodología de trabajo sugerida

El equipo en consenso ha decidido trabajar con la metodología ágil SCRUM, debido a que es una herramienta con la cual los integrantes ya están familiarizados, que busca garantizar el trabajo en equipo a través de entregables a corto plazo con alta prioridad, lo más recomendado con proyectos que exigen flexibilidad y en donde los entregables marcan una pauta importante para el desarrollo de las siguientes etapas del proyecto.

A continuación, se presenta una tabla con los integrantes del grupo con sus respectivos roles.

Categoría	Recurso
Product owner y Scrum Master	Victoria Margarita Sura Jiménez
Team	Gissela Verenice Serrano López
	Irma Gabriela Hernández Martínez
	Jony Edenilson Morales López
	Carlos Moisés Pérez Cabrera

*Tabla 6.Roles según Metodología Scrum*

## 4. Técnicas para la recolección de información

Las técnicas básicas para la recolección de información, se puede definir como; el medio a través del cual el investigador se relaciona con los participantes para obtener la información necesaria que le permita alcanzar los objetivos planteados en la investigación. Entre las técnicas de recolección de información se refieren las siguientes:

**Observación:** se capta de forma sistemática y a través de la vista el fenómeno.

- **Recopilación Documental:** recuperación, análisis, crítica e interpretación de datos secundarios.
- **Entrevista:** diálogo entre entrevistador-entrevistado sobre un tema previamente determinado.
- **Encuesta:** información que se extrae de una muestra acerca de un tema en particular.

En nuestro caso hemos decidido escoger dos de las cuatro técnicas de recolección, esto tomando en cuenta el alcance de estas y la información que se puede recopilar, las cuales detallamos a continuación.

### La Observación

La observación consiste en la indagación sistemática, dirigida a estudiar los aspectos más significativos de los objetos, hechos, situaciones sociales o personas en el contexto donde se desarrollan normalmente; permitiendo la comprensión de la verdadera realidad del fenómeno.

### La Encuesta

La encuesta es una técnica de recopilación de información donde el investigador interroga a los investigados los datos que desea obtener. Se trata de conseguir información, de manera sistemática y ordenada de una población o muestra, sobre las variables consideradas en una investigación. La encuesta permite obtener información de un grupo socialmente significativo de personas relacionadas con el problema de estudio; que posteriormente mediante un análisis cuantitativo o cualitativo, generar las conclusiones que correspondan a los datos recogidos. En la encuesta el nivel de interacción del encuestador con la persona que posee la información es mínimo, pues dicha información es obtenida por preguntas realizadas con instrumentos como el cuestionario.

## 5. Recursos Disponibles

A continuación, se presenta una tabla que contiene los recursos con los que el equipo cuenta y la clasificación de dichos recursos.

Categoría	Recurso
Humano	Victoria Margarita Sura Jiménez
	Gissela Verenice Serrano López
	Irma Gabriela Hernández Martínez
	Carlos Moisés Pérez Cabrera

	Jony Edenilson Morales López
	Computadoras
Materiales	Electricidad
	Oficinas/Establecimiento
	Visual Studio Code
	Equipo con Windows (versión 7 en adelante)
	Equipo con Android (versión 6.0 en adelante)
	Equipo con IOS (versión 14.0 en adelante)
Software	Equipo Mac Os (versión Leopard 10.5.7 o Tiger 10.4.11 en adelante)
	Navegadores (Google Chrome, Brave, Opera Mini, etc.)
	GitHub
	Git
	WampServer
Tiempo	50 días hábiles comenzando desde la semana 6

Tabla 7. Recursos Disponibles

## 6. Sistema Actual

¿Qué origino el cambio?

La búsqueda de productos cual no está siendo alimentado ni actualizado. Lo que origino la necesidad del cambio es el hecho de querer alcanzar nuevas formas de vender y alcanzar nuevos clientes a través de los medios digitales actuales, como lo es la tienda online y las redes sociales.

Aspectos relevantes con respecto del sistema

- **Branding:** El diseño del sitio debe ir de acorde con la imagen de la empresa
- **Optimización de buscadores:** Establecer una estructura amigable para los visitantes de tal manera que pueda encontrar lo que buscan.
- **El Producto:** Establecer el producto con la información relevante de los productos tales como nombre, precio, imagen, etc.
- **Adaptado a dispositivos móviles:** El hecho que se pueda visualizar y comprar por cualquiera de los diferentes dispositivos que existen tan tales como el teléfono, Tablet y computadora.
- **El Carrito:** La parte más destable del proyecto, debido a que es la parte en la que se basa el concepto de tienda online, funcional con contador de productos y sumatorio de los precios que se agregan.

- **La búsqueda de productos:** Importante a la búsqueda de productos que el cliente necesita o busca.
- **Métodos de pago y envío:** Es un aspecto importante y delicado en el tema de seguridad

## 7. Procesos de proyecto desde iniciación hasta cierre

### 7.1. Descripción detallada actividades desarrolladas

#### 7.1.1. Procesos de Iniciación

Las presentes actividades tienen como finalidad conseguir entregables que ayuden a la autorización formal del proyecto.

- Conformar el equipo inicial. Esto fue desarrollado en la semana 2 del C01-2022, igualmente se definió la metodología a utilizar que es SCRUM, para la asignación de tareas y se definieron los roles del equipo, que ayuda a definir niveles jerárquicos y una organización interna de trabajo, quedando de la siguiente manera:

Categoría	Recurso
Product owner y Scrum Master	Victoria Margarita Sura Jiménez
Team	Gissela Verenice Serrano López
	Irma Gabriela Hernández Martínez
	Jony Edenilson Morales López
	Carlos Moisés Pérez Cabrera

Tabla 8.Roles Scrum

- Desarrollo del concepto del proyecto. Esto se puede ver reflejado en la ‘Definición del proyecto’ en donde se identifica la problemática, en este caso el desarrollo de una aplicación web de ‘comercio electrónico’ para la Ferretería Sumersa, basado en las necesidades expuestas por el cliente, que marca un hito para la definición precisa de las actividades a desarrollar.
- Organizar una oficina de proyecto. Debido a la modalidad de trabajo del presente ciclo 01-2022, como equipo se trabajará de manera remota, y con reuniones por videollamadas, a fin de evitar lo más posible las reuniones presenciales.
- Redactar el enunciado preliminar del alcance. De igual manera que al ‘desarrollo del concepto del proyecto’ esto fue tratado en el apartado de ‘Definición del proyecto’

específicamente en el subapartado de ‘Procesos que serán automatizados con la realización del proyecto’, donde se estipulan las características y funciones que tendrá el proyecto.

### 7.1.2. Proceso de planificación

Esta etapa permite definir los objetivos y metas del proyecto, además de definir un plan y una estrategia de cómo desarrollarlo.

- Planteamiento de los objetivos. En el apartado de ‘Definición del proyecto’ se detallan los objetivos que se planean alcanzar con la realización de este proyecto, marcando un indicador que debe ser revisado una vez finalizado el desarrollo.
- Definición del sistema actual. Se describen las características y funcionalidades del sistema actual, para así marcar una línea base desde la cual se parte para definir los aspectos que hay que construir y mejorar del sistema para poder alcanzar las metas definidas en el alcance del proyecto.
- Asignación de recursos. Se definen los recursos que serán necesarios para la implementación del proyecto, debido a la modalidad home office con la que el equipo ha decidido trabajar, los recursos necesarios son más con respecto al área de informática.

Categoría	Recurso
Humano	Victoria Margarita Sura Jiménez
	Gissela Verenice Serrano López
	Irma Gabriela Hernández Martínez
	Carlos Moisés Pérez Cabrera
	Jony Edenilson Morales López
Materiales	Computadoras
	Electricidad
	Oficinas/Establecimiento
	Visual Studio Code
Software	Equipo con Windows (versión 7 en adelante)
	Equipo con Android (versión 6.0 en adelante)
	Equipo con IOS (versión 14.0 en adelante)
	Equipo Mac Os (versión Leopard 10.5.7 o Tiger 10.4.11 en adelante)

	Navegadores (Google Chrome, Brave, Opera Mini, etc.)
	GitHub
	Git
	WampServer
Tiempo	50 días hábiles comenzando desde la semana 6

*Tabla 9. Recursos*

- Definición de herramientas de seguimiento. Con esta actividad el equipo definió a Trello como herramienta de seguimiento, en donde se cuenta con un tablero con las actividades a desarrollar y las personas encargadas de ello, permitiendo una mejor comunicación y trabajo grupal.

#### 7.1.3. Proceso de ejecución

En esta etapa nos enfocamos en el desarrollo del proyecto, donde se dan las siguientes actividades:

- La definición de entregables. Se definen las semanas donde se realizarán entregables, se definen las funcionalidades y metas que deben ser alcanzadas en cada entregable y son plasmados en el tablero Trello del equipo.
- Reuniones de retroalimentación después de cada entregable, por medio de la aplicación de videollamadas de Meet.
- Desarrollo del proyecto. Esta parte se realiza de manera individual, una vez definidas las actividades que corresponden a cada miembro, se trabaja de manera separada notificando sus avances en el tablero de Trello.

#### 7.1.4. Proceso de cierre

En esta etapa se definen las aceptaciones finales del proyecto, una vez que el proyecto ha sido probado por el cliente, aún se debe dar un proceso de cierre tanto interno como externo, en donde se desarrollan las siguientes actividades:

- Aceptación interna del proyecto. Hace referencia a las pruebas finales del equipo de trabajo, si bien a lo largo del desarrollo se han llevado pruebas de calidad, se debe hacer una final, en donde el equipo verifica de mano con la definición del problema, si las metas y objetivos han sido alcanzados y si se cumplen las exigencias del cliente.
- Aceptación externa. Se mantiene una reunión con el cliente, es decir el representante de Ferretería Sumersa, para dar las revisiones finales al proyecto y firmar una carta de aceptación del proyecto, donde se reconoce que el producto final cumple con las expectativas estipuladas al inicio del proyecto.
- Cierre. En este punto se realiza la entrega final del proyecto y se archivan los comprobantes finales en donde el proyecto fue aceptado y finalizado.

## 7.2. Responsables y distribución de trabajo

	<b>ACTIVIDADES</b>	<b>RESPONSABLE</b>
<b>PROCESO DE INICIACIÓN</b>	<i>Conformar Equipo Inicial</i>	TODOS
	<i>Desarrollo del concepto del proyecto</i>	Victoria, Moisés
	<i>Organizar Oficina de Proyecto</i>	TODOS
	<i>Redactar Enunciado preliminar del Alcance</i>	Gissela, Irma y Jony
<b>PROCESO DE PLANIFICACIÓN</b>	<i>Planteamiento de los objetivos</i>	Irma
	<i>Definición del sistema actual</i>	Victoria, Gissela
	<i>Asignación de Recursos</i>	Moisés
	<i>Definimiento de herramientas de seguimiento</i>	Jony
<b>PROCESO DE EJECUCIÓN</b>	<i>Definición de entregables</i>	Gissela e Irma
	<i>Reuniones de Retroalimentación</i>	TODOS
	<i>Desarrollo del Proyecto</i>	TODOS
<b>PROCESO DE CIERRE</b>	<i>Aceptación Interna del Proyecto</i>	TODOS
	<i>Aceptación Externa</i>	Gissela y Victoria (Aprobación del cliente)

Tabla 10.Responsables

## 7.3. Artefactos generados al final de cada proceso

<b>ACTIVIDADES</b>	<b>ARTEFACTOS FINALES</b>
<b>PROCESO DE INICIACIÓN</b>	Conformación del equipo inicial, y lineamientos preliminares del proyecto a realizar.
<b>PROCESO DE PLANIFICACIÓN</b>	Lineamientos base para la hoja de ruta del sistema software, para su comienzo y su seguimiento.
<b>PROCESO DE EJECUCIÓN</b>	Desarrollo del software, medido por entregables distribuidos por semanas.
<b>PROCESO DE CIERRE</b>	ENTREGA DEL PROYECTO FINAL COMPLETO, cumpliendo con la aceptación por parte del equipo, como del cliente beneficiado.

Tabla 11.Artefactos

## 8. Herramientas de seguimiento de procesos

### 8.1. Identificación de herramientas / evaluación



Ilustración 1.Logo de Trello

Trello es una herramienta diseñada para la gestión del trabajo en proyectos, en donde se cuenta con un tablero donde el equipo puede ir añadiendo listas, y a estas listas se le pueden ir añadiendo tarjetas donde se detallan las actividades necesarias a desarrollar, y se asignan a las personas que participaran de dicha tarea, igualmente pueden irse añadiendo comentarios y descripciones para ilustrar de mejor manera en lo que consiste la tarea.

Es por esto que ha sido escogido por el equipo de trabajo, debido a su facilidad de trabajo, la manera gráfica y vistosa que tiene la herramienta para mostrar la información, y porque permite mantener al equipo informado sobre el avance del proyecto y dar una retroalimentación de cada paso.

### 8.2. Aplicación de herramientas en el proyecto

A continuación, se muestran capturas acerca de cómo el equipo ha utilizado la herramienta de seguimiento Trello. Cabe aclarar que las actividades mostradas a través de las capturas son una muestra de todas las actividades que han sido desarrolladas, y solo sirven a manera de ejemplo de cómo ha sido utilizada la herramienta por el equipo de trabajo.

A screenshot of a Trello board titled "Actividades ISO". The board is organized into five columns: "Backlog", "To do", "In progress", "Testing", and "Done".

- Backlog:** Contains items like "Revisión Fase 1 y 2" and "Puntos 8.1 y 9".
- To do:** Contains items like "Modificar Quienes somos", "Modificar contactanos", and "Correcciones Fase 1".
- In progress:** Contains items like "Cambiar los inicios de empleados y administrador", "Agregar Footer", and "Puntos 8.1 y 9".
- Testing:** Contains items like "Diseño y Modelado Base de Datos", "Desarrollo de Mock Ups", "Contactanos", and "Loguearse con google".
- Done:** Contains items like "Diseño de mensaje de recuperación", "Encifrar contraseñas", "Diseño de mensajes de activación", "Mostrar sucursales", "Conexión a base de datos", "Mostrar sucursales", and "Mostrar sucursales".

Each card includes status indicators (e.g., green, yellow, red) and user icons.

Ilustración 2.Visualizacion General del Tablero de Trello

Como se puede observar, se cuentan con cinco tipos de lista, en las que se dividen las actividades, las cuales son:

1. Backlog. En este apartado se asignan las actividades de reserva, que deberán ser evaluadas por el equipo al finalizar con los entregables primarios.



Ilustración 3.Imagen de referencia de backlog

2. To do. Es la lista de actividades pendientes que tiene el equipo.

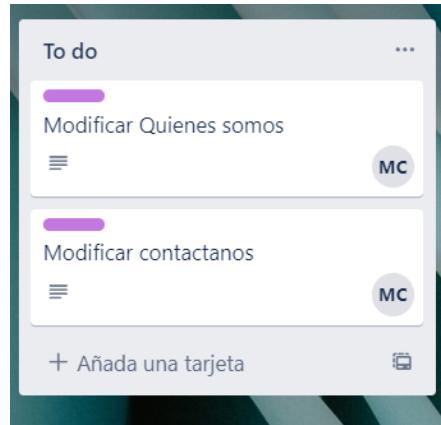


Ilustración 4.Imagen de referencia de To do

3. In progress. Son las actividades que se encuentran en desarrollo.

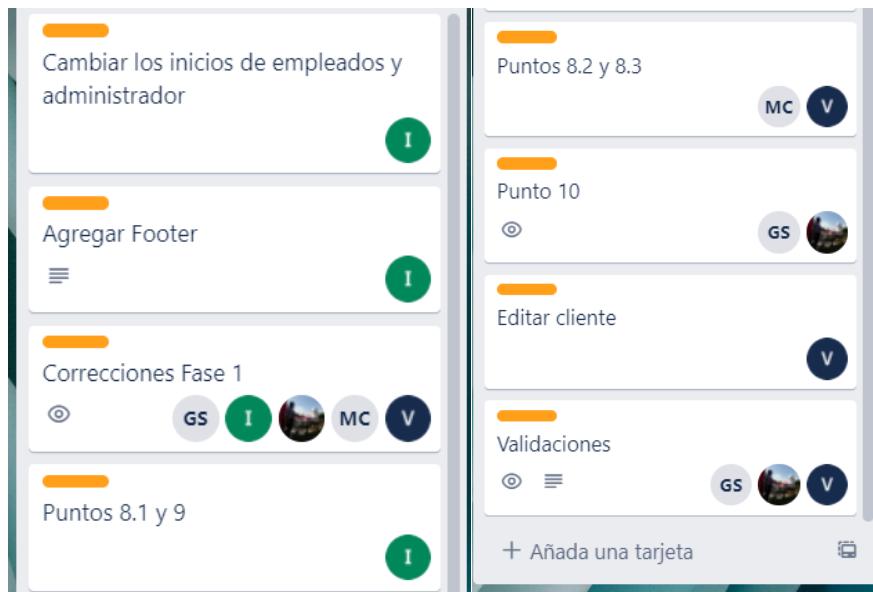


Ilustración 5.Imagen de referencia de In progress

4. Testing. Son los entregables que deben ser revisados para garantizar su funcionalidad.

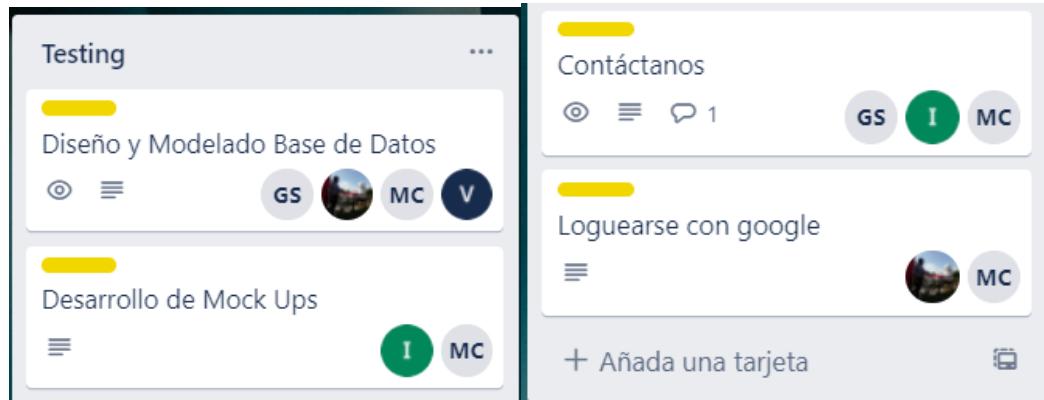
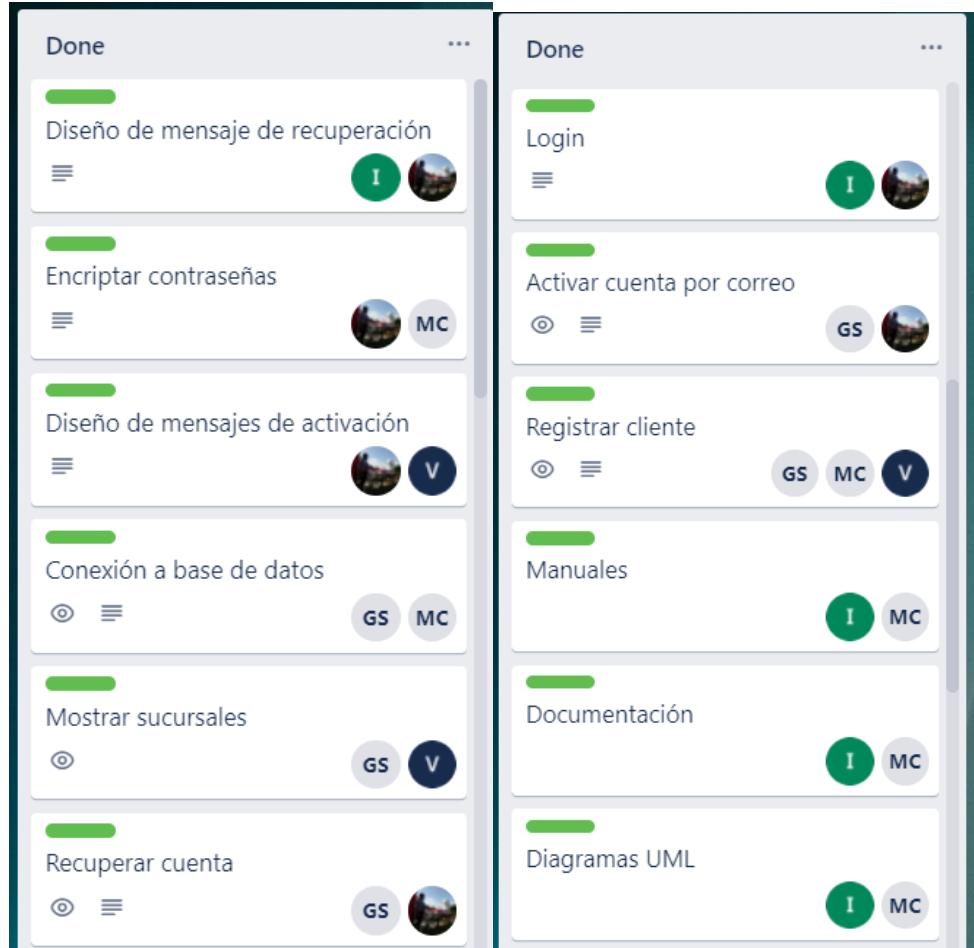


Ilustración 6.Imagen de referencia de Testing

5. Done. Son los apartados que se consideran listos y que cumplen con las exigencias del equipo.



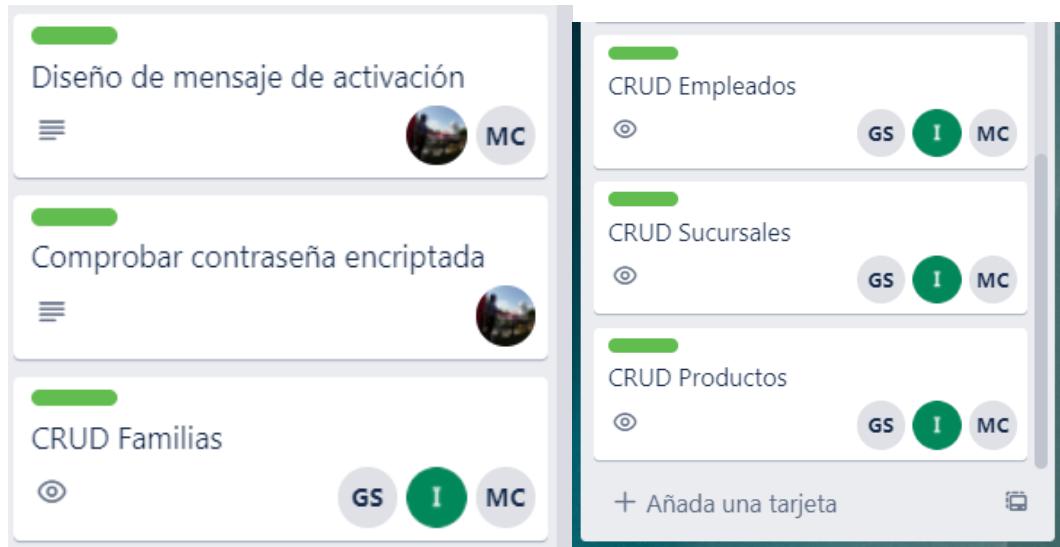
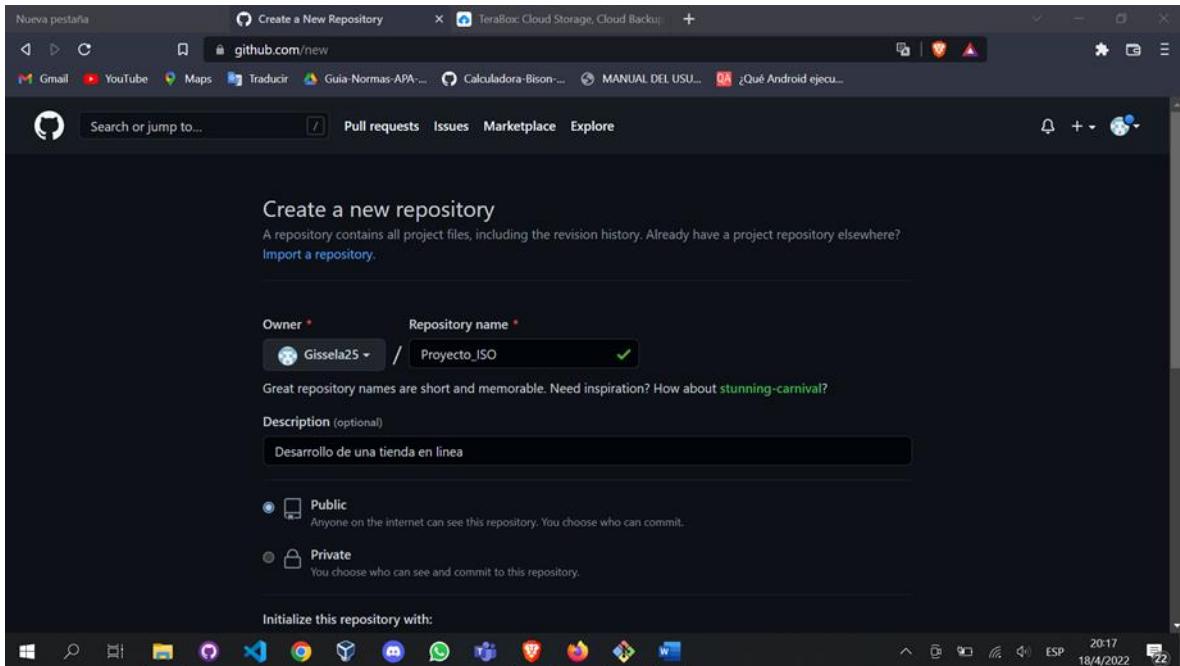


Ilustración 7.Imagen de referencia a Done

## 9. Herramientas de gestión de configuración de software

En el desarrollo del presente proyecto se utilizará git en conjunto con github como herramientas de configuración de software, con el objetivo de facilitar el trabajo en equipo permitiendo que cada uno de los colaboradores pueda trabajar en su propia copia local del proyecto, permitiendo el trabajo simultáneo en diferentes ramas, la comparación entre versiones tanto actuales con anteriores, y locales con las que se encuentra en el repositorio, y su vez asegurando que no existan conflictos con modificaciones introducidas por otros miembros del equipo.

Como primer paso procedemos a crear un nuevo repositorio en GitHub



*Ilustración 8.Creación de repositorio*

Nos dirigimos a la carpeta donde tenemos los archivos que queremos subir al repositorio

```
gisse@DESKTOP-OF2SICA MINGW64 ~
$ cd /c/wamp64/www/ISO
gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (master)
$ ls
Assets/ Controllers/ Core/ DataBase/ Documents/ Models/ README.md Views/ cacert.pem composer.json composer.lock img/ index.php vendor/
gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (master)
$ |
```

*Ilustración 9.Ubicación del repositorio local*

Antes de continuar procedemos a establecer los valores de configuración a nuestro usuario

```
MINGW64:/c/wamp64/www/ISO

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO
$ git config --global user.name Gissela25

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO
$ git config --global email.name gisselaverenice@gmail.com
```

*Ilustración 10.Configurando git*

Procedemos a revisar si la configuración ha sido guardada de forma correcta

```
MINGW64:/c/wamp64/www/ISO  
gisse@DESKTOP-OF2SICA MINGW64 ~  
$ cd /c/wamp64/www/ISO  
gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (master)  
$ git config --global -l  
filter.lfs.process=git-lfs filter-process  
filter.lfs.required=true  
filter.lfs.clean=git-lfs clean -- %f  
filter.lfs.smudge=git-lfs smudge -- %f  
user.name=Gisellea25  
user.email=gisselaverenice@gmail.com  
email.name=gisselaverenice@gmail.com  
gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (master)  
$ |
```

Ilustración 11.Comprobando Configuraciones

A continuación, utilizaremos el comando git init el cual inicializa un repositorio git, creando el directorio .git en un proyecto nuevo o existente

```
gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO  
$ git init  
Initialized empty Git repository in C:/wamp64/www/ISO/.git/  
gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (master)  
$ |
```

Ilustración 12.Inicializando repositorio

Procedemos a utilizar el comando git status para confirmar que archivos ya se agregaron al repositorio en nuestro caso ninguno a sido agregado aún por lo tanto nos aparecerán en color rojo

```

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    Assets/
    Controllers/
    Core/
    DataBase/
    Documents/
    Models/
    README.md
    Views/
    cacert.pem
    composer.json
    composer.lock
    img/
    index.php
    vendor/

nothing added to commit but untracked files present (use "git add" to track)

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (master)
$ |

```

Ilustración 13.Comprobando Estatus

Utilizamos el comando git add \* para agregar nuestros archivos

```

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (master)
$ git add *

```

```

MINGW64/c/wamp64/www/ISO
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in vendor/google/apiclient-services/src/Monitoring/Telemetry.php.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in vendor/google/apiclient-services/src/Monitoring/TimeInterval.php.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in vendor/google/apiclient-services/src/Monitoring/TimeSeries.php.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in vendor/google/apiclient-services/src/Monitoring/TimeSeriesData.php.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in vendor/google/apiclient-services/src/Monitoring/TimeSeriesDescriptor.php.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in vendor/google/apiclient-services/src/Monitoring/TimeSeriesRatio.php.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in vendor/google/apiclient-services/src/Monitoring/Trigger.php.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in vendor/google/apiclient-services/src/Monitoring/Type.php.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in vendor/google/apiclient-services/src/Monitoring/TypedValue.php.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in vendor/google/apiclient-services/src/Monitoring/UptimeCheckConfig.php.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in vendor/google/apiclient-services/src/Monitoring/UptimeCheckZip.php.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in vendor/google/apiclient-services/src/Monitoring/ValueDescriptor.php.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in vendor/google/apiclient-services/src/Monitoring/VerifyNotificationChannelRequest.php.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in vendor/google/apiclient-services/src/Monitoring/WindowsBasedSTI.php.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in vendor/google/apiclient-services/src/MyBusinessAccountManagement.php.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in vendor/google/apiclient-services/src/MyBusinessAccountManagement/AcceptInvitationRequest.php.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in vendor/google/apiclient-services/src/MyBusinessAccountManagement/Account.php.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in vendor/google/apiclient-services/src/MyBusinessAccountManagement/Admin.php.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in vendor/google/apiclient-services/src/MyBusinessAccountManagement/DeclineInvitationRequest.php.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in vendor/google/apiclient-services/src/MyBusinessAccountManagement/Invitation.php.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in vendor/google/apiclient-services/src/MyBusinessAccountManagement/ListAccountAdminsResponse.php.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in vendor/google/apiclient-services/src/MyBusinessAccountManagement/ListAccountsResponse.php.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in vendor/google/apiclient-services/src/MyBusinessAccountManagement/ListInvitationsResponse.php.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in vendor/google/apiclient-services/src/MyBusinessAccountManagement/ListLocationAdminsResponse.php.
The file will have its original line endings in your working directory

```

Ilustración 14.Agregando archivos

Comprobamos que los archivos hayan sido agregados de forma correcta usando el comando git status los archivos cambiarán a color verde. Procedemos a realizar el primer commit

```
gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (master)
$ git commit -m "Archivos principales"

create mode 100644 vendor/monolog/monolog/src/Monolog/Processor/MercurialProcessor.php
create mode 100644 vendor/monolog/monolog/src/Monolog/Processor/ProcessSTDProcessor.php
create mode 100644 vendor/monolog/monolog/src/Monolog/Processor/ProcessorInterface.php
create mode 100644 vendor/monolog/monolog/src/Monolog/Processor/PsrLogMessageProcessor.php
create mode 100644 vendor/monolog/monolog/src/Monolog/Processor/TagProcessor.php
create mode 100644 vendor/monolog/monolog/src/Monolog/Processor/UidProcessor.php
create mode 100644 vendor/monolog/monolog/src/Monolog/Processor/WebProcessor.php
create mode 100644 vendor/monolog/monolog/src/Monolog/Processor/XmlProcessor.php
create mode 100644 vendor/monolog/monolog/src/Monolog/Processor/XmlSerializableProcessor.php
create mode 100644 vendor/monolog/monolog/src/Monolog/SignalHandler.php
create mode 100644 vendor/monolog/monolog/src/Monolog/Utils.php
create mode 100644 vendor/phpseclib/phpseclib/AUTHORS
create mode 100644 vendor/phpseclib/phpseclib/ACKERS.md
create mode 100644 vendor/phpseclib/phpseclib/LICENSE
create mode 100644 vendor/phpseclib/phpseclib/README.md
create mode 100644 vendor/phpseclib/phpseclib/appveyor.yml
create mode 100644 vendor/phpseclib/phpseclib/.gitignore
create mode 100644 vendor/phpseclib/phpseclib/.gitmodules
create mode 100644 vendor/phpseclib/phpseclib/crypt/AES.php
create mode 100644 vendor/phpseclib/phpseclib/phpseclib/crypt/Base.php
create mode 100644 vendor/phpseclib/phpseclib/phpseclib/crypt/Blowfish.php
create mode 100644 vendor/phpseclib/phpseclib/phpseclib/crypt/DES.php
create mode 100644 vendor/phpseclib/phpseclib/phpseclib/crypt/Hash.php
create mode 100644 vendor/phpseclib/phpseclib/phpseclib/crypt/RC2.php
create mode 100644 vendor/phpseclib/phpseclib/phpseclib/crypt/RC4.php
create mode 100644 vendor/phpseclib/phpseclib/phpseclib/crypt/RSA.php
create mode 100644 vendor/phpseclib/phpseclib/phpseclib/crypt/Random.php
create mode 100644 vendor/phpseclib/phpseclib/phpseclib/crypt/Rijndael.php
create mode 100644 vendor/phpseclib/phpseclib/phpseclib/crypt/TripleDES.php
create mode 100644 vendor/phpseclib/phpseclib/phpseclib/crypt/Twofish.php
create mode 100644 vendor/phpseclib/phpseclib/phpseclib/File/ANSI.php
create mode 100644 vendor/phpseclib/phpseclib/phpseclib/File/ASN1.php
create mode 100644 vendor/phpseclib/phpseclib/phpseclib/File/Element.php
create mode 100644 vendor/phpseclib/phpseclib/phpseclib/File/X509.php
create mode 100644 vendor/phpseclib/phpseclib/phpseclib/Math/BigInteger.php
create mode 100644 vendor/phpseclib/phpseclib/phpseclib/Net/SCP.php
create mode 100644 vendor/phpseclib/phpseclib/phpseclib/Net/FTP.php
create mode 100644 vendor/phpseclib/phpseclib/phpseclib/Net/FTPStream.php
create mode 100644 vendor/phpseclib/phpseclib/phpseclib/Net/SSH1.php
create mode 100644 vendor/phpseclib/phpseclib/phpseclib/Net/SSH2.php
create mode 100644 vendor/phpseclib/phpseclib/phpseclib/System/SSH/Agent.php
create mode 100644 vendor/phpseclib/phpseclib/phpseclib/System/SSH/Agent/Identity.php
create mode 100644 vendor/phpseclib/phpseclib/phpseclib/bootstrap.php
create mode 100644 vendor/phpseclib/phpseclib/phpseclib/openssl.cnf
create mode 100644 vendor/psr/cache/CHANGELOG.md
create mode 100644 vendor/psr/cache/LICENSE.txt
create mode 100644 vendor/psr/cache/README.md
create mode 100644 vendor/psr/cache/composer.json
create mode 100644 vendor/psr/cache/src/CacheException.php
create mode 100644 vendor/psr/cache/src/CacheItemInterface.php
```

Ilustración 15. Realizando Commit

El siguiente paso es utilizar git remote para identificar el repositorio

```
gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (master)
$ git remote add origin https://github.com/Gissela25/Proyecto_ISO.git
```

Ilustración 16. Identificando repositorio

Finalmente, para enviar los archivos de nuestra archivo local al repositorio global utilizamos git push

```
gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (master)
$ git push -u origin master
Enumerating objects: 17070, done.
Counting objects: 100% (17070/17070), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2610/2610), done.
Writing objects: 100% (17070/17070), 18.90 MiB | 76.00 KiB/s, done.
Total 17070 (delta 14408), reused 17070 (delta 14408), pack-reused 0
remote: Resolving deltas: 100% (14408/14408), done.
To https://github.com/Gissela25/Proyecto_ISO.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
```

Ilustración 17. Realizando Push

Para comprobar que los archivos hayan sido subidos de forma correcta al repositorio local podemos implementar git status el cual nos confirmara todos nuestros archivos ya fueron agregados

```
gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (master)
$ git status
On branch master
nothing to commit, working tree clean
```

Ilustración 18.Comprobando estatus de archivos

De igual forma si confirmamos desde GitHub veremos los archivos que subimos

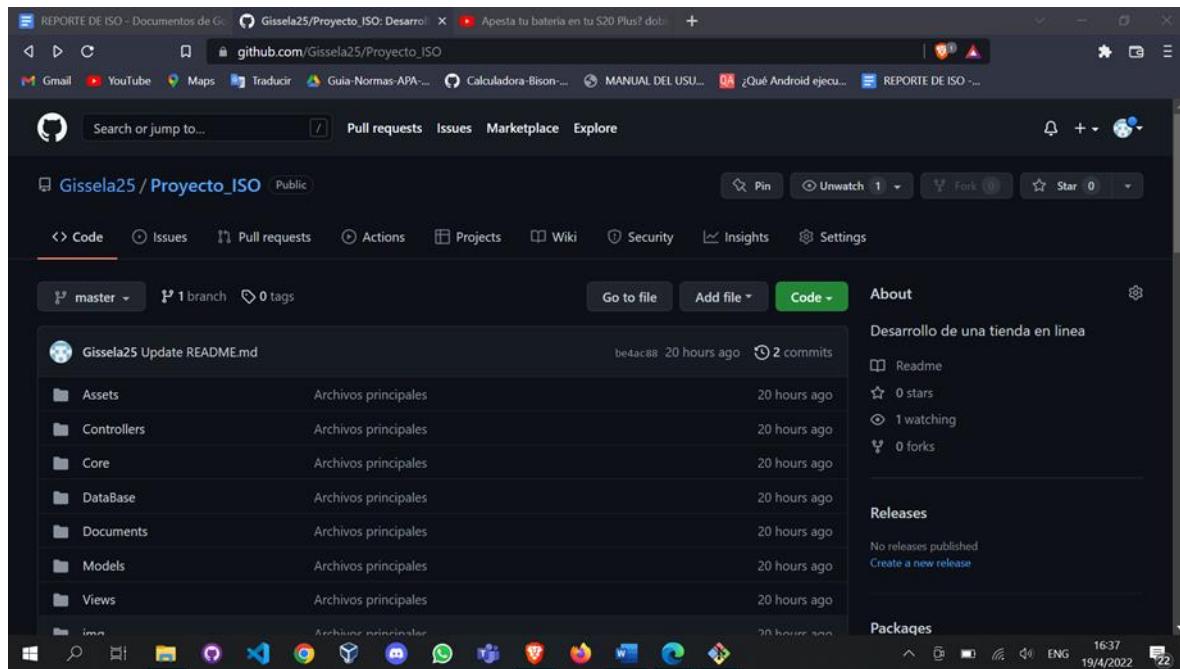


Ilustración 19.Corroboration en GitHub

Otro uso que le daremos a git será para agregar nuevas ramas en este caso crearemos una rama para que cada miembro del equipo pueda trabajar en una rama individual.

```
MINGW64:/c/wamp64/www/ISO
gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (master)
$ git branch Jony

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (master)
$ git branch victoria

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (master)
$ git branch Moises

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (master)
$ git branch Irma

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (master)
$ git branch Gissela

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (master)
$ |
```

Ilustración 20.Agregando Ramas

Listamos nuestras ramas y vemos que estamos trabajando en master

```
gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (master)
$ git branch
Gissela
Irma
Jony
Moises
* master
victoria

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (master)
$ |
```

Ilustración 21.Listando Ramas

Para cambiarnos a la Jony rama utilizamos el comando git checkout

```

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (master)
$ git checkout Jony
Switched to branch 'Jony'

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (Jony)
$ |

```

Ilustración 22.Cambiando de Rama

Procedemos a agregar el documento de la primera fase al repositorio local para tener un cambio en la rama Jony

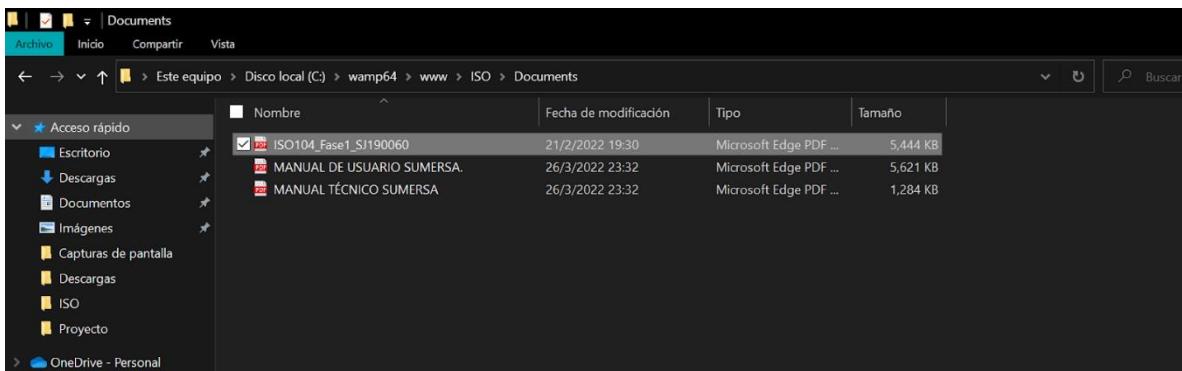


Ilustración 23.Agregando archivos

Confirmamos con Git status que este archivo aún no ha sido agregado

```

MINGW64:/c/wamp64/www/ISO

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (Jony)
$ git status
On branch Jony
Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
    (use "git restore <file>..." to discard changes in working directory)
      deleted:   Documents/PRIMERA FASE LIS.pdf

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    Documents/ISO104_Fase1_SJ190060.pdf

no changes added to commit (use "git add" and/or "git commit -a")

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (Jony)
$ |

```

Ilustración 24.Revisando Estatus

Procedemos a agregarlo

```
gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (Jony)
$ git add Documents/ISO104_Fase1_SJ190060.pdf

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (Jony)
$ |
```

Ilustración 25.Agregando nuevo archivo

Como ya ha sido agregado realizamos commit

```
gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (Jony)
$ git commit -m "Agregando Documentacion"
[Jony e8fea534] Agregando Documentacion
 1 file changed, 80550 insertions(+)
 create mode 100644 Documents/ISO104_Fase1_SJ190060.pdf
```

Ilustración 26.Realizando nuevo commit

Comprobamos que se haya agregado

```
gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (Jony)
$ git status
On branch Jony
Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    deleted:   Documents/PRIMERA FASE LIS.pdf

no changes added to commit (use "git add" and/or "git commit -a")

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (Jony)
```

Ilustración 27.Revisando Estatus

Como se agrego con éxito procedemos hacer Push

```
gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (Jony)
$ git push -u origin Jony
Enumerating objects: 17074, done.
Counting objects: 100% (17074/17074), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2614/2614), done.
Writing objects: 100% (17074/17074), 23.06 MiB | 156.00 KiB/s, done.
Total 17074 (delta 14410), reused 17068 (delta 14408), pack-reused 0
remote: Resolving deltas: 100% (14410/14410), done.
remote:
remote: Create a pull request for 'Jony' on GitHub by visiting:
remote:   https://github.com/Gissela25/Proyecto_ISO/pull/new/Jony
remote:
To https://github.com/Gissela25/Proyecto_ISO.git
 * [new branch]      Jony -> Jony
branch 'Jony' set up to track 'origin/Jony'.

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (Jony)
$ |
```

Ilustración 28.Realizando Push

## Comprobamos

```
gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (Jony)
$ git status
On branch Jony
Your branch is up to date with 'origin/Jony'.

Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
    (use "git restore <file>..." to discard changes in working directory)
      deleted:   Documents/PRIMERA FASE LIS.pdf

no changes added to commit (use "git add" and/or "git commit -a")

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (Jony)
$ |
```

Ilustración 29.Verificando Estatus

De igual forma corroboramos desde GitHub y podemos ver que la rama jony ha sido agregada correctamente al repositorio

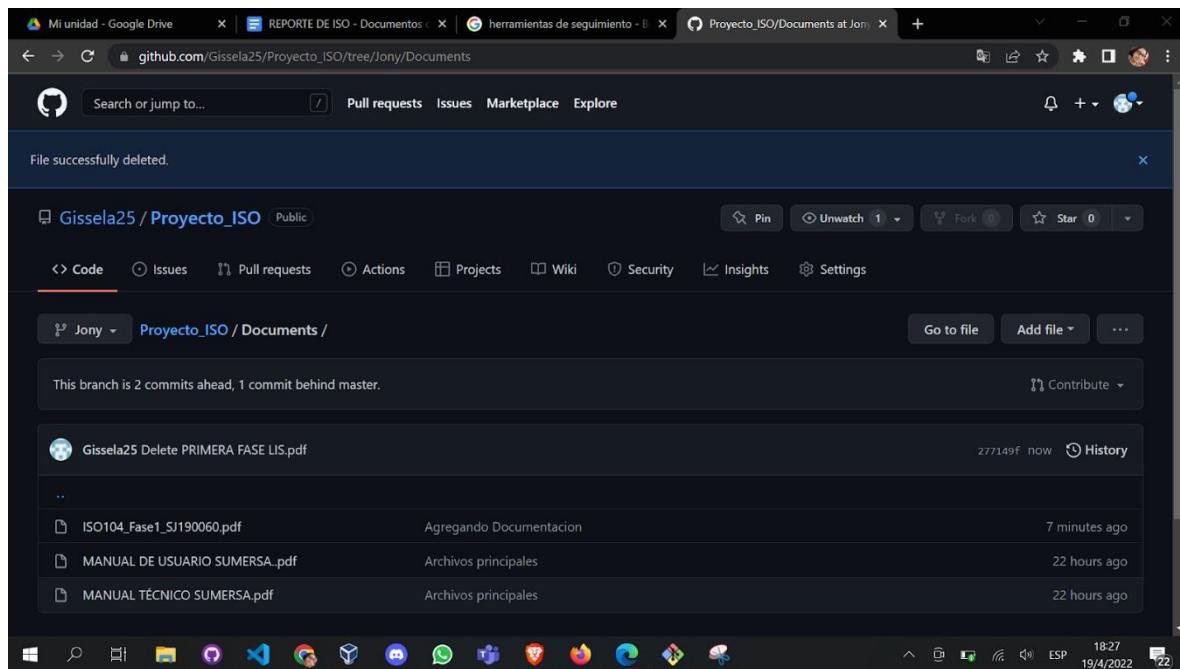


Ilustración 30.Comprobación desde GitHub

Para agregar el contenido al resto de ramas realizamos lo siguiente

```
gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/Proyecto_ISO (master)
$ git push -u origin victoria
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'victoria' on GitHub by visiting:
remote:     https://github.com/Gissela25/Proyecto_ISO/pull/new/victoria
remote:
To https://github.com/Gissela25/Proyecto_ISO.git
 * [new branch]      victoria -> victoria
branch 'victoria' set up to track 'origin/victoria'.

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/Proyecto_ISO (master)
$ git push -u origin Gissela
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'Gissela' on GitHub by visiting:
remote:     https://github.com/Gissela25/Proyecto_ISO/pull/new/Gissela
remote:
To https://github.com/Gissela25/Proyecto_ISO.git
 * [new branch]      Gissela -> Gissela
branch 'Gissela' set up to track 'origin/Gissela'.
```

Ilustración 31.Push a ramas Victoria y Gissela

```
gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/Proyecto_ISO (master)
$ git push -u origin Moises
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'Moises' on GitHub by visiting:
remote:     https://github.com/Gissela25/Proyecto_ISO/pull/new/Moises
remote:
To https://github.com/Gissela25/Proyecto_ISO.git
 * [new branch]      Moises -> Moises
branch 'Moises' set up to track 'origin/Moises'.

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/Proyecto_ISO (master)
$ git push -u origin Irma
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'Irma' on GitHub by visiting:
remote:     https://github.com/Gissela25/Proyecto_ISO/pull/new/Irma
remote:
To https://github.com/Gissela25/Proyecto_ISO.git
 * [new branch]      Irma -> Irma
branch 'Irma' set up to track 'origin/Irma'.

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/Proyecto_ISO (master)
$ |
```

Ilustración 32.Push a ramas Moises y Irma

Corroboramos en GitHub y vemos que tenemos las seis ramas creadas y con los mismos archivos de master

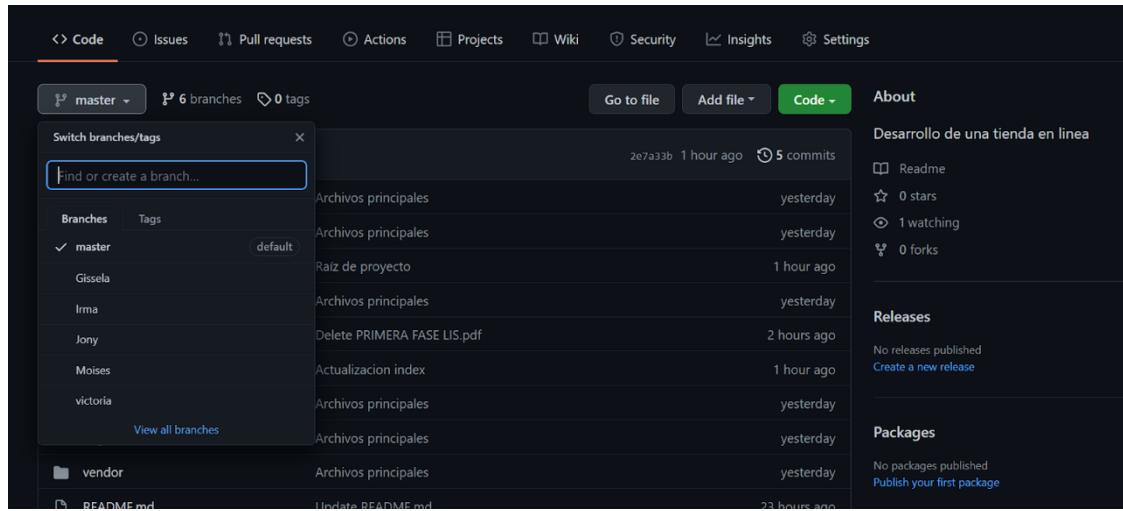


Ilustración 33. Visualizando ramas en GitHub

Aprovechando que nuestro compañero realizó un commit procedemos a emplear el comando git pull para comparar el contenido de mi localhost con el del repositorio

```
gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (Jony)
$ git checkout master
Switched to branch 'master'
D      Documents/PRIMERA FASE LIS.pdf
M      Views/Constant/admin/index.php
Your branch is up to date with 'origin/master'.

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (master)
$ git pull
remote: Enumerating objects: 25, done.
remote: Counting objects: 100% (22/22), done.
remote: Compressing objects: 100% (12/12), done.
remote: Total 15 (delta 8), reused 6 (delta 2), pack-reused 0
Unpacking objects: 100% (15/15), 2.55 KiB | 41.00 KiB/s, done.
From https://github.com/Gissela25/Proyecto_ISO
  d01a2347..8272ae17  master      -> origin/master
    e8fea534..c30d1f1a  Jony      -> origin/Jony
Updating d01a2347..8272ae17
Fast-forward
 Core/config.php          |  2 +-+
 Documents/PRIMERA FASE LIS.pdf | Bin 1411823 -> 0 bytes
 README.md                | 28 +-----
 3 files changed, 3 insertions(+), 27 deletions(-)
 delete mode 100644 Documents/PRIMERA FASE LIS.pdf

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/ISO (master)
$ |
```

Ilustración 34. Realizando Push

posteriormente realizamos un cambio en el código

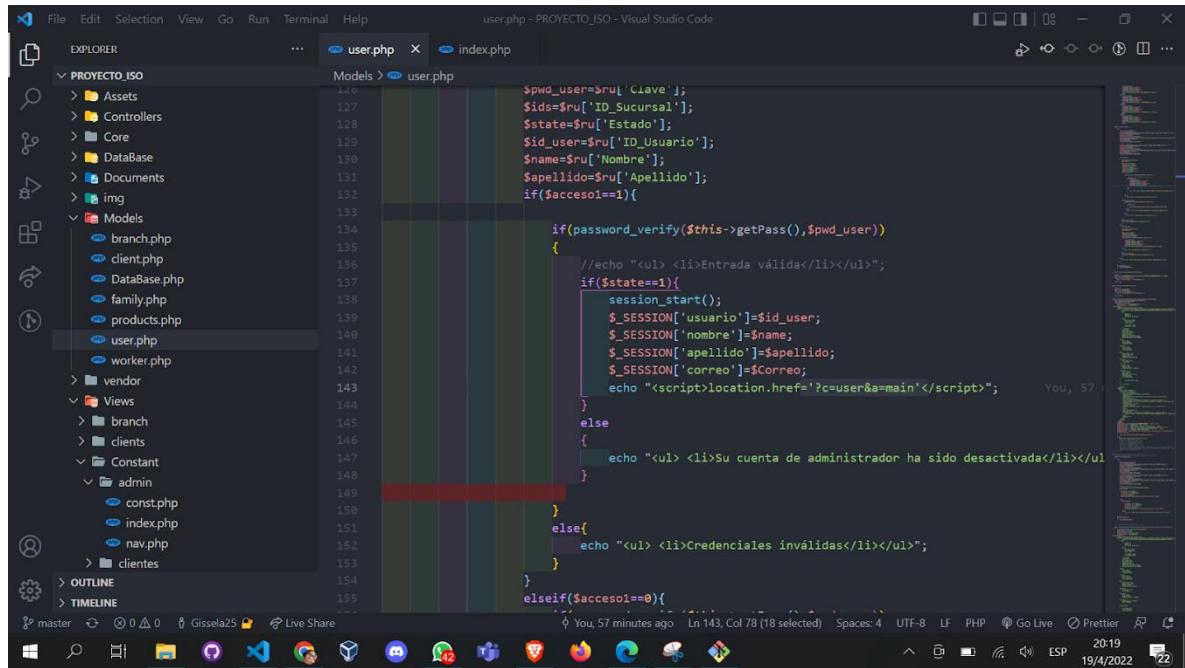


Ilustración 35.Cambio en el código

confirmamos que el cambio no haya sido agregado

```
gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/Proyecto_ISO (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   Models/user.php

no changes added to commit (use "git add" and/or "git commit -a")
```

Ilustración 36.Verificando Estatus del cambio

procedemos agregarlo y confirmar que se haya agregado

```
gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/Proyecto_ISO (master)
$ git add Models/user.php
warning: LF will be replaced by CRLF in Models/user.php.
The file will have its original line endings in your working directory

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/Proyecto_ISO (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   Models/user.php
```

Ilustración 37.Agregando Cambio

Procedemos a hacer commit y agregarlo al repositorio

```

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/Proyecto_ISO (master)
$ git commit -m "Actualizacion index"
[master 2e7a33b6] Actualizacion index
 1 file changed, 1 insertion(+), 1 deletion(-)

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/Proyecto_ISO (master)
$ git push
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 4 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 402 bytes | 402.00 KiB/s, done.
Total 4 (delta 3), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
To https://github.com/Gissela25/Proyecto_ISO.git
  8272ae17..2e7a33b6 master -> master

```

Ilustración 38. Realizando Commit

A continuación nos cambiaremos de rama y realizaremos un cambio en el código

```

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/Proyecto_ISO (Gissela)
$ git checkout Victoria
Switched to branch 'Victoria'

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/Proyecto_ISO (Victoria)
$ |

```

Ilustración 39. Cambiando de rama

```

family.php
products.php
user.php
worker.php
> vendor
  > Views
    > branch
    > clients
    > Constant
      > admin
        const.php
        index.php
        nav.php
      > clientes
    > OUTLINE

119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134

<br>
<iframe
  src="https://www.google.com/maps/embed?pb=!1m18!1m12!1m3!1d941.9268234464838!2d-89.369144246374
  width="385" height="235" style="border:0;" allowfullscreen="" loading="lazy"></iframe>
<div style="font-family:Anton; font-size:260%; color: #000000; border: 1px solid #000000; padding: 5px; margin-bottom: 10px;">
<span style="font-family:Open Sans; font-size:120%;">Teléfono: (503) 2338 - 4821</span>
<p style="font-family:Open Sans; font-size:120%;">Fax: (503) 2338 - 6843</p>
<a href="https://web.whatsapp.com/send?phone=50370391253&text=" target="_blank">
  
</a>
</div>
<div class="col-md-4">
<br>
<iframe
  src="https://www.google.com/maps/embed?pb=!1m18!1m12!1m3!1d1342.98830731772!2d-89.3687084051561
  width="385" height="235" style="border:0;" allowfullscreen="" loading="lazy"></iframe>

```

Ilustración 40. Realizando Cambio

Subimos los cambios al repositorio en la rama Victoria

```

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/Proyecto_ISO (Victoria)
$ git add Views/Constant/admin/index.php

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/Proyecto_ISO (Victoria)
$ git commit -m "Update"
[Victoria 7ba3cef0] Update
 1 file changed, 1 insertion(+), 1 deletion(-)

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/Proyecto_ISO (Victoria)
$ git pull
There is no tracking information for the current branch.
Please specify which branch you want to merge with.
See git-pull(1) for details.

      git pull <remote> <branch>

If you wish to set tracking information for this branch you can do so with:

      git branch --set-upstream-to=origin/<branch> Victoria

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/Proyecto_ISO (Victoria)
$ git push
fatal: The current branch Victoria has no upstream branch.
To push the current branch and set the remote as upstream, use

      git push --set-upstream origin Victoria

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/Proyecto_ISO (Victoria)
$ git push -u origin Victoria
Enumerating objects: 11, done.
Counting objects: 100% (11/11), done.
Delta compression using up to 4 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 540 bytes | 540.00 KiB/s, done.
Total 6 (delta 4), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (4/4), completed with 4 local objects.
remote:
remote: Create a pull request for 'Victoria' on GitHub by visiting:
remote:     https://github.com/Gissela25/Proyecto_ISO/pull/new/Victoria
remote:
To https://github.com/Gissela25/Proyecto_ISO.git
 * [new branch]      Victoria -> Victoria
branch 'Victoria' set up to track 'origin/Victoria'.

```

*Ilustración 41. Actualizando Rama*

A veces necesitamos deshacer nuestros cambios el comando para realizar esta acción es git revert pero primero debemos emplear git log --oneline: para ver nuestro historial de commits tomamos el número del commit que queremos revertir y lo usamos de la siguiente forma

```

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/Proyecto_ISO (Victoria)
$ git log --oneline
7ba3cef0 (HEAD -> Victoria, origin/Victoria) Update
d01a2347 (origin/Moises, origin/Irma, Moises, Irma) Archivos principales

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/Proyecto_ISO (Victoria)
$ git revert 7ba3cef0
[Victoria a3a0199f] Revert "Update"
 1 file changed, 1 insertion(+), 1 deletion(-)

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/Proyecto_ISO (Victoria)
$ git push
Enumerating objects: 11, done.
Counting objects: 100% (11/11), done.
Delta compression using up to 4 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 563 bytes | 563.00 KiB/s, done.
Total 6 (delta 4), reused 1 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (4/4), completed with 4 local objects.
To https://github.com/Gissela25/Proyecto_ISO.git
  7ba3cef0..a3a0199f  Victoria -> Victoria

gisse@DESKTOP-OF2SICA MINGW64 /c/wamp64/www/Proyecto_ISO (Victoria)
$ |

```

Ilustración 42.Revirtiendo Commit

confirmamos que se revirtió el cambio

This branch is 4 commits ahead, 4 commits behind master.

	Gissela25 Update2	2ea2c9f 2 minutes ago	5 commits
	Assets	Archivos principales	yesterday
	Controllers	Archivos principales	yesterday
	Core	Archivos principales	yesterday
	DataBase	Archivos principales	yesterday
	Documents	Agregando Documentacion	3 hours ago
	Models	Archivos principales	yesterday
	Views	Revert "Update"	18 minutes ago
	img	Archivos principales	yesterday
	vendor	Archivos principales	yesterday
	README.md	Archivos principales	yesterday

Releases

No releases published  
Create a new release

Packages

No packages published  
Publish your first package

Languages

PHP 96.7% CSS 3.0

Ilustración 43.Confirmando reversión

Luego que uno de los miembros hace un commit a su rama

	Irma-1247 Cambio inicio y footer	e7d6806 33 minutes ago	6 commits
	Assets	Cambio inicio y footer	33 minutes ago
	Controllers	Archivos principales	33 minutes ago

Ilustración 44.Revisando nuevo commit

Realizaremos un merge con la rama master en este caso utilizaremos GitHub para realizar el proceso

The screenshot shows a GitHub repository page for 'Gissela25 / Proyecto\_ISO'. The 'Code' tab is selected. A tooltip appears over a branch commit, stating 'This branch is 1 commit ahead of master.' with a link to 'Open pull request'. The repository structure shows several folders like Assets, Controllers, Core, and DataBase. On the right side, there's an 'About' section with a description 'Desarrollo de una tienda en linea', a 'Readme' file, and stats for stars (0), forks (0), and watching (1). There's also a 'Releases' section indicating no releases have been published.

Ilustración 45. Realizando merge desde GitHub

Hacemos click en open pull request y nos aparecerá la siguiente pantalla donde podemos agregar un comentario al merge y procedemos a crear pull request

The screenshot shows the 'Open a pull request' dialog box. It has fields for 'base: master' and 'compare: Irma'. A green checkmark indicates 'Able to merge'. The main area is titled 'Cambio inicio y footer' and contains a 'Write' tab with a rich text editor and a 'Preview' tab. A note says 'Leave a comment' and 'Attach files by dragging & dropping, selecting or pasting them.' At the bottom is a 'Create pull request' button. To the right, there are sections for 'Reviewers' (No reviews), 'Assignees' (No one—assign yourself), 'Labels' (None yet), 'Projects' (None yet), 'Milestone' (No milestone), and 'Development' (with a gear icon).

Ilustración 46. Revisando merge

Como paso final realizamos el merge

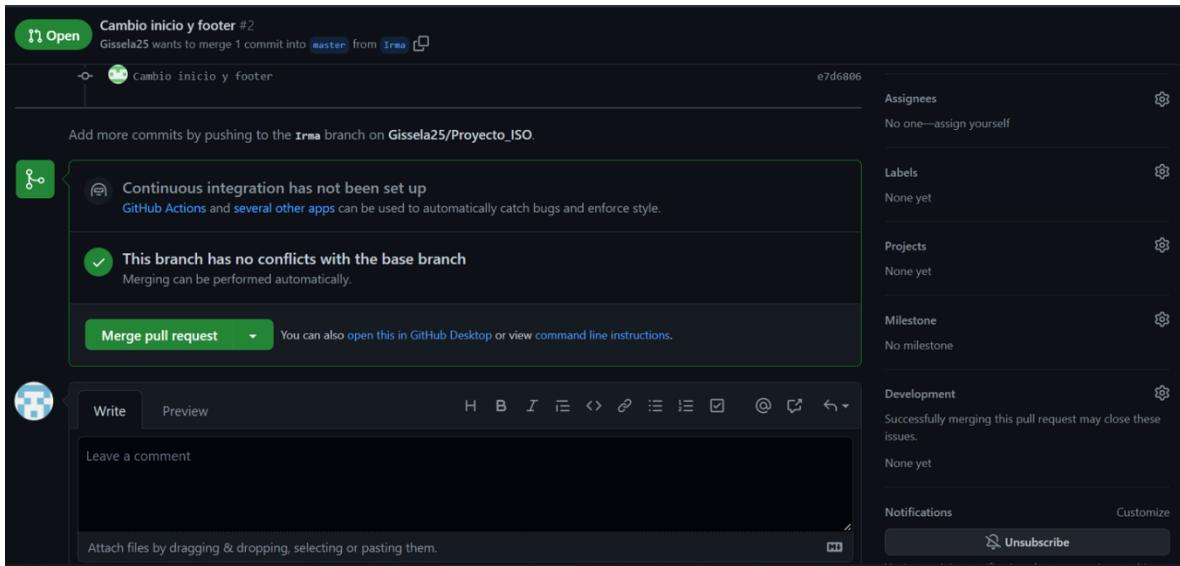


Ilustración 47.Finalizando merge

Procedemos a comprobar que el merge haya sido exitoso

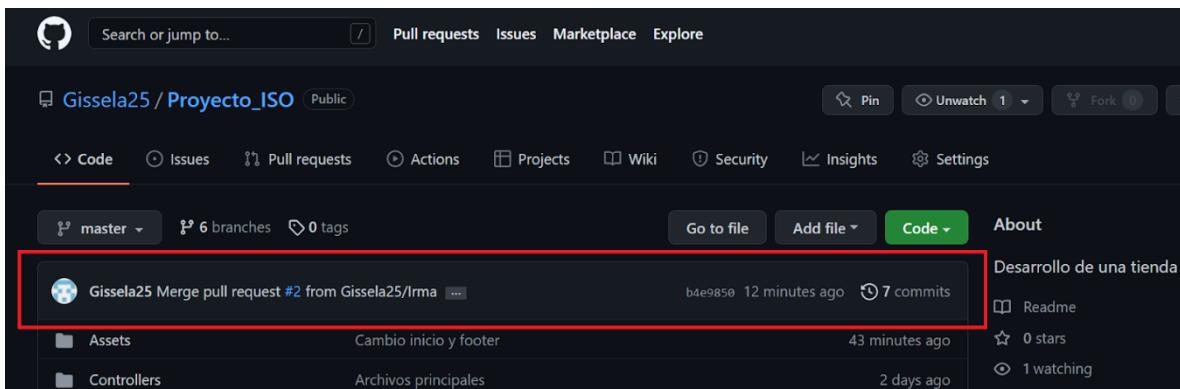


Ilustración 48.Comprobando merge

Clonación de repositorio

Antes que nada, nos aseguramos de que estén nuestros datos

```
jony2@DESKTOP-SACQ985 MINGW64 /c/wamp64/www
$ git config --global -l
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
user.name=Jony-07
user.email=jony25lopezml@gmail.com
init.defaultbranch=main

jony2@DESKTOP-SACQ985 MINGW64 /c/wamp64/www
$
```

*Ilustración 49.Revisando datos*

empezamos la clonación

```
jony2@DESKTOP-SACQ985 MINGW64 /c/wamp64/www
$ git clone https://github.com/Gissela25/Proyecto_ISO.git Proyecto_ISO
Cloning into 'Proyecto_ISO'...
remote: Enumerating objects: 17083, done.
remote: Counting objects: 100% (17083/17083), done.
remote: Compressing objects: 100% (2621/2621), done.
remote: Total 17083 (delta 14416), reused 17073 (delta 14410), pack-reused 0
Receiving objects: 100% (17083/17083), 23.06 MiB | 1.64 MiB/s, done.
Resolving deltas: 100% (14416/14416), done.
Updating files: 100% (16481/16481), done.

jony2@DESKTOP-SACQ985 MINGW64 /c/wamp64/www
$
```

*Ilustración 50.Clonando Repositorio*

y comprobamos con dir, que este haya sido clonado

```
jony2@DESKTOP-SACQ985 MINGW64 /c/wamp64/www
$ dir
add_vhost.php      DP1           IMGs_SUMERSA       LIS2          Practica6    test_sockets.php
BK_LIS             DWVA-master   IMGs_SUMERSA_backup  LPMVC        probando    test-login
bk_update          ejemploMVC   IMGs_SUMERSA_bk     MVC          Proyecto_ISO  testmysql.php
Course_PHP          EMVC          index.php         NMVC        pwd_test   wamplangues
demo_laravel       Expresiones_Regulares laravel_test  Portafolio  Sumersa    wampthemes
desafio-practico-1 favicon.ico    LIS              Practica_4  Taller2_ISO  web.html
desafio-practico-2 Guia2_DL190272 LIS_PROYECTO    Practica-5  test       WebDev

jony2@DESKTOP-SACQ985 MINGW64 /c/wamp64/www
$
```

*Ilustración 51.Revisando Clonación*

Ahora, comprobamos cuales ramas existen y nos cambiamos a la rama Jony

```
jony2@DESKTOP-SACQ985 MINGW64 /c/wamp64/www/Proyecto_ISO (master)
$ git branch
  Jony
* master

jony2@DESKTOP-SACQ985 MINGW64 /c/wamp64/www/Proyecto_ISO (master)
$ git checkout Jony
Switched to branch 'Jony'
Your branch is up to date with 'origin/Jony'.

jony2@DESKTOP-SACQ985 MINGW64 /c/wamp64/www/Proyecto_ISO (Jony)
$
```

*Ilustración 52.Cambiando de rama*

Ahora revisamos desde github desktop. En la sección current branch, aparece la rama que elegimos

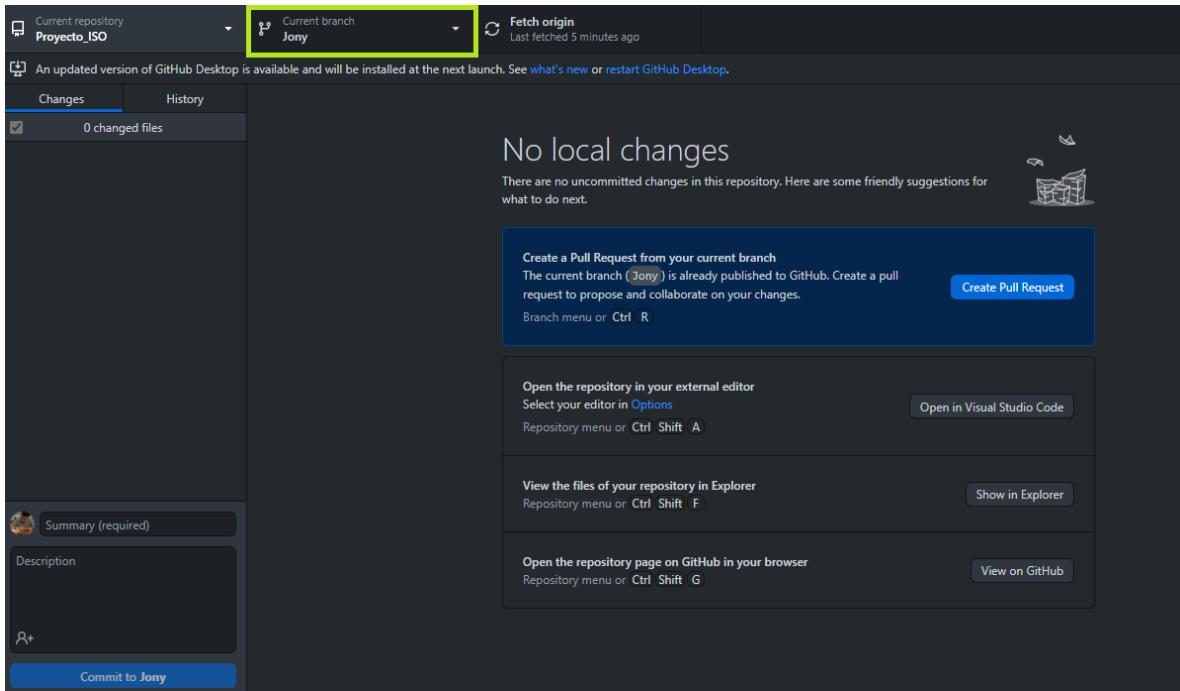


Ilustración 53.Corraborando cambio de rama desde GitHub Desktop

Ahora nos vamos a vs code:

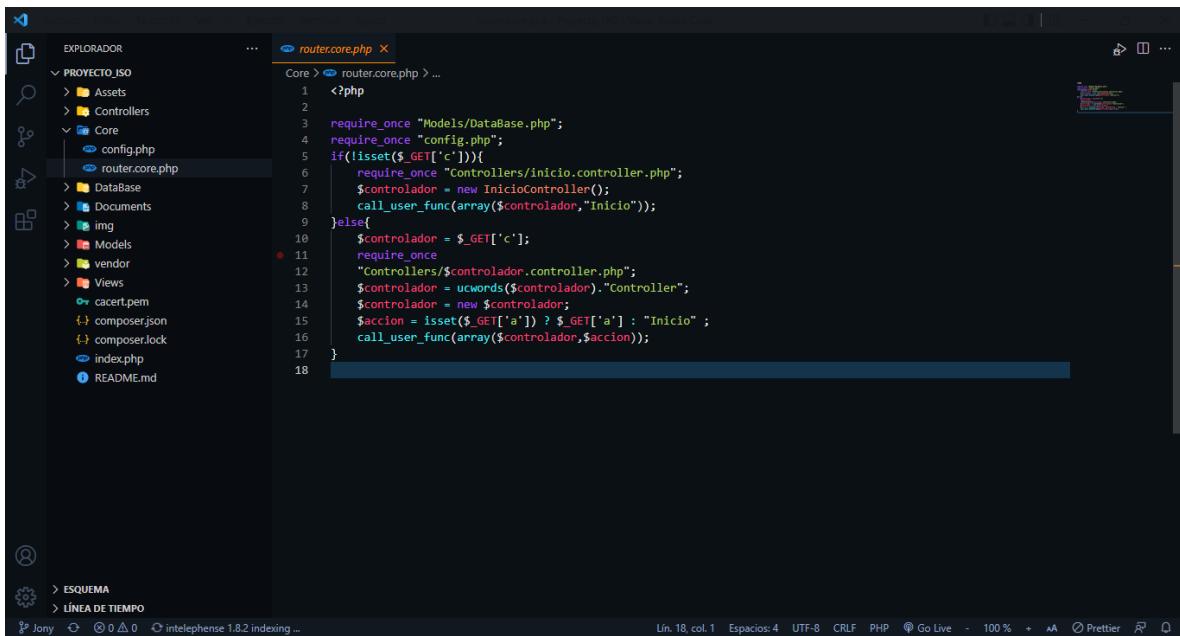


Ilustración 54.VS code

Ahora realizaremos un cambio, para la dirección de la raíz del proyecto.



Ilustración 55. Realizando cambio en el código

Ahora comprobamos el cambio con git status, realizamos el commit

```

jony2@DESKTOP-SACQ985 MINGW64 /c/wamp64/www/Proyecto_IS0 (Jony)
$ git status
On branch Jony
Your branch is up to date with 'origin/Jony'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
    (use "git restore <file>..." to discard changes in working directory)
      modified:   Core/config.php

no changes added to commit (use "git add" and/or "git commit -a")

jony2@DESKTOP-SACQ985 MINGW64 /c/wamp64/www/Proyecto_IS0 (Jony)
$ git add .

jony2@DESKTOP-SACQ985 MINGW64 /c/wamp64/www/Proyecto_IS0 (Jony)
$ git commit -m "Actualizando raíz de proyecto"
[Jony c30d1f1a] Actualizando raíz de proyecto
 1 file changed, 1 insertion(+), 1 deletion(-)

jony2@DESKTOP-SACQ985 MINGW64 /c/wamp64/www/Proyecto_IS0 (Jony)
$ 

```

Ilustración 56. Revisando estatus y realizando commit

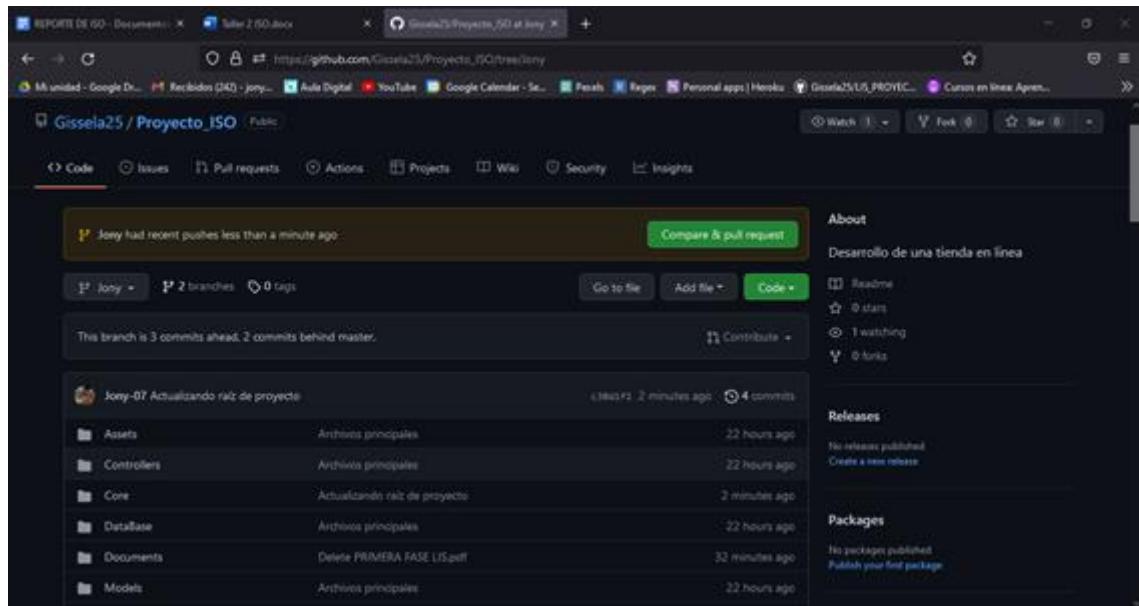
Realizamos el push, para subir al repositorio

```
jony2@DESKTOP-SACQ985 MINGW64 /c/wamp64/www/Proyecto_ISO (Jony)
$ git push origin Jony
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 408 bytes | 204.00 KiB/s, done.
Total 4 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/Gissela25/Proyecto_ISO.git
  277149f4..c30d1f1a Jony -> Jony

jony2@DESKTOP-SACQ985 MINGW64 /c/wamp64/www/Proyecto_ISO (Jony)
$
```

*Ilustración 57. Realizando Push*

Comprobamos la subida del repositorio



*Ilustración 58. Comprobando Push*

Y comprobamos la raíz del proyecto

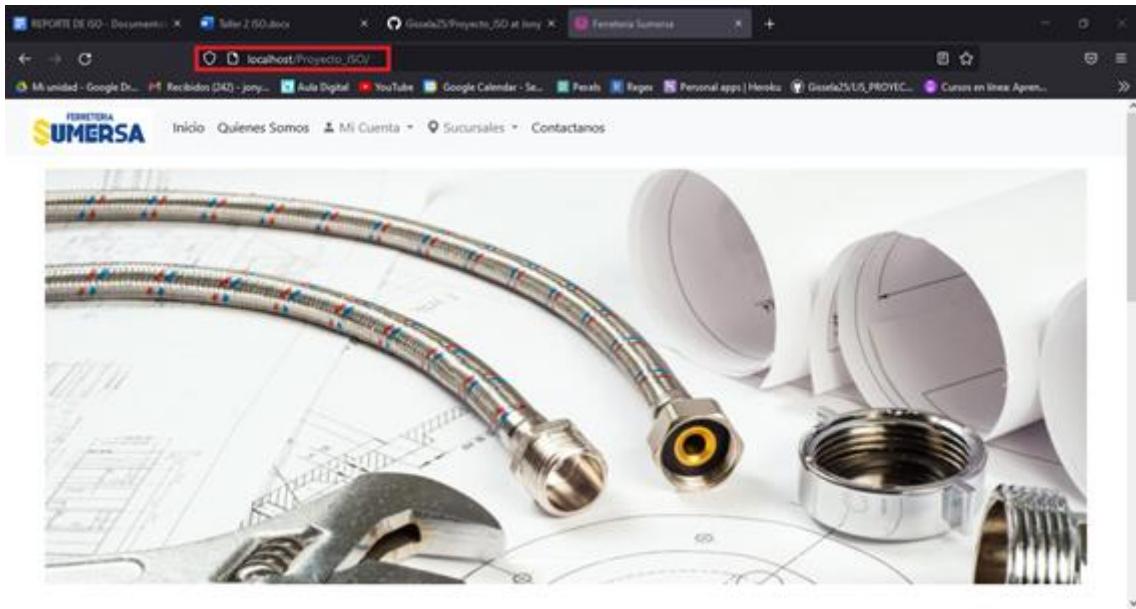


Ilustración 59.Revisando raíz del proyecto

vemos el historial de commits

```
jony2@DESKTOP-SACQ985 MINGW64 /c/wamp64/www/Proyecto_ISO (Jony)
$ git log --pretty=oneline --graph --decorate --all
* 8272ae179b5c7fc7c3211f7029533daebbb5b2d (origin/master, origin/HEAD, master) Raíz de proyecto
* ca115841815d58dc7a6c44b21812dcb7fbc68bb7 Delete PRIMERA FASE LIS.pdf
* be4ac88fa09da027981b6250053b6ba6f18e8635 Update README.md
| * c30d1f1a5417a715f71a72055937572ce9369516 (HEAD -> Jony, origin/Jony) Actualizando raíz de proyecto
| * 277149f4bb1b31357a108e25ff99375e93a390a0 Delete PRIMERA FASE LIS.pdf
| * e8fea5342f48c4bb83a8f83c6713660e44056373 Agregando Documentacion
|/
* d01a23471a37395858e4a12d6fbdbdfaccad30b68 Archivos principales

jony2@DESKTOP-SACQ985 MINGW64 /c/wamp64/www/Proyecto_ISO (Jony)
$
```

Ilustración 60.Revisando historial de commits

## 10. Avance de Software

Herramienta de Seguimiento

Tablero completo en Trello

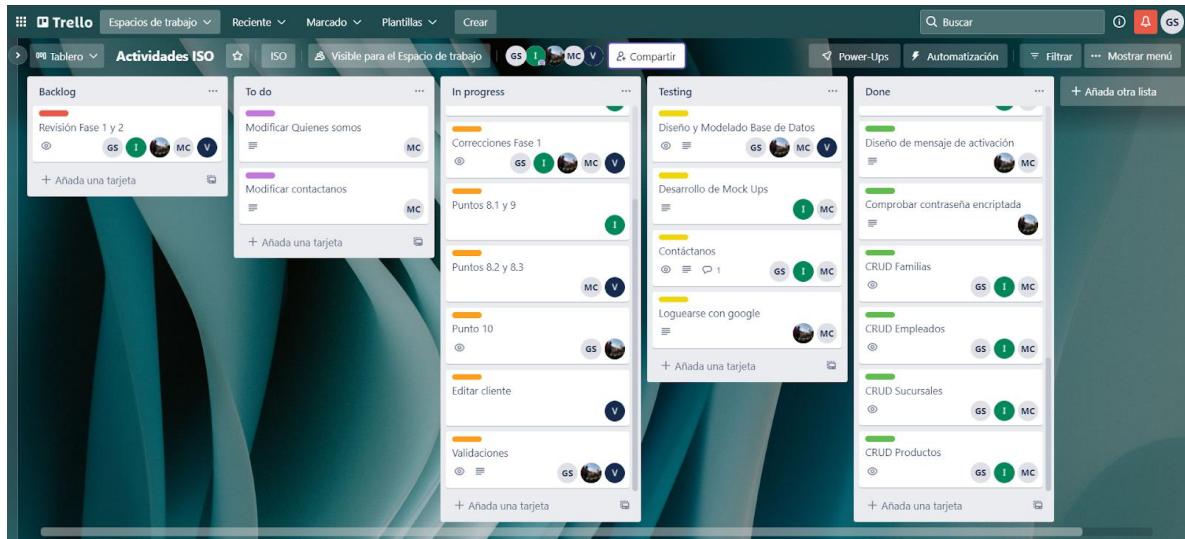


Ilustración 61.Tablero Trello

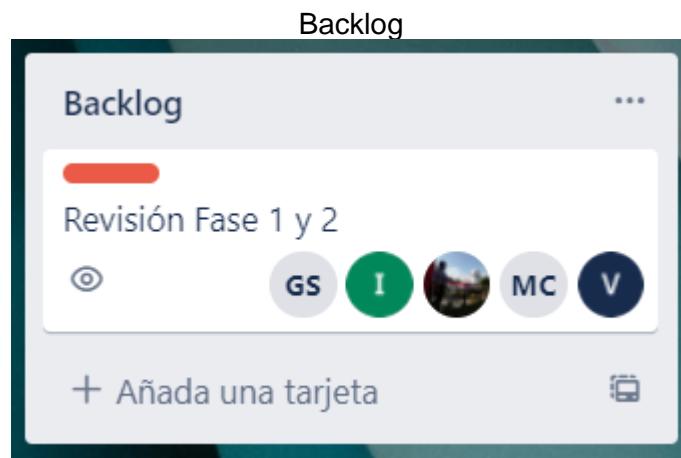


Ilustración 62.Backlog

To do



Ilustración 63. To do

In progress



Ilustración 64. In progress 1



Ilustración 65.In progress 2

### Testing

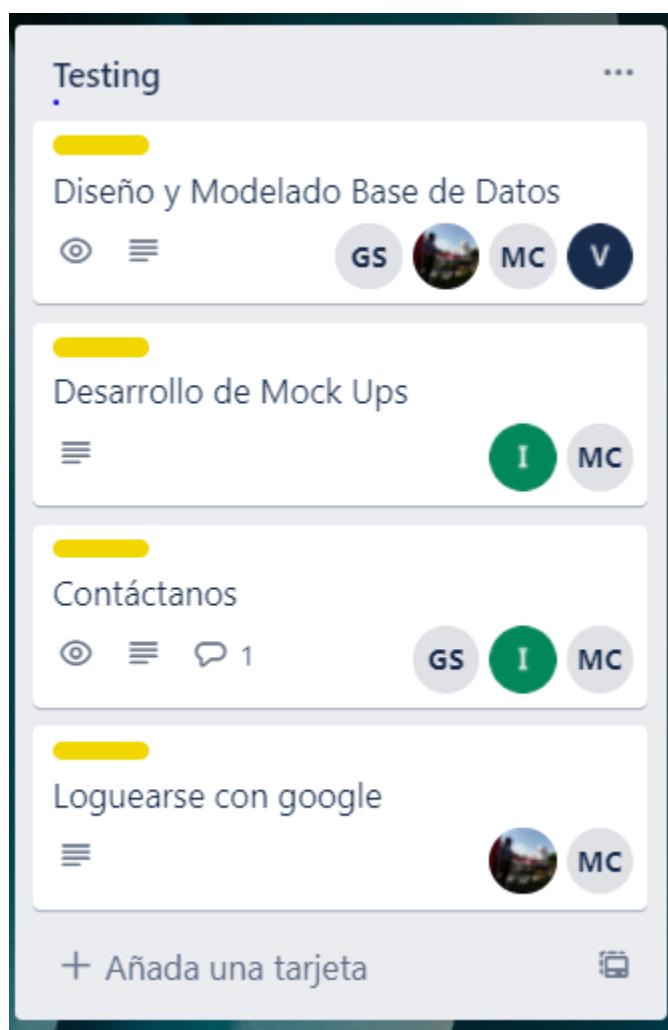


Ilustración 66.Testing

Done

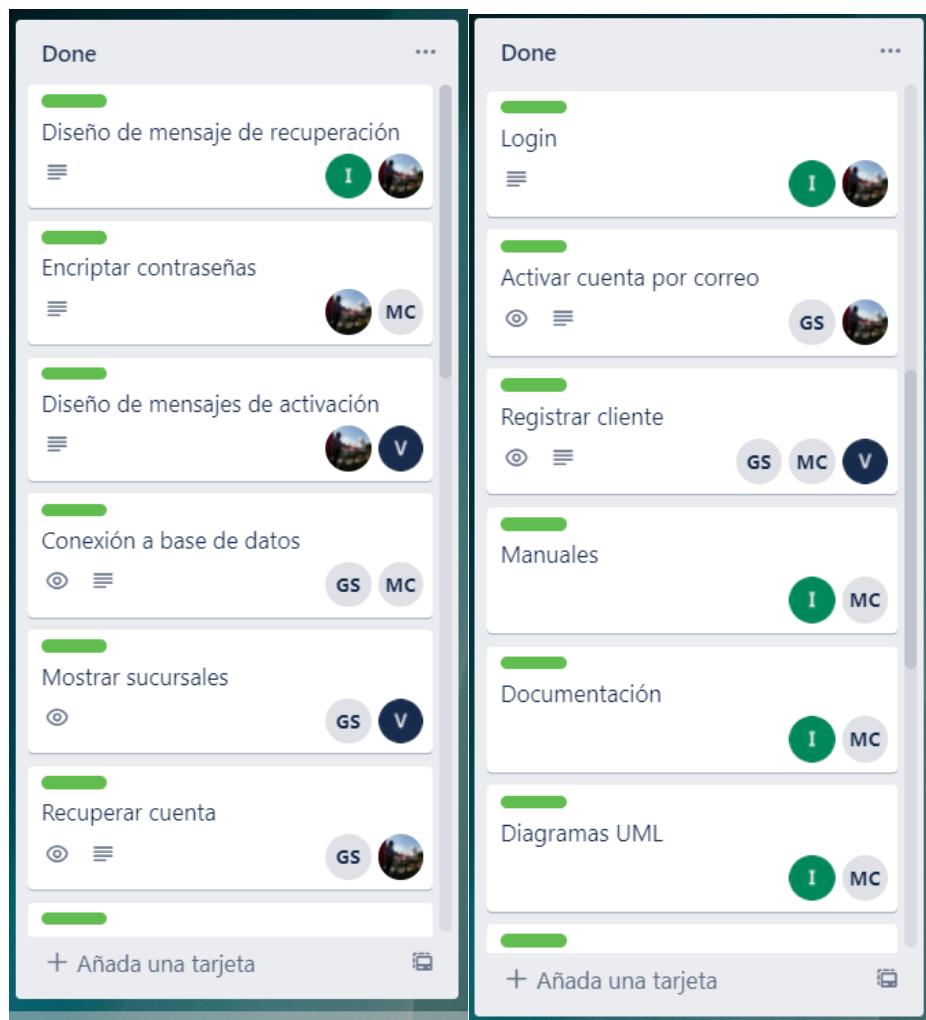


Ilustración 67.Done 1



Ilustración 68.Done 2

### Prototipo en Figma

Ilustración 69.Prototipo 1

Desktop - 1

FERRETERIA  
**SUMERSA**

**INICIAR SESIÓN**

Ingrese su correo electrónico  
Ingrese su contraseña  
¿OLVIDE MI CONTRASEÑA?

INGRESAR  
ACTIVAR CUENTA  
REGISTRAR

Desktop - 2

FERRETERIA  
**SUMERSA**

**RECUPERAR CONTRASEÑA**

Ingrese su email  
CONFIRMAR  
REGRESAR

Ilustración 70.Prototipo 2

Desktop - 3

FERRETERIA  
**SUMERSA**

**VERIFICAR CUENTA**

Ingrese su email  
CONFIRMAR  
REGRESAR

Desktop - 4

**VERIFICAR CUENTA**

DNI  
INGRESE SU DNI

NOMBRE  
INGRESE SU NOMBRE

APELLIDO  
INGRESE SU APELLIDO

NUMERO DE TELEFONO  
INGRESE SU NUMERO DE TELEFONO

DIRECCION  
INGRESE SU DIRECCION

**DATOS DE INICIO DE SESION**

CORREO  
INGRESE SU CORREO

CLAVE  
INGRESE SU CLAVE

CONFIRMAR CLAVE  
CONFIRMAR SU CLAVE

INGRESAR

Ilustración 71.Prototipo 3

Branch

FERRETERIA  
**SUMERSA**

INICIO SUCURSALES EMPLEADOS FAMILIA PRODUCTOS

INSERTAR

ID	NOMBRE	ACCIONES
1	SAN SALVADOR	EDITAR
2	SANTA TECLA	EDITAR
3	LOURDES	EDITAR
4	OPGO	EDITAR
5	ZARAGOZA	EDITAR
6	SANTITA ANA	EDITAR

INSERTAR BRANCH

**INGRESAR SUCURSAL**

NOMBRE SUCURSAL  
ENVIAR

Ilustración 72.Prototipo 5

ACTUALIZAR BRANCH

FERRETERIA SUMERSA

ACTUALIZAR SUCURSAL

NOMBRE SUCURSAL

Desktop - 5

USUARIO ACTIVO

ID USUARIO	NOMBRE	APELLIDO	CORREO	SUCURSAL	ACCIONES
					EDITAR

USUARIO INACTIVO

ID USUARIO	NOMBRE	APELLIDO	CORREO	SUCURSAL	ACCIONES
					EDITAR

Ilustración 73. Prototipo 6

INSERTAR EMPLEADOS

FERRETERIA SUMERSA

INGRESAR EMPLEADO

ID USUARIO

NOMBRE

APELLIDO

CORREO

CLAVE

SUCURSAL

Desktop - 6

FERRETERIA SUMERSA

ACTUALIZAR EMPLEADO

ID USUARIO

NOMBRE

APELLIDO

CORREO

SUCURSAL

ESTADO

ACTIVO

INACTIVO

Ilustración 74. Prototipo 7

FAMILIA

FERRETERIA SUMERSA

INICIO SUCURSALES EMPLEADOS FAMILIA PRODUCTOS

INSERTAR

ID	FAMILIA	ACCIONES
F094	FERRETERIA	EDITAR ELIMINAR
F153	TECHO	EDITAR ELIMINAR
F251	PINTURA	EDITAR ELIMINAR
F315	FONTANERIA	EDITAR ELIMINAR
F856	ELECTRICO	EDITAR ELIMINAR
F869	CONSTRUCCION	EDITAR ELIMINAR

Desktop - 7

FERRETERIA SUMERSA

INGRESAR FAMILIA

ID FAMILIA

FAMILIA

Ilustración 75. Prototipo 8



Ilustración 76. Prototipo 9

Desktop - 10

The screenshot displays a user interface for a hardware store. On the left, there is a grid of four product cards: "Lámpara" (Price: 226 \$), "Hierro" (Price: 2.26 \$), "Pintura" (Price: 33.00 \$), and "cinta métrica" (Price: 2.50 \$). Each card includes a "Ver más" button. On the right, there is a detailed view of a product named "mortero". The view includes the product code (PROD0047), name (mortero), description (MEZCLA MORTERO LISTO PARA UTILIZAR), image (a stack of bags of mortar), family (Construction), price (\$6.70), and stock (33).

Código Producto	Nombre	Descripción	Imagen	Familia	Precio	Existencias
PROD0047	Mortero	MEZCLA MORTERO LISTO PARA UTILIZAR.		Construction	\$6.70	33

Ilustración 77.Prototipo 10

Ilustración 78.Prototipo 11

Ilustración 79.Prototipo 12

### Diagramas UML

En los diagramas de caso y de secuencia por parte del consumidor vemos el proceso con la siguiente lógica:

El cliente ingresará a la página web de la ferretería, tendrá la información de las sucursales, de la empresa y podrá ingresar desde la barra de navegación a la sucursal deseada, posteriormente saldrán los artículos que están en la sucursal de la ferretería seleccionada, elegimos el artículo y se detallará la información de este, seguido de la disponibilidad en la sucursal y la opción de agregar al carrito, luego de agregar todos los productos a comprar, se podrá ingresar al carrito y se pedirá iniciar sesión o registrar, si aún no se cuenta con las credenciales; después de ingresar la información, se pedirá el método de entrega, el cual puede ser retirar en la sucursal o que se envíe a domicilio, en seguida se debe seleccionar el método de pago, el cual puede ser en efectivo o por transferencias bancarias, para finalizar se debe enviar el pedido y el sistema mandará un ticket o

comprobante de la compra el cual será recibido por los empleados de la sucursal seleccionada, en lo cual ellos se encargarán de tener listo el pedido o enviarlo con un repartidor hasta el domicilio.

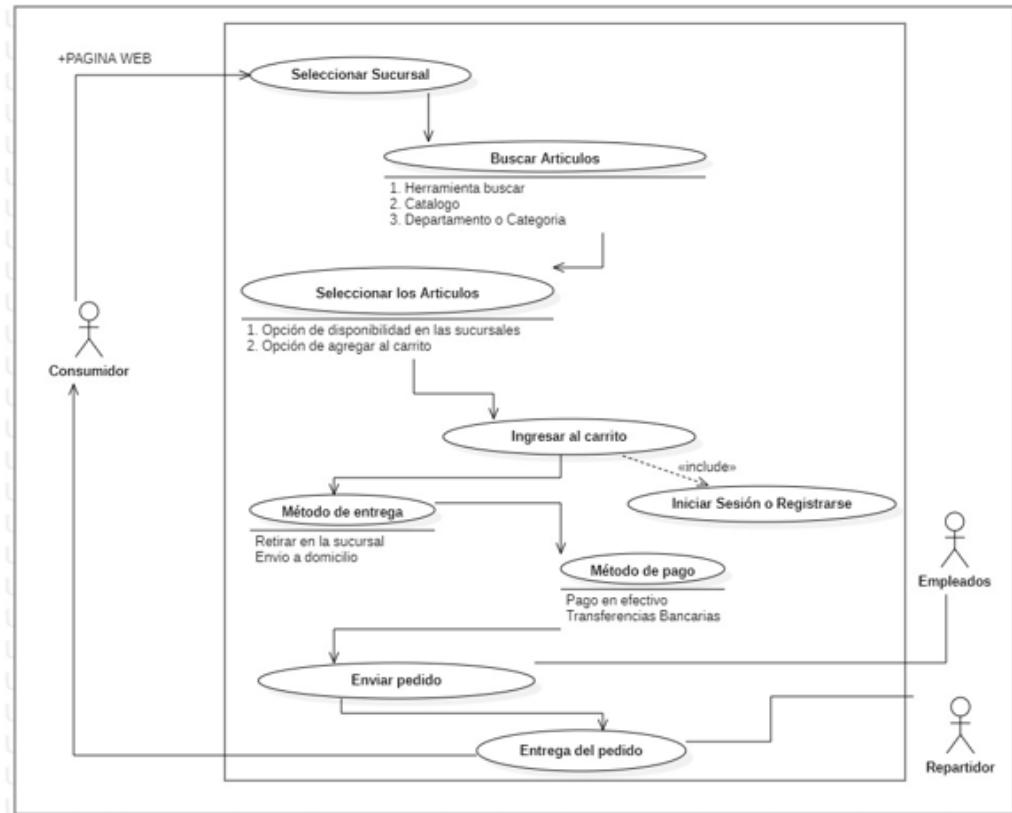


Ilustración 80.UML 1

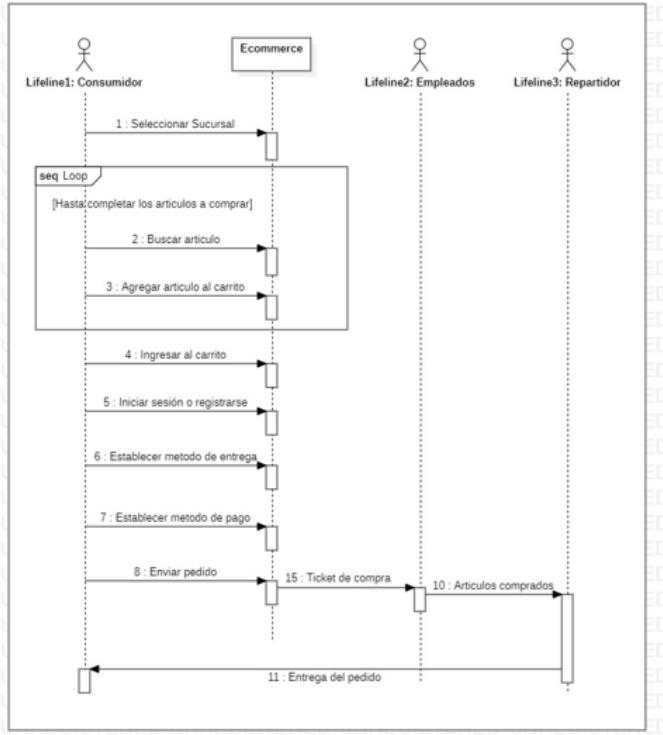


Ilustración 81.UML 2

En los diagramas de caso y de secuencia por parte del administrador vemos el proceso con la siguiente lógica:

El administrador podrá registrar las sucursales de la ferretería y registrar los empleados que existen en dichas sucursales; tanto el administrador como los empleados tienen la facultad de poder ingresar los artículos disponibles en las diferentes sucursales.

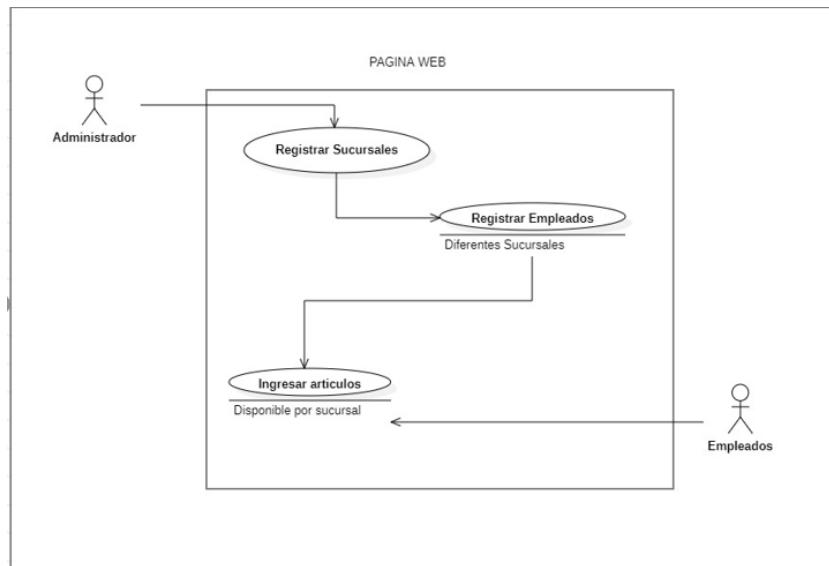


Ilustración 82.UML 3

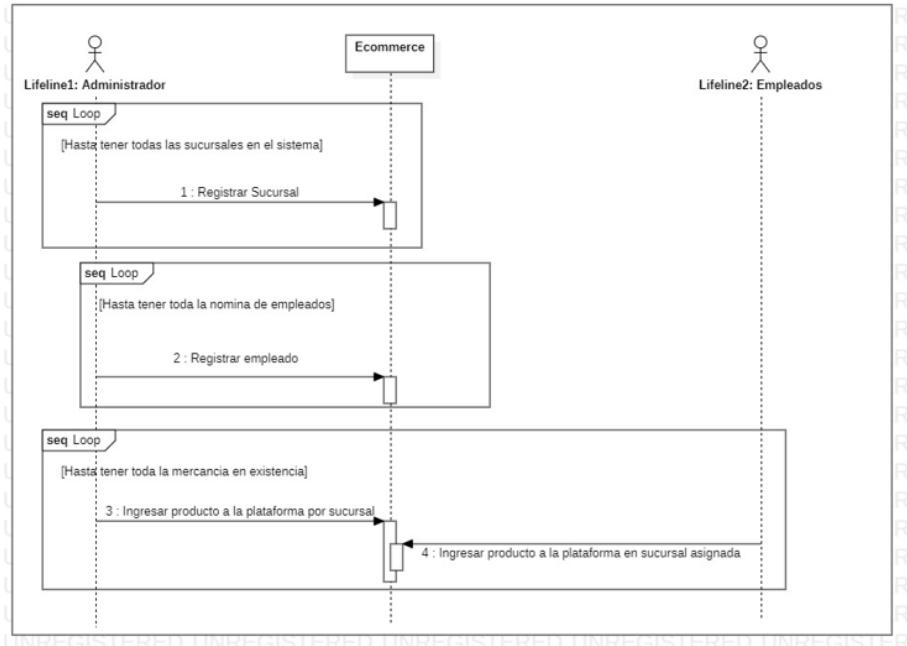


Ilustración 83.UML 4

## 11. Fuentes de Información

- IntegralIT. (2021, 11 noviembre). Conoce los principales roles de Scrum y sus responsabilidades. Integra IT Soluciones. Recuperado 20 de febrero de 2022, de <https://integrait.com.mx/blog/roles-de-scrum/#:~:text=Conoce%20los%203%20roles%20principales%20de%20Scrum.&text=Es%C3%A1%20conformado%20por%203%20roles,Miembros%20del%20Equipo%20de%20Desarrollo>.
- ¿Sabes cuáles son los Requisitos para instalar Safari? (Recuperado el día 20 de febrero de 2022) de <https://siaguanta.com/c-navegadores/requisitos-para-instalar-safari/>
- Epitech. (2021, mayo 19) ¿Qué es PHP y para qué sirve este lenguaje de código abierto?. Recuperado el día 20 de febrero de 2022, de <https://www.epitech-it.es/que-es-php/#:~:text=El%20lenguaje%20PHP%2C%20es%20un,y%20a%20la%20interfaz%20del%20usuario>
- PADAWAN. Definición, características y nuevas etiquetas de HTML5. Recuperado el día 20 de febrero de 2022, de <http://www.elpadawan.com/html/definicion-caracteristicas-y-nuevas-etiquetas-de-html5>
- IEP. ¿Qué es el Ecommerce?. Recuperado el día 20 de febrero de 2022, de <https://www.iep.edu.es/que-es-el-ecommerce/#:~:text=El%20Ecommerce%20consiste%20en%20el,pago%20o%20los%20aspectos%20legales>
- Conecta Software. Aspectos clave de una Tienda online. Recuperado el día 20 de febrero de 2022, de <https://conectasoftware.com/tienda-online/aspectos-clave-de-una-tienda-online/#:~:text=A%20diferencia%20de%20una%20web,imagen%20corporativa%20de%20la%20empresa>

- Y. Fernández. (2019, octubre 2019). ¿Qué es GitHub y qué es lo que ofrece a los desarrolladores?. Recuperado el día 20 de febrero de 2022, de <https://www.xataka.com/basics/que-github-que-que-le-ofrece-a-desarrolladores>
- M. Romain. Sobre el Scrum Master y el Product Owner. Recuperado el día 20 de febrero de 2022, de <https://www.media.thiga.co/es/scrum-master-y-product-owner>
- Gustavo B. (Dic 14, 2020). Qué es Wamp (Guía para principiantes). Recuperado el día 20 de febrero de 2022, de <https://www.hostinger.es/tutoriales/que-es-wamp>
- (Universidad de Alicante). Modelo Vista Controlador (MVC). Recuperado el día 22 de abril de 2022 de <https://si.ua.es/es/documentacion/asp-net-mvc-3/1-dia/modelo-vista-controlador-mvc.html>

## 12. Glosario de Términos

- **Bootstrap:** es una biblioteca multiplataforma o conjunto de herramientas de código abierto para el diseño de sitios y aplicaciones web
- **Controlador:** Actúa como intermediario entre el modelo y la Vista, gestionando el flujo de información entre ellos y las transformaciones para adaptar los datos a las necesidades.
- **Ecommerce:** Sistema de marketing y venta de productos a través de Internet.
- **Git:** Es un software de control de versiones, pensando para la eficiencia, confiabilidad y compatibilidad del mantenimiento
- **GitHub:** Es un sitio que permite alojar proyectos utilizando el sistema de control de versiones Git.
- **JavaScript:** Es un lenguaje de programación, que se utiliza para añadir características interactivas a los sitios web.
- **Modelo:** Es hace referencia a la representación de los datos que maneja el sistema, su lógica de negocio, y sus mecanismos de persistencia.
- **MVC:** Modelo Vista Controlador, es un estilo de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.
- **PHP:** Es un lenguaje de código abierto de uso general, que se adapta especialmente al desarrollo web del lado del servidor
- **Product Owner:** Es un miembro del equipo Scrum responsable de maximizar el valor del producto entregado por el equipo.
- **Scrum Master:** Entrenador y facilitador de equipos Scrum
- **Visual Studio Code:** Es un editor de código fuente desarrollado por Microsoft para Windows, Linux, macOs y Web.
- **Vista:** Es la interfaz del usuario, que compone la información que se envía al cliente y los mecanismos de interacción con éste.
- **WampServer:** Es un entorno de desarrollo web para Windows que permite crear aplicaciones web con Apache, PHP y bases de datos MySQL database

## 13. Anexos

Características de los lenguajes de programación a utilizar

### **HTML**

- Utilizado para el maquetado de sitios web
- Se puede visualizar desde cualquier plataforma
- Estable y en continua mejora

### **CSS**

- Utilizado para mostrar el contenido de los sitios web de forma más interactiva
- Se puede visualizar desde cualquier plataforma
- Estable y en continua mejora

### **PHP**

- Es apto para incrustar el lenguaje HTML
- Permite realizar conexiones con bases de datos
- Favorece la conexión entre el servidor y el cliente

### **JavaScript**

- Utilizado para añadir características interactivas a los sitios web
- Es capaz de crear sliders que se adapten a cualquier pantalla

Herramientas por utilizar

- Visual Studio Code
- GitHub
- Git
- WampServer

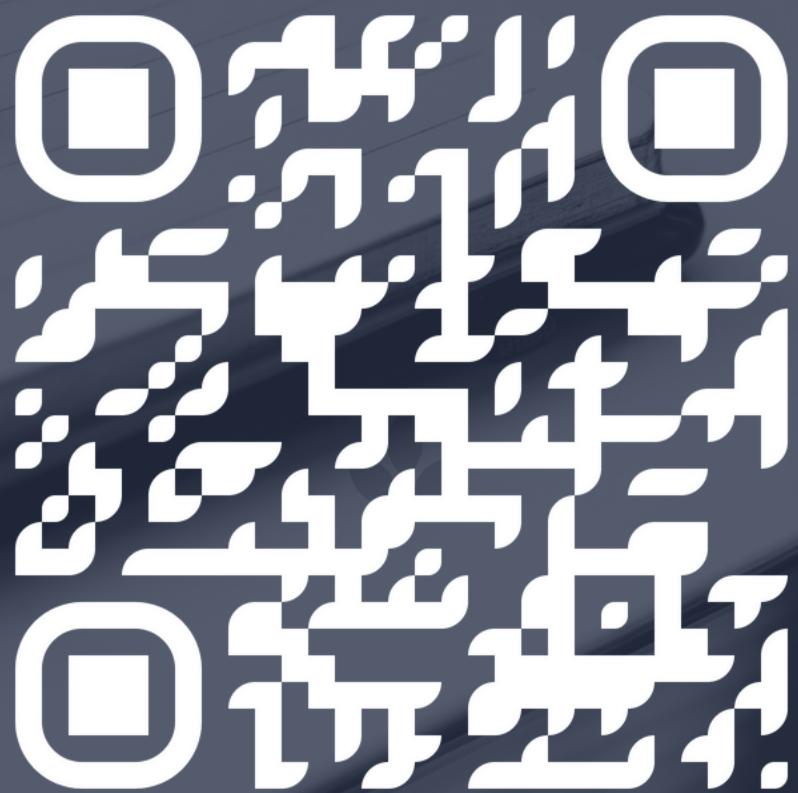
Enlace de Manual de Usuario:

<https://drive.google.com/file/d/1WFeemL3Y34f5xwdnllqLpvpA9zkMKU1P/view?usp=sharing>

Enlace de Manual de Técnico:

<https://drive.google.com/file/d/1w0KBfgtWMGMKWuPmUBD4L6avVpfIg7yh/view?usp=sharing>

# LINK DE VIDEO



[https://drive.google.com/file/d/  
1gVrzlJioqBthX-  
Q7DejNEGbzSndVmrs/view?  
usp=sharing](https://drive.google.com/file/d/1gVrzlJioqBthX-Q7DejNEGbzSndVmrs/view?usp=sharing)