



**UNIVERSIDAD NACIONAL DE TRUJILLO**

**FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS**

**ESCUELA PROFESIONAL DE INFORMÁTICA**

**INFORME DE PRÁCTICAS PREPROFESIONALES  
REALIZADO EN EL ÁREA DE DESARROLLO DE  
SISTEMAS DE LA OFICINA DE TECNOLOGÍAS DE  
INFORMACIÓN**

**AUTOR(A):**

**REBAZA AGUILAR, MARCO JAVIER**

**ASESOR:**

**Ms. Anthony José Gómez Morales**

**Trujillo - Perú  
2023**

## ÍNDICE DE CONTENIDO

ÍNDICE DE CONTENIDO.....	i
ÍNDICE DE TABLAS .....	iv
ÍNDICE DE FIGURAS.....	v
RESUMEN.....	viii
CAPÍTULO I: DE LA EMPRESA .....	1
1.1. Descripción De La Empresa:.....	1
1.1.1. Generalidades:.....	1
1.1.2. Breve Descripción .....	1
1.1.3. Misión.....	1
1.1.4. Visión.....	1
1.1.5. Organigrama .....	2
1.2. Descripción de la Unidad donde se realiza la Práctica (Funciones) ...	2
1.2.1. Funciones .....	2
CAPÍTULO II: DE LAS PRÁCTICAS.....	4
2.1. Semana 01 (28/08/2023 – 01/09/2023).....	4
2.1.1. Objetivos semanales .....	4
2.1.2. Descripción.....	4
2.1.3. Resultados .....	5
2.2. Semana 02 (04/09/2023 – 08/09/2023).....	5
2.2.1. Objetivos semanales .....	5
2.2.2. Descripción.....	5
2.2.3. Resultados .....	12
2.3. Semana 03 (11/09/2023 – 15/09/2023).....	12
2.3.1. Objetivos semanales .....	12
2.3.2. Descripción.....	12
2.3.3. Inconvenientes .....	16
2.3.4. Resultados .....	16
2.4. Semana 04 (18/09/2023 – 22/09/2023).....	16
2.4.1. Objetivos semanales .....	16
2.4.2. Descripción.....	16
2.4.3. Resultados .....	22
2.5. Semana 05 (25/09/2023 – 29/09/2023).....	23

2.5.1. Objetivos semanales .....	23
2.5.2. Descripción .....	23
2.5.3. Resultados .....	24
2.6. Semana 6 (02/10/2023 – 06/10/2023).....	25
2.6.1. Objetivos semanales .....	25
2.6.2. Descripción .....	25
2.6.3. Resultados .....	28
2.7. Semana 7 (09/10/2023 – 13/10/2023).....	29
2.7.1. Objetivos semanales .....	29
2.7.2. Descripción .....	29
2.7.3. Resultados .....	31
2.8. Semana 8 (16/10/2023 – 20/10/2023).....	31
2.8.1. Objetivos semanales .....	31
2.8.2. Descripción .....	31
2.8.3. Resultados .....	32
2.9. Semana 9 (23/10/2023 – 27/10/2023).....	33
2.9.1. Objetivos semanales .....	33
2.9.2. Descripción .....	33
2.9.3. Resultados .....	34
2.10. Semana 10 (30/10/2023 – 03/11/2023).....	35
2.10.1. Objetivos semanales .....	35
2.10.2. Descripción .....	35
2.10.3. Resultados .....	36
2.11. Semana 11 (06/11/2023 – 10/11/2023).....	37
2.11.1. Objetivos semanales .....	37
2.11.2. Descripción .....	37
2.11.3. Resultados .....	38
2.12. Semana 12 (13/11/2023 – 17/11/2023).....	39
2.12.1. Objetivos semanales .....	39
2.12.2. Descripción .....	39
2.12.3. Resultados .....	41
2.13. Semana 13 (20/11/2023 – 24/11/2023).....	41
2.13.1. Objetivos semanales .....	41
2.13.2. Descripción .....	42
2.13.3. Resultados .....	44
2.14. Semana 14 (27/11/2023 – 01/12/2023).....	45

2.14.1. Objetivos semanales.....	45
2.14.2. Descripción.....	45
2.14.3. Resultados.....	47
CONCLUSIONES .....	48
REFERENCIAS BIBLIOGRÁFICAS.....	49
ANEXOS.....	50

## ÍNDICE DE TABLAS

Tabla 1. Repartición de roles de acuerdo a la metodología.....	6
--	---

## ÍNDICE DE FIGURAS

Figura 1. Organigrama de la Oficina de Tecnologías de Información.....	2
Figura 2. D.P. Administrador y Trabajador – Ingresar al Sistema .....	7
Figura 3. D.P. Administrador y Trabajador – Cambiar contraseña .....	8
Figura 4. D.P. Administrador - Buscar resolución y/o actas. ....	8
Figura 5. D.P. Administrador o trabajador(publicador) - Subir resolución y/o acta..	8
Figura 6. Historia de Usuario - Acceder a las actas y resoluciones.....	9
Figura 7. Historia de Usuario - Tener un mantenedor de archivos.....	10
Figura 8. Historia de Usuario - Acceso a la configuración de perfil.....	10
Figura 9. Historia de Usuario - Tener un mantenedor de Órganos.....	10
Figura 10. Historia de Usuario - Tener un mantenedor de archivos de facultad.....	11
Figura 11 Plan del proyecto.....	11
Figura 12. Diagrama de proceso.....	12
Figura 13. Plataforma online Lucidchart .....	14
Figura 14. Diagrama del modelo entidad-relación de la base de datos .....	15
Figura 15. Diagrama del modelo físico de la base de datos .....	15
Figura 16. XAMPP .....	17
Figura 17. Composer.....	18
Figura 18. Node.js.....	18
Figura 19. GIT.....	19
Figura 20. Visual Studio Code.....	20
Figura 21. Consola git bash.....	21
Figura 22. Carpeta de htdocs desde git bash.....	21
Figura 23. Crear proyecto Formulario-app.....	22
Figura 24. Carpeta htdocs .....	22
Figura 25. Base de datos del sistema resoluciones.....	23
Figura 26. Diagrama entidad-relación para el sistema de resoluciones.....	24
Figura 27. Actualización del plan del proyecto del sprint 1 (MBD).....	25
Figura 28. Consulta para obtener la data para la tabla persona.....	25

Figura 29. Consulta para obtener la data para la tabla organo.....	26
Figura 30. Consulta para obtener la data para la tabla documentos.....	26
Figura 31. Consulta para obtener la data para la tabla tipo_documento.....	27
Figura 32. Consulta para obtener la data para la tabla subtipo_documento.....	27
Figura 33. Consulta para obtener la data para la tabla cargo.....	27
Figura 34. Tablas del sistema resoluciones subidas en DBeave.....	28
Figura 35. Actualización del plan del proyecto del sprint 1.....	28
Figura 36. Arquitectura de NestJS.....	29
Figura 37. Endpoints de autenticación documentado en Swagger.....	31
Figura 38. CRUD para la tabla “documento_tipo” documentado en Swagge.....	31
Figura 39. CRUD para la tabla “documento_jerarquia” documentado en Swagge.....	32
Figura 40. CRUD para la tabla “órgano” documentado en Swagge.....	32
Figura 41. CRUD para la tabla “multitabla” documentado en Swagger.....	32
Figura 42. Actualización del plan del proyecto del sprint 1 finalizado.....	33
Figura 43. CRUD para el manejo de archivos documentado en Swagger.....	34
Figura 44. CRUD para la tabla “documento” documentado en Swagger.....	34
Figura 45. Actualización del plan del proyecto del sprint 2 en progres.....	34
Figura 46. Actualizar ruta content/ .....	36
Figura 47. Actualizar ruta actas_au_cu/.....	36
Figura 48. Actualizar ruta actas/.....	36
Figura 49. Actualización del plan del proyecto del sprint 2(HU-1 Y HU-2).....	36
Figura 50. Diseño de la interfaz del buscador de resoluciones y actas.....	37
Figura 51. Implementación de la interfaz del buscador de resoluciones y actas.....	38
Figura 52. Actualización del plan del proyecto del sprint 2(HU-1).....	39
Figura 53. Diseño de la interfaz del mantenedor de archivos.....	40
Figura 54. Implementación de la interfaz del mantenedor de archivos.....	40
Figura 55. Actualización del plan del proyecto del sprint 2 finalizado.....	41
Figura 56. Diseño de la interfaz del perfil de usuario.....	42
Figura 57. Diseño de la interfaz de lista de usuarios.....	42

Figura 58. Implementación de la interfaz perfil de usuario.....	43
Figura 59. Implementación de la interfaz lista de usuarios.....	43
Figura 60. Actualización del plan del proyecto del sprint 3 en progreso.....	44
Figura 61. Diseño de la interfaz del mantenedor de órganos.....	44
Figura 62. Implementación de la interfaz del mantenedor de órganos.....	46
Figura 63. Actualización del plan del proyecto del sprint 3 finalizado.....	46



## **RESUMEN**

Durante mis prácticas preprofesionales, asumí un papel integral en el desarrollo del sistema de resoluciones, abarcando tanto el backend como el frontend, así como la crucial tarea de migración de datos. En el backend, enfrenté desafíos significativos al optimizar el rendimiento y asegurar la seguridad y escalabilidad del sistema. Participé activamente en la migración de datos así como en el modelo de la base de datos, garantizando una transición sin problemas. En el frontend, me centré en diseñar interfaces intuitivas y amigables, incorporando funcionalidades modernas para una experiencia de usuario óptima. Esta experiencia integral me permitió fortalecer mis habilidades técnicas y destacar la importancia de la colaboración entre el frontend y el backend para lograr un producto final coherente y eficiente.

## **CAPÍTULO I: DE LA EMPRESA**

### **1.1. Descripción De La Empresa:**

#### **1.1.1. Generalidades:**

- **Razón social:** UNIVERSIDAD NACIONAL DE TRUJILLO.
- **RUC:** 20172557628
- **Dirección física:** Av. Juan Pablo II S/N Urb. San Andrés Trujillo – La Libertad
- **URL:** <https://dsc.unitru.edu.pe>
- **Gerente general o equivalente:** Edwin Gerson Monzón Llempen

#### **1.1.2. Breve Descripción**

Órgano de apoyo encargado de planificar, implementar y gestionar sistemas de información, infraestructura tecnológica, de cómputo y de comunicaciones de la Universidad Nacional de Trujillo.

#### **1.1.3. Misión**

Dirigir el Sistema Institucional de Informática, con ética y talento humano experto; liderando el Gobierno, la Transformación Digital, y la Modernización de la Gestión Pública, para beneficio de la comunidad universitaria.

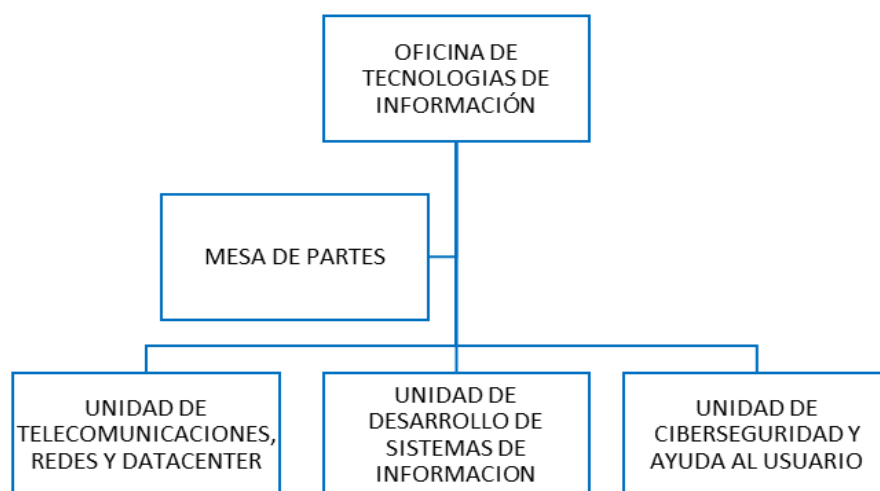
#### **1.1.4. Visión**

Al 2024, la Dirección de Sistemas y Comunicaciones destaca por su liderazgo en Gobierno y Transformación Digital, impulsando la excelencia académica, organizacional y la producción científica de impacto.

### 1.1.5. Organigrama

**Figura 1**

*Organigrama de la Oficina de Tecnologías de Información*



### 1.2. Descripción de la Unidad donde se realiza la Práctica (Funciones)

La unidad donde se realizó las prácticas es la unidad de Desarrollo de Sistemas de Información, la cual se encarga de coordinar y ejecutar los procesos, procedimientos y actividades relacionados al desarrollo, actualización e implementación de los aplicativos informáticos de la Universidad. Esto incluye el diseño y elaboración con las diferentes herramientas de tecnología para la aplicación de la creación de los sistemas.

#### 1.2.1. Funciones

- **Programar:** Coordinar y ejecutar los procesos, procedimientos y actividades relacionados al desarrollo, actualización e implementación de los aplicativos informáticos de la Universidad.
- **Desarrollar:** Uniformizar los aplicativos informáticos en el ámbito de la gestión académica y administrativa de la Universidad.
- **Coordinar:** Desarrollar los proyectos y sistemas en proceso, considerando las necesidades de información de las unidades orgánicas, de conformidad con los principios técnicos y

metodológicos que dicta la administración de proyectos informáticos.

- **Validar:** Los prototipos, formatos y sistemas funcionales como las soluciones viables a las necesidades informáticas de la institución, mediante la continua interacción con las áreas usuarias; generando la documentación técnica y los respectivos manuales de cada sistema.
- **Capacitar:** A los usuarios respecto a la operatividad de la nueva tecnología y las aplicaciones desarrolladas, así como brindar la asistencia técnica oportuna en los sistemas implantados.
- **Proponer:** La automatización de procesos que faciliten el uso de servicios universitarios a la población estudiantil, docente y administrativa de la Universidad.
- **Diseñar:** Desarrollar mecanismos de auditoría de la información de los sistemas desarrollados, para garantizar su consistencia e integridad.
- **Implementar:** Ejecutar y mantener actualizado el Sistema de Control Interno en el marco de las funciones y competencias de su unidad operativa.
- **Otros:** Funciones de su competencia y las establecidas por normas y dispositivos legales vigentes.

## CAPÍTULO II: DE LAS PRÁCTICAS

### 2.1. Semana 01 (28/08/2023 – 01/09/2023):

#### 2.1.1. Objetivos semanales:

- Buscar lugar de prácticas.
- Tener conocimiento sobre el trabajo que se realizará a lo largo del proceso de prácticas.
- Conocer el ambiente laboral.
- Recolectar información para entender el modelo de la base de datos.

#### 2.1.2. Descripción:

Durante esta semana se buscó el lugar donde se realizarán las practicas preprofesionales, el cual se pudo encontrar satisfactoriamente en la Oficina de Tecnologías de la Información en el área de Unidad de Desarrollo de Sistemas de información, el cual está ubicado en el interior de la ciudad universitaria de la Universidad Nacional de Trujillo.

Se nos asignó un jefe de proyectos el cual nos guiará en el proceso del desarrollo y mejoramiento de un nuevo sistema ya existente llamado “**Resoluciones**”, el cual nos brindó las herramientas informáticas a usar.

##### 2.1.2.1. Herramientas informáticas a usar:

Las herramientas informáticas fueron brindadas mediante un correo electrónico por el jefe de proyectos, el cual se puede evidenciar en el anexo 1, los cuales son.

- **Frameworks:** Quasar v1 y NestJS
- **Base de datos:** PostgreSQL
- **Repositorios:** GitLab
- **Editor de código:** Visual Studio Code

Cabe destacar que estas no son todas las herramientas que se usarán en el transcurso.

#### **2.1.2.2. Ambiente Laboral:**

También fue presentado el ambiente laboral donde se desarrollarán las prácticas, así como los compañeros que nos acompañarán y nos apoyarán en el desarrollo y mejoramiento del nuevo sistema, el cual se evidencia en el anexo 2.

#### **2.1.2.3. Limitaciones:**

- Falta de acceso a computadoras portátiles o equipos de desarrollo en el entorno de trabajo.

#### **2.1.3. Resultados:**

- Se logró encontrar el lugar donde se realizarán las practicas preprofesionales.
- Se conoció el sistema a desarrollar.
- Se conoció las herramientas informáticas a usar.
- Se conoció el ambiente laboral.
- Se conoció al equipo de desarrollo.
- No se pudo recolectar información para entender el modelo de la base de datos porque no se nos dio acceso.

### **2.2. Semana 02 (04/09/2023 – 08/09/2023):**

#### **2.2.1. Objetivos semanales:**

- Elegir una metodología informática a utilizar en el desarrollo el nuevo sistema, así como la distribución de roles.
- Recabar información sobre el funcionamiento del sistema actual.
- Elaborar las historias de usuario de acuerdo a la metodología informática usada

#### **2.2.2. Descripción:**

Durante esta semana se realizó se realizó varias tareas, las cuales fueron apoyadas por algunas herramientas informáticas, las cuales, tanto como las herramientas y las tareas, se detallan a continuación:

#### 2.2.2.1. Herramientas informáticas:

Para esta semana se utilizó las siguientes herramientas informáticas:

- Microsoft Excel
- Bizagi Modeler

#### 2.2.2.2. Metodología y distribución de roles del proyecto:

La repartición de los roles se presenta en la Tabla 1, los cuales son fundamentales para la aplicación adecuada de la metodología SCRUM.

**Tabla 1**

*Repartición de roles de acuerdo a la metodología*

<b>Roles</b>	<b>Miembro Asignado</b>
<b>Product Owner</b>	Edwin Monzon Llemen
<b>Scrum Master</b>	Jean Pierre Graos Neciosup
<b>Equipo de desarrollo</b>	Marco Javier Rebaza Aguilar Roberto Núñez Casaux

#### 2.2.2.3. Descripción:

Se recabó información en las visitas que se tuvo al sistema actual de resoluciones tanto para subida y descarga como al buscador, ya que se necesita comprender el funcionamiento actual de estos sistemas. Ya teniendo una idea del funcionamiento del sistema se detalla lo siguiente a continuación:

##### **a) Objetivo:**

El sistema de “Resoluciones” es una plataforma digital que permite a la Universidad Nacional de Trujillo subir y descargar las distintas resoluciones y/o actas. Este sistema automatiza el proceso de subida, consulta y

descarga de estos documentos, lo que ahorra tiempo y reduce errores.

**b) Funcionalidades:**

- Permite al administrador y al trabajador acceder al sistema mediante cuentas previamente brindadas y dar una vista si no es un trabajador de la institución.
- Permite al administrador y al trabajador poder cambiar su contraseña desde el interior del sistema.
- Permite al administrador poder subir las resoluciones y/o actas al sistema.
- Permite al administrador mostrar las resoluciones y/o actas que están en el sistema, además de que se puede buscar mediante filtros ya sea por el número, órgano al que pertenece, título, descripción, etc.
- Permite al trabajador y no trabajador buscar las distintas resoluciones y/o actas que estén en el sistema.

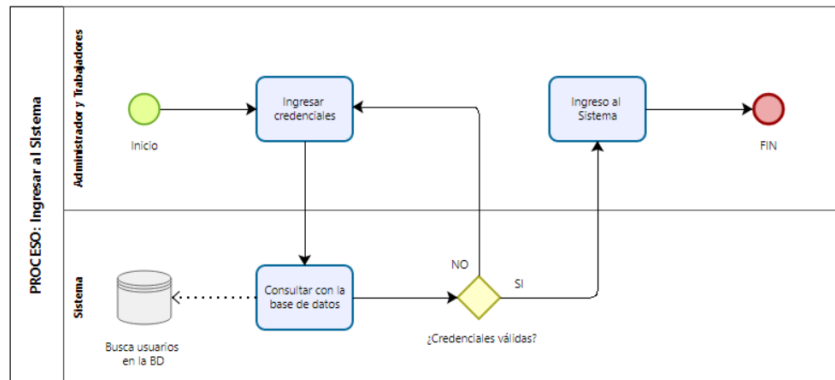
**c) Diagrama de procesos**

A continuación, se mostrarán los diferentes diagramas de procesos correspondientes a los diferentes tipos de usuario, los cuales son el administrador y los trabajadores que en este caso tienen el rol de publicadores.



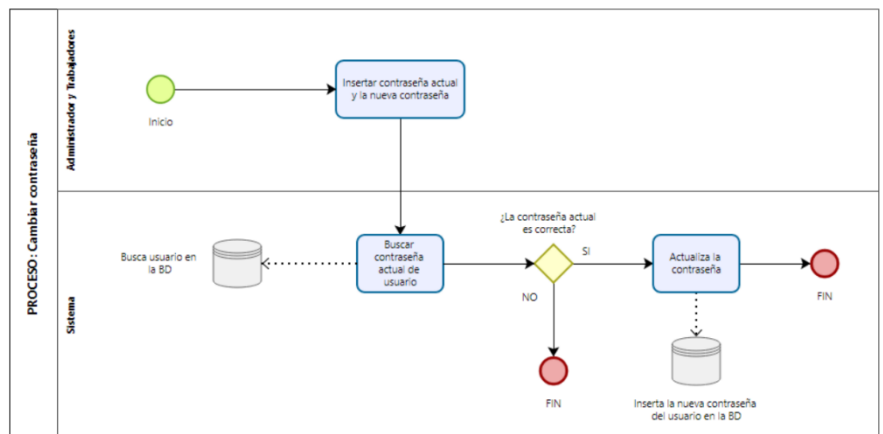
**Figura 2**

*D.P. Administrador y Trabajador – Ingresar al Sistema*



**Figura 3**

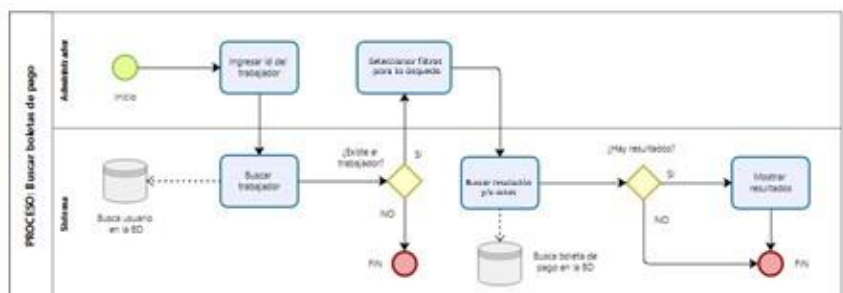
*D.P. Administrador y Trabajador - Cambiar contraseña*



*Nota: Como pre-requisito se necesita estar “logueado” en el sistema.*

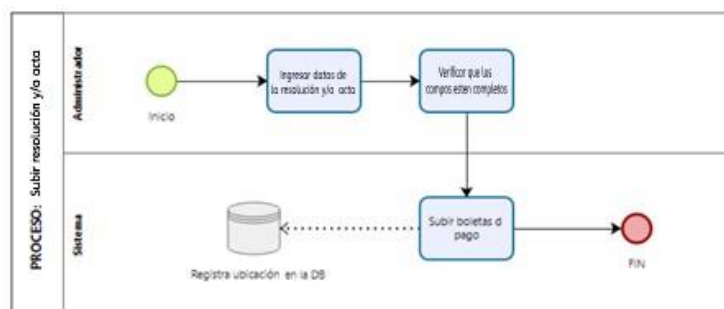
**Figura 4**

*D.P. Administrador - Buscar resolución y/o actas.*



**Figura 5**

*D.P. Administrador o trabajador(publicador) - Subir resolución y/o acta.*



*Nota: Como pre-requisito se necesita estar “logueado” en el sistema como administrador o trabajador.*

#### 2.2.2.4. Historias de Usuario:

**Figura 6**

*Historia de Usuario - Acceder a las actas y resoluciones.*

<b>HISTORIA DE USUARIO #1</b>		Acceder a las actas y resoluciones
<b>Como</b> Usuario		
<b>Quiero</b> tener acceso a todas las actas y resoluciones en el sistema		
<b>Para</b> poder consultarlas y descargarlas en cualquier momento		
<b>Validación</b>		<b>Prioridad:</b> Alta
El usuario puede tener acceso al los archivos publicos mediante una interfaz que permite buscar y descargar los archivos de actas y resoluciones.		<b>Puntos estimados:</b> 3

## Figura 7

### *Historia de Usuario - Tener un mantenedor de archivos.*

<b>HISTORIA DE USUARIO #2</b> <span style="float: right;">Tener un mantenedor de archivos</span>  <b>Como</b> Administrador y Publicador  <b>Quiero</b> tener un mantenedor de los archivos correspondientes a actas o resoluciones  <b>Para</b> poder crearlas, visualizarlas, eliminarlas o editarlas en cualquier momento	
<b>Validación</b>  El administrador y publicador puede realizar las acciones de creación, visualización, edición o eliminación, teniendo así un CRUD completo para dicha sección de archivos que involucran actas y resoluciones de la UNT.	<b>Prioridad:</b> Alta  <b>Puntos estimados:</b> 4

## Figura 8

### *Historia de Usuario - Tener un mantenedor de Órganos.*

<b>HISTORIA DE USUARIO #3</b> <span style="float: right;">Acceso a la configuración de perfil</span>  <b>Como</b> Administrador  <b>Quiero</b> tener un mantenedor de los órganos correspondientes a la Universidad Nacional de Trujillo  <b>Para</b> poder crearlos, visualizarlos, eliminarlos o editarlos en cualquier momento	
<b>Validación</b>  El administrador puede realizar las acciones de creación, visualización, edición o eliminación, teniendo así un CRUD completo para dicha sección de órganos existentes en la Universidad Nacional de Trujillo.	<b>Prioridad:</b> Alta  <b>Puntos estimados:</b> 4

## Figura 9

### *Historia de Usuario - Acceso a la configuración de perfil.*

<b>HISTORIA DE USUARIO #4</b> <span style="float: right;">Tener un mantenedor de órganos</span>  <b>Como</b> Administrador y Publicador  <b>Quiero</b> tener disponible la sección de configurar perfil  <b>Para</b> poder modificar mis datos personales o clave de acceso en el sistema	
<b>Validación</b>  El administrador y publicador tendrá acceso a la sección para configurar su perfil de usuario, pudiendo así, modificar sus datos personales de usuario, además, podrá modificar su clave de acceso en caso de que crea que sea necesario para evitar que otro usuario ingrese con sus credenciales.	<b>Prioridad:</b> Alta  <b>Puntos estimados:</b> 4

## Figura 10

*Historia de Usuario - Tener un mantenedor de archivos de facultad.*

<b>HISTORIA DE USUARIO #5</b>		Tener un mantenedor de archivos de facultad
<b>Como</b> Publicador de Facultad		
<b>Quiero</b> tener un mantenedor de los archivos correspondientes a actas o resoluciones de facultad		
<b>Para</b> poder crearlas, visualizarlas, eliminarlas o editarlas en cualquier momento		
<b>Validación</b>	<b>Prioridad:</b> Alta	
El publicador de facultad puede realizar las acciones de creación, visualización, edición o eliminación, teniendo así un CRUD completo para dicha sección de archivos de facultad que involucran actas y resoluciones de la UNT.	<b>Puntos estimados:</b> 4	

### 2.2.2.5. Plan de proyecto:

Como la metodología ágil que usaremos será de scrum, estableceremos el plan de proyecto ágil en el cual estableceremos tiempos de cada tarea, tal y como se muestra en la **Figura 11**.

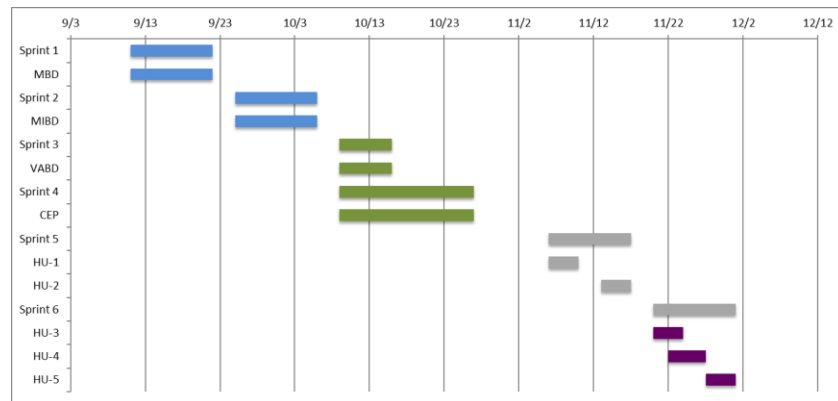
## Figura 11

*Plan del proyecto.*

Nombre de la tarea	Responsable	Fecha de inicio	Fecha final	Días	Estado
Sprint 1	Marco	9/11	9/22	11	Sin Empezar
MBD	Marco	9/11	9/22	11	Sin Empezar
Sprint 2	Marco	9/25	10/6	11	Sin Empezar
MIBD	Marco	9/25	10/6	11	Sin Empezar
Sprint 3	Marco	10/9	10/16	7	Sin Empezar
VABD	Marco	10/9	10/16	7	Sin Empezar
Sprint 4	Roberto	10/9	10/27	18	Sin Empezar
CEP	Marco	10/9	10/27	18	Sin Empezar
Sprint 5	Roberto	11/6	11/17	11	Sin Empezar
HU-1	Marco	11/6	11/10	4	Sin Empezar
HU-2	Marco	11/13	11/17	4	Sin Empezar
Sprint 6	Roberto	11/20	12/1	11	Sin Empezar
HU-3	Marco	11/20	11/24	4	Sin Empezar
HU-4	Roberto	11/22	11/27	5	Sin Empezar
HU-5	Roberto	11/27	12/1	4	Sin Empezar

**Figura 12**

*Diagrama de proceso.*



### 2.2.3. Resultados

- Se escogió la metodología ágil.
- Se logró recabar información sobre los procesos que hace el sistema actual.
- Se recabó información sobre los nuevos requerimientos por parte de los usuarios.
- Se creó las historias de usuario de acuerdo a la metodología SCRUM.

### 2.3. Semana 03 (11/09/2023 – 15/09/2023):

#### 2.3.1. Objetivos semanales:

- Analizar y crear un nuevo modelo de la base datos en base al anterior.
- Diseñar los diagramas de los casos de uso tanto del sistema como del negocio.

#### 2.3.2. Descripción:

Por motivos del desarrollo del sistema no se me pudo dar acceso al código ni a la base de datos, puesto que los encargados se encuentran realizando el levante de este sistema, sin embargo se encargo otra tarea, que es la realización de un formulario para la recolección de datos de los distintos ambientes de la UNIVERSIDAD NACIONAL DE TRUJILLO, en el cual se me especificaron ciertos requerimientos y consideraciones a la hora de realizar este formulario.

Para realización de este formulario, se planteó el uso de la metodología KanBan como herramienta de control de actividades, debido a la facilidad y adaptabilidad con respecto a la gestión ágil de proyectos, ya que se pensó en usar el Framework de LARAVEL, es por eso que se buscó una metodología eficaz y sencilla de comprender al momento del desarrollo de un proyecto

#### **2.3.2.1. Diseño de la base de datos para el formulario:**

Diseñar la base de datos para un formulario de recolección de datos de equipos de la Universidad Nacional de Trujillo (UNT) implica definir las tablas, campos y relaciones necesarias para almacenar y gestionar la información de manera eficiente. Primero debemos de analizar los requerimientos funcionales que nos han proporcionado:

##### **1. Registro de equipo:**

- El formulario debe permitir al usuario registrar los campos necesarios para la identificación del equipo tal como: nombre, serie, descripción, categoría, estado, modelo, marca

##### **2. Registro de ambiente:**

- El formulario debe permitir registrar que tiempo de ambiente es, estos pueden ser: Laboratorio, salón, dirección, secretariado, etc. Así mismo la capacidad de cada uno de estos.

##### **3. Registro de Facultades:**

- El formulario debe permitir al usuario registrar cada ambiente según la facultad que corresponda.

##### **4. Registro el tipo de unidad:**

- El formulario debe permitir registrar al usuario el tipo de unidad a la cual esta registrando esta puede ser académica (Los diferentes tipos de facultades) o

administrativa (Tesorería, Unidades especiales, Secretariados, Rectorado, etc.).

## 5. Generar reportes:

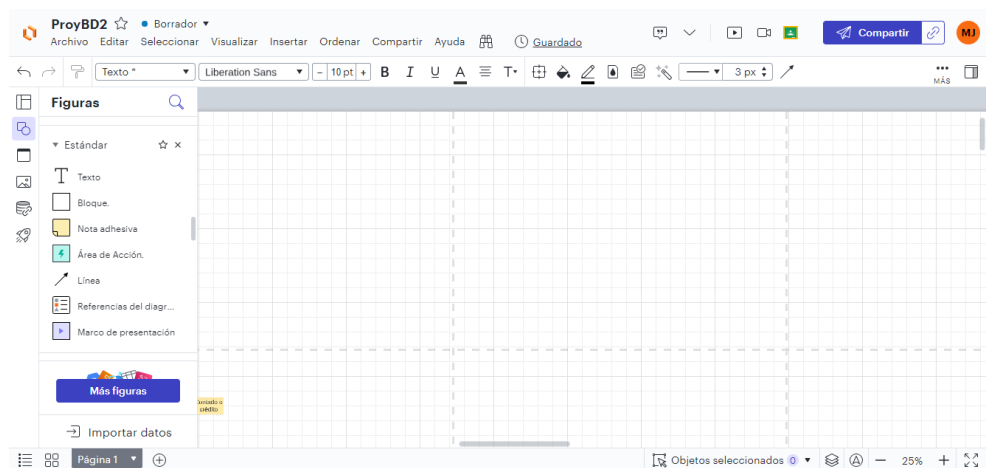
- El formulario debe ser capaz de generar diferentes tipos de reportes, como por ejemplos, el estado de los equipos de un ambiente.

Analizando estos requisitos funcionales llegamos a un posible diseño de la base de datos, dicho diseño fue realizado en Lucidchart.

Lucidchart es una plataforma en línea de creación de diagramas y gráficos que se utiliza para crear visualmente diagramas de flujo, diagramas de Gantt, mapas mentales, organigramas, diagramas de red, diagramas UML y muchos otros tipos de diagramas y representaciones visuales.

### **Figura 13**

#### *Plataforma online Lucidchart*

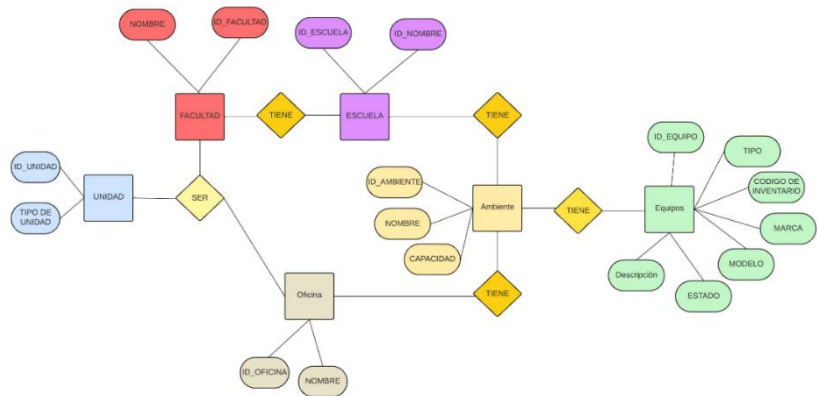


El diseño Entidad - Relación de la base de datos obtenido se muestra en la figura 3, este diseño de la base de datos fue obtenido luego de analizar los requerimientos funcionales, se ha considerado la creación de distintas tablas, las cuales se hizo

pensando en un diseño estructurado, ya que se usará PHP para la realización de la base de datos.

**Figura 14**

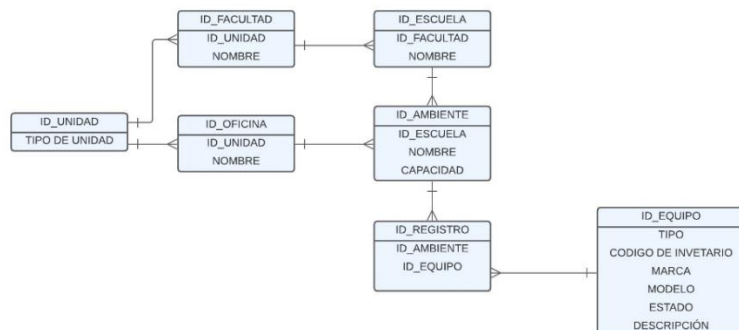
*Diagrama del modelo entidad-relación de la base de datos*



En la figura 4, se muestra el diagrama del modelo físico de la base de datos

**Figura 15**

*Diagrama del modelo físico de la base de datos*



Los diseños presentados, aún no han sido aprobados para iniciar con la implementación de la base de datos en un SGBD, aun se puede optimizar el diseño sin embargo es una



solución adecuada y simple para representar la data que se va a trabajar.

#### **2.3.3. Inconvenientes:**

- Debido a que, por problemas de los desarrolladores encargados del anterior sistema, que no lograron pasar las scripts con la data necesaria para empezar a trabajar, causa un atraso para su desarrollo, llevando consigo perdida de tiempo lo cual altera a lo que teníamos proyectado realizar, sin embargo, se nos asignó un nuevo proyecto momentáneo que solo es modelar la base de datos de este nuevo sistema, que servirá para el registro de equipos de la UNT. Por lo que implica darle una pausa a nuestro sistema de “Resoluciones”.

#### **2.3.4. Resultados:**

- No se logró realizar el nuevo modelo de la base de datos del sistema resoluciones.
- No se logró diseñar los casos de uso.
- Se logro modelar la base de datos del sistema para registrar equipos de la UNT.

### **2.4. Semana 04 (18/09/2023 – 22/09/2023):**

#### **2.4.1. Objetivos semanales:**

- Creación del proyecto usando el framework de laravel, establecer las rutas

#### **2.4.2. Descripción:**

Para la creación del proyecto usando el framework Laravel, se tiene que instalar algunas tecnologías.

##### **2.4.2.1. Herramientas informáticas a usar:**

**XAMPP:** XAMPP es una pila de soluciones de servidor web de código abierto y multiplataforma que incluye componentes esenciales para el desarrollo web. El nombre XAMPP significa "Cross-Platform (X), Apache (A), MariaDB (M), PHP (P), y Perl (P)". Proporciona una manera conveniente de configurar un entorno de desarrollo local para aplicaciones web. XAMPP

incluye el servidor web Apache, MySQL o MariaDB (un sistema de gestión de bases de datos relacionales), PHP (un lenguaje de programación del lado del servidor) y Perl (un lenguaje de programación), lo que lo hace adecuado para desarrollar y probar sitios web y aplicaciones web dinámicas en tu propia computadora antes de implementarlas en un servidor de producción.

**Figura 16**

*XAMPP*



**Composer:** Composer es una herramienta de gestión de dependencias para PHP, un lenguaje de programación ampliamente utilizado en el desarrollo web. Simplifica el proceso de administración e instalación de bibliotecas y paquetes externos de PHP necesarios para un proyecto. Con Composer, los desarrolladores pueden declarar las dependencias que necesita su proyecto PHP en un archivo "composer.json", y Composer descargará e instalará automáticamente los paquetes especificados junto con sus dependencias. Esto simplifica la tarea de mantener actualizadas las bibliotecas de software y ayuda a mantener la integridad de un proyecto al garantizar que todos los desarrolladores estén utilizando las mismas versiones de las dependencias.

**Figura 17**

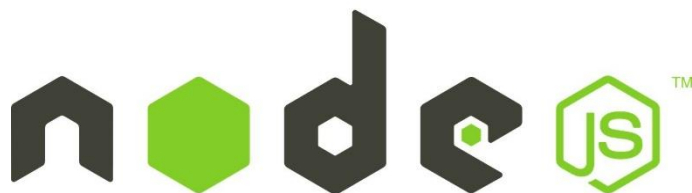
*Composer*



**Node.js:** Node.js es un entorno de ejecución de código abierto en el lado del servidor que permite a los desarrolladores crear y ejecutar aplicaciones JavaScript fuera de un navegador web. Está construido sobre el motor JavaScript V8 de Google y proporciona un modelo de E/S (entrada/salida) sin bloqueo y basado en eventos, lo que lo hace especialmente adecuado para crear aplicaciones de red escalables y eficientes. Node.js cuenta con un amplio ecosistema de bibliotecas y paquetes disponibles a través de npm (Gestor de Paquetes de Node), lo que lo hace popular para desarrollar varios tipos de aplicaciones, incluyendo servidores web, aplicaciones en tiempo real y APIs.

**Figura 18**

*Node.js*



**Git:** Git es un sistema de control de versiones distribuido (VCS) utilizado para rastrear cambios en el código fuente durante el desarrollo de software. Permite que varios desarrolladores colaboren en un proyecto, hagan un seguimiento de las revisiones de código y gestionen y fusionen fácilmente cambios de diferentes contribuyentes. Git es conocido por su velocidad, flexibilidad y robustas capacidades de ramificación y fusión. Es ampliamente utilizado en la industria del desarrollo de software y es la base de plataformas como GitHub y GitLab, que proporcionan alojamiento web para repositorios Git.

**Figura 19**

*GIT*

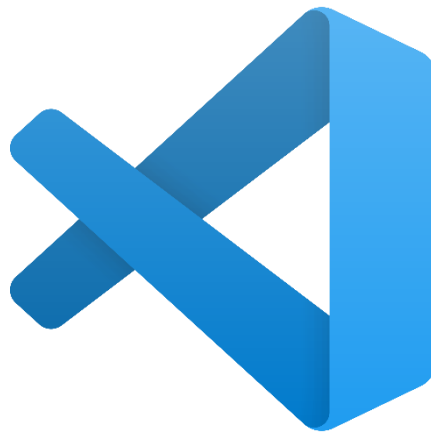


**Visual Studio code:** Visual Studio Code (VS Code) es un editor de código gratuito y de código abierto desarrollado por Microsoft. Es altamente personalizable y popular entre los desarrolladores por su versatilidad, eficiencia y soporte para una amplia gama de lenguajes de programación y extensiones. VS Code ofrece características como resaltado de sintaxis, autocompletado de código, soporte para depuración e integración

incorporada de control de versiones (incluido Git). Es una herramienta versátil para diversos tipos de tareas de desarrollo, incluyendo desarrollo web, desarrollo de aplicaciones y scripting. Se pueden agregar extensiones para personalizar el editor para entornos y lenguajes de desarrollo específicos.

**Figura 20**

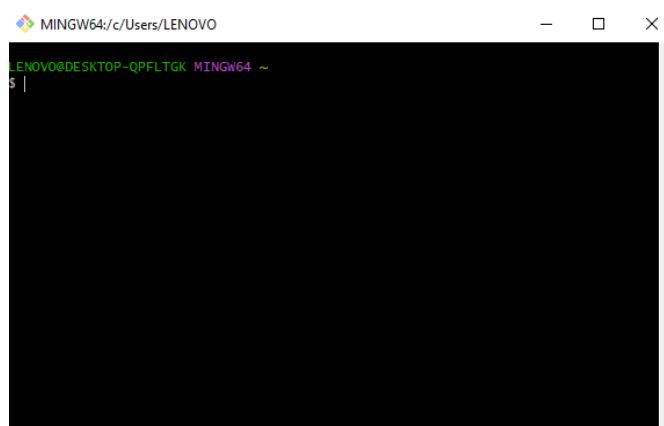
*Visual Studio Code*



Para crear el proyecto LARAVEL, debemos de usar XAMPP, precisamente el servidor de apache, es por esto que debemos de ubicarnos en la carpeta XAMPP en htdocs que es donde se ejecuta los proyectos para el servidor, abrimos el terminal git bash para hacer conexiones ssh, en la figura 21, se muestra la terminal de git bash:

## **Figura 21**

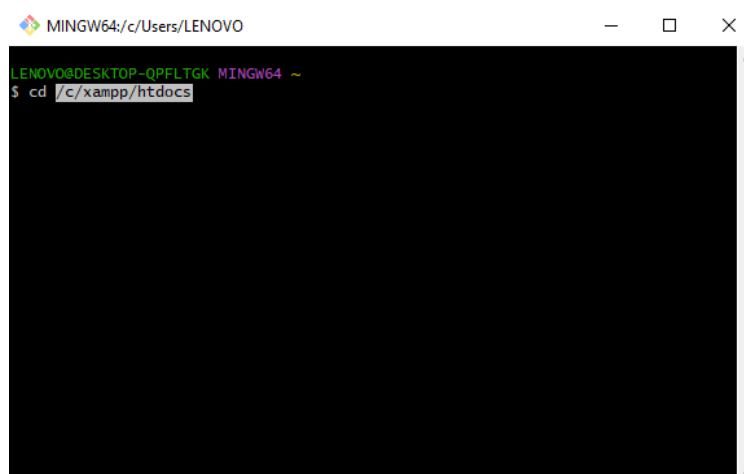
### *Consola git bash*



Entramos en la carpeta de htdocs desde la terminal, tal como se muestra en la figura 22.

## **Figura 22**

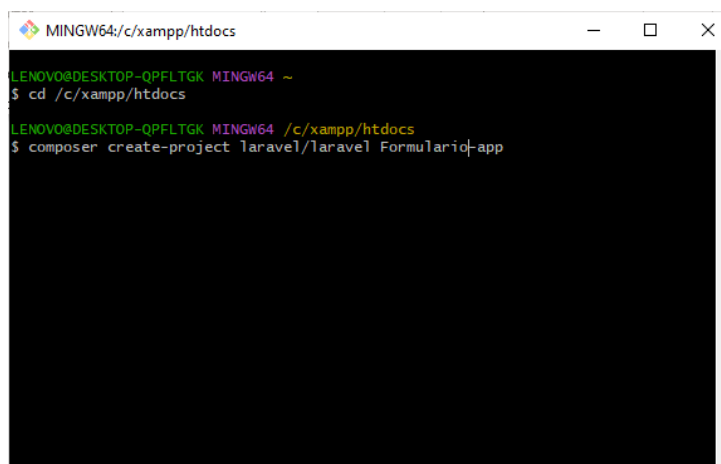
### *Carpeta de htdocs desde git bash.*



Ahora usaremos Composer para crear el proyecto con el nombre que nosotros escojamos, que en este caso lo llamaremos Formulario-app, tal como se muestra en la figura 23.

**Figura 23**

*Crear proyecto Formulario-app.*

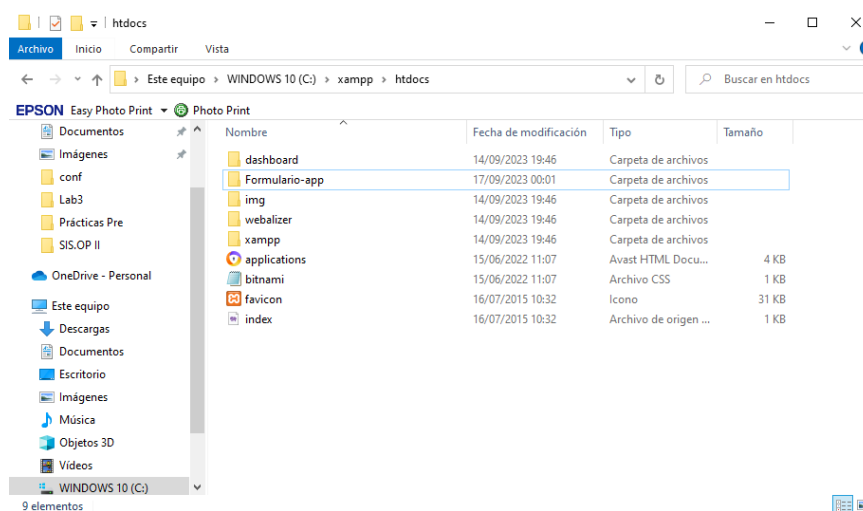


```
MINGW64:/c:/xampp/htdocs  
  
LENOVO@DESKTOP-QPFLTGK MINGW64 ~  
$ cd /c:/xampp/htdocs  
  
LENOVO@DESKTOP-QPFLTGK MINGW64 /c:/xampp/htdocs  
$ composer create-project laravel/laravel Formulario-app
```

Ahora verifiquemos que el proyecto se ha creado, tal como se muestra en la figura 24

**Figura 24**

*Carpeta htdocs.*



### 2.4.3. Resultados:

- Se creo el proyecto usando el framework de laravel.
- Se lograron crear las rutas para cada una de las vistas.

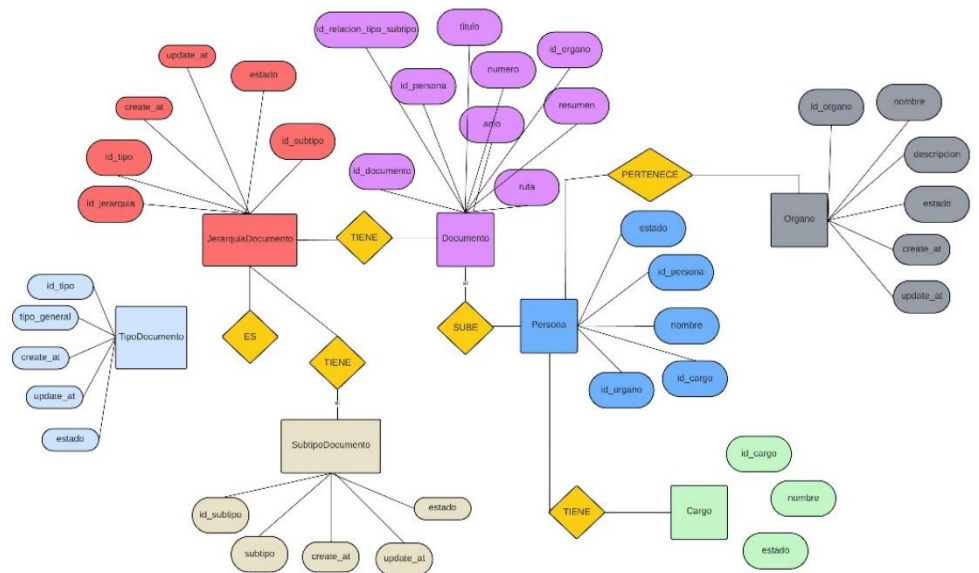




En la figura 27 se muestra una posible solución al modelado de la base datos en un diagrama-entidad relación.

**Figura 26**

*Diagrama entidad-relación para el sistema de resoluciones*



Después de analizar todo, llegamos a obtener este diagrama entidad-relación de la base de datos, el cual ha sido aprobado con éxito y faltaría crear las consultas, para migrar la data del sistema anterior.

### 2.5.3. Resultados:

- Se logró crear el nuevo modelo de la base de datos para el sistema de resoluciones.
- No se logró migrar la data debido a que aún falta crear las consultas para obtener la data.
- En la **Figura 27** se muestra la actualización del plan del proyecto.

**Figura 27**

*Actualización del plan del proyecto del sprint 1 (MBD).*

Nombre de la tarea	Responsable	Fecha de inicio	Fecha final	Días	Estado
Sprint 1	Marco	9/11	9/22	11	Finalizado
MBD	Marco	9/11	9/22	11	Finalizado
Sprint 2	Marco	9/25	10/6	11	Sin empezar
MIBD	Marco	9/25	10/6	11	Sin empezar
Sprint 3	Marco	10/9	10/16	7	Sin empezar
VABD	Marco	10/9	10/16	7	Sin empezar
Sprint 4	Roberto	10/9	10/27	18	Sin empezar
CEP	Marco	10/9	10/27	18	Sin empezar
Sprint 5	Roberto	11/6	11/17	11	Sin empezar
HU-1	Marco	11/6	11/10	4	Sin empezar
HU-2	Marco	11/13	11/17	4	Sin empezar
Sprint 6	Roberto	11/20	12/1	11	Sin empezar
HU-3	Marco	11/20	11/24	4	Sin empezar
HU-4	Roberto	11/22	11/27	5	Sin empezar
HU-5	Roberto	11/27	12/1	4	Sin empezar

## 2.6. Semana 6 (02/10/2023 – 06/10/2023):

### 2.6.1. Objetivos semanales:

- Migrar en tablas a la base de datos “bd\_intranet” en sistema de gestión de base de datos PostgreSQL.

### 2.6.2. Descripción:

Como ya se ha modelado la base de datos anteriormente y está ha sido aprobada, nos queda realizar las consultas para poder migrar la data, las consultas que realizaremos serán un poco complejas en algunos casos debido a que tienes que enlazar hasta 2 tablas para poder dar la data.

En las siguientes figuras, se mostrará las consultas que se realizaron:

**Figura 28**

*Consulta para obtener la data para la tabla persona.*

```
SELECT ROW_NUMBER() OVER (ORDER BY nombre) AS id_persona,
nombre,
id_organo,
id_cargo,
estado
FROM (
    SELECT nombre,
           idoficina AS id_organo,
           idtipo AS id_cargo,
           1 as estado
    FROM usuario
    UNION ALL
    SELECT nombres AS nombre,
           id_dep AS id_organo,
           NULL AS id_cargo,
           estado
    FROM acta_usuario
) AS combined_data;
```

**Figura 29**

*Consulta para obtener la data para la tabla organo.*

```
WITH Organos AS (  
  SELECT  
    idoficina AS id_organos,  
    upper(oficina) AS nombre,  
    1 as estado  
  FROM oficina  
  UNION ALL  
  SELECT  
    id_dep AS id_organos,  
    upper(dependencia) AS nombre,  
    estado  
  FROM acta_au_cu_dependencia  
  UNION ALL  
  SELECT  
    id_dep AS id_organos,  
    upper(dependencia) AS nombre,  
    estado  
  FROM acta_dependencia  
)  
SELECT  
  ROW_NUMBER() OVER (ORDER BY nombre) AS id_organos,  
  nombre,  
  NULL AS descripcion,  
  COALESCE(estado, 1) AS estado  
FROM Organos  
ORDER BY id_organos;
```

**Figura 30**

*Consulta para obtener la data para la tabla documentos.*

```
SELECT  
  id_documento,  
  id_relacion_tipo_subtipo,  
  titulo,  
  numero,  
  anio,  
  fecha_documento,  
  fecha_publicacion,  
  resumen,  
  id_organos,  
  id_persona,  
  ruta,  
  estado  
FROM "ACTAS 3"  
UNION  
SELECT  
  id_documento,  
  id_relacion_tipo_subtipo,  
  titulo,  
  numero,  
  anio,  
  fecha_documento,  
  fecha_publicacion,  
  resumen,  
  id_organos,  
  id_persona,  
  ruta,  
  estado  
FROM "RESOLUCIONES 3";
```

**Figura 31**

*Consulta para obtener la data para la tabla tipo\_documento.*

```
INSERT INTO documentos (tipo_general, estado) VALUES ('1','resolucion', true);
INSERT INTO documentos (tipo_general, estado) VALUES ('2','acta', true);

*****Jerarquia
WITH CombinedData AS (
    SELECT
        "SubtipoDocumento".id_subtipodocumento AS id_subtipo,
        CASE
            WHEN "SubtipoDocumento".subtipo LIKE '%Resolucion%' THEN TipoDocumentoResolucion.id
            ELSE TipoDocumentoActa.id
        END AS id_tipo,
        1 AS estado
    FROM "SubtipoDocumento"
    LEFT JOIN "TipoDocumento" AS TipoDocumentoResolucion ON TipoDocumentoResolucion.tipo_general =
    'resolucion'
    LEFT JOIN "TipoDocumento" AS TipoDocumentoActa ON TipoDocumentoActa.tipo_general = 'acta'
)
SELECT
    row_number() OVER () AS id,
    id_tipo,
    id_subtipo,
    estado
FROM CombinedData
```

**Figura 32**

*Consulta para obtener la data para la tabla subtipo\_documento.*

```
WITH TodasLasPersonas AS (
    SELECT idusuario AS id_persona, nombre, idoficina AS id_organos, idtipo AS id_cargo, 1 AS estado
    FROM usuario
    UNION ALL
    SELECT id_usuario AS id_persona, nombres AS nombre, id_dep AS id_organos, 2 AS id_cargo, estado
    FROM acta_usuario
    UNION ALL
    SELECT id_usuario AS id_persona, nombres AS nombre, id_dep AS id_organos, 2 AS id_cargo, estado
    FROM acta_au_cu_usuario
)
SELECT
    ROW_NUMBER() OVER (ORDER BY nombre) AS id_persona,
    upper(nombre),
    id_organos,
    id_cargo,
    COALESCE(estado, 1) AS estado
FROM TodasLasPersonas;
```

**Figura 33**

*Consulta para obtener la data para la tabla cargo*

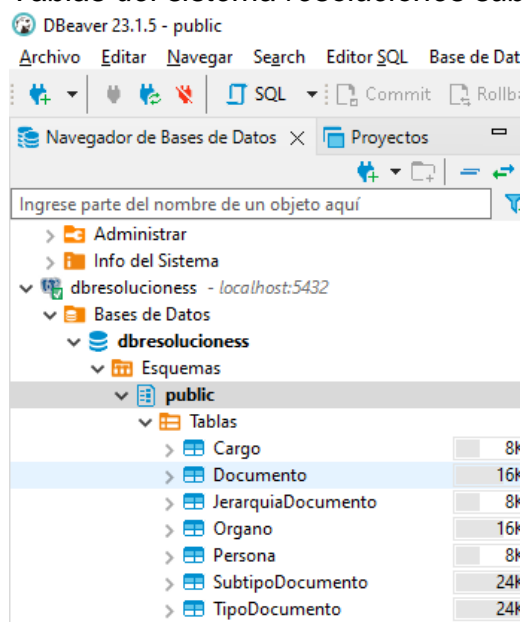
```
INSERT INTO documentos (tipo_general, estado) VALUES ('1','administrador', true)
INSERT INTO documentos (tipo_general, estado) VALUES ('2','publicador', true);
INSERT INTO documentos (tipo_general, estado) VALUES ('3','usuario', true);
```

Al migrar la data, se contabilizo 50008 documentos entre resoluciones y actas. Siendo 47024 resoluciones y 2,984 actas.

En la figura 34 se muestran las tablas subidas al SGBD DBeaver donde manejamos todas nuestras scripts.

**Figura 34**

*Tablas del sistema resoluciones subidas en DBeaver*



### 2.6.3. Resultados:

- Se logró modelar la nueva base de datos como se puede observar en la figura 34, en base al diagrama que se presentó en la figura 26.
- Se logro subir la data del sistema antiguo al nuevo.
- En la **Figura 35** se muestra al actualización del plan de proyecto.

**Figura 35**

*Actualización del plan del proyecto del sprint 2(MIBD).*

Nombre de la tarea	Responsable	Fecha de inicio	Fecha final	Días	Estado
Sprint 1	Marco	9/11	9/22	11	Finalizado
MBD	Marco	9/11	9/22	11	Finalizado
Sprint 2	Marco	9/25	10/6	11	Finalizado
MIBD	Marco	9/25	10/6	11	Finalizado
Sprint 3	Marco	10/9	10/16	7	Sin empezar
VABD	Marco	10/9	10/16	7	Sin empezar
Sprint 4	Roberto	10/9	10/27	18	Sin empezar
CEP	Marco	10/9	10/27	18	Sin empezar
Sprint 5	Roberto	11/6	11/17	11	Sin empezar
HU-1	Marco	11/6	11/10	4	Sin empezar
HU-2	Marco	11/13	11/17	4	Sin empezar
Sprint 6	Roberto	11/20	12/1	11	Sin empezar
HU-3	Marco	11/20	11/24	4	Sin empezar
HU-4	Roberto	11/22	11/27	5	Sin empezar
HU-5	Roberto	11/27	12/1	4	Sin empezar

## 2.7. Semana 7 (09/10/2023 – 13/10/2023):

### 2.7.1. Objetivos semanales:

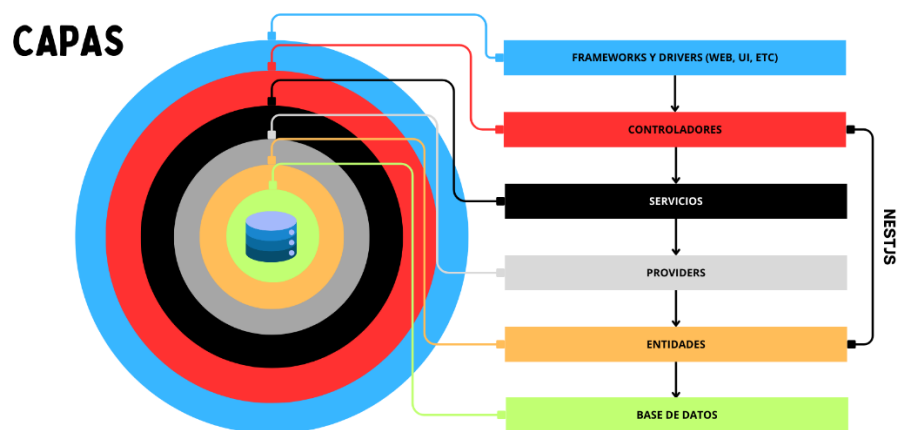
- Aprender la arquitectura del framework para backend **NestJS**
- Hacer la conexión entre la base de datos y el framework **NestJS**
- Empezar el armado de algunos endpoints con respecto a los casos de uso.

### 2.7.2. Descripción:

Primero hay que entender el funcionamiento y/o la arquitectura del framework, el cual se puede observar en la **Figura 36**, además de también tener en cuenta que **NestJS** trabaja con POO, Typescript, DTO's, decoradores, filters, pipes, interceptores, entidades, providers, módulos, servicios y una gran variedad de paquetes disponibles para poder trabajar y codificar de una manera más eficiente y segura; teniendo esto en cuenta, podemos seguir avanzando con el aprendizaje de este framework.

**Figura 36**

*Arquitectura de NestJS.*



#### 2.7.2.1. Herramientas informáticas:

Para esta semana se utilizó las siguientes herramientas informáticas:

- Visual Studio Code
- NodeJS
- NestJS

- Sequelize (ORM)

#### 2.7.2.2. Procedimiento:

Para este proyecto se utilizó una **ORM** llamada **Sequelize**, el cual es el Mapeo de Objeto Relacional, el cual nos brinda muchas facilidades a la hora de hacer la conexión, realizar consultas, hacer transacciones y mucho más con nuestra base de datos.

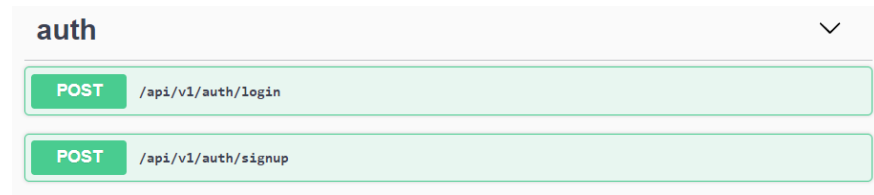
- Para realizar la conexión con la base de datos es necesario utilizar las **entidades**, el cual es el que vincula nuestra base de datos y nuestro backend, y para poder implementarlas junto con **Sequelize** es necesario hacer una serie de configuraciones previas antes de poder realizar la conexión que nos brinda la propia documentación oficial de NestJS.
- Luego de haber creado la entidad esta se usa dentro de un **“provider”**, la cual a su vez esta se asocia a una constante. Un **“provider”**, como su mismo nombre en inglés lo dice, provisiona a un modelo el uso de una entidad.
- Después, el provider creado se importa dentro de un módulo para su uso usando el **spread operator**.
- Finalmente se puede usar la entidad dentro de los **“services”**, el cual contiene la lógica de negocio para poder hacer consultas a la base de datos, el cual se retorna al **controlador**, el cual es el que comunica los dispositivos del exterior con nuestro backend.

#### 2.7.2.3. Armado de Endpoints:

Para el armado de los endpoints se utilizó el procedimiento anteriormente mencionado, el cual fue documentado usando **“swagger”** como se puede evidenciar en la **Figura 37**.

**Figura 37**

*Endpoints de autenticación documentado en Swagger*



### 2.7.3. Resultados:

- Se logró realizar la conexión entre el backend y la base de datos.
- Se armó los endpoints para el logueo de usuarios.

## 2.8. Semana 8 (16/10/2023 – 20/10/2023):

### 2.8.1. Objetivos semanales:

- Armar los endpoints correspondientes a la tabla de publicadores y las tablas que se le relacionan.

### 2.8.2. Descripción:

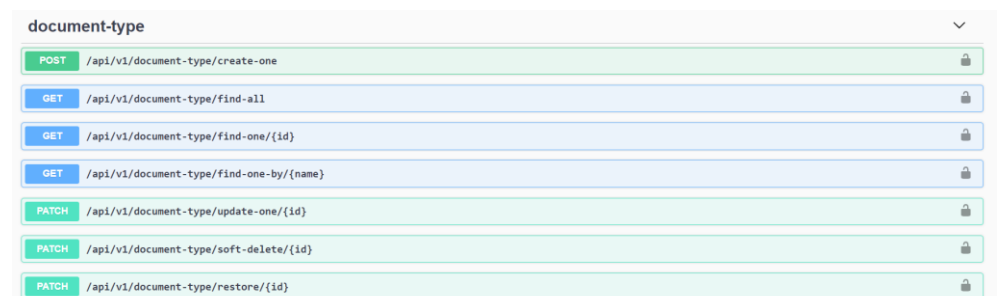
#### 2.8.2.1. Herramientas informáticas:

- Visual Studio Code
- NodeJS
- NestJS
- Sequelize (ORM)
- Swagger

#### 2.8.2.2. Armado de endpoints:

**Figura 38**

*CRUD para la tabla “documento\_tipo” documentado en Swagger*





**Figura 39**

*CRUD para la tabla “documento\_jerarquia” documentado en Swagger*

document-hierarchy			▼
POST	/api/v1/document-hierarchy/create-one		🔒
GET	/api/v1/document-hierarchy/find-all		
GET	/api/v1/document-hierarchy/find-all-by		
GET	/api/v1/document-hierarchy/find-one/{id}		🔒
PATCH	/api/v1/document-hierarchy/update-one/{id}		🔒
PATCH	/api/v1/document-hierarchy/soft-delete/{id}		🔒
PATCH	/api/v1/document-hierarchy/restore/{id}		🔒

**Figura 40**

*CRUD para la tabla “órgano” documentado en Swagger*

organ			▼
POST	/api/v1/organ/create-one		🔒
GET	/api/v1/organ/find-all		🔒
GET	/api/v1/organ/find-all-by/{name}		🔒
PATCH	/api/v1/organ/update-one/{id}		🔒
PATCH	/api/v1/organ/soft-delete/{id}		🔒
PATCH	/api/v1/organ/restore/{id}		🔒

**Figura 41**

*CRUD para la tabla “multitabla” documentado en Swagger*

multitable			▼
POST	/api/v1/multitable/create-one		🔒
GET	/api/v1/multitable/find-all		🔒
GET	/api/v1/multitable/find-all-by		🔒
PATCH	/api/v1/multitable/update-one/{id}		🔒
PATCH	/api/v1/multitable/soft-delete/{id}		🔒
PATCH	/api/v1/multitable/restore/{id}		🔒

### 2.8.3. Resultados:

- Se logró armar los endpoints planteados en esta semana.
- En la **Figura 42** se muestra la actualización del plan del proyecto.

**Figura 42**

*Actualización del plan del proyecto del sprint 3(VABD).*

Nombre de la tarea	Responsable	Fecha de inicio	Fecha final	Días	Estado
Sprint 1	Marco	9/11	9/22	11	Finalizado
MBD	Marco	9/11	9/22	11	Finalizado
Sprint 2	Marco	9/25	10/6	11	Finalizado
MIBD	Marco	9/25	10/6	11	Finalizado
Sprint 3	Marco	10/9	10/16	7	Finalizado
VABD	Marco	10/9	10/16	7	Finalizado
Sprint 4	Roberto	10/9	10/27	18	Sin empezar
CEP	Marco	10/9	10/27	18	Sin empezar
Sprint 5	Roberto	11/6	11/17	11	Sin empezar
HU-1	Marco	11/6	11/10	4	Sin empezar
HU-2	Marco	11/13	11/17	4	Sin empezar
Sprint 6	Roberto	11/20	12/1	11	Sin empezar
HU-3	Marco	11/20	11/24	4	Sin empezar
HU-4	Roberto	11/22	11/27	5	Sin empezar
HU-5	Roberto	11/27	12/1	4	Sin empezar

## **2.9. Semana 9 (23/10/2023 – 27/10/2023):**

### **2.9.1. Objetivos semanales:**

- Crear los endpoints para subir, editar y eliminar los archivos (resoluciones y actas)

### **2.9.2. Descripción:**

#### **2.9.2.1. Herramientas Informáticas**

- DBeaver
- Visual Studio Code
- NestJS
- Sequelize
- NodeJS

#### **2.9.2.2. Procedimiento**

Para subir los archivos, estos deben ser subidos primero a un servidor de archivos, en este caso se usa el servicio de NextCloud, teniendo en cuenta ello, el procedimiento para subir un archivo es el siguiente:

- La tabla documento requiere varios campos, los cuales también incluye el path del archivo, en este caso, el path es la ruta en la que NextCloud guarda el archivo subido.
- Una vez obtenido esa ruta, se puede guardar ese y los demás datos necesarios dentro de la base de datos.

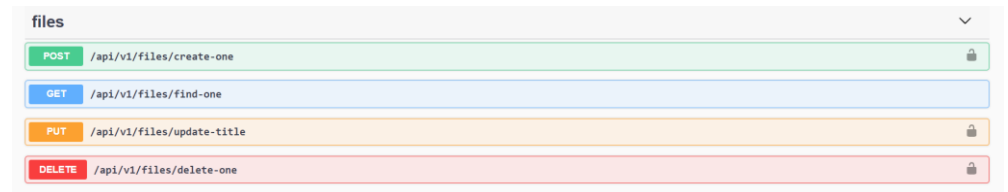
Para editar un archivo, si se desea cambiar el título del archivo o eliminarlo, se elaboró endpoints para ello.

### 2.9.2.3. Armado de los endpoints:

Las evidencias se pueden observar en la **Figura 43** y la **Figura 44** los cuales fueron documentados usando **Swagger**.

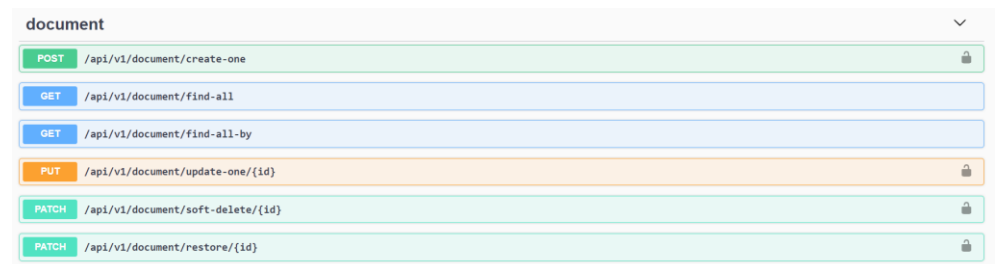
**Figura 43**

*CRUD para el manejo de archivos documentado en Swagger*



**Figura 44**

*CRUD para la tabla “documento” documentado en Swagger*



### 2.9.3. Resultados:

- Se logró armar los endpoints planteados en esta semana.
- En la **Figura 45** se muestra la actualización del plan del proyecto.

**Figura 45**

*Actualización del plan del proyecto del sprint 2 en progreso.*

Nombre de la tarea	Responsable	Fecha de inicio	Fecha final	Días	Estado
Sprint 1	Marco	9/11	9/22	11	Finalizado
MBD	Marco	9/11	9/22	11	Finalizado
Sprint 2	Marco	9/25	10/6	11	Finalizado
MIBD	Marco	9/25	10/6	11	Finalizado
Sprint 3	Marco	10/9	10/16	7	Finalizado
VABD	Marco	10/9	10/16	7	Finalizado
Sprint 4	Roberto	10/9	10/27	18	Finalizado
CEP	Marco	10/9	10/27	18	Finalizado
Sprint 5	Roberto	11/6	11/17	11	Sin empezar
HU-1	Marco	11/6	11/10	4	Sin empezar
HU-2	Marco	11/13	11/17	4	Sin empezar
Sprint 6	Roberto	11/20	12/1	11	Sin empezar
HU-3	Marco	11/20	11/24	4	Sin empezar
HU-4	Roberto	11/22	11/27	5	Sin empezar
HU-5	Roberto	11/27	12/1	4	Sin empezar

## **2.10. Semana 10 (30/10/2023 – 03/11/2023):**

### **2.10.1. Objetivos semanales:**

- Actualizar las rutas(path) en la base de datos de los archivos mediante consultas.

### **2.10.2. Descripción:**

#### **2.10.2.1. Herramientas Informáticas:**

- DBeaver

#### **2.10.2.2. Procedimiento:**

Para corregir los path de la tabla documento que es donde se encuentran tanto las resoluciones como actas, crearemos 3 consultas las consultas servirán para corregirlas.

Se busca corregirlas ya que los archivos se subirán a un servidor externo que es nextcloud los cuales deberán de tener el siguiente formato para su corrección de cada ruta(path):

**/App\_Resoluciones/Resoluciones\_Files/**

Por ejemplo:

Si el archivo tiene la siguiente ruta:

- **conten/res002-2005unt.zip**

Al actualizarlo será:

- **/App\_Resoluciones/Resoluciones\_Files/res002-2005unt.zip**

Se encontraron 3 tipos de formatos de la ruta los cuales son:

- **conten/**
- **actas\_au\_cu/**
- **actas/**

Entonces las consultas que se realizarán serán las siguientes:

**Figura 46**

*Actualizar ruta content/ .*

```
UPDATE "DOCUMENTO_GENERAL" //esto cambia segun el nombre asignado
SET doc_ruta = REPLACE(doc_ruta, 'content/', '/App_Resoluciones/Resoluciones_Files/')
WHERE doc_ruta LIKE 'content/%';
```

**Figura 47**

*Actualizar ruta actas\_au\_cu/ .*

```
UPDATE "DOCUMENTO_GENERAL"
SET doc_ruta = REPLACE(doc_ruta, 'actas_au_cu/', '/App_Resoluciones/Resoluciones_Files/')
WHERE doc_ruta LIKE 'actas_au_cu/%';
```

**Figura 48**

*Actualizar ruta actas/ .*

```
UPDATE "DOCUMENTO_GENERAL"
SET doc_ruta = REPLACE(doc_ruta, 'actas/', '/App_Resoluciones/Resoluciones_Files/')
WHERE doc_ruta LIKE 'actas/%';
```

### 2.10.3. Resultados:

- Se logró actualizar las rutas(path) en la base de datos.
- En la **Figura 49** se muestra la actualización del plan del proyecto.

**Figura 49**

*Actualización del plan del proyecto del sprint 5(HU-1 Y HU-2).*

Nombre de la tarea	Responsable	Fecha de inicio	Fecha final	Días	Estado
Sprint 1	Marco	9/11	9/22	11	Finalizado
MBD	Marco	9/11	9/22	11	Finalizado
Sprint 2	Marco	9/25	10/6	11	Finalizado
MIBD	Marco	9/25	10/6	11	Finalizado
Sprint 3	Marco	10/9	10/16	7	Finalizado
VABD	Marco	10/9	10/16	7	Finalizado
Sprint 4	Roberto	10/9	10/27	18	Finalizado
CEP	Marco	10/9	10/27	18	Finalizado
Sprint 5	Roberto	11/6	11/17	11	En Progreso
HU-1	Marco	11/6	11/10	4	En Progreso
HU-2	Marco	11/13	11/17	4	En Progreso
Sprint 6	Roberto	11/20	12/1	11	Sin empezar
HU-3	Marco	11/20	11/24	4	Sin empezar
HU-4	Roberto	11/22	11/27	5	Sin empezar
HU-5	Roberto	11/27	12/1	4	Sin empezar

## 2.11. Semana 11 (06/11/2023 – 10/11/2023):

### 2.11.1. Objetivos semanales:

- Diseño e implementación de una interfaz para el buscador de resoluciones y actas.

### 2.11.2. Descripción:

#### 2.11.2.1. Herramientas Informáticas:

Para esta semana se utilizó las siguientes herramientas informáticas:

- Vs code
- Balsamiq

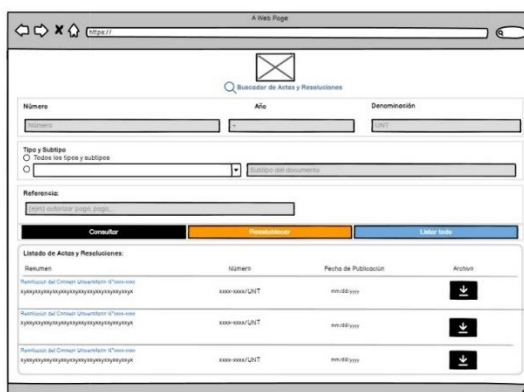
#### 2.11.2.2. Procedimiento:

##### a. Diseño:

- El diseño de la interfaz se realizó en el software de Balsamiq, tal como se muestra en la **Figura 50**.

#### Figura 50

*Diseño de la interfaz del buscador de resoluciones y actas.*



**Observación:** La usabilidad de esta interfaz fue establecida por el líder de proyecto, el cual se basó en el sistema anterior como una mejora para esta migración del sistema.

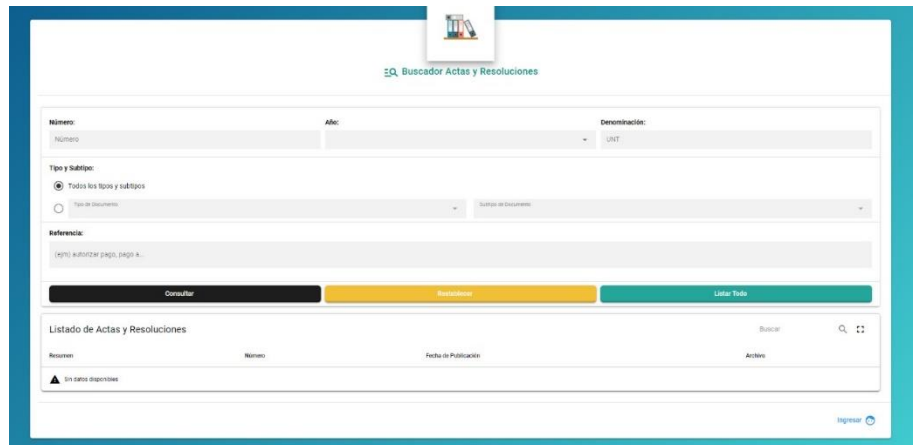
##### b. Implementación:

Para su implementación se usó las tecnologías vue 3 y quasar 1, los resultados se muestran en la **Figura 51**.

**Observación:** Por motivos de la empresa, no se permite poner código solamente el resultado de la implementación.

### Figura 51

*Implementación de la interfaz del buscador de resoluciones y actas.*



#### c. Finalidad:

- Estas interfaces buscan mejorar la experiencia del usuario y optimizar la gestión de usuarios del sistema. Ambas interfaces están diseñadas para satisfacer necesidades específicas y ofrecer funcionalidades distintas que se adaptan a los roles y responsabilidades de los usuarios.

#### 2.11.3. Resultados:

- Se logró diseñar e implementar la interfaz para el perfil de usuario.
- Se logró diseñar e implementar la interfaz para el listado de usuarios.
- En la **Figura 52** se muestra la actualización del plan del proyecto.

**Figura 52**

*Actualización del plan del proyecto del sprint 5(HU-1).*

Nombre de la tarea	Responsable	Fecha de inicio	Fecha final	Días	Estado
Sprint 1	Marco	9/11	9/22	11	Finalizado
MBD	Marco	9/11	9/22	11	Finalizado
Sprint 2	Marco	9/25	10/6	11	Finalizado
MIBD	Marco	9/25	10/6	11	Finalizado
Sprint 3	Marco	10/9	10/16	7	Finalizado
VABD	Marco	10/9	10/16	7	Finalizado
Sprint 4	Roberto	10/9	10/27	18	Finalizado
CEP	Marco	10/9	10/27	18	Finalizado
Sprint 5	Roberto	11/6	11/17	11	En Progreso
HU-1	Marco	11/6	11/10	4	Finalizado
HU-2	Marco	11/13	11/17	4	En Progreso
Sprint 6	Roberto	11/20	12/1	11	Sin empezar
HU-3	Marco	11/20	11/24	4	Sin empezar
HU-4	Roberto	11/22	11/27	5	Sin empezar
HU-5	Roberto	11/27	12/1	4	Sin empezar

## **2.12. Semana 12 (13/11/2023 – 17/11/2023):**

### **2.12.1. Objetivos semanales:**

- Diseño e implementación de un mantenedor de archivos para el rol de administrador y publicador.

### **2.12.2. Descripción:**

#### **2.12.2.1. Herramientas Informáticas:**

Para esta semana se utilizó las siguientes herramientas informáticas:

- Vs code.
- Balsamiq.

#### **2.12.2.2. Procedimiento:**

##### **a. Diseño:**

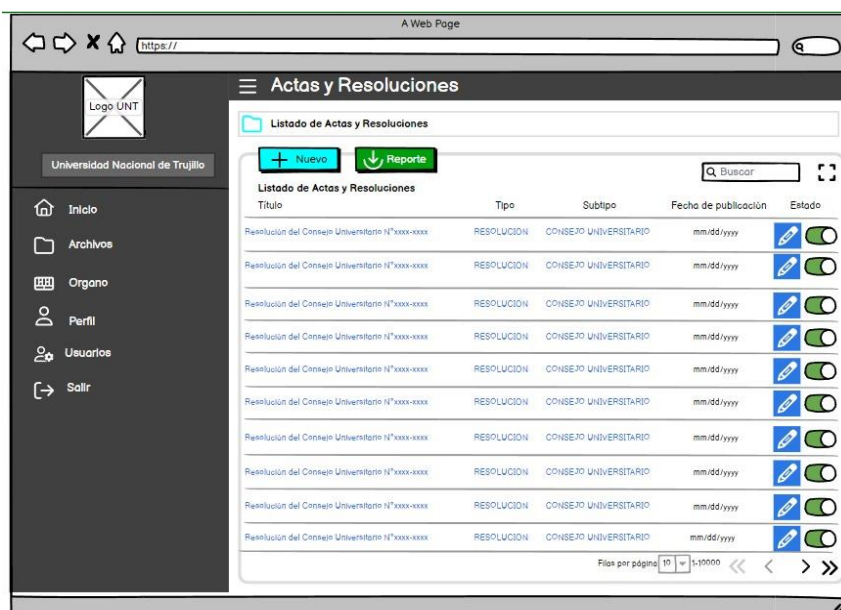
- El diseño de la interfaz se realizó en el software de Balsamiq, tal como se muestra en la **Figura 53**.

**Observación:** La usabilidad de esta interfaz fue establecida por el líder de proyecto, el cual se basó en el sistema anterior como una mejora para esta migración del sistema.



**Figura 53**

*Diseño de la interfaz del mantenedor de archivos.*



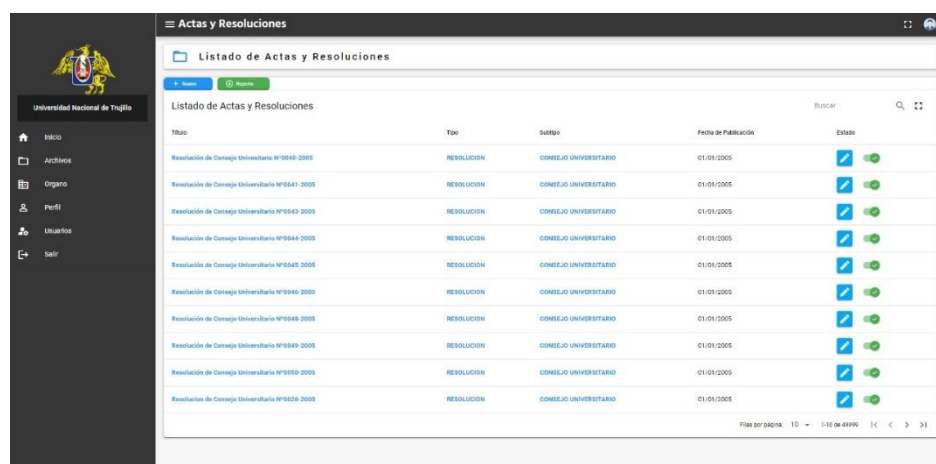
**b. Implementación:**

Para su implementación se usó las tecnologías vue 3 y quasar 1, el resultado se muestra en la **Figura 54**.

**Observación:** Por motivos de la empresa, no se permite poner código solamente el resultado de la implementación.

**Figura 54**

*Implementación de la interfaz del mantenedor de archivos*



### c. Finalidad:

- La implementación del mantenedor tiene como finalidad optimizar y facilitar la gestión eficiente de archivos dentro del sistema ya que desempeña un papel crucial en la organización, control y distribución de información digital, garantizando un flujo de trabajo efectivo y seguro para los usuarios con roles específicos, como administradores y publicadores.

### 2.12.3. Resultados:

- Se logró diseñar e implementar el mantenedor de archivos para el rol de administrador y publicador.
- En la **Figura 55** se muestra la actualización del plan del proyecto.

**Figura 55**

*Actualización del plan del proyecto del sprint 5 finalizado.*

Nombre de la tarea	Responsable	Fecha de inicio	Fecha final	Días	Estado
Sprint 1	Marco	9/11	9/22	11	Finalizado
MBD	Marco	9/11	9/22	11	Finalizado
Sprint 2	Marco	9/25	10/6	11	Finalizado
MIBD	Marco	9/25	10/6	11	Finalizado
Sprint 3	Marco	10/9	10/16	7	Finalizado
VABD	Marco	10/9	10/16	7	Finalizado
Sprint 4	Roberto	10/9	10/27	18	Finalizado
CEP	Marco	10/9	10/27	18	Finalizado
Sprint 5	Roberto	11/6	11/17	11	En Progreso
HU-1	Marco	11/6	11/10	4	Finalizado
HU-2	Marco	11/13	11/17	4	Finalizado
Sprint 6	Roberto	11/20	12/1	11	Sin empezar
HU-3	Marco	11/20	11/24	4	Sin empezar
HU-4	Roberto	11/22	11/27	5	Sin empezar
HU-5	Roberto	11/27	12/1	4	Sin empezar

### 2.13. Semana 13 (20/11/2023 – 24/11/2023):

#### 2.13.1. Objetivos semanales:

- Diseño e implementación de una interfaz para el perfil de usuario.
- Diseño e implementación de una interfaz para el listado de usuarios.

## 2.13.2. Descripción:

### 2.13.2.1. Herramientas Informáticas:

Para esta semana se utilizó las siguientes herramientas informáticas:

- Vs code
- Balsamiq

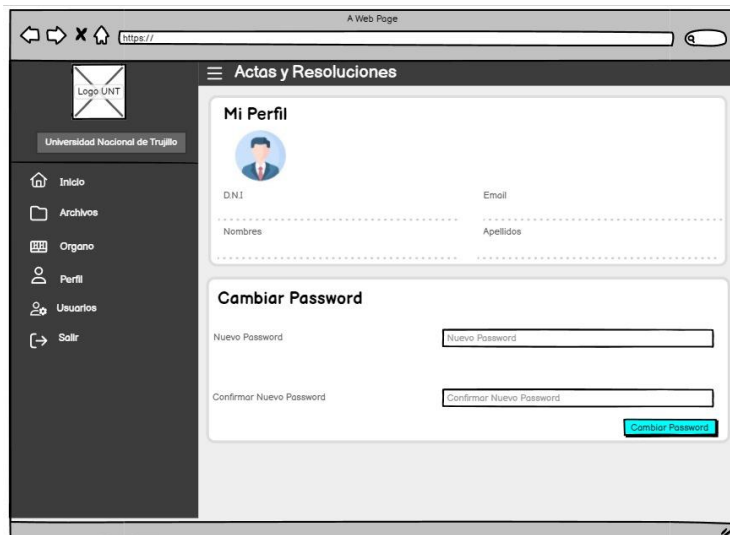
### 2.13.2.2. Procedimiento:

#### d. Diseño:

- El diseño de la interfaz se realizó en el software de Balsamiq, tal como se muestra en la **Figura 56** y **Figura 57**.

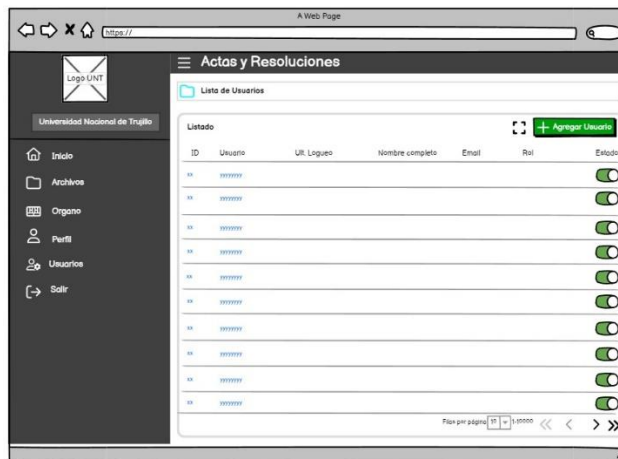
#### Figura 56

*Diseño de la interfaz del perfil de usuario*



**Figura 57**

*Diseño de la interfaz de lista de usuarios*



**Observación:** La usabilidad de esta interfaz fue establecida por el líder de proyecto, el cual se basó en el sistema anterior como una mejora para esta migración del sistema.

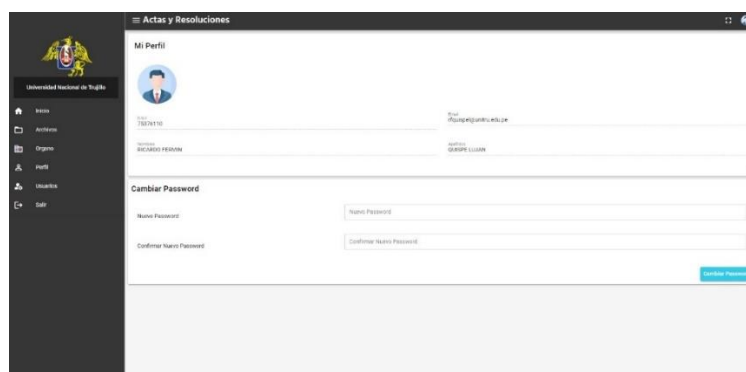
**e. Implementación:**

Para su implementación se usó las tecnologías vue 3 y quasar 1, los resultados se muestran en la **Figura 58** y **Figura 59**.

**Observación:** Por motivos de la empresa, no se permite poner código solamente el resultado de la implementación.

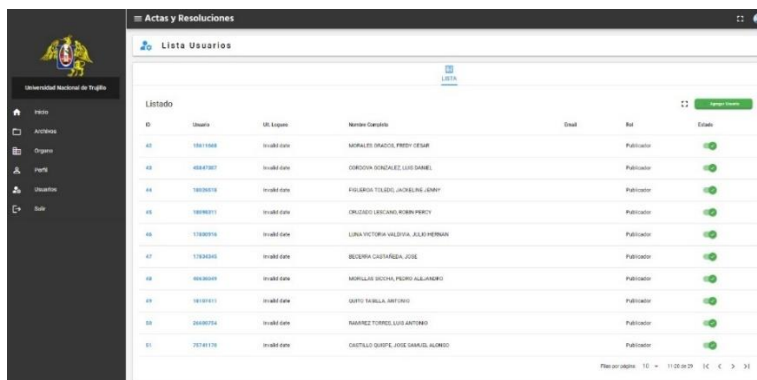
**Figura 58**

*Implementación de la interfaz perfil de usuario*



**Figura 59**

*Implementación de la interfaz lista de usuarios*



ID	Usuario	Ult. Acceso	Nombre Completo	Email	Rol	Estado
43	156115668	Inactivo date	MARIALI PRADO PERRY CESAR		Publicador	Activo
44	46847587	Inactivo date	GONZALO SANCHEZ LUIS DANIEL		Publicador	Activo
45	18036278	Inactivo date	EUGENIA TOLEDO JACQUELINE JENNY		Publicador	Activo
46	18036271	Inactivo date	ORLANDO GONZALEZ ROMAN PERRY		Publicador	Activo
47	17336276	Inactivo date	LUNA VICTORIA VALENTIA JULIA HERMAN		Publicador	Activo
48	17336265	Inactivo date	BELENIA CASTAÑEDA JOSE		Publicador	Activo
49	46936049	Inactivo date	MARULLAN SANCHEZ PEDRO AULANERO		Publicador	Activo
50	18101611	Inactivo date	MARTY MARILLA MARTINO		Publicador	Activo
51	26600774	Inactivo date	RAMIREZ TORRES LUIS ANTONIO		Publicador	Activo
52	76701170	Inactivo date	CASTILLO QUISPE JOSE DANIEL ALBERTO		Publicador	Activo

**f. Finalidad:**

- Estas interfaces buscan mejorar la experiencia del usuario y optimizar la gestión de usuarios del sistema. Ambas interfaces están diseñadas para satisfacer necesidades específicas y ofrecer funcionalidades distintas que se adaptan a los roles y responsabilidades de los usuarios.

**2.13.3. Resultados:**

- Se logró diseñar e implementar la interfaz para el perfil de usuario.
- Se logró diseñar e implementar la interfaz para el listado de usuarios.
- En la **Figura 60** se muestra la actualización del plan del proyecto.

**Figura 60**

*Actualización del plan del proyecto del sprint 6 en progreso.*

Nombre de la tarea	Responsable	Fecha de inicio	Fecha final	Días	Estado
Sprint 1	Marco	9/11	9/22	11	Finalizado
MBD	Marco	9/11	9/22	11	Finalizado
Sprint 2	Marco	9/25	10/6	11	Finalizado
MIBD	Marco	9/25	10/6	11	Finalizado
Sprint 3	Marco	10/9	10/16	7	Finalizado
VABD	Marco	10/9	10/16	7	Finalizado
Sprint 4	Roberto	10/9	10/27	18	Finalizado
CEP	Marco	10/9	10/27	18	Finalizado
Sprint 5	Roberto	11/6	11/17	11	Finalizado
HU-1	Marco	11/6	11/10	4	Finalizado
HU-2	Marco	11/13	11/17	4	Finalizado
Sprint 6	Roberto	11/20	12/1	11	En Progreso
HU-3	Marco	11/20	11/24	4	Finalizado
HU-4	Roberto	11/22	11/27	5	Finalizado
HU-5	Roberto	11/27	12/1	4	En Progreso

## **2.14. Semana 14 (27/11/2023 – 01/12/2023):**

### **2.14.1. Objetivos semanales:**

- Diseño e implementación de un mantenedor de órganos para el rol de administrador.

### **2.14.2. Descripción:**

#### **2.14.2.1. Herramientas Informáticas:**

Para esta semana se utilizó las siguientes herramientas informáticas:

- Vs code
- Balsamiq

#### **2.14.2.2. Procedimiento:**

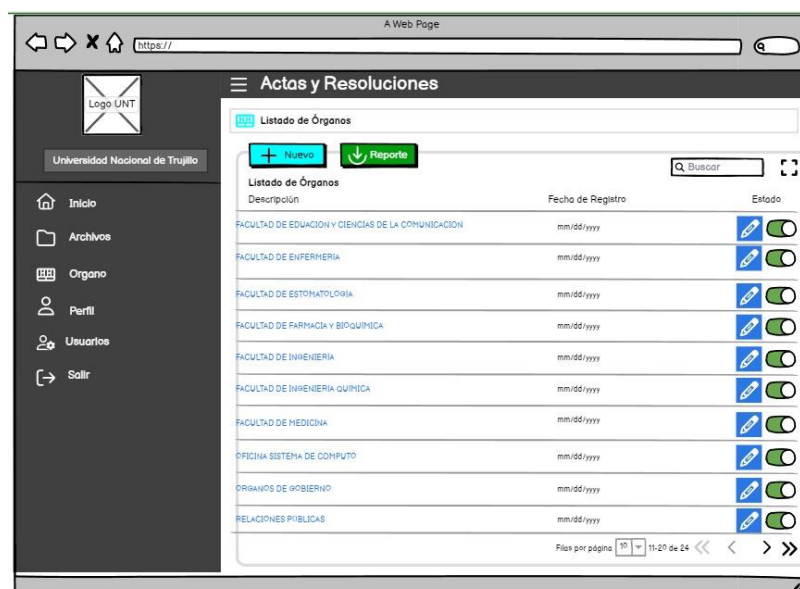
##### **a. Diseño:**

- El diseño de la interfaz se realizó en el software de Balsamiq, tal como se muestra en la **Figura 61**.

**Observación:** La usabilidad de esta interfaz fue establecida por el líder de proyecto, el cual se basó en el sistema anterior como una mejora para esta migración del sistema.

**Figura 61.**

*Diseño de la interfaz del mantenedor de órganos.*



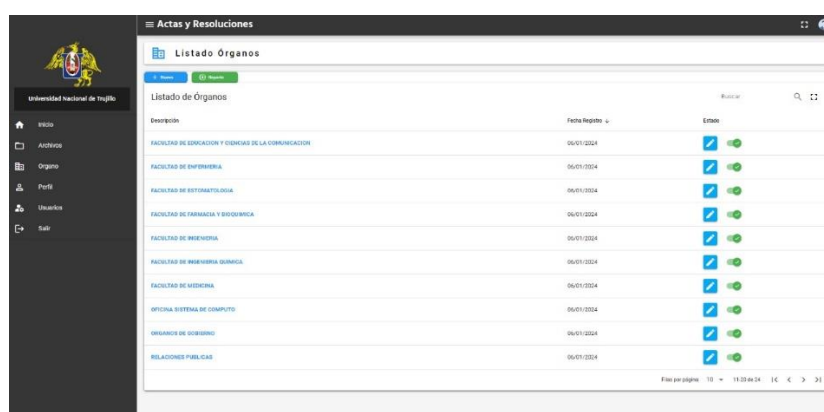
**b. Implementación:**

Para su implementación se usó las tecnologías vue 3 y quasar 1, el resultado se muestra en la **Figura 62**.

**Observación:** Por motivos de la empresa, no se permite poner código solamente el resultado de la implementación.

**Figura 62**

*Implementación de la interfaz del mantenedor de órganos*



**c. Finalidad:**

El mantenedor de órganos es para el administrador general (OTI) y para el administrador de facultades, su

finalidad es hacer un CRUD, ya que los órganos son las subentidades tales como:

- Facultades
- OTI
- Rectorado
- Etc

### 2.14.3. Resultados:

- Se logró diseñar e implementar el mantenedor de órganos para el rol de administrador.
- En la **Figura 63** se muestra la actualización del plan del proyecto.

#### **Figura 63**

*Actualización del plan del proyecto del sprint 3 finalizado.*

Nombre de la tarea	Responsable	Fecha de inicio	Fecha final	Días	Estado
Sprint 1	Marco	9/11	9/22	11	Finalizado
MBD	Marco	9/11	9/22	11	Finalizado
Sprint 2	Marco	9/25	10/6	11	Finalizado
MIBD	Marco	9/25	10/6	11	Finalizado
Sprint 3	Marco	10/9	10/16	7	Finalizado
VABD	Marco	10/9	10/16	7	Finalizado
Sprint 4	Roberto	10/9	10/27	18	Finalizado
CEP	Marco	10/9	10/27	18	Finalizado
Sprint 5	Roberto	11/6	11/17	11	Finalizado
HU-1	Marco	11/6	11/10	4	Finalizado
HU-2	Marco	11/13	11/17	4	Finalizado
Sprint 6	Roberto	11/20	12/1	11	Finalizado
HU-3	Marco	11/20	11/24	4	Finalizado
HU-4	Roberto	11/22	11/27	5	Finalizado
HU-5	Roberto	11/27	12/1	4	Finalizado



## CONCLUSIONES

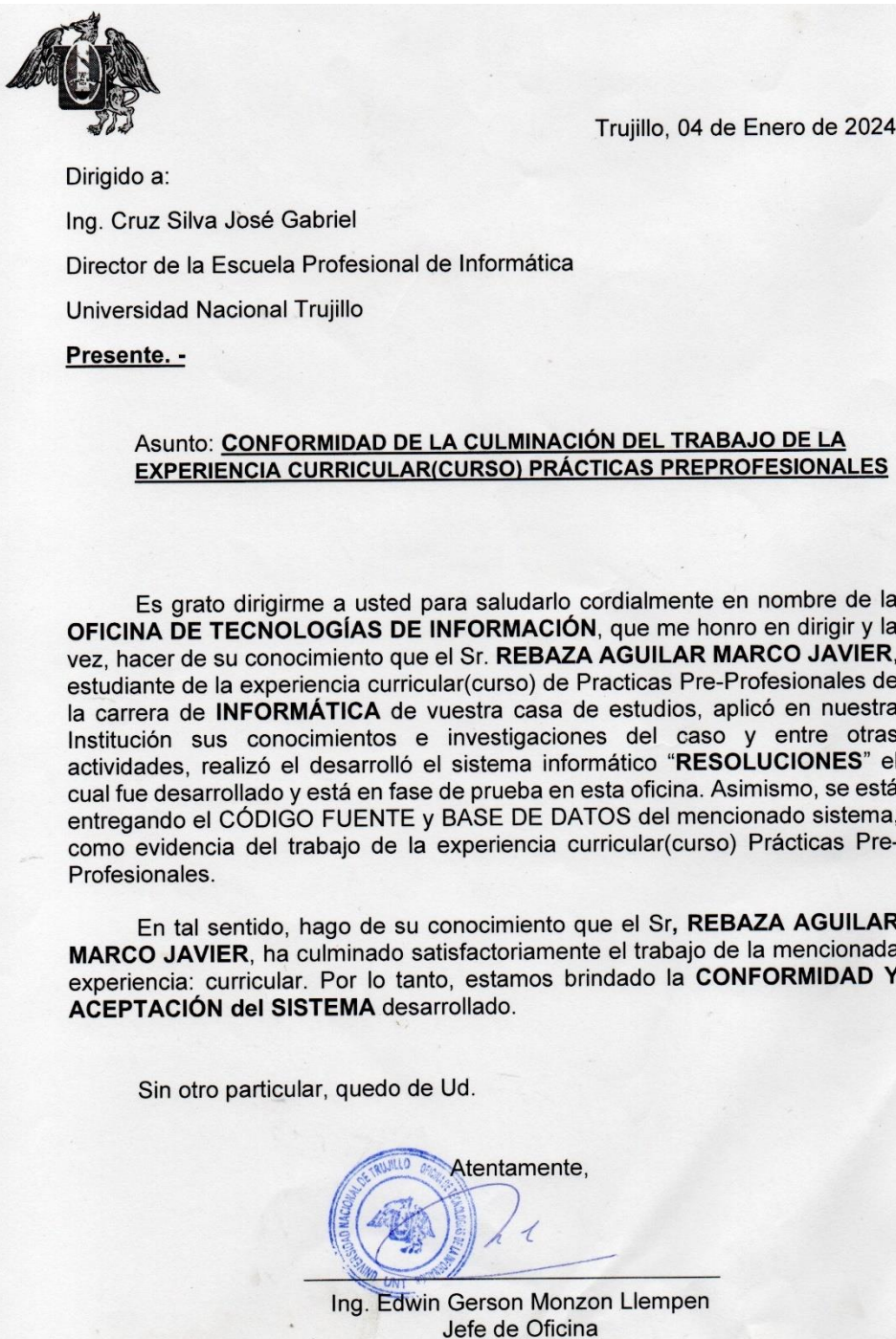
- Se seleccionó y estableció un entorno propicio para el desarrollo de las prácticas, permitiendo una inmersión efectiva en el proyecto.
- Se adquirió un conocimiento profundo sobre las tareas y responsabilidades, abarcando tanto el desarrollo del backend como del frontend, y la migración de datos.
- Se integró de manera efectiva al ambiente laboral, comprendiendo la dinámica de trabajo y estableciendo relaciones colaborativas con el equipo.
- Se llevó a cabo una recolección exhaustiva de información, facilitando la comprensión completa del modelo de la base de datos existente.
- Se seleccionó una metodología informática adecuada, estableciendo roles y responsabilidades de manera eficiente para el desarrollo del nuevo sistema.
- Se recopiló información detallada sobre el sistema actual, identificando áreas de mejora y oportunidades para la optimización.
- Se desarrollaron historias de usuario conforme a la metodología informática, y se analizó y creó un nuevo modelo de la base de datos.
- Se realizó con éxito la migración de tablas al sistema de gestión de base de datos PostgreSQL, garantizando la integridad de los datos.
- Se adquirió conocimiento en la arquitectura de NestJS y se logró la conexión efectiva entre la base de datos y el framework.
- Se inició y continuó el desarrollo de endpoints según los casos de uso, incluyendo la creación de endpoints para la gestión de archivos, resoluciones y actas.
- Se cumplió con éxito con los objetivos semanales, incluyendo la actualización de rutas, implementación de mantenedores y diseño de interfaces específicas.

## REFERENCIAS BIBLIOGRÁFICAS

- Scrum.org. (s.f.). What is scrum? <https://www.scrum.org/resources/what-is-scrum>
- Universidad Nacional de Trujillo. (s.f.). Organigrama. <https://bit.ly/3t0UVPf>
- Oficina de Tecnologías de la Información. (s.f.). Nosotros. <https://bit.ly/3Rrah9S>
- Oficina de Tecnologías de la Información. (s.f.). Unidad de Desarrollo de Sistemas de Información (UDSI). <https://bit.ly/3PwfHOk>
- Plataforma del Estado Peruano. (s.f.). Universidad Nacional de Trujillo – UNT. <https://www.gob.pe/unitru>
- Universidad Nacional de Trujillo. (s.f.). Misión. <https://bit.ly/452X2iH>

## ANEXOS

ANEXO I: Copia escaneada de Constancia de Prácticas emitida por la empresa donde se realizó la práctica preprofesional (obligatorio).



## ANEXO II: Ambiente laboral en el cual se desarrollarán las prácticas preprofesionales



## ANEXO III: Temario proporcionado por el SCRUM Master

La Intranet [la.intranet@uniru.edu.pe](mailto:la.intranet@uniru.edu.pe) puede probar ingresando con tu usuario y Contraseña, ya que tu también entregaste credenciales.

En el sistema de control de versiones gitlab(<http://versiones.uniru.edu.pe/>) ya te he brindado permisos para que puedas visualizar otros sistema Frontend y Backend de lo que

El lenguaje de programación principal es **Javascript**.  
La curva de aprendizaje para estas nuevas tecnologías es mediana, creo que de acuerdo a tus expertis en 3 semanas puedes dominar lo básico e intermedio y para migrarlo a meses aprox.

**Temario:**

1. **Git** (Ya hemos trabajado con solvencia), tenemos servidor Gitlab
2. **Postgresql**, brindaré accesos a la bd\_intranet
3. **Backend Nodejs** (tutorial de Fazt [aquí](#)).  
Usaremos ORM **Sequalize** para conectar a la base de datos como mysql, sql server, postgres, mongodb.  
La autenticación es con **Json Web Token**.  
Crearemos ApiRest por subsistemas.  
Luego pasamos a el Framework **Feathersjs** que estamos usando, este Framework ya incluye o nestjs(aun por definir).
4. **Frontend Vuejs version 2**, tutorial Bluuweb de youtube [aquí](#)  
Usaremos **Quasar Framework** para el frontend. <https://v1.quasar.dev/>  
Tutorial de Quasar [aquí](#)  
Usaremos axios para consumir apis.
5. **Reportes**: Tenemos un servidor **JasperServer community** y utilizamos **jaspersoft studio** para la creación de reportes, consumimos estos reportes desde el backend.
6. **Files**: Tenemos un servidor **Nextcloud** para la gestión de archivos, consumimos estos archivos desde el backend.