

FACULTAD DE INGENIERÍA

ASIGNATURA: TALLER DE PROYECTOS 2

SEMESTRE: 2024- 10

**NOMBRE DEL PROFESOR: GAMARRA MORENO
DANIEL**

INTEGRANTES:

1. GRANDE CONTRERAS, Mario
2. ROLDAN HINOJOSA, Jhon
3. VERA CHINCHAY, Javier
4. MALPARTIDA ZEVALLOS, Franco Luis
5. BALDEON LARA, Jhonatan Moisés
6. COLLACHAGUA VENTURA JUNIOR
7. SANABRIA TINOCO, Williams Anthony
8. JAUREGUI PONCE Jhon Kennedy
9. POVIS URIBE Jean Anthony
10. Diaz Loayza Diego Juan

**FECHA DE ENTREGA : 9/04/2024
HUANCAYO – PERU
2024**

Capítulo I: Sprint 0	3
1.1 Configuración del Entorno	3
1.2 Producto Inicial	5
1.2.1 Definición del Producto Inicial	5
1.2.2 Visión inicial del producto	6
1.2.3 Identificar las funcionalidades básicas	6
1.2.4 Borrador product Backlog	7
1.3 Creación de Equipos	7
1.4 Capacitación	8
1.5 Definición de las Reglas del Equipo	8
1.6 Planificación de Infraestructura	9
1.7 Gestión de Proyectos con enfoque Kanban	9
1.7.1 Tableros Kanban	9
1.7.2 Gestión de proyectos	9
1.7.3 Seguimiento del progreso	10
1.7.4 Flujo de trabajo.	10
1.8 Gestión Efectiva del Product Backlog	10
1.8.1 Refinar el product backlog	11
1.8.2 Priorizar el product backlog	11
1.8.3 Historias de usuario según INVEST	11
1.09 Planificación y Ejecución del Primer Sprint	11
1.09.1 Historias de usuario del primer sprint	11
1.10 Gestión de Reuniones en Scrum	22
1.10.1 Preparación para las Reuniones de Scrum:	22
Para seleccionar las historias de usuario y definir las tareas.	22
1.10.2 Preparación de las reuniones diarias de Scrum: Para el seguimiento del progreso y la identificación de obstáculos.	23
1.10.3 Preparación de la reunión de planificación del sprint: Para demostrar las funcionalidades completadas.	23
1.10.4 Preparación de la reunión de revisión del sprint	23
1.10.5 Preparación de la reunión de la Retrospectiva del Sprint:	23
Para revisar el sprint y planificar mejoras.	23

Capítulo I: Sprint 0

1.1 Configuración del Entorno

Proyecto: Desarrollo de un sistema web para estudiantes académicos de odontología, con un servidor web local para pruebas y luego desplegar en un servidor en la nube (Azure).

Paso 0: Configuración de Docker

Descargar e instalar Docker

<https://www.docker.com/products/docker-desktop/>

Descargar e instalar WSL2

<https://learn.microsoft.com/es-es/windows/wsl/install>

Paso 1: Entorno de desarrollo integrado (IDE)

Descargar e instalar un IDE como Visual Studio Code (VS Code), que es ampliamente utilizado y ofrece muchas extensiones útiles para el desarrollo web.

<https://code.visualstudio.com/docs/?dv=win>

Descargar e instalar el motor de base de datos MySQL

<https://dev.mysql.com/downloads/mysql/8.0.html>

Paso 2: Configuración GitHub

Instalar Git en tu sistema para el control de versiones.

<https://github.com/git-for-windows/git/releases/download/v2.42.0.windows.2/Git-2.42.0.2-64-bit.exe>

Configurar tu nombre de usuario y dirección de correo electrónico en Git.

```
git config --global user.name "Tu Nombre"
```

```
git config --global user.email "tu@email.com"
```

Paso 3: Instalar Node.js y npm

Descargar e instalar Node.js, que incluye npm (Node Package Manager), para gestionar paquetes y dependencias de JavaScript.

<https://code.visualstudio.com/docs/?dv=win>

Paso 4: Crear una carpeta de proyecto

Crea una carpeta en tu sistema dónde alojar tu proyecto web.

Paso 5: Inicializar un repositorio Git o clona el repositorio

Inicia un repositorio Git en la carpeta del proyecto.

```
> git init
```

```
> git clone "repositorio" "nombre que quieres darle"
```

Repositorio:

<https://github.com/MarioJGC/proyecto-historia-clinica.git>

Paso 6: Abrir el proyecto en tu IDE

Abre la carpeta del proyecto en tu IDE (VS Code).

Paso 7: Instala los framework tanto para frontend y backend

Segun corresponda al rol instala el framework requerido frontend:

```
> npm install -g @angular/cli
```

Segun corresponda al rol instala el framework requerido backend:

```
> npm install -g @nestjs/cli
```

Paso 8: Crear el proyecto backend y frontend

Crea el proyecto mediante los comandos frontend:

```
> ng new "nombre del proyecto"
```

backend:

```
> nest new "nombre del proyecto"
```

Paso 9: Ingresar a la carpeta del proyecto backend y frontend

Crea el proyecto mediante los comandos frontend:

```
> cd ./muelitas-front
```

backend:

```
> cd ./muelitas-back
```

Paso 10: Instalar dependencias y dependencias de desarrollo en el proyecto

Leer archivo README.md adjuntado en el repositorio de github.

Paso 11: Crear módulos, componentes, interfaces, servicios, etc

Crea mediante los comandos frontend:

```
> ng new nombre-del-proyecto // Crea un nuevo proyecto Angular.  
> ng generate module nombre-del-modulo // Genera un nuevo módulo Angular.  
> ng generate component nombre-del-componente // Crea un nuevo componente.  
> ng generate service nombre-del-servicio // Genera un nuevo servicio Angular.  
> ng generate class nombre-de-la-clase // Crea una nueva clase TypeScript.  
> ng generate interface nombre-de-la-interfaz // Genera una nueva interfaz TypeScript.  
> ng generate pipe nombre-del-pipe // Crea un nuevo pipe (filtro) Angular.  
> ng generate directive nombre-de-la-directiva // Genera una nueva directiva Angular.  
> ng generate guard nombre-del-guard // Crea un nuevo guardia para la protección de rutas.  
> ng generate application nombre-de-la-aplicacion // Genera una nueva aplicación dentro del  
proyecto Angular existente para permitir múltiples aplicaciones en un solo proyecto.
```

Paso 12: Inicializa el proyecto

Inicializa el proyecto mediante el comando

frontend:

```
ng serve -o
```

backend:

```
yarn start:dev
```

Paso 13: Actualiza el repositorio

Antes de empezar ejecuta el siguiente comando

```
git pull origin "rama asignada"
```

Paso 14: Sube tus cambios

git add .	guardar todo
git commit -m "comentario breve de lo que hiciste"	comentar lo hecho
git checkout -b "nueva rama creada"	crear una rama
git push origin "rama asignada"	subir a la rama

Paso 15: Desarrolla y sigue probando

Inicializa el proyecto mediante el comando

frontend:

ng serve -o

backend:

yarn start:dev

Paso 16: Unir ramas (solo encargados)

git switch "nombre rama destino"

git checkout "nombre rama destino"

git merge "nombre rama origen"

ojo: si hay conflictos de merge resolverlo y publicar en otra rama "de preferencia dev"

Paso 17: Crear el proyecto para subirlo a un servidor en la nube

Comando para crear el proyecto para subirlo en un hosting o vps

frontend:

ng build --prod

backend:

yarn build

frontend: angular - bootstrap

backend: typescript - Nest js

orm: TypeOrm

base de datos: mysql

1.2 Producto Inicial

1.2.1 Definición del Producto Inicial

La definición del Producto Inicial es un paso fundamental en el desarrollo ágil de productos, ya que establece las bases para la planificación y ejecución del proyecto. Permite al equipo de desarrollo y al Product Owner tener una comprensión compartida de lo que se espera lograr y cuáles son las funcionalidades clave que deben entregarse en las primeras etapas del desarrollo.

Proyecto: Desarrollo de un sistema web para estudiantes académica de odontología, con un servidor web local para pruebas y luego desplegar en un servidor en la nube (INMOTION HOSTING).

Sistema web odontológico “Muelitas felices”

Mudar el actual sistema que se tiene y mejorarlo lo más que se pueda para tener un seguimiento de los pacientes por las sesiones que pase en los consultorios y ver el progreso de mejora mediante el odontograma.

1.2.2 Visión inicial del producto

Visión del Proyecto

Desarrollar un sistema que, a través de la vanguardia en innovación tecnológica, garantice la seguridad de los datos, optimice la eficiencia operativa y promueva la formación avanzada de estudiantes en odontología. Este sistema busca proporcionar una atención de primera categoría, mejorar la calidad de vida de los pacientes y preparar a los profesionales del futuro para los desafíos de la odontología.

Visión de la Carrera de Odontología

Servir y solucionar los problemas de salud de su entorno, al romper esquemas y proponer nuevas alternativas en los diagnósticos y tratamientos de la odontología, mostrando el liderazgo profesional.

Funcionalidades básicas

1. **Gestión de Historias Clínicas:** Permite registrar información detallada sobre los pacientes, incluyendo datos personales, antecedentes médicos y odontológicos.
2. **Odontograma Interactivo:** Facilita la creación y visualización de odontogramas que ayudan a planificar y llevar a cabo procedimientos odontológicos de manera efectiva.
3. **Diagnóstico y Tratamiento:** Permite registrar diagnósticos, tratamientos y seguimientos de los pacientes, garantizando un registro completo de la atención.
4. **Generación de Informes:** Puede generar informes detallados que respaldan la toma de decisiones clínicas informadas.
5. **Gráficos de Evolución:** Ayuda a seguir el progreso de la salud bucal del paciente a lo largo del tiempo, mediante gráficos y registros visuales.
6. **Facilidad de Acceso y Búsqueda:** Permite acceder rápidamente a la información de los pacientes y buscar registros específicos.
7. **Cumplimiento Normativo:** Cumple con los estándares y regulaciones de privacidad de datos para garantizar la confidencialidad de la información del paciente.

1.3 Creación de equipos

INTEGRANTE	ROL
Baldeon Lara Jhonatan Moises	Equipo de desarrollo (base de datos)
Sanabria Tinoco Wiliams Anthony	Equipo de desarrollo (backend)
Vera Chinchay Javier	Equipo de desarrollo (fronted)
Povis Uribe Jean Anthony	Equipo de desarrollo (backend)
Jauregui Ponce Jhon Kennedy	Equipo de desarrollo (fronted)
Grande Contreras Mario	Equipo de desarrollo (fronted)
Roldan Hinojosa Jhon	Equipo de desarrollo (base de datos)
Malpartida Zevallos Franco	Equipo de desarrollo (backend)
Diaz Loayza Diego Juan de Dios	Equipo de desarrollo
Collachagua Ventura Junior Jesus	Equipo de desarrollo

1.4 Capacitación

La capacitación juega un papel crucial en la gestión de proyectos, y en este contexto, el hecho de que la mayoría de los estudiantes estén familiarizados con Scrum representa una ventaja considerable. Scrum es una metodología ágil ampliamente aceptada y utilizada en la industria de la gestión de proyectos, lo que facilita su implementación en este proyecto mediano. Al contar con un equipo que ya tiene experiencia en Scrum, podemos aprovechar este conocimiento para asegurar una planificación, ejecución y control del proyecto más efectivos.

1.5 Definición de las Reglas del Equipo

Estas reglas del equipo proporcionan un marco claro para la colaboración y la comunicación efectiva, lo que contribuye a un ambiente de trabajo productivo y colaborativo en el equipo de desarrollo de software. Es importante que todos los miembros del equipo estén al tanto de estas reglas y se adhieran a ellas para lograr un funcionamiento armonioso y eficiente.

Reglas de Generales

- Los encargados son los responsables de entregar el trabajo solicitado a su grupo en el tiempo solicitado y de manera.
- Los miembros del equipo deben comprometerse a contribuir de manera activa y responsable al proyecto.

- Se espera que los miembros del equipo se comuniquen de manera abierta y honesta. Esto incluye compartir actualizaciones sobre el progreso del trabajo, expresar cualquier impedimento o problema que puedan enfrentar, y solicitar ayuda cuando sea necesario.
- Establecer reuniones periódicas, como reuniones de equipo, reuniones de seguimiento o reuniones de retroalimentación, para asegurarse de que todos estén alineados en los objetivos del proyecto y para abordar cualquier preocupación o problema de manera oportuna.
- Cada miembro del equipo es responsable de completar sus tareas asignadas en tiempo y forma. Si un miembro del equipo no cumple con esta responsabilidad, se aplicarán consecuencias, como la disminución del porcentaje de participación en el proyecto de 3% por cada tarea no entregada y no se le va a considerar en las presentaciones de las tareas.
- Si un miembro del equipo no puede cumplir con un plazo, debe informar al equipo lo antes posible para que se puedan tomar las medidas necesarias.
- Todos los miembros del equipo deben seguir los estándares y buenas prácticas de desarrollo de software establecidos por la organización.
- Comentar cualquier funcionalidad para entender de qué se trata.
- Cuando surjan desacuerdos o conflictos, el equipo debe abordarlos de manera constructiva y profesional.
- Antes de entregar cualquier trabajo o funcionalidad, se deben realizar pruebas exhaustivas para asegurarse de que el software funcione correctamente y cumpla con los requisitos.

Reglas de Documentación

- El equipo debe establecer estándares claros para la documentación, incluyendo formatos, estructuras y herramientas a utilizar. Esto garantiza que la documentación sea coherente en todo el proyecto.
- Antes de que comience el desarrollo de una nueva funcionalidad, se debe crear una documentación detallada de las definiciones de usuario o historias de usuario. Esto incluye los criterios de aceptación y los requisitos funcionales.
- Todas las reuniones de Scrum, como las reuniones diarias de stand-up, las reuniones de planificación de sprint y las retrospectivas, deben estar documentadas adecuadamente. Esto ayuda a rastrear el progreso y las decisiones tomadas durante el sprint.
- Cada tarea o funcionalidad desarrollada debe incluir documentación detallada de las pruebas realizadas, tanto las pruebas de unidad como las pruebas de integración.
- La documentación debe mantenerse actualizada a lo largo del proyecto. Cualquier cambio en los requisitos, la arquitectura o el diseño debe reflejarse de manera oportuna en la documentación correspondiente.
- Programar revisiones regulares de la documentación para garantizar su calidad y precisión. Estas revisiones pueden ser parte de las tareas de revisión de sprint o llevarse a cabo de manera separada.

- Se debe de asegurarse de que todos los miembros del equipo tengan acceso a la documentación relevante y establecer una política clara de gestión de permisos para garantizar que la información esté disponible solo para aquellos que la necesiten.

Reglas de Backend

- Ser claros con las ubicaciones donde se realizarán las modificaciones.
- Implementar medidas de seguridad adecuadas para garantizar el acceso seguro a las funcionalidades.
- Generar una documentación clara, que incluya ejemplos de solicitudes y respuestas, para facilitar la integración con el frontend
- Escribir pruebas unitarias para asegurar la calidad del código. Automatizar las pruebas siempre que sea posible.
- Manejar los errores de manera adecuada y proporcionar respuestas de error claras y útiles
- Colaborar con el equipo de Base de Datos para asegurarse de que el diseño de la base de datos sea coherente con las necesidades del backend.

Reglas de Frontend: Diseño

- Los colores serán de acuerdo a la empresa que colabore con nosotros.
- Priorizar lo estético al momento de diseñar (no mucho texto).
- Al realizar listas solo colocar unas cinco de ejemplo debido a que al unir con lo programado solo se necesita uno para que el bucle corra en eso.
- Tratar de poner barras de estado cuando ya estemos en los procesos.
- Se tiene que poder tener responsividad excepto cuando tengamos el odontograma.
- El dashboard si o si tendrá un navegador, sidebar y su contenido, por supuesto.
- Para el odontograma diseñar solo para desktop.

Reglas de Frontend: Programación

- Comenzar con el crud para los usuarios(administradores, odontólogos, recepcionista, etc).
- Las tareas asignadas serán según el rol (administrador, docente o estudiante) que elijas para no desviarte.
- Basaremos los modelos y servicios de acorde al backend.
- Crear toast "mensaje de estado" para probar tu avance o con console.log("mensaje").
- Esperaremos a los de diseño para luego unir lo programado con lo visual.
- Para los procesos de la empresa nos basaremos en bpmn que se diseñe.

Reglas de Base de Datos

- Programar revisiones del progreso del diseño de la base de datos.
- Utilizar la misma versión del software para el desarrollo de la base de datos.
- Crear usuarios para todos (según rol elegido administrador, doctor, recepcionista, etc).

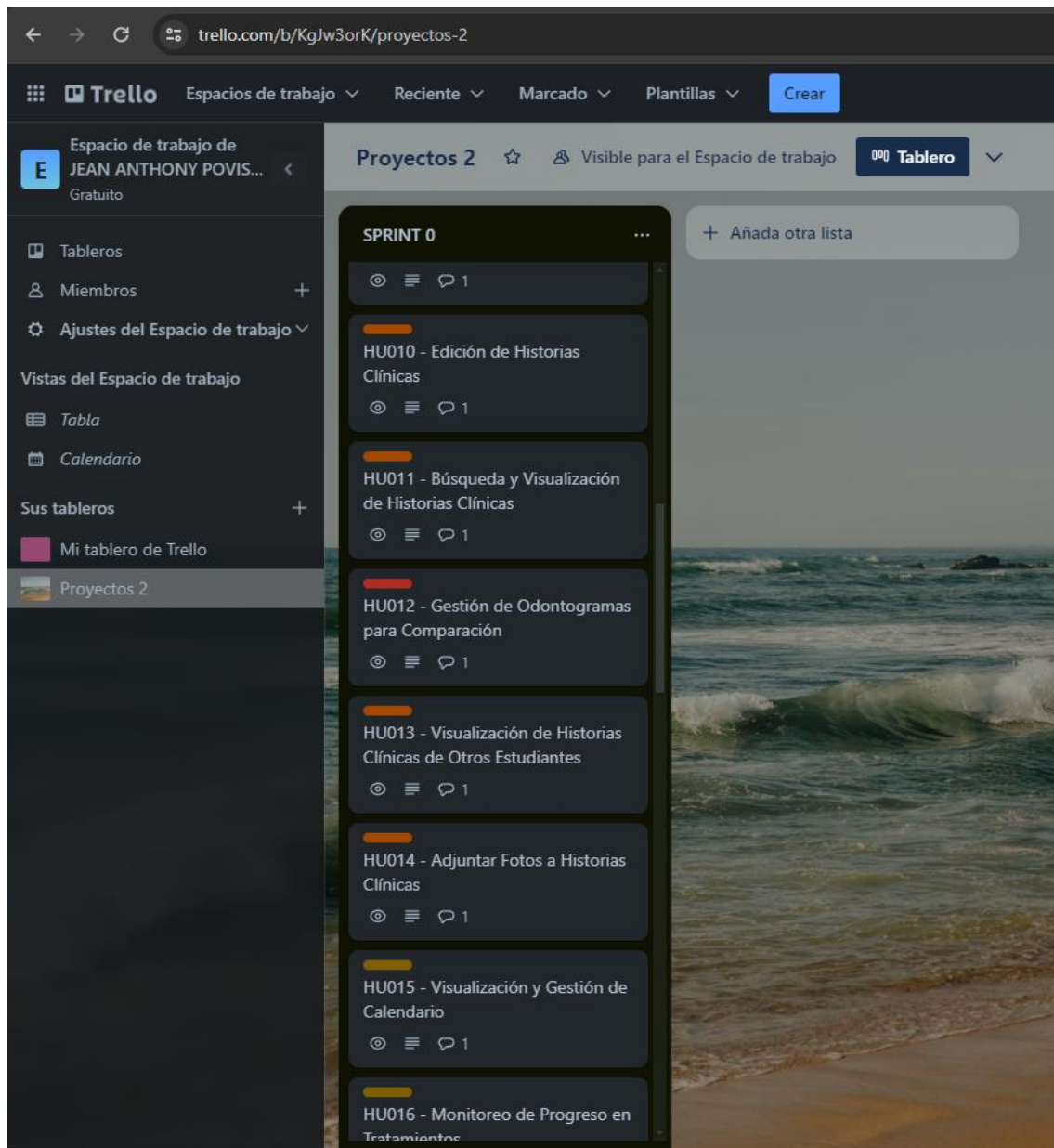
1.6 Planificación de Infraestructura

Entornos de Desarrollo y Pruebas:

- **Identificación de Entornos:**
 - **frontend:** Angular - TypeScript - Bootstrap
 - **backend:** Nest js - TypeScript
 - **orm:** TypeOrm
 - **base de datos:** Mysql
 - **Cloud:** Docker
- **Herramientas:**
 - IDE: Visual Studio Code
 - Modelado: Bizagi y Erwin Data Modeler
 - Diseño: Photoshop, Adobe Illustrator, Corel Draw
 - Presentación: Canva

1.7 Tablero de Desarrollo de Historias de Usuario

<https://trello.com/b/KqJw3orK/proyectos-2>



Borrador del Product Backlog:

1.8 Gestión Efectiva del Product Backlog

[2023/10/01_05_Backlog Detallado del Producto.xlsx](#)

[illegible]

HISTORIA DE USUARIO				OTRO1 DATO 1 DE		HISTORIA DE USUARIO			
ID de la ra	Como (Rol)	Desco	Para	Criterio de Aceptación	Epica	Edición de	Dependencias	Epicini	Estado
HU01	Anal instructor o	Gestionar docentes y estudiantes	Acceder a Software	Se debe poder buscar a los docentes registrados en el	1	59	-	1	Por Hacer
HU08	Estudiante	Poder acceder a todos los cursos	Registrar nuevos casos	El formulario debe captar un email y validar todos los	1	48	OTRO2	2	Por Hacer
HU12	Estudiante	Ajustar la foto de paciente a foto (historia del paciente, la	complementar	Las fotos y radiografías se cargarán y validarán correctamente	1	40	HU09	2	Por Hacer
HU15	Estudiante	Subir al sistema la firma del paciente en el documento	Garantizar el	Debe tener la funcionalidad que permita cargar la firma del	1	35	HU09	3	Por Hacer
HU17	Estudiante	Los odontólogos tienen la opción de inicio de caso de	documentar y con	Se pueden visualizar y comparar ambas versiones del	1	35	HU09	3	Por Hacer
HU18	Estudiante	Registrar odontograma al inicio y al final del paciente	Registrar la información	Debe haber una interfaz para registrar el odontograma	1	75	HU19	3	Por Hacer
HU19	Estudiante	Visualizar odontograma de seguimiento del paciente que me fue	Registrar información de los	Debe mostrar una interfaz donde se registre la información de	1	75	HU19	4	Por Hacer
HU20	Estudiante	Modificar el odontograma de seguimiento del paciente que me fue	Mantener actualizada la	Debe buscar y seleccionar al paciente derivado	1	75	HU20	4	Por Hacer
HU22	Docente	Acceder al historial clínico completo de un paciente	Superar y gestionar los	Debe mostrar información relevante como la causa, quien lo	1	50	HU20	5	Por Hacer
HU23	Docente	Registrar el historial clínico de un paciente	Registrar la información de	Debe poder visualizar el historial de	1	50	HU20	5	Por Hacer
HU25	Docente	Recibir notificaciones de historias clínicas donde se requiera de	Tener conocimiento de los	Se debe avisar al registro de función de notificaciones	1	50	HU25	5	Por Hacer
HU26	Docente	Evaluar y decidir si aprueba o rechaza las actividades que va a	Garantizar la calidad de su	Debe notificar a los docentes la actividades a realizar por	1	50	HU26	5	Por Hacer
HU09	Anal instructor o	Tener la opción de gestionar personal académico, técnicos, aux	Asignar a docentes y	Debe mostrar la interfaz para la asignación de cursos, cursos	2	50	HU26	1	Por Hacer
HU09	Anal instructor o	Realizar historias clínicas	Actualizar y encontrar la	Se pueden actualizar y encontrar la información de	2	50	HU26	1	Por Hacer
HU13	Estudiante	Visualizar un manual de capacitación con días y actividades	Organizar mejor	Las fotos y radiografías se cargarán y validarán correctamente	3	53	HU09	2	Por Hacer
HU16	Docente	Visualizar y asignar una calificación a cada estudiante basándose en	Ver el avance de	Debe tener la funcionalidad para el ingreso de calificaciones	3	55	HU15	2	Por Hacer
HU05	Anal instructor o	Realizar sesión de	Realizar la sesión de	Se debe registrar salud y condiciones	4	24	HU11	2	Por Hacer
HU11	Anal instructor o	Ver historias clínicas de otros estudiantes	Aprender de otros casos	Se pueden visualizar historias clínicas generales en emails	4	24	HU11	2	Por Hacer
HU14	Estudiante	Visualizar el record de tratamientos dentales asignados al inicio de	Seguimiento de cumplimiento	La lista de tratamientos asignados se muestra claramente	4	24	HU09	2	Por Hacer
HU21	Estudiante	Imprimir odontograma	Generar información de	Debe tener la opción de guardar o imprimir el odontograma	5	50	HU21	4	Por Hacer
HU10	Anal instructor o	Buscar y actualizar mis historias clínicas	Actualizar los registros	La lista debe producir resultados relevantes cuando se hace una	5	50	HU10	4	Por Hacer
HU03	Anal instructor o	Tener opción de recuperar contactos	Acceder a mi cuenta en	Se debe poder realizar un proceso de verificación de identidad	7	55	HU02	1	Por Hacer
HU05	Anal instructor o	Tener la opción de modificar los datos de los usuarios que real	Reafirmar y/o actualizar la	Se debe poder realizar la búsqueda de usuarios a realizar	7	24	HU01	1	Por Hacer
HU04	Anal instructor o	Actualizar los datos de los usuarios o usuarios que se	Actualizar la lista	Se debe poder buscar a estudiantes y docentes que se hayan	8	25	HU04	1	Por Hacer
HU07	Opme	Visualizar y modificar perfil	Actualizar mi información	Debe contar con una interfaz de perfil de usuario para el	9	24	HU01 / HU02	1	Por Hacer

HU001 - Gestionar docentes:

- **Independiente:** Esta historia puede ser implementarse sin depender de otras historias.
- **Negociable:** Los detalles específicos de cómo gestionar docentes pueden ser discutidos y acordados con el equipo.
- **Valuable:** Gestionar docentes es valioso para permitirles acceder al software y utilizarlo de manera efectiva.

- **Estimable:** Es posible estimar el tiempo y recursos necesarios para esta tarea.
- **Small (Pequeña):** La historia es lo suficientemente pequeña como para ser completada en una iteración.
- **Testable:** Se pueden definir criterios de aceptación claros para evaluar si la historia está completa.

HU002 - Iniciar sesión:

- **Independiente:** Puede implementarse de manera independiente.
- **Negociable:** Los detalles sobre cómo iniciar sesión pueden ser discutidos y acordados con el equipo.
- **Valuable:** El inicio de sesión es valioso ya que permite a administradores, docentes y estudiantes acceder al sistema.
- **Estimable:** Es posible estimar el esfuerzo requerido para implementar el inicio de sesión.
- **Small (Pequeña):** La historia es lo suficientemente pequeña como para completarse en una sola iteración.
- **Testable:** Se pueden definir criterios de aceptación para evaluar si el inicio de sesión funciona correctamente.

HU003 - Recuperación de contraseña:

- **Independiente:** Puede implementarse sin depender de otras historias.
- **Negociable:** Los detalles de la recuperación de contraseña pueden ser discutidos y acordados con el equipo.
- **Valuable:** La recuperación de contraseña es valiosa para usuarios que olviden sus contraseñas.
- **Estimable:** Es posible estimar el esfuerzo requerido para implementar esta funcionalidad.
- **Small (Pequeña):** La historia es lo suficientemente pequeña como para completarse en una iteración.
- **Testable:** Se pueden definir criterios de aceptación claros para evaluar si la recuperación de contraseña funciona correctamente.

HU004 - Verificar desempeño de estudiantes:

- **Independiente:** Puede implementarse de manera independiente.
- **Negociable:** Los detalles específicos de cómo verificar el desempeño de los estudiantes pueden ser discutidos y acordados con el equipo.
- **Valuable:** Verificar el desempeño de los estudiantes es valioso para los docentes.
- **Estimable:** Es posible estimar el tiempo y recursos necesarios para esta funcionalidad.
- **Small (Pequeña):** La historia es lo suficientemente pequeña como para completarse en una iteración.
- **Testable:** Se pueden definir criterios de aceptación para evaluar si la funcionalidad funciona correctamente.

HU005 - Autorizar derivaciones de pacientes:

- **Independiente:** Puede ser implementada sin depender de otras historias.
- **Negociable:** Los detalles de cómo autorizar las derivaciones de pacientes pueden ser discutidos y acordados con el equipo.
- **Valuable:** Autorizar derivaciones de pacientes es valioso para la gestión de tratamientos en el sistema.
- **Estimable:** Es posible estimar el esfuerzo necesario para implementar esta funcionalidad.
- **Small (Pequeña):** La historia es lo suficientemente pequeña como para completarse en una iteración.
- **Testable:** Se pueden definir criterios de aceptación claros para evaluar si la funcionalidad de autorización de derivaciones funciona correctamente.

1.9 Planificación y Ejecución del Primer Sprint

1.9.1 Historias de usuario del primer sprint

Épica	ID	Historia de Usuario
E1. Gestión la cuenta de Usuarios	HU001	Como administrador, quiero gestionar docentes para que puedan acceder al software.
	HU023	Como administrador, quiero ser el único quien agregue cuentas de docentes.
	HU024	Como administrador, quiero suspender (restringir el acceso al sistema) de docentes no activos (retirados).
	HU007	Como administrador, deseo la opción de modificar los datos de los usuarios registrados.
	HU022	Como usuario (administrador, docente y estudiante), quiero visualizar y modificar mi perfil para mantener mi información actualizada.

E2. Autenticación y Seguridad	HU002	Como administrador, docente o estudiante, quiero poder iniciar sesión en el aplicativo web para acceder a sus funciones.
	HU003	Como administrador, docente o estudiante, deseo tener la opción de recuperar mi contraseña para acceder a mi cuenta en caso de olvidarla.
	HU025	Como usuario (docente, estudiante), quiero cambiar mi contraseña predeterminada para aumentar mi seguridad.
E3. Gestión Académica	HU008	Como administrador, deseo tener la opción de gestionar periodo académico, NRC, aulas, cursos y/o especialidades para asignar al docente y estudiantes para los tratamientos respectivos.
	HU031	Como docente, quiero poder subir un registro de celdas con los estudiantes y que el sistema cree sus cuentas respectivamente.
E4. Historias Clínicas	HU009	Como estudiante, quiero poder agregar nuevas historias clínicas para registrar nuevos casos.
	HU010	Como estudiantes, quiero editar historias clínicas para actualizar la información de un paciente.
	HU011	Como estudiante, quiero buscar y visualizar mis historias clínicas para estudiar casos específicos.
	HU013	Como estudiante, quiero ver historias clínicas de otros estudiantes para aprender de otros casos.

	HU014	Como estudiante, quiero adjuntar la foto del paciente, la foto intraoral del paciente, la radiografía periapical y radiografía, a las historias clínicas para complementar la información del paciente.
	HU018	Como docente, quiero acceder al historial clínico completo de un paciente para supervisar y gestionar los tratamientos realizados por el estudiante.
	HU019	Como docente, quiero visualizar el historial clínico para la revisión/calificación del paciente atendido por el estudiante.
	HU030	Como estudiante, quiero subir al sistema la firma del paciente en el documento de consentimiento informado.
E5. Odontogramas	HU012	Como estudiante, quiero gestionar dos odontogramas (uno al inicio y otro al final del tratamiento) para documentar y comparar de forma clara el progreso y los resultados obtenidos con cada paciente.
	HU026	Como estudiante, quiero registrar el odontograma inicial, de seguimiento y final del paciente.
	HU027	Como estudiante, quiero visualizar el odontograma de seguimiento del paciente que me fue derivado de otro curso.
	HU028	Como estudiante de semestre superior, quiero modificar el odontograma de seguimiento del paciente que me fue derivado.
	HU029	Como estudiante, quiero poder imprimir el odontograma.

E6. Evaluación y Seguimiento	HU004	Como docente, quiero poder verificar el desempeño de los estudiantes.
	HU005	Como docente, quiero poder autorizar a los estudiantes de Clínica 1 de derivar a sus pacientes a Clínica 2 para continuar con su tratamiento.
	HU006	Como Administrador o docente, quiero poder suspender las cuentas de los estudiantes que se retiren de la asignatura, para tener actualizada la lista de estudiantes activos.
	HU015	Como estudiante, quiero visualizar un manejo de calendario con citas y actividades para organizarme mejor.
	HU016	Como estudiante, quiero visualizar el récord de tratamientos dentales asignados al inicio del ciclo por el doctor, y monitorear mi progreso para asegurarme de cumplir con la meta establecida.
	HU017	Como docente, quiero evaluar y asignar una calificación a cada estudiante basándose en sus tratamientos dentales realizados en comparación con las metas establecidas.
	HU020	Como docente, quiero evaluar y decidir si apruebo o rechazo las actividades que va a realizar el estudiante para garantizar la calidad de su trabajo antes de atender al paciente.
	HU021	Como docente, quiero recibir notificaciones de historias clínicas donde se requiera de una firma.

1.09.2 Tareas de las historias de usuario del primer Sprint

HU001: Como administrador, quiero gestionar docentes para que puedan acceder al software

- **Tarea 1:** Diseñar las funciones de creación y modificación de perfiles de docentes.
- **Tarea 2:** Implementar una función de búsqueda de docentes para facilitar la gestión por parte de los administrativos.
- **Tarea 3:** Diseñar un proceso de confirmación para agregar nuevos docentes al sistema.

HU002: Como administrador, docente o estudiante, quiero poder iniciar sesión en el aplicativo web para acceder a sus funciones.

- **Tarea 1:** Diseñar un formulario de inicio de sesión.
- **Tarea 2:** Implementar la autenticación de usuarios basada en sus credenciales (nombre de usuario y contraseña).
- **Tarea 3:** Configurar mecanismos de encriptación para proteger las contraseñas almacenadas.

HU003: Como administrador, docente o estudiante, deseo tener la opción de recuperar mi contraseña para acceder a mi cuenta en caso olvidarla.

- **Tarea 1:** Diseñar una interfaz de usuario que permita a los usuarios solicitar la recuperación de contraseña.
- **Tarea 2:** Desarrollar un proceso de verificación de identidad seguro para confirmar la identidad del usuario que solicita la recuperación.
- **Tarea 3:** Configurar el envío de correos electrónicos o envío de mensaje por whatsapp de recuperación de contraseña con instrucciones claras y seguras.
- **Tarea 4:** Crear una página para que los usuarios ingresen una nueva contraseña después de la verificación exitosa.
- **Tarea 5:** Implementar medidas de seguridad adicionales, como limitar el número de intentos de recuperación de contraseña en un período de tiempo.

HU004: Como docente, quiero poder verificar el desempeño de los estudiantes

- **Tarea 1:**Mostrar los detalles de la historia clínica.
- **Tarea 2:** Diseñar una interfaz de usuario para verificar.
- **Tarea 3 :** Diseñar un botón para que el docente verifique los trabajos.

HU005: Como docente quiero poder autorizar a los estudiantes de Clínica 1 de derivar a sus pacientes a Clínica 2 para continuar con su tratamiento.

- **Tarea 1:** Elaborar una interfaz que permita autorizar la

derivación de pacientes.

- **Tarea 2:** Completar formulario de registro para solicitar autorización
- **Tarea 3:** Mensaje de validación de formulario guardado.
- **Tarea 4:** Recibir notificación para validar que el paciente derivado va a ser atendido.
- **Tarea 5:** Notificar a los docentes de clínica 1 cuando los pacientes derivados hayan terminado su tratamiento en clínica 2.

HU006: Como Administrador o docente, quiero poder suspender las cuentas de los estudiantes que se retiren de la asignatura, para tener actualizada la lista de estudiantes activos.

- **Tarea 1:** Elaborar una interfaz de administración de cuentas de usuarios.
- **Tarea 2:** Buscar los usuarios de los estudiantes retirados.
- **Tarea 3:** Seleccionar un botón para deshabilitar cuentas y bloquear el acceso.
- **Tarea 4:** Mostrar mensaje de confirmación.
- **Tarea 5:** Guardar en una lista los usuarios suspendidos con fecha y asignatura
- **Tarea 6:** Elaborar una función para habilitar las cuentas si un estudiante vuelve a llevar la asignatura

HU007: Como administrador deseo la opción de modificar los datos de los usuarios registrados.

- **Tarea 1:** Diseñar un interfaz con la opción que permita modificar los datos registrados al momento de la creación de usuarios.
- **Tarea 2:** Realizar la búsqueda de usuario usuario criterios de búsqueda documento o celular, para los docentes correo electrónico.
- **Tarea 3:** Contar un botón para redireccionar a la interfaz de modificación y realizar cambios de la información registrada a los usuarios creados.
- **Tarea 4:** Se mostrará un mensaje de confirmación o error para asegurar si la acción se ejecutó correctamente.

HU008: Como administrador deseo tener la opción de gestionar periodo académico, NRC, aulas, cursos y/o especialidades para asignar al docente y estudiantes para los tratamientos respectivos.

- **Tarea 1:** Diseñar una interfaz con las opciones de creación de aulas y cursos para estudiantes y odontólogos.

- **Tarea 2:** Contar con botones de modificar y eliminar para realizar cambios de las aulas, cursos y/o especialidades.
- **Tarea 3:** Se mostrará un mensaje de confirmación o error para asegurar si la acción se ejecutó correctamente.
- **Tarea 4:** Implementar la opción de impresión de lista de aulas, cursos y/o especialidades.

Registro de Historias Clínicas (como estudiantes):

Gestión de historias clínicas

HU009: Como estudiante, quiero poder agregar nuevas historias clínicas para registrar nuevos casos.

- **Tarea 1:** Diseñar un formulario para la entrada de datos de la historia clínica.
- **Tarea 2:** Implementar validaciones para asegurarse de que todos los campos obligatorios estén completos.
- **Tarea 3:** Registrar odontograma del paciente
- **Tarea 4:** Crear una base de datos o estructura de almacenamiento para las historias clínicas.
- **Tarea 5:** Implementar la lógica para guardar la historia clínica en la base de datos.
- **Tarea 6:** Diseñar e implementar una notificación o mensaje de confirmación tras guardar exitosamente una historia clínica.

HU010: Como estudiantes, quiero editar historias clínicas para actualizar la información de un paciente.

- **Tarea 1:** Implementar una función de búsqueda para encontrar la historia clínica específica que se quiere editar.
- **Tarea 2:** Mostrar la historia clínica en un formulario editable.
- **Tarea 3:** Implementar validaciones para cambios en los campos.
- **Tarea 4:** Crear la lógica para actualizar la historia clínica en la base de datos.
- **Tarea 5:** Diseñar e implementar una notificación o mensaje de confirmación tras editar exitosamente una historia clínica.

HU011: Como estudiante, quiero buscar y visualizar mis historias clínicas para estudiar casos específicos.

- **Tarea 1:** Diseñar e implementar una función de búsqueda basada en diferentes criterios (por ejemplo, nombre del paciente, fecha, condición, etc.).
- **Tarea 2:** Mostrar una lista de resultados que coincidan con la búsqueda.

- **Tarea 3:** Permitir seleccionar una historia clínica de la lista de resultados para verla en detalle.

Documentación y visualización

HU012: Como estudiante, quiero gestionar dos odontogramas (uno al inicio y otro al final del tratamiento) para documentar y comparar de forma clara el progreso y los resultados obtenidos con cada paciente.

- **Tarea 1:** Diseñar una interfaz que permita ver las diferentes versiones de un odontograma.
- **Tarea 2:** Permitir la edición del odontograma cuando sea necesario.
- **Tarea 3:** Implementar un botón para visualizar las modificaciones del odontograma.
- **Tarea 4:** Mostrar la fecha de modificación del odontograma.
- **Tarea 5:** En las historias clínicas, priorizar la primera y la última modificación del odontograma.

HU013: Como estudiante, quiero ver historias clínicas de otros estudiantes para aprender de otros casos.

- **Tarea 1:** Diseñar una interfaz que permita visualizar detalles generales de las historias clínicas subidas a la base de datos.
- **Tarea 2:** Implementar la función de visualizar a detalle la historia de la clínica seleccionada.
- **Tarea 3:** Implementar una función que notifique al estudiante cuando otro estudiante haya descargado su historia clínica.
- **Tarea 4:** Implementar la opción de visualizar y descargar la historia clínica en el formato adecuado.

HU014: Como estudiante, quiero adjuntar la foto de paciente, la foto intraoral del paciente, la radiografía periapical y radiografía, a las historias clínicas para complementar la información del paciente.

- **Tarea 1:** Diseñar una interfaz amigable donde los estudiantes puedan cargar fotos clasificadas por tipo (fotos del paciente, intraorales, radiografías periapicales, etc.).
- **Tarea 2:** Implementar una función que permita a los estudiantes subir fotos de acuerdo a la categoría seleccionada, con la capacidad de previsualizar antes de guardar.
- **Tarea 3:** En la historia clínica, integrar una sección donde se puedan visualizar las fotos subidas, categorizadas por

su tipo.

- **Tarea 4:** Asegurar que la información y las fotos estén almacenadas de forma segura y privada.

Gestión de actividades y evaluación

HU015: Como estudiante, quiero visualizar un manejo de calendario con citas y actividades para organizarme mejor.

- **Tarea 1:** Crear una opción en el menú principal del sistema web que sea intuitiva, permitiendo a los estudiantes acceder rápidamente al calendario.
- **Tarea 2:** Diseñar una interfaz de visualización para el calendario donde se muestren claramente las citas médicas.
- **Tarea 3:** Permitir a los estudiantes navegar por el calendario por fechas y asegurarse de que la transición entre estos modos sea fluida.
- **Tarea 4:** Facilitar a los estudiantes la opción de agregar, editar, eliminar y actualizar citas directamente desde el calendario. Cada entrada debe permitir agregar detalles, como hora, salón y descripción.
- **Tarea 5:** Desarrollar la lógica necesaria para que, al realizar cambios en el calendario, estos se reflejen en la base de datos en tiempo real. Además, al acceder al calendario, este debe cargar las citas y actividades previamente guardadas.

HU016: Como estudiante, quiero visualizar el récord de tratamientos dentales asignados al inicio del ciclo por el doctor, y monitorear mi progreso para asegurarse de cumplir con la meta establecida.

- **Tarea 1:** Diseñar una interfaz de usuario para mostrar la lista de tratamientos dentales asignados al inicio del ciclo.
- **Tarea 2:** Desarrollar una función que recupere desde la base de datos los tratamientos específicos asignados por el docente a cada estudiante.
- **Tarea 3:** Asegurarse de que la interfaz muestre de manera clara cuál es la meta de cada tratamiento.
- **Tarea 4:** Desarrollar un contador o barra de progreso que se actualice en tiempo real cada vez que se registre un tratamiento realizado.
- **Tarea 5:** Permitir al docente ingresar la calificación final basada en los tratamientos realizados y las metas alcanzadas.
- **Tarea 6:** Proporcionar al estudiante una vista para acceder a su calificación y cualquier feedback del docente.

HU017: Como docente, quiero evaluar y asignar una calificación a cada estudiante basándose en sus tratamientos dentales realizados en comparación con las metas establecidas.

- **Tarea 1:** Diseñar una interfaz que muestre una lista de estudiantes junto con su progreso.
- **Tarea 2:** Implementar una función para que el docente pueda ingresar calificaciones y comentarios.
- **Tarea 3:** Permitir al docente proporcionar feedback específico sobre áreas de mejora o destacar logros particulares.
- **Tarea 4:** Crear una sección donde los estudiantes puedan ver sus calificaciones y comentarios.
- **Tarea 5:** Implementar notificaciones para informar a los estudiantes cuando su calificación esté disponible.

Revisión de Historias Clínicas (Docente):

HU018: Como docente, quiero acceder al historial clínico completo de un paciente para supervisar y gestionar los tratamientos realizados por el estudiante.

- **Tarea 1:** Interfaz que muestre a los pacientes de la clase vinculado al alumno que realizó el tratamiento filtrado por fecha de clase
- **Tarea 1:** Interfaz que muestre a los estudiantes junto con el paciente atendido filtrado por fecha de clase
- **Tarea 2:** seleccionar al paciente
- **Tarea 3:** mostrar tratamiento hecho por el estudiante
- **Tarea 4:** observar o aprobar el tratamiento realizado
- **Tarea 5:** mostrar mensaje de estado cambiado
- **Tarea 6:** redireccionar al listado de la tarea 1

HU019: Como docente, quiero visualizar el historial clínico para la revisión/calificación del paciente atendido por el estudiante.(HU0XX)

- **Tarea 1:** Visualizar el listado de estudiantes de mi nrc que toca.
- **Tarea 2:** Permitir que el docente seleccione un estudiante específico cuyo historial clínico de su paciente desea revisar y calificar.
- **Tarea 3:** Visualizar lista de pacientes tratados por estudiantes.
- **Tarea 4:** Seleccionar paciente para ver su historial clínico mostrando al docente la información relevante del paciente, que puede incluir diagnósticos previos, tratamientos realizados y cualquier otra información relevante.

- **Tarea 5:** Campo que permita que el docente agregue comentarios específicos sobre el desempeño del estudiante en el tratamiento dental.
- **Tarea 6:** Opción para guardar comentarios y calificación del docente.
- **Tarea 7:** Notificar al estudiante que su retroalimentación ya está disponible.
- **Tarea 8:** Redireccionar a la tarea 1 por si el docente quiere seguir dando feedback.

HU020: Como docente, quiero evaluar y decidir si apruebo o rechazo las actividades que va a realizar el estudiante para garantizar la calidad de su trabajo antes de atender al paciente.

- **Tarea 1:** El Estudiante realiza su plan de actividades para su paciente.
- **Tarea 2:** Notifica al Docente el plan de actividades a realizar por el Estudiante.
- **Tarea 3:** El Docente evalúa las actividades.
- **Tarea 4:** Si es aprobado, se le informará al Estudiante para que inicie con sus actividades, en caso de que sea desaprobado se le notificará al Estudiante para generar sus correcciones.
- **Tarea 5:** El estudiante realiza las modificaciones y se notifica la acción al docente y se regresa a la tarea 3.
- **Tarea 6:** El estudiante realiza un cronograma de actividades y va atendiendo según plan de actividades.
- **Tarea 7:** El estudiante registra la actividad realizada en la fecha que se realice la actividad.

HU021: Como docente, quiero recibir notificaciones de historias clínicas donde se requiera de una firma.

- **Tarea 1:** Estudiante termina de registrar datos que requieren firma.
- **Tarea 2:** Asociar al registro función de notificaciones de requerimiento de firma por parte del docente.
- **Tarea 3:** Docente recibe la notificación en su bandeja de notificaciones del sistema(debatible).
- **Tarea 4:** Ver listado en la bandeja de notificaciones del sistema.
- **Tarea 5:** Seleccionar actividad de la bandeja de notificaciones del sistema y redireccionar donde corresponda.
- **Tarea 6:** Funcionalidad para firmar u observar la actividad realizada.
- **Tarea 7:** Testear.

Configuración de cuenta y Odontograma:

HU022: Como usuario(administrador, docente y estudiante), quiero visualizar y modificar mi perfil para mantener mi información actualizada.

- **Tarea 1:** Diseñar la interfaz de perfil para el usuario.
- **Tarea 2:** Implementar funcionalidad para editar el perfil del usuario.
- **Tarea 3:** Realizar pruebas de los datos modificados correctamente.
- **Tarea 4:** Realizar pruebas de los datos almacenados en la BD.

HU023: Como administrador, quiero ser el único quien agregue cuentas de docentes.

- **Tarea 1:** Diseñar la interfaz de administrador para la creación de cuentas.
- **Tarea 2:** Implementar la funcionalidad de agregar cuenta para los docentes.
- **Tarea 3:** Verificar los datos de la cuenta.

HU024: Como administrador, quiero suspender (restringir el acceso al sistema) la cuenta de docentes no activos (retirados).

- **Tarea 1:** Diseñar la interfaz que permita seleccionar las cuentas de docentes a suspender (cambio de estado).
- **Tarea 2:** Implementar la funcionalidad de suspensión cuenta de docentes.
- **Tarea 3:** Implementar una ventana emergente de confirmación de suspensión de cuenta.
- **Tarea 4:** Permitir al administrador seleccionar un docente libre.
- **Tarea 5:** Realizar pruebas del funcionamiento de la eliminación de cuenta.

HU025: Como usuario (docente, estudiante), quiero cambiar mi contraseña predeterminada para aumentar mi seguridad.

- **Tarea 1:** Diseñar la interfaz que permita el cambio de contraseña
- **Tarea 2:** Implementar una ventana emergente de confirmación de cambio de contraseña.
- **Tarea 3:** Implementar funcionalidad de cambio de contraseña.
- **Tarea 4:** Validar el cambio de contraseña.

HU026: Como estudiante quiero registrar el odontograma inicial, de seguimiento y final del paciente.

- **Tarea 1:** Definir el formato y la estructura del odontograma, especificando los datos que deben ser registrados para los odontogramas inicial y final.
- **Tarea 2:** Diseñar y desarrollar una interfaz de usuario que permita al estudiante ingresar la información del odontograma.
- **Tarea 3:** Crear funciones de validación para asegurar que los datos ingresados en el odontograma sean coherentes y cumplan con los requisitos establecidos.
- **Tarea 4:** Asegurarse de que la información esté vinculada al paciente correspondiente y sea fácilmente accesible para su recuperación.

HU027: Como estudiante quiero visualizar el odontograma de seguimiento del paciente que me fue derivado de otro curso.

- **Tarea 1:** Diseñar y desarrollar una interfaz que permita a los estudiantes visualizar el odontograma de seguimiento del paciente derivado de otro curso.
- **Tarea 2:** Integrar la funcionalidad con el sistema de gestión de pacientes para que los estudiantes puedan acceder a los odontogramas derivados de otros cursos.
- **Tarea 3:** Implementar un sistema de notificación automática que alerte al estudiante cuando se le haya asignado un paciente derivado de otro curso.

HU028: Como estudiante de semestre superior quiero modificar el odontograma de seguimiento del paciente que me fue derivado.

- **Tarea 1:** Configurar un sistema de verificación que permita a los estudiantes acceder a las historias clínicas de los pacientes que les fueron derivados.
- **Tarea 2:** Permitir a los estudiantes buscar y seleccionar al paciente derivado.
- **Tarea 3:** Implementar una herramienta de edición que permita al estudiante agregar, eliminar o editar dientes, registrar procedimientos, y tomar notas relevantes en el odontograma.
- **Tarea 6:** Habilitar la opción para guardar y actualizar el odontograma una vez que el estudiante haya realizado las modificaciones necesarias.

HU029: Como estudiante quiero poder imprimir el odontograma

- **Tarea 1:** Implementar un botón que permita descargar o imprimir el odontograma.

HU030: Como estudiante quiero subir al sistema la firma del paciente en el documento de consentimiento informado.

- **Tarea 1:** Implementa una función que permita al estudiante subir la firma del paciente.
- **Tarea 3:** Implementa una funcionalidad de almacenamiento seguro para los documentos de autorización cargados.

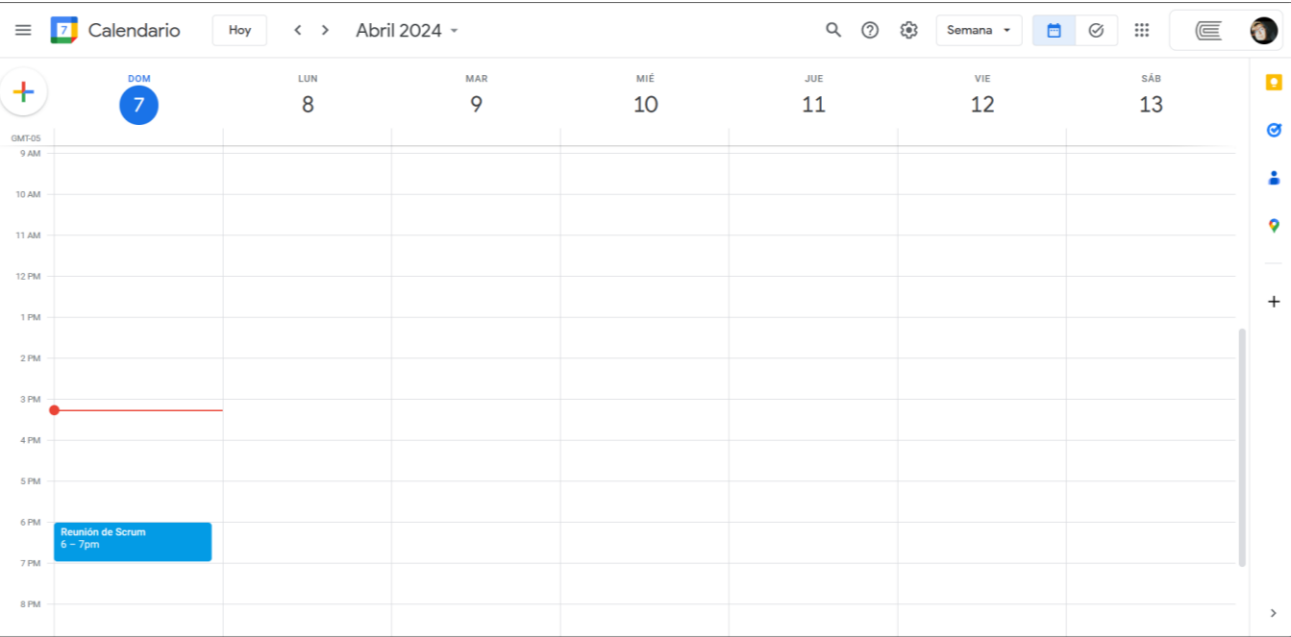
HU031: Como docente, quiero poder subir un registro de celdas con los estudiantes y que el sistema cree sus cuentas respectivamente.

- **Tarea 1:** Definir el formato del archivo de registro que el docente utilizará para cargar la información de los estudiantes.
- **Tarea 2:** Desarrolla una interfaz de usuario o apartado donde el docente pueda cargar el archivo de registro.
- **Tarea 3:** Implementa una función de validación que verifique si el archivo cumple con el formato especificado.
- **Tarea 4:** Utiliza la información del archivo de registro para crear cuentas de usuario para cada estudiante en el sistema. Esto incluye la generación automática de nombres de usuario y contraseñas seguras.
- **Tarea 5:** Notificar al estudiante mediante mensajes automáticos al correo institucional una vez creada la cuenta.

1.10 Gestión de Reuniones en Scrum

1.10.1 Preparación para las Reuniones de Scrum:

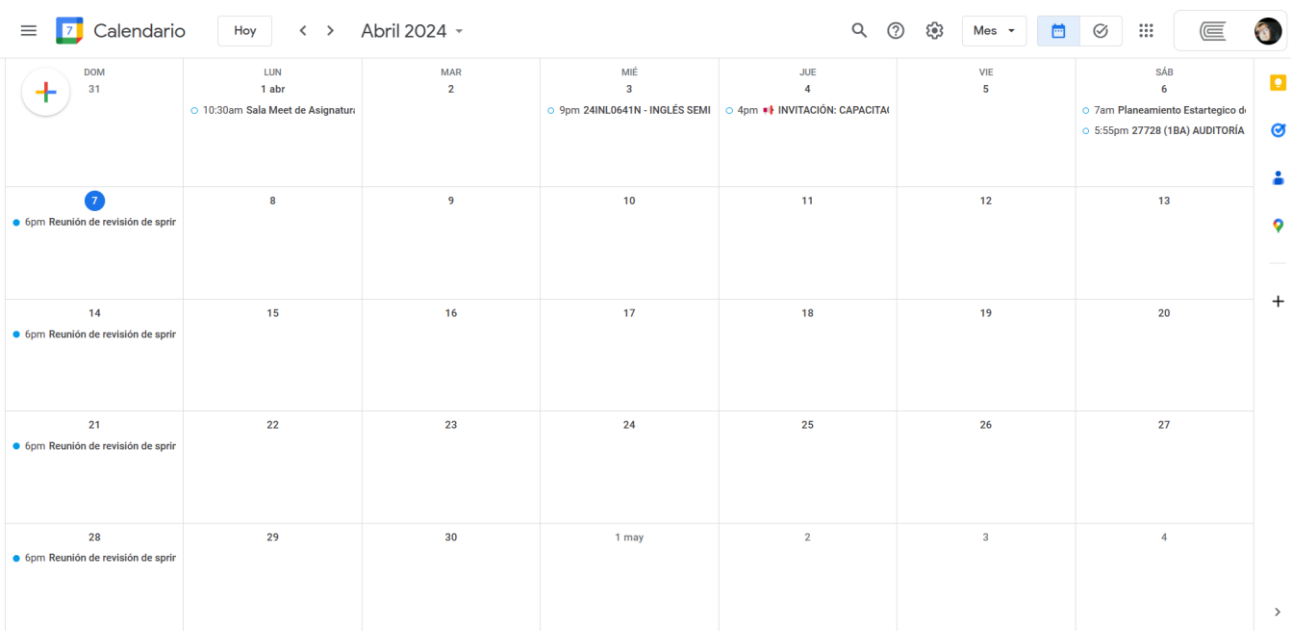
Para seleccionar las historias de usuario y definir las tareas.



1.10.2 Preparación de las reuniones diarias de Scrum: Para el seguimiento del progreso y la identificación de obstáculos.

1.10.3 Preparación de la reunión de planificación del sprint: Para demostrar las funcionalidades completadas.

1.10.4 Preparación de la reunión de revisión del sprint



1.10.5 Preparación de la reunión de la Retrospectiva del Sprint:

Para revisar el sprint y planificar mejoras.

Este Sprint Backlog detallado incluye tareas específicas para cada historia de usuario seleccionada. El equipo se enfocará en completar estas tareas durante el sprint para asegurarse de que las funcionalidades estén listas para su revisión y prueba al final del sprint.