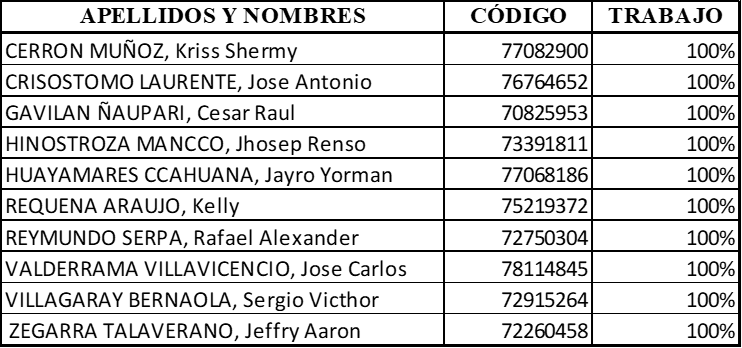
**UNIVERSIDAD CONTINENTAL**

**FACULTAD DE INGENIERÍA**

**ESCUELA ACADÉMICO PROFESIONAL DE INGENIERÍA DE SISTEMAS E INFORMÁTICA**

**PROYECTO**

**“ ”**

****

**PRESENTADO POR:ASESOR: GAMARRA MORENO Daniel**

**HUANCAYO – PERÚ**

**2024**

**Índice**

[**1. Capítulo I: Sprint 0 3**](#_ioy35k4ri6df)

[**1.1. Configuración del Entorno 3**](#_w51g72btcg1h)

[**1.2. Producto inicial 3**](#_l0zpkvq0z33z)

[**1.3. Creación de Equipos 5**](#_xujcwwv44uzk)

[**1.4. Capacitación 5**](#_xesazddy2l19)

[**1.5. Definición de las reglas del Equipo 5**](#_mi2bhq7vzr6a)

[1.5.1. Reglas de Documentación 6](#_e8adzrwxrzbu)

[1.5.2. Reglas de Backend 6](#_m3w1a9lprrwa)

[1.5.3. Reglas de Frontend: Diseño 7](#_dpjkd8lzqjcw)

[1.5.4. Reglas de Frontend: Programación 7](#_m690pqkh3iq6)

[**1.6. Planificación de Infraestructura 7**](#_6741bhg1ehpk)

[**1.7. Gestión de Proyectos con enfoque Kanban 8**](#_aqzqtkv96m2q)

[**1.8. Gestión Efectiva del Product Backlog 8**](#_qa8do28niful)

[1.8.1. Historias de Usuario 8](#_bdzbjmi3guce)

[**1.9. Planificación y Ejecución del Primer Sprint 11**](#_vdint7yeipo4)

[**1.10. Preparación de las reuniones de Scrum 16**](#_3dg2jnlvkppl)

[**1.11. Preparación de las reuniones diarias de Scrum 17**](#_y4kyvne915wa)

[**1.12. Preparación de la reunión de planificación del sprint 17**](#_ripicjh54vrm)

[**1.13. Preparación de la reunión de revisión del sprint 17**](#_3b5gwsfovhug)

[**1.14. Preparación de la reunión de la Retrospectiva del Sprint 17**](#_ab3qsaudjd9o)

# Capítulo I: Sprint 0

# Configuración del Entorno

**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://sourceforge.net/projects/xampp/files/XAMPP%20Windows/8.2.4/xampp-windows-x64-8.2.4-0-VS16-installer.exe/download>)

**Paso 2: Crear una carpeta de proyecto**

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

**Paso 3: 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 back:

Repositorio front:

<https://github.com/proyecto-clinica-odontologica/muelitas-front.git>

# Producto inicial

Desarrollo de un sistema odontológico para los estudiantes de la escuela académica de odontología, con un servidor web desplegado en un servidor en la nube.

El sistema odontológico reserva citas a pacientes de estudiantes de odontología para su debida atención.

1.2.1. Visión inicial del producto

**Visión del proyecto**

Crear un sistema que asegure la protección de datos, mejore la eficiencia en operaciones y fomente el aprendizaje avanzado en odontología. Este sistema se enfoca en brindar atención de calidad en citas reservadas, elevar la calidad de vida de los pacientes y hacer los procesos más ágiles para los usuarios del sistema.

**Visión de la E.A 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

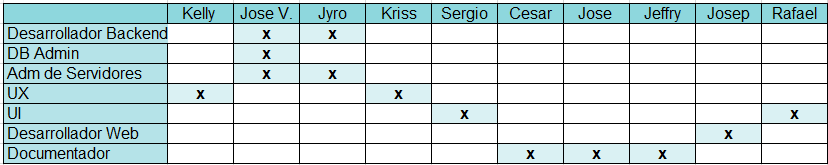
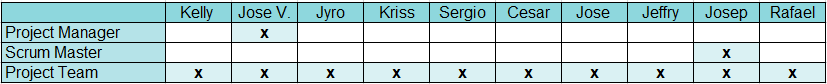
1.2.2. Identificar las funcionalidades básicas

1. Registro de pacientes: Permitir a los pacientes crear perfiles personales con información básica como nombre, edad, historial médico y datos de contacto.
2. Programación de citas: Permitir a los pacientes seleccionar la fecha y hora deseada para su cita odontológica, según la disponibilidad del calendario de la clínica.
3. Recordatorios de citas: Enviar recordatorios automáticos a los pacientes a través de correo electrónico, mensajes de texto o notificaciones móviles para recordarles sus próximas citas.
4. Confirmación de citas: Solicitar a los pacientes que confirmen sus citas programadas, ya sea mediante un enlace en el correo electrónico o un mensaje de texto de confirmación.
5. Cancelación y reprogramación de citas: Permitir a los pacientes cancelar o reprogramar sus citas en línea, siempre que lo hagan dentro de un período de tiempo especificado y sujeto a las políticas de cancelación de la clínica.
6. Historial de citas: Mantener un registro detallado del historial de citas de cada paciente, incluidas las fechas de las citas anteriores, los tratamientos realizados y los profesionales que los llevaron a cabo.

1.2.3. Borrador Product Backlog

# Creación de Equipos





# Capacitación

La capacitación es un componente crucial de la gestión de proyectos, y la familiaridad de la mayoría de los estudiantes con Scrum es una ventaja significativa en este contexto. Scrum es una metodología ágil popular en la industria de la gestión de proyectos que facilita su aplicación en proyectos medianos. Para garantizar una mejor planificación, ejecución y control del proyecto, se puede utilizar este recurso al contar con un equipo que ya conoce Scrum.

# Definición de las reglas del Equipo

Estas reglas del equipo crean un ambiente de trabajo productivo y colaborativo en el equipo de desarrollo de software. Para lograr un funcionamiento armonioso y eficiente, es fundamental que todos los miembros del equipo estén al tanto de estas reglas y se adhieran a ellas.

**Reglas Generales**

* Los encargados son los responsables de entregar el trabajo solicitado a su grupo en el tiempo solicitado y de manera eficiente, caso contrario se les reducirá el porcentaje de participación.
* El encargado del rol asignará los avances de los integrantes por porcentaje.
* Los encargados proporcionarán actividades a su grupo.
* Todos los miembros del equipo deben promover la comunicación abierta y transparente.
* Se acordarán por votación mutua las reuniones viendo la disponibilidad de horarios de los integrantes del grupo.
* Esto incluye compartir de manera regular información relevante, problemas, avances y obstáculos, sin ocultar nada que pueda afectar al proyecto
* Mantenga reuniones regulares, como reuniones de equipo, reuniones de seguimiento o reuniones de retroalimentación, para asegurarse de que todos estén alineados con los objetivos del proyecto y para abordar cualquier preocupación o problema de manera oportuna.
* Entrar a los módulos según su posición como administrador, profesor o estudiante.
* Discuta cualquier funcionalidad para comprenderla.
* Cuando surgen desacuerdos o conflictos, el equipo debe abordarlos de manera profesional y constructiva.
* Todos los miembros del equipo deben cumplir con las normas y prácticas de desarrollo de software establecidas por la organización.
* Deben documentar adecuadamente su trabajo, incluidos el código, los diseños y los procesos.
* Se deben realizar pruebas exhaustivas antes de entregar cualquier trabajo o funcionalidad para asegurarse de que el software funcione correctamente y cumpla con los requisitos.

## **Reglas de Documentación**

* El equipo debe establecer pautas claras para la documentación, que incluyan formatos, estructuras y herramientas. Esto ayuda a asegurarse de que la documentación sea coherente a lo largo del proyecto.
* La documentación detallada de las definiciones de usuario o historias de usuario debe prepararse antes de comenzar el desarrollo de nuevas funcionalidades. Esto incluye los requisitos funcionales y los criterios de aceptación.
* Todas las reuniones de Scrum, incluidas las reuniones de stand-up diarias, las reuniones de planificación de sprint y las retrospectivas, deben registrarse. Esto facilita el seguimiento del progreso y la toma de decisiones durante el sprint.
* La documentación detallada de las pruebas realizadas, tanto de unidad como de integración, debe incluirse en cada tarea o funcionalidad desarrollada. Esto garantiza que el software haya sido probado de manera adecuada antes de su implementación.
* La documentación debe mantenerse actualizada a lo largo del proyecto. La documentación correspondiente debe incluir de manera oportuna cualquier cambio en los requisitos, la arquitectura o el diseño.
* La documentación debe revisarse con frecuencia para garantizar su calidad y precisión. Estas revisiones pueden llevarse a cabo de manera independiente o formar parte de las tareas de revisión de sprint.
* Se debe establecer una política clara de gestión de permisos para garantizar que solo aquellos que la necesiten tengan acceso a la documentación relevante.

## **Reglas de Backend**

* Mostrar claramente dónde se llevarán a cabo las modificaciones.
* Tome las medidas de seguridad adecuadas para garantizar que los usuarios puedan acceder a las funcionalidades de manera segura.
* Para facilitar la integración con el frontend, crea documentación clara que incluya ejemplos de solicitudes y respuestas.
* Realizar pruebas unitarias para garantizar la calidad del código. Siempre que sea posible, automatizar las pruebas.
* Manejar adecuadamente los errores y brindar respuestas útiles a los errores
* Cooperar con el equipo de base de datos para garantizar que el diseño de la base de datos cumpla con las necesidades del backend.

## **Reglas de Frontend: Diseño**

* Los colores se ajustarán a la empresa que trabaja con nosotros.
* Al diseñar, dar prioridad a lo estético (no mucho texto).
* Al hacer listas, solo coloque alrededor de cinco como ejemplo porque solo se necesita uno para que el bucle corra.
* Cuando ya estemos en los procesos, intentemos poner barras de estado.
* Se requiere la capacidad de responder, a menos que se realice un examen dental.
* El dashboard, si o si, incluirá un navegador, una barra lateral y su contenido, por supuesto.
* Diseñar un odontograma solo para computadoras de escritorio.

## **Reglas de Frontend: Programación**

* Para evitar desviarte, las tareas serán asignadas de acuerdo con tu rol (administrador, estudiante o docente).
* Construiremos los modelos y servicios de acuerdo con el backend.
* Para probar su progreso, cree un toast "mensaje de estado" o use console.log "mensaje".
* Esperaremos a que los de diseño combinen lo programado con lo visual.
* Para los procesos de la empresa, utilizaremos el diseño de bpmn.
* Las normas de la base de datos
* Programar revisiones del diseño de la base de datos.
* La versión del software para el desarrollo de la base de datos debe usarse igual.
* Crear usuarios para todos (de administrador, médico, recepcionista, etc.).

# 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

● *Herramientas:*

○ IDE: Visual Studio Code

○ Modelado: Bizagi

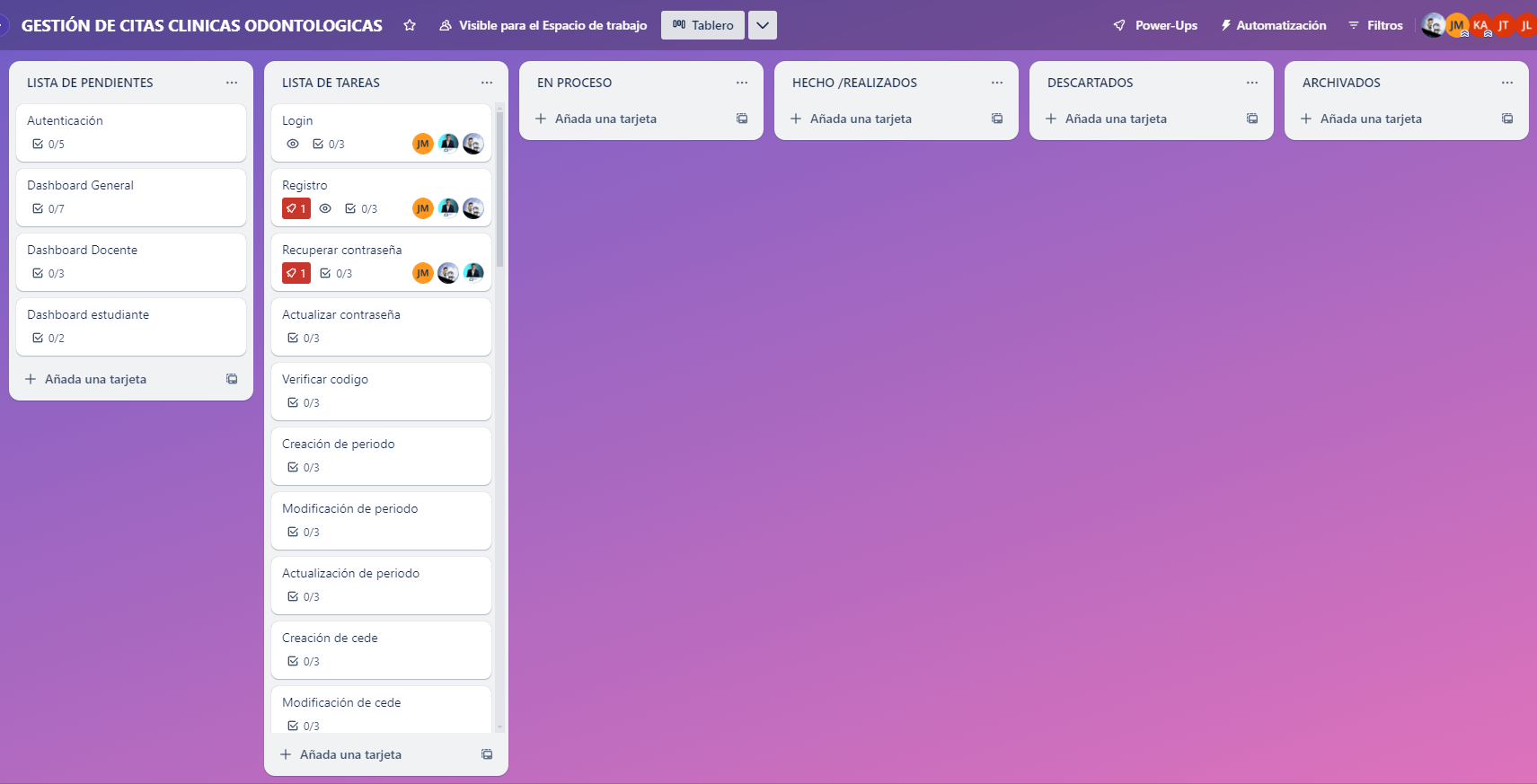
○ Diseño: Figma.

○ Presentación: Canva

○ Organización: Trello

○ Contenedor de Software: Docker

# Gestión de Proyectos con enfoque Kanban



# 

# Planificación y Ejecución del Primer Sprint

HU001: Como usuario, 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.

HU002: Como usuario deseo tener la opción de entrar a mi cuenta con una cuenta google

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

HU003: Como estudiante odontólogo, quiero visualizar mi perfil para gestionar mis datos personales, información de contacto, etc.

HU004: Como estudiante odontólogo, quiero visualizar un dashboard principal con información general para tener una visión rápida del estado actual

HU005: Como estudiante odontólogo, quiero poder registrar nuevos pacientes en el sistema para mantener un registro actualizado de la base de datos de pacientes.

HU006: Como estudiante odontólogo, quiero poder editar//eliminar pacientes en el sistema para mantener un registro actualizado de la base de datos de pacientes.

HU007: Como estudiante odontólogo, deseo poder registrar una cita por nombre o número de documento para acceder rápidamente a su información.

HU008: Como estudiante odontólogo, deseo poder programar citas para mis pacientes, asignando fechas y horas específicas para sus consultas.

HU009: Como estudiante odontólogo, quiero poder cancelar o reprogramar citas existentes en caso de cambios en la agenda.

HU010: Como docente quiero visualizar un dashboard principal con información general para tener una visión rápida del estado actual del consultorio entre otros.

HU011: Como docente, quiero visualizar mi perfil para gestionar mis datos personales, información de contacto, especialidades, etc

HU012: Como docente, quiero la creación, modificación y listado de sedes para gestionar adecuadamente la ubicación de las actividades académicas, como clases teóricas, prácticas y seminarios, facilitando la asignación de recursos y la organización logística.

HU013: Como docente, quiero la creación, modificación y listado de periodos académicos para planificar y estructurar el calendario escolar, incluyendo fechas de inicio y fin de clases, períodos de exámenes y vacaciones, asegurando una gestión eficiente del tiempo académico.

HU014: Como docente, quiero la creación, modificación y listado de cursos de evaluación para establecer los momentos adecuados para la evaluación del desempeño académico de los estudiantes, asegurando una evaluación equitativa y transparente.

HU015: Como docente, quiero revisar y actualizar la información de un tratamiento para mantener actualizados los registros clínicos de los pacientes y asegurar la calidad en la prestación de servicios odontológicos.

HU016: Como docente, quiero generar reportes de los tratamientos realizados para analizar el desempeño de los estudiantes odontólogos, evaluar la eficacia de los procedimientos y ofrecer retroalimentación para mejorar la práctica clínica y el aprendizaje.

HU017: Como administrador, quiero visualizar un dashboard principal con información general para tener una visión rápida del estado actual del consultorio entre otros.

HU018: Como administrador, quiero tener control total de registros de usuarios para realizar cambios y actualizaciones del sistema.

HU019: Como administrador, quiero realizar la creación, modificación y listado de docentes para gestionar eficientemente el personal académico, asignar responsabilidades docentes, gestionar horarios y garantizar la calidad.

HU020: Como administrador, quiero poder suspender las cuentas de los estudiantes que se retiren de la asignatura, para tener actualizada la lista de estudiantes activos.

HU021: 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.

HU022: Como administrador, quiero enviar mensajes al perfil del estudiante para comunicar información relevante sobre su programa académico, observaciones sobre sus procesos,proporcionar instrucciones sobre procedimientos administrativos y mantener una comunicación efectiva para mejorar la experiencia.

.

# Preparación de las reuniones de Scrum

Para las reuniones de scrum tomamos la decisión en grupo de tenerlas los días martes por la tarde para poder estar todos y tener ideas frescas luego de clase.

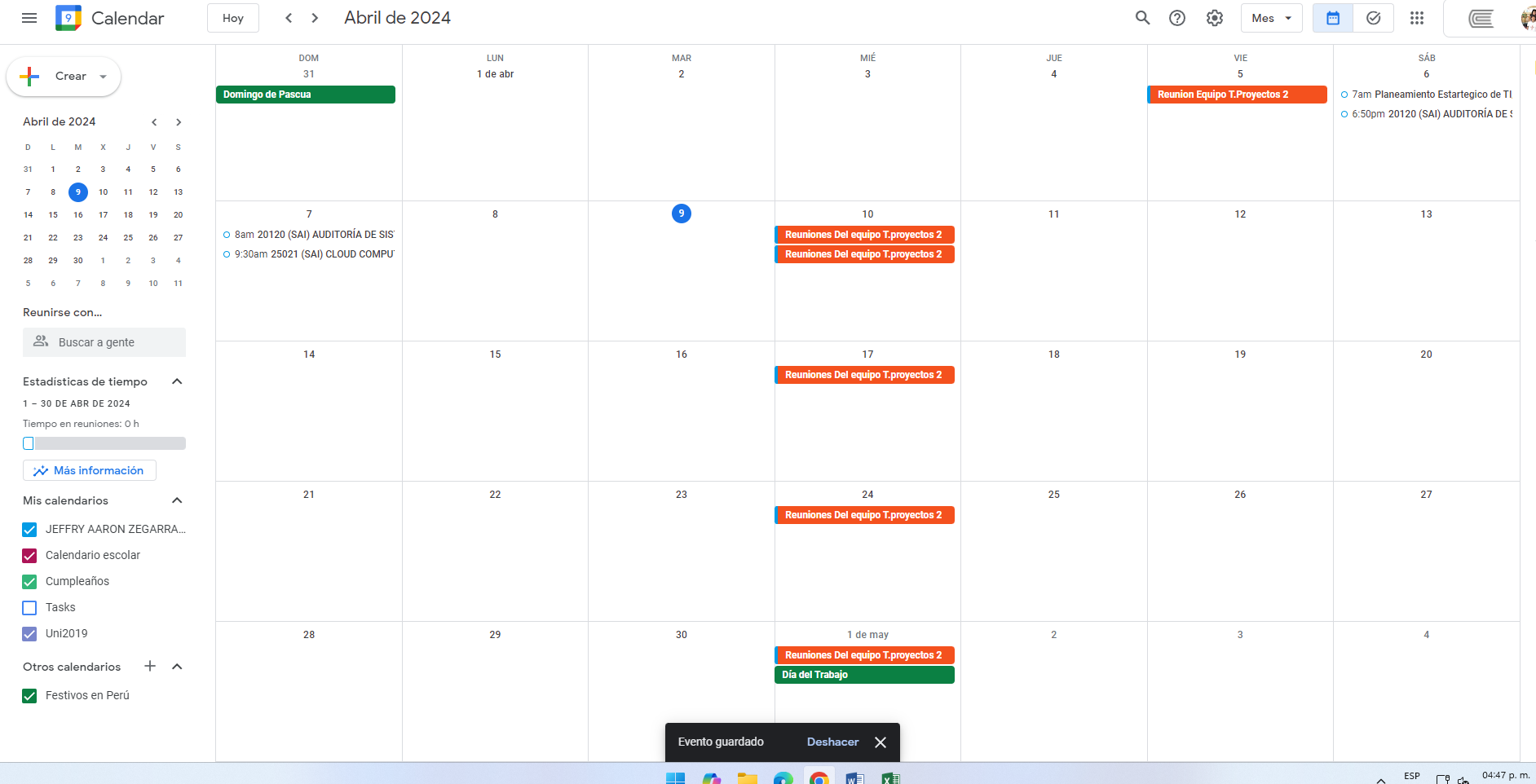
Con el objetivo de avanzar de manera efectiva en nuestro proyecto de software,nos reuniremos semanalmente los días martes para coordinar nuestras tareas y establecer los días de nuestras reuniones virtuales vía Google Meet.Nuestra primera reunión se llevará a cabo todos los viernes a las 9 pm, y se espera la asistencia de todos los miembros del equipo. Durante esta sesión inaugural, discutiremos los detalles del proyecto, asignaremos roles y responsabilidades, y acordaremos los días específicos en los que nos reuniremos vía Meet para revisar nuestro progreso y abordar cualquier pregunta o problema que surja.

# Preparación de las reuniones diarias de Scrum

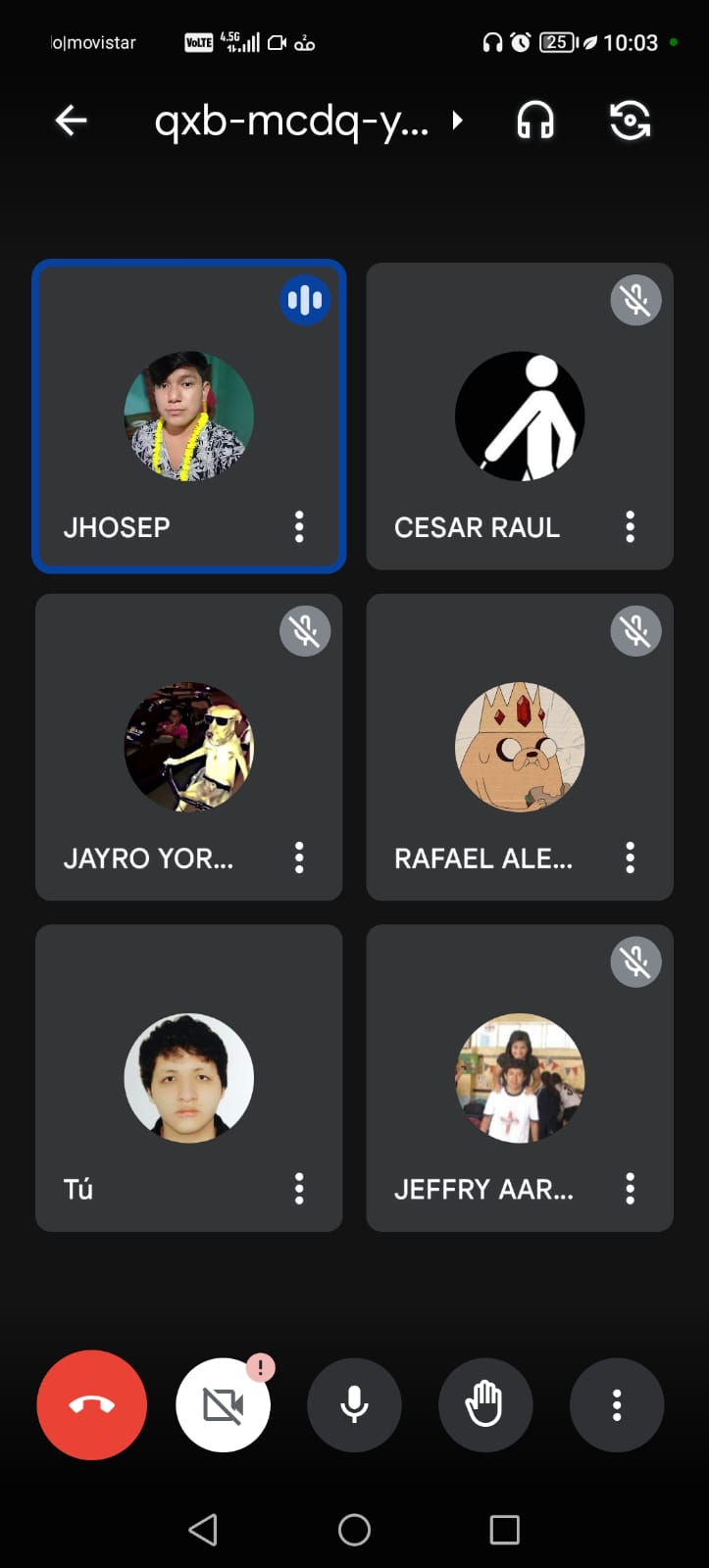
# Preparación de la reunión de planificación del sprint

Se llegó al acuerdo que las reuniones de planificación de sprint serán los días Martes para poder tener mayor concurrencia del grupo y tomar decisiones más asertivas.

# Preparación de la reunión de revisión del sprint



# Preparación de la reunión de la Retrospectiva del Sprint



Primera Reunión 5/04: Se llegó a un acuerdo en referencias a los requerimientos funcionales y no funcionales y a su vez, que tendríamos que modificar algunas historias de usuario y aumentar en base a lo que queríamos modificar, concluimos que tendremos reuniones de retrospectiva los días viernes.

**1.8 Gestión Efectiva del Product Backlog**

<https://drive.google.com/drive/folders/1uX7wbHI7M2NKIKLnW72kg5obaeFkmtgJ?usp=drive_link>

| login |  | estudiante | docente | general (admin) |
| --- | --- | --- | --- | --- |
| mejorar diseño |  | gestiona pacientes | gestionar estudiantes | gestionar periodo, cede, docentes |
|  |  | gestionar citas | gestionar pacientes estudiantes | gestionar integrantes por periodo |
|  |  |  | gestionar lo que realiza el estudiante | gestionar mensajes |
|  |  |  |  | gestionar usuarios |

Velocidad del equipo:

DuracionIteración(2 a 4 semans, recomendación 2) \*Numero Integrantes \* dedicación

2-4 semanas

2 semanas = 2 \*5 = 10 días \*8 horas = 80 horas

80 \* 10 \* 0.50 = 400 horas

Numero de sprints = TotalPuntosHistoriaProyecto / velocidadEquipo

Numero de sprints = 2000 / 400 = 5 iteraciones

trabajar en horas