

PFP MANAGER

Johan Guarin, Julian Ortega, David Medina

**GRUPO U2
TEAM #9**

DOCENTE JUAN MARIÑO

**CAMPUSLANDS
SALON ARTEMIS
RUTA NODEJS
FLORIDABLANCA
2025**

PFP MANAGER

1. SITUACIÓN PROBLEMA

En un mundo que evoluciona a diario en el tema digital, entendemos que en muchas ocasiones un trabajo formal de oficina no es exactamente lo que se está buscando, e incluso existe también la opción para aquellas personas que cuentan con un trabajo remoto, de querer generar más ingresos de manera independiente, por ello hemos tomado la decisión de crear PFP MANAGER, una aplicación creada no sólo para cubrir la necesidad de poder encontrar un trabajo y una estabilidad económica, sino también donde podrás dar a conocer tu trabajo a través de tu portafolio, brindando oportunidades de contacto con clientes que estén interesados en tu trabajo y quieran trabajar contigo, pudiendo compartir desde propuestas, proyectos, entregables, hasta poder gestionar tus contratos y finanzas desde la misma aplicación.

2. LEVANTAMIENTO DE REQUERIMIENTOS

Para el desarrollo del sistema de gestión integral orientado a la administración eficiente de clientes, propuestas, proyectos y demás puntos planteados en nuestro proyecto, se realizó una investigación con gestores reales provenientes de distintos sectores. El objetivo principal fue comprender las necesidades, retos y flujos de trabajo cotidianos de diferentes freelances, que enfrentan en la gestión de proyectos y clientes, con el fin de diseñar funcionalidades que respondan efectivamente a sus requerimientos.

Durante la investigación se analizaron algunos procesos y sistemas existentes que usan para la gestión, con un enfoque en identificar puntos críticos, redundancias y oportunidades de obtener un mayor crecimiento.

La información obtenida permitió definir claramente las funcionalidades clave que el sistema debe ofrecer para mejorar la productividad y la trazabilidad de las operaciones.

3. REQUERIMIENTOS

3.1. Requerimientos Funcionales:

1. ACTUALIZAR ESTADO DE PROPUESTA:

El usuario podrá modificar el estado de una propuesta existente, teniendo como opciones: "Aceptada" o "Rechazada".

2. CREACIÓN DE CLIENTE:

El sistema debe permitir al usuario registrar un nuevo cliente, solicitando los siguientes campos: "Nombre, correo electrónico, teléfono & empresa".



3. LISTAR PROPUESTAS POR ESTADO:

El sistema debe listar las propuestas por estado, teniendo como opciones: "Pendiente, aceptada o rechazada".

4. LISTAR CLIENTES:

El sistema debe permitir listar los clientes registrados en la plataforma, permitiendo listar por nombre del cliente o empresa.

5. ACTUALIZAR INFORMACIÓN DEL CLIENTE:

Se debe poder seleccionar un cliente de una lista, mostrar sus datos actuales y poder editarlos para mantener su información actualizada.

6. ELIMINAR CLIENTE:

El sistema debe permitir la eliminación de un cliente cuando este no tenga proyectos activos, antes de su eliminación, se debe mostrar un aviso de advertencia.

7. CREACIÓN DE CONTRATO:

El sistema debe generar un contrato para un proyecto, formalizando el acuerdo con el cliente, debe crearse con los siguientes campos: Condiciones, fechas (Inicio y fin), valor total y términos de pago.

8. ACTUALIZACIÓN DE CONTRATO:

El sistema debe permitir modificar un contrato ya existente, para ajustar términos según su negociación, sólo debe permitir su edición en estado "Borrador".

9. CREACIÓN DE ENTREGABLE:

El sistema debe permitir la creación de entregable para un proyecto, con los siguientes campos: Nombre, descripción, fecha límite y estado, esto para poder tener un orden a las entregas parciales.

10. ACTUALIZAR ESTADO DE ENTREGABLE:

Se debe permitir la actualización de un entregable, permitiendo evidenciar su progreso, las opciones de sus estados son: pendiente, en progreso, entregado, aprobado, rechazado.

11. CREACIÓN DE PROYECTO:

El sistema debe permitir crear un proyecto manualmente o desde una propuesta aceptada y poder tener una organización mejor al trabajo a realizar, debe tener los siguientes campos: Nombre, descripción, fecha (Inicio y fin), valor y estado.

12. LISTAR PROYECTOS:

Se debe listar todos los proyectos, para poder monitorear su progreso, su listado debe ser por estado: Activo, pausado, finalizado y cancelado, o por cliente.



13. ACTUALIZACIÓN DE PROYECTO:

Se debe permitir la modificación de información de un proyecto y mantenerlo actualizado, se puede modificar su descripción, fechas, valor y estado.

14. REGISTRAR AVANCE DE PROYECTO:

Se debe poder registrar el avance de un proyecto, ingresando un porcentaje de avance (0 a 100%), agregar comentarios del progreso, registrar fecha del avance y mostrar el historial de avances.

15. LISTAR ENTREGABLES:

El sistema debe permitir listar los entregables por proyecto, mostrando su estado y fechas límite, resaltar entregables vencidos y mostrando el progreso general del proyecto.

16. REGISTRAR UN INGRESO:

El sistema debe permitir registrar un ingreso para poder llevar el control de mis ganancias, si se desea se puede asociar a un proyecto.

17. REGISTRAR UN EGRESO:

El sistema debe permitir registrar los egresos, al igual que un ingreso, este puede ser relacionado o no a un proyecto.

18. VER BALANCE FINANCIERO:

El sistema debe permitir ver el balance financiero, mostrando los ingresos, egresos y un balance total, se puede filtrar por fechas, proyecto o cliente.

19. REGISTRO PAGO DE CLIENTE:

Se debe poder registrar pagos recibidos de clientes, para poder actualizar el estado de cobros, se puede asociar a un proyecto en específico.

3.2. Requerimientos No Funcionales

1. TIEMPO DE RESPUESTA:

Todas las operaciones deben obtener una respuesta en un segundo aproximadamente.

2. PERSISTENCIA DE DATOS:

La aplicación debe utilizar una base de datos local para asegurar que los datos permanezcan almacenados entre sesiones.

3. CONSISTENCIA TRANSACCIONAL:

Las operaciones críticas como el registro de pagos o la modificación de entregables deben realizarse mediante transacciones para garantizar la integridad de los datos.



4. PORTABILIDAD:

Descripción: El sistema debe ser ejecutable en Windows, Linux y macOS

4. HISTORIAS DE USUARIO CON CRITERIOS DE ACEPTACIÓN

1.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	HU-021	Actor	Freelance
NOMBRE DEL REQUERIMIENTO	Configurar Conexión Base de Datos		
Descripción			
Como freelancer, quiero configurar la conexión a MongoDB, para que la aplicación funcione correctamente.			
Funcionalidad			
Permitir su validación al conectarse e iniciar sesión, debe mostrar un error si falla la conexión, configurar varios ambientes y la validación de esquema.			
Criterios de aceptación	1. El sistema de debe permitir la configuración para la conexión a MongoDB.		
Restricciones			
El freelance puede configurar la conexión a MongoDB y asegurar de su funcionamiento correctamente.			

2.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	HU-005	Actor	Freelance
NOMBRE DEL REQUERIMIENTO	Crear propuesta		
Descripción			
Como freelancer quiero crear una propuesta para un cliente, para poder formalizar una oferta de trabajo.			
Funcionalidad			
Permitir la creación de una propuesta de trabajo para un cliente, ingresando unos datos solicitados y al finalizar poder generar un número único de propuesta.			
Criterios de aceptación	<div>1. El sistema debe permitir crear una propuesta de trabajo para un cliente.</div> <div>2. El sistema debe iniciar esta propuesta con un estado inicial de “Pendiente”, una vez diligenciados los datos solicitados.</div>		

Restricciones
El freelance puede configurar la conexión a MongoDB y asegurar de su funcionamiento correctamente.

3.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	HU-006	Actor	Freelance
NOMBRE DEL REQUERIMIENTO	Listar propuestas.		
Descripción			
Como freelancer, quiero ver todas mis propuestas, para hacer seguimiento de su estado.			
Funcionalidad			
Permitir listar las propuestas por cliente, respondiendo con los siguientes datos: número, cliente, título, precio, estado y fecha.			
Criterios de aceptación	1. El sistema debe permitir listar las propuestas de trabajo.. 2. El sistema debe mostrar mostrar: número, cliente, título, precio, estado, fecha. 3. El sistema debe listar por cliente y por estado.		
Restricciones			

4.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	HU-007	Actor	Freelance
NOMBRE DEL REQUERIMIENTO	Actualizar estado propuesta.		
Descripción			
Como freelancer, quiero cambiar el estado de una propuesta para reflejar la decisión del cliente.			
Funcionalidad			
Permitir actualizar el estado de una propuesta a: Pendiente, aceptada o rechazada y si es aceptada, generar proyecto automaticamente.			
Criterios de aceptación	<div>1. El sistema debe permitir actualizar el estado de una propuesta.</div> <div>2. El sistema debe registrar la fecha en la que se realizó el cambio.</div> <div>3. El sistema no debe permitir cambios desde “Aceptada” o “Rechazada”.</div>		

Restricciones
El freelance sólo podrá actualizar el estado de una propuesta entre: pendiente, aceptada, rechazada.

5.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	HU-001	Actor	Freelance
NOMBRE DEL REQUERIMIENTO	Crear cliente.		
Descripción			
Como freelancer, quiero registrar un nuevo cliente con sus datos básicos, para poder asociarlo a propuestas y proyectos..			
Funcionalidad			
Permitir registrar un nuevo cliente con los siguientes datos: nombre, email, teléfono, empresa, dirección			
Criterios de aceptación	1. 6.El sistema debe permitir la creación de un nuevo cliente. 2. El sistema debe solicitar: nombre, email, teléfono, empresa, dirección. 3. El sistema debe validar formato de email y teléfono 4. El sistema no debe permitir emails duplicados. 5. El sistema debe asignar ID único automáticamente.		
Restricciones			
El freelance no podrá registrar un mismo correo dos veces.			

6.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	HU-002	Actor	Freelance
NOMBRE DEL REQUERIMIENTO	Listar clientes.		
Descripción			
Como freelancer, quiero ver todos mis clientes registrados, para consultar su información y seleccionar uno específico.			
Funcionalidad			



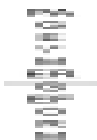
Permitir listar los clientes, visualizando 10 por página,	
Criterios de aceptación	<ol style="list-style-type: none"> 1. El sistema debe mostrar lista paginada de clientes (10 por página). 2. El sistema debe incluir: ID, nombre, empresa, email, teléfono. 3. El sistema debe permitir búsqueda por nombre o empresa 4. El sistema debe mostrar mensaje si no hay clientes registrados 5. El sistema debe dar la opción para ver detalles completos de un cliente
Restricciones	
El sistema sólo podrá mostrar de a 10 clientes por página.	

7.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	HU-003	Actor	Freelance
NOMBRE DEL REQUERIMIENTO	Actualizar cliente.		
Descripción			
Como freelancer, quiero modificar la información de un cliente existente, para mantener sus datos actualizados.			
Funcionalidad			
Permitir registrar un nuevo cliente con los siguientes datos: nombre, email, teléfono, empresa, dirección			
Criterios de aceptación	<div>1. El sistema debe permitir seleccionar el cliente de una lista.</div> <div>2. El sistema debe mostrar datos actuales antes de editar.</div> <div>3. El sistema debe validar nuevos datos con las mismas reglas de creación.</div> <div>4. El sistema debe confirmar cambios antes de guardar.</div> <div>5. El sistema debe mostrar mensaje de éxito o error.</div>		
Restricciones			

8.

HISTORIA DE USUARIO



Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	HU-004	Actor	Freelance
NOMBRE DEL REQUERIMIENTO	Eliminar cliente.		
Descripción			
Como freelancer, quiero eliminar un cliente del sistema, para mantener limpia mi base de datos.			
Funcionalidad			
Permitir eliminar un cliente para mantener un orden y limpieza en la base de datos.			
Criterios de aceptación	1. El sistema sólo debe permitir eliminar clientes sin proyectos activos. 2. El sistema debe mostrar advertencia antes de eliminar. 3. El sistema debe requerir confirmación explícita. 4. El sistema debe mostrar mensaje de confirmación 5. El sistema debe manejar error si tiene dependencias.		
Restricciones			
Sólo se permitirá eliminar un cliente si este no tiene proyectos activos.			

9.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	HU-013	Actor	Freelance
NOMBRE DEL REQUERIMIENTO	Actualizar contrato.		
Descripción			
Como freelancer, quiero modificar un contrato existente, para ajustar términos según negociación.			
Funcionalidad			
Permitir actualizar la información del contrato en estado de “Borrador” para mejorar las condiciones con el cliente, una vez esté todo correcto, se debe cambiar a estado “Firmado” o “Cancelado”.			
Criterios de aceptación	<ol style="list-style-type: none">1. El sistema sólo debe permitir edición en estado "Borrador".2. El sistema debe cambiar estado a "Firmado" o "Cancelado".3. El sistema debe registrar fecha de firma.4. El sistema debe validar coherencia con datos del proyecto.		
Restricciones			
Sólo se permite editar el contrato en estado de “Borrador”.			

10.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	HU-014	Actor	Freelance
NOMBRE DEL REQUERIMIENTO	Crear entregable.		
Descripción			
Como freelancer, quiero definir entregables para un proyecto, para organizar las entregas parciales.			
Funcionalidad			
Poder crear entregables a un proyecto, asociándolo al mismo, para poder tener una mejor organización de entrega con los siguientes campos: nombre, descripción, fecha límite, estado.			
Criterios de aceptación	1. Asociar a proyecto existente 2. Campos: nombre, descripción, fecha límite, estado 3. Estado inicial: "Pendiente" 4. Validar fecha límite dentro del rango del proyecto 5. Permitir múltiples entregables por proyecto		
Restricciones			

11.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	HU-015	Actor	Freelance
NOMBRE DEL REQUERIMIENTO	Actualizar estado de entregable.		
Descripción			
Como freelancer, quiero cambiar el estado de un entregable, para reflejar su progreso.			
Funcionalidad			
Permitir la actualización de un entregable, usando los estados: pendiente, en progreso, entregado, aprobado, rechazado, se usan transacciones para cambios críticos, y registrar la fecha de cada cambio de estado.			
Criterios de aceptación	1. Estados: pendiente, en progreso, entregado, aprobado, rechazado		



	2. Usar transacciones para cambios críticos 3. Registrar fecha de cada cambio de estado 4. Permitir rollback si es necesario 5. Actualizar progreso del proyecto automáticamente
Restricciones	

12.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	HU-008	Actor	Freelance
NOMBRE DEL REQUERIMIENTO	Crear proyecto.		
Descripción			
Como freelancer, quiero crear un proyecto manualmente o desde propuesta aceptada, para organizar el trabajo a realizar.			
Funcionalidad			
Permitir la creación de un proyecto manualmente o desde una propuesta aceptada, permitiendo tener una mejor organización de trabajo, asociándolo a un cliente ya existente, con los siguientes campos:nombre, descripción, fecha inicio/fin, valor, estado, para al final poder generar un código de proyecto único.			
Criterios de aceptación	1. Asociar a cliente existente 2. Campos: nombre, descripción, fecha inicio/fin, valor, estado 3. Estado inicial: "Activo" 4. Si viene de propuesta, heredar datos relevantes 5. Generar código de proyecto único 6. Estado por colores		
Restricciones			

13.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	HU-009	Actor	Freelance

NOMBRE DEL REQUERIMIENTO	Listar proyectos.		
Descripción			
Como freelancer, quiero ver todos mis proyectos, para monitorear su progreso.			
Funcionalidad			
Permitir listar los proyectos, mostrando los siguientes datos: código, nombre, cliente, estado, fechas, progreso. Se podrá filtrar por estado y por cliente, ordenándolos por fecha de inicio.			
Criterios de aceptación	<div><div>1. Mostrar: código, nombre, cliente, estado, fechas, progreso</div><div>2. Filtrar por estado (activo, pausado, finalizado, cancelado)</div><div>3. Filtrar por cliente</div><div>4. Mostrar indicador visual de progreso</div><div>5. Ordenar por fecha de inicio</div></div>		
Restricciones			
Sólo se ordenarán por fecha de inicio.			

14.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	HU-010	Actor	Freelance
NOMBRE DEL REQUERIMIENTO	Actualizar proyecto.		
Descripción			
Como freelancer, quiero modificar la información del proyecto, para mantenerlo actualizado.			
Funcionalidad			
Permitir actualizar un proyecto, cambiando los siguientes campos: descripción, fechas, valor, estado, validando que las fechas sean coherentes, registrando también el historial de cambios de estado.			
Criterios de aceptación	<div>1. Permitir cambiar: descripción, fechas, valor, estado</div> <div>2. Validar fechas coherentes</div> <div>3. Registrar historial de cambios de estado</div> <div>4. Confirmar cambios críticos (cancelación, finalización)</div>		
Restricciones			

15.

HISTORIA DE USUARIO



Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	HU-011	Actor	Freelance
NOMBRE DEL REQUERIMIENTO	Registrar avance proyecto.		
Descripción			
Como freelancer, quiero registrar el progreso del proyecto, para llevar control del avance.			
Funcionalidad			
Permitir registrar avance de un proyecto, ingresando un porcentaje entre 0 a 100%, agregando tambien comentarios, la fecha en la que se registró el avance, permitiendo calcular un progreso automático, basado en los entregables.			
Criterios de aceptación	1. Ingresar porcentaje de avance (0-100%) 2. Agregar comentarios del progreso 3. Registrar fecha del avance 4. Calcular progreso automático basado en entregables 5. Mostrar historial de avances		
Restricciones			

16.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	HU-016	Actor	Freelance
NOMBRE DEL REQUERIMIENTO	Listar entregables.		
Descripción			
Como freelancer, quiero ver entregables por proyecto, para hacer seguimiento de las entregas.			
Funcionalidad			
Permitir listar los entregables de un proyecto, filtrando por proyecto, luego mostrando el estado y las fechas límite, ordenando por fecha límite y mostrando el progreso general del proyecto.			
Criterios de aceptación	<div>1. Filtrar por proyecto</div> <div>2. Mostrar estado y fechas límite</div> <div>3. Resaltar entregables vencidos</div> <div>4. Ordenar por fecha límite</div> <div>5. Mostrar progreso general del proyecto</div>		
Restricciones			

17.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	HU-017	Actor	Freelance
NOMBRE DEL REQUERIMIENTO	Registrar ingreso.		
Descripción			
Como freelancer, quiero registrar un ingreso, para llevar control de mis ganancias.			
Funcionalidad			
Permitir registrar un ingreso, teniendo la opción de asociarlo con un proyecto, solicitando los siguientes campos: monto, descripción, fecha, categoría, usando también transacciones para consistencia y generando un comprobante único por ingreso.			
Criterios de aceptación	1. Asociar a proyecto (opcional) 2. Campos: monto, descripción, fecha, categoría 3. Validar monto > 0 4. Usar transacciones para consistencia 5. Generar comprobante único		
Restricciones			

18.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	HU-018	Actor	Freelance
NOMBRE DEL REQUERIMIENTO	Registrar egreso.		
Descripción			
Como freelancer, quiero registrar un gasto, para controlar mis gastos operativos.			
Funcionalidad			
Permitir registrar un egreso, también teniendo la opción de asociarlo a un proyecto, solicitando los siguientes campos: monto, descripción, fecha, categoría, teniendo un nuevo campo de categorías, donde se pueden seleccionar las siguientes opciones: herramientas, marketing, oficina, otros.			
Criterios de aceptación	1. Asociar a proyecto (opcional) 2. Campos: monto, descripción, fecha, categoría 3. Validar monto > 0		

	4. Categorías: herramientas, marketing, oficina, otros 5. Usar transacciones para consistencia
Restricciones	

19.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	HU-019	Actor	Freelance
NOMBRE DEL REQUERIMIENTO	Ver balance financiero.		
Descripción			
Como freelancer, quiero consultar mi balance financiero, para conocer mi situación económica.			
Funcionalidad			
Permitir ver el balance financiero, mostrando los ingresos, egresos y un balance total, esto se podría filtrar por rango de fechas, por cliente o proyecto.			
Criterios de aceptación	1. Mostrar ingresos, egresos y balance total 2. Filtrar por rango de fechas 3. Filtrar por cliente o proyecto		
Restricciones			

20.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	HU-020	Actor	Freelance
NOMBRE DEL REQUERIMIENTO	Registrar pago de cliente.		
Descripción			
Como freelancer, quiero registrar pagos recibidos de clientes, para actualizar el estado de cobros.			
Funcionalidad			
Permitir registrar un pago del cliente, asociándolo a un proyecto en específico, marcándolo como un			

ingreso automáticamente y actualizando el estado de pago del proyecto.	
Criterios de aceptación	<ol style="list-style-type: none"> 1. Asociar a proyecto específico 2. Marcar como ingreso automáticamente 3. Actualizar estado de pago del proyecto 4. Usar transacciones para evitar duplicados
Restricciones	

5. METODOLOGÍA

Iniciamos este proyecto asignando los roles de los integrantes del grupo, que son:

- Jose Julian Ortega: Developer.
- David Alberto Medina Herrera: Product Owner.
- Johan Andrey Guarin: Scrum Master.

Seguido a ello, en un Git Project, se crearon las diferentes historias de usuario para poder tener un orden organizacional de cómo llevar a cabo la producción del proyecto, donde eventualmente se iban asignando las diferentes historias de usuario a cada integrante del equipo.

Se inició el desarrollo de cada historia de usuario en un orden donde cada integrante iba seleccionando qué historia desarrollar, dependiendo su tiempo o necesidad, al mismo tiempo se iba diligenciando el documento evidenciando el proceso desde su preproducción hasta la posproducción del proyecto.

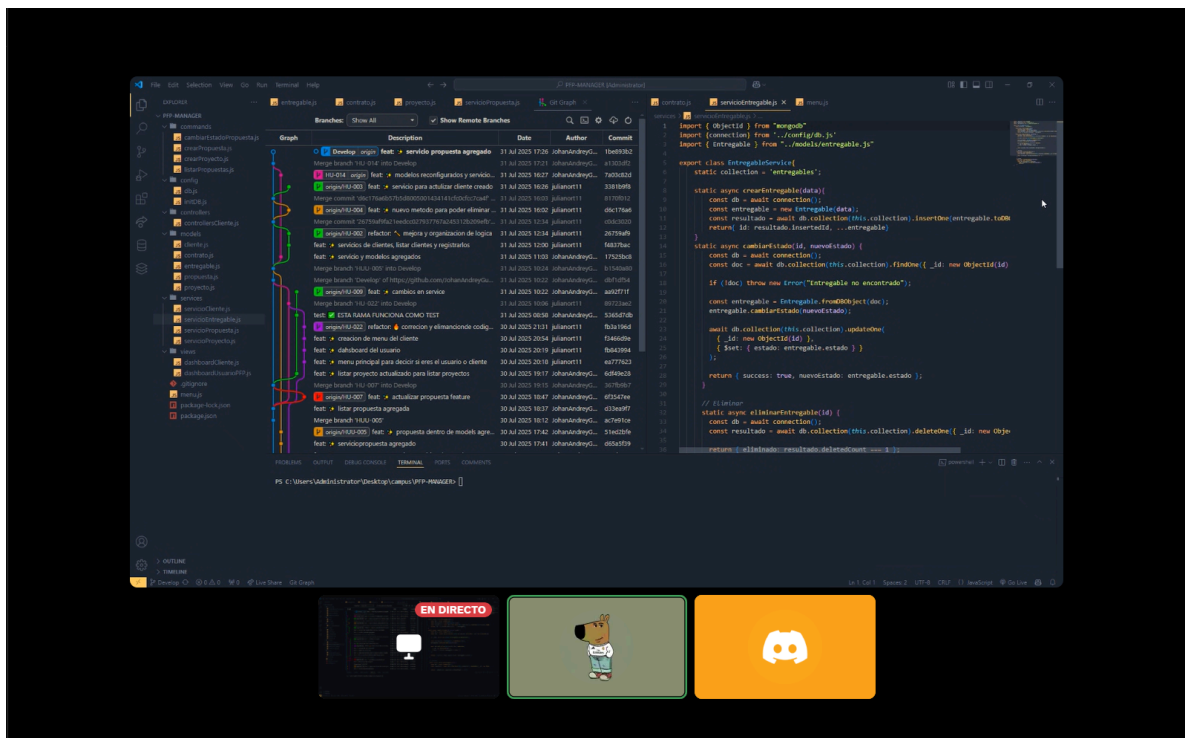
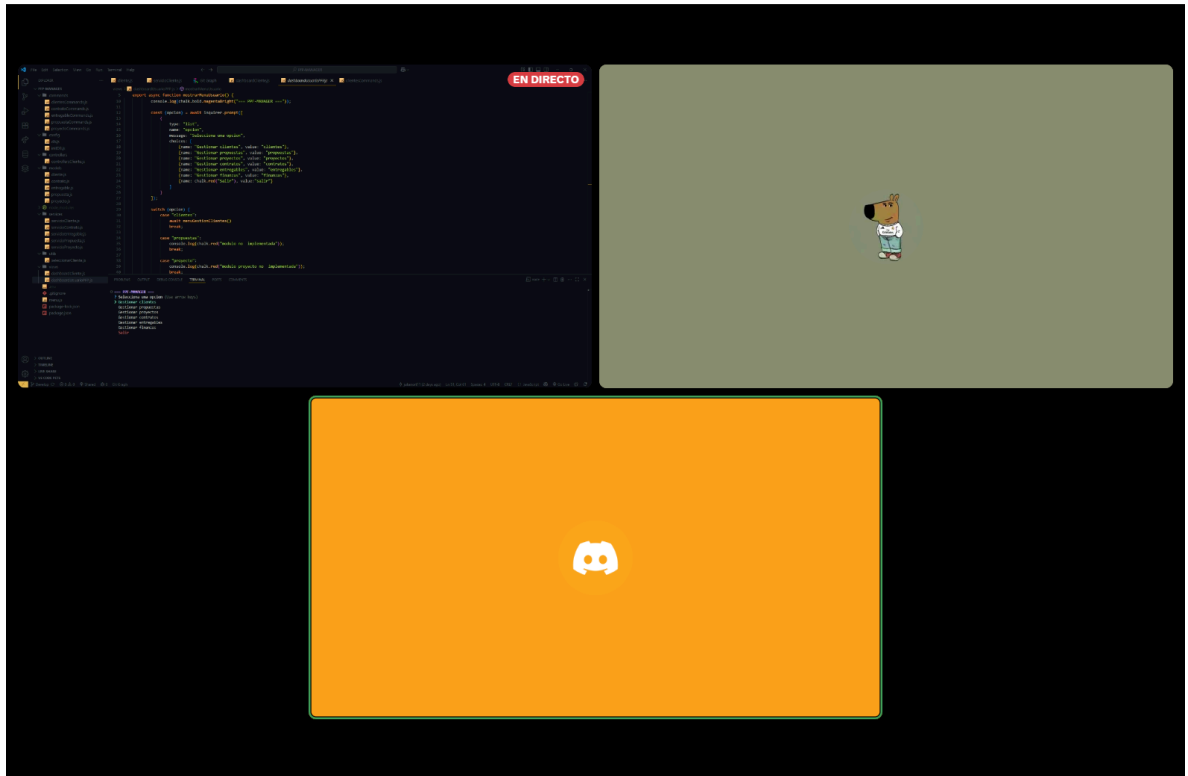
6. EVIDENCIA DE PLANTEAMIENTO DE PLATAFORMA DE TRABAJO

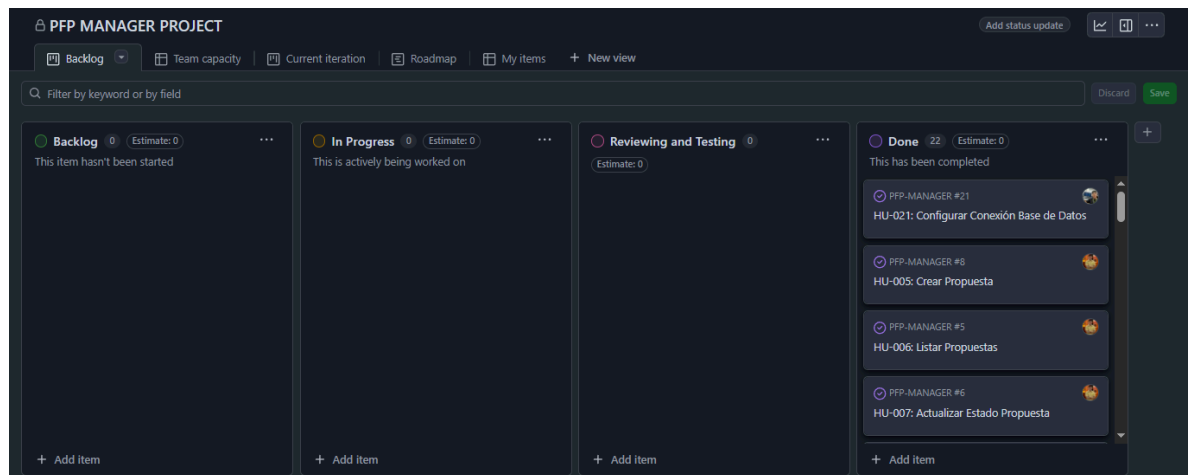
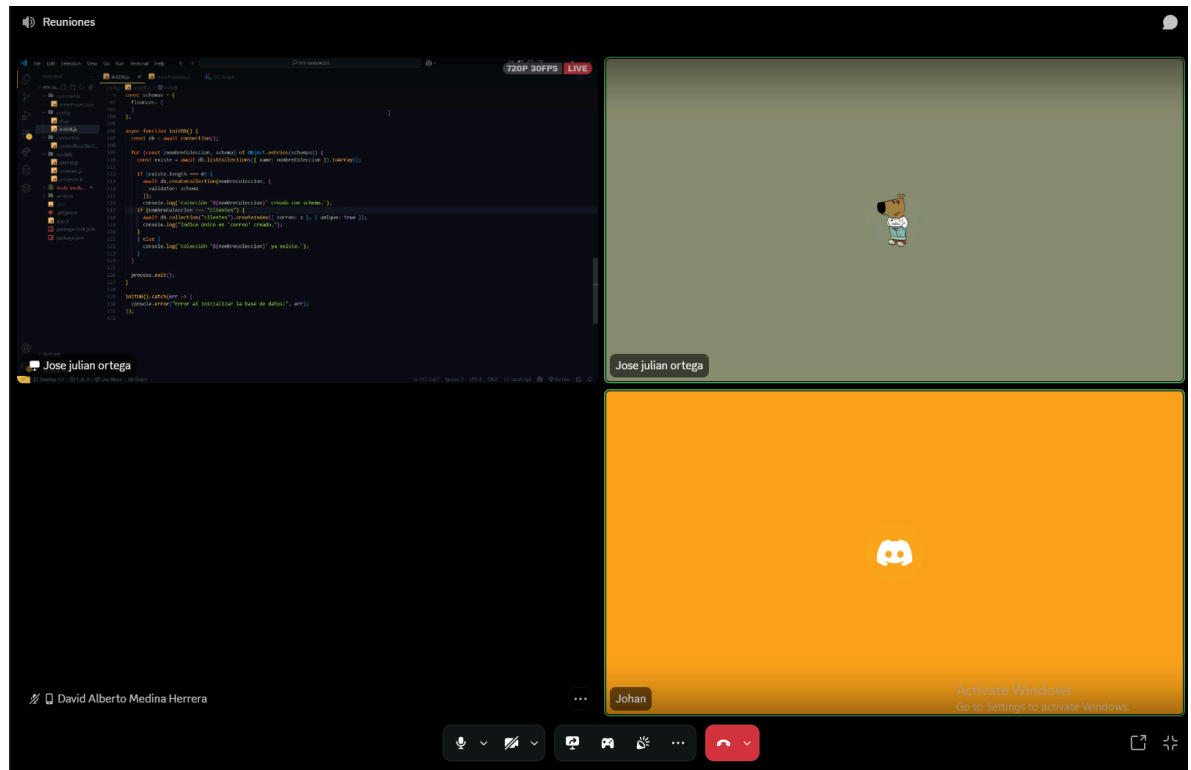
- **LINK DEL REPOSITORIO:**

<https://github.com/JohanAndreyGuarinReatiga/PFP-MANAGER>

- **EVIDENCIA (CAPTURAS):**







7. CONCLUSIONES

- La finalización de este proyecto marca el exitoso desarrollo de la aplicación PFP MANAGER, una aplicación robusta, eficiente y orientada a la gestión integral del portafolio profesional de un freelancer. La herramienta cumple con todos los requerimientos planteados, integrando principios sólidos de arquitectura y buenas prácticas de programación. Se aplicaron todas las funciones requeridas y los requisitos técnicos, lo cual permitió una estructura limpia, modular y fácil de mantener.

Al mismo tiempo se pudo evidenciar que el trabajar en equipo y dividir el proyecto, asignando diferentes tareas a los integrantes, permitía un desarrollo un poco más rápido y



eficiente, dando también la importancia al orden del desarrollo del proyecto desde un principio, logrando una finalización en el tiempo correspondido.

LOGROS ALCANZADOS:

- Se completó con éxito el desarrollo de todas las funcionalidades clave del sistema: gestión de clientes, propuestas, proyectos, contratos, entregables y finanzas.
- La aplicación cumple también con todos los requisitos técnicos.

ASPECTOS QUE FUNCIONARON BIEN:

- La planificación del Sprint fue clara y realista, permitiendo una distribución efectiva de tareas.
- La colaboración del equipo fue constante y fluida, con buena comunicación y resolución de bloqueos en tiempos razonables.
- Las pruebas manuales durante el desarrollo permitieron detectar errores de lógica y validar flujos de negocio en tiempo real.

ASPECTOS A MEJORAR:

- Hubo ciertos retrasos en la documentación, lo que impactó un poco al acercarse la hora de entrega.
- El manejo de errores podría ser aún más robusto, con respuestas más detalladas y personalizadas para el usuario final.

LECCIONES APRENDIDAS:

- Dividir tareas por conocimientos fue fundamental para el buen desarrollo del proyecto.
- Documentar constantemente las decisiones técnicas redujo la carga cognitiva y facilitó la incorporación de cambios.

