

Resumen de la Reunión Retrospectiva

Información de la empresa y proyecto:

Empresa / Organización	Duoc UC. Escuela de Informática y Telecomunicaciones
Proyecto	MisViáticos - Plataforma Integral de Gestión de Viáticos y Gastos Empresariales

Información de la reunión:

Lugar	Sala de reuniones de la empresa
Fecha	24/08/2025
Número de iteración / sprint	1
Personas convocadas a la reunión	SCRUM Master - Daniel Iturra Mario Bronchuer Carolina Pérez Jorge Rivas Paula Sandoval
Personas que asistieron a la reunión	Gabriel Muñoz Manuel Gonzales SCRUM Master - Valeria Vidal Daniel Iturra

Instrucciones:

La reunión retrospectiva es una herramienta del marco de trabajo Scrum, que pertenece a la familia de marcos de trabajo de desarrollo ágil, se realiza en cada iteración (denominado Sprint en Scrum), justo después de la reunión de revisión de la iteración (Sprint Review Meeting) con el dueño del Producto (Product Owner). En esta reunión deben revisarse tres aspectos, lo que salió bien durante la iteración (aciertos), lo que no salió tan bien (errores) y las mejoras que pudieran hacerse en la próxima iteración para evitar errores y mantener aciertos.

El dueño del producto (Product Owner) no asiste a la reunión, por lo que es una oportunidad para el equipo para poder hablar sin tapujos de los éxitos y fracasos, siendo importante para el equipo el analizar su propio desempeño e identificar estrategias para mejorar sus procesos. De forma similar, el Scrum Master (quien es el coach del equipo Scrum) puede observar impedimentos comunes que están afectando al equipo y tomar acciones para resolverlos.

La reunión usualmente se restringe a tres horas.

Formulario de reunión retrospectiva

¿Qué salió bien en la iteración? (aciertos)	¿Qué no salió bien en la iteración? (errores)	¿Qué mejoras vamos a implementar en la próxima iteración? (recomendaciones de mejora continua)
<p>Acta de Constitución del Proyecto completada</p> <ul style="list-style-type: none"> - Definimos claramente el alcance y objetivos de MisViáticos - Establecimos criterios de éxito medibles y restricciones del proyecto <p>Registro de Interesados completo</p> <ul style="list-style-type: none"> - Identificación exitosa de todos los stakeholders relevantes - Estrategias de comunicación definidas para cada grupo de interés <p>Matriz RACI establecida correctamente</p> <ul style="list-style-type: none"> - Responsabilidades claras entre Daniel (Backend/Arquitectura) y Mario (Frontend/UX) - Definición precisa de roles para cada entregable del proyecto 	<p>Subestimación del tiempo para EDT y MDI</p> <ul style="list-style-type: none"> - Requirió horas extras al final del sprint para completar la documentación - Falta de experiencia en descomposición de estructura de trabajo <p>Indefinición inicial del alcance del MVP</p> <ul style="list-style-type: none"> - Retrabajos en el documento de alcance por falta de claridad sobre límites - Dificultad para priorizar funcionalidades esenciales vs deseables <p>Validación tardía de Google Vision OCR</p> <ul style="list-style-type: none"> - No validamos tempranamente limitaciones de cuota del servicio - Descubrimiento de restricciones técnicas al final del sprint 	<p>Establecer daily standups de 15 minutos</p> <ul style="list-style-type: none"> - Reuniones diarias para sincronizar avances y bloqueos - Formato: ¿Qué hice ayer? ¿Qué haré hoy? ¿Tengo impedimentos? <p>Implementar tablero Kanban en GitHub Projects</p> <ul style="list-style-type: none"> - Visualización en tiempo real del estado de las tareas - Columnas: Backlog, To Do, In Progress, In Review, Done <p>Validar integraciones externas con POCs</p> <ul style="list-style-type: none"> - Crear pruebas de concepto antes de comprometer funcionalidades - Validar límites, costos y restricciones de servicios externos (OCR, AWS S3) <p>Definir horarios compartidos de trabajo</p> <ul style="list-style-type: none"> - Establecer bloques de 3 horas, 3 veces por semana para pair programming

¿Qué salió bien en la iteración? (aciertos)	¿Qué no salió bien en la iteración? (errores)	¿Qué mejoras vamos a implementar en la próxima iteración? (recomendaciones de mejora continua)
<p>Requerimientos funcionales y no funcionales documentados</p> <ul style="list-style-type: none"> - Levantamiento exhaustivo de 45+ requerimientos funcionales - Criterios de aceptación detallados para cada funcionalidad - Requerimientos no funcionales de seguridad, performance y escalabilidad definidos <p>Modelo Entidad-Relación diseñado</p> <ul style="list-style-type: none"> - Base de datos normalizada contemplando gastos, políticas, aprobaciones y auditoría - Identificación de índices necesarios para optimización de consultas <p>Comunicación efectiva del equipo</p> <ul style="list-style-type: none"> - Coordinación fluida durante la fase de planificación - Reuniones de sincronización semanales establecidas 	<p>Plan de Comunicaciones genérico**</p> <ul style="list-style-type: none"> - Falta de especificación de canales concretos para reportar avances - No se definieron protocolos para bloqueos o impedimentos <p>Dificultades de sincronización de horarios*- Otras asignaturas afectaron disponibilidad para trabajo colaborativo</p> <ul style="list-style-type: none"> - Productividad reducida por falta de sesiones conjuntas 	<ul style="list-style-type: none"> - Sesiones de revisión de código los viernes <p>Registro de decisiones arquitectónicas (ADR)</p> <ul style="list-style-type: none"> - Documentar decisiones técnicas importantes para evitar retrabajos - Formato: Contexto, Decisión, Consecuencias <p>Buffers de tiempo en estimaciones</p> <ul style="list-style-type: none"> - Agregar 20% de margen en todas las estimaciones futuras - Considerar contingencias para aprendizaje de nuevas tecnologías

Nota:

- Se recomienda utilizar viñetas (bullets) para enumerar los aciertos, errores y recomendaciones de mejora continua.
- El formulario se puede extender cuantas páginas sea necesario para registrar todos los aciertos, errores y recomendaciones.