



Esta pauta tiene como objetivo ayudarte a monitorear el desarrollo de tu Proyecto APT, reflexionando sobre tus avances de acuerdo con lo planificado en la fase anterior y recibiendo retroalimentación de tus pares y docentes que te permita hacer los ajustes necesarios para cumplir con los objetivos de tu proyecto. Esta pauta debe ser respondida con tu grupo.

Puedes completar esta guía y, posteriormente, cargarla en la sección de reflexión de la Fase 2, para retroalimentación de tu docente.

1. Mira tu carta Gantt y reflexiona sobre los avances de tu Proyecto APT

¿Has podido cumplir todas las actividades en los tiempos definidos? ¿Qué factores han facilitado o dificultado el desarrollo de las actividades de tu plan de trabajo?

Sí, hemos logrado cumplir con la mayoría de las actividades planificadas según nuestro roadmap de 18 semanas. Actualmente nos encontramos entre el **Sprint 6-7** (semanas 11-14), habiendo completado exitosamente:

Completado:

- Fundamentos y diseño (Sprint 1)
- Arquitectura base y configuración del entorno (Sprint 2)
- Backend core con Prisma y PostgreSQL (Sprint 3)
- APIs REST completas (Sprint 4)
- Frontend principal con React/Next.js (Sprint 5)
- Sistema de facturación con Stripe (Sprint 6)
- Portal del cliente completo (Sprint 6)
- Sistema de historiales médicos (Sprint 6-7)

Factores que facilitaron:

- Uso de tecnologías modernas y bien documentadas (Next.js, Prisma, Stripe)
- Arquitectura clara desde el inicio con separación frontend/backend
- Base de datos bien estructurada con Prisma Schema
- Metodología ágil con sprints bien definidos

Factores que dificultaron:

- Integración inicial de Stripe con manejo de pagos en pesos chilenos (CLP)
- Configuración del servicio de emails con Resend
- Gestión de autenticación entre diferentes roles (Admin, Veterinario, Cliente)
- Ajustes en las rutas de Next.js 13+ con App Router

2.

¿De qué manera has enfrentado y/o planeas enfrentar las dificultades que han afectado el desarrollo de tu Proyecto APT?

Hemos aplicado varias estrategias para superar los obstáculos:

Problemas técnicos resueltos:

- **Stripe/CLP:** Investigamos la documentación de Stripe para pagos sin centavos en pesos chilenos
- **Autenticación:** Implementamos middleware robusto con JWT y verificación de roles
- **PDFs:** Integramos jsPDF y jsPDF-AutoTable para generar comprobantes profesionales
- **Rutas duplicadas:** Depuramos sistemáticamente verificando estructura de carpetas

Metodología aplicada:

- Debug paso a paso con **console.log** para identificar errores
- Consulta de documentación oficial (Next.js, Prisma, Stripe)
- Pruebas iterativas: implementar → probar → corregir → mejorar
- Versionamiento con nombres claros de archivos y rutas consistentes

Planificación futura:

- Para el Sprint 8 (Testing): Implementar pruebas unitarias y de integración
- Para el Sprint 9 (Deploy): Configurar variables de entorno para producción
- Documentación técnica completa del sistema

3. Hasta el momento:

¿Cómo evalúas tu trabajo? ¿Qué destacas y qué podrías hacer para mejorar tu trabajo?

Evaluación general: 9/10

Aspectos destacados:

- Sistema funcional y completo con todas las funcionalidades críticas
- Diseño UI/UX profesional, intuitivo y responsive
- Arquitectura escalable y bien organizada
- Integración exitosa de servicios externos (Stripe, Resend)
- Gestión completa del flujo: desde agendar cita hasta historial médico
- Seguridad implementada con autenticación y autorización por roles

Áreas de mejora:

- Agregar más validaciones en formularios (frontend y backend)
- Implementar manejo de errores más robusto con mensajes personalizados
- Optimizar carga de imágenes (lazy loading, compresión)
- Agregar tests automatizados (Jest, React Testing Library)
- Mejorar accesibilidad (ARIA labels, navegación por teclado)
- Documentar mejor el código con comentarios explicativos

4. Después de reflexionar sobre el avance de tu Proyecto APT

¿Qué inquietudes te quedan sobre cómo proceder? ¿Qué pregunta te gustaría hacerle a tu docente o a tus pares?

Inquietudes:

1. ¿Cómo implementar notificaciones en tiempo real para recordatorios de citas?
2. ¿Qué estrategia de testing es más apropiada para este tipo de aplicación?
3. ¿Cómo preparar el deploy en producción (Vercel para frontend, Railway/Render para backend)?
4. ¿Es necesario implementar un sistema de respaldos automáticos de la base de datos?

Preguntas para el docente:

- ¿El alcance actual del proyecto es suficiente para la evaluación final?
- ¿Deberíamos priorizar más funcionalidades o enfocarnos en testing y documentación?
- ¿Qué aspectos de seguridad adicionales deberíamos considerar antes del deploy?
- ¿Es recomendable implementar un panel de analytics/reportes para el admin?

Preguntas para pares:

- ¿Cómo han manejado la integración de pagos en sus proyectos?
- ¿Qué herramientas usan para el testing de aplicaciones fullstack?
- ¿Han implementado sistemas similares de gestión de citas/historiales?

5. A partir de esta instancia de monitoreo de su Proyecto APT

¿Consideran que las actividades deben ser redistribuidas entre los miembros del grupo? ¿Hay nuevas actividades que deban ser asignadas a algún miembro del grupo?

N/A - Proyecto Individual

Este proyecto ha sido desarrollado de manera individual, por lo que no aplica la redistribución de tareas entre miembros. Sin embargo, he organizado mi trabajo personal de la siguiente manera:

Organización de actividades realizadas:

- **Semanas 1-4:** Planificación, diseño de arquitectura y configuración inicial
- **Semanas 5-8:** Desarrollo del backend (APIs, base de datos, autenticación)
- **Semanas 9-12:** Desarrollo del frontend (componentes, páginas, integraciones)
- **Semanas 13-14:** Funcionalidades avanzadas (pagos, PDFs, historiales médicos)

Próximas actividades personales (Sprints 8-9):

1. **Testing completo:** Pruebas unitarias y de integración
2. **Documentación técnica:** README, guías de instalación, documentación de API
3. **Optimización:** Revisar rendimiento, tiempos de carga, y experiencia de usuario
4. **Deploy:** Preparar y desplegar en producción (Vercel + Railway/Render)
5. **Video demo:** Grabar demostración del sistema funcionando

Gestión del tiempo: He logrado mantenerme organizado trabajando en sprints de 2 semanas, enfocándome en módulos específicos del sistema para evitar dispersión y mantener la productividad.

6. APT grupal

¿Cómo evalúan el trabajo en grupo? ¿Qué aspectos positivos destacan? ¿Qué aspectos podrían mejorar?

Evaluación del trabajo individual: 8.5/10

Aspectos positivos que destaco:

- **Autonomía y autogestión:** Logré planificar y ejecutar todas las fases del proyecto de manera independiente
- **Aprendizaje técnico:** Adquirí conocimientos sólidos en tecnologías modernas (Next.js, Prisma, Stripe, etc.)
- **Resolución de problemas:** Desarrollé habilidades para depurar y solucionar errores técnicos complejos de forma autónoma
- **Constancia:** Mantuve un ritmo de trabajo constante siguiendo el roadmap planificado
- **Calidad del producto:** El sistema es funcional, completo y profesional
- **Cumplimiento de plazos:** Logré cumplir con los tiempos establecidos en cada sprint
- **Iniciativa:** Implementé funcionalidades adicionales como PDFs, emails automáticos y portal del cliente

Aspectos a mejorar en mi trabajo:

- **Documentación durante el desarrollo:** Debí documentar mejor las decisiones técnicas en el momento, no al final
- **Testing desde el inicio:** Implementar pruebas desde las primeras fases del proyecto, no dejarlo para el final
- **Gestión del tiempo:** Aunque cumplí los plazos, algunas tareas tomaron más tiempo del esperado (ej. integración de Stripe)
- **Code review personal:** Establecer momentos para revisar y refactorizar mi propio código con más frecuencia
- **Priorización:** En algunos momentos me enfoqué en detalles estéticos antes que en funcionalidades críticas
- **Descansos programados:** Trabajar de forma más sostenible con pausas planificadas para evitar agotamiento