**Etapas y modelos del Ciclo de Vida en el que se encuentra el proyecto**

Actualmente, el proyecto se encuentra en la **fase de análisis y diseño funcional**, enmarcado dentro del **ciclo de vida del desarrollo de software**. El objetivo es construir una **plataforma digital para el cuidado de mascotas**, con una evolución proyectada hacia una aplicación móvil.

Para alcanzar este objetivo, es fundamental seleccionar una metodología de desarrollo que permita una planificación estructurada, pero que al mismo tiempo sea flexible, iterativa y centrada en el usuario. En este contexto, se analizan diferentes metodologías aplicables al ciclo de vida del software.

**Comparativa de metodologías de desarrollo de software**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Metodología | Características principales | Ventajas | Desventajas | Adecuación al proyecto |
| **Cascada** | Modelo secuencial y lineal. En cada fase debe completarse antes de pasar a la siguiente. (“Progamacion submodulo - Quizlet”) | Alta documentación, claridad estructural, útil para proyectos sin cambios | Poco flexible, difícil adaptación a cambios posteriores | Útil en la fase de planificación y documentación inicial, pero insuficiente por sí solo |
| **Scrum** | Metodología ágil basada en sprints (ciclos de desarrollo cortos e iterativos) y roles definidos (Scrum Máster, Product Owner, Equipo) | Adaptabilidad, entregas incrementales, retroalimentación continua | Requiere compromiso constante y buena gestión del backlog | **Altamente recomendable**, especialmente para validar funcionalidad y mejorar según el uso |
| **Kanban** | Flujo continuo de tareas visualizadas en un tablero (To Do, Doing, Done), sin sprints fijos | Visibilidad del flujo, simplicidad, ideal para mantenimiento o mejora continua | Menor estructura para desarrollos nuevos | Útil para mantenimiento y seguimiento posterior al MVP |
| **XP (Extreme Programming)** | Fuerte enfoque en buenas prácticas de programación, pruebas constantes y trabajo en parejas | Alta calidad del código, reducción de errores | Requiere madurez técnica y trabajo colaborativo riguroso | Solo recomendable si se forma un equipo con alta capacidad técnica |
| **Modelo en espiral** | Iterativo, con énfasis en análisis de riesgos en cada ciclo | Control de riesgos, enfoque técnico profundo | Costoso, complejo, poco ágil | No ideal para productos orientados al usuario final desde la primera entrega |

**Metodología seleccionada: SCRUM**

De acuerdo con la naturaleza del proyecto (orientado al usuario, con funcionalidades diversas y en evolución), **Scrum es la metodología más adecuada**, ya que:

* Permite **dividir el desarrollo en iteraciones (sprints)** de 2 a 4 semanas para generar entregables funcionales en corto tiempo.
* Facilita la **retroalimentación temprana y constante** de los usuarios (veterinarios, clientes, comercios).
* Es ideal para validar hipótesis, ajustar prioridades y evolucionar la plataforma desde un MVP hacia una solución robusta.
* Al integrarse con prácticas de **Kanban** (por ejemplo, para visualizar flujos de tareas), se mejora la **gestión operativa del equipo de desarrollo**.

**Etapas actuales del proyecto dentro del ciclo de vida**

|  |  |  |
| --- | --- | --- |
| Etapa | Descripción | Estado actual |
| 1. Levantamiento de requerimientos | Identificación de necesidades del usuario, benchmarking, análisis funcional | En desarrollo |
| 2. Diseño técnico y de arquitectura | Definición de módulos, flujos de información, modelo de datos, elección tecnológica | En desarrollo |
| 3. Desarrollo del MVP (web) | Implementación de funcionalidades básicas: login, compras, historial clínico, filtros | En planificación |
| 4. Pruebas continuas y feedback | Validación con usuarios reales en entornos controlados | Planificada |
| 5. Lanzamiento del MVP | Liberación de versión funcional desde navegador, accesible multiplataforma | Planeado |
| 6. Monitoreo y mejora continua (Kanban) | Uso de tableros visuales para priorizar mejoras, correcciones y nuevas funciones | Planeado |
| 7. Transición a aplicación móvil | Desarrollo nativo con foco en notificaciones, QR, GPS, pagos integrados | Planeado (fase 2) |

**Justificación Final de Metodologia**

**Scrum es la mejor opción para el proyecto** porque permite trabajar de manera iterativa con base en prioridades cambiantes, optimizando recursos y centrándose en la entrega de valor constante al usuario. La adopción de tableros Kanban como complemento permitirá mantener visibilidad, flexibilidad y eficiencia en la evolución del producto una vez que el MVP esté en producción.