Requerimientos del Proyecto

Los requerimientos son la base esencial para el éxito de cualquier proyecto de desarrollo de software. Establecer y comprender adecuadamente estos requisitos desde el principio es crucial para un desarrollo eficiente y satisfactorio. Aquí tienes una guía paso a paso para gestionar los requerimientos en tu equipo de desarrollo de software

1. Nombre del proyecto

|  |
| --- |
| *Sistema de Gestión de Productos Farmacéuticos - App Farmacia* |

1. Institución o empresa destinada como beneficiario del proyecto

|  |
| --- |
| *Cliente externo y Universidad del Valle - UNIVALLE* |

1. **Contextualización:** Transcripción de la historia del usuario desde el punto de vista de su contexto

|  |
| --- |
| *El cliente externo administra una farmacia en crecimiento, donde se enfrenta diariamente al desafío de mantener el control adecuado de los productos farmacéuticos disponibles, su vencimiento, ubicación en estantes, precios y stock actualizado. Actualmente, gran parte de estos procesos se realiza de forma manual o con herramientas no especializadas, lo que genera riesgos de errores humanos, pérdidas por productos vencidos, retrasos en la atención y complicaciones en la toma de decisiones comerciales.*  *Ante esta situación, el usuario expresa la necesidad de una aplicación móvil que centralice y digitalice toda la gestión del inventario farmacéutico. Este sistema debe permitirle registrar nuevos productos con sus respectivos atributos (nombre, proveedor, stock, vencimiento, imagen, etc.), editarlos conforme se actualicen los datos, eliminarlos si ya no están vigentes, y filtrarlos por nombre, proveedor o estante para su rápida localización.*  *Además, busca tener una vista clara del estado de vencimiento de cada producto mediante indicadores visuales (por colores o notificaciones), y desea registrar ventas con cálculo automático de subtotales y totales, integrando un historial y resumen estadístico. También desea organizar los productos según estantes físicos, para que tanto el inventario físico como el digital estén sincronizados.*  *El objetivo del cliente externo es tener* ***control total, visualización clara y operación eficiente****, desde su dispositivo móvil, para garantizar una atención farmacéutica segura, rápida y profesional.* |

1. **Definición de Objetivos:** El objetivo de un proyecto de software es desarrollar una solución eficiente y efectiva para satisfacer las necesidades específicas del cliente, garantizando calidad, funcionalidad y cumplimiento de plazos. Debe redactarse de manera clara y concisa, enfocándose en los resultados deseados y los beneficios para los usuarios finales.

|  |
| --- |
| ***Objetivo General del Proyecto:***  *Desarrollar una aplicación móvil integral que permita al cliente externo gestionar de forma eficiente el inventario de productos farmacéuticos de su farmacia, facilitando el registro, edición, eliminación, visualización y venta de productos, incluyendo el control de stock, vencimientos, organización por estantes y generación de reportes, todo ello con conectividad a Firebase para garantizar sincronización, seguridad y acceso desde múltiples dispositivos.*  ***Objetivos Específicos:***   1. ***Digitalizar el inventario farmacéutico*** *mediante el registro completo de productos con campos como nombre, descripción, imagen, precio, stock, proveedor y fecha de vencimiento.* 2. ***Permitir la edición y eliminación controlada*** *de productos obsoletos o con datos actualizados, con retroalimentación visual e interacción amigable.* 3. ***Implementar búsqueda y filtrado dinámico*** *por nombre, proveedor o descripción para facilitar el acceso rápido a la información.* 4. ***Incorporar alertas y colores visuales*** *que adviertan sobre productos próximos a vencer o ya caducados, minimizando pérdidas y errores en ventas.* 5. ***Organizar los productos por estantes físicos y digitales*** *para mantener orden estructural en el inventario.* 6. ***Registrar ventas de productos con cálculos automáticos*** *de subtotales, totales y descuentos, asociando cada orden a su código único.* 7. ***Visualizar un historial y dashboard de ventas****, incluyendo filtros por fecha, estadísticas diarias/mensuales y vista por calendario.* 8. ***Garantizar la sincronización de datos en tiempo real*** *mediante integración con Firebase, asegurando acceso confiable desde diferentes dispositivos.* 9. ***Proporcionar una experiencia de usuario moderna****, profesional e intuitiva, incluyendo un splash screen de bienvenida con identidad institucional.* |

1. **Definición del Alcance**

El alcance de un proyecto de software se refiere a los límites y las fronteras que definen qué se incluirá y qué no se incluirá en el proyecto. Es esencial establecer un alcance claro desde el principio para evitar malentendidos, cambios constantes y posibles desviaciones del objetivo principal. La definición del alcance se realiza generalmente durante la fase de planificación del proyecto y se documenta en el documento de especificación de requerimientos.

Cuando redactes el alcance del proyecto, es crucial involucrar a todas las partes interesadas relevantes para garantizar una comprensión común y un consenso sobre lo que se espera del proyecto. Además, el documento de alcance debe ser revisado y actualizado según sea necesario a lo largo del ciclo de vida del proyecto para reflejar cualquier cambio o ajuste en los requisitos.

La redacción del alcance del proyecto debe ser clara, detallada y comprensible para todas las partes involucradas. Aquí hay algunos elementos clave que deben abordarse al definir el alcance del proyecto de software:

1. **Inclusiones y Exclusiones**
   1. Enumerar detalladamente las características y funciones específicas que estarán incluidas en el software.
   2. Especificar cualquier característica o funcionalidad que esté explícitamente excluida del alcance del proyecto.

|  |
| --- |
| ***A. Inclusiones***  *El sistema* ***App Farmacia*** *incluirá las siguientes funcionalidades y características:*   1. ***Gestión de Productos Farmacéuticos***    * *Registro de productos con los campos: nombre, descripción, precio, stock, proveedor, fecha de vencimiento e imagen.*    * *Subida de imágenes desde cámara, galería o URL.*    * *Validación de campos obligatorios y formato de imágenes (con recorte previo a la subida).*    * *Edición de productos existentes (nombre, stock, precio, proveedor, imagen).*    * *Eliminación de productos obsoletos o duplicados con confirmación previa.* 2. ***Visualización e Inventario***    * *Lista de productos con imagen, nombre, stock y precio.*    * *Búsqueda dinámica por nombre, proveedor o descripción.*    * *Filtro y resaltado de productos por estado de vencimiento (vencido, por vencer, vigente).*    * *Mensajes contextuales y colores visuales (rojo, naranja, verde) según vencimiento.*    * *Visualización de detalles completos de cada producto, incluyendo estante asignado.* 3. ***Gestión de Estantes***    * *Registro, edición, eliminación y visualización de estantes físicos (nombre y descripción).*    * *Asignación de productos a estantes específicos.*    * *Consulta de productos por estante y resumen visual de cantidades.* 4. ***Gestión de Ventas***    * *Registro de ventas con selección directa desde inventario.*    * *Cálculo automático de subtotales, totales y cantidades.*    * *Asignación de código único por orden.*    * *Historial de ventas filtrado por fecha y detalle de cada orden.*    * *Visualización de productos vendidos con imagen y acceso a su ficha.*    * *Resumen visual en calendario de productos vendidos por día.*    * *Refrescado manual del historial de ventas.* 5. ***Alertas y Notificaciones***    * *Notificaciones visuales y automáticas para productos con vencimiento a 0, 30, 60 o 90 días.*    * *Agrupación de alertas por estantes afectados y productos próximos a vencer.*    * *Visualización de estadísticas de vencimientos por estante y gráficos resumidos.* 6. ***Interfaz de Usuario y Experiencia Visual***    * *Splash screen institucional con logotipo, fondo blanco y animaciones.*    * *Redirección automática a la vista principal tras la carga.*    * *Personalización nativa del splash en Android (rendimiento optimizado).*    * *Configuración de permisos para cámara y galería (ImagePicker).*    * *Iconografía institucional y metadatos de marca.* 7. ***Base de Datos y Backend***    * *Integración completa con Firebase Firestore y Clevercloud.*    * *Sincronización en tiempo real de productos, ventas y estantes.*    * *Compatibilidad multi-dispositivo.*   ***B. Exclusiones***  *Las siguientes funcionalidades están* ***fuera del alcance*** *del presente proyecto en esta fase:*   * *No se incluye integración con pasarelas de pago en línea.* * *No se contempla módulo de gestión de usuarios o autenticación (multirol).* * *No se considera impresión de tickets o facturas fiscales.* * *No se implementa respaldo o recuperación de datos fuera de Firebase.* * *No se integra con sistemas ERP, farmacias externas o seguros médicos.* * *No se realiza gestión de recetas médicas electrónicas.* * *No se incluyen estadísticas financieras detalladas (solo resumen de ventas).* * *No se contempla control de acceso por niveles o permisos personalizados por usuario.* |

1. **Límites Geográficos y Temporales**
   1. Establecer cualquier restricción geográfica si aplica.
   2. Definir el periodo de tiempo durante el cual se llevará a cabo el proyecto.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***c. Límites Geográficos***  *El desarrollo e implementación del sistema* ***App Farmacia*** *está destinado inicialmente a operar en el contexto geográfico de* ***Cochabamba, Bolivia****, específicamente en beneficio del* ***Cliente externo*** *y su entorno profesional o institucional inmediato. Sin embargo, al estar basado en tecnologías de nube como Firebase, el sistema es* ***potencialmente escalable*** *y* ***accesible desde cualquier región*** *con conectividad a internet, lo que permite su futura expansión a otros departamentos del país u otras farmacias interesadas.*  ***d. Límites Temporales***  *El desarrollo del proyecto* ***“Sistema de Gestión para Farmacia”*** *está programado para ejecutarse en un* ***periodo total de 6 semanas calendario****, distribuidas en fases específicas que abarcan desde la planificación hasta la puesta en marcha. A continuación, se detalla el cronograma tentativo:*   | ***Fase del Proyecto*** | ***Duración Estimada*** | | --- | --- | | *Análisis y diseño del sistema* | *1 semana* | | *Desarrollo de la interfaz móvil (frontend)* | *2 semanas* | | *Integración con Firebase (backend)* | *1 semana* | | *Implementación de ventas y vencimientos* | *1 semana* | | *Pruebas funcionales y ajustes finales* | *1 semana* |   ***Total, estimado:******6 semanas***  ***Fecha de inicio estimada:*** *Semana 1 de implementación académica*  ***Fecha de finalización estimada:*** *Semana 6 del calendario académico correspondiente.*  *Este límite temporal considera condiciones normales de desarrollo y pruebas. En caso de contingencias técnicas o solicitudes de mejora por parte del cliente, el cronograma puede ajustarse con la debida justificación y aprobación del tutor o cliente asignado.* |

1. **Restricciones y Dependencias**
   1. Identificar restricciones, como limitaciones tecnológicas o presupuestarias.
   2. Describir las dependencias con otros proyectos, sistemas o recursos.

|  |
| --- |
| ***e. Restricciones***  *El desarrollo del proyecto* ***“Sistema de Gestión para Farmacia”*** *presenta las siguientes restricciones técnicas, económicas y operativas:*   * ***Tecnológicas:***   + *La aplicación está diseñada exclusivamente para dispositivos con* ***sistema operativo Android versión 10 o superior****.*   + *Se utilizarán* ***Firebase*** *y* ***Firestore*** *como backend, lo cual limita la personalización avanzada del servidor.*   + *El sistema funcionará* ***únicamente en línea (requiere conexión a Internet)****; no se contempla funcionalidad offline.* * ***Presupuestarias:***   + *Se utilizarán únicamente herramientas gratuitas y de código abierto (Flutter, Firebase plan Spark).*   + *No se considerará la adquisición de licencias de software ni la contratación de servicios externos.* * ***Operativas:***   + *No se incluirán funcionalidades de* ***autenticación o gestión de usuarios*** *en esta primera versión.*   + *El sistema no tendrá* ***integraciones con pasarelas de pago****,* ***reportes en PDF/Excel****, ni* ***módulos de proveedores****.*   + *No se implementará control de acceso por roles ni encriptación avanzada de datos en esta fase.*   ***f. Dependencias***  *El proyecto depende de los siguientes factores y componentes externos:*   * ***Firebase****: Para la persistencia de datos, autenticación futura y almacenamiento de imágenes. El sistema está fuertemente acoplado a esta plataforma.* * ***Permisos del sistema operativo Android****: La correcta funcionalidad del sistema depende del acceso a cámara, almacenamiento y galería del dispositivo.* * ***Red estable de Internet****: El acceso a la base de datos y a los recursos remotos requiere conectividad continua.* * ***Soporte institucional****: Se necesita coordinación con el tutor (Ing. Christian Montaño Salvatierra) para validación de requerimientos, pruebas funcionales y retroalimentación.* * ***Equipo de desarrollo académico****: El cumplimiento del cronograma depende de la disponibilidad del equipo asignado (desarrollador y tutor).* |

1. **Usuarios y Stakeholders:**
   1. Definir quiénes son los usuarios finales y otros stakeholders del sistema.

**Usuario Final Primario:**

* **Rol:** Administrador de Farmacia (Cliente externo)
* **Perfil:** Persona con conocimientos básicos de gestión farmacéutica, sin conocimientos técnicos avanzados en sistemas.

**Stakeholders Clave:**

* **Cliente externo** – Beneficiario del sistema y solicitante del proyecto.
* **Universidad Privada del Valle (UNIVALLE)** – Institución educativa supervisora del proyecto.
* **Equipo de Desarrollo** – Responsable de análisis, diseño, implementación y entrega del sistema.
* **Asesor Académico/Técnico** – Encargado de supervisar y validar la calidad técnica del desarrollo.
  1. Especificar las expectativas y necesidades de los usuarios clave.

|  |  |  |
| --- | --- | --- |
| Usuario | Necesidades Específicas | Expectativas |
| Administrador de Farmacia | - Registrar y consultar productos - Realizar ventas rápidamente - Visualizar vencimientos - Tener trazabilidad de estantes | - Interfaz simple y rápida - Notificaciones claras - Sistema confiable sin errores - Carga rápida en dispositivos Android |
| Beneficiario (Cliente Externo) | - Validar funcionalidad propuesta - Obtener resultados útiles para gestión real | - Solución adaptable - Potencial comercial en futuro - Cumplimiento académico y técnico |
| UNIVALLE | - Validación del proceso metodológico del desarrollo | - Entregables formales y funcionales - Cumplimiento de estándares académicos |

* 1. Análisis de Interesados:

|  |  |  |  |
| --- | --- | --- | --- |
| Interesado | Nivel de Influencia | Nivel de Interés | Necesidades |
| Cliente Externo | Alto | Alto | Gestión digital completa de la farmacia |
| Usuarios de Farmacia | Medio | Alto | Facilidad de uso, confiabilidad |
| UNIVALLE | Alto | Medio | Evaluación metodológica y técnica |
| Desarrollador | Alto | Alto | Información precisa para desarrollo fluido |
| Tutor Académico | Medio | Medio | Control de calidad, avance progresivo |

* 1. Entrevistas con Usuarios Finales:

Durante la etapa de recolección de requerimientos se realizó una entrevista estructurada con el beneficiario del proyecto:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Entrevistado | Cargo | Fecha | Medio | Temas Tratados |
| Cliente Externo | Beneficiario / Propietario de Farmacia | 8 de mayo de 2025 | Reunión presencial | Requerimientos funcionales, flujo de ventas, vencimientos, usabilidad |
| Edwin Heredia Saravia | Desarrollador FullStack | 8 de mayo de 2025 | Reunión presencial | Validación de estructura de datos, mockups, flujo del sistema |
| Tutor UNIVALLE | Asesor Técnico Académico | 8 de mayo de 2025 | Reunión académica | Alcance del sistema, viabilidad, evaluación continua |

1. **Documentación de Requerimientos (Product Backlog)**

Detallar los requerimientos funcionales, que describen lo que el sistema debe hacer.

Especificar los requerimientos no funcionales, como los relacionados con el rendimiento, la seguridad y la usabilidad.

* **Requerimientos Funcionales:** Definir las funciones y operaciones que el software debe realizar.

|  |
| --- |
| *Transcripción de los Requerimientos Funcionales*   1. ***Registro de Productos Farmacéuticos****: El sistema debe permitir al administrador registrar productos farmacéuticos con todos sus datos obligatorios, incluyendo nombre, descripción, imagen, precio, stock disponible, proveedor y fecha de vencimiento. Además, debe ofrecer la opción de cargar imágenes desde la cámara o galería del dispositivo móvil.* 2. ***Modificación de Productos****: Los administradores deben poder editar los datos de productos previamente registrados. Esto incluye modificar el nombre, stock, precio, descripción, proveedor y fecha de vencimiento, garantizando siempre la validación de campos obligatorios.* 3. ***Eliminación de Productos****: El sistema debe permitir la eliminación de productos del inventario, solicitando confirmación mediante un diálogo modal antes de proceder con la acción para evitar eliminaciones accidentales.* 4. ***Visualización de Inventario****: El sistema debe mostrar una lista completa de productos con sus atributos clave (imagen, nombre, precio, stock y proveedor), actualizada en tiempo real tras cada modificación o eliminación.* 5. ***Búsqueda de Productos****: Los administradores deben poder buscar productos por nombre utilizando un campo de búsqueda dinámico, con filtrado en tiempo real, sin sensibilidad a mayúsculas/minúsculas y con mensajes claros si no se encuentran coincidencias.* 6. ***Gestión de Estantes****: El sistema debe permitir registrar nuevos estantes indicando nombre y descripción, así como editar o eliminar estantes existentes. Además, debe poder asignar productos a estantes específicos, visualizar el listado completo con el conteo de productos y consultar el detalle de cada estante con sus respectivos productos.* 7. ***Procesamiento de Ventas****: El sistema debe permitir realizar ventas de productos seleccionados desde el inventario, especificando cantidades, calculando automáticamente subtotales por producto y el total final. Al finalizar, debe guardarse la venta con la fecha y los datos relevantes.* 8. ***Historial y Detalle de Ventas****: Los usuarios deben poder consultar el historial de ventas realizadas, con la posibilidad de ver detalles completos de cada orden, incluyendo productos vendidos, cantidades, precios unitarios, subtotal y total.* 9. ***Visualización de Actividad Comercial****: El sistema debe mostrar un calendario que indique visualmente los días con ventas registradas, permitiendo revisar cuántos productos fueron vendidos por día.* 10. ***Notificaciones por Vencimiento****: El sistema debe identificar automáticamente productos próximos a vencer en 0, 30, 60 y 90 días, resaltándolos con colores diferenciados (rojo, naranja, amarillo, verde). Debe permitir al usuario forzar una actualización manual de vencimientos y realizar búsquedas específicas dentro de esta categoría.* 11. ***Dashboard Visual de Inventario y Ventas****: El sistema debe presentar un panel visual que muestre un resumen de productos próximos a vencer, cantidad de productos por estante y un gráfico de ventas diarias y mensuales.* 12. ***Splash Screen de Bienvenida****: Al iniciar la aplicación, el sistema debe mostrar una pantalla de bienvenida con fondo blanco, el logotipo institucional centrado y una animación de carga elegante. Esta pantalla debe permanecer visible durante 3 segundos antes de redirigir automáticamente a la vista principal del sistema.* |

* **Requerimientos No Funcionales:** Especificar aspectos no relacionados directamente con la funcionalidad, como rendimiento, seguridad y usabilidad.

|  |
| --- |
| *Transcripción de los Requerimientos NO Funcionales*  ***Usabilidad***  *El sistema debe tener una interfaz intuitiva, moderna y minimalista, diseñada para facilitar el uso a usuarios con conocimientos técnicos básicos.*  *Los formularios deben incluir validaciones visuales claras (mensajes de error, colores) para evitar entradas incorrectas.*  *La navegación debe ser coherente, fluida y con botones de acción fácilmente identificables en cada pantalla.*  *Todos los textos e íconos deben estar correctamente etiquetados y utilizar un lenguaje comprensible.*  ***Rendimiento***  *La aplicación debe cargar completamente la vista principal (HomeView) en menos de 2 segundos después del splash screen.*  *La búsqueda dinámica de productos y estantes debe mostrar resultados en menos de 500 milisegundos con hasta 100 registros.*  *El sistema debe ser capaz de funcionar correctamente en dispositivos Android con 2GB de RAM y Android 10 como mínimo.*  ***Persistencia y Conectividad***  *El sistema debe almacenar todos los datos en la nube mediante Firebase Firestore, garantizando disponibilidad desde múltiples dispositivos.*  *La aplicación debe manejar correctamente la pérdida de conectividad, mostrando mensajes adecuados al usuario e impidiendo operaciones no válidas.*  *La sincronización de datos debe realizarse en tiempo real para cambios críticos (ventas, productos, vencimientos).*  ***Seguridad y Privacidad***  *El sistema debe solicitar permisos explícitos para acceder a la cámara, galería y almacenamiento del dispositivo móvil.*  *El acceso a funcionalidades críticas como eliminación de productos debe requerir confirmación visual para evitar acciones accidentales.*  *El sistema no debe exponer información sensible a través de logs visibles o impresiones por consola.*  *Toda imagen capturada debe almacenarse temporalmente en rutas seguras controladas por la app (FileProvider).*  ***Mantenibilidad y Escalabilidad***  *El código debe estar modularizado siguiendo buenas prácticas de Flutter (separación de vistas, rutas, servicios y modelos).*  *El sistema debe permitir la futura incorporación de funcionalidades como autenticación, roles, estadísticas avanzadas o exportación de reportes sin reescritura total. El sistema debe incluir comentarios de código relevantes y archivos de configuración claramente documentados (firebase\_options.dart, app\_routes.dart, etc.).*  ***Compatibilidad Visual***  *La interfaz debe adaptarse correctamente a dispositivos de distintos tamaños: teléfono y tablet en orientación vertical.*  *El diseño debe respetar la identidad visual institucional, usando los colores institucionales (ej. verde esmeralda o teal), logotipo y estilo gráfico definidos.* |

1. **Criterios de Aceptación:**

Establecer los criterios que se utilizarán para evaluar si el proyecto ha sido completado satisfactoriamente.

|  |
| --- |
| *Transcripción de los Criterios de Aceptación*   * **Funcionalidad Completa y Verificada**:   + Todas las funcionalidades descritas en los requerimientos funcionales (CRUD de productos, ventas, vencimientos, estantes, splash, dashboard) han sido implementadas y verificadas.   + Las vistas y formularios permiten ingreso, edición, eliminación y visualización con datos almacenados de forma persistente en Firebase. * **Correcto Almacenamiento en Firebase**:   + Los datos registrados por el usuario deben almacenarse correctamente en Firebase Firestore y reflejarse en tiempo real (productos, ventas, estantes).   + Las operaciones de lectura y escritura no deben generar errores en la consola ni duplicaciones indebidas. * **Validaciones de Datos**:   + Todos los campos obligatorios en los formularios (producto, venta, estante) deben contar con validaciones visibles y no permitir envíos con datos vacíos o inválidos.   + Las imágenes deben ser válidas y provenir de fuentes controladas (cámara o galería con permiso). * **Interfaz de Usuario Intuitiva**:   + El diseño debe ser coherente, profesional e intuitivo en todas las pantallas del sistema, con navegación clara y sin errores de diseño.   + El splash screen debe mostrar el logotipo institucional, fondo blanco, animación de carga, y redirigir automáticamente al HomeView. * **Pruebas de Usabilidad y Fluidez**:   + El sistema debe responder fluidamente a acciones del usuario (ej. búsqueda dinámica, carga de vistas, interacción con listas).   + La carga de la pantalla principal no debe exceder los 2 segundos en condiciones normales. * **Manejo de Vencimientos**:   + El sistema debe mostrar notificaciones visuales de productos próximos a vencer (0, 30, 60, 90 días), usando códigos de color.   + Debe ser posible filtrar productos por vencimiento o forzar una actualización manual. * **Visualización de Datos**:   + El sistema debe mostrar en pantalla una lista de productos con imagen, stock, precio y vencimiento.   + Debe mostrar un dashboard básico con resumen de ventas por día/mes y productos por estante. * **Pruebas Superadas**:   + El sistema debe pasar al menos el 90% de los **casos de prueba funcionales** definidos.   + Los errores conocidos deben documentarse y no deben interferir con las funciones críticas del sistema. * **Documentación Técnica y de Usuario**:   + El proyecto debe incluir una guía de instalación y uso básico.   + El código fuente debe estar bien organizado y comentado, con archivo README.md, estructura de carpetas y dependencias documentadas (pubspec.yaml, firebase\_options.dart, etc.). |

1. **Estructuración del equipo de desarrollo**

|  |  |
| --- | --- |
| **Nombre del Equipo:** | **Equipo de Desarrollo UNIVALLE - Farmacia App** |
| **Cliente / Empresa:** | **Cliente Externo – Universidad Privada del Valle (UNIVALLE)** |
| **Enlace al Teams:** | [**General | PR2-25-APP-FARMACIA | Microsoft Teams**](https://teams.microsoft.com/l/team/19%3AcRu_Pu72PhtH2q4VNzs8VBQfDHwu6rajGmvAPufUUT41%40thread.tacv2/conversations?groupId=34b0c356-b723-418b-9c6d-4e33caaad38b&tenantId=77cab36c-76b2-48c3-9c54-cb9e4ee09b72) |
| **Enlace al Repositorio GIT:** | [**https://github.com/Young-Heredia/PR2-25-APP-FARMACIA.git**](https://github.com/Young-Heredia/PR2-25-APP-FARMACIA.git) |
| **Enlace al Figma:** | **N/A debido al tiempo de desarrollo récord de 6 semanas calendario académico.** |
| **Enlace al Tareas BackLog:** | [**2025.05.02 BACKLOG**](https://univalleedu.sharepoint.com/:f:/s/PR2-25-APP-FARMACIA/Ei-PG9dq5c1Mm627XdP0eCkBoAMWl2cV9W2ypl-JYS_zqg?e=frGqSi)  [**https://univalleedu.sharepoint.com/:f:/s/PR2-25-APP-FARMACIA/Ei-PG9dq5c1Mm627XdP0eCkBoAMWl2cV9W2ypl-JYS\_zqg?e=frGqSi**](https://univalleedu.sharepoint.com/:f:/s/PR2-25-APP-FARMACIA/Ei-PG9dq5c1Mm627XdP0eCkBoAMWl2cV9W2ypl-JYS_zqg?e=frGqSi) |

**Roles: [GitMaster | DBA Architech | Designer UX/UI | Q.A testing| ]**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nro** | **Nombre Completo** | **Roles: [GitMaster | DBA Architech | Designer UX/UI | Q.A testing| ]** | **Team Leader** |
| 1 | Edwin Heredia Saravia | Coordinador, Desarrollador FullStack, Git Máster, DB Architech, QA | Si |
| 2 | N/A | N/A |  |
| 3 | N/A | N/A |  |
| 4 | N/A | N/A |  |
| 5 | N/A | N/A |  |

1. **Definición del Stack de Tecnologías**

|  |  |
| --- | --- |
| **Área** | **Frameworks, Lenguajes, Motores, Compiladores, B.D.** |
| BackEnd | Firebase (Firestore como backend serverless en tiempo real) |
| FrontEnd | Flutter (Dart) |
| Base de datos | Firebase Cloud Firestore (NoSQL en la nube, escalable) |
| Otros | Cloudinary (gestión y almacenamiento de imágenes), Image Picker, Image Cropper, Firebase Core,  Smooth Page Indicator, Intl, Table Calendar, Permission Handler, Android Studio / VS Code, GitHub. |

1. **Asignación del Calendario**

|  |  |  |
| --- | --- | --- |
| **ACTIVIDAD** | **Semana** | **Fecha - Hora** |
| 1er. Sprint | Sem 17 marzo – 31 marzo | Hora de Clases |
| 2do.Sprint | Sem 31 marzo – 14 abril | Hora de Clases |
| 3er. Sprint | Sem 14 abril – 28 abril | Hora de Clases |
| Publicación - Hosteado | Sem 28 abril | Hora de Clases |
| Integración DevOops | Sem 28 abril – 12 mayo | Hora de Clases |
| Q.A. - Testing | Sem 12 mayo – 19 mayo | Hora de Clases |
| Defensa Interna | Sem 26 mayo | Hora de Clases |
| Defensa Publica (Con el Stakeholder) Únicamente equipos aprobados en la instancia anterior | Sem 2 Junio | Hora de Clases |
| Entrega Documentos, URL, y Carta de aceptación del Cliente | Sem 2 Junio | Hora de Clases |