**Requisitos Funcionales (Clasificación MoSCoW)**

**M - Must-have (Indispensable para la v1.0)**

* **Gestión de Pacientes:**
  + **Crear Perfil de Paciente:** Registrar un nuevo paciente con campos esenciales (Ej: Nombre, Apellido, Cédula, Teléfono, Fecha de Nacimiento).
  + **Listar Pacientes:** Ver una lista o tabla con todos los pacientes registrados en el sistema.
  + **Ver Detalle de Paciente:** Abrir el perfil de un paciente para consultar su información completa.
* **Gestión de Citas:**
  + **Agendar Nueva Cita:** Asignar una fecha, hora y tipo de terapia a un paciente existente.
  + **Visualizar Citas:** Ver un calendario o una lista de las citas programadas (por día, semana o mes).

*(****Observación:*** *Con estas 5 funcionalidades, el sistema ya es útil. Cumple su ciclo principal: registrar a alguien y darle una cita).*

**S - Should-have (Importante, pero para después del MVP)**

* **Sistema de Usuarios y Roles:**
  + Crear cuentas de usuario con login (usuario/contraseña).
  + Definir roles (Ej: "Administrativo" y "Fisioterapeuta") para controlar el acceso a la información.
* **Búsqueda y Filtros:**
  + Implementar una barra de búsqueda para encontrar un paciente rápidamente por nombre o cédula.
* **Historial Clínico Básico:**
  + Añadir un campo de "Notas de Sesión" en cada cita para que el fisioterapeuta registre el progreso del paciente.

**C - Could-have (Deseable, si hay tiempo y recursos)**

* **Edición de Pacientes y Citas:** Permitir modificar la información de un paciente o reprogramar una cita existente.
* **Dashboard Principal:** Una pantalla de inicio que muestre un resumen del día (citas de hoy, nuevos pacientes, etc.).
* **Generación de Reportes Simples:** Poder exportar una lista de pacientes o un reporte de citas entre dos fechas.
* **Alertas o Recordatorios Visuales:** Marcar citas pendientes o importantes en la interfaz.

**W - Won't-have (Explícitamente Fuera de Alcance)**

* *(Esta lista ya era muy buena, solo la he limpiado un poco)*
* **Sin Módulo de Facturación:** El sistema no gestionará cobros, facturas ni pagos.
* **Sin Autenticación de Dos Factores (2FA):** El login será simple (usuario/contraseña).
* **Sin Integraciones con APIs de Terceros:** No se conectará con calendarios externos, sistemas de email, etc.
* **Sin Personalización de Interfaz:** No habrá temas, modo oscuro o cambios de layout.

**Requisitos No Funcionales y Restricciones Técnicas**

Aquí definimos las **condiciones** y **limitaciones** del proyecto.

1. **Plataforma:**
   * Será una aplicación de escritorio desarrollada exclusivamente para el sistema operativo **Windows**.
2. **Alojamiento y Base de Datos (Hosting):**
   * La base de datos será **centralizada y auto-alojada** en un servidor dentro de la red local de la clínica (LAN).
   * **No se utilizará ningún servicio de hosting en la nube** (AWS, Azure, etc.), evitando costos recurrentes.
3. **Usabilidad:**
   * La interfaz de usuario debe ser **intuitiva y fácil de usar**, requiriendo mínima capacitación para el personal.
4. **Acceso a la Información:**
   * Toda la información de los pacientes será visible para los usuarios autorizados. **Los registros no estarán aislados** por el usuario que los creó.
5. **Idioma:**
   * El sistema estará disponible únicamente en **español**.

**Competidor Principal: El Sistema de Gestión Actual**

Este es tu punto de referencia. El equipo ya lo conoce, con sus virtudes y defectos.

* **Fortalezas (Hipótesis):**
  + **Familiaridad:** El personal ya está acostumbrado a sus flujos de trabajo, aunque sean ineficientes. El miedo al cambio puede ser un obstáculo.
  + **Datos Históricos:** Toda la información de los pacientes ya reside aquí. La migración de datos será un punto crítico de tu nuevo proyecto.
* **Debilidades (Tu Oportunidad):**
  + **Inflexibilidad (El problema central):**
    - **Campos Rígidos:** Probablemente no se pueden añadir nuevos campos al perfil del paciente (ej: "Referido por", "Diagnóstico Secundario") sin pedir un cambio al programador.
    - **Reportes Fijos:** Genera siempre los mismos reportes. Si se necesita un listado con columnas diferentes, es imposible para el usuario.
    - **Flujos de Trabajo Codificados:** El proceso para dar una cita o registrar un pago es siempre el mismo y no se puede adaptar a nuevos procedimientos de la clínica.
  + **Tecnología Obsoleta:** Puede ser lento, propenso a errores, o incompatible con sistemas operativos modernos.
  + **Interfaz Poco Intuitiva (UX Pobre):** Requiere muchos clics para tareas sencillas, el diseño es anticuado y visualmente cargado, lo que cansa al usuario y aumenta la probabilidad de error.
  + **Costo Cero**: Al ser un trabajo comunal se tiene la facilidad de parte de el personal administrativo de dejar de pagar una mensualidad y tener uno propio.

**Recopilación de Requisitos:**

* Se hará un login (Usuario y contraseña). Estos con la finalidad de saber quien esta operando el sistema.
* Proceso de recuperación de contraseña.
* Proceso de creación de usuarios.
* Se crea un icono para hacer log out.
* Se tiene una navbar con los apartados Pacientes, Reportes, Agenda, Test Evaluativos, Casos Relevantes
* PACIENTES
  + ~~En el apartado de Pacientes se tiene un símbolo para añadir, además de mostrar todos los pacientes~~
  + ~~Se tiene una barra para buscar por nombre o por numero de cedula~~
  + ~~Al hacer click en ese icono se abre una interfaz de CRUD para los pacientes~~
  + ~~En el apartado de leer se debe de generar una interfaz que indique todas las citas y el resumen de lo que se realizo en ella además de todos los datos.~~
  + ~~Al Crear un paciente se pide lo siguiente: Nombre completo, numero de cedula, enfermedades padece, genero, email, numero de teléfono, ¿Cuenta con pension? Si cuanto gana por pensión, tipo de pensión: No esta en tramite la pensión ,fecha de ingreso, fecha de nacimiento, personas con las que habita, ¿Tiene diabetes?, fecha de diabetes, observaciones, dirección de paciente ,~~ cantidad de encargados y habilitar la cantidad de campos basado en este dato, numero de teléfono encargado, cedula de encargado, dirección de encargado, foto de epicrisis.
  + ~~Al estar dentro de un paciente se tiene un icono el cual permite modificar todos los campos de ese paciente en especifico~~
  + ~~También se debe de tener la opción de eliminar un paciente al hacerlo debe de abrir un cuadro que indique el porque se esta eliminando y dos botones de cancelar y aceptar~~
  + ~~Esto no desaparece de la base de datos si no que se marca como inactiva eh igual si se quisiera consultar por ellos en un futuro y reactivarlos se podría hacer~~
* REPORTES
  + Se debe de tener un apartado en el cual se puedan elegir 2 fechas entre esas fechas se filtraran las citas que tuvieron acción en esos periodos de tiempo y estas no se mostraran solo se tomaran las estadísticas, además de los casos relevantes generados.
  + Al hacer click en el botón de generar vista previa se mostraran el siguiente texto:

\*\* De acá hasta   
Mes: {NombreDelMes}

Año: {Numero de año}

Persona que genera el reporte: {Persona que tenga el inicio de sesión}

Codigo: {Codigo de la persona iniciada}

Consultorio: Terapia Fisica

Resumen del mes:

Se atendieron un total de {Numero pacientes (Solo se cuentan una vez} usuarios. Se han realizado {Numero de citas} sesiones de fisioterapia.

Estadisticas de tratamiento

Numero total de usuarios atendidos: {Numero pacientes (Solo se cuentan una vez}

Numero total de electroterapias realizadas: {Numero de citas en las que se haya marcado electroterapia}

Numero total de sesiones de ejercicios terapéuticos: {Numero de citas en las que se haya marcado ejercicio terapeutico}

Numero total de terapias cognitivas PAM con DC: {Numero de citas en las que se haya marcado ejercicio terapias cognitivas PAM con DC}

Numero de ausencias justificadas en TF: {Cantidad de veces que no asistió el paciente}

Numero de ausencias injustificadas en TF: { Cantidad de veces que no asistió el paciente }

Numero de evaluaciones realizadas: {cantidad de pacientes generados }

Numero de terapias grupales: {Numero de citas marcadas como terapias grupales}

Aca el usuario no puede modificar\*\*

Medios terapéuticos mas utilizados: { Espacio para que el usuario escriba }

Dificultades encontradas {Espacio para que el usuario escriba}

\*\*Aca el usuario no puede modificar

Casos destacados:  
{tarjetas que muestren los casos destacados de ese mes

\*\*

{Espacio de comentarios para el usuario}

* + Al hacer click en generar reporte se debe de tomar la prevista y se descargara un pdf automáticamente
* AGENDA
  + ~~Se pondrá un calendario al costado derecho en el cual se verá el mes, el año y los días de el mes~~
  + ~~Se tendrá un icono en el cual al hacer click se desplegara un formulario en el cual se pedirá un paciente registrado y si es diabético o no y la información relacionada, el motivo de la consulta, la hora y la fecha, si la cita es grupal se puede indicar~~
  + ~~Al hacer click en el dia y si hay una cita registrada se mostrara en estilo de una carta en el costado izquierdo y se tendrá la siguiente información: Nombre del paciente, Hora de la cita y las enfermedades, estas cartas se ordenaran desde la que tenga la hora mas temprana hasta la mas tardia.~~
  + ~~Al hacer click en una de estas citas se dará la opción de marcarla como completa (Al hacer esto se cierra la cita y no se puede editar a no ser que se haya marcado como ausencia ahí si se puede modificar para justificar), como si hay una ausencia (Esta se puede justificar mas adelante), nombre de la persona logeada, y se dan varias casillas que se pueden seleccionar mas de una (Electroterapia,Ejercicio Terapeutico, Ejercicio Terapias Cognitivas PAM con DC), también se dará un botón para mostrar el historial clínico registrado, además de que se podrán anotar los siguientes datos: donde duele, porque se hizo la cita, la razón de la consulta, Terapia realizada (Puede ser mas de una y se habilitan los campos si se hace click en un icono), procedimiento, signos vitales (Pulso (Numero bpm), Oxigeno (Numero SpO2)), medidas (Peso (Numero kg), Altura (numero cm)), Presion Arterial(Sistolica(numero mmHg),Diastolica(numero mmHg))~~
* TEST EVALUATIVOS
  + Se presentara una sección en donde se debe escoger al paciente y luego escoger la prueba a realizar
* CASOS ESPECIALES
  + Al abrir se vera todos los casos se vera nombre del paciente, titulo del caso, comentario y fecha
  + Se tendrá un buscador para buscar por paciente
  + Se tendrá un apartado para generar un caso que pedirá seleccionar al paciente, un texto a grandes rastros que dira que fue lo que paso, un espacio de texto para decir que sucedió y como y se autogenerara la fecha a la hora de hacer click en el botón

**Requisitos No Funcionales**

**Elección de la Metodología de Desarrollo**

Se elegira el método cascada dado que los requerimientos son muy claros y no van a cambiar dado que el cliente ya mostro su total aprobación a los requerimientos

**Planificación del Proyecto (Project Plan)**

**Plan de Proyecto: Gestión de Terapias - Centro Diurno Francisca Valerio**

**Estudiante: Jason Reynaldo Orozco Ruiz**

**Total de Horas: 150**

**Metodología: Cascada**

**Fase 1: Planificación y Análisis (Total: 12 Horas)**

**|**

**| Tarea | Descripción | Horas Estimadas |**

**| 1.1. Planificación Inicial | Ajuste y desglose del cronograma de actividades. Definición de entregables y configuración del entorno de trabajo inicial (control de versiones Git, carpetas de proyecto). | 6 |**

**| 1.2. Refinamiento de Requerimientos | Revisión final de los requerimientos detallados con el supervisor para confirmar prioridades. Aclaración de dudas sobre los campos específicos del CRUD de pacientes y el formato de reportes. | 6 |**

**Fase 2: Diseño del Sistema (Total: 30 Horas)**

**| Tarea | Descripción | Horas Estimadas |**

**| 2.1. Diseño de la Base de Datos | Creación del diagrama Entidad-Relación. Definición de tablas, campos, tipos de datos y relaciones (Usuarios, Pacientes, Citas, Encargados, Notas, etc.). Escritura de los scripts SQL para la creación de la base de datos. | 14 |**

**| 2.2. Diseño de Arquitectura | Definición de la estructura del proyecto en Django y React. Diseño de la API REST para la comunicación entre el backend y el frontend. Planificación de los componentes principales en React. Configuración inicial de Docker (Dockerfile, docker-compose.yml). | 16 |**

**Fase 3: Desarrollo (Total: 78 Horas)**

**Este es el núcleo del proyecto. Se divide en sub-módulos priorizados según MoSCoW.**

**| Tarea | Módulo | Descripción | Horas Estimadas |**

**| 3.1. Configuración del Entorno | Montaje del proyecto base en Django y React. Conexión a la base de datos. Creación de la estructura de carpetas y archivos inicial. | 8 | |**

**| 3.2. Autenticación y Usuarios (S) | Creación de modelos de usuario y roles. Implementación de login, logout, y protección de rutas. | 12 | |**

**| 3.3. Gestión de Pacientes (M) | Desarrollo del CRUD completo para pacientes. Incluye el formulario con todos los campos detallados, la vista de lista, la búsqueda (S) y el perfil detallado. | 22 | |**

**| 3.4. Gestión de Citas (M) | Creación del formulario para agendar citas, vinculación con pacientes, y lógica para marcar estados (completa, ausencia). | 16 | |**

**| 3.5. Agenda y Visualización (M) | Implementación del calendario en el frontend y la visualización de citas por día en formato de tarjetas. | 10 | |**

**| 3.6. Historial Clínico (S) | Implementación de la funcionalidad para añadir y ver notas de sesión en cada cita. | 10 | |**

**Fase 4: Pruebas y Ajustes (Total: 18 Horas)**

**| Tarea | Descripción | Horas Estimadas |**

**| 4.1. Pruebas Funcionales | Pruebas manuales de cada funcionalidad desarrollada: creación de pacientes, agendamiento de citas, login, etc. Se verifica que todo funcione según los requerimientos. | 10 |**

**| 4.2. Corrección de Errores | Resolución de los bugs y errores encontrados durante la fase de pruebas funcionales. | 8 |**

**Fase 5: Implementación y Cierre (Total: 12 Horas)**

**| Tarea | Descripción | Horas Estimadas |**

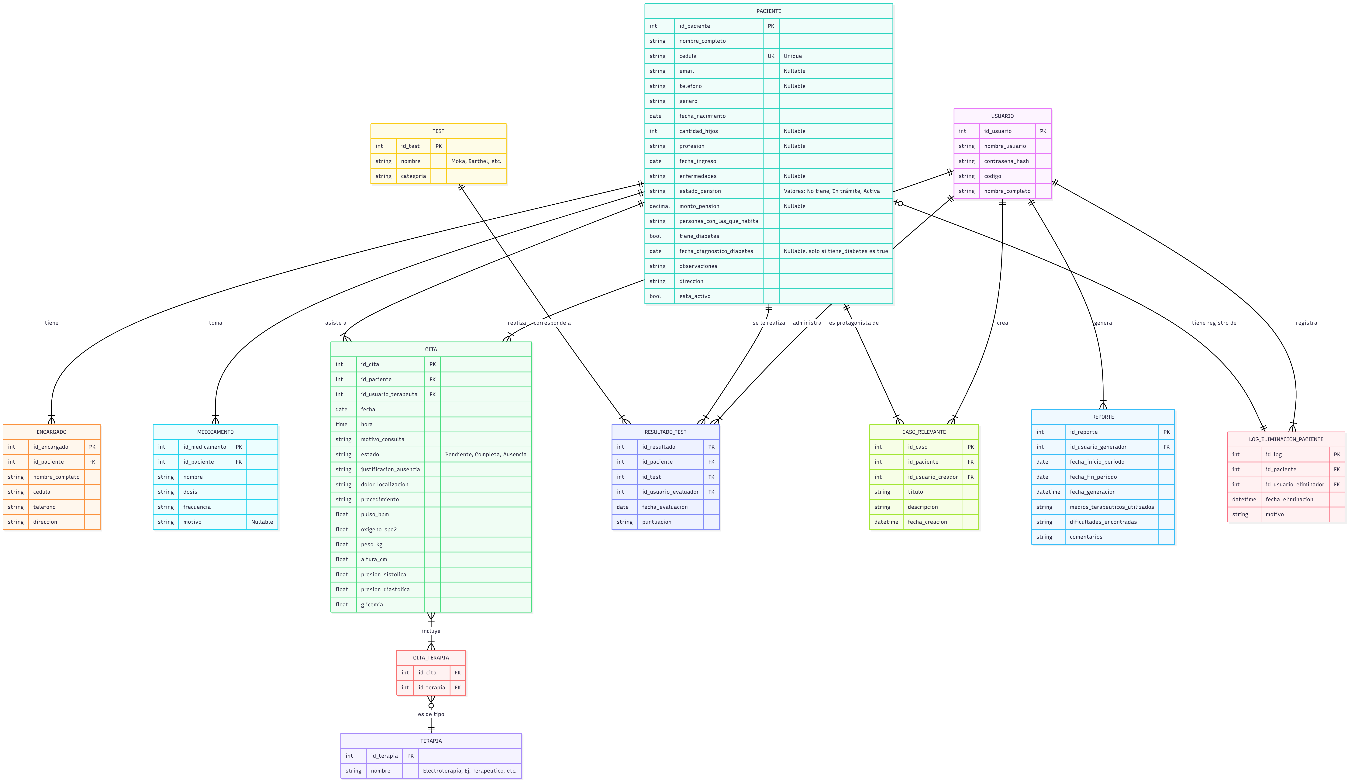
**| 5.1. Despliegue Local | Preparación del entorno en la red local de la clínica y despliegue de la aplicación usando Docker. | 6 |**

**| 5.2. Documentación y Capacitación | Creación del manual de usuario final y realización de una sesión de capacitación con el personal del centro. | 6 |**

**TOTAL GENERAL: 150 Horas**

**Diseño de la Arquitectura del Software**

Se elegirán los lenguajes de sql para la base de datos, Python para el backedn, js para el frontend y los frameworks serán Django para apoyar el backend y React para construir el frontend, además de usar Docker el cual nos dejara levantar el proyecto donde sea.  
  
**Diseño de la Base de Datos**

****

**Diseño de la Interfaz de Usuario (UI) y Experiencia de Usuario (UX)**Estos machotes se encuentran en la carpeta en formato html

**Escribir la Documentación Inicial**El readme se encuentra en el repositorio git o en la carpeta actual