|  |
| --- |
|  |

|  |
| --- |
| Pre-Fall – Sistema inteligente para la prevención y predicción de caídas |

|  |
| --- |
| **E4.1 – Plan de Pruebas** |

|  |  |
| --- | --- |
| Proyecto | Pre-Fall – Sistema inteligente para la prevención y predicción de caídas |
| Entregable | E4.1 – Plan de Pruebas |
| Fecha | 19/09/2023 |

Contenido

[Contenido 1](#_Toc151393589)

[Resumen Ejecutivo 2](#_Toc151393590)

[Índice de Figuras y Tablas 3](#_Toc151393591)

[1 Introducción 5](#_Toc151393592)

[1.1 Objetivos del plan de pruebas 5](#_Toc151393593)

[1.2 Documentos relacionados 5](#_Toc151393594)

[2 Alcance de las pruebas 7](#_Toc151393595)

[2.1 Elementos de pruebas 7](#_Toc151393596)

[2.2 Funcionalidades a no probar 9](#_Toc151393597)

[2.3 Objetivo de Calidad 9](#_Toc151393598)

[2.4 Roles y responsabilidades 9](#_Toc151393599)

[3 Metodología de las pruebas 10](#_Toc151393600)

[3.1 Descripción general 10](#_Toc151393601)

[3.2 Niveles de pruebas 10](#_Toc151393602)

[3.3 Criterios de Aceptación y Rechazo 10](#_Toc151393603)

[3.4 Entregables 10](#_Toc151393604)

[3.5 Recursos 10](#_Toc151393605)

[3.6 Entorno de Pruebas 11](#_Toc151393606)

[4 Planificación y Organización 12](#_Toc151393607)

[5 Referencias 13](#_Toc151393608)

Resumen Ejecutivo

En este Entregable 4.X

Índice de Figuras y Tablas

[Tabla 1: Datos del proyecto 5](#_Toc151038442)

[Figura 1: Plan de Captura de Datos 10](#_Toc151038443)

# Introducción

|  |  |  |  |
| --- | --- | --- | --- |
| **Proyecto(s)** | | **Tipo de Proyecto** | |
| **PreFALL** | | CDTI | |
| **Documentos Evaluación relacionados** | | | |
| E1.2 – Catálogo de requisitos (E12.pdf)  E1.3 – Diseño conceptual del sistema (E13.pdf) | | | |
| **Equipo de Proyecto** | | | |
| **Jefe de Equipo** | Jose Luis Parreño Catalán  (IBC) | **Arquitecto de Producto** | Iván Jiménez Utiel (IBC)  Bernat Díaz (IBC) |

Tabla 1: Datos del proyecto

## Objetivos del plan de pruebas

* Evaluar la eficacia del sistema de prevención de caídas en la identificación de riesgos de caídas en pacientes.
* Garantizar que el sistema funcione de manera precisa y confiable para pacientes con y sin riesgo de caídas.
* Garantizar la calidad y fiabilidad del software de registro de datos de sensores y del modelo de IA utilizado para prevenir las caídas en pacientes.
* Evaluar la capacidad del modelo de IA para emitir un porcentaje de riesgo de caída de manera precisa.

## Documentos relacionados

Las pruebas abarcan todas las etapas del proceso, desde la captura de datos de movimiento de la marcha del paciente hasta la generación de informes de riesgo de caída. Se evaluará el rendimiento en 67 casos, incluyendo 46 hombres y 21 mujeres.

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Descripción** | **Entregable del proyecto** |
| E12.pdf | Informe de Requisitos | Entregable 1.2 |
| E13.pdf | Informe de Diseño conceptual y Casos de Uso | Entregable 1.3 |
| E23.pdf | Captura de datos inicial | Entregable 2.3 |

Tabla 2: Documentos relacionados

# Alcance de las pruebas

Las pruebas se realizarán a los módulos y submódulos del sistema PreFALL.

## Elementos de pruebas

|  |  |
| --- | --- |
| **COMPONENTES A SER PROBADOS** | **MÓDULOS Y PROCESOS** |
| **Web App (WA)** | **WA1: Módulo de Autenticación**   * Proceso de login y logout * Proceso de cambio de contraseña   **WA: Módulo de Administrador general**   * Proceso de creación centros * Proceso de edición de la información de cada centro * Proceso de creación y borrado del *admin* del centro * Proceso de edición y borrado de todos los usuarios   **WA3: Módulo de Administración centro**   * Proceso de edición de la información de su centro * Proceso de creación de usuarios * Proceso de edicion y borrado de usuarios   **WA4: Módulo de Auxiliar**   * Proceso de creación de pacientes * Proceso de edición de detalles personales de cada paciente * Proceso de subida de tests de la marcha * Proceso de asociación y desasociación de médicos para un paciente   **WA5: Módulo Médico:**   * Proceso de creación de pacientes * Proceso de edición de detalles clínicos de cada paciente * Proceso de subida de tests de la marcha * Proceso de análisis de tests de la marcha * Proceso de borrado de tests de la marcha. * Proceso de descarga de tests de la marcha. * Proceso de escribir y editar el diagnostico del test. * Proceso de recomendaciones (deben cambiar dependiendo de la probabilidad de caída) * Proceso de informes:   + Informes individuales   + Informes de comparativas (deben salir tambien las métricas del último informe seleccionado)   + Descargar los gráficos en pdf.   + Limpiar los gráficos seleccionados   + Seleccion de tests sin revisar y tests sin diagnosticar   **WA6: Módulo Paciente**   * Ver detalles de cada test de la marcha * Proceso de descarga del test. * Proceso de informes:   + Añadir tests a comparar   + Descargar gráficos en pdf.   + Limpiar los gráficos seleccionados. |
| **Software de Dispositivo (DA)** | **DA1: Módulo de Login**   * Comprobación de rol médico o axiliar.   **DA2: Módulo de subida de tests**  **DA3: Módulo de Gestión de dispositivos:**   * Proceso de escaneo de los sensores y cámaras. * Proceso de conexión y desconexión de los sensores. * Proceso de apertura y cierre de las cámaras. * Proceso de captura de los sensores. * Proceso de pausar la captura de los sensores y grabación de los sensores y cámara. (comprobar que la grabación se detiene correctamente) * Proceso de grabar un nuevo test para un paciente. * Proceso de detener la grabación del test (stop). * Proceso de pasar registro datos y/o video.   **DA3: Módulo de Gráficos**   * Proceso de renderizar gráficos para ACC, MAG, GYR, QUAT, * Proceso de importar gráficos de CSV * Proceso de carga de Video desde fichero AVI   **DA4: Módulo de Grabación a Fichero**   * Proceso de grabación de streaming de datos a fichero CSV * Proceso de grabación de streaming de video a fichero AVI.   **DA5: Módulo de timeline**   * Proceso de carga de test * Cuadro de control de movimientos * Mover el timeline para cambiar también el tiempo. |
| **API para Modelo de Predicción (M)** | **M1: Módulo de API para ejecución modelo**   * Acceso a predicción mediante URI * Proceso de ejecución de modelo * Retorno de resultados:   + Porcentaje de riesgo   + Métricas: duración fases, medias, desviaciones típicas. |
| **Base de Datos (BD)** | **BD1: Módulo CRUD de gestión de bases de datos**   * Conexión/Desconexión * Inserción * Modificación * Borrado |

Tabla 1: Módulos para realizar las pruebas

## Funcionalidades a no probar

Por ser elementos secundarios que intervienen por detrás en los componentes anteriores o se han probado en otra fase, no se tendrán en cuenta los siguientes módulos:

* Módulo de modelo: pruebas internas del modelo de pro
* Módulo de HW: Los sensores ya están probados así como cámaras
* Módulo de gestión de tareas Celery: para encolar tareas de predicciónde modelos
* Módulo de envío de mensajes RabbitMQ: para enviar mensajes cuando tareas han finalizado.
* Lógica de base de datos
* Seguridad web y rendimiento

## Objetivo de Calidad

El objetivo de las pruebas es verificar la funcionalidad del software de aplicación y la aplicación web.

Este plan se enfocará en cómo el software de aplicación es capaz de capturar datos de los sensores que podrá interpretar en el mismo tiempo real. Dichos datos se podrán subir a la plataforma web.

La plataforma web será responsable de poder comunicarse mediante un API para predecir el riesgo de caída y devolver unos informes y gráficos, además del porcentaje de riesgo de caída.

## Roles y responsabilidades

|  |  |  |
| --- | --- | --- |
| Número | Miembro | Tareas |
| 1 | Gestor de pruebas | Gestiona todo el proyecto.  Define las direcciones a llevar a cabo  Adquiere los recursos apropiados |
| 2 | Analista de pruebas | Diseña las pruebas de forma que sean ágiles e integra una integración y mejora continua de las mismas. |
| 3 | Desarrollador de pruebas | Implementa los casos de prueba o suites de pruebas. |

# Metodología de las pruebas

## Descripción general

En el proyecto PreFALL se desarrollará una metodología que permita el desarrollo de pruebas relacionadas al diseño arquitectónico del sistema y su validación.

## Niveles de pruebas

|  |  |
| --- | --- |
| Pruebas de Integración | Módulos de software individuales son combinados y probados como un grupo. Esto se llevará a cabo en los casos de pruebas desplegados a partir de los Casos de Uso. |
| Pruebas de Sistema | Realizadas en un sistema completo e integrado para evaluar el cumplimiento del sistema con sus requisitos especificados. |

## Criterios de Aceptación y Rechazo

* Criterios de Aceptación o Rechazo
  + Completar al 100% las pruebas de integración.
  + Completar al 100% las pruebas de sistema.
  + Completar un mínimo del 90% de casos de éxito.
* Criterios de Suspensión
  + Porcentaje de fallas superior al 50%.
  + Fallo de un componente a la hora de realizar las pruebas de integración.
  + Cambio de requerimientos.
* Criterios de Reanudación
  + Correcciones de fallos, porcentaje de fallos bajado a menos del 50%.
  + Correcciones de módulos para integración.
  + Comprobación de módulos mediante pruebas unitarias con éxito.

## Entregables

La ejecución del plan de pruebas tendrá la siguiente documentación:

* Documento de Plan de Pruebas (este documento).
* Casos de Pruebas Funcionales
* Casos de Pruebas de Sistema
* Logs de errores.

## Recursos

|  |  |  |
| --- | --- | --- |
| **Num** | **Recursos** | **Descripciones** |
| 1 | Servidor Cloud | Basado en Docker donde estarán expuestos los microservicios que componen el ecosistema:   * Frontend web * API * Base de Datos * Desktop App |
| 2 | Herramienta de tests | Manuales |
| 3 | PC | Al menos Core i7 con 8 GB de Ram , SSD Disco duro y GPU |

## Entorno de Pruebas

|  |
| --- |
| **Requerimientos de Entornos – Hardware**   * Servidor será alojado en un hosting de la empresa (PLESK) que nos dotara de un servidor con procesadores Xenon y 48GB de Ram. * TurboSSD discos duro de alto rendimiento en RAID. * Equipos de escritorio Core7 3 generación, 8gb de Ram. * Sensor IMU y dongle |
| **Requerimientos de Entornos – Software**   * Docker * Contenedor para Python Flask * Contenedor para Python Fast-API * Contenedor para Mysql 8.0 * Contenedor para RabbitMQ * Contenedor para Celery |
| **Requerimientos de Entornos – Pruebas**   * Genere Data. * JMeter. * Karma. |

# Planificación y Organización

## Objetivos principales

1. Establecer Prioridades:

* Identificar módulos críticos para la funcionalidad general del sistema.
* Priorizar la secuencia de prueba en función de la importancia y dependencias entre los módulos.

1. Definir Estrategia de Pruebas:

* Seleccionar entre pruebas manuales y automatizadas según la criticidad y la repetitividad de las funcionalidades.
* Establecer estrategias específicas para módulos complejos, como el Módulo de Autenticación y Módulo de API.

1. Asignar Recursos:

* Designar roles y responsabilidades específicos para cada miembro del equipo de pruebas.
* Considerar la especialización de habilidades al asignar tareas (por ejemplo, la automatización puede ser manejada por el Desarrollador de Pruebas).

1. Establecer Cronograma:

* Dividir el período de 20 días laborables en fases específicas, asignando tiempo a cada módulo de prueba.
* Considerar posibles solapamientos para maximizar la eficiencia.

1. Identificar y Preparar Entornos de Pruebas:

* Asegurar que el entorno de pruebas refleje con precisión el entorno de producción.
* Coordinar con el equipo de desarrollo para garantizar la disponibilidad de entornos adecuados.

## Tareas específicas de cada módulo

* Módulos de Web App (WA1 a WA6):
  + Iniciar con el Módulo de Autenticación (WA1) dada su importancia.
  + Progresar hacia los módulos de administración y usuarios, considerando las dependencias lógicas.
* Módulos de Desktop App (DA1 a DA5):
* Comenzar con el Módulo de Login (DA1) y avanzar hacia la gestión de dispositivos y funciones específicas.
* Módulos Generales (M1, BD1):
  + Integrar pruebas del Módulo de API (M1) con los módulos de aplicación.
  + Validar las funciones CRUD del Módulo de Gestión de Bases de Datos (BD1).

## Planificación del procedimiento

|  |  |  |
| --- | --- | --- |
| **Días** | **Tareas** | **Módulo** |
|  | **Módulos de la Aplicación Web** |  |
| 1-2 | Análisis y Diseño:   * Revisar requisitos y diseñar casos de prueba para el Módulo de Autenticación (WA1).   Ejecución de Pruebas:   * Iniciar pruebas funcionales y de sistema en el Módulo de Autenticación. | WA1, WA2 y WA3 |
| 3-4 | Análisis y Diseño:   * Diseñar casos de prueba para el Módulo de Administrador General (WA2).   Ejecución de Pruebas:   * Continuar pruebas funcionales y de sistema en el Módulo de Administrador General. |
| 5 | Análisis y Diseño:   * Diseñar casos de prueba para el Módulo de Administración de Centro (WA3).   Ejecución de Pruebas:   * Iniciar pruebas funcionales y de sistema en el Módulo de Administración de Centro. |
|  | **Módulos de Desktop App** |  |
| 6-7 | Análisis y Diseño:   * Revisar requisitos y diseñar casos de prueba para el Módulo de Login (DA1).   Ejecución de Pruebas:   * Iniciar pruebas funcionales y de sistema en el Módulo de Login. | DA1,  DA2,  DA3,  DA4,  DA5 |
| 8-9 | Análisis y Diseño:   * Diseñar casos de prueba para los Módulos de Subida de Tests y Gestión de Dispositivos (DA2 y DA3).   Ejecución de Pruebas:   * Continuar pruebas funcionales y de sistema en los Módulos de Subida de Tests y Gestión de Dispositivos. |
| 10 | Análisis y Diseño:   * Diseñar casos de prueba para los Módulos de Gráficos, Grabación a Fichero y Timeline (DA4 y DA5).   Ejecución de Pruebas:   * Iniciar pruebas funcionales y de sistema en los Módulos de Gráficos, Grabación a Fichero y Timeline. |
|  | **Módulos Generales: Modelo y CRUD BD** |  |
| 11-12 | Análisis y Diseño:   * Revisar requisitos y diseñar casos de prueba para el Módulo de API para Ejecución de Modelo (M1).   Ejecución de Pruebas:   * Iniciar pruebas funcionales y de sistema en el Módulo de API. | M1 y  BD1 |
| 13-14 | Análisis y Diseño:   * Diseñar casos de prueba para el Módulo CRUD de Gestión de Bases de Datos (BD1).   Ejecución de Pruebas:   * Continuar pruebas funcionales y de sistema en el Módulo CRUD de Bases de Datos. |
|  | **Módulo Paciente (WA5, WA6), Validación y Retesting** |  |
| 15 | Análisis y Diseño:   * Diseñar casos de prueba para el Módulo Médico (WA5).   Ejecución de Pruebas:   * Iniciar pruebas funcionales y de sistema en el Módulo Médico. | WA5 |
| 16 | Análisis y Diseño:  Diseñar casos de prueba para el Módulo Paciente (WA6).  Ejecución de Pruebas:  Iniciar pruebas funcionales y de sistema en el Módulo Paciente.   * . | WA6 |
| 17-18 | Validación Final:   * Realizar validación final del sistema, asegurándose de que todos los módulos interactúan correctamente.   Ejecución de Pruebas:   * Realizar pruebas adicionales según sea necesario y corregir defectos encontrados. |  |
| 19-20 | Retesting y Ajustes Finales:   * Realizar pruebas de regresión para asegurarse de que las correcciones de defectos no introduzcan nuevos problemas.   Hacer ajustes finales según sea necesario. |  |

## Matriz de responsabilidades

Roles / Responsabilidades: R: Responsable, A: Aprobador, C: Consultado,

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Actividad | Jose Luis Parreño | Iván Jiménez | Bernat Díaz |
| 1 | WA1-WA6 | A | R | C |
| 2 | DA1-DA5 | A | R | C |
| 3 | M1 | A | R | C |
| 4 | BD1 | A | R | C |

## Dependencias y riesgos

Los riesgos identificados con la ejecución de las pruebas son:

* Dificultades con la disponibilidad del servidor de pruebas: Se debe tener una base de datos de respaldo sobre la cual trabajar
* Pruebas que dependen de factores externos como la conexión a internet, la disponibilidad de recursos por parte del usuario: La no disponibilidad de recursos externos puede ser reemplazado con otro similar
* La disponibilidad de los integrantes del plan de pruebas. La falta de algunos de ellos ocasionará retrasos con la finalización a tiempo: La distribución adecuada de tareas determinará la funcionalidad del plan de pruebas.
* El proyecto no depende de otros proyectos, pero está aunado a la migración de los datos del usuario, los datos erróneos generarán errores en las pruebas: La verificación de los datos antes de realizar una migración en cadena minimizará errores
* El no cumplimiento de los tiempos estipulados: Los tiempos tienen un tiempo de tolerancia a novedades que generen retrasos, pero no incumplimiento
* Fallas en el software de automatización de pruebas: Se debe contar con varias herramientas de automatización, en caso que se difícil de realizar se debe tener un soporte del software de automatización.

.

# Referencias

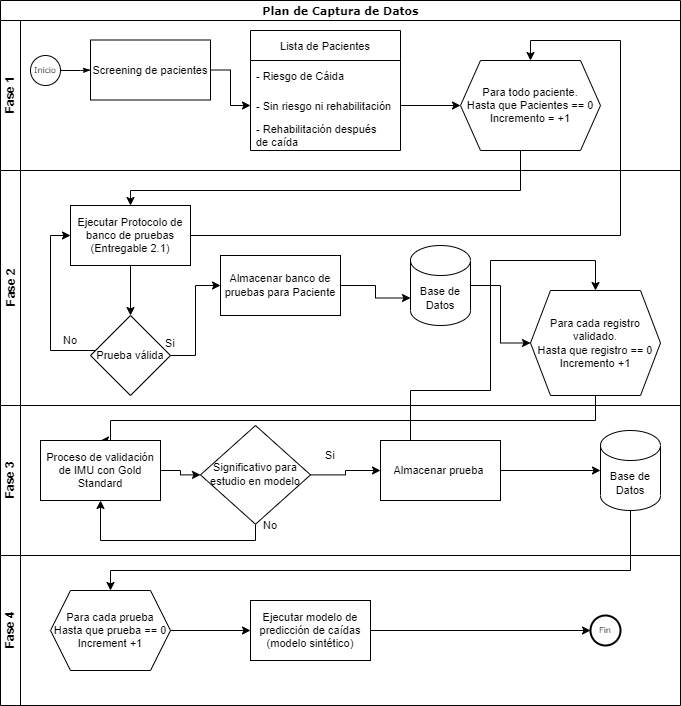


Figura 1: Plan de Captura de Datos