

Contenido

[Histórico de Revisiones 6](#_heading=h.1fob9te)

[Información del Proyecto 6](#_heading=h.3znysh7)

[Propósito del plan de pruebas 7](#_heading=h.tyjcwt)

[Alcance de las pruebas 7](#_heading=h.3dy6vkm)

[Definición de roles y responsabilidades 8](#_heading=h.1t3h5sf)

[Tipos de pruebas a realizar 8](#_heading=h.4d34og8)

[Estrategia y técnicas de pruebas a aplicar 9](#_heading=h.26in1rg)

[Criterios de Inicio 10](#_heading=h.4i7ojhp)

[Criterios de Aceptación o Rechazo 10](#_heading=h.2xcytpi)

[Criterios de Suspensión 11](#_heading=h.ihv636)

[Criterios de Reanudación 11](#_heading=h.3fwokq0)

[Definición del proceso de testing 12](#_heading=h.19c6y18)

[Elementos de Pruebas 12](#_heading=h.37m2jsg)

[Requerimientos de Software 12](#_heading=h.1mrcu09)

[Módulos para probar: 14](#_heading=h.46r0co2)

[Herramientas involucradas 15](#_heading=h.3q5sasy)

[Calendarización de las actividades de pruebas 16](#_heading=h.25b2l0r)

[Resumen de riesgos aplicados a las pruebas 16](#_heading=h.kgcv8k)

[Clasificación de los defectos 18](#_heading=h.34g0dwd)

# 

# 

# Histórico de Revisiones

| **Versión** | **Fecha** | **Descripción/cambio** | **Autor** |
| --- | --- | --- | --- |
| 1.0 | 09-08-2024 | Creación documento | Jose Muñoz |
| 1.1 | 16-09-2024 | Revisión documento | Jose Muñoz |
|  |  |  |  |

# Información del Proyecto

| **Organización** | Duoc UC. Escuela de Informática y Telecomunicaciones |
| --- | --- |
| **Sección** | 009D |
| **Proyecto (Nombre)** | Don Terapia |
| **Fecha de Inicio** | 09-08-2024 |
| **Fecha de Término** | 16-11-2024 |
| **Patrocinador principal** | Duoc UC |
| **Docente** | Cindy Contador |

# Propósito del plan de pruebas

El Propósito del plan de pruebas es trazar los lineamientos de todos los tipos de pruebas que se van a presentar en el proyecto para desear la mejor calidad posible al programa de Feria Virtual.

El objetivo general de este plan es establecer la cronología y las condiciones para las pruebas a desarrollar, para así obtener un sistema informático que pueda cumplir con las necesidades del cliente y tener un rendimiento y funcionamiento completamente funcional para su puesta en marcha.

# Alcance de las pruebas

El alcance de las pruebas que realizaremos consistirá en la prueba y evaluación de los distintos módulos de nuestro programa, siendo los principales; la capa visual: la interfaz que se visualizará al utilizar la solución; la capa lógica: la programación de los distintos módulos de la solución y toda su lógica; y la capa de negocios: las funcionalidades especificadas por el cliente y evaluadas por el equipo, las cuales son necesarias para el correcto funcionamiento del modelo de negocio al cual estaremos dando solución.

Como equipo de desarrollo hemos decidido utilizar el lenguaje Python y su entorno de trabajo Python. Las pruebas serán realizadas utilizando las herramientas que nos otorga esta IDE y además utilizaremos SonarQube, una plataforma para evaluar código, la cual nos servirá para analizar los errores de programación y darles seguimiento y solución.

Se aprobará el proyecto con la totalidad de pruebas ejecutadas, pero con un 70% de aceptación, ósea, el 70% de las pruebas deben de ser exitosas y sin errores. El restante 30% pueden ser de medio o bajo error, pero no graves o fallidas.

# Definición de roles y responsabilidades

| Nombre | Responsabilidad |
| --- | --- |
| Felipe Vargas | Jefe de Proyecto, Responsable de evaluar las condiciones de término para el proceso de pruebas. |
| Jose Muñoz | Tester, Responsable de generación del plan de pruebas. |
| Manuel Cartes | Tester, Responsable de la ejecución e implementación de las pruebas |
| Joaquin Brito | Analista Funcional, Responsable de la resolución de incidentes. |

# Tipos de pruebas a realizar

El líder del proyecto se encargará de aprobar las pruebas propuestas por el equipo y tendrá la responsabilidad de ejecutar las pruebas planificadas previamente por los mismos integrantes del equipo de trabajo.

Se llevarán a cabo pruebas funcionales dadas por los requerimientos, además de pruebas de integración. Con estas pruebas se planifica controlar las funcionalidades de la solución abarcando cada módulo con estricto rigor, además de la integración correcta de los distintos componentes de datos y de la integridad de los datos que son procesados y almacenados en la base de datos y el programa. También se quiere probar la completitud del sistema a nivel de usabilidad, para dar sentido a que el usuario no tenga problemas a la hora de utilizarlo.

En síntesis, los tipos de prueba a realizar son:

Pruebas Funcionales

Pruebas de Integración

Pruebas de Usabilidad

# Estrategia y técnicas de pruebas a aplicar

La estrategia de pruebas consistirá en etapas dedicadas a las pruebas funcionales y no funcionales, de Integración y de Usabilidad.

Para poder llevar a cabo las estrategias de prueba, se requiere lo siguiente:

* Planificación del Plan de Pruebas
* Diseño de las Pruebas
* Implementación de las Pruebas
* Ejecución de las Pruebas
* Evaluación de las Pruebas

Ya descritos los pasos previos, se detalla la estrategia de las pruebas en general:

1. Funcionalidades de los módulos funcionales del proyecto para ver si están operativos.
2. Funcionalidades Integradas de los módulos funcionales del proyecto, donde se aprobarán o se rechazará para ver si cumplen con la integración de todos los módulos.
3. Funcionalidades de Usabilidad, donde se pondrá a prueba el sistema para visualizar la capacidad de adaptabilidad del usuario.

# Criterios de Inicio

Para dar inicio a la ejecución de pruebas, se debe de cumplir con lo siguiente:

* Aceptación del plan de pruebas, revisión y aceptación del documento que contiene los casos de pruebas para la certificación del proyecto.
* Aceptación de paquetes desarrollo.
* Aceptación de ambiente y que se cumpla con las condiciones de aceptación.

# Criterios de Aceptación o Rechazo

**Errores Graves:** Información crítica presentada erróneamente, información mal registrada en la base de datos, caídas de programas, incumplimiento de objetivos en funciones principales, entre otros.

**Errores Medios** (Comunes): Errores en presentación de datos, incumplimiento de objetivos en funciones secundarias, caídas de servicio o comunicación de consulta, entre otros.

**Errores Leves:** Errores en presentación de datos secundarios, no adecuación a estándares, comportamientos correctos pero diferentes en situaciones similares, dificultades de operación, entre otros.

Para determinar si se ha completado el Plan de Pruebas, se debe de haber realizado y completado los siguientes puntos:

* Completar 100% de pruebas.
* El 70% de las pruebas Exitosos
* Casos de prueba con defectos están siendo trabajados
* Software funcional
* Porcentaje de defectos corregidos

# Criterios de Suspensión

Para determinar si el Plan de Pruebas debe de ser suspendido deben de pasar los siguientes puntos:

* Problemas externos (ejemplo: Ambientales, técnicos, salud)
* Algunos defectos vinculados con los módulos de prueba impiden la ejecución de más módulos de prueba
* Acotación del tiempo: Se ha alargado demasiado el tiempo y no habrá tiempo para realizar la ejecución de las pruebas restantes.
* No abastecer la totalidad de las pruebas realizadas en los casos de Prueba

# Criterios de Reanudación

Para la reanudación de las pruebas, se determinará bajo los siguientes puntos:

* Haber corregido los defectos que afectan los demás casos de prueba
* Arreglo de software y Hardware, que impedían la continuación de las pruebas

# Definición del proceso de testing

El proceso de testing será llevado a cabo por la IDE Python y el motor de base de datos de Oracle o MySQL. En este proceso se llevará a cabo la realización de los distintos casos de uso del proyecto para hacer el seguimiento del correcto comportamiento del programa.

Hemos planificado tres ciclos de pruebas a ejecutar, los cuales probarán el código construido hasta ese momento y harán seguimiento de la correcta integridad de los datos desde un comienzo.

* El primer ciclo de pruebas se realizará cerca del final de la segunda iteración.
* El segundo ciclo de pruebas se realizará en la mitad de la tercera iteración.
* El tercer ciclo de pruebas se realizará cerca del final de la tercera iteración y debe cubrir todas las pruebas que hayan fallado en el segundo ciclo de pruebas.

# Elementos de Pruebas

# Requerimientos de Software

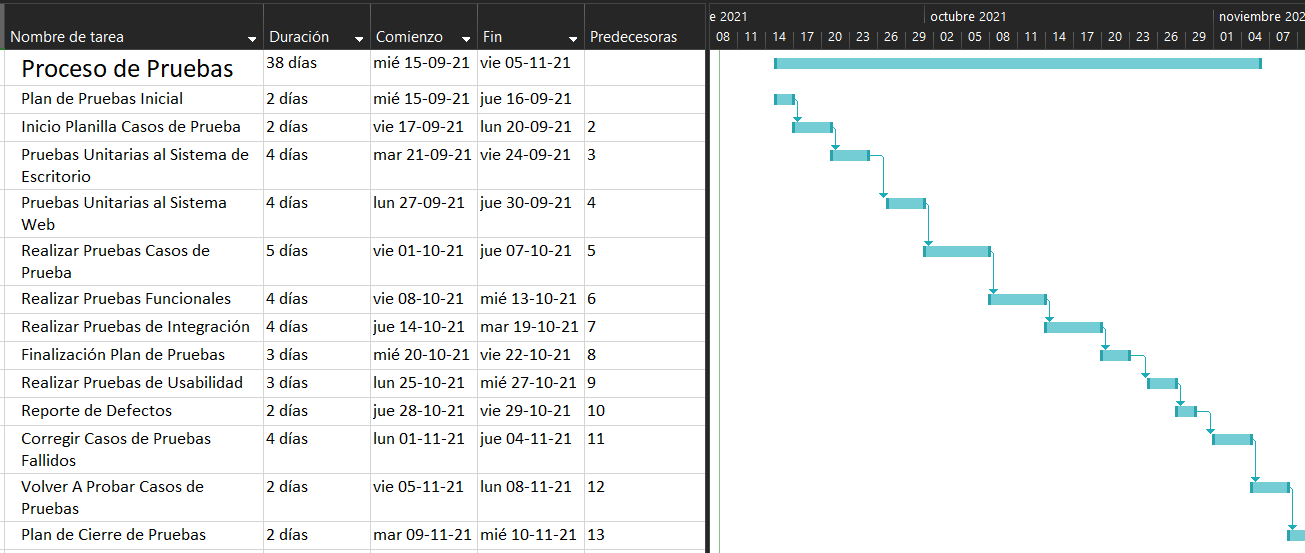
| Código | Requerimiento |
| --- | --- |
| RF-01 | El sistema debe permitir el registro y autenticación de tres tipos de usuarios: pacientes (niños), tutores (padres/madres) y terapeutas |
| RF-02 | Los terapeutas deben poder asignar actividades pre definidas por el a cada paciente. |
| RF-03 | El sistema debe registrar y mostrar el progreso de los pacientes en base a diferentes fuentes (actividades y notas terapéuticas). |
| RF-04 | El sistema debe permitir que el terapeuta evalúe pruebas y evaluaciones terapéuticas de cada usuario paciente. |
| RF-05 | El sistema debe enviar recordatorios automáticos sobre tareas pendientes hacia los usuarios tutores y terapeutas. |
| RF-06 | El sistema debe analizar la conducta y generar recomendaciones en base las evaluaciones hechas por el paciente y las notas dadas por el terapeuta. |
| RF-07 | El sistema debe de tener un sistema de recompensas para los pacientes que realicen a tiempo sus actividades. |
| RF-08 | El sistema debe permitir al usuario terapeuta subir sus notas terapéuticas de cada paciente de una forma sencilla. |
| RF-09 | El sistema debe permitir al usuario terapeuta crear notas terapéuticas de cada paciente. |
| RF-10 | La aplicación debe tener una interfaz intuitiva y accesible, adaptable a diferentes dispositivos. |
| RF-11 | Los terapeutas deben poder interactuar dentro de los perfiles de cada paciente. |
| RF-12 | Los terapeutas deben poder gestionar las actividades que están dentro del sistema. |
| RF-13 | El sistema debe integrar un sistema de inteligencia artificial para simplificar las funcionalidades. |
| RF-14 | El sistema debe permitir el registro de nuevos pacientes por parte del terapeuta. |
| RF-15 | El terapeuta podrá definir el tiempo de cada actividad. |

# 

# Herramientas involucradas

| **Funcionalidad** | **Herramienta** |  | **Versión** |
| --- | --- | --- | --- |
| Testing Aplicación escritorio (Pruebas unitarias) | Apache NetBeans IDE |  | 12.3 |
| Testing para carga de datos y rendimiento | Oracle MySQL Workbench |  | 8.0 CE |
| Testing para Testeo de pruebas |  |  | JDE 8.0 |
| Plataforma para evaluar código fuente. |  |  | 8.2.26 |
| Sistema Web que evalúa y pone a prueba una aplicación Web | Test Link | - | 1.9.20 |

# Calendarización de las actividades de pruebas



# Resumen de riesgos aplicados a las pruebas

| **N.º Riesgo** | **Descripción** | **Gravedad** | **Acción** |
| --- | --- | --- | --- |
| R001 | Documentación incompleta o errónea | Mayor | Comprobar documentación antes de seguir a la siguiente fase |
| R002 | Fallas en la comunicación | Menor | Establecer un plan de comunicación eficiente |
| R003 | Softwares incompletos | Crítica | Dedicar mayor tiempo al desarrollo del software, para que esté 100% funcional |
| R004 | Hardware inestable | Menor | Revisión y mantención constante al hardware o sustituir |
| R005 | No seguir planificación | Mayor | Organizar el proyecto para que sea definida una planificación nueva en caso de ser necesario |
| R006 | Poca eficiencia del equipo de Testing | Crítica | Asegurar la experiencia y prestigio del equipo de Testing o bien, contratar a otro personal del área del Testing. |
| R007 | Herramientas de Testing mal optimizadas | Crítica | Adecuarse a las necesidades del Testing y si es necesario cambiar las herramientas. |
| R008 | Restricciones de Tiempo | Mayor | Priorizar las acciones que son más puntuales para el proyecto |
| R009 | Poca disponibilidad de Recursos | Menor | Asegurarse con los recursos que hay disponibles y ver con el tiempo si se pueden añadir más |
| R010 | No funcionamiento en los botones de la aplicación | Crítica | Contratar a personal externo que ayude con él funcionamiento de la aplicación |
| R011 | Los casos de Prueba son insuficientes | Crítica | Asignar otro personal para que evalué más casos de prueba |
| R012 | No se siguen al pie de la letra las reglas de negocio | Crítico | Realizar una reunión de contingencia para evaluar al personal y asignar nuevos roles. |

# Clasificación de los defectos

| **Nivel de Severidad** | **Descripción** |
| --- | --- |
| **Crítico** | Afecta de manera voraz al sistema o al proyecto |
| **Mayor** | Afecta de manera grave el proyecto o sistema |
| **Menor** | Afecta de manera rigurosa el proyecto o sistema |