

**SERVICIO NACIONAL DE APRENDIZAJE**

**PRUEBAS DE FUNCIONALIDAD UNITARIAS PRODUCTO  
SOFTWARE**

**EJERCICIO DE PLANEACIÓN PARA LA EJECUCIÓN DE PRUEBAS MANUALES DE FUNCIONALIDAD DEL  
PROYECTO SOFTWARE CANCHEROS**

**ACTIVIDAD TRIMESTRE V**

**Autores:**

**MIGUEL ANDRES PARRA SALGUERO**

**JOSTIN TAFUR**

**BRIT NICOLAS GIL**

**Bogotá D.C**

**5 septiembre-2024**

## INDICE

|  |    |
|--|----|
| 1. Introducción .....                        | 3  |
| 2. Objetivo .....                            | 4  |
| 3. Alcance .....                             | 5  |
| 4. Metodología y Métodos .....               | 6  |
| 5. Recursos.....                             | 7  |
| 6. Funcionalidad .....                       | 8  |
| 7. Cronograma.....                           | 9  |
| 8. Revisión y Ajustes .....                  | 10 |
| 9. Interfaz de usuario .....                 | 10 |
| 10. Rendimiento .....                        | 11 |
| 11. Planilla y descripción del proceso ..... | 12 |
| 12. Conclusiones .....                       | 13 |

## 1. INTRODUCCION

El proyecto **CANCHEROS** es un sistema de reservas en línea diseñado para facilitar la reserva de dos canchas de fútbol delegadas al salón comunal de bonanza que participa como intermediario para el **IDRD**. Los usuarios deben registrarse en la plataforma proporcionando información personal, y serie de datos adicionales, para luego iniciar sesión y poder realizar o cancelar reservas en fechas y horas específicas de manera gratuita.

**El propósito de este documento es detallar la planificación de las pruebas unitarias funcionales para asegurar el correcto funcionamiento de cada una de las funcionalidades del sistema, utilizando el método de caja negra. Estas pruebas se realizarán de manera manual y se centrarán en la validación de la operabilidad desde la perspectiva del usuario final.**

## 2. OBJETIVO

El objetivo principal de las **pruebas unitarias funcionales** es verificar que todas las funcionalidades del sistema **CANCHEROS** (registro de usuario, inicio de sesión, creación y cancelación de reservas) funcionen correctamente y sin errores, proporcionando una experiencia de usuario fluida y fiable. Y de la misma manera que se capten de manera eficiente y logica dentro de la base de datos para la facilitación del análisis de dicha información

Las pruebas unitarias funcionales se realizarán a través del método de caja negra, lo que implica que el equipo de prueba no tendrá conocimiento del código interno, sino que se centrará en la salida esperada en respuesta a entradas específicas. Pruebas funcionales: Las pruebas funcionales tienen como objetivo validar que nuestro software haga lo que se supone que debe hacer. Por ejemplo, van a probar el mecanismo de autenticación de logueo para verificar que los usuarios legítimos puedan autenticarse exitosamente mientras se rechazan los intentos de inicio de sesión no válidos. incluyendo comprobaciones de cordura, pruebas de integración y pruebas de sistema.

### 3. ALCANCE

El alcance de las pruebas funcionales cubrirá las siguientes áreas clave del sistema CANCHEROS:

- **Registro de usuario:** Verificación del correcto registro, validación de campos (nombres, apellidos, teléfono, correo, tipo y número de documento, y contraseña).
- **Inicio de sesión:** Confirmación de que los usuarios pueden iniciar sesión de manera exitosa tras el registro.
- **Realización de reserva:** Comprobación de la funcionalidad que permite seleccionar una cancha, fecha, hora, deporte, cancha, cantidad de jugadores petos, arbitraje, color de uniforme, para hacer una reserva.
- **Cancelación de reserva:** Verificación de la opción para cancelar una reserva previamente realizada.

#### 4. METODOLOGIA Y METODOS DE ANALISIS

Se utilizará el método de **caja negra** para llevar a cabo las pruebas. Este método consiste en proporcionar entradas a las funcionalidades y observar las salidas esperadas, sin conocimiento del código fuente subyacente. las pruebas funcionales vamos a probar en el mecanismo de autenticación de nuestra aplicación va a verificar que los usuarios legítimos puedan autenticarse exitosamente mientras se rechazan los intentos de inicio de sesión no válidos. Los tipos de pruebas funcionales que incluiremos serán comprobaciones de cordura, pruebas de integración y pruebas de sistema, y recordemos que estos procesos de validación lo haremos de manera manual haciendo uso directo del sistema

Pasos de las pruebas funcionales manuales:

- **Identificación de las entradas:** Cada campo o acción a evaluar se probará con datos válidos e inválidos.
- **Ejecución de pruebas:** Se realizarán manualmente las pruebas siguiendo los casos de prueba definidos.
- **Validación de resultados:** Se observará si las salidas coinciden con el comportamiento esperado.
- **Documentación de resultados:** Cada prueba se documentará en términos de éxito o fracaso.
- **Reporte de errores:** si hay fallos, se generarán reportes detallados para corregirlos.

## 5. RECURSOS

Los recursos necesarios para llevar a cabo las pruebas funcionales son:

- ◆ Equipos: Computadoras con acceso a internet.
- ◆ Software: Navegador web actualizado y acceso a la plataforma CANCHEROS.
- ◆ Internet Acceso a a una fuente de navegabilidad por internet sea local como wifi o conexión a cable de red y/o datos móviles

Personal:

- **1 analista de pruebas.** (Jostin Tafur)
- **1 revisor** para soporte y corrección de errores. (Nicolas Gil)
- **1 jefe de proyecto** para la supervisión. (Miguel Parra)

Herramientas de documentación: Herramientas de seguimiento de pruebas como Excel o Google Sheets.

## 6. FUNCIONALIDAD

La funcionalidad de la prueba manual se enfocará en validar, práctica y sin acceso al código fuente, que las interacciones entre el usuario y el sistema CANCHEROS operen según los requerimientos establecidos. Utilizando el método de caja negra, se simularán diferentes escenarios para asegurarse de que las funcionalidades respondan de manera adecuada a entradas válidas e inválidas, evaluando el comportamiento del sistema sin necesidad de revisar su implementación interna.

En este caso haremos uso de datos ficticios ingresando información correcta e incorrecto haciendo prueba de las diferentes restricciones en cada campo para evaluar que efectivamente estos limitantes de acceso se están ejecutando de manera correcta y que la información que se está migrando tiene forma, fondo y lógica para el análisis y uso requerido que se va a entregar al administrador del software y para facilitar efectivamente que el registro , agendamiento y separación de las canchas se formalice de manera más eficiente al manejo anterior que se le daba



7. CRONOGRAMA

El cronograma para la ejecución de las pruebas unitarias funcionales se detalla en un periodo de 6 semanas.

| Semana | Actividad                                    | Fecha          |
|--------|--|----------------|
| 1      | Panificación y diseño de los casos de prueba | 06 Julio 2024  |
| 2      | Pruebas de registro de usuario               | 20 Julio 2024  |
| 3      | Pruebas de inicio de sesión                  | 31 Julio 2024  |
| 4      | Pruebas de reserva de canchas                | 10 Agosto 2024 |
| 6      | Pruebas de cancelación de reservas           | 18 agosto 2024 |
| 6      | Reporte de errores, revisión y ajustes       | 27 Agosto 2024 |

## 8. REVISION Y AJUSTES

Durante la semana 6, se realizará una revisión exhaustiva de los resultados obtenidos en las pruebas unitarias. En esta fase, se identificarán los errores encontrados y se coordinará con el equipo de desarrollo para la implementación de las correcciones.

Una vez realizados los ajustes, se repetirán las pruebas en las funcionalidades corregidas para verificar su correcto funcionamiento. De igual manera se realizarán las modificaciones que interpretemos necesarias para que el uso sea ágil y de fácil interpretación para cualquier tipo de publico

## 9. INTERFAZ DE USUARIO

El interfaz de usuario se refiere a las formas en que el usuario o persona que interactúa con el software, y en donde la aplicación permite completar una serie de tareas, incluyen los menús con los que se trabaja, los botones específicos para cada función presente en el sistema y la marca de todo el software. De igual manera que el diseño tenga UX, para que la experiencia del usuario sea agradable a la vista y a la manipulación de navegabilidad

## 10. RENDIMIENTO

Además de funcionar con normalidad y tener buen aspecto, el rendimiento de una aplicación es esencial para agradar a los clientes. Por ellos realizaremos **pruebas de rendimiento en el cual se medirán los tiempos promedio en que una persona de conocimiento nulo de tecnología entienda el proceso y lo compararemos con el tiempo promedio que una persona de manejo intermedio alto puede tomar haciendo el registro** y separación de la cancha, esto busca también medir el impacto en tiempos de agendamiento de las canchas que fue uno de los malestares que presentaba la administración por la cantidad de solicitudes que tenía al día para el agendamiento y los pocos recursos digitales que utilizaban

El rendimiento se refiere a varios factores, como la velocidad e intervalo de tiempos de función de la aplicación a la hora de responder a las entradas del usuario y los recursos que consume en un dispositivo determinado.

## 11. PLANTILLA Y DESCRIPCCION DEL PROCESO

Daremos uso de la plantilla a continuación presentada como modelo para la realización de las pruebas de funcionalidad de cada uno del espacio de ingreso y recepción de datos del software, de esta manera tener un orden en el registro de la información de la prueba y llevar un control de la efectividad de dicha actividad. Y así mismo evaluar el resultado haciendo un promedio del porcentaje de efectividad de las pruebas versus el 100 % esperado.

### Casos de Prueba

### FASE - 1

|                                  |  |  |   |
|----------------------------------|--|--|---|
| <b>Número de Prueba</b>          |  | MDB-1 (FASE 1)   |   |
| <b>Título del caso de prueba</b> |  | Cambio de estado <i>inactivo</i> a <i>deshabilitado</i> .  |   |
| <b>Resumen de la prueba</b>      |  | El objetivo de la prueba es comprobar que se realiza el cambio de estado correctamente y para ello se realizarán las acciones oportunas para comprobar el correcto funcionamiento del TPV cuando se encuentra en estado <i>inactivo</i> y queremos pasar a estado <i>deshabilitado</i> . |   |
| <b>Precondiciones</b>            |  | El TPV tiene que estar en estado <i>inactivo</i> .   |   |
| <b>Pasos</b>                     |  | <b>Resultados</b>  |   |
| <b>1</b>                         | El VMC envía el comando 'SETUP' subcomando 'Config Data' al TPV. | <b>1</b>   | El TPV tiene que haber entrado en estado <i>deshabilitado</i> . |

## 12. CONCLUSIONES

Las pruebas unitarias funcionales con el método de caja negra proporcionarán una visión clara sobre la efectividad del sistema CANCHEROS desde la perspectiva del usuario. Esta estrategia garantizará que los usuarios puedan registrarse, iniciar sesión y realizar o cancelar reservas de manera sencilla y sin errores.

Con un enfoque basado en la validación de funcionalidades, se asegurará que el sistema cumple con los requerimientos iniciales y proporciona una experiencia de usuario eficiente, lo que garantizará la calidad final del software.

Las pruebas también nos permitirán darle una mejor visión al proceso de uso del software y poder modificar errores de diseño y no solo funcionalidad que se vayan identificando dentro del proceso