



**Universidad Libre de Colombia**  
**Facultad de Ingeniería**  
**Programa de Ingeniería de Sistemas**

## **Pruebas sobre sistema de información**

**Curso:** Proyecto de Asignatura

**Fecha:** 6 de mayo de 2025

**Ciudad:** Bogotá D.C

### **Integrantes:**

Favio Bernal

Carlos Lamus

Juan Roncancio

Kevin Collazos

## Índice

1. Portada
2. Resumen del Avance
3. Introducción
4. Actividades a Realizar
  - 4.1. Listado de requerimientos y/o historias de usuario (JIRA)
  - 4.2. Documentación de los sprints con metodología SCRUM
    - 4.2.1. Daily
    - 4.2.2. Planificación (Planning)
    - 4.2.3. Revisión (Review)
    - 4.2.4. Retrospectiva
  - 4.3. Burndown chart y análisis por sprint
  - 4.4. Evidencias de las ceremonias SCRUM
  - 4.5. Escenarios de casos de prueba
  - 4.6. Casos de prueba
  - 4.7. Pruebas de aceptación
  - 4.8. Scripts con Selenium
  - 4.9. Descripción y análisis de las pruebas funcionales
  - 4.10. Tablas e imágenes con numeración y descripción
5. Próximas Etapas
  - 5.1. Pruebas de carga
  - 5.2. Preparación del informe final
  - 5.3. Defensa del proyecto

## 2. Resumen del Avance

Este documento presenta el avance del proyecto asociado a la asignatura Ingeniería de Software 3.

Se usarán algoritmos de pruebas basadas en Selenium para personalizar las sugerencias de libros.

Los resultados obtenidos muestran un avance significativo hacia los objetivos planteados.

## 3. Introducción

El presente proyecto tiene como objetivo desarrollar una documentación y scripts de un Conjunto de pruebas funcionales. Este avance documenta el trabajo realizado durante el final del semestre.

## 4. Actividades a Realizar

### 4.1. Listado de requerimientos y/o historias de usuario (JIRA)

▼

Tablero Sprint 1

6 may – 14 jun (15 actividades)

0

14

27

Completar sprint

...

<input checked="" type="checkbox"/>	<del>TUC-34</del>	Registro de Usuario	FINALIZADA ▼	3	CL
<input checked="" type="checkbox"/>	<del>TUC-35</del>	Validación de Campos Vacíos en Registro	FINALIZADA ▼	2	FG
<input checked="" type="checkbox"/>	<del>TUC-36</del>	Redirección Post-Registro	FINALIZADA ▼	1	K
<input checked="" type="checkbox"/>	<del>TUC-37</del>	Inicio de Sesión	FINALIZADA ▼	4	JC
<input checked="" type="checkbox"/>	<del>TUC-38</del>	Error de Inicio de Sesión	FINALIZADA ▼	2	CL
<input checked="" type="checkbox"/>	<del>TUC-39</del>	Redirección tras Inicio de Sesión	FINALIZADA ▼	1	FG
<input checked="" type="checkbox"/>	<del>TUC-40</del>	Redirección si no se ha Iniciado Sesión	FINALIZADA ▼	2	K
<input checked="" type="checkbox"/>	<del>TUC-41</del>	Mostrar Reservas del Usuario	FINALIZADA ▼	3	JC
<input checked="" type="checkbox"/>	<del>TUC-42</del>	Mensaje sin Reservas	FINALIZADA ▼	1	CL
<input checked="" type="checkbox"/>	<del>TUC-43</del>	Cerrar Sesión	FINALIZADA ▼	4	FG
<input checked="" type="checkbox"/>	TUC-44	Realización de Pruebas (Pruebas Funcionales)	EN CURSO ▼	4	K
<input checked="" type="checkbox"/>	TUC-45	Realización de Pruebas de Aceptación	EN CURSO ▼	4	JC
<input checked="" type="checkbox"/>	TUC-46	Documentación del Sistema	EN CURSO ▼	2	CL
<input checked="" type="checkbox"/>	<del>TUC-47</del>	Conexión a la Base de Datos	FINALIZADA ▼	4	FG
<input checked="" type="checkbox"/>	TUC-48	Finalización de la Interfaz de Usuario	EN CURSO ▼	4	K

**Figura 1. Tablero de tareas del Sprint 1**

Esta figura muestra el tablero de tareas correspondiente al Sprint 1 del proyecto "TU CANCHA", con un rango de fechas del 6 de mayo al 14 de junio. En ella se presentan las tareas asignadas a diferentes miembros del equipo, su estado actual (finalizadas o en curso) y el progreso alcanzado hasta el momento. Cada tarea está identificada con su código (TUC-34, TUC-35, etc.), y se incluyen las pruebas funcionales y de aceptación, así como las tareas relacionadas con la interfaz de usuario y la base de datos.

ID RF	Requerimiento Funcional	Caso de Prueba Asociado
RF01	El usuario debe poder registrarse ingresando nombre, correo y contraseña.	CP01 – Registro de usuario exitoso
RF02	El sistema debe validar que todos los campos del registro estén completos.	CP02 – Registro fallido por campos vacíos
RF03	Tras un registro exitoso, debe mostrarse una alerta y redirigir al login.	CP03 – Redirección post-registro
RF04	El usuario debe poder iniciar sesión con email y contraseña válidos.	CP03 – (mismo que redirección)
RF05	Si las credenciales son incorrectas, debe mostrarse un mensaje de error.	CP04 – Inicio de sesión incorrecto
RF06	Al iniciar sesión correctamente, debe redirigir a `reservas.html`.	CP03 – Redirección validada
RF07	Si el usuario no ha iniciado sesión, debe redirigir al login al intentar acceder a reservas.	CP05 – Acceso no autorizado a reservas
RF08	Si hay reservas guardadas, deben mostrarse en una lista.	CP06 – Mostrar reservas del usuario

RF09	Si no hay reservas, debe mostrarse un mensaje indicando que no existen.	CP07 – Mensaje sin reservas
RF10	El botón "Cerrar sesión" debe eliminar la sesión y redirigir al login.	CP08 – Cerrar sesión

**Tabla 1. Trazabilidad de requerimientos funcionales y casos de prueba**

*Esta tabla muestra la relación entre los requerimientos funcionales del sistema y los casos de prueba diseñados para validarlos. Cada requerimiento tiene asociado al menos un caso de prueba que permite verificar su cumplimiento dentro del desarrollo del proyecto "TU CANCHA".*

## **4.2. Documentación de los sprints con metodología SCRUM**

### **4.2.1. Daily**

Durante el desarrollo del proyecto, el equipo no realizó reuniones diarias formales, ya que todos los integrantes se ven regularmente en la universidad. El seguimiento del trabajo se realizó los días lunes y miércoles, durante el horario de clase de la asignatura, donde se discutían los avances, dificultades y próximos pasos. En esos espacios también se acordaban las tareas a realizar dentro de cada sprint, el cual se definió con una duración de 15 días.

### **4.2.2. Planificación (Planning)**

El proceso de planificación se realizó en tres sprints, cada uno con sus respectivas tareas y objetivos.

- **Primer Sprint (18 de marzo):** Se centró en la creación y asignación de los requerimientos funcionales y las historias de usuario en Jira. Cada historia de usuario fue acompañada de los casos de prueba correspondientes. Este sprint sirvió como base para las funcionalidades principales del sistema, asegurando que se cumplieran los requisitos iniciales.
- **Segundo Sprint (2 de abril):** En esta fase, se revisaron las tareas del sprint anterior para garantizar que todas estuvieran completas. Además, se asignaron tareas para

terminar las funcionalidades del software y se comenzó con las pruebas funcionales en Selenium, asegurando la correcta integración del sistema con las pruebas automatizadas.

- Tercer Sprint (17 de abril): En este sprint, se finalizó el desarrollo del software y las pruebas funcionales. Se asignaron tareas de pruebas de aceptación y de mejora de funciones, con el objetivo de completar y validar todas las características del sistema antes de la entrega final.

#### **4.2.3. Revisión (Review)**

La revisión del proyecto se llevó a cabo al final de cada sprint, donde se evaluaron las tareas completadas, los objetivos alcanzados y las tareas pendientes para asegurar el progreso continuo del proyecto.

- Primer Sprint (18 de marzo): Durante la revisión, se verificó que se completaron las tareas iniciales de la creación y asignación de los requerimientos funcionales y las historias de usuario en Jira, junto con sus respectivas pruebas. Sin embargo, quedaron pendientes dos requerimientos que no fueron completamente definidos.
- Segundo Sprint (2 de abril): En esta fase, se revisaron las tareas del sprint anterior para asegurar que todo estuviera completo. Se completaron las tareas de cerrar sesión y se finalizaron las pruebas funcionales en Selenium. También se revisaron los requerimientos pendientes del sprint anterior, y se confirmaron que se habían resuelto. Aunque muchas tareas fueron completadas, algunas mejoras en la interfaz de usuario aún estaban por terminar.
- Tercer Sprint (17 de abril): En la revisión de este sprint, se verificó que todas las pruebas funcionales se completaron, aunque quedaron pendientes dos pruebas funcionales que no se pudieron finalizar. Además, se asignaron tareas para realizar las pruebas de aceptación y para finalizar las funcionalidades que aún no estaban completamente implementadas.

#### **4.2.4. Retrospectiva**

La retrospectiva del proyecto se realizó al finalizar cada sprint, con el objetivo de identificar lo que funcionó bien, los aspectos que se deben mejorar y las acciones a tomar para los siguientes sprints.

- Primer Sprint (18 de marzo):

Lo que funcionó bien: La asignación de requerimientos funcionales e historias de usuario en Jira fue eficiente y permitió tener claridad sobre las tareas a desarrollar.

Áreas de mejora: Quedaron pendientes dos requerimientos que no fueron completamente definidos, lo que retrasó la ejecución de tareas relacionadas.

Acciones: Para el próximo sprint, se definió una reunión adicional para asegurar que todos los requerimientos fueran claros y completos antes de empezar.

- Segundo Sprint (2 de abril):

Lo que funcionó bien: Se completaron las tareas de cerrar sesión y las pruebas funcionales en Selenium. Además, se revisaron y completaron los requerimientos del sprint anterior.

Áreas de mejora: Aunque se completaron las tareas asignadas, se notó que algunas mejoras de interfaz de usuario quedaron fuera de la revisión, lo que afectó la experiencia general del sistema.

Acciones: Se acordó asignar un enfoque más claro a las mejoras de la interfaz, en el próximo sprint y no dejar tareas fuera del alcance.

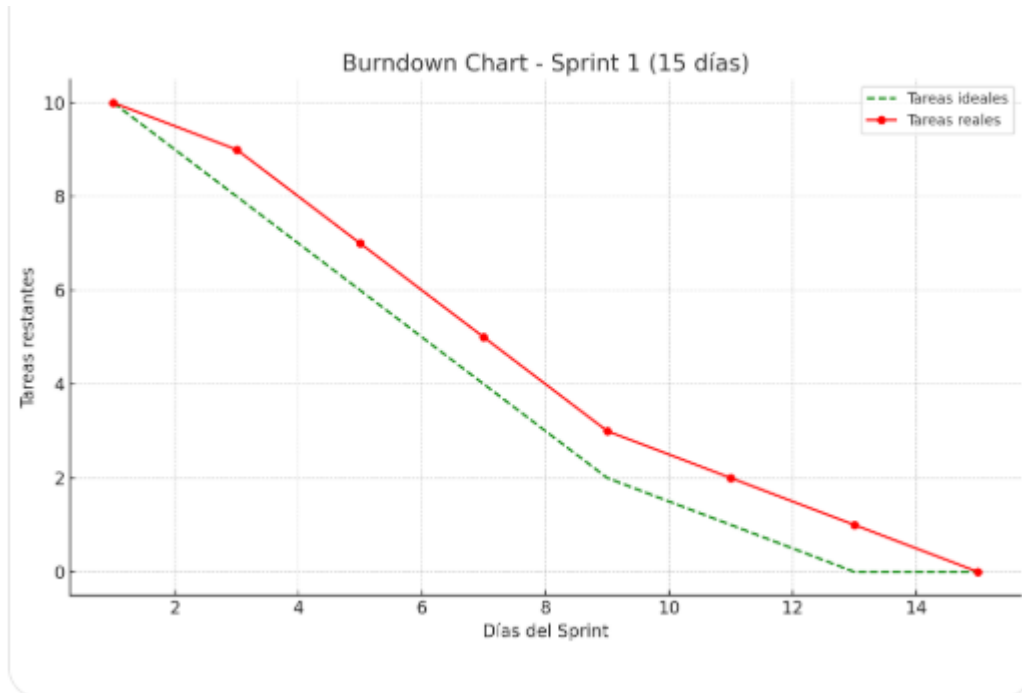
- Tercer Sprint (17 de abril):

Lo que funcionó bien: Se completaron las pruebas funcionales en su mayoría y se avanzó en las pruebas de aceptación.

Áreas de mejora: Las dos pruebas funcionales pendientes indicaron que algunos aspectos del software necesitaban más tiempo para pulirse.

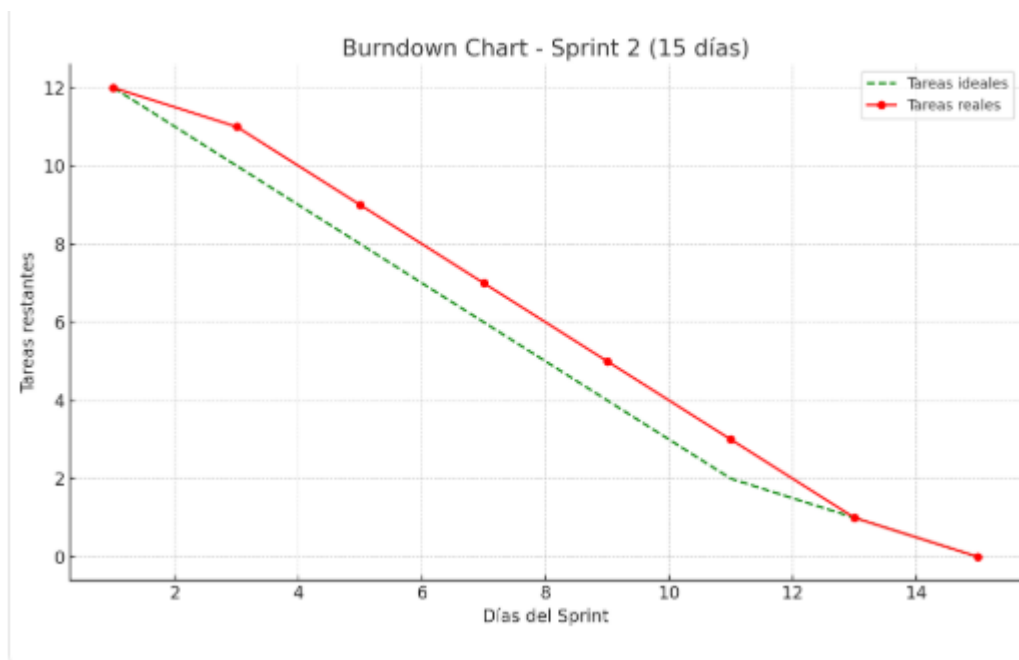
Acciones: Se decidió que para el próximo sprint, se destinaría más tiempo a las pruebas finales y a la corrección de errores antes de pasar a las pruebas de aceptación.

### 4.3. Burndown chart y análisis por sprint



**Figura 2:** *Burndown Chart - Sprint 1 (15 días)*

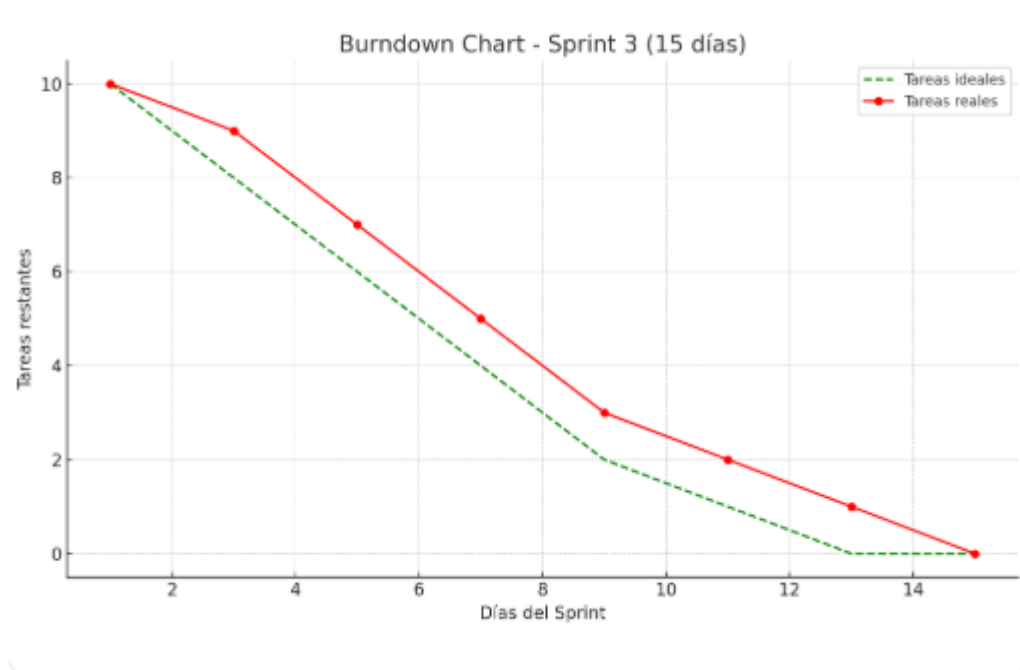
El gráfico presenta el progreso de un Sprint de 15 días. La línea roja muestra las tareas reales restantes, mientras que la línea verde punteada refleja el progreso ideal. Ambos ejes indican los días del Sprint (X) y las tareas restantes (Y).





**Figura 3:** *Burndown Chart - Sprint 2 (15 días)*

Este gráfico muestra el progreso de un Sprint de 15 días. La línea roja indica las tareas reales restantes, mientras que la línea verde punteada muestra el progreso ideal. Ambos ejes representan los días del Sprint (X) y las tareas restantes (Y).



**Figura 4:** *Burndown Chart - Sprint 3 (15 días)*

Este gráfico muestra el progreso de un Sprint de 15 días, con las tareas restantes representadas por la línea roja (tareas reales) y el progreso ideal señalado por la línea verde punteada. Se observa la evolución del trabajo en función del tiempo del Sprint.

### **Análisis de cada Sprint:**

- Sprint 1: En este sprint, el avance fue moderado, pero se completaron las tareas definidas para el inicio del proyecto, aunque algunos requerimientos quedaron pendientes.
- Sprint 2: Este sprint mostró una mayor estabilidad en el avance, con casi todas las tareas completas. Las mejoras de interfaz de usuario quedaron como una tarea pendiente.

- Sprint 3: El avance fue bastante bueno, pero algunas pruebas funcionales no se completaron a tiempo. Sin embargo, se asignaron las tareas de pruebas de aceptación y la finalización de las funcionalidades pendientes.

#### **4.4. Evidencias de las ceremonias SCRUM**

Las evidencias de las ceremonias SCRUM, que incluyen las reuniones de planificación, revisión, daily stand-ups y retrospectivas, fueron grabadas y almacenadas en formato de video. Los enlaces a los videos correspondientes se encuentran disponibles en el siguiente enlace de OneDrive.

 **Accede aquí a los enlaces de los videos de las ceremonias SCRUM:**

[Sprints TUCANCHA](#)

#### **4.5. Escenarios de casos de prueba**

##### **CP01 – Registro de Usuario Exitoso**

**Escenario 1.1:** El usuario completa todos los campos correctamente y se registra. El sistema muestra una alerta de “Registro exitoso” y redirige al login.

**Escenario 1.2:** El usuario intenta registrarse con un correo ya registrado. El sistema muestra una alerta indicando que el correo ya existe.

##### **CP02 – Registro Fallido por Campos Vacíos**

**Escenario 2.1:** El usuario ingresa solo el correo electrónico. El sistema solicita completar los demás campos.

**Escenario 2.2:** El usuario ingresa solo la contraseña. El sistema solicita completar los demás campos.

##### **CP03 – Inicio de Sesión Correcto**

**Escenario 3.1:** El usuario ingresa un correo y contraseña válidos previamente registrados. El sistema valida los datos y redirige a la página de reservas.

##### **CP04 – Inicio de Sesión Incorrecto**

**Escenario 4.1:** El usuario ingresa un correo que no está registrado. El sistema muestra una alerta de “Credenciales incorrectas.”

**Escenario 4.2:** El usuario ingresa una contraseña incorrecta para un correo válido. El sistema muestra una alerta de “Credenciales incorrectas.”

#### **CP05 – Acceso no autorizado a reservas**

**Escenario 5.1:** El usuario intenta acceder directamente a la página de reservas sin haber iniciado sesión. El sistema muestra una alerta de advertencia y redirige al login.

**Escenario 5.2:** El usuario inicia sesión correctamente y accede a la página de reservas sin inconvenientes.

#### **CP06 – Mostrar Reservas del Usuario**

**Escenario 6.1:** El usuario tiene reservas guardadas en “sessionStorage”. Al ingresar a la página de reservas, estas se muestran en forma de lista.

**Escenario 6.2:** El usuario tiene datos corruptos o ilegibles en el almacenamiento. El sistema no muestra reservas o lanza error (dependiendo de la implementación).

#### **CP07 – Mensaje sin reservas registradas**

**Escenario 7.1:** El usuario inicia sesión pero no tiene reservas almacenadas. El sistema muestra un mensaje informando que aún no hay reservas.

#### **CP08 – Cerrar Sesión**

**Escenario 8.1:** El usuario hace clic en el botón “Cerrar sesión”. El sistema elimina los datos de sesión y redirige al login.

**Escenario 8.2:** Después de cerrar sesión, el usuario intenta volver a la página de reservas. El sistema lo redirige nuevamente al login, ya que no hay sesión activa.

### **4.6. Casos de prueba**

#### **CP01 – Registro de Usuario Exitoso**

Objetivo: Validar que un nuevo usuario pueda registrarse correctamente.

Datos de entrada: Nombre, correo, contraseña.

Resultado esperado: Aparece alerta “Registro exitoso” y se cambia vista al login.

Procedimiento de prueba:

1. Ir al formulario de registro
2. Ingresar datos válidos.
3. Clic en “Registrarse”.

## **CP02 – Registro Fallido por Campos Vacíos**

Objetivo: Verificar que no se permita el registro si faltan datos.

Datos de entrada: Uno o más campos vacíos.

Resultado esperado: Alerta “Por favor, completa todos los campos.”

Procedimiento de prueba:

1. Ir al registro.
2. Dejar campos vacíos.
3. Clic en “Registrarse”.

## **CP03 – Inicio de Sesión Correcto**

Objetivo: Verificar que el usuario pueda iniciar sesión con credenciales válidas.

Datos de entrada: Correo y contraseña válidos registrados previamente.

Resultado esperado: Redirección a “reservas.html”.

Procedimiento de prueba:

1. Ingresar credenciales válidas.
2. Clic en “Entrar”.

## **CP04 – Inicio de Sesión Incorrecto**

Objetivo: Validar comportamiento ante credenciales inválidas.

Datos de entrada: Correo o contraseña incorrectos.

Resultado esperado: Alerta “Credenciales incorrectas.”

Procedimiento de prueba:

1. Ingresar datos inválidos.
2. Clic en “Entrar”.

### **CP05 – Acceso no autorizado a reservas**

Objetivo: Impedir acceso a “reservas.html” si no hay sesión activa.

Condición inicial: Usuario no logueado.

Resultado esperado: Redirección a “index.html” y alerta: “Debes iniciar sesión primero.”

Procedimiento de prueba:

1. Abrir reservas.html directamente sin login.
2. Verificar redirección y alerta.

### **CP06 – Mostrar Reservas del Usuario**

Objetivo: Verificar que se muestran las reservas guardadas en “sessionStorage”.

Condición inicial: Usuario logueado con reservas.

Resultado esperado: Lista visible de reservas en pantalla.

Procedimiento de prueba:

1. Iniciar sesión.
2. Cargar reservas.html.
3. Verificar la visualización de reservas.

### **CP07 – Mensaje sin reservas registradas**

Objetivo: Validar que se muestre mensaje si no hay reservas.

Condición inicial: Usuario logueado sin reservas.

Resultado esperado: Mensaje: “No tienes reservas aún.”

Procedimiento de prueba:

1. Iniciar sesión sin reservas.
2. Verificar mensaje en reservas.html.

### **CP08 – Cerrar Sesión**

Objetivo: Verificar que el botón cierre sesión correctamente y redirija al login.

Entrada: Clic en botón “Cerrar sesión”.

Resultado esperado: Eliminación de sesión y redirección a “index.html”.

Procedimiento de prueba:

1. Estar logueado.
2. Clic en “Cerrar sesión”.
3. Verificar redirección y que el “sessionStorage” se vacíe.

#### **4.7. Pruebas de aceptación**

Las pruebas de aceptación se documentaron detalladamente en una hoja de cálculo colaborativa, donde se especifican los requerimientos del sistema, los criterios de aceptación, los escenarios de prueba, los resultados obtenidos y la validación correspondiente. Esta herramienta permitió asegurar que cada funcionalidad implementada cumpliera con las expectativas del usuario y con los objetivos del proyecto.

 **Accede aquí a la hoja de cálculo con las pruebas de aceptación:**

 Pruebas De Aceptacion

#### **4.8. Scripts con Selenium**

Las pruebas funcionales automatizadas fueron desarrolladas utilizando Selenium, permitiendo validar el correcto funcionamiento de las funcionalidades clave del sistema, como el registro, el inicio de sesión y la navegación hacia la sección de reservas. Estas pruebas fueron documentadas en un archivo independiente, donde se incluyen los scripts ejecutados, los casos de prueba automatizados, los resultados obtenidos y las respectivas capturas de evidencia.

 **Accede aquí al documento con los scripts y resultados de pruebas Selenium:**

 PRUEBAS FUNCIONALES

#### **4.9. Descripción y análisis de las pruebas funcionales**

Durante la fase de verificación del sistema "TU CANCHA", se desarrollaron pruebas funcionales automatizadas utilizando Selenium con Python para garantizar que los componentes principales del sistema operaran correctamente frente a distintos escenarios de usuario. A continuación, se detallan los escenarios evaluados y sus respectivos resultados.

##### **CP01 – Registro de Usuario**

- **Escenario 1.1:** El usuario completa correctamente el formulario de registro. El sistema muestra la alerta *“Registro exitoso. Ahora inicia sesión”* y redirige automáticamente al login.  
**Resultado funcional:** Registro validado correctamente y mensaje mostrado.  
**Estado:** OK.
- **Escenario 1.2:** El usuario intenta registrarse con un correo ya registrado. El sistema impide el registro duplicado y lanza la alerta *“Este correo ya está registrado”*.  
**Resultado funcional:** El sistema valida correctamente el duplicado y bloquea el registro.  
**Estado:** OK.

#### **CP02 – Validación de Campos en Registro**

- **Escenario 2.1:** El usuario deja vacío el campo “nombre” y trata de registrarse. El sistema lanza la alerta *“Por favor, completa todos los campos”*.  
**Resultado funcional:** El formulario bloquea el registro incompleto.  
**Estado:** OK.
- **Escenario 2.2:** El usuario deja vacío el campo “correo” o “nombre” e ingresa solo la contraseña. El sistema responde con la misma alerta.  
**Resultado funcional:** Validación y bloqueo exitosos.  
**Estado:** OK.

#### **CP03 – Inicio de Sesión con Credenciales Válidas**

- **Escenario 3.1:** El usuario inicia sesión con credenciales válidas. El sistema redirige a reservas.html, donde se muestra el panel *“Mis Reservas”*.  
**Resultado funcional:** Redirección correcta.  
**Estado:** OK.

#### **CP04 – Inicio de Sesión con Credenciales Incorrectas**

- **Escenario 4.1:** El usuario utiliza un correo no registrado. El sistema lanza la alerta *“Credenciales incorrectas”*.  
**Resultado funcional:** Control de autenticación exitoso.  
**Estado:** OK.

- **Escenario 4.2:** El usuario utiliza una contraseña errónea para un correo válido. El sistema también muestra la alerta correspondiente.

**Resultado funcional:** Autenticación fallida correctamente gestionada.

**Estado:** OK.

#### **CP05 – Acceso con o sin sesión activa**

- **Escenario 5.1:** El usuario intenta acceder directamente a reservas.html sin haber iniciado sesión. El sistema muestra una alerta y redirige al login.

**Resultado funcional:** Seguridad validada, acceso bloqueado.

**Estado:** OK.

- **Escenario 5.2:** Usuario autenticado accede correctamente a reservas.html.

**Resultado funcional:** Interfaz desplegada exitosamente.

**Estado:** OK.

#### **CP06 – Visualización de Reservas**

- **Escenario 6.1:** El sessionStorage contiene reservas. El sistema recupera los datos y los muestra en la interfaz.

**Resultado funcional:** Visualización funcional validada.

**Estado:** OK.

- **Escenario 6.2:** El sessionStorage contiene datos corruptos. El sistema muestra alerta de error y solicita nueva autenticación.

**Resultado funcional:** Manejo de errores exitoso.

**Estado:** OK.

#### **CP07 – Usuario sin reservas**

- **Escenario 7.1:** El usuario no tiene reservas almacenadas. El sistema muestra el mensaje *“No tienes reservas aún”*.

**Resultado funcional:** Mensaje informativo correcto.

**Estado:** OK.

#### **CP08 – Cierre de Sesión**

- **Escenario 8.1:** Al hacer clic en *“Cerrar sesión”*, el sistema elimina los datos de sesión y redirige al login.



**Resultado funcional:** Flujo de cierre ejecutado correctamente.

**Estado:** OK.

- **Escenario 8.2:** Usuario intenta acceder a reservas.html luego de cerrar sesión. El sistema muestra alerta por sesión inválida y redirige.

**Resultado funcional:** Validación de sesión implementada correctamente.

**Estado:** OK.

#### 4.10. Tablas e imágenes con numeración y descripción

A continuación se presenta la numeración, título y descripción de las tablas utilizadas en el desarrollo del proyecto "TU CANCHA". Cabe resaltar que otras tablas e imágenes adicionales que complementan este informe se encuentran referenciadas en los documentos vinculados en los apartados 4.7 y 4.8. Además, ahora se incluye una Tabla de Figuras, que presenta la numeración y descripción de todas las imágenes relevantes utilizadas en el informe.

**Tabla de Tablas**

Nº	Título	Descripción
1	Trazabilidad de requerimientos funcionales y casos de prueba	Relación entre los requerimientos funcionales definidos y los casos de prueba asociados que permiten validarlos.

**Tabla de figuras**

Nº	Título	Descripción
1	Tablero de tareas del Sprint 1	Muestra el tablero de tareas del Sprint 1, con la lista de tareas asignadas, su estado de avance (finalizadas o en curso) y los responsables de cada tarea durante el período del 6 de mayo al 14 de junio.

2	Burndown Chart - Sprint 1 (5 días)	Gráfico de las tareas restantes en 5 días de Sprint 1.
3	Burndown Chart - Sprint 1 (15 días)	Comparación de tareas restantes en Sprint 1, durante 15 días.
4	Burndown Chart - Sprint 2 (15 días)	Visualización de tareas restantes en Sprint 2, durante 15 días.

## 5. Próximas Etapas

### 5.1. Pruebas de carga

Hasta el momento no se han realizado pruebas de carga sobre el sistema. Sin embargo, se contempla la realización de estas pruebas en fases futuras del desarrollo, con el fin de evaluar el rendimiento del sistema.

### 5.2. Preparación del informe final

### 5.3. Defensa del proyecto

