**Plan de Pruebas**

***Caso: Proyecto Vecinal RedBarrio***

***Fecha: [17/10/2025]***

**Asignatura: Portafolio de Título**

**Integrantes:**

**Joaquin Fuentes**

**Simon Dinamarca**

**Raul Nuñez**

**Fecha de entrega: *17/10/2025***

**Profesor: Arturo Vargas**

**Historial de revisiones**

| **Fecha** | **Versión** | **Descripción** | **Autor** |
| --- | --- | --- | --- |
| 17/10/2025 | <1.0> | Se desarrolla el documento incluyendo todos los puntos referentes a: Introducción, Requerimiento para pruebas, Estrategia de pruebas, Recursos, Actividades, Hitos y Entregables | Raul Nuñez |
| **Fecha** | **Versión** | **Descripción** | **Autor** |
|  |  |  |  |

**TABLA DE CONTENIDO**

[**1.**](#_heading=h.gjdgxs) **Introducción** 4

[**1.2**](#_heading=h.30j0zll) **Alcance de las pruebas** 4

[**1.2.1**](#_heading=h.1fob9te) **Elementos de pruebas** 4

[**1.2.2**](#_heading=h.3znysh7) **Pruebas Funcionales** 4

[**1.2.3**](#_heading=h.2et92p0) **Riesgos** 5

[**1.2.3.1**](#_heading=h.tyjcwt) **Matrices de riesgos** 5

[**1.2.3.1.1**](#_heading=h.3dy6vkm) **Pruebas** 5

[**1.3**](#_heading=h.1t3h5sf) **Tipos de pruebas** 5

[**1.3.1**](#_heading=h.4d34og8) **Pruebas Funcionales** 5

[**1.3.2**](#_heading=h.2s8eyo1) **Herramientas involucradas** 5

[**1.3.3**](#_heading=h.17dp8vu) **Preparación del ambiente de pruebas** 6

[**1.3.4**](#_heading=h.3rdcrjn) **Diseño del ambiente de pruebas** 6

[**1.3.5**](#_heading=h.26in1rg) **Diseño ambiente de pruebas** 9

[**2.**](#_heading=h.lnxbz9) **Resultados de las pruebas** 9

[**2.1 Reporte de defectos** 10](#_heading=h.44sinio)

# **Introducción**

* 1. **Resumen ejecutivo**

**Resumen Ejecutivo**

En el presente documento se detalla el Plan de Pruebas asociado al proyecto “Sistema de Gestión Vecinal RedBarrio”, una plataforma móvil desarrollada para la Junta de Vecinos, cuyo propósito es agilizar la comunicación, la gestión de solicitudes y la interconectividad entre la comunidad vecinal y la administración comunal.

El objetivo del plan es garantizar la calidad y correcto funcionamiento del sistema, definiendo los módulos, funcionalidades, tipos de pruebas, entornos y recursos necesarios para su validación. En este documento se establecen los lineamientos para verificar que el software cumple con los requerimientos funcionales actuales, de usabilidad y desempeño esperados por los usuarios finales.

El sistema integra diferentes módulos que permiten la interacción entre los actores principales (vecinos, directiva y administrador), buscando optimizar los procesos comunitarios y la trazabilidad de la información.

Los principales módulos para evaluar durante la fase de pruebas son:

* Módulo de Autenticación y Perfil de Usuario: autenticar usuario, registrar usuario, cambiar contraseña, perfil
* Módulo de Solicitudes y Certificados: registro, seguimiento y aprobación de solicitudes vecinales.
* Módulo de Actividades y Proyectos Comunitarios: generar actividades, generar proyecto, inscripción proyectos, inscripción actividades
* Módulo Administrativo: gestión de usuarios, roles, reportes y parámetros del sistema.
* Módulo de Noticias: gestión de noticias, publicaciones e interacciones del vecino
* Módulo de espacios: generar espacios, arrendar un espacio
* Módulo de votaciones: visualización de votaciones, generar votaciones

Este plan de pruebas busca establecer un enfoque sistemático que asegure que cada componente del sistema funcione de forma correcta, integrada y estable, contribuyendo al cumplimiento de los objetivos del proyecto: fortalecer la participación vecinal, mejorar la transparencia en la gestión comunitaria y reducir los tiempos de respuesta administrativa.

## **Alcance de las pruebas**

El presente plan de pruebas tiene como objetivo validar los componentes funcionales y de integración del sistema RedBarrio, asegurando que cada módulo cumpla con los requerimientos definidos en las especificaciones del proyecto.

El alcance incluye la verificación de la correcta interacción entre los módulos de usuario, solicitudes, proyectos, actividades, noticias, espacios y votaciones, considerando los distintos perfiles del sistema: vecino, directiva y administrador.

### **Elementos de pruebas**

Dentro de los módulos que serán que serán usados en el plan de pruebas nos encontramos con los siguientes:

| Sistema | Tecnologías | Detalle |
| --- | --- | --- |
| APK (aplicación móvil) | Aplicación construida con Ionic + Angular + Capacitor. | Módulos:   * Autenticación, perfil y gestión de usuario * solicitudes, reservas y certificados * generación e inscripción actividades y proyectos * noticias, espacios y votaciones. |
| Backend | * Supabase * PostgreSQL * API REST * Plugins * Supabase Realtime * Netlify * n8n | Usos de las tecnologías:   * Webpay * MapBox * Base de datos en la nube * CRUD, OCR * Alojamiento de pagina puente para transacciones * automatizaciones |

### **Pruebas Funcionales**

Se empleará la técnica de testing de caja negra, enfocándose en la validación del comportamiento del sistema desde el punto de vista del usuario y la correcta inserción de la información en la base de datos (supabase). El propósito principal es verificar que cada flujo funcional cumpla con los resultados esperados al rellenar los datos.

**Objetivos de las pruebas funcionales:**

* Comprobar que cada módulo cumple su propósito funcional.
* Validar los flujos de navegación, creación, edición y eliminación de registros.
* Garantizar la correcta interacción entre la aplicación móvil y el backend Supabase.

### **Riesgos**

Durante la ejecución de las pruebas se identifican posibles riesgos que pueden afectar la estabilidad o disponibilidad del sistema. Estos se clasifican según su gravedad (Baja, Media o Alta), con el fin de priorizar la atención y mitigación.

| N° | Riesgo | Descripción | Gravedad | Acción Correctiva / Mitigación |
| --- | --- | --- | --- | --- |
| 1 | Fallo en la conexión con Supabase | La aplicación no se conecta con el proyecto BD de supabase | Alta | validar ruta conexión, Verificar tablas de visualización de información |
| 2 | El usuario no puede registrarse o logearse | El usuario no logra registrarse o iniciar sesión en el sistema | Media | Verificar funcionamiento de proceso de CRUD o validaciones de id en la base de datos (supabase) |
| 3 | Fallo en la carga, envío, generacion o redirección de módulos | la aplicación no redirecciona, genera, envía o no carga correctamente las rutas y CRUD de inserción de datos | Alta | Verificar archivo de enrutamientos, revisar CRUD de inserción, validar conexion con las tablas relacionadas |
| 4 | relleno de formulario incompletos o inconsistentes en generación o inscripción de actividades, proyectos, votaciones, espacios, edición de perfil | La aplicación registra a medias o no registra campos claves para la generación o inscripción de módulos | alta | El sistema debe aclarar los campos obligatorios, ocultando o controlando el botón de envío de formulario para evitar inconsistencias. |
| 5 | Interrupción del servicio móvil | La aplicación deja de funcionar repentinamente en un uso general de los módulos y sistemas | Alta | Validar el correcto flujo de toda la aplicación, asegurando la navegación sin caídas repentinas. |
| 6 | Fallo en visualización de votaciones o espacios | La aplicación no muestra correctamente las votaciones o espacios disponibles | Medio | Asegurar que la aplicación recepciona y muestra correctamente la información cargada en la base de datos. |
| 7 | El OCR no escanea correctamente la Cedula de identidad. | La aplicacion le dificulta o le es imposible identificar los campos que autorellena el ORC | Medio | Validar plugin de OCR para escanear y rellenar correctamente los campos requeridos, Asegurar y aclarar ángulo de fotografía para poder escanear correctamente los datos |
| 8 | El simulador de Webpay no procesa el pago, deja de funcionar, queda en loop infinito o no redirecciona a la aplicación. | El simulador de pago no funciona como lo esperado fuera de la aplicación, dejando en loop, no procesa el pago o no redirecciona directo a la aplicacion. | Alto | Validar ApiRest de transbank, asegurar que los datos y el formulario es enviado de manera optima, Corroborar correcta conexión con la Base de datos (SupaBase) o puente de conexión (Netlify) |
| 9 | MapBox caído o no detecta dirección | La aplicacion no identifica la direccion y visualización del mapa | Alta | Verificar integración ApiRest de MapBox, validar que se ingrese correctamente el detalle de la dirección y corroborar la correcta visualización del mapa |
| 10 | Página puente no recepciona y no entrega resultados esperados. | El simulador de transbank no envía la información o la pagina puente | Alta | Verificar conexión entre simulador transbank con la pagina puente y redireccionamiento a la aplicación |
| 11 | La automatización n8n no recepciona o dispara procesos automatizados. | El flujo de orquestación no recepciona o no dispara los procesos de automatización de manera correcta o no envía correctamente el resultado a la aplicación | Alta | Validar conexiones y enrutamiento del servicio n8n, Verificar posicionamiento y envío de la información, como redirección al correo |

#### **Matrices de riesgos**

##### **Pruebas**

| **N° Riesgo** | **Descripción** | **Gravedad** | **Acción** |
| --- | --- | --- | --- |
| 1 | Fallo en la conexión con Supabase: la aplicación no se conecta con el proyecto de base de datos Supabase. | Alta | Validar la ruta de conexión y verificar la correcta visualización de las tablas y datos. |
| 2 | El usuario no puede registrarse o logearse: el usuario no logra completar el proceso de registro o inicio de sesión. | Media | Verificar funcionamiento de los procesos CRUD y validaciones de ID en la base de datos Supabase. |
| 3 | Fallo en la carga, envío, generación o redirección de módulos: la aplicación no redirecciona, genera, envía o carga correctamente los CRUD o rutas. | Alta | Verificar archivo de enrutamientos, revisar CRUD de inserción y validar la conexión con las tablas relacionadas. |
| 4 | Relleno de formularios incompletos o inconsistentes en generación o inscripción de actividades, proyectos, votaciones, espacios o edición de perfil. | Alta | El sistema debe aclarar los campos obligatorios y controlar el botón de envío para evitar registros inconsistentes. |
| 5 | Interrupción del servicio móvil: la aplicación deja de funcionar repentinamente durante el uso de los módulos. | Alta | Validar el flujo completo de la aplicación y asegurar la navegación sin caídas repentinas. |
| 6 | Fallo en la visualización de votaciones o espacios: la aplicación no muestra correctamente los datos cargados. | Media | Asegurar que la aplicación reciba y muestre correctamente la información proveniente de la base de datos. |
| 7 | El OCR no escanea correctamente la cédula de identidad: el sistema no identifica adecuadamente los campos para el autorelleno. | Media | Validar el plugin OCR, asegurar el correcto escaneo de campos y recomendar ángulo y condiciones de fotografía. |
| 8 | El simulador de Webpay no procesa el pago, se detiene o no redirecciona a la aplicación. | Alta | Validar la API REST de Transbank, revisar envío de formulario, conexión con Supabase o puente intermedio (Netlify). |
| 9 | MapBox caído o sin detección de dirección: el sistema no identifica o no muestra correctamente el mapa. | Alta | Verificar la integración de la API REST de MapBox y validar los datos de dirección ingresados. |
| 10 | Página puente no recepciona ni entrega resultados esperados del simulador Transbank. | Alta | Verificar conexión entre el simulador Transbank y la página puente, asegurando redirección correcta a la app. |
| 11 | La automatización n8n no recepciona o dispara los procesos programados. | Alta | Validar conexiones y enrutamiento del servicio n8n, revisando envío de datos y redirección de resultados al correo o app. |

## **Tipos de pruebas**

### **Pruebas Funcionales**

El testing funcional se realizará sobre los requerimientos funcionales y sus respectivos casos de uso del sistema RedBarrio. El fin es asegurar la funcionalidad y capacidad de la aplicación frente a diferente situaciones de altas cargas y diversas formas de interactuar con el sistema

* **Objetivo de la prueba:** El propósito de estas pruebas es comprobar que cada módulo cumpla correctamente las reglas del negocio, asegurando la interacción adecuada entre los componentes del sistema y el flujo de procesos de la aplicación (frontend, backend y base de datos Supabase).

Garantizar que la navegación y el flujo de uso dentro de cada módulo reflejen correctamente las reglas del negocio y los requerimientos definidos, comprobando la estabilidad general del sistema y la coherencia de los resultados frente a entradas válidas e inválidas.

**Objetivos específicos:**

* Validar que cada flujo funcional responda según las acciones del usuario.
* Asegurar que la interacción entre aplicación móvil, API REST y base de datos Supabase sea estable y consistente.
* Verificar que las funciones críticas (login, solicitudes, pagos, votaciones, OCR, n8n, etc.) operen de forma correcta tras cada integración.
* **Técnica para utilizar:** Para cada módulo funcional del sistema se aplicarán los siguientes tipos de pruebas:

| Tipo de Prueba | Descripción | Objetivo Específico |
| --- | --- | --- |
| Caja Negra (Black Box Testing) | Se evalúan entradas y salidas enfocadas en la interacción con la aplicación. | verificar que los datos válidos producen resultados esperados y manejo de inconsistencias. |
|  | Verificación inicial tras despliegues o integraciones. | Asegurar que las funciones principales del sistema (inicio de sesión, registro, formularios, navegación, conexión Supabase, transacciones de pago, etc.) no estén afectadas por cambios recientes. |

* **Criterio de validación:**

Las pruebas se considerarán exitosas cuando:

* Todas las pruebas planificadas se ejecutan correctamente sin errores críticos.
* Las funciones principales del sistema responden según los requerimientos definidos.
* Los defectos detectados serán registrados en el reporte de defectos y asignados para su corrección.
* Los flujos prioritarios (autenticación, registro, envío de solicitudes, generación de proyectos, pagos y visualización de votaciones) funcionan de manera estable y sin interrupciones.
* La conexión con Supabase, Transbank, MapBox y n8n se mantiene activa y operativa durante la ejecución.

### **Herramientas involucradas**

A continuación, se presentarán las diferentes herramientas que se utilizaron para probar las distintas funcionalidades que tiene el programa.

|  | **Herramienta** | **Versión** |
| --- | --- | --- |
| **Herramienta de Base de Datos** | **SupaBase** | **2.0** |
| **Framework** | **Ionic**  **Angular** | **7.x**  **17.x** |
| **editor de código fuente** | **Visual Studio Code** | **1.94** |
| **Entorno de ejecución JavaScript** | **Node.js** | **20.x** |
| **Servicio web de despliegue de pagina puente** | **Netlify** | **-** |
| **Orquestador de automatizaciones y flujos de procesos** | **n8n** | **1.72** |
| **Simulador Webpay** | **Transbank SDK** | **3.2** |
| **API de mapas para geolocalización de direcciones** | **MapBox** | **2.16** |
| **Tecnología de de escaneo de documentos y fotografías** | **Plugin OCR** | **1.3** |
| **Entorno de ejecución y depuración de la app en dispositivos Android.** | **Android Studio** | **2024.2** |

### **Preparación del ambiente de pruebas**

Para la configuración del ambiente de pruebas, se definieron los componentes de software, hardware y servicios en línea necesarios para garantizar un entorno controlado, estable y representativo del entorno de producción.

El sistema RedBarrio utiliza una arquitectura híbrida conectada a servicios en la nube. Por ello, el entorno de pruebas incluye tanto recursos locales (para depuración y ejecución móvil) como recursos remotos (para base de datos, automatizaciones y despliegue de API).

**Detalles del entorno de pruebas:**

* Aplicación móvil (cliente): desarrollada en Ionic + Angular, ejecutada en entorno Android mediante Capacitor.
* Backend: administrado por Supabase, con autenticación integrada, base de datos PostgreSQL y funciones automatizadas.
* Entorno de desarrollo: Visual Studio Code y Android Studio, utilizados para compilación, ejecución, depuración de la aplicación y generación de la APK.
* Automatización: flujos n8n para validación de procesos automáticos (envío de correos, registro de solicitudes, notificaciones).
* API externas: integración con servicios REST (Transbank, MapBox y OCR) para pruebas de pago, geolocalización y reconocimiento de texto.
* Red y despliegue: pruebas ejecutadas en entorno de conexión estable (fibra óptica 200 Mbps), con página puente alojada en Netlify para simulaciones de pago.

Este entorno permite ejecutar pruebas funcionales sobre los distintos módulos de la aplicación, validando la correcta comunicación entre los componentes y la estabilidad general del sistema.

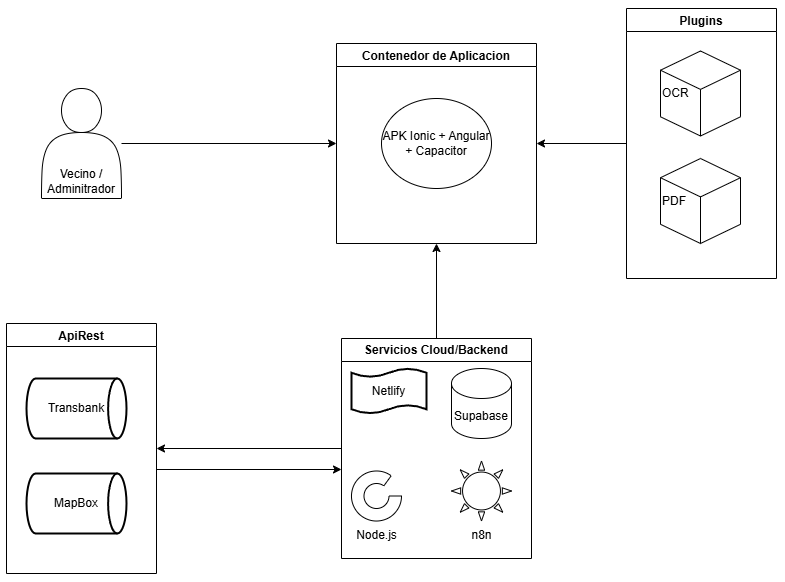
### **Diseño del ambiente de pruebas**

| **ITEM** | **DESCRIPCIÓN** | **OBSERVACIONES** |
| --- | --- | --- |
| **HARDWARE** |  |  |
| **Estación de Pruebas (Administrador/vecino)** |  |  |
|  | **Computador** | **Telefono Android** |
| Procesador | Intel Core i5-12400 o equivalente | Octa-Core 2.0 GHz (Snapdragon 665 / MediaTek Helio G80 o superior) |
| Memoria RAM | 16 GB DDR4 3200 MHz | 4 GB |
| Espacio en Disco | SSD 512 GB | 64 GB |
| Tipo Monitor y Resolución | Monitor Full HD 1080p 24’’ | Pantalla táctil Full HD (1080 x 2400 px) de 6.0’’ o superior |
| Unidad de Disquete | N/A | N/A |
| Tarjeta de red | Integrada 1 Gb Ethernet / Wi-Fi 6 | Wi-Fi 802.11 a/b/g/n/ac + Bluetooth 5.0 |
| Modem | Arris TG2482A (fibra óptica 200 Mbps) | 4G LTE (recomendado 5G) |
| Mouse | Logitech MK270 inalámbrico | Opcional (por Bluetooth o USB OTG) |
| Tipo Enlace | Red doméstica LAN / Wi-Fi | Conexión inalámbrica (Wi-Fi o datos móviles) |
| Gráfica | integrada de la CPU | Adreno 610 / Mali-G52 o superior |
| Versión de Android | N/A | Android 11 (API 30) o superior |
| Batería | N/A | 4000 mAh o superior |
| Puertos / Conectividad | USB-C o microusb, soporte para depuración ADB | USB-C o microusb, soporte para depuración ADB |
| **Servidores** |  |  |
| ***Base de Datos (pruebas)*** |  |  |
| Base de Datos (Supabase – PostgreSQL Cloud) | Infraestructura gestionada por Supabase. Incluye autenticación, políticas RLS y triggers automáticos. |  |
| Aplicación Web (Netlify) | Despliegue de la página puente y endpoints públicos para integración con Transbank. |  |
| Orquestador (n8n Cloud) | Automatización de procesos: envío de correos, notificaciones y validación de flujos internos. |  |
| API externas | Servicios conectados: Transbank SDK (simulador Webpay), MapBox (geolocalización), OCR (Capacitor plugin). |  |
| **RED** |  |  |
| Topología | Estrella |  |
| Medio | Fibra Óptica |  |
| Velocidad | 20Mb/s |  |
| Protocolo | TCP/IP, HTTPS |  |
| Módems | Arris TouchStone CM820 |  |
| Conexión Internet | Permanente |  |
| **SOFTWARE** |  |  |
| **Estación de Pruebas (Cliente)** |  |  |
| Sistema Operativo | Windows 11 Pro x64 y Teléfono Android |  |
| Herramienta de *testing* | Postman, Android Emulator, Chrome DevTools, Android Studio |  |
| Entorno de Desarrollo | Visual Studio Code (1.94), Android Studio (2024.2) |  |
| Gestor de Dependencias | Node.js (20.x) / npm (10.x) |  |
| Browser | Google Chrome |  |
| Frameworks Frontend | Ionic (7.x), Angular (17.x), Capacitor (6.x) |  |
| **Base de Datos (pruebas)** |  |  |
| Sistema Operativo | Supabase (PostgreSQL Cloud) |  |
| Software de Red | Netlify CLI, PuTTY, Git |  |
| Dominio/Cuenta | Colaboradores |  |
| **Aplicación (pruebas)** |  |  |
| Sistema Operativo | Android |  |
| Software de Red | Conexion WIFI |  |
| Dominio/Cuenta | Admin/Vecino |  |
| **Repositorios** |  |  |
| GitHub | https://github.com/BatyMonkey/CapstoneJF-RN-SD |  |
| Supa Base Project | Proyecto “RedBarrio” |  |

### **Diseño ambiente de pruebas**

El siguiente diagrama representa el ambiente de pruebas del sistema RedBarrio, donde se visualizan los principales componentes tecnológicos que interactúan entre sí durante la ejecución de los casos de prueba.

La arquitectura definida busca garantizar una comunicación eficiente entre la aplicación móvil, los servicios en la nube y las APIs externas utilizadas para la validación de procesos clave como pagos, geolocalización y automatización de flujos.



A nivel de arquitectura, de forma general, el sistema está compuesto por una aplicación móvil híbrida (Ionic + Angular + Capacitor) que se comunica con un backend gestionado en Supabase, complementado por servicios de integración como n8n, Netlify, Transbank y MapBox.

Esta configuración permite aislar los entornos de desarrollo, pruebas y producción, asegurando la estabilidad y trazabilidad de los resultados durante las pruebas funcionales.

### **Ventajas del diseño del ambiente**

* Permite realizar pruebas completas sobre la interacción entre frontend móvil, backend y servicios externos, validando la integridad del flujo end-to-end.
* La arquitectura en la nube (Supabase, Netlify y n8n) reduce la necesidad de servidores locales y facilita la disponibilidad constante del entorno de pruebas.
* Los servicios externos, como Transbank (pagos) y MapBox (mapas), pueden ser simulados o controlados mediante APIs REST, permitiendo un control seguro del entorno de test.
* La estructura modular de la aplicación facilita la detección y corrección de errores en cada componente, sin afectar al resto del sistema.
* Los plugins de Capacitor (OCR y PDF) permiten validar funcionalidades nativas en un entorno híbrido, asegurando la compatibilidad con dispositivos Android.

### **Desventajas del diseño del ambiente**

* Al depender de servicios en la nube, una falla en Supabase o Netlify puede afectar temporalmente la ejecución de pruebas hasta restablecer la conexión.
* La integración de múltiples servicios externos (Transbank, MapBox, n8n) requiere mantener actualizados los tokens y credenciales de acceso.
* Las pruebas en dispositivos móviles reales pueden variar según la versión de Android o el rendimiento del hardware, por lo que se recomienda mantener dispositivos de prueba estandarizados.
* Las automatizaciones mediante n8n deben ser cuidadosamente monitoreadas, ya que una configuración incorrecta podría disparar procesos múltiples o generar duplicidades.

# **Resultados de las pruebas**

Durante la ejecución de las pruebas del sistema RedBarrio, se aplicaron pruebas funcionales de caja negra, ejecutadas por módulos según las áreas funcionales definidas.  
 Cada área fue verificada mediante los casos de prueba especificados en el documento “Casos de Prueba.xlsx”, donde se detallan los pasos, condiciones, resultados esperados y obtenidos.

El objetivo principal fue validar el correcto funcionamiento de cada módulo del sistema y asegurar que los flujos principales (autenticación, gestión de solicitudes, actividades, votaciones, pagos y automatizaciones) operen conforme a los requerimientos definidos.

En los siguientes cuadros se presentan los resultados por área funcional, los cuales mantienen la estructura utilizada en el plan de pruebas original.

| Área Funcional: Autenticación y Perfil de Usuario | | | | | |
| --- | --- | --- | --- | --- | --- |
| Paso | Propósito | Condiciones | Resultado esperado | Resultado obtenido | Observaciones |
| 1.- El usuario ingresa el correo registrado. 2.- El usuario ingresa la contraseña correspondiente con el correo. 3.- El usuario presiona el botón ingresar. | Iniciar sesion en el sistema | El usuario debe ingresar credenciales previamente registradas en el sistema | El usuario logra entrar al sistema, llegando a la página principal de este. | El usuario logra entrar al sistema, llegando a la página principal de este. |  |
| 1.- El usuario ingresa un correo no registrado en el sistema. 2.- El usuario ingresa una contraseña no registrada en el sistema. 3.- El usuario presiona el botón ingresar. | Validaciones al iniciar sesión | El usuario ingresa credenciales que no se encuentran registradas en el sistema | El sistema muestra un mensaje de error y deniega la entrada al sistema. | El sistema muestra un mensaje de error y deniega la entrada al sistema. |  |

| AREA FUNCIONAL : Registrar usuario | | | | | |
| --- | --- | --- | --- | --- | --- |
| Paso | propósito | Condiciones | Resultado esperado | Resultado obtenido | Observaciones |
| 1.- El usuario ingresa el primer nombre  2.- El usuario ingresa el segundo nombre(opcional).  3.- El usuario ingresa el primer apellido.  4.- El usuario ingresa el segundo apellido.  5.- El usuario ingresa el RUT.  6.- El usuario ingresa la fecha de nacimiento.  7.- El usuario selecciona el sexo.  8.- El usuario ingresa la dirección.  9.- El usuario ingresa el teléfono.  10.- El usuario ingresa el correo.  11.- El usuario ingresa una contraseña (mínimo 6 caracteres).  12.- El usuario presiona el botón "Registrarme" | Registrarse en el sistema como nuevo vecino validando datos básicos. | El usuario no debe estar registrado en el sistema | El usuario queda registrado en el sistema y puede acceder al sistema | El usuario queda registrado en el sistema y puede acceder al sistema |  |
| 1.-El usuario presiona el botón "Escanear Cédula".  2.- El usuario verifica que los datos sean correctos.  3.- El usuario ingresa los datos faltantes  4.- El usuario presiona el botón "Registrarme" | Registrarse en el sistema utilizando OCR | El usuario no debe estar registrado en el sistema | El usuario queda registrado en el sistema y puede acceder al sistema | El usuario queda registrado en el sistema y puede acceder al sistema |  |
| 1.-Ingresar los datos dejando en blanco algún campo obligatorio | Registrarse en el sistema sin ingresar datos obligatorios | El usuario no debe estar registrado en el sistema y debe intentar registrarse olvidando datos obligatorios | Debe salir un error mencionando el campo que falta por completar y no se registra el usuario en el sistema | Debe salir un error mencionando el campo que falta por completar y no se registra el usuario en el sistema |  |

| AREA FUNCIONAL : Modulo generar proyecto(administrador) | | | | | |
| --- | --- | --- | --- | --- | --- |
| Paso | propósito | Condiciones | Resultado esperado | Resultado obtenido | Observaciones | |
| 1. El usuario de tipo Administrador logeado.  2. Despliega el Menu de hamburguesa (superior izquierda).  3. Click en el boton "Generar Proyeto/Actividad" | Boton de menu "Generar proyecto/Actividad" | El usuario debe estar logeado, ademas debe ser un usuario de tipo administrador | Al clickear en "Generar proyecto/Actividad" el usuario es redireccionado e ingresa al modulo de "Generar publicacion - Proyecto" | Al clickear en "Generar proyecto/Actividad" el usuario es redireccionado e ingresa al modulo de "Generar publicacion - Proyecto" |  | |
| 1. Se ingresa al Formulario de generacion de Proyecto  2. Se elige el tipo de publicacion a "Proyecto"  3. se rellena el campo "titulo del proyecto"  4. se rellena el campo "Descripcion del proyecto"  5. se sube una imagen de referencia  6. apretar el boton "Crear proyecto"  7. apretar el boton "Limpiar" si se requiere limpiar los Labels | Rellenar Formulario de generacion de proyecto | El usuario Administrador debe estar dentro del modulo de Generar publicacion en el path "/generar/proyecto" | una vez rellenado el formulario de generacion de proyecto y apretar el boton "crear proyecto" de manera exitosa, deberia sobresalir una caja de texto | una vez rellenado el formulario de generacion de proyecto y apretar el boton "crear proyecto" de manera exitosa, deberia sobresalir una caja de texto |  | |

| AREA FUNCIONAL : Cambiar contraseña | | | | | |
| --- | --- | --- | --- | --- | --- |
| Paso | Descripción | Condiciones | Resultado esperado | Resultado obtenido | Observaciones |
| 1.- El usuario ingresa el correo eletrónico registrado.  2.- El usuario presiona el botón "Recuperar Contraseña" | Correo electrónico de recuperación de contraseña | El correo debe estar registrado en el sistema | El sistema envía un correo de restauración al correo registrado en el sistema | El sistema envía un correo de restauración al correo registrado en el sistema |  |
| 1.- El usuario utiliza el correo de recuperación.  2.- El usuario ingresa la nueva contraseña (mínimo 6 caracteres).  3.- El usuario presiona el botón "Cambiar Contraseña". | Cambio de contraseña | El usuario debe entrar al formulario desde el correo de recuperación | El usuario logra cambiar la contraseña | El usuario logra cambiar la contraseña |  |
| 1.- No ingresar correo en el campo.  2.- Ingresar correo en el campo que aparece.  3.- Presionar botón "Enviar Enlace". | Cambio de contraseña | El correo debe estar registrado en el sistema y no debe ser ingresado en el campo correo | El sistema envía un correo de restauración al correo registrado en el sistema | El sistema envía un correo de restauración al correo registrado en el sistema |  |
| 1.- Ingresar un correo electrónico no registrado en el sistema | Cambio de contraseña correo no registrado | El correo no debe estar registrado | El sistema envía un correo de restauración al correo ingresado | El sistema envía un correo de restauración al correo ingresado |  |

| AREA FUNCIONAL : Modulo generar Actividad (administrador) | | | | | |
| --- | --- | --- | --- | --- | --- |
| Paso | Descripción | Condiciones | Resultado esperado | Resultado obtenido | Observaciones |
| 1. El usuario de tipo Administrador logeado.  2. Despliega el Menu de hamburguesa (superior izquierda).  3. Click en el boton "Generar Proyeto/Actividad" | Boton de menu "Generar proyecto/Actividad" | El usuario debe estar logeado, ademas debe ser un usuario de tipo administrador | Al clickear en "Generar proyecto/Actividad" el usuario es redireccionado e ingresa al modulo de "Generar publicacion - Actividad" | Al clickear en "Generar proyecto/Actividad" el usuario es redireccionado e ingresa al modulo de "Generar publicacion - Actividad" |  |
| 1. Se ingresa al Formulario de generacion de Proyecto/Actividad  2. Se elige el tipo de publicacion a "Actividad"  3. se rellena el campo "titulo de la actividad"  4. se rellena el campo "Descripcion de actividad"  5. se sube una imagen de referencia  6. definir los cupos disponibles  7. definir las fechas de inicio y termino  8. apretar boton "crear actividad" | Rellenar Formulario de generacion de actividad | El usuario Administrador debe estar dentro del modulo de Generar publicacion en el path "/generar/proyecto" | una vez rellenado el formulario de generacion de actividad y apretar el boton "crear actividad" de manera exitosa, deberia sobresalir una caja de texto | una vez rellenado el formulario de generacion de actividad y apretar el boton "crear actividad" de manera exitosa, deberia sobresalir una caja de texto |  |

| AREA FUNCIONAL : Noticias | | | | | |
| --- | --- | --- | --- | --- | --- |
| Paso | Descripción | Condiciones | Resultado esperado | Resultado obtenido | Observaciones |
| 1.- El usuario entra al inicio de la aplicación | El usuario puede ver las noticias de la comunidad | El usuario debe estar autenticado en el sistema y debe estar en el inicio | El usuario puede ver las noticias del sistema | El usuario puede ver las noticias del sistema |  |
| 1.- El usuario entra en el menú lateral a "Gestionar Noticias"  2.- El usuario presiona el botón "Nueva Noticia".  3.- El usuario rellena con los datos obligatorios (Parrafo 1, Imágen de portada).  4.- El usuario presiona el botón "Guardar Noticia". | El administrador puede crear noticias en el sistema | El usuario debe estar autenticado en el sistema como administrador | La noticia se guarda y se muestra en "Gestionar Noticias" y en la página de inicio del sistema | La noticia se guarda y se muestra en "Gestionar Noticias" y en la página de inicio del sistema |  |

| AREA FUNCIONAL : Modulo de inscripcion a Proyecto | | | | | |
| --- | --- | --- | --- | --- | --- |
| Paso | Descripción | Condiciones | Resultado esperado | Resultado obtenido | Observaciones |
| 1. El usuario logeado de tipo vecino  2. Despliega el menu de hamburguesa(Superior izquierda)  3. Realiza click en el boton "Inscribirse a proyecto/actividad" | Boton de menu "inscribirse a proyecto/actividad" | El usuario debe estar logeado, ademas el usuario debe ser de tipo Vecino  . | El usuario ingresa al modulo de "/inscripcion/proyectos" | El usuario ingresa al modulo de "/inscripcion/proyectos" |  |
| 1. El usuario dentro del modulo "Proyectos y actividades"  2. Usuario filtra de "Todos" a "Proyectos"  3. Usuario navega entre los proyectos disponibles  4. Usuario Selecciona con un click el proyecto para inscribirse | Navegacion y seleccion de proyecto a inscribirse | El usuario debe estar dentro del modulo "Proyectos y actividades", en el path /inscripcion/proyectos | El usuario al seleccionar el proyecto de interes, le redirecciona al modulo de inscripcion del proyecto seleccionado | El usuario al seleccionar el proyecto de interes, le redirecciona al modulo de inscripcion del proyecto seleccionado |  |
| 1. el usuario dentro del modulo de Inscripcion de proyecto  2. Usuario rellena la caja "Motivacion"  3. Usuario apreta el boton de "Inscribirse" | Relleno del formulario para postulacion del proyecto | El usuario tiene que haber escogido un proyecto en la lista de navegacion de proyectos disponibles, estar dentro del path /inscripcion/inscripcion-proyecto/... | El usuario al rellenar el cuadro de "motivacion" y apretar el boton "Inscribirse" deberia sobresalir una caja de texto como se muestra en la imagen | El usuario al rellenar el cuadro de "motivacion" y apretar el boton "Inscribirse" deberia sobresalir una caja de texto como se muestra en la imagen |  |

| AREA FUNCIONAL : Perfil | | | | | |
| --- | --- | --- | --- | --- | --- |
| Paso | Descripción | Condiciones | Resultado esperado | Resultado obtenido | Observaciones |
| 1.- El usuario se autentica en el sistema.  2.- El usuario navega hasta "Mi perfil" en el menú lateral.  3.- El usuario presiona "Mi perfil". | El usuario puede ver la información de su perfil | El usuario debe estar registrado en el sistema | El usuario es capáz de ver la información de su perfil. | El usuario es capáz de ver la información de su perfil. |  |
| 1.- El usuario se autentica en el sistema.  2.- El usuario navega hasta "Mi perfíl" en el menú lateral.  3.- El usuario modifica los datos disponibles (Segundo nombre, Segundo apellido, Teléfono).  4.- El usuario presiona el botón "Guardar Cambios". | Editar información del perfíl | El usuario debe estar registrado en el sistema | El usuario es capáz de ver la nueva información de su perfíl. | El usuario es capáz de ver la nueva información de su perfíl. |  |

## 

| AREA FUNCIONAL : Modulo de inscripcion actividad (Vecino) | | | | | |
| --- | --- | --- | --- | --- | --- |
| Paso | Descripción | Condiciones | Resultado esperado | Resultado obtenido | Observaciones |
| 1. El usuario logeado de tipo vecino  2. Despliega el menu de hamburguesa(Superior izquierda)  3. Realiza click en el boton "Inscribirse a proyecto/actividad" | Boton de menu "inscribirse a proyecto/actividad" | El usuario debe estar logeado, ademas el usuario debe ser de tipo Vecino | El usuario ingresa al modulo de "/inscripcion/proyectos" | El usuario ingresa al modulo de "/inscripcion/proyectos" |  |
| 1. El usuario dentro del modulo "Proyectos y actividades"  2. Usuario filtra de "Todos" a "Actividades"  3. Usuario navega entre las actividades disponibles  4. Usuario Selecciona con un click la actividad para inscribirse | Navegacion y seleccion de actividad a inscribirse | El usuario debe estar dentro del modulo "Proyectos y actividades", en el path /inscripcion/proyectos | El usuario al seleccionar la actividad de interes, le redirecciona al modulo de inscripcion del actividad seleccionada | El usuario al seleccionar la actividad de interes, le redirecciona al modulo de inscripcion del actividad seleccionada |  |
| 1. el usuario dentro del modulo de Inscripcion de actividad  2. Usuario rellena la caja "comentario"  3. Usuario apreta el boton de "Inscribirse" | Relleno del formulario para postulacion de la actividad | El usuario tiene que haber escogido una actividad en la lista de navegacion de actividades disponibles, estar dentro del path /inscripcion/inscripcion-proyecto/... | El usuario al rellenar el cuadro de "comentario" y apretar el boton "Inscribirse" deberia sobresalir una caja de texto como se muestra en la imagen | El usuario al rellenar el cuadro de "comentario" y apretar el boton "Inscribirse" deberia sobresalir una caja de texto como se muestra en la imagen |  |

| AREA FUNCIONAL : Espacios | | | | | |
| --- | --- | --- | --- | --- | --- |
| Paso | Descripción | Condiciones | Resultado esperado | Resultado obtenido | Observaciones |
| 1.- El usuario se autentica en el sistema.  2.- El usuario navega a "Espacios Comunitarios" en el menú lateral.  3.- El usuario ingresa a "Espacios Comunitarios" | El usuario puede ver los espacios de la junta de vecinos | El usuario debe estar autenticado en el sistema.  Deben haber espacios creados en el sistema. | El usuario puede ver los espacios de la junta de vecinos | El usuario puede ver los espacios de la junta de vecinos |  |
| 1.- El usuario se autentica como administrador en el sistema.  2.- El usuario navega hasta "Espacios comunitarios" en el menú lateral  3.- El usuario presiona el botón "Espacios comunitarios".  4.- El usuario presiona el botón "Nuevo Espacio".  5.- el usuario rellena el formulario con los datos solicitados.  6.- El usuario presiona el botón "Guardar Espacio". | El administrador puede añadir espacios al sistema | El usuario debe estar autenticado como administrador en el sistema. | El espacio queda visible en el menú "Nuestros Espacios". | El espacio queda visible en el menú "Nuestros Espacios". |  |

| AREA FUNCIONAL : Votación | | | | | |
| --- | --- | --- | --- | --- | --- |
| Paso | Descripción | Condiciones | Resultado esperado | Resultado obtenido | Observaciones |
| 1.- El usuario se autentica en el sistema.  2.- El usuario navega hasta "Votaciones" en el menú lateral.  3.- El usuario despliega las votaciones activas | El usuario puede ver las votaciones activas | El usuario debe estar autenticado en el sistema.  Deben haber votaciones activas en el sistema. | Se visualizan todas las votaciones activas del sistema | Se visualizan todas las votaciones activas del sistema |  |
| 1.- El usuario se autentica en el sistema.  2.- El usuario navega hasta "Votaciones" en el menú lateral.  3.- El usuario despliega las votaciones finalizadas. | El usuario puede ver las votaciones finalizadas | El usuario debe estar autenticado en el sistema.  Deben haber votaciones finalizadas en el sistema. | Se visualizan todas las votaciones finalizadas del sistema | Se visualizan todas las votaciones finalizadas del sistema |  |
| 1.- El usuario se autentica en el sistema con una cuenta administrador.  2.- El usuario navega hasta "Generar votación" en el menú lateral.  3.- El usuario completea el formulario con los datos obligatorios: Título, Fecha Inicio, Fecha Término y Opciones.  4.- El usuario presiona el botón "Crear Votación" | El administrador puede generar votaciones | El usuario debe estar autenticado con una cuenta administrador. | Se visualiza la votación recien creada en el sistema | Se visualiza la votación recien creada en el sistema |  |
| 1.- El usuario se autentica en el sistema.  2.- El usuario navega hasta "Votaciones" en el menú lateral.  3.- El usuario selecciona la votacion de la cual quiere ver detalles. | El usuario puede ver el detalle de las votaciones | El usuario debe estar autenticado en el sistema | El sistema muestra el detalle de la votación incluyendo porcentaje de votos en cada opción, el total de votos y el estado de la votación (tiempo restante o votacion finalizada) | El sistema muestra el detalle de la votación incluyendo porcentaje de votos en cada opción, el total de votos y el estado de la votación (tiempo restante o votacion finalizada) |  |

| AREA FUNCIONAL : Certificados | | | | | |
| --- | --- | --- | --- | --- | --- |
| Paso | Descripción | Condiciones | Resultado esperado | Resultado obtenido | Observaciones |
| 1.- El usuario se autentica en el sistema.  2.- El usuario navega hasta "Solicitar certificado" en el menú lateral.  3.- El usuario ingresa Fecha y lugar (opcional).  4.- El usuario presiona el botón "Generar y enviar por correo". | El usuario puede solicitar certificados de residencia por correo | El usuario debe estar autenticado en el sistema. | El usuario recibe un correo electrónico con el certificado de residencia | El usuario recibe un correo electrónico con el certificado de residencia |  |
| 1.- El usuario se autentica en el sistema.  2.- El usuario navega hasta "Solicitar certificado" en el menú lateral.  3.- El usuario ingresa Fecha y lugar (opcional).  4.- El usuario presiona el botón "Generar y descargar". | El usuario puede solicitar certificados de residencia y descargarlo desde la app | El usuario debe estar autenticado en el sistema | El usuario genera y comienza la descarga del pdf | El usuario genera y comienza la descarga del pdf |  |
| 1.- El usuario se autentica en el sistema.  2.- El usuario navega hasta "Solicitar certificado" en el menú lateral.  3.- El usuario ingresa el correo de destino  4.- El usuario ingresa Fecha y lugar (opcional).  5.- El usuario presiona el botón "Generar enviar por correo". | El usuario puede enviar el certificado a cualquier correo electrónico | El usuario debe estar autenticado en el sistema | El sistema envía el certificado al correo de destino | El sistema envía el certificado al correo de destino |  |
| 1.- El usuario se autentica en el sistema.  2.- El usuario navega hasta "Solicitar certificado" en el menú lateral.  3.- El usuario presiona el botón "Ver PDF" en el certificado que desee. | El usuario puede ver un certificado previamente generado | El usuario debe estar autenticado en el sistema.  Debe haber generado algún certificado previamente. | El sistema muestra el certificado generado anteriormente. | El sistema muestra el certificado generado anteriormente. |  |

| AREA FUNCIONAL : Arrendar un espacio(vecino) | | | | | |
| --- | --- | --- | --- | --- | --- |
| Paso | Descripción | Condiciones | Resultado esperado | Resultado obtenido | Observaciones |
| 1. El usuario una vez logeado, despliega el menu hamburguesa  2. Apretar el boton "Arrendar espacio" | Boton "arrendar espacio" | El usuario debe estar logeado.  El usuario debe ser de tipo Vecino | Cuando se clickea el boton de arrendar espacio, la aplicacion redirige al usuario al modulode /solicitud. Para iniciar el proceso de arrendar un espacio | Cuando se clickea el boton de arrendar espacio, la aplicacion redirige al usuario al modulode /solicitud. Para iniciar el proceso de arrendar un espacio |  |
| 1. El usuario una vez dentro del modulo de solicitud de espacio. rellena los campos solicitados  2. Define la fecha y hora de inicio y termino del espacio  3. Presiona el boton "reservar espacio" | Formulario "Reserva de Espacio/terreno"  El usuario debe ser de tipo Vecino | El usuario debe estar logeado.  El usuario tuvo que haber apretado el boton "arrendar espacio"  El usuario debe ser de tipo Vecino | Una vez rellenado todos los campos del formulario y apretar el boton de "reservar espacio". La aplicacion redirige al usuario al simulador de transbank para procesar el pago del arriendo del espacio | Una vez rellenado todos los campos del formulario y apretar el boton de "reservar espacio". La aplicacion redirige al usuario al simulador de transbank para procesar el pago del arriendo del espacio |  |
| 1. el usuario una vez redirigido, rellena los datos de la tarjeta simulada.  2. se rellena el rut y la clave de simulacion  3. se elige aceptar o rechazar pago, dependiendo de la eleccion la respuesta será pago aceptado o pago rechazado | Simulador de transbank | El usuario debe ser de tipo Vecino.  El usuario debe tener el formulario rellenado. | Una vez hecho el proceso, la pagina deberia mostrar una respuesta del proceso de pago y redirigir a la aplicacion, con un mensaje de pago exitoso o rechazado, como se muestra en la imagen | Una vez hecho el proceso, la pagina deberia mostrar una respuesta del proceso de pago y redirigir a la aplicacion, con un mensaje de pago exitoso o rechazado, como se muestra en la imagen |  |
| 1. Una vez realizado el pago, la simulacion de transbank redirecciona al link de puente para entregar el pago aceptado o rechazado  2. la pagina puente redirecciona a la aplicacion para entregar el mensaje de pago rechazado o pago exitoso. | Redireccion de pago aceptado o rechazado | El usuario debe haber hecho el proceso de pago en la simulacion de transbank | Una vez redireccionado a la aplicacion, debería mostrar un mensaje de pago aprobado o rechazado, como se muestra en la imagen | Una vez redireccionado a la aplicacion, debería mostrar un mensaje de pago aprobado o rechazado, como se muestra en la imagen |  |

## 