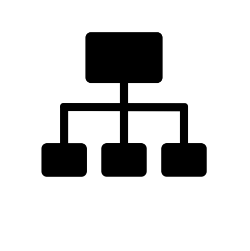
**Documento de Plan de pruebas**

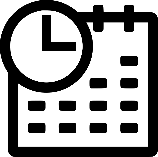
**Cliente - Luis Eduardo Fuertes**

**Sistema - Resvit**

**Versión 1.0**

****

**Organización: BigCode**

****

**Fecha de actualización: 21/10/19**

**Control de versiones**

|  |  |  |
| --- | --- | --- |
| Versión | Fecha | Descripción |
| 1.0 | 18/10/19 | Versión inicial plan de pruebas |
| 1.1 | 20/10/19 | Casos de prueba |
| 1.2 | 21/10/2019 | Casos de prueba, banco de pruebas, iteración de pruebas |

**Lista de distribución y roles**

|  |  |  |
| --- | --- | --- |
| Nombre | Rol – Influencia Plan de pruebas | Organización |
| Elizabeth Moncada | Administrador del proyecto, creador documento Plan de pruebas. | BigCode |
| David Hernández Cárdenas | Desarrollo Base de datos, editor documento Plan de pruebas. | BigCode |
| Cristian Osorio Paz | Desarrollo Backend, probador encargado. | BigCode |
| Camila López | Jefe de solución, descripción casos de prueba. | BigCode |
| Germán Caycedo | Arquitecto de producto, descripción casos de prueba. | BigCode |

**Objetivos del documento**

El documento tiene como objetivo principal la especificación del Plan de pruebas que se llevará a cabo en el proyecto Resvit. Por medio del plan se describirán aspectos tales como:

* Atributos de calidad a ser resaltados.
* Tipos de pruebas a realizar.
* Encargados del desarrollo y despliegue de pruebas.
* Técnicas y estrategias de pruebas a realizar.

|  |  |
| --- | --- |
| Nombre | Descripción |
| Requerimientos del sistema Resvit | Definición de requerimientos del sistema a desarrollar. |

**Documentos Relacionados**

**Requerimientos en software para la realización de pruebas**

**Usuario-Comensal:**

**RT-02 Listado de ciudades:** El sistema debe desplegar un listado de restaurantes disponibles para esa ubicación de los cuales el usuario-comensal podrá escoger.

**RT-03 Listado de restaurantes:** El sistema debe permitir que cada restaurante tenga la opción de estar activo o inactivo, esto determinará si el restaurante aparece en la lista que visualiza el usuario-comensal.

**RT-09 Sección de comentarios:** El sistema debe tener una sección de comentarios para calificar cada uno de los restaurantes.

**RT-10 Comentario de restaurante:** El sistema debe permitir escribir un comentario por parte del usuario-comensal para dar su opinión acerca de un restaurante.

**RT-11 Puntuación de restaurante:** El sistema debe que un usuario-comensal califique un restaurante entre 0 y 5 estrellas.

**RT-17 Sección de puntuación:** El sistema debe tener una sección para puntuar un restaurante entre 0 y 5 estrellas.

**Usuario-Restaurante:**

**RT-04 Visualizar información restaurante:** El sistema debe poseer una interfaz donde el usuario-restaurante pueda visualizar la información de su restaurante.

**RT-05 Modificar información:** El sistema debe permitir editar con facilidad el nombre, tipo de comida, dirección, descripción, teléfono, correo electrónico.

**RT-06 Añadir imágenes:** El sistema debe permitir cargar hasta 8 imágenes de su restaurante.

**RT-07 Eliminar imágenes:** El sistema debe permitir eliminar imágenes siempre y cuando haya imágenes para eliminar.

**RT-14 Historial de reservas:** El sistema debe mostrar al usuario-restaurante un registro de reservas pasadas y actuales de su restaurante.

**RT-16 Agregar mesas:** El sistema debe permitir al usuario-restaurante agregar una mesa con su cantidad de sillas a su restaurante.

**Vistas:**

**AD-01 Clasificación de usuarios:** Los usuarios están clasificados como usuario-comensal, usuario-restaurante y usuario- administrador.

**AD-02 Vista usuario-comensal:**    El sistema debe desplegar la vista del usuario-comensal cuando este inicie sesión.

**AD-03 Vista usuario-restaurante:** El sistema debe desplegar la vista del usuario-restaurante cuando este inicie sesión.

**Atributos de Calidad del Software a ser probados**

Los atributos de calidad a ser probados principalmente son los siguientes:

1. **Operabilidad:** Es la capacidad del software de permitir al usuario manejarlo y controlarlo de manera sencilla.
2. **Modularidad:** Capacidad del software que permite que un cambio en un componente tenga un impacto mínimo en los demás.
3. **Confidencialidad**: Capacidad de protección contra el acceso de datos e información no autorizados, ya sea accidental o deliberadamente.
4. **Disponibilidad:** Capacidad de la aplicación o componente de estar operativo.

**Tipos de Pruebas**

Se realizarán dos tipos de pruebas que se subdividen de la siguiente manera:

1. **Pruebas funcionales**
   1. **Aceptación:** Consiste en verificar la aceptación del sistema por parte del usuario.
   2. **Regresión:** Consiste en descubrir errores funcionales y del comportamiento, causados por algún cambio en el sistema.
2. **Pruebas no funcionales**
   1. **Estrés:** Consiste en generar una gran carga sobre la aplicación y llevarla hasta su límite, es decir hasta que falle para poder establecer el límite de la realidad en la aplicación.
   2. **Ejecución:** Consiste en verificar que la aplicación alcance un nivel esperado en producción.
   3. **Funcional:** Consiste en probar que el software funcione de la manera que debería, es decir que funcione según lo esperado.

**Estrategias de Prueba**

Se ejecutarán los siguientes tipos de pruebas, haciendo énfasis en que el probador también cuenta con el acceso al código fuente:

1. **Caja negra:** Consiste observar el comportamiento del software al parametrizar datos de entrada para cada componente y esperando una respuesta que permitirá la verificación del cumplimiento de cada requerimiento. No se tiene en cuenta el funcionamiento interno del código, solo resultados.
2. **Caja blanca:** Consiste en observar el comportamiento del software al parametrizar datos de entrada para cada componente y esperando una respuesta que permitirá la verificación del cumplimiento de cada requerimiento. Se tiene en cuenta la traza que toma el dato al momento de ser ingresado para así verificar el criterio de falla en caso de que suceda.

**Técnicas de Pruebas**

Con ayuda de la herramienta Selenium se harán uso de técnicas de pruebas como:

1. **Pruebas de volumen:** Creación de tipos de datos de pruebas específicos, que permiten evaluar cuando se alcanzan o se exceden los límites del sistema.
2. **Análisis de control de datos:** Asegura que las variables, estructuras, librerías, etc., usadas han sido definidas apropiadamente (tipos, estándares), y usadas adecuadamente.
3. **Prueba de desastre:** Simulaciones de fallas operacionales o de sistema, que sirven como base para probar el proceso de recuperación del sistema.

**Descomposición Funcional**

**Imagen que contiene captura de pantalla

Descripción generada automáticamente**

**Banco de pruebas**

Se hará uno de una matriz de trazabilidad como banco de pruebas:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Id Condición de prueba | Casos de prueba | Requerimiento | Regla de negocio | Requerimiento anterior | Documento técnico | Otro documento |
| CondP01 | CP0001\_ListadoCiudades | **RT-02** | **Requerimiento funcional RT-02** | **NA** | Requerimientos del sistema Resvit | **NA** |
| CondP02 | CP0002\_ListadoRestaurante | **RT-03** | **Requerimiento funcional RT-03** | **NA** | Requerimientos del sistema Resvit | **NA** |
| CondP03 | CP0003\_InformacionRestaurante | **RT-04** | **Requerimiento funcional RT-04** | **NA** | Requerimientos del sistema Resvit | **NA** |
| CondP04 | CP0004\_ModificarInformacion | **RT-05** | **Requerimiento funcional RT-05** | **NA** | Requerimientos del sistema Resvit | **NA** |
| CondP05 | CP0005\_AñadirImagenes | **RT-06** | **Requerimiento funcional RT-06** | **NA** | Requerimientos del sistema Resvit | **NA** |
| CondP06 | CP0006\_EliminarImagenes | **RT-07** | **Requerimiento funcional RT-07** | **NA** | Requerimientos del sistema Resvit | **NA** |
| CondP07 | CP0007\_Comentarios | **RT-09, RT-10** | **Requerimiento funcional RT-09 y RT-10** | **NA** | Requerimientos del sistema Resvit | **NA** |
| CondP08 | CP0008\_Puntuaciones | **RT-11, RT-17** | **Requerimiento funcional RT-11 y RT-17** | **NA** | Requerimientos del sistema Resvit | **NA** |
| CondP09 | CP0009\_HistorialReservas | **RT-14** | **Requerimiento funcional RT-14** | **NA** | Requerimientos del sistema Resvit | **NA** |
| CondP10 | CP0010\_AgregarMesa | **RT-16** | **Requerimiento funcional RT-16** | **NA** | Requerimientos del sistema Resvit | **NA** |
| CondP11 | CP0011\_VistaDesplegada | **AD-01, AD-02, AD-03** | **Requerimiento funcional AD-01, AD-02 y AD-03** | **NA** | Requerimientos del sistema Resvit | **NA** |

**Casos de prueba**

Especificación de los diferentes casos de prueba por cada requerimiento funcional definido previamente:

|  |  |  |  |
| --- | --- | --- | --- |
| Caso de prueba | Validación de listado de ciudades inicial. | | |
| Identificador caso de prueba | CP0001\_ListadoCiudades | | |
| Nomenclatura del nombre definida: CP0001\_ListadoCiudades | | |
| Función probar | Se probará el listado de las diferentes ciudades del usuario-comensal. | | |
| Objetivo | Se visualizarán las diferentes ciudades que se encuentren con restaurantes registrados en el sistema. | | |
| Descripción | El usuario-comensal visualizará los diferentes restaurantes registrados para una ciudad escogida. | | |
| Criterios de éxito | El sistema evidencia el despliegue de los diferentes restaurantes registrados para una ciudad escogida. | | |
| Criterios de falla | El sistema no evidencia el despliegue de los diferentes restaurantes registrados para una ciudad escogida. | | |
| Precondiciones | Se debe tener acceso al sistema por medio de una cuenta existente para un usuario-comensal y haber seleccionado una ciudad del listado de ciudades de la ventana de inicio. | | |
| Perfil del usuario | Usuario-comensal. | | |
| Necesidades para el caso de prueba | Se debe haber ingresado al sistema por medio de una cuenta única asignada a un usuario-comensal. Posterior a esto, visualizar las diferentes ciudades que se encuentren con restaurantes registrados en el sistema. Finalmente, seleccionar una de las ciudades del listado. | | |
| Autor | BigCode. | | |
| Fecha de creación | 18/10/19 | | |
| Flujo del caso de prueba | N° paso | Usuario del sistema | Sistema |
| 1. Ingreso al sistema como usuario-comensal. 2. Visualización de ciudades registradas en el sistema.  3. Seleccionar una ciudad del listado desplegado. | Selección de una ciudad desplegada en el listado. | Despliegue de los restaurantes registrados en el sistema para la ciudad seleccionada. |
| Post condiciones | El sistema evidenció el despliegue de los diferentes restaurantes registrados para una ciudad escogida. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Caso de prueba | Validación de listado de ciudades inicial. | | |
| Identificador caso de prueba | CP0002\_ListadoRestaurante | | |
| Nomenclatura del nombre definida: CP0002\_ListadoRestaurante | | |
| Función probar | Se probará las opciones activo/inactivo del usuario-restaurante que determina la visualización del este en el usuario-comensal. | | |
| Objetivo | Se visualizarán los diferentes restaurantes que estén activos. | | |
| Descripción | El usuario-restaurante probará las opciones activo/inactivo y el usuario-comensal verificará la visualización del restaurante. | | |
| Criterios de éxito | El sistema evidencia el funcionamiento de las opciones activo/inactivo de cada restaurante. | | |
| Criterios de falla | El sistema no evidencia el funcionamiento de las opciones activo/inactivo de cada restaurante. | | |
| Precondiciones | Se debe tener acceso al sistema por medio de una cuenta existente para un usuario-restaurante y haber seleccionado una de las opciones activo/inactivo.  Además, se debe tener acceso al sistema por medio de una cuenta existente para un usuario-comensal y haber seleccionado una ciudad del listado de ciudades de la ventana de inicio. | | |
| Perfil del usuario | Usuario-comensal y usuario-restaurante | | |
| Necesidades para el caso de prueba | Se debe haber ingresado al sistema por medio de una cuenta única asignada a un usuario-restaurante, ingresar al perfil y seleccionar una de las opciones activo/inactivo.  Además, se debe haber ingresado al sistema por medio de una cuenta única asignada a un usuario-comensal, visualizar las diferentes ciudades que se encuentren con restaurantes registrados en el sistema y seleccionar una de las ciudades del listado. | | |
| Autor | BigCode. | | |
| Fecha de creación | 20/10/19 | | |
| Flujo del caso de prueba | N° paso | Usuario del sistema | Sistema |
| 1. Ingreso al sistema como usuario-restaurante. 2. Seleccionar una de las opciones activo/inactivo.  3. Ingreso al sistema como usuario-restaurante.  4. Visualización de ciudades registradas en el sistema.  5. Seleccionar una ciudad del listado desplegado. | Restaurante, selección de una de las opciones activo/inactivo.  Comensal, selección de una ciudad desplegada en el listado. | Despliegue de los restaurantes registrados en el sistema para la ciudad seleccionada que previamente activaron la opción *activo.* |
| Post condiciones | El sistema evidenció el funcionamiento de las opciones activo/inactivo de cada restaurante. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Caso de prueba | Validación de listado de ciudades inicial. | | |
| Identificador caso de prueba | CP0003\_InformacionRestaurante | | |
| Nomenclatura del nombre definida: CP0003\_InformacionRestaurante | | |
| Función probar | Se probará la visualización de la información del restaurante desde la vista usuario-restaurante, tales como: fotos, nombre, dirección, descripción, teléfono, correo electrónico, comentarios de comensales, puntuación. | | |
| Objetivo | Se visualizarán la información de cada restaurante desde la vista usuario-restaurante, tales como: fotos, nombre, dirección, descripción, teléfono, correo electrónico, comentarios de comensales, puntuación. | | |
| Descripción | El usuario-restaurante probará ver la información de su propio restaurante, tales como: fotos, nombre, dirección, descripción, teléfono, correo electrónico, comentarios de comensales, puntuación. | | |
| Criterios de éxito | El sistema evidencia la visualización de la información del restaurante, tales como: fotos, nombre, dirección, descripción, teléfono, correo electrónico, comentarios de comensales, puntuación. | | |
| Criterios de falla | El sistema no evidencia la visualización de la información del restaurante, tales como: fotos, nombre, dirección, descripción, teléfono, correo electrónico, comentarios de comensales, puntuación. | | |
| Precondiciones | Se debe tener acceso al sistema por medio de una cuenta existente para un usuario-restaurante y haber seleccionado el perfil. | | |
| Perfil del usuario | Usuario-restaurante | | |
| Necesidades para el caso de prueba | Se debe haber ingresado al sistema por medio de una cuenta única asignada a un usuario-restaurante y seleccionar el perfil. | | |
| Autor | BigCode. | | |
| Fecha de creación | 20/10/19 | | |
| Flujo del caso de prueba | N° paso | Usuario del sistema | Sistema |
| 1. Ingreso al sistema como usuario-restaurante. 2. Seleccionar el perfil. | Selección del perfil. | Visualización de la información correspondiente, tales como: fotos, nombre, dirección, descripción, teléfono, correo electrónico, comentarios de comensales, puntuación. |
| Post condiciones | El sistema evidenció la visualización de la información del restaurante, tales como: fotos, nombre, dirección, descripción, teléfono, correo electrónico, comentarios de comensales, puntuación. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Caso de prueba | Validación de listado de ciudades inicial. | | |
| Identificador caso de prueba | CP0004\_ModificarInformacion | | |
| Nomenclatura del nombre definida: CP0004\_ModificarInformacion | | |
| Función probar | Se probará la edición de la información del restaurante desde la vista usuario-restaurante, tales como: nombre, dirección, descripción, teléfono, correo electrónico. | | |
| Objetivo | Se probará la edición efectiva de la información de cada restaurante desde la vista usuario-restaurante, tales como: nombre, dirección, descripción, teléfono, correo electrónico. | | |
| Descripción | El usuario-restaurante probará editar la información de su propio restaurante, tales como: nombre, dirección, descripción, teléfono, correo electrónico. | | |
| Criterios de éxito | El sistema evidencia la edición efectiva de información del restaurante, tales como: fotos, nombre, dirección, descripción, teléfono, correo electrónico. | | |
| Criterios de falla | El sistema no evidencia la edición efectiva de información del restaurante, tales como: fotos, nombre, dirección, descripción, teléfono, correo electrónico. | | |
| Precondiciones | Se debe tener acceso al sistema por medio de una cuenta existente para un usuario-restaurante y haber seleccionado el perfil. | | |
| Perfil del usuario | Usuario-restaurante | | |
| Necesidades para el caso de prueba | Se debe haber ingresado al sistema por medio de una cuenta única asignada a un usuario-restaurante, seleccionar el perfil y editar los campos deseados. | | |
| Autor | BigCode. | | |
| Fecha de creación | 20/10/19 | | |
| Flujo del caso de prueba | N° paso | Usuario del sistema | Sistema |
| 1. Ingreso al sistema como usuario-restaurante. 2. Seleccionar el perfil. | Editar la información del perfil. | Edición efectiva de la información correspondiente, tales como: nombre, dirección, descripción, teléfono, correo electrónico. |
| Post condiciones | El sistema evidenció la edición efectiva de información del restaurante, tales como: fotos, nombre, dirección, descripción, teléfono, correo electrónico. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Caso de prueba | Validación de listado de ciudades inicial. | | |
| Identificador caso de prueba | CP0005\_AñadirImagenes | | |
| Nomenclatura del nombre definida: CP0005\_AñadirImagenes | | |
| Función probar | Se probará la carga de imágenes del restaurante desde la vista usuario-restaurante. | | |
| Objetivo | Se probará la carga efectiva de imágenes de cada restaurante desde la vista usuario-restaurante. | | |
| Descripción | El usuario-restaurante probará cargar imágenes de su propio restaurante. | | |
| Criterios de éxito | El sistema evidencia la carga de imágenes del restaurante. | | |
| Criterios de falla | El sistema no evidencia la carga de imágenes del restaurante. | | |
| Precondiciones | Se debe tener acceso al sistema por medio de una cuenta existente para un usuario-restaurante, posterior a eso seleccionar el perfil y finalmente pulsar en el botón de carga de imágenes. | | |
| Perfil del usuario | Usuario-restaurante | | |
| Necesidades para el caso de prueba | Se debe haber ingresado al sistema por medio de una cuenta única asignada a un usuario-restaurante, seleccionar el perfil y pulsar en el botón de carga de imágenes. | | |
| Autor | BigCode. | | |
| Fecha de creación | 20/10/19 | | |
| Flujo del caso de prueba | N° paso | Usuario del sistema | Sistema |
| 1. Ingreso al sistema como usuario-restaurante. 2. Seleccionar el perfil. | Cargar una imagen. | Carga efectiva de las imágenes. |
| Post condiciones | El sistema evidenció la carga de imágenes del restaurante. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Caso de prueba | Validación de listado de ciudades inicial. | | |
| Identificador caso de prueba | CP0006\_EliminarImagenes | | |
| Nomenclatura del nombre definida: CP0006\_EliminarImagenes | | |
| Función probar | Se probará la eliminación de imágenes del restaurante desde la vista usuario-restaurante. | | |
| Objetivo | Se probará la eliminación efectiva de imágenes de cada restaurante desde la vista usuario-restaurante. | | |
| Descripción | El usuario-restaurante probará eliminar imágenes de su propio restaurante. | | |
| Criterios de éxito | El sistema evidencia la eliminación de imágenes del restaurante. | | |
| Criterios de falla | El sistema no evidencia la eliminación de imágenes del restaurante. | | |
| Precondiciones | Se debe tener acceso al sistema por medio de una cuenta existente para un usuario-restaurante, posterior a eso seleccionar el perfil y finalmente seleccionar la imagen que desea eliminar. | | |
| Perfil del usuario | Usuario-restaurante | | |
| Necesidades para el caso de prueba | Se debe haber ingresado al sistema por medio de una cuenta única asignada a un usuario-restaurante, seleccionar el perfil y seleccionar la imagen que desea eliminar. | | |
| Autor | BigCode. | | |
| Fecha de creación | 20/10/19 | | |
| Flujo del caso de prueba | N° paso | Usuario del sistema | Sistema |
| 1. Ingreso al sistema como usuario-restaurante. 2. Seleccionar el perfil. | Seleccionar una imagen a eliminar. | Eliminación efectiva de las imágenes. |
| Post condiciones | El sistema evidenció la eliminación de imágenes del restaurante. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Caso de prueba | Validación de listado de ciudades inicial. | | |
| Identificador caso de prueba | CP0007\_Comentarios | | |
| Nomenclatura del nombre definida: CP0007\_Comentarios | | |
| Función probar | Se probará la visualización y creación efectiva de un comentario realizado por un usuario-comensal. | | |
| Objetivo | Se probará poder realizar comentarios por parte de un usuario-comensal a un restaurante. | | |
| Descripción | El usuario-comensal probará escribir un comentario sobre su experiencia en el restaurante. | | |
| Criterios de éxito | El sistema evidencia la creación correcta del comentario por parte del usuario-comensal a un restaurante. | | |
| Criterios de falla | El sistema no evidencia la creación correcta del comentario por parte del usuario-comensal a un restaurante. | | |
| Precondiciones | Se debe tener acceso al sistema por medio de una cuenta existente para un usuario-restaurante, seleccionar una ciudad, seleccionar un restaurante, y finalmente realizar una reserva. | | |
| Perfil del usuario | Usuario-comensal | | |
| Necesidades para el caso de prueba | Se debe haber ingresado al sistema por medio de una cuenta única asignada a un usuario-comensal, seleccionar una ciudad, seleccionar un restaurante, realizar una reserva y finalmente escribir en la sección de comentarios cuando la reserva se cumpla. | | |
| Autor | BigCode. | | |
| Fecha de creación | 21/10/19 | | |
| Flujo del caso de prueba | N° paso | Usuario del sistema | Sistema |
| 1. Ingreso al sistema como usuario-restaurante. 2. Seleccionar una ciudad del listado de ciudades.  3. Seleccionar un restaurante.  4. Realizar una reserva en el restaurante seleccionado.  5. Cuando la reserva se cumpla podrá escribir un comentario en la sección de comentarios. | Escribir un comentario en la sección de comentarios del restaurante donde reservó. | Creación efectiva del comentario escrito por el usuario. |
| Post condiciones | El sistema evidenció la creación correcta del comentario por parte del usuario-comensal a un restaurante. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Caso de prueba | Validación de listado de ciudades inicial. | | |
| Identificador caso de prueba | CP0008\_Puntuaciones | | |
| Nomenclatura del nombre definida: CP0008\_Puntuaciones | | |
| Función probar | Se probará la visualización y creación efectiva de la puntuación de 0 a 5 realizada por un usuario-comensal. | | |
| Objetivo | Se probará poder realizar puntuaciones por parte de un usuario-comensal a un restaurante. | | |
| Descripción | El usuario-comensal probará puntuar de 0 a 5 su experiencia en el restaurante. | | |
| Criterios de éxito | El sistema evidencia la creación correcta de la puntuación realizada por parte del usuario-comensal a un restaurante. | | |
| Criterios de falla | El sistema no evidencia la creación correcta de la puntuación realizada por parte del usuario-comensal a un restaurante. | | |
| Precondiciones | Se debe tener acceso al sistema por medio de una cuenta existente para un usuario-restaurante, seleccionar una ciudad, seleccionar un restaurante, y finalmente realizar una reserva. | | |
| Perfil del usuario | Usuario-comensal | | |
| Necesidades para el caso de prueba | Se debe haber ingresado al sistema por medio de una cuenta única asignada a un usuario-comensal, seleccionar una ciudad, seleccionar un restaurante, realizar una reserva y finalmente puntuar cuando la reserva se cumpla. | | |
| Autor | BigCode. | | |
| Fecha de creación | 21/10/19 | | |
| Flujo del caso de prueba | N° paso | Usuario del sistema | Sistema |
| 1. Ingreso al sistema como usuario-restaurante. 2. Seleccionar una ciudad del listado de ciudades.  3. Seleccionar un restaurante.  4. Realizar una reserva en el restaurante seleccionado.  5. Cuando la reserva se cumpla podrá puntuar en la sección de puntuación. | Puntuar en la sección de puntuación del restaurante donde reservó. | Creación efectiva de la puntuación de 0 a 5 realizada por el usuario. |
| Post condiciones | El sistema evidenció la creación correcta de la puntuación realizada por parte del usuario-comensal a un restaurante. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Caso de prueba | Validación de listado de ciudades inicial. | | |
| Identificador caso de prueba | CP0009\_HistorialReservas | | |
| Nomenclatura del nombre definida: CP0009\_ HistorialReservas | | |
| Función probar | Se probará la visualización de las reservas pasadas y actuales del usuario-restaurante. | | |
| Objetivo | Se probará poder ver las reservas realizadas por los usuarios en el usuario-restaurante. | | |
| Descripción | El usuario-restaurante probará visualizar sus reservas previas y actuales. | | |
| Criterios de éxito | El sistema evidencia la visualización de las reservas previas y actuales del usuario-restaurante. | | |
| Criterios de falla | El sistema no evidencia la visualización de las reservas previas y actuales del usuario-restaurante. | | |
| Precondiciones | Se debe tener acceso al sistema por medio de una cuenta existente para un usuario-restaurante y haber seleccionado el historial. | | |
| Perfil del usuario | Usuario-restaurante | | |
| Necesidades para el caso de prueba | Se debe haber ingresado al sistema por medio de una cuenta única asignada a un usuario-restaurante y seleccionar la sección del historial. | | |
| Autor | BigCode. | | |
| Fecha de creación | 21/10/19 | | |
| Flujo del caso de prueba | N° paso | Usuario del sistema | Sistema |
| 1. Ingreso al sistema como usuario-restaurante. 2. Seleccionar el historial. | Seleccionar el historial. | Visualizar las reservas pasadas y actuales. |
| Post condiciones | El sistema evidenció la visualización de las reservas previas y actuales del usuario-restaurante. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Caso de prueba | Validación de listado de ciudades inicial. | | |
| Identificador caso de prueba | CP0010\_AgregarMesa | | |
| Nomenclatura del nombre definida: CP0010\_AgregarMesa | | |
| Función probar | Se probará la creación de mesas con sus respectivas sillas del usuario-restaurante. | | |
| Objetivo | Se probará crear mesas y sillas por el usuario-restaurante. | | |
| Descripción | El usuario-restaurante probará crear las meas y las sillas de su restaurante par las reservas. | | |
| Criterios de éxito | El sistema evidencia la creación de mesas y las sillas respectivas para las reservas. | | |
| Criterios de falla | El sistema no evidencia la creación de mesas y las sillas respectivas para las reservas. | | |
| Precondiciones | Se debe tener acceso al sistema por medio de una cuenta existente para un usuario-restaurante y haber seleccionado las mesas. | | |
| Perfil del usuario | Usuario-restaurante | | |
| Necesidades para el caso de prueba | Se debe haber ingresado al sistema por medio de una cuenta única asignada a un usuario-restaurante y seleccionar la sección de las reservas. | | |
| Autor | BigCode. | | |
| Fecha de creación | 21/10/19 | | |
| Flujo del caso de prueba | N° paso | Usuario del sistema | Sistema |
| 1. Ingreso al sistema como usuario-restaurante. 2. Seleccionar las reservas. | Seleccionar las reservas. | Visualizar las reservas pasadas y actuales. |
| Post condiciones | El sistema evidenció la creación de mesas y las sillas respectivas para las reservas. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Caso de prueba | Validación de listado de ciudades inicial. | | |
| Identificador caso de prueba | CP0011\_VistaDesplegada | | |
| Nomenclatura del nombre definida: CP0011\_VistaDesplegada | | |
| Función probar | Se probará el despliegue de la vista correspondiente a los usuarios: restaurante y comensal. | | |
| Objetivo | Se probará la visualización de la vista comensal o restaurante según corresponda quién inicie sesión. | | |
| Descripción | El usuario-restaurante y el usuario-comensal visualizarán las interfaces correspondientes a cada uno cuando inicie sesión. | | |
| Criterios de éxito | El sistema evidencia la visualización de las interfaces correspondientes a los usuarios: restaurante y comensal, cuando inicien sesión. | | |
| Criterios de falla | El sistema no evidencia la visualización de las interfaces correspondientes a los usuarios: restaurante y comensal, cuando inicien sesión. | | |
| Precondiciones | Se debe tener acceso al sistema por medio de una cuenta existente para un usuario-restaurante/usuario-comensal y haber iniciado sesión. | | |
| Perfil del usuario | Usuario-restaurante y usuario-comensal | | |
| Necesidades para el caso de prueba | Se debe haber ingresado al sistema por medio de una cuenta única asignada a un usuario-restaurante/usuario-comensal. | | |
| Autor | BigCode. | | |
| Fecha de creación | 21/10/19 | | |
| Flujo del caso de prueba | N° paso | Usuario del sistema | Sistema |
| 1. Ingreso al sistema como usuario-restaurante/usuario-comensal. | Iniciar sesión. | Visualizar las interfaces que corresponden al usuario que inició sesión. |
| Post condiciones | El sistema evidenció la visualización de las interfaces correspondientes a los usuarios: restaurante y comensal, cuando inicien sesión. | | |

**Iteración de pruebas**

Definición del calendario semanal para la realización de los casos de prueba propuestos en la matriz de trazabilidad:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Id Condición de prueba | Semanas | | | | | |
|  | **11** | **12** | **13** | **14** | **15** | **16** |
| CondP01 |  |  |  |  |  |  |
| CondP02 |  |  |  |  |  |  |
| CondP03 |  |  |  |  |  |  |
| CondP04 |  |  |  |  |  |  |
| CondP05 |  |  |  |  |  |  |
| CondP06 |  |  |  |  |  |  |
| CondP07 |  |  |  |  |  |  |
| CondP08 |  |  |  |  |  |  |
| CondP09 |  |  |  |  |  |  |
| CondP10 |  |  |  |  |  |  |
| CondP11 |  |  |  |  |  |  |