

## **Plan de Pruebas**

**Versión <1.0>**

	Versión: <1.0>
Plan de Pruebas	Fecha: <15/12/2022>
PPS2G	

## Histórico de revisiones

Fecha	Versión	Descripción	Autor
<15/12/2022>	<1.0>	Creación del documento	Equipo Sándwich

	Versión: <1.0>
Plan de Pruebas	Fecha: <15/12/2022>
PPS2G	

# Índice

1.	Introducción	1
1.1	Objetivos	1
1.2	Ámbito	1
1.3	Referencias	2
2.	Elementos de prueba	2
3.	Características que se deben probar	2
4.	Características que no se van a probar	3
5.	Aproximación	3
5.1	Pruebas funcionales	3
5.2	Pruebas de carga	3
5.3	Pruebas de rendimiento	4
5.4	Pruebas de seguridad	<b>iError! Marcador no definido.</b>
6.	Criterios para decidir si pasa/falla un elemento	4
7.	Criterios de suspensión y reanudación	4
7.1	Suspensión	5
7.2	Reanudación	5
8.	Pruebas que se deben entregar	5
9.	Tareas de prueba	6
10.	Necesidades ambientales	7
11.	Responsabilidades.	7
12.	Aprobación	8

	Versión: <1.0>
Plan de Pruebas	Fecha: <15/12/2022>
PPS2G	

# Plan de Pruebas

## 1. Introducción

Este es el plan maestro de pruebas para el proyecto de Sandwich2Go.

Este plan abarca solo aquellos artículos y elementos que se relacionan con Sandwich2Go. Se afrontarán los elementos afectados tanto directa como indirectamente.

El primer enfoque de este plan es garantizar que la aplicación Sandwich2Go proporcione el mismo nivel de información y detalle que los acordados en los casos de uso, al tiempo que permite mejoras y aumentos en la adquisición de datos y el nivel de detalle disponible.

El proyecto tendrá 2 niveles de prueba, pruebas unitarias, y pruebas funcionales. Los detalles para cada una de las pruebas serán definidos en los documentos específicos para cada nivel.

El tiempo estimado para este proyecto es de 4 meses, por lo que algún retraso en el proceso de desarrollo o en la instalación. La instalación de software de terceros podría tener un efecto negativo en las pruebas.

### 1.1 Objetivos

El principal objetivo de las pruebas descritas en este documento es comprobar que la aplicación cumple con los Casos de Uso establecidos al principio del desarrollo.

En consecuencia, durante la elaboración de estas pruebas, debemos establecer otros objetivos, como encontrar y solucionar errores en el software.

Otro de los objetivos principales es comprobar que nuestro software cumple con un mínimo de requisitos, garantizando una calidad para el usuario final de la aplicación

### 1.2 Ámbito

Nuestro ámbito de pruebas será la aplicación Sandwich2Go, que consiste en una aplicación con dos partes:

Una parte para los clientes, que le permitirá hacer un pedido con unos sándwiches predefinidos, o crear su propio sándwich y pedirlo.

La otra parte será la visión de la aplicación del gerente, donde podrá crear nuevas ofertas para los sándwiches, o realizar un pedido para reabastecer el stock del restaurante.

	Versión: <1.0>
Plan de Pruebas	Fecha: <15/12/2022>
PPS2G	

### 1.3 Referencias

Para poder hacer las pruebas, el equipo se ha basado en los documentos de Casos de Uso, los cuales, para poder aceptar la implementación, debe cumplir con todos los requisitos mencionados en el mismo, funcionando correctamente.

Los cuatro casos de uso son los siguientes:

- Caso de uso Comprar Sándwich
- Caso de uso Crear Sándwich
- Caso de Uso Crear Oferta
- Caso de Uso Pedido Proveedor

Estos casos de prueba podrán encontrarse en la carpeta "Guías Casos de Uso" en el Design del proyecto.

## 2. Elementos de prueba

Los elementos que probaremos serán las diferentes clases que han sido necesarias para implementar los casos de uso son los siguientes:

- Las diferentes clases necesarias para almacenar la información.
- La base de datos creada a partir de las clases
- Los controladores necesarios para transferir información de, y a las vistas.
- Las diferentes vistas proporcionadas al usuario final
- Los diferentes View Model, que actúan como puente entre las vistas y el controlador.

## 3. Características que se deben probar

A continuación, vamos a enumerar las características que se deben probar de nuestro software:

- A. Filtrados en las listas de selección.
- B. Los elementos obligatorios que debe introducir el usuario no pueden tener valor nulo.
- C. La información que se muestra se tiene que corresponder con la introducida previamente.
- D. La autenticación de usuarios debe funcionar correctamente denegando el acceso a usuarios no autorizados para esa sección.
- E. Comprobar el correcto funcionamiento del flujo principal de los casos de uso.

	Versión: <1.0>
Plan de Pruebas	Fecha: <15/12/2022>
PPS2G	

#### 4. Características que no se van a probar

Vamos a enumerar las características que no se deben probar y su motivo:

- A. Cantidad de solicitudes simultaneas que puede recibir la web debido a que no es una web pública.
- B. La disponibilidad de la web debido a que accederán a la web solo cuando sea necesario.
- C. No se comprobará que el método de pago funcione de forma realista debido a que necesitaríamos añadir nuevas funcionalidades a nuestro software para comprobarlo.
- D. Los tiempos de repuesta de nuestro software.

#### 5. Aproximación

Tenemos diferentes pruebas en el sistema, para la cual hacemos una descripción:

Describe la aproximación que se va a utilizar: quien la hace, actividades principales, técnicas y herramientas para cada grupo/característica. ¿Cómo se decidirá si un grupo de características se ha probado satisfactoriamente?

##### 5.1 Pruebas unitarias

Para las pruebas unitarias, las realizará el propio programador del caso de uso, y consistirán en unas pruebas de caja blanca, donde el programador introduce unos datos que espera encontrar cuando ejecuta sus controladores, y busca los resultados producidos son iguales a los esperados.

Si todos los resultados esperados coinciden con los resultados producidos, aceptaremos las pruebas.

##### 5.2 Pruebas funcionales

Para las pruebas funcionales, cada caso de uso le será asignado a un tester, el cual realizará unas pruebas de caja negra, pues no tiene por qué conocer realmente como han sido implementados los casos de uso.

Establecerá salidas del programa esperadas, y si coinciden con las reales, las pruebas serán aceptadas.

	Versión: <1.0>
Plan de Pruebas	Fecha: <15/12/2022>
PPS2G	

### 5.3 Pruebas de seguridad

Respecto a las pruebas de seguridad, cada programador del caso de uso será el responsable de controlar quien puede acceder a sus vistas.

Esto será aprobado por el tester, que tendrá que comprobar la correcta asignación de los Roles a los controladores.

Si estas asignaciones coinciden con el Caso de Uso, pasaremos las pruebas.

### 5.4 Pruebas de rendimiento

No aplicable

## 6. Criterios para decidir si pasa/falla un elemento

El conjunto de criterios en los que nos hemos basado para decidir si un elemento pasa o falla es el siguiente:

En primer lugar, hemos depurado cada prueba para comprobar en cada paso que selecciona las cosas que se desean.

También, hemos ejecutado de manera individual por un lado cada prueba y después de manera conjunta todas para ver si se ejecutaban correctamente, así podemos decidir también que dichas pruebas pasan.

Por el contrario, a estos dos criterios citados previamente, si alguno de los pasos no coge el elemento que deseamos o ni siquiera coge ningún valor (es nulo), se podría decir que ese elemento falla.

## 7. Criterios de suspensión y reanudación

En este apartado vamos a describir el procedimiento a realizar cuando debamos suspender la realización de las pruebas por algún fallo.

	Versión: <1.0>
Plan de Pruebas	Fecha: <15/12/2022>
PPS2G	

## 7.1 Suspensión

Suspenderemos la realización de pruebas cuando encontremos algún fallo como testers en alguna implementación de algún caso de uso. El protocolo que seguir será el siguiente:

- Crearemos un "Work Item" del tipo "Bug".
- Documentaremos el fallo, con una trazabilidad para llegar a reproducirlo.
- Le asignaremos el "Work Item" al programador del caso de uso con el fallo.

## 7.2 Reanudación

Para poder reanudar la implementación de las pruebas, debe ocurrir lo siguiente:

- El programador debe marcar su "Work Item" como abierto.
- El programador debe corregir el error en el software.
- Debe marcar su "Work Item" como Resolved y asignárselo al Tester.
- El Tester comprobará si realmente ha sido resuelto. Si no ha sido resuelto, se volverá a asignar al programador y marcar como Active. En caso de estar resuelto, se cerrará el Work Item, y seguirá implementando las pruebas.

## 8. Pruebas que se deben entregar

Para este software debemos entregar lo siguiente:

- Documentos de los casos de uso
- Plan de pruebas informe de pruebas
- Los casos de prueba que hemos realizado para documentar las pruebas funcionales de cada caso de uso.
- La implementación de dichas pruebas en el proyecto.



	Versión: <1.0>
Plan de Pruebas	Fecha: <15/12/2022>
PPS2G	

## 9. Tareas de prueba

Para comprobar el correcto funcionamiento de nuestro software hemos tenido que realizar pruebas unitarias y pruebas funcionales.

En las pruebas unitarias cada integrante del grupo ha comprobado que el caso de uso que ha realizado funciona correctamente.

Marcos González Ruiz ha realizado las pruebas unitarias del caso de uso Crear oferta.

Sandra Mendoza Martínez ha realizado las pruebas unitarias del caso de uso Crear sándwich.

Alonso Illán Martínez del Santo ha realizado las pruebas unitarias del caso de uso Comprar sándwich.

Alberto López Hernández ha realizado las pruebas unitarias del caso de uso Crear pedido proveedor.

Las pruebas unitarias consisten en probar el correcto funcionamiento de los controladores y de los View Models creados.

Las pruebas se dividen en tres partes:

- Select
- Create
- Details

Para la realización de las pruebas funcionales a cada integrante del grupo se le ha asignado un caso de uso de un compañero. Para poder realizar estas pruebas previamente hemos tenido que terminar todas las funcionalidades de nuestro software.

Marcos González Ruiz ha realizado las pruebas funcionales del caso de uso Comprar sándwich.

Sandra Mendoza Martínez ha realizado las pruebas funcionales del caso de uso Crear pedido proveedor.

Alonso Illán Martínez del Santo ha realizado las pruebas unitarias del caso de uso Crear oferta

Alberto López Hernández ha realizado las pruebas unitarias del caso de uso Crear sándwich.

Las pruebas funcionales consisten en comprobar que el flujo principal se realiza acorde a los detallado en los casos de uso y de que todos los flujos alternativos funcionan correctamente.

	Versión: <1.0>
Plan de Pruebas	Fecha: <15/12/2022>
PPS2G	

## 10. Necesidades ambientales

Para poder realizar las pruebas, necesitaremos los siguientes elementos:

### 10.1 Hardware

No aplica.

### 10.2 Software

Necesitaremos el Software de Visual Studio Enterprise, el software de pruebas Selenium, y un navegador Web para realizar las pruebas funcionales.

### 10.3 Seguridad

No aplica.

### 10.4 Herramientas

Como herramienta de apoyo a las pruebas, usaremos Azure DevOps, para poder crear los "Work Items" y garantizar el seguimiento del desarrollo de la aplicación.

### 10.5 Publicaciones

No aplica.

## 11. Responsabilidades.

Los miembros encargados tanto del manejo como del diseño de la aplicación son los integrantes que pertenecen al Equipo Sándwich.

También se encargan de la preparación de esta y de arreglar los errores que puedan aparecer en cualquier momento.

En cuanto al equipo, se ha empleado el ordenador de cada uno de los miembros para poder llevar a cabo el proyecto.

Las comprobaciones del código las ha llevado a cabo prioritariamente el Scrum Máster. Los testigos empleados han sido los mismos integrantes del grupo.

	Versión: <1.0>
Plan de Pruebas	Fecha: <15/12/2022>
PPS2G	

## 12. Aprobación

Sandra Mendoza Martínez

Fecha: 15/12/2022

Firma: Sandra Mendoza Martínez

Alonso Illán Martínez del Santo

Fecha: 15/12/2022

Firma: Alonso Illán Martínez del Santo

Marcos González Ruiz

Fecha: 15/12/2022

Firma: Marcos González Ruiz

Alberto López Hernández

Fecha: 15/12/2022

Firma: Alberto López Hernández