PRUEBAS SOFTWARE BARDIS

JERSON CAMILO MISAS SERNA

DAVINSON POSADA TORRES

LUISA FERNANDA URREGO VÉLEZ

SERVICIO NACIONAL DE APRENDIZAJE

FICHA 1828182

ANÁLISIS Y DESARROLLO DE SISTEMAS DE INFORMACIÓN

MEDELLÍN

2020

PRUEBAS SOFTWARE BARDIS

JERSON CAMILO MISAS SERNA

DAVINSON POSADA TORRES

LUISA FERNANDA URREGO VÉLEZ

EDILFREDO PINEDA FLOREZ

SERVICIO NACIONAL DE APRENDIZAJE

FICHA 1828182

ANÁLISIS Y DESARROLLO DE SISTEMAS DE INFORMACIÓN

MEDELLÍN

2020

Contenido

[Introducción 1](#_Toc43466250)

[¿En qué módulos y por qué? 2](#_Toc43466251)

[Pruebas de Caja Blanca 3](#_Toc43466252)

[Módulo Pedido: 4](#_Toc43466253)

[Modulo Producto: 5](#_Toc43466254)

[Módulo Canción: 6](#_Toc43466255)

[Pruebas de Caja Negra 7](#_Toc43466256)

[Módulo Pedido: 7](#_Toc43466257)

[Módulo Producto: 8](#_Toc43466258)

[Módulo Canción: 8](#_Toc43466259)

[Casos de Prueba 10](#_Toc43466260)

[Módulo Producto: 10](#_Toc43466261)

[Módulo Canción: 11](#_Toc43466262)

[Módulo Contacto: 11](#_Toc43466263)

[Módulo Inicio de sesión 12](#_Toc43466264)

[Módulo Empleados 12](#_Toc43466265)

[Pruebas Automatizadas con Selenium 14](#_Toc43466266)

[Módulo Pedido 15](#_Toc43466267)

[Módulo Producto 22](#_Toc43466268)

[Módulo Canción 33](#_Toc43466269)

[Resultados pruebas 38](#_Toc43466270)

[Pruebas de caja blanca y de caja negra: 38](#_Toc43466271)

[Casos de prueba: 38](#_Toc43466272)

[Pruebas automatizadas: 38](#_Toc43466273)

# Introducción

En el presente documento se dan a conocer las diferentes pruebas de software realizadas al sistema Bardis para verificar el correcto funcionamiento del sistema, las cuales son pruebas de caja blanca, de caja negra, casos de prueba, pruebas automatizadas realizadas con Selenium IDE y a su vez, también se presentan los resultados que arrojaron estas pruebas.

# ¿En qué módulos y por qué?

Las pruebas fueron realizadas solamente en algunos módulos del sistema Bardis, los cuales son el módulo de los productos, módulo de los pedidos y el módulo de solicitar las canciones; y los casos de prueba se realizaron en esos mismos módulos, pero también en el módulo de los empleados, módulo de inicio de sesión y módulo de contacto, para las pruebas automatizadas se tomaron varios casos pruebas para los módulos de pedidos, productos y canciones.

Las diferentes pruebas se realizaron en esos módulos debido a que representan la funcionalidad más importante del sistema Bardis.

# Pruebas de Caja Blanca

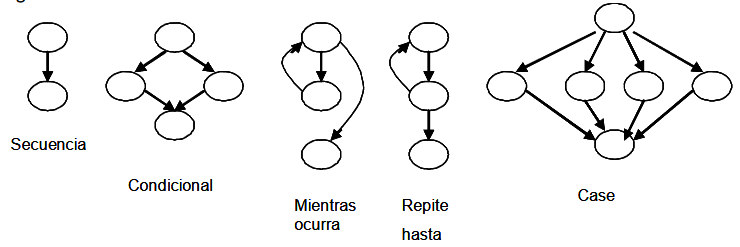
Estas pruebas se realizaron de manera manual, donde se probó la lógica y estructura interna de diferentes módulos del sistema Bardis, con el objetivo de examinar los caminos de una estructura correspondiente a un módulo. Los pasos que se realizaron en esta prueba, para los diferentes módulos fueron:

1. Tomar una parte de código para examinar.
2. Dibujar el grafo de flujo correspondiente a esa parte de código.
3. Determinar la complejidad ciclomatica de ese grafo.
4. Extraer los caminos independientes de esa estructura.

Hay varios tipos de grafos, su uso depende de la estructura que se va a examinar, cada grafo está compuesto por los nodos que son aquellos que representan una o más acciones secuenciales y los arcos que representan los cambios de dirección.

Nodos

Arcos



A continuación, se presenta la información mencionada anteriormente, pero con los módulos

del sistema Bardis:

## Módulo Pedido:

|  |  |  |
| --- | --- | --- |
| **Pseudocódigo**  A. Inicio.  B. Leer id\_producto, por la URL.  C. Si id\_producto url = id\_producto array entonces  D. Cantidad=Cantidad+1  Total = (Cantidad\*Precio) +Total.  E. Sino: Agregue nuevo producto en el array. | **Grafo Condicional-Realizar pedido** | **Caminos**  A-H  A-B-C-D-F-G-A-H  A-B-C-E-F-G-A-H  **CDM**  V(G)= #Aristas- #Nodos +2  V(CDM) = 9-8+2 = **3** |
| **Código** | **Grafo** | **Caminos y CDM** |

## Modulo Producto:

|  |  |  |
| --- | --- | --- |
| **Pseudocódigo**  A. Inicio  B. Leer nombre\_producto.  B. Leer precio\_producto  B. Leer descripción\_producto. Leer categoría\_producto  C. Mostrar “Se ha registrado exitosamente”.  D. Fin | **Grafo Secuencial-Agregar producto** | **Caminos**  A-D  A-B-C-A-D  **CDM**  V(G)= #Aristas- #Nodos +2  V(CDM) = 4-4+2 = **2** |
| **Código** | **Grafo** | **Caminos y CDM** |

## Módulo Canción:

|  |  |  |
| --- | --- | --- |
| **Pseudocódigo**  A. Inicio.  B. Leer nombre\_canción.  B. Leer intérprete\_canción.  C. Mostrar “Se ha enviado correctamente”.  D. Fin. | **Grafo Secuencial-Solicitar canción** | **Caminos**  A-D  A-B-C-A-D  **CDM**  V(G)= #Aristas- #Nodos +2  V(CDM) = 4-4+2 = **2** |
| **Código** | **Grafo** | **Caminos y CDM** |

# Pruebas de Caja Negra

Estas pruebas se realizaron de manera manual, donde se probó la funcionalidad del sistema sin tener en cuenta la lógica y la estructura interna de los diferentes módulos del sistema Bardis, con el objetivo de verificar la entrada y la salida esperada correspondiente a un módulo. A cada módulo se lo colocó una descripción del caso, es decir, una explicación de lo realizado en ese módulo, luego se coloca el dato de entrada y el resultado esperado del sistema en una acción determinada.

## Módulo Pedido:

**Nombre:** Realizar pedido.

**Descripción del caso:** Se realizarán pruebas para verificar si el pedido se realiza correctamente.

**Caso 1.1:** ***Datos de entrada:*** El pedido se encuentra vacío, es decir no hay productos agregados a ese pedido.

**Resultado esperado(Salida):** El sistema le solicita al usuario mesero que ingrese un producto en el pedido.

**Caso 1.2:** ***Datos de entrada:*** El pedido no tiene mesa ingresada.

**Resultado esperado(Salida):** El sistema le solicita al usuario mesero ingresar la mesa del pedido.

**Caso 1.3: *Datos de entrada:*** El mesero está enviando el pedido.

**Resultado esperado(Salida):** El pedido fue realizado exitosamente.

## Módulo Producto:

**Nombre:** Registrar producto.

**Descripción del caso:** Se realizarán pruebas para verificar si se agrega un producto correctamente.

**Caso 2.1: Datos de entrada:** El formulario para agregar los productos se encuentra vacío.

**Resultado esperado(Salida):** El sistema le solicita al usuario administrador los datos para agregar un producto.

**Caso 2.2: Datos de entrada:** El usuario administrador está ingresando los datos que se le piden.

**Resultado esperado(Salida):** El producto se agregó correctamente.

## Módulo Canción:

**Nombre:** Solicitar canción.

**Descripción del caso:** Se realizarán pruebas para verificar si se solicita una canción correctamente.

**Caso 2.1: *Datos de entrada:*** El formulario para solicitar una canción se encuentra vacío.

**Resultado esperado(Salida):** El sistema le solicita al cliente los datos para solicitar la canción.

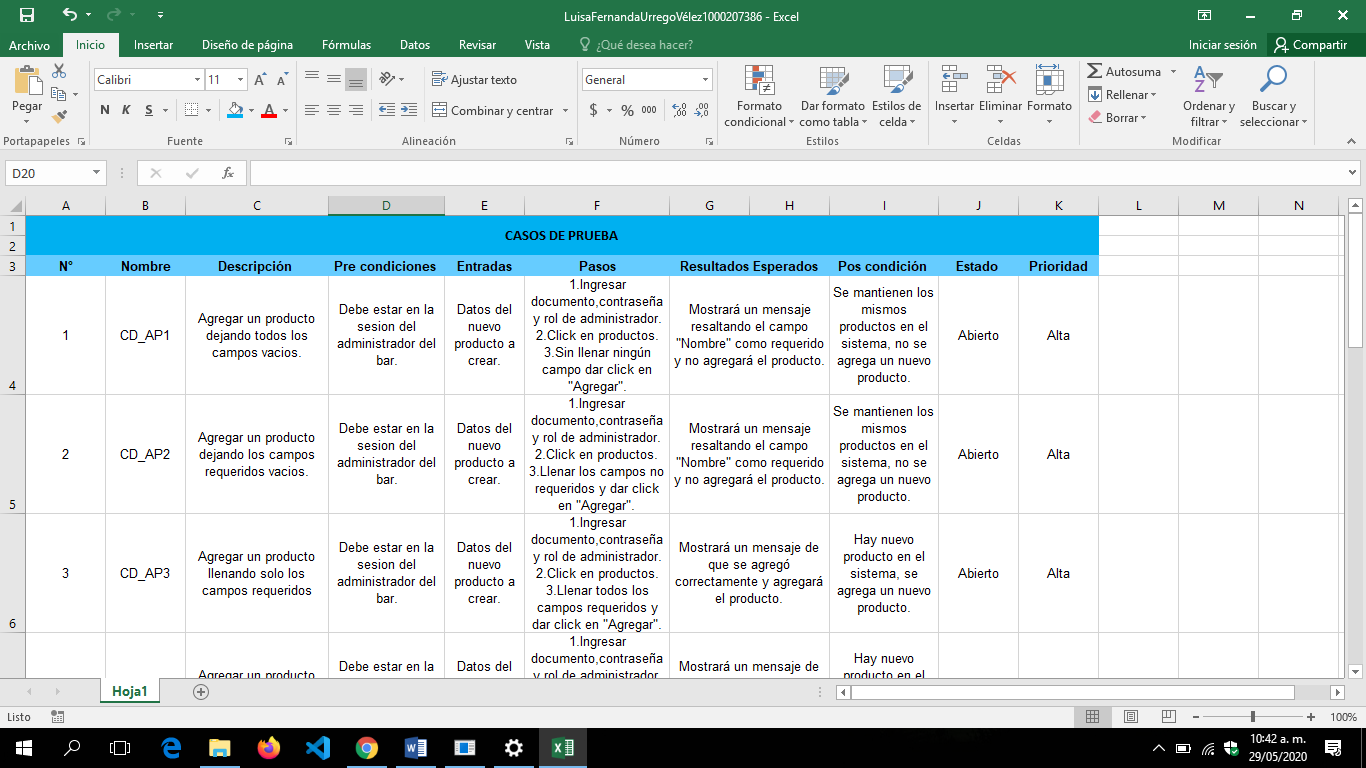
**Caso 2.2: Datos de entrada:** El usuario cliente está llenando los datos que se le piden

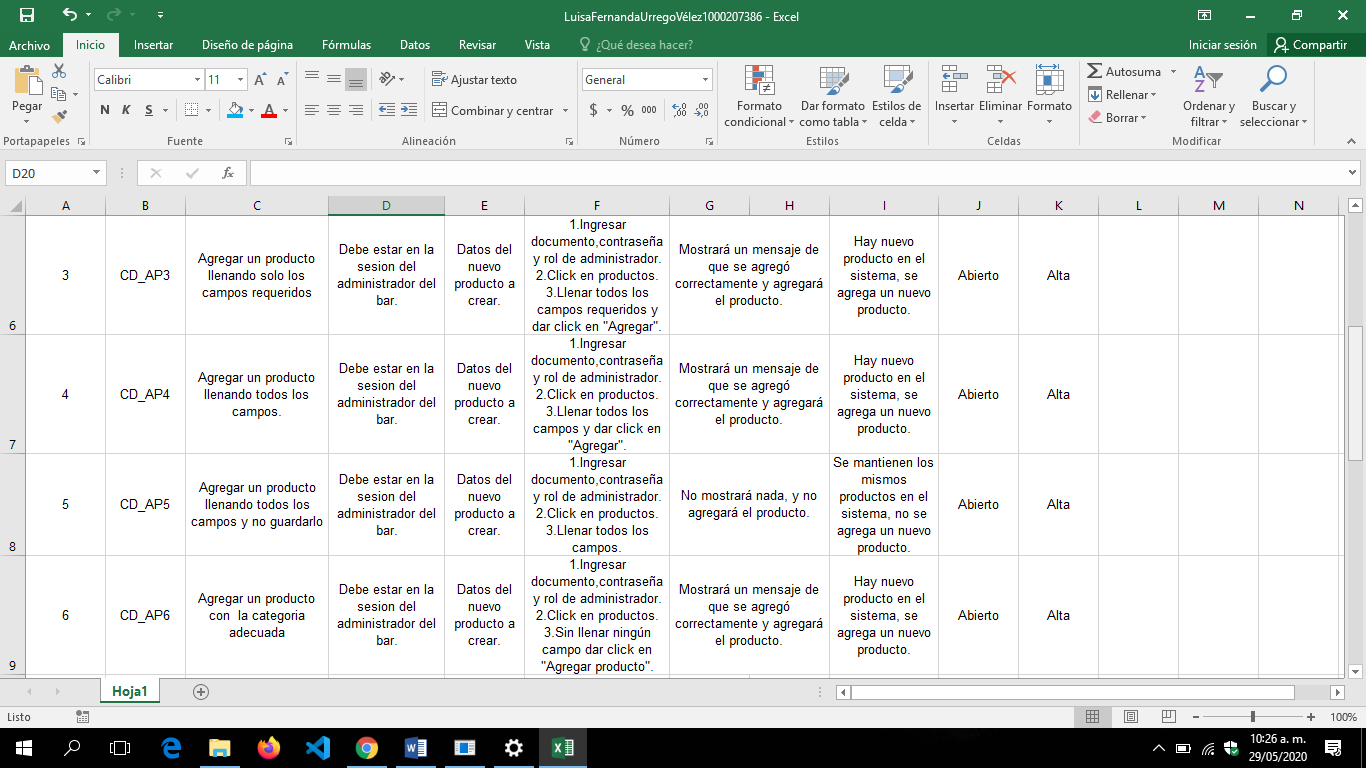
**Resultado esperado(Salida):** La canción se solicitó correctamente.

# Casos de Prueba

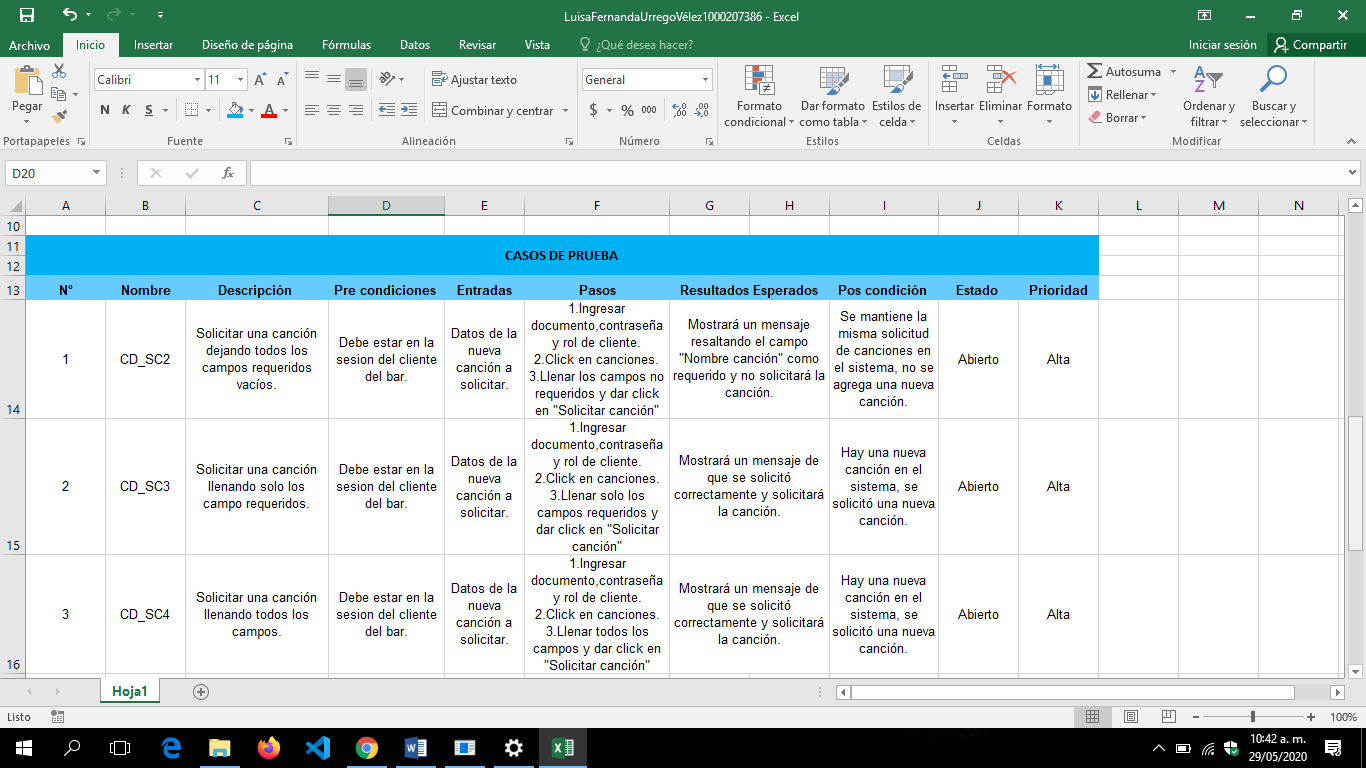
Se realizó una matriz con los casos de prueba de diversos módulos del sistema, donde se evaluó lo que sucedía en el respectivo modulo realizando diversas acciones. Los casos de prueba se componen por un número, nombre o identificador, descripción, pre condiciones, entradas, pasos, resultados esperados, pos condición, estado y prioridad.

## Módulo Producto:

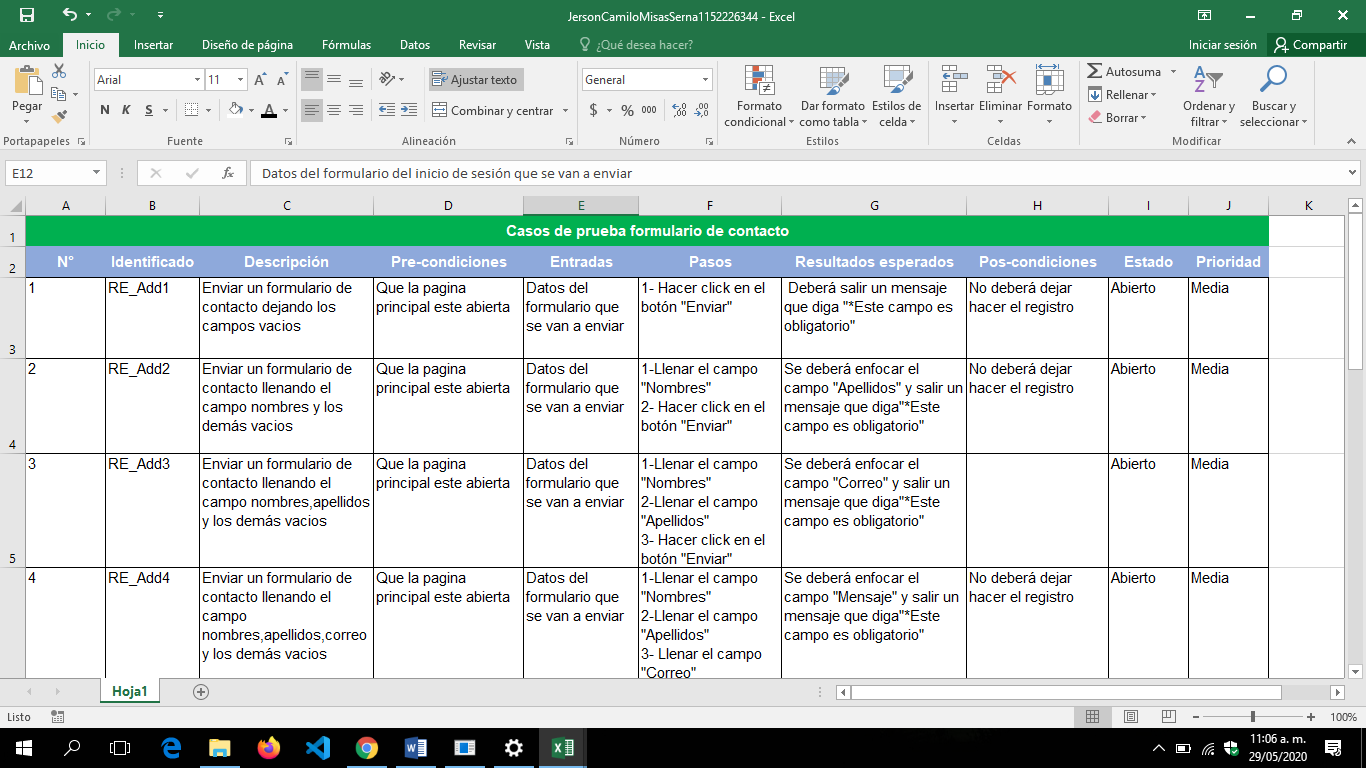


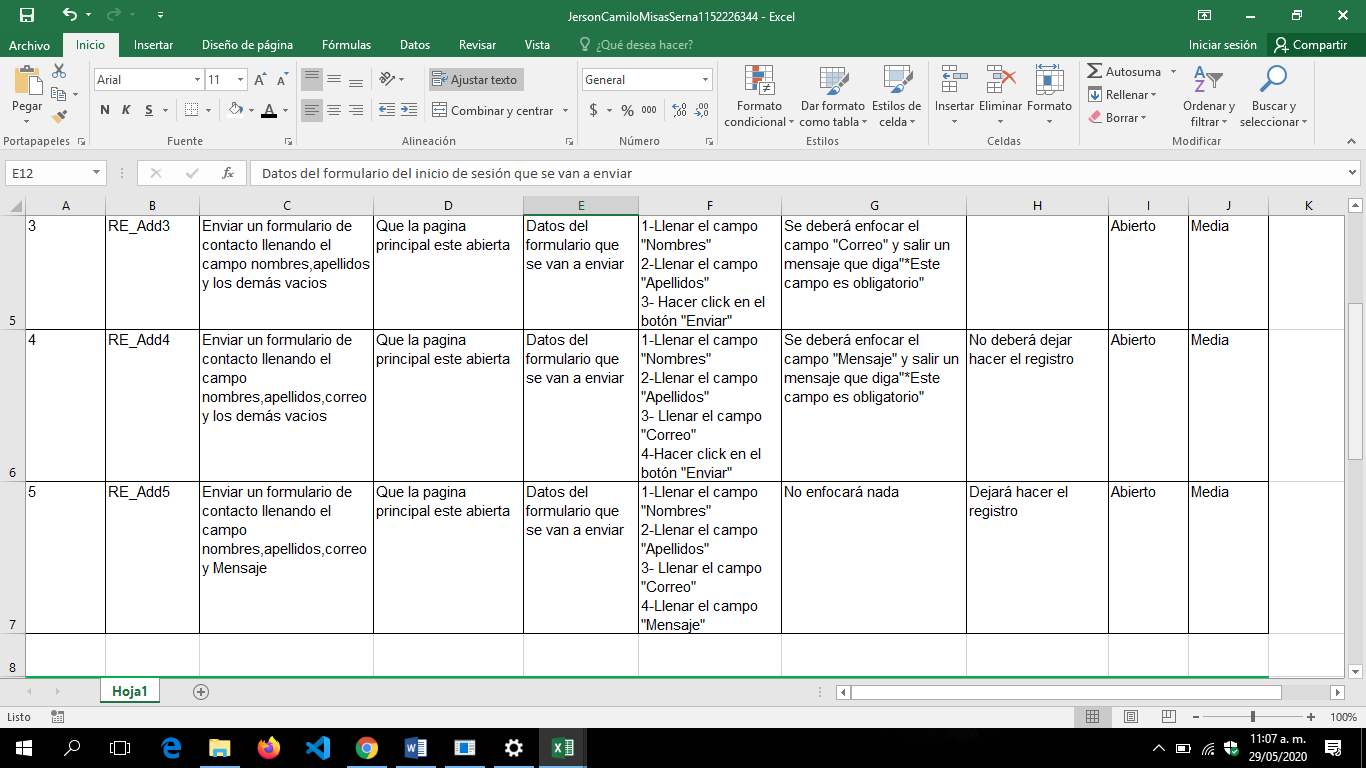


## Módulo Canción:

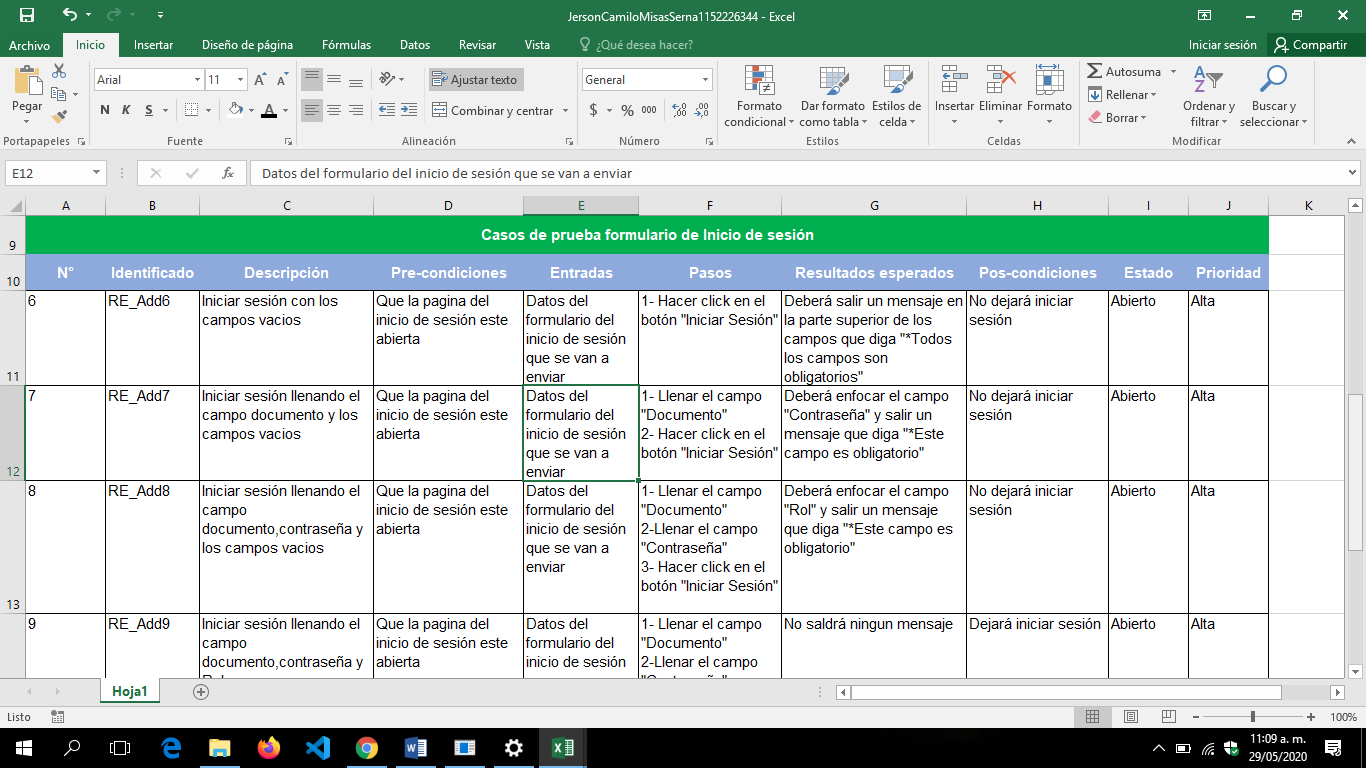


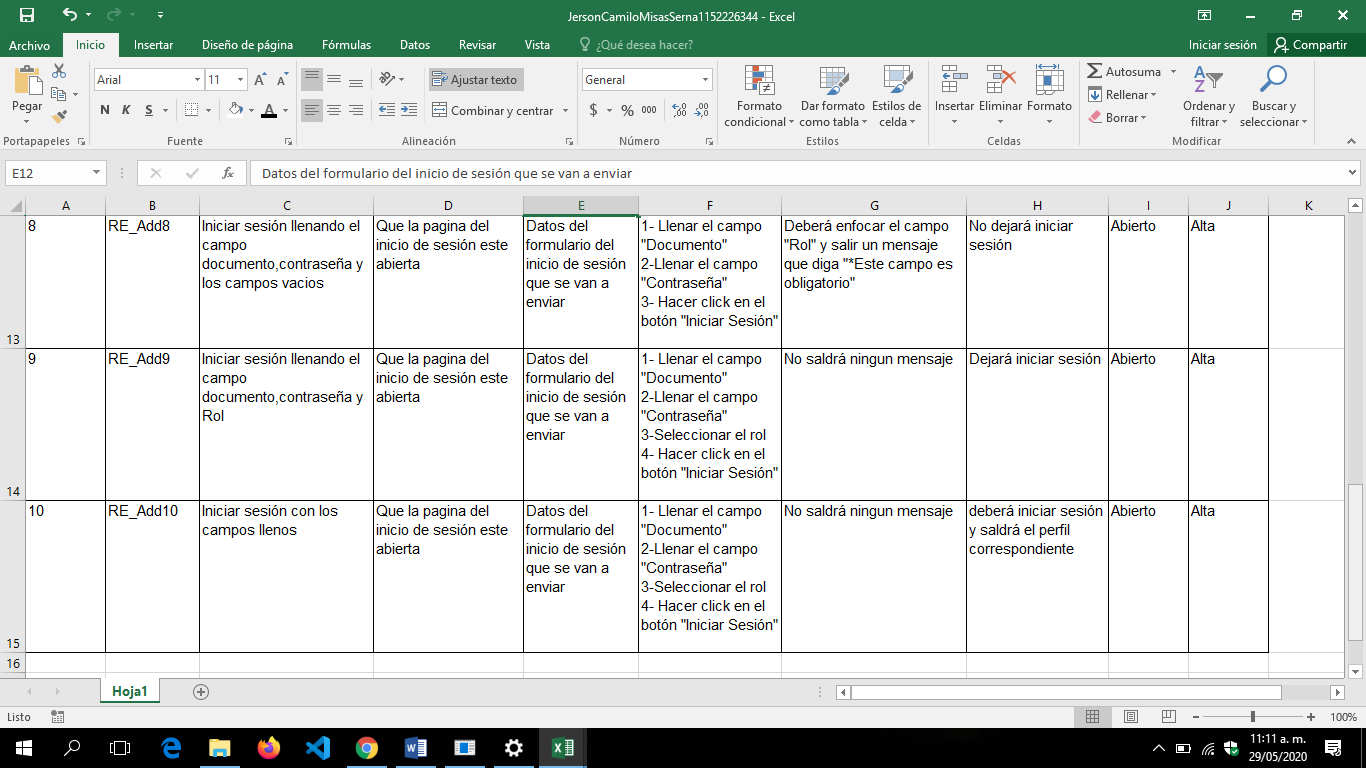
## Módulo Contacto:



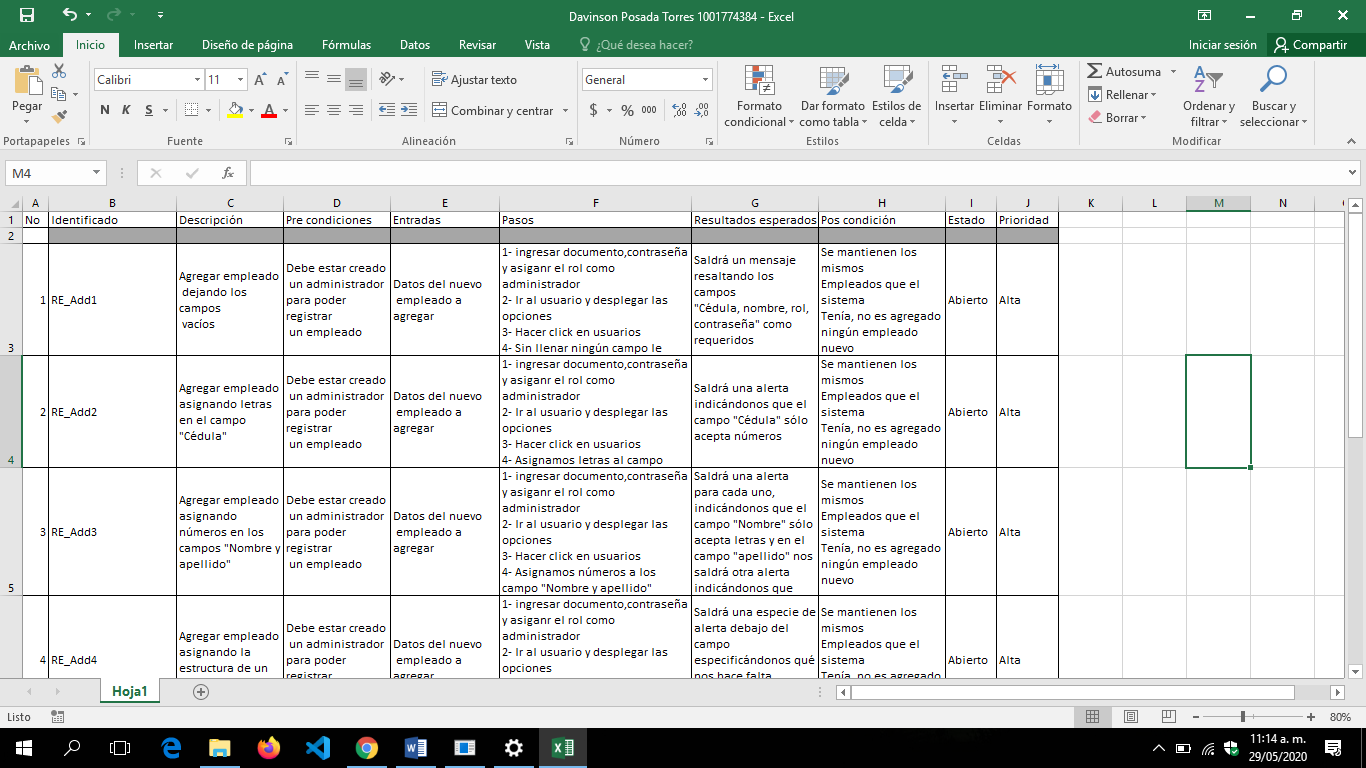


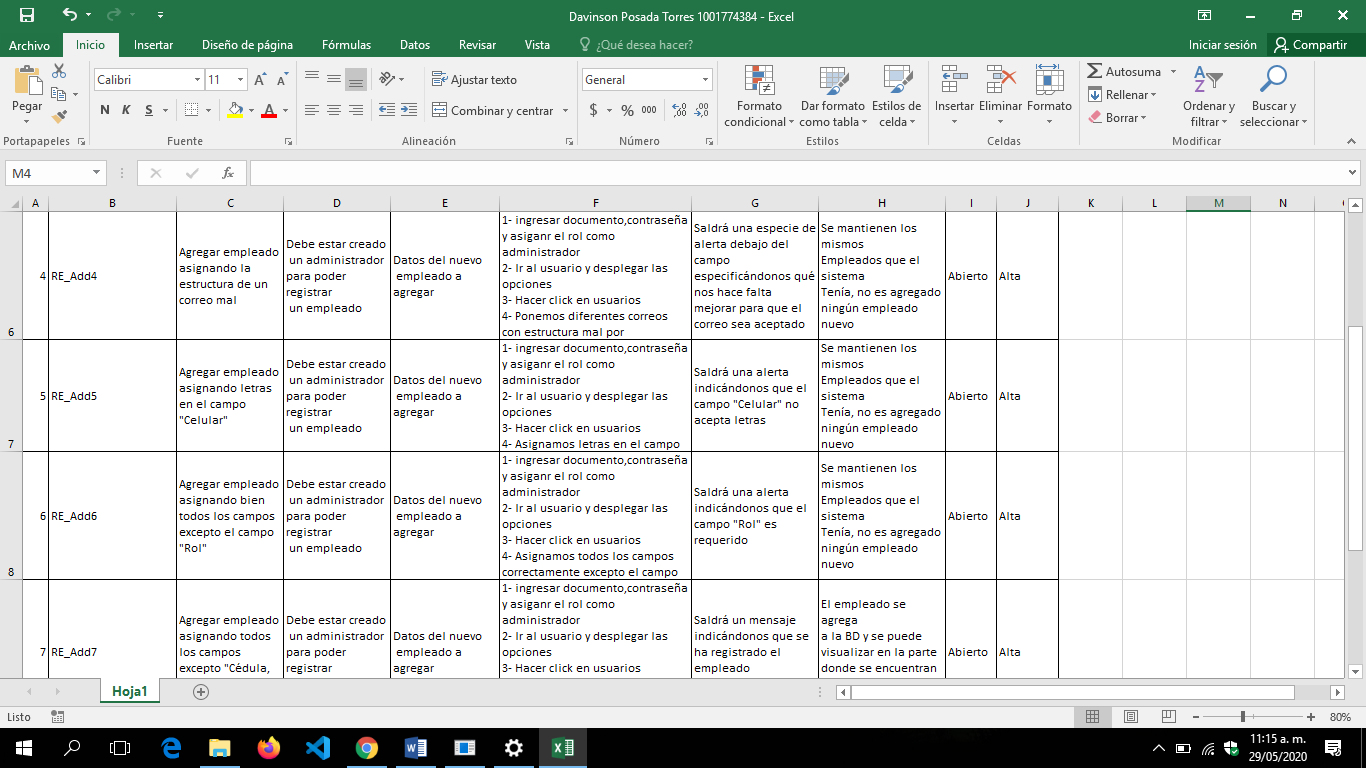
## Módulo Inicio de sesión

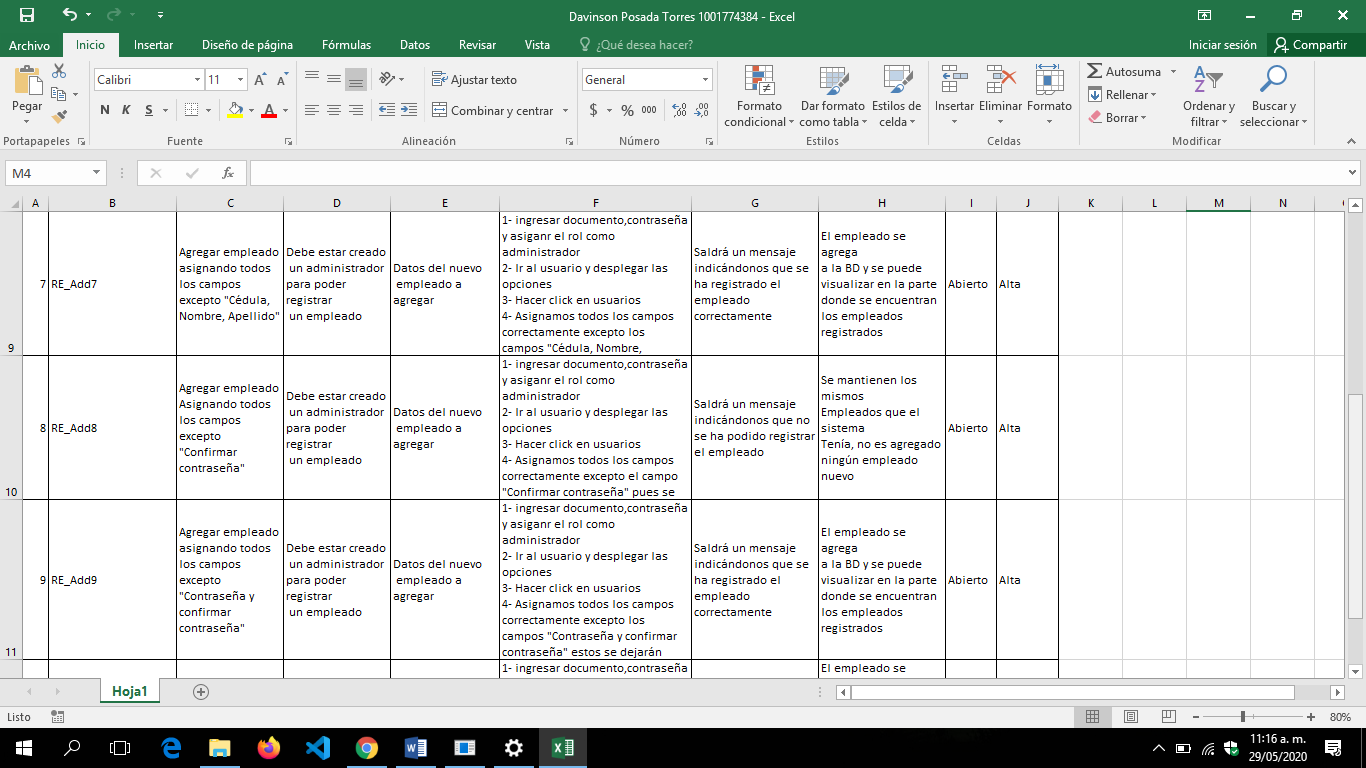


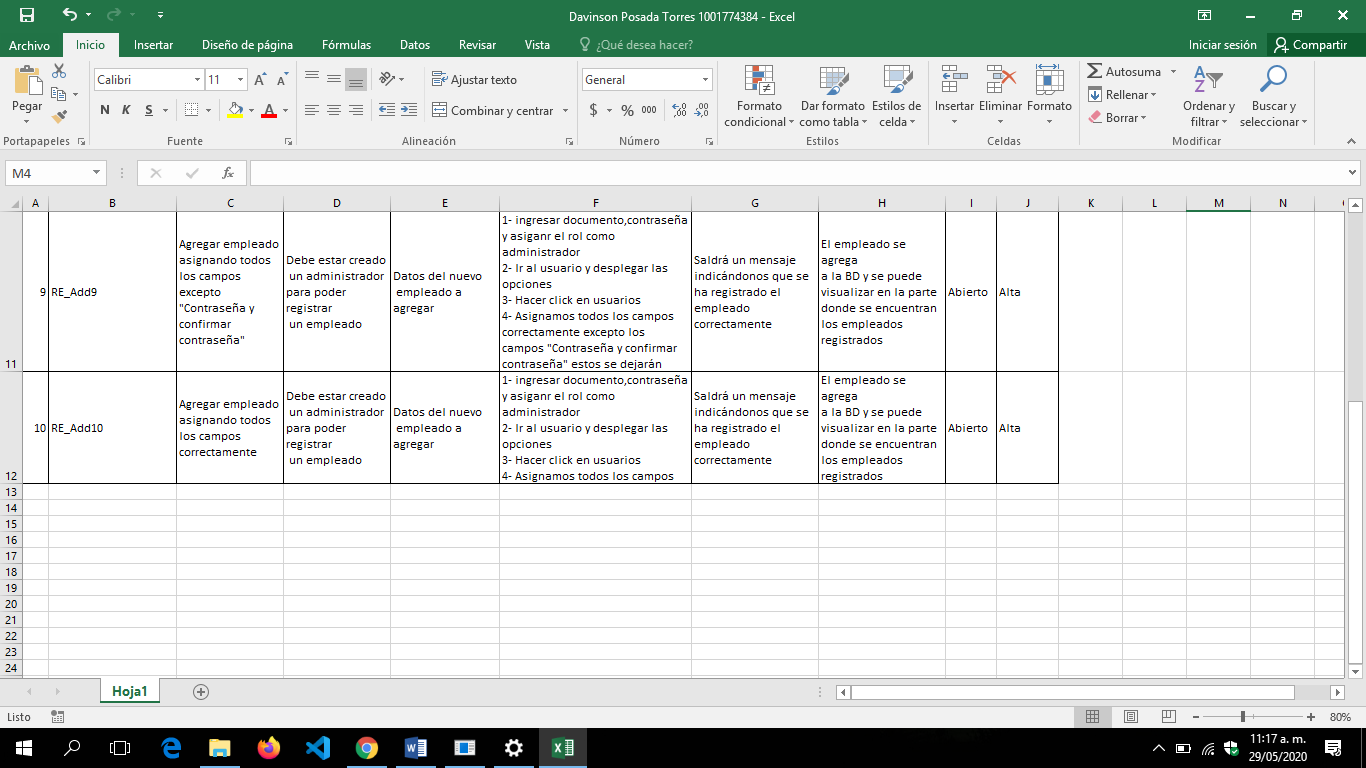


## Módulo Empleados









# Pruebas Automatizadas con Selenium

Las pruebas automatizadas nos permiten agilizar y automatizar todo el proceso de verificación de un sistema, en este caso, las pruebas automatizadas se realizaron con la herramienta Selenium IDE, el cual es un entorno de pruebas de software para aplicaciones basadas en la web, que permite grabar/reproducir para crear pruebas sin usar un lenguaje de scripting para pruebas; es una extensión que se puede añadir al navegador Google Chrome desde el siguiente enlace:

<https://chrome.google.com/webstore/detail/selenium-ide/mooikfkahbdckldjjndioackbalphokd>

Para la realización de las pruebas automatizadas a cada módulo del proyecto se necesita tener previamente:

Pruebas de caja blanca.

Pruebas de caja negra.

Casos de prueba.

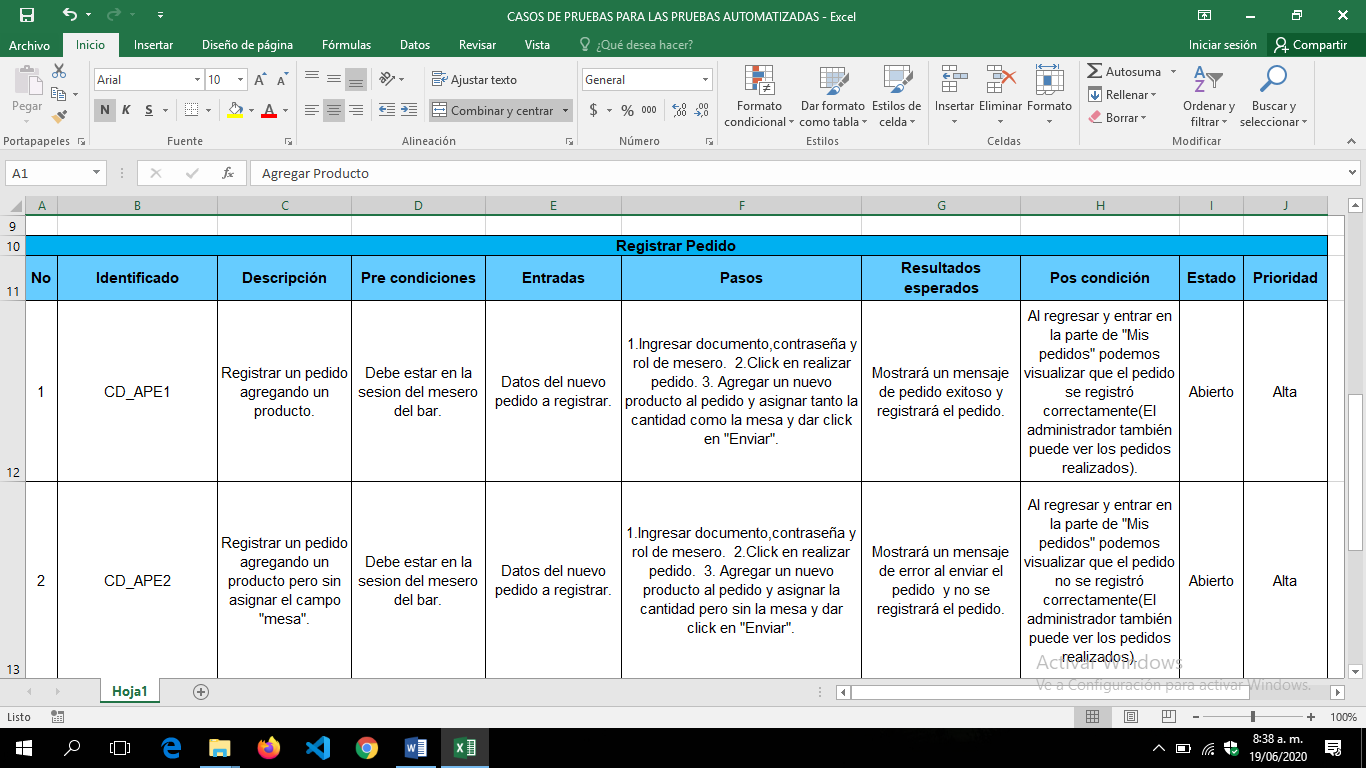
Esta información se encuentra en este mismo documento, las cuales han sido especificadas anteriormente.

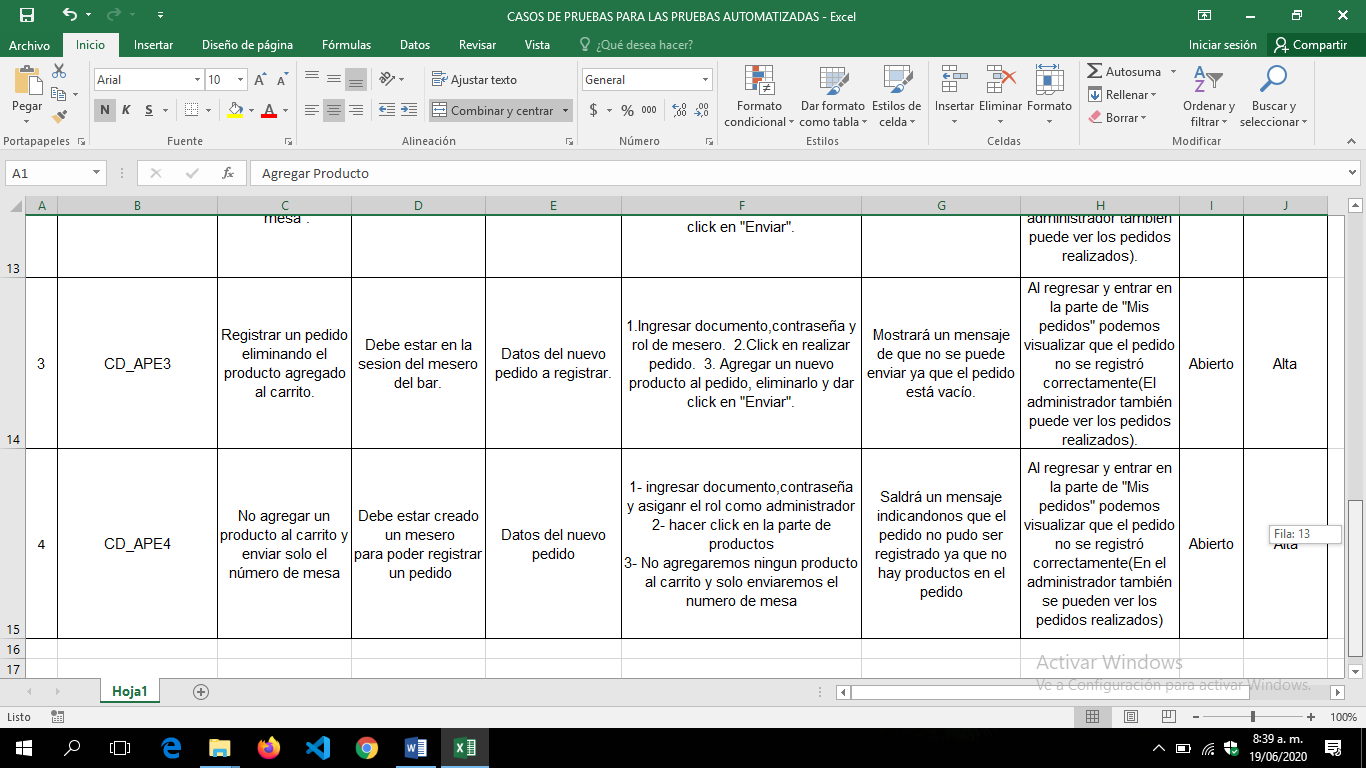
Estas pruebas se realizaron en tres módulos, los cuales son: módulo de pedidos, de productos y de canciones, aplicándole seis (6), cuatro (4) y tres (3) casos de prueba respectivamente.

## Módulo Pedido

Se evalúa el comportamiento del sistema Bardis enviando un nuevo pedido desde el módulo del empleado mesero, sometiéndolo a varios casos de prueba para así, finalmente analizar su funcionamiento.

Los casos de prueba aplicados a este módulo fueron:





**Caso de prueba #1:** Enviar un pedido agregando un producto.

**Nombre:** Realizar pedido.

**Descripción del caso:** Se realizarán pruebas para verificar si el pedido se realiza correctamente.

**Caso 1.1:** ***Datos de entrada:*** El pedido se encuentra vacío, es decir no hay productos agregados a ese pedido.

**Resultado esperado(Salida):** El sistema le solicita al usuario mesero que ingrese un producto en el pedido.

**Caso 1.2:** ***Datos de entrada:*** El pedido no tiene mesa ingresada.

**Resultado esperado(Salida):** El sistema le solicita al usuario mesero ingresar la mesa del pedido.

**Caso 1.3: *Datos de entrada:*** El mesero está enviando el pedido.

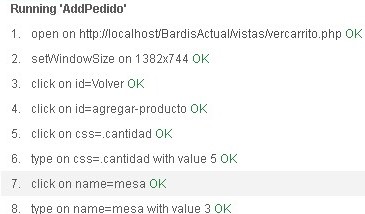
**Resultado esperado(Salida):** El pedido fue realizado exitosamente.

Se ejecuta la prueba automatizada con los siguientes valores:



**Resultado obtenido:** El pedido se realizó correctamente

**Estado:** Ok.

****

****

**Caso de prueba #2:** Enviar un pedido agregando un producto, pero sin ingresar la mesa.

**Nombre:** Realizar pedido.

**Descripción del caso:** Se realizarán pruebas para verificar si el pedido se realiza correctamente.

**Caso 2.1:** ***Datos de entrada:*** El pedido se encuentra vacío, es decir no hay productos agregados a ese pedido.

**Resultado esperado(Salida):** El sistema le solicita al usuario mesero que ingrese un producto en el pedido.

**Caso 2.2:** ***Datos de entrada:*** El pedido no tiene mesa ingresada.

**Resultado esperado(Salida):** El sistema le solicita al usuario mesero ingresar la mesa del pedido.

**Caso 2.3: *Datos de entrada:*** El mesero no ingresa la mesa y envía el pedido.

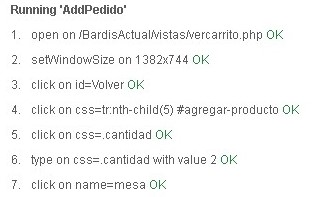
**Resultado esperado(Salida):** El pedido no se realizó correctamente.

Se ejecuta la prueba automatizada con los siguientes valores:



**Resultado obtenido:** El pedido no se realizó correctamente.

**Estado:** Ok.

****

****

**Caso de prueba #3:** Agregar un pedido, eliminarlo y enviar el pedido.

**Nombre:** Realizar pedido.

**Descripción del caso:** Se realizarán pruebas para verificar si el pedido se realiza correctamente.

**Caso 3.1:** ***Datos de entrada:*** El pedido se encuentra vacío, es decir no hay productos agregados a ese pedido.

**Resultado esperado(Salida):** El sistema le solicita al usuario mesero que ingrese un producto en el pedido.

**Caso 3.2:** ***Datos de entrada:*** El pedido no tiene mesa ingresada.

**Resultado esperado(Salida):** El sistema le solicita al usuario mesero ingresar la mesa del pedido.

**Caso 3.3: *Datos de entrada:*** El mesero elimina el producto que agregó y envía el pedido.

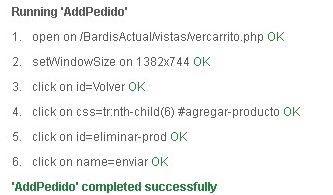
**Resultado esperado(Salida):** El pedido no se realizó correctamente.

Se ejecuta la prueba automatizada, sin agregar nada:



**Resultado obtenido:** El pedido no se realizó correctamente.

**Estado:** Ok.

****

**Caso de prueba #4:** No agregar un producto al carrito y enviar solo el número de mesa.

**Nombre:** Realizar pedido.

**Descripción del caso:** Se realizarán pruebas para verificar si el pedido se realiza correctamente.

**Caso 4.1:** ***Datos de entrada:*** El pedido se encuentra vacío, es decir no hay productos agregados a ese pedido.

**Resultado esperado(Salida):** El sistema le solicita al usuario mesero que ingrese un producto en el pedido.

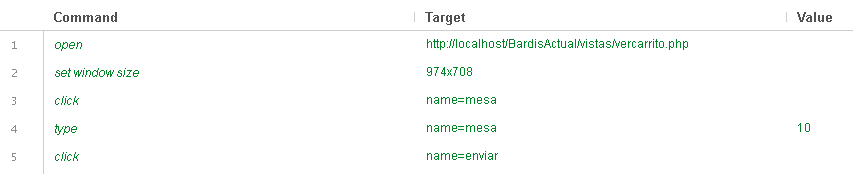
**Caso 4.2:** ***Datos de entrada:*** El pedido no tiene mesa ingresada.

**Resultado esperado(Salida):** El sistema le solicita al usuario mesero ingresar la mesa del pedido.

**Caso 4.3: *Datos de entrada:*** El mesero no agrega ningún producto al carrito y asigna un numero de mesa para luego darle enviar pedido.

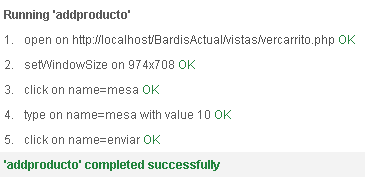
**Resultado esperado(Salida):** El pedido no se realizó correctamente porque no hay productos en el pedido.

Se ejecuta la prueba automatizada con los siguientes valores:



**Resultado obtenido:** El pedido no se realizó correctamente.

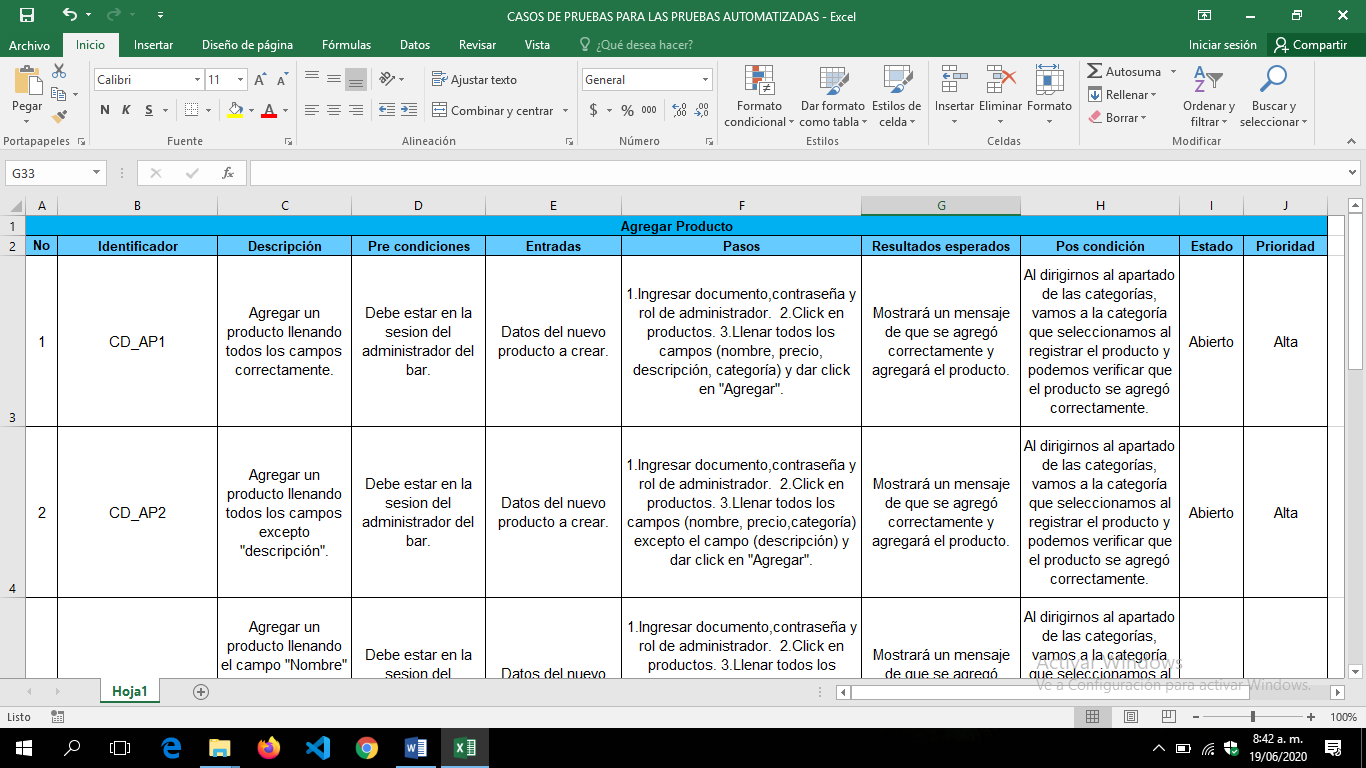
**Estado:** Ok.

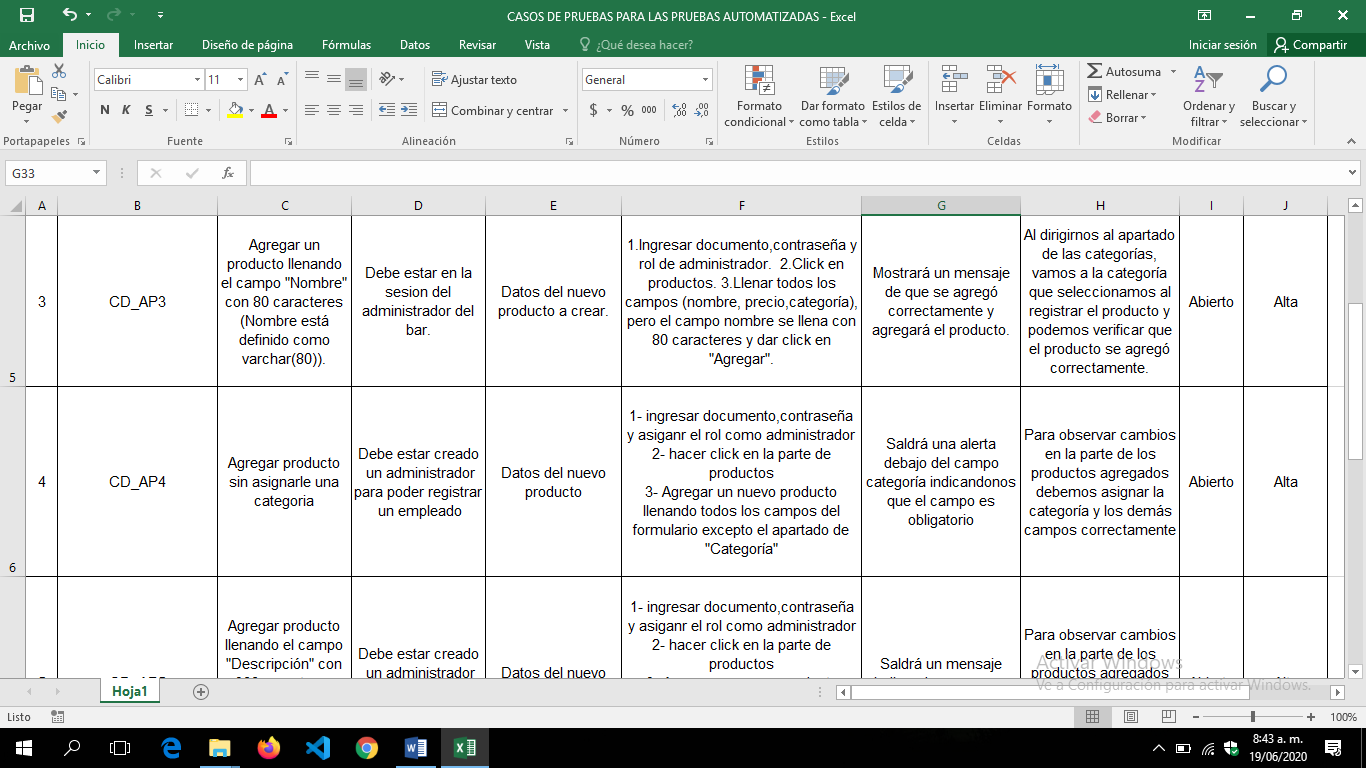


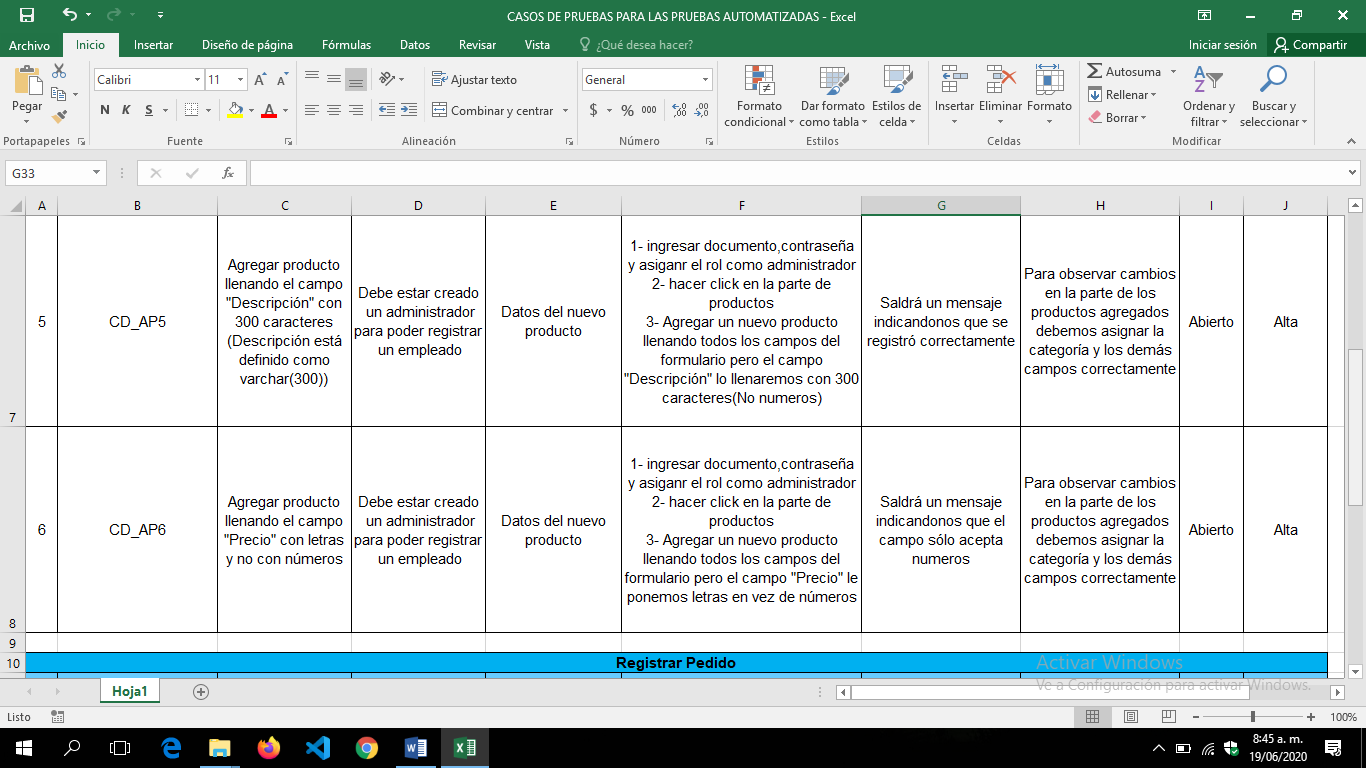
## Módulo Producto

Se evalúa el comportamiento del sistema Bardis agregando un nuevo producto desde el módulo Administrador, sometiéndolo a varios casos de prueba para así, finalmente analizar su funcionamiento.

Los casos de prueba aplicados a este módulo fueron:







**Caso de prueba #1:** Agregar un producto llenando todos los campos.

**Nombre:** Registrar producto.

**Descripción del caso:** Se realizarán pruebas para verificar si se agrega un producto correctamente.

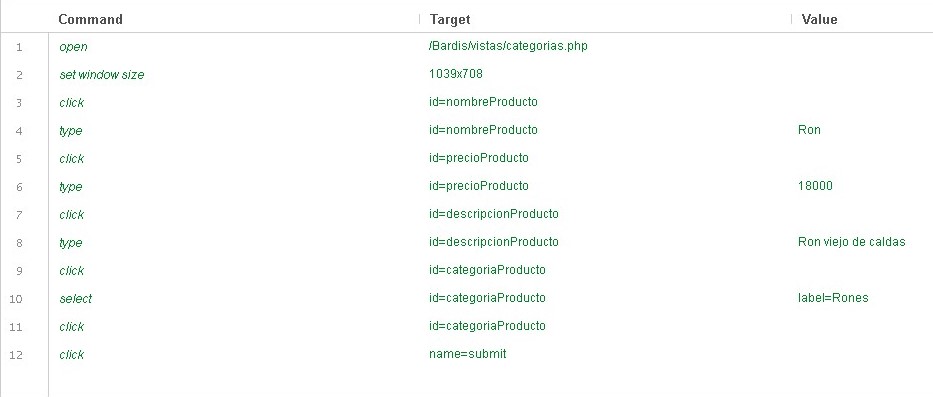
**Caso 1.1: Datos de entrada:** El formulario para agregar los productos se encuentra vacío.

**Resultado esperado(Salida):** El sistema le solicita al usuario administrador los datos para agregar un producto.

**Caso 1.2: Datos de entrada:** El usuario administrador está ingresando los datos que se le piden.

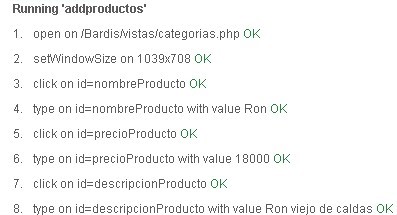
**Resultado esperado(Salida):** El producto se agregó correctamente.

Se ejecuta la prueba automatizada con los siguientes valores:



**Resultado obtenido:** El producto se agregó correctamente.

**Estado:** Ok.





**Caso de prueba #2:** Agregar un producto llenando todos los campos excepto el campo descripción.

**Nombre:** Registrar producto.

**Descripción del caso:** Se realizarán pruebas para verificar si se agrega un producto correctamente.

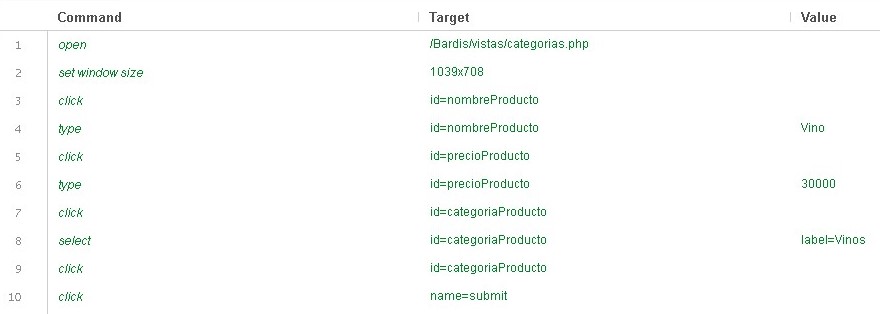
**Caso 2.1: Datos de entrada:** El formulario para agregar los productos se encuentra vacío.

**Resultado esperado(Salida):** El sistema le solicita al usuario administrador los datos para agregar un producto.

**Caso 2.2: Datos de entrada:** El usuario administrador está ingresando los datos que se le piden excepto la descripción del producto.

**Resultado esperado(Salida):** El producto se agregó correctamente.

Se ejecuta la prueba automatizada con los siguientes valores:



**Resultado obtenido:** El producto se agregó correctamente.

**Estado:** Ok.

****

****

**Caso de prueba #3:** Agregar un producto llenando el campo nombre con 80 caracteres, ya que el nombre está definido como varchar (80).

**Nombre:** Registrar producto.

**Descripción del caso:** Se realizarán pruebas para verificar si se agrega un producto correctamente.

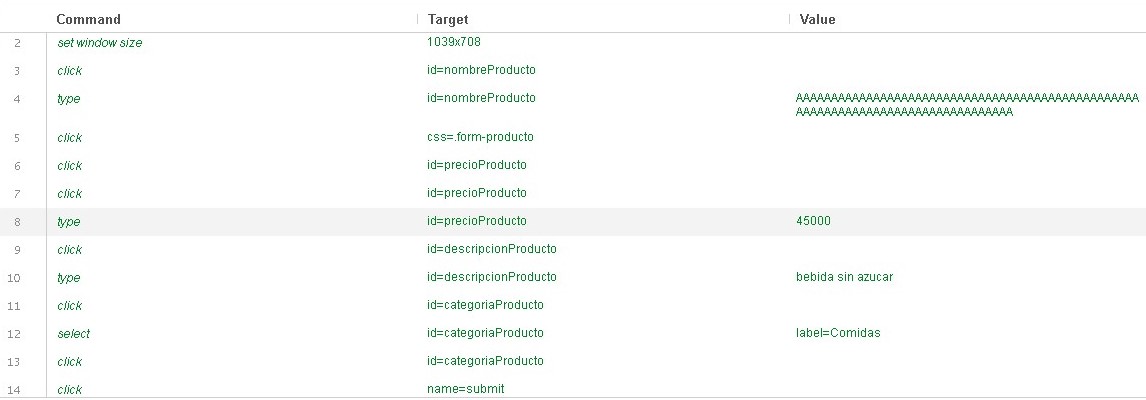
**Caso 3.1: Datos de entrada:** El formulario para agregar los productos se encuentra vacío.

**Resultado esperado(Salida):** El sistema le solicita al usuario administrador los datos para agregar un producto.

**Caso 3.2: Datos de entrada:** El usuario administrador está ingresando los datos que se le piden, y al campo nombre le pone 80 caracteres (letras).

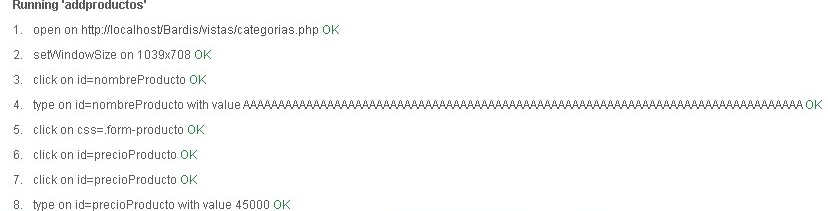
**Resultado esperado(Salida):** El producto se agregó correctamente.

Se ejecuta la prueba automatizada con los siguientes valores:



**Resultado obtenido:** El producto se agregó correctamente.

**Estado:** Ok.

****

****

**Caso de prueba #4:** Agregar producto sin asignarle una categoría.

**Nombre:** Registrar producto.

**Descripción del caso:** Se realizarán pruebas para verificar si se agrega un producto correctamente.

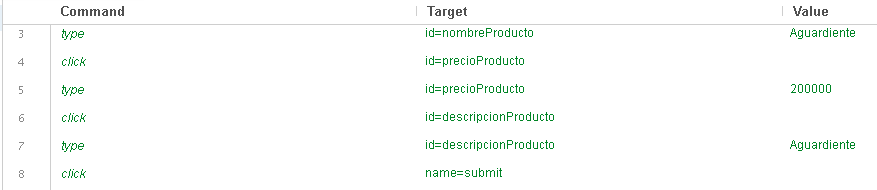
**Caso 4.1: Datos de entrada:** El formulario para agregar los productos se encuentra vacío.

**Resultado esperado(Salida):** El sistema le solicita al usuario administrador los datos para agregar un producto.

**Caso 4.2: Datos de entrada:** El usuario administrador está ingresando los datos que se le piden, pero en el apartado categoría no selecciona ninguna.

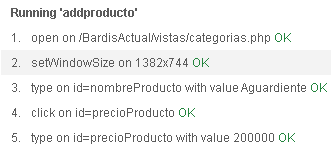
**Resultado esperado(Salida):** El producto no se puede agregar porque el campo “Categoría” es obligatorio.

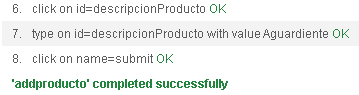
Se ejecuta la prueba automatizada con los siguientes valores:



**Resultado obtenido:** El producto no se agregó

**Estado:** Ok.





El producto no se agregó debido a que el campo categoría es obligatorio:



**Caso de prueba #5:** Agregar un producto llenando el campo "Descripción" con 300 caracteres (Descripción está definido como varchar (300)).

**Nombre:** Registrar producto.

**Descripción del caso:** Se realizarán pruebas para verificar si se agrega un producto correctamente.

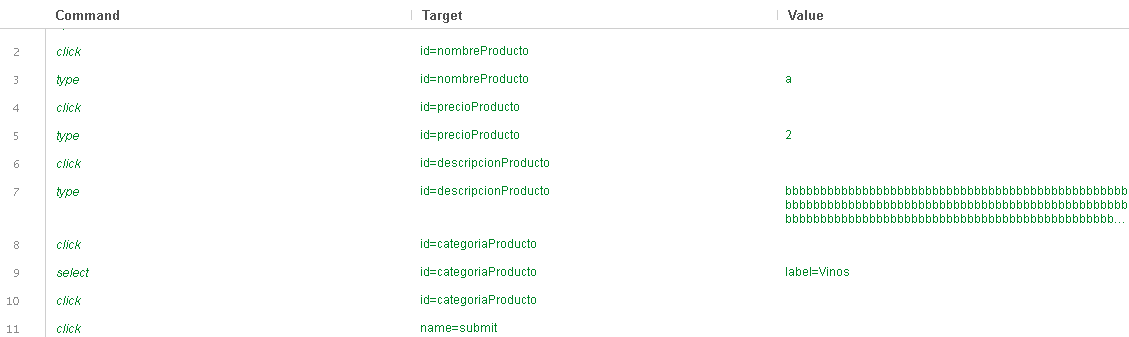
**Caso 5.1: Datos de entrada:** El formulario para agregar los productos se encuentra vacío.

**Resultado esperado(Salida):** El sistema le solicita al usuario administrador los datos para agregar un producto.

**Caso 5.2: Datos de entrada:** El usuario administrador está ingresando los datos que se le piden, pero en el campo descripción pone 300 caracteres.

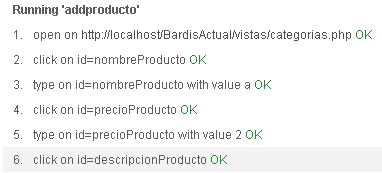
**Resultado esperado(Salida):** El producto se registró correctamente.

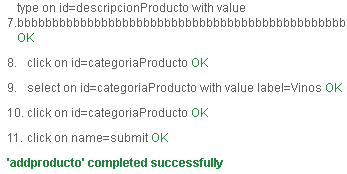
Se ejecuta la prueba automatizada con los siguientes valores:



**Resultado obtenido:** El producto se agregó correctamente.

**Estado:** Ok.





**Caso de prueba #6:** Agregar un producto llenando el campo "Precio" con letras y no con números.

**Nombre:** Registrar producto.

**Descripción del caso:** Se realizarán pruebas para verificar si se agrega un producto correctamente.

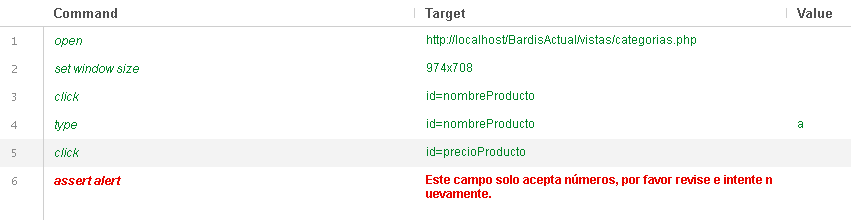
**Caso 6.1: Datos de entrada:** El formulario para agregar los productos se encuentra vacío.

**Resultado esperado(Salida):** El sistema le solicita al usuario administrador los datos para agregar un producto.

**Caso 6.2: Datos de entrada:** El usuario administrador está ingresando los datos que se le piden, pero en el campo precio en vez de poner números asigna letras.

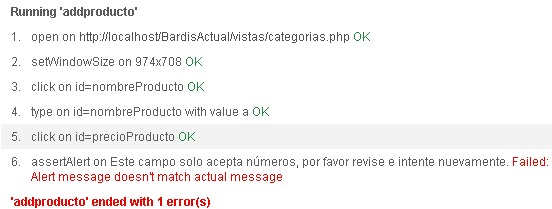
**Resultado esperado(Salida):** El producto no se registró correctamente (La herramienta Selenium IDE detecta que no pusimos el campo con los valores permitidos).

Se ejecuta la prueba automatizada con los siguientes valores:



**Resultado obtenido:** El producto no se agregó.

**Estado:** Ok.

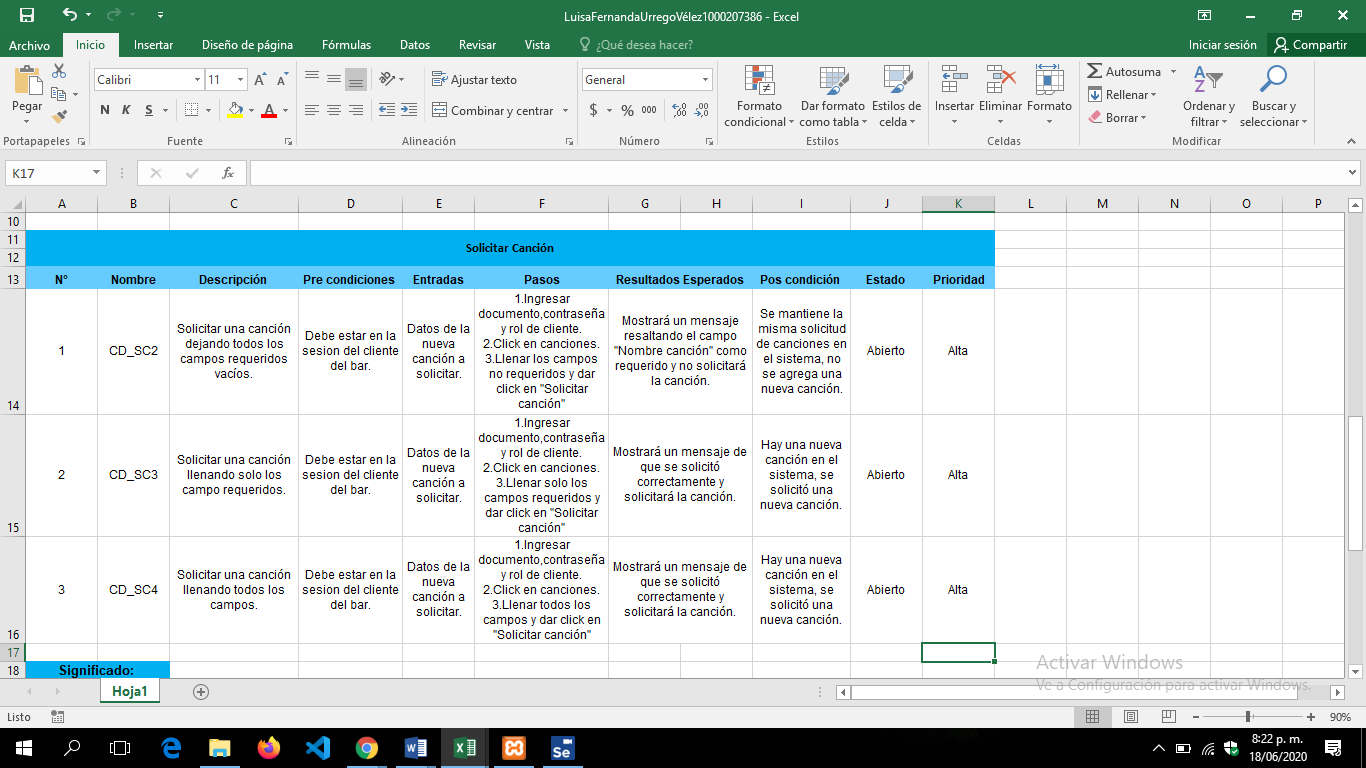


**Nota:** Sale una alerta en el campo precio debido a que éste solo acepta números, y se intentó ingresar letras.

## Módulo Canción

Se evalúa el comportamiento del sistema Bardis solicitando una nueva canción desde el módulo Cliente, sometiéndolo a varios casos de prueba para así, finalmente analizar su funcionamiento.

Los casos de prueba aplicados a este módulo fueron:



**Caso de prueba #1:** Solicitar una canción llenando solo el campo del nombre del intérprete de la canción.

**Nombre:** Solicitar canción.

**Descripción del caso:** Se realizarán pruebas para verificar si se solicita una canción correctamente.

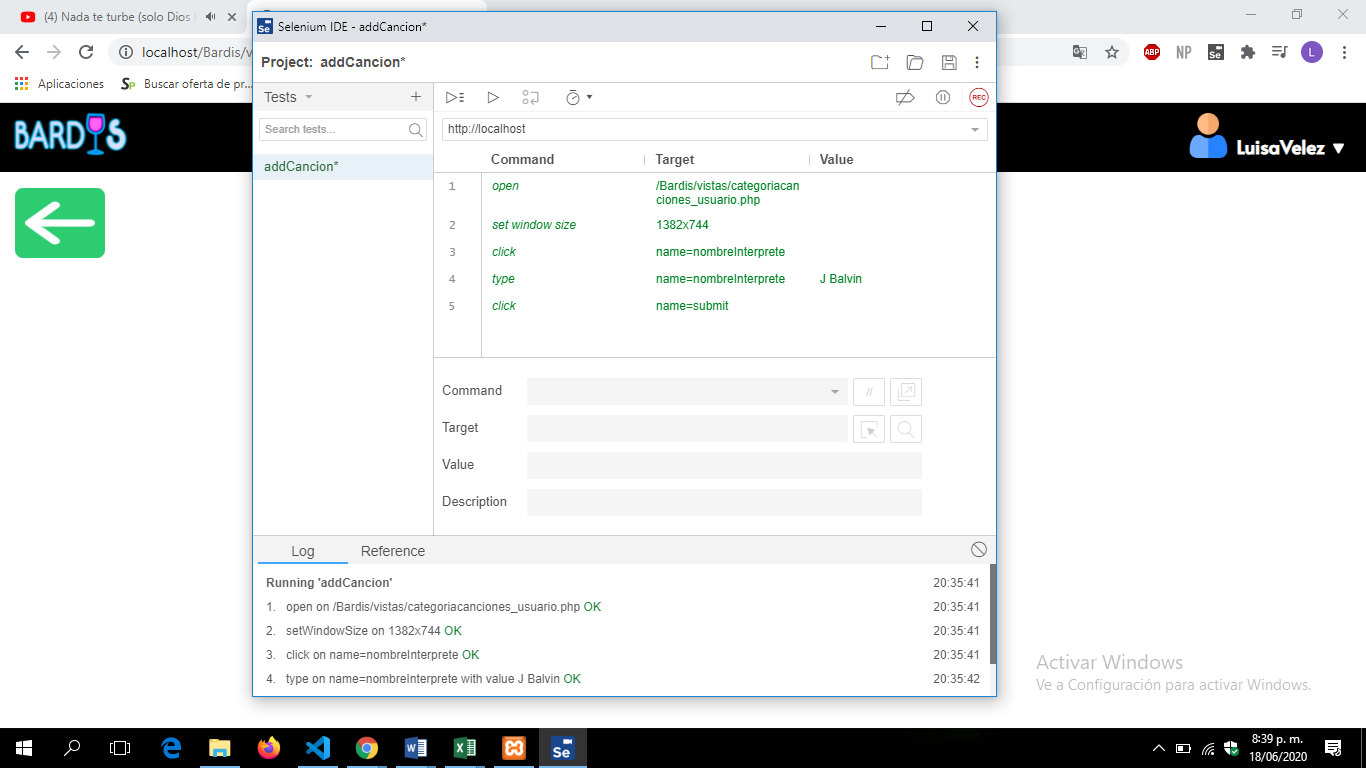
**Caso 1.1: *Datos de entrada:*** El formulario para solicitar una canción se encuentra vacío.

**Resultado esperado(Salida):** El sistema le solicita al cliente los datos para solicitar la canción.

**Caso 1.2: Datos de entrada:** El usuario cliente está ingresando solamente el nombre del intérprete de la canción.

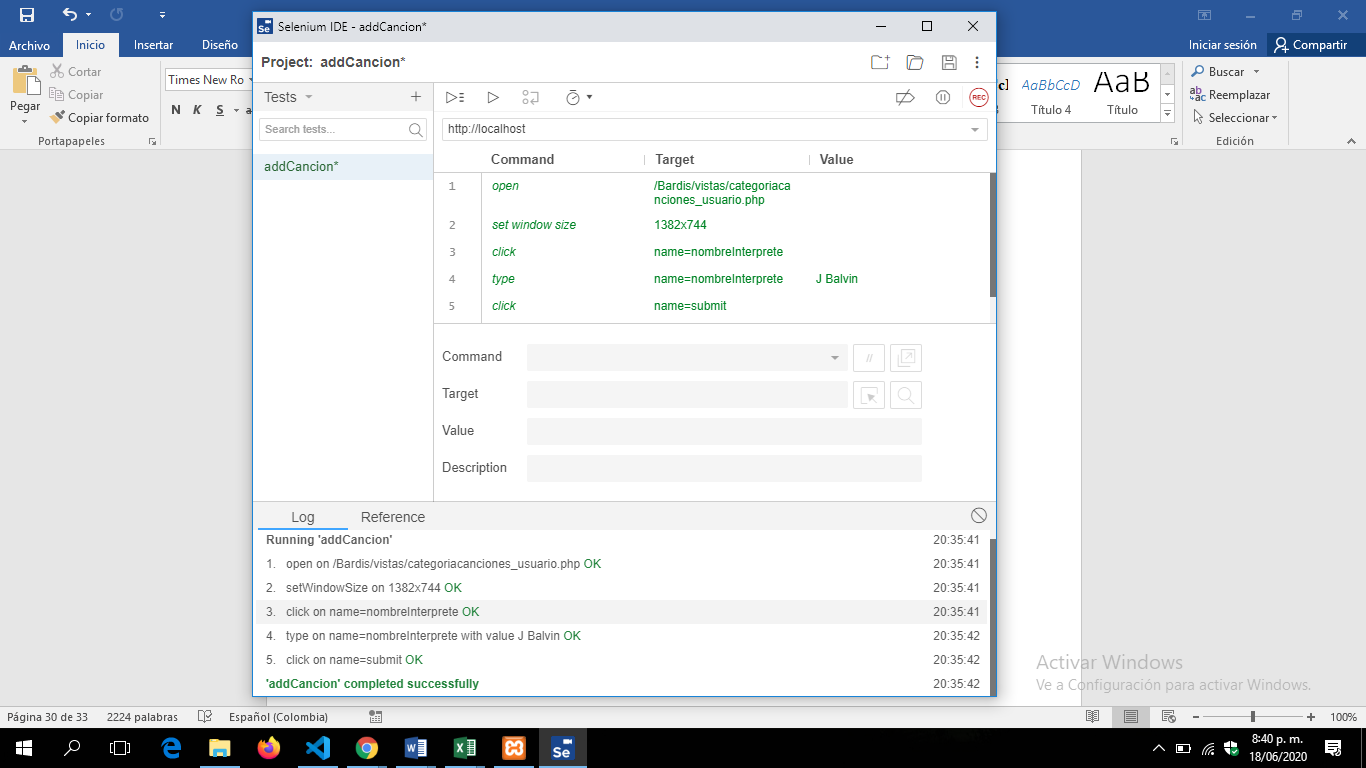
**Resultado esperado(Salida):** La canción no se solicitó.

Se ejecuta la prueba automatizada con los siguientes valores:



**Resultado obtenido:** La canción no se solicitó.

**Estado:** Ok.



**Caso de prueba #2:** Solicitar una canción llenando solo el campo del nombre de la canción.

**Nombre:** Solicitar canción.

**Descripción del caso:** Se realizarán pruebas para verificar si se solicita una canción correctamente.

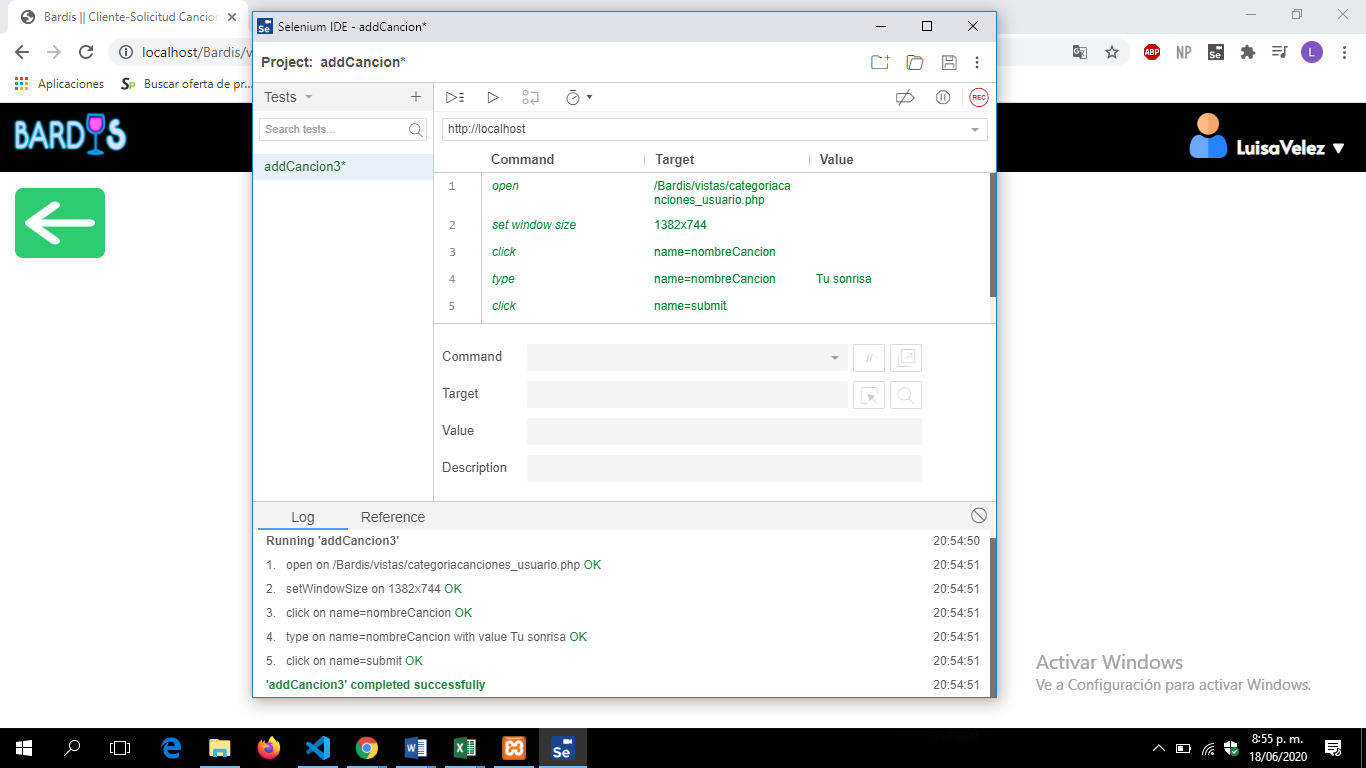
**Caso 2.1: *Datos de entrada:*** El formulario para solicitar una canción se encuentra vacío.

**Resultado esperado(Salida):** El sistema le solicita al cliente los datos para solicitar la canción.

**Caso 2.2: Datos de entrada:** El usuario cliente está ingresando solamente el nombre de la canción.

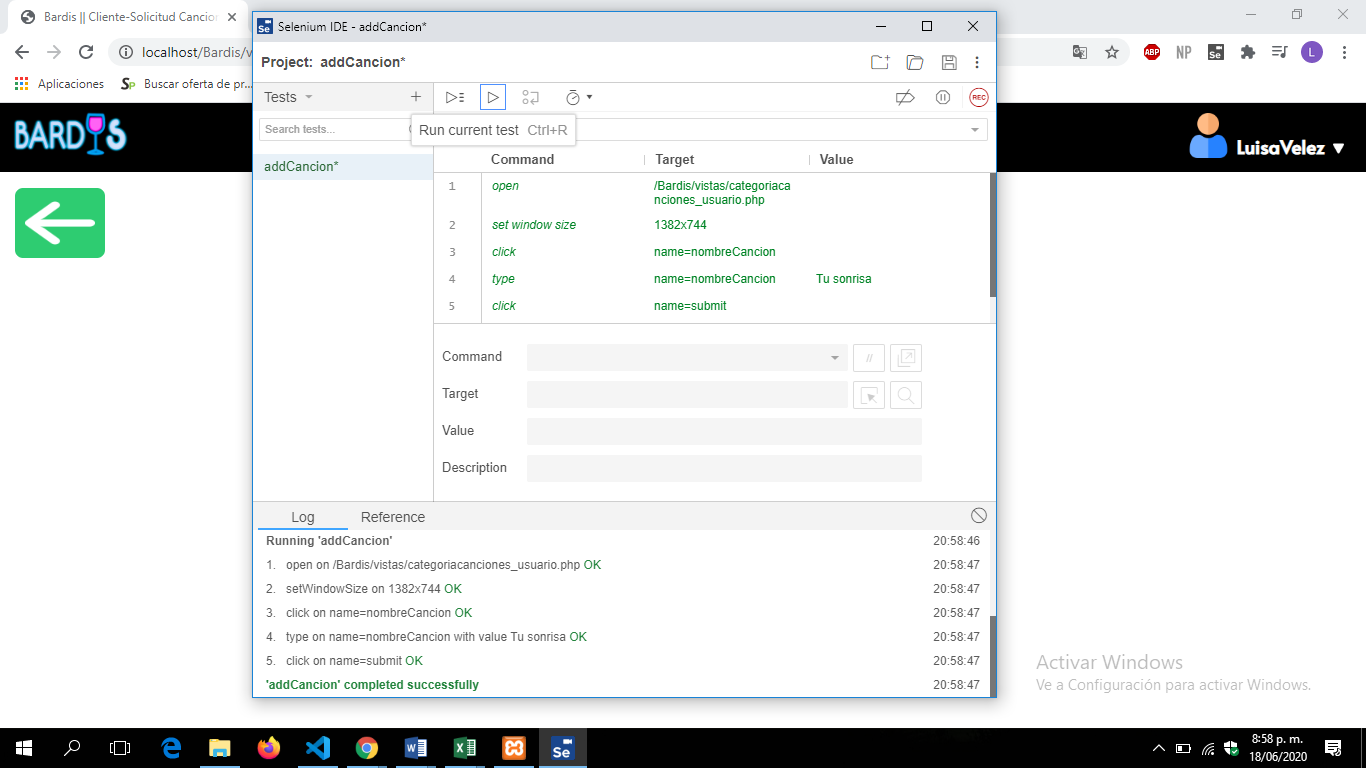
**Resultado esperado(Salida):** La canción se solicitó correctamente.

Se ejecuta la prueba automatizada con los siguientes valores:



**Resultado obtenido:** La canción se solicitó correctamente.

**Estado:** Ok.



**Caso de prueba #3:** Solicitar una canción llenando todos los campos.

**Nombre:** Solicitar canción.

**Descripción del caso:** Se realizarán pruebas para verificar si se solicita una canción correctamente.

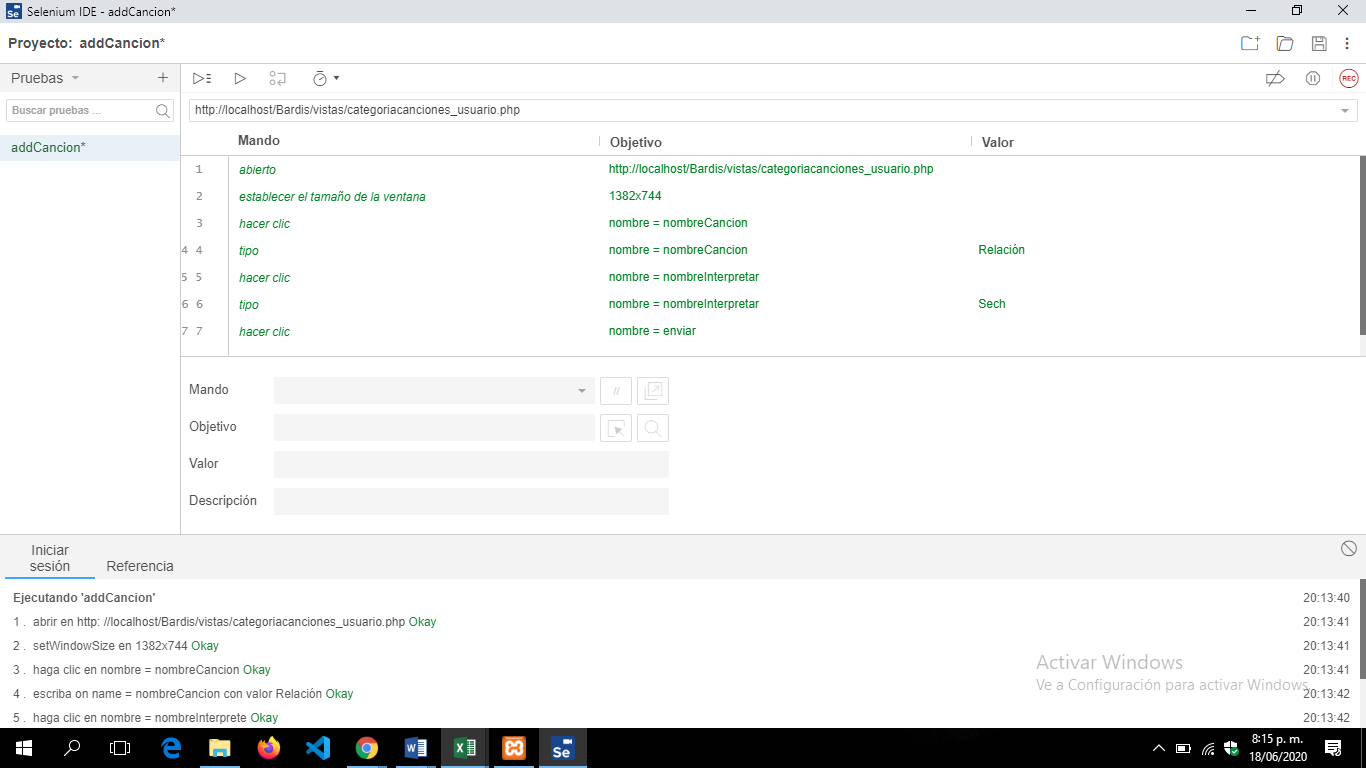
**Caso 3.1: *Datos de entrada:*** El formulario para solicitar una canción se encuentra vacío.

**Resultado esperado(Salida):** El sistema le solicita al cliente los datos para solicitar la canción.

**Caso 3.2: Datos de entrada:** El usuario cliente está llenando los datos que se le piden

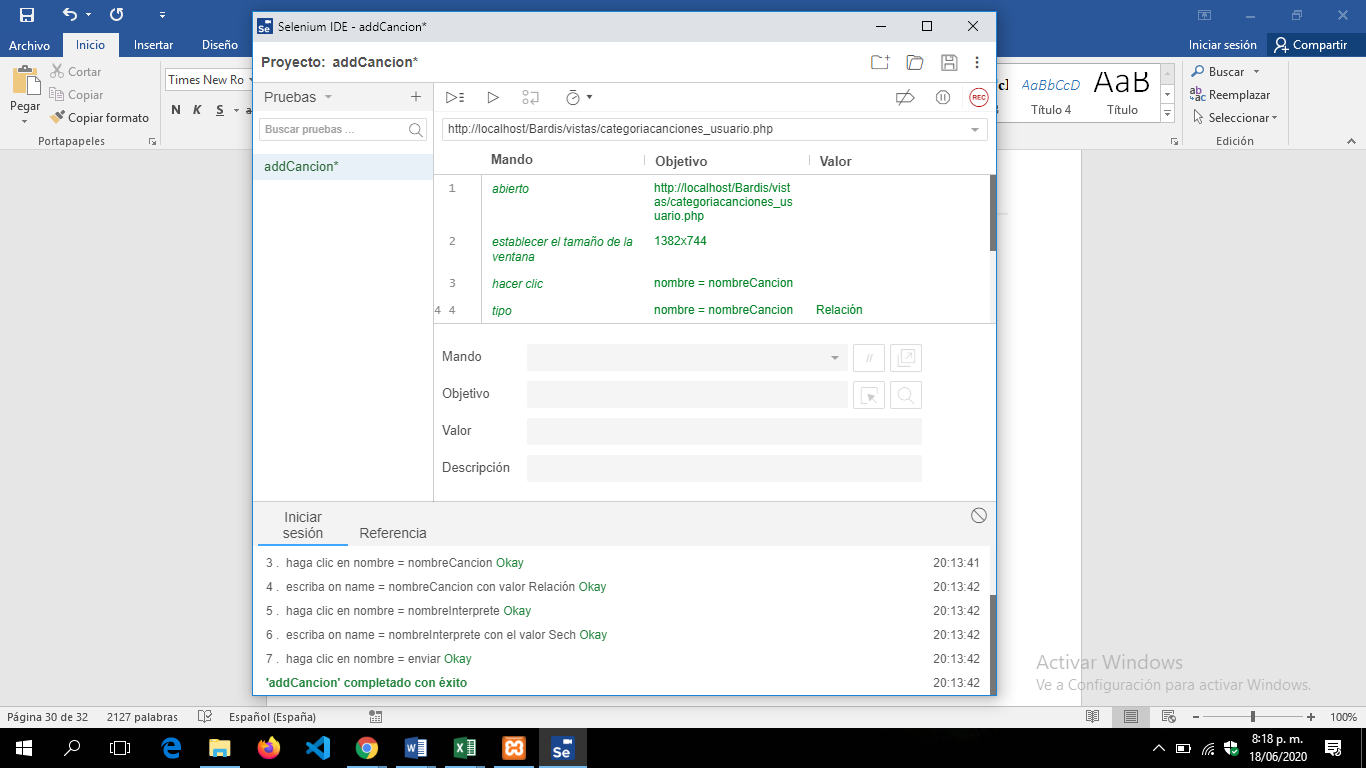
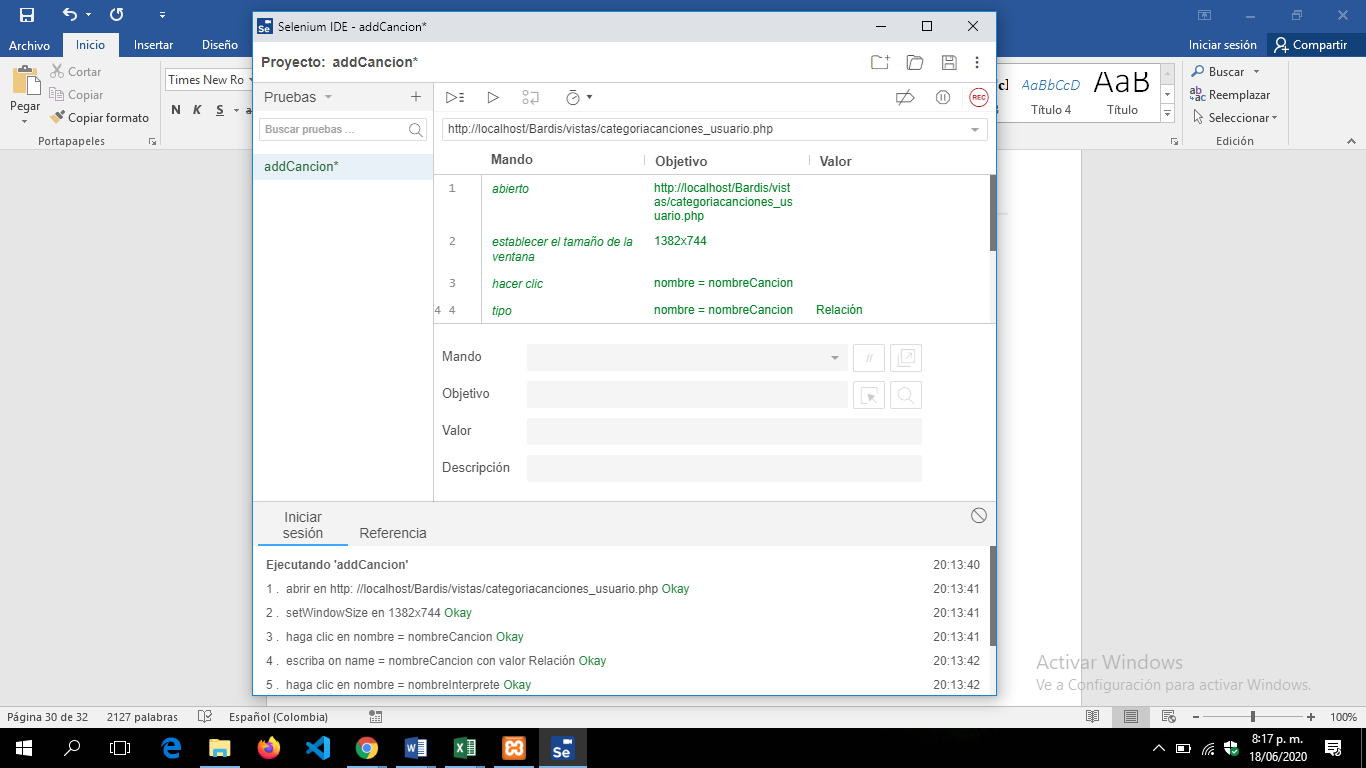
**Resultado esperado(Salida):** La canción se solicitó correctamente.

Se ejecuta la prueba automatizada con los siguientes valores:



**Resultado obtenido:** La canción se solicitó correctamente.

**Estado:** Ok.



# Resultados pruebas

### Pruebas de caja blanca y de caja negra:

En estas pruebas, en el módulo pedido se encontró un error, donde se evidenció que se permitía enviar un pedido sin productos, lo que generaba conflicto a la hora de atender los pedidos, para lo cual se realizó una intervención en el código que permite registrar un pedido, donde se validó que el pedido no estuviera vacío.

### Casos de prueba:

En los casos de prueba los errores encontrados fueron de validación, es decir el sistema permitía enviar algunos formularios con campos vacíos pero que realmente estos eran obligatorios, por esta razón se realizaron validaciones en los formularios de agregar empleado e inicio de sesión.

### Pruebas automatizadas:

Las pruebas automatizadas se realizaron correctamente ya que los errores mencionados anteriormente fueron solucionados, por lo tanto, no se encontraron errores en estas, la ejecución de las pruebas automatizadas, arrojaron los resultados esperados.