|  |  |
| --- | --- |
| **Pybuys**  25/05/2023 | **Descripción breve**  Pybuys es una aplicación de comercio electrónico intuitiva y completa para la venta de productos tecnológicos, que brinda una experiencia de compra satisfactoria y eficiente.  **Tutores del proyecto** Miguel Ángel González Santos  Gregorio Granja Granja  **VÍCTOR GONZÁLEZ COBOS**  2º Desarrollo de Aplicaciones Multiplataforma. |

# Estudio del problema y análisis del sistema

## Descripción del software

**Resumen:**

PyBuys es una solución de comercio electrónico integral diseñada para satisfacer las necesidades de las pequeñas y medianas empresas. Con PyBuys, estas empresas pueden establecer su presencia en línea de manera rápida y eficiente, ofreciendo sus productos a una audiencia más amplia.

**Detalles:**

Desarrollado sobre la sólida plataforma de Django y utilizando Python como lenguaje de programación, PyBuys ofrece una amplia gama de características que se adaptan a las necesidades de las empresas en crecimiento. Las características principales incluyen la gestión de cuentas de usuario, la navegación y búsqueda de productos, la gestión del carrito de compras, así como la posibilidad de añadir, modificar y eliminar productos.

Una de las fortalezas de PyBuys radica en su robusto sistema de gestión de cuentas. Permite a los nuevos usuarios crear cuentas fácilmente, iniciar sesión y modificar la información de su cuenta con facilidad. Además, la plataforma valida la información ingresada por los usuarios para asegurar su veracidad y seguridad.

El sistema de gestión de productos de PyBuys es otra característica destacada. No sólo permite a los usuarios buscar y seleccionar productos, sino que también permite a los administradores y al personal gestionar el inventario de manera eficaz. Esto incluye la adición de nuevos productos, la modificación de la información de los productos existentes y la actualización de las existencias de los productos.

También ofrece la búsqueda de productos por su nombre, y una gran gestión de compras y ventas para el staff y administradores.

Pybuys ofrece una increíble modularidad que permite a cualquier pequeña o mediana empresa iniciar su camino como e-commerce en cuestión de segundos, si bien está enfocada a productos tecnológicos nada impide que pueda ser usada para cualquier otro tipo de productos.

Ha sido desarrollada en Python, ofrece una solución para guardar toda tu información en MySQL (fácilmente migrable a cualquier otro tipo de bdd gracias a su modularidad). Ha sido creada para que el servidor tenga Windows.

**Campos de aplicación y conexión con el ciclo:**

Los campos de aplicación de PyBuys son variados y amplios, centrados principalmente en el sector del comercio electrónico. Esta plataforma puede ser utilizada por cualquier pequeña o mediana empresa (PYME) que desee establecer o expandir su presencia en línea. Esto puede abarcar una amplia gama de industrias, desde la moda y los electrodomésticos hasta la tecnología y los libros, entre otros.

El potencial de PyBuys también se extiende a empresas que buscan digitalizar sus operaciones internas, como la gestión de inventario y el seguimiento de las ventas. La plataforma puede ayudar a estas empresas a optimizar sus operaciones, mejorar la eficiencia y reducir los errores.

En términos de la conexión con el ciclo de formación de Desarrollo de Aplicaciones Multiplataforma (DAM), PyBuys se alinea estrechamente con varias áreas clave de estudio.

* Back end: El sistema se basa en Python y Django, lo que permite la implementación de una lógica de negocio sólida, la gestión de las interacciones del usuario, y la conexión con la base de datos. PyBuys utiliza MySQL para almacenar y gestionar los datos de los usuarios, los productos y las transacciones, ofreciendo a los estudiantes la oportunidad de aplicar y expandir sus conocimientos en el manejo de bases de datos.
* Front end: PyBuys se centra en proporcionar una interfaz de usuario intuitiva y atractiva, lo que requiere habilidades de desarrollo front end. Aunque Django maneja gran parte del trabajo de back end, los desarrolladores de PyBuys necesitan entender cómo presentar los datos a los usuarios de una manera que sea fácil de entender y de navegar.
* Full stack: Finalmente, PyBuys representa un ejemplo de desarrollo full stack, ya que combina tanto el desarrollo de front end como de back end para crear una aplicación web completa.

## Motivación y finalidad

El comercio electrónico ha experimentado un crecimiento exponencial en los últimos años, y se espera que continúe en esta trayectoria en el futuro previsible. Sin embargo, a pesar del gran potencial que ofrece el comercio electrónico, muchas pequeñas y medianas empresas (PYMES) todavía luchan por establecer su presencia en línea y aprovechar este canal de ventas.

La motivación detrás de PyBuys es ayudar a estas PYMES a superar los desafíos asociados con la creación de una tienda en línea y facilitar su entrada al mercado digital. A través de su interfaz amigable, su conjunto de características bien pensadas y su flexibilidad de configuración, PyBuys permite a las PYMES establecer su tienda en línea de manera rápida y eficiente, sin necesidad de tener un conocimiento técnico profundo.

La finalidad de PyBuys es ser un catalizador para el crecimiento de las PYMES en el espacio digital. Al proporcionar una solución de comercio electrónico asequible y fácil de usar, PyBuys puede ayudar a estas empresas a aumentar su alcance de mercado, mejorar sus ventas, y finalmente, mejorar su rentabilidad.

Además, PyBuys está diseñado para ser escalable y adaptable a las necesidades cambiantes de las empresas. A medida que la empresa crece, PyBuys puede crecer con ella, lo que permite a las PYMES mantener su tienda en línea actualizada y relevante para su base de clientes en constante expansión.

En resumen, la motivación y finalidad de PyBuys es ser el socio de confianza para las PYMES en su viaje de comercio electrónico, ofreciéndoles una plataforma robusta y versátil que puede facilitar su transición al espacio digital y apoyar su crecimiento continuo.

## Requisitos

* **Requisito de Sistema (RS1):** El sistema debe permitir a los nuevos usuarios crear su propia cuenta.
  + Requisito Funcional (RF1): La plataforma debe validar que el correo electrónico ingresado tenga un formato válido y que no esté ya en uso.
  + Requisito No Funcional (RNF1): La creación de la cuenta debe ser rápida y no debe demorar más de unos segundos.
* **RS2:** El sistema debe permitir a los usuarios registrados iniciar sesión.
  + Requisito Funcional (RF2): La plataforma debe verificar que la combinación de correo electrónico y contraseña sea correcta antes de permitir el inicio de sesión.
* **RS3:** El sistema debe permitir a los usuarios modificar la información de su cuenta.
  + Requisito Funcional (RF3): La plataforma debe validar que todos los campos de información del usuario sean completados antes de permitir la modificación.
  + Requisito No Funcional (RNF3): Las modificaciones a la cuenta del usuario deben ser guardadas y reflejadas inmediatamente en la plataforma.
* **RS4:** El sistema debe permitir a los usuarios agregar, modificar y eliminar productos de su carrito.
  + Requisito Funcional (RF4): La plataforma debe verificar que hay stock suficiente de un producto antes de permitir que se añada al carrito.
  + Requisito No Funcional (RNF4): Las modificaciones al carrito de un usuario deben ser guardadas y reflejadas inmediatamente en la plataforma.
* **RS5:** El sistema debe permitir a los usuarios del staff o administradores añadir nuevos productos.
  + Requisito Funcional (RF5): La plataforma debe validar que todos los campos de información del producto sean completados antes de permitir la adición de un nuevo producto.
  + Requisito No Funcional (RNF5): La adición de nuevos productos debe ser registrada y reflejada inmediatamente en la plataforma.
* **RS6:** El sistema debe permitir a los usuarios del staff o administradores añadir o quitar stock.
  + Requisito Funcional (RF6): La plataforma debe validar que la cantidad de stock a añadir o quitar sea un número válido y que no exceda la cantidad de stock disponible.
  + Requisito No Funcional (RNF6): Las modificaciones al stock de un producto deben ser registradas y reflejadas inmediatamente en la plataforma.
* **RS7:** El sistema debe permitir a los usuarios del staff o administradores visualizar las compras de stock realizadas.
  + Requisito Funcional (RF7): La plataforma debe mostrar todas las compras realizadas en orden cronológico, incluyendo la fecha, producto y cantidad.
  + Requisito No Funcional (RNF7): La visualización de las compras de stock debe ser actualizada en tiempo real para reflejar las últimas transacciones.
* **RS8:** El sistema debe permitir a cualquier usuario registrado buscar un producto.
  + Requisito Funcional (RF8): La plataforma debe proporcionar resultados de búsqueda relevantes basados en el término de búsqueda ingresado por el usuario.
  + Requisito No Funcional (RNF8): El sistema de búsqueda debe ser rápido y debe proporcionar resultados en tiempo real a medida que el usuario escribe su búsqueda.

# Recursos necesarios para el desarrollo

## Para el desarrollo

**Hardware:**

* Sistema Operativo: Windows 10
* Procesador: Intel Core i5
* Memoria: 12 GB de RAM
* Almacenamiento: 100 GB de espacio libre en disco

**Software:**

* Sistema Operativo: Windows 10
* Python 3.8 o superior
* MSSQL Server Express Edition
* Dependencias instaladas: Django, django-mssql-backend, Pillow

## Para su ejecución

### Servidor

**Hardware:**

* Sistema Operativo: Windows 10 o superior.
* Procesador: CPU de 2 o 3 núcleos
* Memoria: 4 GB de RAM
* Almacenamiento: 10 GB de espacio libre en disco

**Software:**

* Sistema Operativo: Windows 10 o superior
* Python 3.7 o superior
* MSSQL Server Express Edition
* Dependencias instaladas: Django, django-mssql-backend, Pillow

### Cliente

**Hardware:**

* Navegador web actualizado (Chrome, Firefox, Safari, etc.)

**Software:**

* Navegador web actualizado (Chrome, Firefox, Safari, etc.)

# Desarrollo del proyecto

## Estimación de tiempos

|  |  |  |
| --- | --- | --- |
| Fase de desarrollo | Tiempo estimado | Tiempo realizado |
| Análisis | 13 | 10 |
| Diseño | 17 | 20 |
| Codificación | 24 | 20 |
| Pruebas | 27 | 10 |
| Total | 81 | 60 |

Para desarrollo total del proyecto he necesitado alrededor de unas 60 horas totales.

## Trabajo realizado.

### Análisis.

Recoger requisitos, plantear herramientas a usar y estimar tiempos. Elegir framework a usar y elección de lenguaje de programación.

### Diseño

Diseñar base de datos, diagrama de clases, casos de uso. Estructurar como iba a ser la codificación en una aplicación de django y como podía adaptar los requisitos a este framework.

Diseño de interfaces y creación de logo.

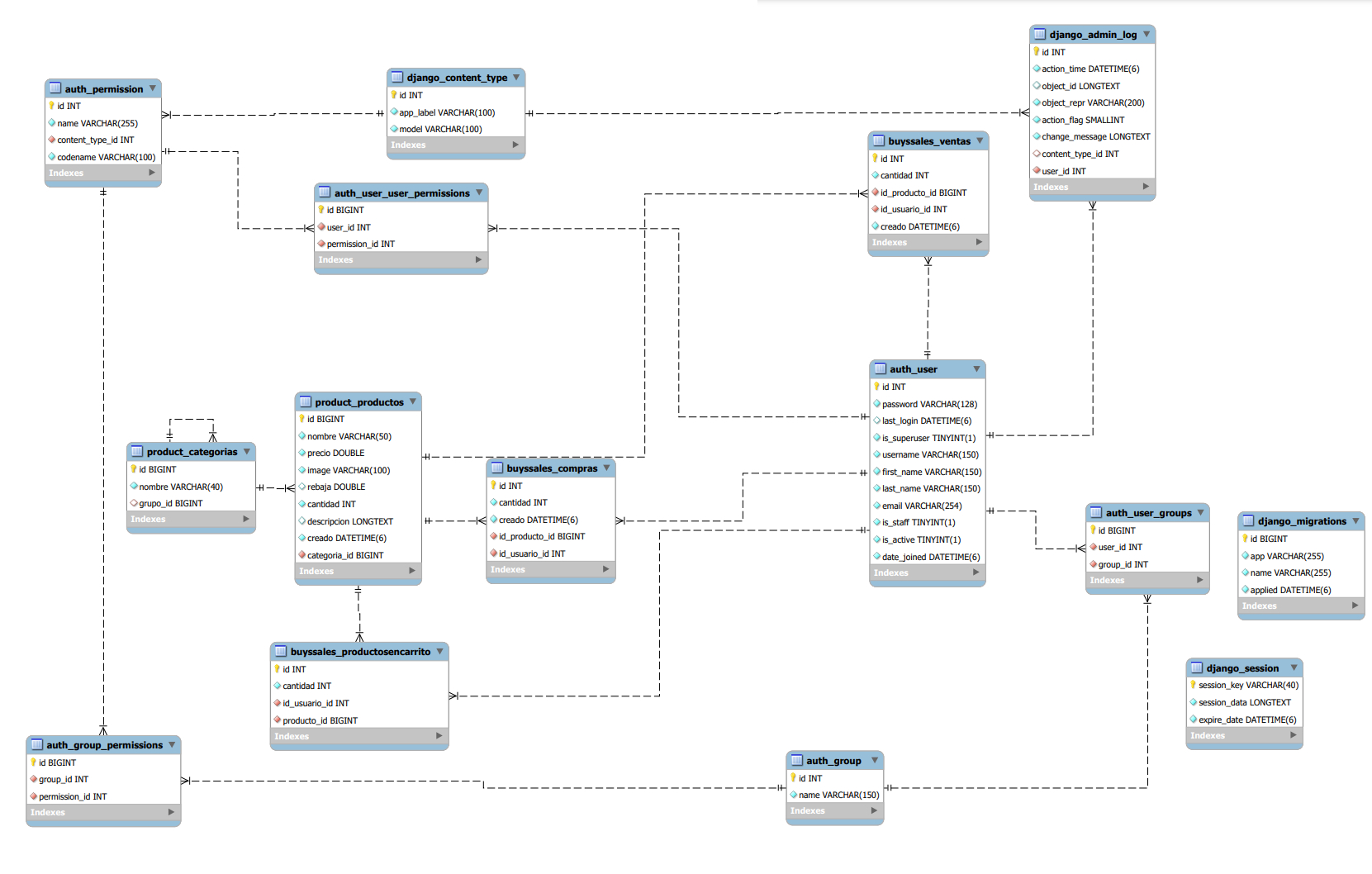
### Codificación

Implementación de funcionalidades al código. Adaptar apartado de administrador para las funcionalidades. E implementar interfaces diseñadas con anterioridad.

### Pruebas

Probadas todas las funcionalidades implementadas. Hechas pruebas unitarias en código y pruebas integradas, de pico, estabilidad y estrés documentadas en Excel.

## Diseño de datos

**Diagrama relacional**

## Diagrama de clases

## Diagrama de flujoDiagrama Descripción generada automáticamente

## Interacción con el usuario

### Casos de uso

Diagrama

Descripción generada automáticamente**Diagrama de casos de uso**

**Casos de uso detallados**

|  |  |
| --- | --- |
|  | CU1 – Crear cuenta |
| Descripción | Un usuario nuevo podrá crear su propia cuenta. Al iniciar sesión o registrarse, el usuario es redirigido a la página de inicio. |
| Actores | Nuevo usuario |
| Precondiciones | Ninguna |
| Pasos del caso de uso | 1. El nuevo usuario ingresa su nombre de usuario, dirección de correo electrónico, contraseña y repite la contraseña en la pantalla de registro. 2. El usuario hace clic en "Crear cuenta". 3. Si algún campo obligatorio está vacío o no se cumple con los requisitos básicos (el formato del correo electrónico), se muestra un mensaje de error correspondiente y se solicita al usuario que complete correctamente los campos. 4. Si todos los campos están vacíos y se cumplen requisitos básicos, se envía la información para que el servidor la procese. 5. Si el correo electrónico o el nombre de usuario ya existen en el sistema, se muestra un mensaje de error indicando que ya están en uso. El usuario es redirigido a la página de crear cuenta. 6. Si la contraseña no cumple con los requisitos de seguridad establecidos en Django (por ejemplo, debe contener al menos 8 caracteres, incluir letras y números), se muestra un mensaje de error y se solicita al usuario que elija una contraseña más segura. El usuario es redirigido a la página de crear cuenta. 7. Si todos los campos se completan correctamente y la contraseña cumple con los requisitos, se crea la cuenta del usuario y se le redirige a la página de inicio de sesión. |

|  |  |
| --- | --- |
|  | CU2– Inicio de sesión |
| Descripción | El usuario introduce sus datos previamente registrados para poder acceder. |
| Actores | Usuario registrado |
| Precondiciones | El usuario debe tener una cuenta registrada previamente para poder iniciar sesión con éxito. |
| Pasos del caso de uso | 1. El usuario ingresa su nombre de usuario y contraseña en la pantalla de inicio de sesión. 2. El usuario hace clic en "Iniciar sesión". 3. El servidor procesa la información. 4. Si la combinación de nombre de usuario y contraseña es incorrecta, se muestra un mensaje de error indicando que la contraseña es incorrecta. El usuario puede intentar nuevamente ingresando la contraseña correcta. 5. Si el nombre de usuario y la contraseña coinciden con los registros del sistema, el inicio de sesión tiene éxito y el usuario es redirigido a la página de inicio de la aplicación. |

|  |  |
| --- | --- |
|  | CU3 – Modificar información de la cuenta |
| Descripción | El usuario modifica los datos de su cuenta. |
| Actores | Usuario registrado |
| Precondiciones | Tener una cuenta creada y estar con la sesión iniciada |
| Pasos del caso de uso | 1. El usuario inicia sesión en la página web. 2. En la interfaz de usuario, el usuario selecciona la opción "Configuración de cuenta". 3. Se muestra un formulario con los datos actuales del usuario, incluyendo su nombre y dirección de correo electrónico. 4. El usuario realiza las modificaciones necesarias en los campos que desea cambiar. 5. Si el correo no es correcto no permite enviar la información. 6. Al intentar guardar los cambios, se realizan las siguientes validaciones: 7. a. Si algún campo está vacío, se muestra un mensaje de error indicando que todos los campos deben ser completados. 8. b. Si el usuario intenta cambiar su dirección de correo electrónico y esa dirección ya está en uso por otro usuario, se muestra un mensaje de error indicando que la dirección de correo electrónico ya está en uso. 9. Si todas las validaciones son exitosas, se actualizan los datos de la cuenta del usuario en la base de datos. 10. Se muestra un mensaje de éxito indicando que los cambios se han guardado correctamente. 11. El usuario puede continuar utilizando la aplicación con los datos actualizados en su cuenta. |

|  |  |
| --- | --- |
|  | CU4 – Añadir/Modificar/ Eliminar del carrito |
| Descripción | El usuario hace operaciones en el carrito. |
| Actores | Usuario con sesión iniciada. |
| Precondiciones | El usuario debe haber iniciado sesión en el sistema. |
| Pasos del caso de uso | 1. El usuario navega a la ventana de detalle de un producto y desde allí tiene la opción de añadir el producto a su carrito si hay stock disponible. 2. El usuario puede acceder a la ventana del carrito, donde puede realizar modificaciones en los elementos añadidos, como modificar la cantidad (+, -), siempre y cuando haya stock disponible. 3. Después de realizar las modificaciones deseadas, el usuario selecciona el botón de "Actualizar carrito" para confirmar los cambios. 4. Si el usuario decide eliminar un producto del carrito, puede seleccionar la opción correspondiente para eliminarlo por completo. 5. El usuario puede continuar navegando por la página y repetir los pasos anteriores para añadir, modificar o eliminar más elementos en su carrito. 6. Cuando el usuario está satisfecho con los elementos en su carrito y desea proceder con la compra, selecciona la opción de "Comprar". 7. Una vez finalizada la compra, se genera una venta con los productos seleccionados. 8. El usuario recibe una notificación de compra realizada. |
|  | CU5 – Añadir producto |
| Descripción | El usuario del staff o administrador podrá añadir nuevos productos |
| Actores | Usuario del staff o administrador |
| Precondiciones | Haber iniciado sesión y entrado en el panel de administrador. |
| Pasos del caso de uso | 1. El usuario del staff o administrador selecciona la opción "Añadir producto" desde el panel de administrador. 2. Se muestra un formulario que permite al usuario ingresar los campos básicos del producto, como nombre, descripción, precio y cantidad disponible. Todos los campos esenciales deben ser completados para poder continuar. 3. El usuario completa todos los campos requeridos y selecciona la opción "Confirmar" o "Guardar". 4. Se verifica si toda la información necesaria está completa. Si falta algún campo esencial, se muestra un mensaje de error indicando qué campo debe ser completado. 5. Si toda la información requerida está completa, se guarda el producto en el sistema y se realiza una compra inicial con la cantidad especificada. Esto significa que se añade el producto al inventario y se registra una transacción de compra con la cantidad ingresada. 6. Una vez que se ha guardado el producto exitosamente, se muestra un mensaje de confirmación y se redirige al usuario a la página de gestión de productos. |

|  |  |
| --- | --- |
|  | CU6 – Añadir/Quitar stock |
| Descripción | El usuario del staff o administrador podrá añadir nuevos productos |
| Actores | Usuario del staff o administrador |
| Precondiciones | El usuario debe haber iniciado sesión como staff o administrador. |
| Pasos del caso de uso | 1. El usuario del staff o administrador selecciona un producto y accede a su detalle. 2. En la página de detalle del producto, se muestra un contador que refleja la cantidad de stock actual. El contador tiene un valor mínimo igual a la cantidad de stock actual en negativo, lo que permite realizar operaciones para quitar stock. 3. El usuario introduce la cantidad adecuada de stock que desea añadir o quitar en el contador. 4. El usuario hace clic en "Añadir stock" para confirmar la operación. 5. Si se añade o quita el stock correctamente, se muestra un mensaje de confirmación indicando que la operación se ha realizado exitosamente. 6. Si se intenta realizar una operación inválida, por ejemplo, si un cliente compra una cantidad mayor a la que se intenta quitar, se muestra un mensaje de error. En este caso, el usuario puede actualizar la página y repetir los pasos de nuevo. |

|  |  |
| --- | --- |
|  | CU7 – Visualización de compras de stock |
| Descripción | El usuario del staff o administrador podrá ver las compras de stock realizadas |
| Actores | Usuario del staff o administrador |
| Precondiciones | Haber iniciado sesión y entrar en el apartado de “Compras” |
| Pasos del caso de uso | 1. El usuario del staff ingresa al apartado de "Compras" y puede ver todas sus compras realizadas en orden cronológico. Las compras se muestran en una tabla con las siguientes columnas: fecha, producto y cantidad. 2. El administrador ingresa al apartado de "Compras" y puede ver todas las compras realizadas por todos los usuarios del sistema, incluyendo el nombre de quien las realizó. Las compras se presentan en una tabla similar a la del usuario del staff, con las mismas columnas mencionadas anteriormente. 3. Tanto el usuario del staff como el administrador tienen la capacidad de ordenar la tabla por cualquier columna. Al hacer clic en el encabezado de una columna, las compras se reorganizan en función de los valores de esa columna en orden ascendente o descendente. |

|  |  |
| --- | --- |
|  | CU8 – Búsqueda de un producto |
| Descripción | Cualquier usuario con sesión iniciada podrá buscar el producto que busca. |
| Actores | Usuario. |
| Precondiciones | Haber iniciado sesión. |
| Pasos del caso de uso | 1. El usuario ingresa en la barra de búsqueda el término o nombre del producto que desea buscar. 2. El sistema realiza una búsqueda en la base de datos en busca de todas las coincidencias de productos existentes con el término ingresado. Estas coincidencias pueden incluir productos que actualmente no tengan stock disponible. 3. El sistema muestra los resultados de la búsqueda, presentando al usuario las coincidencias encontradas. Para cada resultado, se el nombre del producto. 4. El usuario puede seleccionar una de las coincidencias de productos mostradas haciendo clic en ella. 5. Al seleccionar un producto de la lista de resultados, el sistema realizará una búsqueda con ese nombre específico, o simplemente si le da a buscar realiza una búsqueda con el patrón introducido en la barra de búsqueda. |

**Diagramas de secuencia**

**CU1- Crear cuenta**

Tabla

Descripción generada automáticamente con confianza baja

**CU2- Inicio de sesión**

Diagrama, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica

Descripción generada automáticamente**CU3-** **Modificar información de la cuenta**

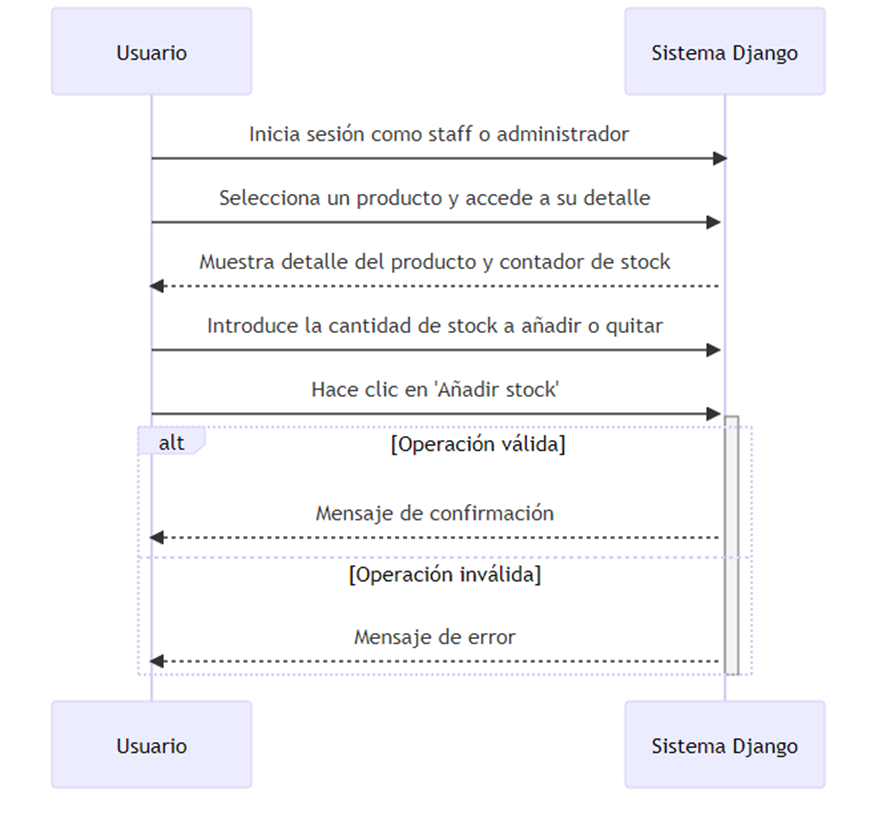
**CU4 -** **Añadir/Modificar/ Eliminar del carrito**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**CU5 – Añadir producto**

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**CU6 – Añadir/Quitar stock**

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**CU7 – Visualización de compras de stock**

Diagrama

Descripción generada automáticamente**CU8 – Búsqueda de un producto**

4.2.2.Diseño de la interfaz

4.3. Código fuente. Aquellos procesos, métodos y/o funciones más representativas del

proyecto, ficheros de configuración, scripts de instalaciones… en el caso de que el

proyecto incluya el desarrollo de una aplicación informática.

5. Fase de pruebas (las que resulten necesarias según el tipo de proyecto)

5.1. Pruebas realizadas (descripción de la prueba e historia de la misma (datos de prueba,

resultado y solución).

Creación de una batería de pruebas (tanto para describir errores lógicos como

posibles mejoras en la ejecución) que corroboran de forma razonable el buen

funcionamiento del sistema.

6. Conclusiones finales

Propuesta de modificaciones o ampliaciones futuras del sistema implementado.

7. Documentación del sistema desarrollado (en anexos)

7.1. Manual de instalación

7.2. Manual de uso

8. Bibliografía

Incluirá toda la documentación consultada: libros, apuntes, páginas web, foros…9. Anexos

Se enumeran en orden de uso y numeración romana (Anexo I, Anexo II…) Los elementos

que el alumno utilice para la exposición y defensa del proyecto se incluirán también como

anexos.

FUNCIONALIDAD DE LA APLICACIÓN Y COMO ESTA HECHA