



LAUREATE INTERNATIONAL UNIVERSITIES®

Universidad del Valle de México

MOD Y MET para el Desarrollo de Software

Historias de usuario

Docente:

Julio César Villa López

Alumnos

Jesús Donaldó Guerrero Palacios - 440255348

Eduardo Alberto Tarazón García - 440267148

Monterrey, Nuevo León

27/06/2024

Historias de usuario generadas

Historia de usuarios generadas

- Dicha interfaz debe permitir guardar, actualizar, eliminar y limpiar

Historia de usuario	
Número: 1	Usuario: Administrador
Nombre historia: Funciones principales de la interfaz	
Prioridad en negocio: Alta	Riesgo en desarrollo: Bajo
Puntos estimados: 5	Iteración asignada: Sprint 1
Programador responsable: Eduardo Tarazón	
Descripción: Como usuario administrador del sistema, quiero poder registrar un nuevo producto en la base de datos, ingresando su código único, nombre descripción, precio, cantidad en existencia y asignándole una categoría, para mantener un inventario actualizado y categorizado de los productos disponibles.	
Observaciones: La interfaz debe ser intuitiva y fácil de usar para agilizar el proceso de registro de productos. <ul style="list-style-type: none">◦ Se debe validar que el código del producto sea único en la base de datos para evitar duplicados.◦ Los campos como precio y existencias deben aceptar solo valores numéricos válidos.◦ La categoría del producto debe seleccionarse de una lista predefinida para mantener la consistencia en la base de datos.	

- Se necesita que almacene los datos

Historia de usuario	
Número: 2	Usuario: Administrador
Nombre historia: Almacenamiento en base de datos	
Prioridad en negocio: Media	Riesgo en desarrollo: Medio
Puntos estimados: 7	Iteración asignada: Sprint 1
Programador responsable: Jesús Donaldo	

Descripción: Como desarrollador de backend, necesito crear y configurar la estructura de la base de datos para almacenar la información de los productos, incluyendo su código único, nombre, descripción, precio, cantidad en existencia y categoría asociada, de manera que el sistema pueda gestionar eficientemente los datos relacionados con los productos.

Observaciones:

- Se deben definir correctamente las tablas, campos y relaciones necesarias en la base de datos para almacenar la información de los productos de forma coherente y eficiente.
- Se debe implementar la lógica necesaria para garantizar la integridad de los datos, como la validación de campos obligatorios y la gestión de relaciones entre tablas.
- Se debe considerar la escalabilidad del sistema al diseñar la estructura de la base de datos, para permitir futuras expansiones y modificaciones sin comprometer el rendimiento.
- Se deben establecer los índices adecuados para optimizar las consultas relacionadas con la gestión de productos, como búsquedas por código, nombre o categoría.

- Se debe identificar el inicio de sesión y se debe controlar a quien inicia sesión

Historia de usuario	
Número: 3	Usuario: Administrador
Nombre historia: Inicio de sesión	
Prioridad en negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 5	Iteración asignada: 2
Programador responsable: Eduardo Tarazón	
Descripción: Como usuario del sistema, quiero poder iniciar sesión utilizando mis credenciales de usuario (nombre de usuario y contraseña) para acceder a las funcionalidades del sistema y realizar acciones como registrar productos. Además, el sistema debe controlar el acceso y autenticar a los usuarios correctamente, permitiendo solo a usuarios autorizados iniciar sesión.	
Observaciones: <ul style="list-style-type: none">• Se deben implementar medidas de seguridad, como el almacenamiento seguro de contraseñas utilizando técnicas de hash y salting, para proteger la información de los usuarios.	

- Se deben manejar los casos de usuarios bloqueados o desactivados, impidiendo su acceso al sistema durante el inicio de sesión.
Se debe proporcionar retroalimentación clara al usuario en caso de credenciales incorrectas o problemas durante el inicio de sesión, para mejorar la experiencia del usuario.
Se debe registrar y auditar los intentos de inicio de sesión, incluyendo la hora y el resultado (éxito o fallo), para mantener un registro de actividad y detectar posibles intentos de acceso no autorizados.

- Codificación de la interfaz gráfica

Historia de usuario	
Número: 4	Usuario: Administrador
Nombre historia: Interfaz gráfica	
Prioridad en negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 5	Iteración asignada: Sprint 2
Programador responsable: Jesús Donaldo	
Descripción: Como desarrollador frontend, necesito codificar la interfaz gráfica del sistema para permitir a los usuarios registrar un nuevo producto, ingresando su código, nombre, descripción, precio, existencias y categoría asociada de manera intuitiva y eficiente. La interfaz debe ser coherente con el diseño visual establecido y garantizar una experiencia de usuario fluida	
Observaciones: • Se debe implementar la validación de campos en la interfaz para garantizar que los datos ingresados por el usuario sean correctos y cumplan con los requisitos establecidos <ul style="list-style-type: none"> • Se debe considerar la usabilidad de la interfaz, incluyendo la disposición de los elementos, la claridad de las etiquetas y la facilidad de navegación, para mejorar la experiencia del usuario. • Se deben realizar pruebas exhaustivas de la interfaz para identificar y corregir posibles errores o problemas de rendimiento antes de su implementación en el entorno de producción. 	

- Creación de esquema de base de datos y tablas con campos solicitados

Historia de usuario	
Número: 5	Usuario: Administrador de base de datos
Nombre historia: Creación de esquema de base de datos y tablas con campos solicitados	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Puntos estimados: 3	Iteración asignada: 2
Programador responsable: Eduardo Tarazón	
Descripción: Para la creación de nuestra base de datos con TiendaVirtual y Producto bajo los siguiente requerimientos solicitados... <ul style="list-style-type: none"> • Id tipo entero not null auto_increment • Código varchar 20 not null • Nombre varchar 45 not null • Descripción varchar 100 • Precio decimal 10..2 not null • Existencias entero not null 	
Observaciones: Se proporciona plantilla, pero se acepta modificaciones que aporten al diseño y estructura de está.	

- Edición, conexión de base de datos y el botón agregar

Historia de usuario	
Número: 5	Usuario: Administrador de base de datos
Nombre historia: Edición y conexión de base de datos y el botón agregar	
Prioridad en negocio: Baja	Riesgo en desarrollo: Media
Puntos estimados: 6	Iteración asignada: 3
Programador responsable: Jesús Donaldo	
Descripción: <ul style="list-style-type: none"> • Comprobar conexión con base de datos TiendaVirtual y tabla Producto • Codificar para que la interfaz se conecte y almacene de manera correcta con los registros de la base de datos TiendaVirtual fueron agregados a nuestra tabla Producto • Validar que los campos son correctos • Validar funcionamiento de interfaz 	
Observaciones: Validar que la interfaz se conecte cada vez que se abre el servicio	

--

Historias de Usuario

Botón Guardar

Historia de usuario	
Número: 1	Usuario: Administrador Base de Datos, Empleado
Nombre historia: Creación botón Guardar	
Prioridad en negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 5	Iteración asignada: 1
Programador responsable: Jesús Donaldo Guerrero Palacios	
Descripción: Botón de guardar en la interfaz principal para validar y almacenar datos ingresados en la base de datos. El botón debe ser claramente visible y etiquetado. Al hacer clic, los datos deben validarse. Si son válidos, se almacenan en la base de datos; en caso contrario, se muestra un mensaje de error claro. Se proporciona una confirmación visual cuando los datos se guardan con éxito.	
Observaciones: Realizar una revisión para asegurar que el botón sea intuitivo y accesible. Implementar pruebas unitarias y de integración exhaustivas. Considerar un sistema de backup o rollback para los datos guardados en caso de problemas inesperados.	

Botón Buscar

Historia de usuario	
Número: 2	Usuario: Administrador Base de Datos, Empleado
Nombre historia: Creación botón Buscar	
Prioridad en negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 3	Iteración asignada: 2
Programador responsable: Eduardo Alberto Tarazón García	

<p>Descripción: Botón de buscar en la interfaz principal que permite a los usuarios ingresar criterios de búsqueda y recuperar datos de la base de datos. El botón debe ser visible y fácil de identificar. Al hacer clic, debe ejecutar una consulta en la base de datos y mostrar los resultados correspondientes. En caso de error, debe mostrar mensajes claros y sugerencias para ajustar la búsqueda.</p>
<p>Observaciones: Realizar pruebas de rendimiento para asegurar que las consultas se ejecuten de manera eficiente. Implementar cambios para garantizar que el botón sea intuitivo. Implementar manejo de errores para consultas fallidas y considerar la paginación de resultados para mejorar la usabilidad</p>

Botón Reemplazar

Historia de usuario	
Número: 3	Usuario: Administrador Base de Datos, Empleado
Nombre historia: Creación botón Reemplazar	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 4	Iteración asignada: 2
Programador responsable: Eduardo Alberto Tarazón García	
<p>Descripción: Actualizar en la interfaz principal que permite a los usuarios modificar los datos existentes en la base de datos. El botón debe ser claramente visible y etiquetado. Al hacer clic, los datos ingresados deben ser validados y actualizados en la base de datos. En caso de error, se debe proporcionar un mensaje claro al usuario. Una confirmación visual debe mostrarse cuando los datos se hayan actualizado correctamente</p>	
<p>Observaciones: Realizar una revisión para asegurar que el botón sea intuitivo y accesible. Implementar pruebas unitarias y de integración exhaustivas. Considerar el manejo de conflictos de concurrencia y un sistema de rollback en caso de fallos durante la actualización</p>	

Botón eliminar

Historia de usuario	
Número: 1	Usuario: Administrador Base de Datos, Empleado
Nombre historia: Creación botón eliminar	
Prioridad en negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 3	Iteración asignada: 3
Programador responsable: Jesús Donaldo Guerrero Palacios	

Descripción: Botón de eliminar en la interfaz principal que permite a los usuarios eliminar datos específicos de la base de datos. El botón debe ser claramente visible y etiquetado. Al hacer clic, debe mostrar una confirmación de eliminación. Si se confirma, debe eliminar los datos correspondientes de la base de datos. En caso de error, debe mostrar un mensaje claro al usuario. Una confirmación visual debe mostrarse cuando los datos se hayan eliminado correctamente.

Observaciones: Realizar una revisión para asegurar que el botón sea intuitivo y accesible. Implementar un diálogo de confirmación para evitar eliminaciones accidentales. Realizar pruebas unitarias y de integración exhaustivas. Considerar un sistema de backup o undo para recuperarse de eliminaciones accidentales.

Botón limpiar

Historia de usuario	
Número: 2	Usuario: Administrador Base de Datos, Empleado
Nombre historia: Creación botón limpiar	
Prioridad en negocio: Media	Riesgo en desarrollo: Bajo
Puntos estimados: 2	Iteración asignada: 1
Programador responsable: Jesús Donaldo Guerrero Palacios	
Descripción: Botón de limpiar en la interfaz principal que permite a los usuarios restablecer todos los campos del formulario a su estado inicial. El botón debe ser claramente visible y etiquetado. Al hacer clic, debe limpiar todos los campos de entrada del formulario, permitiendo al usuario comenzar de nuevo sin recargar la página.	
Observaciones: Realizar una revisión para asegurar que el botón sea intuitivo y accesible. Implementar pruebas para asegurarse de que todos los campos se restablezcan correctamente. Considerar la adición de una confirmación opcional para evitar la limpieza accidental de datos ingresados.	

Validación de campos numéricos

Historia de usuario	
Número: 1	Usuario: Administrador Base de Datos, Empleado
Nombre historia: Validación de campos numéricos	
Prioridad en negocio: Alta	Riesgo en desarrollo: Bajo
Puntos estimados: 3	Iteración asignada: 1
Programador responsable: Jesús Donaldo Guerrero Palacios	

Descripción: Como usuario, quiero que los campos de código, precio y existencia acepten solo números, para asegurar la integridad de los datos ingresados y evitar errores en la base de datos. Esto implica que al ingresar valores en estos campos, la interfaz debe validar que únicamente se introduzcan caracteres numéricos. Si el usuario intenta ingresar caracteres no numéricos, la interfaz debe mostrar un mensaje de error inmediato indicando que solo se permiten números.

Observaciones: Esta funcionalidad ayudará a prevenir errores de entrada de datos y a mantener la consistencia en la base de datos. Implementar validaciones en tiempo real mejora la experiencia del usuario al proporcionar retroalimentación inmediata.

Confirmación y limpieza al borrar elementos

Historia de usuario	
Número: 2	Usuario: Administrador Base de Datos, Empleado
Nombre historia: Confirmación y limpieza al borrar elementos	
Prioridad en negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 5	Iteración asignada: 1
Programador responsable: Jesús Donaldo Guerrero Palacios	
Descripción: Como usuario, quiero que al borrar un elemento de la base de datos, los campos de las cajas de texto se limpien automáticamente y aparezca un mensaje de confirmación con la opción de sí/no, para evitar borrar registros por error. Cuando el usuario decide eliminar un registro, la interfaz debe mostrar un cuadro de diálogo que pregunte si está seguro de querer eliminar el registro.	
Observaciones: Es crucial para la integridad de los datos y la usabilidad de la interfaz que los usuarios confirmen las acciones destructivas como el borrado de registros. Esta confirmación previene eliminaciones accidentales, lo cual es fundamental para mantener la confianza del usuario en la aplicación.	

Habilitación condicional de botones

Historia de usuario	
Número: 3	Usuario: Administrador Base de Datos, Empleado
Nombre historia: Habilitación condicional de botones	
Prioridad en negocio: Media	Riesgo en desarrollo: Medio

Puntos estimados: 5	Iteración asignada: 2
Programador responsable: Jesús Donaldo Guerrero Palacios	
Descripción: Como usuario, quiero que los botones de eliminar y editar estén habilitados solo después de realizar una búsqueda, para asegurar que no se realicen operaciones sin un contexto adecuado. Esto significa que cuando la interfaz se carga inicialmente, estos botones deben estar deshabilitados.	
Observaciones: Esta funcionalidad evita errores de edición o eliminación sin contexto, mejorando la seguridad y la usabilidad de la interfaz. Garantiza que las acciones solo se puedan realizar cuando se ha seleccionado un registro específico, reduciendo el riesgo de operaciones incorrectas.	

Confirmación y Limpieza al Editar Elementos

Historia de usuario	
Número: 4	Usuario: Administrador Base de Datos, Empleado
Nombre historia: Confirmación y Limpieza al Editar Elementos	
Prioridad en negocio: Media	Riesgo en desarrollo: Medio
Puntos estimados: 5	Iteración asignada: 2
Programador responsable: Jesús Donaldo Guerrero Palacios	
Descripción: Como usuario, quiero que al editar un elemento, aparezca un mensaje de confirmación con la opción de sí/no y que los campos se limpien automáticamente después de la operación, para asegurar que las modificaciones se realicen intencionalmente y no haya datos residuales.	
Observaciones: Esta funcionalidad es importante para mantener la integridad de los datos y asegurar que las operaciones de edición se realicen correctamente.	

Deshabilitar Botones Después de Operaciones

Historia de usuario	
Número: 5	Usuario: Administrador Base de Datos, Empleado
Nombre historia: Deshabilitar Botones Después de Operaciones	
Prioridad en negocio: Media	Riesgo en desarrollo: Baja
Puntos estimados: 3	Iteración asignada: 3

Programador responsable: Jesús Donaldo Guerrero Palacios
Descripción: Como usuario, quiero que los botones de eliminar y editar se deshabiliten después de realizar una operación (eliminación o modificación), para prevenir múltiples acciones sin contexto. Esto significa que después de que una eliminación o edición se haya completado, los botones correspondientes deben deshabilitarse automáticamente hasta que se realice una nueva búsqueda.
Observaciones: Mejora la seguridad y la consistencia de la interfaz al asegurarse de que los usuarios no puedan realizar operaciones sucesivas sin un nuevo contexto. Esta práctica ayuda a evitar errores y asegura que cada acción del usuario esté respaldada por la información más reciente y precisa disponible

Prevención de Códigos Duplicados

Historia de usuario	
Número: 6	Usuario: Administrador Base de Datos, Empleado
Nombre historia: Prevención de Códigos Duplicados	
Prioridad en negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 5	Iteración asignada: 3
Programador responsable: Jesús Donaldo Guerrero Palacios	
Descripción: Como usuario, quiero recibir una advertencia si ingreso un código que ya existe en la base de datos, para evitar duplicados y asegurar la unicidad de los registros. Al intentar guardar un nuevo registro o actualizar uno existente, la interfaz debe verificar si el código ingresado ya existe en la base de datos.	
Observaciones: Esta funcionalidad es crucial para mantener la integridad y la unicidad de los datos en la base de datos. Prevenir códigos duplicados asegura que cada registro sea único y fácilmente identificable, lo que es esencial para la correcta gestión y recuperación de datos.	

Consulta de Datos en la Interfaz

Historia de usuario	
Número: 7	Usuario: Administrador Base de Datos, Empleado
Nombre historia: Consulta de Datos en la Interfaz	
Prioridad en negocio: Baja	Riesgo en desarrollo: Alto

Puntos estimados: 8	Iteración asignada: 4
Programador responsable: Jesús Donaldo Guerrero Palacios	
Descripción: Como usuario, opcionalmente deseo poder consultar todos los datos de la base de datos en una interfaz, para tener una visión completa de los registros y asegurar que no haya campos vacíos. Esta funcionalidad implicaría desarrollar una vista o módulo dentro de la interfaz donde se puedan visualizar todos los registros de la base de datos.	
Observaciones: Aunque esta funcionalidad es opcional, mejorará significativamente la experiencia del usuario al permitir una gestión más completa y transparente de los datos.	

Verificación de Campos No Vacíos

Historia de usuario	
Número: 8	Usuario: Administrador Base de Datos, Empleado
Nombre historia: Verificación de Campos No Vacíos	
Prioridad en negocio: Alta	Riesgo en desarrollo: Bajo
Puntos estimados: 3	Iteración asignada: 4
Programador responsable: Jesús Donaldo Guerrero Palacios	
Descripción: Como usuario, no deseo que ninguno de los campos se quede vacío, para asegurar que todos los registros estén completos y válidos. La interfaz debe implementar validaciones que obliguen a los usuarios a llenar todos los campos requeridos antes de permitirles guardar un nuevo registro o actualizar uno existente.	
Observaciones: Esta funcionalidad asegura la integridad y la completitud de los datos ingresados, evitando problemas futuros en la gestión de la base de datos.	