***Universidad de las Fuerzas Armadas***

*Un dibujo de salsa de tomate

Descripción generada automáticamente con confianza baja*

***Tecnologías de la Información y Comunicación***

***Ingeniería de Software II, NRC 16417***

***Actividad de Aprendizaje No. 3***

***Integrantes***

* *Almeida Garófalo Alan Joel*
* *Flores Ortiz, Jose Miguel*

***15 de enero del 2024***

**Plan de Desarrollo de Software para la Modernización del Sistema de Gestión de Biblioteca**

**Definición del Proyecto:**

Este proyecto tiene como objetivo modernizar el sistema de gestión de bibliotecas mediante la implementación de una arquitectura cliente-servidor. El sistema se actualizará para mejorar la eficiencia y la experiencia del usuario.

**Propósito del Documento de Requisitos:**

Definir los requisitos del sistema de aplicación de software desarrollado para la librería de venta electrónica de libros. Estos requisitos servirán como base para el diseño, desarrollo, implementación y verificación del sistema.

**Alcance del Proyecto:**

El alcance del proyecto incluye el desarrollo de una aplicación web de tres capas para la venta electrónica de libros, integrando servicios de pago externos.  
**Definiciones, Acrónimos y Abreviaturas:**

* **TI**: Tecnologías de la Información
* **Backend**: Capa de procesamiento y almacenamiento de datos en la aplicación de tres capas.
* **Cliente-Servidor**: Una arquitectura usada para el desarrollo de software.
* **JavaScript**: Lenguaje de programación interpretado, dialecto del estándar ECMAScrip.
* **CSS**: Lenguaje de diseño gráfico para definir y crear la presentación de un documento.
* **HTML**: Lenguaje de marcado utilizado en la creación de páginas web.

**Resumen del Resto del Documento:**

El sistema heredado se actualizará mediante la introducción de una arquitectura cliente-servidor. AngularJS se utilizará para la interfaz de usuario, mientras que Java se empleará para desarrollar servicios RESTful para el backend. La migración se realizará incrementalmente para minimizar el impacto en los usuarios.

**Descripción General**

**Perspectiva del Producto:**

Integración del sistema como un sistema de venta electrónica en el mercado de libros en línea.

**Funciones del Producto:**

* Registro y autenticación de clientes.
* Búsqueda y gestión de productos.
* Proceso de pedidos con opciones de pago.
* Gestión de descuentos por parte de agentes de ventas.
* Administración de clientes, productos e información por parte de los administradores.

**Características de los Usuarios:**

* **Clientes**: Utilizan la aplicación para buscar, comprar y gestionar libros.
* **Agentes de Ventas**: Conceden descuentos a clientes.
* **Administradores TI**: Administran clientes, productos e información.

Requisitos Específicos

**Requisitos Funcionales:**

**RF1**: Proceso de Registro de Clientes

**Descripción**:   
Los clientes deben poder registrarse en la aplicación proporcionando información específica.

**Verificación**:

Se asegurará la claridad, consistencia, completitud y rastreabilidad de los requisitos.

**RF2**: Búsqueda y Gestión de Productos

**Descripción**:

Los usuarios deben poder buscar y gestionar productos dentro del sistema.

**Verificación**:

* Se verificará que la función de búsqueda proporcione resultados relevantes y precisos.
* Se comprobará que la gestión de productos permita agregar, editar y eliminar productos de manera efectiva.

**RF3**: Proceso de Pedidos con Opciones de Pago

**Descripción**:

Los clientes deben poder realizar pedidos y seleccionar opciones de pago adecuadas.

**Verificación**:

* Se asegurará que el proceso de pedido sea intuitivo y fácil de seguir.
* Se comprobará que las opciones de pago estén integradas correctamente y funcionen según lo esperado.

**RF4**: Gestión de Descuentos por Parte de Agentes de Ventas

**Descripción**:

Los agentes de ventas deben poder aplicar descuentos a los productos para los clientes.

**Verificación**:

Se verificará que los agentes de ventas puedan aplicar descuentos de manera efectiva y que se reflejen correctamente en los precios.

**RF5**: Administración de Clientes, Productos e Información por Parte de los Administradores

**Descripción**:

Los administradores deben tener acceso a herramientas para administrar clientes, productos e información del sistema.

**Verificación**:

* Se asegurará que los administradores puedan agregar, editar y eliminar clientes y productos según sea necesario.
* Se comprobará que los administradores puedan acceder a informes y estadísticas relevantes sobre el sistema.

**Requisitos No Funcionales:**

**RNF1**: Seguridad de la Aplicación

**Descripción**:

* La aplicación debe cumplir con estándares de seguridad específicos.

**Verificación**:

* Se verificará la claridad, consistencia, completitud y rastreabilidad de los estándares de seguridad.

**RNF2**: Rendimiento del Sistema

**Descripción**:

* El sistema debe ser capaz de manejar una carga de trabajo esperada sin degradación significativa del rendimiento.

**Verificación**:

* Se realizarán pruebas de carga para evaluar el rendimiento del sistema bajo diferentes condiciones.
* Se establecerán métricas de rendimiento y se verificará que se cumplan durante las pruebas.

**RNF3**: Escalabilidad

**Descripción**:

* El sistema debe ser capaz de escalar para manejar un aumento en el número de usuarios y transacciones.

**Verificación**:

* Se evaluará la capacidad del sistema para escalar vertical y horizontalmente según sea necesario.
* Se comprobará que el sistema pueda manejar el aumento de carga de manera eficiente y sin comprometer la funcionalidad.

**RNF4**: Usabilidad

**Descripción**:

* La interfaz de usuario debe ser intuitiva y fácil de usar para los diferentes tipos de usuarios.

**Verificación**:

* Se realizarán pruebas de usabilidad con usuarios reales para evaluar la facilidad de uso de la interfaz.
* Se recopilarán comentarios y sugerencias de los usuarios para mejorar la usabilidad del sistema.

**RNF5**: Mantenibilidad

**Descripción**:

* El sistema debe ser fácil de mantener y actualizar en el futuro.

**Verificación**:

* Se utilizarán prácticas de desarrollo limpio y modular para facilitar la mantenibilidad del código.
* Se proporcionará documentación detallada para los desarrolladores y administradores del sistema.

**Diseño Arquitectónico**

El diseño arquitectónico de la modernización del sistema de gestión de bibliotecas se basará en una arquitectura de tres capas: presentación, lógica de negocio y datos. Esta arquitectura distribuida permitirá una separación clara de responsabilidades y una escalabilidad adecuada del sistema.

**Capa de Presentación:**

En esta capa se encuentra la interfaz de usuario, que será desarrollada utilizando AngularJS, HTML y CSS para proporcionar una experiencia moderna e intuitiva para los usuarios. La interfaz de usuario permitirá a los clientes buscar libros, realizar pedidos y gestionar su cuenta, mientras que los agentes de ventas y los administradores podrán acceder a funciones adicionales para la gestión de productos y clientes.

**Capa de Lógica de Negocio:**

En esta capa se encuentra la lógica de negocio de la aplicación, que será desarrollada utilizando Java para la creación de servicios RESTful. Estos servicios serán responsables de procesar las solicitudes de los usuarios, realizar operaciones de negocio como el registro de clientes, la gestión de pedidos y la aplicación de descuentos, y coordinar la interacción entre la capa de presentación y la capa de datos.

**Capa de Datos:**

En esta capa se encuentra la base de datos, que almacenará toda la información relevante para el sistema de gestión de bibliotecas. Se utilizará una base de datos relacional para garantizar la integridad y la consistencia de los datos. La base de datos será accedida por la capa de lógica de negocio a través de consultas SQL para realizar operaciones de lectura y escritura de datos.

**Componentes Adicionales:**

Además de las tres capas principales, se integrarán componentes adicionales para mejorar la funcionalidad y la seguridad del sistema. Estos incluyen:

* Servicios de autenticación y autorización para garantizar que solo los usuarios autorizados puedan acceder a determinadas funciones.
* Integración con servicios de pago externos para procesar transacciones financieras de manera segura.
* Mecanismos de caché para mejorar el rendimiento y la escalabilidad del sistema, almacenando en caché datos frecuentemente accedidos para reducir la carga en la base de datos.

**Diagrama de Arquitectura:**

Diagrama de casos de uso

Diagrama

Descripción generada automáticamente

[Diagrama de casos de uso](https://lucid.app/lucidchart/c19dc266-3c7d-43f7-a565-a2cc85c4d939/edit?viewport_loc=-1299%2C-1363%2C4020%2C1990%2C0_0&invitationId=inv_d2c42dda-64de-4e7b-9e89-25249f867ab4)

Diagrama de flujos

Diagrama, Esquemático

Descripción generada automáticamente

[Diagrama de flujo](https://lucid.app/lucidchart/578162f0-e4fd-4213-a4fc-3d7dd30e957c/edit?viewport_loc=-484%2C417%2C2979%2C1475%2C0_0&invitationId=inv_a4596ab6-aafa-45d8-bb44-86ee97d02b28)

**Gestión de la Configuración**

La gestión de la configuración es esencial para garantizar la integridad y coherencia del sistema de gestión de bibliotecas modernizado. A continuación, se detalla cómo se abordarán los aspectos clave de la gestión de la configuración en este proyecto:

**Control de Versiones:**

* Se implementará y mantendrá un sistema de control de versiones utilizando Git y GitHub.
* Cada iteración del software será etiquetada y documentada para facilitar la identificación de hitos y la reversión a estados anteriores si es necesario.

**Gestión de Cambios:**

* Se establecerá un proceso formal para solicitar, evaluar y aprobar cambios en los artefactos de software.
* Se mantendrá un registro detallado de las modificaciones realizadas, incluyendo la justificación y el impacto previsto en el sistema.

**Configuración de Ambientes:**

* Se definirán y mantendrán configuraciones específicas para entornos de desarrollo, prueba y producción.
* Se gestionarán dependencias y configuraciones de terceros para garantizar la consistencia entre entornos.

**Trabajo Colaborativo:**

* Se facilitará la colaboración entre los diferentes equipos involucrados en el desarrollo, como los encargados de la interfaz de usuario, lógica de negocio y administración de bases de datos.
* Se implementará la integración continua para garantizar la compatibilidad y funcionalidad conjunta de las distintas partes del sistema.

**Auditoría y Seguimiento:**

* Se establecerán mecanismos de auditoría para registrar cambios y acceder al historial de configuración.
* Se implementará un proceso de seguimiento para evaluar el rendimiento de la gestión de configuración y realizar ajustes según sea necesario.

**Roles y Responsabilidades:**

|  |  |
| --- | --- |
| **Roles** | **Responsabilidades** |
| **Gerente de Configuración:** | - Supervisará la planificación, identificación, control y auditoría de todos los elementos de configuración.  - Desarrollará y ejecutará el plan de gestión de configuración y liderará evaluaciones del proceso. |
| **Coordinador de Configuración:** | - Registrará adecuadamente todos los elementos de configuración, garantizando su integridad, y reportará discrepancias al Gerente de Configuración.  - Reportar discrepancias o no conformidades al Gerente de Configuración. |
| **Responsable de Elementos de Configuración:** | - Registrar en la base de datos los elementos de configuración bajo su responsabilidad con datos apropiados.  - Verificar que los cambios en los elementos sigan el proceso definido.  - Colaborar con el Gerente de Configuración para abordar discrepancias identificadas en auditorías e implementar acciones correctivas. |
| **Gestor de Cambio:** | - Evaluará el impacto y riesgo de los cambios propuestos, y asegurará la actualización de los históricos de los elementos afectados. |

**Actividades de Gestión de Configuración:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Actividad** | **Rol Responsable** | **Descripción** | **Entradas** | **Salidas** |
| Planificación de la Gestión de Configuración. | Gerente de Configuración | Elaborar el plan de gestión de configuración para la aplicación de venta electrónica de libros. | Requisitos del proyecto. Planificación de proyecto. | Plan de gestión de configuración aprobado. |
| Identificación de Elementos de Configuración. | Gerente de Configuración | Identificar y registrar los elementos de configuración relevantes para la aplicación. Establecer la estructura del directorio de gestión de configuración. | Documentación de requisitos. Productos del proyecto. | Elementos de configuración identificados. Línea base. Estructura del directorio de gestión de configuración. |
| Control de Cambios y Líneas Base. | Responsable del Elemento de Configuración | Evaluar y controlar los cambios en los elementos de configuración. Obtener la aprobación para solicitudes de cambio en productos de trabajo de línea base. | Solicitudes de cambio. | Registro de solicitud de cambio. Solicitud de cambio aprobada. Línea base actualizada. |
| Informe de Estado de Configuración. | Gerente de Configuración | Mantener y publicar de forma regular el estado actualizado de los elementos de configuración. | Elementos de configuración. | Informe de estado de elementos de configuración. |
| Verificación y Auditoría. | Gerente de Configuración | Realizar auditorías para evaluar la eficacia de la gestión de configuración. | Registros de gestión de configuración. Línea base. Registros de cambios. | Informe de auditoría de gestión de configuración. |

**Identificación de Elementos de Configuración:**

Para desarrollar de manera más completa la identificación de elementos de configuración, podemos profundizar en algunos aspectos clave y proporcionar ejemplos adicionales:

**Asignación de Identificadores Únicos:**

* Establecer un formato estándar para los identificadores únicos, como prefijos que indiquen el tipo de elemento seguido de un número secuencial.
* Ejemplo: PRJ-001 para el primer plan de proyecto, REQ-001 para el primer requisito, etc.

**Definición de Estructura de Almacenamiento:**

* Organizar los elementos de configuración en una estructura jerárquica que refleje la arquitectura del proyecto y facilite la navegación y búsqueda.
* Ejemplo: Carpeta raíz del proyecto > Documentos > Planes > Plan de proyecto.

**Control de Acceso:**

* Establecer grupos de usuarios y asignar permisos específicos según roles y responsabilidades.
* Ejemplo: Los desarrolladores tienen permisos de escritura en el repositorio de código, mientras que los miembros del equipo de pruebas solo tienen permisos de lectura.

**Criterios para Inclusión bajo Gestión de Configuración:**

* Definir claramente cuándo un elemento debe ser incluido en el sistema de gestión de configuración, por ejemplo, al alcanzar un estado de revisión o aprobación.
* Ejemplo: Los documentos se incluyen en la gestión de configuración después de la revisión formal por parte del equipo de calidad.

**Autorización y Línea Base:**

* Establecer un proceso para obtener la autorización para incluir un elemento en la línea base, que marque un hito en su evolución y congele su estado.
* Ejemplo: La línea base se establece para el código fuente después de completar una versión estable del software.

**Procedimientos de Cambios:**

* Definir un flujo de trabajo para solicitar, evaluar, aprobar e implementar cambios en los elementos de configuración.
* Ejemplo: Los desarrolladores presentan solicitudes de cambio que son revisadas por el equipo de gestión de configuración antes de ser implementadas.

**Criterios para la Selección de Elementos de Configuración:**

* Identificar criterios específicos para determinar qué elementos deben ser gestionados, como su impacto en la integridad y la calidad del sistema.
* Ejemplo: Se incluyen en la gestión de configuración aquellos artefactos que son críticos para la entrega del producto final.

**Métricas de Software, Proceso y Diseño:**

* Establecer métricas específicas para evaluar la calidad del software, la eficiencia del proceso y la robustez del diseño.
* Ejemplo: Se realizan mediciones periódicas de la cobertura de código y la frecuencia de despliegues para mejorar continuamente el proceso de desarrollo.

**Proceso de Control de Cambios:**

El proceso de control de cambios para el proyecto de desarrollo de la aplicación podría seguir una estructura similar a la proporcionada, adaptada a las necesidades y características específicas del proyecto. A continuación, se detalla cómo podría ser este proceso:

**Recepción de Solicitud de Cambio:**

* Cuando surge la necesidad de un cambio en la aplicación, ya sea debido a un error identificado, una mejora propuesta o un nuevo requisito, el proceso se inicia enviando una solicitud de cambio a través de un formulario designado o herramienta de seguimiento de cambios.

**Registro y Documentación:**

* Las solicitudes de cambio se registran en el sistema de gestión de configuración, donde se documentan detalles como la descripción del cambio, el motivo, la prioridad, la fecha de recepción y cualquier otra información relevante.

**Evaluación Inicial:**

* El Gerente de Configuración y el equipo de gestión de configuración revisan inicialmente la solicitud para determinar su viabilidad y evaluar el impacto potencial en los productos y el proyecto en general.

**Análisis de Impacto y Evaluación Técnica:**

* Se lleva a cabo un análisis detallado del impacto del cambio propuesto, considerando factores como la complejidad técnica, los recursos requeridos y la alineación con los objetivos del proyecto.

**Revisión Formal:**

* Se convoca a una reunión formal para considerar la solicitud de cambio, en la que se presentan los resultados del análisis de impacto. Durante esta reunión, se evalúa si el cambio es viable y beneficioso para el proyecto.

**Aprobación o Rechazo:**

* Basándose en la revisión formal, se toma una decisión sobre la aprobación o el rechazo del cambio. Si se aprueba, se procede a la siguiente etapa.

**Implementación del Cambio:**

* Con la aprobación, se procede a implementar el cambio en los productos afectados, siguiendo los procedimientos definidos en el plan de gestión de configuración y utilizando herramientas como GitHub para gestionar el código fuente.

**Comunicación de Cambios:**

* Se notifica a todas las partes interesadas y personas afectadas sobre la implementación exitosa del cambio, proporcionando detalles sobre las modificaciones realizadas y cómo pueden verse afectados.

**Actualización de la Línea Base:**

* Después de la implementación exitosa, se actualiza la línea base del proyecto para reflejar la nueva configuración de los productos, asegurando la coherencia en el seguimiento de versiones.

**Registro y Seguimiento:**

* Cada cambio desde la solicitud hasta la implementación se registra detalladamente en el sistema de gestión de configuración, lo que permite un seguimiento continuo para evaluar el impacto a lo largo del tiempo y garantizar la trazabilidad.

**Informe de Estado de Configuración:**

* La información sobre los cambios realizados se incluye en informes de estado de configuración, que se mantienen y publican periódicamente para mantener informadas a todas las partes interesadas sobre el progreso del proyecto y los cambios realizados.

**Informe del Estado de los Elementos de Configuración:**

El informe del estado de los elementos de configuración para el proyecto de desarrollo de la aplicación se estructuraría de la siguiente manera, tomando como referencia las pautas proporcionadas:

**Identificación y Recopilación de Información:**

Se identificarán y recopilarán los siguientes tipos de información para respaldar el proceso de gestión de la configuración del proyecto:

* Registro de documentación de configuración aprobada.
* Designación de responsables de elementos de configuración.
* Estado de cambios propuestos y desviaciones de configuración.
* Implementación del estado de los cambios aprobados.
* Configuración de todas las unidades de los elementos de configuración en el inventario.
* Resultados de auditorías de configuración.

**Herramienta Automatizada de Soporte:**

Se implementará GitHub como herramienta de gestión de configuración automatizada para respaldar la recopilación, el seguimiento y la presentación de informes. GitHub proporcionará características como:

* Registro y aprobación de cambios.
* Asignación de responsables de configuración.
* Seguimiento del estado de implementación de cambios.
* Mantenimiento de inventario de configuración.
* Auditorías automatizadas para garantizar la coherencia.

**Métricas e Información a Incluir en el Informe:**

El informe de estado de configuración incluirá, entre otras, las siguientes métricas e información:

* Número total de cambios propuestos.
* Estado actual de implementación de cambios aprobados.
* Designación y carga de trabajo de responsables de configuración.
* Inventario actualizado de todas las unidades de configuración.
* Resultados de auditorías, destacando cualquier no conformidad.

**Periodicidad y Distribución del Informe:**

* Los informes se generarán periódicamente según un cronograma predeterminado (por ejemplo, semanal o mensual) para garantizar que la información se actualice continuamente. El informe se compartirá con todas las partes interesadas, incluido el equipo de desarrollo, los clientes y los responsables de la configuración.

**Evaluación Continua y Mejora:**

* Se establecerá un proceso de evaluación continua para verificar la eficacia de los informes y herramientas automatizadas. Se realizarán los ajustes necesarios para mejorar la calidad y la idoneidad de los informes para la gestión de la configuración.

**Definición de Plan de Gestión de la Configuración:**

**Objetivo del Plan:**

El objetivo principal del Plan de Gestión de la Configuración (PGC) es proporcionar la estructura organizacional y los procesos necesarios para definir, controlar y probar los componentes de configuración del proyecto de desarrollo de la aplicación de libros electrónicos. Se busca asegurar la consistencia y calidad del software producido, manteniendo la integridad y coherencia del producto o sistema.

**Alcance:**

El alcance del PGC abarca todos los elementos de configuración relevantes para el proyecto de desarrollo de la aplicación de libros electrónicos. Esto incluye, pero no se limita a, código fuente, documentación técnica, artefactos de diseño, recursos multimedia y cualquier otro producto relacionado con el desarrollo de software. Se establecerán claramente los límites del control de configuración y se especificarán las actividades que serán gestionadas por el plan.

**Identificación de Elementos de Configuración:**

Se establecerá un esquema para identificar de manera única cada elemento de configuración. Esto incluirá convenciones de nomenclatura para versiones, lanzamientos y otras entidades no estándar. Se asignarán identificadores únicos a cada elemento de configuración para facilitar su seguimiento y control a lo largo del ciclo de vida del proyecto.

**Proceso de Control de Cambios:**

Se describirán los procedimientos para realizar comprobaciones de configuración y gestionar los cambios de manera efectiva. Esto implicará revisar sistemáticamente los elementos de configuración para garantizar que cumplan con las regulaciones y estándares especificados. Se establecerán mecanismos para evaluar el impacto de los cambios y se realizarán pruebas de integración y validación según sea necesario para mantener la integridad del sistema.

**Roles y Responsabilidades:**

Se definirán claramente los roles y responsabilidades de los miembros del equipo relacionados con la gestión de la configuración. Esto incluirá roles como administrador de configuración, coordinador de configuración y propietario de elementos de configuración, junto con sus responsabilidades asociadas. Se establecerá una estructura organizativa clara para la gestión eficaz de la configuración.

**Herramientas:**

Se proporcionará una lista de herramientas y recursos necesarios para una gestión de configuración eficaz. Esto puede incluir herramientas de control de versiones como Git, sistemas de seguimiento de problemas como Jira, herramientas de integración continua como Jenkins, entre otras herramientas relevantes para el proyecto. Se garantizará que estas herramientas estén disponibles y configuradas adecuadamente para respaldar los procesos de gestión de configuración.

**Métricas y Evaluación Continua:**

Se establecerán métricas para evaluar la efectividad del Plan de Gestión de la Configuración y el proceso de gestión de configuración en general. Se recomendará un proceso de evaluación continua para realizar ajustes y mejoras según sea necesario. Se realizarán revisiones periódicas del plan y se actualizará según sea necesario para garantizar su relevancia y eficacia durante todo el ciclo de vida del proyecto. **Especificación del Entorno Tecnológico:**

En esta tarea, se delineará el entorno tecnológico específico que respaldará la gestión de configuración del sistema de información de la aplicación de libros electrónicos. Se considerará la automatización de procesos y controles definidos en el plan de gestión de la configuración, garantizando la eficacia y eficiencia en la mecanización de los procesos y proporcionando la base tecnológica necesaria para el control de versiones, gestión de cambios y seguimiento de la configuración.

**Revisión del Plan de Gestión de la Configuración:**

Se realizará una revisión detallada del Plan de Gestión de la Configuración (PGC) para comprender los requisitos específicos y los controles establecidos que guiarán la selección y configuración del entorno tecnológico.

**Identificación de Componentes Tecnológicos Necesarios:**

En colaboración con el equipo de desarrollo, se identificarán los componentes hardware y software necesarios para respaldar los procesos de control de versiones, gestión de cambios y auditoría de la configuración. Esto puede incluir servidores, sistemas de almacenamiento, redes, así como software específico para la gestión de configuración.

**Selección de Herramientas de Control de Versiones:**

Se evaluarán y seleccionarán herramientas de control de versiones que se alineen con los requisitos del proyecto. Se considerarán sistemas como Git o SVN, teniendo en cuenta la preferencia del equipo y la compatibilidad con el entorno de desarrollo.

**Herramientas de Gestión de Proyectos y Comunicación:**

Se seleccionarán herramientas de gestión de proyectos y comunicación, como Jira o Trello, para facilitar la colaboración y la asignación de tareas relacionadas con la gestión de configuración.

**Automatización de Integración Continua:**

Se evaluarán y configurarán herramientas de integración continua, como Jenkins o GitLab CI, para automatizar la verificación del código y la detección temprana de problemas de integración.

**Herramientas de Auditoría y Métricas:**

Se seleccionarán herramientas de auditoría automatizadas, como SonarQube, y herramientas de presentación de métricas, como Grafana o Kibana, para evaluar la calidad del código y realizar un seguimiento del rendimiento de la gestión de configuración.

**Entornos de Desarrollo, Prueba y Producción:**

Se establecerán entornos utilizando tecnologías como Docker para asegurar ambientes consistentes y reproducibles para desarrollo, prueba y producción. Esto garantizará que el software se pueda desarrollar y probar de manera efectiva en diferentes etapas del ciclo de vida del proyecto.

**Seguridad y Control de Acceso:**

Se implementarán prácticas de seguridad, incluida la autenticación de dos factores, y se establecerán niveles de acceso controlado en las herramientas utilizadas para garantizar la seguridad de los artefactos y la información de configuración.

**Documentación del Entorno Tecnológico:**

Se documentará detalladamente el entorno tecnológico especificado, proporcionando información clara sobre las herramientas seleccionadas, configuraciones y flujos de trabajo. Esto garantizará que el equipo tenga acceso a recursos y orientación necesarios para trabajar de manera efectiva en el entorno tecnológico definido.

**Validación del Entorno Tecnológico:**

Se validará el entorno tecnológico con el equipo de desarrollo para asegurar su compatibilidad y eficacia en la gestión de configuración. Se realizarán pruebas y ajustes según sea necesario para garantizar que el entorno tecnológico cumpla con los requisitos del proyecto.

**Participantes:**

* Responsable de Gestión de Configuración
* Equipo de Desarrollo
* Aspecto Herramienta/Recurso

|  |  |
| --- | --- |
| **Aspecto** | **Herramienta/Recurso** |
| **Control de Versiones** | Git |
| **Plataforma de Colaboración** | GitHub |
| **Gestión de Ramas** | Gitflow |
| **Herramienta de Seguimiento de Problemas** | Jira, Redmine |
| **Automatización de Auditorías** | SonarQube, Jenkins |
| **Base de Datos de Configuración** | MySQL, PostgreSQL |
| **Herramienta de Entregas Continuas** | Jenkins, Travis CI |
| **Documentación** | Confluence |
| **Comunicación y Notificación** | Slack, Email |
| **Herramienta de Pruebas** | Selenium, JUnit, Jest |
| **Ambiente de Desarrollo Integrado (IDE)** | IntelliJ IDEA, Visual Studio Code |

**Prototipo**

Para el prototipado crearemos un proyecto en Apache Netbeans, el cual lo llamaremos prototipo:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

También crearemos una base de datos la cual usaremos para poder almacenar la información de la Biblioteca, para lo cual usaremos xampp, como veremos a continuación:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Lo siguiente seria crear la información de la base de datos, para lo cual crearemos las tablas y las celdas de la información que deseamos agregar:



En este caso crearemos una tabla con los Libros que vamos a almacenar como es solo un prototipo, es un buen comienzo, para lo cual tendrán las siguientes columnas:

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

En el cual ya tenemos los datos que vamos a llenar, lo siguiente es crear la conexión con la base datos a nuestro proyecto, para lo cual debemos importar una librería:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Una vez incluida la librería procederemos a crear un MVC el cual nos ayudara a comunicar la base de datos con el proyecto:

Texto

Descripción generada automáticamente

Con esto ya comunicamos al proyecto con la base de datos, lo siguiente seria crear los datos que usaremos para la página y los paquetes:

Interfaz de usuario gráfica, Aplicación, Word

Descripción generada automáticamente

A continuación, la página con la tabla para los libros,

Tabla

Descripción generada automáticamente con confianza baja

Y en esta es la de agregar:

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

**Practica:**

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Con esto se agregaría en la base de datos, como se verá a continuación:

Tabla

Descripción generada automáticamente