Diagrama

Descripción generada automáticamente

## Introducción

### Propósito

El desarrollo de una página web similar a Diagrams.net se entiende como una oportunidad para crear una herramienta que permita a los usuarios diseñar y gestionar diagramas de manera intuitiva, eficiente y accesible desde cualquier lugar. Este sistema busca facilitar la organización de ideas, procesos y estructuras a través de una interfaz interactiva y personalizable que fomente la creatividad y el análisis visual.

Además, el proyecto está diseñado con un enfoque en la colaboración en tiempo real, habilitando a múltiples usuarios para trabajar simultáneamente en un mismo diagrama, con actualizaciones instantáneas gracias a la integración de tecnologías como WebSockets. Esto asegura que los cambios realizados por un usuario sean reflejados de forma inmediata en los demás clientes conectados, mejorando la comunicación y la productividad en entornos académicos, laborales y personales.

Desde el punto de vista técnico, el desarrollo incorpora patrones y arquitecturas modernas que promueven la escalabilidad y la modularidad, como el uso de callbacks para gestionar eventos y flujos asincrónicos. Estos mecanismos fortalecen la capacidad de manejar interacciones complejas entre los distintos componentes del sistema, optimizando tanto el rendimiento como la experiencia del usuario.

Este proyecto no solo busca proporcionar una solución práctica y funcional para la creación de diagramas, sino también servir como una plataforma de aprendizaje para el equipo de desarrollo, permitiendo profundizar en el uso de tecnologías emergentes y consolidar conocimientos en áreas clave del desarrollo web, como la gestión de datos en tiempo real, la interacción cliente-servidor y la implementación de sistemas dinámicos y colaborativos.

## Análisis de requerimientos

### Alcance del producto

La página web estará enfocada en ofrecer funcionalidades básicas pero esenciales para la creación de diagramas colaborativos en tiempo real. Los usuarios podrán colocar y manipular íconos en un lienzo digital, diseñando diagramas de manera sencilla e intuitiva. El sistema permitirá que múltiples usuarios conectados puedan observar en tiempo real las modificaciones realizadas por otros, asegurando una experiencia de colaboración fluida y sincronizada.

Además, los diagramas creados podrán ser exportados en formatos estándar, lo que garantiza su portabilidad y uso fuera de la plataforma.

### Clases de usuario y características

|  |  |
| --- | --- |
| **Clase de usuario** | **Descripción** |
| Usuario | En esta plataforma, el usuario es completamente anónimo y no requiere registro ni autenticación para acceder al sistema. Al ingresar, tiene acceso inmediato al lienzo de trabajo, donde puede interactuar de forma intuitiva con las herramientas disponibles. Entre sus capacidades se encuentra la posibilidad de colocar y manipular íconos, mover, redimensionar y eliminar elementos del diagrama según sus necesidades. |

### Requisitos funcionales

1. El sistema debe permitir al usuario colocar íconos en el lienzo de manera libre.
2. El sistema debe permitir al usuario mover, redimensionar, girar y eliminar los íconos en el lienzo.
3. El sistema debe ofrecer herramientas básicas, como líneas, flechas y elementos que conforman al diagrama de flujo y el diagrama de casos de uso.
4. El sistema debe sincronizar automáticamente los cambios realizados por cualquier usuario en el lienzo para que sean visibles en tiempo real por otros usuarios conectados.
5. El sistema debe permitir que múltiples usuarios trabajen simultáneamente en el mismo espacio sin conflictos.
6. El sistema debe actualizar la posición, tamaño y estado de los íconos en tiempo real en todos los clientes conectados.
7. El sistema debe permitir al usuario exportar el diagrama en formatos estándar, como PNG.
8. El sistema debe ser accesible sin necesidad de registro o autenticación.
9. El sistema debe cargar el lienzo automáticamente al acceder a la página web.

### Requisitos no funcionales

1. La sincronización de los cambios en el lienzo debe realizarse con una latencia máxima de 200 ms para garantizar fluidez en la interacción.
2. La exportación de un diagrama debe completarse en menos de 3 segundos.
3. La interfaz de usuario debe ser intuitiva, con herramientas claramente identificables y accesibles para cualquier usuario, incluso sin conocimientos previos.
4. El sistema debe mantener la conexión en tiempo real mientras el usuario esté activo, con mecanismos de reconexión automática en caso de pérdida de conexión.
5. Los mensajes de sincronización en tiempo real deben garantizar la entrega y procesarse en el orden correcto.

## Casos de uso

### Lista de casos de uso

CU-01 Crear diagrama

CU-02 Exportar diagrama

CU-03 Consultar atajos del teclado

### Matriz de trazabilidad de requisitos

|  |  |  |  |
| --- | --- | --- | --- |
|  | CU1 | CU2 | CU3 |
| 1 | X |  |  |
| 2 | X |  | X |
| 3 | X |  |  |
| 4 | X |  |  |
| 5 | X |  |  |
| 6 | X |  |  |
| 7 |  | X |  |
| 8 | X |  |  |
| 9 | X |  |  |

### Descripción de los casos de uso

|  |  |
| --- | --- |
| ID | CU01 |
| Nombre del CU | Crear diagrama |
| Actor(es) | USUARIO |
| Descripción | El USUARIO realiza la creación de un diagrama básico de casos de uso utilizando los elementos proporcionados en la página. |
| Flujo normal | 1. El USUARIO ingresa a la página de diagramas. 2. El SISTEMA crea un ID aleatorio y se lo asigna al USUARIO. 3. El USUARIO da clic en el ícono de un actor y lo arrastra al área del lienzo. 4. El USUARIO da clic en un rectángulo y lo arrastra al lado del ícono del actor. 5. El USUARIO da clic en una flecha y la arrastra en medio del actor y del rectángulo. 6. El USUARIO arrastra un óvalo del caso de uso y lo coloca dentro del rectángulo. 7. El USUARIO visualiza el diagrama terminado. (ver FA01, FA02, FA03, FA04) 8. Fin del caso de uso. |
| Flujo alterno | FA01. Modificar ícono   1. El USUARIO da clic en el ícono. 2. El SISTEMA muestra la barra lateral que permite añadir texto al ícono, modificar su posicionamiento y alinear el texto escrito. 3. Continúa en el paso 7 del flujo normal.   FA02. Eliminar ícono   1. El USUARIO da clic un ícono 2. El USUARIO da clic en la tecla de retroceso 3. El SISTEMA elimina el ícono del lienzo del diagrama. 4. Continúa en el paso 7 del flujo normal   FA03. Redimensionar ícono   1. El USUARIO da clic un ícono 2. El USUARIO arrastra una esquina del ícono para hacerlo más grande o pequeño 3. El SISTEMA agranda o achica el ícono dependiendo de la acción del usuario 4. Continúa en el paso 7 del flujo normal   FA04. Exportar diagrama   1. Continúa en el CU02. 2. Continúa en el paso 7 del flujo normal.   FA05. Consultar atajo   1. Continúa en el CU03. 2. Continúa en el paso 7 del flujo normal. |
| Flujo de excepción | EX01. No hay conectividad al servidor   1. El SISTEMA muestra una alerta indicando que se perdió la conexión con el servidor. 2. Termina el caso de uso. |

|  |  |
| --- | --- |
| ID | CU02 |
| Nombre del CU | Exportar diagrama |
| Actor(es) | USUARIO |
| Descripción | El USUARIO realiza la exportación en formato .jpg del diagrama que realizó. |
| Flujo normal | 1. El USUARIO da clic en el botón de exportar 2. El SISTEMA genera la imagen en formato .jpg y la descarga automáticamente. (ver FA01). 3. Fin del caso de uso. |
| Flujo alterno | FA01. Diagrama sin elementos   1. El SISTEMA (muestra un mensaje indicando que no se puede exportar un diagrama vacío). |
| Flujo de excepción | EX01. No hay conectividad al servidor   1. El SISTEMA muestra una alerta indicando que se perdió la conexión con el servidor. 2. Termina el caso de uso. |

|  |  |
| --- | --- |
| ID | CU03 |
| Nombre del CU | Consultar atajos del teclado |
| Actor(es) | USUARIO |
| Descripción | El USUARIO realiza la consulta de atajos del teclado para realizar diversas acciones en el diagrama. |
| Flujo normal | 1. El USUARIO da clic en el botón de ayuda para ver los comandos. 2. El SISTEMA despliega una ventana flotante con los comandos del teclado disponibles en el sistema. 3. El USUARIO da clic en aceptar. 4. Termina el caso de uso |
| Flujo alterno | N/A |
| Flujo de excepción | EX01. No hay conectividad al servidor   1. El SISTEMA muestra una alerta indicando que se perdió la conexión con el servidor. 2. Termina el caso de uso. |

## Diagrama de clases

Diagrama

Descripción generada automáticamente

## Diagrama de componentes

Diagrama

Descripción generada automáticamente

## Explicación de la solución

El proyecto desarrollado consiste en una plataforma de diagramación colaborativa basada en la web, similar a diagrams.net pero más sencilla en cuanto a funcionalidad. Su principal objetivo es permitir que los usuarios coloquen íconos en una interfaz, les añadan texto y vean los cambios realizados en tiempo real por otros usuarios conectados. Esto se logra gracias a la implementación de Websockets para la comunicación bidireccional entre cliente y servidor.

El sistema se divide en dos componentes principales: Servidor y Cliente.

1. Servidor

El servidor se ha desarrollado utilizando Spring Boot, con el objetivo de gestionar las conexiones y mensajes en tiempo real mediante Websockets. A continuación, se detallan sus aspectos más relevantes:

a. Websockets

Se emplean Websockets para habilitar una comunicación bidireccional eficiente entre el cliente y el servidor. Los Websockets permiten enviar y recibir mensajes en tiempo real, lo que es esencial para una experiencia colaborativa fluida.

b. Manejadores de Mensajes

El servidor utiliza manejadores personalizados para gestionar los mensajes enviados desde los clientes. Estos manejadores:

* Reciben y procesan los mensajes.
* Manejan la distribución de actualizaciones a todos los clientes conectados.
* Administran las sesiones activas para asegurar que todos los usuarios reciban los cambios en tiempo real.

c. Contenedor con Docker

El proyecto incluye un archivo Dockerfile que permite empaquetar y desplegar el servidor en un contenedor. Esto asegura:

* Facilidad en la distribución y despliegue del servidor en diferentes entornos.
* Escalabilidad y portabilidad gracias a Docker.

2. Cliente

El cliente es una aplicación web ligera que interactúa con el servidor mediante Websockets. A continuación, se destacan los aspectos clave:

a. Estructura del Cliente

La estructura del cliente es minimalista, enfocada en la simplicidad y eficiencia. Incluye:

* Carpeta de íconos: Contiene las imágenes utilizadas para representar los íconos en los diagramas.
* Archivo index.html: Es el punto de entrada principal del sistema, donde se define la interfaz gráfica.
* Archivo script.js: Contiene el código JavaScript responsable de:
  + Manejar la interacción del usuario con la interfaz.
  + Administrar la comunicación con el servidor mediante Websockets.

b. Estilos con SASS

Para el diseño de la interfaz gráfica, se utilizó SASS (Syntactically Awesome Stylesheets), que permite una escritura más estructurada y reutilizable de los estilos CSS. Esto facilita:

* La organización de los estilos en módulos.
* La personalización rápida de la apariencia visual del sistema.

c. Conexión al Backend

La comunicación con el backend se realiza mediante Websockets. Cada vez que un usuario coloca un ícono o añade texto, esta información se envía al servidor, que la distribuye a los demás clientes conectados para mantener la sincronización en tiempo real.

d. Contenedor con Podman

El cliente también cuenta con un archivo para empaquetar su aplicación en un contenedor utilizando Podman, lo que permite:

* Desplegar la interfaz gráfica en cualquier entorno compatible con contenedores.
* Integrar el cliente y servidor en una misma infraestructura contenedorizada para simplificar el despliegue.

e. Modelo Simplicidad

Debido a que el cliente se enfoca exclusivamente en la interfaz gráfica, no se implementó un modelo basado en patrones como MVC (Modelo-Vista-Controlador). Esta decisión está alineada con la naturaleza del proyecto y su objetivo de ser una herramienta sencilla y funcional.