

**Desarrollo de Aplicación web canvas interactivo para crear modelos de negocio en  
CEIDEA de Uniagustiniana**

Daniel Stivens Ramírez Chitiva  
Joan Sebastian Saavedra Medina

Universitaria Agustiniana  
Facultad de Ingeniería  
Programa de Tecnología en Desarrollo De Software  
Bogotá, D.C.  
2022

**Desarrollo de Aplicación web canvas interactivo para crear modelos de negocio en  
CEIDEA de Uniagustiniana**

Daniel Stivens Ramírez Chitiva  
Joan Sebastian Saavedra Medina

Director  
Jhon Alexander Rojas Tavera  
David Alejandro Pirateque Guerrero

Trabajo de grado para optar por el título de Tecnólogo en Desarrollo de Software

Universitaria Agustiniana  
Facultad de Ingeniería  
Programa de Tecnología en Desarrollo De Software  
Bogotá, D.C.  
2022

## **Resumen**

En este documento se evidencia la implementación de una forma diferente y practica de gestionar proyectos académicos en el Centro de Emprendimiento Innovación y Desarrollo Empresarial Agustiniano CEIDEA, la cual busca que los emprendedores puedan aclarar sus dudas respecto a su idea y/o proyecto. Al momento de emprender o tener una idea de negocio, no siempre se suele tener todos los aspectos claros y esto nos conduce a cometer errores que pueden significar pérdidas en diferentes áreas, como pueden ser la organización, la economía o el valor agregado que nuestro emprendimiento tenga para llegar a ser un producto de éxito. Para mejorar estos aspectos, basados en el método de modelo de negocio de Ash Maurya denominado Lean canvas, que se desempeña en identificar evaluar y corregir, para así encontrar una óptima solución junto a otros factores a considerar para el desarrollo efectivo del modelo de negocio, el cual hace este proceso más sencillo y rápido para los emprendedores, además de contar con la asesoría de expertos del CEIDEA quienes dentro de su plataforma Gestemp donde se gestionan los proyectos de la Uniagustiniana y donde se manejará el sistema interactivo Lean canvas, manejaran los respectivos procesos; el modelo de negocio además contará con un nuevo bloque el cual hará al emprendedor ir de la mano con los objetivos de desarrollo sostenible y así cubrir los aspectos social económico y del medio ambiente en su respectivo proyecto.

*Palabras clave:* gestionar proyectos, idea de negocio, modelo de negocio, objetivos de desarrollo sostenible.

### **Abstract**

This document shows the implementation of a different and practical way of managing academic projects at the Center for Entrepreneurship, Innovation and Business Development Agustiniño CEIDEA, which seeks to enable entrepreneurs to clarify their doubts regarding their idea and/or project. When starting or having a business idea, not always all aspects are clear and this leads us to make mistakes that can mean losses in different areas, such as the organization, the economy or the added value that our venture has to become a successful product. To improve these aspects, based on the Ash Maurya business model method called Lean canvas, which works to identify, evaluate and correct, in order to find an optimal solution along with other factors to consider for the effective development of the business model, which makes this process easier and faster for entrepreneurs, in addition to having the advice of experts from CEIDEA who, within its Gestemp platform where Uniagustiniana projects are managed and where the Lean canvas interactive system will be managed, will manage the respective processes; The business model will also have a new block which will make the entrepreneur go hand in hand with the objectives of sustainable development and thus cover the social, economic and environmental aspects in their respective project.

**Keywords:** manage projects, business idea, business model, objectives of sustainable development.

## Tabla de contenidos

Resumen .....	3
Abstract.....	4
Tabla de contenidos .....	5
Introducción.....	8
1. Título del proyecto de software .....	10
2. Planeación del proyecto.....	10
2.1 Objetivos del proyecto.....	10
2.2 Planteamiento del problema y/o necesidad. ....	11
2.3 Alcance del proyecto. ....	12
2.4 Metodología de desarrollo de software .....	13
Fase 1: Planificación.....	14
Fase 2: Diseño. ....	16
Fase 3: Codificación y/o programación.....	16
Fase 4: Pruebas. ....	16
Fase 5: Lanzamiento.....	16
3. Marco teórico.....	17
3. Estado del arte .....	19
4. Especificación de Requisitos de Software (IEEE 830) .....	21
4.1 Perspectiva del Producto. ....	21
4.2 Funcionalidad del Producto. ....	21
4.3 Características de los Usuarios. ....	21
4.4 Restricciones.....	22
4.5 Suposiciones y Dependencias.....	22

4.6 Requisitos Específicos.....	23
4.6.1 Actores/Roles. ....	23
4.6.2 Requisitos Funcionales .....	24
4.6.3 Diagramas de casos de uso .....	25
4.6.4 Especificaciones de los casos de uso. ....	25
4.7 Requisitos de Rendimiento .....	30
Eficiencia. ....	30
4.8 Restricciones de Diseño .....	30
4.9 Atributos del Software del Sistema. ....	30
Seguridad y Lógica De Datos. ....	30
Usabilidad.....	31
Dependibilidad. ....	31
5. Diseño del software (ISO -12207-1) .....	32
5.1 Diseño de la Arquitectura de software. ....	32
Contexto del sistema: .....	32
Diagrama del contenedor:.....	33
Diagrama de componentes: .....	34
5.2 Diseño detallado del software. ....	35
5.2.1 Diagrama de clases. ....	35
5.3 Diseño de la Interfaz.....	39
5.3.1 Interfaz Gráfica de Usuario. ....	39
5.3.2 Interfaces de Entrada. ....	42
5.3.3 Interfaces de Salida.....	43
6. Implementación .....	44
6.1 Plataformas de desarrollo. ....	44

Frontend.....	44
Funcional y lógica. ....	44
Frameworks de diseño .....	44
Backend. ....	44
6.2 Base de datos. ....	44
6.3 Infraestructura de hardware y redes. ....	45
Especificaciones de hardware.....	45
Especificaciones de software.....	45
Sistema operativo. ....	45
Navegadores. ....	45
7. Pruebas del software.....	46
7.1 Pruebas del software.....	46
Pruebas heurísticas .....	46
Pruebas de validación y verificación .....	61
7.2 Pruebas de usabilidad. ....	72
Conclusiones.....	81
Anexos.....	82
Manual de usuario. ....	82
Vista Principal .....	82
Gestión por módulos.....	83
Nota: Los módulos que contienen tablas similares son los siguientes: .....	84
Módulos con diferente función:.....	85
ODS: .....	85
Canales. ....	87
Referencias .....	89

## **Introducción**

Para el CEIDEA de la Uniagustiniana es muy importante el manejo de la gestión de los distintos tipos de proyectos que se desarrollan para aclarar las ideas que tengan los emprendedores, es por esto que para obtener un mejor funcionamiento de estos procesos se responde la pregunta del problema que presenta la estructura manejada actualmente la cual es: ¿Es el sistema obsoleto?

En respuesta a la anterior pregunta: Para empezar la estructura desarrollada es un archivo de Excel el cual contiene una tabla, este, aunque bien puede gestionar los datos, aun así, no es fácil de manejar debido a que muchas veces su tamaño es demasiado para el envío echo por correo, puesto que además es la única forma de trabajar las actualizaciones. Dificulta o atrasa los procesos lo cual genera molestias para los usuarios, que en este caso son emprendedores y asesores, además esta se desacomoda al momento de ingresar mucho texto en los diferentes espacios generando confusión y una posible pérdida de información.

Luego de diversas investigaciones y dar una solución practica el objetivo es: Desarrollar una aplicación web para analizar y crear modelos de negocio mediante un canvas interactivo para la actualización de la plataforma Gestemp en el CEIDEA, para abordar el problema y efectuar la respectiva solución se trabaja la metodología de desarrollo Extreme Programing más conocida como X.P. En esta metodología se trabaja el proceso de manera circular, empieza por la planificación donde se definen todos los requisitos y requerimientos; sigue con el diseño donde se revisa la estructura del software y como su nombre lo dice, su diseño, luego se trabaja el desarrollo y para terminar se realizan las pruebas de caja negra y caja blanca, estas se evidencian por medio de estadísticas de encuestas realizadas, si se presentan correcciones se debe repetir el proceso con los avances ya alcanzados para realizar las respectivas correcciones.

Como resultado en el desarrollo del software se encontró una manera más dinámica para gestionar los proyectos en el CEIDEA y se concluyó en un diseño tipo dashboard además los datos que se ingresen podrán ser vistos por los demás usuarios de manera simultánea y a su vez los cambios que se quieran realizar por parte de los usuarios emprendedores, cumpliendo con los criterios de aceptación del desarrollo pedidos por el usuario y con sus respectivos cambios para un desarrollo más completo.

También se abordará la teoría correspondiente en la cual fue inspirado el método empleado para gestionar los diferentes tipos de proyectos la cual emplea un modelo de negocio y mediante este se puede trabajar de mejor manera la idea o proyecto que tenga el emprendedor.



Ya para la parte del desarrollo se hablará de la metodología de programación implementada que da su función mediante la programación orientada a objetos, donde se tendrá como lenguaje principal a Java junto a otras tecnologías para el correcto funcionamiento de las rutas y en el diseño el manejo de Bootstrap.

Para la base de datos se mostrará la estructura con su respectivo orden y relaciones y puesto a diversas investigaciones y practica en el desarrollo para una mayor agilidad en los procesos y un funcionamiento más optimo se decidió el manejo de MySQL.

Por último, se observará el manual de usuario en el que se encontrará el respectivo manejo de la aplicación web como una guía para los diferentes manejadores del sistema y así buscar que se obtenga una mejor experiencia y cumplir con todos los estándares para dar la confianza que necesita el usuario.

## **1. Título del proyecto de software**

Desarrollo de aplicación web canvas interactivo para crear modelos de negocio en el CEIDEA de Uniagustiniana.

## **2. Planeación del proyecto**

### **2.1 Objetivos del proyecto**

**2.1.1 Objetivo general.** Desarrollar una aplicación web para crear modelos de negocio mediante un canvas interactivo para la actualización de la plataforma Gestemp en el CEIDEA

#### **2.1.2 Objetivos específicos.**

- Analizar y comprender la metodología de implementación del lienzo canvas.
- Desarrollar un diseño intuitivo, llamativo e interesante para el usuario en cuanto a su contenido y calidad.
- Codificar o implementar la funcionalidad de cada módulo de la aplicación.
- Hacer pruebas del funcionamiento de la aplicación.

## 2.2 Planteamiento del problema y/o necesidad.

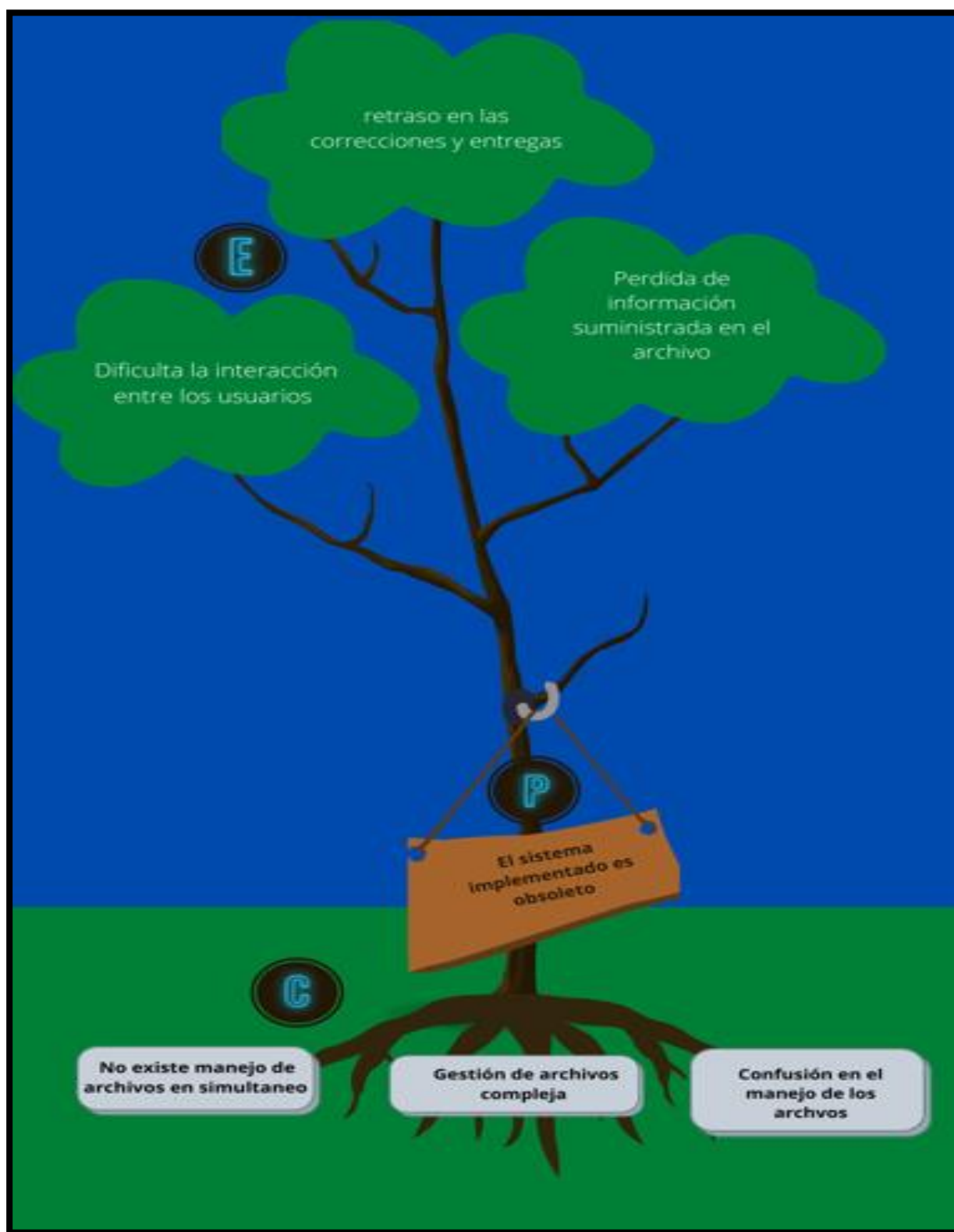


Figura 1. Dibujo árbol de problema. Autoría propia, (2022).

Tabla 1.

*Descripción del árbol de problemas.*

<b>P</b> <b>ROBLEMA</b>	Teniendo en cuenta que el lienzo implementado es un documento de Excel, el cual no permite que los usuarios interactúen en tiempo real y además las cargas de información en el documento dificultan su manejo.
<b>C</b> <b>AUSAS</b>	Estas se derivan puesto a que no se permiten cargas de información de manera simultánea, la gestión de los archivos es compleja, causa confusión en los usuarios por la manipulación indiscriminada del archivo las cuales muestran el deterioro del sistema.
<b>E</b> <b>EFECTOS</b>	Estos se crean producto de las causas que generó el problema, como son:  La complejidad en la interacción de los usuarios, el retraso en las entregas y la pérdida de información suministrada en el archivo.

*Nota.* Autoría propia (2022)

### 2.3 Alcance del proyecto.

Este trabajo se realiza con el fin de buscar una mejoría en la gestión de proyectos del CEIDEA

Teniendo en cuenta los siguientes aspectos:

- Consideración del orden de la funcionalidad y diseño del software.
- Diseño tipo dashboard para la gestión de los datos del REU.
- El Emprendedor podrá crear, actualizar, ver y borrar datos los demás usuarios solo verán los datos.
- Poder ver los cambios efectuados en el formato del lienzo, (agregados y/o quitados).
- Solo el emprendedor podrá gestionar los módulos y los demás usuarios ver el resultado de esa gestión.
- Solo el emprendedor validará por módulos y filtrará los datos para verlos en el lienzo.
- El usuario tendrá una guía expuesta en la aplicación para manejarla correctamente.

El proyecto tendrá como alcance los siguientes aspectos específicos:

- canvas intuitivo y dinámico.
- Funcionalidad de Objetivos de Desarrollo Sostenible ODS dentro del canvas.
- Interacción en tiempo real de los datos.

- Poder crear, actualizar, eliminar leer y filtrar los datos.

## 2.4 Metodología de desarrollo de software

La metodología de investigación que se utiliza en el desarrollo de la aplicación web es cualitativa puesto que lo que se busca es reducir la falta de organización al momento de estructurar una idea de proyecto o negocio por medio de un lienzo Canvas interactivo el cual tendrá la opción de incluir varios datos en todos los módulos buscando con esto la mejor opción para estructurar las ideas del usuario. El enfoque será subjetivo ya que busca mejorar la percepción personal que los usuarios tienen respecto a sus ideas.

Conociendo las necesidades para el contexto del desarrollo del software se está empleando una metodología ágil, la cual nos será expuesta gracias a (Elena Bello, 2021):

Extreme Programming o XP Programming es un marco de desarrollo de software ágil que tiene como objetivo producir un software de mayor calidad para mejorar la eficiencia del equipo de desarrollo. Se trata de una metodología de desarrollo cuyo objetivo es promover la aplicación de prácticas de ingeniería apropiadas para la creación de software. Esta metodología la formuló Kent Beck, autor del primer libro sobre este ámbito llamado «Extreme Programming Explained: Embrace Change», publicado en 1999.

Extreme Programming fue la metodología dominante en el mundo ágil en los años 2.000. Luego, el *framework* Scrum le quitó el puesto en cuanto a popularidad.

A pesar de ello, se trata de una de las metodologías ágiles de desarrollo de software más exitosas. La combinación de esta, junto con la Scrum, asegura un enorme control sobre los proyectos y una implementación mucho más efectiva.

Para que nos hagamos una idea, Extreme Programming está diseñada para ofrecer el software que los usuarios necesitan en el momento adecuado. En este sentido, ayuda a los desarrolladores a ajustarse a los requerimientos cambiantes de los clientes.

Este tipo de programación se diferencia de las metodologías tradicionales en que pone más énfasis en la adaptabilidad que en la previsibilidad. El Extreme Programming considera que los cambios de requisitos sobre la marcha son acciones naturales e inevitables en el desarrollo de un proyecto. Creer que ser capaces de adaptarse a los cambios que puedan surgir en cualquier punto del ciclo de vida de un proyecto es una mejor previsión y más realista que intentarlos definir todos en un principio y que no varíen más.

Para la comprensión de la función de la metodología a continuación se darán las fases que la componen y como se deben desempeñar:



**Figura 2.** Mapa metodología X.P. OpenWebinars, (2022).

### Fase 1: Planificación.

En esta fase se deberá tener en cuenta los criterios del cliente para con su aplicación y así mismo darle constantes avances en los cuales él pueda decidir si el desarrollo cumple con lo que él quiere o si por el contrario se debe cambiar a su gusto, para esto trabajamos con las historias de usuario.

Tabla 2.

*Descripción de historias de usuario.*

Historias de usuario				
ID	Rol	Característica o Funcionalidad	Resultado	Criterio de aceptación
1	Desarrolladores y cliente	Documentación	Definición de requisitos.	Consideración del orden de la funcionalidad y diseño del software.
2	Cliente	Creación de entorno manejable a través de una conexión a internet.	Generar una versión aplicable a la Ruta del Emprendimiento Uniagustiniano REU.	Diseño tipo dashboard para la gestión de los datos del REU.

3	Cliente	Validación de datos por cada módulo según el usuario.	Gestión de los datos.	El Emprendedor podrá crear, actualizar, ver y borrar datos los demás usuarios solo verán los datos.
4	Cliente	Creación del formato de lienzo.	Estructura funcional y dinámica.	Poder ver los cambios efectuados en el formato del lienzo, (agregados y/o quitados).
5	Cliente	Organización tipo dashboard.	Módulos mediante un menú según el usuario.	Solo el emprendedor podrá gestionar los módulos y los demás usuarios ver el resultado de esa gestión.
6	Desarrolladores	Diseño de una interfaz llamativa.	Validación por módulos y filtrar los datos para verlos en el lienzo.	Solo el emprendedor validará por módulos y filtrará los datos para verlos en el lienzo.
7	Cliente	Administración de vistas	La información que se muestra es la correspondiente al usuario ingresado.	El administrador podrá crear y eliminar los usuarios que interactuarán en el sistema
8	Cliente	Experiencia de usuario intuitiva	Se muestra la información referente a los eventos, validaciones y posibles errores.	Se debe tener validación y muestra de posibles errores de manera más intuitiva y atrayente para el usuario.

9	Desarrolladores	Manuales	Guía de la funcionalidad de la aplicación.	El usuario tendrá una guía expuesta en la aplicación para manejarla correctamente.
---	-----------------	----------	--	--

*Nota.* Autoría propia (2022).

### **Fase 2: Diseño.**

Para el diseño se debe tener en cuenta también el requerimiento del cliente y se desarrollará a medida que se avanza en el desarrollo. Esta fase la encontramos en el literal: **Diseño del software (ISO -12207-1)** pg. 21.

### **Fase 3: Codificación y/o programación.**

Aquí se debe tener en cuenta un trabajo colaborativo: En este caso uno desarrollará la aplicación y el otro la documentación, sin dejar de lado que se pueden ayudar mutuamente para las diversas tareas.

### **Fase 4: Pruebas.**

Para las pruebas se debe tener en cuenta una constante revisión de la funcionalidad del software a medida que se avanza y así mismo permitir al cliente corroborar estos avances probando el software. En este caso se dispondrá el software al cliente y distintas personas para una mejor revisión y realización del mismo.

### **Fase 5: Lanzamiento.**

Para que este punto sea finiquitado se debe tener en cuenta que los anteriores deben cumplirse por completo y así solo resta subir el software a un servidor o como en nuestro caso cumpliendo los requerimientos del cliente para entregarlo.



### 3. Marco teórico

Para la Uniagustiniana es indispensable el dar a conocer los orígenes puesto a que por medio de una idea de negocio se puede lograr un excelente emprendimiento o mejorar una idea mucho más y facilitar la comprensión para quienes quieran o deban integrarse y comprender de mejor manera la gestión de sus proyectos. Para hablar del modelo de negocio creado por Ash Maurya, se tiene que hablar primero de que es un modelo de negocio y para qué, cómo nos será dicho a continuación gracias a (Molina, 2022):

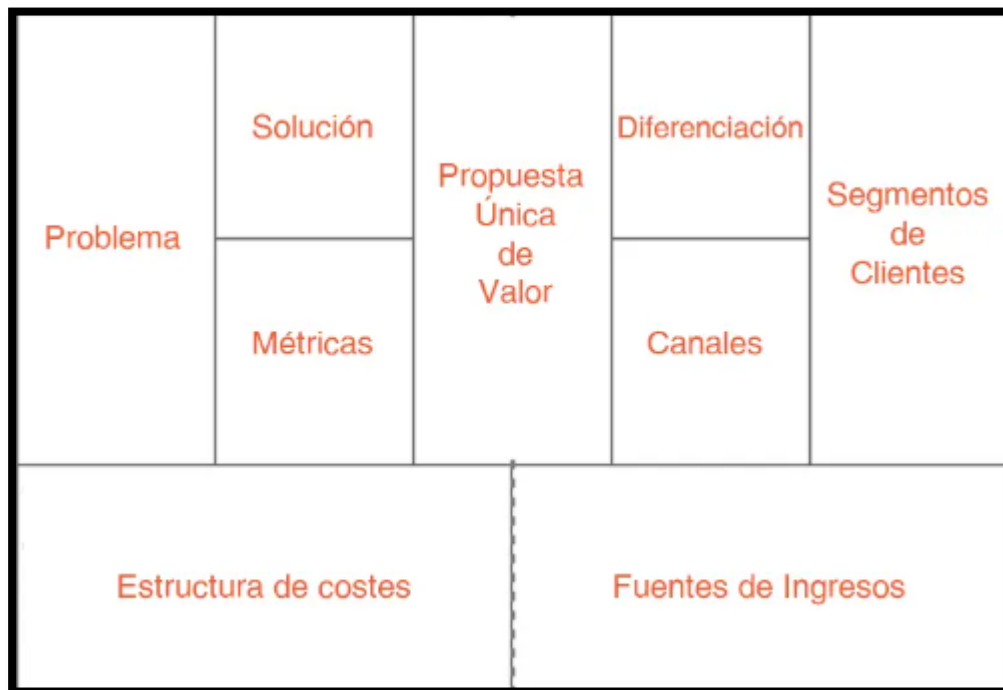
El Business Model Canvas o Modelo Canvas es un modelo muy visual con el que podremos ordenar nuestras ideas a la hora de definir cuál será nuestro modelo de negocio. Desarrollado por Alexander Osterwalder, se trata de un modelo ideal para determinar y crear modelos innovadores con el objetivo de generar valor para los clientes, definiendo y creando modelos de negocio innovadores a través de cuatro grandes áreas (los clientes, la oferta, la infraestructura y la viabilidad económica) que se desarrollan en nueve divisiones, apartados o casillas.

Hablamos de un modelo simple que permite su uso con facilidad, sin que eso suponga que se dejen de lado las cuestiones más estratégicas que repercuten en un negocio. Abordando la problemática de la empresa desde un punto de vista interno y otro externo, el Modelo Canvas permite que tengamos una idea clara del proyecto con un simple golpe de vista. Eso sí, para llevarlo a cabo se deben ir completando sus apartados en el orden correcto.

Ash Maurya formula el siguiente esquema, en donde se puede ver los principales módulos que conforman el lienzo Lean Canvas los cuales se entienden de la siguiente manera gracias a (Sanchez,2020):

- Propuesta única de valor: Que ofrecemos a nuestros clientes, siempre enfocados a beneficios y no funcionalidades.
- Segmentos de clientes: A quién nos dirigimos.
- Estructura de costes: En qué tenemos que gastar para desarrollar y ofrecer nuestra propuesta de valor.
- Fuentes de ingresos: Cómo vamos a monetizar nuestro proyecto. Junto con la estructura de costes forman la rentabilidad del modelo de negocio.
- Problema: Qué problema hemos detectado que es suficientemente doloroso para un buen número de personas. Podemos listar los 3 subproblemas más importantes que hayamos detectado sobre una problemática concreta.
- Solución: Qué solución proponemos para paliar el problema. Podemos listar las 3 características principales de la solución.

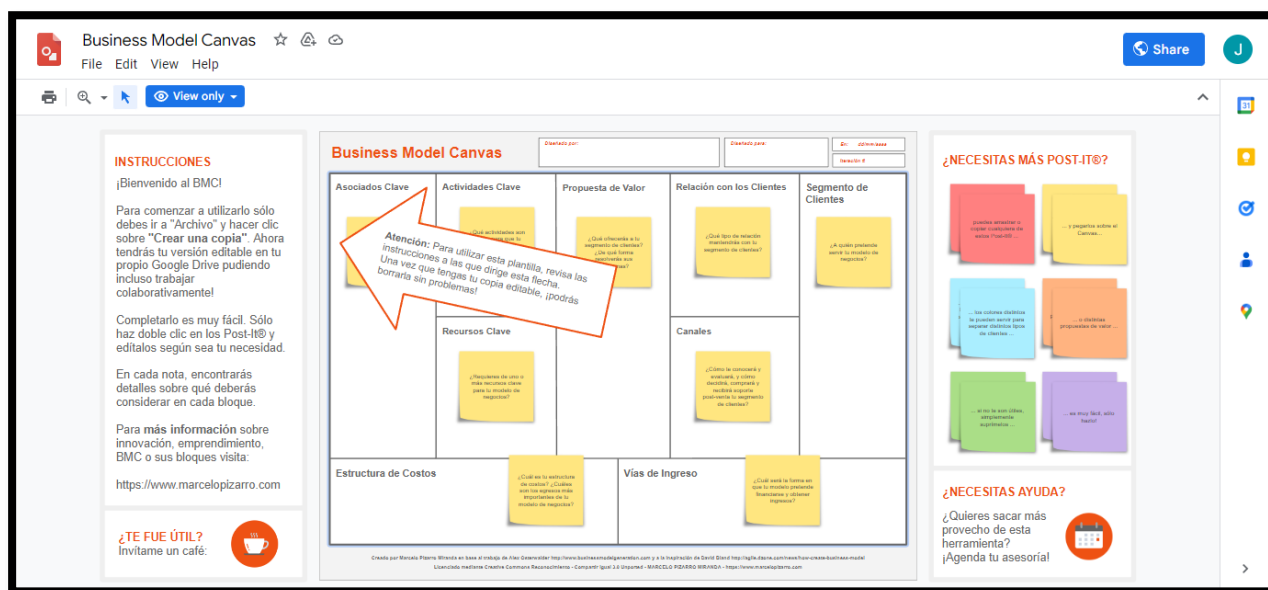
- Métricas: En qué números nos tenemos que fijar para saber el estado de nuestro negocio.
- Diferenciación (o ventaja competitiva): Qué nos diferencia realmente de la competencia de forma que no sea fácil imitar.
- Canales: A través de qué medios proporcionamos nuestra propuesta de valor a nuestros clientes.



**Figura 3.** Figura Lean Canvas. Xavi Sánchez, (2020).

### 3. Estado del arte

Como referencia encontramos las siguientes aplicaciones las cuales comparten similitudes con nuestro proyecto las cuales nos son una competencia directa ya que nuestro enfoque es para gestionar proyectos dentro de la Uniagustiniana y además tener en cuenta los Objetivos de Desarrollo Sostenible:



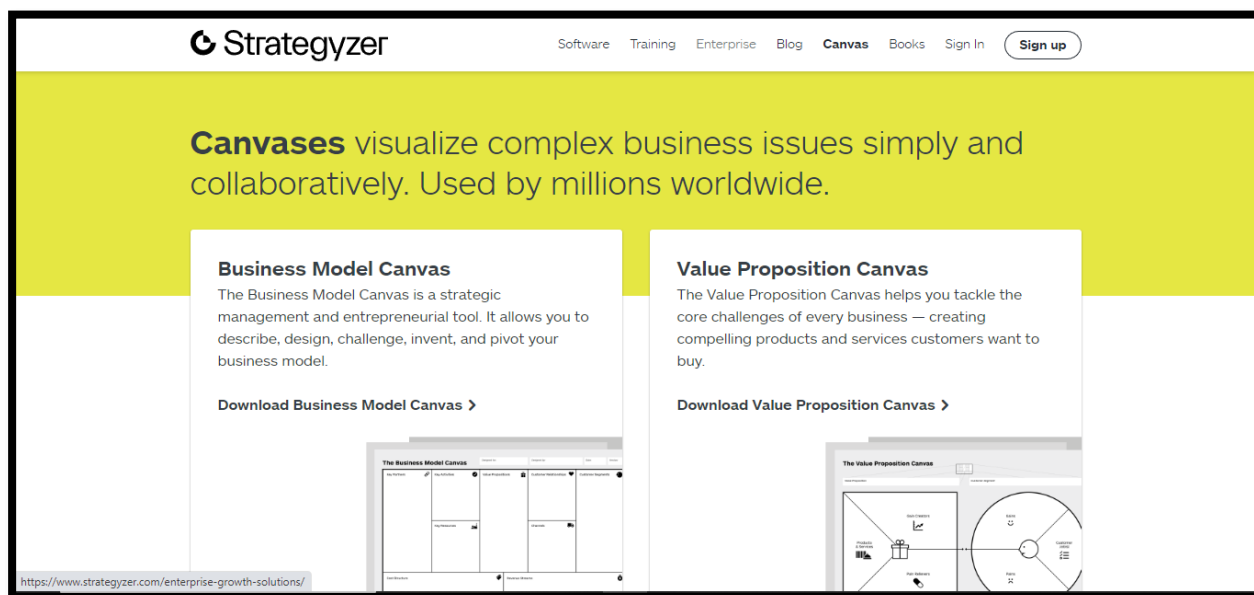
**Figura 4.** Plantilla de Business Model Canvas. Pizarro Miranda, (2011).

Esta plantilla maneja el método de modelo de negocio lean canvas, y además se trabaja por medio de la herramienta google donde además se encuentra el modo de uso de la misma, en este caso hablamos del clásico ya que tiene una versión 2.0 la cual es paga.



**Figura 5.** Página Canvanizer. Canvanizer, (2019).

Es una plataforma en la cual se puede escoger o crear lienzos y poder exportarlos, no solo para un modelo de negocio sino para diferentes tipos.



**Figura 6:** Imagen página Strategyser. Strategyser, (2020).

Este es un software que permite trabajo en tiempo real solo que se basa en el modelo de negocio canvas tradicional y es pago.



**Figura 7:** Imagen Página web EDIT. EDIT.org, (s.f).

Este modelo de negocios se maneja como aplicación, permite compartir el tablero por redes sociales, además cuenta con el modelo SWOT, o bien conocido como análisis de alcance.

## 4. Especificación de Requisitos de Software (IEEE 830)

### 4.1 Perspectiva del Producto.

Esta aplicación contiene una página inicial donde se observa una estructura llamada lienzo la cual al inicio se encuentra vacía, al lado izquierdo tiene un menú en el cual se encuentran los módulos donde se ingresa el texto correspondiente, y al enviar los datos los guarda en ese módulo para poder editarlos y seleccionarlos, al seleccionarlos esto hace que también aparezca esa selección en el lienzo principal y el máximo de cartas o textos que deja crear por módulo es indefinido.

### 4.2 Funcionalidad del Producto.

La funcionalidad se explicará por la interacción que corresponde a cada usuario dentro del sistema:

Emprendedor:

- Ingresa y observa un lienzo canvas totalmente vacío mostrando sus datos básicos
- Puede crear, actualizar, leer, eliminar y filtrar datos.
- Al filtrar un dato este se verá en el lienzo.
- Al realizar algún cambio este se revelará instantáneamente.

Administrador:

- Ingresa y observa un lienzo canvas totalmente vacío mostrando sus datos básicos.
- Puede ver los diferentes usuarios que creo y su respectiva información.
- Puede eliminar un usuario de ser necesario.
- Puede observar los cambios de los lienzos de los emprendedores.

Asesor:

- Ingresa y observa un lienzo canvas totalmente vacío mostrando sus datos básicos.
- Puede observar los cambios de los lienzos de los emprendedores.

### 4.3 Características de los Usuarios.

Tabla 3.

*Consideraciones de los usuarios.*

<b>Características generales:</b>	<ul style="list-style-type: none"> <li>• La edad de los usuarios será desde los 14 años en adelante</li> </ul>
-----------------------------------	--

	<ul style="list-style-type: none"> <li>• El usuario debe leer el manual para entender el manejo básico de la plataforma.</li> </ul>
<b>Usuario administrador:</b>	<ul style="list-style-type: none"> <li>• Conocimientos básicos de manejo de redes e internet.</li> <li>• Conocimientos básicos de ofimática.</li> <li>• Conocimientos técnicos sobre el manejo de la plataforma.</li> <li>• Conocimientos técnicos sobre el control de demás usuarios.</li> </ul>
<b>Usuario asesor:</b>	<ul style="list-style-type: none"> <li>• Conocimientos básicos de manejo de redes e internet.</li> <li>• Conocimientos técnicos sobre el manejo de la plataforma.</li> <li>• conocimientos técnicos para asesorar a los emprendedores.</li> </ul>
<b>Usuario Emprendedor:</b>	<ul style="list-style-type: none"> <li>• Conocimientos básicos de manejo de redes e internet.</li> <li>• Conocimientos técnicos de su área de estudio.</li> <li>• Conocimientos básicos de su idea de emprendimiento</li> </ul>

*Nota.* Autoría propia (2022).

#### 4.4 Restricciones.

- El lenguaje que se utilizará será Java.
- No se realizará envío de archivos
- Cada posible error o carga tendrá una respectiva animación.
- La creación de un nuevo lienzo será a partir de la filtración de los datos correspondientes al modelo de negocio del usuario.

#### 4.5 Suposiciones y Dependencias.

- Los equipos donde se maneja la aplicación deberán contar con el mínimo de los recursos para su correcta función.

- El uso del software dependerá de una conexión a internet.
- El sistema dependerá del ingreso a la plataforma Gestemp para ser utilizado.
- El Front sólo será implementado por medio de Thymeleaf y Bootstrap.

#### 4.6 Requisitos Específicos.

##### 4.6.1 Actores/Roles.

Tabla 4.

*Acciones de los usuarios.*

Usuarios	Permitido	No permitido
Administrador	<ul style="list-style-type: none"> <li>• Agregar datos en la plataforma</li> <li>• Actualizar datos en la plataforma</li> <li>• Ver los datos agregados por el usuario emprendedor y el usuario asesor</li> <li>• Agregar o quitar cualquier usuario</li> </ul>	<ul style="list-style-type: none"> <li>• Eliminar los datos agregados por el usuario emprendedor y el usuario asesor</li> <li>• Actualizar los datos agregados por el usuario emprendedor y el usuario asesor</li> </ul>
Asesor	<ul style="list-style-type: none"> <li>• Agregar datos en la plataforma</li> <li>• Actualizar datos en la plataforma</li> <li>• Ver los datos agregados por el usuario emprendedor</li> <li>• Notificar al usuario emprendedor un cambio a realizar</li> </ul>	<ul style="list-style-type: none"> <li>• Eliminar datos en la plataforma</li> <li>• Ver los datos agregados por el usuario administrador</li> <li>• Eliminar los datos agregados por el usuario emprendedor</li> <li>• Eliminar los datos agregados por el usuario administrador</li> <li>• Actualizar los datos agregados por el usuario emprendedor</li> <li>• Actualizar los datos agregados por el usuario administrador</li> </ul>

Emprendedor	<ul style="list-style-type: none"> <li>• Agregar datos en la plataforma</li> <li>• Actualizar datos en la plataforma</li> <li>• Seleccionar datos de la plataforma</li> <li>• Modificar la selección de los datos</li> </ul>	<ul style="list-style-type: none"> <li>• Eliminar datos de la plataforma</li> <li>• Ver los datos agregados por el usuario administrador</li> <li>• Eliminar los datos agregados por el usuario emprendedor</li> <li>• Eliminar los datos agregados por el usuario administrador</li> </ul>
-------------	--	---

*Nota.* Autoría propia (2022).

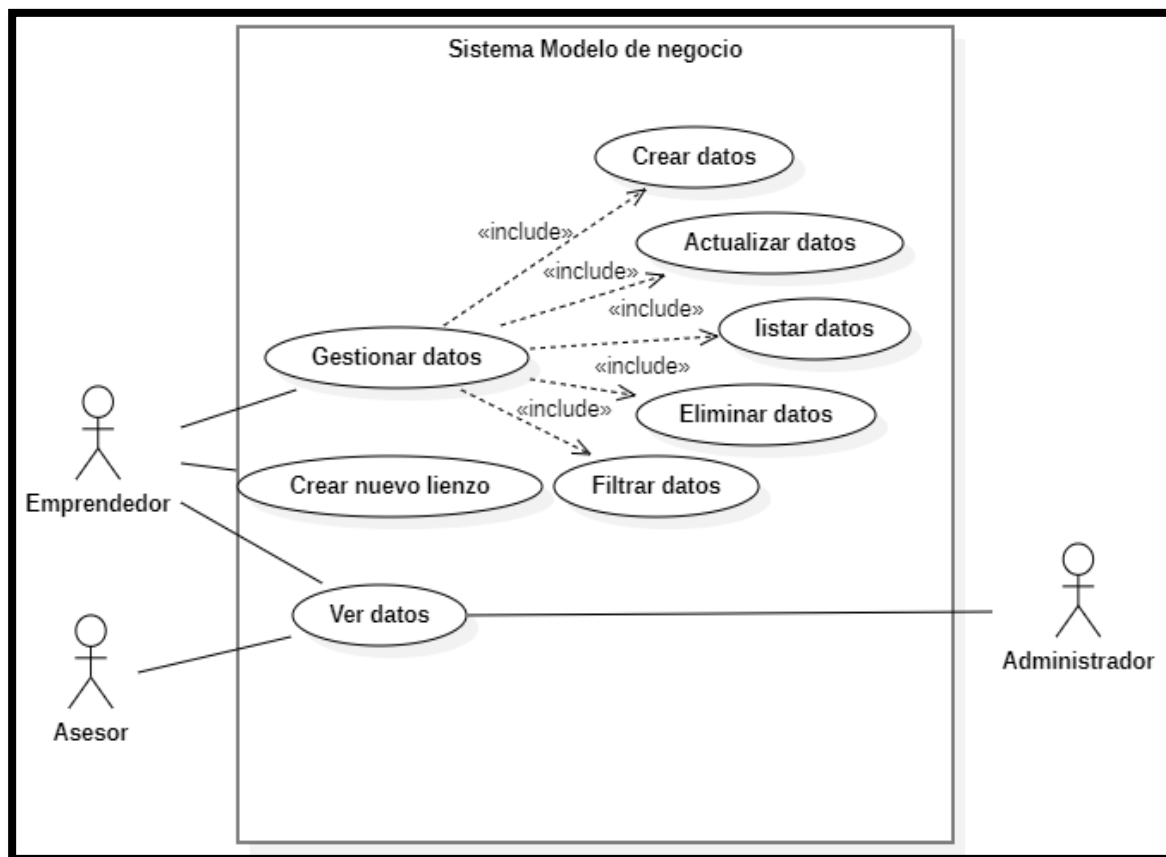
#### 4.6.2 Requisitos Funcionales

- El sistema asignará a los usuarios emprendedores un lienzo Lean Canvas en blanco.
- El sistema permitirá el envío de datos al usuario emprendedor referentes a la idea de negocio para iniciar en el modelo canvas.
- El sistema permitirá al usuario emprendedor ingresar la información de cada módulo donde sea necesario.
- El sistema permitirá al usuario emprendedor actualizar la información de cada módulo
- El sistema permitirá crear un nuevo lienzo Lean Canvas a través de la selección filtrada entre los módulos segmento, solución y problema.
- El sistema permitirá al usuario emprendedor filtrar los datos deseados del lienzo Lean Canvas.
- El sistema permitirá al usuario emprendedor imprimir o capturar la imagen del lienzo Lean Canvas completo o cuando está en progreso.
- El sistema permitirá al usuario administrador ver la información básica de los demás usuarios.
- El sistema permitirá al usuario administrador ver los cambios de todos los lienzos
- El sistema permitirá a los usuarios asesores ver los cambios de todos los lienzos.
- El sistema contará con una interfaz dinámica e intuitiva para el correcto manejo de los usuarios.
- El sistema mostrará mensajes previos en la carga de algún espacio de ser necesario.
- El sistema contará con validaciones en los campos para la correcta diligencia de estos.



- El sistema contará con guías y/o imágenes según el módulo para orientar al usuario.

#### 4.6.3 Diagramas de casos de uso



**Figura 8.** Diagrama de casos de uso. Autoría propia, (2022).

#### 4.6.4 Especificaciones de los casos de uso.

Tabla 5.

*Especificación caso de uso 1.*

Nombre:	Gestionar datos
Autor:	Joan Saavedra, Daniel Ramírez
Fecha:	18/08/2022
Descripción: El sistema maneja una estructura en blanco donde se verán los cambios realizados por los emprendedores.	

Actores: Emprendedor.	
Precondiciones: Es necesario aplicar los cambios por cada módulo para ver los cambios del lienzo.	
<p>Flujo Normal: Emprendedor, Sistema.</p> <ol style="list-style-type: none"> <li>1. El emprendedor accede al sistema.</li> <li>2. El sistema muestra principalmente el lienzo al emprendedor.</li> <li>3. El emprendedor crea datos.</li> <li>4. El sistema guarda el dato creado</li> <li>5. El emprendedor actualiza un dato.</li> <li>6. El sistema muestra el dato actualizado</li> <li>7. El emprendedor lista los datos.</li> <li>8. El sistema muestra los datos listados.</li> <li>9. El emprendedor procede a filtrar los datos para llenar su lienzo.</li> <li>10. El sistema ubica los datos filtrados en el lugar que corresponde en el lienzo.</li> <li>11. El emprendedor procede a ver los datos que selecciona en su lienzo.</li> <li>12. El sistema muestra los datos que están en el lienzo.</li> </ol>	<p>Flujo Alternativo:</p> <p>1A. El sistema muestra un error 400 de conexión de red y pide volver a recargar la página debido a que la red se cayó.</p> <p>2A. El emprendedor diligenció mal un dato por lo tanto tuvo que actualizar y corregir el dato.</p> <p>1B. Algún campo no fue llenado y al enviarlo pedirá que por favor lo diligencie para su correcto envío.</p>
Postcondiciones: El emprendedor verá toda la información seleccionada plasmada en el lienzo.	
Prioridad: Alta ya que se desarrolla la plataforma con esta información para mantener el proceso de los emprendedores.	

Nota. Autoría propia (2022).

Tabla 6.  
Especificación caso de uso 2.

Nombre:	Crear nuevo lienzo
Autor:	Joan Saavedra, Daniel Ramírez
Fecha:	18/08/2022
<p>Descripción:</p> <p>El sistema maneja una estructura en blanco donde se verán los cambios realizados por los emprendedores.</p>	
Actores: Emprendedor.	
<p>Precondiciones:</p> <p>Es necesario aplicar los cambios por cada módulo para ver los cambios del lienzo.</p>	
<p>Flujo Normal:</p> <p>Emprendedor, sistema.</p> <ol style="list-style-type: none"> <li>1. El emprendedor procede a eliminar los datos filtrados.</li> <li>2. El sistema elimina los datos seleccionados.</li> <li>3. El emprendedor procede a filtrar los datos para formar un nuevo lienzo.</li> <li>4. El sistema muestra los datos que se filtran en el lienzo.</li> </ol>	<p>Flujo Alternativo:</p> <p>1A.El sistema muestra un error 400 de conexión de red y pide volver a recargar la página debido a que la red se cayó.</p> <p>2A.El emprendedor seleccionó mal un dato por lo tanto eliminará ese dato y seleccionará el correcto.</p>
<p>Postcondiciones: El emprendedor verá toda la información seleccionada plasmada en el lienzo.</p>	
<p>Prioridad: Alta Puesto que sin esta información no se puede validar ningún proyecto.</p>	

Nota. Autoría propia (2022).

Tabla 7.  
Especificación caso de uso 3.

Nombre:	Ver datos
Autor:	Joan Saavedra, Daniel Ramírez
Fecha:	18/08/2022
<p>Descripción:</p> <p>El sistema maneja una estructura en blanco donde se verán los cambios realizados por los emprendedores.</p>	
Actores: Asesor.	
<p>Precondiciones:</p> <p>Es necesario aplicar los cambios por cada módulo para ver los cambios del lienzo.</p>	
<p>Flujo Normal:</p> <p>Asesor, sistema.</p> <ol style="list-style-type: none"> <li>1. El asesor accede al sistema.</li> <li>2. El sistema muestra la interfaz del asesor.</li> <li>3. El asesor escoge un lienzo para revisar.</li> <li>4. El sistema muestra los datos que el asesor selecciona.</li> <li>5. El administrador ve los datos del lienzo correspondiente.</li> <li>6. El sistema muestra los datos que el administrador selecciona.</li> </ol>	<p>Flujo Alternativo:</p> <p>1A.El sistema muestra un error 400 de conexión de red y pide volver a recargar la página debido a que la red se cayó.</p>
<p>Postcondiciones: El emprendedor y el Asesor verán toda la información seleccionada plasmada en el lienzo.</p>	

Prioridad: Alta ya que se requiere la plataforma con esta información para ver el proceso de los emprendedores.

*Nota.* Autoría propia (2022).

Tabla 8.

*Especificación caso de uso 4.*

Nombre:	Ver datos
Autor:	Joan Saavedra, Daniel Ramírez
Fecha:	18/08/2022
<p>Descripción:</p> <p>El sistema maneja una estructura en blanco donde se verán los cambios realizados por los emprendedores.</p>	
Actores: Administrador.	
<p>Precondiciones:</p> <p>Es necesario aplicar los cambios por cada módulo para ver los cambios del lienzo.</p>	
<p>Flujo Normal:</p> <p>Administrador, sistema.</p> <ol style="list-style-type: none"> <li>7. El Administrador accede al sistema.</li> <li>8. El sistema muestra la interfaz del asesor.</li> <li>9. El administrador crea un usuario.</li> <li>10. El usuario habilita ese usuario.</li> <li>11. El asesor escoge un usuario para revisar.</li> <li>12. El sistema muestra los datos que el asesor selecciona.</li> <li>13. El asesor elimina un usuario.</li> <li>14. El sistema elimina el usuario.</li> </ol>	<p>Flujo Alternativo:</p> <p>2A.El sistema muestra un error de 400 conexión de red y pide volver a recargar la página debido a que la red se cayó.</p>

Postcondiciones: El emprendedor y el Asesor verán toda la información seleccionada plasmada en el lienzo.

Prioridad: Alta ya que se requiere la plataforma con esta información para ver el proceso de los emprendedores.

*Nota.* Autoría propia (2022).

#### **4.7 Requisitos de Rendimiento**

##### **Eficiencia.**

- El sistema al almacenará los datos en un tiempo estimado de 4 segundos
- Los datos modificados en la base de datos deberán ser actualizados para todos los usuarios que acceden en menos de 2 segundos.

#### **4.8 Restricciones de Diseño**

- El diseño implementado en el software y su respectivo análisis se crea según los principios del paradigma orientado a objetos
- El lenguaje escogido para trabajar este paradigma es Java, puesto que es el lenguaje del cual se tiene dominio por parte del equipo de desarrollo.
- Como infraestructura de Java se trabajará Spring boot el cual por medio de JPA configura el mapeo de los datos XML en los cuales se trabajan con el motor de plantillas Thymeleaf
- El framework de diseño utilizado para este software es Bootstrap
- La base de datos implementada para el desarrollo es SQL - MySQL puesto que es multiplataforma fácil de entender y es open source y lo más importante se puede crear BD relacionales.

#### **4.9 Atributos del Software del Sistema.**

##### **Seguridad y Lógica De Datos.**

- Los permisos de acceso al sistema podrán ser cambiados solamente por el administrador de acceso a datos.
- El sistema aplicará patrones y recomendaciones de programación que incrementen la seguridad de los datos.
- El sistema se inicia nuevamente luego de 5 minutos de inactividad.

**Usabilidad.**

- El tiempo de aprendizaje del sistema por un usuario será menor a una hora
- El sistema mantendrá un orden intuitivo que permite disminuir el margen de error por equivocaciones.
- El sistema contará con manuales de usuario estructurados adecuadamente
- El sistema proporcionará mensajes de error al usuario con el fin de orientarlo correctamente
- La comunicación se establecerá a través de protocolos HTTP (cliente, servidor).
- El sistema poseerá una interfaz gráfica bien formada
- Al tratarse de una aplicación web, se requiere un ordenador con un navegador convencional y una conexión a Internet básica.

**Dependibilidad.**

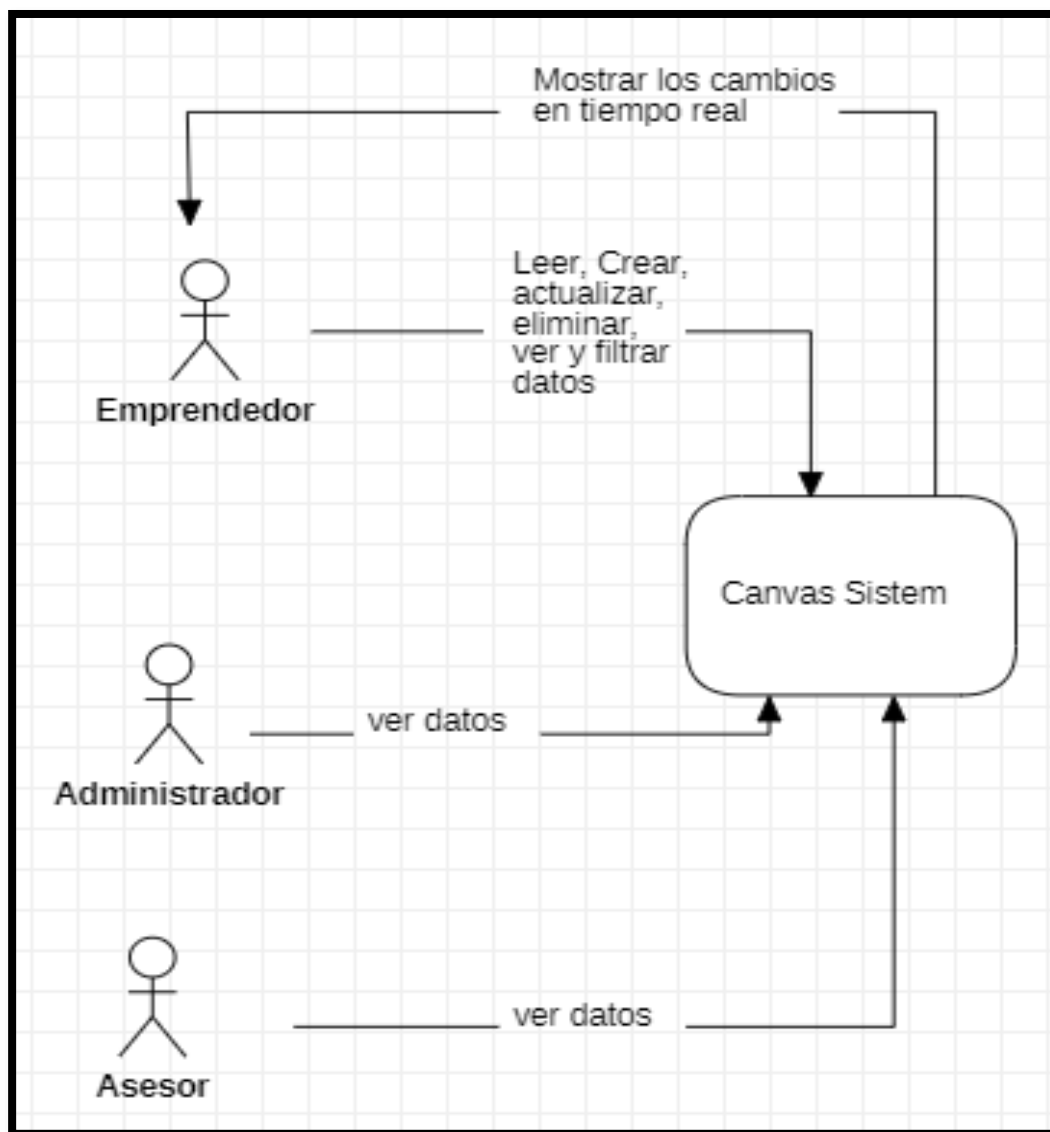
- El sistema tendrá una disponibilidad del 99,99% de las veces en que un usuario intente acceder.
- El tiempo para iniciar o reiniciar el sistema será menor a 5 minutos.
- El tiempo para iniciar el sistema no podrá ser mayor a 2 minutos.
- El sistema será compatible con los siguientes navegadores; Google, Firefox y Opera.

## 5. Diseño del software (ISO -12207-1)

### 5.1 Diseño de la Arquitectura de software.

#### Modelos C4

#### Contexto del sistema:



**Figura 9.** Imagen de contexto del sistema. Autoría propia, (2022).



### Diagrama del contenedor:

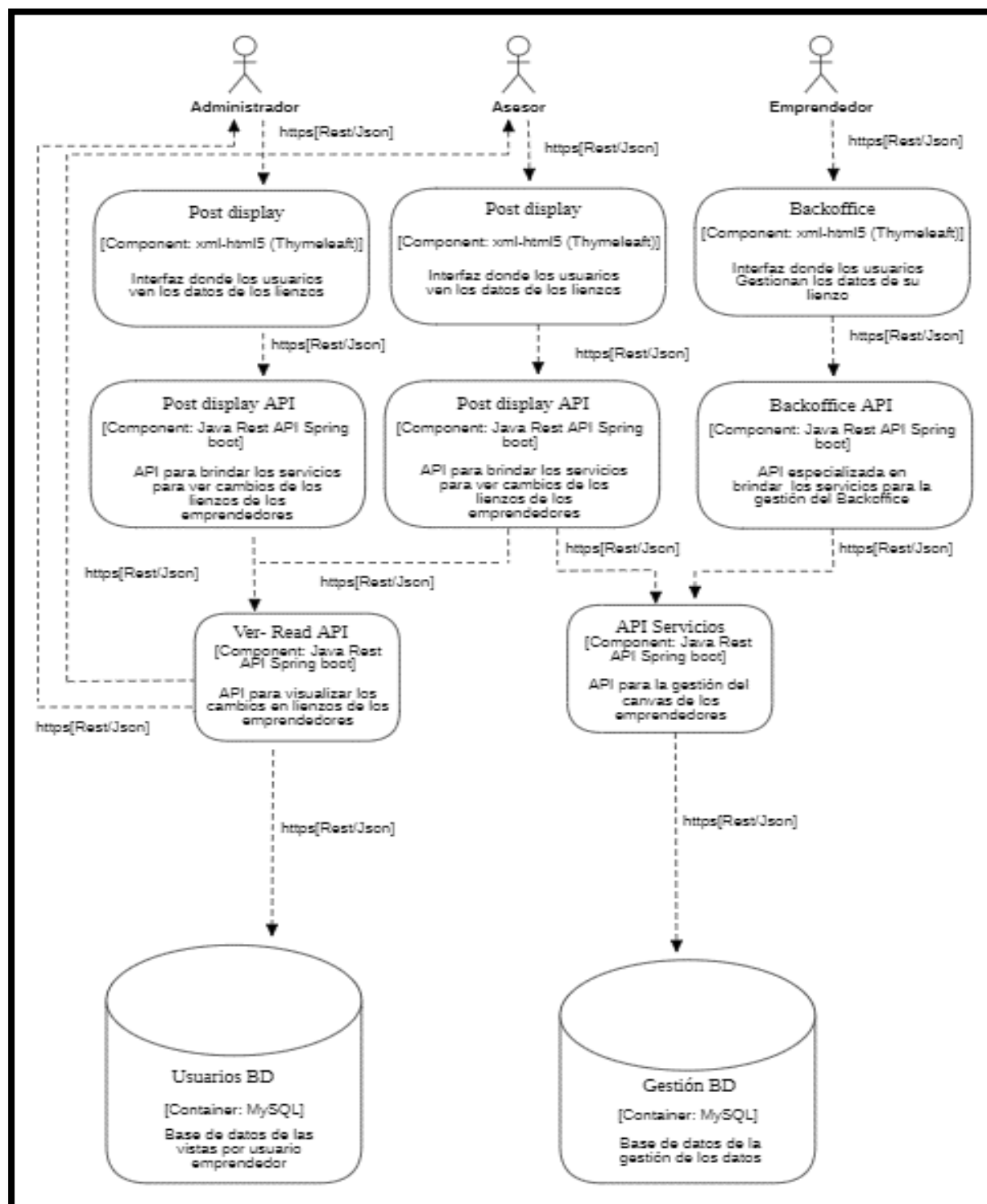
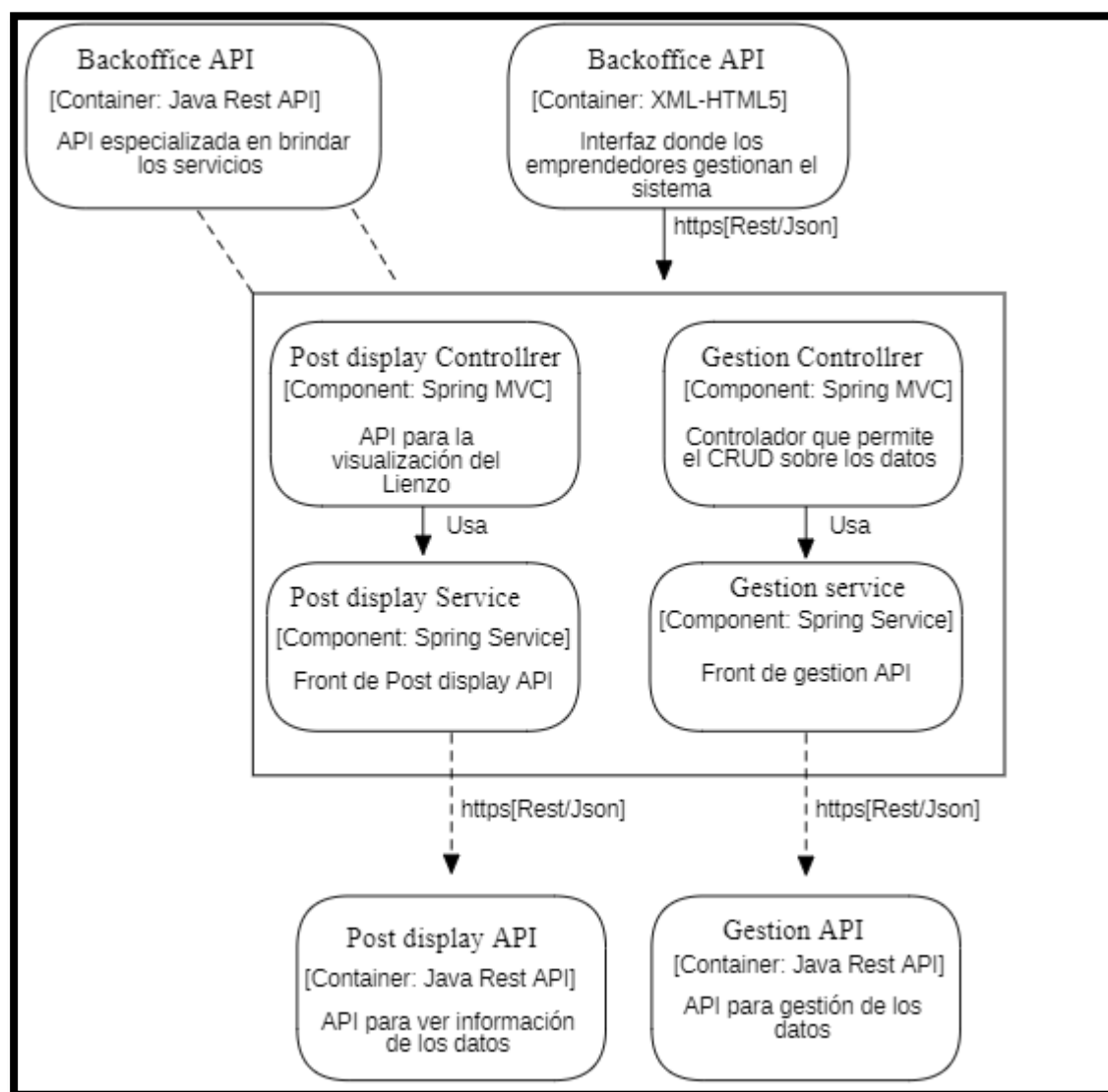


Figura 10. Imagen de diagrama del contenedor. Autoría propia, (2022).

### Diagrama de componentes:



**Figura 11.** Imagen de diagrama de componentes. Autoría propia, (2022).

## 5.2 Diseño detallado del software.

### 5.2.1 Diagrama de clases.

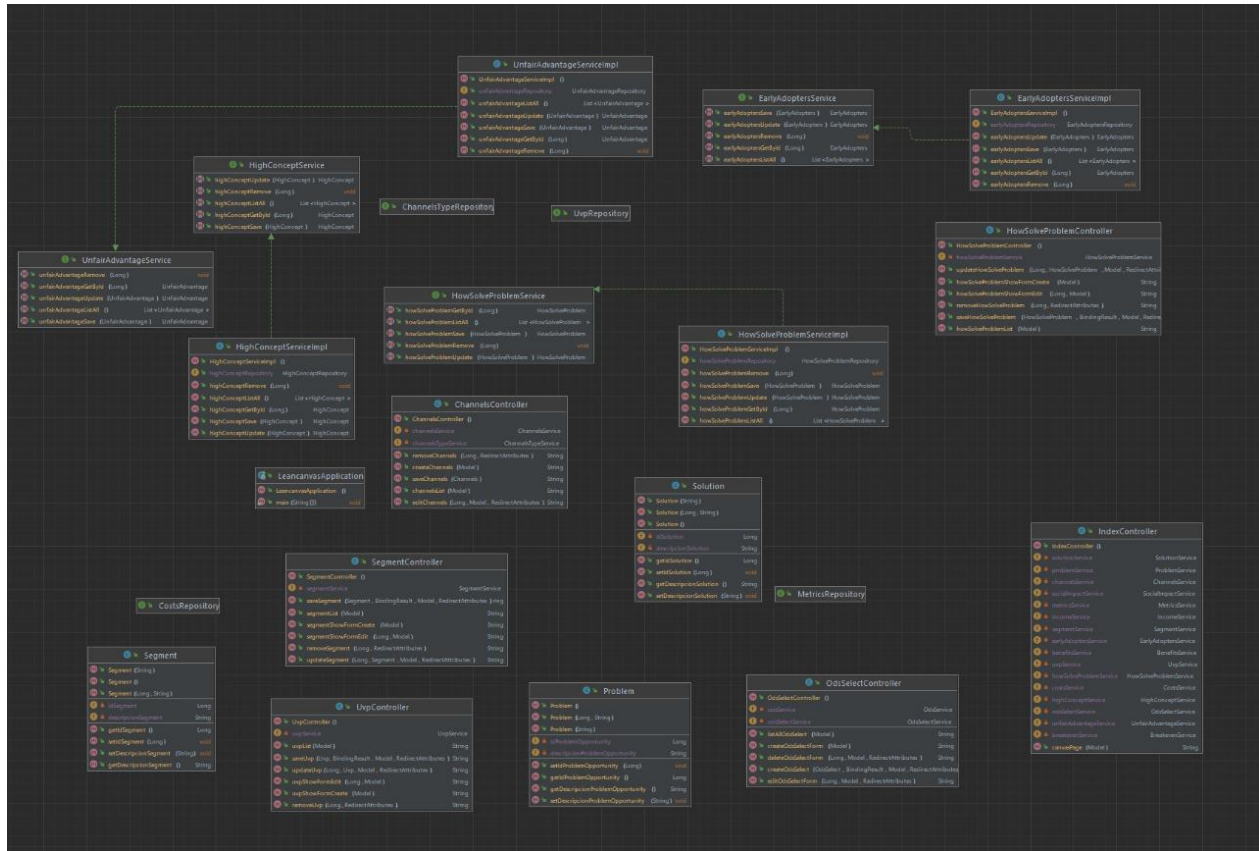
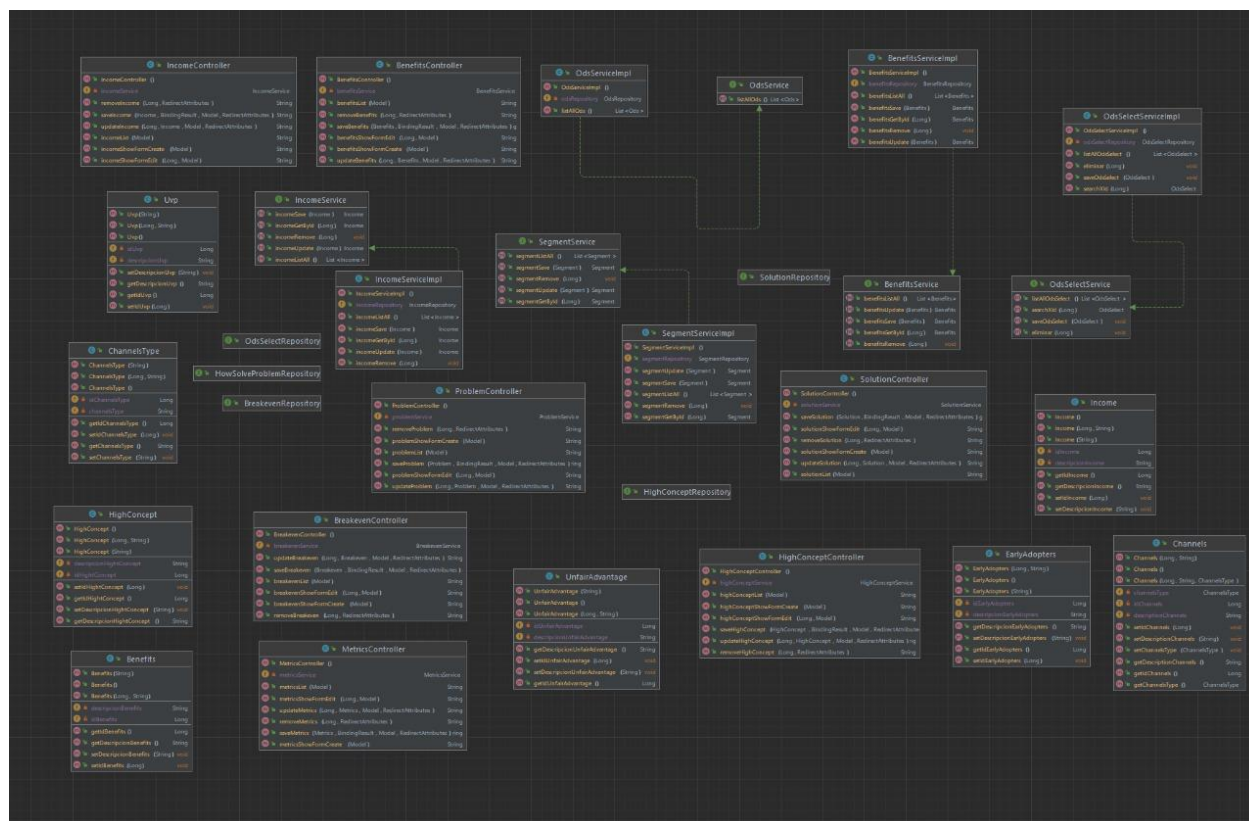


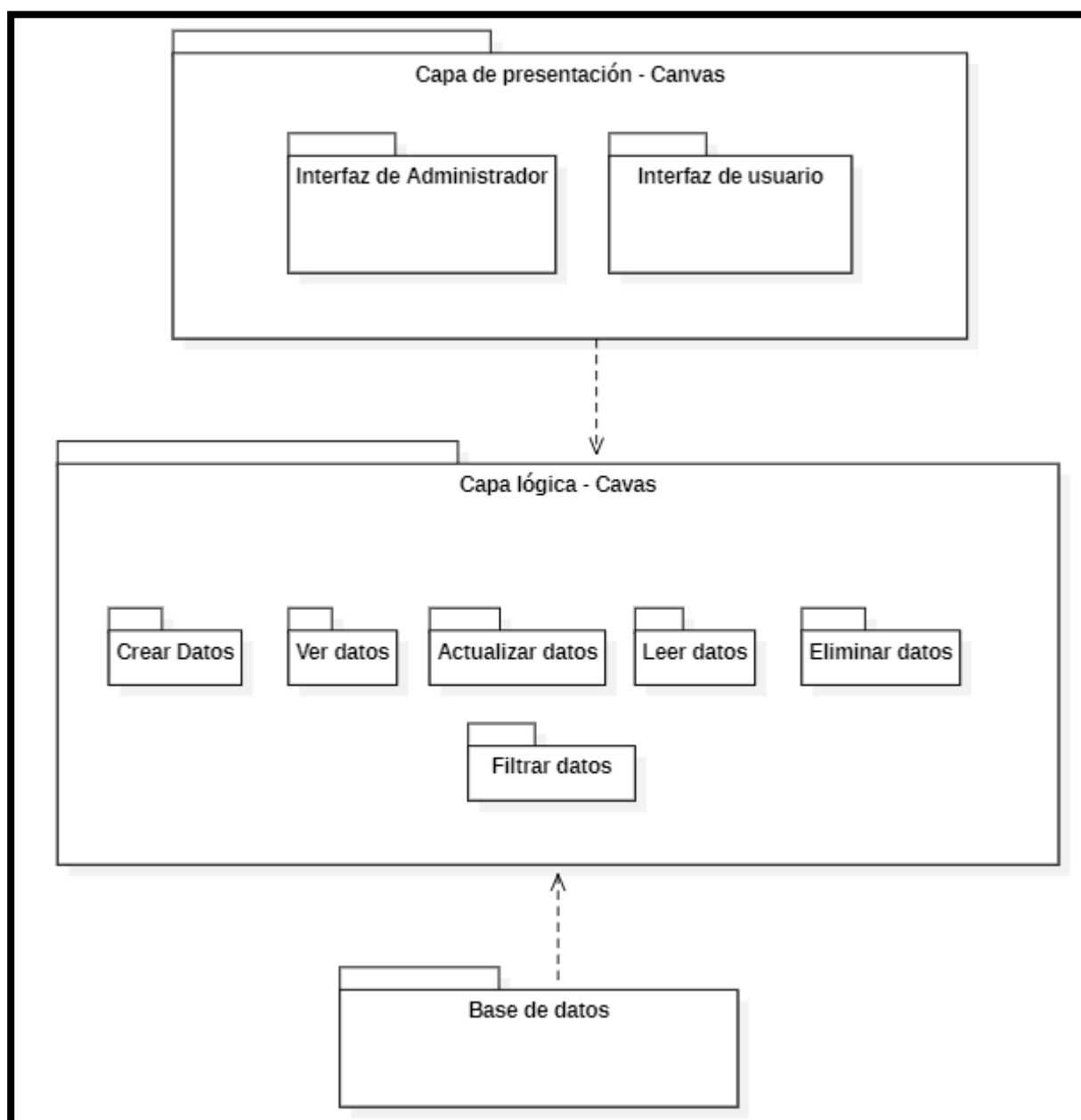
Figura 12. Diagrama de clases 1. Autoría propia, (2022).





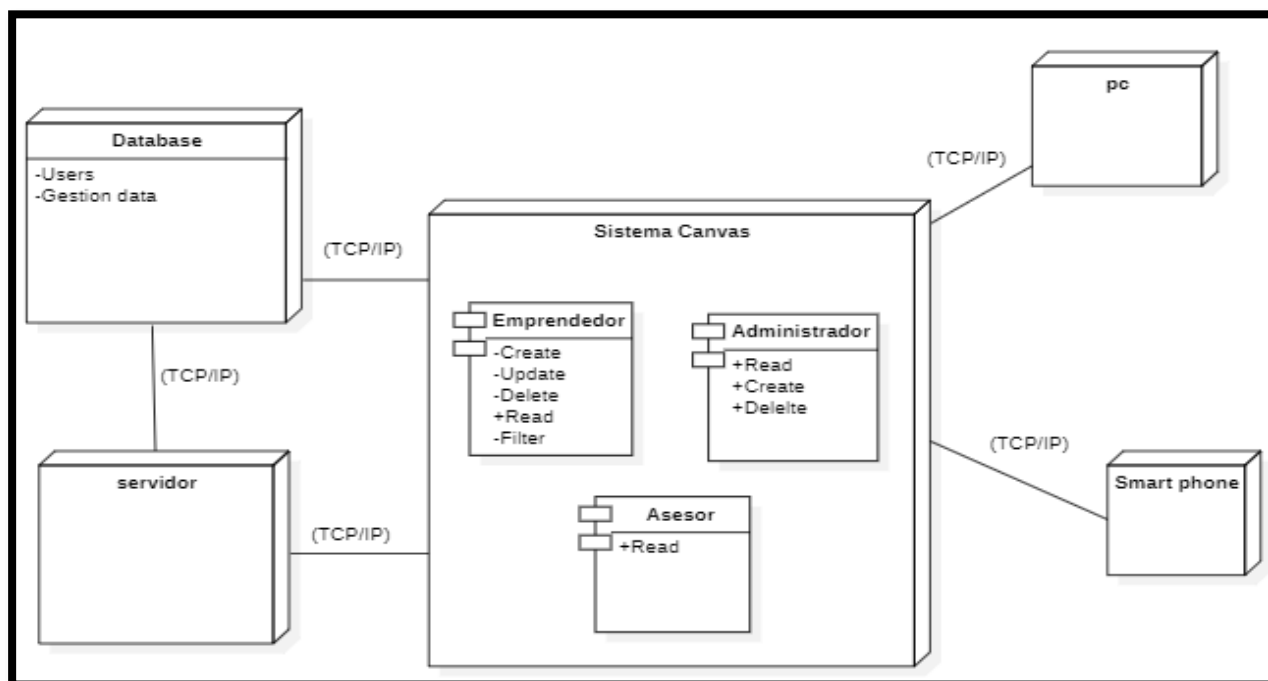
**Figura 14.** Diagrama de clases 3. Autoría propia, (2022).

### 5.2.2 Diagrama de paquetes.



**Figura 15.** Imagen de diagrama de paquetes. Autoría propia, (2022).

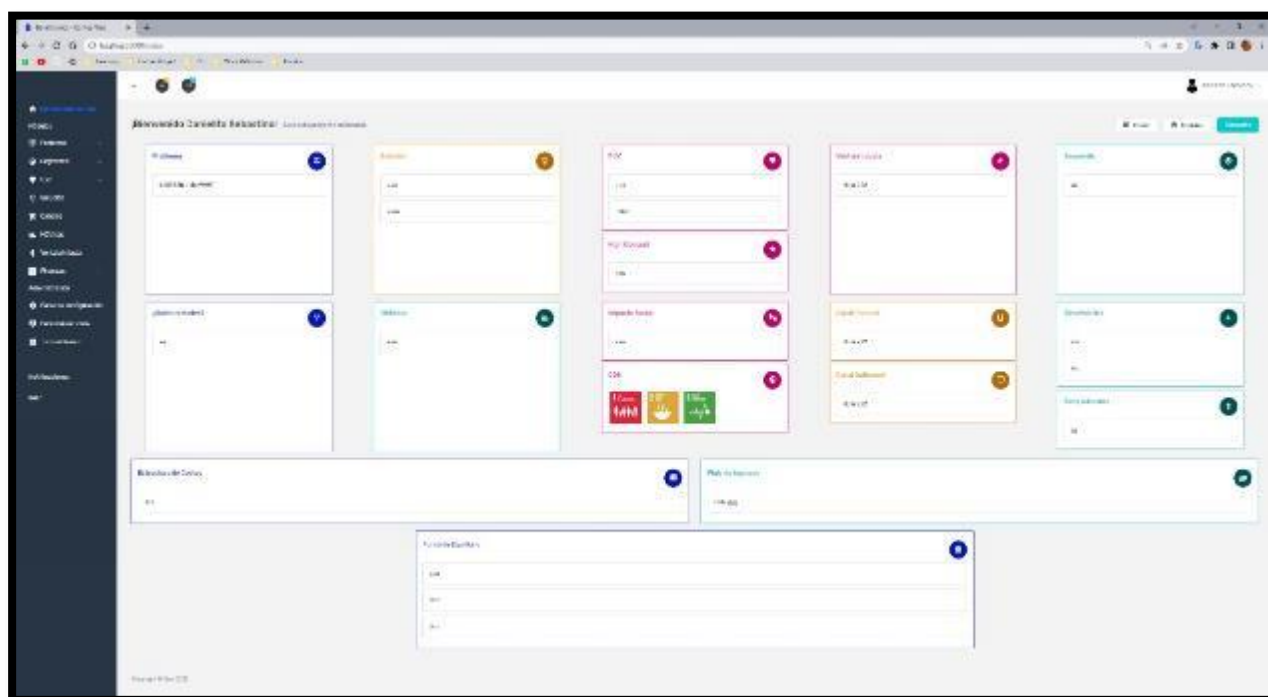
### 5.2.3 Diagramas de despliegue.



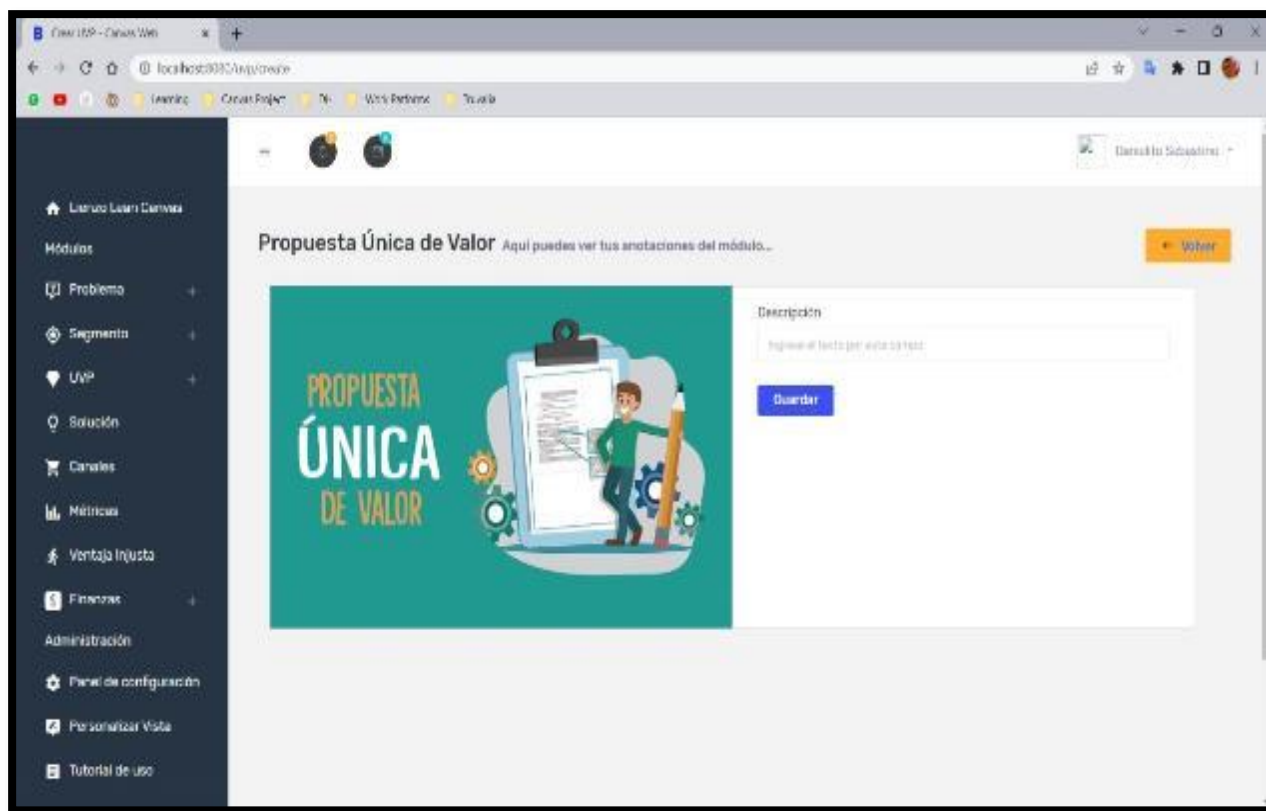
**Figura 16:** Imagen de diagrama de despliegue. Autoría propia, (2022).

## 5.3 Diseño de la Interfaz.

### 5.3.1 Interfaz Gráfica de Usuario.



**Figura 17:** Imagen interfaz gráfica 1. Autoría propia, (2022).



**Figura 18:** Imagen interfaz gráfica 2. Autoría propia, (2022).



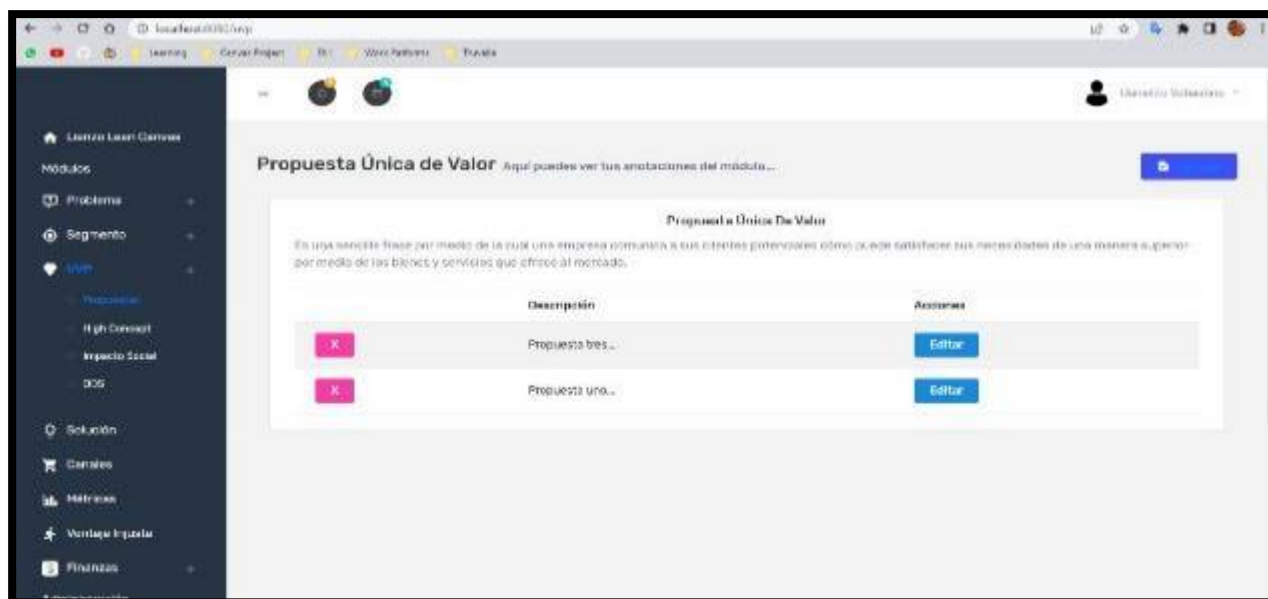


Figura 19: Imagen interfaz gráfica 3. Autoría propia, (2022).

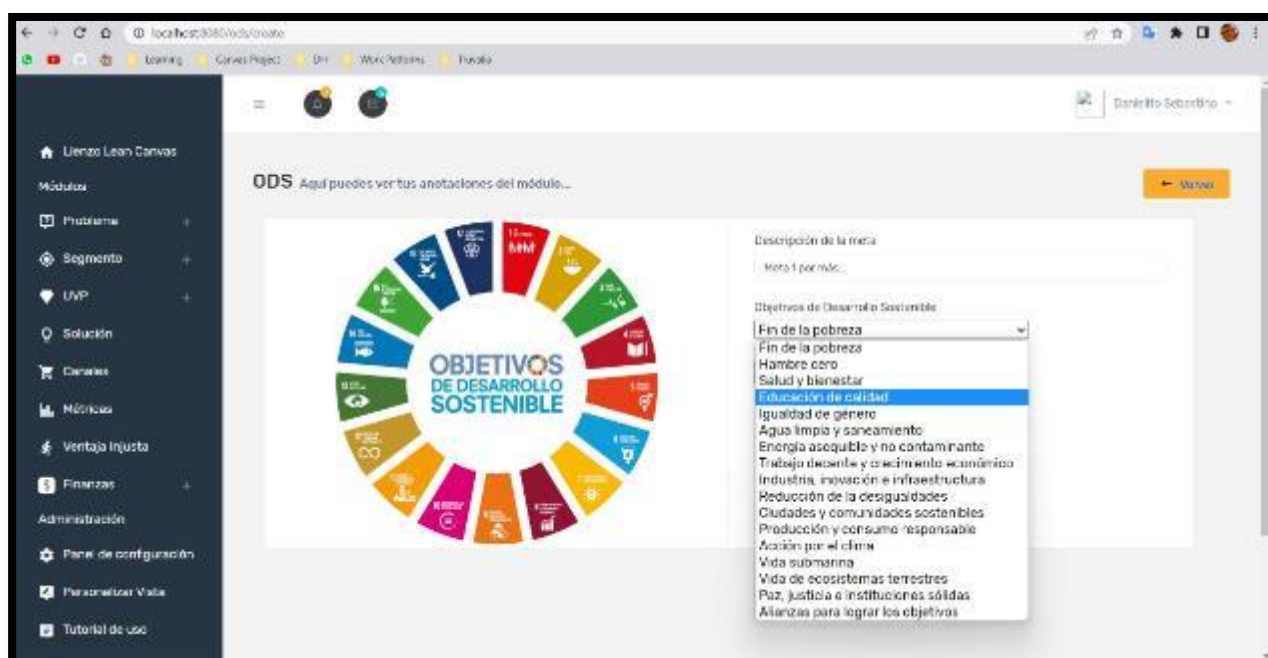
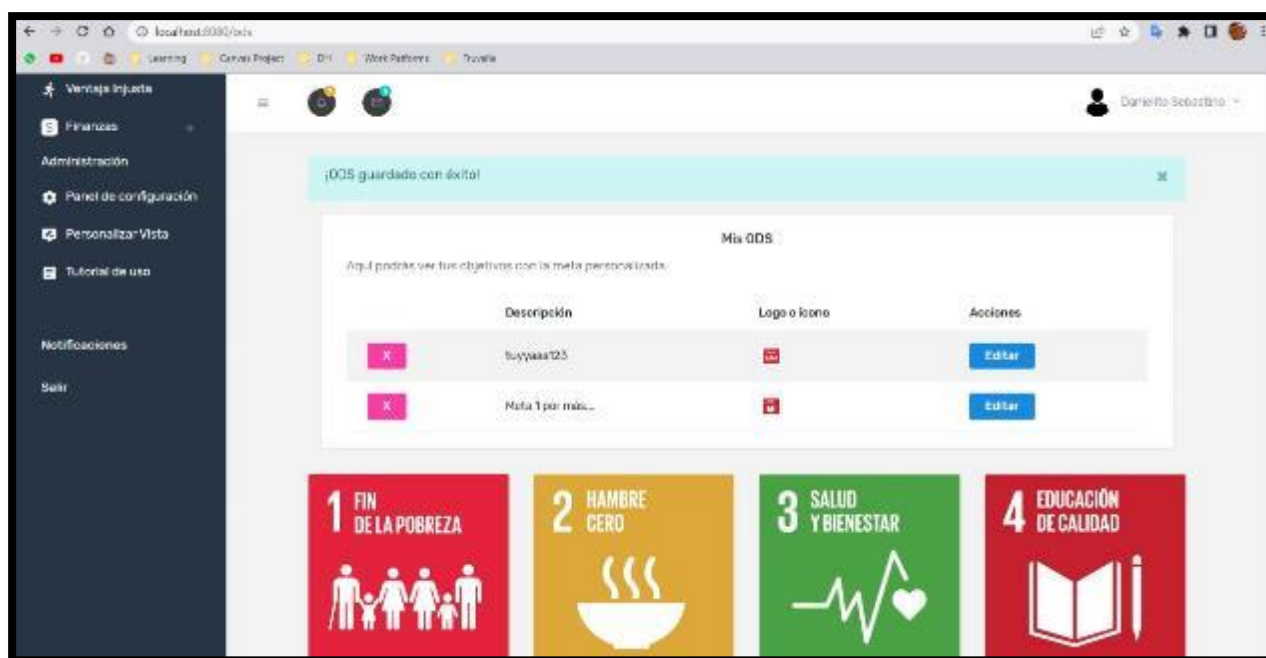
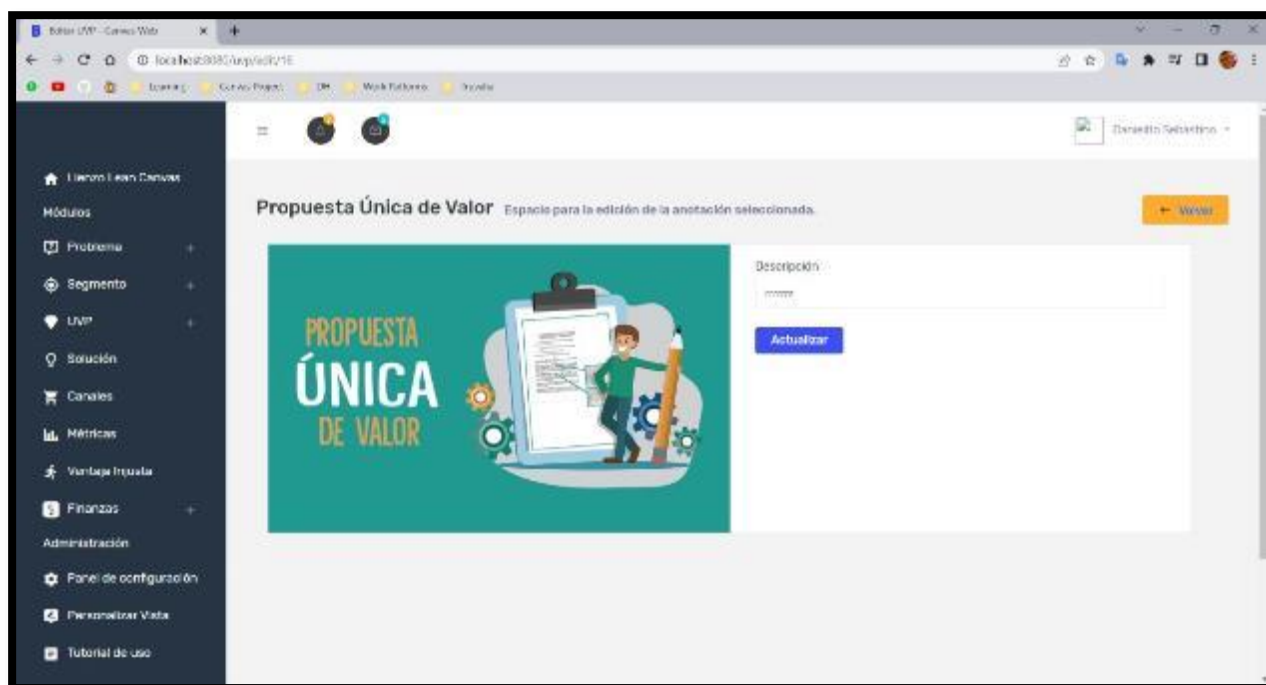


Figura 20. Imagen interfaz gráfica 4. Autoría propia, (2022).



**Figura 21.** Imagen interfaz gráfica 5. Autoría propia, (2022).

### 5.3.2 Interfaces de Entrada.



**Figura 22.** Imagen interfaz de entrada 1. Autoría propia, (2022).

### 5.3.3 Interfaces de Salida.

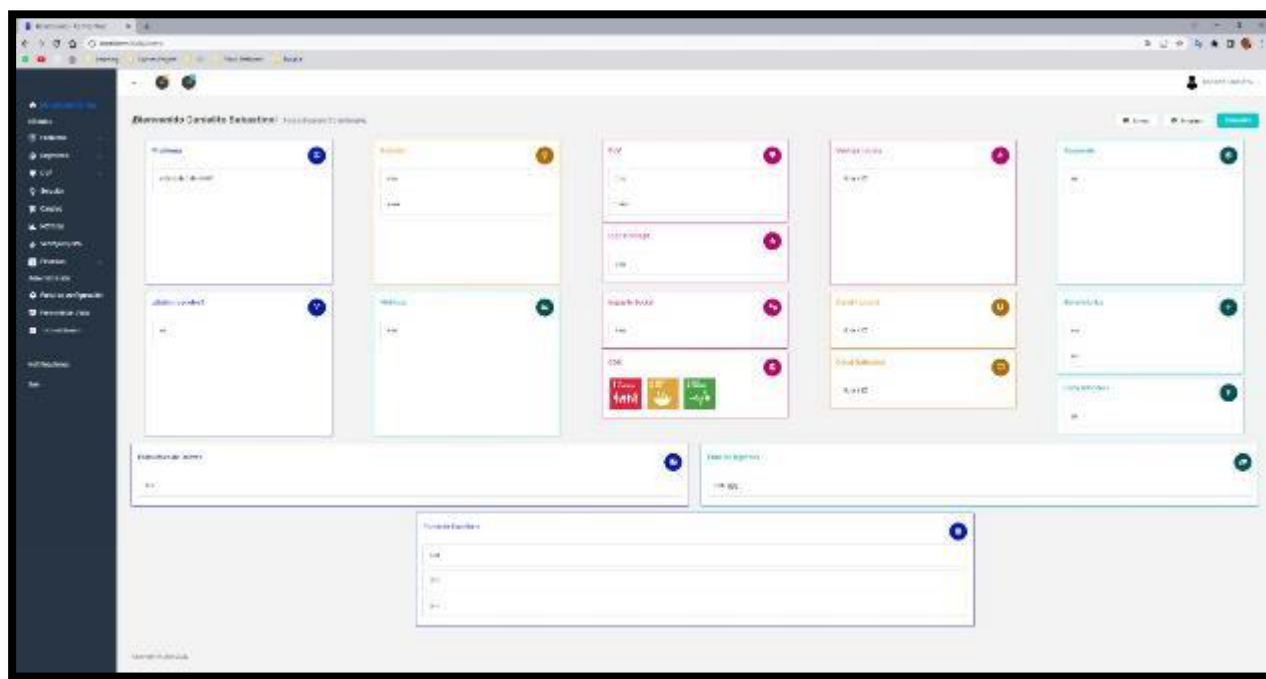


Figura 23: Imagen interfaz de salida 1. Autoría propia, (2022).

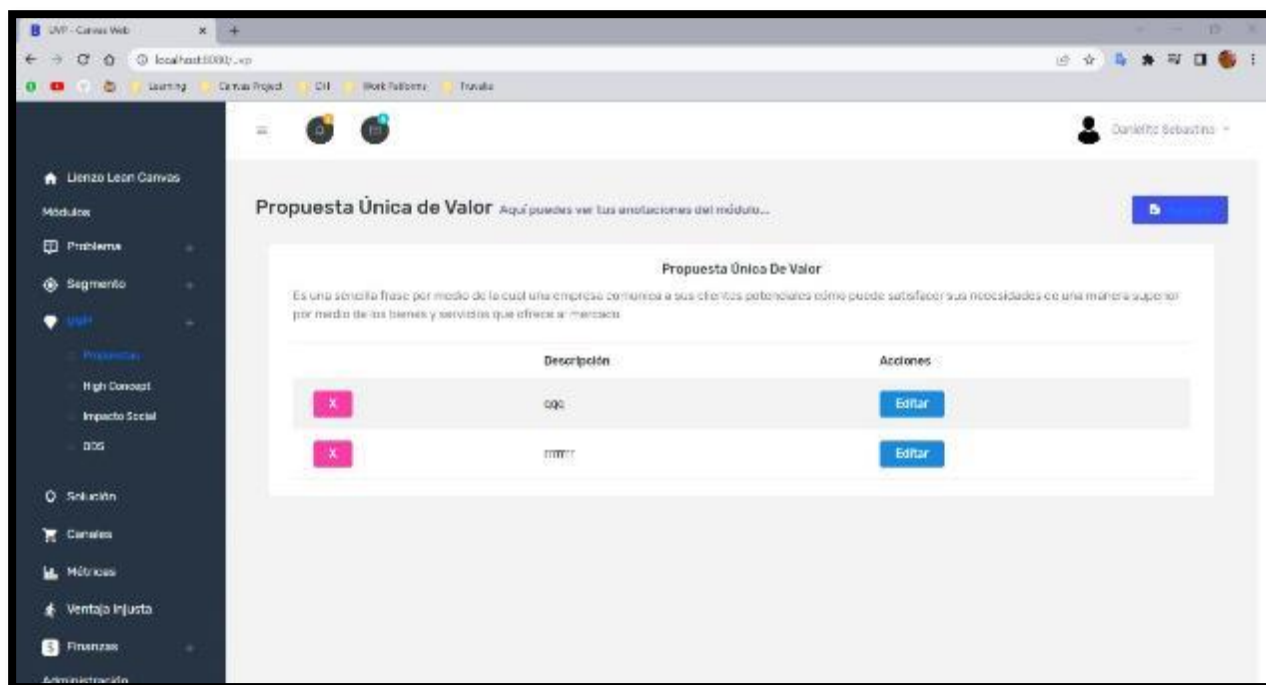


Figura 24. Imagen interfaz de salida 2. Autoría propia, (2022).

## 6. Implementación

### 6.1 Plataformas de desarrollo.

#### Frontend.

Para el desarrollo de la interfaz gráfica se optó por implementar:

**HTML.** Con el cual se creó una estructura de diseño básica en este caso lo llamaremos el esqueleto del sistema.

**CSS.** Para darle formato a la estructura del lienzo, los textos, tamaño de los módulos y el menú de la aplicación.

#### Funcional y lógica.

**Thymeleaf.** Se utilizó para la elaboración de la capa de visión de la aplicación basada en MVC (modelo vista controlador).

#### Frameworks de diseño

**Bootstrap.** Es el marco de desarrollo para trabajar el diseño de la aplicación el cual utiliza HTML, CSS y JavaScript.

#### Backend.

Para esta parte del desarrollo, se implementó las siguientes tecnologías:

**Java-Spring boot - JPA.** Como bien nos dicen en JavaTpoint, (2021): Es una especificación de Java para administrar datos *relacionales* en aplicaciones Java. Nos permite acceder y persistir datos entre el objeto/clase de Java y la base de datos relacional.

Dicho en otras palabras, sirve para interactuar con la base de datos por medio de objetos convirtiéndolos en instrucciones, para así manejar el mapeo de las rutas.

**Hibernate.** Apoyados también en esta herramienta para un mejor desarrollo de la aplicación se da una breve explicación. Según Equipo Geek (2019):

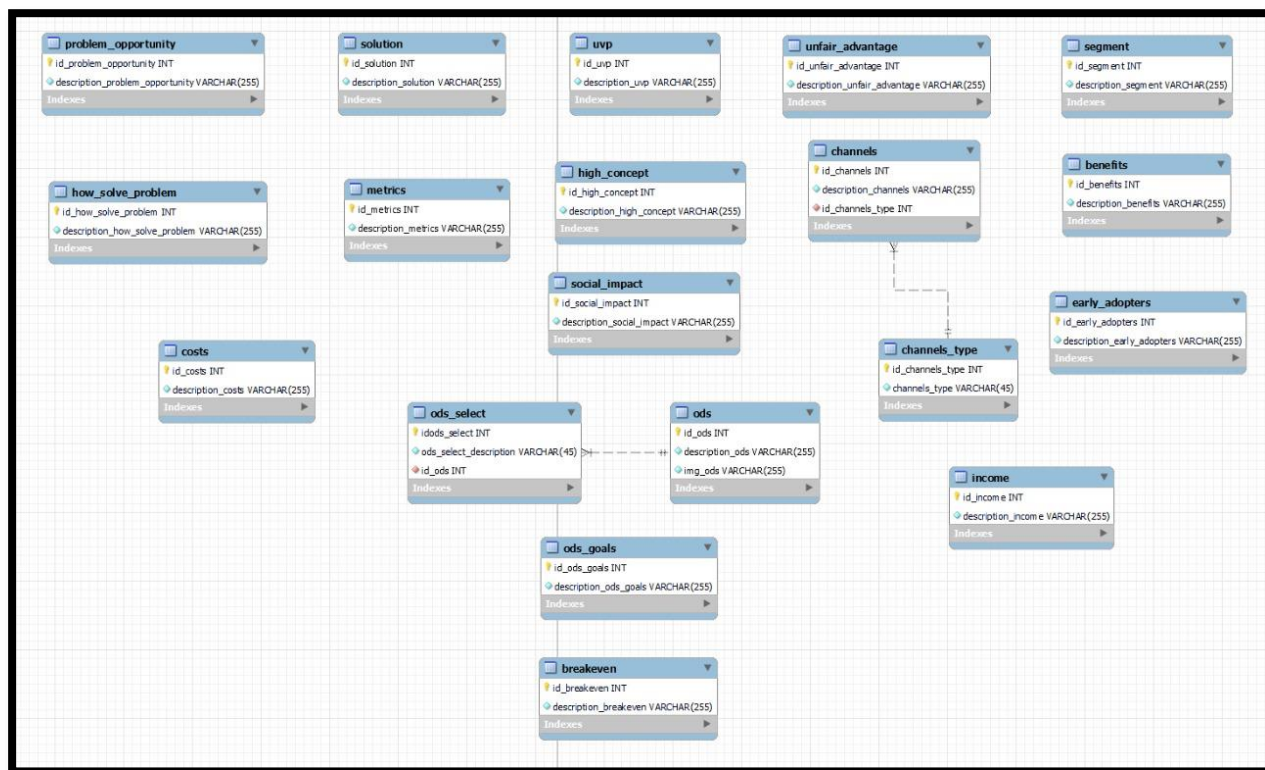
Es una herramienta de mapeo objeto-relacional (ORM) bajo licencia GNU LGPL para Java, que facilita el mapeo de atributos en una base de datos tradicional, y el modelo de objetos de una aplicación mediante archivos declarativos o anotaciones en los beans de las entidades que permiten establecer estas relaciones.

### 6.2 Base de datos.

Para el almacenamiento de la información, se incorporó el software de gestión de bases de datos MySQL Workbench, el cual tiene una gran compatibilidad con las tecnologías antes

mencionadas, además es de código abierto, es fácil de usar y la seguridad es robusta para almacenar datos en ella.

En esta herramienta se llevó a cabo el diseño del modelo entidad-relación, el cual se había trabajado con el aplicativo de Visual Paradigm, el cual nos permitió crear todas las tablas con sus respectivas relaciones, para este caso, se creó teniendo en cuenta los siguientes parámetros:



**Figura 25.** Autoría propia, (2022).

### 6.3 Infraestructura de hardware y redes.

#### Especificaciones de hardware

- Disco duro de 500 Gb.
- Memoria RAM min, 2 Gb.

#### Especificaciones de software

##### Sistema operativo.

Windows 7 todas sus versiones y en adelante.

Linux.

##### Navegadores.

Chrome, Opera, Mozilla.

## 7. Pruebas del software

### 7.1 Pruebas del software.

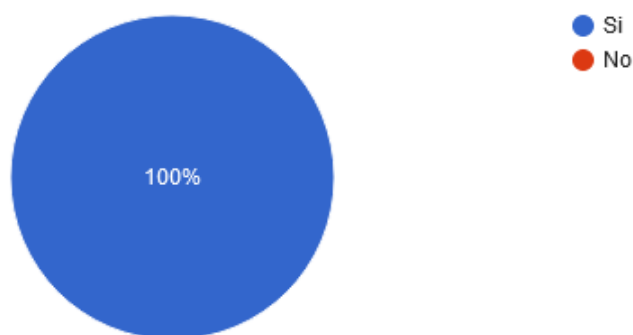
Para validación de las pruebas se procedió a la validación de la funcionalidad del software por medio de formularios obteniendo los siguientes resultados:

#### Pruebas heurísticas

##### Visibilidad del estado del sistema

¿Cada parte de la interfaz comienza con un título o encabezamiento que describa el contenido de la pantalla?

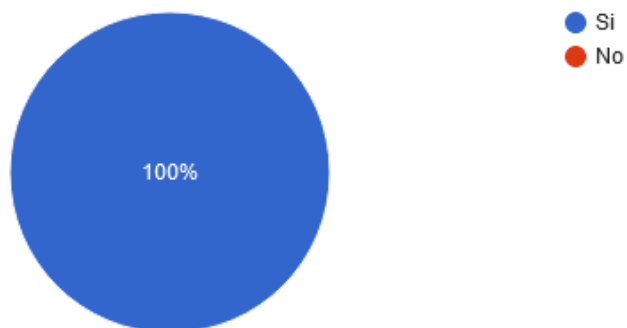
9 responses



**Figura 26.** Imagen pruebas heurísticas 1. Autoría propia, (2022).

¿El esquema de diseño de los iconos y su estética es consistente en todo el sistema?

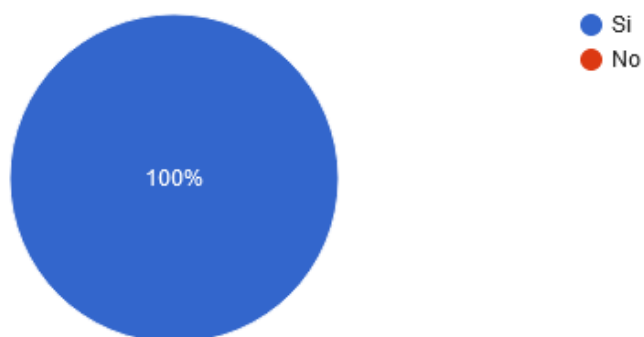
9 responses



**Figura 27.** Imagen pruebas heurísticas 2. Autoría propia, (2022).

Cuándo se selecciona un icono particular rodeado por otros iconos, ¿se distingue el icono seleccionado claramente?

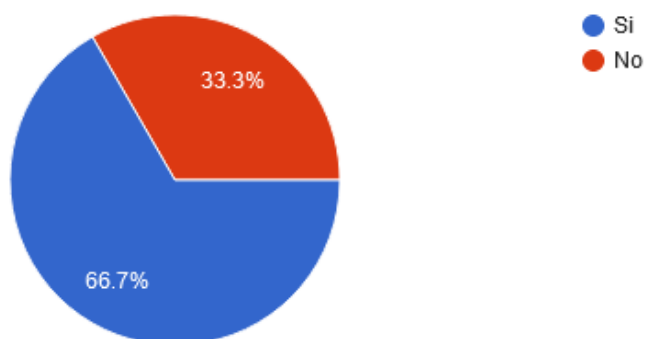
9 responses



**Figura 28.** Imagen pruebas heurísticas 3. Autoría propia, (2022).

El menú de instrucciones, puntos de entrada de datos y mensajes de error ¿aparecen en el mismo lugar de la pantalla?

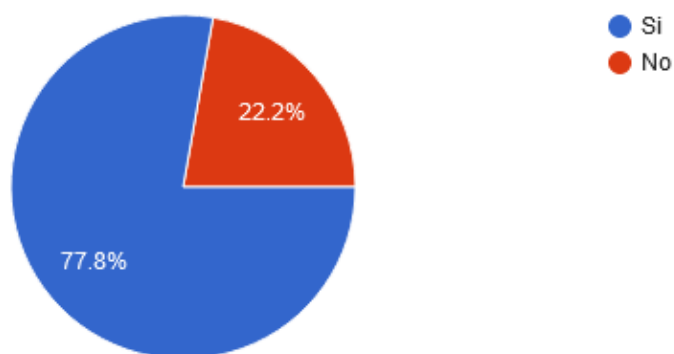
9 responses



**Figura 29.** Imagen pruebas heurísticas 4. Autoría propia, (2022).

En pantallas múltiples para entrada de datos ¿cada página esta etiquetada para mostrar su relación con las otras?

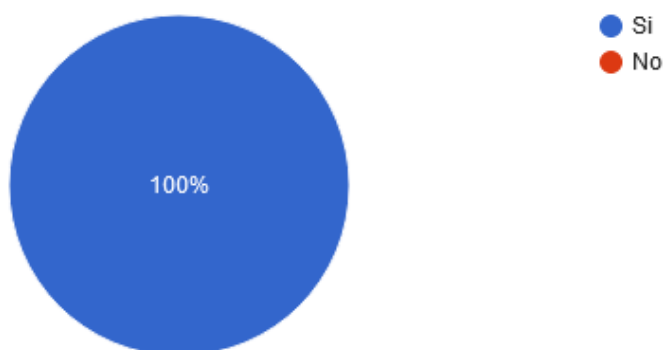
9 responses



**Figura 30.** Autoría propia, Imagen pruebas heurísticas 5. (2022).

Si se utilizan ventanas emergentes (pop-up) para mostrar mensajes de error, ¿permiten estas ventanas que el usuario visualice el error en la interfaz cuando se despliegan?

9 responses

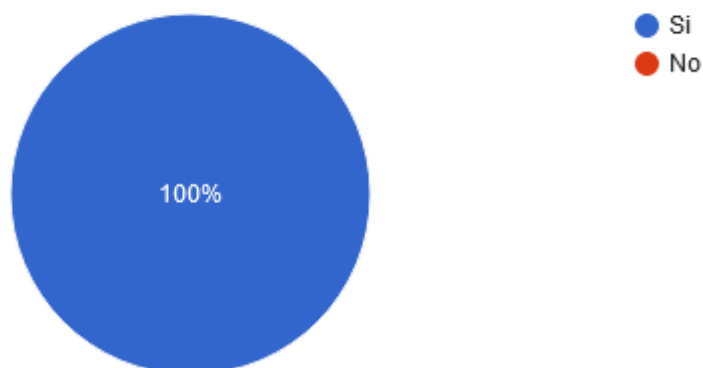


**Figura 31.** Imagen pruebas heurísticas 6. Autoría propia, (2022).



¿Hay algún tipo de "feedback" para cada acción u operación?

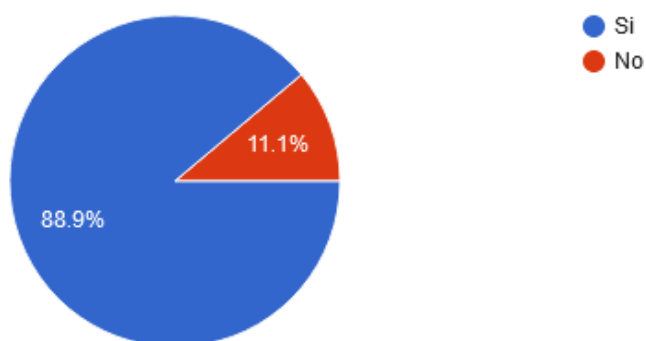
9 responses



**Figura 32.** Imagen pruebas heurísticas 7. Autoría propia, (2022).

Luego que el usuario completa una acción o un grupo de acciones, ¿el "feedback" del sistema indica que el siguiente grupo de acciones puede comenzarse?

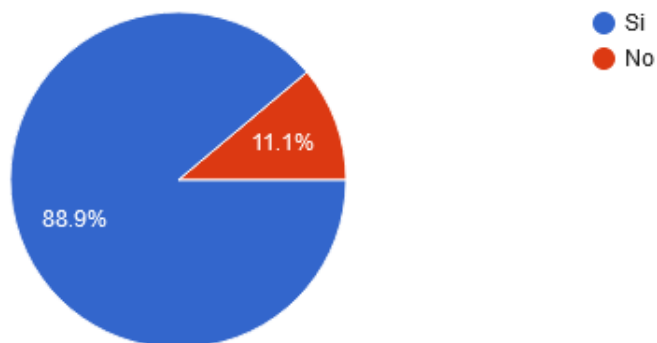
9 responses



**Figura 33.** Imagen pruebas heurísticas 8. Autoría propia, (2022).

¿El sistema provee algún tipo de “feedback” visual en menús o cajas de diálogo que indiquen las opciones que pueden seleccionarse?

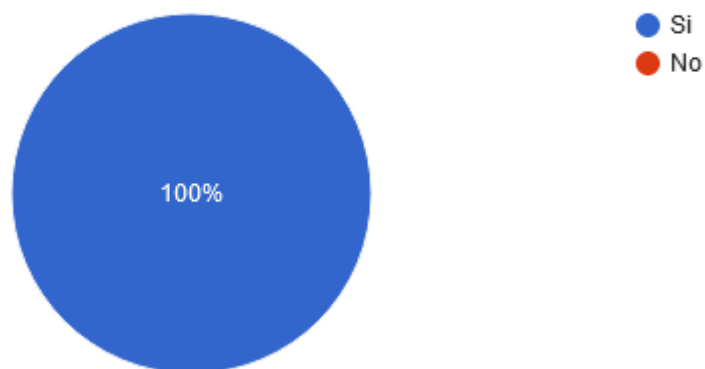
9 responses



**Figura 34.** Imagen pruebas heurísticas 9. Autoría propia, (2022).

¿Los tiempos de respuestas son apropiados para cada tarea?

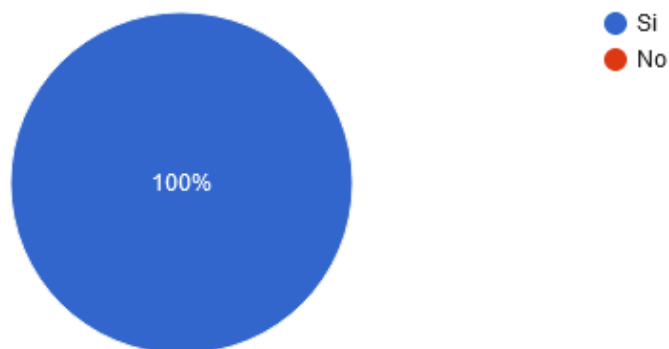
9 responses



**Figura 35.** Imagen pruebas heurísticas 10. Autoría propia, (2022).

¿Los tiempos de respuesta del sistema son adecuados al proceso cognitivo del usuario?

9 responses

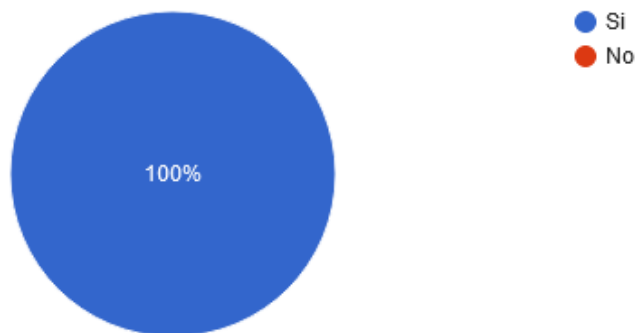


**Figura 36.** Imagen pruebas heurísticas 11. Autoría propia, (2022).

#### Lenguaje de Los Usuarios

Dado un determinado usuario, una determinada lista de ítems y variables para realizar tareas. ¿las opciones en los segmentos (nombres de los ítems) están ordenadas en la manera más lógica para el usuario?

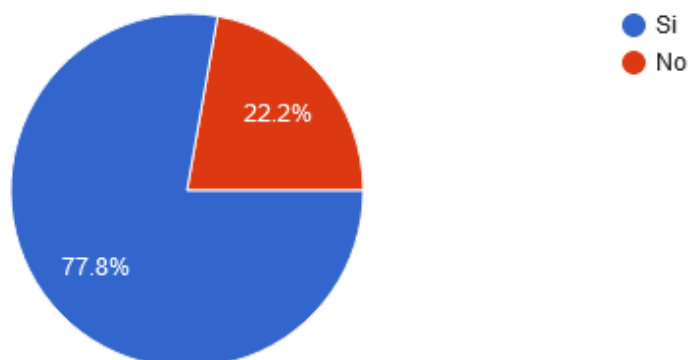
9 responses



**Figura 37.** Imagen pruebas heurísticas 12. Autoría propia, (2022).

Si existe una secuencia natural para la selección de elementos del canvas ¿esta implementada esa secuencia?

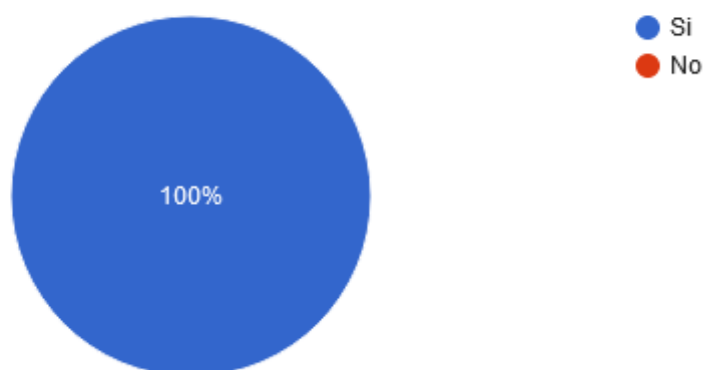
9 responses



**Figura 38.** Imagen pruebas heurísticas 13. Autoría propia, (2022).

¿Los iconos son concretos y familiares para el usuario?

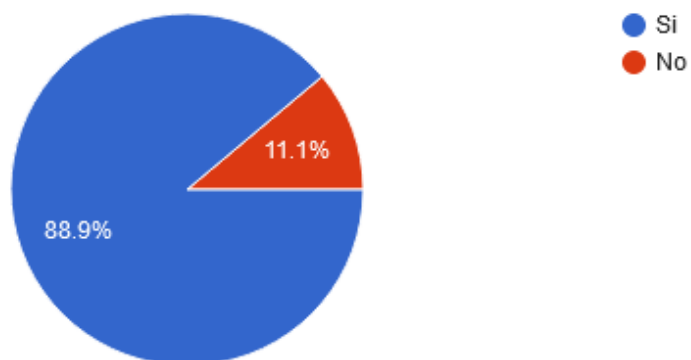
9 responses



**Figura 39.** Imagen pruebas heurísticas 14. Autoría propia, (2022).

Los campos relacionados e interdependientes, ¿aparecen en la misma pantalla?

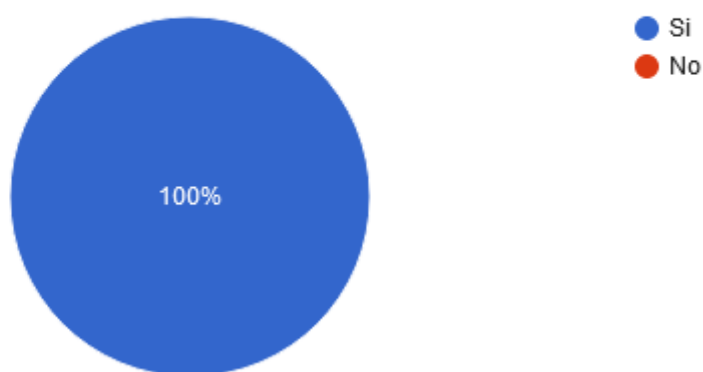
9 responses



**Figura 40.** Imagen pruebas heurísticas 15. Autoría propia, (2022).

¿Los colores seleccionados corresponden a cada segmento?

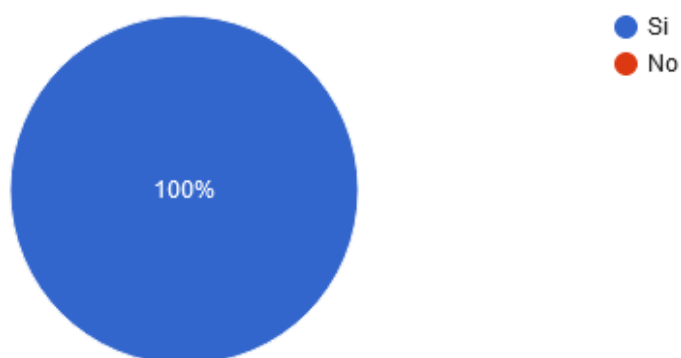
9 responses



**Figura 41.** Imagen pruebas heurísticas 16. Autoría propia, (2022).

Los botones incluyen un mensaje con palabras consistentes con esa acción?

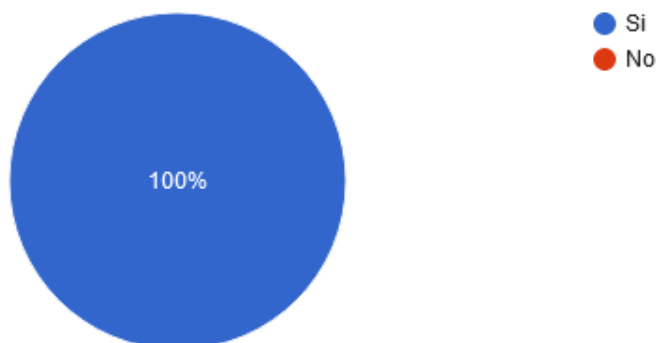
9 responses



**Figura 42.** Imagen pruebas heurísticas 17. Autoría propia, (2022).

Cuando la pantalla incluye preguntas que debe ser respondidas, ¿el lenguaje de esas preguntas es simple y claro?

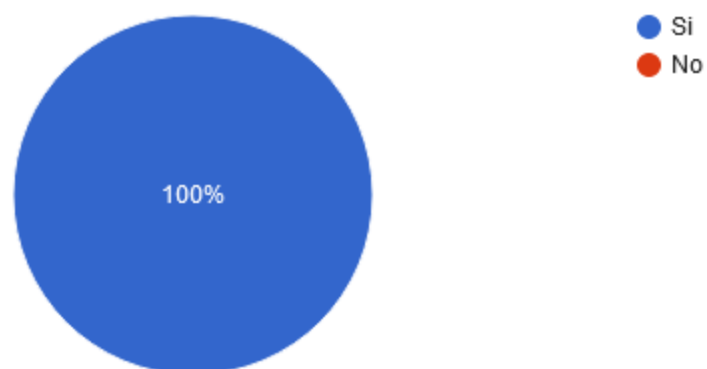
9 responses



**Figura 43.** Imagen pruebas heurísticas 18. Autoría propia, (2022).

¿El orden de los segmentos es el adecuado e intuitivo para el usuario?

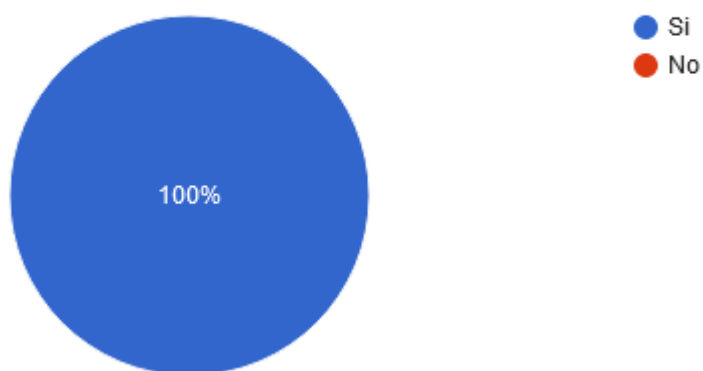
9 responses



**Figura 44.** Imagen pruebas heurísticas 19. Autoría propia, (2022).

¿En el diseño del canvas se maneja un mismo estilo gramatical y ordenado?

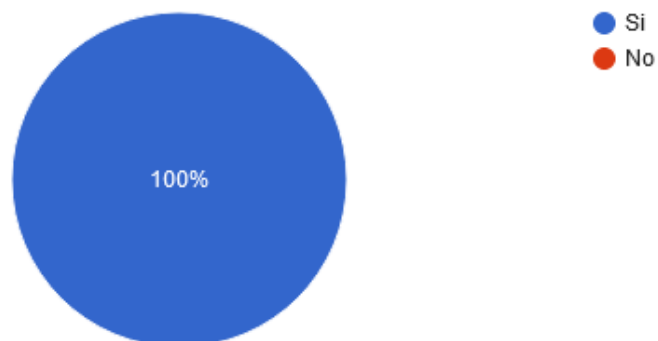
9 responses



**Figura 45.** Imagen pruebas heurísticas 20. Autoría propia, (2022).

El lenguaje empleado en la plataforma ¿utiliza la jerga de los usuarios evitando el uso vocablos computacionales específicos?

9 responses

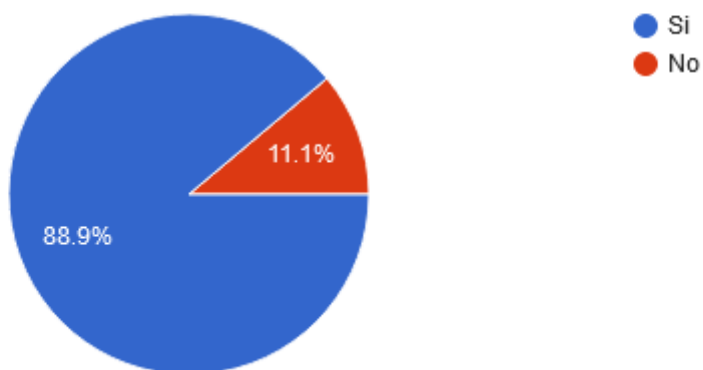


**Figura 46.** Imagen pruebas heurísticas 21. Autoría propia, (2022).

### Control y Libertad para el Usuario

¿Es fácil reacomodar (reubicar) las ventanas superpuestas en la pantalla?

9 responses

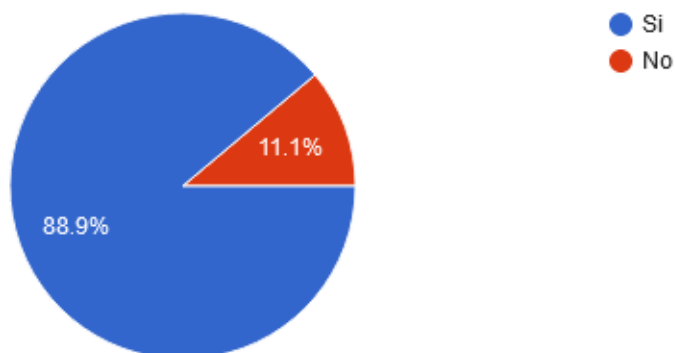


**Figura 47.** Imagen pruebas heurísticas 22. Autoría propia, (2022).



Cuando una tarea efectuada por el usuario se completa, ¿el sistema espera alguna señal del usuario (mensaje) antes de procesar la tarea?

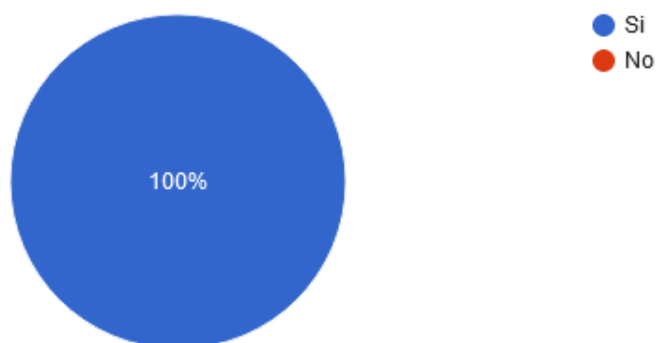
9 responses



**Figura 48.** Imagen pruebas heurísticas 23. Autoría propia, (2022).

¿Existe una función para "deshacer" al nivel de cada acción simple, cada entrada de datos y cada grupo de acciones completadas?

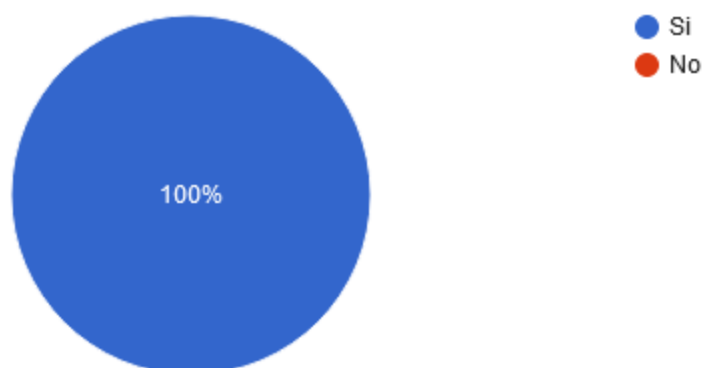
9 responses



**Figura 49.** Imagen pruebas heurísticas 24. Autoría propia, (2022).

¿La edición de caracteres está permitida en los comandos?

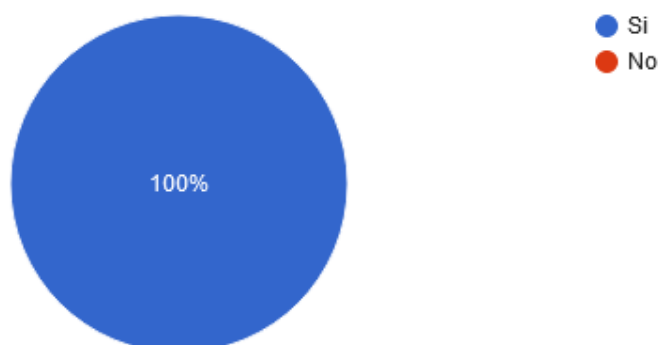
9 responses



**Figura 50.** Imagen pruebas heurísticas 25. Autoría propia, (2022).

¿Los usuarios pueden reducir el tiempo de entrada de datos copiando y modificando datos existentes?

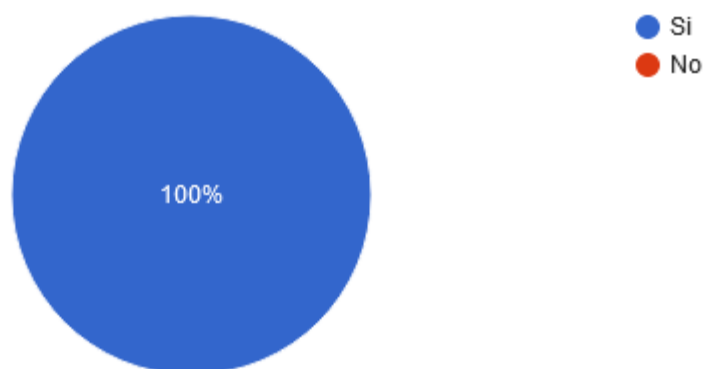
9 responses



**Figura 51.** Imagen pruebas heurísticas 26. Autoría propia, (2022).

¿La edición de caracteres está permitida en los campos de entrada de datos?

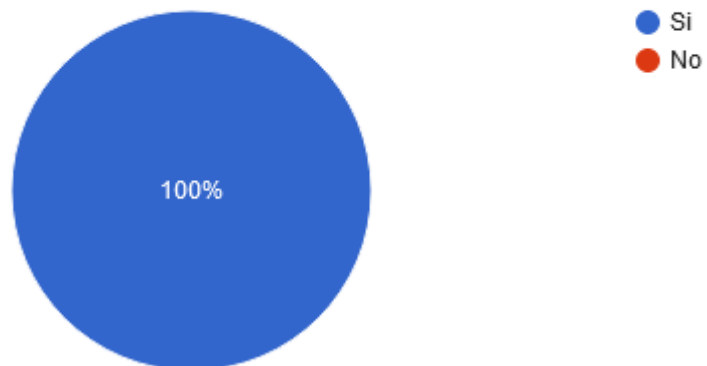
9 responses



**Figura 52.** Imagen pruebas heurísticas 27. Autoría propia, (2022).

¿Pueden los usuarios seleccionar varios ODS (Objetivos y metas de desarrollo sostenible)?

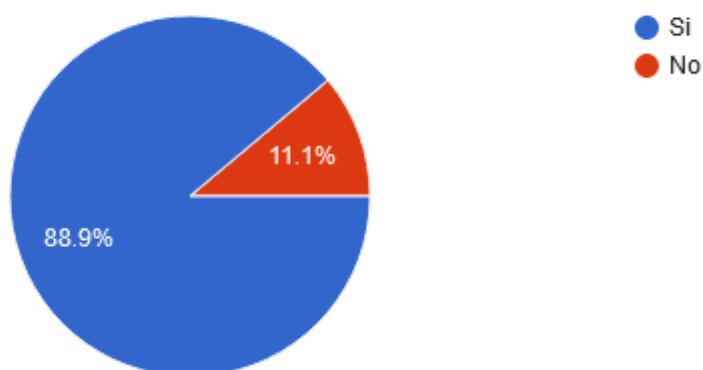
9 responses



**Figura 53.** Imagen pruebas heurísticas 28. Autoría propia, (2022).

¿Puede el usuario eliminar algo que no desee dentro del canvas?

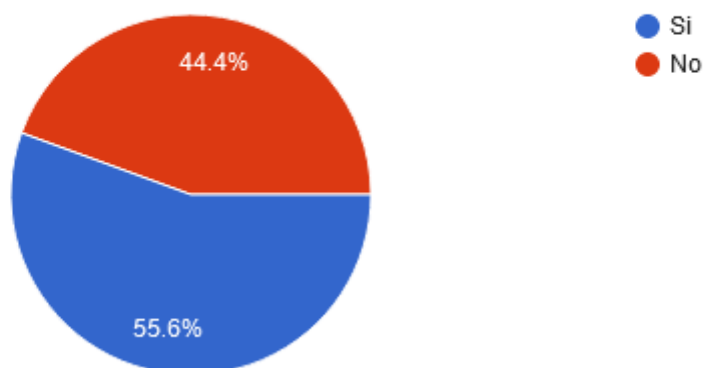
9 responses



**Figura 54.** Imagen pruebas heurísticas 29. Autoría propia, (2022).

¿Al ingresar datos en algún segmento el usuario tiene un limite para hacerlo?

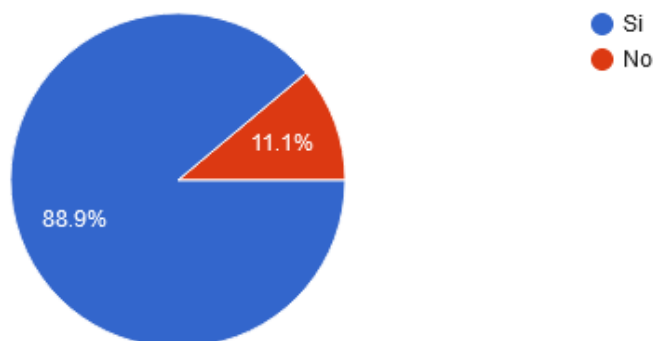
9 responses



**Figura 55.** Imagen pruebas heurísticas 30. Autoría propia, (2022).

Si el sistema utiliza dispositivos de tipo puntero, ¿los usuarios tienen la opción tanto de hacer “clic” en una lista de ítems como de utilizar atajo usando el teclado?

9 responses



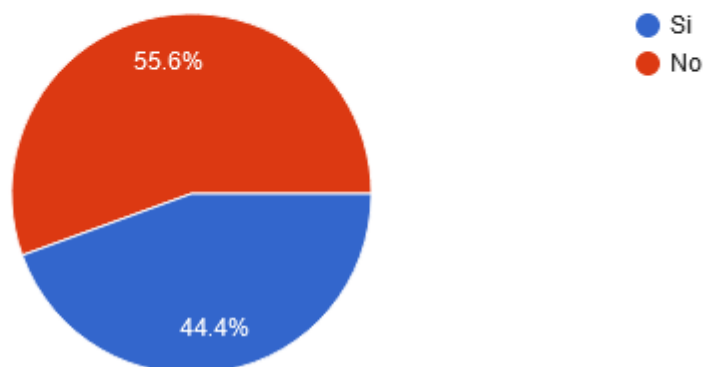
**Figura 56.** Imagen pruebas heurísticas 31. Autoría propia, (2022).

### Pruebas de validación y verificación

#### VALIDACIÓN

¿El sistema permite la interacción del aplicativo a los usuarios?

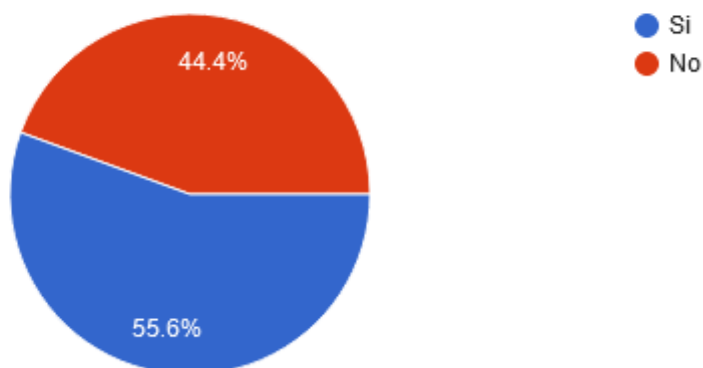
9 responses



**Figura 57.** Imagen pruebas de validación y verificación 1. Autoría propia, (2022).

¿El sistema cuenta con un ingreso de inicio de sesión?

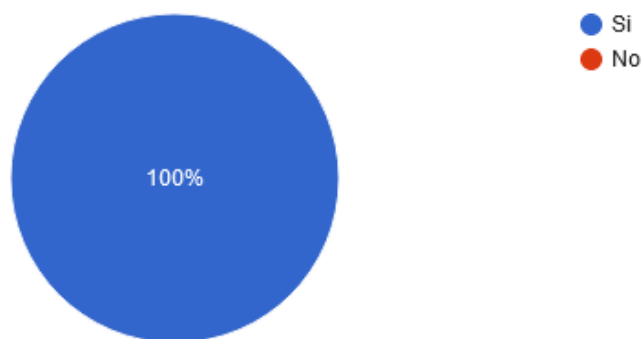
9 responses



**Figura 58.** Imagen pruebas de validación y verificación 2. Autoría propia, (2022).

¿El sistema permite el envío de datos referentes a la idea de negocio para iniciar en el modelo canvas?

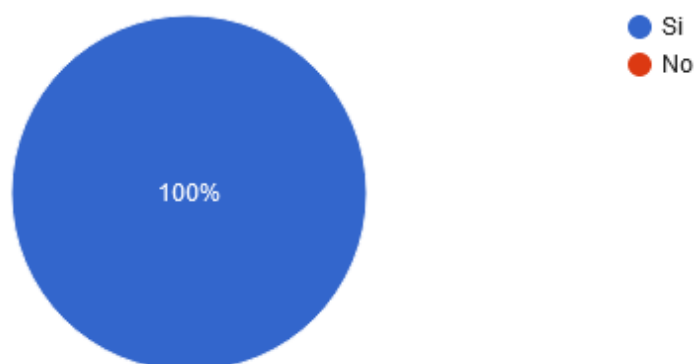
9 responses



**Figura 59.** Imagen pruebas de validación y verificación 3. Autoría propia, (2022).

¿El sistema permite insertar información en cada modulo donde es necesario?

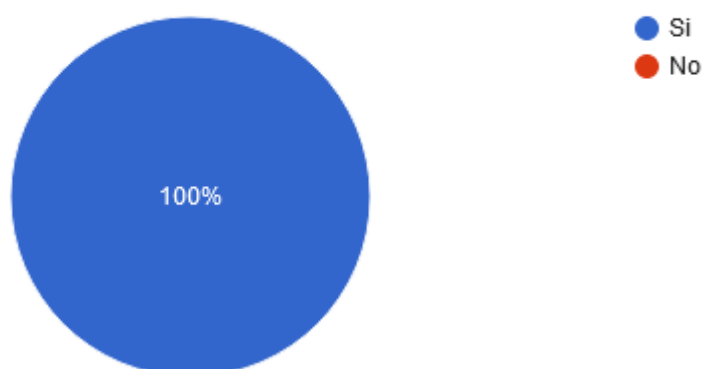
9 responses



**Figura 60.** Imagen pruebas de validación y verificación 4. Autoría propia, (2022).

¿El sistema permite actualizar la información de cada modulo?

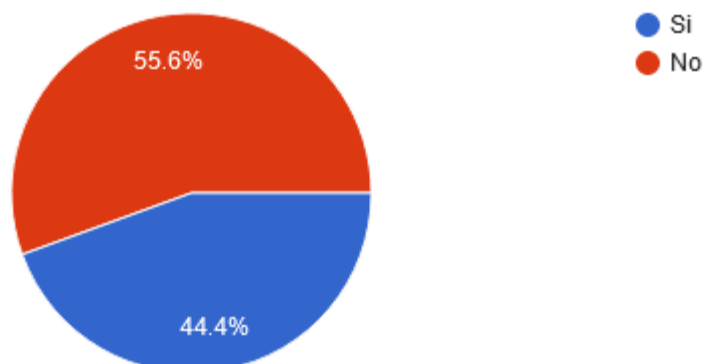
9 responses



**Figura 61.** Imagen pruebas de validación y verificación 5. Autoría propia, (2022).

¿El sistema permite al usuario llenar solo en un orden específico los datos del canvas?

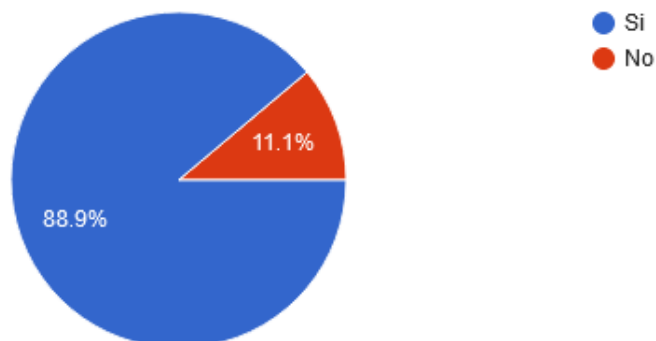
9 responses



**Figura 62.** Imagen pruebas de validación y verificación 6. Autoría propia, (2022).

¿El sistema muestra los datos agregados en la ventana sobre el canvas al guardar la información?

9 responses



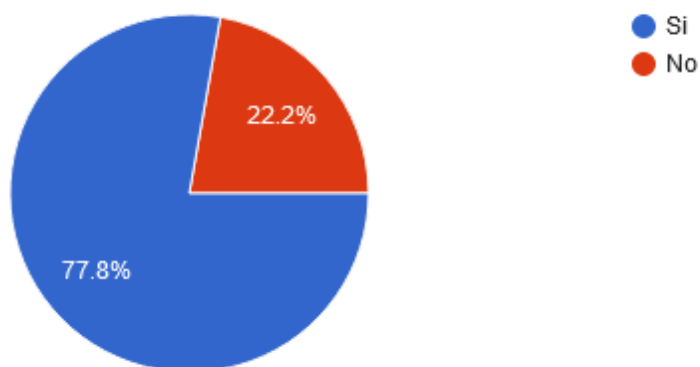
**Figura 63.** Imagen pruebas de validación y verificación 7. Autoría propia, (2022).



## Validación-legalidad

¿El sistema controla el acceso y lo permite solamente a usuarios autorizados.?

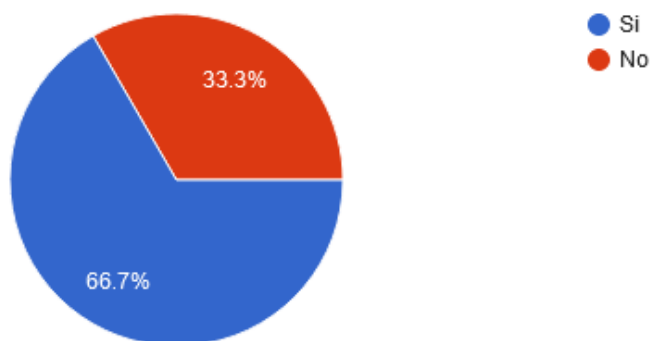
9 responses



**Figura 64.** Imagen pruebas de validación y verificación 8. Autoría propia, (2022).

¿El sistema permite la seguridad de los datos al ser agregados y cambiar de página poniendo un control de acceso si no se es usuario del mismo?

9 responses



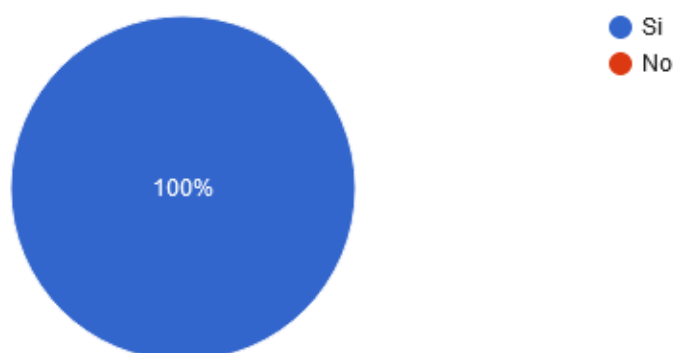
**Figura 65.** Imagen pruebas de validación y verificación 9. Autoría propia, (2022).

## VERIFICACIÓN

### EFICIENCIA

¿El sistema al almacenar los datos lo hace en un tiempo estimado de 4 segundos?

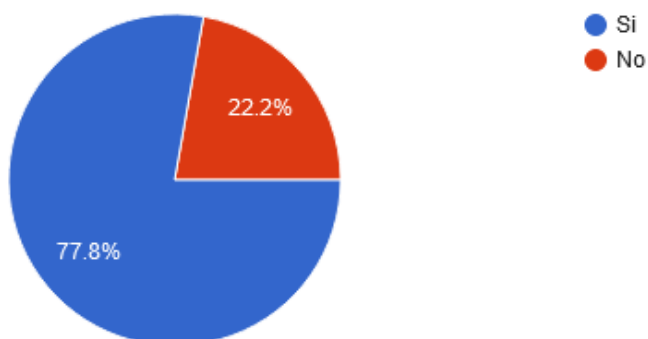
9 responses



**Figura 66.** Imagen pruebas de validación y verificación 10. Autoría propia, (2022).

¿El sistema es capaz de operar adecuadamente con múltiples usuarios con sesiones concurrentes.?

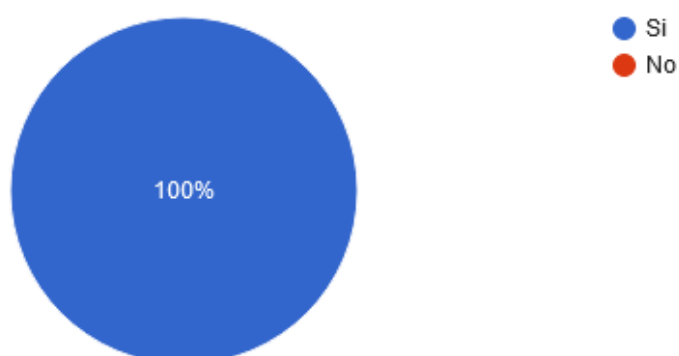
9 responses



**Figura 67.** Imagen pruebas de validación y verificación 11. Autoría propia, (2022).

Los datos modificados en la base de datos deben ser actualizados para todos los usuarios que acceden en menos de 2 segundos. ¿Es verídico?

9 responses



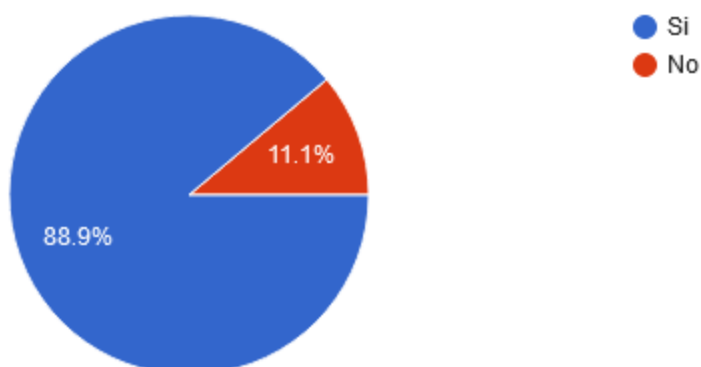
**Figura 68.** Imagen pruebas de validación y verificación 12. Autoría propia, (2022).

## SEGURIDAD Y LÓGICA DE DATOS

---

¿Los permisos de acceso al sistema pueden ser cambiados solamente por el administrador de acceso a datos.?

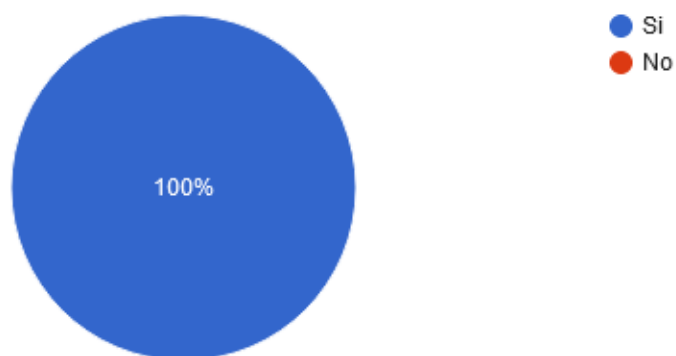
9 responses



**Figura 69.** Imagen pruebas de validación y verificación 13. Autoría propia, (2022).

¿El sistema se desarrolló aplicando patrones y recomendaciones de programación que incrementen la seguridad de datos.?

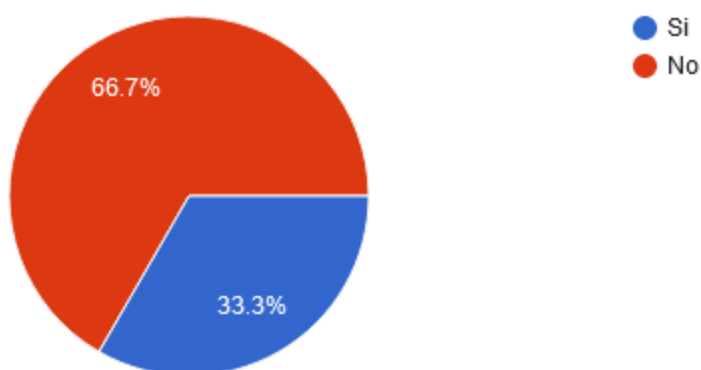
9 responses



**Figura 70.** Imagen pruebas de validación y verificación 14. Autoría propia, (2022).

¿El sistema se inicia nuevamente luego de 5 minutos de inactividad?

9 responses

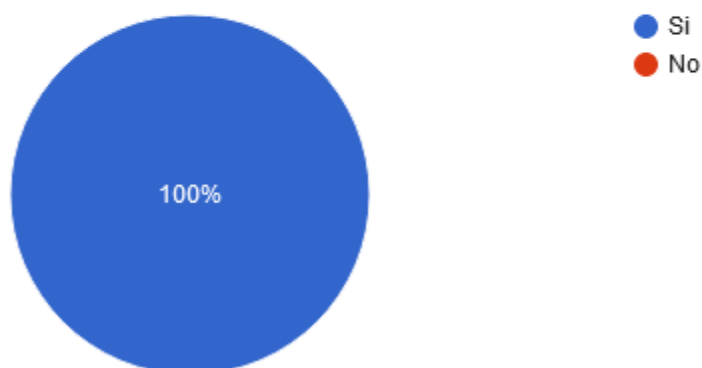


**Figura 71.** Imagen pruebas de validación y verificación 15. Autoría propia, (2022).

## USABILIDAD

¿El tiempo de aprendizaje del sistema por un usuario es menor a una hora?

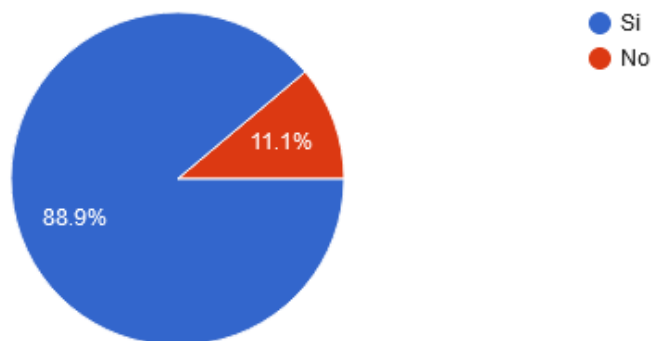
9 responses



**Figura 72.** Imagen pruebas de validación y verificación 16. Autoría propia, (2022).

¿El sistema mantiene un orden intuitivo que permite disminuir el margen de error por equivocaciones?

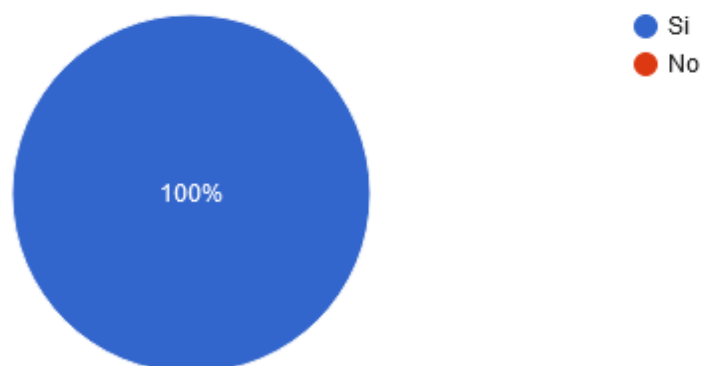
9 responses



**Figura 73.** Imagen pruebas de validación y verificación 17. Autoría propia, (2022).

¿El sistema cuenta con manuales de usuario estructurados adecuadamente?

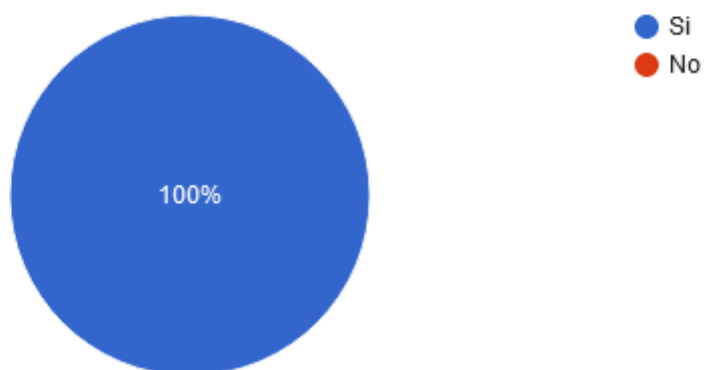
9 responses



**Figura 74.** Imagen pruebas de validación y verificación 18. Autoría propia, (2022).

¿El sistema proporciona mensajes de error al usuario con el fin de orientarlo correctamente?

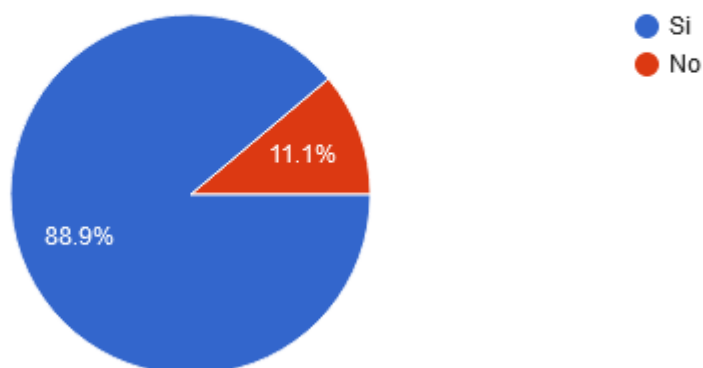
9 responses



**Figura 75.** Imagen pruebas de validación y verificación 19. Autoría propia, (2022).

¿El sistema posee una interfaz gráficas bien formada?

9 responses

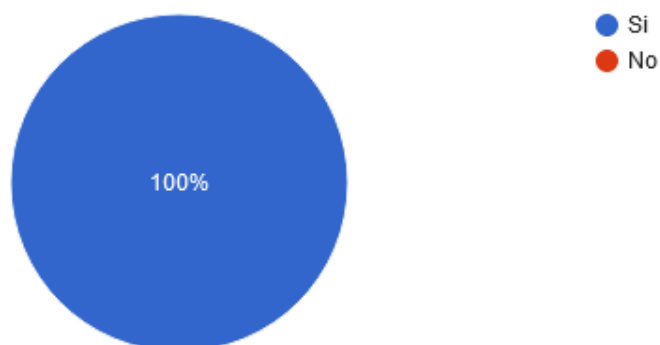


**Figura 76.** Imagen pruebas de validación y verificación 20. Autoría propia, (2022).

#### DEPENDIBILIDAD

El sistema debe tener una disponibilidad del 99,99% de las veces en que un usuario intente acceder ¿Es verídico?

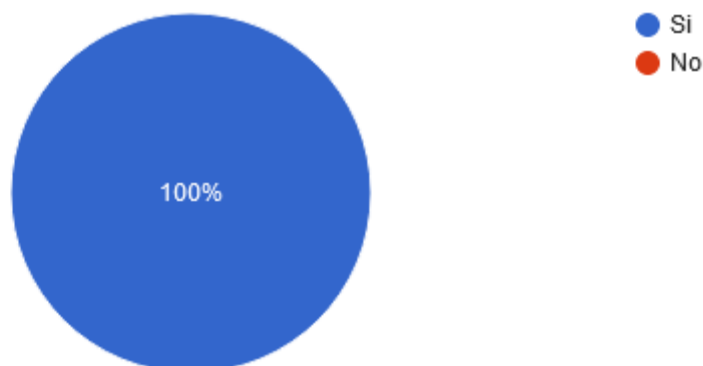
9 responses



**Figura 77.** Imagen pruebas de validación y verificación 21. Autoría propia, (2022).

¿El tiempo para iniciar o reiniciar el sistema es menor a 5 minutos?

9 responses



**Figura 78.** Imagen pruebas de validación y verificación 22. Autoría propia, (2022).

## 7.2 Pruebas de usabilidad.

### Identidad del software

¿Con la información que se ofrece en pantalla, es posible saber a qué institución o empresa corresponde el sitio? ¿Cómo lo sabe?

9 responses

Si, aparece en la pestaña de navegación

Por la pestaña de navegacion

Si , por el contenido corresponde a una institución educativa

Se observa por medio de la pestaña de navegación

Se muestra por medio de la pestaña de la navegación de la página

Es un proyecto independiente ya que no posee un logotipo de alguna institución.

Si, en la parte de arriba se logra visualizar

Por medio del nombre en la pestaña de navegacion.

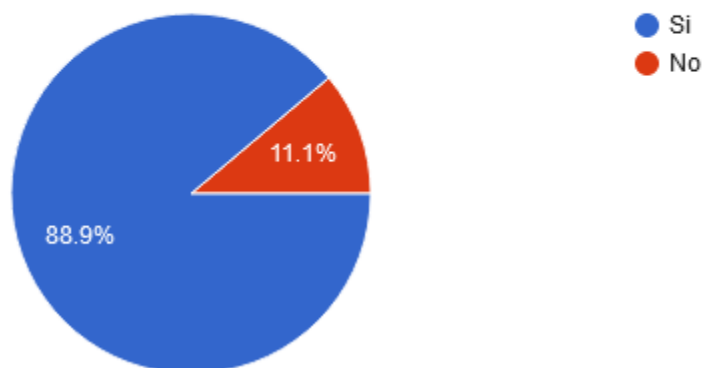
En el menú y ventana de navegación.

**Figura 79.** Imagen pruebas de usabilidad 1. Autoría propia, (2022).



¿Hay algún elemento gráfico o de texto que le haya ayudado a entender más claramente a que institución o empresa pertenece el sitio?

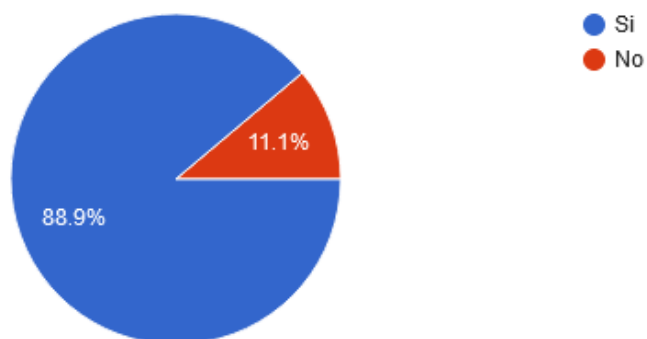
9 responses



**Figura 80.** Imagen pruebas de usabilidad 2. Autoría propia, (2022).

¿Relaciona los colores predominantes en el sitio web con la institución? ¿Relaciona la dirección del sitio web con la institución?

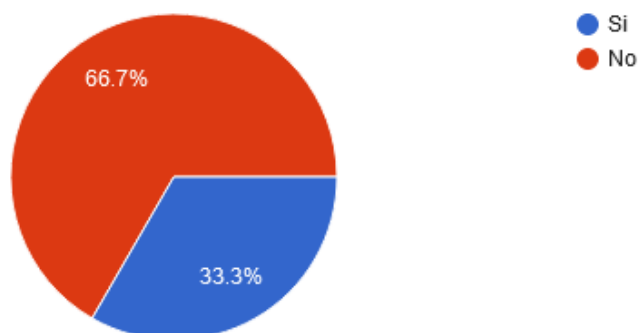
9 responses



**Figura 81.** Imagen pruebas de usabilidad 3. Autoría propia, (2022).

¿De los elementos que muestra esta pantalla, hay algo que usted crea que está fuera de lugar, porque no pertenece a la institución o empresa que usted identifica como propietaria?

9 responses

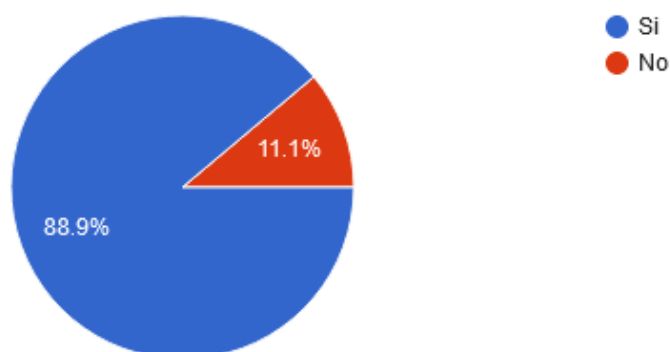


**Figura 82.** Imagen pruebas de usabilidad 4. Autoría propia, (2022).

#### Contenido del software

¿Al ver el sitio, pudo distinguir de una sola mirada cuál es el orden del contenido?

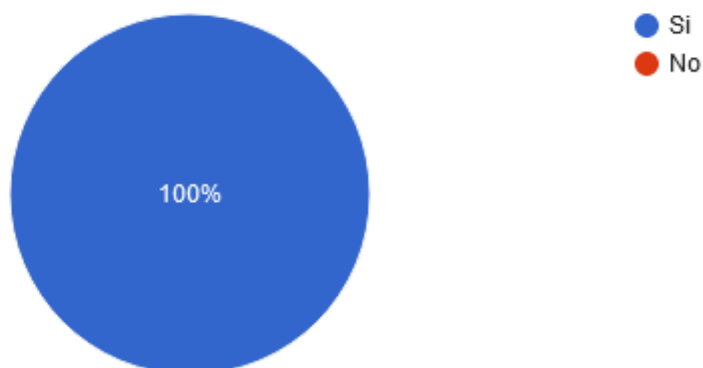
9 responses



**Figura 83.** Imagen pruebas de usabilidad 5. Autoría propia, (2022).

¿Es fácil distinguir los nuevos contenidos que presenta el sitio web?

9 responses

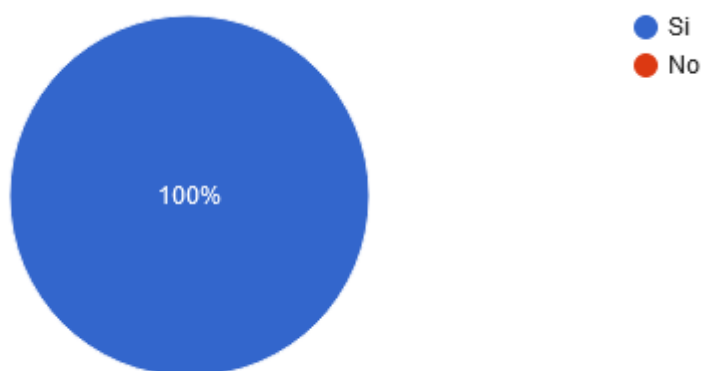


**Figura 84.** Imagen pruebas de usabilidad 6. Autoría propia, (2022).

#### Navegación del software

¿Puede distinguir fácilmente la forma de entrar y salir del modelo Canvas

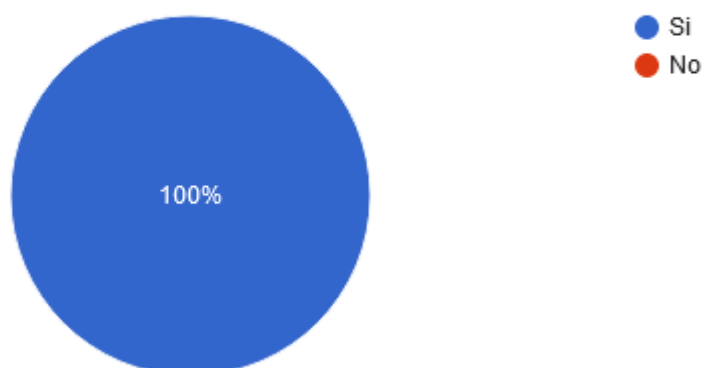
9 responses



**Figura 85.** Imagen pruebas de usabilidad 7. Autoría propia, (2022).

¿Puede saber con detalle el orden de diligenciamiento del Canvas?

9 responses

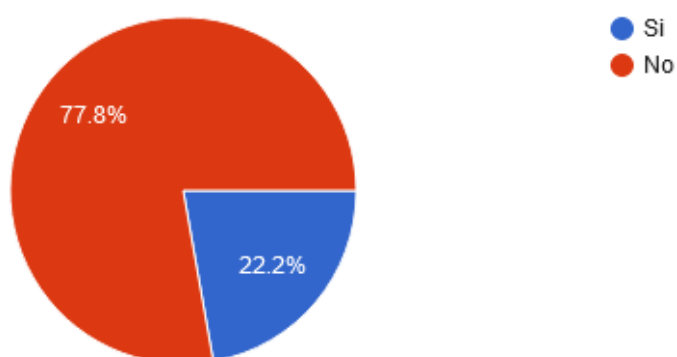


**Figura 86.** Imagen pruebas de usabilidad 8. Autoría propia, (2022).

#### Feed back del software

¿Encuentra alguna forma online y offline de ponerse en contacto con la empresa o institución, para hacer sugerencias o comentarios?

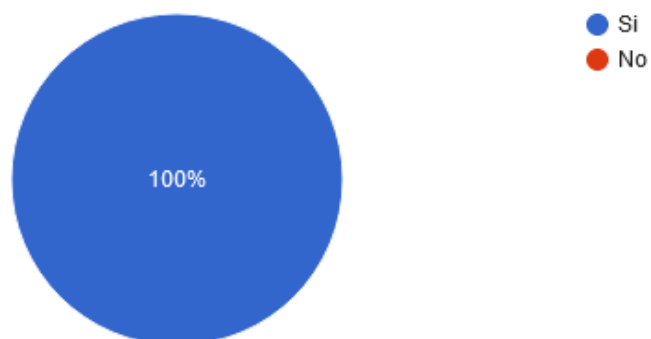
9 responses



**Figura 87.** Imagen pruebas de usabilidad 9. Autoría propia, (2022).

¿Al mandar datos mediante un formulario, la web le avisa si los recibió correctamente o no?

9 responses

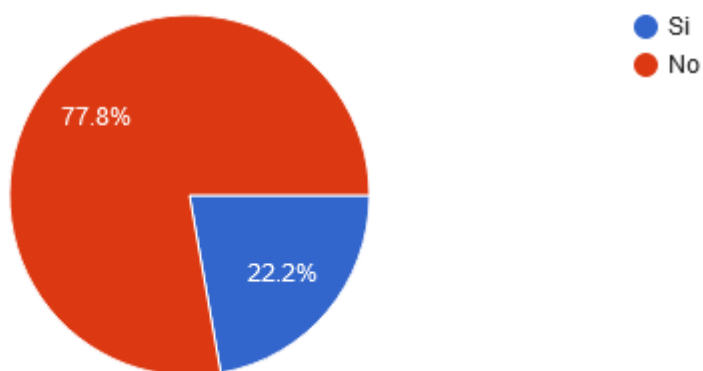


**Figura 88.** Imagen pruebas de usabilidad 10. Autoría propia, (2022).

#### Búsqueda del software

¿Distinguió si en este sitio se ofrecía un buscador?

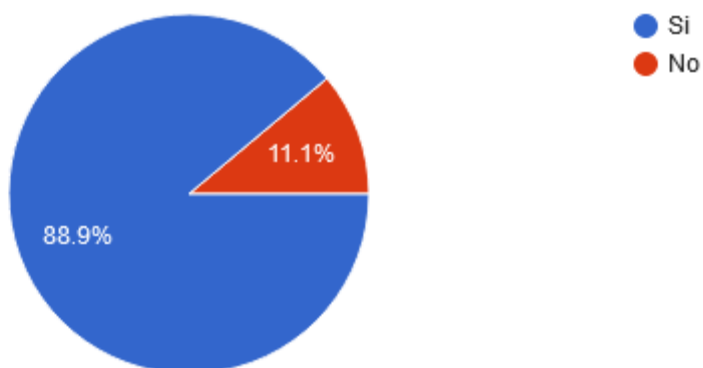
9 responses



**Figura 89.** Imagen pruebas de usabilidad 11. Autoría propia, (2022).

¿Al buscar la plataforma GesTemp en el navegador es fácil encontrarla?

9 responses

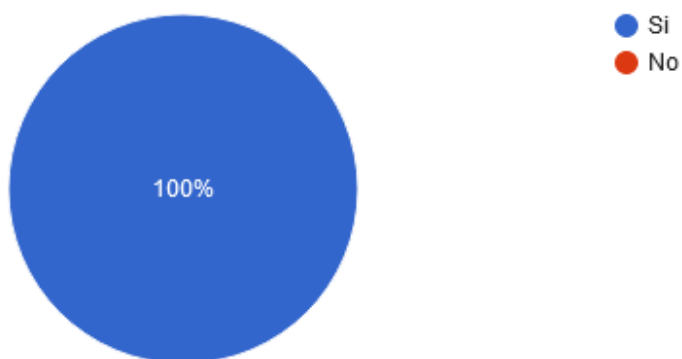


**Figura 90.** Imagen pruebas de usabilidad 12. Autoría propia, (2022).

#### Gráfica web del software

¿Le pareció adecuada la forma en que se muestran los iconos de la plataforma?

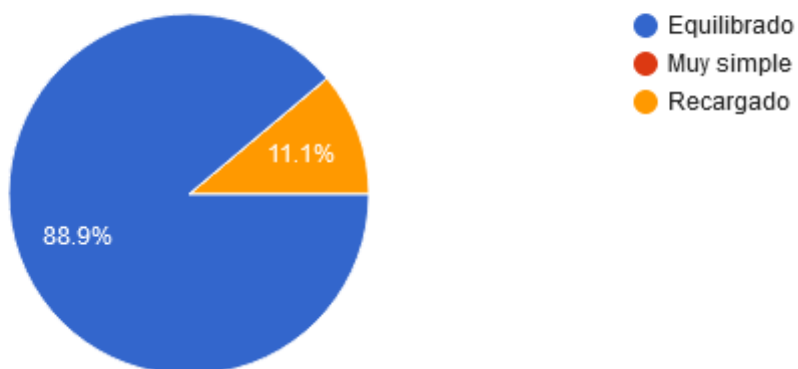
9 responses



**Figura 91.** Imagen pruebas de usabilidad 13. Autoría propia, (2022).

Considera que gráficamente el sitio está:

9 responses

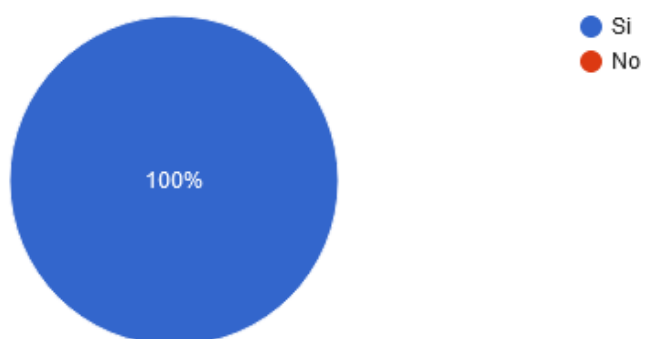


**Figura 92.** Imagen pruebas de usabilidad 14. Autoría propia, (2022).

#### Utilidad del software

¿Tras una primera mirada, le queda claro cuál es el objetivo del sitio? ¿Qué contenidos y servicios ofrece? ¿Los puede enumerar?

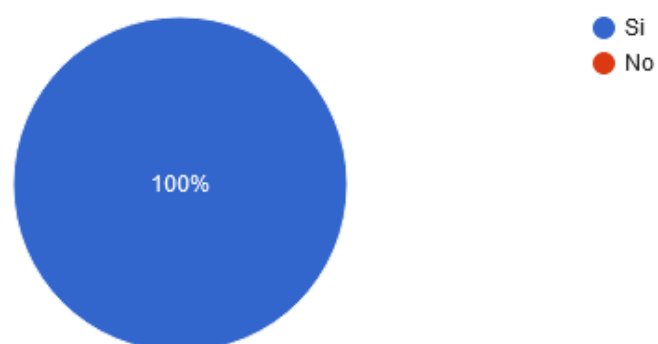
9 responses



**Figura 93.** Imagen pruebas de usabilidad 15. Autoría propia, (2022).

¿Cree que los contenidos y servicios que se ofrecen en este sitio son de utilidad para su caso personal?

9 responses



**Figura 94.** Imagen pruebas de usabilidad 16. Autoría propia, (2022).



### **Conclusiones.**

El manejo de las herramientas tecnológicas para la implementación del desarrollo como lo son Java y los demás frameworks, permiten que la funcionalidad del software sea más optima y escalable, gracias a esto y tomando en cuenta la metodología para el desarrollo la cual permitió una retroalimentación más rigurosa permitiendo al cliente sentirse a gusto con los resultados, se puede resaltar lo siguiente:

Estructura adecuada e intuitiva del manejo de los datos.

Creación, actualización, eliminación lectura y filtración correcta de los datos.

Modelo Lean canvas bien estructurado y funcional.

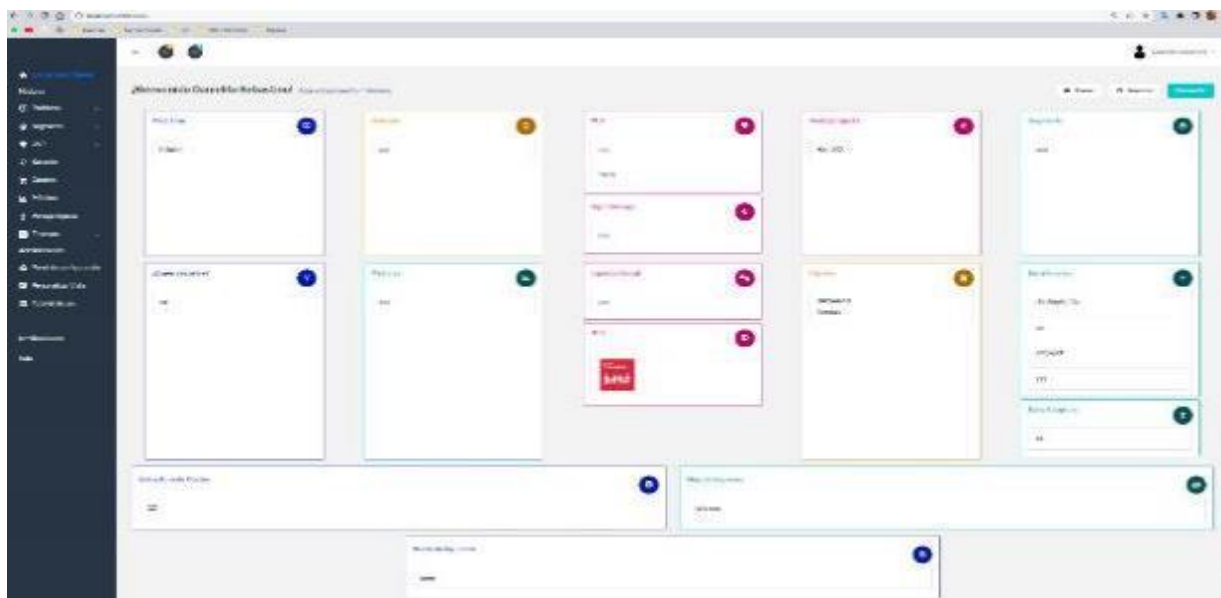
Implementación del módulo de Objetivos de Desarrollo Sostenible (ODS).

## Anexos

### Manual de usuario.

#### Vista Principal

En esta primera vista se encuentra el lienzo totalmente vacío, y en la parte izquierda se encontrará el menú de navegación donde están cada uno de los módulos que se deberán gestionar.



**Figura 95.** Imagen vista principal. Autoría propia, (2022).

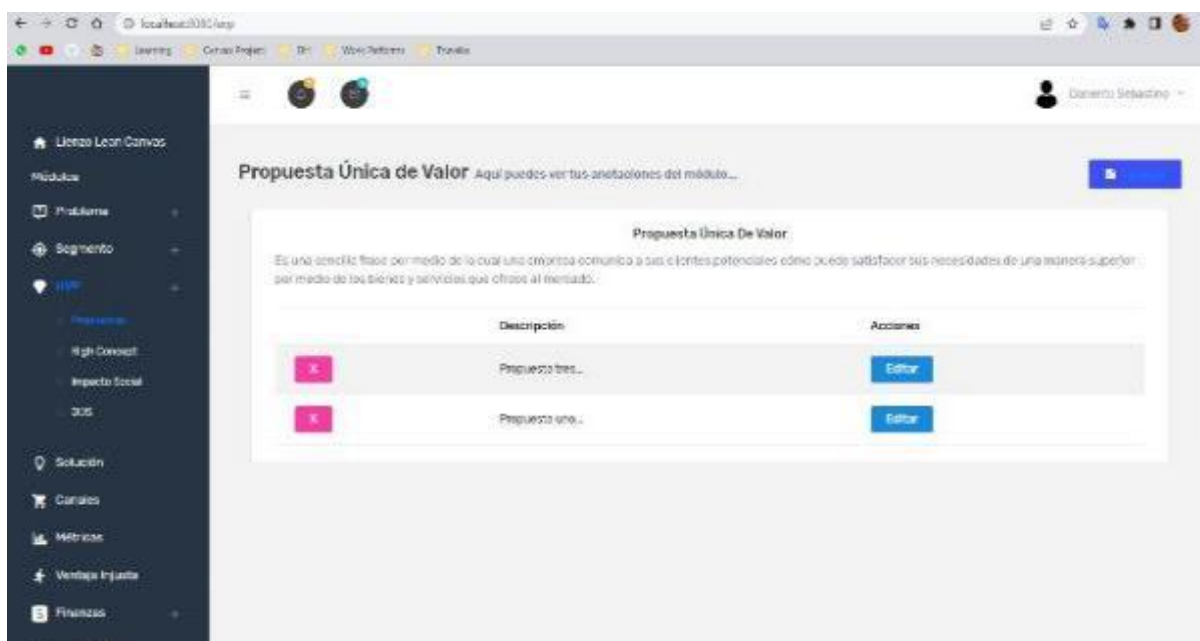
### Gestión por módulos

En este apartado se encuentra el paso a paso que se debe realizar por cada módulo, como el proceso es repetitivo para algunas tablas sólo se hará referencia con un módulo para aclarar el procedimiento que es similar, y aquellas que tengan otro procedimiento serán expuestas para su correcta navegación:

Paso 1: Se podrá agregar los datos en el campo requerido, debajo de descripción:

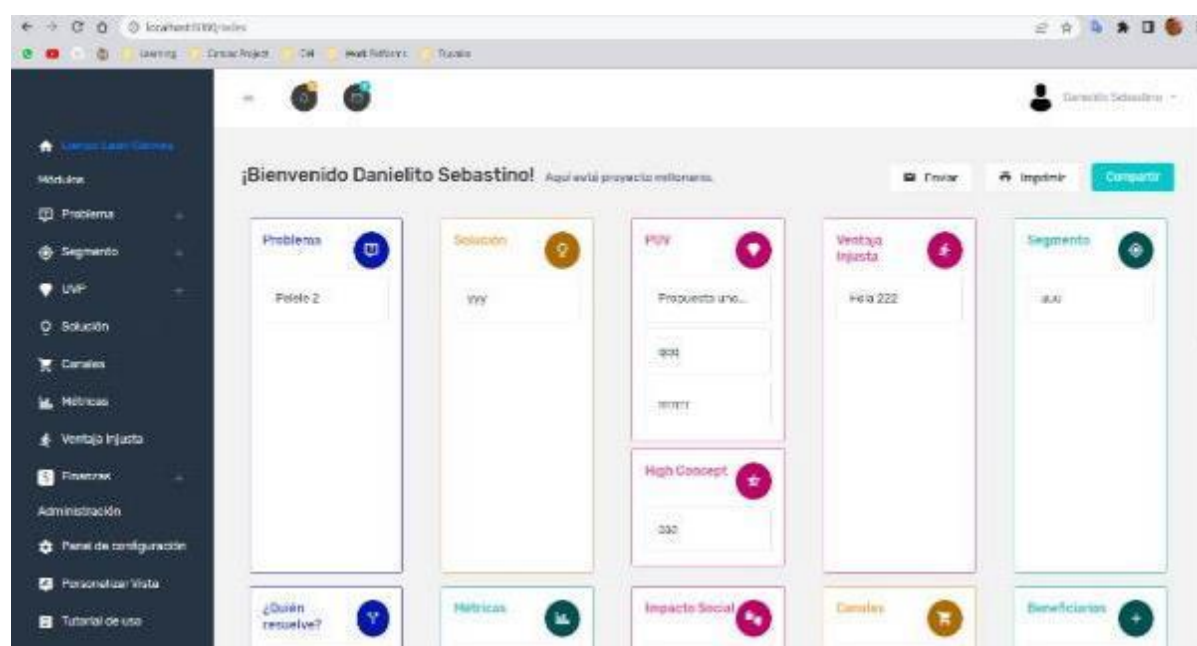
**Figura 96.** Imagen vista formulario de ingreso. Autoría propia, (2022).

Paso 2: En el siguiente ítem se verán los datos ingresados, en esta sección se podrán eliminar los datos y también al dar clic al botón editar se verá el texto para editar.



**Figura 97.** Imagen gestión de datos. Autoría propia, (2022).

Paso 3: se mostrarán posteriormente los datos ingresados.



**Figura 98.** Imagen lienzo. Autoría propia, (2022).

**Nota:** Los módulos que contienen tablas similares son los siguientes:

- Problema
- Segmento
- UVP – Sin contar el submódulo (ODS)
- Solución

- Métricas
- Ventaja injusta

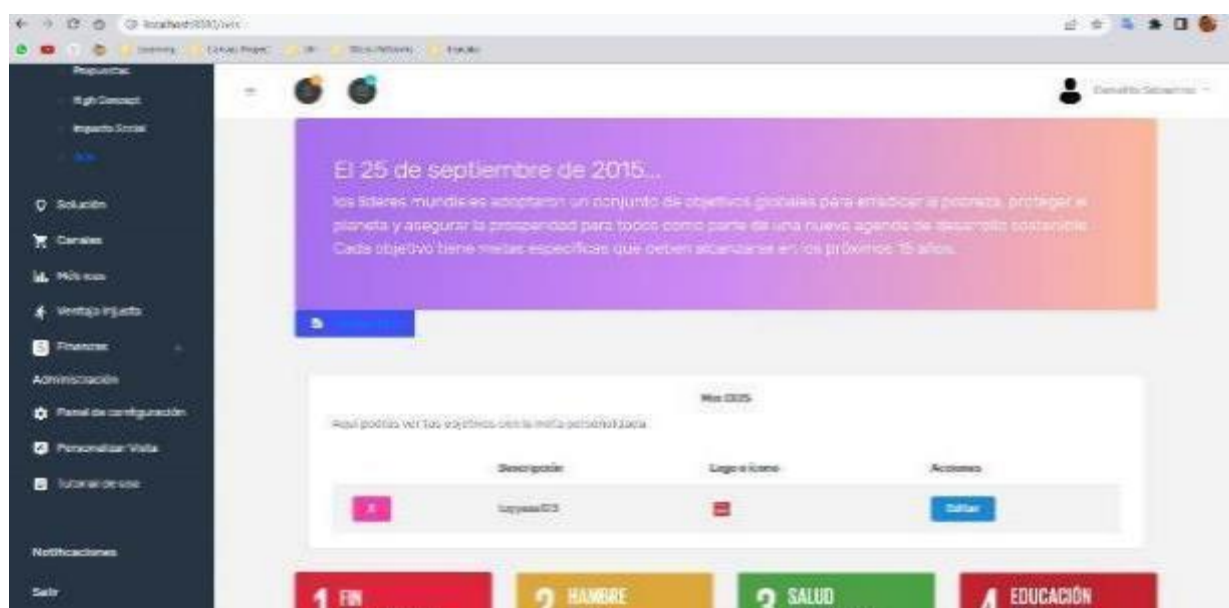
### Módulos con diferente función:

**ODS:** En este módulo aparecerán los objetivos de desarrollo sostenible los cuales se podrán seleccionar y posterior mente se verán en la parte superior con la opción de editarlos y eliminarlos.



**Figura 99.** Vista ODS. Autoría propia, (2022).

Aquí se verán los datos que se han seleccionado



**Figura 100.** Vista ODS 2. Autoría propia, (2022).

Luego si se desea **editar** saldrá el mismo formulario con la opción a escoger nuevamente y para cambiar la meta



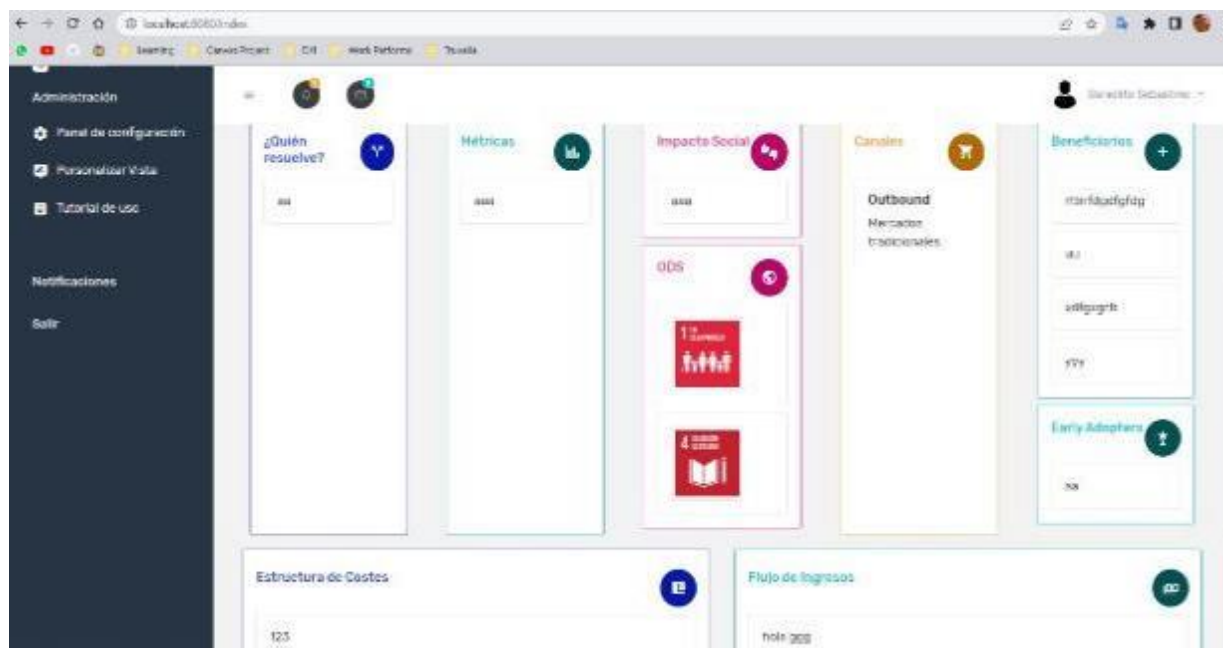
**Figura 101.** Vista ODS 3. Autoría propia, (2022).

Este es el resultado al escoger dos ODS



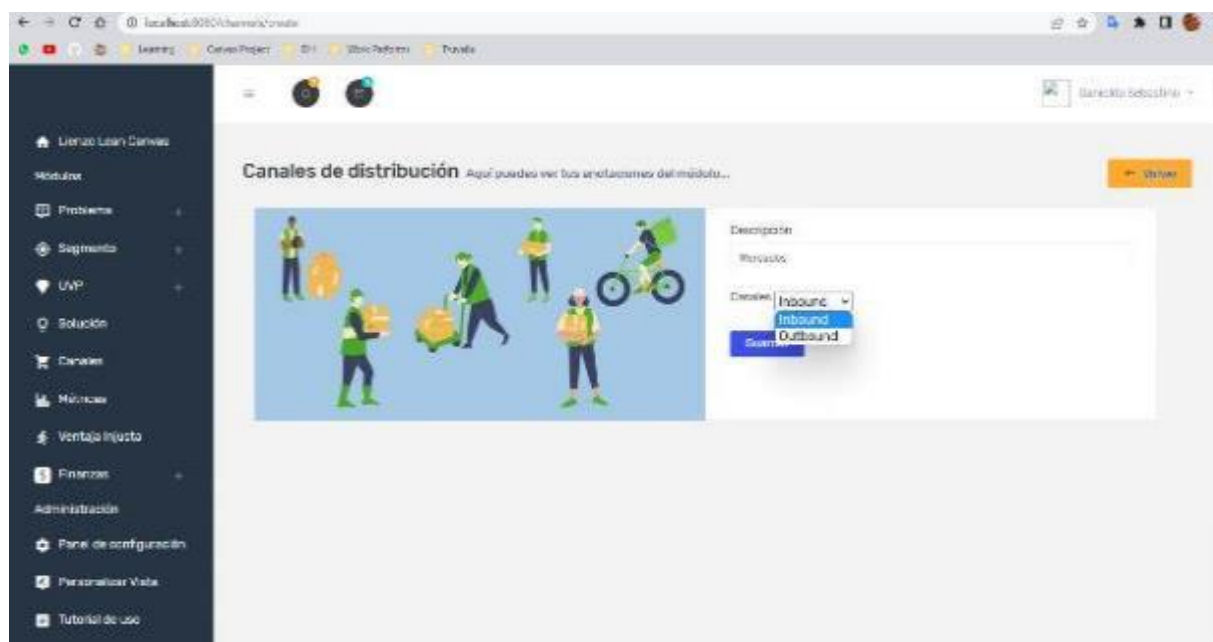
**Figura 102.** Vista ODS 4. Autoría propia, (2022).

A continuación, tendremos la vista de estos objetivos en el lienzo.



**Figura 103.** Vista ODS 5. Autoría propia, (2022).

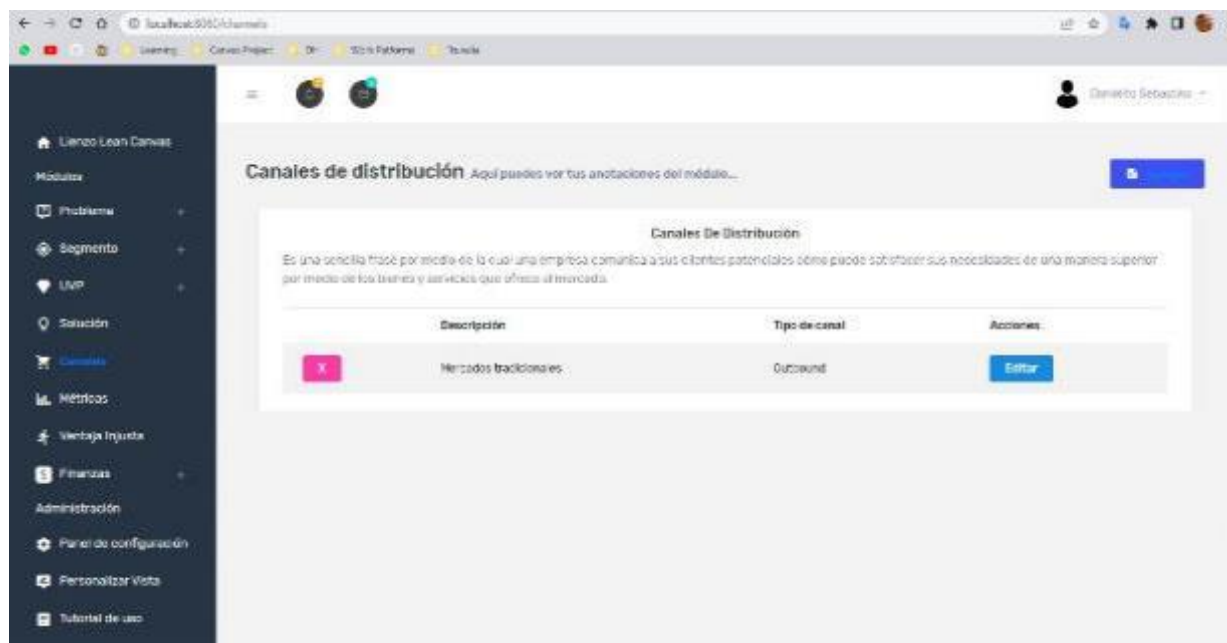
**Canales.** Este módulo se deberá escoger si se quiere un canal inbound o outbound



**Figura 104.** Vista ingreso de canales. Autoría propia, (2022).

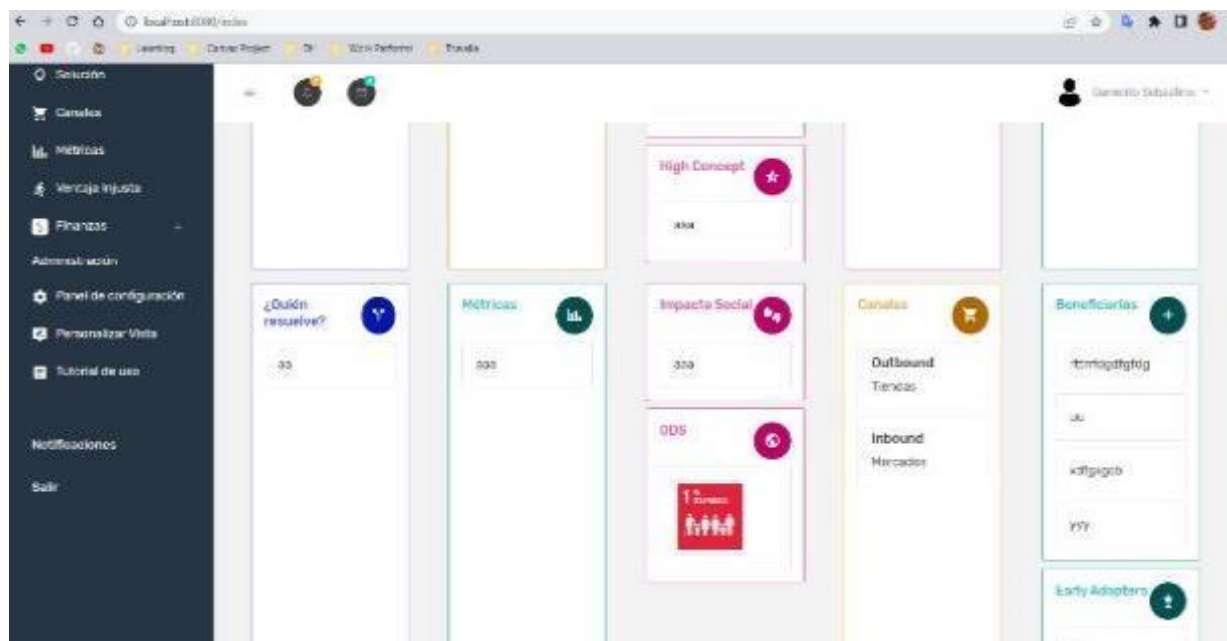
Luego se mostrarán los datos escogidos según la selección y en esta sección se podrán eliminar o editar los datos al oprimir el botón de editar que permitirá cambiar lo anteriormente puesto en el formulario de envío





**Figura 105.** Vista gestión de canales. Autoría propia, (2022).

Luego que los datos que se quieren estén correctamente diligenciados se verán estos cambios en el lienzo.



**Figura 106.** Vista de canales en el lienzo. Autoría propia, (2022).



## Referencias

- Bello, E. (2021). Descubre qué es el Extreme Programming y sus características. IEBS. [Entrada de Blog]. Recuperado de: <https://www.iebschool.com/blog/que-es-el-xp-programming-agile-scrum/>
- Canvanizer. (2019). [Página Canvanizer]. Recuperado de: <https://canvanizer.com/choose-canvas>.
- EDIT.org. (s.f.a). Plantillas para realizar el business Model Canvas online. EDIT.org. [Entrada de Blog]. Recuperado de: <https://edit.org/es/blog/plantillas-business-canvas-model-online>.
- Equipo Geek. (2019). ¿Qué es Java Hibernate? ¿Por qué usarlo? if geek then. Recuperado de: <https://ifgeekthen.nttdata.com/es/que-es-java-hibernate-por-que-usarlo>.
- JavaTpoint. (2021). Sprint Boot JPA. javaTpoint. Recuperado de: <https://www.javatpoint.com/spring-boot-jpa>.
- Molina, D. (2022). Qué es el Modelo Canvas y ejemplos de Canvas reales. IEBS. [Entrada de Blog]. Recuperado de: <https://www.iebschool.com/blog/que-es-el-modelo-canvas-y-como-aplicarlo-a-tu-negocio-agile-scrum/>.
- Muradas, Y. (2022). Conoce las 3 metodologías ágiles más usadas. OpenWebinars. [Entrada de Blog]. Recuperado de: <https://openwebinars.net/blog/conoce-las-3-metodologias-agiles-mas-usadas/>.
- Pizarro, M. (2011). Business Model Canvas. Business Innovation. Recuperado de: <https://www.marcelopizarro.com/2011/12/business-model-canvas/>.
- Pizarro, M. (2011). Business Model Canvas. [Plantilla de Business Model Canvas.]. Recuperado de: <https://www.marcelopizarro.com/2011/12/business-model-canvas/>.
- Sánchez, X. (2020a). Aprende a diseñar tu modelo de negocio con Lean Canvas. Emprenderalia magazine. Recuperado de: <https://www.emprenderalia.com/lean-canvas-disenar-modelos-de-negocio/>.
- Sánchez, X. (2020b). Los 9 bloques del lienzo Lean [Figura Lean Canvas]. Recuperado de: <https://www.emprenderalia.com/lean-canvas-disenar-modelos-de-negocio/>.
- Strategyser. (2020). [Imagen página Strategyser]. Recuperado de: <https://www.strategyzer.com/canvas>.