



UNIVERSIDAD PRIVADA DE TACNA

FACULTAD DE INGENIERÍA

Escuela Profesional de Ingeniería de Sistemas

**“Implementación de plataforma
Crowdsourcing - Sustain Partners”**

Curso: Programación III

Docente: Mag. Elard Rodriguez Marca

Integrantes:

Agreda Ramirez, Jesús Eduardo (2021069823)

Contreras Lipa, Alvaro Javier (2021070020)

Ortiz Fernández, Ximena Andrea (2021071080)

**Tacna – Perú
2023**

Implementación de plataforma Crowdsourcing - Sustain Partners Documento de Especificación de Requerimientos de Software

Versión 7.0

| CONTROL DE VERSIONES | | | | | |
|----------------------|-----------|--------------|--------------|-------|-----------------------|
| Versión | Hecha por | Revisada por | Aprobada por | Fecha | Motivo |
| 1.0 | AJX | AJX | AJX | 25-10 | Revisión |
| 2.0 | AJX | AJX | AJX | 30-10 | Revisión |
| 3.0 | AJX | AJX | AJX | 02-11 | Revisión |
| 4.0 | AJX | AJX | AJX | 08-11 | Revisión |
| 5.0 | AJX | AJX | AJX | 13-11 | Revisión |
| 6.0 | AJX | AJX | AJX | 15-11 | Revisión |
| 7.0 | AJX | AJX | AJX | 02-12 | Evaluación y Revisión |

ÍNDICE GENERAL

| | |
|---------------------------------------------------------------------|----|
| INTRODUCCIÓN | 4 |
| I. Generalidades de la Empresa | 5 |
| 1. Nombre de la Empresa | 5 |
| 2. Visión | 5 |
| 3. Misión | 5 |
| 4. Organigrama | 5 |
| II. Visionamiento de la Empresa | 5 |
| 1. Descripción del Problema | 5 |
| 2. Objetivos de Negocios | 5 |
| 3. Objetivos de Diseño | 5 |
| 4. Alcance del proyecto | 5 |
| 5. Viabilidad del Sistema | 5 |
| 6. Información obtenida del Levantamiento de Información | 6 |
| III. Análisis de Procesos | 6 |
| a) Diagrama del Proceso Actual – Diagrama de actividades | 6 |
| b) Diagrama del Proceso Propuesto – Diagrama de actividades Inicial | 7 |
| IV Especificación de Requerimientos de Software | 7 |
| a) Cuadro de Requerimientos funcionales Inicial | 7 |
| b) Cuadro de Requerimientos No funcionales | 7 |
| c) Cuadro de Requerimientos funcionales Final | 8 |
| d) Reglas de Negocio | 9 |
| V Fase de Desarrollo | 12 |
| 1. Perfiles de Usuario | 12 |
| 2. Modelo Conceptual | 5 |
| a) Diagrama de Paquetes | 5 |
| b) Diagrama de Casos de Uso | 12 |
| c) Escenarios de Caso de Uso (narrativa) | 14 |



| | |
|----------------------------------------|----|
| 3. Modelo Lógico | 23 |
| a) Análisis de Objetos | 23 |
| b) Diagrama de Actividades con objetos | 32 |
| c) Diagrama de Secuencia | 37 |
| d) Diagrama de Clases | 42 |

Contenido

I. Generalidades de la Empresa

1. Nombre de la Empresa

Sustain Partners

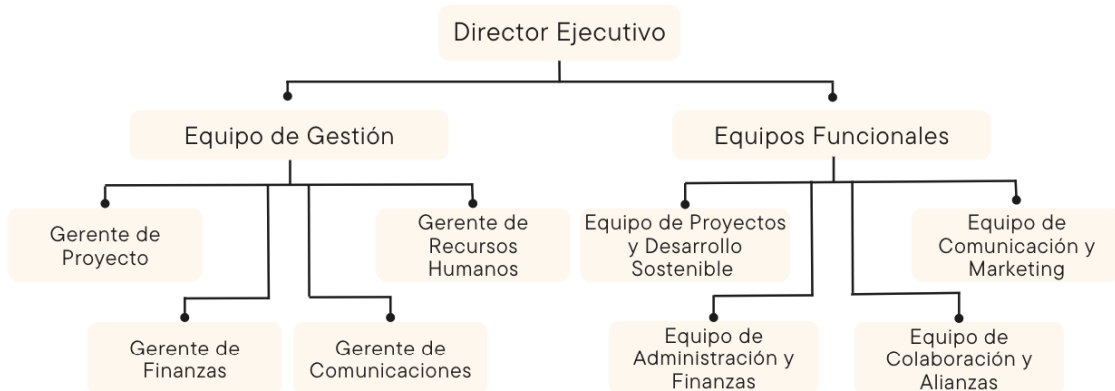
2. Visión

“Ser la plataforma líder en Perú que facilita la colaboración y la acción para lograr los Objetivos de Desarrollo Sostenible (ODS) de las Naciones Unidas a nivel local y nacional, promoviendo un Perú más sostenible y equitativo. ”

3. Misión

"Nuestra misión en Perú es empoderar a individuos y organizaciones para abordar los desafíos locales relacionados con los ODS. Proporcionamos la plataforma que facilita la colaboración, ofrece recursos y promueve la innovación, uniendo a la comunidad peruana en la creación de un Perú más sostenible y equitativo."

4. Organigrama



En Sustain Partners, el Director Ejecutivo supervisa la estrategia general. El Gerente de Proyecto coordina las actividades diarias relacionadas con los ODS. Además, hay equipos funcionales especializados en proyectos sostenibles, colaboraciones clave y comunicaciones. El equipo de administración y finanzas se encarga de la gestión diaria. Juntos, trabajamos para lograr un Perú más sostenible y equitativo.

II. Visionamiento de la Empresa

1. Descripción del Problema

La plataforma Sustain Partners surge como respuesta a la creciente necesidad de un espacio de colaboración dedicado a promover la sostenibilidad y la acción en Perú. Se reconoce que actualmente no existe un lugar centralizado para que individuos, organizaciones y comunidades se unan y ejecuten proyectos relacionados con los Objetivos de Desarrollo Sostenible (ODS) de las Naciones Unidas. La falta de una plataforma integral dificulta la colaboración efectiva y la consecución de estos objetivos.

2. Objetivos de Negocios

- Facilitar la colaboración entre ciudadanos, organizaciones y comunidades en Perú para abordar desafíos locales relacionados con los ODS.
- Promover una cultura de sostenibilidad y acción en Perú, donde la participación activa y la colaboración sean pilares clave.
- Convertir a Sustain Partners en un símbolo de empoderamiento para la comunidad peruana, destacando que la plataforma no solo se trata de proyectos, sino de una experiencia compartida hacia un futuro más sostenible.

3. Objetivos de Diseño

- Diseñar una interfaz de usuario intuitiva y fácil de usar que permita a los usuarios proponer, buscar y unirse a proyectos de sostenibilidad de manera eficiente.
- Optimizar los flujos de trabajo en la plataforma para garantizar que los usuarios puedan unirse a proyectos y colaborar de manera rápida y efectiva.

4. Alcance del proyecto

a. Inclusiones

- Desarrollo de la plataforma Sustain Partners: El proyecto abarca el diseño, desarrollo e implementación de la plataforma Sustain Partners en Perú, que permitirá a los usuarios proponer, buscar y unirse a proyectos de sostenibilidad en diversas áreas.
- Interfaz de usuario amigable: La plataforma incluirá una interfaz de usuario intuitiva y accesible desde dispositivos móviles y equipos de escritorio, lo que facilitará su uso tanto para individuos como para organizaciones.
- Registro y seguimiento de proyectos: Los usuarios podrán registrar proyectos de sostenibilidad.
- Facilitar la colaboración: La plataforma permitirá a los usuarios unirse como colaboradores a proyectos existentes y participar en su desarrollo.
- Aplicación de Escritorio para administradores: La aplicación de escritorio permitirá realizar un análisis de datos y reportes para la toma de decisiones, fortaleciendo la capacidad de supervisión y gestión.
- API endpoint: La plataforma brindará facilidades a otros proyectos que involucren planteo de proyectos o la recopilación de los mismos brindando un endpoint desde el cual podrán obtener datos de los proyectos de Sustain Partners.

b. Exclusiones

- Procesos de ejecución de proyectos: El alcance del proyecto no incluye la ejecución o implementación directa de proyectos de sostenibilidad. Se centra en proporcionar la plataforma y herramientas para la colaboración.
- Mantenimiento continuo: Aunque se implementarán acuerdos de soporte técnico y mantenimiento, el alcance no cubre actividades de mantenimiento a largo plazo después de la implementación inicial.

- Capacitación general en sostenibilidad: A pesar de ofrecer orientación sobre el uso de la plataforma, el proyecto no incluye capacitación general en sostenibilidad o desarrollo sostenible.
- Cambio de infraestructura tecnológica: El alcance no involucra la actualización o modificación significativa de la infraestructura tecnológica existente, a menos que sea necesario para la integración de la plataforma.

5. Viabilidad del Sistema

5.1 Viabilidad Técnica

Hardware:

Lista de Hardware requerido para el Sistema

| Herramienta | Descripción | Cantidad de herramientas |
|-------------|-------------------------------------------|--------------------------|
| Laptops | Nos permitirá usar y codificar el sistema | 3 unidades |

Software:

Lista de Software requerido para el Sistema

| Herramienta | Descripción |
|---------------------|-------------------------------------------------------------------------------------------------------------------------------------------------|
| Navegador | La compatibilidad de los navegadores web utilizados en la empresa con las aplicaciones basadas en web que formarán parte del sistema de ventas. |
| Internet | Acceso a Internet de alta velocidad |
| Licencia de Windows | Se cuenta con Windows 10 Pro. |
| Visual Studio Code | Versión Gratuita |
| Neon PostgreSQL | Versión Gratuita |
| Servidor en la Nube | Compra de un servidor básico en Linux adquirido en Elastika para la publicación de la página. |
| Redis | Base de datos en memoria para optimizar el código |
| Docker | Para ejecutar el servicio de Redis que se encontrará en un contenedor. |

| | |
|----------------------------|-------------------------------------------------------------------------------------------------------------------|
| Java Development Kit (JDK) | OpenJDK versión 17 de Java. Asegurarse de que las variables de entorno estén configuradas correctamente |
| Python | Para realizar scripting en el proceso de envío de correos electrónicos |
| Apache NetBeans | Versión Gratuita para el desarrollo de la aplicación de escritorio. |
| Apache Maven | Herramienta de manejo de dependencias, gestionar las bibliotecas y dependencias del proyecto de manera eficiente. |

5.2 Viabilidad Económica

Costos Totales del Desarrollo del Sistema

| Tipos de Costo | Subtotal |
|---------------------|---------------------|
| Costos generales | S/ 10,118.00 |
| Costos operativos | S/ 1,120.00 |
| Costos de personal | S/ 29,600.00 |
| Costos del ambiente | S/ 640.00 |
| TOTAL | S/ 41,478.00 |

Cuadro resumen de los Costos generales, Costos operativos, Costos de personal y Costos del ambiente

5.3 Viabilidad Operativa

Capacidad de la empresa para mantener el sistema funcionando:

- **Recursos Financieros:** Se han asignado recursos financieros adecuados para el mantenimiento del sistema a largo plazo. Se han contemplado los costos de alojamiento, licencias de software y soporte técnico.

Lista de interesados:

Usuarios Finales:

Individuos y organizaciones interesadas en participar en proyectos sostenibles. Personas que buscan colaborar con iniciativas alineadas con los Objetivos de Desarrollo Sostenible.

Cientes o Socios Comerciales:

Posibles colaboradores y patrocinadores que pueden beneficiarse de la visibilidad en la plataforma. Entidades comerciales interesadas en promover proyectos sostenibles.

Comunidad y Ciudadanía:

Ciudadanos locales interesados en participar en proyectos que impacten positivamente en la comunidad. Grupos y organizaciones comunitarias que buscan oportunidades para contribuir al desarrollo sostenible.

Educadores y Instituciones Académicas:

Profesores y estudiantes interesados en utilizar la plataforma como recurso educativo. Instituciones académicas que pueden beneficiarse de colaboraciones en proyectos de investigación.

Empresas con objetivos afines:

Empresas que deseen recopilar proyectos para el impacto de las ODS tendrán acceso a la API implementada para retribuir los proyectos.

6. Información obtenida del Levantamiento de Información

1. ¿Existen iniciativas o programas en la ciudad relacionados con el desarrollo sostenible y los Objetivos de Desarrollo Sostenible (ODS) de las Naciones Unidas?

Si existen programas dentro de la ciudad, los cuales competen sobre todo en lo que es el apoyo humanitario, Proyecto País es uno de estos programas que desarrollan actividades relacionadas con las ODS, sobre todo en la gestión de riesgos de desastres. INDECI tiene relación con USAID, el cuál tiene un representante a nivel nacional, los cuales vienen capacitando y formando talleres del sistema de comandos de incidentes en caso de algún desastre.

2. ¿Cuáles son los principales desafíos o necesidades que enfrenta la ciudad en términos de sostenibilidad y desarrollo sostenible?

El primer desafío es el económico. El tema que hay que enfrentar es la adjudicación de un fondo a través de los gobiernos para poder desarrollar estos programas.

3. **¿Existe interés en promover la colaboración entre ciudadanos y organizaciones locales para abordar estos desafíos sostenibles? Si es así, ¿de qué manera se ha fomentado hasta ahora?**

Si existe interés por parte de los gobiernos locales y regionales. Desde el punto de vista de la gestión de riesgos desastres están los voluntariados. El gobierno regional busca crear un grupo de voluntarios en reserva para que puedan aportar ante alguna emergencia.

4. **¿El municipio ha considerado la posibilidad de utilizar una plataforma en línea para facilitar la colaboración y la ejecución de proyectos sostenibles en la ciudad?**

Si, el gobierno regional tiene plataformas virtuales en las cuales puedes acceder a diferentes tipos de ayuda, como el Consulta Ciudadano.

5. **¿Cuáles son las áreas prioritarias de acción para el municipio en términos de sostenibilidad y desarrollo sostenible? ¿Existen proyectos específicos o metas relacionadas con los ODS?**

Sí, dentro de las áreas prioritarias podría considerarse a la seguridad ciudadana.

6. **¿El municipio está dispuesto a respaldar o colaborar en la implementación de un proyecto como Sustain Partners, que fomente la colaboración y la ejecución de proyectos sostenibles en la ciudad?**

Si, si el proyecto facilita la comunicación, el enlace entre las personas y las autoridades.

7. **¿Cómo considera que un proyecto como Sustain Partners podría beneficiar a la ciudad y a sus ciudadanos en términos de desarrollo sostenible, participación ciudadana y logro de los ODS?**

El proyecto beneficiaría a la ciudad en distintos aspectos, sobre todo en el tema de la limpieza y seguridad ciudadana.

8. **¿Cuáles son las políticas o regulaciones locales que podrían influir en la implementación de un proyecto de este tipo?**

Para poder desarrollarse un proyecto de esta magnitud, debe de pasar por la aprobación por de un consejo regional, al final ellos son los que aprueban los proyectos que se van a ejecutar, entonces primero se elevaría esta solicitud al

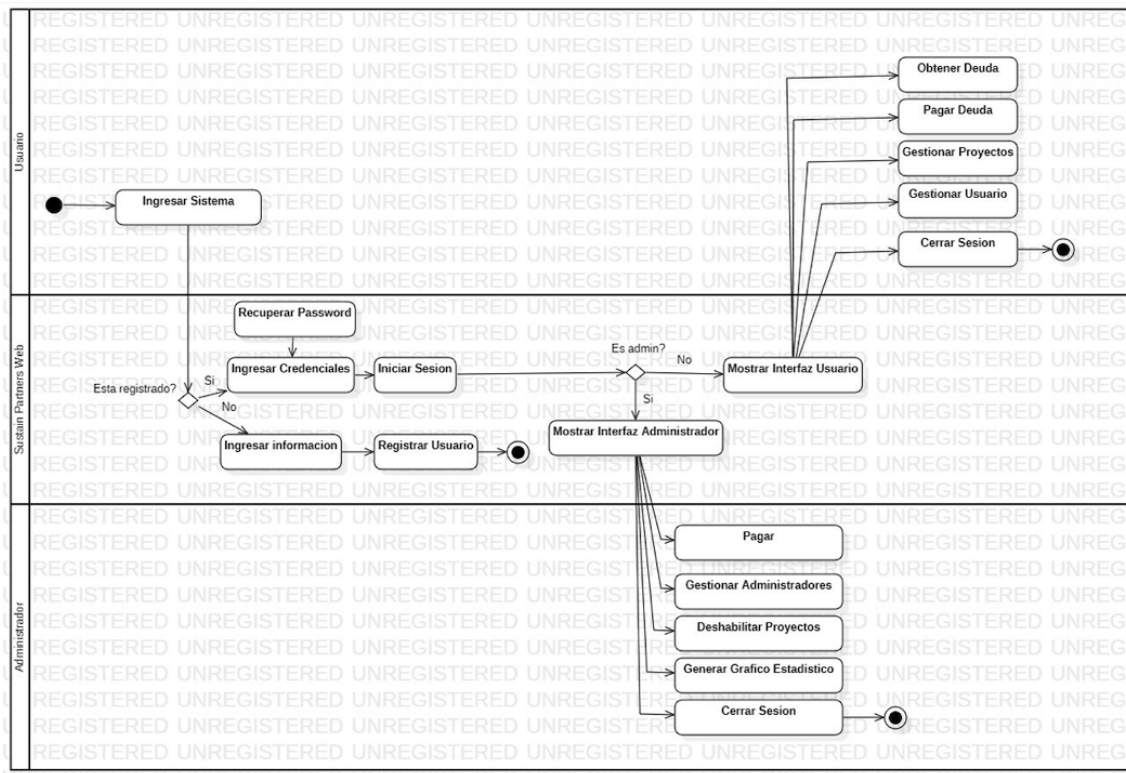
consejo regional para que ellos en una sesión ordinaria a través de una ordenanza regional se pueda ejecutar.

9. ¿El municipio estaría dispuesto a colaborar en la promoción y difusión de Sustain Partners entre la comunidad local?

Si, es importante la difusión, mediante la redes sociales se podría colaborar continuamente esta plataforma digital.

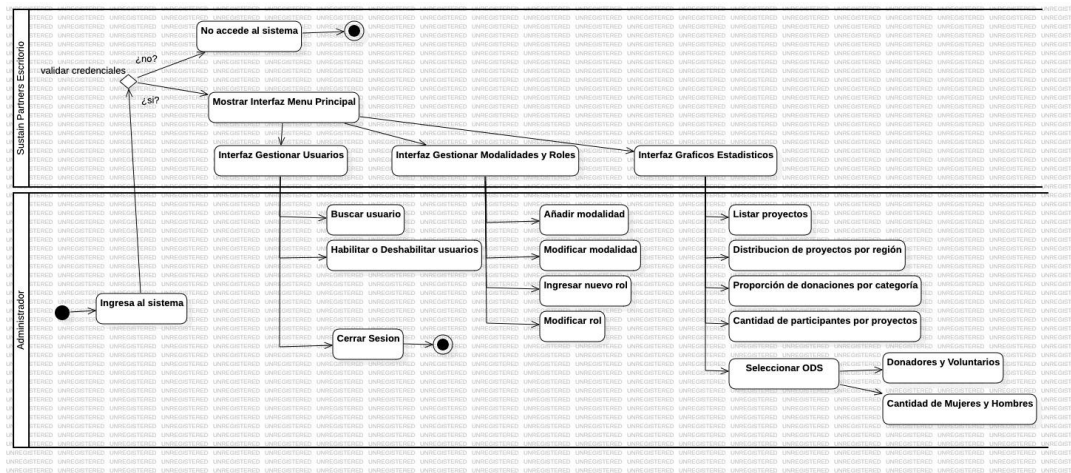
III. Análisis de Procesos

Diagrama de Procesos Propuesto en Sustain Partners Aplicación Web



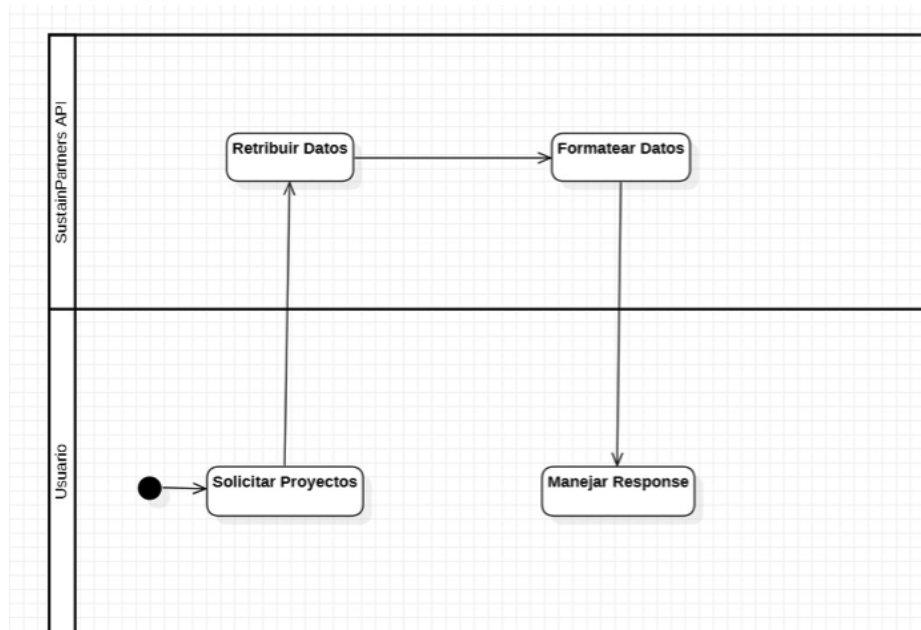
El diagrama de actividades presentado, muestra todas las opciones que presenta la aplicación web y el cómo llegar a estas. Se dividen por 3 diferentes pools, los cuales son el sistema, el usuario y el administrador, comenzando desde el usuario ingresando a página y finalizando en varias ocasiones dependiendo de las opciones que opte por realizar.

Diagrama de Procesos Propuesto en Sustain Partners Aplicación de Escritorio



El diagrama de actividades presentado, muestra todas las opciones que presenta la aplicación de escritorio y el cómo llegar a estas. Se dividen por 2 diferentes pools, debido a que solo administradores tendrán acceso a este, para realizar gestiones y procedimientos necesarios.

Diagrama de Procesos Propuesto en Sustain Partners API



El diagrama de actividades presentado, muestra la interacción entre un usuario y la API de Sustain Partners, comenzando con el usuario o el sistema solicitando los proyectos. El endpoint de sustain partners recuperara los datos solicitados y les brindara el formato JSON respectivo para devolver los datos en una response, la cual será manejada como el usuario decida.

IV. Especificación de Requerimientos de Software

a) Cuadro de Requerimientos Funcionales Inicial

| Código | Requerimiento | Descripción |
|--------|----------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| RF-01 | Registrar usuario | El sistema deberá permitir el registro de la persona en el sistema. |
| RF-02 | Inicio sesión como usuario | El usuario podrá ingresar al sistema luego de proporcionar su usuario, que será su DNI, y contraseña |
| RF-03 | Modificar usuario | El usuario podrá editar sus datos previamente guardados en la aplicación web. |
| RF-04 | Búsqueda de Proyectos | Los usuarios deben poder buscar proyectos sostenibles usando palabras clave. |
| RF-05 | Proposición de Proyectos | Las comunidades locales deben poder proponer proyectos sostenibles en la plataforma, proporcionando detalles como descripción, objetivos, ubicación y recursos necesarios. |
| RF-07 | Gestión de Proyectos | Los creadores de proyectos deben poder administrar y actualizar la información de sus proyectos, así como agregar o eliminar colaboradores, inversores o voluntarios. |

b) Cuadro de Requerimientos No Funcionales

| Código | Requerimiento | Descripción |
|----------|---------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| RNF - 01 | Interfaz intuitiva | La interfaz de usuario debe ser fácil de usar y amigable para adultos mayores, con una navegación clara y sencilla. |
| RNF - 02 | Seguridad de acceso | El sistema deberá tener implementadas medidas sólidas de seguridad de acceso a la página. |
| RNF - 03 | Rendimiento óptimo | La plataforma debe ser rápida y eficiente, con tiempos de carga mínimos y una capacidad para manejar múltiples usuarios concurrentes sin degradación significativa del rendimiento. |
| RNF - 04 | Alta accesibilidad | El sistema debe ser altamente accesible para adultos mayores, incluyendo soporte para tecnologías de asistencia como lectores de pantalla, opciones de alto contraste y fuentes legibles. |

c) Cuadro de Requerimientos Funcionales Final

| Código | Requerimiento | Descripción |
|---------------|----------------------------------------|------------------------------------------------------------------------------------------------------------------------------------|
| RF - 01 | Gestionar Usuario | El sistema permitirá a los usuarios registrarse y modificar sus datos personales. |
| RF - 02 | Iniciar Sesión | Los usuarios podrán ingresar al sistema proporcionando sus credenciales. |
| RF - 03 | Gestionar Proyectos | Incluye la proposición de proyectos, la inscripción a proyectos existentes, donaciones y la elección de roles en proyectos. |
| RF - 04 | Gestionar Administradores | Incluye la asignación de roles de administrador y la visualización de usuarios registrados. |
| RF - 05 | Habilitar y Deshabilitar Proyecto | Permite a los administradores cambiar el estado de un proyecto a "Deshabilitado" o "Habilitado". |
| RF - 06 | Generar Gráficos Estadísticos | Permite a los administradores ver reportes referentes a las estadísticas de los proyectos. |
| RF - 07 | Gestionar Roles y Modalidades | Permite a los administradores agregar y modificar las modalidades y roles existentes del sistema. |
| RF - 08 | Recuperar Contraseña | Los usuarios podrán solicitar un restablecimiento de su contraseña. |
| RF - 09 | Historial de pagos por correo | Los usuarios podrán solicitar un historial de todos los pagos que realizaron. Este será enviado al correo enlazado con su usuario. |
| RF - 10 | Realizar pago de donaciones pendientes | Los usuarios podrán realizar los pagos de las donaciones pendientes que tienen, ingresando sus datos financieros. |
| RF - 11 | Obtener proyectos | Los usuarios podrán solicitar información de los proyectos mediante el endpoint de la API. |

d) Reglas de negocios

| Nombre | Descripción | Autoridad |
|------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|---------------|
| Registro Obligatorio para Propuestas | Un usuario debe estar registrado e iniciar sesión para proponer un proyecto sostenible en la plataforma. | Usuario |
| Donación Necesaria para Donadores | Si un usuario elige el rol de donador al inscribirse en un proyecto, es necesario que realice una donación monetaria al proyecto en el que se inscribe | Usuario |
| Restricción de Acceso a Datos | Los administradores no tendrán acceso a la modificación de datos personales del usuario. | Administrador |
| Restricción de usuarios inscritos en proyectos | Los administradores no podrán editar a los participantes de los proyectos. | Administrador |
| Cambio de estado en Proyecto | Solo los administradores tendrán acceso a cambiar el estado de algún proyecto (Activo o Inactivo) | Administrador |

V. Fases de Desarrollo

1. Perfiles de Usuario

Usuario: Este perfil representa a los usuarios que utilizan la plataforma para buscar proyectos sostenibles, inscribirse en ellos, donar dinero, proponer nuevos proyectos y administrar los proyectos que han propuesto. Las capacidades de este perfil incluirían:

- Buscar proyectos sostenibles.
- Inscribirse en proyectos.
- Donar dinero a proyectos.
- Proponer nuevos proyectos sostenibles, proporcionando detalles como descripción, objetivos y ubicación.
- Participar como donador o voluntario en proyectos.

Administrador: El perfil de administrador tendrá acceso completo al sistema, tanto en la aplicación web, como en la aplicación de escritorio y será responsable de la gestión y supervisión general de la plataforma. Las capacidades de este perfil incluirían:

- Gestionar el estado de los proyectos propuestos.
- Deshabilitar cuentas de usuario en casos de incumplimiento de políticas o problemas.
- Visualizar gráficos estadísticos en la aplicación web y la aplicación de escritorio
- Gestionar Modalidades y Roles

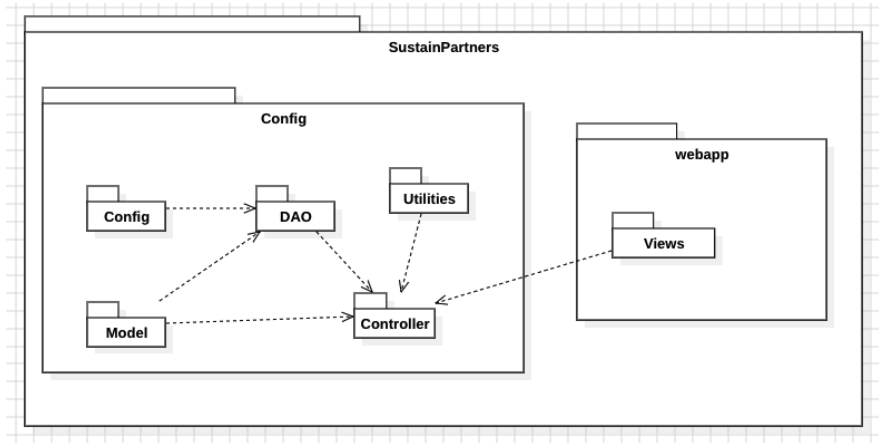
Empresa: El perfil de empresa sólo tendrá acceso a consultar datos de proyectos en el endpoint mediante la API. Las capacidades de este perfil incluirían:

- Consultar al endpoint por los datos de los proyectos.

2. Modelo Conceptual

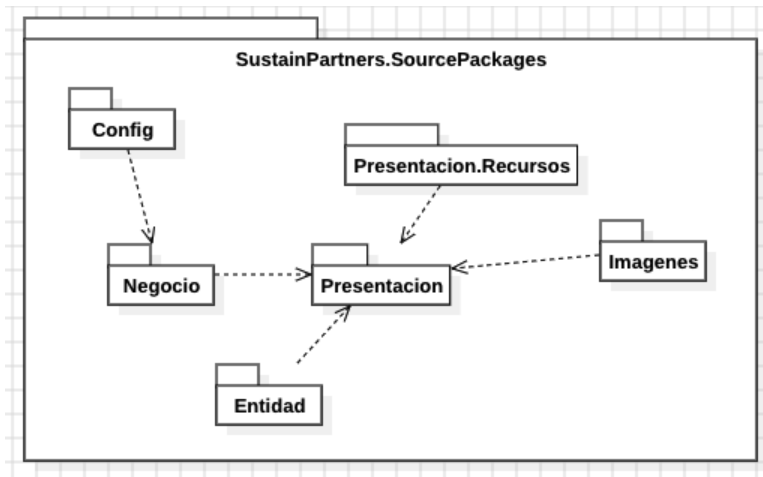
a) Diagrama de Paquetes

Diagrama de Paquetes Aplicación web - Sustain Partners



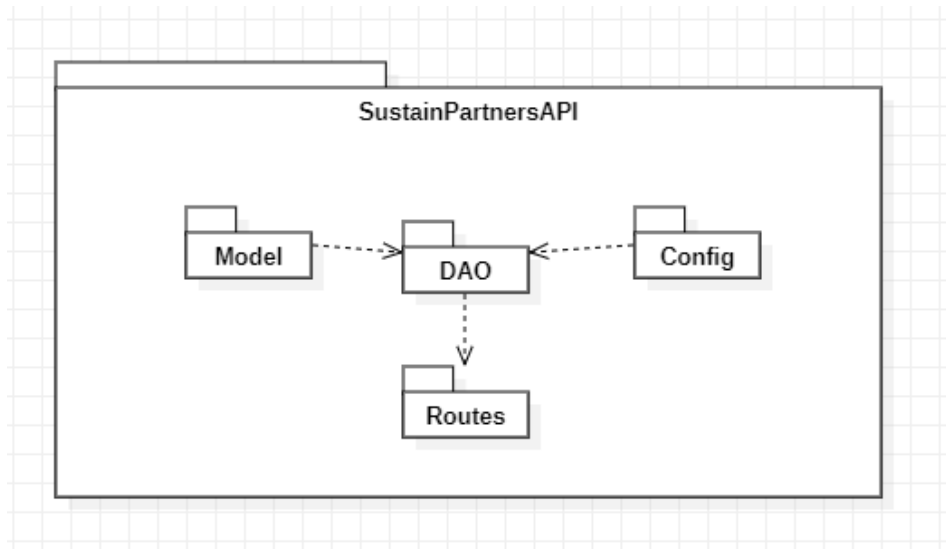
El sistema "Sustain Partners Web" se organiza en paquetes clave: "Config" para archivos de configuración, "DAO" para interacción con la base de datos, "Model" para entidades, "Controller" para lógica y "Utilities" para scripts de correo. Además, el paquete externo "webapp", con su subpaquete "Views", se enfoca en la interfaz web, brindando una estructura modular eficiente para gestionar configuración, datos, lógica y la interfaz web.

Diagrama de Paquetes Aplicación de escritorio - Sustain Partners



El diagrama muestra un paquete "Config" para archivos de conexión, "Negocio" gestiona la interacción y métodos con la base de datos, y "Entidad" almacena entidades relacionadas. "Presentación" contiene interfaces de usuario, con subpaquetes como "Presentación.Recursos" para botones en tablas.

Diagrama de Paquetes API - Sustain Partners



En el diagrama de paquetes de la API podemos observar el paquete Config el cual alberga la conexión a la base de datos en nube postgresql(Neon) el paquete Model el cual tiene la clase Proyecto para la retribución de datos y en el DAO se realiza los procesos de obtención de datos usando Proyecto y la conexión a la DB por último el paquete Routes es el encargado de los endpoints de la API donde se retribuirá la data al usuario.

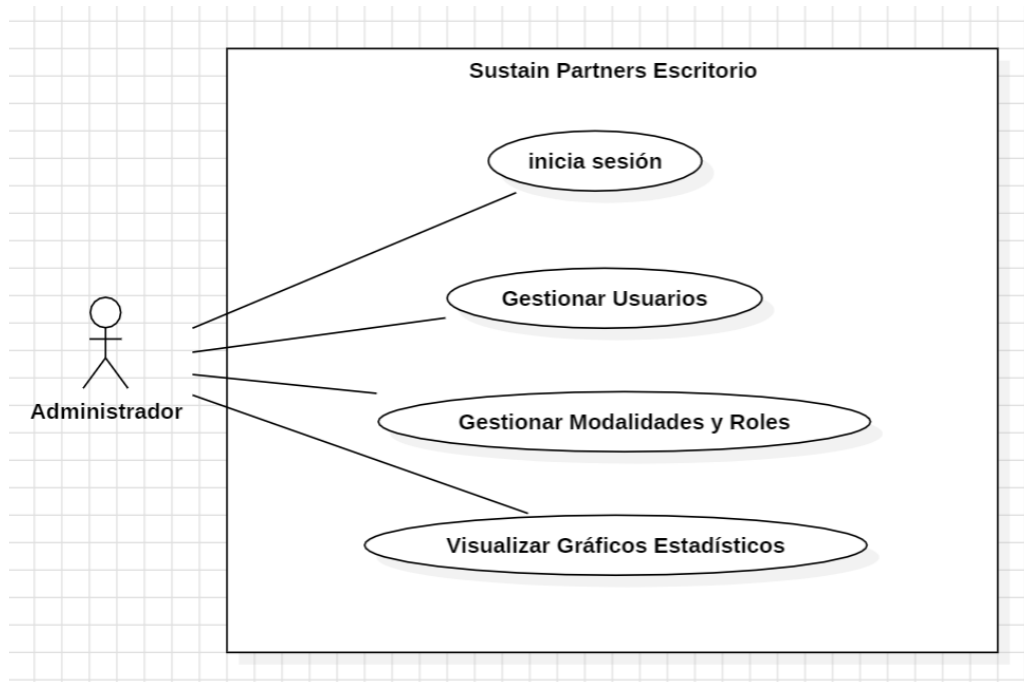
b) Diagrama de Casos de Uso

Diagrama de Casos de Uso Aplicación web - Sustain Partners



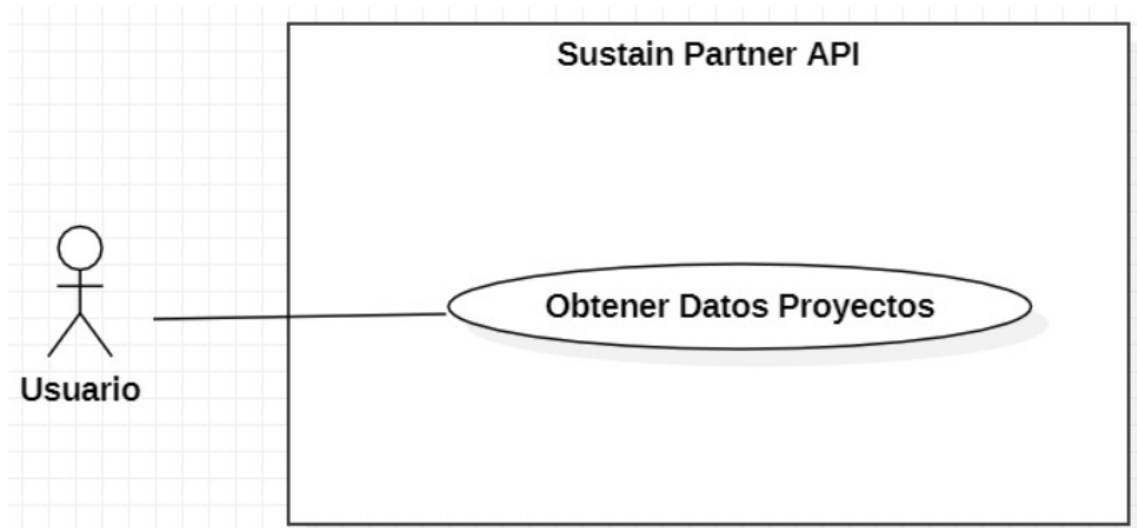
El diagrama presenta todos los casos de uso de la aplicación web relacionados con los actores (administradores y usuarios) los cuales tienen permitidos acceder a diferentes opciones las cuales están indicadas en el diagrama.

Diagrama de Casos de Uso Aplicación de escritorio - Sustain Partners



El diagrama presenta todos los casos de uso de la aplicación de escritorio relacionados con el actor administrador, el cual será el único en tener acceso al sistema

Diagrama de Casos de Uso API Sustain Partners



El diagrama presenta el caso de uso para retribuir los datos del endpoint en la API en este caso el usuario representa un sistema, empresa o usuario que desee obtener los datos de los proyectos.

c) Escenarios de Caso de Uso (narrativa)

Escenarios de Caso de Uso de Aplicación web - Sustain Partners

i) Caso de uso: Gestionar Usuario

- 1) Actores: Usuario
- 2) Propósito: El usuario podrá registrar y modificar sus datos en el sistema.
- 3) Precondición: El usuario debe de haber dado click en el enlace "Editar Perfil"
- 4)

| Acción del actor | Curso del sistema |
|-----------------------------------------------------------------------------------------|-----------------------------------------------------------------------|
| 1. El usuario ingresa al apartado de su Perfil. | 2. El sistema acepta la solicitud y muestra la interfaz de su perfil. |
| 3. El usuario procede a realizar los cambios para posteriormente grabarlo en su perfil. | 4. El sistema registra los cambios de edición. |

ii) Caso de uso: Iniciar Sesión

- 1) Actores: Usuario o Administrador
- 2) Propósito: Proporcionar a los usuarios una forma segura de acceder, utilizando su usuario y contraseña.
- 3) Precondición: El usuario debe estar registrado en el sistema.
- 4)

| Acción del actor | Curso del sistema |
|-------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------|
| 1. El usuario ingresa su usuario y contraseña en el formulario de inicio de sesión. | 4. El sistema valida las credenciales ingresadas por el usuario y el captcha. |
| 2. El usuario realiza el captcha de seguridad | 5. El sistema verifica la existencia del usuario en la base de datos. |
| 3. El usuario selecciona la opción "Iniciar Sesión." | 6. El sistema otorga acceso al usuario y muestra el menú principal correspondiente a su rol. |

iii) Caso de uso: Gestionar Proyectos

- 1) Actores: Usuarios y Administradores
- 2) Propósito: Poder participar en los proyectos eligiendo una modalidad de apoyo, así como poder crearlos.
- 3) Precondición: El usuario debe haber iniciado sesión.
- 4)

| Acción del actor | Curso del sistema |
|------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------|
| 1. El usuario accede a un proyecto haciendo clic sobre los que están desplegados en la interfaz principal. | 2. El sistema muestra la interfaz del proyecto seleccionado |
| 3. El usuario da clic en la opción participar escogiendo uno de los roles disponibles para el proyecto. | 4. El sistema muestra una interfaz para donar dinero o colaborar en el proyecto |
| 5. El usuario escoge la opción de donar. | 6. El sistema despliega una interfaz para ingresar un monto con el que se desea ayuda. |
| 7. El usuario ingresa el monto y confirma el pago en la página principal | 8. El sistema detecta que la tarjeta es original y procesa el pago. |

iv) Caso de uso: Gestionar Administradores

- 1) Actores: Administrador
- 2) Propósito: Designar a un administrador para que el trabajo de monitorear los proyectos sea más sencillo.
- 3) Precondición: El administrador debía iniciar sesión.
- 4)

| Acción del actor | Curso del sistema |
|----------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------|
| 1. El administrador ingresa a la lista de usuarios registrados en el sistema | 2. El sistema realiza una búsqueda en la base de datos de todos los usuarios y los muestra. |
| 3. El administrador selecciona un usuario para poder convertirlo en administrador de igual manera. | 4. El sistema accede al cambio de rol del usuario a administrador. |

v) Caso de uso: Obtener Historial de Pagos

- 1) Actores: Usuarios y Administradores
- 2) Propósito: Solicitar el historial de pagos que se enviará al correo electrónico.
- 3) Precondición: El usuario debe haber iniciado sesión y tener pagos realizados.
- 4)

| Acción del actor | Curso del sistema |
|---------------------------------------------------------------|-------------------------------------------------------------------------|
| 1. El usuario accede a su perfil. | 2. El sistema muestra la interfaz del perfil del usuario. |
| 3. El usuario da clic en la opción enviar historial de pagos. | 4. El sistema llama al script de envío de historial de pagos en Python. |
| 5. El usuario revisa su correo. | |

vi) Caso de uso: Pagar Deuda

- 1) Actores: Usuarios y Administradores
- 2) Propósito: Poder culminar el pago de las donaciones a los proyectos que seleccionó el usuario..
- 3) Precondición: El usuario debe haber iniciado sesión y tener deudas pendientes.
- 4)

| Acción del actor | Curso del sistema |
|-------------------------------------|------------------------------------------------------------------|
| 1. El usuario accede al menú pagar. | 2. El sistema muestra el formulario de pago. |
| 3. El usuario ingresa su tarjeta. | 4. El sistema valida la tarjeta reconociendo si es real o falsa. |
| 5. El usuario acepta el pago. | 6. El sistema realiza el pago y actualiza la base de datos. |

vii) Caso de uso: Deshabilitar Proyecto

- 1) Actores: Administrador
- 2) Propósito: Marcar como finalizados los proyectos que alcanzaron sus metas en donaciones.
- 3) Precondición: El administrador debe haber iniciado sesión
- 4)

| Acción del actor | Curso del sistema |
|---------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------|
| 1. El administrador accede a la plataforma y entra a un proyecto. | 2. El sistema muestra la interfaz del proyecto. |
| 3. El administrador da clic en el botón “Cambiar estado del Proyecto” para no poder participar en el proyecto | 4. El sistema restringe el acceso a los usuarios que quieran participar. |

viii) Caso de uso: Generar Gráfico Estadístico

- 1) Actores: Administrador
- 2) Propósito: Permitir tener el conocimiento de los proyectos que más se financian y se llegaron a completar.
- 3) Precondición: El Administrador debe haber iniciado sesión.
- 4)

| Acción del actor | Curso del sistema |
|------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------|
| 1. El administrador da clic en el botón “Ver estadísticas” | 2. El sistema mostrará un gráfico de barras que indiquen los proyectos que se concluyeron o alcanzaron la meta de las donaciones. |

Escenarios de Caso de Uso de Aplicación de escritorio - Sustain Partners

i) Caso de uso: Inicia sesión

- 1) Actores: Administrador
- 2) Propósito: El administrador ingresará a la aplicación de escritorio.
- 3) Precondición: La cuenta con la que se ingrese debe tener el rol de administrador
- 4)

| Acción del actor | Curso del sistema |
|------------------------------------------------------------|-----------------------------------------------------------------------|
| 1. El administrador ingresa al la aplicación de escritorio | 3. El sistema valida las credenciales y permite ingreso. |
| 2. El administrador ingresa su usuario y contraseña | 4. El sistema mostrará la interfaz de menú principal al administrador |

ii) Caso de uso: Gestionar Usuario

- 1) Actores: Administrador
- 2) Propósito: El administrador podrá buscar y habilitar o deshabilitar usuarios
- 3) Precondición: El administrador da clic al botón “Gestionar usuarios”
- 4)

| Acción del actor | Curso del sistema |
|-------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------|
| 2. El administrador da clic en el botón “Deshabilitar” o “Habilitar” para cambiar el estado del usuario | 1. El sistema mostrará la interfaz habilitar o deshabilitar usuarios. |
| 4. Si el administrador desea buscar un usuario en específico debe ingresar el username en la caja de texto y da clic en el botón “Buscar” | 3. El sistema cambia el valor del estado y también el texto del botón 5. El sistema busca el usuario y se visualiza en la tabla |
| 6. Si el administrador desea volver al menú principal, deberá dar clic en el botón “Atrás” | 7. El sistema muestra la interfaz de menú principal |

iii) Caso de uso: Gestionar Modalidades y Roles

- 1) Actores: Administrador
- 2) Propósito: El administrador podrá añadir y modificar modalidades y también ingresar y modificar roles
- 3) Precondición: Encontrarse en el menu principal y darle clic al botón “Gestionar modalidades y roles”

| Acción del actor | Curso del sistema |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 2. El administrador deberá rellenar los campos código y nombre para modalidad y para rol y luego de eso dar click en el botón “Agregar” | 1. El sistema mostrará la interfaz gestionar modalidades y roles |
| 4. Si el administrador desea modificar alguna modalidad o un rol, puede dar click sobre el texto en la tabla, después de esto debe de dar click en el botón “Modificar” que funciona como botón de guardado | 3. El sistema registra el nuevo rol y modalidad y lo imprime en ambas tablas para el usuario 5. El sistema registra las modificaciones y actualiza la tabla |

iv) Caso de uso: Visualizar Gráficos Estadísticos

- 1) Actores: Administrador
- 2) Propósito: El administrador podrá listar los proyectos y visualizar diferentes gráficos estadísticos para la toma de decisiones
- 3) Precondición: Encontrarse en el menú principal y darle click al botón “Gráficos Estadísticos”

| Acción del actor | Curso del sistema |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------|
| 1. Si el administrador desea ver gráficos generales podrá seleccionar los diversos botones como “Listar Proyectos”, “Distribución de proyectos por región”, | 2. El sistema muestra gráfico estadístico general respecto al botón seleccionado. 4. El sistema verifica opción seleccionada y |

| | |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------|
| <p>“Proporción de donaciones por categoría” y “Cantidad de participantes por proyectos”</p> <p>3. Si el administrador desea ver gráficos específicos debe de seleccionar una ODS y luego seleccionar el botón “Donadores y Voluntarios” o “Cantidad de Mujeres y Hombres”</p> | <p>muestra gráfico según ODS seleccionada por el usuario</p> |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------|

Escenarios de Caso de Uso de API Sustain Partners

i) Caso de uso: Obtener Datos Proyecto

- 5) Actores: Usuario
- 6) Propósito: El usuario podrá obtener datos de los proyectos para su uso en otros sistemas.
- 7) Precondición: -

| Acción del actor | Curso del sistema |
|--------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------|
| <p>1. Usuario realiza una petición al endpoint</p> <p>4. El usuario procede a realizar los cambios para posteriormente grabarlo en su perfil</p> | <p>2. El sistema obtiene la información de la base de datos.</p> <p>3. El sistema le da un formato a los datos(JSON).</p> |

3. Modelo Lógico

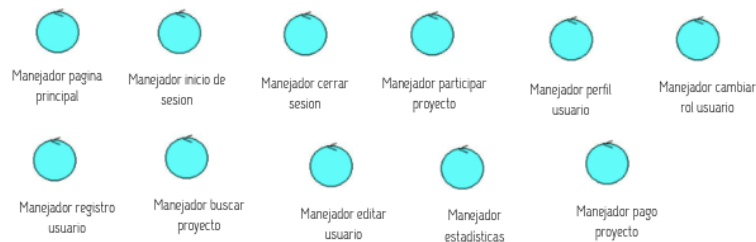
a) Análisis de Objetos

Análisis de Objetos de Aplicación web de Sustain Partners

OBJETOS FRONTERA



OBJETOS CONTROL



OBJETOS ENTIDAD



Análisis de Objetos de Aplicación de escritorio de Sustain Partners

OBJETOS FRONTERA



OBJETOS CONTROL



OBJETOS ENTIDAD



Análisis de Objetos de API Sustain Partners

OBJETOS CONTROL



Manejador Listado
Proyectos

OBJETOS ENTIDAD

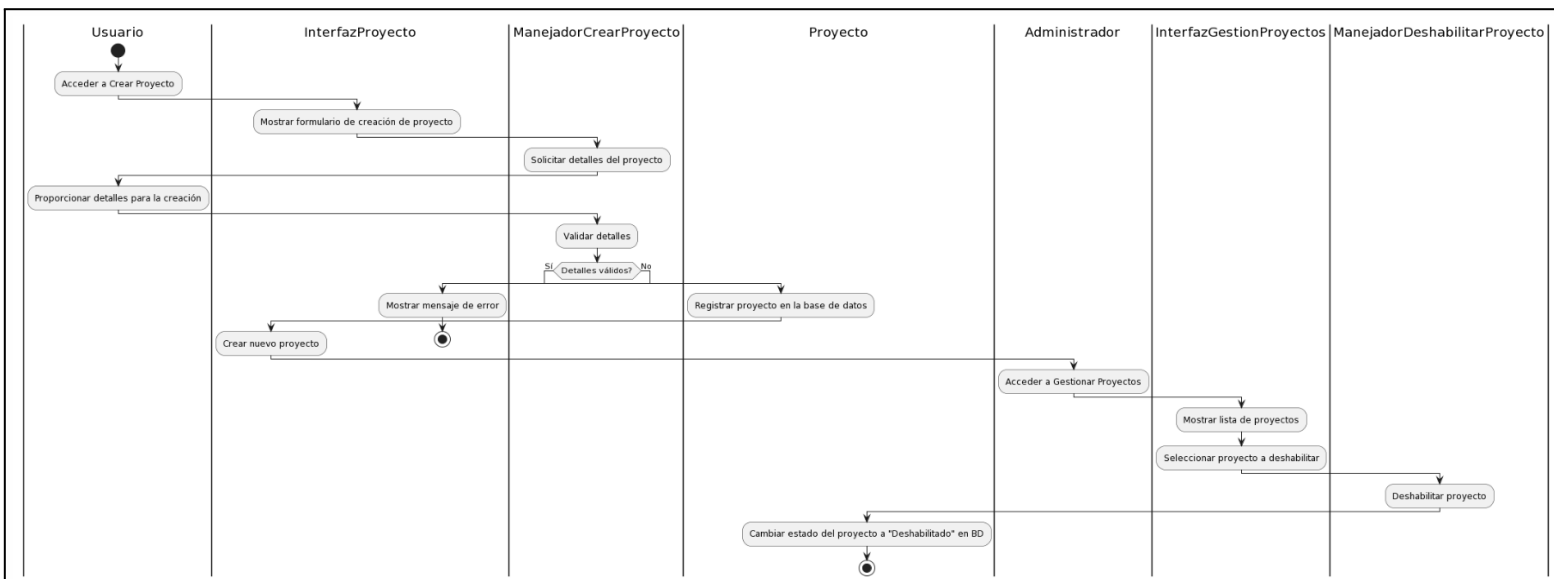


Proyecto

b) Diagrama de Actividades con objetos

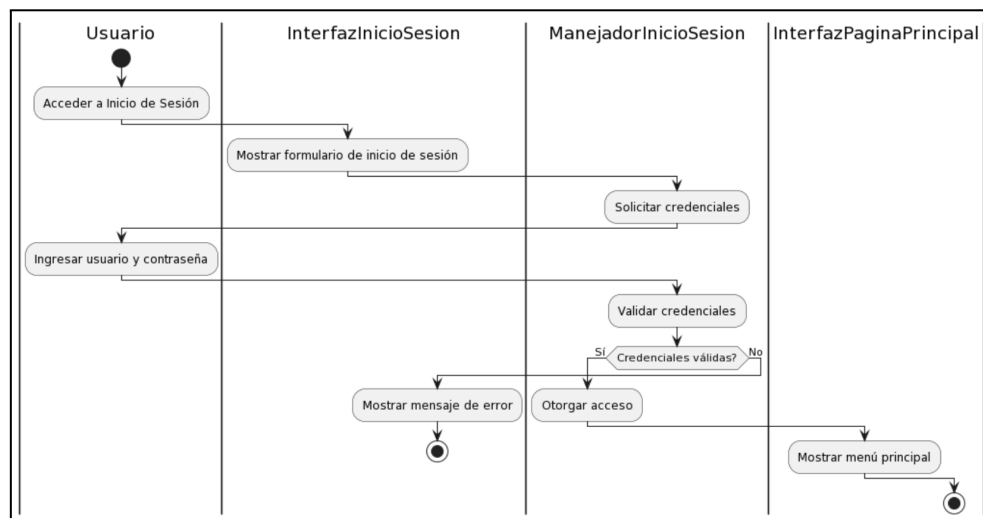
Diagrama de Actividades con Objetos de Aplicación web de Sustain Partners

- Diagrama actividades de Creación y Deshabilitación de Proyectos



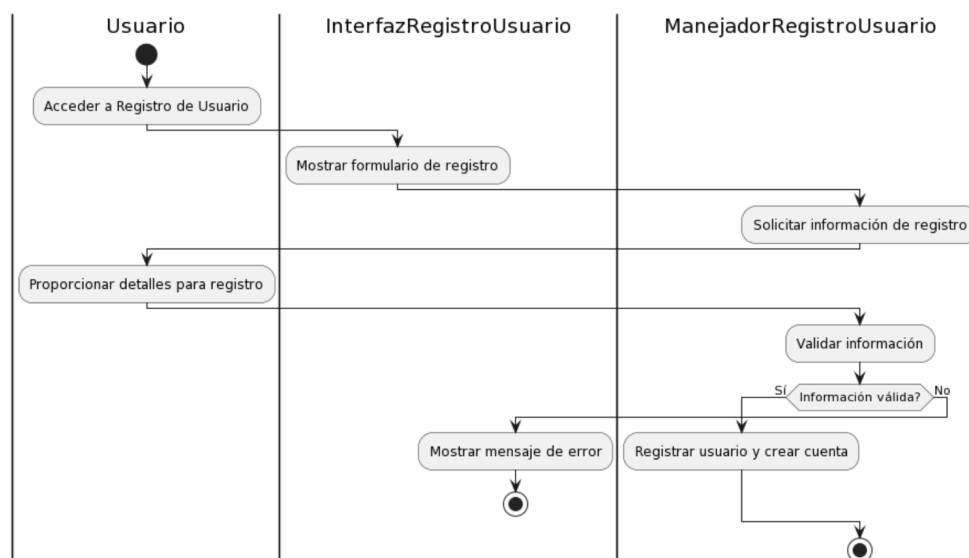
El diagrama de actividades muestra cómo los usuarios crean proyectos completando un formulario, y el sistema valida y registra la información. Los administradores pueden deshabilitar proyectos seleccionando uno de la lista, cambiando su estado a "Deshabilitado" en la base de datos.

- **Diagrama de actividades de inicio de sesión**



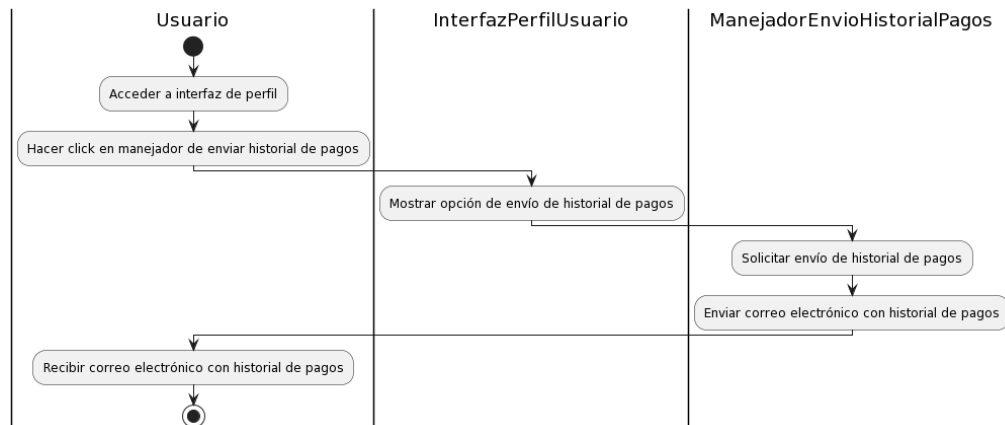
El diagrama describe el simple proceso de inicio de sesión en la plataforma. El usuario completa un formulario con sus credenciales, el sistema las valida y, si son correctas, se le concede acceso al menú principal. En caso de error, se muestra un mensaje al usuario.

- **Diagrama de actividades de registro de usuario**



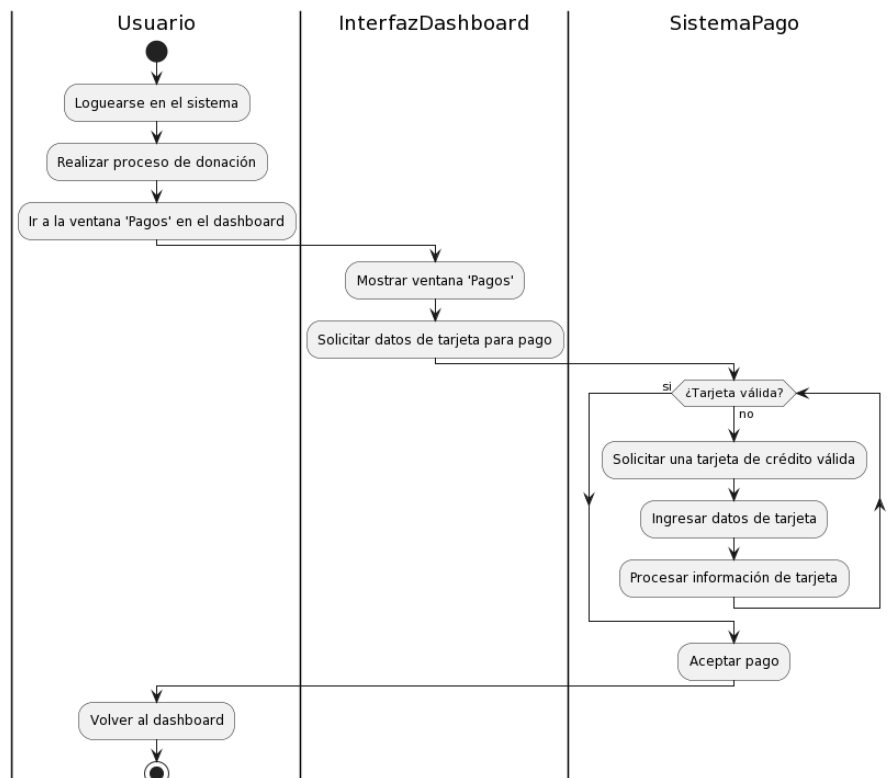
El diagrama detalla cómo los usuarios se registran en la plataforma: completan un formulario, proporcionan información, el sistema valida y crea la cuenta si es correcta. Si hay errores, se muestra un mensaje de error. Enfoca la interacción usuario-sistema durante el registro.

- **Diagrama de actividades de solicitar historial de pagos**



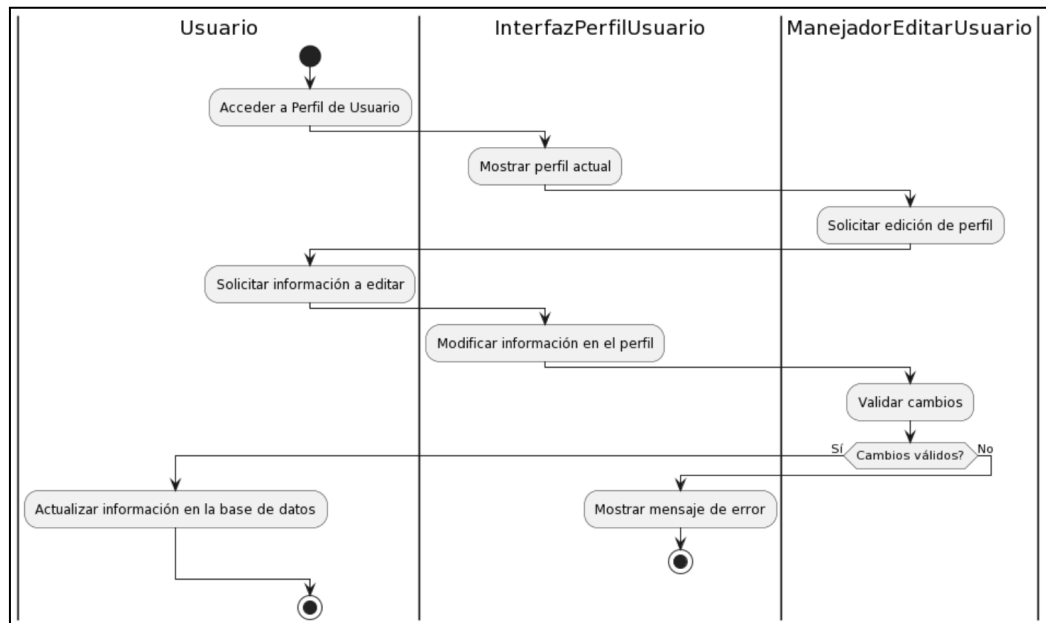
Este diagrama de actividades detalla el proceso de envío de historial de pagos que se realiza cuando el usuario solicita la interfaz de perfil y le hace click al manejador de enviar historial de pagos, el cual envía un correo electrónico al usuario.

- **Diagrama de actividades de realizar pago**



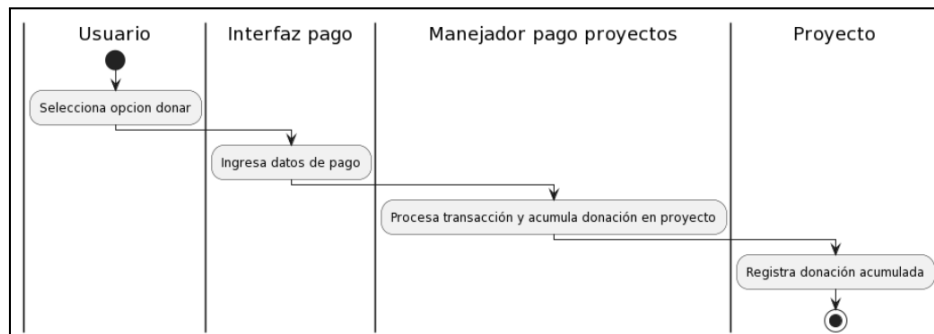
El usuario, después de donar a proyectos, accede a la sección de pagos en el panel. Se le pide información de la tarjeta para saldar deudas. Si la transacción se aprueba, vuelve al panel; si no, se solicita una tarjeta válida.

- **Diagrama de actividades de editar perfil**



El usuario edita su perfil en la plataforma: accede a su información, solicita ediciones, proporciona cambios, y si son válidos, la información se actualiza en la base de datos. Si hay errores, se muestra un mensaje.

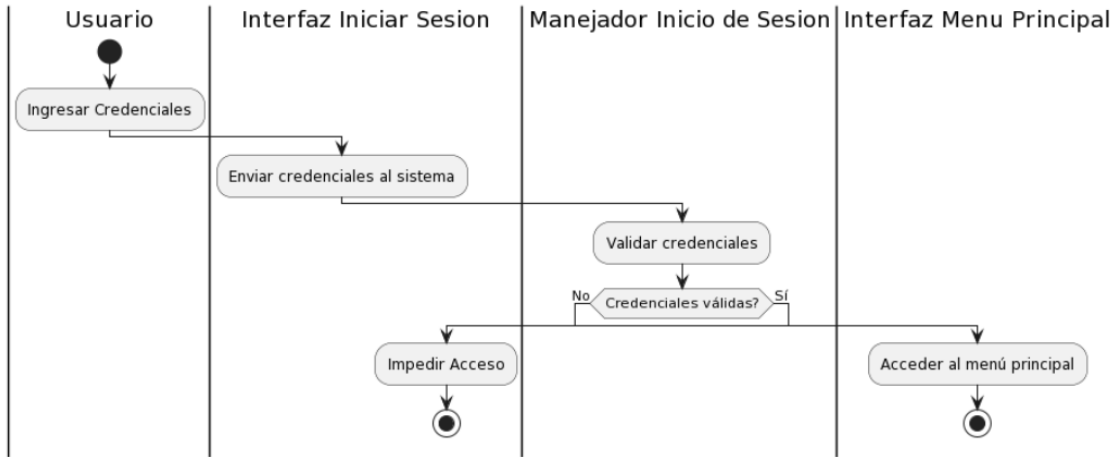
- **Diagrama de actividades de donación**



El usuario inicia el proceso de donación seleccionando la opción correspondiente, ingresando datos de pago. El manejador de pagos procesa la transacción, asegurando su éxito, y acumula la donación en los proyectos seleccionados previamente.

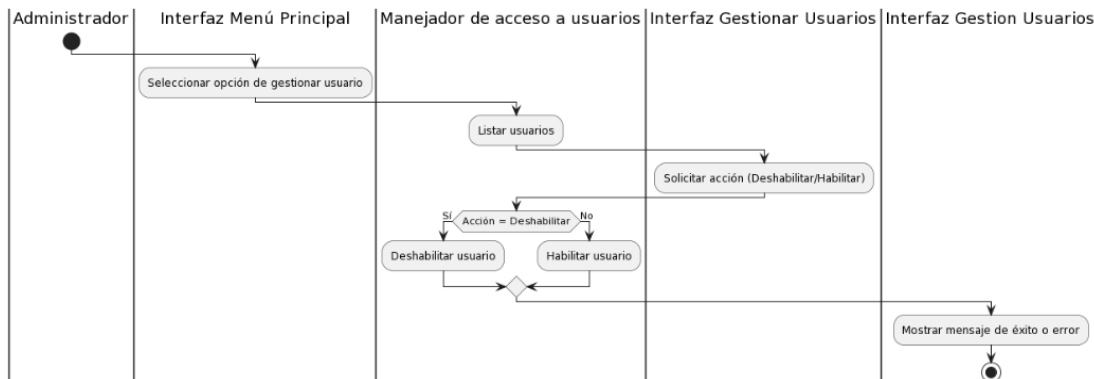
Diagrama de Actividades con Objetos de Aplicación de escritorio de Sustain Partners

- Diagrama de actividades de inicio de sesión



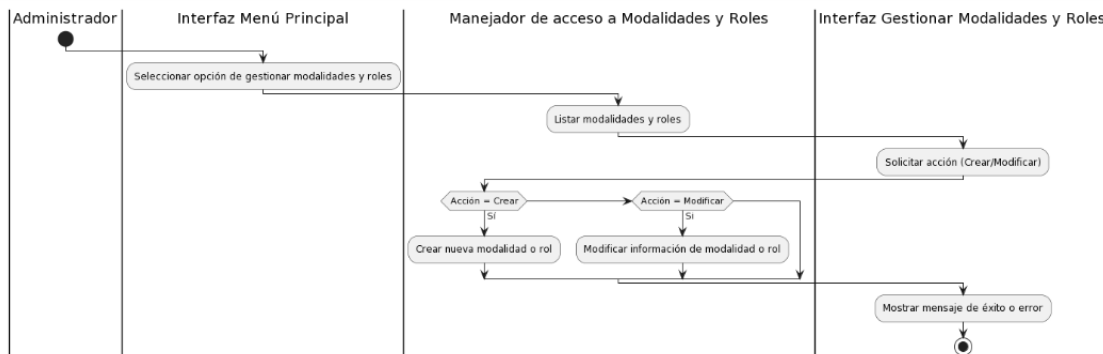
El usuario inicia sesión en la aplicación de escritorio, luego el sistema valida las credenciales y si son válidas el usuario accede al menú principal

- Diagrama de actividades de gestionar usuarios



El usuario selecciona la opción gestionar usuarios y luego se listan los usuarios, donde se podrá cambiar el estado de este (Habilitado o Deshabilitado)

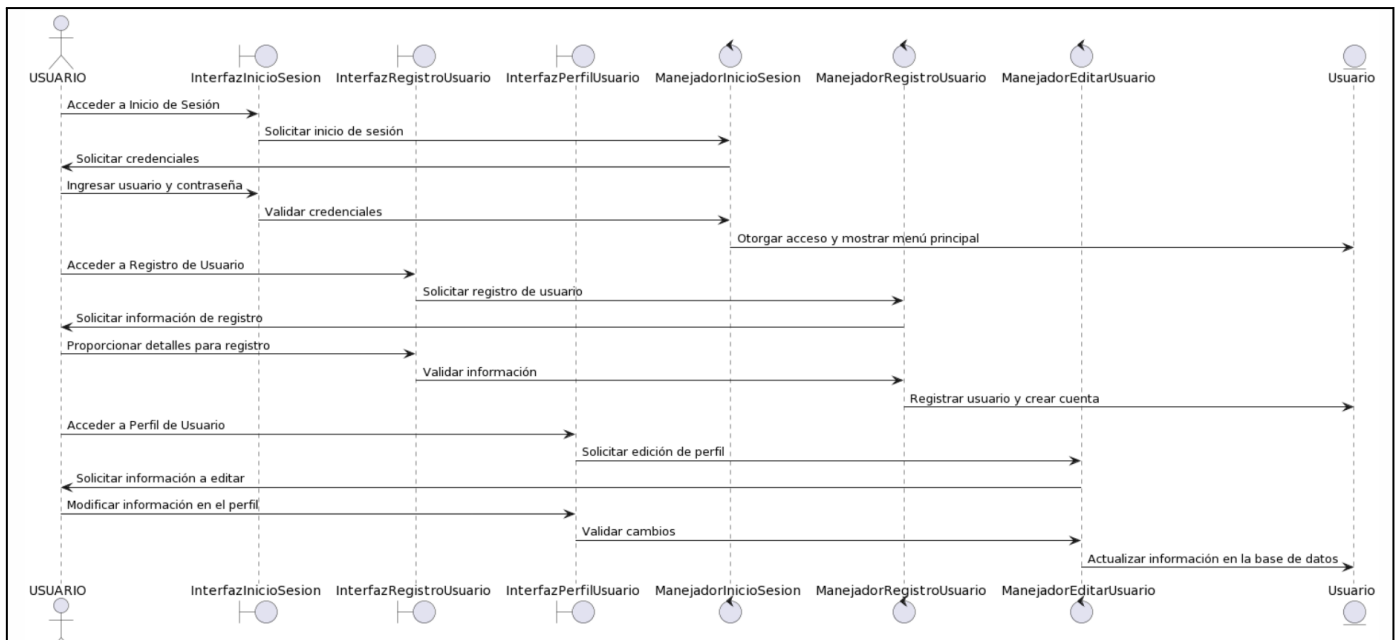
- **Diagrama de actividades de gestionar modalidades o roles**



El usuario selecciona la opción gestionar modalidades y roles posterior a esto se listarán las modalidades y roles, el usuario decidirá si desea crear o modificar algún rol o modalidad

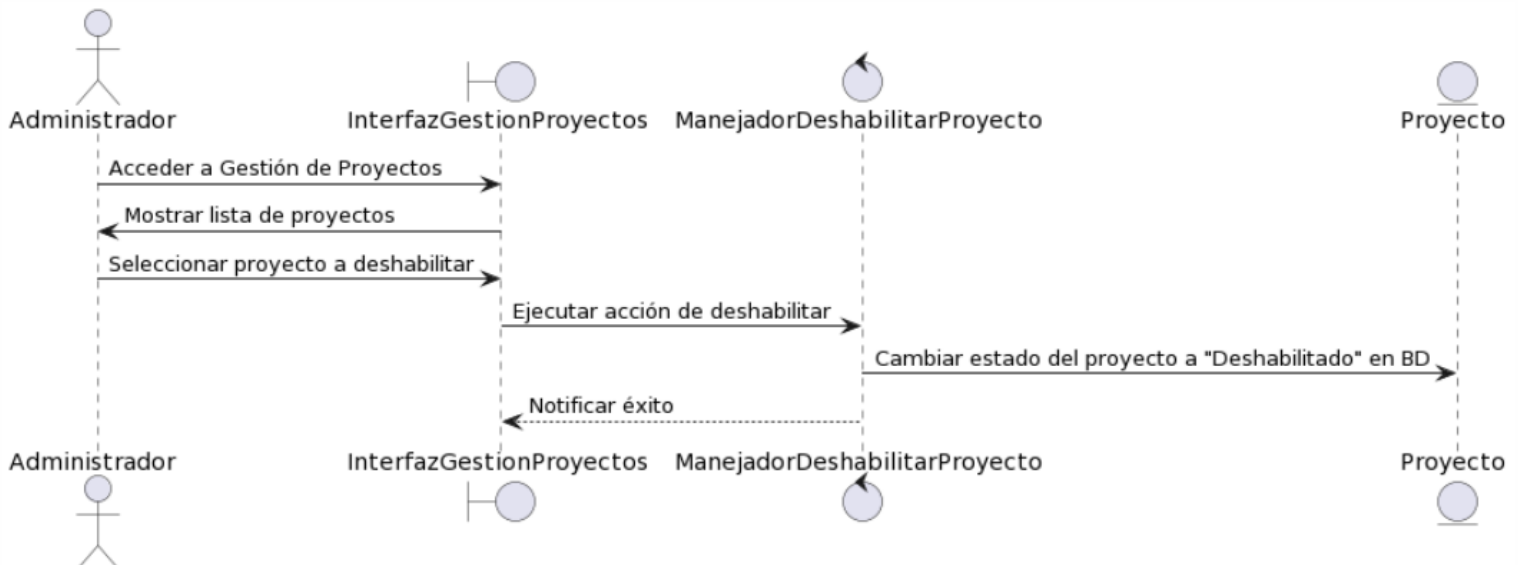
c) Diagrama de Secuencia

- **Diagrama de secuencia de inicio sesión, registro usuario y edición de usuario**



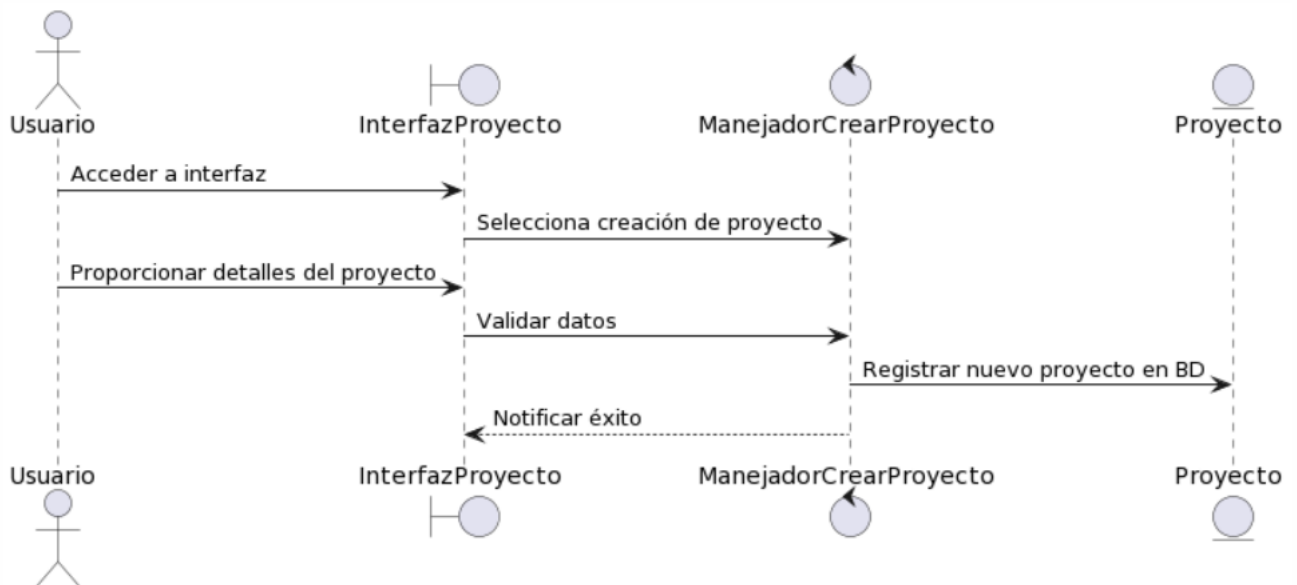
El diagrama de secuencia muestra cómo el usuario interactúa con el sistema en tres procesos: inicio de sesión, registro y edición de perfil. Se inicia con la validación de credenciales para el acceso al menú principal. En el registro, la información validada se guarda en la base de datos, y la edición de perfil permite al usuario actualizar datos, validados y almacenados en la base de datos.

- **Diagrama de secuencia creación de proyectos**



El diagrama de secuencia ilustra la creación de proyectos en la plataforma. El usuario inicia el proceso desde la interfaz del proyecto, proporciona detalles solicitados, y el sistema valida y registra la información en la base de datos. El usuario recibe una notificación de éxito. Este diagrama sintetiza la interacción eficientemente durante la creación de proyectos.

- **Diagrama de secuencia de deshabilitar proyecto**



El diagrama de secuencia representa la deshabilitación de proyectos por parte del administrador en la plataforma. Inicia con el administrador accediendo a la gestión de proyectos, seleccionando un proyecto específico, y activando el proceso de deshabilitación. El manejador correspondiente ejecuta la acción, actualizando el estado del proyecto en la base de datos a "Deshabilitado".

```

sequenceDiagram
    actor U as Usuario
    participant IP as Interfaz Pago
    participant MP as Manejador Pago Proyectos
    participant P as Proyecto

    U->>IP: Selecciona opción donar
    IP->>MP: Ingresa datos de pago
    MP->>P: Procesa transacción y acumula donación
  
```

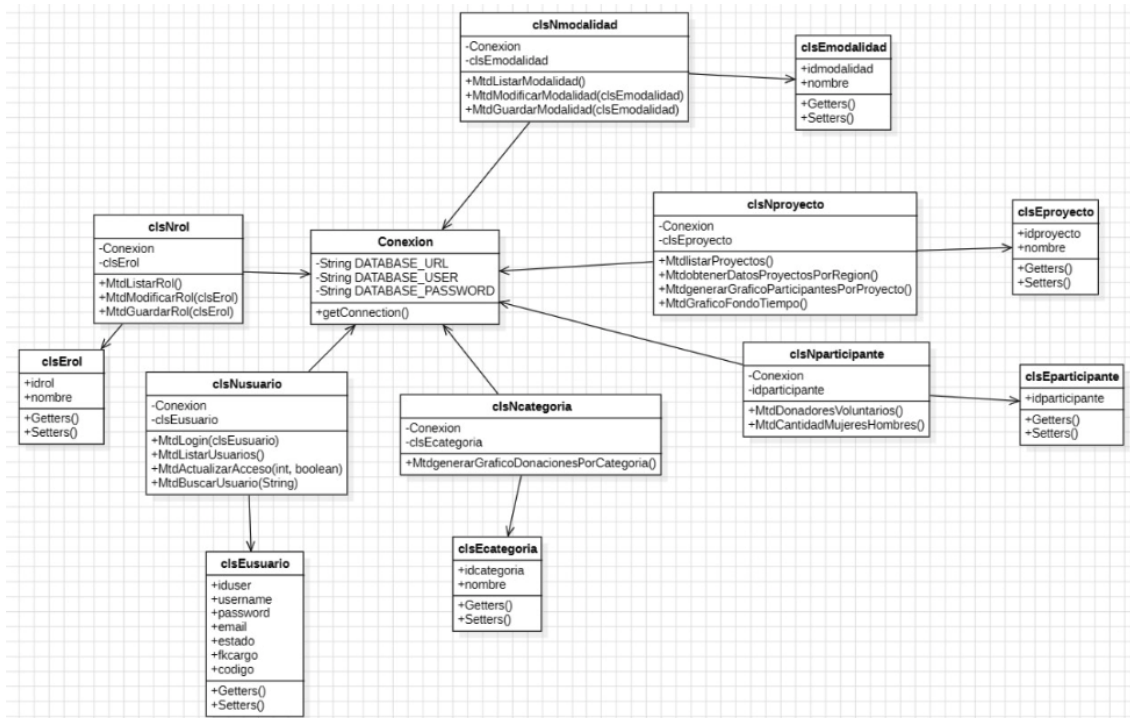
d) Diagrama de Clases

```

classDiagram
    class BoletaController {
        +Usuario
        +ndoGet(HttpServletRequest, HttpServletResponse)
        +enviarBoleta(HttpServletRequest, HttpServletResponse)
    }
    class UsuarioController {
        +UsuarioDao
        +ndoGet(HttpServletRequest, HttpServletResponse)
        +ndoPost(HttpServletRequest, HttpServletResponse)
    }
    class PagoDAO {
        +NeoConnection
        +Pago
        +obtenerPago()
        +crearPago()
        +obtenerDeuda()
    }
    class PagoController {
        +PagoDAO
        +ndoGet(HttpServletRequest, HttpServletResponse)
        +ndoPost(HttpServletRequest, HttpServletResponse)
    }
    class Participante {
        +idParticipante
        +idUsuario
        +Activo
        +idProyecto
        +getters()
        +setters()
    }
    class ParticiparController {
        +ProyectoDAO
        +ParticiparDAO
        +ndoGet(HttpServletRequest, HttpServletResponse)
        +ndoPost(HttpServletRequest, HttpServletResponse)
    }
    class ProyectoDAO {
        +NeoConnection
        +Proyecto
        +RedisConnection
        +obtenerProyecto(int)
        +obtenerParticipantesProyecto(int)
        +obtenerMonitoProyecto(int)
        +obtenerProyectoById(int)
        +obtenerNumeroCondoresVoluntarios(int)
        +crearProyecto(Proyecto, int)
        +obtenerProyectoByIdPersonal(int)
        +buscarProyectos(String)
        +cambiarEstadoProyecto(int, boolean)
        +obtenerCategoriasProyecto(int)
        +obtenerModalidadesProyecto(int)
    }
    class ProyectoController {
        +ProyectoDAO
        +ndoGet(HttpServletRequest, HttpServletResponse)
        +ndoPost(HttpServletRequest, HttpServletResponse)
    }
    class SearchBarController {
        +ProyectoDAO
        +ndoGet(HttpServletRequest, HttpServletResponse)
        +ndoPost(HttpServletRequest, HttpServletResponse)
    }
    class PerfilController {
        +ProyectoDAO
        +ndoGet(HttpServletRequest, HttpServletResponse)
    }
    class EstadisticasController {
        +ProyectoDAO
        +ndoGet(HttpServletRequest, HttpServletResponse)
    }
    class DashboardController {
        +ProyectoDAO
        +PersonaDAO
        +ndoGet(HttpServletRequest, HttpServletResponse)
        +establecerUsuario(HttpServletRequest)
        +establecerProyecto(HttpServletRequest)
    }
    class PersonaController {
        +PersonaDAO
        +ndoGet(HttpServletRequest, HttpServletResponse)
        +ndoPost(HttpServletRequest, HttpServletResponse)
    }
    class RecuperarController {
        +UsuarioDAO
        +ndoGet(HttpServletRequest, HttpServletResponse)
        +ndoPost(HttpServletRequest, HttpServletResponse)
        +enviarCodigo(HttpServletRequest, HttpServletResponse)
        +validarCodigo(HttpServletRequest, HttpServletResponse)
        +cambiarContraseña(HttpServletRequest, HttpServletResponse)
    }
    class LoginController {
        +UsuarioDAO
        +String SecretKey
        +ndoGet(HttpServletRequest, HttpServletResponse)
        +ndoPost(HttpServletRequest, HttpServletResponse)
        +verify(String, String)
    }
    class RegisterController {
        +UsuarioDAO
        +ndoGet(HttpServletRequest, HttpServletResponse)
        +ndoPost(HttpServletRequest, HttpServletResponse)
    }
    class LogoutController {
        +ndoGet(HttpServletRequest, HttpServletResponse)
    }
    class Usuario {
        +idUser
        +username
        +password
        +email
        +estado
        +fechaReg
        +Crear()
        +Setters()
    }
    class UsuarioDAO {
        +NeoConnection
        +authenticate(String, String)
        +actualizarRolUsuario(int)
        +obtenerUsuario(int)
        +obtenerUsuarios(int)
        +crearUsuario(Usuario)
        +actualizarUsuario(Usuario)
        +buscarUsuarios(String)
    }
    class Pago {
        +idPago
        +idUsuario
        +monto
        +fechaPago
        +Getters()
        +Setters()
    }
    class ParticiparDAO {
        +NeoConnection
        +crearParticipante(Participante, int)
    }
    class Proyecto {
        +idProyecto
        +nombre
        +descripcion
        +objetivo
        +foto
        +estado
        +idRegion
        +idUser
        +idUser
        +Getters()
        +Setters()
    }
    class Persona {
        +idPersona
        +nombre
        +apellido
        +celular
        +fechaNacimiento
        +fotoPersona
        +sexo
        +idUser
        +Getters()
        +Setters()
    }
    class PersonaDAO {
        +NeoConnection
        +RedisConnection
        +obtenerPersona(int)
        +obtenerPersonas(int)
        +crearPersona(Persona)
        +actualizarPersona(Persona)
    }
    class ProyectoDAO {
        +NeoConnection
        +Proyecto
        +RedisConnection
        +obtenerProyecto(int)
        +obtenerParticipantesProyecto(int)
        +obtenerMonitoProyecto(int)
        +obtenerProyectoById(int)
        +obtenerNumeroCondoresVoluntarios(int)
        +crearProyecto(Proyecto, int)
        +obtenerProyectoByIdPersonal(int)
        +buscarProyectos(String)
        +cambiarEstadoProyecto(int, boolean)
        +obtenerCategoriasProyecto(int)
        +obtenerModalidadesProyecto(int)
    }
    class ParticiparDAO {
        +NeoConnection
        +crearParticipante(Participante, int)
    }
    class UsuarioDAO {
        +NeoConnection
        +authenticate(String, String)
        +actualizarRolUsuario(int)
        +obtenerUsuario(int)
        +obtenerUsuarios(int)
        +crearUsuario(Usuario)
        +actualizarUsuario(Usuario)
        +buscarUsuarios(String)
    }
    class NeoConnection {
        +String DATABASE_URL
        +String DATABASE_USER
        +String DATABASE_PASSWORD
        +getConnection()
    }
    class RedisConnection {
        +Jedis
        +getConnection()
    }
    BoletaController --> UsuarioController
    UsuarioController --> UsuarioDAO
    UsuarioController --> PagoDAO
    PagoDAO --> PagoController
    PagoController --> Participante
    Participante --> ParticiparController
    ParticiparController --> ProyectoDAO
    ProyectoDAO --> ProyectoController
    ProyectoController --> SearchBarController
    SearchBarController --> PerfilController
    PerfilController --> EstadisticasController
    EstadisticasController --> DashboardController
    DashboardController --> PersonaController
    PersonaController --> RecuperarController
    RecuperarController --> LoginController
    LoginController --> RegisterController
    RegisterController --> LogoutController
    Usuario --> UsuarioDAO
    Pago --> PagoDAO
    Participante --> ParticiparDAO
    Proyecto --> ProyectoDAO
    Persona --> PersonaDAO
    
```

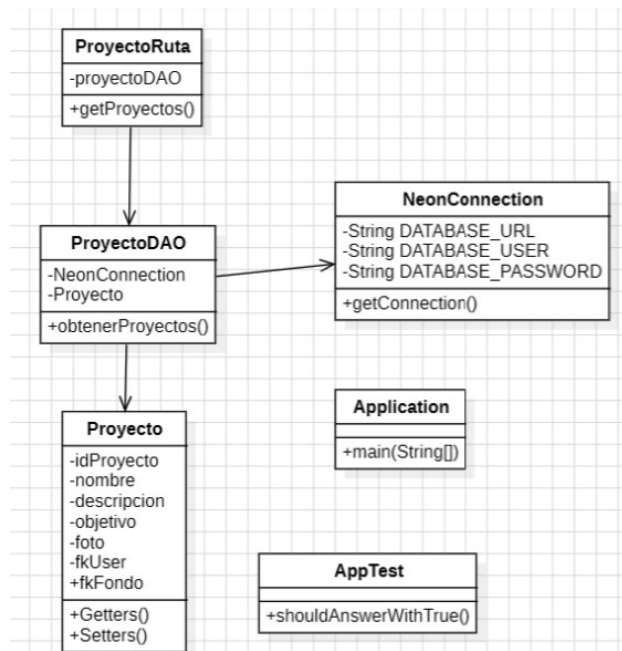
37

Diagrama de Clases de Aplicación de escritorio de Sustain Partners



Este diagrama de clases muestra las relaciones y dependencias que existen en la aplicación de escritorio, junto a todas las clases y sus respectivos métodos.

Diagrama de Clases de API Sustain Partners



Este diagrama de clases muestra las relaciones y dependencias que existen en la API de Sustain Partners, junto a todas las clases y sus respectivos métodos.