

DIAGRAMAS

**JUANITA MOSQUERA SANTOS
MARIA JOSE MIRA MARTINEZ**

**SERVICIO NACIONAL DE APRENDIZAJE SENA
ANÁLISIS Y DESARROLLO DE SOFTWARE**

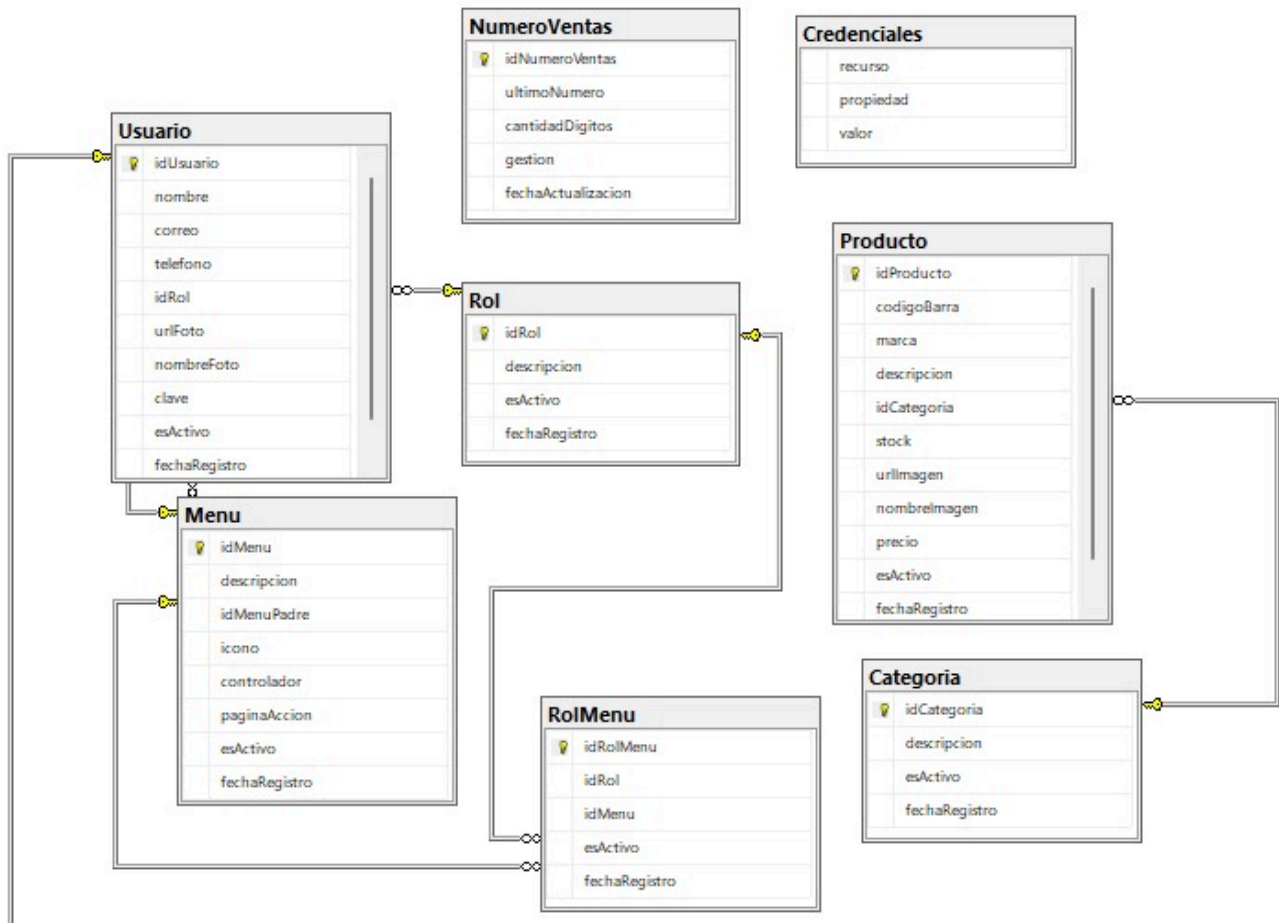
(2694526)

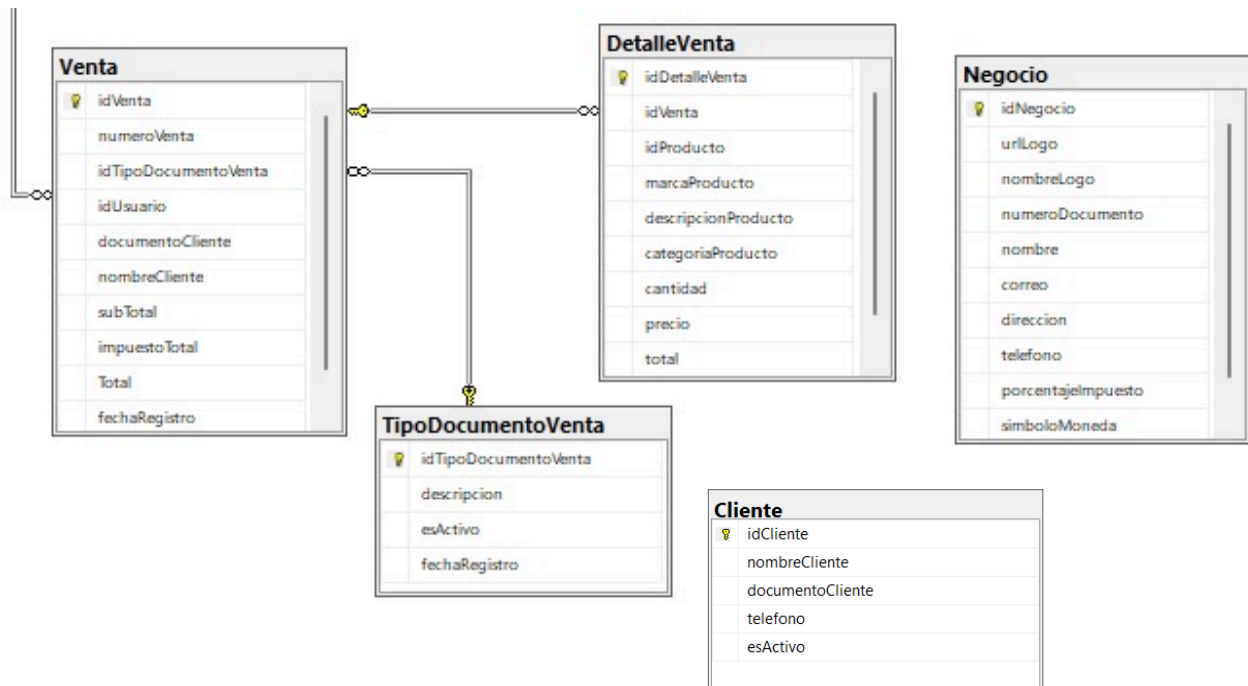
2024

MODELO RELACIONAL

Este documento presenta un modelo relacional diseñado para representar y gestionar los datos asociados con un sistema de inventarios de cerveza. El modelo se basa en los principios del diseño de bases de datos relacionales, que permiten la estructuración organizada y coherente de la información, facilitando su almacenamiento, recuperación y manipulación.

El sistema de inventarios de cerveza abarca una amplia gama de datos, que incluyen información sobre productos, proveedores, pedidos, inventarios actuales, estadísticas y más. El modelo relacional propuesto define las relaciones entre estas entidades y establece las reglas de integridad necesarias para garantizar la consistencia y la precisión de los datos.

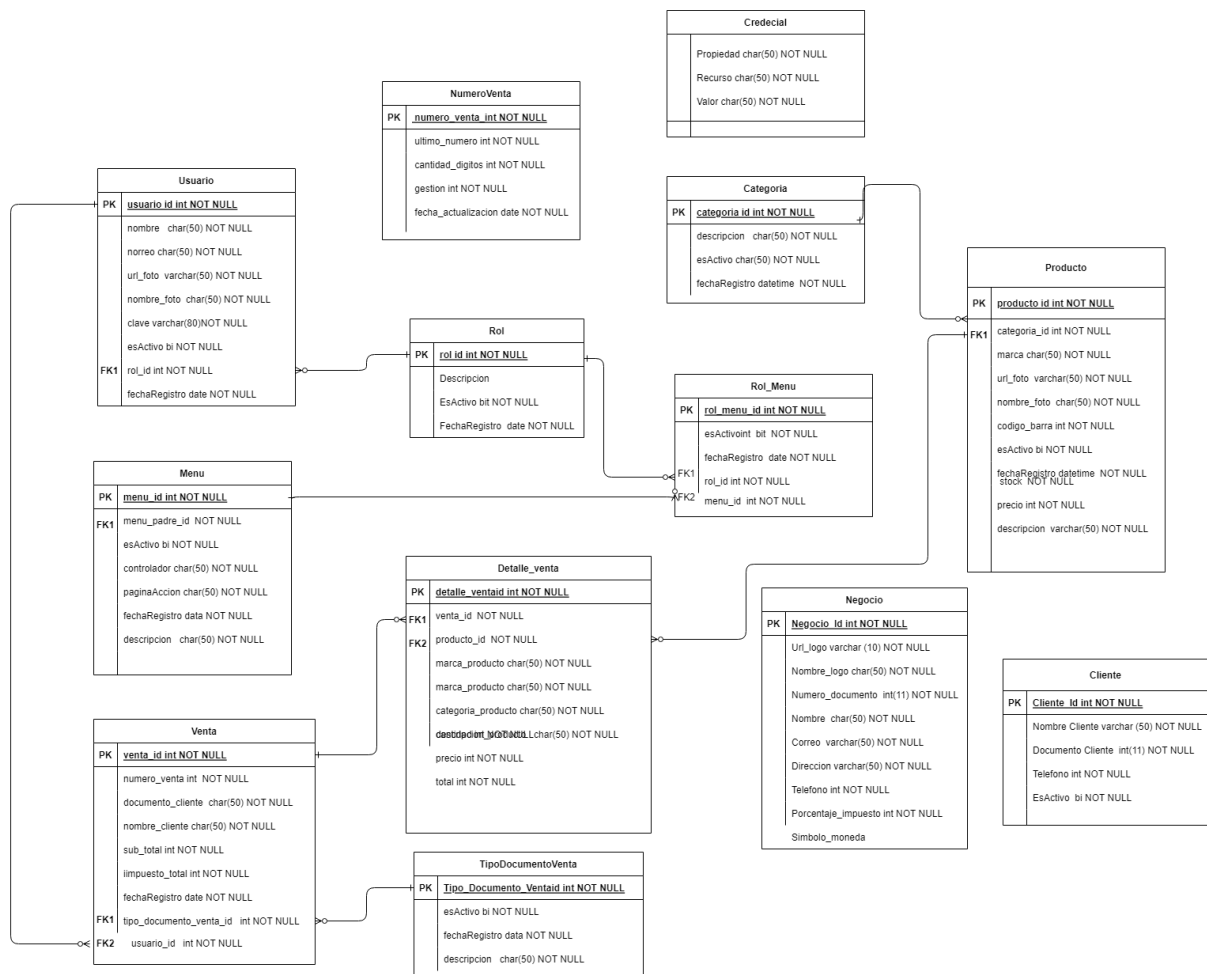




MODELO ENTIDAD RELACIÓN

Este documento presenta un modelo entidad-relación diseñado para representar las diferentes entidades y sus relaciones en un sistema de inventarios de cerveza. El modelo entidad-relación es una herramienta visual que ayuda a comprender la estructura y las interacciones entre las entidades clave en un sistema de base de datos.

En el contexto de un sistema de inventarios de cerveza, las entidades principales pueden incluir productos de cerveza, proveedores, pedidos, cajas de cerveza en almacenamiento. Estas entidades están interrelacionadas de diversas maneras, reflejando los flujos de datos y las operaciones comerciales en el proceso de gestión de inventarios.



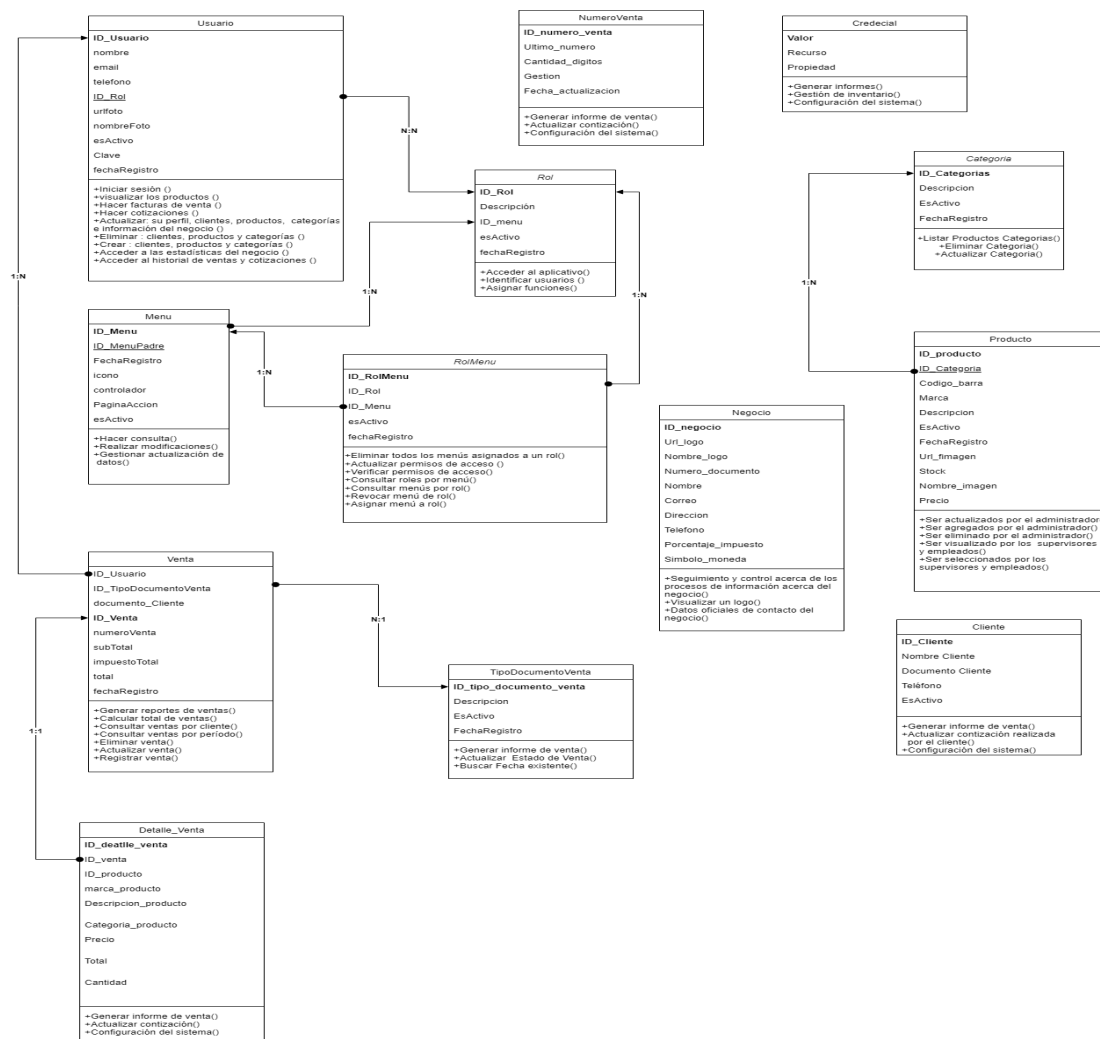
→

https://drive.google.com/file/d/1N73MV_BCapK7gtjyXb1MWkG74OicVPQz/view?usp=s
haring

DIAGRAMA DE CLASE

Este documento presenta un modelo de clases diseñado específicamente para la gestión de inventarios de cerveza. El modelo de clases es una herramienta fundamental en el diseño orientado a objetos que permite organizar y representar las entidades relevantes en el dominio del problema, así como las relaciones y comportamientos asociados con ellas.

El modelo de clases propuesto identifica las clases principales y define sus atributos y métodos distintivos. Además, establece las relaciones entre estas clases, como la composición, la agregación y la asociación, lo que permite modelar con precisión las interacciones entre las entidades en el sistema de inventarios de cerveza.

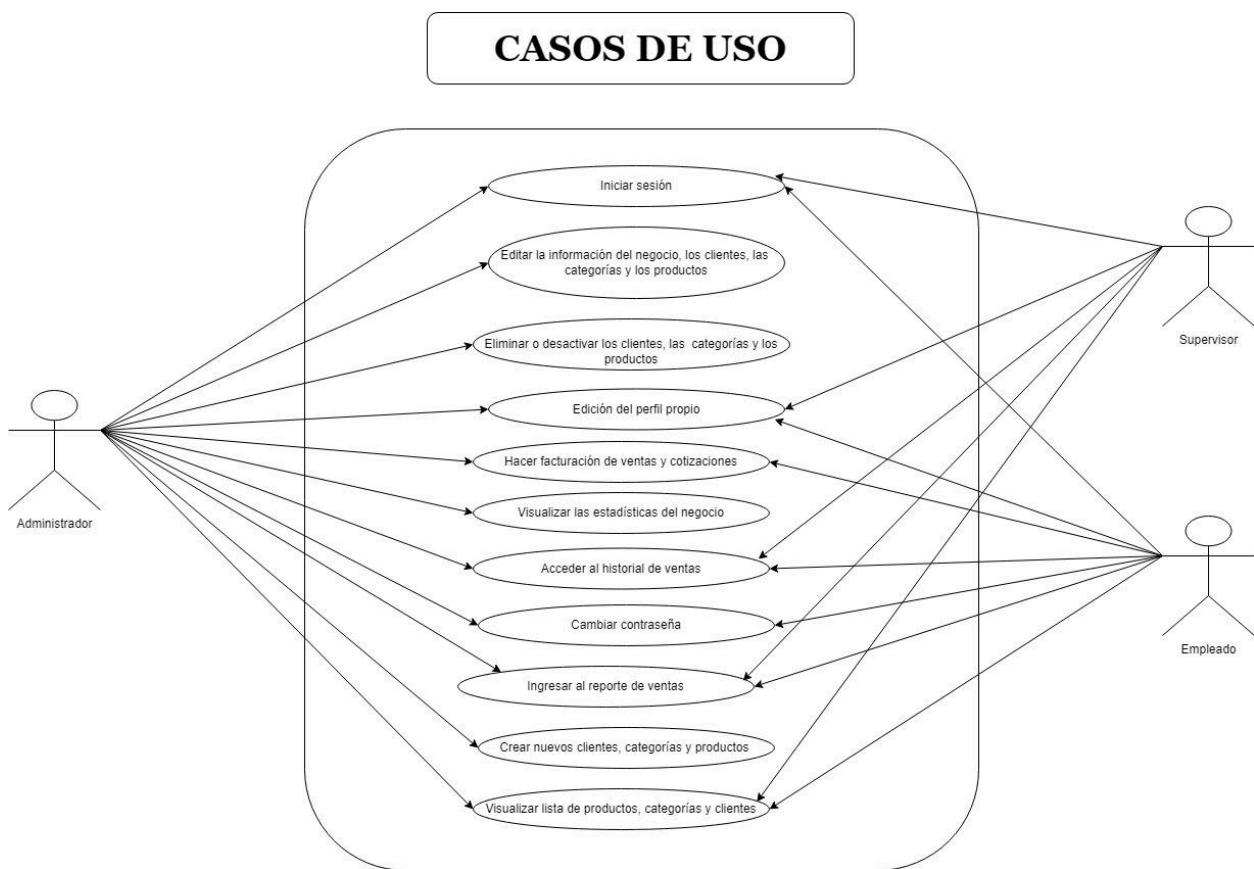


<https://app.diagrams.net/#G1qgVYHjVMcbVqel6lNwCOEYuu1NHg3QlIf>

DIAGRAMA DE CASO DE USO

Diagrama de clases del proyecto, por esta razón se ha tomado como requerimientos funcionales del software y las especificaciones del usuario para el desarrollo del Diagrama Caso de Uso en su clasificación de Diagrama de Clases, con el objetivo de trazar claramente la estructura de dicho sistema al modelar sus clases y métodos de ejecución durante el proceso en el aplicativo web.

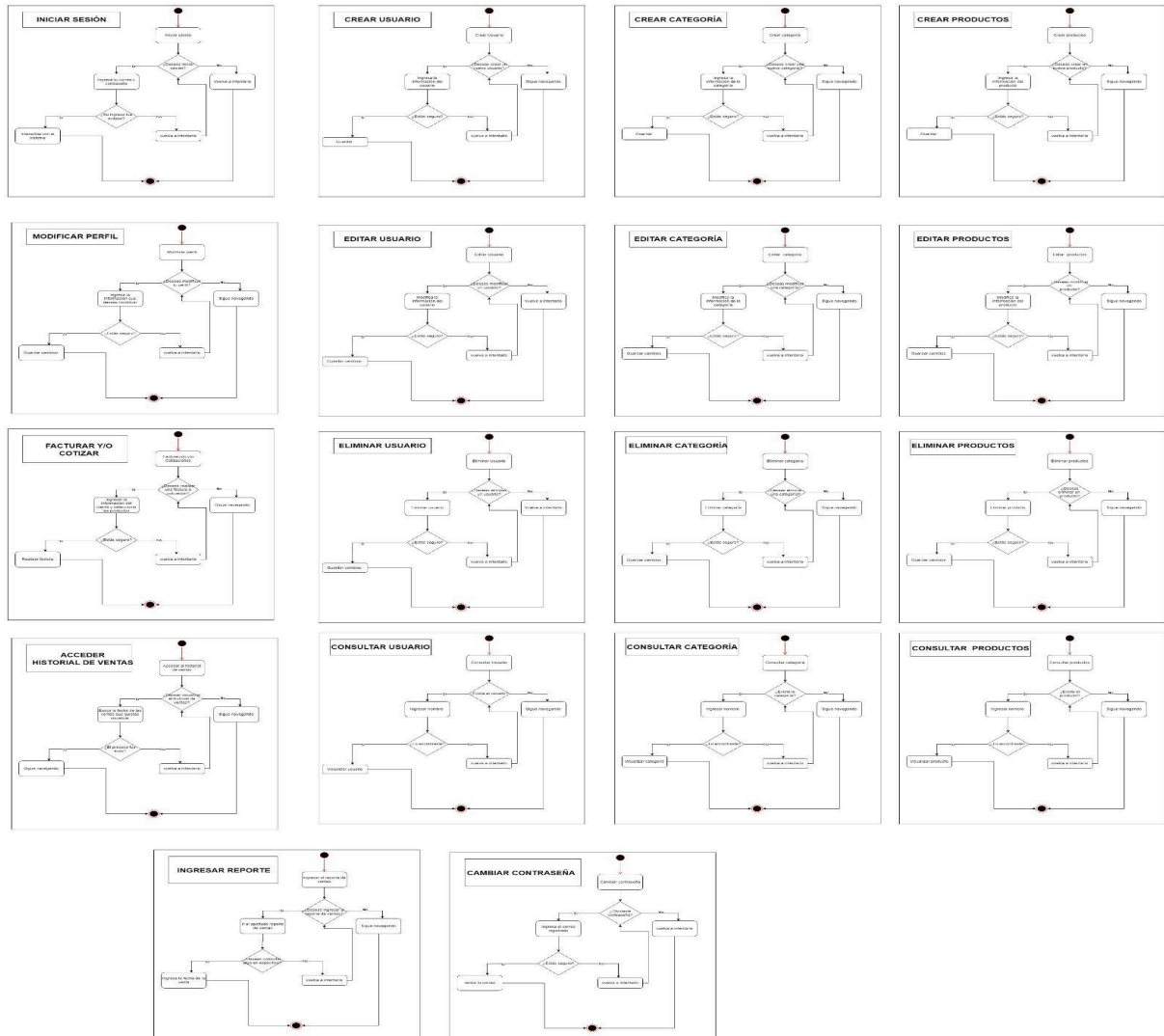
Lo anterior, se refiere al proceso de descripción textual de cada uno de estos detallando los flujos y eventos que interactúan con el sistema y los actores que participan en él. Usualmente el analista, que es la persona encargada del diseño de los casos de uso a medida es quien va identificando los requerimientos y los va documentando.



<https://drive.google.com/file/d/1OeoXBS2YqUJEloFLfNzqDzr2DRTXnds/view?usp=sharing>

DIAGRAMA DE ACTIVIDADES

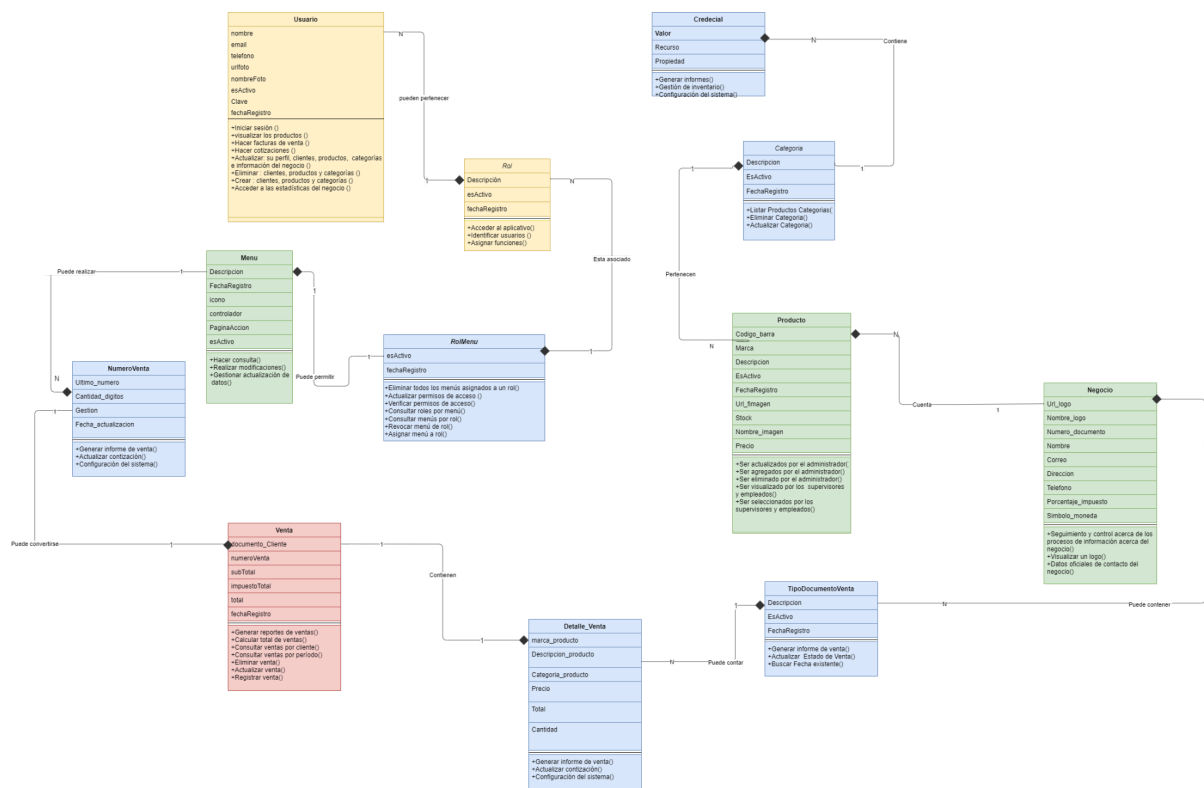
Representación visual que muestra el flujo de acciones o tareas en un proceso o sistema. Ayuda a entender el orden de las actividades y las relaciones entre ellas, facilitando la comunicación y la identificación de mejoras en el proceso. Se utiliza en ingeniería de software y otros campos para modelar sistemas y procesos.



https://drive.google.com/file/d/1P627w_aaUxmZJvtVbhy-vaXefUv5yMtk/view?usp=sharing

MODELO DE DOMINIO

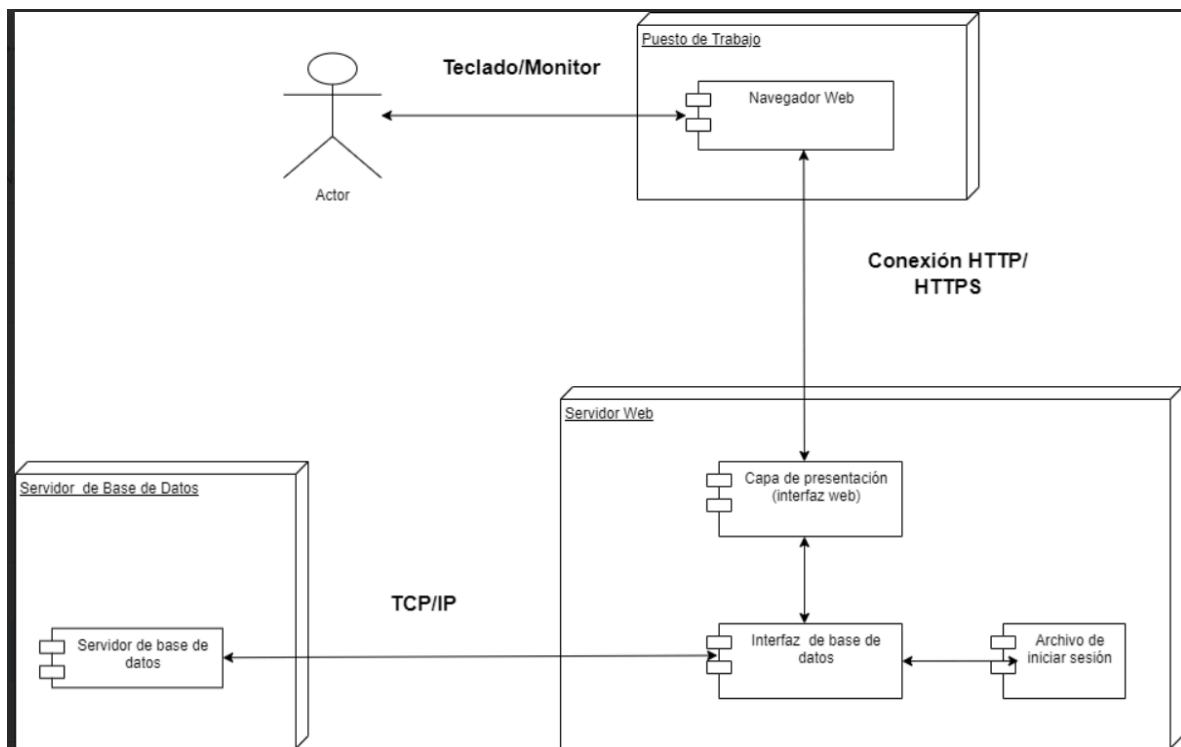
Una representación visual de las entidades fundamentales y sus relaciones en un dominio específico de aplicación. Utiliza diagramas de clases para mostrar las clases principales, sus atributos y métodos, así como las asociaciones entre ellas. Este modelo proporciona una vista estructurada y clara del dominio del problema, ayudando a los desarrolladores a comprender mejor los conceptos clave y las interacciones dentro del sistema. El modelo de dominio en UML es una herramienta crucial en el análisis y diseño de sistemas de información, ya que establece una base sólida para el desarrollo de software al identificar y organizar las entidades relevantes y sus relaciones.



https://drive.google.com/file/d/1P0Ejk_jeWEq9y9aZxKR72C7fpagzYWM0/view?usp=sharing

DIAGRAMA DESPLIEGUE

El diagrama de despliegue desempeña un papel fundamental en el desarrollo de software, ya que proporciona una representación visual de cómo los distintos componentes del sistema interactúan y se distribuyen físicamente en el entorno de ejecución. Esto resulta crucial para comprender la arquitectura física del sistema, permitiendo una planificación más precisa de los recursos necesarios, tanto en términos de hardware como de software. Al identificar la ubicación de cada componente, los equipos de desarrollo pueden anticipar y mitigar posibles cuellos de botella en la red o en los recursos de hardware, así como identificar puntos únicos de falla que podrían comprometer la disponibilidad y el rendimiento del sistema. Además, este diagrama facilita la integración de nuevos componentes y el mantenimiento continuo del sistema a lo largo del tiempo, ya que proporciona una visión clara de cómo se conectan entre sí los diversos elementos del sistema.



https://app.diagrams.net/#G1rgjRFInut8_P40E-9WyK-2IX52FLhJ_p#%7B%22pageld%22%3A%22LVe3ZiUKt4aBDEtxLXB3%22%7D%7D