



Contenido

[1. Introducción 2](#_gjdgxs)

[1.1 Propósito 2](#_30j0zll)

[1.2 Alcance 2](#_1fob9te)

[1.3 Definiciones, acrónimos y abreviaturas 2](#_3znysh7)

[1.4 Referencias (Pendiente) 2](#_2et92p0)

[2. Representación arquitectural 3](#_tyjcwt)

[2.1 Metas y limitaciones arquitectónicas 3](#_3dy6vkm)

[2.2 Vista de casos de uso 3](#_1t3h5sf)

[2.3 Descripción general de la arquitectura: capaz de paquetes y subsistemas. 4](#_4d34og8)

[2.4 Vista del proceso 4](#_2s8eyo1)

[2.5. Vista despliegue 7](#_17dp8vu)

# 1. Introducción

## Propósito

Este documento proporciona una descripción arquitectónica integral del sistema "Donde Deme Otra", un gestor de ventas diseñado para facilitar la administración de ventas y usuarios a los dueños de emprendimientos que se dedican a la venta de desayunos, tales como churrascas y sopaipillas. Utilizando diversas vistas arquitectónicas, se busca capturar y comunicar las decisiones clave que guían el desarrollo del sistema.

## Alcance

El documento abarca la arquitectura del sistema desarrollado por Miguel Arias, José Guerra y Nicole Ulloa para "Donde Deme Otra" (DOO). Se ha generado directamente a partir del modelo de diseño y análisis del sistema, utilizando la plantilla de documento de arquitectura de software.

## Definiciones, acrónimos y abreviaturas

Las definiciones, acrónimos y abreviaturas son las siguientes

| DOO | Donde deme otra |
| --- | --- |
| CRUD | Create (Creación), Read (Lectura), Update (Modificación) y Delete (Eliminación). |
|  |  |
|  |  |
|  |  |

## 

# 2. Representación arquitectural

## 2.1 Metas y limitaciones arquitectónicas

Las metas y limitaciones clave del sistema incluyen:

* **Escalabilidad:** El sistema debe ser capaz de manejar un aumento en el número de usuarios y transacciones sin degradar el rendimiento.
* **Seguridad:** Asegurar que los datos de los usuarios y transacciones están protegidos contra accesos no autorizados.
* **Usabilidad:** El sistema debe ser intuitivo y fácil de usar para los propietarios de los emprendimientos.
* **Mantenibilidad:** La arquitectura debe permitir la fácil actualización y mantenimiento del sistema.

## 2.2 Vista de casos de uso

La vista de casos de uso se centra en los escenarios y funcionalidades centrales del sistema. Los casos de uso incluyen:

* **Gestión de ventas:** Permite a los usuarios registrar y gestionar ventas diarias.
* **Administración de usuarios:** Facilita el registro y gestión de usuarios del sistema.
* **Reportes de ventas:** Genera reportes detallados sobre las ventas realizadas en un periodo determinado.
* **Notificaciones:** Envía notificaciones a los usuarios sobre ventas, actualizaciones y eventos importantes.

**Ejemplos de casos de uso:**

* **Registro de venta:** Un usuario puede registrar una nueva venta, especificando detalles como el producto vendido, cantidad y precio.
* **Generación de reporte:** El usuario puede generar un reporte de ventas para un periodo específico.
* **Notificación de ventas:** El sistema envía una notificación al usuario con el resumen de las ventas diarias.

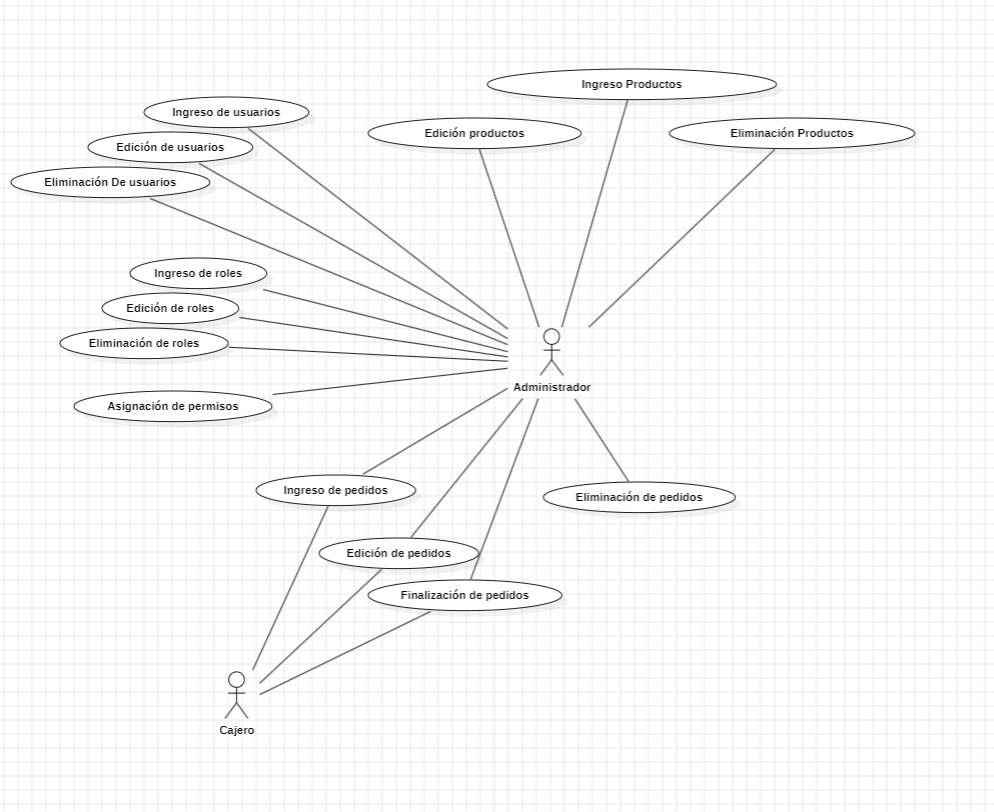


Figura 1: Caso de uso de “The Oven Wizard”.

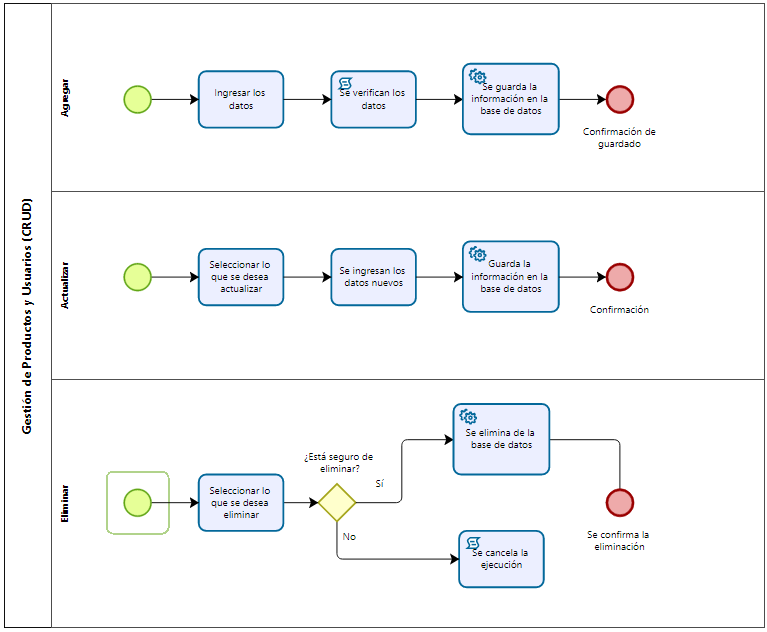
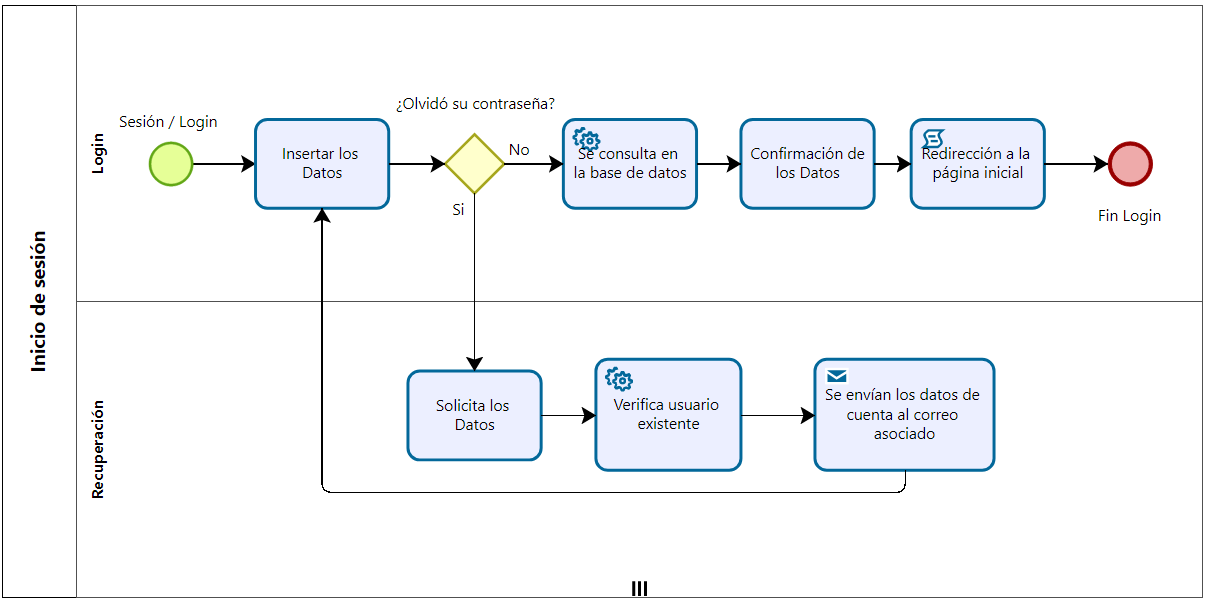
## 2.3 Descripción general de la arquitectura: capaz de paquetes y subsistemas.

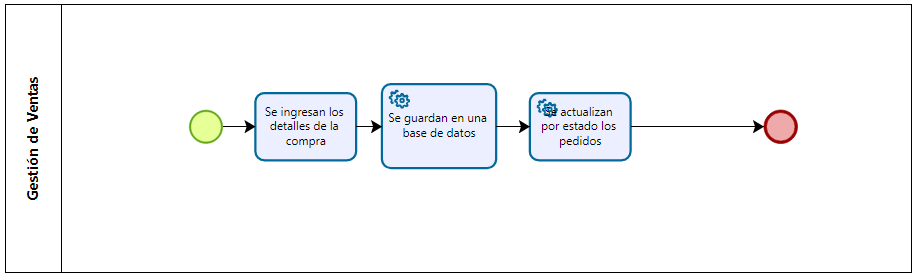
La arquitectura del sistema se divide en varias capas y subsistemas que incluyen:

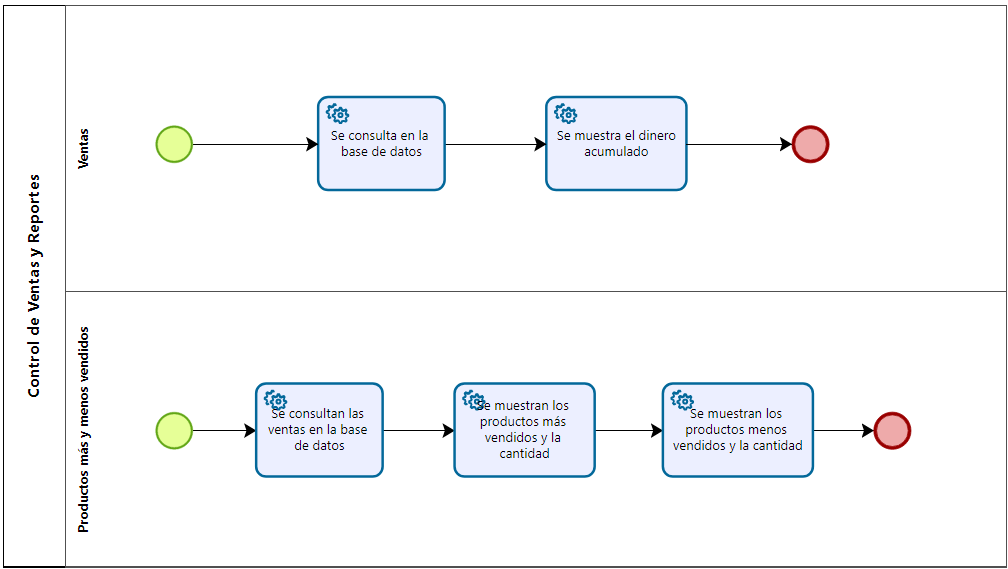
* **Capa de presentación:** Interfaz de usuario web.
* **Capa de negocio:** Lógica de negocios y reglas del sistema.
* **Capa de datos:** Gestión y almacenamiento de datos.
* **Capa de integración:** Servicios y API para la integración con sistemas externos.

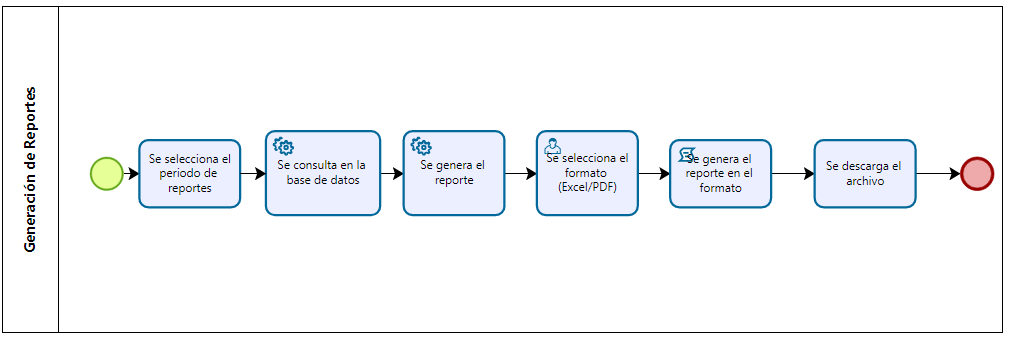
## 2.4 Vista del proceso

El proceso del sistema comienza con la autenticación del usuario. Una vez autenticado, el usuario puede acceder a diferentes funcionalidades como la gestión de ventas y la generación de reportes. Cada acción realizada por el usuario es procesada por la lógica de negocios y registrada en la base de datos.









## 2.5. Vista despliegue

El despliegue del sistema se realiza en un entorno web, accesible a través de navegadores. La arquitectura incluye servidores de aplicaciones para manejar las solicitudes del usuario y servidores de bases de datos para almacenar la información.

