INFORME FINAL DE PROYECTO

NOMBRE EQUIPO: Sysmasters

INTEGRANTES EQUIPO DESARROLLADOR:

1. Garcia Chambilla Luis Mario.
2. Flores Mamani Alex
3. Coca Alvarez Mauricio Gerson
4. Alanes Ortega Kevin Eduardo
5. Introducción

En el presente proyecto se desarrolla un sistema de gestión para un salón de eventos y restaurante, que permite organizar y controlar actividades esenciales, como reservas, gestión de pedidos, manejo del personal y notificaciones de limpieza. El sistema está diseñado para mejorar la eficiencia operativa y la experiencia del cliente, proporcionando herramientas de administración y automatización de procesos clave.

1. Descripción del problema

*Descripción del problema que resuelve el curso*

En los salones y restaurantes, la gestión de actividades es un proceso complejo que involucra múltiples roles y áreas de trabajo. Actualmente, muchos establecimientos realizan estas tareas de forma manual o con sistemas no integrados, lo cual genera ineficiencias como la sobrecarga de trabajo para el personal, retrasos en la atención a los clientes, y problemas de coordinación entre diferentes áreas, como la cocina, el servicio de limpieza, y el control de eventos.

Al no contar con un sistema unificado para la **gestión de reservas, pedidos, inventarios,** **eventos**, y **personal**, se presentan dificultades para mantener la calidad en el servicio, lo que impacta directamente en la satisfacción del cliente y la rentabilidad del negocio.

*El último párrafo debe incluir descripción del objetivo como equipo*

1. Objetivo de la aplicación

*2.1 Objetivo general*

*2.2 Objetivo equipo (tema(s) asignado(s)*

1. Actores involucrados

\*Administrador

\*Cajero

\*Mesero

\*Cliente

\*Personal de limpieza

\*Chef(Personal de cocina)

\*Recursos humanos

\*Recepcionista

\*Seguridad

1. Objetivos de usuarios

**Administrador**: Gestiona eventos, controla el presupuesto, gestiona el inventario, y agenda las fechas de los eventos.

**Cajero**: Registra ventas y pedidos, lee las reservas ,asigna pedidos a delivery, y anula ventas cuando es necesario.

**Mesero**: Registra pedidos, reserva platos, actualiza pedidos,lee reservas de mesas, y gestiona la disponibilidad de mesas.

**Cliente**: Realiza reservas de mesas y platos, visualiza el menú y eventos disponibles, realiza pedidos a domicilio,realiza pagos , deja opiniones, y accede a ofertas.

**Personal de limpieza**: Recibe notificaciones de limpieza, consulta horarios, y reporta el estado de la limpieza.

**Chef**: Planifica el menú del día, consulta inventarios, y revisa pedidos y reseñas.

**Recursos humanos**: Selecciona personal, registra empleados ,gestiona turnos, horarios, y controla la asistencia del personal.

**Recepcionista**: Gestiona reservas, consulta sugerencias, recibe notificaciones de llegadas, y mantiene una lista de contactos.

**Seguridad**: Supervisa el entorno del establecimiento, controla la entrada y salida, recibe turnos de vigilancia, recibe notificacioes de disturbios

1. Requerimientos funcionales[[1]](#footnote-1)
2. Requerimientos no funcionales[[2]](#footnote-2)
3. Especificación de casos de uso o historias de usuario ------------------ desde aquí: solo lo concerniente al trabajo de equipo

**Caso de Uso 1: Gestión de Reservas de Restaurante**

**Actor principal**: Cliente

**Descripción**: El cliente accede al sistema para realizar una reserva en el restaurante.

**Flujo principal**:

El cliente inicia sesión en el sistema.

Selecciona la opción de "Reservar mesa".

Escoge la fecha, hora y el número de personas.

El sistema verifica la disponibilidad de mesas.

El cliente confirma la reserva.

El sistema muestra un mensaje de confirmación.

**Flujo alternativo**: Si no hay disponibilidad de mesas, el sistema ofrece otras fechas u horarios.

**Precondiciones**: El cliente debe estar registrado e iniciar sesión.

**Postcondiciones**: La reserva queda registrada en el sistema.

**Caso de Uso 2: Gestión de Pedidos**

**Actor principal**: Mesero, Cliente

**Descripción**: El mesero o el cliente realizan un pedido de platos.

**Flujo principal**:

El mesero inicia sesión en el sistema.

Selecciona la opción de "Registrar Pedido".

Introduce los platos y bebidas seleccionados por el cliente.

El sistema registra el pedido y lo envía a la cocina.

El chef visualiza los pedidos en la cocina y comienza a preparar los platos.

**Flujo alternativo**: Si un cliente realiza el pedido a domicilio, el sistema asigna un delivery.

**Precondiciones**: El cliente debe haber hecho una reserva o estar en una mesa.

**Postcondiciones**: El pedido queda registrado en el sistema y la cocina recibe la solicitud.

**Caso de Uso 3: Gestión de Eventos**

**Actor principal**: Administrador, Cliente

**Descripción**: El administrador gestiona los eventos programados en el salón.

**Flujo principal**:

El administrador inicia sesión en el sistema.

Selecciona la opción "Gestionar eventos".

Programa o modifica eventos, seleccionando fecha, hora, espacio requerido y servicios adicionales.

El sistema verifica la disponibilidad del salón y confirma la reserva.

El cliente puede visualizar y reservar eventos a través del sistema.

**Precondiciones**: El administrador debe estar registrado y haber iniciado sesión.

**Postcondiciones**: El evento queda registrado en el sistema, disponible para los clientes.

**Caso de Uso 4: Gestión de Inventario**

**Actor principal**: Chef, Administrador

**Descripción**: El chef y el administrador gestionan el inventario de alimentos y bebidas.

**Flujo principal**:

El chef inicia sesión en el sistema.

Selecciona la opción "Consultar Inventario".

Visualiza el inventario de ingredientes y bebidas.

El administrador puede actualizar el inventario cuando llegan nuevos insumos.

El sistema notifica al chef sobre los niveles bajos de algún insumo.

**Precondiciones**: El chef y administrador deben estar registrados e iniciar sesión.

**Postcondiciones**: El inventario queda actualizado en el sistema.

**Caso de Uso 5: Gestión de Limpieza**

**Actor principal**: Personal de Limpieza

**Descripción**: El personal de limpieza recibe notificaciones sobre las tareas de limpieza.

**Flujo principal**:

El personal de limpieza inicia sesión en el sistema.

Consulta las notificaciones de las áreas que requieren limpieza.

El sistema muestra los horarios de limpieza.

El personal reporta el estado de limpieza de cada área.

**Precondiciones**: El personal de limpieza debe estar registrado e iniciar sesión.

**Postcondiciones**: El sistema actualiza el estado de limpieza de cada área.

**Historias de Usuario**

**Como cliente**, quiero poder realizar una reserva de mesa en el restaurante desde la plataforma para asegurar mi lugar en el local.

**Como mesero**, quiero registrar pedidos en el sistema para que la cocina pueda prepararlos de manera eficiente.

**Como administrador**, quiero poder gestionar los eventos y asegurarlos en el calendario para maximizar el uso del salón de eventos.

**Como chef**, quiero consultar el inventario de ingredientes y bebidas, para asegurarme de que no falten insumos para los pedidos.

**Como personal de limpieza**, quiero recibir notificaciones automáticas cuando una área requiera limpieza, para mantener el restaurante en óptimas condiciones.

ANALISIS

1. Lista de clases participantes en cada caso de uso[[3]](#footnote-3)

DISEÑO

1. Diagramas de secuencia por cada caso de uso
2. Diagrama de clases a nivel de diseño[[4]](#footnote-4) *-------------clases de todas sus unidades o módulos*
3. Arquitectura de la aplicación

*(Puntos adicionales: arquitectura del proyecto de curso)*

1. Diagrama de despliegue
2. Diseño de interfaces

1. Lista de casos de uso o historias de usuario. Diagrama de casos de uso. [↑](#footnote-ref-1)
2. Si existen [↑](#footnote-ref-2)
3. Emplear el patrón MVC [↑](#footnote-ref-3)
4. Debe incluir atributos, métodos y relaciones entre clases. Todos los integrantes del equipo aportan sus clases identificadas. [↑](#footnote-ref-4)