INFORME FINAL DE PROYECTO

NOMBRE EQUIPO: Four

INTEGRANTES EQUIPO DESARROLLADOR:

1. Denis Hamilton Almaquio Melgar
2. Apellidos Nombres
3. Introducción
4. Descripción del problema

*Descripción del problema que resuelve el curso*

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

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
3. ANEXO1

**Notificación automática tras desocupación**

**Como** personal del restaurante (mesero o cajero),  
**quiero** marcar una mesa como desocupada,  
**para** que el personal de limpieza reciba una notificación y pueda limpiar la mesa a tiempo.

**Criterios de aceptación**:

* Opción para que el mesero marque una mesa como "desocupada".
* Notificación automática al personal de limpieza cuando la mesa está lista para limpiar.
* El personal de limpieza puede confirmar la finalización de la tarea en el sistema.

**Limpieza tras eventos finalizados**

**Como** personal de eventos,  
**quiero** que el sistema envíe una notificación automática al personal de limpieza cuando termine un evento,  
**para** que el área del evento sea limpiada sin necesidad de intervención manual.

**Criterios de aceptación**:

* Notificación automática al personal de limpieza cuando un evento es marcado como finalizado.
* Información sobre el área específica y tipo de limpieza requerido (superficial o profunda).

**Limpieza programada por ocupación**

**Como** sistema automático,  
**quiero** detectar cuando una zona (como el baño o área común) ha sido utilizada frecuentemente,  
**para** enviar una notificación al personal de limpieza y mantener las áreas limpias.

**Criterios de aceptación**:

* El sistema registra cuántas veces se ha usado una zona.
* Se envía una notificación cuando se alcanza un umbral predefinido de uso.

**Limpieza en tiempos de baja ocupación**

**Como** administrador, **quiero** que el sistema envíe notificaciones de limpieza durante las horas de baja ocupación,  
**para** asegurar que el personal de limpieza no interfiera con el servicio al cliente.

**Criterios de aceptación:**

* El sistema detecta periodos de baja ocupación en tiempo real.
* Se generan notificaciones automáticas al personal de limpieza para aprovechar tiempos muertos

**Limpieza de áreas prioritarias**

**Como** personal de limpieza,  
**quiero** recibir notificaciones de las áreas con mayor prioridad para limpieza,  
**para** poder atender las zonas más críticas primero.

**Criterios de aceptación**:

* El sistema clasifica las áreas según prioridad (baños, cocinas, áreas comunes).
* El personal recibe notificaciones con detalles de la prioridad.

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)