

# ANTEPROYECTO DEL TRABAJO DE FIN DE GRADO

## INFORMACIÓN GENERAL

<b>Alumno/a</b>	Pedro Antonio Benito Rojano				
<b>Titulación:</b>	Grado en Ingeniería del Software				
<b>Tutor/es:</b>	José Carlos Canal Velasco				
<b>Título</b>	QRest: Web de gestión de comandas con QR para restaurantes.				
<b>Subtítulo</b> <i>(solo si en grupo)</i>					
<b>Título en inglés</b>	Qrest: QR order management website for restaurants.				
<b>Subtítulo en inglés</b> <i>(solo si en grupo)</i>					
<b>Trabajo en grupo:</b>	<b>Sí</b>	<input type="checkbox"/>	<b>No</b>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<b>Otros integrantes del grupo:</b>					

## INTRODUCCIÓN

*Contextualización del problema a resolver. Describir claramente de dónde surge la necesidad de este TFG y el dominio de aplicación. En caso de que el TFG se base en trabajos previos, debe aclararse cuáles son las aportaciones del TFG.*

El sector de la restauración ha tenido un notable crecimiento a nivel mundial en los últimos 50 años, y no para de crecer. El transporte es más accesible y el turismo se ha visto impulsado y popularizado como forma de ocio en el ciudadano medio; las mejoras en los medios de comunicación han hecho que las personas sean más sociales y típicamente se reúnan en bares o restaurantes; muchas personas acosumbran a desayunar y/o almorzar antes o después del trabajo, mientras que las cenas y meriendas suelen ser comidas de ocio, de forma que el sector de la restauración no descansa.

La restauración es uno de los modelos de negocio más antiguos. Y por desgracia no hemos avanzado mucho en modernizar este sector. Solo las grandes cadenas de restaurantes han llegado a implementar sistemas para mejorar y automatizar la gestión de comandas; y generalmente son costosos a nivel hardware, ya que hacen uso de dispositivos específicos, además de software especializado para la cadena de restaurantes en concreto. La mayoría de restaurantes son pequeñas empresas de un solo local, por lo que no se pueden permitir el desarrollo de un sistema específico y generalmente terminan tomando la carta a papel o usando sistemas POS [1] en el mejor de los casos. El problema de los sistemas POS es que necesitas una gran inversión inicial que puede llegar a los miles de Euros, una mensualidad y una comisión por transacciones.

El sistema de gestión de comandas es solo uno de los problemas que encuentra un restaurante para constituirse, pero hay muchos más. ¿Qué pasa con los pedidos a domicilio? Por simplicidad la mayoría de restaurantes atienden estos pedidos de forma telefónica. ¿Qué pasa con las reservas? De igual forma suelen ser atendidos de forma telefónica. ¿Qué pasa con los datos online? Si quieren pedir a domicilio, querrán poder pagar online, muchos de los restaurantes lo cobran en efectivo en la entrega, o en el mejor de los casos cada repartidor lleva un datáfono. ¿Qué pasa con la web o carta online? Hoy día es típico tener una web de tu restaurante, pero esto es costoso, por lo que muchos restaurantes recurren a crear un enlace con el PDF de su carta. ¿No sería genial un sistema íntegro para la gestión de tu restaurante que te solucionase todos estos problemas? Y que además tuviese muy bajo coste de instalación. Este sería del sueño de miles de emprendedores que deciden montar un restaurante cada año en España.

Además cada pedido son datos que actualmente no se están utilizando. Podríamos utilizar los datos de estos pedidos para sacar estadísticas de comandas y así analizar y predecir las futuras.

España es el país con más bares y restaurantes del mundo [2] lo que ofrece una gran oportunidad para abordar una solución a este problema. Así surge QRest, una solución software basada en web que proveerá de un sistema integral para la gestión de restaurantes: Pedida de comandas a través de QR, gestión de comandas a través de la web, web específica para cada restaurante, sugerencias personalizadas, pedidos a domicilio, pagos online e impresión de facturas. Un sistema integral para una experiencia completa del cliente y una solución sencilla para todo tipo de restaurantes.

Los restaurantes que más se pueden beneficiar de QRest son restaurantes nuevos que quieran usar una solución simple y sin mucha barrera de entrada, restaurantes en los que se suelen pedir comandas en varias veces (ej: bares, tapas, etc.) y/o con mucho personal que puedan ahorrar tiempo o reducir personal al no tener que tomar las comandas, restaurantes que les falten algunas de estas funcionalidades típicas o quieran mejorarlas y restaurantes que quieran ahorrar al hacer uso de un solo sistema en vez de todos estos por separado.

Una ventaja adicional del uso del sistema es que la web del restaurante será accesible a través de la siguiente URL: [URL QRest]/[Nombre del restaurante]. De forma que el restaurante no tendrá que hacerse con un dominio. Además obtendrá “publicidad gratuita”, ya que en la web de QRest se mostrarán los restaurantes que cuentan con este sistema.

## OBJETIVOS

*Descripción detallada de en qué consistirá el TFG. En caso de que el objeto principal del TFG sea el desarrollo de software, además de los objetivos generales deben describirse sus funcionalidades a alto nivel.*

El principal objetivo es que desarrollar una aplicación web a través de la cual puedas pedir comandas a través de los QR de las mesas, llegando estas a cocina, y una vez marcadas como listas que los camareros puedan servirlos. Y finalmente generar una factura que poder imprimir. Con la opción de realizar el pago online a través de la web.

El QR en la mesa redireccionará al pedido asociado a esta, de forma que todos los clientes de la mesa puedan pedir simultáneamente a través de este, pudiendo ver en tiempo real si alguien añade o elimina un plato a la comanda. Cada vez que una mesa confirme una comanda, de haber comandas anteriores de la misma mesa estas se añadirán a una misma factura hasta que se pague. Estas comandas deben llegar a cocina al confirmar el pedido. Para ello necesitaremos una tablet en cocina para ver los platos pendientes, en caso de no ser plato (ej: bebida, postre, etc) este elemento pasará directamente a la lista de camareros. En la tablet de cocina se mostrarán los platos ordenados por orden de comanda. Una vez que un plato esté listo, es marcará como listo y pasará a la lista de camareros. Los camareros servirán los elementos mostrados en la lista de camareros, esta lista puede ser visualizada a través de la misma tablet de cocina o en una a parte de así desearlo. Finalmente cuando el cliente pida la factura, el camarero podrá imprimirla directamente a través de la web, la cual enviará esta a la impresora de tickets. Si el pago se hace en efectivo o con datáfono, el camarero deberá marcar el pedido como pagado en la web. En caso de hacer el pago online a través de la web, el pedido se cerrará automáticamente.

Para hacer este sistema escalable y personalizado para cada restaurante, se dispondrá de un fichero de configuración y de una base de datos específica para cada uno de ellos. Además de un CRUD [3] para añadir elementos a la carta bajo un inicio de sesión autorizado.

Como funcionalidades adicionales, se incluirán: Posibilidad de pedidos a domicilio, reservas online y sugerencias personalizadas a usuarios.

Para los pedidos a domicilio se pedirá la dirección y se procederá de igual forma que los pedidos en restaurante pero con las siguientes diferencias: el pedido será una sola comanda, es decir, una vez confirmado el pedido no se podrán añadir nuevas comandas a este; que los camareros una vez empaquetado el pedido este pasará a una lista de reparto; y que el restaurante podrá exigir el pago de este al hacer el pedido.

Los clientes podrán hacer reservas a través de la web, en caso de que el restaurante así lo desee.

Para una experiencia de usuario más personalizada se podrán ofrecer sugerencias a clientes frecuentes en base a pedidos anteriores.

## ENTREGABLES

**Listado de resultados que generará el TFG (aplicaciones, estudios, manuales, etc.)**

Aplicación web desarrollada en python.

Despliegue del sistema.

Documento General de Requisitos (DGR).

Manual de Instalación.

Manual de Usuario.

Memoria del TFG.

**MÉTODOS Y FASES DE TRABAJO**

**METODOLOGÍA:**

*Descripción de la metodología empleada en el desarrollo del TFG. Especificar cómo se va a desarrollar. Concretar si se trata de alguna metodología existente y, en caso contrario, describir y justificar adecuadamente los métodos que se aplicarán.*

Para realizar el TFG, se aplicará una metodología de desarrollo ágil, donde tendrán lugar reuniones periódicas con el tutor asignado, obteniendo así una opinión y retroalimentación para que en cada fase se resuelvan los problemas existentes y sea posible el desarrollo óptimo del trabajo, tratando de proporcionar la mayor flexibilidad al proyecto y obteniendo un resultado con la mejor calidad posible.

**FASES DE TRABAJO:**

*Enumeración y breve descripción de las fases de trabajo en las que consistirá el TFG.*

**1. Análisis.**

- a. Estudio e investigación de las tecnologías a utilizar.
- b. Extracción de los requisitos del proyecto.
- c. Elaboración del Documento General de Requisitos (DGR).

**2. Diseño.**

- a. Diseño y modelado de la Aplicación.
- b. Estudio, diseño y modelado de la Base de Datos No Relacional.
- c. Estudio, diseño y modelado del archivo de configuración.
- d. Maquetado de la Aplicación.

**3. Desarrollo e implementación.**

- a. Sistema de Pedidos en tiempo real.
- b. Sistema de Gestión de comandas.
- c. Sistema de Generación de facturas.
- d. Sistema de Impresión de tickets.
- e. Sistema de Pago.
- f. Sistema de Gestión de reservas.
- g. Sistema CRUD de carta.
- h. Sistema de Configuración de restaurantes.
- i. Sistema de Sugerencias a clientes.

**4. Verificación y pruebas.**

- a. Realización de Pruebas unitarias.
- b. Realización de Pruebas de integración.
- c. Realización de Pruebas de extremo a extremo.
- d. Realización de Pruebas en un restaurante.

**5. Documentación.**

- a. Elaboración del Manual de Usuario.
- b. Elaboración del Manual de Instalación.
- c. Elaboración de la Memoria del TFG.

## TEMPORIZACIÓN:

La siguiente tabla deberá contener una fila por cada una de las fases enumeradas en la sección anterior. En caso de tratarse de un trabajo en grupo, se añadirá una columna HORAS por cada miembro del equipo. Debe especificarse claramente el número de horas dedicado por cada alumno/a y la suma de horas individual deberá ser también de 296.

FASE	HORAS
	Pedro Antonio Benito Rojano
<b>1. Análisis.</b>	<b>26</b>
a. Estudio e investigación de las tecnologías a utilizar.	10
b. Extracción de los requisitos del proyecto.	10
c. Elaboración del Documento General de Requisitos (DGR).	6
<b>2. Diseño.</b>	<b>45</b>
a. Diseño y modelado de la Aplicación.	15
b. Estudio, diseño y modelado de la Base de Datos No Relacional.	10
c. Estudio, diseño y modelado del archivo de configuración.	10
d. Maquetado de la Aplicación.	10
<b>3. Desarrollo e implementación.</b>	<b>140</b>
a. Sistema de Pedidos en tiempo real.	20
b. Sistema de Gestión de comandas.	10
c. Sistema de Generación de facturas.	10
d. Sistema de Impresión de tickets.	15
e. Sistema de Pago.	10
f. Sistema de Gestión de reservas.	15
g. Sistema CRUD de carta.	20
h. Sistema de Configuración de restaurantes.	25
i. Sistema de Sugerencias a clientes.	15
<b>4. Verificación y pruebas.</b>	<b>50</b>
a. Realización de Pruebas unitarias.	20
b. Realización de Pruebas de integración.	10
c. Realización de Pruebas de extremo a extremo.	5
d. Realización de Pruebas en un restaurante.	15
<b>5. Documentación.</b>	<b>35</b>
a. Elaboración del Manual de Usuario.	10
b. Elaboración del Manual de Instalación.	10
c. Elaboración de la Memoria del TFG.	15
	<b>296</b>

**ENTORNO TECNOLÓGICO****TECNOLOGÍAS EMPLEADAS:**

*Enumeración de las tecnologías utilizadas (lenguajes de programación, frameworks, sistemas gestores de bases de datos, etc.) en el desarrollo del TFG.*

Lenguaje de programación: Python + HTML + CSS + JS

Base de de datos no relacional (JSON): MongoDB [4]

Framework web: FastAPI [5]

Framework CSS: Bulma [6]

**RECURSOS SOFTWARE Y HARDWARE:**

*Listado de dispositivos (placas de desarrollo, microcontroladores, procesadores, sensores, robots, etc.) o software (IDE, editores, etc.) empleados en el desarrollo del TFG.*

Ordenador personal

IDE: Visual Studio Code [6]

Plataforma de despliegue: Vercel [8]

Tablets para probar el sistema en en restaurante

Impresora de Tickets para imprimir facturas y QRs

**REFERENCIAS**

*Listado de referencias (libros, páginas web, etc.)*

[1] “Sistema POS”. ¿Qué es el sistema pos y cómo funciona?. <https://www.certus.edu.pe/blog/que-es-sistema-pos/> (accedido el 12 de febrero de 2023)

[2] “España es el país con más bares y restaurantes del mundo”. España es el país con más bares y restaurantes del mundo. <https://www.elmundo.es/viajes/espana/2020/06/19/5eeb49defdddf66b8b4587.html> (accedido el 12 de febrero de 2023)

[3] “CRUD”. CRUD – Glosario de MDN Web Docs. <https://developer.mozilla.org/es/docs/Glossary/CRUD> (accedido el 12 de febrero de 2023)

[4] “MongoDB”. MongoDB: The Developer Data Platform. <https://www.mongodb.com/> (accedido el 12 de febrero de 2023)

[5] “FastAPI”. FastAPI. <https://fastapi.tiangolo.com/> (accedido el 12 de febrero de 2023)

[6] “Bulma”. Bulma: Free, open source, and modern CSS framework based on Flex box. <https://bulma.io/> (accedido el 12 de febrero de 2023)

[7] “Visual Studio Code”. Visual Studio Code – Code Editing. Redefined. <https://code.visualstudio.com/> (accedido el 12 de febrero de 2023)

[8] “Vercel”. Vercel: Develop. Preview. Ship. For the best Frontend teams. <https://vercel.com/> (accedido el 12 de febrero de 2023)

Málaga, 12 de Febrero de 2023

Firma tutor/tutora:

Firma cotutor/a:

Firma tutor/a coordinador/a: