****

Memoria TFG Desarrollo Aplicaciones Web



Victor Manuel Omar Allende

Tutor: Carlos Rufiángel

# Abstract

Este proyecto llevará a cabo el desarrollo de una aplicación web para pequeñas y medianas empresas del sector de la hostelería que necesiten realizar tareas como la gestión de su oferta de menú diario y además obtener funcionalidades imprescindibles en el negocio como la reserva online de sus clientes.

Este proyecto surge como una solución alternativa ante las aplicaciones grandes que son más complejas y costosas y no son adecuadas para las pequeñas y medianas empresas de hostelería.

HostApp tiene un importante potencial de crecimiento ya que su patrón de diseño, al ser MVC, modulariza cada componente, lo que facilita cualquier tipo de evolución que cubra necesidades futuras. El poder de personalización de HostApp es lo que lo hace realmente interesante y válido para las empresas. pequeñas y medianas empresas del sector.

This project will carry out the development of a web application for small and medium-sized companies in the hospitality sector that need to perform tasks such as managing their daily menu offer and also obtain essential functionalities in the business such as the online reservation of their clients.

This project arises as an alternative solution to large applications that are more complex and expensive and are not suitable for small and medium-sized hospitality companies.

HostApp has significant growth potential since its design pattern, being MVC, modularizes each component, which facilitates any type of evolution that covers future needs. HostApp's power of customization is what makes it really interesting and valid for companies. small and medium-sized companies in the sector.

# Índice

[Abstrac 1](#_Toc135059626)

[Índice 2](#_Toc135059627)

[Objetivos 4](#_Toc135059628)

[Diagrama RFTP 4](#_Toc135059629)

[Introducción 9](#_Toc135059630)

[Justificación del Proyecto 10](#_Toc135059631)

[Desarrollo 11](#_Toc135059632)

[Arquitectura del proyecto 11](#_Toc135059633)

[Entornos de desarrollo 12](#_Toc135059634)

[Control de versiones 13](#_Toc135059635)

[Cliente control de versiones 13](#_Toc135059636)

[Estructura del proyecto 14](#_Toc135059637)

[Modelo 14](#_Toc135059638)

[Vista 15](#_Toc135059639)

[Controlador 17](#_Toc135059640)

[Configuración del proyecto 17](#_Toc135059641)

[Modelo de datos 18](#_Toc135059642)

[Describe usuarios 18](#_Toc135059643)

[Describe menu 18](#_Toc135059644)

[Describe reservas 18](#_Toc135059645)

[Diseño 19](#_Toc135059646)

[Diseño MVC 19](#_Toc135059647)

[Sesión 19](#_Toc135059648)

[Validaciones 20](#_Toc135059649)

[Encriptación 22](#_Toc135059650)

[Diseño de vistas 22](#_Toc135059651)

[Diagrama de flujo de navegación 23](#_Toc135059652)

[Flujo de navegación usuarios tipo ‘n’ 24](#_Toc135059653)

[Flujo de navegación usuarios tipo ‘a’ 25](#_Toc135059654)

[Diagrama de peticiones web 25](#_Toc135059655)

[Diagrama de clases 28](#_Toc135059656)

[Diagrama entidad relación 28](#_Toc135059657)

[Tabla usuarios 29](#_Toc135059658)

[Tabla Menu 29](#_Toc135059659)

[Tabla Reservas 29](#_Toc135059660)

[Diagrama de base de datos 30](#_Toc135059661)

[Diagrama de casos de uso 31](#_Toc135059662)

[Metodología 33](#_Toc135059663)

[Diagrama de Gantt 33](#_Toc135059664)

[Cuadro de tiempos 35](#_Toc135059665)

[Tecnologías 36](#_Toc135059666)

[Trabajos futuros 37](#_Toc135059667)

[Pruebas 38](#_Toc135059668)

[Conclusiones 64](#_Toc135059669)

[Bibliografía 65](#_Toc135059670)

[Glosario 66](#_Toc135059671)

# Objetivos

Objetivos generales

Og\_Diseñar una aplicación o solución informática dirigida al negocio de la restauración, con una solución simple, económica y mantenible para la gestión de cualquier negocio de la restauración, abarcar las necesidades o requerimientos en la gestión de reservas de clientes, así como también proporcionar una gestión personalizada para los administradores del negocio.

Objetivos específicos.

Oe.1\_Desarrollar una interfaz de usuario simple e intuitiva.

Oe.2\_Proporcionar una administración directa y sencilla sobre las tareas.

Oe.3\_Incorporar un sistema de reservas a sus usuarios de forma simple rápida y segura.

Oe.4\_Permitir en todo momento conocer el detalle de las reservas realizadas.

## Diagrama RFTP

Para lograr los objetivos detallados anteriormente, se deben llevar a cabo las tareas de programación que detallan a continuación en el diagrama RFTP, en este diagrama solo se contemplan las tareas relativas a los desarrollos, las tareas de análisis, formación y demás tareas que no computan horas de desarrollo no están incluidas en este diagrama.

**R-2000 Configuraciones iniciales de PHP**

F-2001 Configuración del proyecto, variables, define etc.

T-2001-1 Configuración inicial en config.php

**R-3000 Crear base de datos MySQL**

F-3001 Creación de la base de datos en MySQL

T-3001-1 Definición de Script de creación de base de datos

T-3001-2 Mock de datos para probar la aplicación

T-3001-3 Crear base de datos por script en MySQL

**R-4000 Vistas de login, páginas principales de bienvenida, header y footer**

F-4001 Desarrollo MVC, vistas, formularios.

T-4001-1 login.php vista y formulario para realizar de login en la app

T-4001-2 bienvenida\_admin.php vista principal de administradores

T-4001-3 bienvenida.php vista principal de los clientes registrados

T-4001-1 Desarrollo de Header y Footer aplicables a toda la app

**R-5000 Desarrollo de Session PHP**

F-5001 Desarrollar inicio de Session para los usuarios

T-5001-1 Desarrollo de Session session.php

**R-6000 Login y Registro MVC**

F-6001 Funciones en controlador y modelo

T-6001-1 Realizar login

T-6001-2 Alta nuevos clientes

T-6001-3 Desarrollo de la clase Controlador y Modelo CRUD

T-6001-4 Desarrollo de index.php llamadas GET o POST

P-6002-1 Realizar login sin datos en formulario

P-6002-2 Realizar login DNI erróneo mal formato

P-6002-3 Realizar login password incorrecto

P-6002-4 Realizar login correcto

**R-7000 Estilos CSS**

F-7001 Desarrollar estilos globales

T-7001-1 Estilos de la aplicación CSS, desarrollo parcial

**R-8000 Insertar nuevas transaccione, obtener listados de estas**

F-8001 Insertar datos y listar datos desde la base de datos

T-8001-1 Controlador nueva transacción insertar datos

T-8001-2 Controlador mostrar listas desde base de datos

T-8001-3 Mostrar reservas admin, funciones Modelo y CRUD

**R-9000 Incorporar nuevas vistas funcionales**

F-9001 Crear menu, crear reserva vistas de actualización de datos

T-9001-1 crear\_menu.php vista y formulario de los administradores

T-9001-2 crear\_reserva.php vista y formulario

T-9001-3 mostrar\_reservas\_admin.php vista para administradores

T-9001-4 mostrar\_reservas.php vista de usuarios o cliente de la app

T-9001-5 mostrar\_menu.php vista de los administradores de la app

T-9001-6 actualizar\_menu.php vista y formulario

T-9001-7 actualizar\_reserva.php vista y formulario

P-9002-1 Crear un menú sin datos usuario admin

P-9002-2 Crear un menú con datos erróneos usuario admin

P-9002-3 Crear un menú con datos usuario admin

P-9002-4 Listar todos los menús usuarios admin

P-9002-5 Listar los menús propios usuarios admin

P-9002-6 Modificar un menú sin datos usuario admin

P-9002-7 Modificar un menú con erróneos usuarios admin

P-9002-8 Modificar un menú con datos usuario admin

P-9002-9 Eliminar un menú usuario admin

P-9002-10 Listar reservas usuario admin

P-9002-11 Eliminar reservas usuario admin

P-9002-12 Listar todos los menús usuario cliente

P-9002-13 Crear una reserva usuario cliente sin datos

P-9002-14 Crear reserva usuario cliente con datos erróneos

P-9002-15 Crear una reserva usuario cliente con datos

P-9002-16 Listar reservas usuario cliente

P-9002-17 Actualizar reserva usuario cliente sin datos

P-9002-18 Actualizar reserva cliente con datos erróneos

P-9002-19 Actualizar reserva usuario cliente con datos

P-9002-20 Eliminar reserva usuario cliente

**R-10000 Navegación (funciones en controlador y modelo)**

F-10001 Navegación entre paginas

T-10001-1 Redirección entre páginas devueltas por controlador

P-10002-1 Pruebas de navegabilidad entre páginas

**R-11000 Registrar usuarios vista y formulario**

F-11001 Desarrollo de vista de registro

T-11001-1 registro.php vista y formulario de registro de clientes

T-11001-2 Guardar y recuperar datos de registro

P-11002-1 Realizar registro sin datos en formulario

P-11002-2 Realizar registro nombre incorrecto (caracteres)

P-11002-3 Realizar registro DNI erróneo mal formato)

P-11002-4 Realizar registro Email erróneo mal formato

P-11002-5 Realizar registro usuario ya existente

P-11002-6 Realizar registro correcto

**R-12000 Ampliación estilos**

R-12001 Ampliar estilos a las nuevas vistas

T-12001-1 Estilos de la aplicación CSS, desarrollo parcial

**R-14000 Incluir menú de cierre de sesión**

F-14000 Incluir en las vistas opción de menú cierre de sesión

T-14001-1 Modificaciones en modelo.php, perfil usuarios

T-14001-2 Incluir opciones en vistas de la app

P-14002-1 Visualización de cerrar sesión ambos perfiles

**R-15000 Link de navegación a cierres de sesión**

F-15001 Incluir nuevos links a despedida Sessión Destroy

T-15001-1 Modificaciones en registro.php login.php vista y formulario

T-15001-2 logout.php vista para cerrar sesión para todos los usuarios

T-15001-3 despedida.php, Sesión Destroy para todos los usuarios

P-15002-1 Proceso de cierre de sesión ambos perfiles

**R-16000 Mostrar reservas a usuarios**

F-16001 Modificar, Controlador y Modelo (parametrización y query)

T-16001-1 Parametrizar llamadas a base de datos query join

P-16002-1 Visualizar reservas usuarios

**R-17000 Incluir banner e iconos a la app e incluir menu de cierre de sesión**

F-17001 Incluir banner e iconos, ver menu cierre sesión

T-17001-1 Crear banner e incluirlos en toda la app

T-17001-2 Modificaciones opciones de menú

T-17001-3 Incluir iconos en la aplicación

P-17002-1 Ver banner e icono en la app

**R-18000 Validaciones de formularios**

F-18001 Desplegar utils.php encargado de las validaciones

T-18001-1 Validaciones, texto, email, dni, password Encriptación md5

P-18002-1 Probar formulario y datos en md5 en base de datos

R-19000 **Ampliación estilos**

F-90001 Incluir nuevos estilos, desarrollo parcial

T-19001-1 Estilos de la aplicación CSS

# Introducción

En el presente proyecto se va a desarrollar una aplicación web como solución para las pequeñas y medianas empresas del sector de la hostelería, que necesiten o demanden actualizar su sistema de reservas para sus clientes a un sistema de reservas online.

Se va a cubrir la necesidad de actualización online de los menús ofertados el día a día, manteniendo actualizada su oferta gastronómica, de esta manera, serán los propios administradores de ***HostApp*** los que tendrán en la herramienta la posibilidad de crear menús, actualizar o eliminarlos de forma rápida y sencilla.

***HostApp*** ofrece mantener en todo momento un control exacto de las reservas que hacen sus clientes en su negocio y poder administrarlas.

Seofrece a los clientes de los negocios de restauración que tengan como sistema de gestión nuestro producto ***HostApp*** la cómoda posibilidad de realizar reservas online en cualquier momento. Así como conocer la oferta gastronómica que se ofrece día a día totalmente online.

# Justificación del Proyecto

En la actualidad donde cada vez se implementan más soluciones informáticas en todo ámbito de negocio, se ha detectado que medianas o pequeñas empresas del sector de la hostelería no tienen la suficiente oferta de soluciones informáticas a medida o para tereas simples, la mayoría de estas soluciones son grandes, complejas y costosas.

Observado esto, nace la idea de ***HostApp***, que es una solución simple y adaptable a cualquier negocio a los cuales les permite entrar en el ámbito web, permitiendo a sus clientes hacer reservas online, modificarlas o cancelarlas, conocer su oferta día a día sin necesidad de poseer complejas soluciones informáticas, también administrar, crear ofertas (menú) actualizar o eliminar los mismos, conocer el estado de reservas y administrarlas.

Es por estos motivos que ***HostApp*** se posiciona como una potente alternativa para pequeñas y medianas empresas que busquen una solución específica para estas necesidades.

Además, se ha optado a realizarlo con tecnología PHP, ya que para aplicaciones web es muy potente, si bien es cierto que, con el avance de nuevas tecnologías, PHP tiende a perder el dominio que tiene desde el año 2004, pero sigue estando muy presente en el mundo web a nivel empresarial, con lo cual es una oportunidad única para aprender más de esta tecnología y poder desarrollar nuevos conocimientos.

# Desarrollo

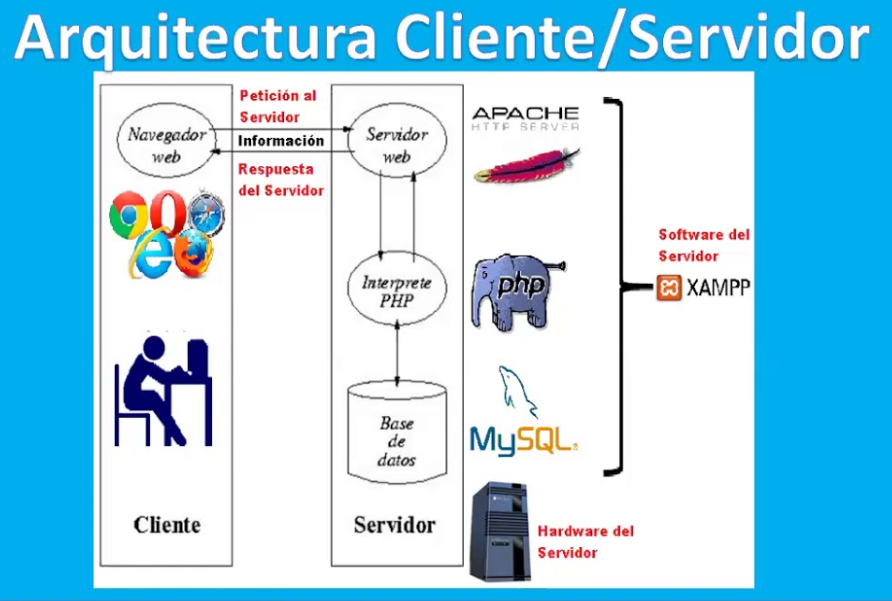
Durante el desarrollo se va a llevar a cabo la construcción de una aplicación web, que se encargará de gestionar tareas diarias de los pequeños y medianos negocios de la restauración y ofrecer a clientes y administradores las siguientes funcionalidades:

* Administradores:
  + Gestionar online los menús ofertados crear, ver, modificar y eliminar.
  + Gestionar las reservas online de sus clientes, ver reservas, cancelarlas o eliminarlas.
* Clientes:
  + Ver los menús ofertados y seleccionarlos al reservar.
  + Realizar reservas, ver reservas, modificarlas o eliminarlas.

## Arquitectura del proyecto

La arquitectura de la aplicación ***HostApp*** es la siguiente:

|  |  |  |
| --- | --- | --- |
| **Servidor web** | **Base de datos** | **Tecnología** |
| Apache XAMMP | MySQL MariaDB | PHP 7.4.27 |



Fuente: <https://www.youtube.com/@AreaCurricularInformatica>

**El patrón de arquitectura** del software es **Modelo-Vista-Controlador o MVC** que es el estándar que permite una separación de los datos o lógica del negocio de las interfaces, dando así una independencia de funcionamiento y mayor facilidad en su mantenimiento.

El módulo encargado de la comunicación entre vistas y modelo es el módulo de controlador.

Más detalles en el apartado de tecnologías.

Diagrama

Descripción generada automáticamente

Fuente: <https://hocphp.net/tong-quan/tin-tuc/mo-hinh-mvc-trong-php/>

## Entornos de desarrollo

Como ide de desarrollo se ha seleccionado **Visual Studio Code** en su Version 1.74.2, por su potencial como ide de desarrollo de aplicaciones web y ser de libre distribución, no se necesita licencia, también se ha utilizado el kit de herramientas de código libre que incorpora **XAMPP**, para los despliegues del proyecto en el servidor Apache, así como la base de datos MySQL que incorpora xampp.

## Control de versiones

Como sistema de gestión de versiones se ha utilizado GitHub, con una cuenta en su plataforma online.

Esta herramienta es totalmente gratuita, aunque los repositorios son públicos, para hacer los repositorios privados hay que adquirir una licencia de pago.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

<https://github.com/victAll/TFG_Victor_Allende.git>

## Cliente control de versiones

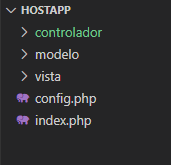
Como cliente GIT he utilizado SourceTree en su versión 3.4.12.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

## Estructura del proyecto

La estructura del proyecto se divide en **3 carpetas principales** *controlador* contiene la capa controladora, *modelo* contiene la clase encargada de la interacción con la base de datos y *vista* que contine las vistas o interfaces de usuario y formularios, el fichero *index.php* es el fichero de inicio del proyecto y también el encargado de distribuir las peticiones que se envían de los formularios desde las diferentes interfaces, ya sean de tipo *GET* o *POST* y el fichero *config.php* que es el encargado de la configuración del proyecto.



### Modelo

Esta carpeta contiene el principal fichero que permite la interacción de la aplicación con el modelo de datos: *modelo.php* También incluyo un fichero *conexión.php* que es el encargado de abrir conexión con la base de datos, el fichero *sesión.php* encargado de gestionar las sesiones y por último una carpeta *scriptDB* con el fichero *scriptdb.sql* que contiene un script para generar la base de datos desde cero y un juego de datos.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

|  |  |  |
| --- | --- | --- |
| **Fichero** | **Clase** | **Descripción** |
| modelo.php | class Modelo | Funciones y métodos para interactuar con el modelo de datos. |
| conexión.php |  | Funciones para gestionar conexiones a la base de datos |
| session.php |  | Gestiona las sesiones de los usuarios. |

### Vista

Contiene las vistas y formularios de la aplicación además de los archivos CSS de la aplicación y recursos:

**Carpeta vista**: Vistas de la aplicación, vistas genéricas, de administración y usuarios clientes.

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

|  |  |
| --- | --- |
| **Fichero** | **Descripción** |
| actualizar\_menu.php | Vista de administrador de la aplicación |
| actualizar\_reserva.php | Vista de usuario de la aplicación |
| bienvenida\_admin.php | Vista de administrador de la aplicación |
| bienvenida.php | Vista de todos los usuarios de la aplicación |
| crear\_menu.php | Vista de administrador de la aplicación |
| crear\_reserva.php | Vista de usuario de la aplicación |
| despedida.php | Vista de todos los usuarios de la aplicación |
| login.php | Vista de todos los usuarios de la aplicación |
| logout.php | Vista de todos los usuarios de la aplicación |
| mostrar\_menu.php | Vista de administrador de la aplicación |
| mostrar\_reservas\_admin.php | Vista de administrador de la aplicación |
| mostrar\_reservas.php | Vista de usuario de la aplicación |
| registro.php | Vista de todos los usuarios de la aplicación |

**Carpeta CSS**: Estilos de la aplicación:



|  |  |
| --- | --- |
| **Fichero** | **Descripción** |
| style.css | Estilos de la aplicación |

**Carpeta layouts**: Vistas genéricas header y footer de la aplicación.

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

|  |  |
| --- | --- |
| **Fichero** | **Descripción** |
| footer.php | Vista de footer común a toda la aplicación |
| header.php | Vista cabecera común a toda la aplicación |

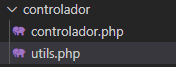
**Carpeta res**: recursos de la aplicación, iconos, banners utilizados por la aplicación.

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

### Controlador

En esta carpeta se almacena el fichero Controller de la aplicación y un fichero de utilidades o de validaciones:



|  |  |  |
| --- | --- | --- |
| **Fichero** | **Clase** | **Descripción** |
| controlador.php | class ModeloController | Métodos y funciones que permiten la comunicación entre la vista y el modelo de datos |
| utils.php |  | Métodos y funciones de validación para los formularios |

### Configuración del proyecto

En la raíz del proyecto se encuentran los ficheros de configuración de HostApp, estos ficheros son *index.php.* y *config.php.*

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente con confianza media

|  |  |
| --- | --- |
| **Fichero** | **Descripción** |
| index.php | Recibe las peticiones *Get* o *Post* desde las vistas o formularios y llama a controlador |
| config.php | Configuración de variables globales a toda la apliicación |

## Modelo de datos

El modelo de datos consiste en 3 tablas, las que se encargan de almacenar *usuarios*, *menú* y *reservas.* A continuación, se mostrará un *“describe table”* de cada tabla, se detallarán en profundidad cada una en el diagrama de base de datos más adelante.

### Describe usuarios

Texto

Descripción generada automáticamente

### Describe menu

Imagen que contiene texto, pizza, tabla, grande

Descripción generada automáticamente

### Describe reservas

Imagen que contiene texto, marcador, pizza, tabla

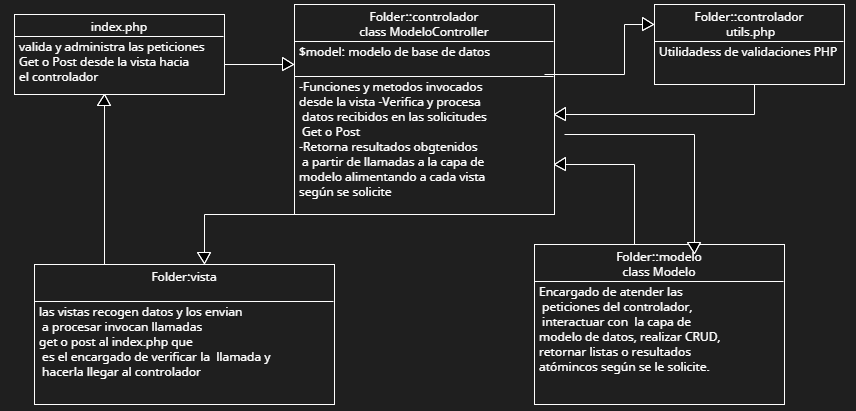
Descripción generada automáticamente

# Diseño

El objetivo de este apartado es presentar y dar a conocer el diseño de ***HostApp*** en su totalidad, se mostrarán los diagramas de diseño MVC, sesión, validaciones, encriptación, diseño de vistas, diagrama de flujo de navegación, esquema de las peticiones web realizadas desde HostApp, cada una con su method sea get o post, diagrama de clases, diagrama de entidad relación, diagrama de base de datos y diagrama de casos de uso.

## Diseño MVC

El patrón de diseño escogido para HostApp es el más utilizad para animaciones web. Por su alta abstracción de capas, separación de capa de lógica, vistas y modelo, a continuación, mostraré el diagrama de MVC utilizado por HostApp.



## Sesión

A continuación, se muestra el ciclo de vida de la Sesión PHP, se inicia al hacer login ***sesión\_start()*** y finaliza ***sesión\_destroy()*** al cerrar sesión por parte del usuario logeado:

Diagrama

Descripción generada automáticamente

## Validaciones

Para las validaciones de los datos inyectados a ***HostApp***, he utilizado un fichero ***utils.php*** que es el encargado de realizar estas funciones. Las funciones para las validaciones son las siguientes:



**(\*1)**

* / = Delimitador de la expresión regular, define el inicio y el fin.
* [] = Definición de una clase, el carácter puede ser cualquiera de los valores que se encuentran definidos dentro de la clase.
* a-z = Rango de letras desde la a hasta la z, sería lo mismo que escribir abcdefg...
* A-Z = Rango de letras desde la A hasta la Z, lo mismo que antes, pero en mayúsculas
* \s = El equivalente a un carácter de tipo espacio
* ñáéíóúÁÉÍÓÚ = caracteres que no están incluidos en a-z y A-Z, para que la expresión coincida cuando estén estos caracteres.
* ^ = Define que la expresión debe aplicar desde el comienzo de la cadena
* ·          $ = Define que la expresión debe aplicar hasta el final de la cadena
* ·          {4,} requiere de una cadena de al menos 4 caracteres

**(\*2”)**

* validar longitud && validar que la 9º posición es '-' ej: 53380908-R
* nos quedamos con los números del dni
* nos quedamos con la letra del dni
* fórmula para encontrar la letra
* resto de los números del dni / 23 tiene que ser el numero de la letra final del dni

## Encriptación

La encriptación utilizada por ***HostApp*** para el tratamiento y seguridad en las password es el ***MD5*** por su sencillez y eficacia, el producto PHP define una función md5($var) preparada para su consumo.

## Diseño de vistas

En el siguiente diagrama se puede **identificar** a **alto nivel** el diseño de las vistas, identificándose por el tipo de usuario ‘n’ o ‘a’, las vistas comunes y las específicas de cada funcionalidad, más adelante, en el diagrama de flujo de navegación identificaré particularmente cada flujo para cada tipo de usuario.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

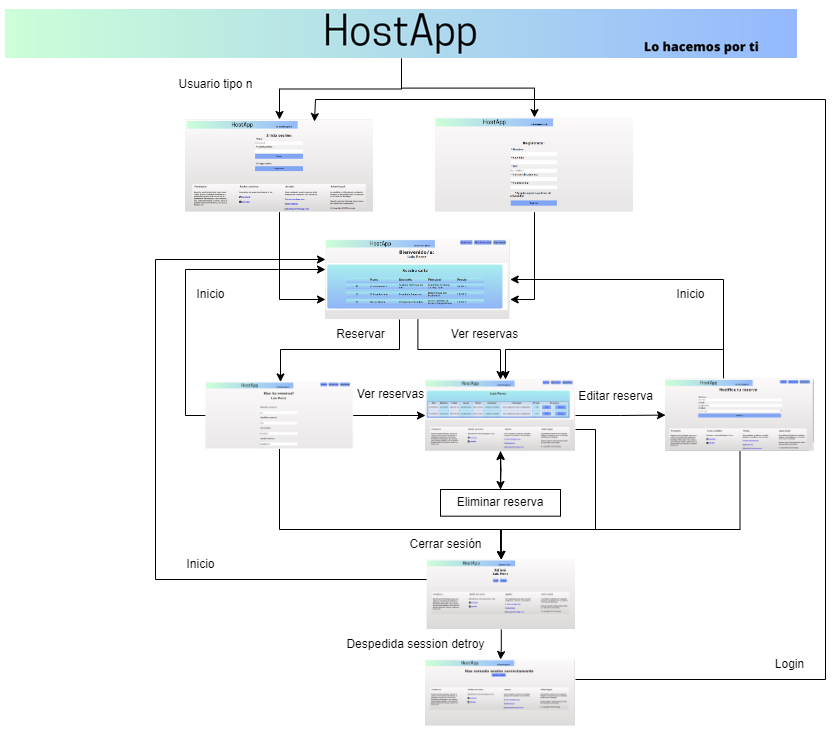
## Diagrama de flujo de navegación

En el siguiente diagrama se muestra el flujo de navegación de **HostApp** en este flujo se definen 2 posibles flujos:

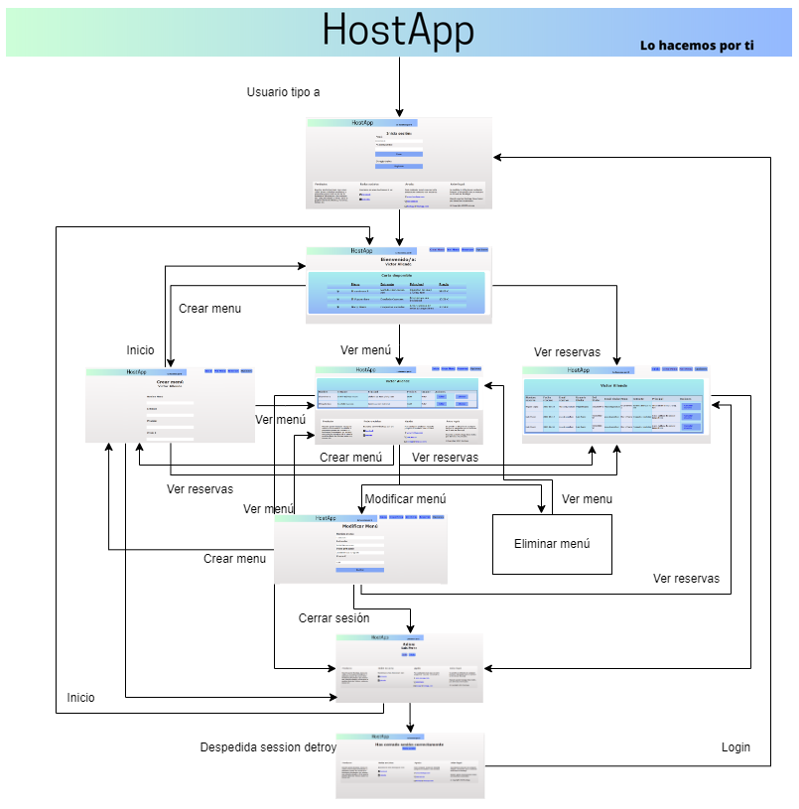
1. Usuario normal o cliente
2. Administrador del sistema

Las acciones de ambos casos o ambos tipos de usuario se verán más en detalle en el diagrama de casos de uso, más adelante.

### Flujo de navegación usuarios tipo ‘n’



### Flujo de navegación usuarios tipo ‘a’



## Diagrama de peticiones web

Esquema de llamadas WEB (Get y Post) y funcionalidades de ***HostApp***, se describe cada llamada desde cada vista o formulario, sean get o post, así como su tratamiento por parde de la capa controler y su transacción con la capa de modelo de negocio.





## Diagrama de clases

El siguiente diagrama enseña las clases y ficheros principales de la aplicación, así también se muestra como interactúan entre los mismos.

Esta es toda la funcionalidad y lógica de negocio de ***HostApp***.

Texto

Descripción generada automáticamente

## Diagrama entidad relación

A continuación, presento el diagrama de entidad relación del modelo de datos utilizado por ***HostApp***, en este diagrama se presentan las 3 tablas con las que ***HostApp*** ofrece su funcionalidad.

Diagrama

Descripción generada automáticamente

### Tabla usuarios

Almacena los usuarios registrados en ***HostApp*** tanto, ***administradores*** tipo ‘a’ de ***HostApp*** dados de alta por el administrador de la base de datos y usuarios tipo ‘n’ o clientes que se registran a través de la WEB.

### Tabla Menu

En esta taba se almacena los menús creados por los administradores, son visibles desde la UI para los usuarios tipo ‘n’ y los tipos ‘a’, en el caso de los administradores tipo ‘a’ pueden ver todos los menús creados por todos los administradores, pero solo se permiten acciones sobre los menús propios creados por el administrador logeado en la sesión, esto da una capa de seguridad a ***HostApp*** no permitiendo el borrado intencional o accidental de los menús del negocio. Los usuarios tipo ‘n’ tienen visibilidad sobre todos los menús para la selección en la fase de reserva.

### Tabla Reservas

En esta tabla se almacenan las reservas realizadas por los usuarios tipo ‘n’ a través de ***HostApp***, cada usuario tipo ‘n’ tiene a su disposición sus reservas en la UI, pudiendo actuar sobre las mismas, editar o eliminar.

Los usuarios tipo ‘a’ o administradores tiene visibilidad sobre todas las reservas, no se permite su edición, pero si su eliminación o cancelación.

## Diagrama de base de datos

El diagrama de base de datos presentado a continuación muestra las 3 tablas detalladas anteriormente.

Tabla **usuario**, almacena los usuarios registrados en ***HostApp***, la tabla **menu** que almacena los menús creados por los administradores de ***HostApp*** tiene como FK un usuario administrador, que es el creador y dueño de ese menú, solo se permiten acciones al administrador creador del menú, tiene visibilidad sobre todos los menús de ***HostApp,*** pero no el control del total sobre los menús en su totalidad,

La tabla **reservas** almacena las reservas realizadas por los clientes registrados en ***HostApp.*** Se permite la edición y eliminación a los clientes realizadores de la reserva, a los administradores se les da permiso de delete sobre las mismas.

Diagrama

Descripción generada automáticamente

## Diagrama de casos de uso

En el siguiente listado de casos de uso, se muestra cómo se interactúa con ***HostApp*** con los diferentes perfiles usuario cliente y usuario administrador,

Inmediatamente después se detallará cada uno de los casos de uso individualmente y se amplía su información.

|  |  |  |  |
| --- | --- | --- | --- |
| **Listado de casos de uso *HostApp*** | | | |
| Código | CU-001 | Caso de uso: | Registro |
| Código | CU-002 | Caso de uso: | Login |
| Código | CU-003 | Caso de uso: | Ver menú |
| Código | CU-004 | Caso de uso: | Crear reserva |
| Código | CU-005 | Caso de uso: | Actualizar reserva |
| Código | CU-006 | Caso de uso: | Eliminar reserva |
| Código | CU-007 | Caso de uso: | Cerrar sesión |
| Código | CU-008 | Caso de uso: | Ver menú administrador |
| Código | CU-009 | Caso de uso: | Crear menú administrador |
| Código | CU-010 | Caso de uso: | Actualizar menú administrador |
| Código | CU-011 | Caso de uso: | Eliminar menú administrador |
| Código | CU-012 | Caso de uso: | Ver reservas administrador |
| Código | CU-013 | Caso de uso: | Eliminar reservas administrador |

Diagrama

Descripción generada automáticamente

|  |  |  |
| --- | --- | --- |
| Diagrama  Descripción generada automáticamente   |  | | --- | |  | | |
| **Caso de uso** | |
| Registro | |
| **Código** | **Descripción** |
| CU-001 | Usuario realiza registro en la app. |
| **Tipo de usuario** | |
| Usuario normal o cliente de la app. | |
| **Condiciones previas** | **Condiciones posteriores** |
| No tener cuenta creada en la app. | Se realiza el registro en la app. Se crea cuenta en la app. Redirección a bienvenida.php |
| **Datos de entrada** | **Datos de salida** |
| Datos personales formulario registro.php | Se crea un nuevo usuario |
| **Tablas modelo de datos** | **Clases** |
| usuarios | ModeloController  Modelo  session.php |
| **Interfaces gráficas / vistas** | |
| registro.php bienvenida.php | |

|  |  |  |
| --- | --- | --- |
| Diagrama  Descripción generada automáticamente   |  | | --- | |  | | |
| **Caso de uso** | |
| Login | |
| **Código** | **Descripción** |
| CU-002 | Usuario realiza login en la app. |
| **Tipo de usuario** | |
| Ambos tipos de usuarios de la app | |
| **Condiciones previas** | **Condiciones posteriores** |
| Tener cuenta creada en la app. | Se Realiza el login en la app. Redirecciona a bienvenida.php o bienvenida\_admin.php con el listado total de menús disponibles. |
| **Datos de entrada** | **Datos de salida** |
| Datos personales formulario login.php | Datos del usuario que realiza login  Inicio sesión |
| **Tablas modelo de datos** | **Clases** |
| Usuarios  menu | ModeloController  Modelo  session.php |
| **Interfaces gráficas / vistas** | |
| login.php bienvenida.php bienvenida\_admin.php | |

|  |  |  |
| --- | --- | --- |
| |  | | --- | |  | | |
| **Caso de uso** | |
| Ver menú | |
| **Código** | **Descripción** |
| CU-003 | Usuario visualiza lista de menú |
| **Tipo de usuario** | |
| Usuario normal o cliente de la app. | |
| **Condiciones previas** | **Condiciones posteriores** |
| Estar logeado en la app. Existir menú creados en la app | Se Realiza la visualización de los menús de la app |
| **Datos de entrada** | **Datos de salida** |
| N/A | Listado de menús |
| **Tablas modelo de datos** | **Clases** |
| menu | ModeloController  Modelo  session.php |
| **Interfaces gráficas / vistas** | |
| mostrar\_menu.php bienvenida.php bienvenida\_admin.php | |

|  |  |  |
| --- | --- | --- |
| |  | | --- | |  | | |
| **Caso de uso** | |
| Crear reserva | |
| **Código** | **Descripción** |
| CU-004 | Usuario realiza una nueva reserva en la app. |
| **Tipo de usuario** | |
| Usuario normal o cliente de la app. | |
| **Condiciones previas** | **Condiciones posteriores** |
| Estar logeado en la app. Existir menú creados en la app | Se Realiza la reserva de un menú.  Redirección a mostrar\_menu.php |
| **Datos de entrada** | **Datos de salida** |
| Datos para crear reserva en formulario en crear\_reserva.php | Una nueva reserva del usuario logeado |
| **Tablas modelo de datos** | **Clases** |
| reservas | ModeloController  Modelo  session.php |
| **Interfaces gráficas / vistas** | |
| crear\_reserva.php  mostrar\_reservas.php | |

|  |  |  |
| --- | --- | --- |
| Diagrama  Descripción generada automáticamente   |  | | --- | |  | | |
| **Caso de uso** | |
| Actualizar reserva | |
| **Código** | **Descripción** |
| CU-005 | Usuario actualiza datos de una reserva en la app. |
| **Tipo de usuario** | |
| Usuario normal o cliente de la app. | |
| **Condiciones previas** | **Condiciones posteriores** |
| Estar logeado en la app. Tener reservas creadas. | Se actualiza la reserva de un menú. Redirección a mostrar\_reservas.php |
| **Datos de entrada** | **Datos de salida** |
| Datos para modificar una reserva en formulario en actualizar\_reserva.php | Una reserva actualizada del usuario logeado |
| **Tablas modelo de datos** | **Clases** |
| reservas | ModeloController Modelo session.php |
| **Interfaces gráficas / vistas** | |
| actualizar\_reserva.php  mostrar\_reservas.php | |

|  |  |  |
| --- | --- | --- |
| Diagrama  Descripción generada automáticamente   |  | | --- | |  | | |
| **Caso de uso** | |
| Eliminar reserva | |
| **Código** | **Descripción** |
| CU-006 | Usuario elimina una reserva en la app. |
| **Tipo de usuario** | |
| Usuario normal o cliente de la app. | |
| **Condiciones previas** | **Condiciones posteriores** |
| Estar logeado en la app. Tener reservas creadas. | Se elimina la reserva de un menú. Redirección a mostrar\_reservas.php |
| **Datos de entrada** | **Datos de salida** |
| La reserva para eliminar | Resto de reservas del usuario logeado |
| **Tablas modelo de datos** | **Clases** |
| reservas | ModeloController Modelo session.php |
| **Interfaces gráficas / vistas** | |
| mostrar\_reservas.php | |

|  |  |  |
| --- | --- | --- |
| Diagrama  Descripción generada automáticamente   |  | | --- | |  | | |
| **Caso de uso** | |
| Cerrar sesión | |
| **Código** | **Descripción** |
| CU-007 | Usuario logeado cierra sesión en la app. |
| **Tipo de usuario** | |
| Ambos tipos de usuarios de la app | |
| **Condiciones previas** | **Condiciones posteriores** |
| Estar logeado en la app. | Se elimina la sesión del usuario logeado. Redirección a despedida.php y  posteriormente a login.php |
| **Datos de entrada** | **Datos de salida** |
| Usuario logeado en sesión | N/A |
| **Tablas modelo de datos** | **Clases** |
| N/A | ModeloController Modelo session.php |
| **Interfaces gráficas / vistas** | |
| logout.php despedida.php | |

|  |  |  |
| --- | --- | --- |
| Diagrama  Descripción generada automáticamente   |  | | --- | |  | | |
| **Caso de uso** | |
| Ver menú administrador | |
| **Código** | **Descripción** |
| CU-008 | Visualización del total de menús disponibles en la app para su gestión. |
| **Tipo de usuario** | |
| Usuarios administradores de la app | |
| **Condiciones previas** | **Condiciones posteriores** |
| Ser administrador de la app. Estar logeado en la app. Que existan menús creados por el administrador logeado | Se retorna una lista de menús creados por el administrador logeado para su gestión. |
| **Datos de entrada** | **Datos de salida** |
| Usuario logeado en sesión | Lista de menús del usuario logeado en sesión |
| **Tablas modelo de datos** | **Clases** |
| menu | ModeloController Modelo session.php |
| **Interfaces gráficas / vistas** | |
| mostrar\_menu.php | |

|  |  |  |
| --- | --- | --- |
| Diagrama  Descripción generada automáticamente   |  | | --- | |  | | |
| **Caso de uso** | |
| Crear menú administrador | |
| **Código** | **Descripción** |
| CU-009 | Formulario de creación de menú por parte del administrador |
| **Tipo de usuario** | |
| Usuarios administradores de la app | |
| **Condiciones previas** | **Condiciones posteriores** |
| Ser administrador de la app. Estar logeado en la app. | Se retorna una lista de menús creados por el administrador logeado incluyendo el nuevo menú para su gestión. Redirección a mostrar\_menu.php |
| **Datos de entrada** | **Datos de salida** |
| Datos de formulario crear\_menu.php | Un nuevo menú creado y añadido a la lista de menús totales. |
| **Tablas modelo de datos** | **Clases** |
| menu | ModeloController Modelo session.php |
| **Interfaces gráficas / vistas** | |
| crear\_menu.php mostrar\_menu.php | |

|  |  |  |
| --- | --- | --- |
| Diagrama  Descripción generada automáticamente   |  | | --- | |  | | |
| **Caso de uso** | |
| Actualizar menú administrador | |
| **Código** | **Descripción** |
| CU-010 | Formulario de actualización de menú por parte del administrador |
| **Tipo de usuario** | |
| Usuarios administradores de la app | |
| **Condiciones previas** | **Condiciones posteriores** |
| Ser administrador de la app. Estar logeado en la app. Tener menús creados. | Se retorna una lista de menús creados por el administrador logeado incluyendo el menú actualizado para su gestión. Redirección a mostrar\_menu.php |
| **Datos de entrada** | **Datos de salida** |
| Datos de formulario actualizar\_menu.php | Un menú actualizado y añadido a la lista de menús totales. |
| **Tablas modelo de datos** | **Clases** |
| menu | ModeloController Modelo session.php |
| **Interfaces gráficas / vistas** | |
| actualizar\_menu.php mostrar\_menu.php | |

|  |  |  |
| --- | --- | --- |
| |  | | --- | | Diagrama  Descripción generada automáticamente | | |
| **Caso de uso** | |
| Eliminar menú administrador | |
| **Código** | **Descripción** |
| CU-011 | Eliminar un menú por parte del administrador. |
| **Tipo de usuario** | |
| Usuarios administradores de la app | |
| **Condiciones previas** | **Condiciones posteriores** |
| Ser administrador de la app. Estar logeado en la app. Tener menús creados. | Se retorna una lista actualizada de menús disponibles o creados por el administrador logeado para su gestión. |
| **Datos de entrada** | **Datos de salida** |
| Menú identificado para su eliminación | Listado de menús actualizado. |
| **Tablas modelo de datos** | **Clases** |
| menu | ModeloController Modelo session.php |
| **Interfaces gráficas / vistas** | |
| mostrar\_menu.php | |

|  |  |  |
| --- | --- | --- |
| |  | | --- | | Diagrama  Descripción generada automáticamente | | |
| **Caso de uso** | |
| Ver reservas administrador | |
| **Código** | **Descripción** |
| CU-012 | Visualizar todas las reservas realizadas en la app. |
| **Tipo de usuario** | |
| Usuarios administradores de la app | |
| **Condiciones previas** | **Condiciones posteriores** |
| Ser administrador de la app. Estar logeado en la app.  Que existan reservas realizadas en la app. | Se retorna una lista de todas las reservas creadas en la app para su gestión. |
| **Datos de entrada** | **Datos de salida** |
| N/A | Listado de todas las reservas realizadas en la app. |
| **Tablas modelo de datos** | **Clases** |
| reservas | ModeloController Modelo session.php |
| **Interfaces gráficas / vistas** | |
| mostrar\_reservas\_admin.php | |

|  |  |  |
| --- | --- | --- |
| |  | | --- | | Diagrama  Descripción generada automáticamente | | |
| **Caso de uso** | |
| Eliminar reservas administrador | |
| **Código** | **Descripción** |
| CU-013 | Eliminación de reservas realizadas en la app. |
| **Tipo de usuario** | |
| Usuarios administradores de la app | |
| **Condiciones previas** | **Condiciones posteriores** |
| Ser administrador de la app. Estar logeado en la app. Que existan reservas realizadas en la app. | Se retorna una lista de las reservas existentes en la app para su gestión. |
| **Datos de entrada** | **Datos de salida** |
| La reserva a eliminar | Listado de todas las reservas existentes y actualizadas de la app. |
| **Tablas modelo de datos** | **Clases** |
| reservas | ModeloController Modelo session.php |
| **Interfaces gráficas / vistas** | |
| mostrar\_reservas\_admin.php | |

# Metodología

Como metodología para los desarrollos y al ser un único recurso, no aplica o no es suficientemente necesario utilizar alguna metodología ágil como ***Scrum*** o ***Kanban*** etc.

Sin embargo, se ha utilizado un sistema de tareas a realizar, tareas en curso, tareas finalizadas, fechas de entrega y poder realizar hitos. Se ha utilizado ***Asana*** en su periodo de pruebas, ya que ***Asana*** requiere licencia:

Escala de tiempo

Descripción generada automáticamente

## Diagrama de Gantt

En el siguiente diagrama de Gantt se puede visualizar la programación de los trabajos de desarrollo cronológicamente.

Se ha de destacar, que en este diagrama solo se han incluido aquellas tareas netamente de programación o desarrollos. Los análisis de las tecnologías a emplear, análisis funcionales, análisis de arquitectura del proyecto, análisis de bases de datos o el modelo de datos a emplear están fuera del alcance de este diagrama de Gantt.

El tiempo utilizado para los desarrollos incluyendo la realización de la memoria es de: **72.5 horas**. El tiempo total de esfuerzo entre desarrollos, formación y la totalidad de los análisis realizados es de **103.5 horas**.



## Cuadro de tiempos

En el siguiente cuadro se muestran los tiempos utilizados en las distintas fases de los desarrollos:



# Tecnologías

Las tecnologías utilizadas en este proyecto son tecnologías que permiten el desarrollo de aplicaciones web, son tecnologías gratuitas y open source y son las siguientes:

**PHP 7.4.27**

Como lenguaje de programación se ha utilizado php, por su potencia en el desarrollo de aplicaciones web, su robustez como lenguaje de entorno servidor y por su implementación del patrón de diseño MVC.

**Servidor Apache de XAMMP**

El entorno de servidor que se ha utilizado es Apache, que viene incorporado en el paquete de aplicaciones que ofrece XAMPP, es de código libre, integra una serie de plataformas para el desarrollo como servidor de bases de datos MySQL además de un intérprete de PHP, por lo que es una alternativa muy potente para los desarrollos de ***HostApp***.

**MySql MariaDB**

La base de datos utilizada es MariaDB que es una bifurcación de MySQL a partir de su versión 5.1

**SQL**

El lenguaje para la comunicación y persistencia de ***HostApp*** en base de datos es SQL

**HTML**

Para las estructuras de las vistas de php se ha utilizado html

**CSS**

La tecnología utilizada para los estilos de las vistas de la aplicación es CSS

**GitHub**

Como repositorio y para el control de versiones se ha utilizado la tecnología Git a través de la plataforma online GitHub, es necesario tener o crear una cuenta de usuario. Es muy eficiente, tiene mucho soporte y una gran cantidad de foros de información.

**SourceTree**

Para gestionar Git se ha instalado el cliente Git SourceTree ya que es de código libre, muy intuitivo, fácil y de gran rendimiento, demás de su sistema gráfico que es muy útil cuando se necesitan realizar operaciones sobre el repositorio

# Trabajos futuros

En este apartado se señalan una serie de trabajos futuros o desarrollos para realizar en ***HostApp,*** son evolutivos que llevarán la aplicación a nuevas funcionalidades e integraciones, a continuación, se presenta una línea de trabajos futuros o evolutivos:

* Implementar o integrar un sistema de notificaciones.
* Implementar confirmación de reservas.
* Parametrizar el número de mesas disponibles según capacidad del negocio.
* Permitir registro con más documentos de identificación, ej: NIF.
* Generación de informes o estadísticas de reservas.
* Ampliar el modelo de negocio para incorporar funcionalidad de escoger ubicación de mesa en fase de reservas.
* Ampliar estilos de la aplicación, evolucionar el CSS diseñado.

# Pruebas

El plan de pruebas se realiza al finalizar los desarrollos de ***HostApp*** siguiendo el diagrama RFTP establecido en los objetivos del presente proyecto.

P-6002-1 Realizar login sin datos en formulario****

P-6002-2 Realizar login DNI erróneo mal formato



P-6002-3 Realizar login password incorrecto



P-6002-4 Realizar login correcto



P-9002-1 Crear un menú sin datos usuario admin

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

P-9002-2 Crear un menú con datos erróneos usuario admin

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

P-9002-3 Crear un menú con datos usuario admin

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Una captura de pantalla de una red social

Descripción generada automáticamente

P-9002-4 Listar todos los menús usuarios admin

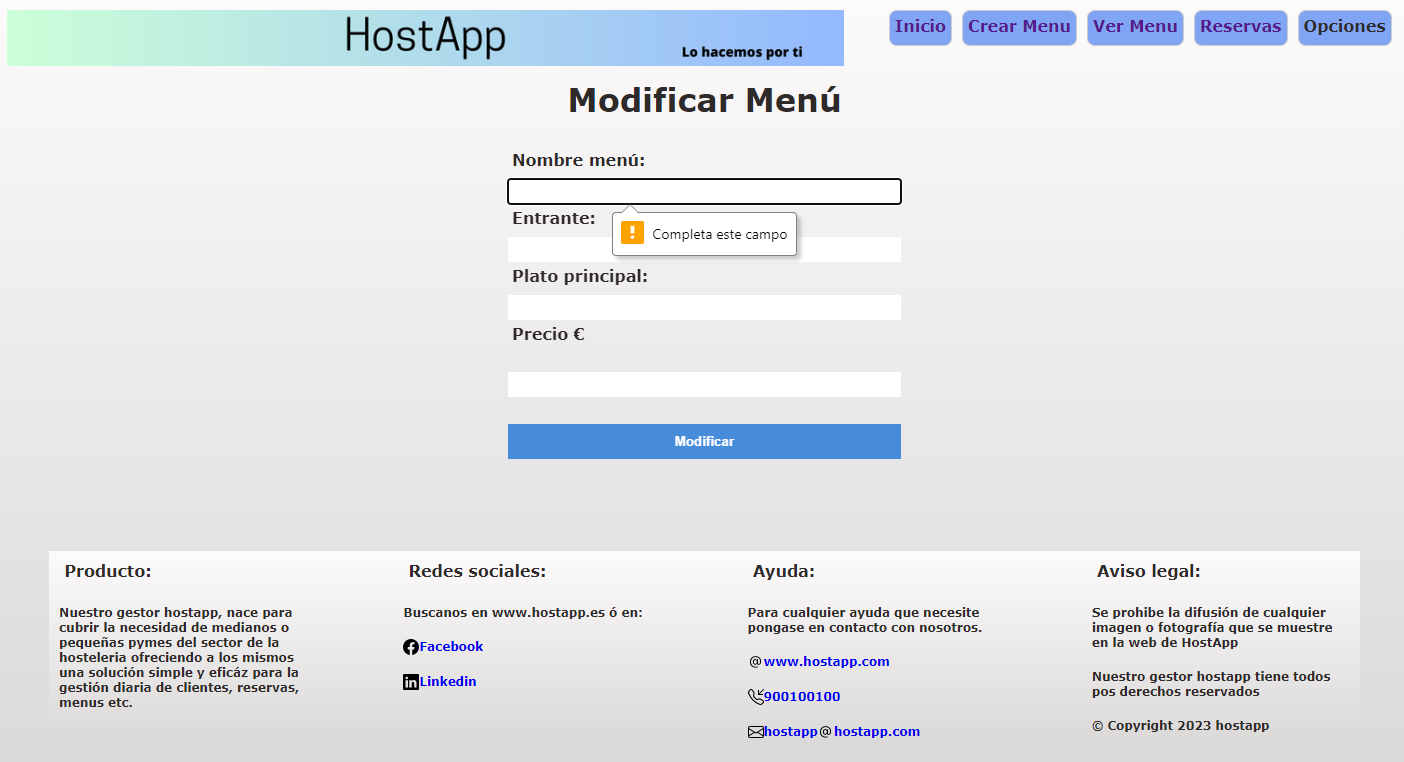
Tabla

Descripción generada automáticamente

P-9002-5 Listar los menús propios usuarios admin



P-9002-6 Modificar un menú sin datos usuario admin



P-9002-7 Modificar un menú con erróneos usuarios admin



P-9002-8 Modificar un menú con datos usuario admin

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica

Descripción generada automáticamente

P-9002-9 Eliminar un menú usuario admin

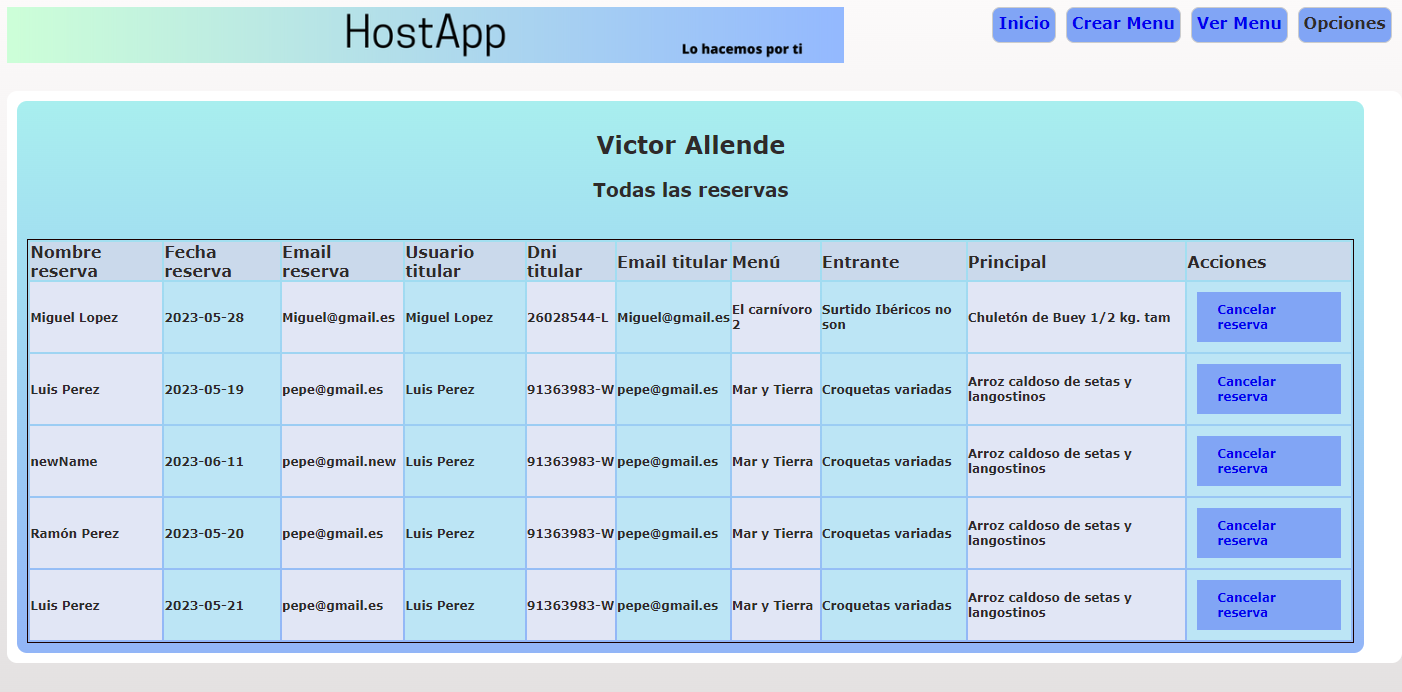
Interfaz de usuario gráfica, Tabla

Descripción generada automáticamente

Interfaz de usuario gráfica

Descripción generada automáticamente

P-9002-10 Listar reservas usuario admin



P-9002-11 Eliminar reservas usuario admin



P-9002-12 Listar todos los menús usuario cliente

Tabla

Descripción generada automáticamente

P-9002-13 Crear una reserva usuario cliente sin datos

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

P-9002-14 Crear reserva usuario cliente con datos erróneos

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

P-9002-15 Crear una reserva usuario cliente con datos

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

P-9002-16 Listar reservas usuario cliente

Interfaz de usuario gráfica

Descripción generada automáticamente

P-9002-17 Actualizar reserva usuario cliente sin datos

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

P-9002-18 Actualizar reserva cliente con datos erróneos



Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

P-9002-19 Actualizar reserva usuario cliente con datos

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Tabla

Descripción generada automáticamente

P-9002-20 Eliminar reserva usuario cliente

**Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente**

P-10002-1 Pruebas de navegabilidad entre páginas

Tabla

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Una captura de pantalla de una red social

Descripción generada automáticamente

P-11002-1 Realizar registro sin datos en formulario

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

P-11002-2 Realizar registro nombre incorrecto (caracteres)

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

P-11002-3 Realizar registro DNI erróneo mal formato)

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

P-11002-4 Realizar registro Email erróneo mal formato

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

P-11002-5 Realizar registro usuario ya existente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

P-11002-6 Realizar registro correcto

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Tabla

Descripción generada automáticamente

P-14002-1 Visualización de cerrar sesión ambos perfiles

Tabla

Descripción generada automáticamente

Tabla

Descripción generada automáticamente

P-15002-1 Proceso de cierre de sesión ambos perfiles

**Interfaz de usuario gráfica, Tabla

Descripción generada automáticamente con confianza media**

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Tabla

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

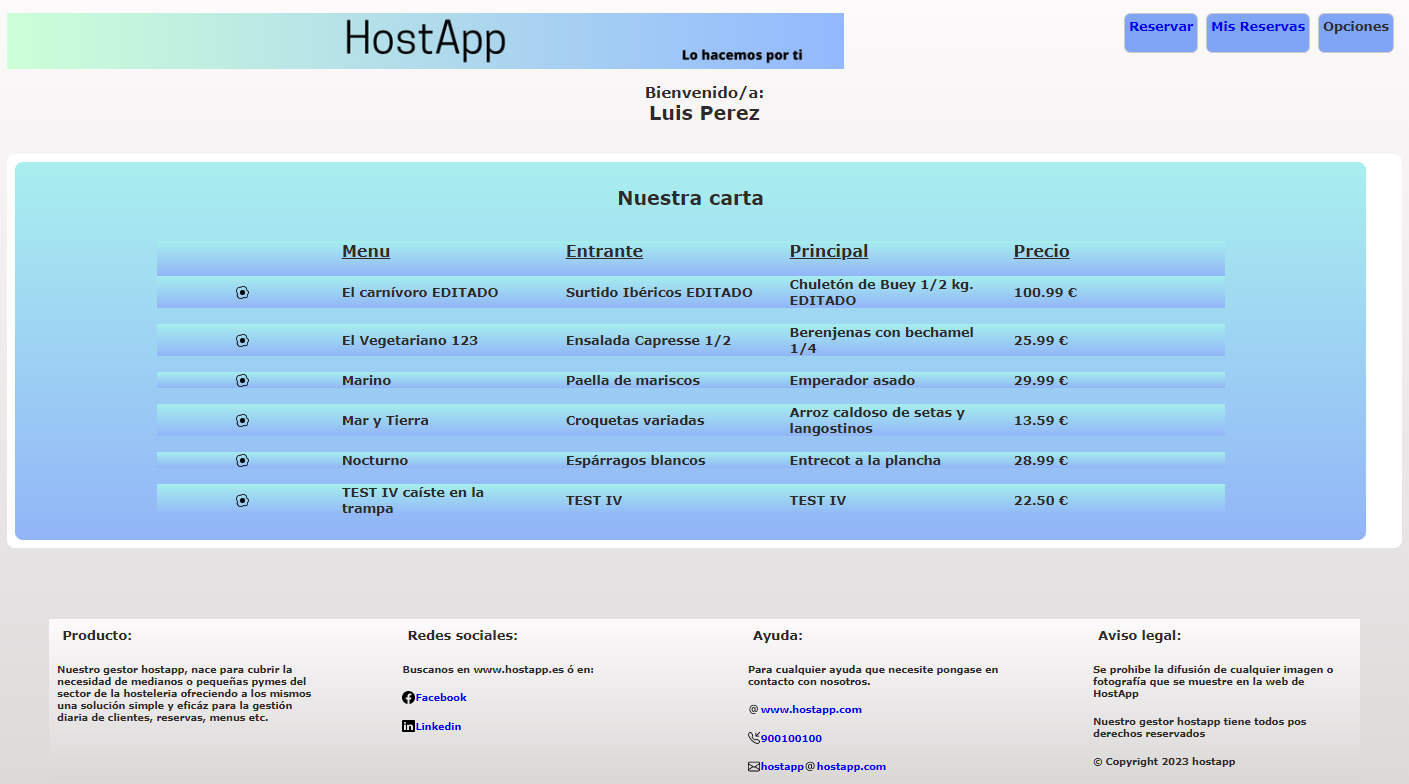
Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

P-16002-1 Visualizar reservas usuarios



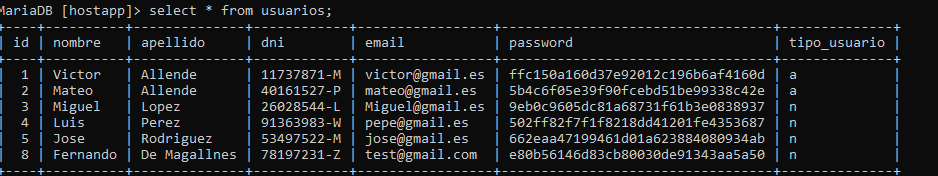
P-17002-1 Ver banner e icono en la app



P-18002-1 Probar formulario y datos en md5 en base de datos

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente



# Conclusiones

A raíz de este trabajo, se puede concluir que las tecnologías informáticas pueden y deben estar al alcance del pequeño y mediano negocio, es este sector, en particular como es la restauración, se hace imprescindible su implementación para ofrecer servicios online a sus clientes, fidelizarlos día a día y por supuesto, ser más competitivos.

Durante el desarrollo de este proyecto se ha aprendido y comprendido las necesidades fundamentales para la realización del negocio de la restauración, y, por ende, lo importante de estar presentes en el mundo web y las posibilidades que se abren una vez este tipo de negocios adaptan nuevas tecnologías.

A medida que avanzaba en el desarrollo, identificaba innumerables alternativas funcionales que serían de gran interés para incorporar a ***HostApp***, debido al tiempo que se ha realizado este proyecto, se dejan muchas ideas para futuros trabajos, que sin duda darán una mayor potencia comercial a ***HostApp***.

El objetivo maestro del presente proyecto, era construir desde cero, una aplicación web que permita operar de forma directa sobre el negocio, este objetivo se pudo cumplir con la realización de ***HostApp*** que ha cubierto en la totalidad los objetivos identificados al inicio de este proyecto, llegando así al final de este, con una aplicación robusta y capaz de ser implementada por cualquier pequeño o mediano negocio de la restauración.

# Bibliografía

A continuación, se muestran las páginas o sitios de consulta para los desarrollos:

Curso de formación MVC en PHP:

<https://www.youtube.com/watch?v=ExAs5ySWqGg>

Validación de email FILTER\_VALIDATE\_EMAIL constant in PHP:

<https://www.tutorialspoint.com/filter-validate-email-constant-in-php>

Validación de email con función filter\_var:

<https://www.w3bai.com/es/php/filter_validate_email.html#gsc.tab=0>

Filter var oficial PHP:

<https://www.php.net/manual/es/function.filter-var.php>

Validación en php campos solo texto

<https://es.stackoverflow.com/questions/453683>

Validación DNI

<https://www.jose-aguilar.com/blog/validar-dni-php/>

Calcular modulo online para validar dni

<https://miniwebtool.com/es/modulo-calculator/>

Encriptación md5

<https://blog.infranetworking.com/encriptar-con-md5-php/>

Html marcar copyright

<https://www.html.am/html-codes/character-codes/html-copyright-code.cfm>

Estilos de tabla “using nth-child in tables tr td”

<https://stackoverflow.com/questions/43493583/tbody-element-breaking-nth-child-selector>

Custom select

<https://www.antofernandez.com/personalizar-un-select-con-css/>

Menú desplegable

<https://devxdev.net/tutoriales/menu-desplegable-html-css/>

Consultas maquetación

<https://www.freecodecamp.org/espanol/news/html-vs-body-como-configurar-el-ancho-y-el-alto-para-el-tamano-de-la-pagina-completa/>

Fuentes de imágenes:

<https://www.youtube.com/@AreaCurricularInformatica>

<https://hocphp.net/tong-quan/tin-tuc/mo-hinh-mvc-trong-php/>

Definición de XAMPP:

https://es.wikipedia.org/wiki/XAMPP

# Glosario

A continuación, se definen los términos técnicos utilizados en esta memoria:

**-Diagrama RFTP**: Es el diagrama de requisitos, funcionalidades, tareas y pruebas, sus códigos agrupan los requisitos de la siguiente forma ej:

**R-2000:** Código de requisito

**F-2001**: Código de la funcionalidad

**T-2001-1**: Código de tarea

**P-2002-1**: Código de prueba

**-CRUD**: Son las siglas de las operaciones contra las bases de datos, que en inglés significan:

* Create: Crear registros
* Read: Leer registros
* Update: Actualizar registros
* Delete: Eliminar registros de la base de datos.

**-XAMPP:** es un paquete de software libre, que consiste principalmente en el sistema de gestión de bases de datos MySQL, el servidor web Apache y los intérpretes para lenguajes de script PHP y Perl. El nombre es en realidad un acrónimo: X, Apache, MariaDB/MySQL, PHP, Perl.

**-Usuario tipo ‘a’**: son los usuarios administradores del sistema

**-Usuario tipo ‘n’**: Usuario normal o cliente registrado en la aplicación

**-filter\_var() FILTER\_VALIDATE\_EMAIL:** Función y variables disponible en el api de PHP, se basa en la especialización [**RFC 882**](http://www.faqs.org/rfcs/rfc822.html) para la sintaxis de direcciones de correo

Texto

Descripción generada automáticamente con confianza media