



**Autores**: Gloria Taboada, Carlos Gómez y Oscar Robles

**Ciclo**: Grado Superior de Desarrollo de Aplicaciones Web (2º curso)

**Clase**: MP12 - Proyecto

**Tutor/a**: Francesca Tomás Artigues

Índice

[1.1. Propósito del proyecto 1](#_Toc515644921)

[1.2. Objetivos 1](#_Toc515644922)

[1.3. Herramientas utilizadas 1](#_Toc515644923)

[2.1. Recogida de requisitos 2](#_Toc515644924)

[2.2. Análisis 4](#_Toc515644925)

[2.2.1 Diagrama casos de uso 4](#_Toc515644926)

[2.2.2 Especificación de casos de uso 5](#_Toc515644927)

[2.2.1 Diagrama de clases de análisis 18](#_Toc515644928)

[2.3. Diseño 19](#_Toc515644929)

[2.3.1 Diagrama de clases de diseño 19](#_Toc515644930)

[2.3.2 Modelo conceptual y lógico de la base de datos 20](#_Toc515644931)

[2.3.3 Interfaces 21](#_Toc515644932)

[2.3. Codificación y pruebas 26](#_Toc515644933)

[3. Conclusión 26](#_Toc515644934)

[3.1. Objetivos técnicos alcanzados/no alcanzados/ampliados 26](#_Toc515644935)

[3.2. Posibles ampliaciones futuras 27](#_Toc515644936)

[3.3. Valoración personal 27](#_Toc515644937)

[4. Bibliografía 28](#_Toc515644938)

[5. Anexos 29](#_Toc515644939)

[5.1. Plan de Marketing 29](#_Toc515644940)

[5.1.1 Estudio de mercado 29](#_Toc515644941)

[5.1.2 Estrategia de marketing 31](#_Toc515644942)

[5.2. Manual de usuario 32](#_Toc515644943)

[5.3. Manual de instalación 39](#_Toc515644944)

**1. Introducción**

## Propósito del proyecto

Este proyecto nace de la idea de comodidad a la hora de reservar una habitación on-line, ampliando dichas reservas a todos los servicios ofrecidos por el hotel-balneario, tanto de servicios de habitación como de tratamientos de belleza, pasando por reservas de mesa en sus restaurantes o actividades que realice.

La idea es que el cliente del balneario pueda reservar todos los servicios on-line, escogiendo el horario disponible que más le convenga, pudiendo cancelarlo en cualquier momento sin necesidad de llamar por teléfono.

## Objetivos

El objetivo del proyecto es facilitar la gestión del personal encargado de hacer las reservas, ahorrando empleados y tiempo al delegar dicha función en el cliente.

El hotel-balneario pondrá a disposición de sus clientes una agenda encargada de mostrar las fechas y horarios disponibles para sus habitaciones, restaurantes y/o tratamientos ofrecidos.

Los clientes tendrán que registrarse en la aplicación para poder gestionar sus reservas.

Como política de www y a fin de que el cliente no pueda acaparar la agenda, se penalizara con el pago de un porcentaje del importe del servicio contratado por reserva anulada sin suficiente anterioridad o según las normas del hotel.

El cliente que no esté registrado y no desee registrarse podrá hacer reservas previo pago, pero no podrá anularlas. De esta manera incentivamos a los clientes a registrarse y facilitarnos sus datos.

## 1.3. Herramientas utilizadas

Los programas utilizados para desarrollar el proyecto han sido los siguientes: Git (GitHub), NetBeans, Composer, MySQL, PhpMyAdmin, Php, Mycript, XAMPP y un navegador web (Firefox).

**2. Desarrollo**

## 2.1. Recogida de requisitos

Interfaz para el cliente

El proyecto cuenta con una interfaz usable, ágil y de fácil navegación para el cliente. Las diferentes imágenes de cabecera y colores utilizados facilitan la vista del apartado donde se encuentra el cliente.

El menú de la barra de navegación siempre esta visible en cada apartado para una rápida navegación, al igual que el acceso de usuarios registrados o el cambio de idioma.

El diseño de cada apartado está cuidadosamente diseñado para un mayor atractivo a la vista del cliente, usando colores relajantes y diferentes tipos de fuente proporcionadas por *Google Fonts*, que están siempre disponibles para cualquier dispositivo con el que se navegue.

El diseño de la aplicación es totalmente *responsive*, atendiendo a la política *First Mobile.* Las imágenes y las fuentes utilizadas están en medidas escalables, como porcentajes o *vw (ancho del viewport),* que permiten ser expuestas en diferentes tamaños sin pérdida de calidad en su representación.

Toda la aplicación está basada en bootstrap 4.

El uso de ventanas modales para mostrar el detalle del producto seleccionado con sus imágenes correspondientes agiliza la carga de cada vista de la aplicación.

La tarea de reserva está simplificada al máximo con sus respectivas notificaciones.

El área personal del cliente registrado está diseñada para que pueda modificar sus datos personales y obtenga un recordatorio de sus anteriores reservas.

Interfaz para el administrador

El proyecto cuenta con un menú personalizado para el administrador de la aplicación que le permite ejecutar tareas de administración como la gestión de empleados, clientes y horarios.

Actores y guiones de la aplicación

|  |  |
| --- | --- |
| Actor | Administrador |
| Descripción | Usuario externo que interactúa con la aplicación para administrarla. |
| Guion | * Tiene control total sobre toda la aplicación. * Puede crear, modificar y eliminar cualquier usuario. * Asigna permisos a todos los usuarios. * Puede gestionar reservas. * Puede modificar horarios. |

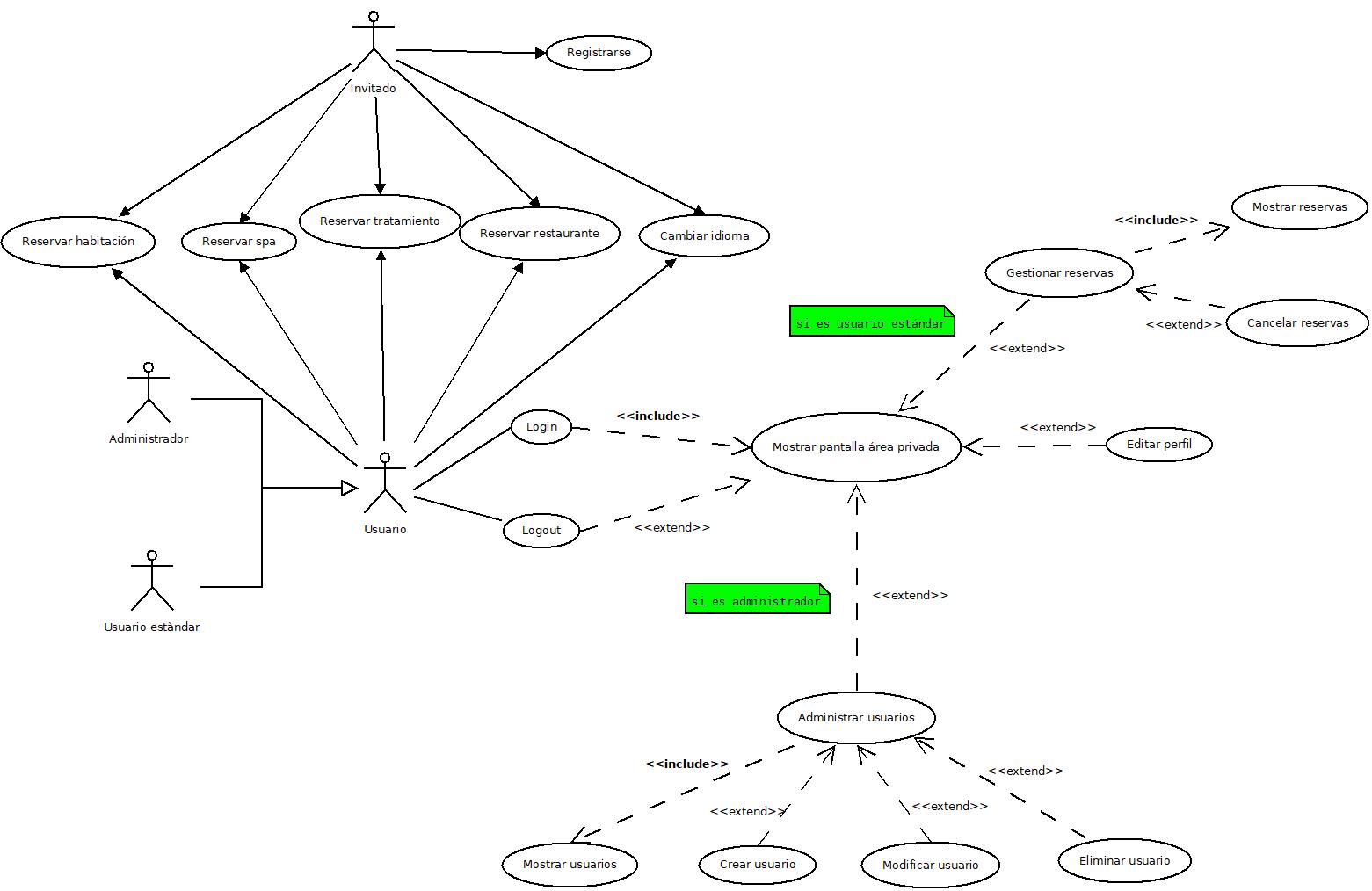
|  |  |
| --- | --- |
| Actor | Usuario |
| Descripción | Usuario externo que interactúa con la aplicación. |
| Guion | * Puede mirar toda la información de la aplicación que no requiera permisos de administrador. * Puede registrarse en la aplicación. * Puede modificar sus datos de perfil. * Puede hacer y gestionar sus reservas. * Puede anular reservas. |

|  |  |
| --- | --- |
| Actor | Visitante |
| Descripción | Usuario externo que interactúa con la aplicación. |
| Guion | * Puede mirar toda la información de la aplicación que no requiera permisos adicionales. |

## 

## 2.2. Análisis

### 2.2.1 Diagrama casos de uso



### 2.2.2 Especificación de casos de uso

|  |  |
| --- | --- |
| Nombre: Registrarse | |
| Descripción: Acceder a la aplicación | |
| Actores: Visitante | |
| PRECONDICIÓN: Nada | |
| Flujo Normal | Flujo Alternativo |
| 1. El usuario selecciona la opción para acceder al área de usuarios. |  |
| 2. Clica en el botón para registrarse y el sistema le muestra una pantalla de formulario donde ha de incluir los datos del registro. |  |
| 3. El sistema verifica los datos. | 3.1 El sistema indica que el usuario ya está registrado mostrándole la pantalla del formulario para que vuelva a insertar los datos de registro  3.2. El sistema verifica los datos.  3.3 Se produce un error. Avisa al visitante que se ha producido un error. |
| 5. El sistema registra al visitante |  |
| POSTCONDICIÓN: El usuario visitante se ha registrado. | |

|  |  |
| --- | --- |
| Nombre: Login | |
| Descripción: Acceder al área privada de la aplicación | |
| Actores: Usuario (estándar y Administrador) | |
| PRECONDICIÓN: Estar registrado en el sistema | |
| Flujo Normal | Flujo Alternativo |
| 1. El usuario selecciona la opción para hacer login. |  |
| 2. El sistema pide nombre de usuario y contraseña. |  |
| 3. El usuario introduce los datos de usuario y contraseña, clica en aceptar. | 3.1 El usuario introduce parcialmente los datos y clica en aceptar. El sistema avisa que completes el login y vuelve al punto 1.  3.2 No introduce ningún dato y clica en aceptar. El sistema avisa que completes el login y vuelve al punto 1.  3.3 El usuario no hace nada. |
| 4. El sistema valida los datos introducidos. | 4.1 El sistema no encuentra el usuario en la BBDD. El sistema avisa de que no existe el usuario y vuelve al punto 1.  4.2 El sistema encuentra el usuario, pero no coincide la contraseña. El sistema avisa de que la contraseña no es válida. |
| 5. El usuario entra en el sistema |  |
| POSTCONDICIÓN: El usuario ha podido acceder al sistema y muestra su área privada | |

|  |  |
| --- | --- |
| Nombre: Mostrar pantalla principal | |
| Descripción: Muestra la interfaz del área privada con las funcionalidades principales | |
| Actores: Usuario (estándar y Administrador) | |
| PRECONDICIÓN: Estar logueado en el sistema | |
| Flujo Normal | Flujo Alternativo |
| 1.1. El sistema muestra la interfaz del área privada y habilita las siguientes funcionalidades:  Si es administrador muestra:  · Editar perfil  · Ver reservas  · Logout  Si es administrador muestra:  · Editar perfil  · Administrar usuarios  · Logout |  |
| POSTCONDICIÓN: El sistema muestra la interfaz del área privada | |

|  |  |
| --- | --- |
| Nombre: Logout | |
| Descripción: Hacer Logout y abandonar el área privada | |
| Actores: Usuario estándar y Administrador | |
| PRECONDICIÓN: Estar logueado | |
| Flujo Normal | Flujo Alternativo |
| 1.El usuario selecciona la opción de logout |  |
| POSTCONDICIÓN: El usuario accede a la página principal de la web | |

|  |  |
| --- | --- |
| Nombre: Editar Perfil | |
| Descripción: El usuario puede cambiar sus datos personales | |
| Actores: Usuario estándar y Administrador | |
| PRECONDICIÓN: Estar logueado | |
| Flujo Normal | Flujo Alternativo |
| 1. El usuario selecciona la opción Editar Perfil |  |
| 2.El sistema muestra los datos de usuario en modo edición:   * Email * Nombre * Numero tarjeta |  |
| 3. El usuario modifica los datos y clica en guardar. | 3.El usuario no modifica los datos |
| 4.El sistema valida les modificaciones y vuelve a la página principal de usuario | 4.1. El usuario no ha modificado ningún dato. Vuelve a la página principal de usuario.  4.2. El sistema encuentra no válidas les modificaciones. |
| POSTCONDICIÓN: El sistema registra les modificaciones y vuelve a la página principal de usuario. | |

|  |  |
| --- | --- |
| Nombre: Mostrar reservas | |
| Descripción: El usuario puede ver sus reservas | |
| Actores: Usuario estándar | |
| PRECONDICIÓN: Estar logueado | |
| Flujo Normal | Flujo Alternativo |
| 1. El sistema busca las reservas que tiene el usuario |  |
| 2. El sistema devuelve las reservas seleccionadas del punto anterior. | 2. El sistema no devuelve nada por que el usuario no tiene reservas |
| POSTCONDICIÓN: Devuelve todas las reservas. | |

|  |  |
| --- | --- |
| Nombre: Gestionar reservas | |
| Descripción: El administrador puede ver las reservas realizadas | |
| Actores: Usuario | |
| PRECONDICIÓN: Estar logueado | |
| Flujo Normal | Flujo Alternativo |
| 1.El sistema busca las reservas que tiene el usuario en la BBDD |  |
| 2. El sistema devuelve las reservas seleccionadas del punto anterior. | 2. El sistema no devuelve nada porque el usuario no tiene reservas |
| 3. Si el usuario selecciona una reserva y desea cancelarla se ejecuta “Cancelar reserva” |  |
| POSTCONDICIÓN: Devuelve todas las reservas. | |

|  |  |
| --- | --- |
| Nombre: Cancelar reservas | |
| Descripción: El usuario cancela reservas | |
| Actores: Usuario estándar | |
| PRECONDICIÓN: Estar logueado | |
| Flujo Normal | Flujo Alternativo |
| 1.El sistema pregunta si desea eliminar una reserva |  |
| 2. El usuario selecciona sí | 2. El usuario selecciona no  2.1 Se vuelve a la pantalla de reservas |
| 3.Se cancela la reserva |  |
| POSTCONDICIÓN: Devuelve todas las reservas. | |

|  |  |
| --- | --- |
| Nombre: Mostrar usuarios | |
| Descripción: Mostrar usuarios del sistema | |
| Actores: Administrador | |
| PRECONDICIÓN: Estar logueado | |
| Flujo Normal | Flujo Alternativo |
| 1. El sistema busca en la bbdd todos los usuarios registrados |  |
| POSTCONDICIÓN: El sistema devuelve todos los usuarios | |

|  |  |
| --- | --- |
| Nombre: Administrar usuarios | |
| Descripción: El administrador puede ver los usuarios registrados | |
| Actores: Administrador | |
| PRECONDICIÓN: Estar logueado | |
| Flujo Normal | Flujo Alternativo |
| 1.El sistema busca los usuarios que tiene el sistema en la BBDD |  |
| 2. El sistema devuelve los usuarios seleccionadas del punto anterior. | 2. El sistema no devuelve nada porque el sistema no tiene usuarios registrados |
| 3.1 Si el administrador selecciona un usuario y desea eliminarlo se ejecuta “Eliminar usuario”  3.2 Si el administrador selecciona un usuario y desea modificarlo se ejecuta “Modificar usuario”  3.3 Si el administrador desea crear un usuario se ejecuta “Crear usuario” |  |
| POSTCONDICIÓN: Devuelve todos los usuarios. | |

|  |  |
| --- | --- |
| Nombre: Eliminar usuario | |
| Descripción: El administrador elimina usuarios | |
| Actores: Administrador | |
| PRECONDICIÓN: Estar logueado | |
| Flujo Normal | Flujo Alternativo |
| 1. El sistema pregunta si desea eliminar ese usuario |  |
| 2. El usuario selecciona “sí” | 2. El usuario selecciona “no”  2.1 Se vuelve a la pantalla de reservas |
| 3.Se elimina el usuario |  |
| POSTCONDICIÓN: El usuario se ha eliminado y devuelve todos los usuarios. | |  |

|  |  |
| --- | --- |
| Nombre: Crear Usuario | |
| Descripción: El administrador da de alta usuarios | |
| Actores: Administrador | |
| PRECONDICIÓN: Estar logueado | |
| Flujo Normal | Flujo Alternativo |
| 1. El usuario selecciona la opción Crear Usuario |  |
| 2.El sistema muestra los datos de usuario en modo edición:   * Email * Nombre |  |
| 3. El sistema verifica los datos. | 3.1 El sistema indica que el usuario ya está registrado |
| 5. El sistema crea el usuario |  |
| POSTCONDICIÓN: El usuario se crea | |

|  |  |
| --- | --- |
| Nombre: Modificar Usuario | |
| Descripción: El administrador modifica usuarios | |
| Actores: Administrador | |
| PRECONDICIÓN: Estar logueado | |
| Flujo Normal | Flujo Alternativo |
| 1. El usuario selecciona la opción Modificar Usuario |  |
| 2.El sistema muestra los datos de usuario en modo edición:   * Email * Nombre |  |
| 3. El administrador modifica los datos y clicka en guardar | 3.1 El administrador no modifica los datos |
| 4.El sistema valida les modificaciones y vuelve a la página principal de administrador | 4.1. El administrador no ha modificado ningún dato. Vuelve a la página principal de administrador  4.2. El sistema encuentra no válidas les modificaciones. |
| POSTCONDICIÓN: El usuario se modifica | |

|  |  |
| --- | --- |
| Nombre: Cambiar idioma | |
| Descripción: Cambiar idioma de la aplicación | |
| Actores: Usuario (estándar y Administrador) y visitante | |
| PRECONDICIÓN: NADA | |
| Flujo Normal | Flujo Alternativo |
| 1. El usuario selecciona la opción per cambiar el idioma de la web. |  |
| POSTCONDICIÓN: La página se ha cambiado de idioma. | |

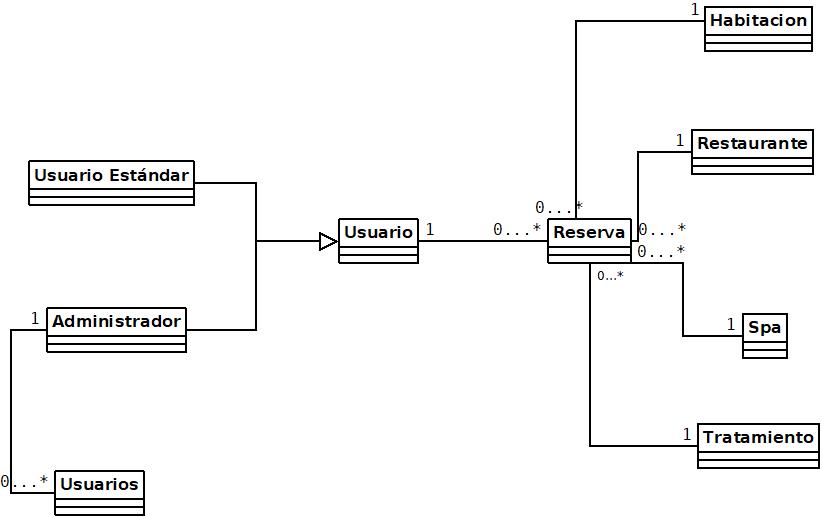
|  |  |
| --- | --- |
| Nombre: Reservar Habitación | |
| Descripción: Reservar una habitación | |
| Actores: Usuario (estándar y Administrador) y visitante | |
| PRECONDICIÓN: Nada | |
| Flujo Normal | Flujo Alternativo |
| 1. El usuario logueado o visitante selecciona la opción de hacer una reserva de habitación. |  |
| 2. El sistema muestra los datos de la reserva de la habitación  · Nombre habitación  · Fecha llegada  · Fecha salida  · Huéspedes  Si es invitado también muestra:  · Número de tarjeta |  |
| 3. El usuario introduce los datos y clica en “reservar” | 3.1 El usuario no introduce datos y clica en “volver”. |
| 3. El sistema verifica los datos | 3.1 El sistema indica que los datos introducidos no son correctos |
| 4. El sistema registra la reserva |  |
| POSTCONDICIÓN: El usuario hace la reserva y vuelve a la página de habitaciones. | |

|  |  |
| --- | --- |
| Nombre: Reservar Restaurante | |
| Descripción: Reserva en un restaurante | |
| Actores: Usuario (estándar y Administrador) y visitante | |
| PRECONDICIÓN: Nada | |
| Flujo Normal | Flujo Alternativo |
| 1. El usuario logueado o visitante selecciona la opción de hacer una reserva de restaurante. |  |
| 2. El sistema muestra los datos de la reserva del restaurante  · Nombre restaurante  · Fecha  · Comensales  · Hora  Si es invitado también muestra:  · Número de tarjeta |  |
| 3. El usuario introduce los datos y clica en “reservar” | 3.1 El usuario no introduce datos y clica en “volver”. |
| 3. El sistema verifica los datos | 3.1 El sistema indica que los datos introducidos no son correctos |
| 4. El sistema registra la reserva |  |
| POSTCONDICIÓN: El usuario hace la reserva y vuelve a la página de restaurantes. | |

|  |  |
| --- | --- |
| Nombre: Reservar Balneario-Spa | |
| Descripción: Reserva en un balneario-spa | |
| Actores: Usuario (estándar y Administrador) y visitante | |
| PRECONDICIÓN: Nada | |
| Flujo Normal | Flujo Alternativo |
| 1. El usuario logueado o visitante selecciona la opción de hacer una reserva de un balneario-spa. |  |
| 2. El sistema muestra los datos de la reserva de la balneario-spa  · Nombre balneario-spa  · Fecha  · Personas  · Hora  Si es invitado también muestra:  · Número de tarjeta |  |
| 3. El usuario introduce los datos y clica en “reservar” | 3.1 El usuario no introduce datos y clica en “volver”. |
| 3. El sistema verifica los datos | 3.1 El sistema indica que los datos introducidos no son correctos |
| 4. El sistema registra la reserva |  |
| POSTCONDICIÓN: El usuario hace la reserva y vuelve a la página de balneario-spa | |

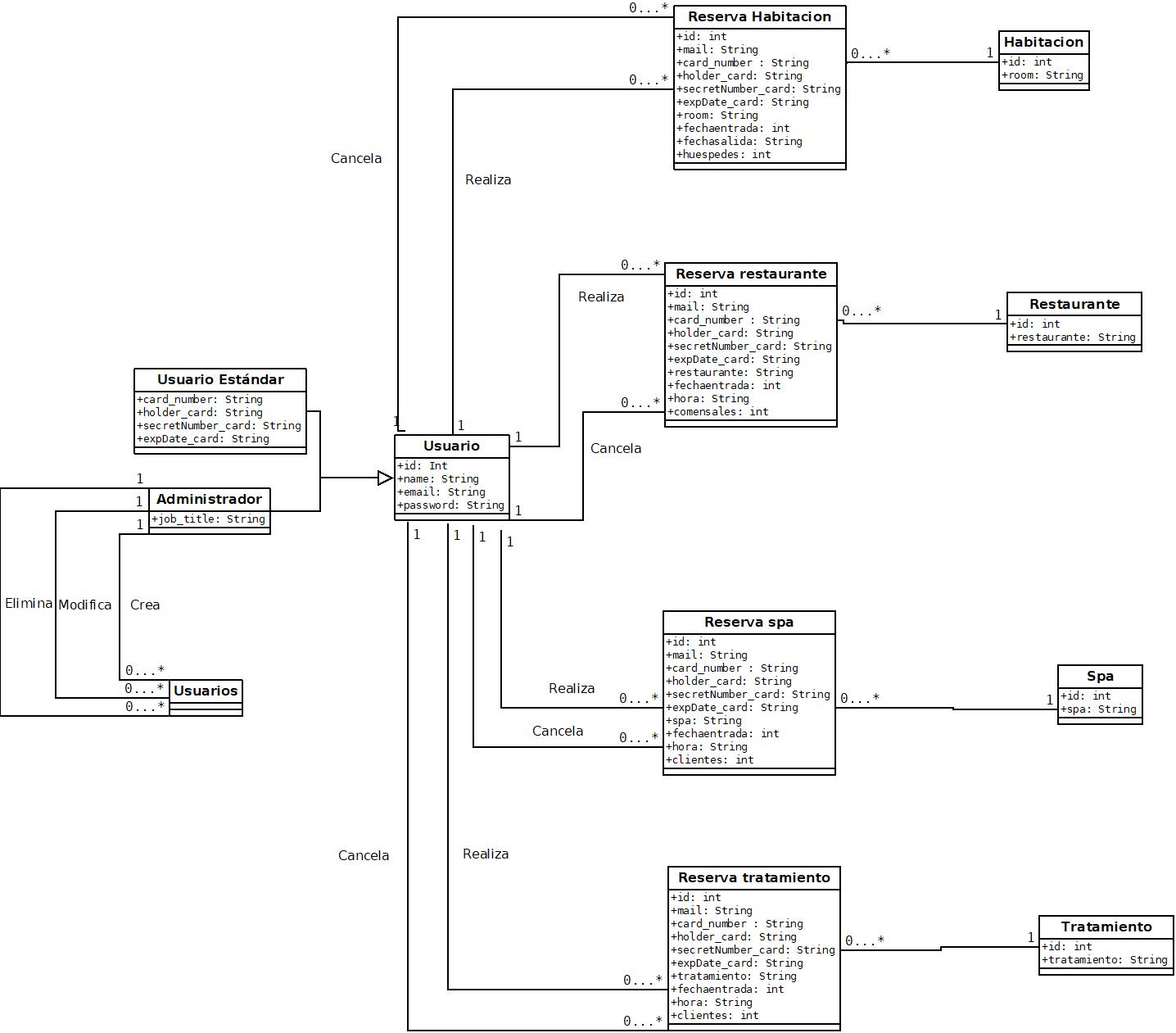
|  |  |
| --- | --- |
| Nombre: Reservar Tratamiento | |
| Descripción: Reserva un tratamiento | |
| Actores: Usuario (estándar y Administrador) y visitante | |
| PRECONDICIÓN: Nada | |
| Flujo Normal | Flujo Alternativo |
| 1. El usuario logueado o visitante selecciona la opción de hacer una reserva de un tratamiento. |  |
| 2. El sistema muestra los datos de la reserva de un tratamiento  · Nombre tratamiento  · Fecha  · Hora  Si es invitado también muestra:  · Número de tarjeta |  |
| 3. El usuario introduce los datos y clica en “reservar” | 3.1 El usuario no introduce datos y clica en “volver”. |
| 3. El sistema verifica los datos | 3.1 El sistema indica que los datos introducidos no son correctos |
| 4. El sistema registra la reserva |  |
| POSTCONDICIÓN: El usuario hace la reserva y vuelve a la página de tratamientos | |

### 2.2.1 Diagrama de clases de análisis

****

## 2.3. Diseño

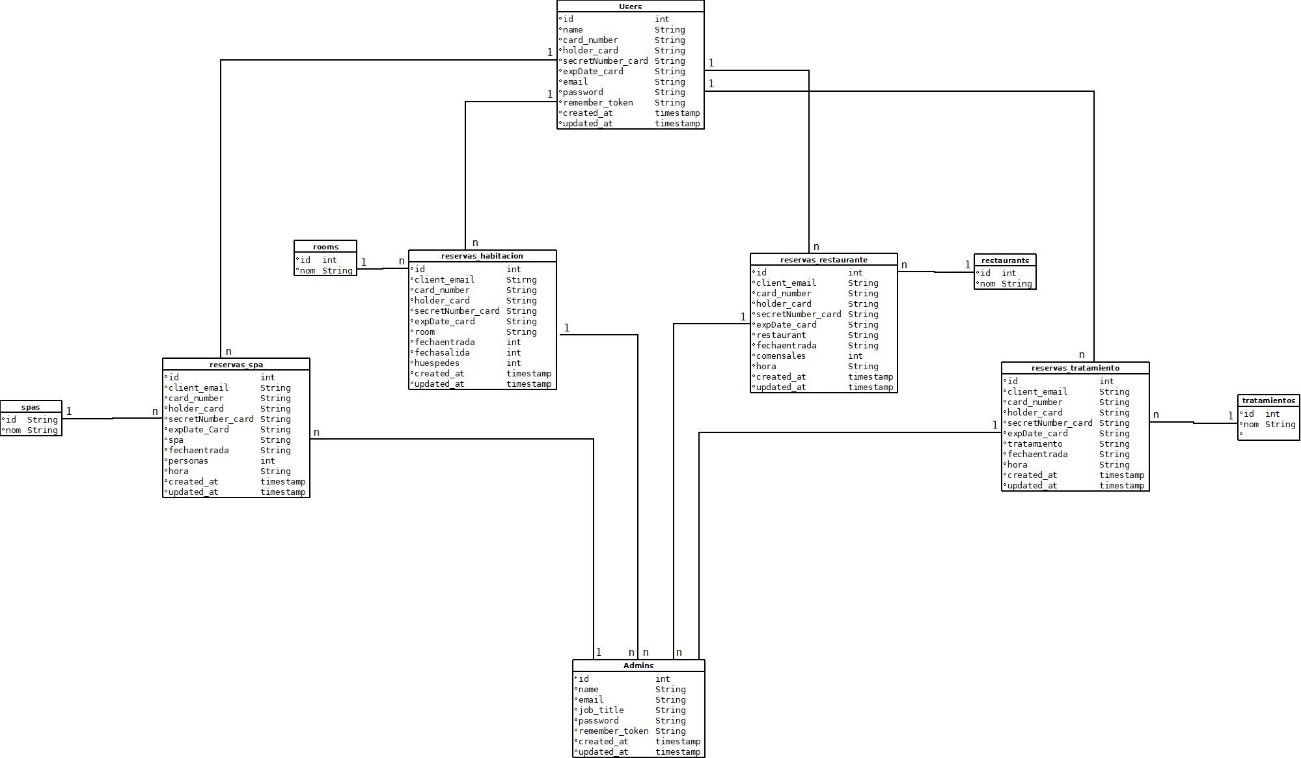
### 2.3.1 Diagrama de clases de diseño



### 

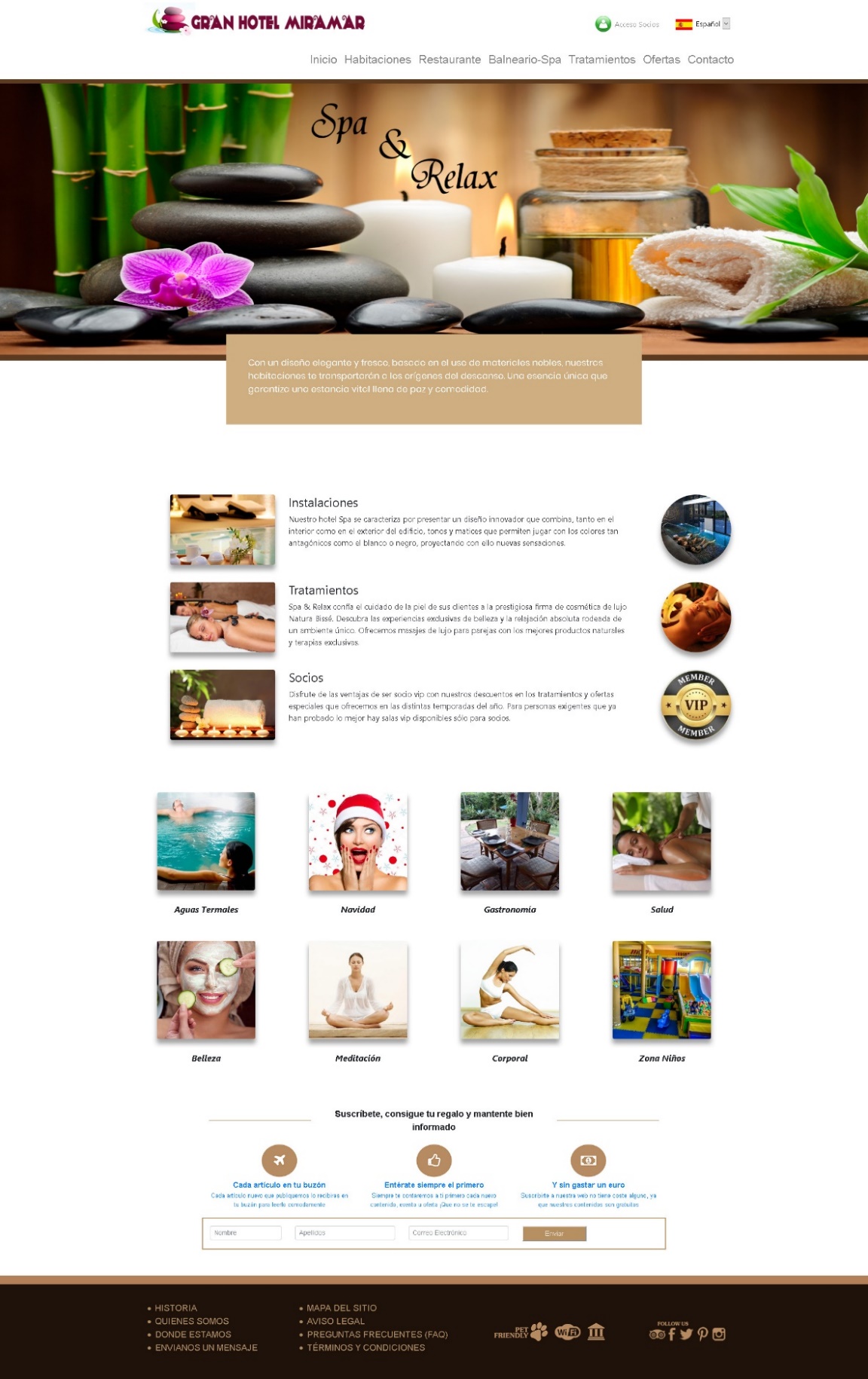
### 

### 2.3.2 Modelo conceptual y lógico de la base de datos

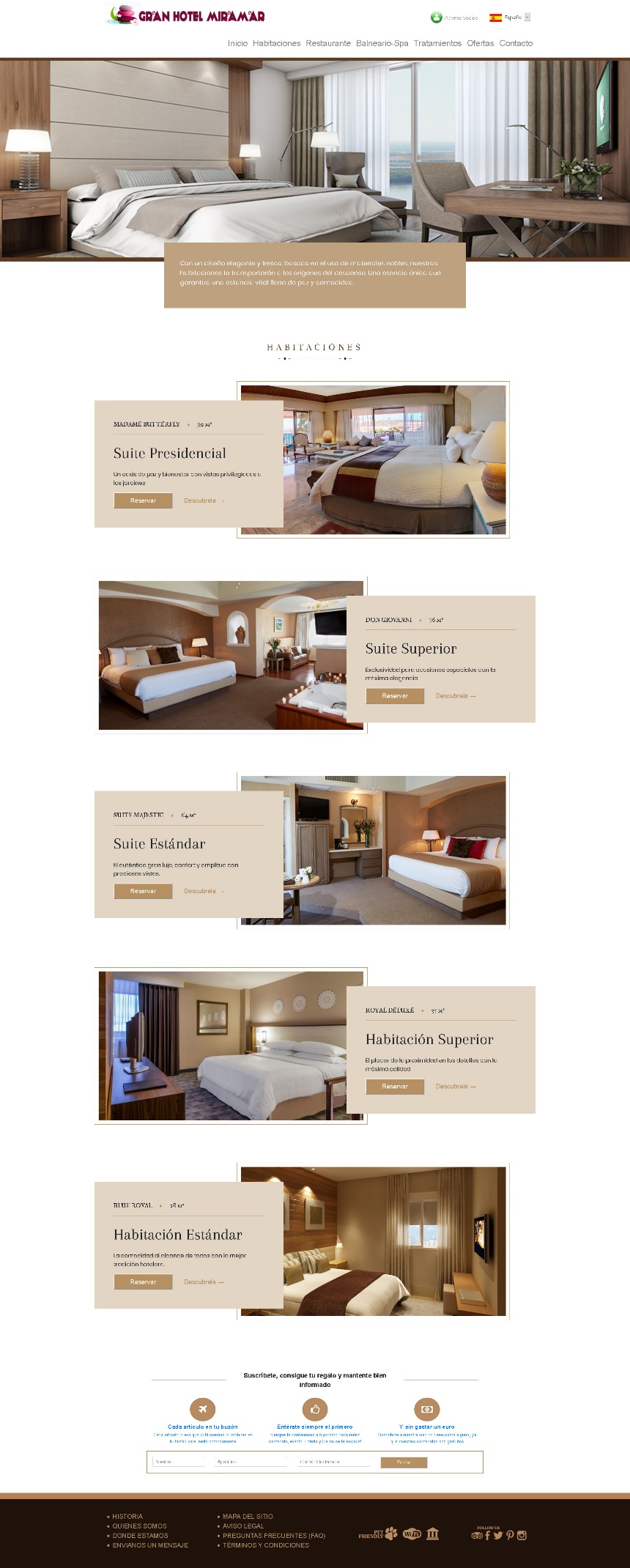


### 2.3.3 Interfaces

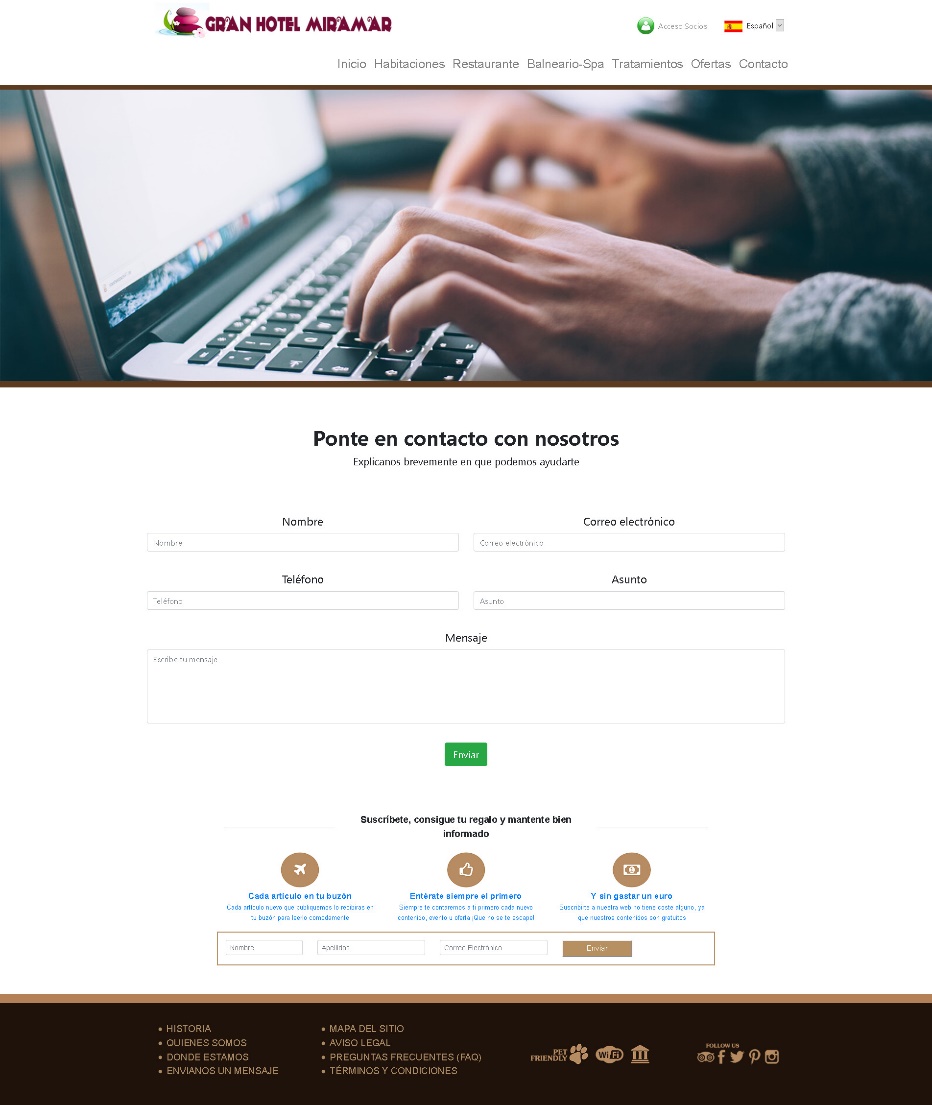
-Página Principal



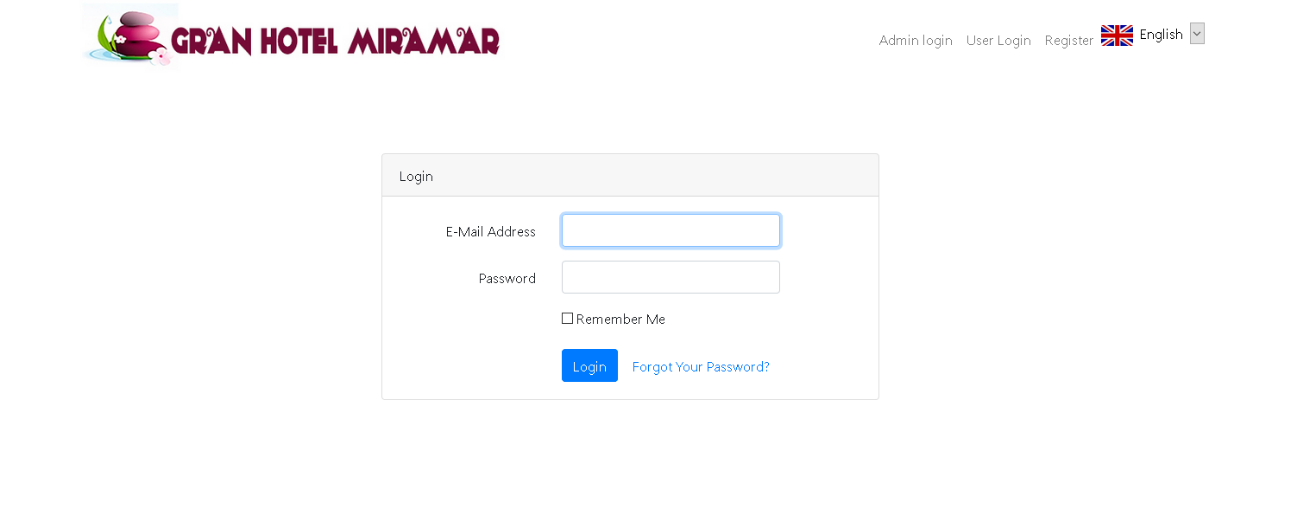
-Página Habitaciones (resto de secciones, misma estructura)



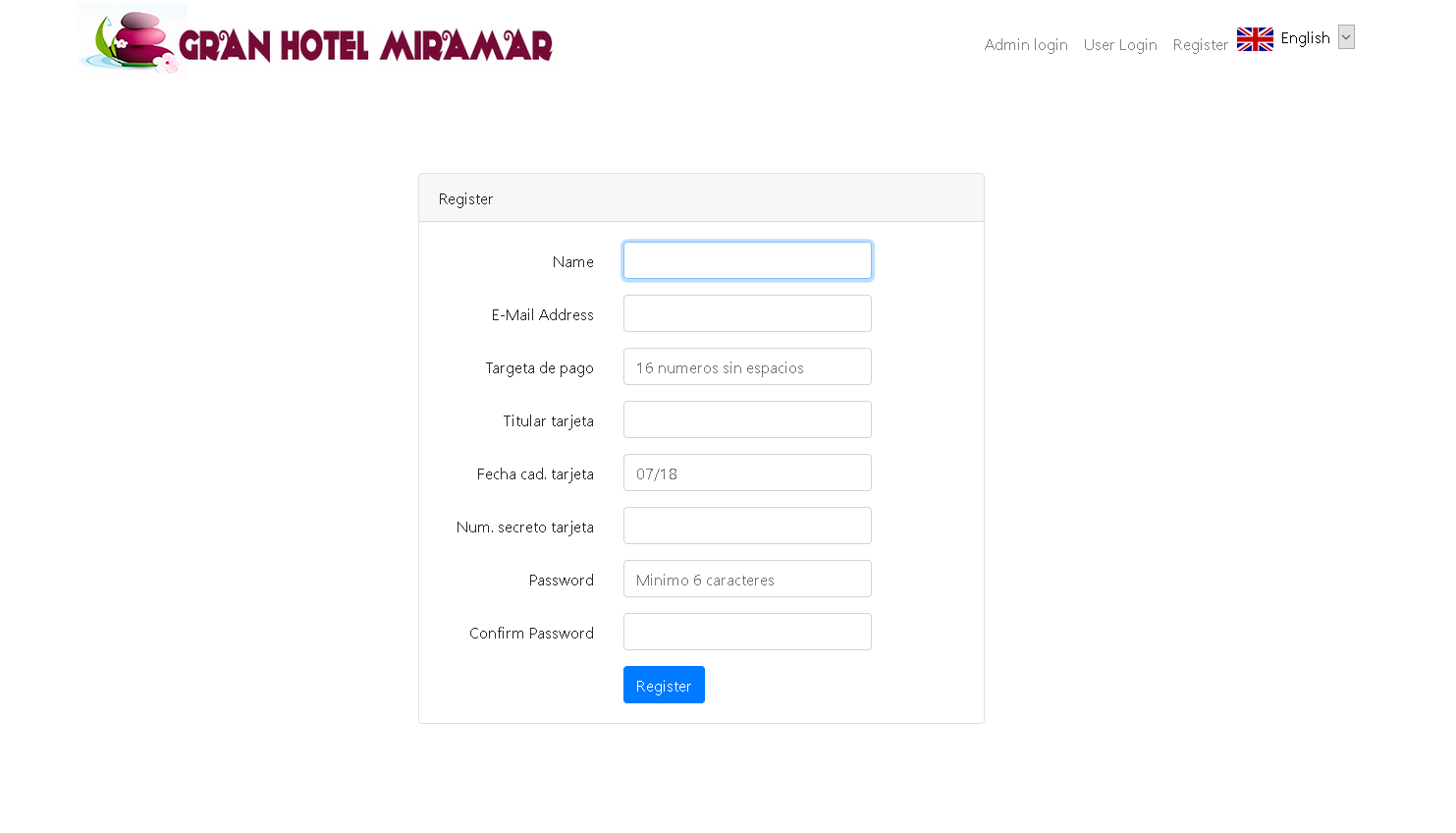
-Página Contacto



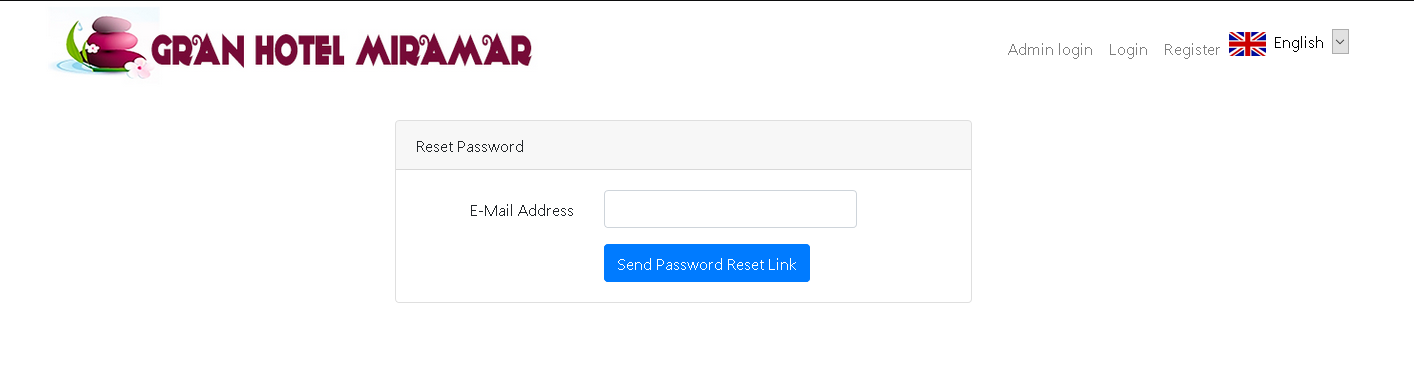
-Inicio de sesión usuario



-Registro de usuario



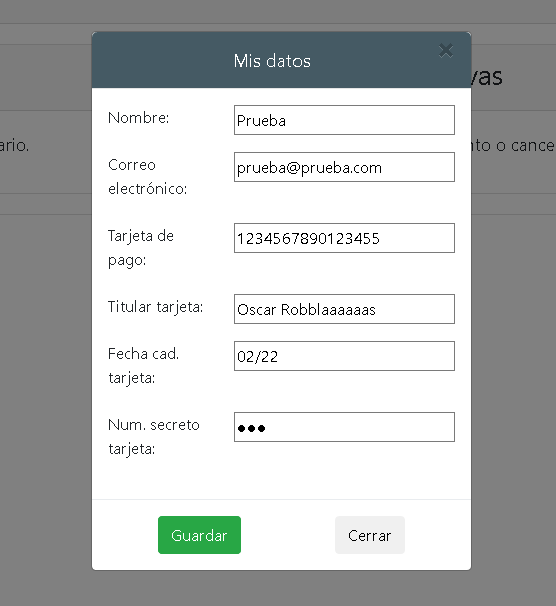
-Recuperación de contraseña



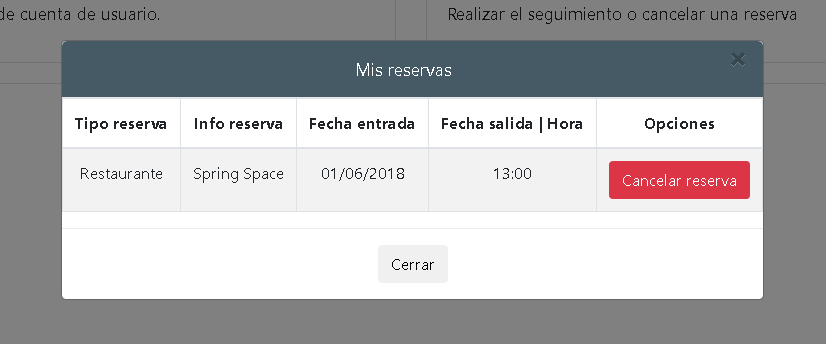
-Panel perfil de usuario



-Vista “Mis datos” (Usuario)



-Vista “Mis reservas” (Usuario)



-Panel perfil de administrador



-Vista “Mis datos” (Administrador)



-Vista “Administrar Usuarios” (Administrador)



## 2.3. Codificación y pruebas

La codificación y las pruebas de la aplicación se han llevado a cabo ente los tres miembros del grupo. Una vez que las pruebas se realizaban con éxito el código utilizado se unía con el resto del proyecto a través de la plataforma GitHub.

# 3. Conclusión

## 3.1. Objetivos técnicos alcanzados/no alcanzados/ampliados

Objetivos técnicos alcanzados

Los objetivos alcanzados son haber empezado un proyecto desde cero, planificando cada apartado y basándonos en un diseño creado a partir de una imagen adjuntada en la propuesta de proyecto, clonándola en formato web con la ayuda de css y bootstrap 4, herramienta que hemos aprendido a usar por nuestra cuenta con espectaculares resultados, ya que durante el curso hemos utilizado la versión 3.

El sistema de autentificación diferencia entre logueo de usuarios y logueo de administrador, teniendo este último todas las herramientas para crear, editar y eliminar usuarios en ventanas modales bajo petición Ajax y así optimizar la carga de información sin requerir muchos recursos.

La aplicación está diseñada para abarcar varios idiomas.

Se ha logrado con todo lo desarrollado crear una página web de reservas de un hotel para usuarios e incluso para invitados para realizar reservas si la necesidad de estar registrados en la aplicación.

Objetivos técnicos no alcanzados

Como objetivo técnico no alcanzado sería la implementación de un calendario donde se reflejarán los días que ya están ocupados.

No obstante, es un objetivo no alcanzado por falta de tiempo y no por falta de conocimientos.

Objetivos técnicos ampliados

Un objetivo técnico ampliado a tener en cuenta es el sistema empleado para traducir el contenido de la página, que facilita mucho la tarea, pudiendo traducir a cualquier idioma el contenido de un apartado solo con añadir un nuevo archivo con la traducción del texto deseado.

## 3.2. Posibles ampliaciones futuras

-Una posible ampliación futura sería añadir los idiomas más usados habitualmente.

-Otra posible ampliación sería enviar un email al usuario que quisiera reservar habitación u otro servicio no habiendo estado está disponible para las fechas seleccionadas, que previamente habría cancelado otro usuario.

-La posibilidad de ponerse en contacto por medio del envío del formulario de contacto a través de correo electrónico con el administrador de la aplicación.

-La importación de otro acceso a reservas donde se puedan ver todos los tipos de reservas del hotel (habitaciones, restaurante, spa-balneario, y tratamientos) en una misma vista.

-Encriptación de datos bancarios.

## 3.3. Valoración personal

*Oscar*

En general este proyecto me ha servido como un trabajo para poner en práctica los conocimientos aprendidos durante este curso y sobre todo en la asignatura de Proyecto (M14 - Desarrollo de una aplicación web) ya que en ella hemos aprendido a gestionar un proyecto desde 0. En general el objetivo principal del proyecto ha sido alcanzado, aunque existen muchas mejoras posibles para la aplicación.

*Carlos*

Me parece que es un buen proyecto, aunque se podrían ampliar funcionalidades, pero hemos tenido que ajustarnos al tiempo. Como comentan mis compañeros gracias a la asignatura de M14 hemos podido poner en practica lo aprendido durante el curso.

Como punto a mejorar sería poder hacer un mejor diseño, análisis y planificación del proyecto, pero como comentaba anteriormente la gran falta de tiempo hacía que tuviésemos que adaptarnos a la entrega final.

*Gloria Taboada*

Mi valoración personal es muy positiva, teniendo en cuenta que este proyecto cuenta con un gran peso en la parte de diseño, cosa que me encanta desarrollar.

Creo que una cosa fundamental a la hora de crear una aplicación es hacerla atractiva a la vista, facilitando así su navegación por los diferentes apartados.

Desde el principio hemos coincidido todos los compañeros en las herramientas con las que trabajaríamos para crear el proyecto, que son las aprendidas durante el curso, y cada uno es más habilidoso en unas o en otras, así que el reparto de tareas ha sido fácil.

# 4. Bibliografía

Páginas web:

* Laravel – Manual de Laravel

<https://laravel.com/>

* Bootstrap – Manual de Bootstrap

<https://getbootstrap.com/>

* YouTube – Guía instalación Laravel multilenguaje

<https://www.youtube.com/watch?v=rUl4O1kspvk>

* YouTube – Guía instalación Laravel con autentificación (usuarios y administradores)

Parte 1: <https://www.youtube.com/watch?v=iKRLrJXNN4M>

Parte 2: [https://www.youtube.com/watch?v=Ir2nAD9UDGg](https://www.youtube.com/watch?v=Ir2nAD9UDGg%20)

Parte 3: <https://www.youtube.com/watch?v=P8T3MjZPDdI>

# 5. Anexos

## 5.1. Plan de Marketing

### 5.1.1 Estudio de mercado

Localización

La sede principal de la empresa seria en Terrassa, pero no solo nos limitamos a un público de esta población.

Características y tendencias del mercado

Las aplicaciones web es un sector económico que mueve millones de euros. El mundo ha evolucionado mucho en estos últimos años y a nivel de aplicaciones web sigue evolucionando, por eso cualquier empresa de servicios desea tener su aplicación web para usuarios o clientes.

A la hora de tener que desarrollar una página web los hoteles son más precavidos. No quieren tener una página poco funcional y con un diseño poco atractivo, desean una página web donde el cliente se sienta cómodo y sea de fácil uso

El mercado de las aplicaciones web para reserva de hoteles está muy desarrollado, pero la expectativa es que siga creciendo debido a la aparición de nuevos lenguajes de programación y necesidades del cliente.

Análisis de la demanda

Con la página web del hotel se pretende que el hotel pueda seguir realizando sus tareas sin tener que preocuparse por atender a los clientes para hacer reservas. Nuestro cliente es muy exigente y sabe que quiere tanto el cómo sus clientes. No dejará que sus clientes reserven de cualquier forma.

La demanda busca la comodidad y buen servicio con un presupuesto ajustado. Por eso es importante que nuestra aplicación web tenga un buen diseño y una buena funcionalidad.

Análisis de la competencia

En Terrassa hay más de una veintena de empresas que se dedican al diseño y creación de páginas web. Con esta cantidad de empresas realizando las mismas funciones se han de realizar esfuerzos para mantener la posición en el mercado.

A parte de realizar un gran trabajo se han de realizar campañas publicitarias para diferenciar los productos que se ofrecen para poder competir con los precios de la competencia o la mejora del servicio a nuestros clientes

Viabilidad comercial (DAFO)

- Puntos fuertes:

* Formación académica sobre desarrollo de aplicaciones web que hemos recibido estos dos últimos años en el instituto Nicolau Copérnic.
* Poca inversión económica
* Alta motivación del proyecto
* Somos personas jóvenes y entusiastas.

- Puntos débiles:

* No tener conocimientos en tener un negocio propio

- Oportunidades:

* La confianza de los clientes en las aplicaciones web en sus negocios

- Amenazas:

* La buena posición de las empresas ya instaladas
* La fuerte competencia del sector
* El gran poder financiero de las grandes empresas

### 5.1.2 Estrategia de marketing

El producto

La mayoría de los que buscan un hotel miran en Internet antes que en una agencia de viajes. 8 de cada 10 personas opinan que la red es el medio más fiable para conocer o ampliar información sobre una marca o producto. Nuestro producto está repartido en los siguientes puntos:

- La web se visualizará desde cualquier dispositivo móvil de forma que la búsqueda de información y la compra desde smartphones y Tablet sean experiencias especialmente cómodas y sencillas.

- Mejora de la visibilidad de la página hacienda que su contenido sea más legibles y atractivo para los buscadores, posicionándose así en las primeras posiciones de búsqueda

- Gestionamos de forma gratuita la contratación del hosting y la disponibilidad y el registro del nombre de dominio de la web.

- Consultoría web de desarrollo a lo largo de todo el proceso y resoluciones técnicas durante el primer mes sin coste adicional alguno.

El precio

Todos nuestros servicios cuentan con precios independientes para todos los clientes. El precio medio por web profesional será de 249€.

La promoción

Los instrumentos a seguir respecto a los clientes será la del descuento en los precios. En la campaña de introducción se hará un descuento del 10% en todos sus productos durante los primeros dos meses de la inauguración, para incentivar al público a conocernos.

La distribución

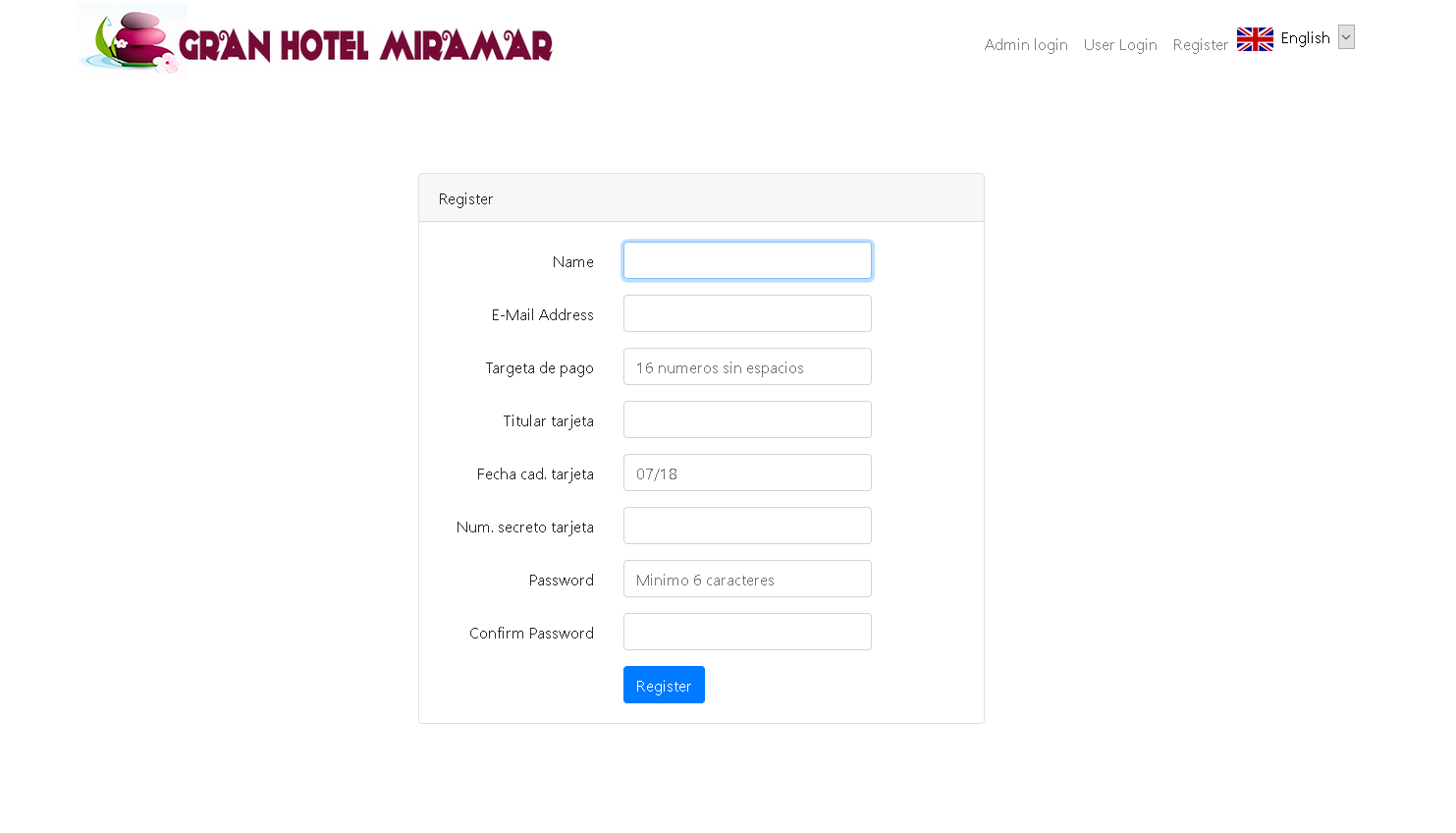
Funcionará como una empresa mayorista. Su función será la de vender productos al minorista, para que este los ofrezca al consumidor final.

## 5.2. Manual de usuario

**-Registro de usuario estándar**

1. Rellena los campos correctamente.

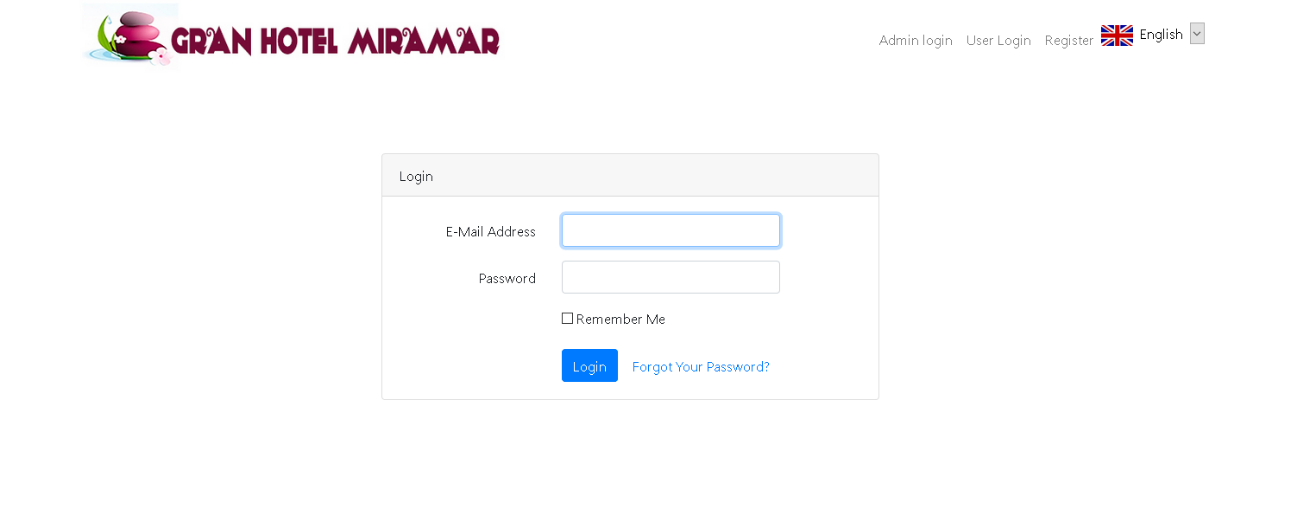
2. Clica el botón de registro.



**-Inicio de sesión de usuario**

1. Introduce tus datos de usuario.

2. Clica en botón de iniciar sesión.

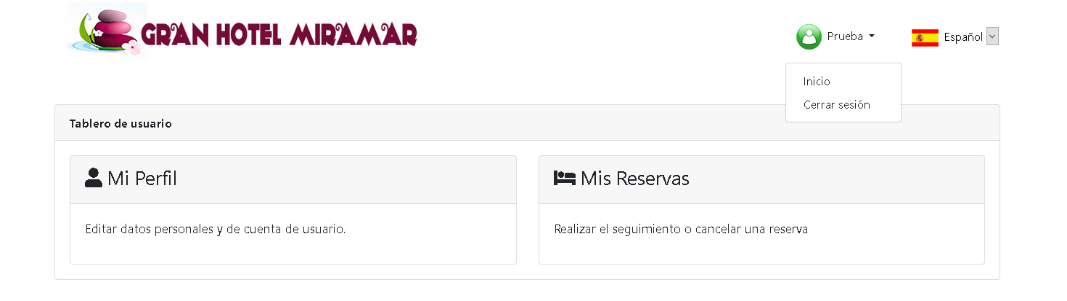


**-Mostrar Panel Usuario**

1. Clica en “Mi Perfil” para acceder a tus datos de usuario.

1.2. Clica en “Mis Reservas “para ver tus reservas y poder cancelarlas.

1.3. Clica en tu usuario y en el desplegable en “Cerrar sesión” para cerrar la sesión de tu usuario.

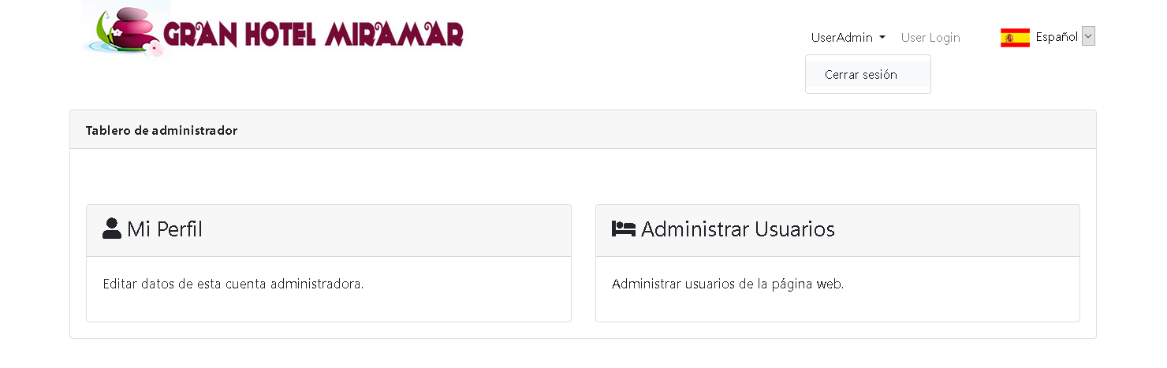


**-Mostrar Panel Administrador**

1. Clica en “Mi Perfil” para acceder a tus datos de usuario administrador.

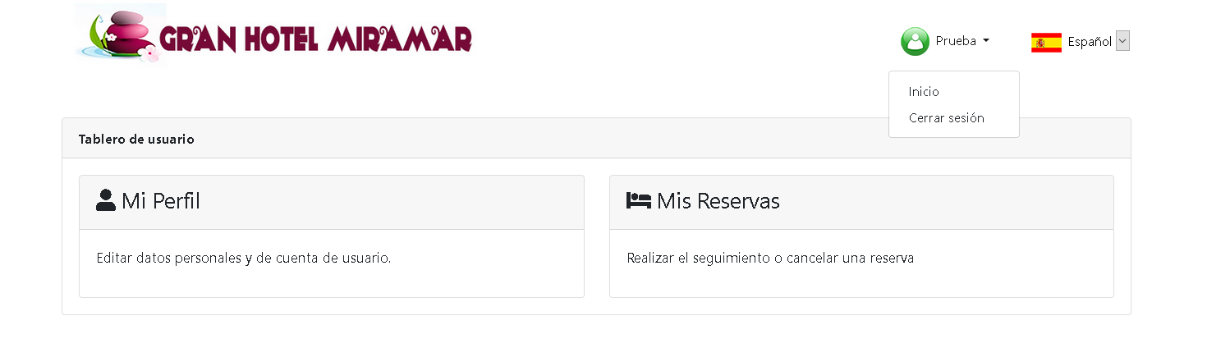
1.2. Clica en “Administrar Usuarios “para ver los usuarios de la aplicación y poder editar sus datos y darlos de baja.

1.3. Clica en tu usuario y en el desplegable en “Cerrar sesión” para cerrar la sesión de tu usuario administrador.



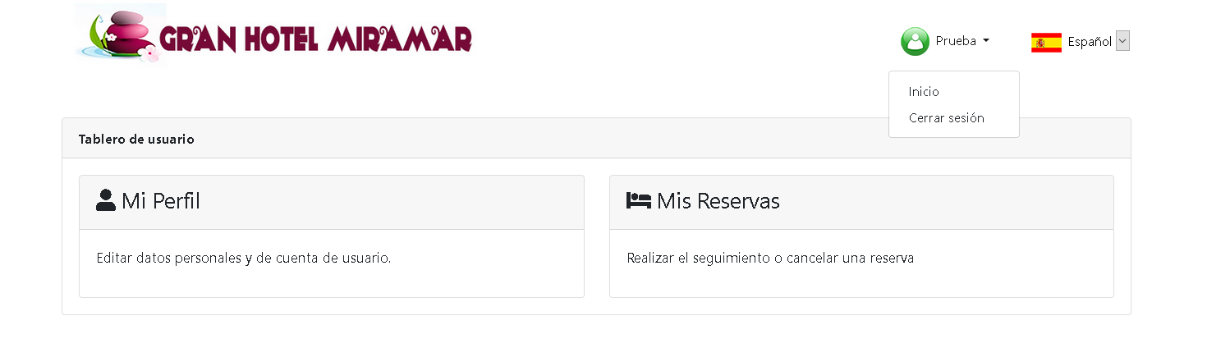
**-Cerrar sesión**

1. Clica en tu usuario y en el desplegable en “Cerrar sesión” para cerrar la sesión de tu usuario estándar o administrador.

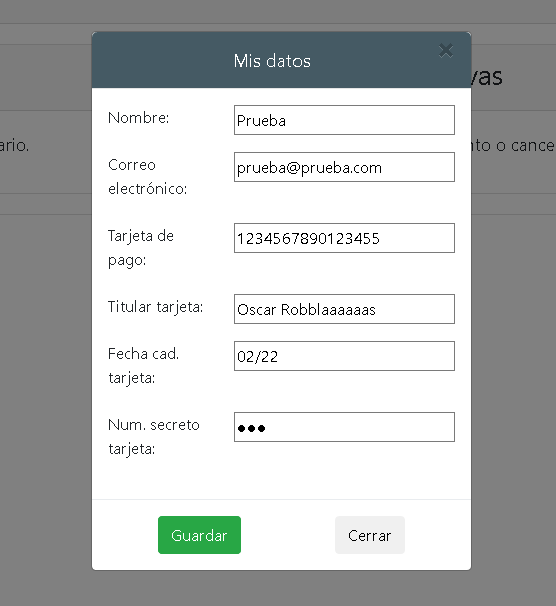


**-Editar Perfil**

1. Clica en “Mi Perfil” para acceder a tus datos de usuario estándar o administrador.



2. Edita los datos que consideres.



USUARIO ESTANDAR USUARIO ADMINISTRADOR

3. Clica guardar para modificar tus datos de usuario estándar o administrador.

**-Mostrar reservas**

1. Clica en “Ver Reservas” para poder ver tus reservas.

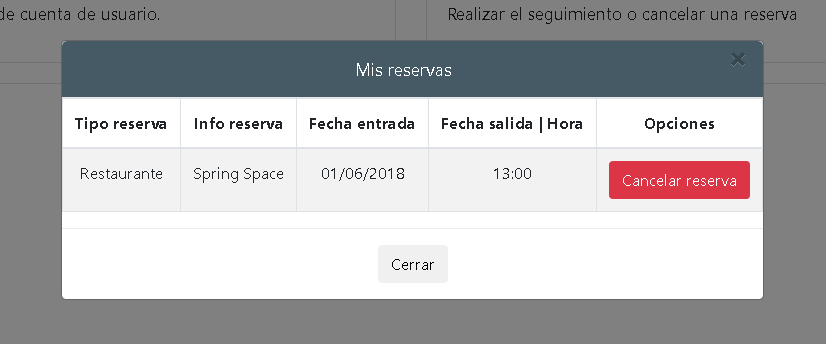


**-Gestionar reservas**

1. Clica en “Ver Reservas” para poder ver tus reservas.



2. Puedes revisar tus reservas y cancelarlas.

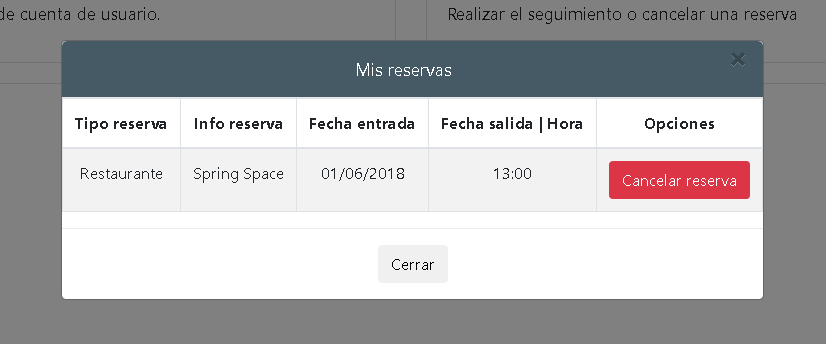


**-Cancelar reserva**

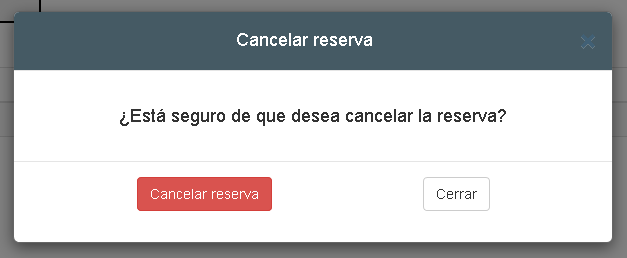
1. Clica en “Ver Reservas” para poder ver tus reservas.



2. Clica en cancelar reserva.



3. Confirma la cancelación de la reserva.



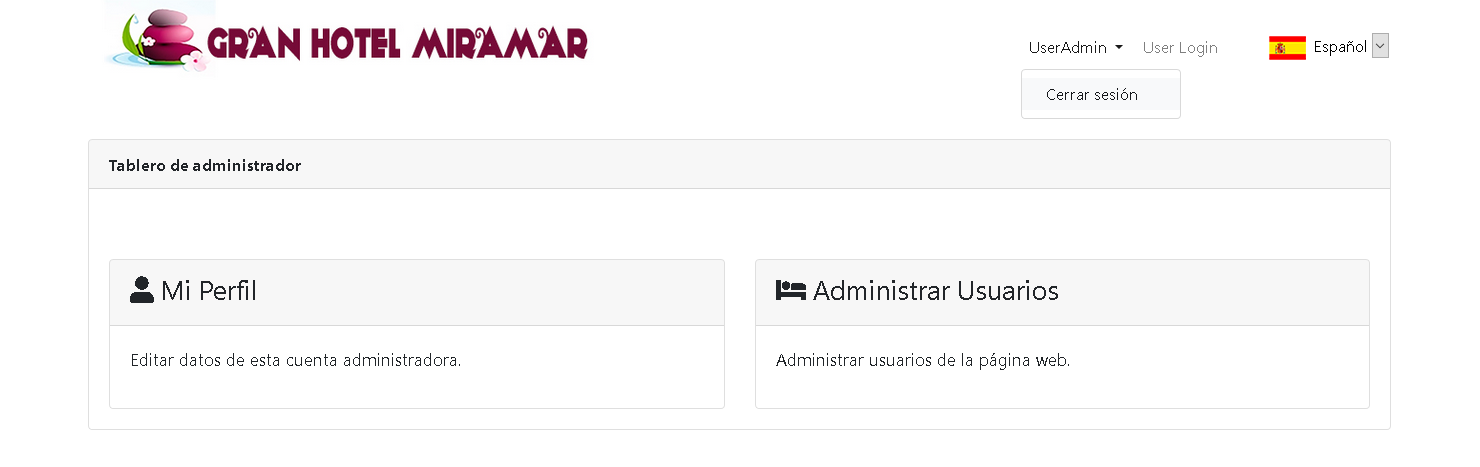
**-Mostrar usuarios (administrador)**

1. Clica en “Administrar Usuarios” para ver los usuarios de la aplicación y poder editar sus datos y darlos de baja.



**-Administrar usuarios (administrador)**

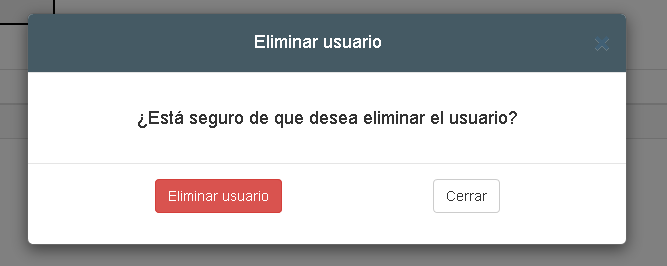
1. Clica en “Administrar Usuarios” para ver los usuarios de la aplicación.



2.1. Clica en “Eliminar usuario “para darlo de baja.



2.1.1 Confirma la eliminación de usuario.

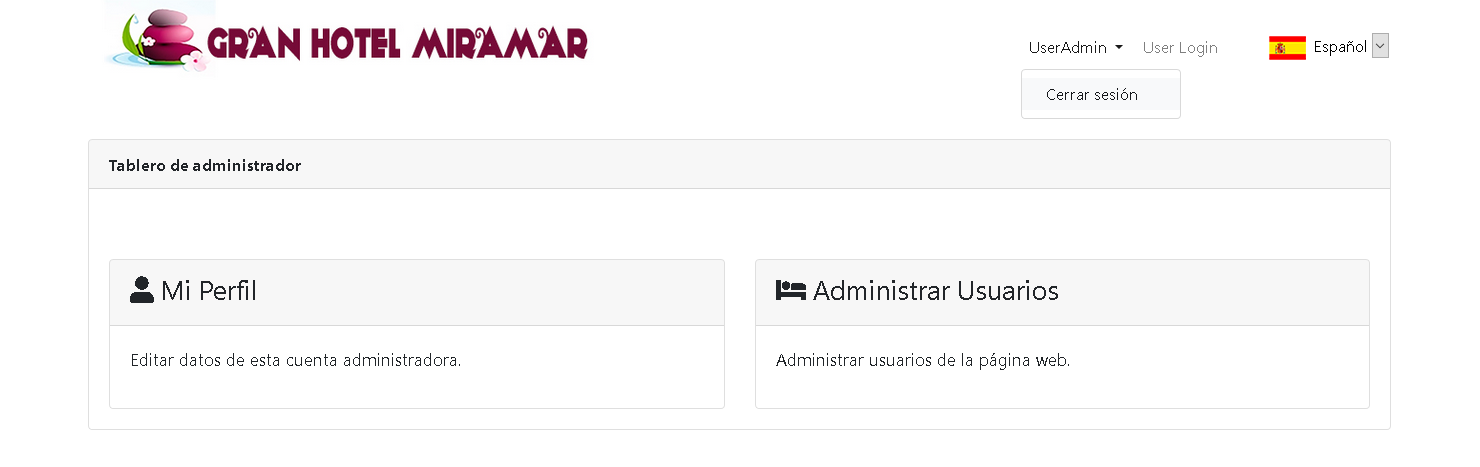


2.2. Edita los campos de los datos de usuario que desees modificar.

2.2.1. Clica en ¨Guardar para modificar los datos editados.

**-Eliminar usuario (administrador)**

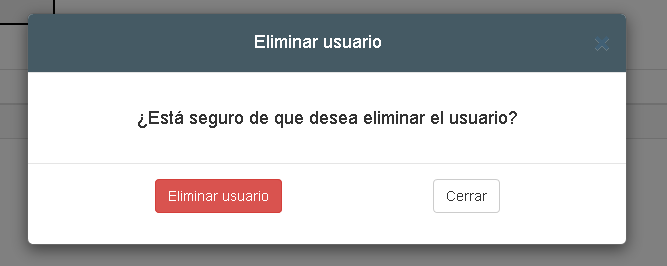
1. Clica en “Administrar Usuarios” para ver los usuarios de la aplicación.



2. Clica en “Eliminar usuario “para darlo de baja.

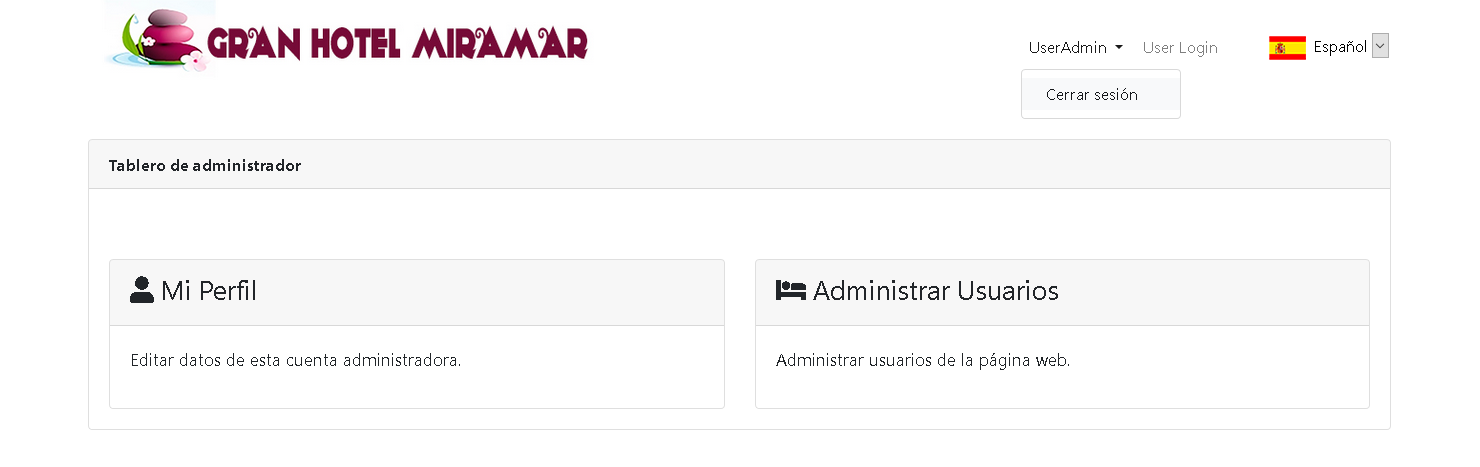


3. Confirma la eliminación de usuario.



**-Crear usuario (administrador)**

1. Clica en “Administrar Usuarios” para ver los usuarios de la aplicación.



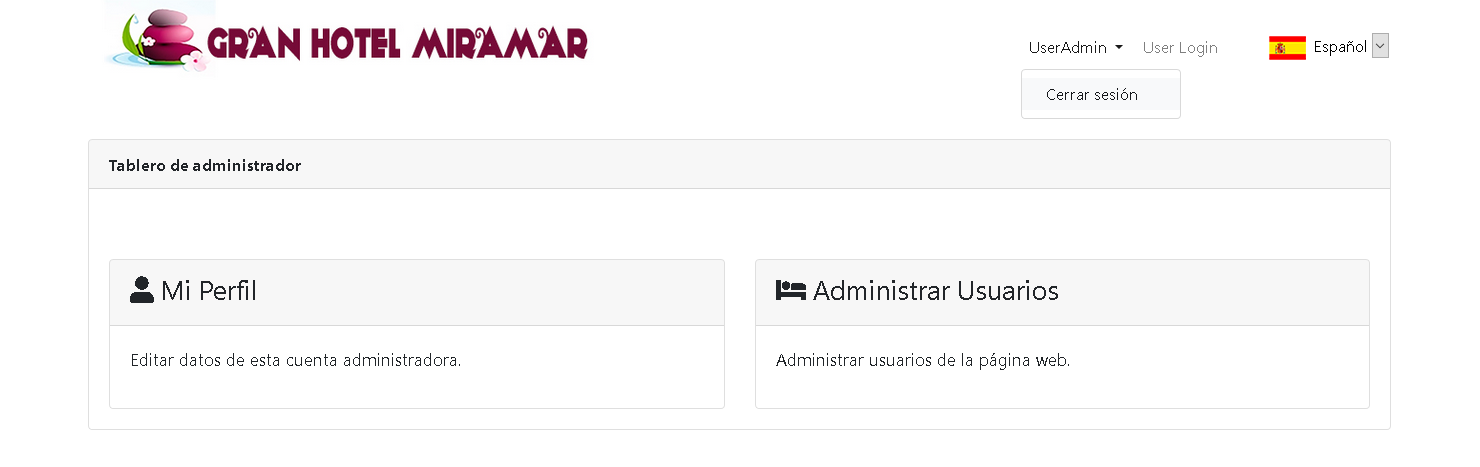
2. Clica en “Crear usuario “para darlo de alta.

3. Introduce sus datos.

4. Clica en “guardar” para crear el usuario.

**-Modificar usuario (administrador)**

1. Clica en “Administrar Usuarios” para ver los usuarios de la aplicación.



2. Edita los campos de los datos de usuario que desees modificar.

3. Clica en ¨Guardar para modificar los datos editados.

**-Cambiar idioma**

1. Clica en el selector de idiomas para ver los diferentes idiomas de la aplicación.



2. Selecciona el idioma que desees.

**-Reservar (habitación, restaurante, tratamiento, spa)**

1. Clica en el botón “Reservar” del servicio que desees reservar.



2. Introducimos los datos necesarios del formulario.



3. Clica en “Reservar” para finalizar la reserva.

## 5.3. Manual de instalación

Guía instalación Proyecto Laravel y GitHub

* Requisitos:

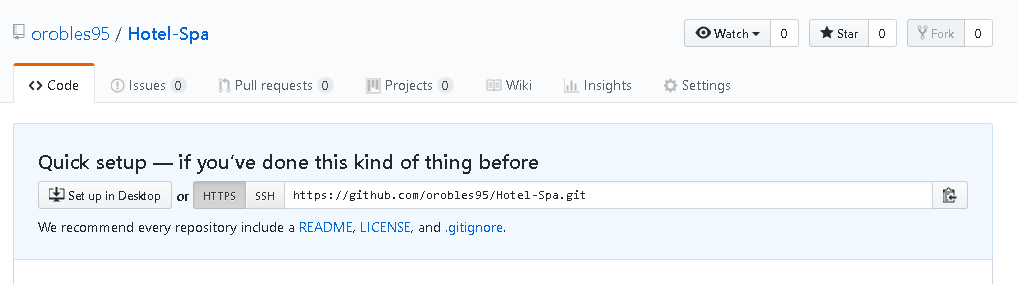
-Necesitaremos tener instalados los siguientes programas en nuestro equipo: Git, NetBeans Composer, Mycript y XAMPP.

* Pasos:

1. Abrimos símbolo de sistema o la consola de GitBash y creamos nuevo proyecto en “xampp/htcdocs”:



2. Creamos nuevo repositorio en GitHub (<https://github.com/orobles95/Hotel-Spa>) para trabajar con el proyecto.



3. Ahora iniciamos la consola de GitHub desde el proyecto para subirlo a nuestro repositorio GitHub. Empezamos ejecutando un “git init” preparar el directorio y crear las carpetas necesarias para compartir el proyecto.

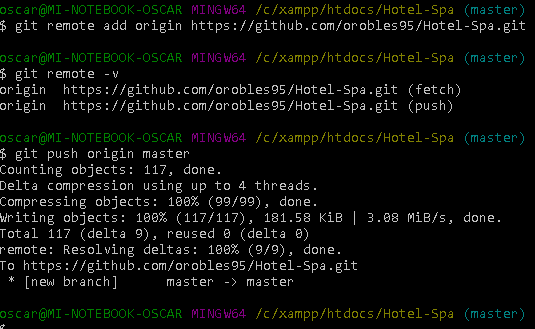
Seguidamente ejecutamos un “git add .” para añadir todos los directorios y archivos contenidos en la carpeta del proyecto:



A continuación, ejecutamos el primer *commit* para poder realizar un *push* de todos los archivos:

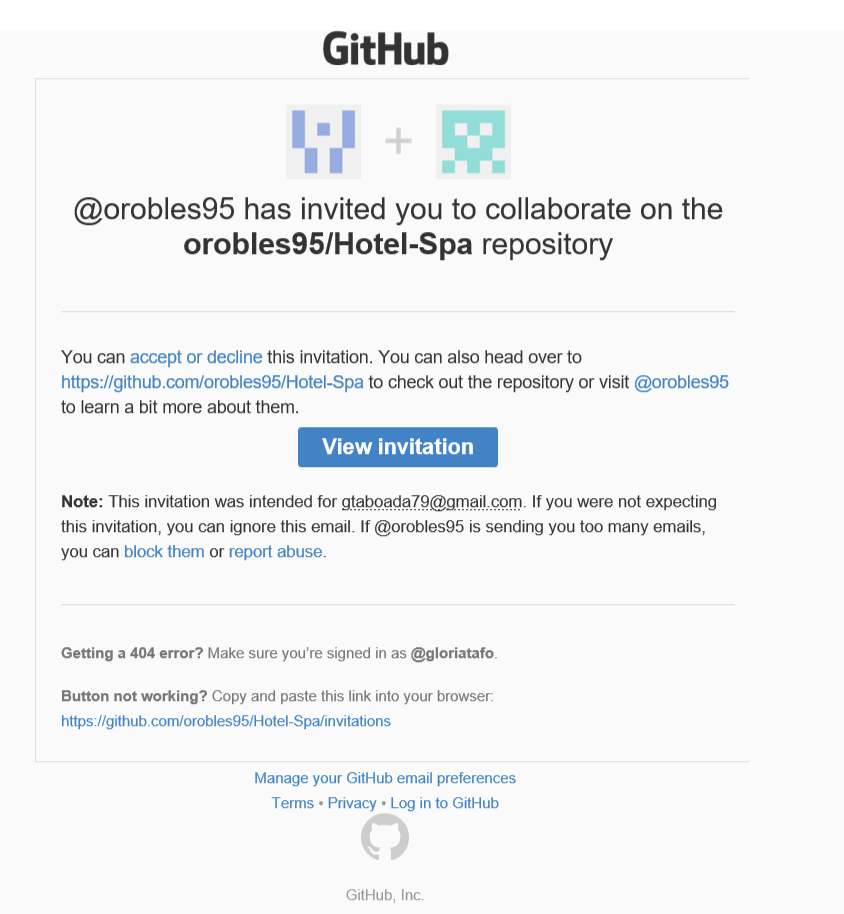


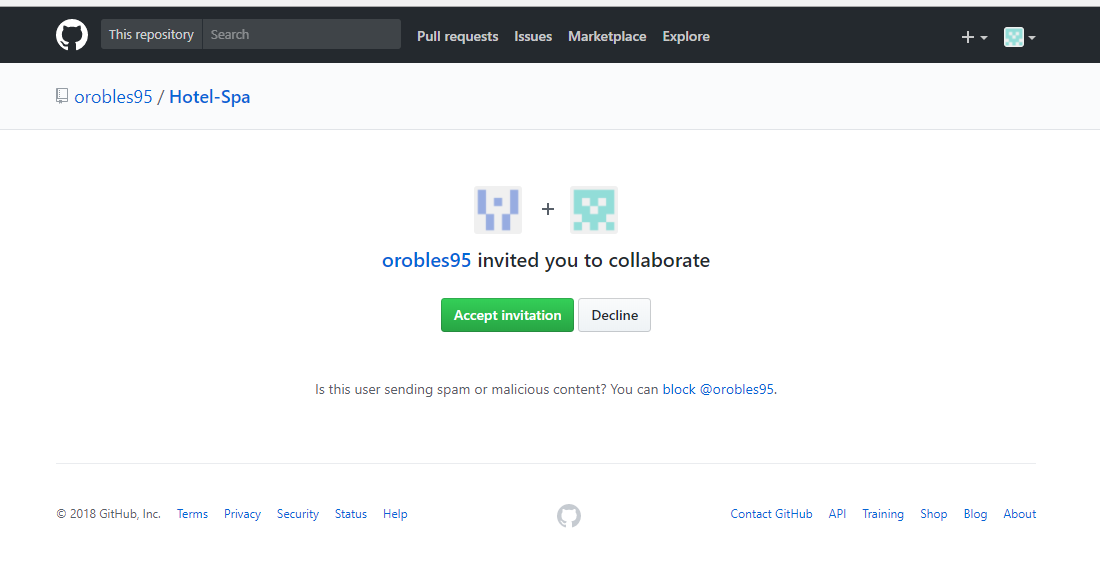
Finalmente le indicamos la dirección de nuestro repositorio GitHub y ejecutamos un *push* de la raíz de nuestro proyecto:

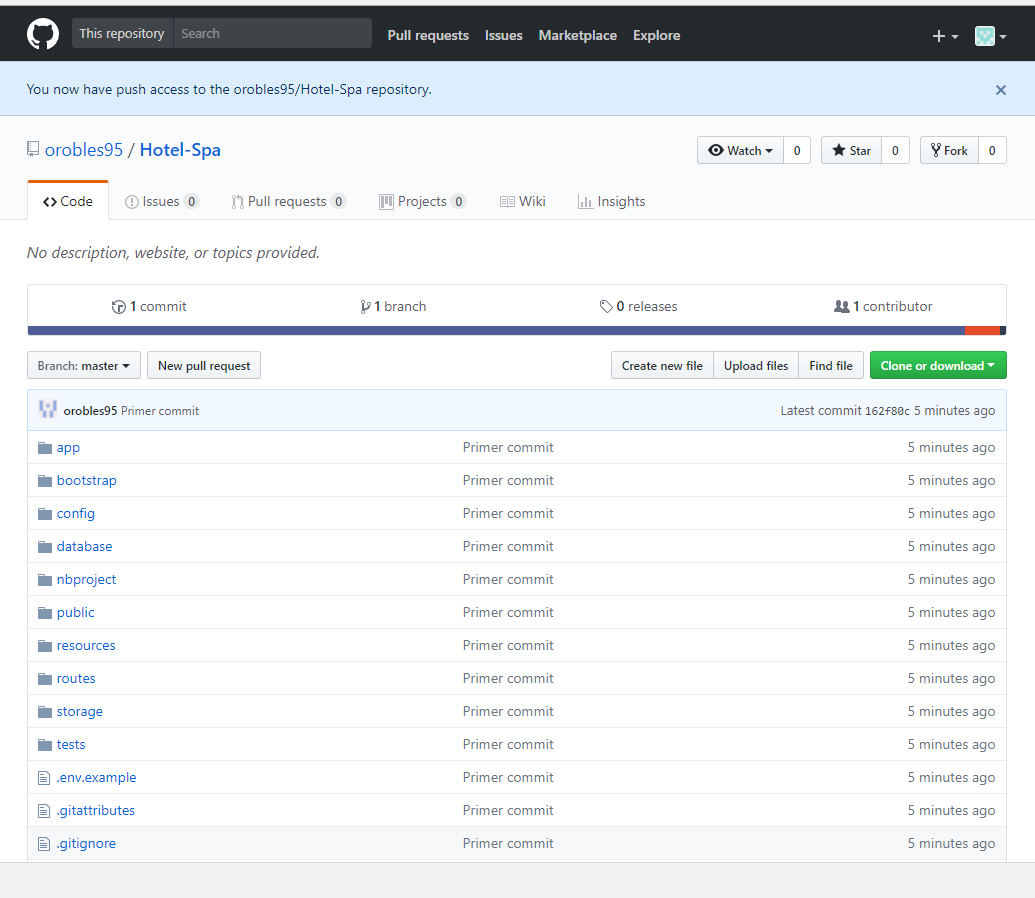


4. El compañero que no ha creado el proyecto debe seguir los siguientes pasos:

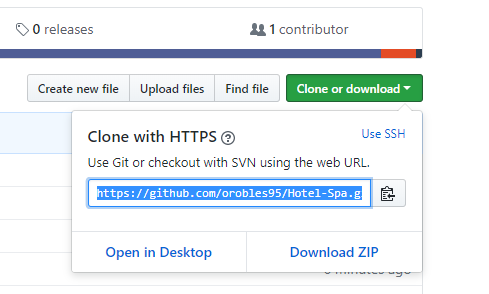
4.1. Unirse al repositorio del proyecto en GitHub (en este caso con invitación del creador del repositorio).



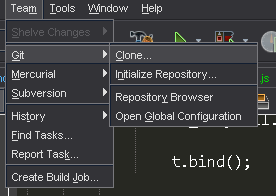


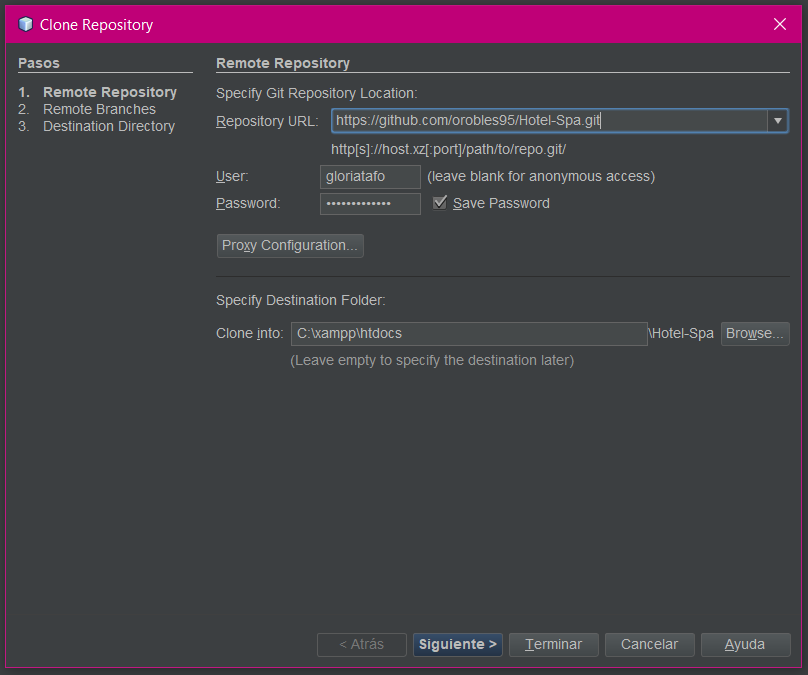


4.2. Después tendremos que clonar el proyecto. Existen varias formas de lograr esto, pero nosotros utilizaremos la dirección del repositorio:

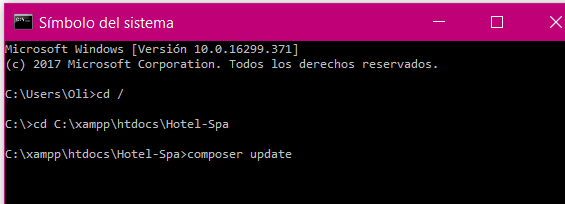


4.3. A continuación, abrimos NetBeans y accedemos a la siguiente opción:

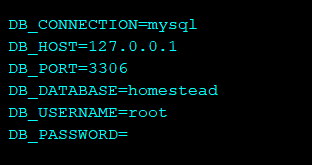




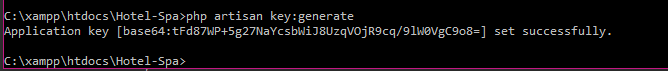
4.4. Ahora desde el terminal deberemos acceder a la carpeta donde clonemos nuestro proyecto para actualizar Composer:



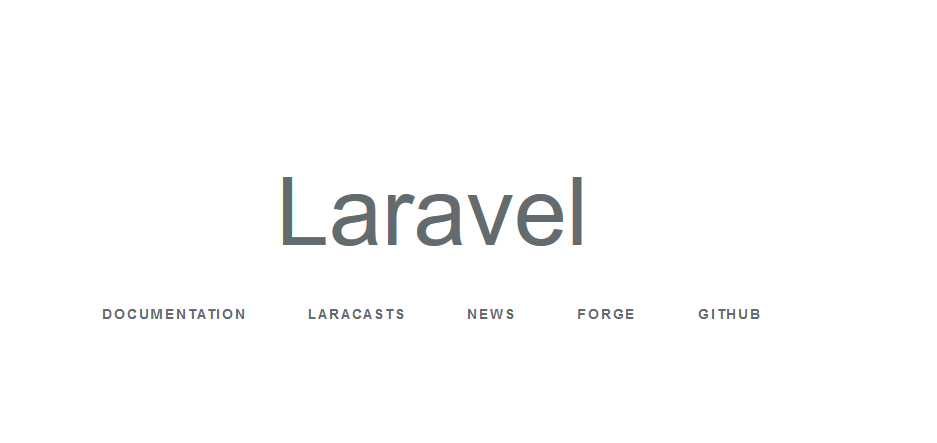
4.5. Lo siguiente que debemos hacer es duplicar el archivo “env.example” del proyecto y cambiarle el nombre como “env.” Y configurar los siguientes datos con nuestra configuración de nuestro servidor, nuestra base de datos y usuario y contraseña para acceder a ella.



4.6. Por último, deberemos acceder de nuevo al terminal a nuestro proyecto y crear una nueva clave para la aplicación:

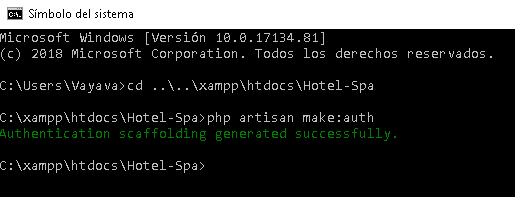


4.7. Ahora ya podremos acceder a nuestro proyecto (por ejemplo, la vista por defecto):

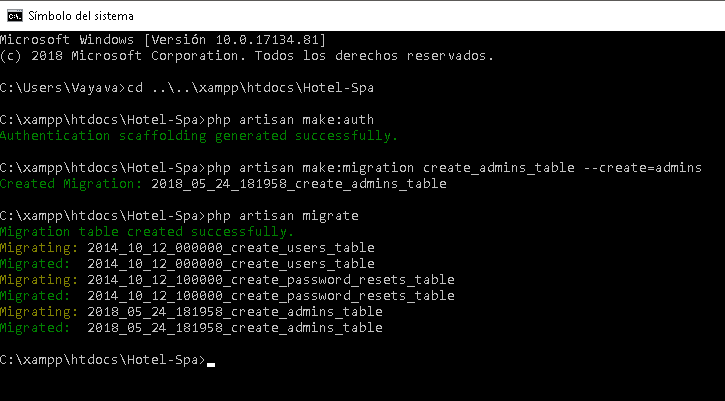


Guía instalación autentificación de usuarios

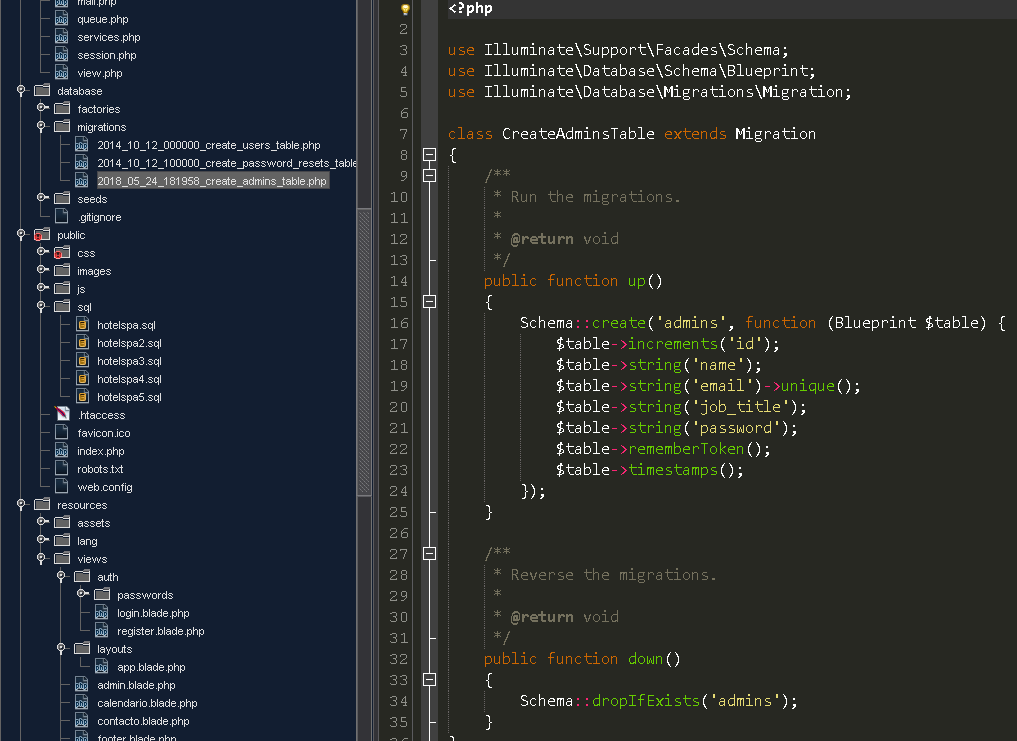
1. En terminal creamos los archivos necesarios para la autentificación en u proyecto Laravel:



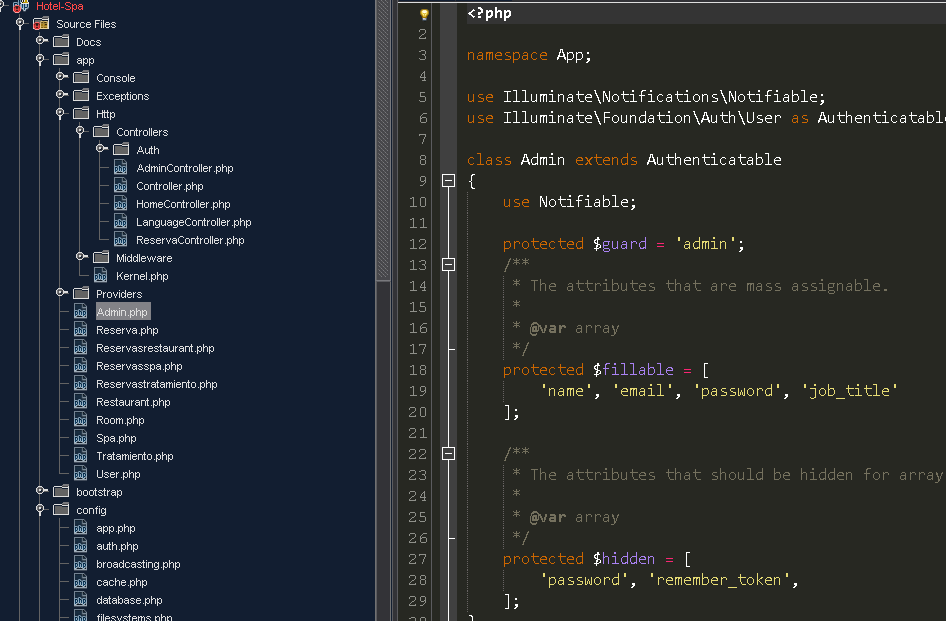
2. Añadimos la table admin a las migraciones y creamos las tablas por defecto de usuarios, reset de contraseña y la tabla admin creada:



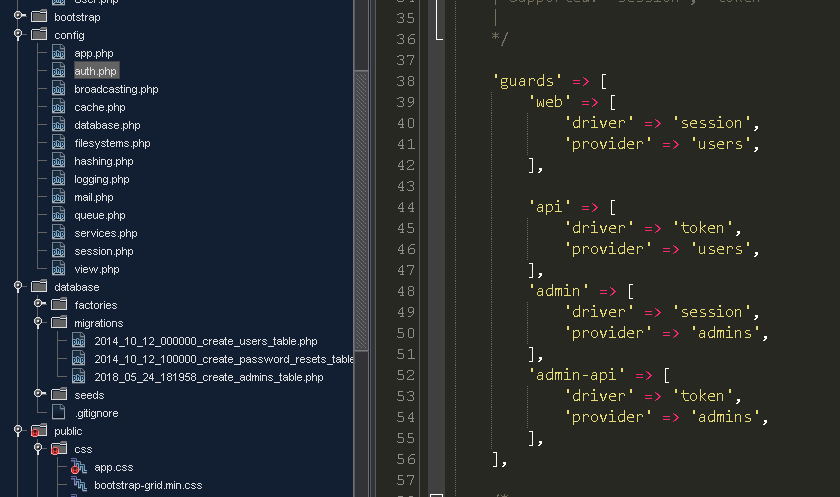
3. Editamos la migración de “admin” con los mismos datos que un usuario norma además de un campo “job\_title” destinado a la función de ese administrador:



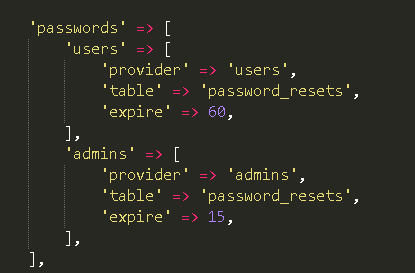
4. Accedemos al modelo de Admin para seleccionar la tabla “admin” al usar el modelo Admin y la editamos con los siguientes datos:



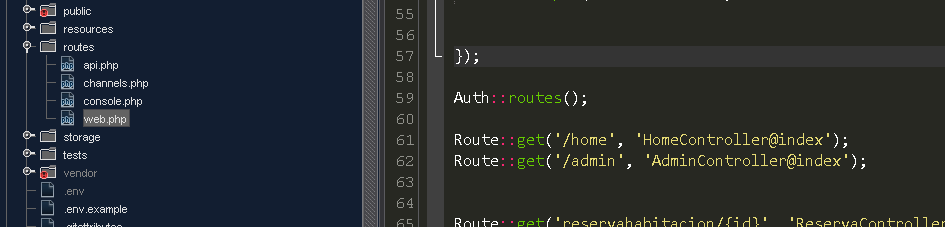
5. A continuación, accedemos a “config/auth.php” y editamos las siguientes secciones añadiendo la configuración necesarios para nuestro administrador:



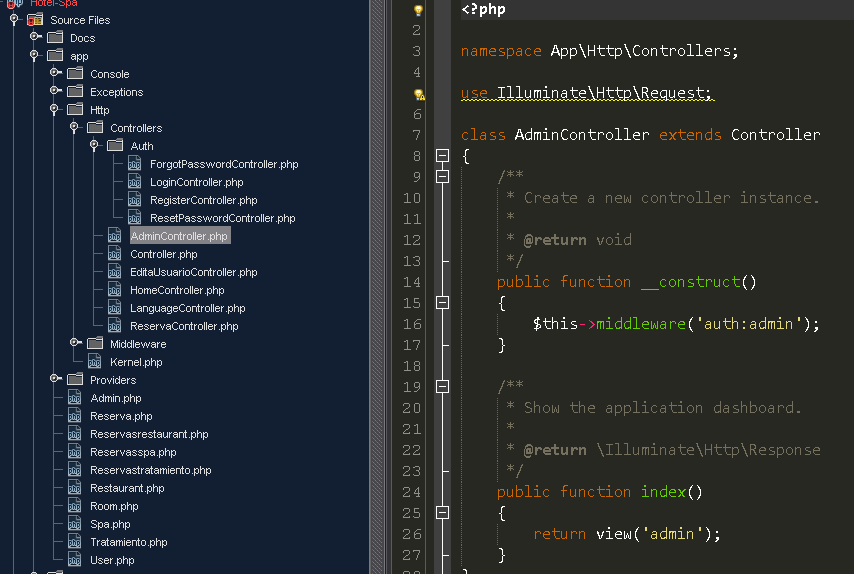




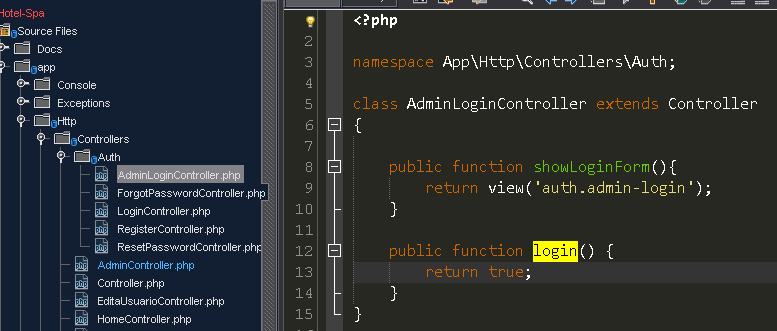
6. Ahora accedemos al archivo de rutas de Laravel (“routes/web.php”) para crear la protección al acceder a las vistas de usuario, admin y el resto de rutas de inicio de sesión y registro relacionadas con la autentificación de usuarios:



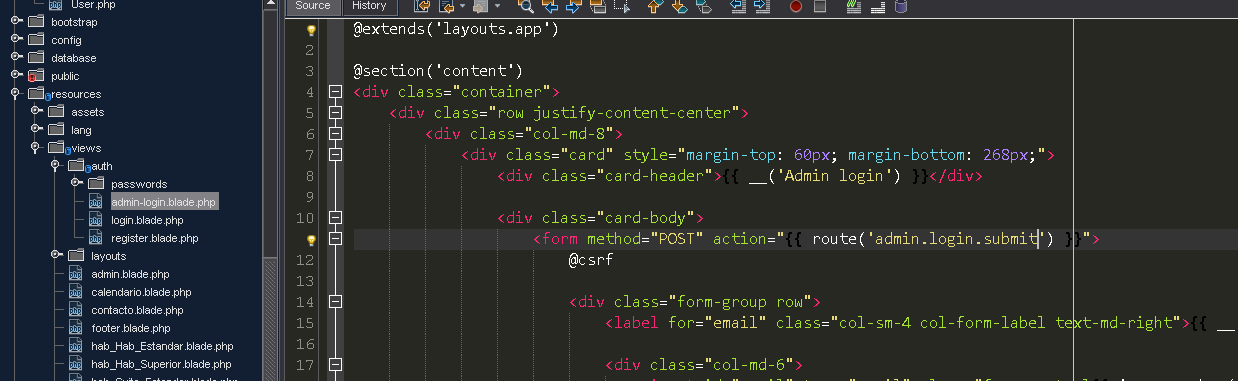
7. Creamos el controlador que usara la vista de administrador agregando en esta la especificación de que solo pueda acceder un usuario de la tabla “admin”:



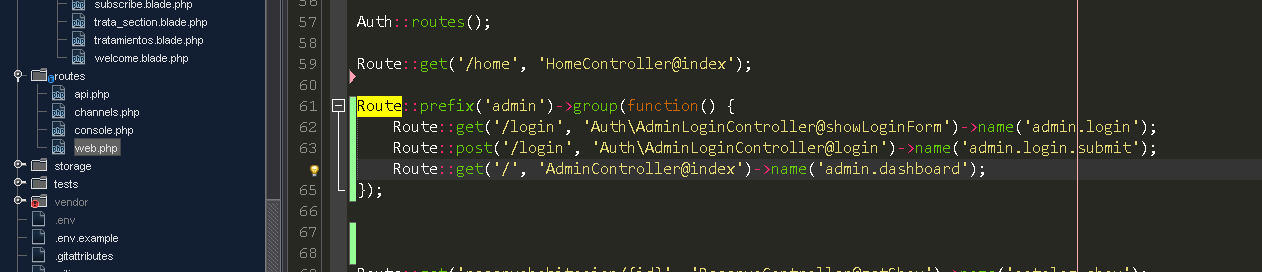
8. A continuación, creamos el controlador de login del administrador en “app/Http/Controllers/AdminLoginController.php”:



9. A continuación creamos la vista de login de administrador en “resources/views/auth/admin-login.blade.php” y editamos la ruta al enviar el formulario como se ve en la imagen:



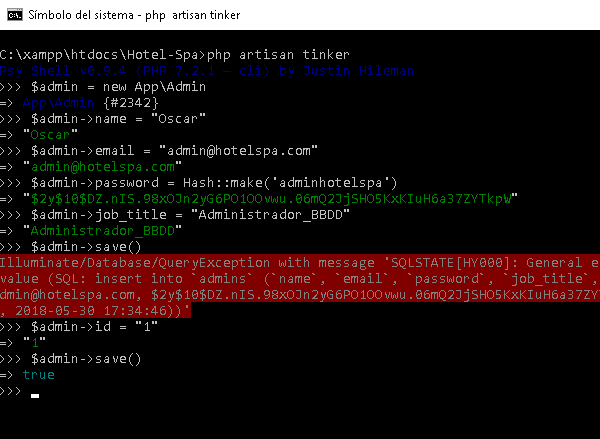
10. Ahora editaremos el fichero de rutas con las siguientes:



11. Terminando editamos el controlador de login de administrador “app/Http/controllers/AdminLoginController.php”:



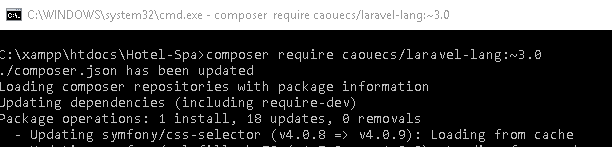
12. Por último, solo nos queda crear un administrador en nuestra tabla Admin desde el terminal para probar que el código funciona correctamente:



13. Finalmente, modificamos el acceso a el dashboard de administrador para que quede accesible si ya hemos iniciado sesión cuando accedamos a la vista de “login user” ubicada en “resources/views/layouts/app.php”:

Guía instalación proyecto multilingüe

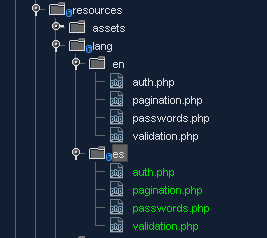
1. El primer paso es acceder a la carpeta raíz de nuestro proyecto y ejecutar el siguiente comando:



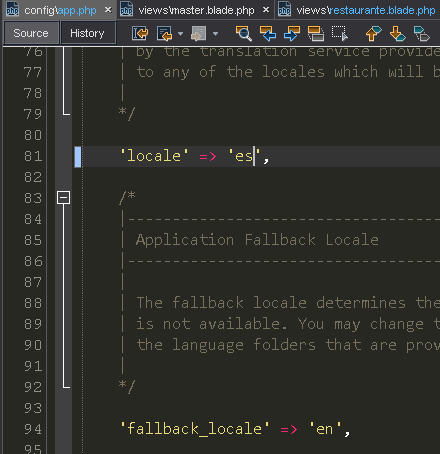
Esto nos creará las siguientes carpetas de varios idiomas en “vendor/caouecs/src” con los archivos necesarios predeterminados de autentificación y más configuraciones:



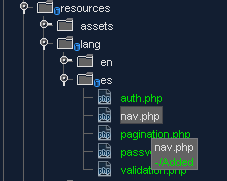
2. Ahora seleccionamos las que queramos implementar en nuestro proyecto y deberemos copiarlas en “resources/lang”:



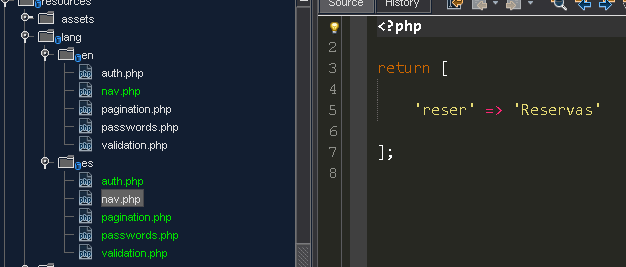
3. Ahora podemos configurar nuestro idioma por defecto en nuestro proyecto en “config/app.php” cambiando el valor de ‘locale’ y seguidamente también podemos configurar el idioma secundario en caso de que no encuentre nuestra librería de idioma principal configurando el valor de ‘fallback\_locale’:



4. Ahora creamos el archivo “nav.php” que utilizaremos en nuestra primera vista traducida tanto en la carpeta ‘en’ (vista ingles) como en la ‘es’ (vista español):



5. Añadimos en estos archivos la siguiente estructura json con sus referencias “name=value”:

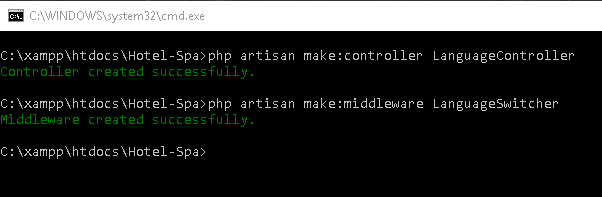




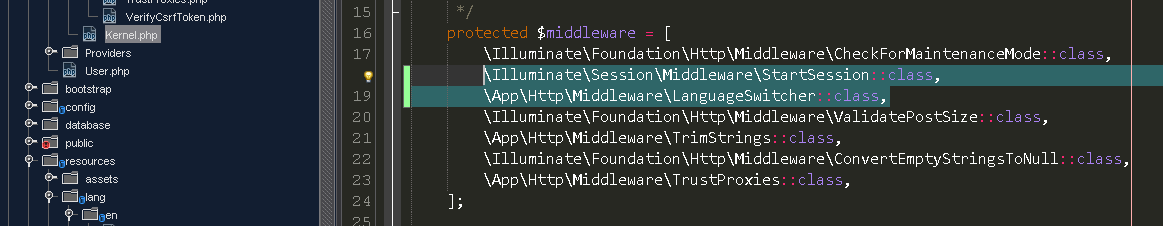
6. Añadimos en la vista la función php que utiliza nuestro archivo ‘nav’ y selecciona el valor de ‘reser’:



7. A continuación, crearemos los archivos necesarios de configuración para el sistema multilingüe. Para ello accedemos al terminal en nuestra carpeta raíz del proyecto y creamos los siguientes archivos middleware:

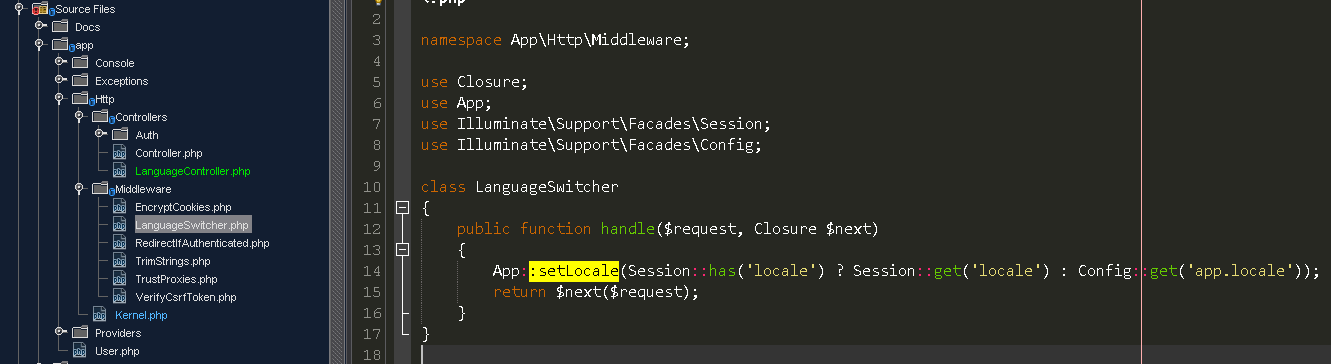


8. Ahora accedemos al archivo “app/Http/Kernel.php” y le añadimos las siguientes líneas:

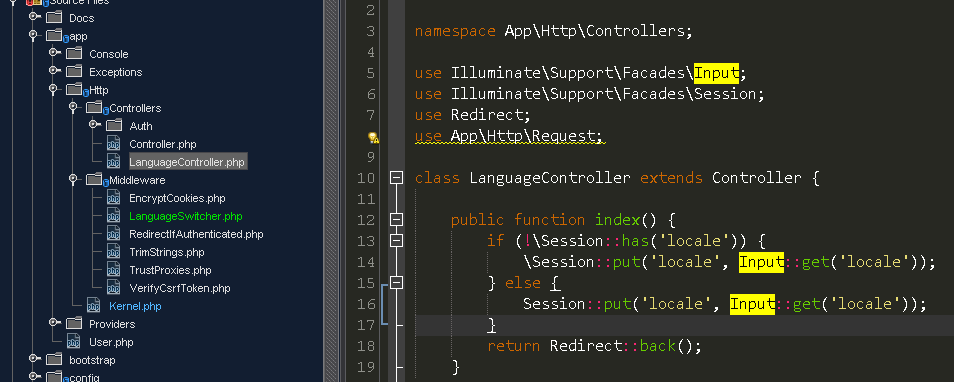


9. Seguidamente configuramos el middleware y el controlador cambiador de idiomas creado anteriormente:

Middleware LanguageSwitcher:



Controlador LanguageController:



10. Por último, accedemos a la vista donde ubicaremos el seleccionador de idioma con un “select” y este será el código que utilizaremos:

