Índice

[Planteamiento de la aplicación 2](#_Toc74793815)

[Mockups e ideas de diseño 3](#_Toc74793816)

[Página de inicio 3](#_Toc74793817)

[Página de contacto 3](#_Toc74793818)

[UML – Diagrama de clases java 4](#_Toc74793819)

[Tecnologías y programas usados en el proyecto 5](#_Toc74793820)

[La empresa, Autoescuelas Consur 6](#_Toc74793821)

[Modelo de negocio 6](#_Toc74793822)

[Inversión inicial 6](#_Toc74793823)

[Costes de despliegue y mantenimiento 6](#_Toc74793824)

[Funcionalidades que me gustaría haber implementado 7](#_Toc74793825)

[Funcionalidades parcialmente implementadas 7](#_Toc74793826)

# Planteamiento de la aplicación

Se pretende desarrollar la web de “autoescuelas consur” (empresa ficticia), la cual debe contar con una página inicial que sirva tanto para atraer a los potenciales clientes de la autoescuela como para proporcionarles acceso a información de contacto, ofertas disponibles, acceso a una sección donde comprar clases presenciales de conducir (solo para alumnos con el teórico aprobado) y un lugar donde iniciar sesión.

En cuanto a la parte funcional de la aplicación web, debe permitir que los usuarios que visiten la web inicien sesión (aunque no debe permitir que se registren ya que el registro será cosa de los administradores, teniendo en cuenta que solo nos interesa dar acceso a las personas que vayan físicamente a la autoescuela a darse de alta).

Queremos que se guarden los intentos que hace cada usuario en cada test (si falla/acierta) ya que en la sección de estadísticas podrán verlas de forma global o individuales para cada test, para así saber en que tests necesitan esforzarse más. Cuando un usuario 75% de tasa de éxito o más de forma general en los test que hace se le recomendará realizar el examen teórico de la DGT.

Otra de las características que debe tener la web es una sección donde publicar ofertas para los distintos tipos de carné, clases de conducir o incluso packs, esta sección debe estar controlada exclusivamente por los administradores y las ofertas deben listarse en algún apartado accesible desde la página principal de la autoescuela.

**Debe haber 3 grupos de usuarios en la aplicación:**

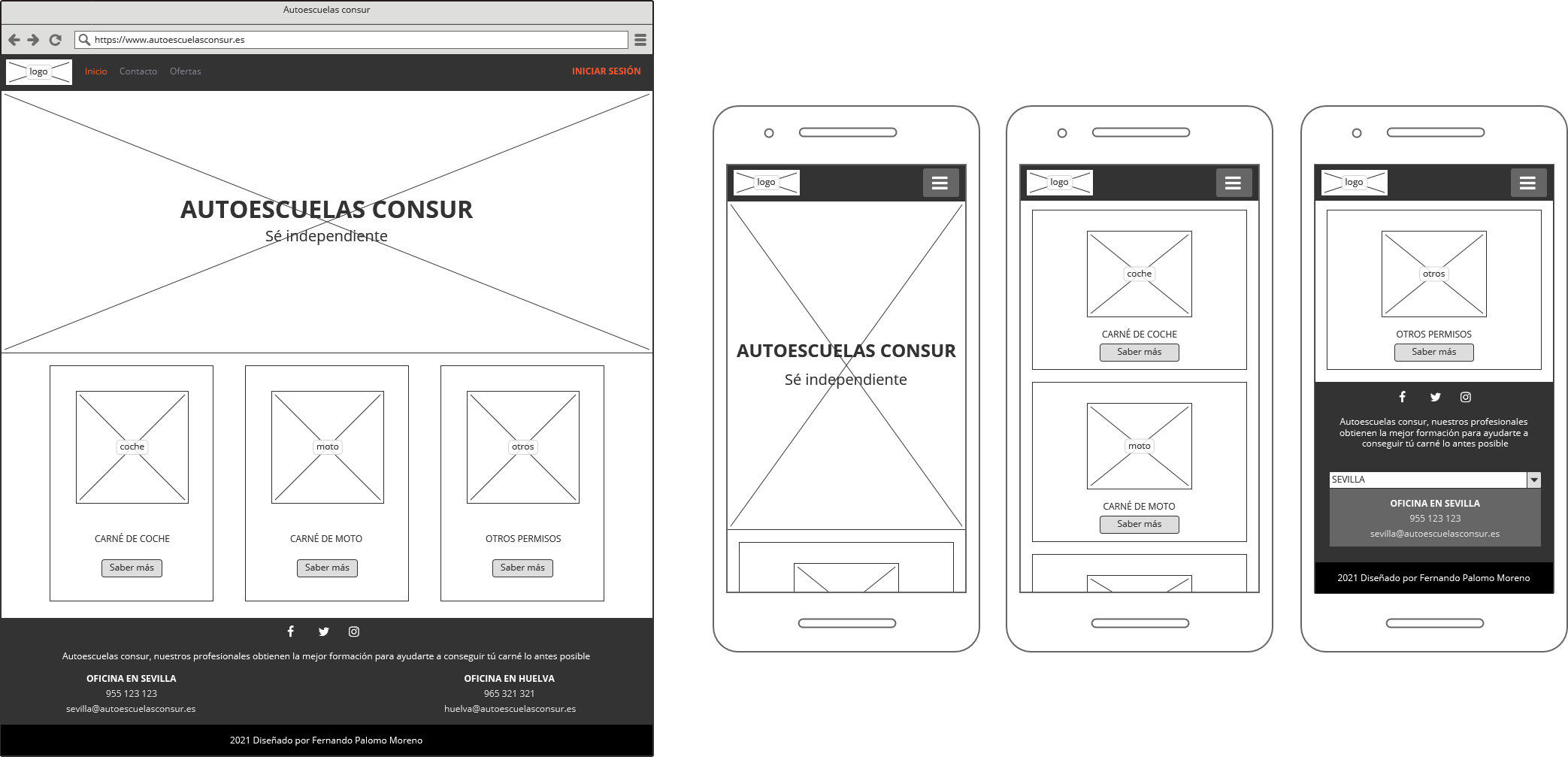
* **Usuarios:** estos son los alumnos, los cuales pueden iniciar sesión, realizar tests y ver estadísticas sobre sí mismos.
* **Profesores**: deben poder acceder a las estadísticas de todos los alumnos para saber cómo están evolucionando, también se les va a permitir añadir/editar/eliminar preguntas de test, pero no se les va a permitir registrar/dar de baja usuarios ni cambiar roles. Los profesores también deben poder seleccionar qué horas tienen libres para dar clases (las cuales deberán estar disponibles en un calendario web en el que los alumnos que hayan aprobado el examen teórico puedan comprar esas horas con la finalidad de asistir a una clase presencial práctica).
* **Administradores:** pueden realizar cualquier acción sobre la aplicación, incluido registrar/dar de baja tanto a alumnos como a profesores, además de cambiar su rol.

**Para mayor comodidad de los administradores y profesores cada uno tendrá su propio panel:**

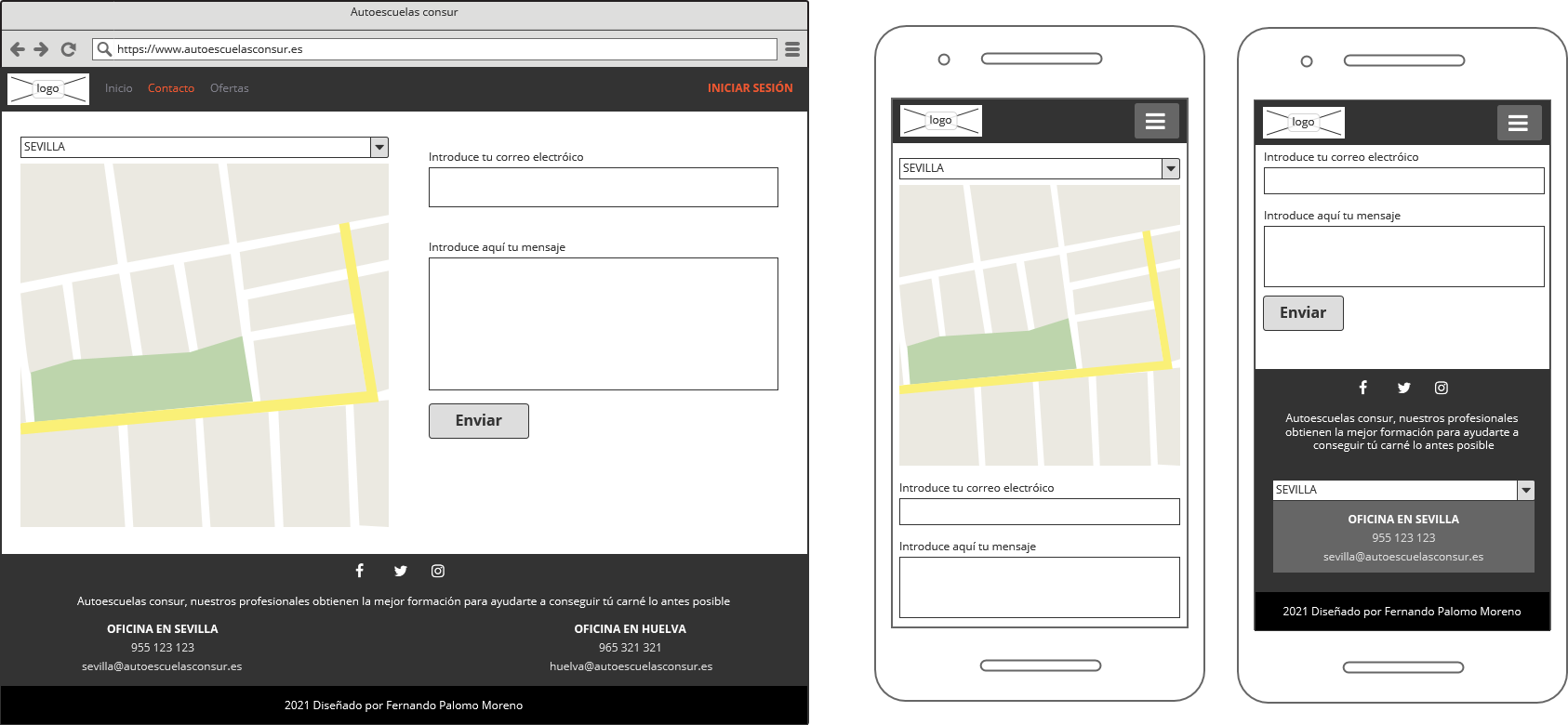
* **/admin ->** Administradores, aquí podrán ver y gestionar todo sobre la aplicación
* **/profesor ->** Informes, estadísticas, búsqueda de alumnos, selección de horas libres, etc

# Mockups e ideas de diseño

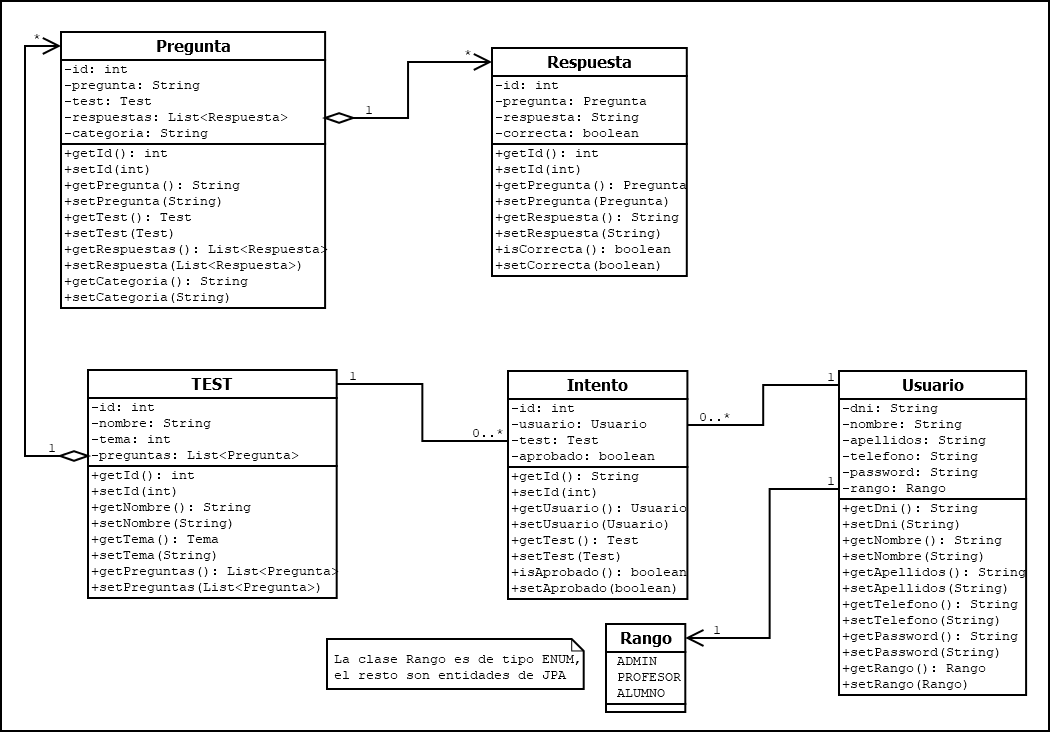
## Página de inicio



## Página de contacto



# UML – Diagrama de clases java



# Tecnologías y programas usados en el proyecto

* **Java 8:** Lo utilizaremos como lenguaje principal para realizar la aplicación web, se usará eclipselink JPA para crear las entidades de la base de datos, acceder a ellas y modificarlas.
* **HTML 5 + JSP:** Frontend (estructura y mostrar datos).
* **SCSS + bootstrap 5:** Frontend (posicionamiento de elementos, estilos en general).
* **JS:** Frontend (Efectos puntuales, gráficas, etc…).
* **MariaDB:** Necesitamos una base de datos así que nos hemos decantado por MariaDB.
* **IntelliJ IDEA:** La potencia y posibilidades de este IDE son increíbles y es con el que más familiarizado estoy, aunque podría haber usado netbeans o eclipse en su lugar sin problema.
* **Docker:** para el despliegue de la base de datos y el war de la aplicación web
* **GitHub Desktop:** para el control del repositorio git de una forma sencilla

La principal razón por las que he elegido usar estas tecnologías es por el manejo que tengo de ellas y la facilidad con la que las trabajo

# La empresa, Autoescuelas Consur

La aplicación será desarrollada para Autoescuelas Consur, una empresa ficticia creada por mí la cual actúa como cliente en este caso, su constitución social es la de una SL

## Modelo de negocio

El modelo de negocio de esta empresa se basa principalmente en atraer a clientes con unas ofertas llamativas en cuanto a precio para los distintos tipos de carné, siendo después su principal fuente de ingresos las clases prácticas que los alumnos deben comprar individualmente o por lotes para poder adquirir experiencia en el manejo del vehículo. Cada clase de 1h y 45 minutos tiene un precio fijo de 20€

## Inversión inicial

El capital inicial ha sido obtenido por medio de un préstamo bancario

## Costes de despliegue y mantenimiento

|  |  |
| --- | --- |
| **MOTIVO** | **COSTE** |
| Servidor para alojar la aplicación | 26.40€ |
| Dominio web | 10.00€ |
| Desarrollo web | 250.00€ |
| **Coste Inicial** | 286.40€ |
| **Coste anual (renovación del dominio + 12 meses de mantenimiento)** | 326.80€ |

# Funcionalidades que me gustaría haber implementado

* Estadísticas de usuarios
* Botones para editar desde el panel de administración (solo están disponibles para algunos objetos)
* Terminar de traducir la aplicación (~85%)
* Implementar el sistema que gestionaría las clases y donde los alumnos podrían comprarlas
* Hay algunos bugs menores que arreglar
* Fotos en las ofertas

# Funcionalidades parcialmente implementadas

* Tests, actualmente se pueden realizar y crear tests, pero no se envían a ningún sitio, aunque si se corrigen
* Botones de edición en los objetos del panel de admin

Funcionalidades completamente implementadas

* Inicio/cierre de sesión con verificación de DNI
* Filtros para rutas sensibles (/admin, /app/a) y tests
* Panel de administración
* Página de ofertas añadidas dinámicamente
* Creación y borrado de usuarios (solo pueden los administradores ya que es lo que más sentido tiene en este caso), los usuarios pueden filtrarse por tipo desde el panel
* Creación y borrado de temas y tests
* Creación y borrado de ofertas
* Posibilidad de añadir coches y posteriormente filtrarlos por su estado
* Vista simplificada de impresión para todos los objetos accesibles desde el panel de administración
* Paginación para cargar los resultados por lotes y así poder crear “páginas” que muestren una parte de los resultados
* Sistema de traducción completamente funcional y expandible
* Estadísticas generales para administradores en la portada del panel de administrador
* Diseño full responsive, se ha pensado primero en hacerlo para dispositivos móviles y luego adaptarlo a resoluciones grandes
* Web optimizada pensando siempre en realizar el mínimo numero de queries y ralentizar la carga lo mínimo posible