

# Memoria Ingenieros S.A

ARQUITECTURA Y DISEÑO DE SISTEMAS WEB Y C/S



INGENIEROS S.A.S.

## Índice

Análisis del Problema: .....	2
Problemas encontrados y soluciones aportadas:.....	2
Implementación y diagrama de clases:.....	3
Bibliografía.....	4

## **Análisis del Problema:**

La situación que se nos plantea es la de una empresa cuyo objetivo es el de desarrollar un sistema informático que la permita gestionar a su personal, dicha aplicación diferenciará dos perfiles de forma clara, el primer perfil sería el de RRHH el cuál debe administrar la información de las empresas y cada uno de los diferentes proyectos que lleva a cabo, quienes son los empleados de dicho proyecto y los calendarios de estos empleados, este perfil será capaz de añadir la información de las empresas clientes nuevas, de añadir proyectos a estas o a las ya existentes, y administrar los empleados que trabajaran para estas empresas y sus proyectos.

Además, este perfil deberá ser capaz de aprobar los días libres pedidos por el empleado

Por otro lado, el perfil del empleado deberá ser capaz de hacer su fichaje en los proyectos para saber las horas invertidas en cada uno y llevar un registro de estas.

También se nos pide controlar los posibles errores en la aplicación, derivándolos a una página diseñada por nosotros para poder administrarlos.

Por último debemos usar para su desarrollo HTML5, CSS3, JavaScript y MySQL, siguiendo el patrón Modelo Vista Controlador.

## **Problemas encontrados y soluciones aportadas:**

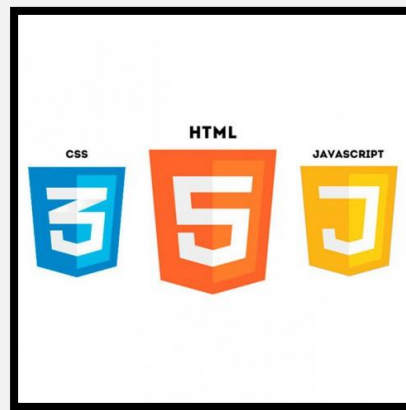
El primer problema que nos encontramos fue con la falta de familiarización práctica para afrontar el desarrollo de esta aplicación, aunque se vio altamente paliada debido a la teoría colgada en la página del curso, ya que gracias a comprender cada tipo de componente y su funcionamiento, conseguimos desarrollarla de una forma óptima.

Otro de los problemas encontrados, sobre todo al principio del desarrollo, fue el de tener que visualizar cada cambio en el front teniendo que cargar el proyecto en la máquina virtual y hacer todo el proceso de subirlo al servidor constantemente, así la solución que acabamos por adoptar fue la de instalar el propio apache-tomcat en Windows, y así ser más eficientes a la hora de trabajar.

Por otro lado, también ha habido contratiempos con la base de datos, que a pesar de ser pequeños detalles, nos han robado tiempo de desarrollo, por lo que tuvimos que trabajar arduamente para conseguir el objetivo que perseguíamos.

## Implementación y diagrama de clases:

Para el diseño de la parte del front-end, además de ayudarnos de HTML5, hemos estilizado los componentes web gracias a CSS, para ello hemos creado un fichero Styles, del que tomar su estilo el resto de componentes HTML, consiguiendo así una interfaz de usuario óptima y preocupándonos por su accesibilidad y compresión. Así planteamos la estructura mostrada en la aplicación, donde queda expuesta de forma clara la jerarquía entre empresa, proyecto y empleado.



Para la implementación hemos utilizado JSP ya que nos permite generar contenido dinámico debido a la inserción de código en lenguaje java, además nos permite el acceso a la base de datos remota, útil para las operaciones CRUD.

Para este proyecto hemos implementado un total de 13 JSP, que son: listUser.jsp, listProject.jsp, listCompany.jsp, listPetition.jsp, listCalendar.jsp, index.jsp, chooseTimeSearch.jsp, assignedProject.jsp, addUser.jsp, addProject.jsp, addPetition.jsp, addCompany.jsp y workHours.jsp.

Además como es lógico y derivado del uso de los JSP, se usan Servlets en esta implementación que se encargan de recibir y responder a las peticiones de un cliente a través de HTTP. De este modo tenemos los servlets; CalendarioController, CalendarioControllerAdmin, CompanyController, LoginServlet, PetitionController, ProjectController, ServletLogin, UserController.

Por último hemos utilizado las clases java para poder declarar el modelo (estructura del calendario, compañía, peticiones, proyectos, usuario) y el controlador de la aplicación. Así tenemos las clases de acceso a la base de datos, estas son: CalendarioDao, CompanyDao, DbUtil, ErrorHandler, Log, PeticionDao, PorjectDao, UserDao

Para un mejor entendimiento de esta implementación adjuntamos el diagrama de clases del programa: (aumentando el zoom se leen claramente todas las clases)

[https://drive.google.com/file/d/1ozMsMSqrPCU\\_-C17c-sRRvOtxz4bfjkF/view?usp=sharing](https://drive.google.com/file/d/1ozMsMSqrPCU_-C17c-sRRvOtxz4bfjkF/view?usp=sharing)

## Bibliografía

Documentación aportada en la BlackBoard por el profesor.

<https://www.w3schools.com/>

<https://dev.mysql.com/doc/>

<https://stackoverflow.com/documentation>

<https://www.youtube.com/watch?v=ugkTau6cWrg>

<https://www.javatpoint.com/jsp-tutorial>