



# WIS Knowledge report

C2.034

<https://github.com/juanlo017/c1.034>

Apellidos, Nombre	Email
Castro Vázquez, Jose Ignacio	joscasvaz@alum.us.es
Junco Obregón, Juan del	juajunobr@alum.us.es

# **Tabla de contenidos**

<b>Resumen ejecutivo</b>	<b>3</b>
<b>Tabla de revisiones</b>	<b>4</b>
<b>Introducción</b>	<b>5</b>
<b>Contenidos</b>	<b>6</b>
<b>Conclusiones</b>	<b>7</b>
<b>Bibliografía</b>	<b>8</b>

## **Resumen ejecutivo**

Se explicarán los conocimientos anteriormente adquiridos a DP2 a lo largo de nuestra formación en referencia a la arquitectura WIS.

## Tabla de revisiones

Versión	Descripción	Fecha
v1.0	Creación del documento, recapitulación de información de la arquitectura WIS.	14/02/2024
v1.1	Primera versión del documento.	15/02/2024

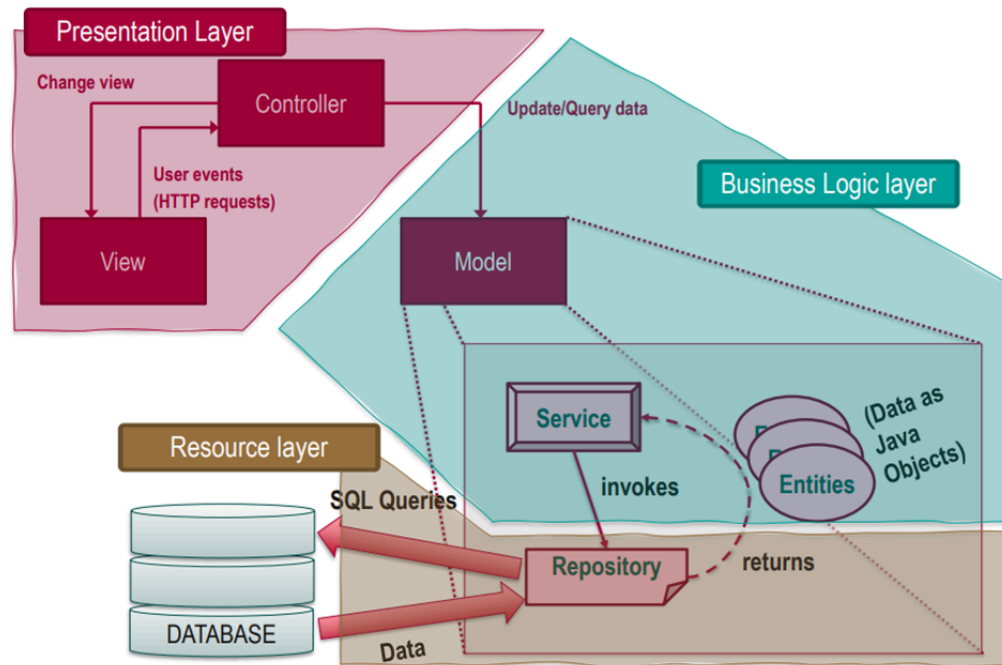
## **Introducción**

En este documento se hará un estudio de los conceptos aprendidos sobre la arquitectura WIS en el grado de Ingeniería del Software, y más concretamente en la asignatura de la asignatura de Diseño y Pruebas I, asignatura predecesora a esta.

Para ello se ha preguntado a los distintos miembros del grupo sobre la información que recordaban de las WIS, posteriormente contrastada con las transparencias de clase del cuatrimestre pasado.

Para la elaboración de este documento, se ha optado por un enfoque realizado comentado de manera más genérica al comienzo, para terminar hablando de manera más avanzada en el tema.

# Contenidos



Entendemos como arquitectura de un Web Information System (WIS) al “esqueleto” que nos muestra las interacciones entre componentes de la aplicación, interfaces de usuario y bases de datos.

Aunque hay muchas arquitecturas posibles en un WIS, nos centraremos en la arquitectura MVC. En este estilo de arquitectura, podemos diferenciar tres capas: la capa de web (controladores y vistas), la capa de dominio (servicios y entidades) y la capa de recursos (repositorio y base de datos).

Las siglas MVC derivan de los principales componentes que interactúan entre sí formando el estilo de la arquitectura. Tenemos un **modelo** que conforma todos los datos, los cuales se muestran en **vistas** que se renderizan cada vez que el servidor recibe una petición HTTP que es gestionada por los **controladores**.

La forma en que trabajamos con un WIS con arquitectura MVC es la siguiente:

1. Los usuarios envían peticiones HTTP a través de la aplicación mediante el navegador al servidor.
2. Estas peticiones son recogidas por los controladores, quienes las gestionan y llaman a los servicios correspondientes.
3. Los servicios, que pertenecen a la capa de dominio, reciben, modifican y proporcionan los datos a los controladores, y también tienen la habilidad de transferir datos a la capa de recursos para su persistencia.
4. Finalmente, la capa de recursos se encarga de la conversión de datos a clases Java y viceversa para la correcta manipulación y/o persistencia de estos, utilizando tecnología JPA o Hibernate en nuestro caso.

## **Conclusiones**

Gracias al estudio realizado en este documento, se ha llevado de manera autodidáctica un análisis sobre la arquitectura de un WIS. Al poner en común los conocimientos básicos que recordábamos del anterior cuatrimestre y al refrescarlos con las transparencias de Diseño y Pruebas I nos encontramos en una mejor situación para entender el contenido de esta asignatura.

## **Bibliografía**

Transparencias de Diseño y Pruebas I.