

# Noviembre D01 – Diseño y Pruebas II

11-11-2022

## Grupo D07

CERRATO SÁNCHEZ, LUIS (luicersan@alum.us.es)

GUITÉRREZ CONTRERAS, ERNESTO (erngutcon@alum.us.es)

## Tutor

PATRICIA JIMÉNEZ AGUIRRE

<https://github.com/erngutcon/Acme-Courses-D0>

## Índice

RESUMEN EJECUTIVO	2
TABLA DE REVISIÓN	3
INTRODUCCIÓN	4
CONTENIDOS	5
CONCLUSIONES	7
BIBLIOGRAFÍA	8

## RESUMEN EJECUTIVO

Este documento es el informe de Sistemas de Información Web de nuestro proyecto para la asignatura de Diseño y Pruebas II.

Es necesario tener una visión general de las aplicaciones, de los lenguajes que se utilizarán o se han utilizado a la hora de realizar cambios, a la hora de implementar nuevo código y de hacer las pruebas necesarias porque de esta forma se podrán repartir mejor las tareas y el trabajo en función de las aptitudes de cada uno de los miembros del grupo.

# Tabla de revisión

V1.0	11-11-2022	Añadida la portada, el índice, el resumen ejecutivo, la tabla de revisión, la introducción, el contenido, las conclusiones y la bibliografía.
------	------------	---

## INTRODUCCIÓN

Los sistemas de información web pueden integrar procesos o sistemas dentro de una única interfaz y permitir el acceso a través de una intranet local o una red de Internet global.

Los sistemas de información web son sistemas que las distintas organizaciones y sus clientes utilizan para realizar entre muchas otras por ejemplo el comercio electrónico.

## CONTENIDOS

Un sistema de información web es un sistema de información que utiliza tecnologías web de Internet para ofrecer información y servicios a usuarios u otros sistemas o aplicaciones de información. Su objetivo principal es publicar y mantener datos utilizando principios basados en hipertexto y por lo tanto podríamos decir que es un "esqueleto" o diseño que muestra las interacciones entre los componentes de la aplicación, los sistemas de middleware, las interfaces de usuario y las bases de datos.

Las aplicaciones de bases de datos basadas en web tienen tres componentes principales:

- un navegador web (o cliente),
- un servidor de aplicaciones web y
- un servidor de base de datos.

Las aplicaciones de bases de datos se basan en un servidor de bases de datos, que proporciona los datos para la aplicación. El servidor de la base de datos ofrece lógica comercial e información o datos relevantes que el servidor almacena y administra, recupera y proporciona la información.

Durante los cursos en la Universidad de Sevilla nos hemos enriquecido de conocimiento sobre por ejemplo cómo utilizar y aplicar un diseño arquitectónico de una aplicación Spring Boot con Spring MVC que es el patrón modelo-vista-controlador y se trata de un patrón arquitectónico que estructura una aplicación en 3 componentes como el modelo que es la representación de la información, incluye los datos y la lógica de negocio necesaria para trabajar con ellos. La lógica de negocio no se debe colocar fuera del modelo, la vista que es la representación del modelo de manera que el usuario puede interactuar con él y suele hacerse mediante una interfaz de usuario y el controlador que responde a eventos en la interfaz de usuario (solicitudes HTTP en nuestro caso), invoca cambios en el modelo y probablemente en la vista.

Hemos aprendido a utilizar los controladores que son las clases/funciones que maneja todas las solicitudes de un sitio web y luego las envía al controlador apropiado. En Java, los controladores se crean como servlets y con la ayuda de Spring nos proporciona una implementación llamada DispatcherServlets. Los controladores responden a eventos en la interfaz de usuario (HTTP request en nuestro caso), invoca cambios en el modelo y probablemente en la vista. Spring da una manera de mapear las HTTP request como métodos con @GetMapping o @PostMapping Los controladores son responsables de transformar entidades en un ModelMap que puede ser procesado por las vistas, validación de la entrada proporcionada por la vista, transformar esta entrada en entidades si es necesario, llamar a servicios para realizar una acción y devolver el nombre de la vista que se debe cargar. Para conectar el controlador con el modelo solo necesitamos inyectar la dependencia de los servicios en el controlador.

Como ya sabemos la arquitectura de información web admite la lógica comercial y las aplicaciones de múltiples capas, desarrolladas con Python, PHP, Java, .NET, Ruby y Node.js.

También hemos aprendido a utilizar las vistas que representan la información dada por el controlador. Spring MVC no te obliga a usar una tecnología de vista específica como JSP, lo que permite la integración de frameworks SPA.

La arquitectura del sistema de información web PHP se basa en un lenguaje de desarrollo menos complejo y altamente funcional por tanto podríamos decir que el lenguaje PHP es uno de los más populares entre la comunidad. Su arquitectura permite una seguridad robusta, un

desarrollo rápido, un marco dedicado, mantenimiento simple y soporte extendido de una comunidad de desarrolladores.

Nos hemos confrontado con los patrones de la capa lógica de negocios como transaction script donde se organiza la lógica empresarial por procedimientos, donde cada procedimiento maneja una sola solicitud de la presentación, domain model que es un modelo de objeto del dominio que incorpora ambos comportamientos y datos y service layer que define el límite de una aplicación con una capa de servicios que establece un conjunto de operaciones disponibles y coordina la respuesta de la aplicación en cada operación, la capa de presentación interactúa con el dominio a través de la capa de servicio y la capa de servicios divide la lógica empresarial en dos tipos diferentes como la lógica de dominio y la lógica de aplicación.

Los frameworks de Python facilitan el desarrollo de aplicaciones web con código breve y conciso, fácil de leer y mantener. Esta es una de las razones por las que a los desarrolladores les gusta Python para usarlo como un lenguaje de secuencias de comandos del lado del servidor. Como lenguaje no solo es dinámico, sino que no requiere una codificación larga, lo que generalmente reduce la longitud de la codificación hasta en 1/5 de otros lenguajes de programación y se podría decir que quizás la mayor ventaja de Python es que el lenguaje se considera el más adecuado para la creación de prototipos.

La arquitectura del sistema de información web Java se basa en que es un lenguaje de programación versátil y es popular en el entorno de desarrollo empresarial. Independientemente de la complejidad o la naturaleza de la aplicación, la arquitectura de aplicaciones web de Java es la plataforma preferida por los desarrolladores para crear soluciones y cumplir con las expectativas. Una de las ventajas distintivas de esta arquitectura es la capacidad de combinar y confiar en las herramientas nativas de Java y los marcos para crear aplicaciones. También hemos aprendido a utilizar la vinculación de parámetros HTTP y objetos Java y cómo podemos completar un objeto java usando los valores de un formulario HTML, que es un caso de uso muy común, necesitamos ir mapeándolos manualmente uno por uno y por ejemplo en Sprint se usa el concepto de binding, que vincula automáticamente un conjunto de parámetros HTTP a los atributos del objeto Java.

Para nombrar algunos ejemplos de sistemas de información que son utilizados por las distintas empresas como los siguientes: sistema de soporte de decisiones, sistema de control de procesos, sistema de Información de Gestión, almacenes de datos, planificación de recursos empresariales, etc.

## CONCLUSIONES

Hemos e intentamos cada día esforzarnos por avanzar en las distintas tareas como el modelado de usuarios, la recuperación de información, el procesamiento del lenguaje natural, los sistemas de bases de datos, la ingeniería web, la gestión de datos web, la interacción del usuario, etc.

Por lo tanto, como estudiantes intentamos prepararnos para desempeñar un papel de liderazgo en la ciencia y la tecnología de los sistemas de información web, desarrollar una comprensión profunda de la construcción, métodos, técnicas y herramientas para el uso de dicho sistema.

## BIBLIOGRAFÍA

“Intentionally blank”.



