

Sistema de Administración de Farmacias

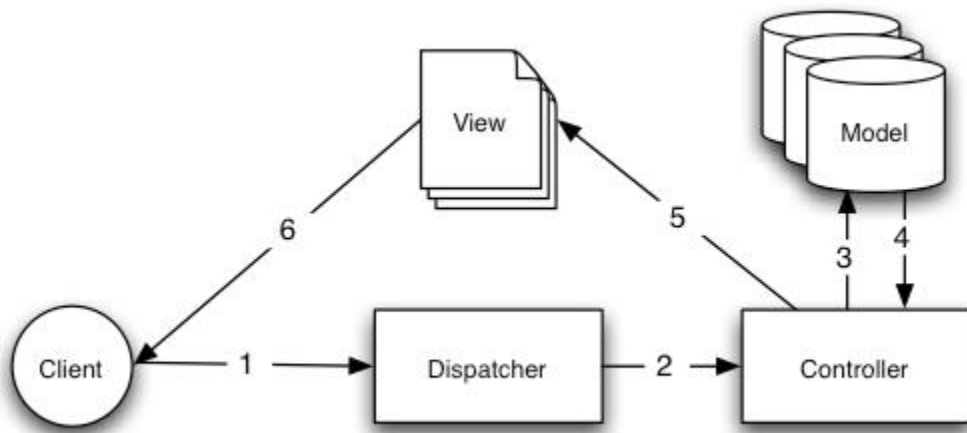
Prototipo Evolutivo Capa de Presentación

Versión 1.0

Introducción

La idea principal de este prototipo, es realizar los primeros pasos de la capa de presentación e implementar la base del patrón Modelo Vista Controlador(MVC) por medio del Framework Spring MVC.

Descripción del patrón utilizado con aclaraciones de la implementación



Cuando el cliente por medio de un navegador solicita una url, esta es capturada en primera instancia por el Dispatcher (Servlet), luego este se encarga de asignarle un controlador según el pedido. El controlador es capaz de interactuar con capas inferiores (como la capa lógica) para conseguir datos de interés y de este modo armar una vista (jsp) la cual es devuelta al cliente.

En este prototipo, está resuelto de modo genérico el Dispatcher, por lo cual agregar nuevas vistas o controladores no requiere la modificación del mismo.

¿Cómo funciona el Dispatcher?

En el web.xml del proyecto, se define las siguientes directivas:

```
<servlet>
  <servlet-name>dispatcher</servlet-name>
  <servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-
class>
  <load-on-startup>1</load-on-startup>
</servlet>
<servlet-mapping>
  <servlet-name>dispatcher</servlet-name>
  <url-pattern>*.htm</url-pattern>
  <url-pattern>*.html</url-pattern>
</servlet-mapping>
```

Se define un único servlet el cual es el dispatcher, el mismo va a aceptar cualquier pedido que concuerde con las siguientes expresiones regulares:

1. `.*\.html$`
2. `.*\.htm$`

Si no concuerda con alguna de estas expresiones se lanza una excepción diciendo que la url solicitada no existe, en caso que concuerde se le intenta asignar un controlador.

La forma de buscar un controlador está dada en el dispatcher, esta se implementó de forma tal que el dispatcher busque en un directorio específico, donde se le indicó que ahí están todos los controladores disponibles.

Esto se realizó con la siguiente línea:

```
<context:component-scan base-package="SAF.UI" />
```

Nota: Esto quiere decir que todos los controladores están en el paquete SAF.UI

¿Cómo defino un Controlador?

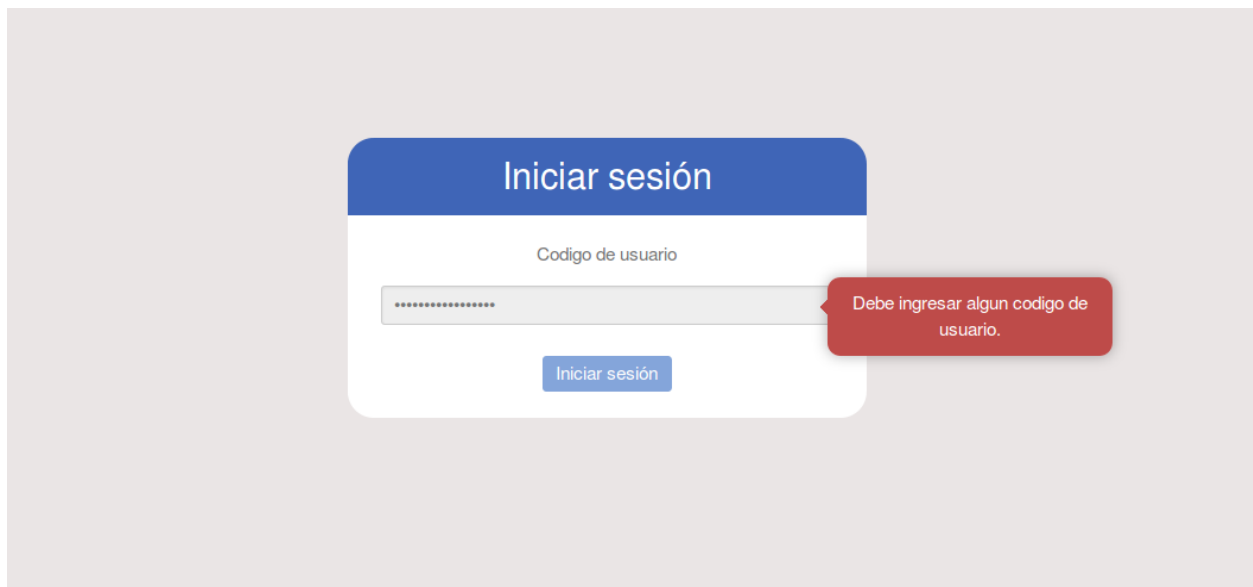
Se crea una clase java con la siguiente estructura:

```
@Controller // Esto indica que esta clase es un controlador
public class nombreDelControlador {
    @RequestMapping(value = "index") //Esto quiere decir que la operación va a
    responder por el pedido index
    public ModelAndView nombreOperacion(){
        -----Lineas de codigo-----
        return ( aquí va una vista ) //Se retorna una vista, la cual es devuelta al
        cliente
    }
}
```

Prototipo de presentación

La capa de presentación esta implementada en base a los framework Bootstrap y JQuery, con agregados personalizados.

Implementa un login inicial con un control de caracteres adecuados y campo no vacío, como se muestra a continuación.



El prototipo muestra un formulario de inicio de sesión con un encabezado azul que dice "Iniciar sesión". Debajo, hay un campo de texto etiquetado "Codigo de usuario" con caracteres ocultos por puntos. Un botón azul "Iniciar sesión" está debajo del campo. Un mensaje de error en un recuadro rojo indica: "Debe ingresar algun codigo de usuario."

Si bien el login está asociado a su controlador, el mismo carece de lógica.

Luego se implementó lo que sería la estructura de los demás cuerpos.

