

PROYECTO #2: SISTEMA PRESCRIPCIÓN Y DESPACHO DE RECETAS FRONTEND-BACKEND-BASEDATOS

EIF206- Programación 3 (2025-II)

Este proyecto consiste en hacer una nueva versión de la aplicación desarrollada para el Proyecto #1, pero bajo una nueva **arquitectura distribuida y usando bases de datos**, donde habrá dos procesos que se ejecutan (corren) de manera independiente (Frontend y Backend), pero que utilizan hilos (threads) y se comunican por medio de sockets; y donde el Backend almacenará los datos en una Base de datos (MySQL). **El Frontend y el Backend deberán poderse correr en máquinas separadas** dentro de una red. Obviamente **múltiples Frontend podrán estar conectados a un único Backend**. Las responsabilidades de cada uno se describen a continuación:

Backend

Corresponde a las capas **Lógica** y de **Datos**. Deberá ser implementado como un Servidor que no tiene ninguna interfaz de usuario. Más bien “escuchará” en un número de puerto seleccionado las peticiones que le lleguen de las instancias de Frontend y las atenderá y responderá. El Backend será el único que se conectará a la base de datos para recuperar o almacenar los datos allí, y utilizar dichos datos para responder las peticiones que le lleguen desde los Frontend.

Además de procesar y responder a cada Frontend las peticiones que éste le envíe, el Backend enviará **notificaciones asíncronas** a todos los demás Frontend's que estén activos, cada vez que un usuario ingrese (login) o salga (logout), o cuando un usuario desee enviar un mensaje (tipo chat) a otro usuario.

Frontend

Corresponde a la capa de **Presentación** de la aplicación (MVC: Modelo-Vista-Controlador). Al iniciar, el Frontend establecerá una conexión con el Backend. Luego usará esa conexión para enviar las peticiones atendiendo a las acciones del usuario; y también para recibir las notificaciones asíncronas del Backend. Este componente NO debe tener ninguna referencia relacionada con la base de datos.

Aparte de las funciones que ya se tenían (según el Proyecto #1), ahora también deberá tenerse las siguientes funcionalidades:

- **Área de Usuarios:** Un área, al lado derecho de la pantalla principal, donde mostrará la lista de los **usuarios** activos (*logueados*). A esta lista se le agregan o quitan usuarios cada vez que lleguen notificaciones asíncronas del Backend indicando que un usuario ingresó o salió, como se explicó antes.

Id	Nombre	Especialidad
MED-111	David Gurdian	Pediatría
MED-222	Miguel Angel	Neurocirugia

Id	Mensajes?
ADM-111	<input type="checkbox"/>
MED-111	<input checked="" type="checkbox"/>
MED-222	<input type="checkbox"/>

- **Enviar mensaje:** El usuario podrá escoger a otro de los usuarios que esté activo y enviarle un mensaje de texto (que se ingresa mediante una ventana emergente)
- **Recibir mensaje:** Se podrá seleccionar al usuario remitente y presionar el botón de “Recibir” para recibir dicho mensaje (que se mostrará en una ventana emergente).

Indicaciones

1. Uso de Proxy: se debe usar el patrón Proxy, de manera que el Frontend utilice una clase (proxy) que “represente” localmente al Service del Backend, y que sea esa clase la que se encargue de enviar y recibir a través de la red (sockets) las peticiones al verdadero Service, que es remoto.

Rúbrica

Funcionalidad	Porcentaje
1- Ingreso (login) y cambio de clave	10%
2- Prescripción	10%
3-Despacho	10%
4-Lista de Médicos	6%
5-Lista de Farmaceutas	6%
6-Lista de Pacientes	6%
7- Catálogo de medicamentos	6%
8-Dashboard (Indicadores)	20%
9-Histórico de recetas	6%
10-Usuarios	10%
11-Mostrar y recibir mensjaes	10%

Reglas

- La solución debe respetar estrictamente la **Arquitectura Distribuida** descrita anteriormente
- La capa de presentación deberá seguir la arquitectura **Modelo-Vista-Controlador** que se estudia en clase.
- La persistencia se debe hacer por medio de **base de datos**.
- Todos los datos deberán validarse y reportar adecuadamente cualquier error
- Entrega: por medio del aula virtual, en la siguiente fecha: **01 de noviembre**
- Deberá entregar los proyectos fuente como un único archivo comprimido, incluyendo el script de la base de datos.
- Se mantendrán los mismos equipos de estudiantes del Proyecto #1, a menos que se solicite algún cambio al profesor y éste lo autorice.
- Los proyectos deberán ser **presentados (defendidos)** por cada equipo. Para ello oportunamente cada equipo deberá seleccionar una cita de un cronograma de citas disponibles que en su momento se publicará.