

Apellido y Nombre: Legajo:

UNISocial

Contexto general

El Ministerio de Educación ha lanzado una licitación para realizar el diseño y desarrollo de una red social para cada una de las Universidades Públicas del país con la finalidad de mejorar y fortalecer las relaciones no formales dentro de cada institución, además de utilizar a ésta como un medio para compartir información. Hemos sido afortunados y logramos ganar la misma.



Nuestros analistas funcionales ya han realizado su trabajo y del mismo surgieron los siguientes requerimientos:

1. Un **usuario** deberá poder identificarse con: nombre, apellido, fecha de nacimiento, legajo o número de identificación dentro de la universidad, mail, número de teléfono móvil y **rol**. Los roles pueden ser: alumno, docente, no docente, directivo, empleado administrativo, empleado comercial.
2. Un usuario podrá pertenecer a uno o varios grupos. Un **grupo** es, en su forma más estricta, una agrupación de personas que tienen algo en común en la vida real.
3. Un usuario deberá poder enviar y recibir mensajes tanto a/de otros usuarios como a/de grupos. Cada vez que se reciba un mensaje, se deberá mostrar una **alerta** (en su versión web) o **notificación push** (en su versión móvil). Se sabe que otro equipo desarrollará esta parte del sistema (alertas y notificaciones) y que lo entregará en una etapa tardía del proyecto.
4. Cada usuario deberá poder realizar aportes. En un **aporte** se podrán compartir diferentes tipos de archivos junto a una especificación sobre qué tópico de la facultad está involucrado. El sistema deberá poder mostrar una vista previa de cada archivo. Por el momento, solamente se soportarán archivos de tipo:

- PDF: nos han sugerido la utilización de una biblioteca llamada Apache PDFBox, pero aún no se ha tomado la decisión (podría cambiar).
- Excel: el equipo de desarrollo evaluará y decidirá si utilizar Apache POI o JXLS.
- Word: aún no se sabe qué biblioteca podría implementarse.

Si importar el tipo de archivo y que bibliotecas se utilicen, se sabe que para mostrar una vista previa se debe:

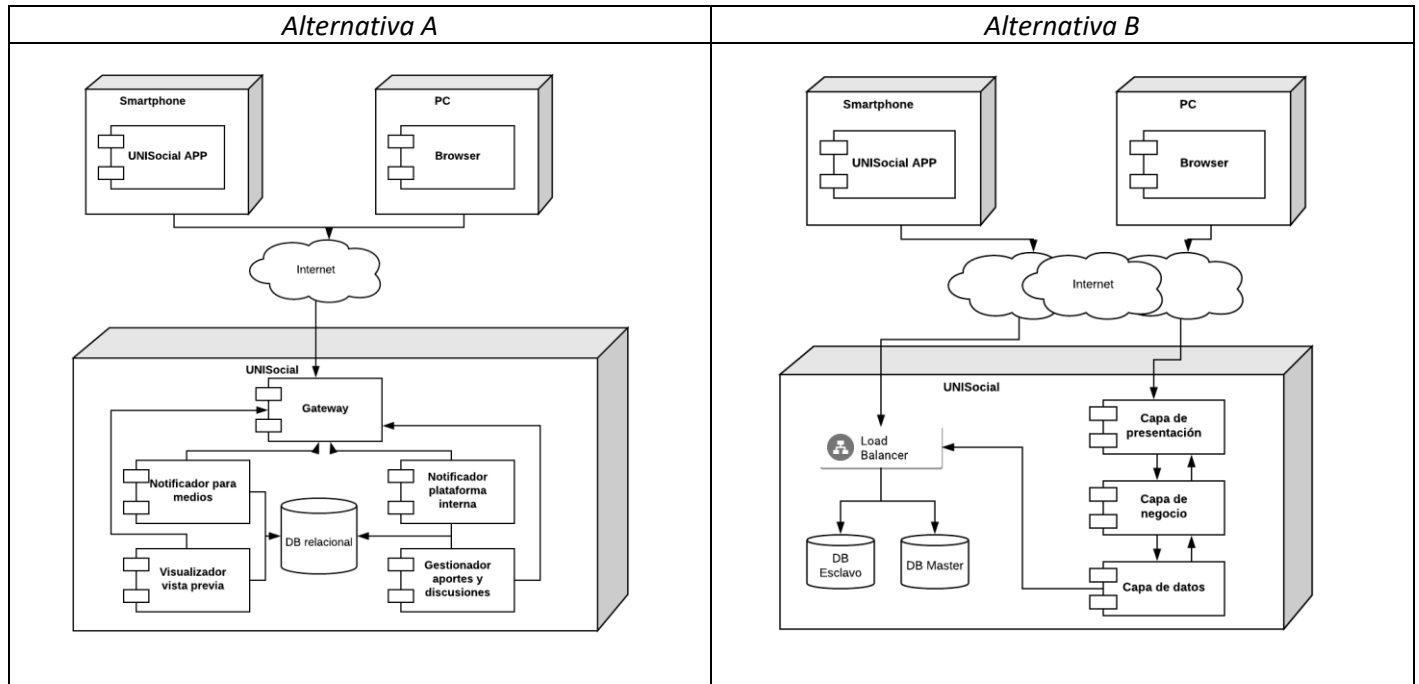
- a. Abrir el archivo.
 - b. Cargar el archivo.
 - c. Generar documento HTML.
 - d. Cargar documento HTML con la vista previa del archivo.
5. Cada usuario deberá poder puntuar aportes: cuando se realiza una puntuación sobre un aporte (del 1 al 5, donde 5 es "aporte genial" y 1 es "pésimo aporte"), ésta influirá positiva o negativamente sobre la **reputación** del usuario aportador. Un usuario nuevo siempre tiene "**buena reputación**", pero si cuando se le hace una puntuación, la sumatoria de todas las puntuaciones sobre la cantidad de aportes multiplicado por dos da <3 entonces pasará a tener "**mala reputación**". Cuando un usuario tiene mala reputación, no podrá realizar aportes. Para volver a tener buena reputación, el usuario administrador deberá darle otra chance de aporte.
 6. Cada usuario deberá poder realizar, contestar o seguir una **discusión**. En una discusión se plantea una pregunta base, y luego se espera que otros usuarios contesten.
 - a. Si A crea una discusión y B contesta, entonces A deberá ser notificado. Si A le vuelve a contestar a B, entonces B deberá ser avisado.
 - b. Si A sigue una discusión y cualquier otro usuario contesta, entonces A deberá ser notificado.

Además, cada usuario deberá poder elegir si, además de recibir notificaciones en la propia plataforma, también quiere recibir alertas vía mail, SMS o WhatsApp.

Por otra parte, el usuario creador de una discusión deberá poder **cerrarla** o **pausarla**. Si la cierra, nadie más podrá contestar a esa discusión, excepto que se reabra. Si la pausa, se deberá especificar por cuánto tiempo no se podrán recibir más respuestas. Solamente se puede pausar una publicación abierta.

Punto 1 – Arquitectura (30 puntos)

Dadas las siguientes propuestas de arquitectura:



Compárelas indicando ventajas y desventajas de cada una. ¿Cuál cree que es la más apropiada para el dominio presentado? ¿Por qué?

Punto 2 - Modelo de Dominio (40 puntos)

- Modelar el dominio presentado utilizando el paradigma orientado a objetos, comunicando su solución mediante un diagrama de clases. Si utiliza patrones de diseño, indíquelos y justifique su uso. **NOTA:** Puede ayudarse para comunicar, además, con código, pseudo-código, prosa u otros diagramas (diagrama de secuencia, de estados, entre otros). **(30 puntos)**
- El equipo de desarrollo nos ha enviado una vieja implementación sobre los medios de notificaciones:

```
>> notificador
public void avisarA(Usuario unUsuario, String mensaje){
    switch (unUsuario.medioDeNotificacion){
        CASE MAIL: {
            MimeMessage message = new MimeMessage(Usuario::session);
            message.setFrom(new InternetAddress(MAIL_NO_REPLY));
            message.addRecipient(Message.RecipientType.TO, new
InternetAddress(unUsuario.mail));
            message.setSubject("Tienes algo pendiente!");
            message.setText(mensaje);
            Transport.send(message);
        }; break;
        CASE WHATSAPP: {
            WhatsappSender.sendMessage(unUsuario.movil, mensaje);
        }; break;
        CASE SMS: {
            this.enviarSMS(unUsuario, mensaje);
        }; break;
        default: throw new Exception("No se puede notificar");
    }
}
```

}

Indique ventajas y desventajas sobre esta implementación. **(5 puntos)**

3. Muestre a través de código o pseudocódigo una propuesta de implementación para la vista previa de los archivos. **(5 puntos)**

Punto 3 – Persistencia (30 puntos)

Utilizando un DER, explique cómo persistirá el modelo del punto anterior indicando claramente:

- Qué elementos del modelo es necesario persistir.
- Las claves primarias, las foráneas y las restricciones según corresponda.
- Si fueran necesarias, estrategias de mapeo de herencia utilizadas. Explicar por qué fueron elegidas y compararlas con alguna otra.
- Justificaciones sobre las decisiones de diseño tomadas anteriormente.

Y sabiendo que el sistema deberá poseer **trazabilidad** con respecto a:

1. La reputación de los usuarios
2. Los estados de las discusiones

NOTA: Explicar supuestos y justificar decisiones de diseño.

Condiciones de aprobación: Para aprobar debe sumar como mínimo 60 puntos y no menos del 50 % en cada sección.