

4. Aplicación “MensajeYa”

Queremos desarrollar una aplicación “MensajeYa” para la integración de diferentes aplicaciones de mensajería instantánea del tipo Whatsapp, Telegram, Skype, etc. y sus contactos asociados. El objetivo es proporcionar a los usuarios una vista única de lectura de los mensajes/contactos a partir de su interacción en diversas aplicaciones de mensajería.

Una aplicación de mensajería viene descrita por su nombre, descripción, Web del desarrollador y una URL para el acceso a su API. Por otro lado, un usuario de “MensajeYa” posee un nombre, un número de teléfono, una dirección de correo electrónico para envío de notificaciones importantes y una fecha de nacimiento. Dicho usuario puede ser miembro de una o más aplicaciones de mensajería, a las que accede mediante una cuenta de usuario y una contraseña. Obviamente, la cuenta de usuario y contraseña variarán en función de la aplicación de mensajería. Por ejemplo, en el caso de Whatsapp la cuenta de usuario estará asociada al número de teléfono del usuario, mientras que en Skype puede ser un nombre y no un número de teléfono.

Cada usuario mantiene una lista de contactos, que habrá que dar de alta, con los contactos de sus aplicaciones de mensajería. Un requisito importante en el sistema es poder listar la totalidad de contactos existentes agrupados por aplicación de mensajería, y también agrupados por contactos “reales”, ya que puede darse el caso de que el mismo contacto esté repetido en más de una aplicación (por ejemplo, en Telegram y Whatsapp, con cuentas de usuario distintas representando al mismo contacto “real”). Por lo tanto, el sistema debe permitir añadir cuentas de usuario distintas de dichos contactos procedentes de las distintas aplicaciones de mensajería.

Cuando un usuario se da de alta en “MensajeYa”, debe proporcionar sus datos de perfil, e identificar de qué aplicaciones de mensajería es usuario, indicando los datos correspondientes. Para ello, el sistema ofrece al usuario una lista de las aplicaciones registradas previamente y el usuario, al seleccionar una de ellas, debe proporcionar la cuenta de usuario y la contraseña para acceder a la misma. A partir de ahí, el sistema debe construir una red de contactos que se almacenará para su uso posterior.

Cuando un usuario inicia sesión en “MensajeYa” se le ofrece la red de contactos; es decir, un contacto en “MensajeYa” es el definido como contacto en al menos una aplicación de mensajería. Para cada contacto se muestra al usuario las aplicaciones de mensajería en las que está presente y relacionado con el usuario. Al seleccionar uno de los contactos el sistema debe ofrecer dos posibilidades: 1) mostrar los últimos 4 mensajes de cada aplicación de mensajería en la que está el contacto, o 2) mostrar los últimos 10 mensajes del total de aplicaciones de mensajería, indicando mediante un icono el nombre de la aplicación de mensajería de la que procede cada mensaje. Con el fin de obtener esta información, se supone que existe un agente de mensajería para cada una de las aplicaciones registradas que extrae la información requerida sobre un contacto y la proporciona al sistema. Adicionalmente, al finalizar cada día se deberá generar un informe con los 5 contactos que más mensajes nos han enviado y enviárselo al usuario por correo electrónico. Nos interesa almacenar un histórico de esos informes, con la fecha y los 5 contactos.

El sistema debe permitir borrarse de una aplicación de mensajería cuando se estime una cantidad de información desmesurada. Esto implicará la pérdida de los contactos de dicha aplicación, aunque pueden seguir siendo contactos a través de otra aplicación. Evidentemente, si se trataba de la última aplicación de mensajería que me relacionaba con ese contacto, se perderá (es decir se deberá eliminar) dicho contacto

El sistema también debe gestionar (altas y bajas) cuál es la aplicación de mensajería favorita de cada contacto y una serie de contactos favoritos. De esta forma, el sistema debe ofrecer dos formas de envío de mensajes a un contacto favorito: 1) envío normal, donde se enviará un mensaje mediante la aplicación de mensajería favorita del contacto, y 2) envío urgente, donde se enviará el mismo mensaje utilizando todas las aplicaciones de mensajería en las que esté registrado el contacto. Además, en el caso de envío urgente se puede programar una repetición automática *insistente* para que se re-envíe el mismo mensaje cada hora hasta que el contacto responda. Para los contactos no favoritos solo se puede realizar el envío normal de mensajes.

Por el momento no se permite generar grupos de contactos con la idea de enviar mensajes (normales y urgentes) a dichos grupos, aunque está previsto que se incorpore en futuras versiones.

A partir de la descripción anterior, se pide:

- a) Obtener el diagrama de casos de uso en UML.
- b) Obtener el diagrama de clases en UML, incluyendo los atributos de las clases que creáis oportunos, así como los nombres de las relaciones que detectéis. No es necesario indicar los métodos.