

## Proyecto 1

### Uso de un protocolo existente (XMPP)

#### 1 Antecedentes

La mensajería instantánea es una de las invenciones que han revolucionado la comunicación de persona a persona. No obstante, muchos servicios requieren el uso de protocolos privados que limitan y obligan al usuario a usar únicamente las aplicaciones desarrolladas por el proveedor.

XMPP (*eXtensible Messaging and Presence Protocol*) es un protocolo abierto con más de 10 años de desarrollo que permite la interconexión entre distintos proveedores de mensajería instantánea. Éste provee las características de manejar cifrado, ser descentralizado, ser extensible, fácil de escalar y ser totalmente comunitario.

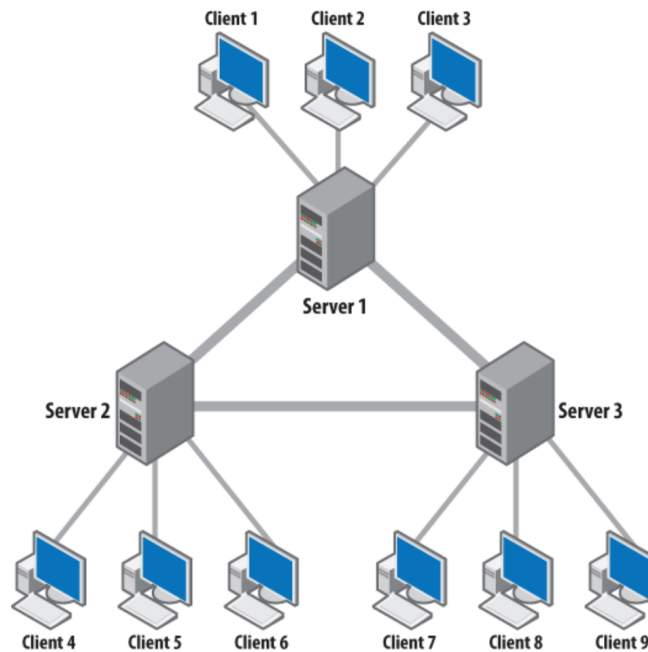


Imagen 1: Arquitectura de XMPP. Fuente: (Saint-Andre, Smith, & Tronçon, 2009)

## 2 Objetivos del proyecto

- Implementar un protocolo en base a los estándares.
- Comprender el propósito del protocolo XMPP.
- Comprender cómo funcionan los servicios del protocolo XMPP.
- Comprender las bases de programación asíncrona requeridas para algunas de las necesidades de desarrollo en redes.

## 3 Desarrollo

### 3.1 Características y limitaciones

El proyecto consiste en implementar un cliente de mensajería instantánea que soporte el protocolo XMPP. A partir de ello debe de soportar como mínimo las siguientes funcionalidades:

Administración de cuentas (20% del funcionamiento, 5% cada funcionalidad)

- 1) Registrar una nueva cuenta en el servidor
- 2) Iniciar sesión con una cuenta
- 3) Cerrar sesión con una cuenta
- 4) Eliminar la cuenta del servidor

Comunicación (80% del funcionamiento, 10% cada funcionalidad)

- 1) Mostrar todos los usuarios/contactos y su estado
- 2) Agregar un usuario a los contactos
- 3) Mostrar detalles de contacto de un usuario
- 4) Comunicación 1 a 1 con cualquier usuario/contacto
- 5) Participar en conversaciones grupales
- 6) Definir mensaje de presencia
- 7) Enviar/recibir notificaciones
- 8) Enviar/recibir archivos

El proyecto debe estar definido con una **interfaz de consola** (CLI). Puede utilizar cualquier lenguaje de programación, siempre y cuando este permita compatibilidad con distintos sistemas operativos.

Pueden utilizarse librerías que faciliten la comunicación con el protocolo XMPP. No obstante, no se puede utilizar una librería o framework que resuelva por sí misma las dificultades de programación concurrente. En caso de duda, consultar.

Puesto que se estará utilizando una interfaz de consola no es necesario que se muestre el avatar del contacto o cualquier imagen que envíe.

El servidor XMPP será proporcionado por los docentes. El dominio para utilizar es: [alumchat.xyz](http://alumchat.xyz)

### 3.2 Documentación y uso de control de versiones

Cada estudiante debe de llevar su proyecto con un sistema de control de versiones de su preferencia, el cual debe de ser privado durante el desarrollo y público hasta el día de entrega, para que sea descargado en el momento de la evaluación del proyecto (o bien, totalmente privado pero brindando acceso al catedrático). El control de versiones debe ser coherente con el desarrollo gradual del proyecto. Por ejemplo, un proyecto con pocos commits no refleja un versionamiento apropiado.

El código debe de estar debidamente documentado, utilizando las mejores prácticas para el lenguaje. Adicionalmente, debe de poseer al menos un documento README.md que describa las características del proyecto, las funcionalidades implementadas y la manera en que un usuario puede instalarlo y usarlo en su computadora (**en inglés**).

No es necesario, pero se motiva que al menos los comentarios para cada *commit* en el proyecto se encuentren también en inglés.

### 3.3 Presentación del proyecto

El día de la presentación los estudiantes serán divididos en tres grupos (asumiendo 2 auxiliares). Cada estudiante debe de demostrar la funcionalidad de su proyecto frente a los demás estudiantes de su grupo, explicando únicamente lo siguiente:

- Características implementadas
- Dificultades
- Lecciones aprendidas

De manera que cada presentación debe de durar un máximo de 5 minutos. Luego se procederá a una revisión individual.

## 4 Evaluación

El proyecto tiene una ponderación de 10 puntos netos, los cuales están distribuidos de la siguiente manera:

- [70%] Funcionamiento del proyecto (las funcionalidades solicitadas)
- [15%] Documentación y uso de sistema de control de versiones
- [15%] Presentación del proyecto

### 4.1 Extras

- Se otorgará 20% extra a aquel alumno que implemente el protocolo XMPP con el intercambio directo de los elementos XML (o sea, armando el XML "a mano" y enviándolo con primitiva send, versus la llamada a la librería utilizada. Consultar si hay dudas), implementando correctamente al menos el 90% de las funcionalidades solicitadas. **Esta calificación es binaria** (+20% xor +0%).
- Se otorgará 20% extra a aquel alumno que **además** de la aplicación de consola, desarrolle una GUI que utilice como API las funciones implementadas en consola, y que se haya implementado el 100% de las funciones en consola. **Esta calificación es binaria** (+20% xor +0%).

## 4.2 Rúbrica

Para las funcionalidades se manejarán los siguientes rangos:

- 100%: la funcionalidad está implementada correctamente
- 99% - 75%: la funcionalidad falla en casos muy concretos o eventuales.
- 75% - 50%: la funcionalidad falla constantemente, pero funciona en algunos casos.
- 50% - 0%: la funcionalidad no está implementada, contiene errores en el código o no funciona en ningún escenario.

Para cada rango, se explicará el por qué de la nota obtenida durante la calificación

---

La documentación manejará los siguientes rangos:

- Documentación (35%)
    - 100%: El archivo README contiene con detalle todas las secciones solicitadas , los comentarios son utilizados apropiadamente en el código.
    - 99% - 75%: el archivo README contiene las secciones solicitadas, pero no se detallan con exactitud (Ej: faltan algunas instrucciones para instalar o usar el cliente), los comentarios son utilizados de forma apropiada.
    - 75% - 50%: el archivo README existe, pero el nivel de detalle es escaso y no es posible instalar o usar el cliente únicamente con las instrucciones proporcionadas. Los comentarios son genéricos.
    - 50% - 0%: no existe el archivo README o no hay comentarios.
  - Control de versiones (75%)
    - 100%: el número de commits es adecuado al desarrollo gradual del proyecto durante las cinco semanas de desarrollo.
    - 99% - 50%: el número de commits no refleja el desarrollo gradual del proyecto, existen commits con varios días de distancia.
    - 50% - 0: existen muy pocos commits, o la mayoría de commits fueron realizados en los días previos a la entrega del proyecto.
- 

La presentación manejará los siguientes rangos:

- 100%: La presentación contiene las funcionalidades realizadas, las dificultades y cómo se resolvieron y las lecciones aprendidas.
- 99% - 50%: la presentación contiene las secciones, pero hay poco detalle visual o verbal.
- 50% - 0%: la presentación no aborda las secciones solicitadas, los detalles son genéricos.

## 5 Recomendaciones

Se sugiere comprender el funcionamiento de los servicios de XMPP y practicar con los distintos tipos de *stanzas* antes de utilizar una librería. Se sugiere el cliente de XMPP Gajim para practicar las stanzas en la consola XML. El libro “XMPP – The definitive guide” contiene una explicación muy completa del funcionamiento de los servicios de XMPP y está disponible en Canvas.

Verificar que las librerías utilizadas estén vigentes. Hay librerías deprecadas o que tienen años sin recibir una actualización.

Revisar la documentación de cada librería. Algunas tienen configuraciones adicionales para conectarse a los servidores XMPP.

Se recomienda utilizar lenguajes como Java y C++ ya que su manejo de concurrencia e hilos es superior al de otros lenguajes. En Python existen librerías pero (1) la concurrencia tiene una curva de aprendizaje alta si no se tiene experiencia previa (2) históricamente ha presentado dificultades su uso (3) la librería disponible para Python está algo olvidada y desactualizada. Por esas razones es que este año se prohíbe el uso de Python para el proyecto 1 (excepto para su GUI si desean esos extras).

Durante el desarrollo y pruebas, verificar que la cuenta o cuentas de pruebas están conectadas en un solo cliente, o cerciorarse que en caso de múltiples conexiones, el cliente que se prueba tenga la prioridad más alta para recibir los mensajes.

## 6 Recursos

Saint-Andre, P., Smith, K., & Tronçon, R. (2009). *XMPP : The Definitive Guide*. Disponible en Canvas.

XMPP protocol Introduction and Overview: <https://www.youtube.com/watch?v=92egt5-UDwo>

XMPP software: <https://xmpp.org/software/>

XEP-0459: XMPP Compliance Suites 2022 <https://xmpp.org/extensions/xep-0459.html>