



UNIVERSIDAD  
**NACIONAL**  
DE COLOMBIA

Universidad Nacional de Colombia - sede Bogotá  
Facultad de Ingeniería  
Departamento de Sistemas e Industrial  
Curso: Ingeniería de Software 1 (2016701)

**Profesor: Oscar Eduardo Alvarez Rodriguez**

**Estudiante(s):**

**Nelson Ivan Castellanos Betancourt**

**Angel Santiago Avendaño Cañon**

**David Sebastian Hurtado Sanchez**

**Nicolas Zuluaga Galindo**

### **Tarea 3**

Durante la clase práctica de análisis de requerimientos se trabajó un desarrollo para crear un medio que permita comunicar estudiantes y profesores. A continuación se presentan las justificaciones para las estimaciones del Anexo 1:

- 1. Registro e inicio de sesión:** El equipo estimó que para este requerimiento se iban a utilizar un total de 5 días. Durante este tiempo se piensa hacer la interfaz gráfica, la encriptación de las claves y el proceso de transición entre la información suministrada por la universidad en una base de datos a la de mongo.
- 2. Lista de contactos dinámica:** Desarrollar la funcionalidad de lista dinámica para el chat entre estudiantes y profesores nos tomará 3 días porque estamos usando MongoDB, una herramienta que no conocemos y necesitamos aprender. Esto incluye entender cómo organizar los datos de usuarios, mensajes y chats grupales. Además, debemos diseñar cómo funcionará la base de datos para soportar los chats en tiempo real e integrar con el sistema actual. También necesitamos pruebas para casos como el envío simultáneo de mensajes, la búsqueda en chats y la gestión de miembros. Al ser una tecnología nueva para nosotros, será necesario investigar y resolver problemas, lo que hace que este tiempo sea razonable.
- 3. Chat individual:** Para este requerimiento se estimaron 3 días de labor, esto debido a que plantea generar un chat básico que sirva para el envío y recepción de mensajes. Es posible utilizar un mayor número de días en caso de que se busque el envío de imágenes, audios y/o documentos.
- 4. Chat Grupal:** Teniendo en cuenta que esta implementación se debe hacer primero que la del Chat Individual (que bien podrían hacerse en el mismo tiempo) es posible que tome más tiempo, pues requiere de diseñar y estructurar los modelos e implementarlos. Una vez la parte del backend esté hecha, en la que yo estimaría 2 días, quedaría el desarrollo de interfaz gráfica. Estimo que puede tomar 1 día aproximadamente desarrollar la parte gráfica (frontend). Esto aliviana la carga del desarrollo de interfaz gráfica para el Chat individual, en caso de requerir más tiempo para el desarrollo del Chat Grupal se podría tomar tiempo del chat individual, pues el

chat individual se puede pensar cómo un caso específico del chat grupal para el cuál ya estarían desarrolladas las bases o serían muy similares.

5. **Historial de mensajes:** En el equipo todos consideramos que 3 días es más que suficiente para tener implementada esta característica. En principio es una característica que se basaría en un MessageRepository con operaciones básicas CRUD, su dificultad es baja/moderada. La mayor dificultad estará en manejar las claves derivadas que cifran la clave del chat y mantiene la seguridad E2E. Una vez claro el diseño que permite mantener la seguridad de la aplicación su desarrollo podrá ser llevado con normalidad.
6. **Notificaciones en tiempo real:**El desarrollo de notificaciones en tiempo real nos tomará 5 días debido a la complejidad técnica y la falta de experiencia previa con MongoDB. Necesitamos aprender cómo gestionar cambios en tiempo real, posiblemente usando herramientas como Change Streams, e integrarlo con un backend que permita enviar notificaciones rápidas y precisas. Esto implica diseñar la lógica para que las notificaciones lleguen a los usuarios adecuados, considerando casos como mensajes nuevos, menciones o actualizaciones en los grupos, además de probar su funcionamiento en diferentes escenarios, como usuarios desconectados o múltiples dispositivos. Este tiempo también incluye investigar y resolver problemas que puedan surgir durante el desarrollo.
7. **Búsqueda de mensajes y contactos:** Según cómo se planteen los modelos de mensajes esta característica tendrá dificultad moderada o alta, pensando en el rendimiento puede llegar a una dificultad alta. Estimamos que 3 días pueden ser suficientes para esta característica, enfocando los esfuerzos en el rendimiento de la búsqueda de mensajes, bajo el supuesto que un chat pueda tener miles de mensajes diferentes. Por otro lado la búsqueda de contactos parece ser más sencilla, pues las búsquedas estarán limitadas a la cantidad de contactos y además estas pueden ser cacheadas cuándo alcanzan una alta recurrencia.
8. **Integración con Calendario académico:**La integración con el calendario académico nos tomará 5 días debido a la necesidad de entender cómo conectar el sistema de chat con una herramienta externa o construir una funcionalidad personalizada. Esto incluye diseñar cómo los eventos del calendario, como fechas de exámenes, entregas o reuniones, se sincronizarán y se mostrarán en el chat, permitiendo a los usuarios interactuar con ellos de manera sencilla. Además, debemos implementar la lógica para actualizar los eventos en tiempo real y garantizar que se reflejen correctamente en los chats grupales e individuales. Finalmente, se requiere tiempo para realizar pruebas, resolver errores y asegurar que la funcionalidad sea intuitiva y eficiente para los usuarios.
9. **Seguridad:** La seguridad es fundamental para cualquier desarrollo, por lo que este requerimiento se debe implementar con un gran detalle. Teniendo en cuenta esto, el equipo estimó un total de 5 días para poder realizar una revisión frente a la librerías de encriptación requeridas y para aprender acerca de su uso y manejo debido a que ningún miembro del equipo las ha implementado.

**10. Rendimiento:** Para asegurar un buen rendimiento en la aplicación, se estima un tiempo de 5 días, un periodo pertinente que será dedicado a optimizar las consultas a la base de datos, mejorar la eficiencia de los algoritmos utilizados para los chats, y asegurar que el sistema sea capaz de manejar gran número de solicitudes de diferentes usuarios de manera simultánea sin afectar significativamente el tiempo de respuesta.

**11. Mantenibilidad:** Se estima un promedio de 5 días para implementar prácticas de mantenibilidad en el proyecto, incluyendo refactorización y organización del código, siguiendo patrones de diseño y código limpio, generación de documentación detallada, implementar arquitectura modular, y elaborar pruebas unitarias que permitan identificar errores rápidamente para futuras modificaciones.

**12. Disponibilidad:** Para garantizar la disponibilidad de la aplicación, se asignaron 5 días para diseñar e implementar estrategias capaces de asegurar un funcionamiento continuo e ininterrumpido de las características principales de la aplicación, incluyendo chats, inicio de sesión y consultas, esto incluye configurar redundancia en los servidores, manejar errores de manera correcta, al igual que excepciones que pueden llegar a generar caídas inesperadas, así mismo, se considerará el uso de herramientas de monitoreo para detectar problemas en tiempo real para realizar mantenimiento preventivo y asegurar una disponibilidad completa del sistema para todos los usuarios, incluso en picos grandes de uso.

**NOTA:** Es posible que la redacción de este documento sea mejorada con la herramienta Chat GPT.

## Referencias

*Experiencias propias de cada integrante del equipo.*

## Anexo 1: Estimaciones individuales según requerimientos

	Requerimientos	Nicolás	Sebastian	Ivan	Angel	Promedio	Max	Min	Desv Std
1	Registro e inicio de sesión	5	5	5	8	5	8	5	1,5
2	Lista de contactos dinámica	3	5	3	5	3	5	3	1,2
3	Chat individual	5	3	3	3	3	5	3	1,0
4	Chat Grupal	1	2	3	5	3	5	1	1,7
5	Historial de mensajes	3	3	3	3	3	3	3	0,0
6	Notificaciones en tiempo real	5	8	3	2	5	8	2	2,6
7	Búsqueda de mensajes y contactos	1	2	3	5	3	5	1	1,7

Facultad de Ingeniería- Departamento de Sistemas e Industrial

8	Integración con Calendario académico	5	8	3	8	5	8	3	2,4
9	Seguridad	5	5	3	5	5	5	3	1,0
10	Rendimiento	3	2	8	5	5	8	2	2,6
11	Mantenibilidad	5	3	13	5	5	13	3	4,4
12	Disponibilidad	3	3	8	4	5	8	3	2,4
	Total	<b>44</b>	<b>49</b>	<b>58</b>	<b>58</b>	<b>52,25</b>	<b>81</b>	<b>32</b>	