



Tarea 3

Ingeniería de software I

Julián Dario Colmenares Saenz

Kevin David Rodriguez Riveros

Frank Sebastian Pardo Amaya

Jorge Andrés Torres Leal

Departamento de Ingeniería de Sistemas e Industrial

Universidad Nacional de Colombia - Sede Bogotá

Facultad de Ingeniería

28 de Noviembre de 2024

TAREA 3

- Estimar por cada estudiante cada una de las funcionalidades

Respuesta de Frank:

- **MUST :**
 - Mantenibilidad.
 - Chat grupal.
 - Registro Inicio de Sesión.
 - Historial de Mensajes.
- **SHOULD :**
 - Búsqueda de Contactos
 - Chat individual.
 - Lista de Contactos Dinámica.
- **COULD :**
 - Seguridad.
 - Rendimiento.
 - Disponibilidad 24/7.
- **WONT :**
 - Notificaciones en tiempo real.
 - Integración con calendario académico.

Respuesta de Jorge:

- **MUST :**
 - Registro e inicio de sesión
 - Chat grupal
 - Seguridad
 - Búsqueda de contactos
 - Mantenibilidad
 - Historial de mensajes
- **SHOULD :**
 - Chat individual
 - Rendimiento
 - Disponibilidad
 - Lista de contactos
- **COULD :**
 - Notificaciones en tiempo real
- **WONT :**
 - Integración con calendario académico

Respuesta de Julián:

- **MUST :**
 - Seguridad
 - Registro Inicio de Sesión.
 - Historial de Mensajes.
 - Mantenibilidad
 - Chat grupal
 - Búsqueda de contactos
- **SHOULD :**
 - Chat individual
 - Rendimiento
 - Disponibilidad
 - Lista de contactos
- **COULD :**
 - Notificaciones en tiempo real
- **WONT :**
 - Integración con calendario académico.

Respuesta de Kevin:

- **MUST :**
 - Registro e inicio de sesión.
 - Chat grupal
 - Historial Mensajes
 - Seguridad
- **SHOULD :**
 - Rendimiento
 - Disponibilidad
 - Chat individual
 - Mantenibilidad
- **COULD :**
 - Búsqueda de contactos
 - Lista de contactos
 - Notificaciones en tiempo real
- **WONT :**
 - Integración con calendario académico
- Individualmente colocar su número de fibonacci y una breve explicación

Respuesta de Frank:

- **MUST :**
 - Mantenibilidad (3): Considero que es un pro tener el código organizado para las futuras actualizaciones, como en principio tendríamos la funcionalidades fundamentales en el desarrollo de esta herramienta. Será útil y más fácil desde

el principio ir clasificando las líneas de código a medida que se ponen las funcionalidades.

- Chat grupal (13): Se necesitará tiempo porque implica almacenamiento de datos y sincronización en tiempo real entre múltiples usuarios en un mismo espacio (en este caso, un chat). También se deben manejar eventos como usuarios que entran o salen de los grupos, asegurando la consistencia de los mensajes y datos.
- Registro Inicio de Sesión (5): Me parece que toca crear al menos una base de datos con ciertos parámetros que identifiquen a cada usuario y que al registrarse se agreguen ahí mismo. Requiere tiempo para pruebas y ajustes.
- Historial de Mensajes (13): Permitir la persistencia de los mensajes requiere diseñar una base de datos eficiente. En este caso, el historial debería estar almacenado en un servidor central para garantizar accesibilidad desde cualquier dispositivo, pero con sincronización adecuada para que funcione sin conexión (opcionalmente). Requiere pensar bien la estructura.
- **SHOULD :**
 - Búsqueda de Contactos (3): Según lo entendido, es un poco sencillo porque la base de datos que se usaría facilitaría este tipo de tareas, pero también implica agregar una barra de búsqueda y filtrado con el historial y la lista de contactos.
 - Chat individual (8): Ya teniendo como base el chat grupal, el cuál es fundamental, se puede implementar un chat individual que implicaría tener usuarios 1:1, pero la sincronización de mensajes para el historial sigue siendo necesario.
 - Lista de Contactos Dinámica (3): Esta se podría sacar fácilmente interactuando con la base de datos del registro de inicio de sesión, siento que no sería tan complicado crearla.
- **COULD :**
 - Seguridad (21): Implementar encriptación puede ser complicado, pero me parece que ya para versiones finales de la herramienta es necesario ya contar con esta característica. Le pongo 21 porque al ser de seguridad debería ser auditado.
 - Rendimiento (8): Optimizar el rendimiento incluye asegurar que las consultas a la base de datos sean rápidas, minimizar los tiempos de carga en el frontend, y utilizar técnicas como compresión de datos o carga diferida (lazy loading). Esto garantizaría que el sistema sea más ágil para los usuarios.
 - Disponibilidad 24/7 (5): Esto requiere configurar servidores eficientes y redundantes, pero me parece más sencillo porque es simplemente tener más infraestructura funcional que mantenga el buen funcionamiento del sistema.
- **WONT :**
 - Notificaciones en tiempo real (21): Muy complejo porque exige un sistema robusto de push notifications y sincronización constante con servicios de los dispositivos de cada usuario, además de servidores configurados para eventos en vivo. No me parece tan factible colocar esta característica puesto que al ser algo académico, los mismos usuarios (estudiantes y docentes) deberían estar pendientes de sus actividades, se llama responsabilidad.

- Integración con calendario académico (21): También complejo porque implica interactuar con sistemas externos o APIs en este caso de la universidad, lo que sería nuevo para mí personalmente.

Disclosure: Algunas respuestas mías fueron revisadas y ajustadas por ChatGPT, un modelo de lenguaje desarrollado por OpenAI, para que tuvieran más técnica y fueran más lógicas para un desarrollador profesional. Los puntajes y clasificación final de las funcionalidades son de mi auditoría basándome en mi conocimiento y análisis del tema.

Respuesta de Jorge:

A continuación uso la misma estructura de mi respuesta anterior, ya que siento que es justamente bajo esto que debemos asignar la cercanía (qué tan cercano respecto a la fecha de inicio o rápido debe de realizarse) con que debe finalizarse, inicié con la mantenibilidad ya que siento que es algo que debe mantenerse desde el día uno para el resto del proyecto o sea literalmente desde la planeación debemos tener la mantenibilidad como algo presente, luego pues pensé que en cuanto al registro básicamente se trata de realizar la autenticación en una base de datos que debemos modelar desde cero y la parte gráfica, además la parte de búsqueda de contactos la puse como 5 días ya que una vez modelada la base de datos realizar la parte que hará la búsqueda será algo muy sencillo, en cuanto al chat individual , seguridad e historial de mensaje siento que es algo que se puede ir implementando de manera paralela ya que una vez tenemos el chat pues almacenaremos los mensajes en la base de datos y los recuperaremos, además la seguridad es algo que debe ponerse en el chat de manera que los mensajes procedan a encriptarse .

En la parte de SHOULD puse el chat grupal como de 13 días ya que al tener implementado el envío de mensaje individuales, seguridad e historial sería fácil simplemente hacer que varios chats se reúnan en un mismo canal, por el lado de la lista de contactos siento que es algo como simplemente filtrar y no es algo esencial por lo tanto le pongo esa fecha, finalmente el rendimiento y la disponibilidad es algo que podemos mejorar una vez hayamos salido de las funcionalidades más esenciales.

Las notificaciones es algo que añadiremos en una segunda versión o de últimas ya que siento que al inicio las personas pueden estar bastante al pendiente de los mensajes y demás ya que es algo muy académico y por lo tanto es algo sobre lo que los estudiantes y profesores pueden estar al pendiente por si mismos.

- **MUST :**

- Registro de inicio de sesión (3)
- Chat individual (8)
- Seguridad (8)
- Búsqueda de contactos (5)
- Mantenibilidad (1)
- Historial de mensajes (8)

- **SHOULD :**

- Chat grupal (13)
- Rendimiento (21)
- Disponibilidad (21)

- Lista de contactos (13)
- **COULD :**
 - Notificaciones en tiempo real (34)

Respuesta de Julián:

- **MUST :**
 - Seguridad (8): Me parece que es importante para todo lo que se vaya a hacer, sin esta podríamos comprometer información nuestra o de los usuarios.
 - Registro Inicio de Sesión.(3): Es un requisito básico para cualquier aplicación.
 - Historial de Mensajes.(8): Los usuarios esperan poder acceder a conversaciones pasadas para revisar, buscar o continuar diálogos previos.
 - Mantenibilidad(1): La mantenibilidad asegura que la plataforma pueda ser actualizada, corregida y mejorada a lo largo del tiempo. Sin una estructura mantenible, sería difícil realizar mejoras o solucionar problemas.
 - Chat grupal(5): El chat grupal es fundamental para aplicaciones que buscan una comunicación más amplia, como en ambientes de trabajo o grupos sociales.
 - Búsqueda de contactos(3): La búsqueda de contactos facilita la localización rápida de personas dentro de la plataforma.
- **SHOULD :**
 - Chat individual(8): Le daremos mas importancia al chat grupal.
 - Rendimiento(21): El rendimiento puede ser optimizado con el tiempo sin que sea un factor limitante en el lanzamiento inicial.
 - Disponibilidad(21): Los usuarios creo que podrían tolerar un pequeño tiempo de inactividad o interrupciones ocasionales, por lo que esta característica se ubica en un nivel inferior en comparación con la seguridad o el historial de mensajes.
 - Lista de contactos(21): Una lista de contactos es útil para organizar y gestionar las personas con las que interactuamos, sin embargo, no es tan esencial como otras funciones, ya que los usuarios pueden agregar contactos manualmente o buscar por otros medios.
- **COULD :**
 - Notificaciones en tiempo real(18): Aunque estas mejoran la experiencia general del usuario, no es una necesidad absoluta para la funcionalidad básica.
- **WONT :**
 - Integración con calendario académico.(21): La integración con el calendario académico es un valor agregado. Dado que se trata de una característica adicional, no es una prioridad y, por lo tanto, queda fuera de nuestro alcance.

Respuesta de Kevin:

- **MUST :**
 - Registro e inicio de sesión. (3): 3 días porque se tendrían factores de funcionamiento y de validación de dicho registro e inicio de sesión. Teniendo en

cuenta que está ubicado en must, así que es de las tareas ha tener como prioridad y realizar lo más pronto posible

- Chat grupal (5): Después de tener un registro e inicio de sesión, necesitamos cumplir con ese objetivo de comunicación entre las partes de la comunidad educativa, en ese orden de ideas me parece lo más apropiado abordarlo por medio del chat grupal
- Historial Mensajes (8): Lo pongo con una escala de 8 porque aunque es importante lo hago partiendo de que al ser días hábiles previamente se habrían utilizado algunos días para tareas previas como lo son las dos ya mencionadas, es decir, el chat grupal para la respectiva comunicación y el debido registro e inicio de sesión para cada persona
- Seguridad (8): La seguridad aunque es un tema fundamental la pondría un poco más de 5 días porque sinceramente no tengo muchas nociones en este tema y debería indagar y adquirir un poco de idea sobre el tema para ya poder ponerlo en práctica, y también teniendo en cuenta que esta seguridad será aplicada especialmente justo en la sesión de cada persona y más específicamente en los chats que existan anclados a algún perfil

- **SHOULD :**

- Rendimiento (21): Como se vio en clase aunque me parece un factor importante porque da mejor impresión al cliente y si lo debería tener, sin embargo, con el ejemplo que tanto se dió del SIA aunque debería tenerlo en mi opinión creo que podría ser un poco más adelante.
- Disponibilidad (13): También considero que debería tener una disponibilidad adecuada para que se puedan ver todas las pruebas en mayor tiempo posible y porque somos todas personas en mundos tan diferentes que podemos usar una misma aplicación pero al tener diferentes hábitos y horarios las podemos visitar en tiempos muy diferentes
- Chat individual (8): Ya teniendo realizado el importante chat grupal, se podría de una manera similar ir pensando en un chat individual, puesto que hay situaciones en las que no todo el mundo debe o más bien necesita saber lo que hablan algunas personas en particular. (Cosas como excusas médicas, situaciones particulares, etc)
- Mantenibilidad (1): Creo que desde el día uno es importante empezar con buenas prácticas, por eso lo pongo en el pronto 1. Haciendo referencia a desde un inicio hacer uso de estas practicas

- **COULD :**

- Búsqueda de contactos (21): Este requisito lo considero que sería un buen plus pero al no ser dentro de lo que destacaría como lo más vital le doy un poco más de plazo de días para realizar
- Lista de contactos (21): Similar a lo que mencione anteriormente creo que tener esta lista de contactos vendría muy bien y podría darle bastante buena forma a todo el desarrollo de la aplicación, pero justamente después de tener realizadas otras partes completadas ubicadas más arriba

- Notificaciones en tiempo real (34): Concuerdo con lo que se socializo en clase, hay aplicaciones que funcionen sin necesariamente estar implementando notificaciones, siento que sería interesante tener la opción pero le doy 34 días porque sería una de las cosas finales a tratar
- **WONT :**
 - Integración con calendario académico (?): Por como se plantea me gustaría, sin embargo, considero que ya este se saldría un poco de los requerimientos iniciales que se solicitan para la aplicación así que no le determinaría un número de días en la escala porque ya sería como algo bastante opcional
- Llegar a un consenso (decir en cuanto tiempo se va a trabajar)

El tiempo total estimado de desarrollo es de 24 días laborales , considerando que algunas tareas pueden realizarse en paralelo. Las funcionalidades más complejas, como rendimiento y mantenibilidad, se trabajarán de manera continua a lo largo del proyecto.