

# Requerimientos

## Funcionales:

- Calendarización (Capacidad de poder asignar a ciertas fechas una entrega de tarea ya sea de forma manual o automática mediante la sincronización).
- Creación de equipos (Permitir la creación de equipos mediante un vínculo o código permitiendo crear asignaciones simultáneas)
- Revisión de actividades (Disponibilidad de revisión de los trabajos para la evaluación de estos)
- Notificaciones (La web tendrá un sistema de notificaciones personalizable o predeterminado)
- Estadística de calificación (Opción de visualización de calificaciones en ciertos trabajos en forma de gráficas)
- Chat (Disponibilidad de un chat para las asignaciones en equipos)
- Sincronización (Opción de vincular tu cuenta de alguna plataforma institucional, teams, Classroom, etc para la calendarización automática de las tareas)

## • No funcionales:

La aplicación debe mostrar las tareas calendarizadas en menos de 2 segundos después de iniciar sesión.

- El chat debe permitir la comunicación en tiempo real con un retardo máximo de 1 segundo.
- El sistema debe soportar hasta 500 usuarios conectados de manera simultánea sin degradar el rendimiento.
- Las cuentas de usuario deben requerir autenticación con correo y contraseña segura.
- La interfaz debe ser intuitiva y accesible desde dispositivos móviles y de escritorio.
- Debe seguir lineamientos de accesibilidad (ejemplo: contraste de colores, textos legibles).
- Debe guardar automáticamente los cambios realizados en las tareas.

- El código debe estar documentado y permitir futuras actualizaciones sin reestructurar toda la aplicación
- La aplicación debe funcionar en Android e iOS y encontrarse en las principales tiendas de aplicación (Google Play y App store)

## Priorización:

### Funcionales:

1. **Calendarización** (Este es el corazón de la aplicación por lo tanto debería ser lo primero en lo que centrarnos)
2. **Notificaciones** (De nada serviría un calendario si este no nos avisa de las entregas por lo que es importante ser de lo primero que se desarrolle)
3. **Revisión de tareas** (Esta será la principal función para los profesores por lo que es de vital importancia que sea de lo primero en desarrollar)
4. **Creación de equipos** (Esto es algo más complementario sin embargo para la siguiente función es necesario desarrollar este primero)
5. **Chat** (Este surge como necesidad del requerimiento anterior por lo que estos van de la mano)
6. **Sincronización** (Este es de los requerimientos más complejos por lo que se requieren más tiempos y recursos, de igual forma no es indispensable por lo que puede quedar como uno de los últimos requerimientos)
7. **Estadísticas de calificación** (Esta se desarrolla como una función para complementar la parte del profesor, sin embargo, se puede hacer de forma manual por lo que lo dejamos hasta el último)

### Todos los requerimientos:

8. Autenticación con correo y contraseña segura (no funcional, pero crítico para seguridad).
9. Rendimiento en tiempo real del chat (no funcional, asegura buena experiencia).
10. Interfaz intuitiva y accesible (no funcional, afecta usabilidad).
11. Disponibilidad multiplataforma (Android/iOS, escritorio).
12. Carga rápida (<2 segundos).
13. Accesibilidad (contraste, textos legibles, etc.).
14. Soporte hasta 500 usuarios.
15. Guardado automático de cambios.
16. Código documentado y actualizable.

## Diagrama casos de uso:

