

Priorización MoSCoW en clase							
Must-Have		Should-Have		Could-Have		Won't-Have	
Registro e Inicio de sesion		Chat individual		Historial de mensajes		Integracion con calendario academico	
Lista de contactos dinamica		Notificaciones en tiempo real		Busqueda de mensajes y contactos			
Chat grupal		Mantenibilidad		Seguridad			
		Disponibilidad		Rendimiento			
Planning Poker (Priorización MoSCoW Profesor)							
Must-Have	Dias	Should-Have	Dias	Could-Have	Dias	Won't-Have	
Registro e Inicio de sesion	8	Chat individual	1	Mantenibilidad	8	Integracion con calendario academico	2
Seguridad	2	Disponibilidad	8			Notificaciones en tiempo real	1
Chat grupal	3	Lista de contactos dinamica	1				
Busqueda de mensajes y contactos	1	Rendimiento	13				
Historial de mensajes	1						
Total	15		23		8		3
Requerimiento	Días	Razón					
Registro e Inicio de sesion	8	Primero se debe estudiar la conexión con el sistema de gestión de la universidad, para así poder realizar una correcta implimentación del inicio de sesion y validar las cuentas, por lo tanto, se requiere tiempo para conocer su funcionamiento					
Seguridad	2	Puesto que es un cifrado end-to-end y se debe cumplir con los estándares, es simple implementar dicha encriptación por la cantidad de recursos disponibles					
Chat grupal	3	Realizar dicho requerimiento no es demasiado complejo, teniendo en cuenta que solo estamos considerando envio y recepcion de textos sin multimedia con una interfaz minima					
Busqueda de mensajes y contactos	1	La base de datos como tal, nos brinda una funcion para ello					
Historial de mensajes	1	Simplemente es llevar un registro en los chats y hacer que no se borre automaticamente					
Chat individual	1	Contemplando la implemenacion de chat grupal, habilitar un chat individual es trivial ya que solo haría falta hacer la respectiva adaptación.					

	Disponibilidad	8	Generalmente, las aplicaciones de este tipo no necesitan que el sistema tenga una disponibilidad tan alta aunque lo deseable es que funcione la mayor parte del tiempo. Implementar una aplicación con alta disponibilidad implica construir sistemas con arquitecturas que manejen los distintos escenarios en donde la aplicación podría fallar.
	Lista de contactos dinamica	1	Este requerimiento funcional podría considerarse sencillo de realizar teniendo en cuenta que es un ejercicio muy común
	Rendimiento	13	Consideramos que de todos los requerimientos funcionales es el más complejo de realizar dado que implica modificar gran parte de lo que se tiene para lograr así una mejoría en el rendimiento general del sistema.
	Mantenibilidad	8	De los comentarios que el profesor hizo en clase sobre cómo en algunas empresas se dedica gran parte del tiempo a realizar la documentación pensamos que este requerimiento tomaría bastante tiempo, y sería de alguna manera proporcional a la magnitud del proyecto como tal
	Integración con calendario académico	2	La razón fundamental sobre el por qué decidimos establecer un tiempo tan reducido para este requerimiento es que realmente gran parte del trabajo será realizado por una API externa y es muy poco el código que se debe crear desde cero para lograr que funcione.
	Notificaciones en tiempo real	1	Al igual que el caso anterior generalmente las plataformas para las cuales se diseña el sistema proveen todos los mecanismos necesarios para lograr tener esta función, por lo general lo que se debe hacer es conectarlos.