



# AppChat

31/12/2019

---

Alfonso López Baños

Manuel González García

Tecnologías de Desarrollo de Software

Grupo 2.1

alfonso.lopezb@um.es

manuel.gonzalezg@um.es

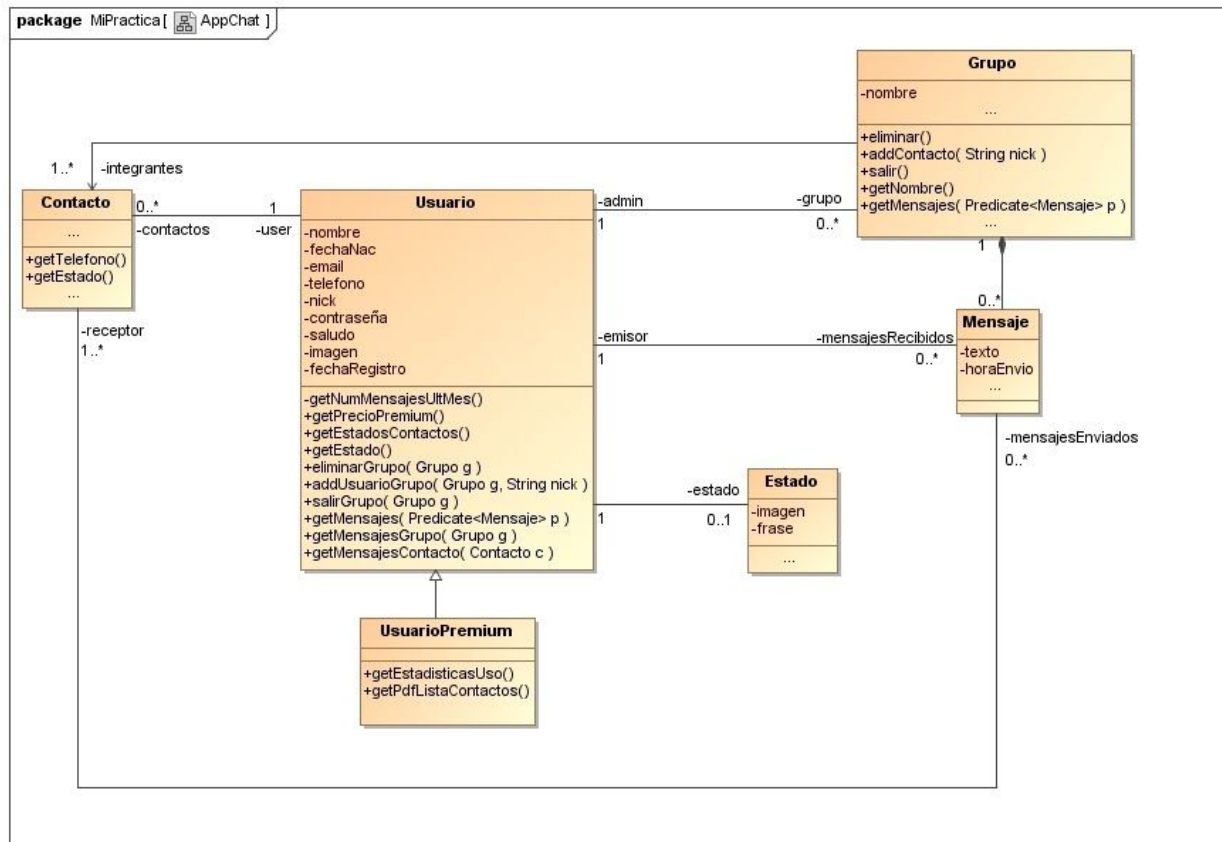
Enero 2020

# Índice

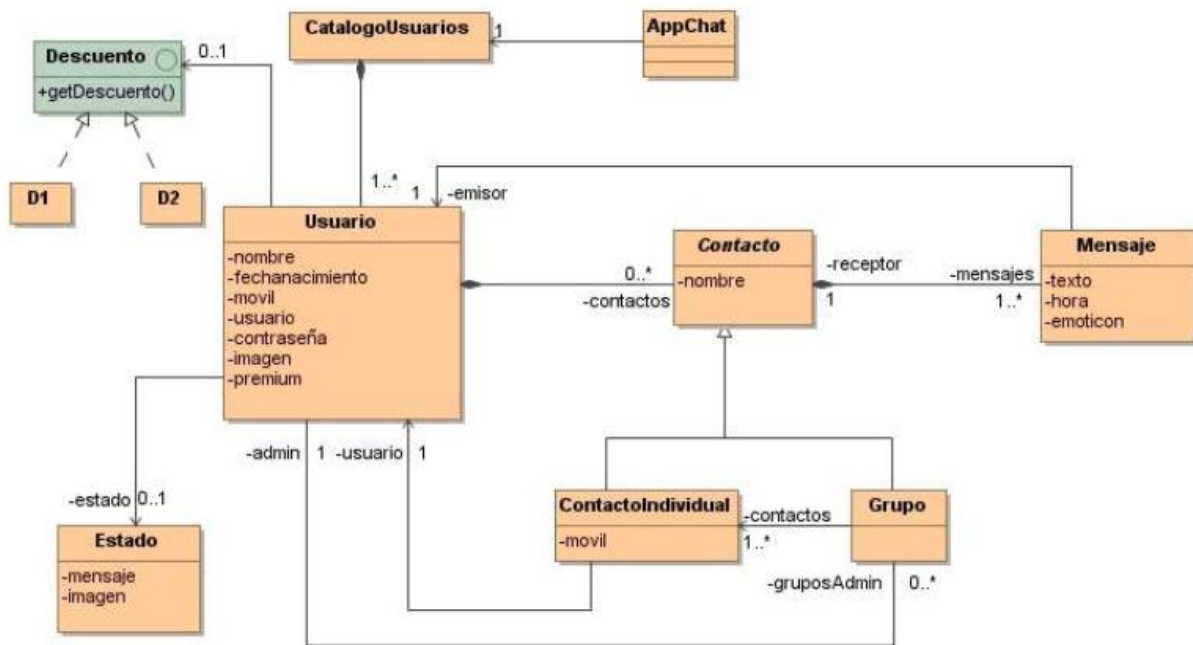
<b>Diagrama de clases del dominio</b>	<b>2</b>
<b>Diagrama de interacción</b>	<b>3</b>
<b>Arquitectura y diseño de la aplicación</b>	<b>4</b>
<b>Patrones de diseño</b>	<b>6</b>
Patrones de diseño utilizados directamente	6
Patrones de diseño utilizados indirectamente	6
<b>Componentes</b>	<b>6</b>
<b>Test unitarios</b>	<b>7</b>
Test Fachada	7
Suite de persistencia	7
<b>Manual de usuario</b>	<b>7</b>
Acceso a la aplicación e inicio de sesión	7
Registro	8
Envío de mensajes	9
Gestionar fotos de perfil y modificar nuestro saludo	11
Cambiar estado	12
Crear contacto	14
Crear grupo	15
Modificar grupo	16
Mostrar contactos y exportarlos a PDF	17
Convertirse en premium	19
Cerrar sesión	20
Ver información del contacto con el que estoy hablando	20
Importar conversaciones	21
Buscar mensajes	22
Mostrar estadísticas	23
Borrar todos los mensajes del chat seleccionado	24
Borrar contacto	25
<b>Observaciones</b>	<b>26</b>

## Diagrama de clases del dominio

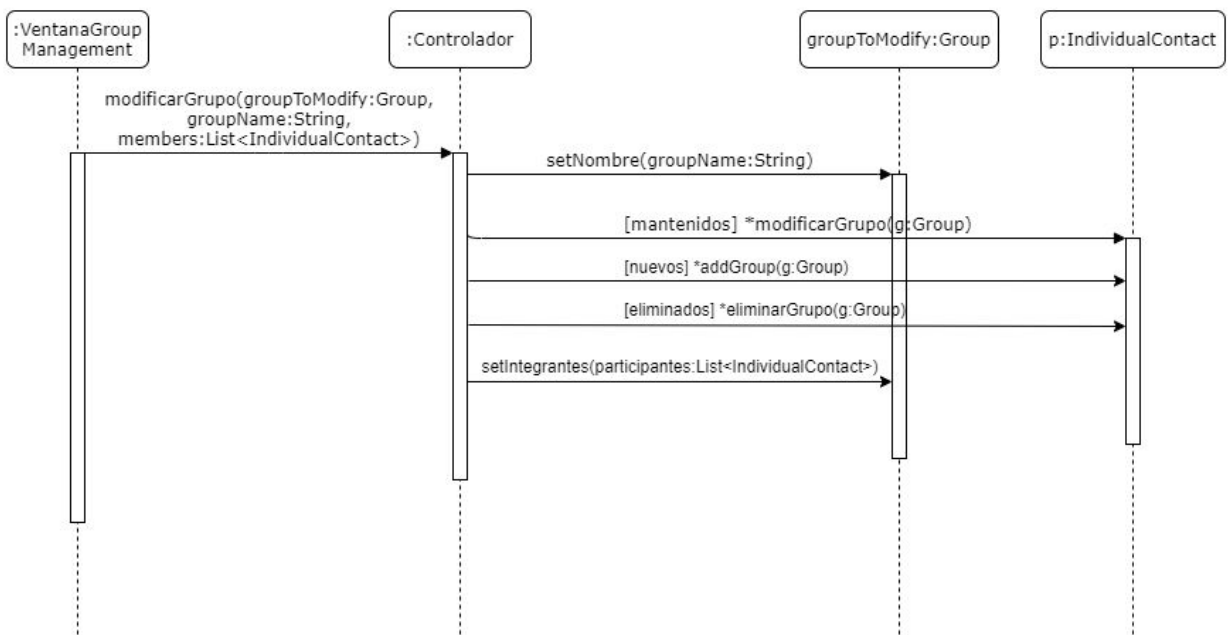
Nuestro diagrama de clases entregado fue el siguiente:



Finalmente hemos seguido para la aplicación el diagrama aportado por el profesorado:



## Diagrama de interacción



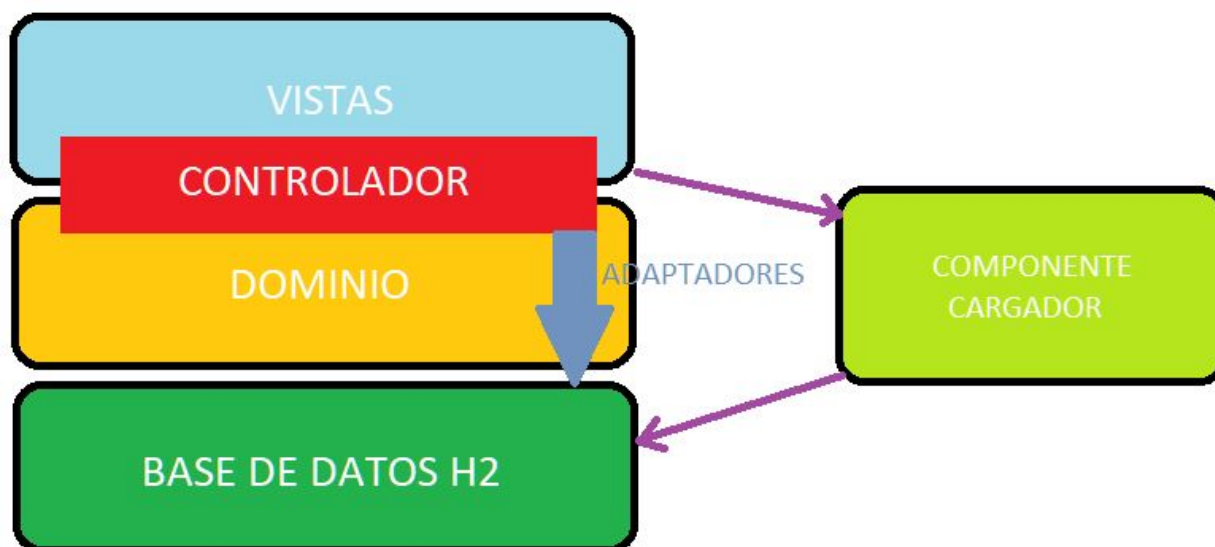
Para modificar un grupo hemos creado una operación genérica, pues el usuario puede tanto cambiar el nombre del grupo como modificar sus participantes (añadir y eliminar usuarios). La ventana le indica al controlador que se ha modificado el grupo pasándole el grupo modificado, su nuevo nombre y su nueva lista de miembros.

Será el controlador el encargado de modificar el grupo insertando su nuevo nombre y sus nuevos participantes. Diferenciamos a los usuarios de la lista de participantes del grupo de la siguiente forma:


- Mantenidos: Les pasaremos el grupo modificado para que lo actualicen y puedan tener disponibles todos los cambios.
- Nuevos: Los añadimos al grupo, para ello le pasamos el nuevo grupo del que ahora son participantes.
- Eliminados: Usuarios para los que se les borra el grupo.

## Arquitectura y diseño de la aplicación

La aplicación consta de 16 clases para las ventanas que formarán parte de la capa de vistas, los adaptadores para las diferentes entidades que se guardan en persistencia, las clases del dominio, el componente cargador de mensajes y el controlador que hace de fachada entre vistas y dominio.



Otro elemento a mencionar es el paquete que se encarga del guardado de los recursos gráficos que utilizan las ventanas de la aplicación. Además del paquete `umu.tds.apps.photos`, que se encarga de guardar las imágenes de los usuarios para que no tengan rutas absolutas en su ordenador sino una ruta relativa al proyecto.



Cada vez que se añade una foto de perfil o se crea un estado y se cierra la aplicación es necesario refrescar el proyecto para que eclipse encuentre las imágenes y pueda cargarlas.

En lo referente al diseño, una de las decisiones tomadas fue la creación de la clase interna Carrousel en la clase UserSettings. Esta clase se encarga de gestionar la lista con las imágenes del usuario y mostrar la imagen adecuada según su índice de recorrido. Es una clase interna porque está estrechamente relacionada con la ventana UserSettings, pues utiliza las etiquetas empleadas como identificador y foto mostrada.

Otra decisión tomada fue la creación de una clase Theme en el paquete de las vistas con el objetivo de reunir las constantes y métodos que consideramos propios de la temática del proyecto y comunes a todas las clases. Se recogen constantes como los colores empleados en las vistas y métodos como aquellos que redimensionan imágenes. Así, en caso de querer cambiar algo del diseño, es tan sencillo como cambiar el valor de una constante en lugar de cambiar los valores en todas las clases.

Debemos comentar que tuvimos que crear la clase interna ChatBurbujas que implementa Scrollable para que el chat con las burbujas no mostrase la barra horizontal y reduciese el espacio que hay entre mensajes enviados y recibidos. Este chat se emplea para mostrar conversaciones en la ventana principal y en la ventana de búsqueda.

Por último añadir que los mensajes de una conversación de Whatsapp sólo se podrán importar si todos los emisores de los mensajes aparecen en el archivo de texto con el mismo nombre que tienen dichos usuarios en la aplicación.

## Patrones de diseño

### I. Patrones de diseño utilizados directamente

Los patrones de diseño implementados por nosotros son:

- Patrón DAO (servicio de persistencia).
- Patrón adaptador (servicio de persistencia).
- Patrón factoría abstracta (servicio de persistencia).
- Patrón singleton (servicio persistencia, controlador y catálogo de usuarios).
- Patrón fachada (controlador).
- Patrón observador (componente de carga de mensajes).
- Patrón command (aplicación de los descuentos a un usuario).
- Patrón composite (los grupos podrían considerarse como un agregado de contactos individuales).

### II. Patrones de diseño utilizados indirectamente

Los patrones de diseño que hemos usado indirectamente:

- Patrón bridge (Java AWT: Botones, listas, scrollbar, etc).
- Patrón adaptador (uso de modelos en tablas y listas).
- Patrón composite (Java AWT: Ventanas y paneles).
- Patrón decorador (bordes y scrollbar).
- Patrón flyweight (bordes).
- Patrón estrategia (bordes).
- Patrón iterador (colecciones).
- Patrón observador (MDE, por ejemplo, en los botones).

## Componentes

Para cargar en la aplicación una conversación de whatsapp exportada como texto hemos creado la clase MessagesCharger, que guarda la lista con los mensajes de Whatsapp importados. Esta clase será el componente que, cuando se produzca un cambio en la lista de mensajes importados, notificará al Controlador para que guarde en base de datos y en el catálogo los mensajes importados. El Controlador será el único oyente que tendrá la clase MessagesCharger.

El cargado de los mensajes se hace por medio de la librería disponible en el Aula Virtual que parsea el archivo de texto según el formato y devuelve una lista con los mensajes.

También hemos utilizado el componente Luz, un componente gráfico para la ventana principal. Este componente está disponible en el aula virtual.

## Test unitarios

Hemos implementado una suite de test para comprobar el buen funcionamiento de la base de datos h2 y un test que se encarga de comprobar que funciona correctamente el controlador (y por tanto varios elementos del dominio).

### I. Test Fachada

Prueba que funcione la realización de funciones básicas a través del controlador como el registro de un usuario, inicio de sesión, creación de contactos y creación de estados. También nos ha servido para inicializar la base de datos y así realizar ciertas pruebas manuales sin tener que inicializar los valores a mano.

### II. Suite de persistencia

Conjunto de clases de test que se encargan de comprobar que funciona el guardado y recuperación de las entidades empleando nuestros adaptadores.

## Manual de usuario

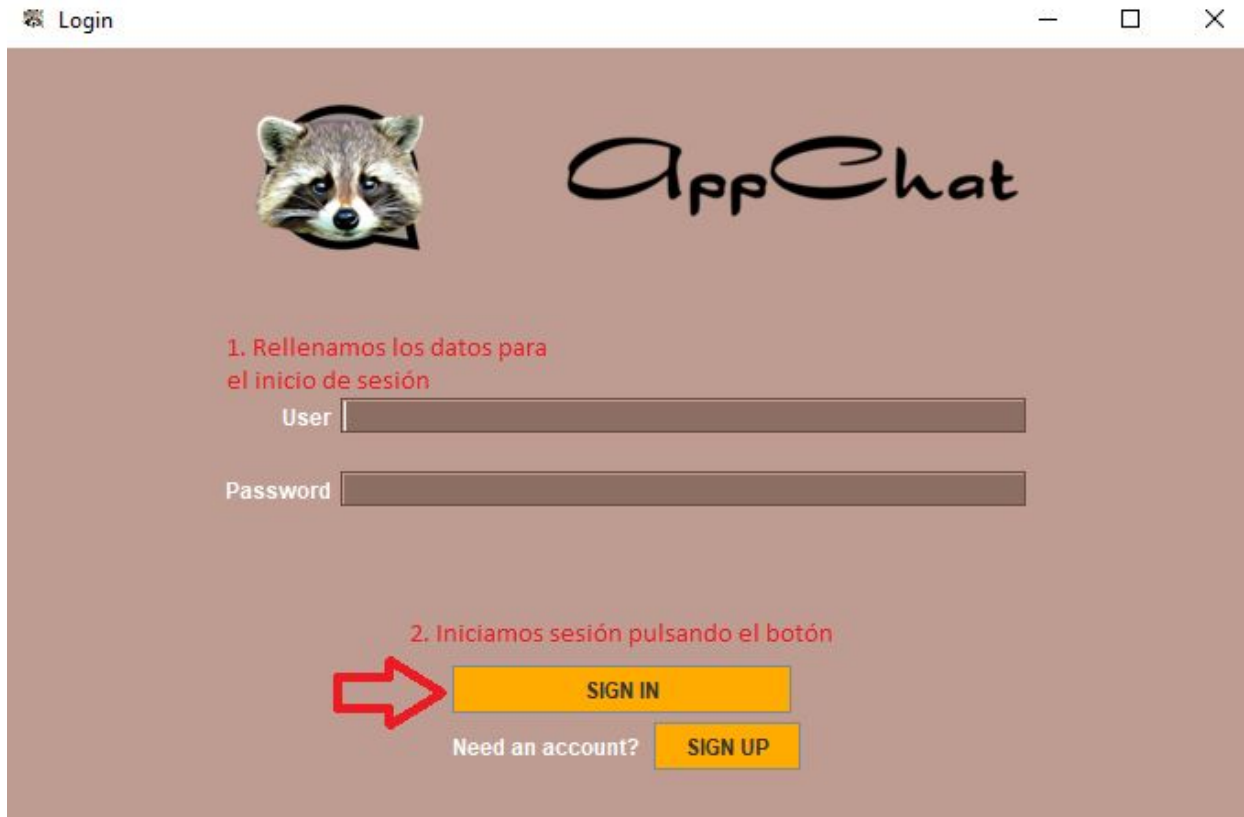
### I. Acceso a la aplicación e inicio de sesión

Para lanzar la aplicación es necesario que esté corriendo el servidor H2 en nuestro ordenador.

Una vez lanzamos la aplicación se nos iniciará la ventana de Login. Si ya estamos registrados, indicamos nuestro usuario y contraseña en los campos correspondientes y pulsamos en el botón "SIGN IN" para iniciar sesión.

En el caso de no tener cuenta y querer registrarnos, pulsamos el botón "SIGN UP" que nos llevará a la ventana de registro.





Login

AppChat

1. Rellenamos los datos para el inicio de sesión

User

Password

2. Iniciamos sesión pulsando el botón

**SIGN IN**

Need an account? **SIGN UP**

## II. Registro

A la hora de registrarse, nos dirigimos a la ventana de registro pulsando el botón "SIGN UP" y una vez en ella, rellenamos todos los campos. Tendremos una foto de avatar por defecto pero si quisiéramos poner una propia, nos ponemos encima de ella y hacemos click, eligiendo así una imagen con extensión .jpg o .png de nuestro ordenador.

Una vez estén rellenados todos los datos pulsamos en el botón "CREATE ACCOUNT", lo cual nos registrará y nos llevará a la ventana principal, iniciando así al mismo tiempo la sesión.



**SIGN IN**

Need an account? **SIGN UP**

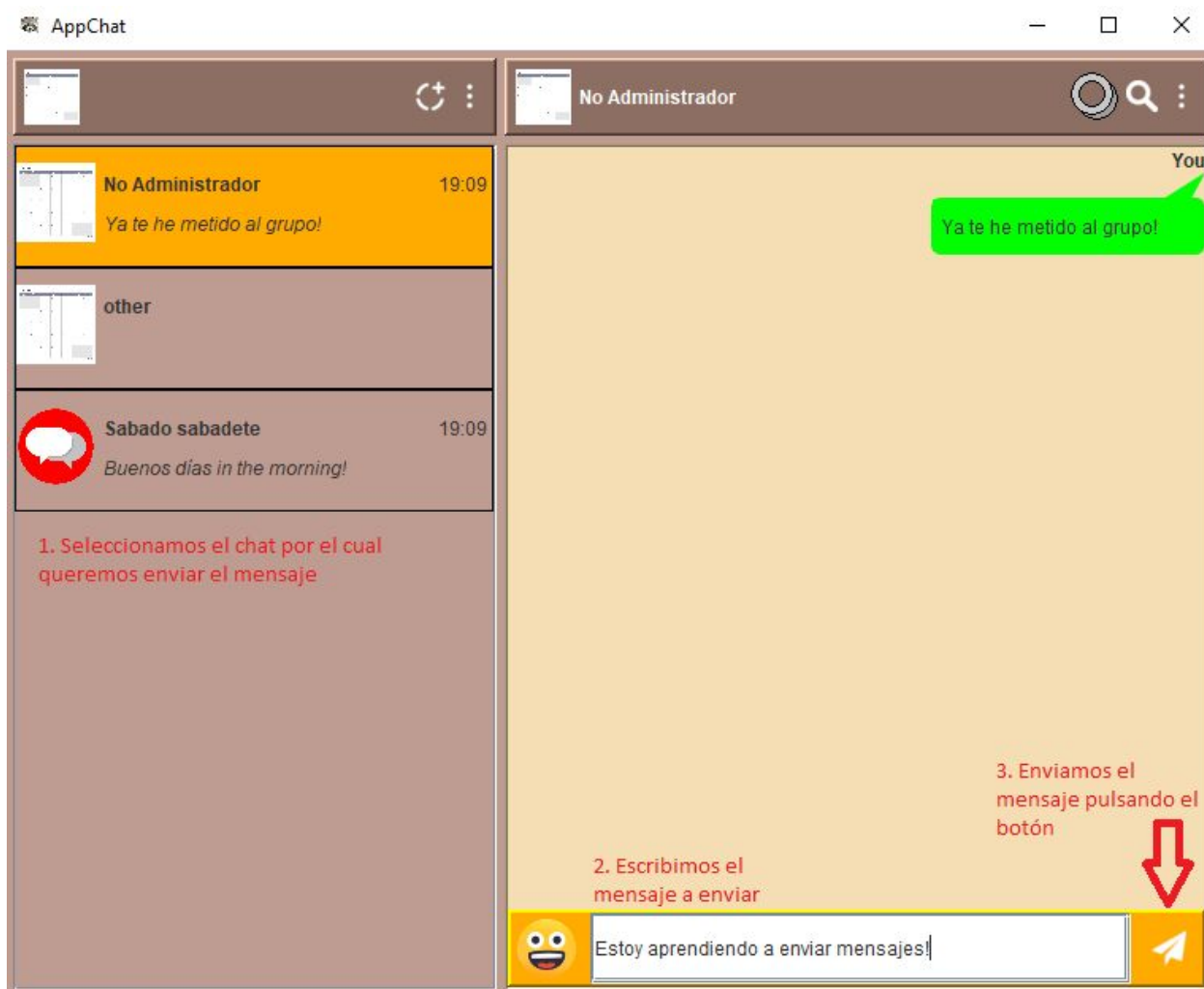
1. Accedemos a la ventana de registro pulsando el botón

The image shows a 'Register' window with a light brown background. At the top left is a small icon and the title 'Register'. At the top right are window control buttons (minimize, maximize, close). In the center is a circular placeholder for a profile picture. To the right of the profile picture, the text '2. Rellenamos los datos de nuestra nueva cuenta' is displayed in red. Below this text are three input fields: 'User', 'Password', and 'Confirm password'. To the left of the profile picture, there are three input fields: 'Email', 'Phone number', and 'Name'. Below the 'Name' field is a 'Birth Date' field with the value '13-ene-2020' and a calendar icon. At the bottom center is a large yellow button labeled 'CREATE ACCOUNT'. A red arrow points to this button. Below the 'CREATE ACCOUNT' button is the text 'Have an account?' and a yellow button labeled 'SIGN IN'.

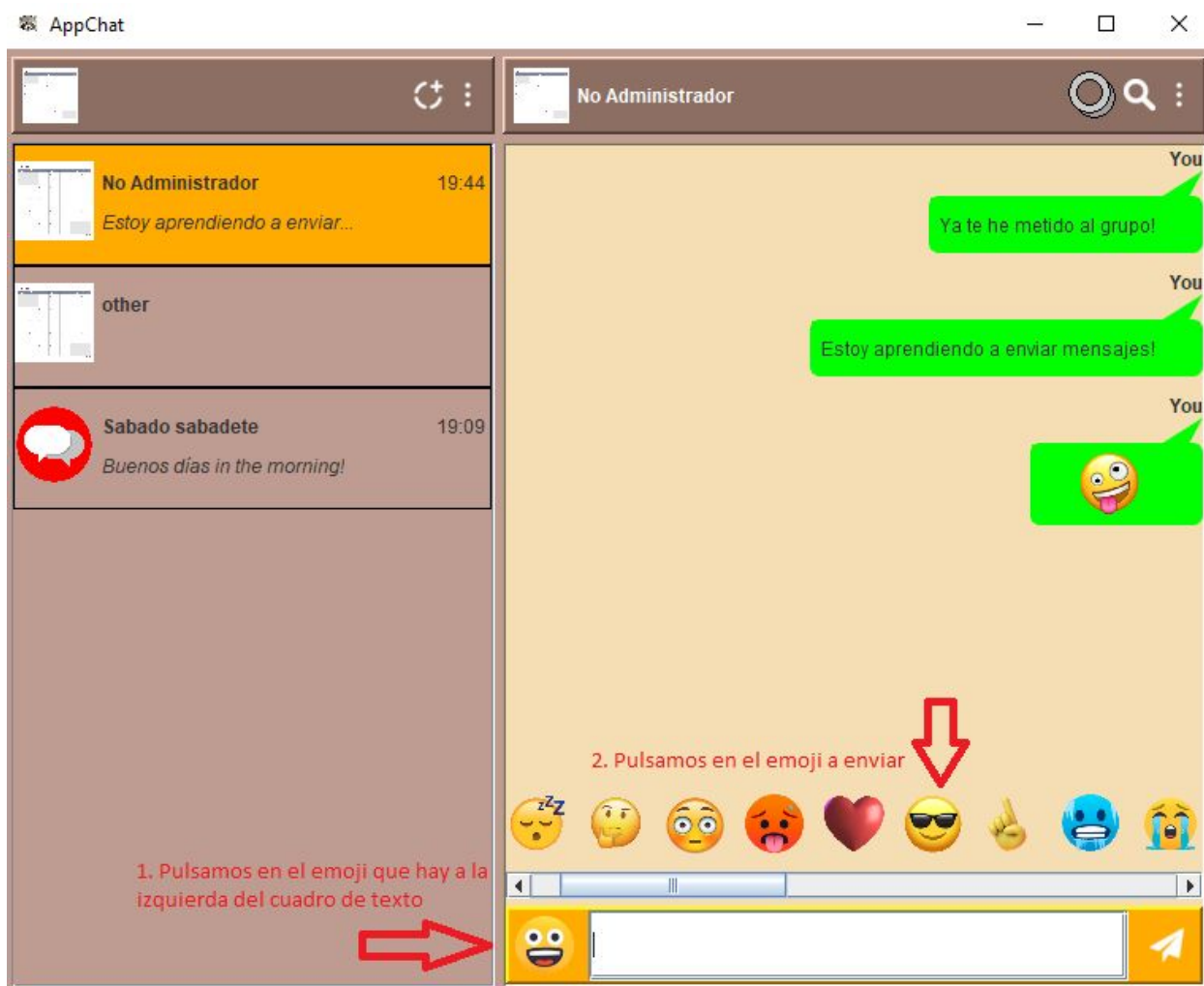
### III. Envío de mensajes

Una vez en la ventana principal, en la izquierda tendremos todos nuestros contactos y nuestros grupos, y a la derecha el chat del contacto seleccionado.

Para hablar con cualquiera de ellos seleccionamos por donde queremos enviar el mensaje, escribimos el mensaje y pulsamos en el botón de envío.

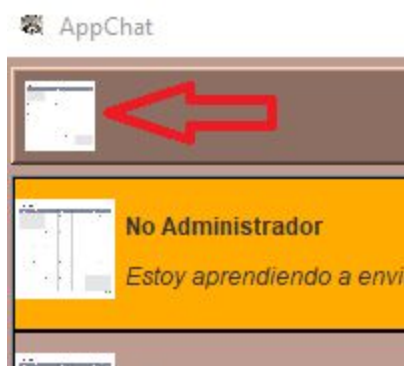


Si el mensaje a enviar es un emoji, realizamos click en el emoji para que nos aparezcan todos los emojis, elegimos el emoji a enviar y lo enviamos haciendo click en él. Una vez enviado si queremos ocultar el listado de emojis disponibles, pulsamos de nuevo el emoji que hay a la izquierda del cuadro de texto.



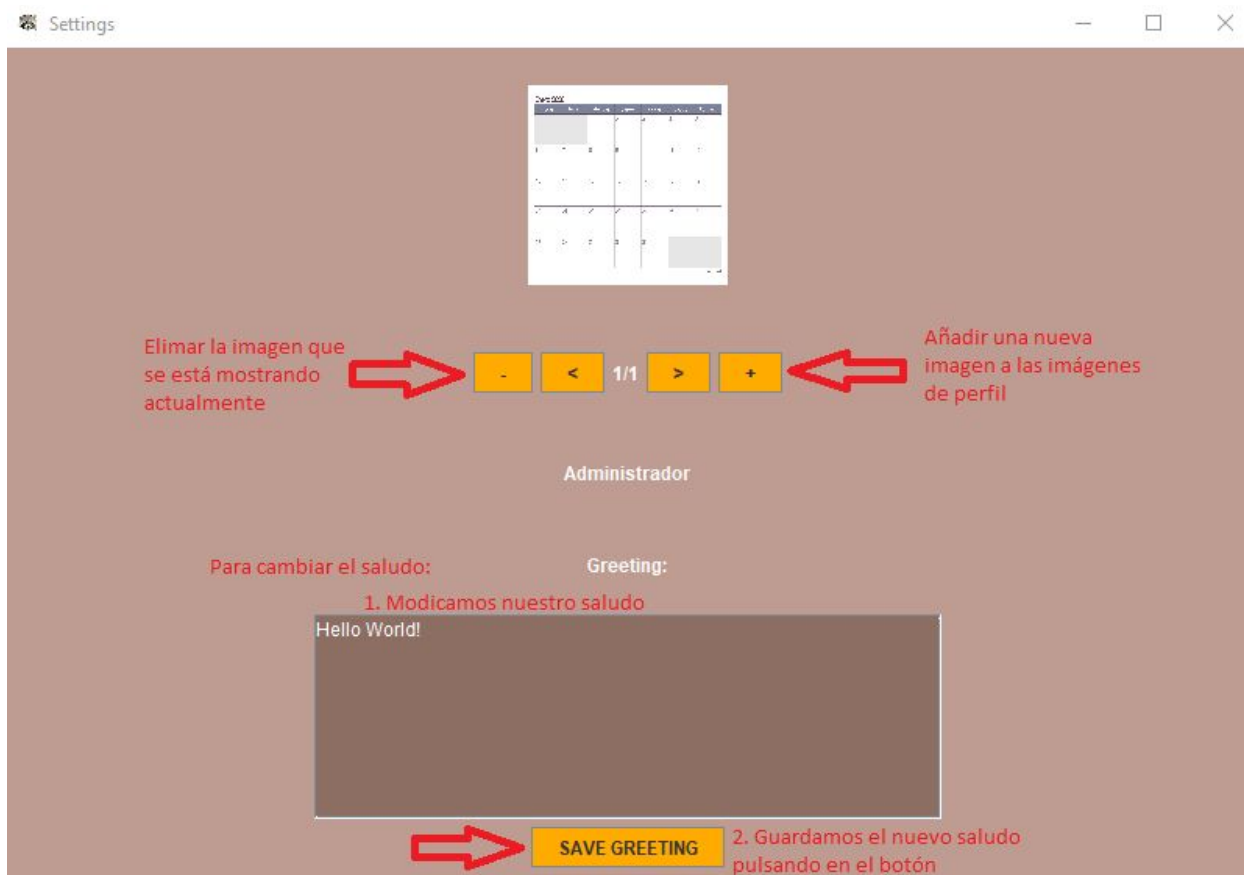
#### IV. Gestionar fotos de perfil y modificar nuestro saludo

Para gestionar nuestras fotos de perfil o modificar nuestro saludo, pulsamos en nuestra imagen de perfil.



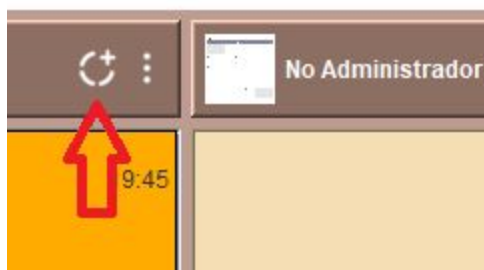
Se nos abrirá una nueva ventana en la cual podemos ver en la parte superior un carrusel con todas nuestras imágenes de perfil, pudiendo añadir o quitar imágenes.

Y en la parte inferior podremos cambiar nuestro saludo. Para ello, modificamos el saludo y pulsamos en el botón "SAVE GREETING".

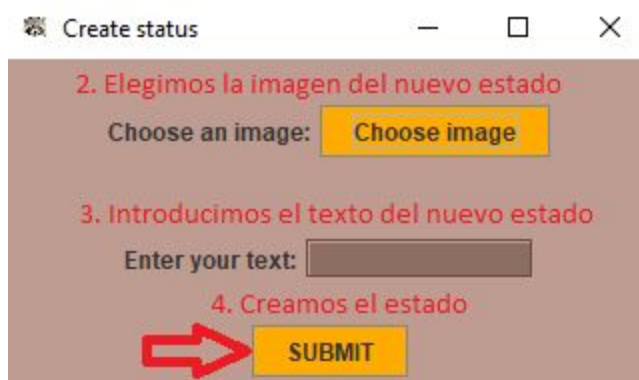
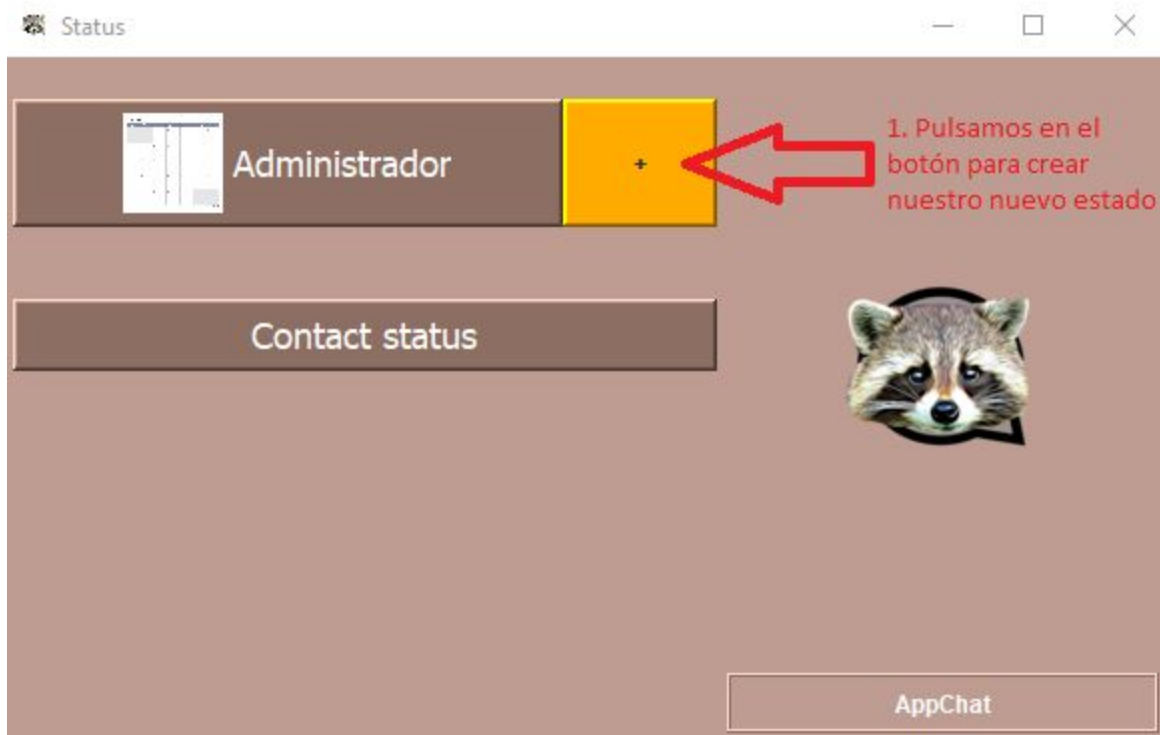


## V. Cambiar estado

Pulsamos el botón de estados para irnos a la ventana de estados.

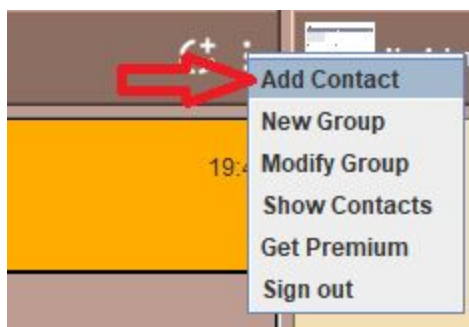


Una vez en la ventana de estados podemos ver nuestro estado y el estado de nuestros contactos. Para cambiar nuestro estado pulsamos en el botón e introducimos el texto e imagen de nuestro nuevo estado.



## VI. Crear contacto

Para crear un contacto pulsamos en las opciones del usuario y seleccionamos la opción "Add Contact".

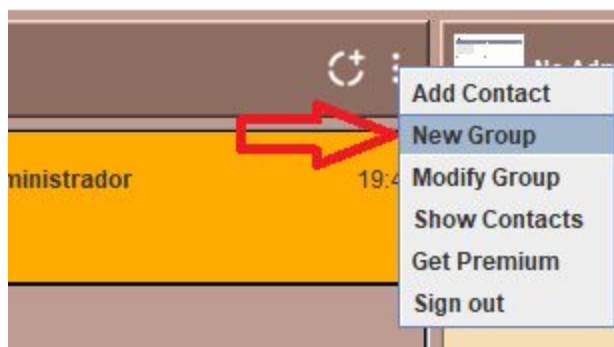


Una vez abro la ventana para añadir un nuevo contacto, solamente tendré que introducir sus datos y pulsar en el botón de añadir.

A screenshot of a window titled 'Add contact'. At the top is a raccoon avatar. Below it, red text reads: '1. Escribimos la información del contacto a añadir.' There are two input fields: 'Name' and 'Phone number'. At the bottom, a red arrow points to a yellow 'ADD' button. To the right of the button, red text reads: '2. Pulsamos en el botón para añadir el nuevo contacto'.

## VII. Crear grupo

Para crear un grupo pulsamos en las opciones del usuario y seleccionamos la opción "New Group".



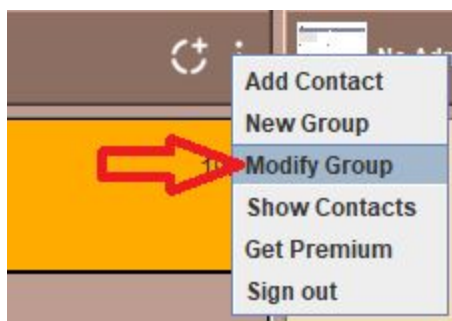
Una vez abro la ventana para añadir un nuevo grupo, indico el nombre del grupo, seleccionamos cuales de mis contactos quiero que sean participantes y pulso en el botón "Create Group".





## VIII. Modificar grupo

Para modificar un grupo pulsamos en las opciones del usuario y seleccionamos la opción "Modify Group".



Seleccionamos el grupo a modificar y pulsamos en el botón "Modify":

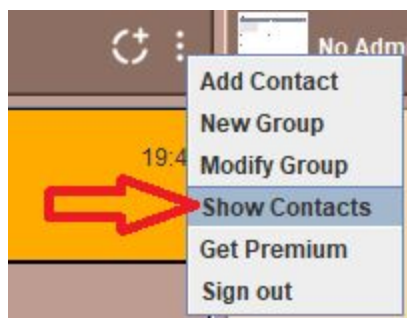


Modificamos el grupo y una vez modificado, pulsamos en el botón "MODIFY GROUP" para que se realice la modificación:

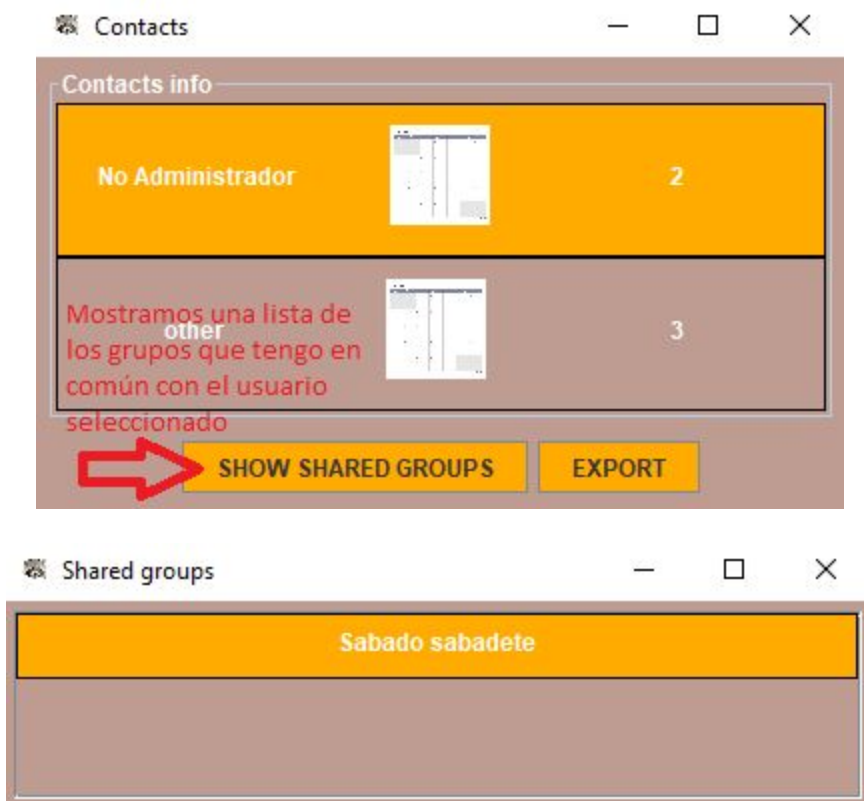


## IX. Mostrar contactos y exportarlos a PDF

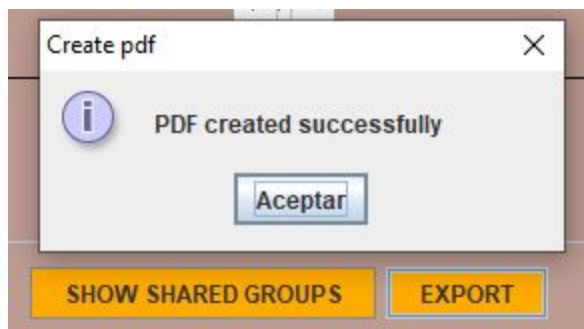
Para ver un listado de los contactos pulsamos en las opciones del usuario y seleccionamos la opción "Show Contacts".



Se nos mostrarán todos los contactos que tenemos y podemos ver los grupos que tienen en común pulsando en el botón "SHOW SHARED GROUPS".



O si queremos exportar a PDF toda la información de nuestros contactos, pulsaremos en el botón "EXPORT" (Se necesita ser premium para tener disponible esta funcionalidad).



Ejemplo del contenido del PDF:

Contact no. 1



- Contact name: No Administrador
- Phone: 2
- Groups shared:
  - + Group name: Sabado sabadete
  - + Group admin: Administrador
  - + Participants:
    - Name: No Administrador
    - Phone: 2
    - Name: other
    - Phone: 3

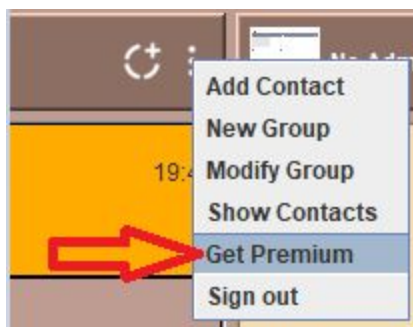
Contact no. 2



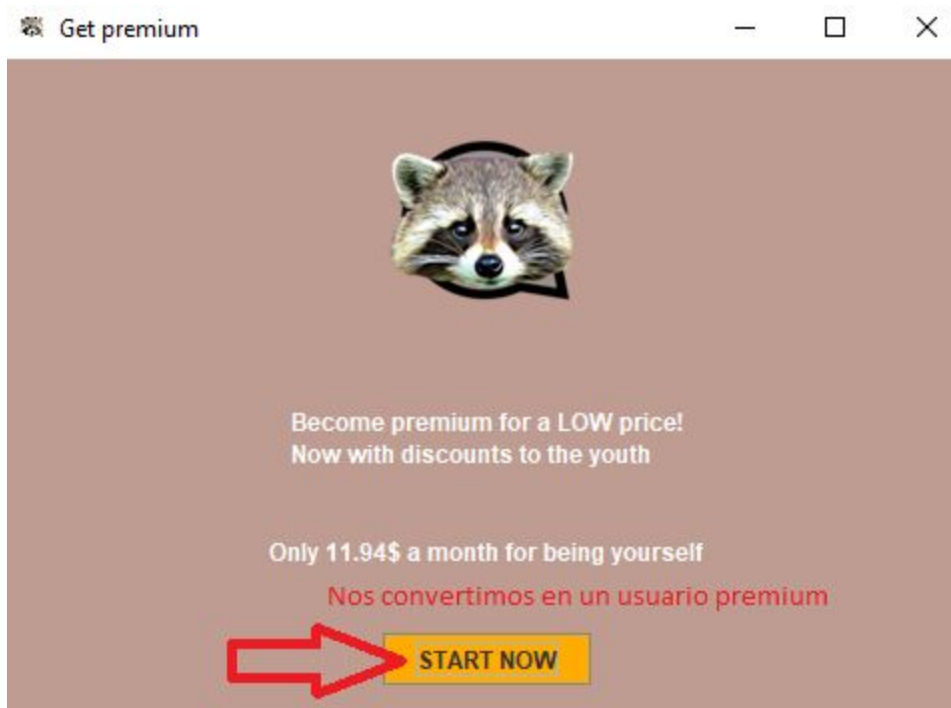
- Contact name: other
- Phone: 3
- Groups shared:
  - + Group name: Sabado sabadete
  - + Group admin: Administrador
  - + Participants:
    - Name: No Administrador
    - Phone: 2
    - Name: other
    - Phone: 3

## X. Convertirse en premium

Para convertirse en usuario premium pulsamos en las opciones del usuario y seleccionamos la opción "Get Premium".



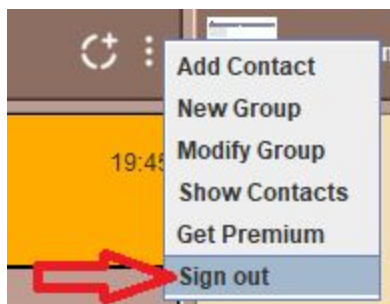
Y pulsamos el botón "START NOW" para aceptar el pago mensual y comenzar a disfrutar de las ventajas de ser un usuario premium.



Es necesario cerrar y abrir sesión de nuevo para poder disfrutar de toda la funcionalidad extra de ser premium.

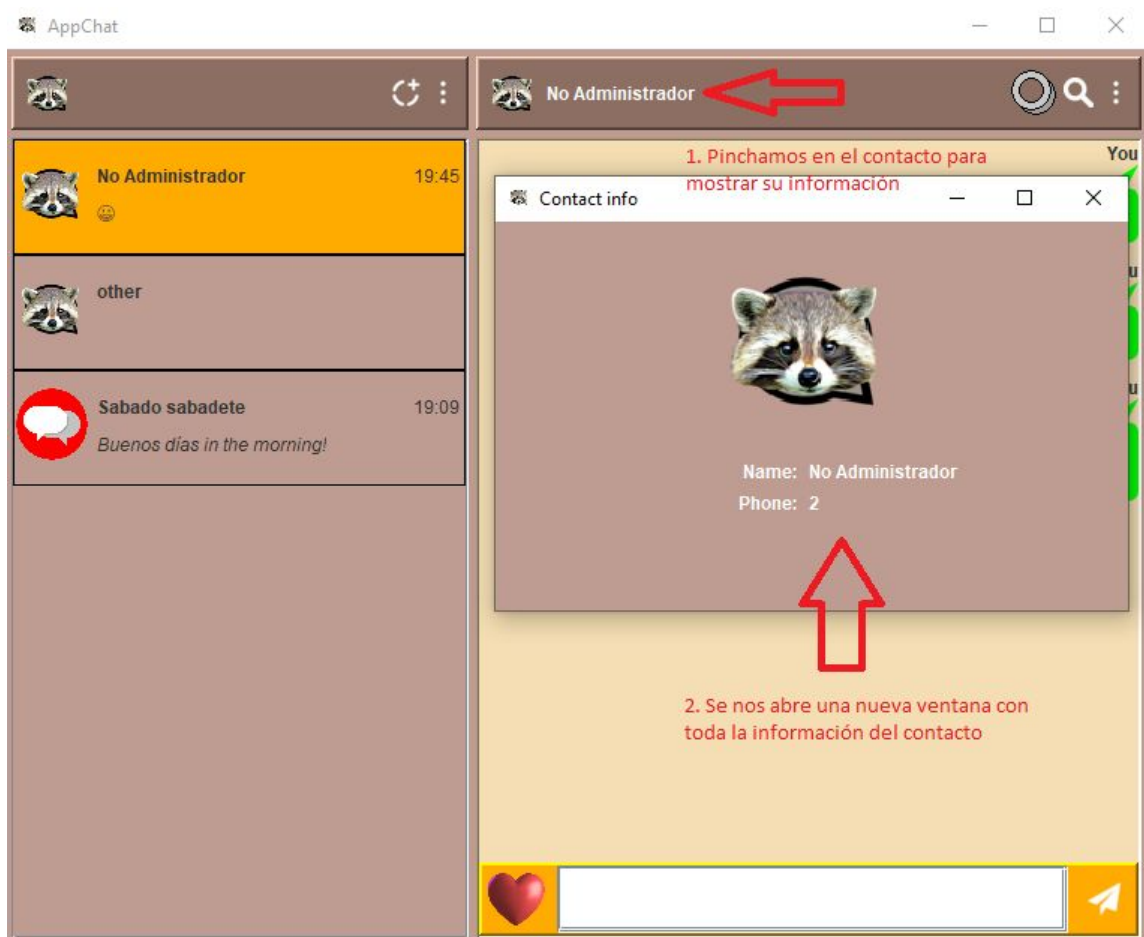
## XI. Cerrar sesión

Para cerrar sesión pulsamos en las opciones del usuario y seleccionamos la opción "Sign Out".



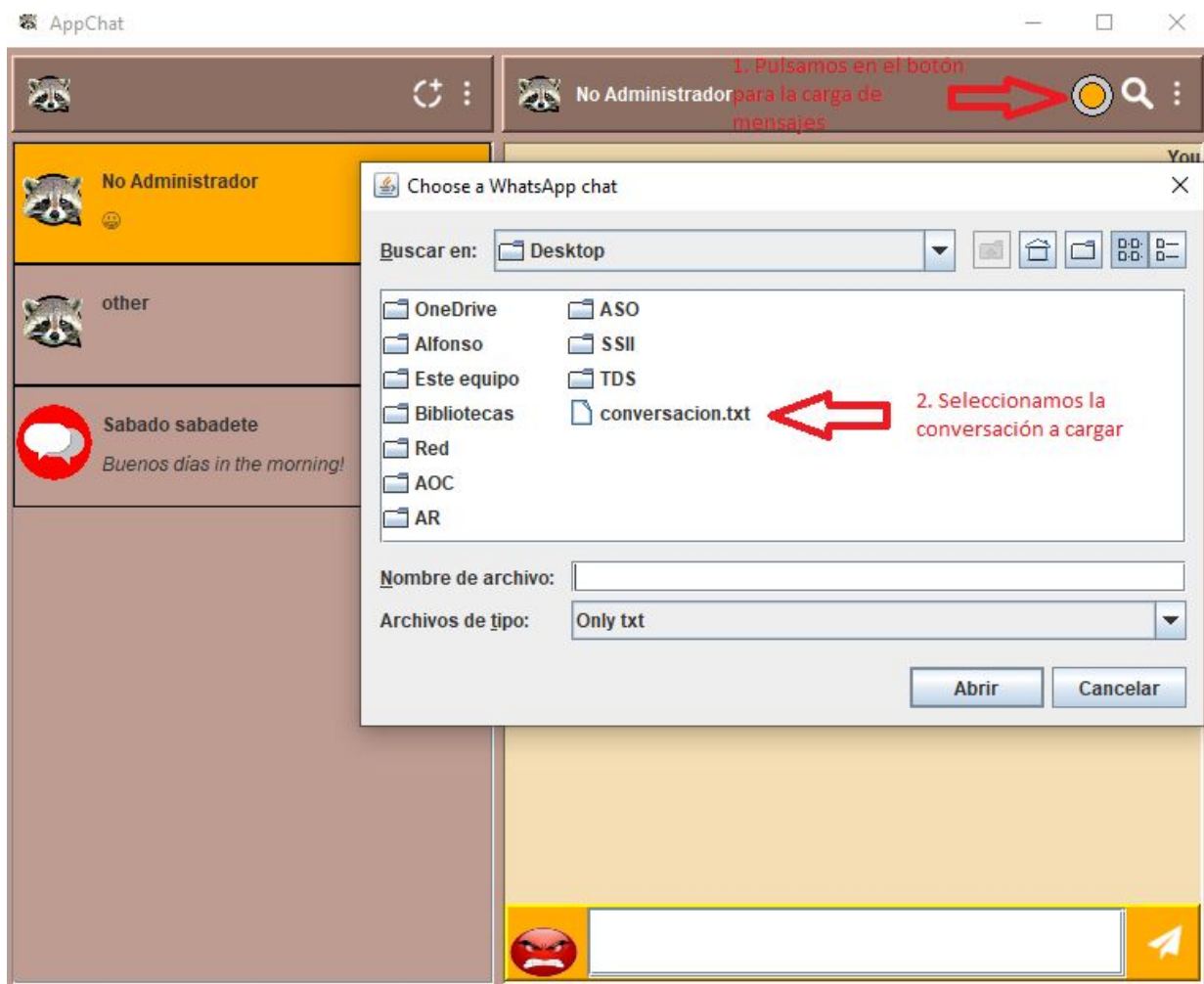
## XII. Ver información del contacto con el que estoy hablando

Pinchamos en el nombre del contacto y se nos mostrará una ventana en la que aparecerá su imagen, nombre y teléfono.



### XIII. Importar conversaciones

Pinchamos en el botón para importar conversaciones y elegimos el fichero con extensión .txt que contiene los mensajes de la conversación a cargar. Una vez seleccionado podremos ver en nuestro chat todos los nuevos mensajes importados.



#### XIV. Buscar mensajes

Pinchamos en el botón de búsqueda de mensajes y se nos abrirá una ventana para realizar la búsqueda.

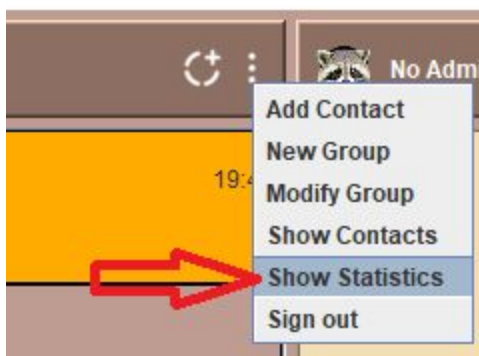


En la ventana de búsqueda de mensajes aplicaremos todos los filtros sobre los mensajes que queremos obtener y pulsaremos en el botón "SEARCH", mostrándonos así todos los mensajes que cumplan con los filtros indicados.



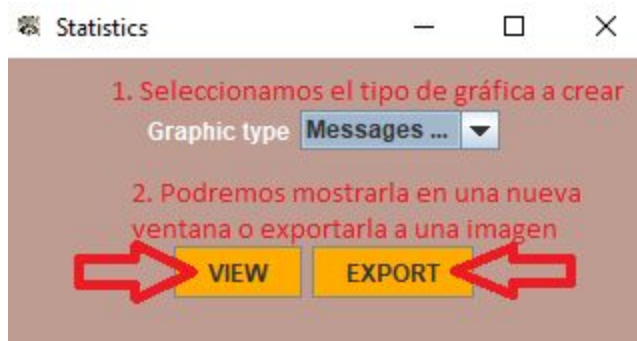
## XV. Mostrar estadísticas

Para ver las estadísticas pulsamos en las opciones del usuario y seleccionamos la opción "Show Statistics" (necesitamos ser premium para tener disponible esta opción).



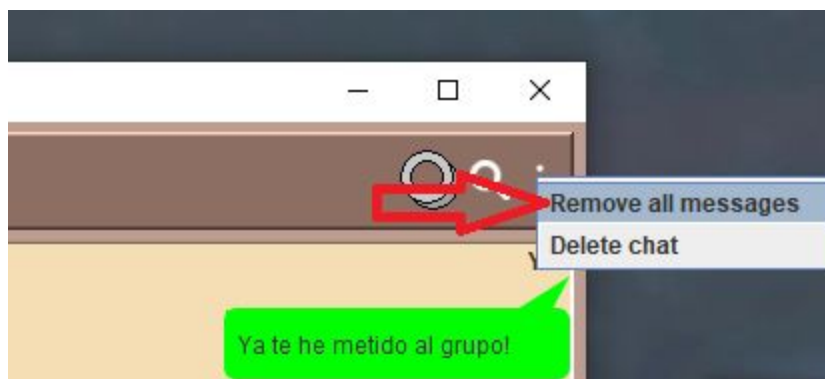


Seleccionaremos el tipo de gráfica que queremos crear, y podremos verla en una nueva ventana o exportarla a una imagen para poder almacenarla en nuestro ordenador.



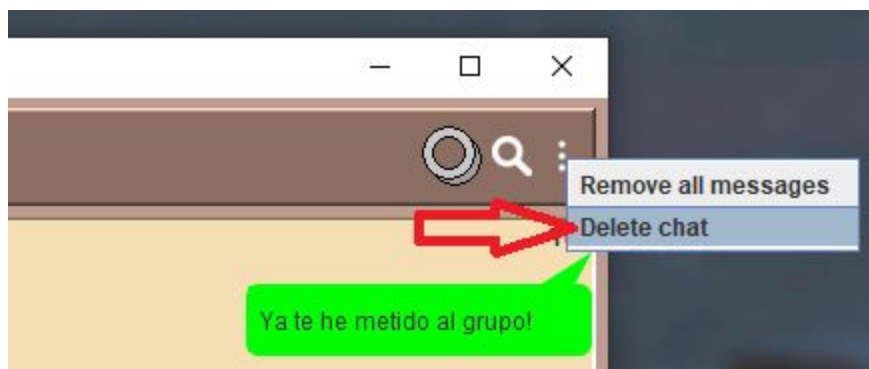
## XVI. Borrar todos los mensajes del chat seleccionado

Para borrar todos los mensajes del chat que tenemos abierto actualmente, pulsamos en las opciones del chat y seleccionamos la opción "Remove all messages".



## XVII. Borrar contacto

Para eliminar mi contacto, teniendo seleccionado al contacto a borrar, pulso en las opciones del chat y seleccionamos la opción "Delete chat".



## Observaciones

Esta práctica nos ha enseñado a afrontar un proyecto de envergadura, que perfectamente podría tratarse de una aplicación comercial. Pese a las numerosas horas dedicadas a la programación y al diseño, hemos disfrutado mucho creando esta aplicación. Empezó como un commit de un proyecto eclipse y a fecha de entrega lleva 234 commits.

mrblacknuel / AppChat Private

Unwatch 1 Unstar 1 Fork 0

Code Issues 1 Pull requests 0 Actions Security Insights Settings

Aplicación de chat creada para la asignatura TDS Edit

chat-application h2-database java-8 java-swing Manage topics

234 commits 20 branches 0 packages 0 releases 2 contributors

Branch: master New pull request Create new file Upload files Find file Clone or download

File	Commit Message	Time Ago
.settings	Minor bug fixed	2 months ago
lib	Añadido cargador	1 hour ago
src	Entrega de enero 2020	7 minutes ago
target	Entrega de enero 2020	7 minutes ago
.classpath	Añadido cargador	1 hour ago
.gitignore	Funcionando el guardado de imagenes	last month
.project	Inicialización del proyecto	3 months ago
pom.xml	Funcionando grafica con histograma. #12	28 days ago

Help people interested in this repository understand your project by adding a README. Add a README

Estos tres meses de desarrollo han estado caracterizados por el compromiso en la calidad del código y el perfeccionamiento de todos los detalles de AppChat. Les damos las gracias por plantear una práctica motivante a la par que extensa.

Un aspecto que nos gustaría comentar es que nos hubiese gustado que algunas sesiones de prácticas se hubiesen dedicado exclusivamente a la realización del proyecto y a la resolución de dudas acerca de este. Es algo que hemos echado en falta en ocasiones.

En cuanto a las horas empleadas, estimamos haber distribuido el tiempo de la siguiente forma:

	Ventanas (porcentaje)	Dominio (porcentaje)	Persistencia (porcentaje)	Servicios de PDF (porcentaje)	Servicios de gráficas (porcentaje)	Cargador de mensajes (porcentaje)	Refactoring y mejoras (porcentaje)	Horas estimadas
<b>Manuel González García</b>	30	15	20	0	10	10	15	<b>200</b>
<b>Alfonso López Baños</b>	30	15	20	10	0	10	9	<b>194</b>

Nos hemos dividido el trabajo de forma que ambos trabajamos todos los aspectos de la práctica siendo la única excepción los servicios generadores de PDF y la generación de gráficas con las estadísticas de usuario.