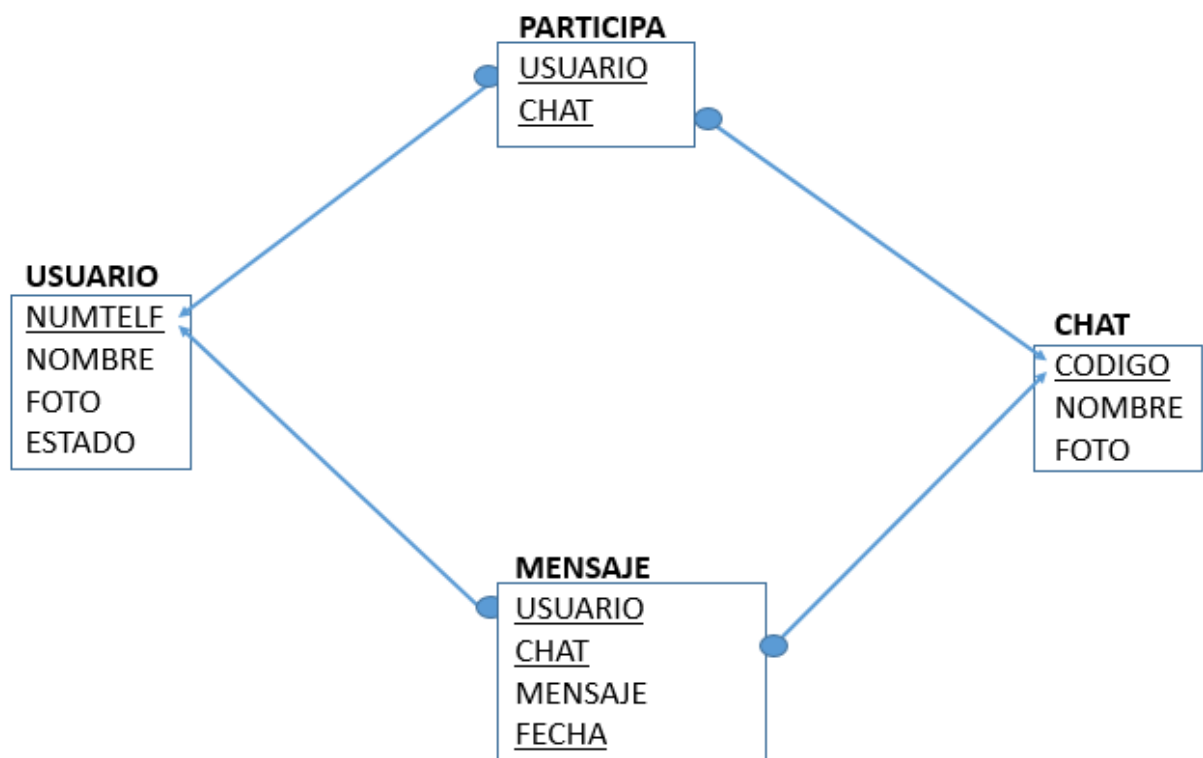


## Supuesto: “Mensajería instantánea”

### *Semántica INICIAL*

Formamos parte del equipo de desarrollo que está creando un sistema de mensajería instantánea dentro del ámbito de la UPNA. En concreto nos encargamos de crear la base de datos para soportar el sistema de mensajería instantánea.

Tras realizar el correspondiente análisis de requerimientos realizan el esquema Entidad/Relación y nos proporcionan ya el diseño de las tablas que deben almacenar los datos relacionados con el sistema de mensajería instantánea:



## Cuestiones

**1.** Crea las tablas descritas en el diseño que nos han proporcionado siguiendo las siguientes instrucciones:

- a. Tabla Usuarios, que debe almacenar los siguientes datos de los usuarios:
  - i. Numtelf: debe almacenar el número de teléfono, que serán números de 9 dígitos, de los usuarios que será la clave primaria.
  - ii. Nombre: debe almacenar el nombre de usuario que se mostrará en el chat, que serán de hasta 15 caracteres, puede estar vacío, ya que en ese caso se mostrará su número de teléfono.
  - iii. Foto: debe almacenar la foto del usuario que se mostrará en su contacto, que contendrán mapas de bits que guardaremos usando el tipo BLOB, puede estar vacío, ya que si no tiene foto no se mostrará.
  - iv. Estado: debe almacenar el estado del usuario que se mostrará junto a su contacto, que puede ser de hasta 30 caracteres, y por defecto sino se inserta ningún valor cogerá el valor 'Disponible'

*Table USUARIO creado.*

- b. Tabla Chat, que debe almacenar los siguientes datos de los chats:
  - i. Código: debe almacenar el código del chat, que serán números de hasta 12 dígitos, que será la clave primaria de los chats.
  - ii. Nombre: debe almacenar el nombre del chat que se mostrará en el chat, que serán de hasta 30 caracteres, NO puede estar vacío.
  - iii. Foto: debe almacenar la foto del chat que se mostrará, que contendrán mapas de bits que guardaremos usando el tipo BLOB, puede estar vacío, ya que si no tiene foto no se mostrará.

*Table CHAT creado.*

- c. Tabla Participa, que almacena la relación entre los usuarios y los chats:
  - i. Usuario: debe almacenar el número de teléfono de los usuarios que participan en el chat, que será del mismo tipo que el número de teléfono de la tabla usuarios, debemos asegurar que los números de teléfonos insertados correspondan a números de teléfonos que existan en la tabla usuarios.
  - ii. Chat: debe almacenar el código de chat de los usuarios que participan en el chat, que será del mismo tipo que el código de chat

de la tabla chat, debemos asegurar que los códigos de chat insertados correspondan a códigos de chat que existan en la tabla chat.

- iii. La clave principal de esta tabla la deben formar las combinaciones de usuario y chat.

*Table PARTICIPA creado.*

- d. Tabla Mensaje, que almacena todos los mensajes de los usuarios en los chats:

- i. Usuario: debe almacenar el número de teléfono de los usuarios que participan en el chat, que será del mismo tipo que el número de teléfono de la tabla usuarios, debemos asegurar que los números de teléfonos insertados correspondan a números de teléfonos que existan en la tabla usuarios.
- ii. Chat: debe almacenar el código de chat de los usuarios que participan en el chat, que será del mismo tipo que el código de chat de la tabla chat, debemos asegurar que los códigos de chat insertados correspondan a códigos de chat que existan en la tabla chat.
- iii. Mensaje: debe almacenar el mensaje que el usuario escribe en el chat, el cual se mostrará, que puede contener mucha información y que guardaremos usando el tipo CLOB, NO puede estar vacío.
- iv. Fecha: debe almacenar la fecha en la que queda guardado el mensaje en la base de datos, que será de tipo date y por defecto sino se inserta valor cogerá el valor de la fecha del sistema SYSDATE (sin comillas).
- v. La clave principal de esta tabla la deben formar las combinaciones de usuario, chat y fecha (suponemos que el tipo date incluye fecha hora hasta milisegundos y que no será posible que un mismo usuario, en el mismo chat pueda guardar un mensaje en el mismo momento y que así queda unívocamente identificado).

*Table MENSAJE creado.*

2. Tras realizar las primeras pruebas de inserción de datos en la base de datos recién creada por los equipos de pruebas e integración, nos indican que tenemos que hacer las siguientes modificaciones:

- a. Han podido introducir números de teléfonos de menos de 9 dígitos, nos indican que probemos a insertar el siguiente usuario:

i. Numtelf: 55512312

NUMTELF	NOMBRE	FOTO	ESTADO
55512312			Disponible

- b. Borrar el usuario con el numtelf introducido

NUMTELF	NOMBRE	FOTO	ESTADO

- c. Incluir una comprobación en la creación de tabla de usuario para que no se puedan insertar números de teléfonos que no sean de 9 dígitos. Suponemos que los números de teléfono no pueden empezar por 0. (Para ello tendremos que eliminar la tabla usuarios y crearla de nuevo)

*Table USUARIO borrado.*

*Table USUARIO creado.*

- d. Realizar las siguientes comprobaciones:

- i. Que ya no es posible introducir el usuario

1. Numtelf: 55512312

*Informe de error -*

*ORA-02290: check constraint (xxx.CK\_USUARIO\_NUMTELF) violated*

- ii. Que sí es posible introducir el usuario

1. Numtelf: 555123123

NUMTELF	NOMBRE	FOTO	ESTADO
555123123			Disponible

**3.** Tras las segundas pruebas de inserción de datos detectan que el sistema permite al mismo estudiante registrarse más de una vez con números de teléfonos diferentes y nos piden, que, aunque podrían controlarlo desde la parte de la app, esto se impida ya desde base de datos.

- a. Incluir NIA (número de identificación del alumno): añadir el campo nia a la tabla usuario, que serán números de 6 dígitos, que NO puede estar vacío y que nos debemos asegurar que sólo se introducirán valores únicos de nias. (Para ello tendremos que eliminar la tabla usuarios y crearla de nuevo)

*Table USUARIO borrado.*

*Table USUARIO creado.*

- b. Realizar las siguientes comprobaciones

- i. Insertar estos 2 usuarios

1. Numtelf: 555123456, nia:1 y Estado: 'En clase'
2. Numtelf: 555111222, nia:2 y Estado: 'Ocupado'

NUMTELF	NIA	NOMBRE	FOTO	ESTADO
555123456	1			En clase
555111222	2			Ocupado

- ii. Comprueba que NO es posible insertar el usuario 1 de nuevo con otro numtelf:

1. Numtelf: 555333444, nia:1 y Estado: 'En clase'

*Informe de error -*

*ORA-00001: unique constraint (xxx.UQ\_USUARIO\_NIA) violated*

**4.** Una vez realizados los cambios solicitados nos piden que insertemos el siguiente juego de datos de prueba para poblar un poco las bases de datos y validar la base de datos:

- a. Si hemos utilizado CASCADE CONSTRAINTS en la sentencia SQL de DROP de la tabla Usuario, habremos eliminado en cascada las restricciones de foreign key en las tablas afectadas (Participa y Mensaje). Recrea de nuevo todas las tablas de la Base de datos.

*Table CHAT borrado.*

*Table PARTICIPA borrado.*

Table MENSAJE borrado.

Table USUARIO borrado.

Table USUARIO creado.

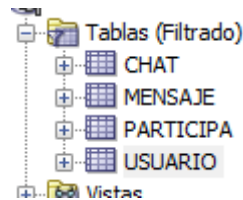
Table CHAT creado.

Table PARTICIPA creado.

Table MENSAJE creado.

b. Tabla usuario: Insertamos los siguientes usuarios

- i. Numtelf: 555123123, nia:1;
  - ii. Numtelf: 555111222, nia:2,nombre:'Irati',estado: 'Ocupado'
  - iii. Numtelf: 555666777, nia:3,nombre:'Martin',estado: 'En clase'
1. Insertamos manualmente una foto a Martin, una vez insertado el usuario Martin, desde el visor de tablas abrimos la tabla usuario



2. Y desde la pestaña datos editamos el campo de la foto y le damos cargar para poder cargar un fichero

1

2

Editar Valor

Información

☐ Datos Guardados [Editor Externo](#) [Descargar](#) [Ver co](#)

Longitud

Vació

Temporal

Abrir

☐ Datos Locales [Cargar](#) [Definir NULL](#) [Ver como:](#) ☐

Nombre de Archivo

Tamaño del Archivo

3. Una vez cargado veremos que ha quedado el campo BLOB con bytes y le daremos a confirmar (commit)

Columnas Datos Model Restricciones Permisos Estadísticas Dispara

Ordenar... Filtrar:

	NUMTELF	NIA	NOMBRE	FOTO	ESTADO
1	555123123	1	(null)	(null)	Disponible
2	555111222	2	Irati	(null)	Ocupado
*3	555666777	3	Martin	(BLOB)	En clase

NUMTELF	NIA	NOMBRE	FOTO	ESTADO
555123123	1			Disponible
555111222	2	Irati		Ocupado
555666777	3	Martin	FFD8FFE16AEF45....	En clase

- c. Tabla chat: Insertamos los siguientes chats
- Código:1, nombre: 'Prueba chat directo'
  - Código:2, nombre: 'Prueba chat de grupo'

1. Insertamos manualmente una foto en el chat de grupo siguiendo el procedimiento para introducir una foto al usuario Martin.

<i>CODIGO</i>	<i>NOMBRE</i>	<i>FOTO</i>
-----		
1	Prueba chat directo	
2	Prueba chat de grupo	966000049492A0...

- d. Tabla participa: Insertamos las siguientes pertenencias a chats:

- i. Usuario: 555123123, chat: 2
- ii. Usuario: 555111222, chat: 1
- iii. Usuario: 555111222, chat: 2
- iv. Usuario: 555666777, chat: 1
- v. Usuario: 555666777, chat: 2

<i>USUARIO</i>	<i>CHAT</i>
-----	
555123123	2
555111222	1
555111222	2
555666777	1
555666777	2

- e. Tabla mensaje: Insertamos los siguientes mensajes:

- i. Usuario: 555111222, chat:1, mensaje: 'Hola chat directo soy Irati'
- ii. Usuario: 555666777, chat:1, mensaje: 'Hola chat directo soy Martin'
- iii. Usuario: 555123123, chat:2, mensaje: 'Hola chat grupo no tengo nombre'
- iv. Usuario: 555111222, chat:2, mensaje: 'Hola chat grupo soy Irati'
- v. Usuario: 555666777, chat:2, mensaje: 'Hola chat grupo soy Martin'

<i>USUARIO</i>	<i>CHAT MENSAJE</i>	<i>FECHA</i>
-----		
555111222	1 Hola chat directo soy Irati	17/11/17
555666777	1 Hola chat directo soy Martin	17/11/17
555123123	2 Hola chat grupo no tengo nombre	17/11/17
555111222	2 Hola chat grupo soy Irati	17/11/17
555666777	2 Hola chat grupo soy Martin	17/11/17

5. Finalmente, para dejar la base de datos limpia para que el equipo de pruebas pueda continuar insertando y comprobando el funcionamiento debemos de borrar todos los registros del juego de datos de pruebas insertados. Para no tener



problemas borrando los registros debemos comenzar borrando los registros de las tablas que contienen las claves foráneas y en último lugar las que contienen las claves primarias.