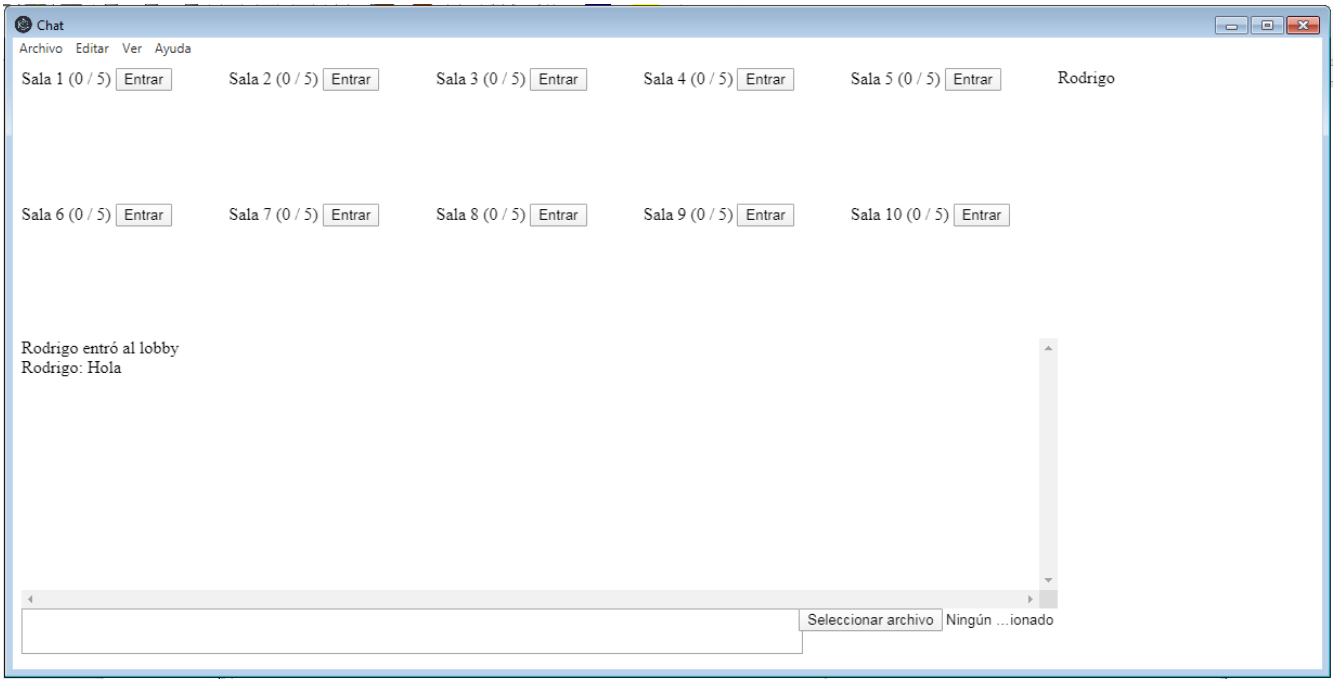
1151076 - Programación Visual Orientada a Eventos **2019-O | CSI01**

Tercera evaluación - especificación del proyecto

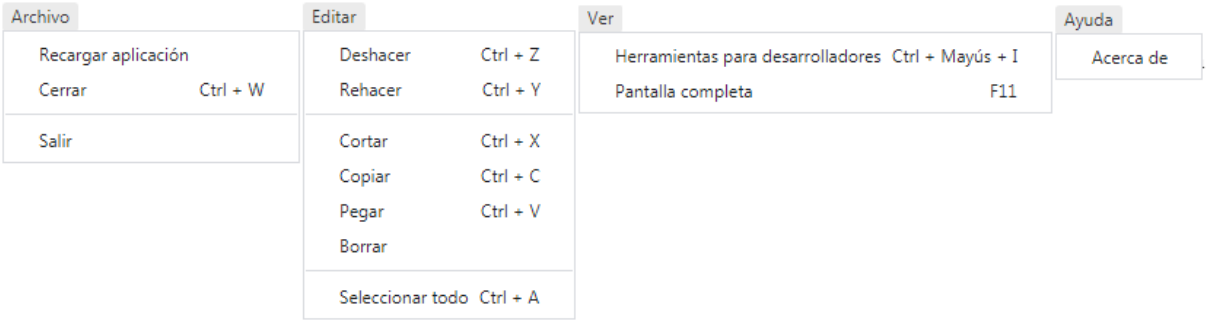
El proyecto consistirá en la implementación del cliente de un chat compatible con el servidor implementado en Node JS y disponible en https://sites.google.com/site/rccuam/home/1151076---programacion-visual-orientada-a-eventos/_servidor.zip. El cliente deberá estar implementado en Electron y deberá empaquetarse para generar la carpeta de aplicación y un ejecutable de Windows usando Electron Packager.

A continuación se muestran capturas de pantalla de una implementación *funcional* del cliente. Se implementará la mayor parte del mismo durante clase, pero el 80% de la evaluación dependerá de que el alumno termine la implementación. El 20% restante dependerá de los estilos adicionales que el alumno le aplique al cliente. Los estilos no necesitan ser vistosos, pero deben darle un aspecto más profesional al cliente.

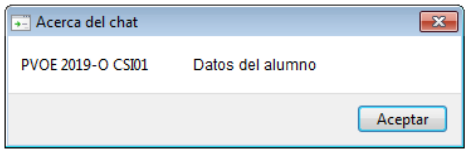
Ventana de la aplicación



La ventana debe contar con los controles usuales de una ventana de Windows. La barra de menú debe tener las siguientes opciones con el comportamiento habitual. Algunas opciones se detallan a continuación.



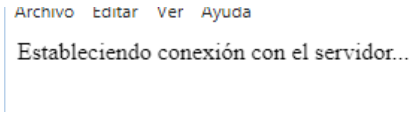
En el menú "Archivo", la opción "Recargar la aplicación" deberá cerrar cualquier subventana y recargar la pantalla inicial en la ventana principal; la opción "Cerrar" deberá cerrar la ventana desde donde se lanza el comando; la opción "Salir" deberá cerrar la aplicación completa. En el menú "Ver", la opción "Herramientas para desarrolladores" es equivalente a la opción del navegador Chrome. En el menú "Ayuda", la opción "Acerca de" deberá desplegar un cuadro de diálogo similar al siguiente:



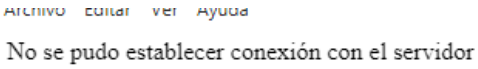
Pantalla inicial y de ingreso

La primera acción del cliente deberá ser intentarse conectar con el servidor. Dentro de la UAM, el servicio está disponible en ws://callix.azc.uam.mx:8000. Se recomienda que el alumno realice pruebas con un servidor local.

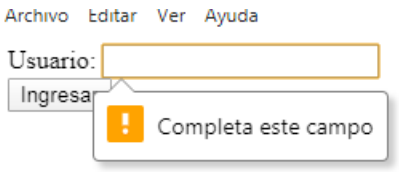
Mientras la conexión no se establezca, deberá mostrarse un letrero similar a éste:



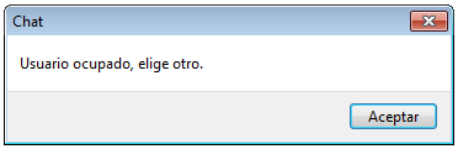
Si la conexión no logra establecerse, deberá informársele al usuario de esta situación.



Si la conexión logra establecerse, deberá mostrársele al usuario un formulario donde podrá elegir un nombre de usuario para el chat. El nombre de usuario no puede estar vacío tampoco contener espacios, debiendo validarse.

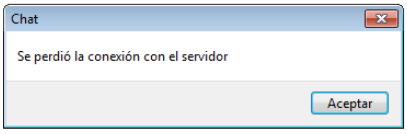


Si el nombre de usuario es válido pero el servidor reporta que ya está ocupado, se debe notificar al usuario.



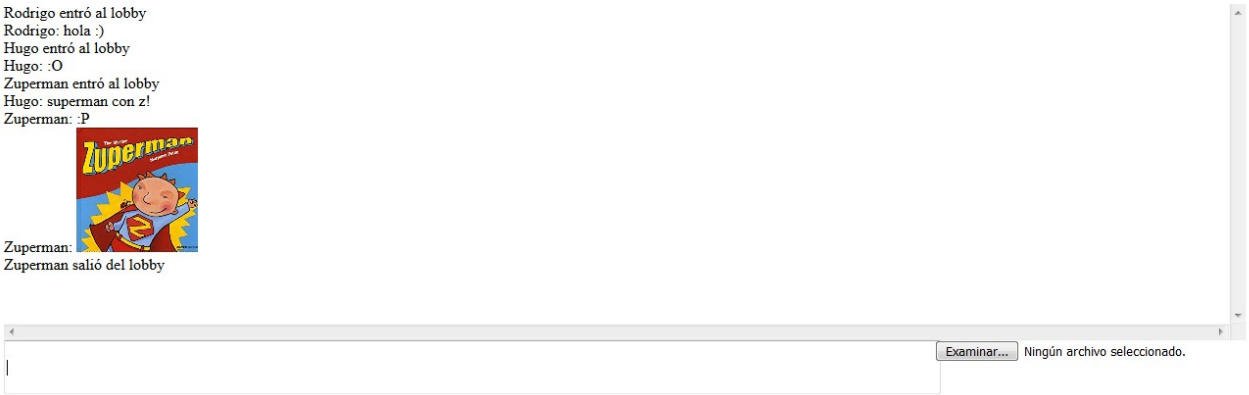
Pérdida de conexión

En caso de que la conexión con el servidor se interrumpa después de haberla establecido, se deberá notificar al usuario y recargar la aplicación.



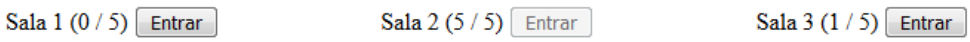
Pantalla de lobby

El chat consta de un *lobby* y de diez salas con capacidad limitada. Los usuarios conectados podrán enviar mensajes al lobby, los cuales deberán ser visibles para otros usuarios conectados. También se deberá mostrar la lista de usuarios conectados (actualizada y en orden alfabético) además de avisos de entrada y salida de usuarios (no es obligatorio mostrar los mensajes de entrada y salida del propio usuario).

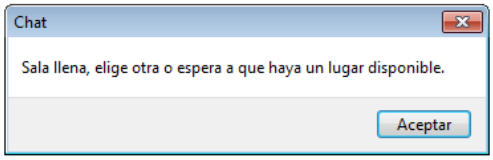


Un mensaje escrito en el cuadro de texto inferior será enviado después de dar *enter*. Si una imagen fue seleccionada en el selector de archivos cuando esto ocurra, también será enviada. El campo de texto y el selector de archivos deberán reiniciarse tras un envío. El cliente no deberá enviar al servidor mensajes de texto vacíos (aunque puede enviar una imagen sin mensaje y puede enviar mensajes que consistan de espacios). El alumno es libre de elegir el formato con el que se envía una imagen, pero no puede modificar la funcionalidad del servidor para implementar dicha característica. Si la implementación limita de alguna forma el envío de mensajes de texto, el alumno deberá documentarlo.

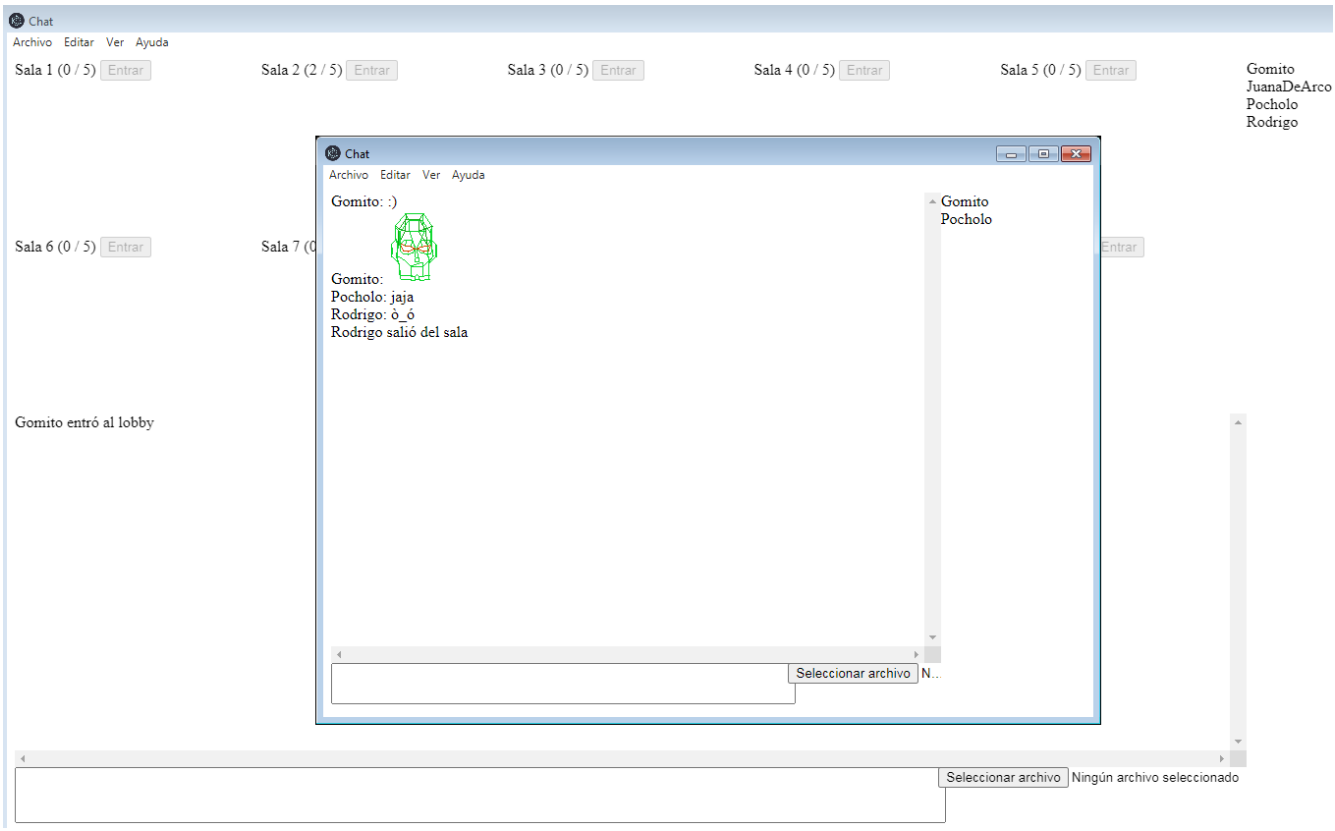
Un usuario podrá seleccionar alguna de las salas disponibles, siempre y cuando la sala no esté llena y el usuario no esté ya en una sala. Los botones de ingreso deberán habilitarse y deshabilitarse según la situación.



Al seleccionar una sala disponible y si la petición de ingreso es aceptada por el servidor, se abrirá otra ventana. En caso contrario (por ejemplo, cuando la petición se hace mientras la sala está disponible pero el servidor la rechaza porque alguien entró primero) se deberá enviar una notificación al usuario.



La subventana de sala



La subventana de sala tiene funcionalidad similar a la ventana principal del lobby, excepto que el contenido del chat ocupa la mayor parte de la ventana al no existir subsalas dentro de una sala. La lista de usuarios de una sala sólo debe mostrar a los usuarios que están en dicha sala. Al cerrar la ventana de sala, el cliente debe solicitar salir de ésta al servidor. Si el chat recarga por alguna razón (se pierde la conexión o el usuario solicita la recarga desde la barra de menú), cualquier subventana debe cerrarse.

El alumno es libre de añadir funcionalidad al cliente que no está señalada en este documento. Aunque esto no es necesario para obtener la máxima calificación de la tercera evaluación, se tomará en cuenta.