

JUEGO DE PREGUNTAS EN MULTIJUGADOR

CONTENIDO

Motor	2
Estructura del proyecto	2
Server central:	2
.....	2
Cliente del server central:	2
Descripción de Clases	2
Info	2
Assets	2
Lobby.....	3
Lobbyclient y ServerMain.....	3
Juego	3
GameServer	3
ClientServer	4
Vistas	4
Inicio.....	4
Main	5
Game.....	5
TableroView.....	6
PanelJugadores.....	6
PodioView	6

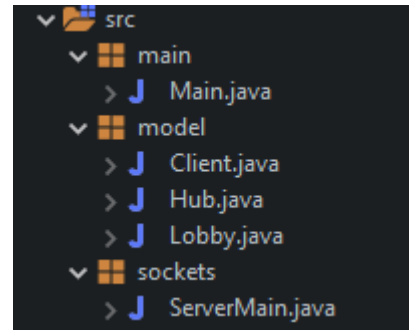
MOTOR

El juego estará creado en java, se utilizará sockets para la conexión con otros clientes y WindowBuilder para los gráficos y los menús

ESTRUCTURA DEL PROYECTO

SERVER CENTRAL:

Un proyecto de un server central con los usuarios que están compartiendo su partida, para que otros usuarios puedan acceder a ellas, esto será un server tcp, ya que no es importante la latencia

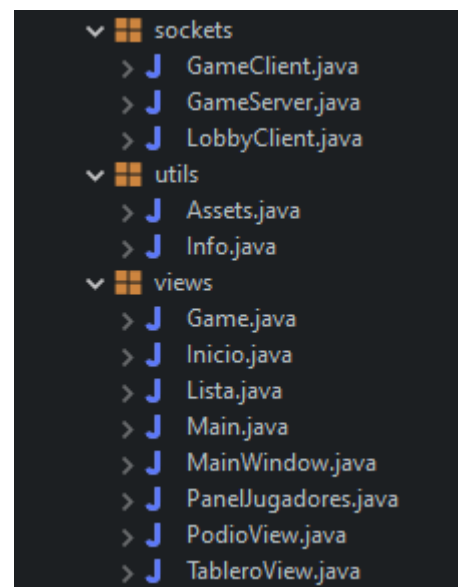


CLIENTE DEL SERVER CENTRAL:

Un cliente con un socket tcp para poder crear, unirse y eliminar salas interactuando con el servidor central

Un Cliente con un socket udp para poder conectarse a las salas de otros usuarios, el cual se encarga de recibir información y dibujarla en la pantalla

Un Servidor de socket udp que se creará al mismo tiempo que una sala en el servidor central, y que permitirá a los usuarios conectarse a este servidor, el cual será el encargado de actualizar el juego



DESCRIPCIÓN DE CLASES

INFO

Es una clase que tiene métodos estáticos que compartirán las diferentes vistas

ASSETS

Es una clase que se inicia al principio del programa y carga en memoria todos los sprites que se van a usar

LOBBY

El servidor central tiene una lista de lobbys donde están los lobbys abiertos, con los clientes que tiene cada uno

LOBBYCLIENT Y SERVERMAIN

Crear (/create)

El cliente creará un server udp y enviará al servidor "create", el servidor recibirá el mensaje, añadirá a su arraylist de host, el nuevo servidor su ip

Entrar (/join)

El cliente enviara al servidor central cual es el lobby que quiere entrar mediante su dirección, el servidor añadirá a ese Lobby al jugador y responderá con un Lobby con la dirección, y el cliente creara un cliente udp para conectarse el servidor udp del lobby mediante el address

List (/list)

El cliente envía al servidor que quiere una lista de todos los lobby y el servidor devolverá un hub el cual tiene una lista de todos los jugadores

Abandonar (/close)

EL cliente enviara al servidor central que quiere abandonar el Lobby y el server central intentara eliminar del Lobby al cliente

Cerrar (/close)

EL cliente enviara al servidor central que quiere cerrar el Lobby y el server central intentara cerrar el lobby, enviando al usuario maestro que lo cierra, para que este por el server udp, envíe a todos los usuarios que cierren el Lobby

JUEGO

El servidor del usuario ira a 60 tps, que es la velocidad de update del servidor

GAMESERVER

Tiene una lista con todos los usuarios conectados a él, tiene una clase Game con un GameLoop que actualiza el juego 60 veces por segundo, sus protocolos son:

Cuando recibe mensajes:

con "/chat {msg}": lo envía a todos los clientes

con "/join {jugador}": añade ese usuario a la lista de jugadores

con "/close {jugador}": elimina a ese jugador de la sala

StartGame:

Inicia la clase Game que es la que da comienzo al juego

Close:

Envía a todos los usuarios “/close “, lo cual hace que salgan todos de su sala

CLIENTSERVER

Tiene la ip del host udp para enviarle mensajes

Cuando recibe mensajes:

con “/chat {msg}”: lo guarda en el textarea Chat

con “/close “: cierra el cliente y vuelve al hub de salas

con “/game {tablero} “: actualiza las vistas con la información traída

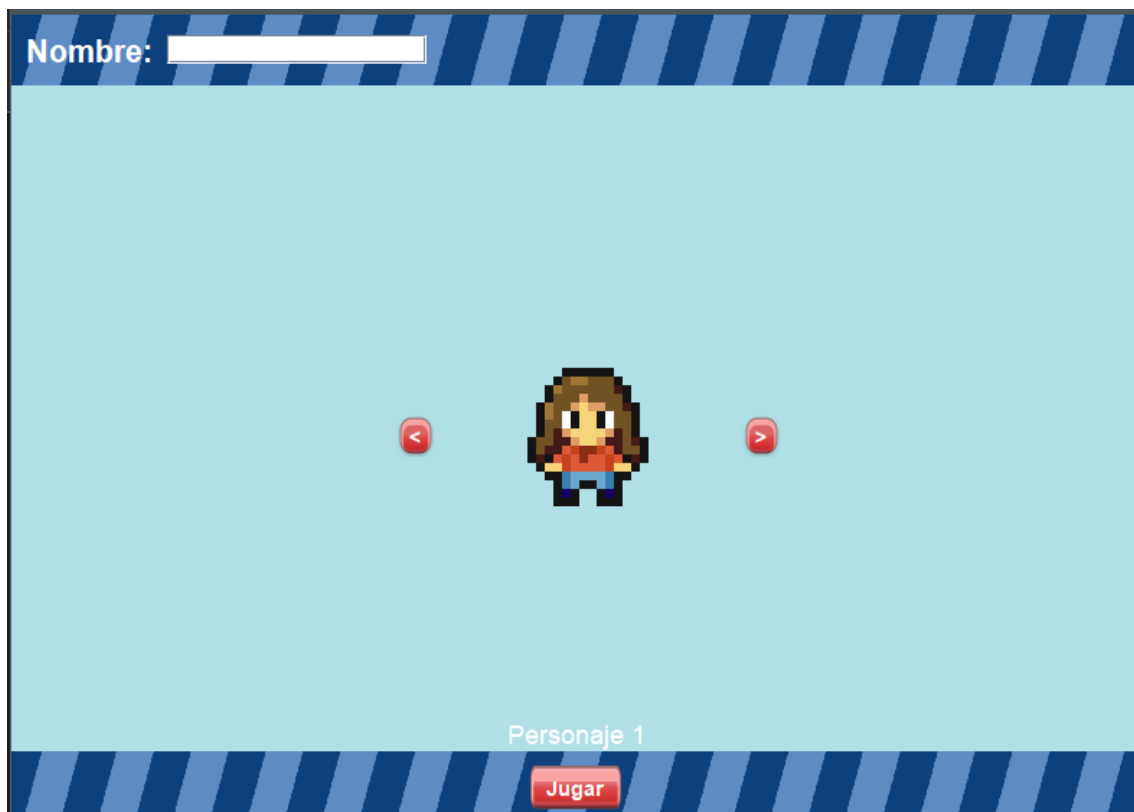
VISTAS

INICIO

Tiene un Jtextfield que guardara en la variable Info.Nombre

Un selector de jugador donde ira cambiando el sprite dependiendo de las flechas de los lados, y que luego se guardara en la variable Info.Sprite

Un botón que te llevara a la vista Main



MAIN

Tiene una tabla donde se muestran objetos de clase Lobby y al clicar en una de las filas, hace un /join y pasa a la vista Game

Dos botones uno llamado refrescar y host:

refrescar pide una /list, y actualiza la información en la tabla

host hace un /create y pasa a la vista Game

host	Jugadores
192.168.1.20	1

Refrescar

Host

GAME

Tiene un panel que representa una carrera en la izquierda, donde se encuentran las posiciones de los jugadores

Un tablero central donde se encuentra la pregunta y las respuestas, al pulsar una de las respuestas, esta y las demás se bloquean, mostrando si has fallado, y señalando la correcta

En la izquierda hay un chat donde enviar mensajes

Y en el sur, un panel con un botón para abandonar la sala, o si eres creador de esta cerrarla, además con un botón para empezar una partida

Si uno de los jugadores llega al final, se terminara la partida y donde antes había un tablero, ahora estará la vista Podio mostrando a los jugadores que han ganado

TABLEROVIEW

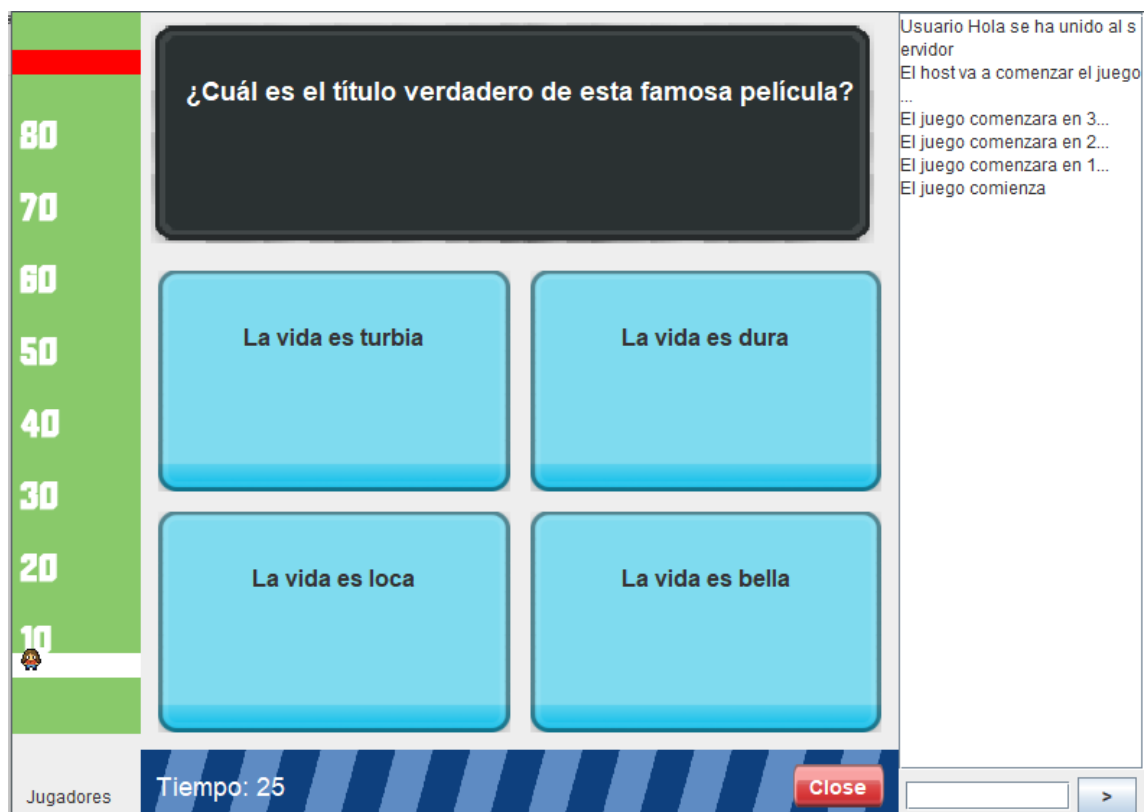
Es el tablero central que tiene la vista Game

PANELJUGADORES

Es el panel que representa las puntuaciones de los jugadores en la vista Game

PODIOVIEW

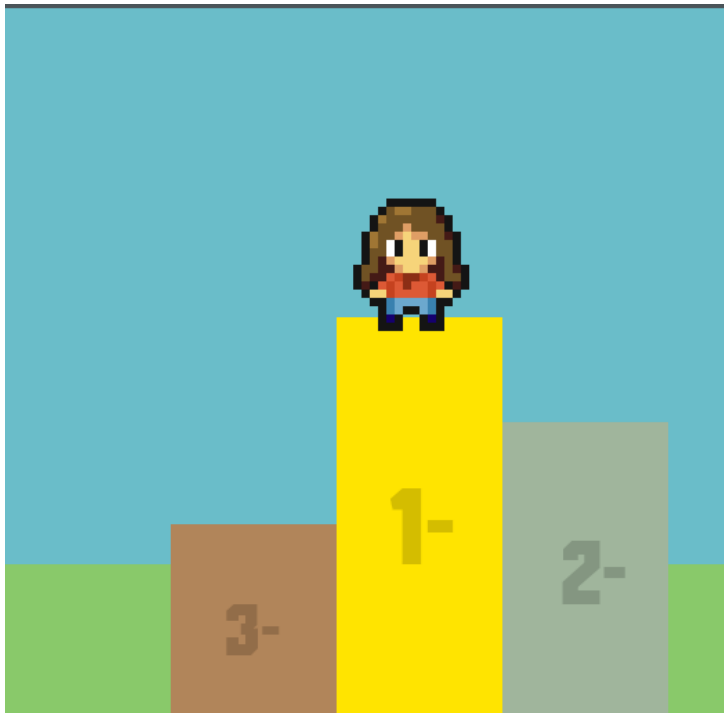
Es el panel que representa a los jugadores que han ganado después de terminar la partida



El tablero central al elegir una pregunta



El tablero central al terminar la partida



Si hubiera más jugadores, saldrían en los puestos siguientes