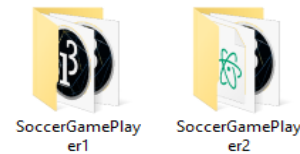


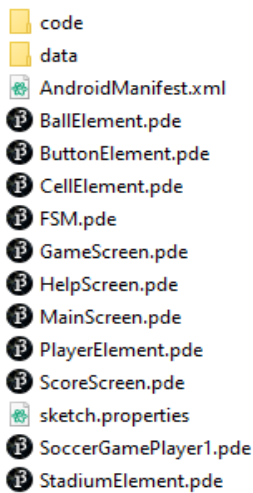
MANUAL RESUMIDO: SOOCER GAME

El objetivo de este proyecto es realizar un juego multijugador online sobre el “Ping Pong”, usando Processing para Android. En el proyecto abordo material de clase, tales como: Cargas de imágenes, formas geométricas de Processing, máquina de estados (FSM) y librerías como Ketai y Oscp5.

El proyecto tiene 2 versiones. Una versión cliente para el jugador 2 y una versión servidor para el jugador 1. La idea es que, mediante la librería Oscp5, permitir la comunicación entre los dos.



Sobre la estructura del proyecto, cuento lo siguiente:



La carpeta “data” que contiene imágenes.

El SoccerGamePlayer1/ SoccerGamePlayer2, es el main del proyecto, la cual inicia todo el juego.

Los archivos acabados en “Screen”, son las diferentes pantallas que contiene el juego.

Los archivos acabados en “Element”, son elementos gráficos que ayudan a los archivos “Screen”.

El archivo FSM, es la máquina de estado que controla que pantalla(estado) se dibuja en cada momento en el loop del juego.

La estrategia que he seguido para organizarme, es la siguiente:

Tengo una máquina de estados (FSM), que administra las diferentes pantallas a dibujar. Esta clase FSM es declarada, con acceso global, en el main del proyecto y ejecutada en el loop.

Las pantallas que contiene mi juego son 4: MainScreen, GameScreen, HelpScreen y ScoreScreen. Dependiendo del valor del estado del FSM, se dibujará una pantalla u otra. Los valores de estados disponibles, están declarados en la clase FSM como constantes globales, así desde cualquier “Screen” puedo cambiar el estado de la variable global FSM al estado correspondiente a la pantalla que quiero dibujar.

Los archivos “Element” de mis proyectos, són elementos gráficos que me ayudan a organizar el código. Me facilitan el dibujado de: la pelota, el estadio, el jugador, los botones, las celdas de la tabla de resultado.

También es en el fichero FSM, donde tengo definido el evento mousePressed().

A continuación, muestro un ejemplo de cómo cambio de estado a la FSM, estando en la pantalla MainScreen.

```
//PRESSED BUTTON MANAGER
void mousePressed(){
    switch(game.state)
    {
        case FSM.ST_MAINSCREEN:
            if(game.mainScreen.btnnPlay.isClicked(mouseX,mouseY)){
                game.state = FSM.ST_GAMESCREEN;
            }
            else if(game.mainScreen.btnnHelp.isClicked(mouseX,mouseY)){
                game.state = FSM.ST_HELPSCREEN;
            }
            else if(game.mainScreen.btnnScores.isClicked(mouseX,mouseY)){
                game.state = FSM.ST_SCORESCREEN;
            }
            else if(game.mainScreen.btnnExit.isClicked(mouseX,mouseY)){
                exit();
            }
            break;
    }
}
```

Se cambia el estado del FSM



Las funciones que realizan cada “Screen”, son las siguientes:

- La pantalla mainScreen, muestra el logo de la versión del juego ejecutado y el menú principal con 4 botones (PLAY, HELP, SCORES, EXIT).
- La pantalla GameScreen, muestra el juego funcional del proyecto. Más adelante esta más comentado.
- La pantalla HelpScreen, muestra una imagen explicativa del juego y un botón RETURN, que nos devuelve al mainScreen.
- La pantalla ScoreScreen, muestra una tabla de los últimos 3 resultados de las partidas, almacenado mediante SQLite de la librería Ketai. En caso de no haber resultados aún, se muestra un mensaje indicándolo, en vez de la tabla. Además, se muestra un botón RETURN, que nos devuelve al mainScreen.

Sobre la pantalla GameScreen. Se trata de la pantalla donde el usuario juega en su lado del estadio. Primeramente, tras acceder desde el botón PLAY del mainScreen, se muestra la mitad de un campo de estadio y una cuenta atrás de 3 segundos. Luego es el jugador 1 el que siempre empieza la partida, con lo cual la pelota se mueve en su lado del campo. En caso de que la pelota colisione con la cabeza de la imagen del jugador, esta será repelida en dirección contraria, y tras sobrepasar el límite de la pantalla superior (centro del campo del estadio), pasara a la pantalla del otro dispositivo móvil con la versión cliente para el jugador2.

En caso de que el jugador no llegue a detener la pelota, y esta, llegue al limite inferior de la pantalla (portería), se mostrara un mensaje en ambos dispositivos, donde pondrá quien ha marcado el gol, y se reinicia la posición de los jugadores y de la pelota. La posición de la pelota estará en el centro del campo del lado del jugador al que le hayan marcado gol.

Para acabar la partida, se necesita 5 goles. Tras alcanzar los 5 goles, se muestra un mensaje en ambos dispositivos donde se indica el jugador ganador de la partida. Debajo del mensaje, se muestra también un botón RETURN, que nos devuelve al MainScreen.

TAREAS (Ordenada)	FECHA INICIO	TIEMPO
Crear pantalla Juego	13/05/2021	2d
Crear pantalla Inicio	15/05/2021	3h
Estructura de Estados (FSM)	15/05/2021	0.5h – 1h
Adaptar pantalla Juego y pantalla Inicio a la FSM	15/05/2021	4h
Crear pantalla Puntuación	16/05/2021	3h
Crear pantalla Ayuda	16/05/2021	2h
Crear versión cliente para jugador 2	17/05/2021	0.5h
Solucionar bugs generales	17/05/2021	2h
TOTAL		2d 17.5h

Vistazo a las diferentes pantallas:

