

## **PLANIFICACION SCRABBLE II**

Este documento non representa todos os métodos das clases pero si os métodos mínimos para a realización do programa.

### **Partida:**

#### **- Atributos:**

1. `puntosVictoria` : os puntos necesarios para gañar a partida
2. `xogadores`: neste caso son dous xogadores
3. `taboleiro`: as casillas vacías, con letras ou con multiplicadores
4. `numTurno`: o número de turno da partida

#### **- Constructores, getters e setters**

#### **- Métodos:**

1. `xogarPartida()`: método para xogar. Ao acabar a partida invócase o método "amosarResultados"
2. `iniciarTaboleiro()`: método que coloca os multiplicadores no taboleiro. O obxectivo inicial é que a súa posición sexa aleatoria.
3. `amosarResultados()`: método para amosar os resultados da partida. Contempla tamén a posibilidade dun empate entre os xogadores.
4. `seguinteXogador()`: método para cambiar de xogador. Non é un método preciso ao ter en conta que son dos xogadores pero separalo permitirá que engadir máis xogadores sexa máis fácil.
5. `colocarPalabra()`: usado para colocar as palabras no taboleiro. Toma como argumentos a posicion, a dirección e a palabra.

### Xogador:

#### - Atributos:

1. **nome:** o nome do xogador
2. **puntuación:** os puntos gañados na partida
3. **numRendicions:** o número de veces que o xogador pasou na partida. En principio no programa contéplase que o xogador so pode pasar tres veces ata perder.
4. **letras:** un array de 7 letras coas letras dispoñibles para colocar no taboleiro.

#### - Constructor, getters e setters

#### - Métodos:

1. **podeColocarPalabra():** o método úsase para comprobar se o xogador ten as letras para compoñer a palabra dada. Se pode colocala devolverá 1, se pode colocala usando algún comodín devolverá 0 e se non a pode colocar devolve -1.
  2. **retirarLetras():** este método retira as letras ou comodíns das letras dispoñibles
- Implantar o método **toString** para reflexar o estado do xogador (a súa puntuación e o seu nome).

### Casilla:

#### - Atributos:

1. **contido:** vacío se é unha casilla baleira. Tamén pode conter "x2" se é un multiplicador, unha letra,"ch" ou "ll".

2. **eMultiplicador:** booleano para facilitar saber se é un multiplicador. Cando se cambie o valor do contido este booleano debe ser actualizado

3. **valor:** cando o contido non é un multiplicador ou unha cadea vacía conterá os puntos da casilla

- Constructor, getters e setters

- Métodos:

1. **calcularPuntuacion():** chamará a clase Scrabble para pedir a puntuacion da cadea e se é un multiplicador multiplicará ese valor polo valor do multiplicador.

## **Scrabble:**

Clase con métodos estáticos para calcular puntuacións

## **EntradaSaida:**

Clase para xestionar a entrada do programa e tamén algúns métodos para imprimir erros ou imprimir con cores.