

Estanciero

Jacobo y su hno. Samuel son unos enfermitos del estanciero y para mejorar sus tácticas y estudiar nuevas estrategias deciden modelar su juego preferido en Wollok (qué capos!).

De cada **jugador nos va a interesar el dinero** que posee (por el hecho de cobrar y pagar) **y las propiedades** que tiene (por el hecho de comprarlas).

Se dispone de un tablero con casilleros. En su mayoría los casilleros son propiedades, donde **de cada propiedad se conoce su precio de compra inicial y su dueño**. Los otros casilleros son premios y el casillero de salida, explicados más adelante.

Caída en una propiedad

Cuando un jugador J cae en una propiedad puede ocurrir alguna de estas tres cosas

- Que la propiedad no pertenezca a nadie (es decir, **que el dueño sea el Banco**):
En este caso el jugador que cayó ahí compra la propiedad pagándole al Banco
- Que el **dueño** de la propiedad sea un **jugador rival** (esto es $J \neq \text{dueño}$)
El jugador debe abonarle al dueño el valor de la renta de esa propiedad.
- Que el **dueño** sea de uno mismo (esto es $J = \text{dueño}$)
No pasa nada

Tipos de Propiedades

Hay 2 tipos de propiedades

I. Campos II. Empresas

I. Campos

Algunas propiedades son "Campos" y están agrupadas en grupos llamados "Provincias".

De cada campo se **conoce su valor de renta fijo** (o sea, el valor de la renta sin estancias), se **conoce el costo de construcción de cada estancia** (en ese campo) y se **conoce la cantidad de estancias** que están construidas en él.

Además, cada campo debe saber decirnos el valor de renta para los casos en que la propiedad tenga construida estancias, que se calcula con la siguiente fórmula:

$$\text{Renta para } N \text{ estancias} = 2^N * \text{valor renta fijo}$$

Ejemplo, si al campo "Campo Bravo" tiene construidas en él 3 estancias y tiene un valor de renta fijo de \$3000 su renta es \$24.000 porque $2^3 * 3000 = 24000$

Construcción de estancias: Deben validarse 2 cosas en la provincia

1. ser monopolio: que todos los campos de la provincia tengan el mismo dueño
2. construcción pareja: la cantidad de estancias en el campo en donde se realiza la construcción debe ser menor o igual a la cantidad de estancias que tiene cada campo de la provincia

Una provincia conoce los campos que tiene y cada campo conoce a qué provincia pertenece.

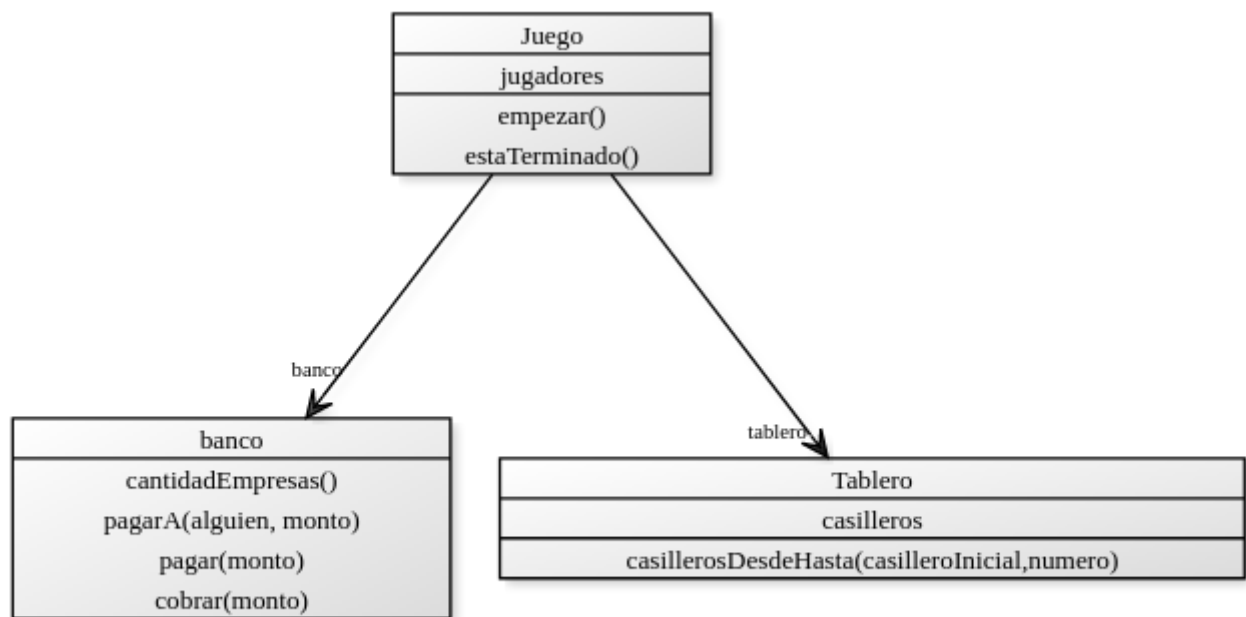
II. Empresas

Otras propiedades son empresas, son tres en todo el juego que conviene acumularlas ya que la renta aumenta al tener mayor cantidad de las mismos.

Si un jugador cae en una empresa ese jugador debe

1. tirar los dados nuevamente (asumamos que saco X)
2. pagarle al dueño de dicha empresa:
 - $X * \$30.000$ de renta si el dueño tiene una sola empresa ($X * 30000 * 1$)
 - $X * \$60.000$ de renta si el dueño tiene 2 empresas ($X * 30000 * 2$)
 - $X * \$90.000$ de renta si el dueño tiene 3 empresas ($X * 30000 * 3$)

Ya se cuenta con las siguientes clases:



Asumimos que todas las propiedades al momento de ser creadas tienen como **dueño** al Banco

```
class Juego >> empezar() {
  if (!self.estaTerminado()) {
    jugadores.forEach { jugador => self.queJuegue(jugador) }
  }
}
```

Los métodos que están en el diagrama ya están implementados y hacen lo que su nombre indica (no los tienen que hacer y los pueden usar si lo creen necesario).

El método del Tablero, `casillerosDesdeHasta(casilleroInicial, unNumero)`

- **Recibe** un casillero (un objeto instancia de alguna clase creada por vos) y un número
- **Devuelve** una lista de casilleros donde el primer elemento es `casilleroInicial`, el segundo es el casillero siguiente a `casilleroInicial`, el tercero es el siguiente al siguiente a `casilleroInicial` y así; siendo el último elemento de la colección el casillero que está a `unNumero` de posiciones de `casilleroInicial`

Requerimientos (por cada método se debe especificar en que clase se ubica)

1.
 - a. Obtener la colección de dueños que tiene una provincia sin repetidos. O sea, los dueños de sus campos (en la colección puede haber jugadores y el banco)
 - b. Agregar una estancia en un campo validando que se pueda construir. Para la validación preguntarle a **su provincia** si se puede construir en ese campo (Ver **Construcción de estancias**) no olvidarse de que el dueño pague lo correspondiente por dicha estancia al Banco (cada campo conoce el precio de construcción)
2.
 - a. Hacer que todas las propiedades entiendan el mensaje `sosEmpresa`, los campos devuelven `false` y las empresas devuelven `true`
 - b. Saber la cantidad de empresas que tiene un jugador
3. Hacer que un jugador entienda el mensaje **tirarDados**. Esto devuelve una cantidad que es la suma de dos números aleatorios entre 1 y 6 (**)
4. Hacer que todas las propiedades entiendan el mensaje **rentaPara(jugadorQueCayo)**
 - Para los **campos** tener en cuenta las estancias que hay construidas allí
 - Para las **empresas** tener en cuenta
 - a. la cantidad que saco en los dados el jugador que cayó (recuerden que el **jugador que cayó** tiene que tirar los dados nuevamente)
 - b. la cantidad de empresas que tiene el **dueño** de la propiedad
5. Hacer que un jugador pague una suma de dinero a un acreedor; si no tiene dinero suficiente se debe lanzar un error. Cuando un jugador paga a otro, su dinero disminuye y el del acreedor aumenta. **Los jugadores y el banco deben ser polimórficos.**
6. Hacer que cualquier casillero entienda el mensaje `pasó(unJugador)`
 - Por defecto cuando un jugador pasa por un **casillero no sucede nada**
 - Si un jugador pasa por la **Salida cobra \$5000**(*)
7. Hacer que cualquier casillero entienda el mensaje `cayó(unJugador)`
 - Cuando un jugador cae en una **propiedad** lo que pasa **depende del dueño**.
 - Si es el banco este le dice al jugador que compre la propiedad
 - Si el dueño es otro jugador rival este le dice al jugador que cayó que le pague la renta de la propiedad.
 - Cuando cae en **Premio Ganadero** el jugador cobra \$2500(*)
 - Cuando cae en **Salida** no pasa nada
8. Hacer que un jugador se **mueva sobre** una colección de casilleros que le llega por parámetro. Esto hace que:
 - El jugador **pase por todos** los casilleros de esa colección
 - El jugador **caiga en el último** casillero de esa colección
 - El **casillero actual** del jugador sea el **último casillero** de esa colección
9. Hacer el método `Juego >> queJuege(unJugador)` en donde `unJugador` tiene que
 - tirar los dados (digamos que sacó X) y,
 - moverse sobre los casilleros correspondientes (OJO! primero tienen que saber en donde está el jugador para que recorra desde ahí hasta X)

(*) esta es plata espontánea, o sea, no se le decrementa a nadie

(**) los numeros entienden el mensaje **randomupTo(limiteSuperiorNoIncluido)**

Estanciero - Extensiones

Se desea agregar al modelo hecho anteriormente algunos nuevos requerimientos:

1. Prisión y pérdida de turnos
2. Estrategias de compra de un jugador cuando cae en una propiedad
3. Hipotecas

1) Se agrega un nuevo casillero que es la **Prisión**.

- Cuando un jugador cae en la prisión no pasa nada.
- Cuando un jugador pasa por la prisión no pasa nada.

Ahora bien, cuando un jugador saca 2 veces seguidas doble en los dados, va preso.

Ir preso significa que:

- El casillero actual del jugador pasa a ser el casillero Prisión
- El jugador deberá permanecer en la prisión 3 turnos consecutivos (o sea, perder 3 turnos).
A excepción de que al comienzo de su turno saque nuevamente doble en los dados.
Ya sea que saca doble estando preso o que pasen 3 turnos desde su encarcelamiento, el jugador deja de estar preso y continua jugando normalmente.

2) Actualmente cuando un jugador cae en una propiedad cuyo dueño es el banco, el jugador está obligado a comprar dicha propiedad. Se detectó que un jugador en un momento dado tiene una de estas 3 estrategias de compra (un jugador a lo largo del juego puede cambiar su estrategia de compra):

Standard: es el comportamiento que se describió anteriormente. El jugador que cae en una propiedad que tiene como dueño al banco, está obligado a comprarle dicha propiedad al banco.

Garca: cuando un jugador garca cae en una propiedad solo le compra al banco dicha propiedad ...

- Si cae en un campo: se tiene que cumplir que entre las propiedades que pertenecen a la provincia de la propiedad en la que cayó el jugador ya hay propiedades que le pertenecen a otro jugador distinto del que cayó.
Ejemplo: si el jugador A cayó en el campo *Buenos Aires Zona Norte* tiene que haber en la provincia *Buenos Aires* (provincia a la que pertenece el campo) alguna otro campo cuyo dueño sea un jugador distinto del jugador A
- Si cae en una empresa: se tiene que cumplir que el dueño de alguna otra empresa del juego sea otro jugador.
Ejemplo: si el jugador A cayó en la empresa *Bodega San Filipo* algún otra empresa del juego, que no sea la *Bodega San Filipo*, debe tener como dueño un jugador distinto del jugador A

Imperialista: un jugador solo compra una propiedad si ...

- Si cae en un campo: se tiene que cumplir que el jugador que cayó ya tiene algún otro campo en la provincia del campo en donde cayó o que todos los campos de esa provincia no tengan dueño (o sea, que el dueño sea el banco)
- Si cae en una empresa: se tiene que cumplir que todas las empresas no tengan dueño (o sea, que el dueño sea el banco)

3) Cuando a un jugador no le alcanza el dinero para pagar un monto, en vez de lanzar un error el jugador hipoteca todas las propiedades que necesite para pagar dicha la suma de dinero.

Si hipotecando todas sus propiedades no alcanza a pagar dicho importe, recién en ese momento el jugador pierde - ahí es donde se deberá lanzar el error.

Cuando se hipoteca una propiedad se debe tener en cuenta

- Para cualquier propiedad el banco le paga al dueño de la propiedad la mitad de su valor de compra y a efectos prácticos la propiedad pasa a ser del banco, con la diferencia de que si un jugador cualquiera (inclusive el pobre flaco que hipoteco todo) cae en esa propiedad para comprarla deberá pagar el valor de compra inicial * 1.5 (o sea, debe levantar la hipoteca) para poder comprarla y ser el nuevo dueño.
- Para los campos además, el banco le paga al dueño de la propiedad, por cada estancia que había en el campo, la mitad del valor de construcción de esa estancia.
- Para las empresas no hay nada particular que hacer.