



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA
DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN
IIC2233 — PROGRAMACIÓN AVANZADA

Actividad 11

2º semestre 2017
19 de octubre de 2017

Interfaces gráficas

Introducción

The walking luddite: una nueva infección ha invadido a nuestra ciudad, ¡pero esta vez está haciendo adicta a la gente jugar en los casinos! Los casinos, Monticello y Enjoy, están colapsando, por lo que le han pedido que programe máquinas tragamonedas virtuales: así las personas podrán jugar desde sus hogares.

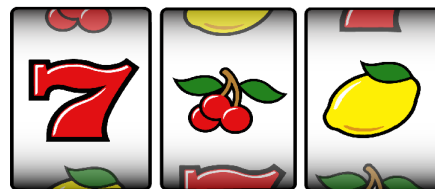
Usted debe crear una interfaz que represente una máquina de casino con las restricciones y funcionalidades que se mencionarán a continuación.

Cómo jugar

Para ingresar al juego el usuario debe escribir su nombre. Si él ya ha jugado anteriormente, entonces se debe poder ver la información de su cuenta: el valor de la última apuesta, el valor de la máxima apuesta ganada, y el saldo al final de la última sesión. Si no ha jugado antes, se debe crear una cuenta, donde la **última apuesta hecha** y la **máxima apuesta ganada** son cero. El **saldo disponible** debe ser 1.500.

Para jugar, el usuario debe hacer una apuesta. Esta debe ser mayor a 500 unidades y menor o igual que el saldo disponible. El usuario debe poder bajar o subir la cantidad apostada con los botones de bajar (-) y subir (+) en 500 unidades la apuesta. Cuando haya decidido su apuesta, debe hacer clic en un botón para que la máquina haga *rodar* las imágenes. Si el usuario no ha realizado una apuesta válida, este botón debe estar deshabilitado.

El tragamonedas tiene tres posibles imágenes: un siete, unas guindas, y un limón.



El valor ganado (o perdido) es el valor de la apuesta multiplicado por un factor. Este factor depende de la combinación de las figuras obtenidas según el cuadro. Si hubiesen dos combinaciones que se cumplen, se tomará la de mejor valor. Cualquier caso no especificado en el cuadro significa que el valor del factor es 0.

Combinación	Factor
Tres sietes	2
Tres guindas	1,5
Tres limones	1,25
Uno o dos sietes	0,9
Dos guindas	0,8

Cuadro 1: factores para obtener valor ganado

Luego de que aparezcan las imágenes definitivas, se debe mostrar un mensaje indicando cuál fue el resultado de la jugada. Los mensajes deben indicar las combinaciones de la tabla y el factor obtenido. Si es cualquier otra combinación debe decir “Mala suerte esta vez. Tu factor es cero”. Al finalizar la jugada, el monto del saldo debe actualizarse dependiendo del resultado de ésta (ganancia o pérdida).

Si el saldo final es menor a quinientas unidades, el juego debe informar al usuario que no podrá seguir jugando. El juego se debe cerrar automáticamente. Si el usuario vuelve a entrar con el mismo nombre deberá comenzar con las mismas condiciones iniciales.

Cuando el usuario quiera dejar de apostar, debe cerrar su sesión y su información debe ser guardada para la próxima vez que quiera ingresar. El formato de archivo no es importante: puede ser texto plano, pero recuerde que deben poder cargarse correctamente en el juego.

Front-end

Las siguientes son las componentes que deben verse en el *front-end* (interfaz).

- Las tres imágenes mencionadas anteriormente que simulan los tres rollos que giran en la máquina. Las imágenes se encuentran en la carpeta anexa **assets**.
- Un recuadro con el saldo actual del jugador.
- Un recuadro con la apuesta que se está por hacer.
- Un botón para subir la apuesta y uno para bajarla.
- Un botón para jugar la apuesta.
- Un recuadro con la máxima apuesta ganada.
- Un recuadro con la última apuesta realizada.

La interfaz debería parecerse a la siguiente imagen:



Al presionar el botón para jugar, todas las imágenes deben iniciar con la imagen de *tapa* (i.e. la imagen que tiene tres sietes que se ve en el ejemplo) y luego de 0,5 segundos deben voltearse, de izquierda a derecha con un tiempo intermedio de 0,5 segundos entre ellas, mostrando la imagen aleatoria resultante.

Front-end y back-end

Todo lo mencionado hasta ahora está referido al enfoque del usuario, es decir, a las funcionalidades del *servicio*. Sin embargo, es importante aprender a separar en *front-end* y *back-end*. **Es obligatorio** hacer la separación *front-end* y *back-end*.

Como consejo, identifique cuáles son las entidades que cambian su estado a través del juego y representélas en el *back-end*. Recuerde que el *back-end* es la parte lógica del programa en la que se hacen los razonamientos y cálculos para cambiar los estados de las entidades, mientras que el *front-end* es sólo la representación visual del programa donde no se hacen grandes razonamientos.

Notas

- Esto probablemente le será útil: si usa `QPixmap` para representar las imágenes, recuerde que `QPixmap` tiene el método `scaled(ancho, alto)` que sirve para cambiar las dimensiones de la imagen.
- Recuerde, además, que puede usar `QTest.qWait` de la librería `PyQt5.Qt`, que detiene la ejecución del programa por la cantidad de milisegundos que ingrese como parámetro.
- El *razonamiento* en la selección de las imágenes aleatorias puede ser implementado en el *front-end*.

Requerimientos

- (3,60 pts) *Front-end*

- (0,80 pts) El cuadro de texto y el botón de ingreso que permiten comenzar el juego luego de ingresar un nombre de usuario.
- (0,80 pts) Se puede disminuir o aumentar la apuesta.
- (0,60 pts) Las imágenes cambian cada 0,5 segundos en orden y comenzando por la imagen *tapa*.
- (0,60 pts) Aumenta o disminuye el saldo dependiendo del resultado.
- (0,80 pts) Se actualiza la interfaz después de realizar la apuesta.
- (2,40 pts) *Back-end*
 - (0,80 pts) Se separa el *front-end* del *back-end*.
 - (0,80 pts) Obtienen los factores de acuerdo a las reglas.
 - (0,80 pts) Se cargan y guardan los usuarios desde la base de datos.

Entrega

- **Lugar:** En su repositorio de GitHub en la **carpeta** Actividades/AC11/
- **Hora:** 16:55
- Si está trabajando en pareja, basta con que un miembro suba la actividad. Si se suben actividades distintas, se corregirá una de las dos al azar.