

**Ingeniería en Tecnologías de Información.**

**Curso: Programación II.**

**Prof. Jonathan Moreno Núñez.**

**Fecha de entrega: 16 DE NOVIEMBRE, 11:59 pm.**

## **PROYECTO PROGRAMADO I**

La Cruz Roja es una institución benemérita sin fines de lucro. Con el fin de recaudar fondos para sufragar sus gastos, fundó el famoso Tico Bingo, donde sortea premios que han sido donados por empresas benefactoras.

Usted ha sido contratado por dicha institución como programador para que elabore un sistema automatizado gráfico en JAVA para el juego del Tico Bingo, con el fin que este proceso sea transparente y más llamativo para los clientes. A continuación, se detallan todas disposiciones que se deben tomar en cuenta para desarrollar dicho sistema:

- El sistema debe contar con 10 cartones (de 5x5, numerados del 1 al 10), **por supuesto, todos diferentes y elaborados por el sistema de manera automática** (aleatoria). La distribución de los números de los cartones debe hacerse de la siguiente forma:
  - En la primera columna solo se pueden asignar números del 1 al 15.
  - En la segunda, números del 16 al 30.
  - En la tercera, números del 31 al 45.
  - En la cuarta, números del 46 al 60.
  - En la quinta, números del 61 al 75.

BINGO				
12	18	41	47	61
7	26	39	54	70
4	27	FREE SPACE	49	63
5	23	35	58	72
3	30	32	52	68

***Nota: Un número X solo puede aparecer una única vez en cada cartón y la casilla central debe quedar vacía como se muestra en la figura.***

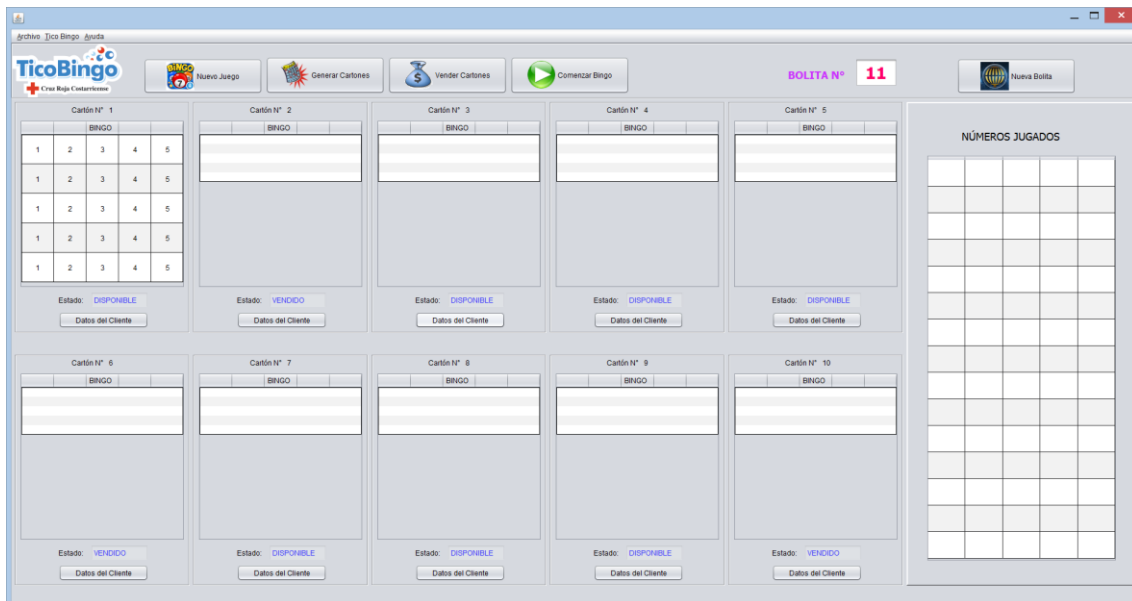
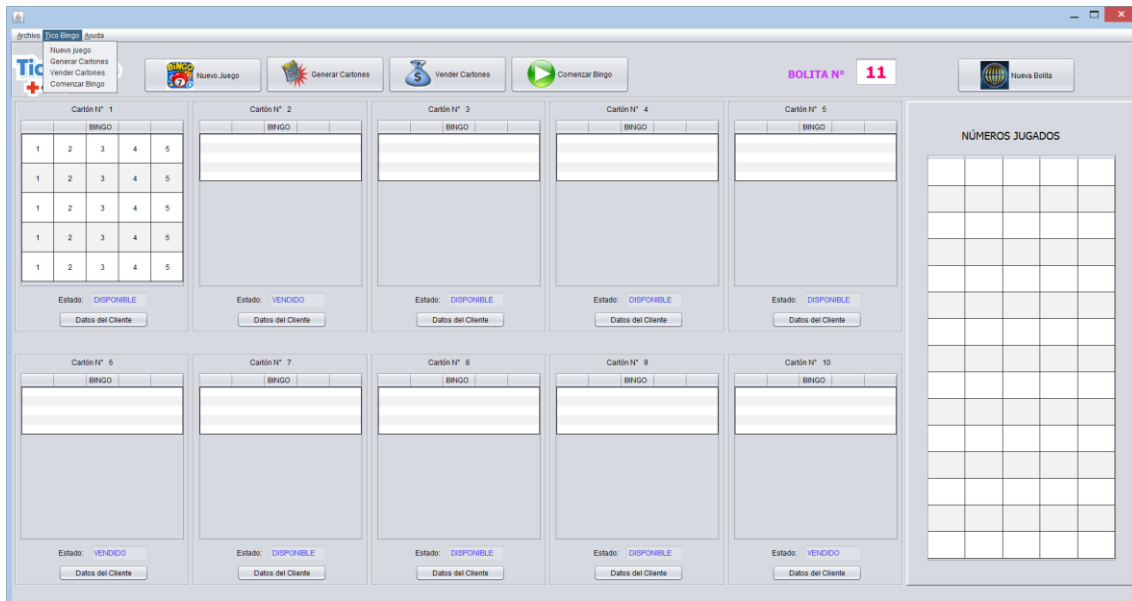
- Cada cartón será 1 hilo separado que llenará su propio cartón y luego se duerme, pero cuando se genera una bolita desde el programa principal (tómbola), se despiertan todos los hilos al mismo tiempo para que busquen en su cartón si ese número está presente y lo marcará (Colorea) en el cartón de la vista. Luego de marcar la casilla con el número de la tómbola que acaba de salir, cada hilo debe revisar si ha ganado la partida, y si gana, el juego termina y el jugador de ese cartón gana, pero si no gana ningún cartón los hilos se duermen para esperar que salga otra bolita de la tómbola.

# UNIVERSIDAD TÉCNICA NACIONAL

## SEDE DE GUANACASTE



- El sistema debe poseer un Formulario principal JFrame con las siguientes opciones de menú:
  - Nuevo Juego
  - Generar cartones (10 cartones, se crean los hilos).
  - Vender cartones.
  - Comenzar Bingo.
  - Ver historial.
- La IGU del sistema se debe ver similar a las siguientes pantallas:



- En el caso de la **primera opción Nuevo Juego** del menú Tico Bingo (botón Nuevo Juego), se deben inicializar todos los parámetros de la aplicación para iniciar un nuevo juego.
- En la **segunda opción** del menú Tico Bingo, **Generar Cartones** (botón Generar Cartones), Se crean e inician los hilos y se deben crear y llenar los cartones al azar, cumpliendo con las disposiciones y reglas que con anterioridad se mencionaron.
- En la **tercera opción** del Menú Tico Bingo, **Vender Cartones** (botón Vender Cartones), cuando un cartón es vendido a una persona, se deben tomar los datos personales de ésta (nombre, apellidos, cédula, teléfono y número de cartón que adquiere) y almacenarlos en un **ArrayList** tipo Clase Persona, con los datos de la persona. Recuerde crear la clase **Persona**.
- Recuerde que el programa **debe ser totalmente orientado a objetos**, y se deben crear las clases necesarias para cumplir con los requerimientos.
- En la **opción 4 Comenzar Bingo** del menú Tico Bingo (botón Comenzar Juego) se debe dar inicio al bingo, **donde los números del Bingo (del 1 al 75) se van a ir generando aleatoriamente y sin repetirse**, después de presionar el botón **Nueva Bolita**. En esta pantalla el juez tendrá la opción de ver el avance de cada cartón y que posibilidades tendrá de ganar cada jugador; **los números que ya han salido se deben resaltar (puede ser cambio de color) en cada cartón** (1-10) del resto de números que no han salido (simulando que ponemos el granito de maíz sobre el número en el cartón).
- Una vez que una bolita es generada, la misma se muestra en la parte superior del programa, pero al mismo tiempo se inserta en orden en la tabla del panel derecho que se muestra en la imagen de la interfaz de ejemplo.
- Un jugador puede ganar llenando cualquier **fila, columna, diagonales o cuatro esquinas solamente**.
- Mientras el bingo se está jugando se debe llevar el tiempo de duración del juego (Utilización del Timer de Java).
- En el momento que uno o varios cartones obtengan la victoria, el sistema debe notificar al usuario y mostrar los números de cartones ganadores, el cartón en sí con la jugada ganadora, además, de mostrar los datos de la persona que lo compró.
- Cuando se gana, los datos de la persona se guardan en una tabla de una base de datos llamada **Historial**, así como la fecha, la hora actual y número de cartón con el que ganó. En la tabla de la base de datos se guarda un historial de los últimos 10 ganadores de los últimos 10 sorteos jugados.

- En la quinta opción Ver Historial, se debe mostrar en un nuevo JDialog un JTable que muestre el historial de los últimos 10 ganadores.
- *El programa debe estar estrictamente orientado a objetos, de lo contrario perderá la calificación total del mismo. Debe pensar en el modelo de programación orientado a objetos. Utilice clases, objetos, atributos, métodos, set, get, polimorfismo.*
- Otro aspecto de suma importancia es que cada cartón en el juego será un Thread. El programa principal llevará un ArrayList de los cartones vendidos (datos de las personas que los compraron) y la tómbola con los números que van saliendo; cada cartón será un hilo por lo que debe estar revisando la mesa de la tómbola para ver que números han salido. En el momento que se presione el botón **Nueva Bolita** el acceso al tablero de bolitas salidas estará bloqueado hasta que se genere una nueva, en ese lapso los hilos estarán **wait()**, cuando la ficha salga se le debe notificar **notify()** a los hilos cartones para que verifiquen si poseen el número que acaba de salir y marcan tanto lógica como visualmente el dato en el cartón, además debe chequear si con ese número gana la partida.
- Gana el cartón que logre completar una de las jugadas ganadoras estipuladas anteriormente.

#### **Notas Aclaratorias:**

1. Debe hacerse la documentación interna respectiva usando Java Doc y comentarios.
2. El código fuente de cada clase debe estar correctamente **identado**, se restarán puntos de lo contrario.
3. Dos tareas iguales tienen un 0(cero) en la nota, así como proyectos copiados de Internet o cualquier otro medio, o que no cumpla con los requerimientos plasmados en este documento.
4. No se reciben proyectos después de la fecha de entrega establecida.
5. Si el sistema se cae será estrictamente calificado.
6. El proyecto será realizado en **grupos de 3 personas máximo**.
7. Debe entregarse por medio del campus virtual, **1 entrega por grupo**.
8. *Si para la elaboración de este proyecto, se requiere de alguna información no vista en clase, la misma debe ser investigada (Clase Timer, Menús, Table).*

#### **Recomendaciones**

Haga frecuentes respaldos de su proyecto, para no perder el avance de cada esfuerzo realizado. Recuerde los respaldos en la NUBE son una excelente opción.