

LOTERIA DE NAVIDAD

-Ficha técnica-



**Xavier García
Biel Palomar
Óscar Zapatero
Kevin Burga**



ÍNDICE

- 1. Resumen del proyecto**
- 2. Diario de desarrollo**
- 3. Retrasos**
- 4. Funcionalidad**
- 5. Contenido del proyecto**
- 6. Infografía**
- 7. Premios de la lotería:**



Resumen del proyecto

El documento redacta el informe técnico del proyecto del grupo FACHERITOS compuesto por: Xavier García, Biel Palomar, Óscar Zapatero y Kevin Burga. El proyecto ha sido desarrollado en las plataformas de NetBeans y GitHub utilizando el lenguaje java para programar el proyecto.

El proyecto consiste en simular el sorteo de la lotería de navidad, donde el cliente que haya comprado un boleto pueda preguntarle al programa si su boleto ha sido premiado o le corresponde otra cantidad por cercanía a premios, entre diferentes premios generados automáticamente.

Para poder trabajar en grupo de manera ordenada, y con el fin de poder preservar el código y otros elementos del proyecto, hemos utilizado la plataforma GitHub para poder guardar nuestro progreso, y así también tener una mayor facilidad para poder acceder al contenido para modificar o documentar el código.

A continuación se detallarán los diferentes contenidos, avances y retrasos que hemos tenido durante el proyecto.



Diario de desarrollo

Dia 12/12/2022 - 16/12/2022

- Entendimiento del proyecto.
- Creación de repositorio en GitHub grupal.
- Creación de grupo de Whatsapp para comunicarnos.

Dia 19/12/2022 - 13/01/2023

- Búsqueda de información para el desarrollo lógico del programa.
- Creación de fichero de los datos necesarios para el proyecto.
- Inicialización de un nuevo proyecto en Java gestionado en Github.
- Desarrollo del programa de manera no modular.
- Creación del sorteo
- Funciones básicas para facilitar la lectura.

Dia 16/01/2023 - 20/01/2023

- Creación de una ficha técnica para recoger la información del proyecto a nivel lógico y técnico.
- Finalización del apartado funcional del programa.
- Comprobación de los cupones

Dia 23/01/2023 - 27/01/2023

- Reestructuración en funciones de todo el programa.
- Creación de constantes útiles.
- Redacción de la ficha técnica.

Dia 30/01/2023 - 3/02/2023

- Documentación del código y comentado.
- Se descarta la idea de la compra de boletos.
- Finalización de la ficha técnica.
- Reunión para defensa y entendimiento del código.
- Casos a probar la funcionalidad del código.



Retrasos

- Uso de una ficha técnica para documentar el proceso de desarrollo del proyecto:
Previsto: periodo 19/12/22 - 13/01/23
Efectuado: periodo 16/01/23 - 20/01/23
- Comentado del código para un mejor entendimiento general:
Previsto: semanalmente
Efectuado: periodo 16/01/23 - 20/01/23
- Casos de pruebas para ver el funcionamiento del código.
Previsto: 22/01/23 - 25/01/23
Efectuado: Indeterminado
- Implementacion de test.
Previsto: 30/01/23 - 02/02/23
Efectuado: Indeterminado



Funcionalidad

- Sorteo de premios
 - Generación de un bombo de premios

```
public static int[] CrearBomboPremios() {  
    final int GORDO = 4000000;  
    final int SEGONPREMI = 1200000;  
    final int TERCERPREMI = 500000;  
    final int CUARTPREMI = 200000;  
    final int QUINTOPREMI = 60000;  
    final int PEDREADA = 1000;  
    final int CANTIDADPREMIOS = 1807;  
  
    int[] bomboPremios = new int[CANTIDADPREMIOS];  
    for (int i = 0; i < bomboPremios.length; i++) {  
        if (i < quantPremis) {  
            for (int j = 0; j < quantGordo; j++) {  
                bomboPremios[i] = GORDO;  
                i++;  
            }  
            for (int j = 0; j < quantSegund; j++) {  
                bomboPremios[i] = SEGONPREMI;  
                i++;  
            }  
            for (int j = 0; j < quantTercer; j++) {  
                bomboPremios[i] = TERCERPREMI;  
                i++;  
            }  
            for (int i = 0; i < quantCuart; i++) {
```

- Generación de números (no repetidos) asignados a cada premio del bombo. Creación de un número premiado extrayendo una bola del bombo de premios.

```
public static NumPremiado[] Sorteo() { //Cada premio  
    int[] bomboPremios = CrearBomboPremios();  
    NumPremiado[] numeros_premiats = new NumPremiado[1807];  
    int num_afegir = 0;  
    int premi_afegir, posP;  
    for (int i = 0; i < numeros_premiats.length; i++) { //Crear todos los premios  
        boolean repeatnum = true;  
        while (repeatnum) {  
            num_afegir = rnd.nextInt( bound:100000); //Generar número aleatorio  
            repeatnum = false;  
            for (int j = 0; j < i && !repeatnum; j++) { //Comprobar que el número  
                if (num_afegir == numeros_premiats[j].numero) {  
                    repeatnum = true;  
                }  
            }  
        }  
  
        //Escoger número del bombo  
        posP = rnd.nextInt(bomboPremios.length - i);  
        premi_afegir = bomboPremios[posP]; //Escoger premio del bombo  
        //Crear premio a partir de el número y premio sacados de los bombos  
        numeros_premiats[i] = new NumPremiado();  
        numeros_premiats[i].numero = num_afegir;  
        numeros_premiats[i].premio = premi_afegir;
```

```
//</editor-fold>  
/**  
 * Clase de números premiados  
 */  
public static class NumPremiado {  
  
    int numero;  
    int premio;  
}
```



- Muestra de los premios. Ordenación previa de los premios por el método de la burbuja.

```
* Función para mostrar el listado completo de números premiados
* @param num_premi array de número premiados
*/
public static void MostrarPremios(NumPremiado[] num_premi) {
    String numeroS = "";
    System.out.println( x: "\nLotería de Navidad");
    System.out.println( x: "*****");
    for (int i = 0; i < num_premi.length; i++) {
        numeroS = "" + num_premi[i].numero;
        //Añadir ceros a los números de menos de 5 cifras
        while (numeroS.length() < 5) {
            numeroS = "0" + numeroS;
        }
        System.out.println((i + 1) + ". Número: " + numeroS + " Premio: " + num_premi[i].premio)
    }
}
```

```
Qué deseas consultar:
1) Revisar premio de cupón
2) Consultar todos los premios
3) Sortir
Opción seleccionada: 2

Lotería de Navidad
*****
1. Número: 73891 Premio: 4000000
2. Número: 62435 Premio: 1200000
3. Número: 44355 Premio: 500000
4. Número: 95769 Premio: 200000
5. Número: 80150 Premio: 200000
6. Número: 88179 Premio: 60000
7. Número: 26229 Premio: 60000
8. Número: 08347 Premio: 60000
9. Número: 04506 Premio: 60000
10. Número: 51934 Premio: 60000
11. Número: 53646 Premio: 60000
12. Número: 85255 Premio: 60000
```

- Comprobación de premios
 - Introducción de un número de cupón[000]

```
Qué deseas consultar:
1) Revisar premio de cupón
2) Consultar todos los premios
3) Sortir
Opción seleccionada: 1
Introduce tu número: 23517
```



- Comprobación de premios

```
public static int ComprobarPremio(int cupon, NumPremiado[] premiados) {
    //Creamos la variable premio
    int premio = 0;

    //Miramos si ha tocado algun premio principal
    premio = ComprobarBombo(cupon, premiados);

    //Si no ha tocado miramos la aproximación
    if (premio == 0) {
        premio = ComprobarAproximacion(cupon, premiados);
    }
    //Si sigue sin tocar nada miramos la centena
    if (premio == 0) {
        premio = ComprobarCentena(cupon, premiados);
    }
    //Si sigue sin tocar nada miramos las 2 últimas cifras del cupón
    if (premio == 0) {
        premio = ComprobarUltimas(cupon, premiados);
    }
    //Si sigue sin tocar nada miramos la pedrea
    if (premio == 0) {
        premio = ComprobarBombo2(cupon, premiados);
    }
    //Por último miramos si se lleva el reintegro
    premio += ComprobarReintegro(cupon, premiados, premioactual:premio);

    //Devolvemos el valor premiado
    return premio;
}
```

1. Comprobación de premios principales. Se revisa si el número corresponde al gordo, segundo premio, tercer premio, cuarto premio o quinto premio.

```
public static int ComprobarBombo(int cupon, NumPremiado[] premiados) {
    /*Creamos la constante con al cantidad de numeros mayores que hay - 1
    1 primer premio
    1 segundo premio
    1 tercer premio
    2 cuartos premios
    8 quintos premios
    */
    final int TOTALGUANYADORS = 12;
    //Creamos las variables del programa
    boolean ganador = false;
    int premio = 0;

    //Miramos si el cupon corresponde con alguno de estos premios y le asignamos el premio si és asi
    for (int i = 0; i <= TOTALGUANYADORS && ganador == false; i++) {
        if (cupon == premiados[i].numero) {
            premio = premiados[i].premio;
            ganador = true;
            nompremi = nombrePremiado(premio, premiados);
        }
    }

    //Devolvemos el valor de premio
    return premio;
}
```

2. Comprobación de aproximación a un premio principal. Si el número de cupón sobrepasa por una unidad o queda una unidad por debajo de uno de los tres premios principales.

```
public static int ComprobarAproximacion(int cupon, NumPremiado[] premiados) {
    //Creamos las constantes con los valores de los premios
    final int PREMIAPROX3 = 9600;
    final int PREMIAPROX2 = 12500;
    final int PREMIAPROX1 = 20000;

    //Creamos la avriable a devolver
    int premio = 0;

    //Miramos si el cupon se aproxima al primer premio
    if (cupon == premiados[0].premio - 1 || cupon == premiados[0].premio + 1) {
        premio = PREMIAPROX1;
        nompremi = "La aproximación al gordo";
    }
    //Miramos si el cupon se aproxima al 2do premio
    else if (cupon == premiados[1].premio - 1 || cupon == premiados[1].premio + 1) {
        premio = PREMIAPROX2;
        nompremi = "La aproximación al 2do premio";
    }
    //Miramos si el cupon se aproxima al 3er premio
    else if (cupon == premiados[2].premio - 1 || cupon == premiados[2].premio + 1) {
        premio = PREMIAPROX3;
        nompremi = "La aproximación al 3er premio";
    }

    //Devolvemos el valor de premio
    return premio;
}
```




3. Comprobación de la centena de un premio principal. Si el número de cupón forma parte de la misma centena que uno de los cuatro premios principales.

```
public static int ComprobarCentena(int cupon, NumPremiado[] premiados) {  
    //Creamos la constante del valor del premio  
    final int PREMICIENTENA = 1000;  
    //Creamos la variable a devolver  
    int premio = 0;  
    //Comprovamos si la centena corresponde a alguna de ls 4 premios mayores  
    for (int i = 0; i <= 4; i++) {  
        if (cupon >= (premiados[i].numero / 100) * 100 && cupon <= (premiados[i].numero / 100) * 100) {  
            premio = PREMICIENTENA;  
            nompremi = "el premio a la centena";  
        }  
    }  
    //Devolvemos el valor de premio  
    return premio;  
}
```

4. Comprobación de las dos últimas cifras de un premio principal. Si el número de cupón comparte sus dos últimas cifras con uno de los tres premios principales.

```
public static int ComprobarUltimas(int cupon, NumPremiado[] premiados) {  
    //Creamos la constante del premio a las 2 últimas cifras  
    final int PREMI2ULTIMAS = 1000;  
    //Creamos la variable premio  
    int premio = 0;  
    //Creamos una variable con las 2 últimas cifras de cupón  
    int ultimas2 = cupon % 100;  
    //Miramos si coinciden las 2 últimas cifras con las 2 ultimas de un premio mayor  
    for (int i = 0; i <= 2; i++) {  
        if (ultimas2 == premiados[i].numero % 100) {  
            premio = PREMI2ULTIMAS;  
            nompremi = "las 2 últimas cifras de un premio mayor";  
        }  
    }  
    //Devolvemos el premio  
    return premio;  
}
```

5. Comprobación de pedreada. Si el número **premiado** es cualquier otro número de los 1807 ganadores no perteneciente a los premios principales.

```
public static int ComprobarBombo2(int cupon, NumPremiado[] premiados) {  
    //Creamos la constante de todos los premios mayores que hay - 1  
    final int TOTALGUANYADORS = 12;  
    //Creamos las variable del programa  
    boolean ganador = false;  
    int premio = 0;  
    //Miramos si el cupon corresponde a la pedrea  
    for (int i = TOTALGUANYADORS; i < premiados.Length && ganador == false; i++) {  
        if (cupon == premiados[i].numero) {  
            premio = premiados[i].premio;  
            ganador = true;  
            nompremi = "la pedrea";  
        }  
    }  
    //Devolvemos el premio  
    return premio;  
}
```



6. Comprobación de última cifra del gordo (reintegro). Como extra, si el número de cupón, sea premiado o no, salvo el gordo, comparte su última cifra con el gordo, obtiene el reintegro.

```
public static int ComprobarReintegro(int cupon, NumPremiado[] premiados, int premioactual) {  
    //Creamos la constante con el valor del premio  
    final int PREMIREINTEGRO = 200;  
    //Creamos la variable premio  
    int premio = 0;  
    //Miramos si el último numero coincide con el último numero del gordo  
    if (cupon % 10 == premiados[0].numero % 10) {  
        //Miramos que el numero no sea el gordo  
        if (cupon != premiados[0].numero) {  
            //Assignamos el valor de la constante a premio  
            premio = PREMIREINTEGRO;  
            //Si todavía no habia ningún premio ponemos la vairable nompremi con reintegro  
            if (premioactual == 0) {  
                nompremi = "el reintegro";  
            }  
        }  
    }  
    //Devolvemos la variable premio  
    return premio;  
}
```

- Comprobación:

```
Opción seleccionada: 1  
Introduce tu número: 23517  
Has ganado la pedrea con una cantidad de 100 al décimo
```



Contenido del proyecto

- Proyecto java Lotería de Navidad
 - Código ejecutable
 - Tests de casos de prueba
 - JavaDoc
- Ficha técnica
- Sumario de funcionamiento de premios de la lotería
- Autoevaluaciones de cada miembro del equipo.
- Coevaluaciones de otros miembros.



Infografía

Premios de la lotería: <https://www.loterias.com/loteria-de-navidad/como-funciona>

En la versión del programa 1.0. El Bombo principal cuenta con un total de 1807 premios, subdivididos en un premio principal de 4.000.000€, un segundo premio de 1.500.000€, un tercer premio de 500.000€, dos cuartos premios de 200.000€ cada uno, ocho quintos premios de 60.000€ por premio, y todos los premios restantes son pedreadas de 1.000€ cada una.

En el caso de aproximarse una unidad por encima o por debajo de un premio primero, segundo o tercero, el cupón ganará 20.000€, 12.500€ y 9.600€ correspondientemente como premio por acercamiento.

Los cupones que entren en la centena de un premio primero, segundo, tercero o cuarto, o aquellos que compartan sus dos últimas cifras con el primer premio, el segundo y el tercero, obtendrán un premio de 1.000€.

Como compensación adicional, todo premio, salvo el gordo, que comparta su última cifra con la cifra final del Gordo, obtendrá un plus de 200€.