

ISW-Programación I

Prof. Efrén Jiménez

Est:

Acuña Paniagua Mario

Campbell Rojas Freivin

Proyecto: I-Creación de app

Fecha: 2 Julio del 2014

**Alicia se rió: “No tiene sentido intentarlo”, dijo: “No se**

**puede creer en cosas imposibles” “Yo más bien diría que**

**es cuestión de práctica” dijo la reina. “Cuando yo era**

**joven, practicaba todos los días durante media hora.**

**Muchas veces llegué a creer en más de seis cosas**

**imposibles antes del desayuno”**

**Lewis Carroll, “Alicia en el país de las maravillas”.**

Tabla de contenido

[INTRODUCCIÓN 4](#_Toc392232157)

[Descripción del problema 5](#_Toc392232158)

[Solución del problema 5](#_Toc392232159)

[Análisis de Resultados 8](#_Toc392232160)

[Conclusión final 8](#_Toc392232161)

[Recomendaciones para el Profesor 9](#_Toc392232162)

[Bibliografía 10](#_Toc392232163)

# INTRODUCCIÓN

Este trabajo se realizó con el propósito de que las estudiantes apliquen los conocimientos básicos e intermedios obtenidos en clase, con el fin de aprender más sobre programación en lenguaje java y su contenido

Una aplicación que simule un juego de estrategia. La implementación del proyecto se realizó por medio de un IDE llamado Netbeans. En él se desarrolló la aplicación con sus respectivas características a líneas de código y además la implementación en el uso de swing de Java para su finalización.

# Descripción del problema

Creación de una aplicación desde cero la cual no se basó en una versión anterior de algún otro juego o código ya establecido, esta aplicación fue creada con el propósito de corregir la problemática donde las personas usan juegos donde su capacidad de razonamiento no es explotada.

La aplicación tendrá una interfaz en cual la cantidad de jugadores será como mínimo y un máximo 2 usuarios, tendrá una partida post-configurada donde el mapa de juego tendrá sus propias configuración de igual forma se pueden modificar cuando el usuario le vea conveniente, ya iniciado el juego los jugadores deberán imaginar un campo de batalla donde su estrategia será la que les de la victoria sobre su contrincante durante la partida deberán realizar su táctica antes de atacar y aceptar el blanco para tener la victoria.

# Solución del problema

Demostrando que la programación orientada a Objetos con interfaz gráfica en java no es simple. Es compleja para el programador. Lo que queremos lograr mediante este proyecto la interacción rápida y sencilla de los usuarios con estas aplicaciones las cuales son herramientas fáciles de utilizar. De igual forma ayudar en la creatividad y razonamiento estratégico de los usuarios.

Para realizar el programa indagamos periódicamente que con el IDE de JAVA: NETBEANS, podíamos realizar una interfaz limpia y detallada para una aplicación la cual se podría implementar el uso de imágenes, etiquetas, botones, menú, Scroll y Slider´s.

Mencionar que la declaración correcta de cada clase, variable, método e instancia en el código del programa llego a la finalización correcta.

De igual forma con la creación de métodos que citaremos algunos de ellos y explicaremos a continuación para la elaboración de la App.

1. **public Jugador(boolean pTipoJugador)** = Con este método indicamos que el jugador será aliado o enemigo de igual forma el nombre del jugador, avatar, puntaje, cantidad de partidas ganadas, jugadas y perdidas.
2. **public Tablero(int pFilas, int pColumnas)** = Método con el que se genera el alto y el ancho del tablero del juego con la cantidad de filas y columnas que el usuario disponga.
3. **public boolean setBarco(int pFila, int pColumna)** = Método utilizado para establecer un barco dentro del terreno de juego.
4. **public boolean hayBarco(int pFila, int pColumna)** = Metodo para verificar si existe un barco en la posición indicada.
5. **public boolean[][] getMatriz()** = Metodo para obtener la matriz.
6. **public boolean destruirBarco(int pFila, int pColumna)** = Metodo para destruir un barco y eliminarlo de la matriz
7. **public AreaJuego()** = método que abrirá la ventana del área de juego.
8. **private void MnI\_SalirActionPerformed(java.awt.event.ActionEvent evt)** = Método para salir de la aplicación al presionar Alt+F4 o al seleccionar la opción salir en el menú.
9. **private void verificarAtaque(java.awt.event.MouseEvent evt)** = Evento que captura cuando un botón ha sido seleccionado en la matriz de botones y se procede a verificar y ejecutar el ataque al elemento seleccionado
10. **private void ejecutarAtaque(int pFilaAtaque, int pColumnaAtaque)** = Método que lleva a cabo el ataque y modifica tanto la matriz grafica (coloca un barco hundido o un vacío como imagen) como la matriz de objeto Tablero (cambia el valor del true a false, en caso que hubiera un barco, indicando que este fue hundido)
11. **private void generarJuego(int filasDe1Usuario, int columnasDelTablero, int cantidadDeBarcos)** = Método utilizado para crear el terreno de juego y los barcos que se colocaran en el de forma aleatoria. Seguido de ello, generara un boolean para asignar quien comienza el juego.
12. **private void generarTerreno(int pFilas, int pColumnas)** = Método utilizado para generar cada una de las casillas del tablero gráfico.
13. **private void generarBarcos(int pFilas, int pColumnas, int pCantidadDeBarcos)** = Método utilizado para crear de manera aleatoria la posición en la que se ubicaran los barcos, tanto aliados como enemigos, en el tablero de juego.
14. **private void MnI\_ConfiguracionActionPerformed(java.awt.event.ActionEvent evt)** = Método utilizado para mostrar la ventana de configuración.
15. **private void MnI\_EditJueg1ActionPerformed(java.awt.event.ActionEvent evt)** = Método utilizado para crear un nuevo jugador Aliado al seleccionar el del menú.
16. **private void MnI\_EditJueg2ActionPerformed(java.awt.event.ActionEvent evt)** = Método utilizado para crear un nuevo jugador Enemigo al seleccionar el elemento del menú.
17. **private void formWindowClosing(java.awt.event.WindowEvent evt)** = Método utilizado para verificar si el usuario desea cerrar la ventana una vez presionado el botón X.
18. **private void MnI\_NuevoJuegoActionPerformed(java.awt.event.ActionEvent evt)** = Método que verifica si los jugadores no han sido creados luego de presionar juego Nuevo en el menú de opciones.
19. **private void Lbl\_Player1PhotoMouseEntered(java.awt.event.MouseEvent evt)** = Muestra las estadísticas del usuario al pasar el puntero sobre la imagen del jugador Aliado.
20. **private void Lbl\_Player1PhotoMouseExited (java.awt.event.MouseEvent evt)** = Evento que desvanece la ventana estadísticas cuando el mouse sale de la imagen del aliado.
21. **private void Lbl\_Player2PhotoMouseEntered (java.awt.event.MouseEvent evt)** = Muestra las estadísticas del usuario al pasar el puntero sobre la imagen del jugador Enemigo.
22. **private void Lbl\_Player2PhotoMouseExited(java.awt.event.MouseEvent evt)** = Evento que desvanece la ventana estadísticas cuando el mouse sale de la imagen del Enemigo.
23. **private void Btn\_AplicarActionPerformed(java.awt.event.ActionEvent evt)** = Metodo que le permite al usuario crear una nueva configuración para crear una nueva partida con dicha información.
24. **private void ChkB\_ConfigActionPerformed(java.awt.event.ActionEvent evt)** = Metodo que le permite al usuario crear una nueva configuración para crear una nueva partida sin límites.
25. **private void verificarEntrada(java.awt.event.KeyEvent evt)** = Metodo que impide que se ingresen letras en la configuración personalizadas.
26. **public void setEstadisticas(String pTitulo, int pGanadas, int pJugadas, int pPerdidas)** = Metodo para establecer los valores que se mostraran en la ventana estadística.

# Análisis de Resultados

En la mayoría de casos, se obtuvieron resultados satisfactorios y se recogieron un conjunto de recomendaciones que se tuvieron en cuenta para mejorar los resultados el proyecto.

Los resultados finales del programa indican que se siguieron todos los pasos a evaluar en el proyecto. Como resultado tangible del mismo se ha obtenido una aplicación totalmente funcional que cumple completamente los requisitos que había solicitado para su elaboración, estando preparada para su implantación inmediata, de forma que se haga un uso real de la misma.

# Conclusión final

Habiendo desarrollado y concluido este proyecto se han obtenido las siguientes conclusiones:

El objetivo principal de este trabajo es desarrollar una app que simule un juego Battleship. Después de completar la implementación y la ejecución de las pruebas en el sistema se puede declarar que el objetivo del proyecto ha sido alcanzado. El presente proyecto ha demostrado que es perfectamente posible crear un ambiente virtual para la elaboración de un Juego o una aplicación para pc.

En la realización de este proyecto, se ha tenido total libertad en la toma de decisiones, tanto en la forma de trabajar, de elegir las tecnologías a utilizar, para diseñar y dar solución a las necesidades que se plantean para el programa sin duda es una ventaja para la conclusión del proyecto y así su finalización con los resultados esperados.

# Recomendaciones para el Profesor

1. Con Respecto al Proyecto

Fue un gran reto el trabajo asignado se debía comprender muy bien la materia vista en clase para su elaboración, conllevo a una inversión de tiempo grande la cual la planificación y estructura del proyecto debía de estar clara para la conclusión del mismo, sin el trabajo en equipo y con lluvias de ideas no claras de ser un reto pasaría a ser algo imposible de elaborar.

1. Con respecto a el Lenguaje de Programación

Queda claro recalcar que en nuestras horas de estudio en la materia de Programación I, el lenguaje único a programar ha sido Java por lo cual se toma como una excelente decisión por parte del profesor el cual calificara el código del proyecto tomando en cuenta la materia vista en clase.

1. Con respecto a las Explicaciones

No cabe duda que la Programación es un trabajo el cual nos lleva a Mejorar el estilo de vida a muchas personas (empresas, negocios, oficinas) lo que queremos recalcar es que la explicación es algo para aclarar nuestras dudas y fue correcta desde nuestro punto de vista, pero debemos dejar claro que la investigación sobre los temas vistos es con lo cual nuestro vacío es llenado.

1. Con respecto a las Horas Consulta

Opinamos en su totalidad en las ocasiones que necesitamos respuesta alguna duda presentada antes, durante y después de la elaboración del proyecto fue aclarada correctamente por el Profesor.

1. Con respecto en el Tiempo de elaboración del Proyecto

Fueron invertidas horas en la el investigación y recolección de datos para su creación pero con la creación de un cronograma el cual nos facilidad las horas de reunión y trabajo en un orden limpio y apegado al cronograma anteriormente citado se podría llevar la elaboración sin ningún problema en el tiempo determinado para su entrega.

# Bibliografía

Núñez de Balboa. (2002) Piensa En JAVA, Madrid, España, PEARSON EDUCACION. SA.

Paul J. y Harvey M. (2008) Cómo programar en Java (7ma ed.) México DC

México, PEARSON EDUCACION. SA.

Introducción a Netbeans Programación 2 (2010-2011). <http://ocw.udl.cat/enginyeria-i-arquitectura/programacio-2/continguts-1/1-introduccioi81n-a-netbeans.pdf>.

Programación de Java Blog. (2013). [http://jedicerocool.blogspot.com/2013/01/ajustar-una-imagen-al-tamano-de-un.html#.U7atNPmBOoh](http://jedicerocool.blogspot.com/2013/01/ajustar-una-imagen-al-tamano-de-un.html%23.U7atNPmBOoh).

FaceYourManga (2008-2011). <http://www.faceyourmanga.com/>.

Universidad de Guadalajara Centro Universitario de Ciencias Exactas e Ingenierías (2006-20014). <http://proton.ucting.udg.mx>.

Aprendiendo de Java Paso a Paso (2013).

<http://aprendiendojavapasoapaso.blogspot.com/2012/05/libros-pdf.html>.

Guía a la redacción en el estilo APA, 6ta edición Biblioteca de la Universidad Metropolitana Sylvia Zavala Trías, MLS <http://www.cibem.org/paginas/img/apa6.pdf>.