**INSTITUTO TECNOLÓGICO DE COSTA RICA**

Escuela de Computación

Programacion orientada a objetos

Grupo 20

II Tarea programada en Java

Juego de ajedrez

Integrantes:

Daniel Solís Méndez

Jose Roberto Arguedas

Melvin Alonso Elizondo Pérez

Nombre del profesor:

Carlos Arias Rodriguez

II Semestre 2013

**Resumen Ejecutivo**

Juego de ajedrez

Este documento presenta una visión desde otra perspectiva del desarrollo de un proyecto, en este caso el desarrollo de un software, donde se explica de forma breve los puntos relevantes de algunos procesos de todo proyecto, como lo son entender, de muy buena forma, cual es el problema a resolver, con esto se fija una meta y se puede entonces empezar a pensar el diseño interno(código y estructura) y externo(interfaz).

También muestra cuales fueron las técnicas para la solución del problema, como la filosofía de divide y vencerás, que consiste en ver un problema grande en varios problemas pequeños para poder así, buscar una solución efectiva y con esto dar resolución al problema mayor.

Dentro de los puntos importantes cabe destacar que no todo proyecto es

perfecto, y muchas veces no sale exactamente como se imagina, pero eso

claramente se puede utilizar de motivación e intentar acercarse lo mayor

posible a esa idea, utilizando siempre la buena comunicación de todas las

partes, aprovechando todas las buenas ideas que surgen y tomando siempre en cuenta que todo lo que se utilice tiene una razón y para esto se prueba en cada parte cual puede ser la solución más óptima, o en el

ambiente poder discernir cual metodología es mejor de acuerdo a las características del problema.

El desarrollo de un proyecto como este siempre deja beneficios, como lo es todo el aprendizaje que se puede obtener, también muchos de los problemas que se pudieron presentar en el trabajo son ahora una herramienta más al conocimiento cuando en otro momento se enfrente alguna situación similar.

Como objetivo principal queremos afianzar todos los conocimientos vistos en clase, y poder aplicarlos a un problema en especifico; así como también investigar para abordar más información relevante que nos ayude en el planteamiento y realización de la tarea programada.

En el siguiente escrito describiremos como fue surgiendo el desarrollo, las ideas que se fueron utilizando y desechando, los errores, y los obstáculos que se encontraron en el trayecto, así como la forma de saltar esos obstáculos para proseguir adelante con el trabajo.

**Introducción:**

El siguiente proyecto fue motivado por la Escuela de Computación del Instituto Tecnológico de Costa Rica como una iniciativa clara para incrementar el conocimiento de los estudiantes de dicha carrera en el curso de Programacion Orientada a Objetos.

El objetivo es utilizar algunos de los conceptos y técnicas de programación

estudiados durante el curso, por ejemplo:   
Implementacion de herramientas para el diseno de clases, el uso de las herencias, contructores, patrones y demas.  
 Así como investigar con más detalle otras herramientas, servicios y técnicas para resolver adecuadamente el proyecto.

La siguiente documentación hace referencia a una tarea programada que debió ser desarrollada utilizando el lenguaje de programación Java y su libreria para el manejo de aplicaciones web Java Servlet.

Se intenta con este documento crear un registro de todo el proceso por el cual pasa el proyecto mostrando problemas, soluciones y recomendaciones para el mismo.

**Análisis del problema:**

En esta sección se explicará lo que es la solución general que se toma para solucionar este proyecto el cual consiste en llevar de forma valida los acontecimientos típicos en una partida normal de ajedrez, como lo son movimientos por turno de cada jugador y los movimientos básicos que tiene cada pieza ya sea para desplazarse o “comer” otra pieza.

Lo primero pasos fueron pensar en como se podían solucionar los diferentes problemas que se presentan a la hora de jugar Ajedrez (movimientos y demás), con esto primero que surge fue establecer los nombres de las clases y métodos que se podrían utilizar para darle solución.

Estas clases son:

Jugador:

Esta clase mantiene los nombres de los jugadores activos durante la sesion que se crea al iniciar el juego, la partida no podria empezar si no se tienen los nobres de los jugadores ademas de esto sirve para en un futuro saber que jugador está en que turno.

Pieza:

En esta clase se define las piezas su ruta, su color, y que tipo es si es alfil, caballo entre otras. Esta clase cuenta con varios constructores, ya que de ser necesario por temas de implementacion es mejor solo tener un contructor sin parametros que pueda crear una pieza “vacia” para llenar la mayoria de espacios en el tablero.

Contiene los métodos de definir pieza, setRuta, sets y gets de x e y entre otros necesarios para las necesidades de las otras clases.

Ajedrez:

En esta clase basicamente donde se manejan las demas clses del juego, ya que una instancia de Ajedrez crea en su contructor las otrsas clases dependientes para poder llevar el juego, lleva la parte para la aplicación web, los request, los response, mostrar el tablero e importar algunos archivos css y js necesarios en la página web.

Partida:

Esta se utiliza para poder referenciar los objetos jug1, jug2 y tabjuego.

Tablero:

Esta clase se usa para hacer el tablero lógico del juego que consta de una matriz 8x8 donde en el constructor se definen todas las piezas en el orden adecuado para iniciar la partida los cuales son: la torre, el caballo, el alfil, la reina, el rey, y los peones y piezas vacias.

Tambien por la parte de los metodos se tienen algunos getters de TipoPieza, ColorPieza, RutaPieza y otros setters. Todo esto para modificar o consultar la matriz logica del juego.

Experto en ajedrez:

Esta clase lleva una de las responsabilidad más fuertes de la partida ya que es la que verifica si los movimientos que se hacen en el tablero son validos, también se da cuenta cuando hay jaque o jaque mate.

Tiene métodos como:

validarMovimento:

Este método recibe como parámetro dos pares ordenados los cuales corresponden a las coordenadas en la matriz donde se ubican las piezas con respecto al “click” que de la persona en la pagina.

Es el método principal de la clase, el cual verifica por medio de las coordenadas las posiciones y posibles movimiento dependiendo del tipo de pieza en el tablero lógico para después mostrarlo en el tablero visual en el explorador.

Se verifica si la posición de destino esta vacía, si el tipo de pieza según corresponda puede realizar ese movimiento además de verificar si el destino puede remover una pieza del contrario, de ser posible entonces se asigna una Pieza vacía a la casilla de destino y se coloca la pieza de la jugada en la casilla correspondiente.

ValidarMovimientoHorizonal:

Este metodo sirve como auxiliar para el metodo de validar el movimiento, en el cual recorre la fila según los valores de los pares ordenas de y1 y y2. Para esto se utilizan las validaciones correspondientes con los condicionales y apartir de esto se hace un ciclo uniciando des el primer punto hasta llegar al seleccionado como destino, el cual verifica que no se encuentre ninguna ficha de mismo color y que los espacios que recorra de ser posible si es mas de uno, que esten vacios. En el caso de que las pieza de destino sea una del color contrario se revisa que no se haya saltado mas de una pieza del otro color y que ademas no quiera saltar a un espacio vacio si hay alguna pieza en medio.

ValidarMovimientoVertical:

Este metodo sigue la misma metodologia que se utiliza en el metodo validarMovimientoHorizontal ya que en este caso se invierten los papeles de los puntos “x” y “y”, ya que para moverse verticalmente en una matriz los puntos que no experimentan cambios son los “y” de la misma tomando “x” como filas y “y| como columnas.

ValidarDiagonalArriba:

validarDiagonalAbajo

Jaque

verificar

JaqueMate

Digamos que esta clase es como la mente del juego.

Con estas clases se no hizo muy fácil llegar a la solución ya que al tener estas clases, así sabíamos que métodos iban en cada clase.

**Conclusiones**:

El proyecto logra finalizar exitosamente, funcionando como debe y siendo visualmente agradable. Se sacan grandes experiencias y nuevos conocimientos que nos servirán de aquí en adelante en nuestra carrera y vida profesional.

Este proyecto generó bastante agrado por ser algo nuevo y diferente a lo que conocíamos hasta ahora.

(...agregar mas....)

En general fue una experiencia enriquecedora que nos ayudo a esculpir un poco más los profesionales que queremos llegar a ser.

**Recomendaciones:**

* Creer que se puede lograr, no tenerle miedo solo por lo que dicen las personas.
* Investigar en internet, especialmente en <http://stackoverflow.com/>. Esta página cuenta con mucha información importante para el desarrollo de cualquier aplicación.
* Dividir el trabajo en partes pequeñas y fáciles de obtener, ir haciéndolo de manera progresiva y uniendo las partes alcanzadas.
* Llevar el código y la documentación de la mano, es mucho mejor para no acumular trabajo al final, y para comprender mejor lo que estamos haciendo y hemos hecho.
* Tomar decisiones en equipo, es mejor así se pueden dialogar ideas y escoger la mejor.
* Cada vez que se visita una página web en busca de información, y se da en el punto, inmediatamente añadir la dirección web a las referencias del trabajo.
* Lograr la funcionalidad del proyecto primero, y luego revisar excepción y restricciones del juego/aplicación.

**Referencias:**

**(Seguir formato APA como el ejemplo)**

Ejm:  
System call. (2013, October 18). In *Wikipedia, The Free Encyclopedia*. Retrieved 04:15, October 24, 2013, from <http://en.wikipedia.org/w/index.php?title=System_call&oldid=577757465>

*(...Referencias...)*

**Anexos:**

(...Adjuntar anexos...)