



ESCUELA POLITECNICA NACIONAL



FACULTAD DE INGENIERÍA DE SISTEMAS

ESTRUCTURA DE DATOS (SIC334)

GRUPO: No. 7

DEBER: Proyecto Final

Alumnos: Carlos Chicaiza

Danny Díaz

Rafael Vinueza

PROFESORA: Dra. María Gabriela Pérez H.

FECHA DE ENTREGA: 06-08-2018



**ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN**

Objetivos:

Al final del curso con la entrega del proyecto de Estructura de Datos lo que se desea lograr es que el alumno desarrolle habilidades para planificar, organizar, así como también capacidad de análisis y de síntesis sobre el estudio de los algoritmos y su implementación, de cada estructura de datos estudiada.

- Reforzar los temas estudiados en clase, permitiéndoles trabajar con todos los algoritmos vistos en la solución de problemas del entorno.
- Potenciar el trabajo colaborativo y en equipo, así como incentivar la investigación formativa del estudiante.
- Aprender a expresar, tanto de forma oral como escrita conocimientos, procedimientos, resultados e ideas en su campo.
- Que el alumno desarrolle habilidades de abstracción y capacidad para planificar y organizar, así como también potenciar la capacidad de análisis y de síntesis sobre los contenidos de la materia.
- Encontrar la manera de describir la solución a un problema plasmando los resultados obtenidos en algún tipo de informe, que en este caso será, un poster científico.

Marco teórico:

El juego se encuentra en la etapa Alpha, lo que quiere decir que está en desarrollo y puede existir uno que otro bug, para la solución de diferentes problemas y bugs hemos implementado estrategias vistas en clase para dar una solución.

LISTAS. - Una lista constituye una colección de elementos, el orden entre estos se establece por medio de punteros o índices.

El personaje “jugable” tiene como habilidad un viaje temporal que le permite recuperar sus estadísticas y posición, para implementar esto usamos las listas que van almacenando la información el jugador tanto sus estadísticas como posición, almacenan la información cada segundo, y cuando el jugador desee usar esta habilidad se recorrerá el arreglo 6 posiciones atrás y la información que haya en dicho estado se la pasara al jugador como estado actual. Se implementó algunas listas como Colas y otras como Pilas según la necesidad.

PILAS. - son estructuras lineales en las que se puede agregar o quitar elementos únicamente por uno de los extremos.

COLA. - Es una estructura lineal de datos e la que los nuevos elementos se introducen por un extremo y los ya existentes se eliminan por el otro.

ÁRBOLES. - Son estructuras jerárquicas que se aplican sobre una colección de elementos u objetos llamados nodos, uno de los cuales es conocido como raíz. Además se crea una relación de parentesco entre los nodos dando lugar a términos como padre, hijo, hermano, antecesor, sucesor, ancestro, etc.

Los arboles fueron usados para el desarrollo del 3 en raya que necesita generar todos sus posibles estados, para esto se genera una estructura que recuerda a la transformación de árboles generales



**ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN**

a árboles binarios, estos árboles nos permiten explorar todas las posibilidades del juego y escoger la mejor opción.

FUNCION HASH. - Son funciones usadas para la búsqueda por transformación de claves que permite aumentar la velocidad de búsqueda sin necesidad de tener elementos ordenados. Cuenta con la ventaja de que el tiempo de búsqueda es independiente del número de componentes del arreglo.

En la interfaz llamada logeo se usa las funciones Hash donde cada usuario tendrá una key que se usará en una función hash, y esta información se ira almacenando en una pila, de esta forma garantizamos que no se mezclen las cuentas de diferentes usuarios.

GLOSARIO

NPC: Un personaje no jugador o personaje no jugable, es un término a menudo abreviado con la sigla PNJ (o NPC por su sigla en inglés), que es un personaje controlado por el director de juego o DJ (también llamado GM, del inglés game master) en el curso de una partida de rol. Cuando esta definición se refiere a los videojuegos de rol, un PNJ es generalmente parte del programa, y no controlado por un humano. El concepto opuesto es el personaje representado por una persona o PJ (personaje jugador).

TRIGGER: Se les llama triggers o disparadores, son eventos que se ejecutan cuando se cumplen condiciones determinadas. La utilidad principal de un trigger es mejorar la administración de una aplicación, ya que no requieren que un usuario los ejecute.

KEY LISTENER: nos provee los métodos para controlar los eventos del teclado. Al implementar la interface debemos implementar también los métodos que la componen, veamos cuales son y su funcionamiento básico:

keyPressed(KeyEvent e): Se ejecuta cuando el usuario presiona una tecla.

keyReleased(KeyEvent e): Se ejecuta cuando el usuario libera una tecla

keyTyped(KeyEvent e): Se ejecuta cuando el usuario presiona una tecla, pero solo cuando la tecla corresponde a caracteres, teclas especiales como F1, F2 entre otras no son identificadas.

SPRITE: Se trata de un tipo de mapa de bits dibujados en la pantalla de ordenador por hardware gráfico especializado (MSX, Atari 400/800, Commodore 64 y Commodore Amiga fueron unos de los pocos ordenadores que soportan sprites reales) sin cálculos adicionales de la CPU. A menudo son pequeños y parcialmente transparentes, dejándoles así asumir otras formas a la del rectángulo.

COLISIÓN (JUEGO 2D): Decimos que dos objetos colisionan cuando uno de ellos se sobrepone a otro en este momento debemos disparar una "señal" y tratar dicha colisión en consecuencia, impidiendo el movimiento si es un sólido, restando vida si es un enemigo, etc. Todo dependerá del tipo de juego en este artículo nos vamos a centrar en detectar dichas colisiones.

HITBOX: Una hitbox es una técnica invisible comúnmente utilizada en los videojuegos para la detección de colisiones en tiempo real. Es sigue un punto en un objeto visible (como un modelo o un sprite), aunque las formas circular y esférica son también comunes. Es usual que objetos



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN

animados dispongan de varias “hitboxes” en cada una de sus partes móviles, para así asegurar una buena precisión durante el movimiento.

TILE: Una tesela o tile es la parte gráfica de cada videojuego que puede ser utilizada para completar partes de un fondo por medio de un set de teselas o tileset. Útil para generar mapas, ya que el conjunto de estos se los puede representar en una matriz de símbolos

REBOOT: Un reinicio en los conceptos de ficción se refiere al relanzamiento de una historia presentando una inflexión de la serie, no forzosamente siguiendo la historia previa sino sólo conservando los elementos más importantes, los que se consideren mejores o más funcionales para darle continuidad y, en consecuencia, ignorando los que resulten más irregulares o poco gustados, todo bajo el parámetro subjetivo de los propietarios del concepto quienes tomarán decisiones presuntamente de acuerdo a lo que deseen ver los seguidores del concepto en cuestión, aunque comenzando todo de nuevo, desde el inicio, con ideas más “frescas” o reinterpretadas.

ALPHA: Es una versión temprana de un proyecto que contiene elementos básicos y esenciales, generalmente tiene suficiente contenido para dar una idea de lo que será el proyecto.

BUG: Un bug es un error o un defecto en el software o hardware que hace que un programa funcione incorrectamente. Bug es español significa insecto. Según se dice, el primer bug de ordenador fue un insecto real, descubierto en 1945 en Harvard, una polilla atrapada en la calculadora Mark II Aiken Relay hizo que la máquina entera se apagara.

Desarrollo del proyecto



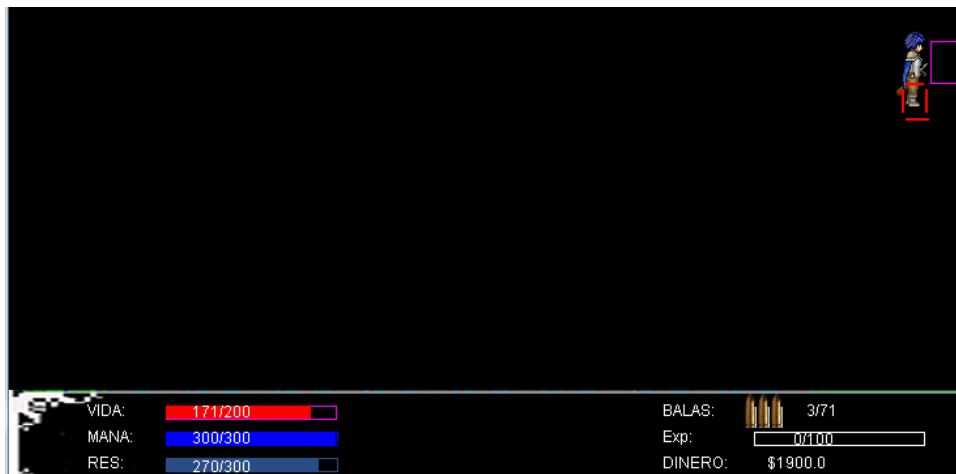
El videojuego principal consiste en un jugador el cual puede moverse a través de distintos mapas y debe eliminar a los enemigos que aparecen en estos. La implementación de esto consiste en colocar al jugador en el centro de la ventana y mediante el uso de fórmulas matemáticas calcular el desfase que genera para que el mapa detrás de él se mueva dependiendo de la tecla que presione el usuario. Se han colocado rectángulos colisionables en los objetos del mapa donde el jugador no es capaz de atravesar ni tampoco los enemigos.



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN



Se ha implementado una clase pistola que contiene Balas las cuales el jugador puede disparar a los enemigos y hacerles daño. Esto se controla mediante el uso de la clase de java Rectangle, ya que la bala y el enemigo están formados por rectángulos se usa la función intersects que comprueba si un rectángulo choca con otro y cada rectángulo tiene un tag específico ya sea enemigo, edificio, naturaleza, etc. A partir de este tag si una bala colisiona con un tag enemigo esta desaparecerá y el enemigo perderá cierta cantidad de vida.

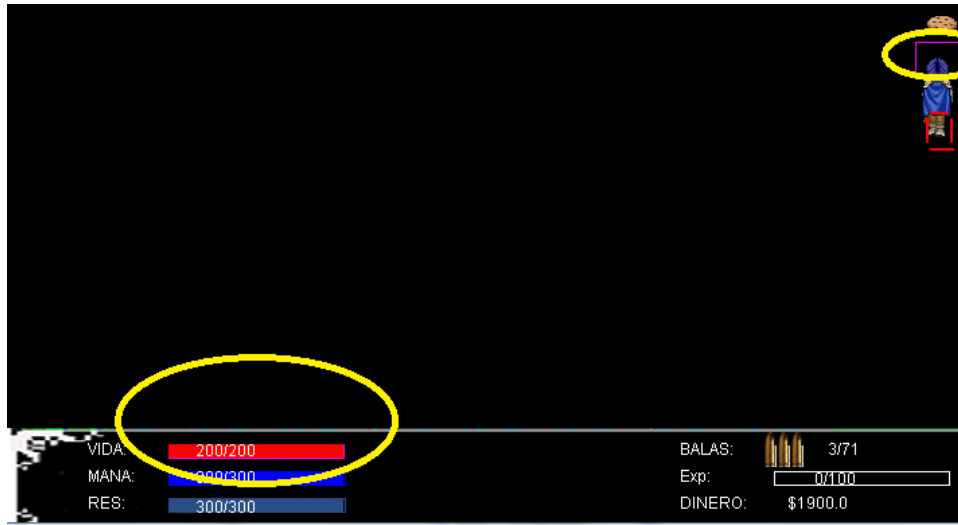


El jugador tiene una pequeña interfaz debajo del mapa donde se muestran sus estadísticas más importantes como son su vida actual, su mana actual, su resistencia, su experiencia, la cantidad de balas que posee, su dinero, etc. a esta pantalla se le denomina HUD y está presente a lo largo de todo el trayecto del personaje.

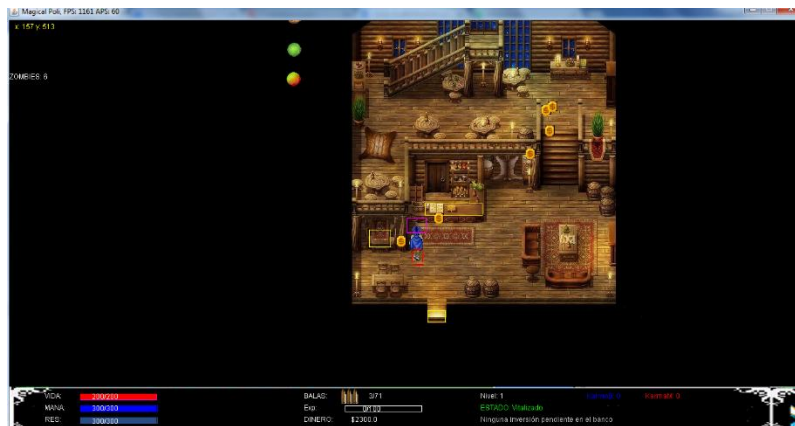


ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN

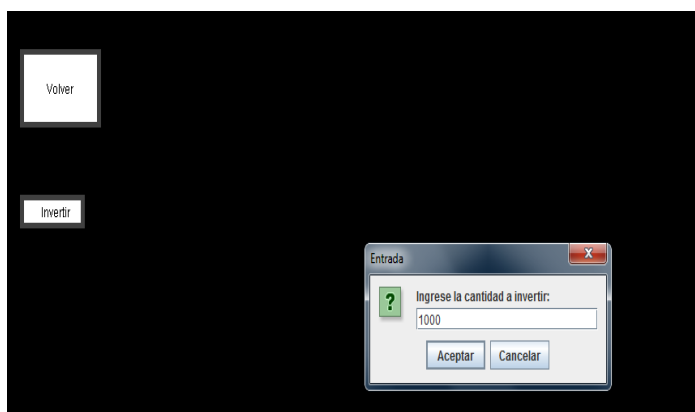
Comida



Se han implementado objetos en el mapa que cambian las estadísticas del personaje, por ejemplo una galleta que al ser consumida aumenta la vida del jugador en 20 siempre y cuando este no tenga su barra de vida llena.



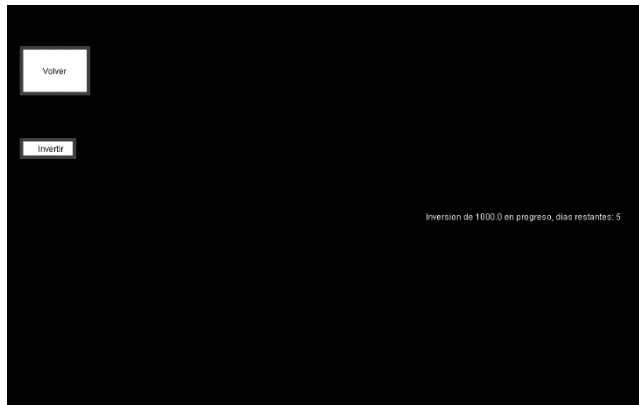
Dentro de uno de los mapas, se ha colocado una casa de inversiones y tienda de objetos donde el personaje puede invertir cierta cantidad de dinero por un determinado tiempo y obtener intereses al finalizar este periodo.



Al ingresar a la casa de inversiones el personaje tiene 2 botones, uno para regresar al estado anterior y otro para realizar una inversión monetaria de la cantidad que el usuario desee, siempre y cuando el jugador posea esa cantidad de dinero.

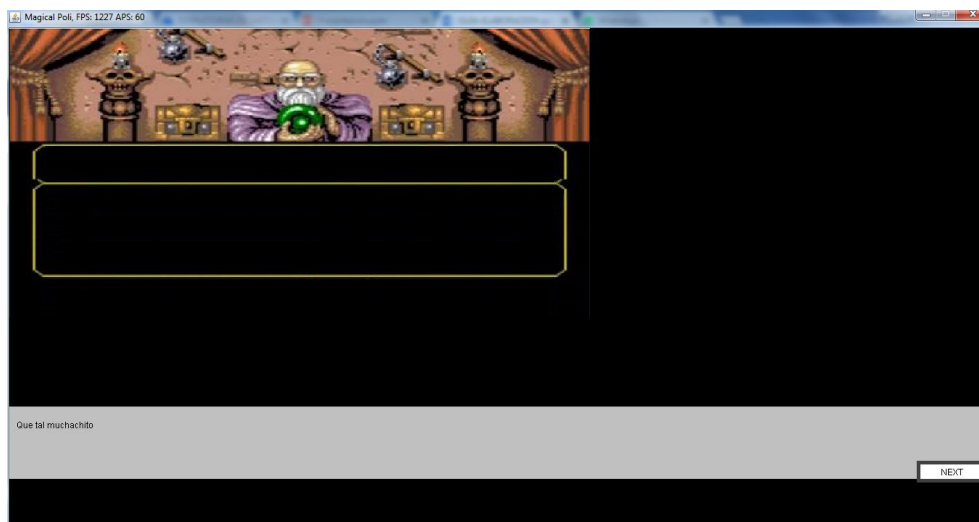
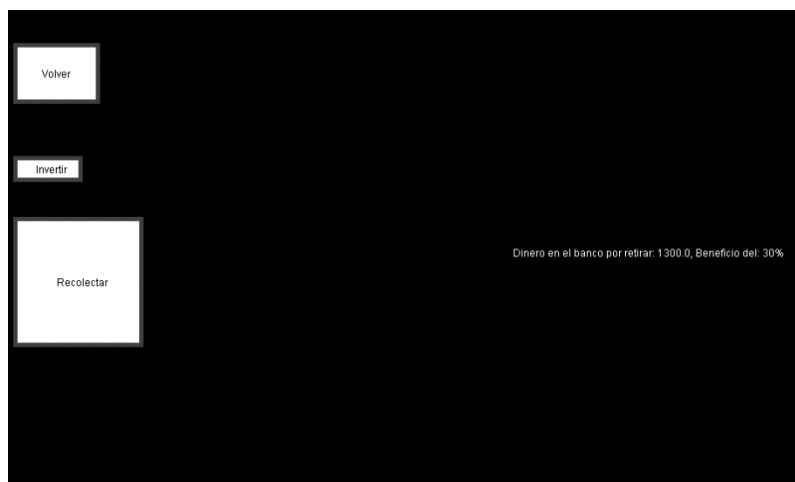


**ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN**



Una vez realizada la inversión el jugador deberá esperar 7 días dentro del juego (7 segundos en este caso) para obtener su inversión mas un interés agregado por invertir.

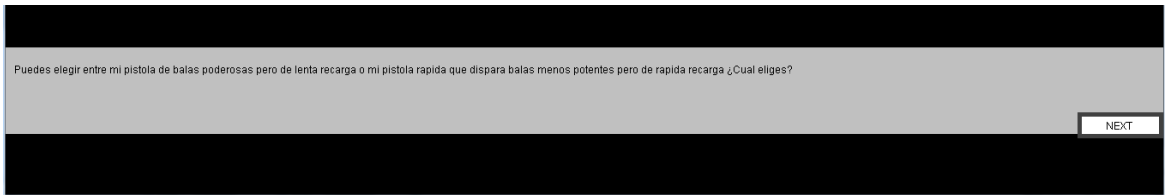
Al finalizar este lapso de tiempo el jugador puede sacar e vuelta su dinero con un beneficio del 30%, esto puede ser útil para cuando el jugador tiene ahorrada una gran cantidad de dinero.



La tienda sirve para que el jugador pueda comprar objetos que le ayuden en su viaje, aquí se ha implementado uno de los puntos clave del proyecto que es una decisión importante. La primera vez que el jugador entre a la tienda se activara este dialogo.



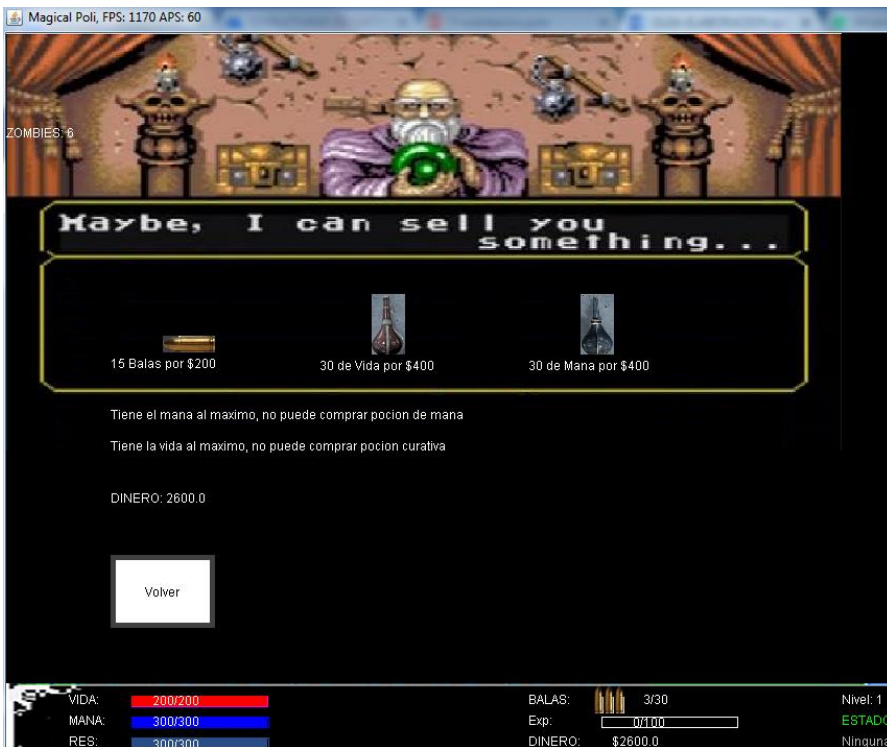
ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN



El jugador podrá elegir cambiar su pistola por una con mayor daño pero con mayor tiempo de espera al recargar o una pistola con menor daño pero una recarga mucho más rápida y mayor cantidad de balas en su inventario.



Se despliegan 2 botones para que el jugador pueda tomar una decisión y esta se guardara.



Una vez realizada la elección se desplegará el menú de tienda normal, donde el jugador podrá comprar distintos objetos si tiene el dinero suficiente. Pero mediante el poder implementado de retroceso en el tiempo mediante el uso de pilas, este podrá retornar al pasado antes de tomar su decisión y cambiarla si así lo desea, esto podrá hacerlo siempre y cuando no haya salido del edificio, caso contrario la decisión que tomo se quedará guardada.

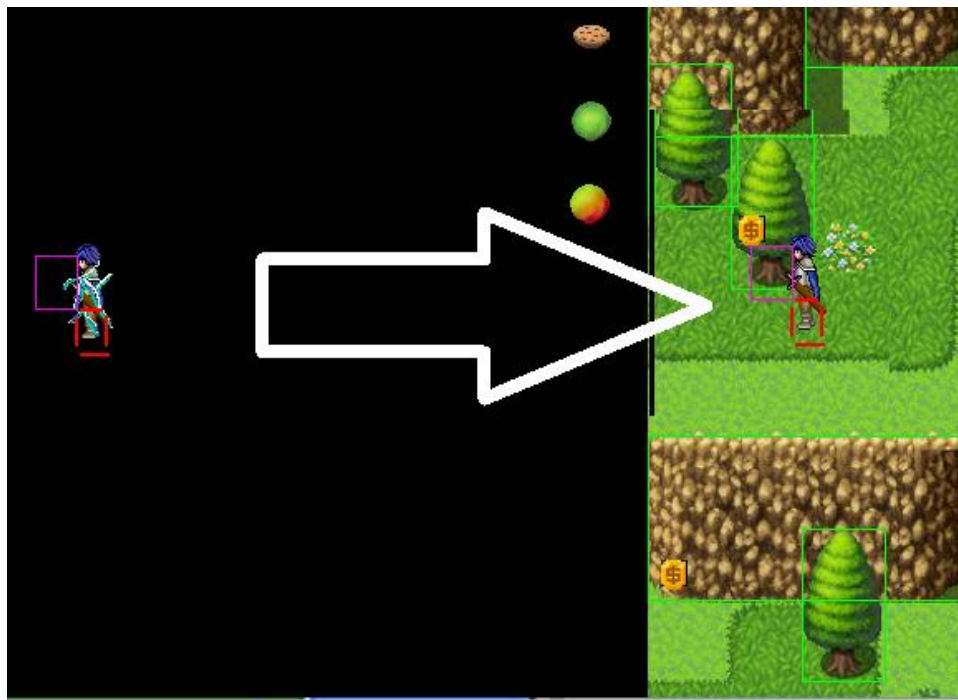


ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN



También se ha implementado un sistema de subida de niveles, a medida que el personaje gane experiencia subirá su barra de experiencia y al subir de nivel sus estadísticas como vida, mana y resistencia aumentaran.

PILA



El poder de viaje en el tiempo es uno de los puntos clave del proyecto, mediante el uso de una pila se almacena los estados del jugador en cada momento, si este activa el poder de viaje en el tiempo estos estados son expulsados de la pila regresándolo hacia atrás en el tiempo con una animación y un sonido de regreso.

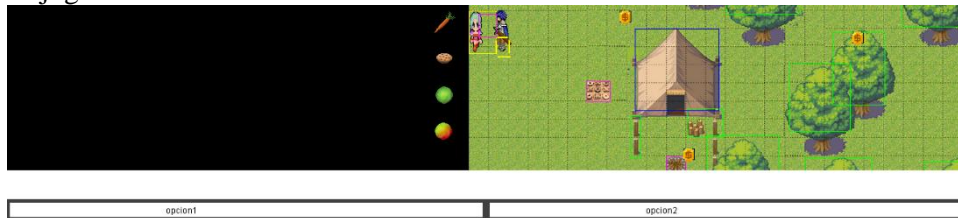


ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN



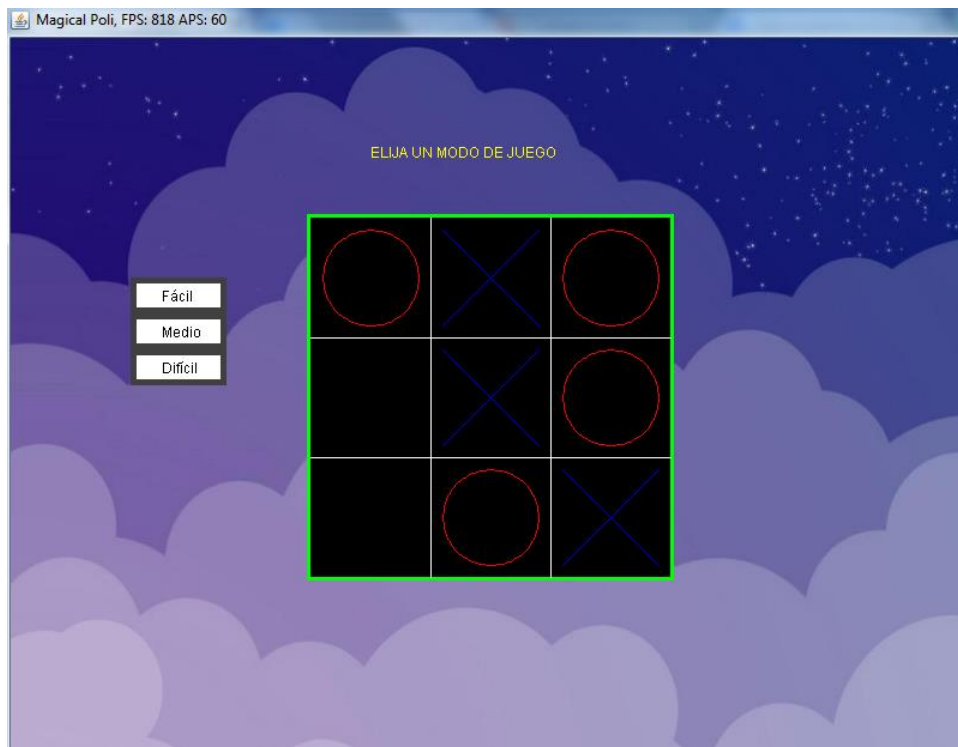
También se ha implementado personajes que no agreden al jugador denominados NPC con los cuales el jugador puede interactuar. Al hacerlo desencadenará un dialogo en el cual podrá tomar una importante decisión.

El jugador decide si salvarle la vida o no.



Se desplegaran dos botones en los cuales el jugador puede elegir, dependiendo de si elige salvarla o no, el karma del jugador se inclinara a ser bueno o malo.

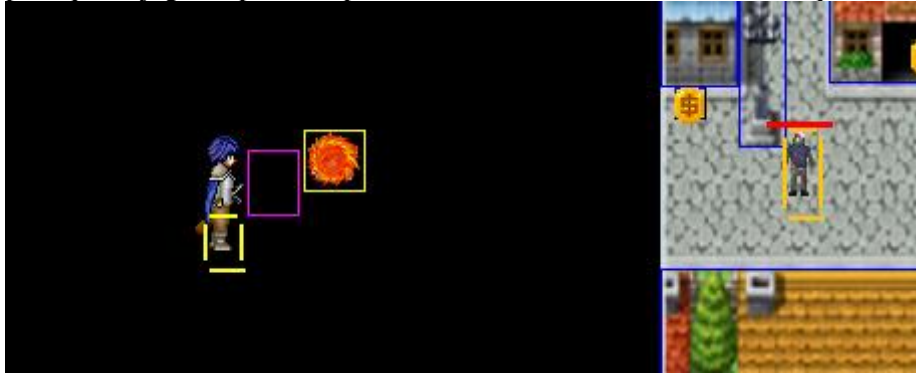
ARBOLES





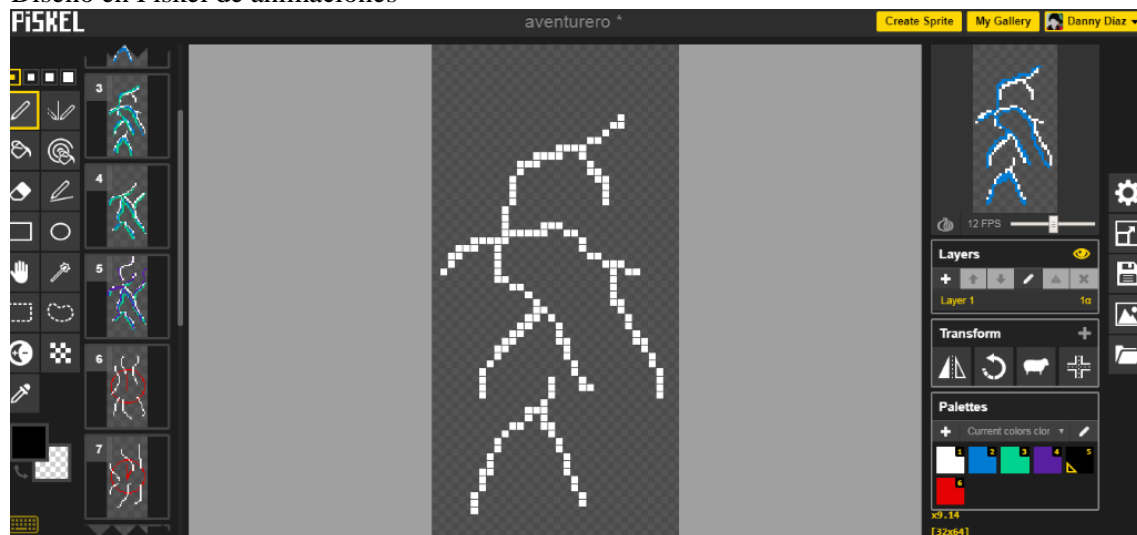
ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN

La implementación de un juego tres en raya o tic tac toe es otro punto clave de nuestro proyecto, mediante el uso de árboles se analizan todas las posibles jugadas que el jugador pueda realizar y en base a esto elige la rama más favorable para la máquina. Se ha desarrollado en 3 dificultades para que el jugador pueda experimentar sus habilidades contra la máquina.



Se ha añadido un poder de bola de fuego que el jugador puede arrojar consumiendo su mana, este poder cumple una función parecida a la bala bajándole vida al enemigo y se puede usar siempre y cuando el jugador posea el mana suficiente.

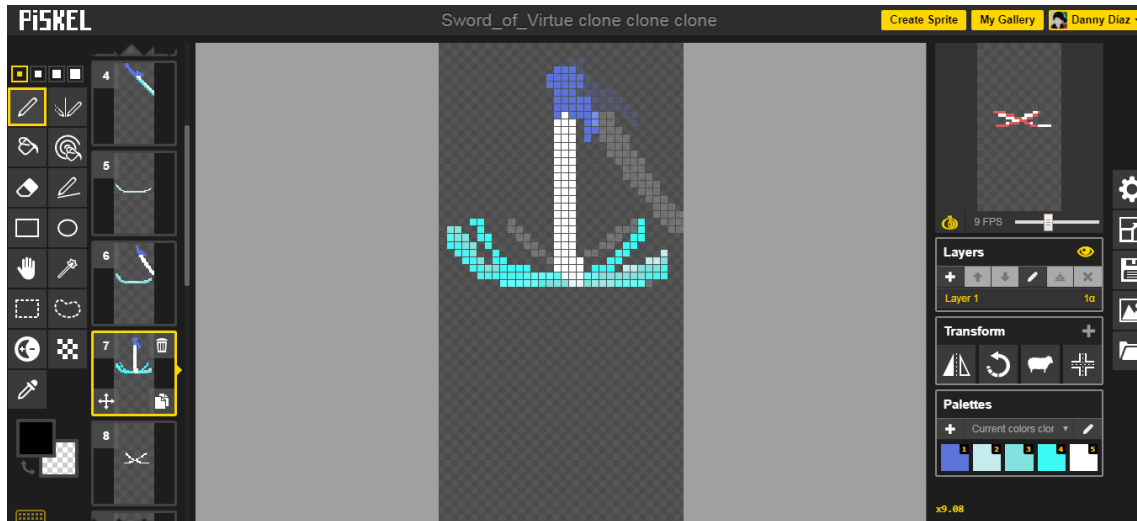
Diseño en Piskel de animaciones



Para el diseño de la animación de retorno en el tiempo se usó Piskel, una página online gratuita que permite el diseño de trazos o dibujos dependiendo de la habilidad del usuario.



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN



Dentro de esta página web se pueden diseñar varios trazos y concatenarlos para reproducirlos y simular una animación. Mediante esto se diseñó la animación de retorno en el tiempo del jugador y la animación de la espada del jugador.

HASH

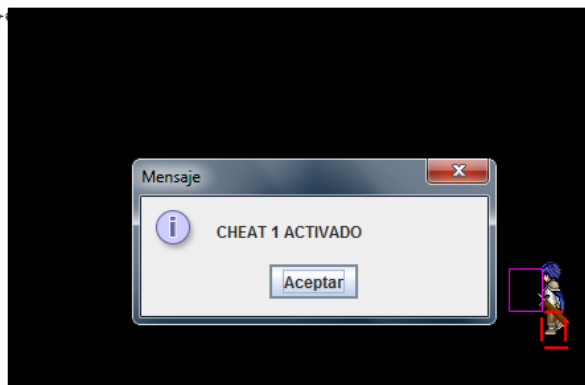
Otro punto clave en nuestro proyecto fue el desarrollo de un sistema de logueo con encriptación de contraseñas mediante el uso de funciones hash. Para ello la contraseña y el usuario se guardan en un archivo, la contraseña se transforma a código ASCII y a este valor se le aplica una función hash, para ser almacenado en la capa1, luego se realiza la función re-hash para generar los valores de las capas 2, 3 y 4. Estas capas se guardan en una pila de manera segura y cuando una persona desea loguearse se extraen los valores de la pila y a la contraseña ingresada se realiza el mismo proceso de transformación a ASCII y encriptación hash, debe coincidir con las 4 capas almacenadas en la pila para acceder.



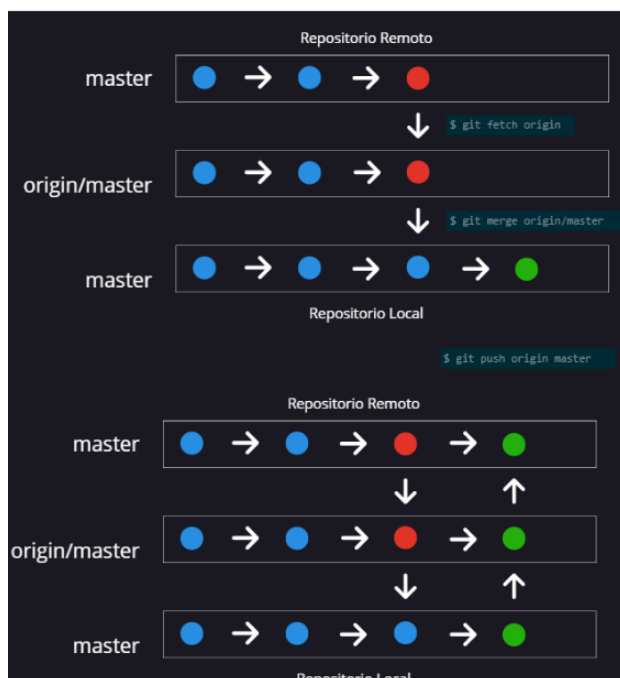
LISTAS

```
87->  
87->  
87->87->  
87->87->87->  
87->87->87->68->  
87->87->87->68->68->  
87->87->87->68->68->68->  
87->87->87->68->68->68->83->  
87->87->87->68->68->68->83->83->  
87->87->87->68->68->68->83->83->83->  
87->87->87->68->68->68->83->83->83->65->  
87->87->87->68->68->
```

Lista de Teclas, solo se guardan si coinciden parcialmente o totalmente con los cheats



Finalmente, mediante el uso de listas se realizó la implementación de trucos secretos con los cuales el jugador puede obtener bonificaciones secretas. Se lee la secuencia de teclas aplastadas y si coincide con uno de los valores de la lista se ejecuta el truco, pero si durante la secuencia falla alguna tecla la lista se vacía y se regresa al estado inicial donde se debe insertar la secuencia correcta otra vez.



Durante todo el transcurso de nuestro proyecto se usó la herramienta git, esta aplicación permite que varios colaboradores diseñen un proyecto conjunto realizando cambios a todo momento y así no acumular versiones inútiles ni realizar uniones de código muy conflictivas. Mediante esta herramienta cada colaborador sube sus avances mediante el comando git push, cuando otro colaborador desea subir sus avances debe fusionar su contenido con el de sus compañeros mediante el comando git merge, encontrar los conflictos de código que resulten y una vez solucionados se procede a volver a subir al almacenamiento remoto. Mediante esta técnica de desarrollo de versiones se ahorra tiempo al unir el trabajo de todos de manera dinámica siempre y cuando exista la debida comunicación entre todos los colaboradores.



**ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN**

Conclusiones:

- Las pilas permiten almacenar varios estados en el tiempo de tal forma que podemos volver a un estado inicial. Esto permitió el "viaje en el tiempo".
- Las listas guardan información de una manera casi limitada. Están acotadas por la cantidad de memoria RAM y los niveles de caché de un computador.
- Los arboles pueden dar un vistazo al futuro en juegos "cerrados" es decir con un límite de jugadas posibles.
- Para desarrollar una IA que pueda jugar ajedrez se requiere un cómputo bastante significativo es por eso que técnicas como el reinforcement learning son mucho más útiles que crear árboles de decisión.
- Tomar decisiones que afectan el futuro del juego afectan personalmente al usuario de forma que se "sumerge" mucho más a la aventura

- Se ha logrado la implementación de un videojuego en java mediante el uso de estructuras de datos tales como pilas, arboles, funciones hash y otras.
- Mediante el uso de arboles se logrado implementar una inteligencia artificial para el juego de 3 en raya o tictac toe capaz de realizar jugadas analizando el estado del tablero en cada movimiento almacenado en sus nodos internos.
- Mediante el uso de pilas se ha logrado implementar un sistema de retroceso en el tiempo durante el transcurso del videojuego principal, se guarda a cada momento los estados del jugador en la pila y al activar el poder de retorno en el tiempo esta expulsa los estados retornando a la posición donde se encontraba al inicio de la pila.
- Mediante el uso de funciones hash se ha logrado crear un sistema de encriptación el cual transforma la contraseña de un usuario a binario y a este resultado se le aplica una función hash modulo 4 veces quedando resguardado tras 4 capas.

Recomendaciones:

- Se recomienda el uso de la plataforma git para un buen desarrollo de proyectos eliminando la necesidad de crear varias versiones y un mejor desarrollo de código en equipo.
- Se recomienda el uso de matrices para la representación del mapa mediante cuadrados definidos, al cargar simplemente el mapa si modelarlo resulta difícil la implementación de algoritmos para el movimiento e inteligencia de enemigos y NPCs.
- Se recomienda el uso de la función System para usar la tarjeta gráfica de la computadora y mejorar el rendimiento.
- Para un mejor desarrollo de videojuegos se recomienda el uso de una IDE de desarrollo diseñada específicamente para este objetivo como GameStudio ya que esto facilita la implementación de interacciones y la creación de elementos dentro de un videojuego

Bibliografía:

- Cairo Osvaldo, Guardati Silvia, Estructura de datos, ISBN: 970105908-5, Tercera Edición 2006. [Online]. Disponible en:



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN

https://drive.google.com/file/d/0B_XimPSyUDLcM2ZtU3VCVHhLUUk/edit?pref=2&pli=1

- Estructura de Datos en Java, JOYANES Luis, ZAHONERO Ignacio. ISBN: 9788448173937, Edición 2008. Disponible en: ftp://soporte.uson.mx/PUBLICO/02_ING.SISTEMAS.DE.INFORMACION/estructura_datos/Estructura%20de%20datos%20en%20java%20Joyanes%201ed.pdf
- Estructura de Datos, GARCIA Iván, GARCIA Magariño, ISBN: 8445419358 ISBN-13, Edición 2011.
- Johnsonbaugh, R. (2005). Árboles de juegos. En R. Johnsonbaugh, *Matemática Discreta, sexta edición* (págs. 429-434). México: PEARSON EDUCACIÓN.
- Tomás Blanco. *PARA JUGAR COMO JUGÁBAMOS*. 2003. ISBN 84-87339-44-1 Ed. John Wiley & sons.
- Cairo Osvaldo, Guardati Silvia, Estructura de datos, ISBN: 970105908-5, Tercera Edición 2006. [Online]. Disponible en: https://drive.google.com/file/d/0B_XimPSyUDLcM2ZtU3VCVHhLUUk/edit?pref=2&pli=1
- Estructura de Datos en Java, JOYANES Luis, ZAHONERO Ignacio. ISBN: 9788448173937, Edición 2008. Disponible en: ftp://soporte.uson.mx/PUBLICO/02_ING.SISTEMAS.DE.INFORMACION/estructura_datos/Estructura%20de%20datos%20en%20java%20Joyanes%201ed.pdf
- Estructura de Datos, GARCIA Iván, GARCIA Magariño, ISBN: 8445419358 ISBN-13, Edición 2011.
- Curso de creación de un juego de rol 2d en java: <https://www.youtube.com/watch?v=qa6GA5p9nQ0>
- Curso de GIT de CódigoFacilito: <https://www.youtube.com/watch?v=zH3I1DZNovk>
- http://ariadne.cti.espol.edu.ec/sidweb36/sidweb_2/3928/958030/formato_informesQGI.pdf
- <http://www.ing.unp.edu.ar/electronica/asignaturas/ee016/tutoriales/informes/CONFECCL%20D3N%20DE%20INFORMES.htm>