

ESCUELA COLOMBIANA DE INGENIERÍA

PROGRAMACIÓN ORIENTADA A OBJETOS

POOAgeOfWar

DESCRIPCIÓN



El Objetivo del proyecto **POOAgeOfWar** es desarrollar una aplicación para una versión simplificada de Age Of War entre dos jugadores: Un usuario y el computador o, un usuario y otro usuario.

El juego consiste en destruir el Castillo enemigo, evitando ser destruido a su vez. El escenario de juego es un escenario bi-dimensional, a cada extremo del tablero deben estar ubicados los castillos de los oponentes y en medio habrá un espacio que los bandos de cada grupo deberán cruzar para intentar destruir a su enemigo.

El Castillo

Cada castillo tiene, nivel de vida, nivel de defensa, almacén de oro, y nivel de desarrollo:


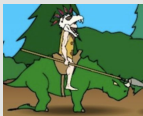
- **Nivel de vida:** Cuando llega a cero el jugador pierde su castillo y el juego. La vida se pierde cuando un soldado enemigo ataca el castillo, se puede recuperar una porción cuando se avanza a un nuevo nivel de desarrollo.
- **Nivel de defensa:** Se desgasta antes que la vida. Se pierde cuando un soldado enemigo ataca el castillo, se recupera totalmente cuando se avanza a un nuevo nivel de desarrollo.
- **Almacén de oro:** Determina la riqueza del castillo, será útil para convocar soldados o poderes especiales. Cada 10 segundos 15 monedas de oro son añadidas al almacén.






- **Nivel de desarrollo:** Representa una mejora del castillo que desbloquearán nuevos soldados y mejorará las defensas. Para avanzar de nivel, el castillo deberá usar una gran suma de dinero de su almacén de oro.

Niveles de desarrollo		
	Nombre	Descripción
	Edad de Piedra	Es el primer nivel de desarrollo y todos los jugadores comienzan con ese nivel.
	Edad Media	Son necesarias 180 monedas para avanzar a este nivel de desarrollo, incrementa la vida del castillo al 50% y la defensa a un 60%.
	Edad Industrial	Son necesarias 270 monedas para avanzar a este nivel de desarrollo, incrementa la vida del castillo al 60% y la defensa a un 70%.
	Edad Moderna	Son necesarias 360 monedas para avanzar a este nivel de desarrollo, incrementa la vida del castillo al 70% y la defensa a un 80%.

Soldados

Cada soldado tiene un nivel de salud, un nivel de defensa, un nivel de ataque y un costo. El nivel de defensa lo pierde a medida que es atacado por otro enemigo, una vez que lo pierde, su nivel de vida es consumido, el nivel de ataque determina el daño que puede causar a sus contrincantes. Finalmente el costo, es el precio que cada jugador debe pagar para convocar un soldado en particular.

Soldados			
	Nombre	Edad	
	Hombre de piedra	Edad de Piedra	<ul style="list-style-type: none"> • 1 puntos de ataque. • 10 puntos de salud. • 0 puntos de defensa. • 5 monedas para su creación. • Al morir devuelve 1 moneda al castillo.
	Dinosaurio de piedra	Edad de Piedra	<ul style="list-style-type: none"> • 2 puntos de ataque. • 15 puntos de salud. • 3 puntos de defensa. • 10 monedas para su creación.

			<ul style="list-style-type: none"> Al morir devuelve 1 punto de nivel de defensa al castillo.
	Hombre de acero	Edad Media	<ul style="list-style-type: none"> 5 puntos de ataque. 10 puntos de salud. 5 puntos de defensa. 15 monedas para su creación.
	Caballero	Edad Media	<ul style="list-style-type: none"> 7 puntos de ataque. 15 puntos de salud. 10 puntos de defensa. 25 monedas para su creación. Al morir devuelve 2 punto de nivel de defensa al castillo.
	Lancero	Edad Industrial	<ul style="list-style-type: none"> 15 puntos de ataque. 10 puntos de salud. 5 puntos de defensa. 20 monedas para su creación. Al morir devuelve 2 moneda al castillo.
	Cañon	Edad Industrial	<ul style="list-style-type: none"> 30 puntos de ataque. 1 puntos de salud. 10 puntos de defensa. 20 monedas para su creación. Al morir devuelve 3 punto de nivel de defensa al castillo.
	Marine	Edad Moderna	<ul style="list-style-type: none"> 35 puntos de ataque. 20 puntos de salud. 5 puntos de defensa. 40 monedas para su creación. Al morir devuelve 3 moneda al castillo.
	Tanque	Edad Moderna	<ul style="list-style-type: none"> 50 puntos de ataque. 100 puntos de salud. 50 puntos de defensa. 90 monedas para su creación. Al morir devuelve 4 punto de nivel de defensa al castillo.

Jugador

El jugador computador puede ser de diferentes perfiles; entre ellos actualmente tenemos:

- **Ingenuo:** Realiza sus movimientos al azar.

- **Tacaño:** Solo usa el soldado más barato y su principal objetivo es mejorar el castillo al máximo nivel de desarrollo.

Campo de Batalla

El campo de batalla es un espacio bi-dimensional. Cada castillo esta ubicado en los costados del campo. En medio hay un espacio por medio del cual los diferentes soldados caminarán con el objetivo de llegar al otro costado para atacar al castillo enemigo; sin embargo si durante el camino se encuentran con un soldado enemigo, deberán destruirlo antes de poder continuar.

REQUISITOS FUNCIONALES

- Permitir crear una instancia del juego dada la dirección de una archivo.
- Permitir identificar el jugador por nombre.
- Permitir seleccionar el tipo de perfil del jugador computador.
- Presentar permanentemente el estado del campo de batalla.
- Presentar permanentemente la información completa del estado de cada uno de los jugadores.
- Permitir que el usuario realice sus jugadas indicando posibles errores. Si el usuario se equivoca se debe notificar con un mensaje amable o con sonido.
- Los movimientos son ejecutados en cualquier momento por cualquier jugador.
- Decidir cuando termina la batalla y comunicar la causa.
- Permitir terminar la batalla en cualquier momento.

REQUISITOS DE DISEÑO

DE EXTENSIÓN

- Permitir generar nuevas versiones de la aplicación que incluyan otros tipos de soldados (catapultas, arqueros, mosqueteros etc.), niveles de desarrollo (Edad romana, Edad Futurista) y otros perfiles de jugadores computador (nuevo perfil con estrategia).

DE VISUALIZACIÓN

- Los icons de cada representación gráfica de cada elemento son de elección propia.
- Los soldados, los castillos, los elementos especiales y las celdas deben ofrecer una representación gráfica detallada que permita explorar el estado del juego.

DE MANEJO DE EXCEPCIONES

- Deben definir mínimo una nueva clase excepción para manejar las excepciones propias.
- Los métodos correspondientes a las acciones deben lanzar una excepción si la acción solicitada no es válida explicando claramente su causa.
- Cuando ocurra una excepción no esperada o una propia grave se debe escribir esta información en el log de errores para los programadores y terminar la ejecución del mismo.

REQUISITOS DE ENTREGA

Revisión inicial		Capa Presentación Boceto de la interfaz gráfica Capa Aplicación Diagrama de clases	A par S12:
Version 1 Presentación	Edad de piedra. Usuario vs usuario. Actividad de los elementos.	Capa Presentación Boceto de la interfaz gráfica Diagrama de clases Código Capa Aplicación Diagrama de clases Diagramas de secuencia Código Programa Código pruebas JUnit	A Par S14: Jueves 16 de Noviembre
Version 2 Persistencia	Edad de piedra. Usuario vs. usuario Actividad de los castillos y de los elementos.	Capa Presentación Boceto de la interfaz gráfica Diagrama de clases Código Capa Aplicación Diagrama de clases Diagramas de secuencia Código Programa Código pruebas JUnit	A Par S16: Jueves 30 de Noviembre
Version 3	Todas las edades Usuario vs. máquina	Capa Presentación Boceto de la interfaz gráfica Diagrama de clases Código Capa Aplicación Diagrama de clases Diagramas de secuencia Código Programa Código pruebas JUnit	A evaluador Inicial Miercoles 6 de Diciembre Final Miercoles 13 de Diciembre
Competencia	Es requisito para participar en la competencia que el equipo se haya presentado a todas las revisiones de pares y que todas las entregas del proyecto hayan sido aprobadas. El equipo ganador tiene 5.0 en la nota del tercer tercio.		Viernes 15 de Diciembre