

jfx-pokemon



Integrantes:

Camilo Campaz Jiménez - A00370258

Camilo González Velasco - A00370263

Daniel Esteban Jaraba Gaviria - A00368822

Johan Stiven Ricardo Sibaja - A0037069

DOCENTE: Juan Manuel Reyes

REQUERIMIENTOS FUNCIONALES	3
REQUERIMIENTOS NO FUNCIONALES	5
DIAGRAMA DE CLASES	6
DISEÑOS DE PRUEBAS UNITARIAS	7

REQUERIMIENTOS FUNCIONALES

El programa debe estar en la capacidad de:

1. Comenzar una nueva aventura con un nuevo entrenador pokémon, con un nombre que será proporcionado por el jugador y un avatar elegido por el mismo, que puede ser masculino o femenino.
2. Cargar los datos de una partida anterior.
3. Almacenar la tabla de posiciones en el torneo y actualizarla cada que participe un nuevo entrenador.
4. Acceder a la información de cualquier pokémon por medio de la pokédex. Ya sean los pokemons que tiene guardado el entrenador o la de un pokemon salvaje.
5. Guardar la información de la partida en cualquier momento, mientras no esté en batalla.
6. Organizar la información de los pokemons registrados en la pokédex.
7. Cambiar el volumen de la música en el menú de configuración.
8. Ver la tabla de posiciones del modo torneo.
9. Listar en orden los pokemons que el entrenador tiene en sus pokeballs.
10. Cambiar de mapa mientras va avanzando en el mapa principal.
11. Permitir que el jugador haga uso de los NPCs y todas sus funciones.
12. Enfrentar de manera automática los personajes que participan en el torneo frente a el jugador actual.
13. Aumentar el nivel de los pokemons y el nivel de fuerza, energía y defensa.
14. Implementar una mecánica de pelea por turnos por cada enfrentamiento que tenga el entrenador.
15. Mostrar un pequeño tutorial del juego cuando se inicie por primera vez en el juego o cuando el usuario quiera verlo.
16. Guardar información de un pokémon en la pokédex.
17. Reproducir música de manera continua en cualquier pantalla hasta que el juego se cierre.
18. Capturar algunos pokémon y guardarlos en su colección de pokemons.
19. Comprar y/o actualizar cualquier objeto que exista en la tienda.

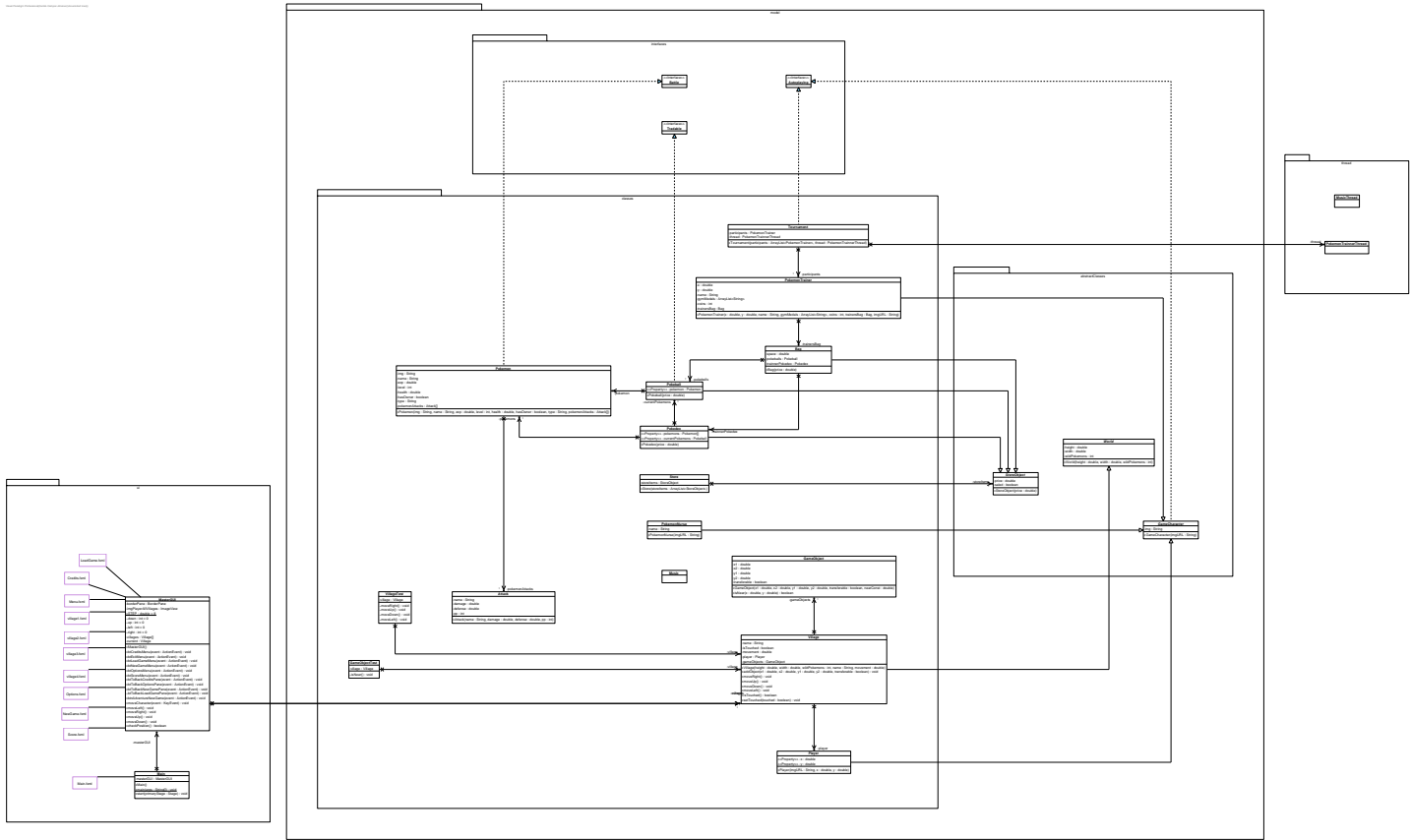
20. Intercambiar objetos en la tienda siempre y cuando este sea intercambiable.
21. Reconocer los objetos en los mapas y que el jugador interactúe correctamente con ellos; es decir, si es objeto sólido (como casas , montañas , rocas, etc) el jugador debe chocar con él.

REQUERIMIENTOS NO FUNCIONALES

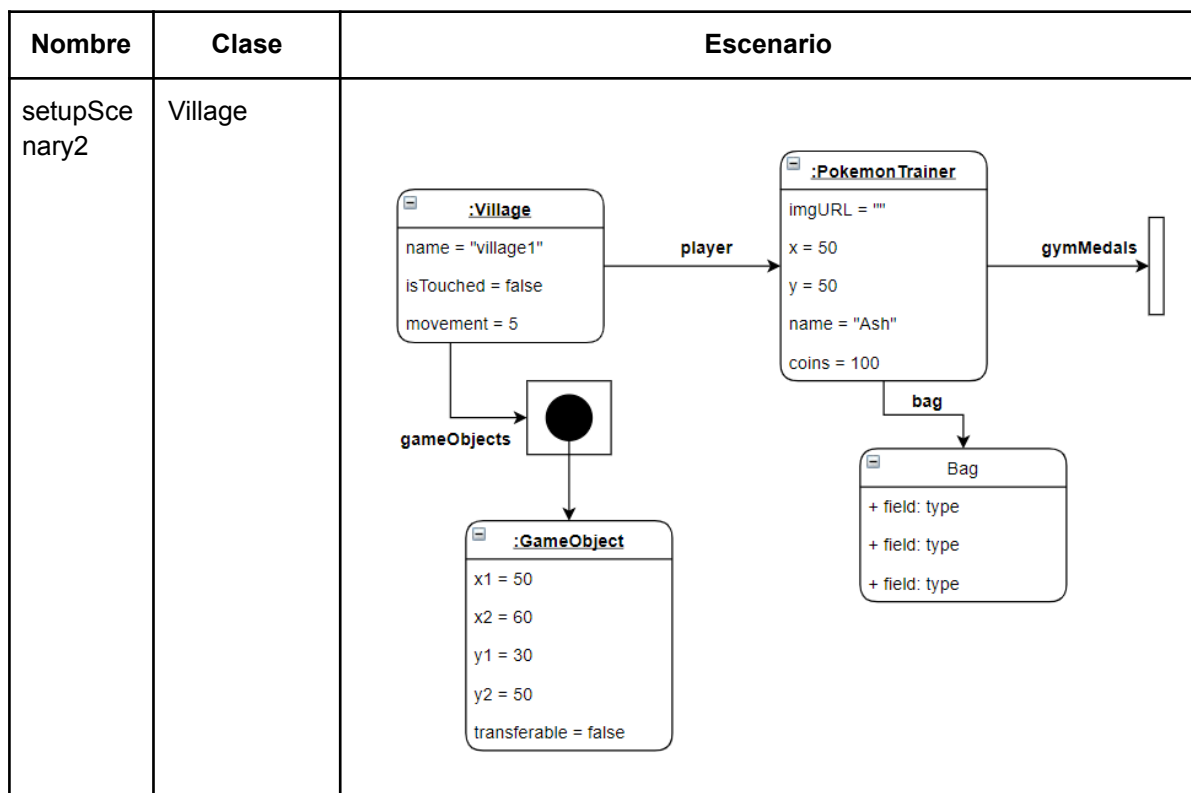
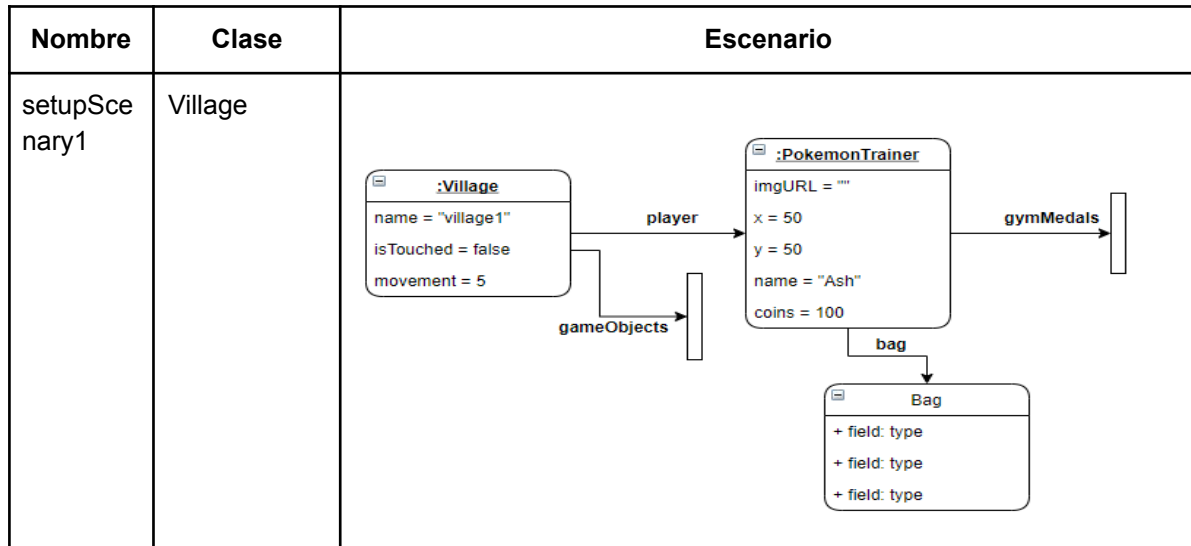
El programa debe cumplir con las siguientes condiciones:

1. Impedir que el usuario salga de los límites del mapa con el movimiento del jugador.
2. Mostrar un texto flotante cada vez que el jugador se acerque a un NPC.
3. Generar un mínimo de 3 objetos de tipo Pokemon por cada mapa del videojuego, a excepción de zonas de intercambio y/o seguras.
4. Cambiar el nombre de cualquier Pokemon que el jugador haya atrapado.
5. Limitar la cantidad de Pokemons que se generan en un mismo mapa.
6. Cancelar el diálogo con cualquier NPC.

DIAGRAMA DE CLASES



DISEÑOS DE PRUEBAS UNITARIAS



Objetivo de la Prueba: Verificar el correcto funcionamiento del sistema de movimiento en dirección ascendente para el personaje modificando de manera correcta las coordenadas del mismo

Clase	Método	Escenario	Valores de Entrada	Resultado
Village	moveUp	setupScenary1	<i>ninguno</i>	se modifica el atributo del objeto PokemonTrainer que posee la clase, restando movement a a su atributo en Y

Objetivo de la Prueba: Verificar el correcto funcionamiento del sistema de movimiento en dirección descendente para el personaje modificando de manera correcta las coordenadas del mismo

Clase	Método	Escenario	Valores de Entrada	Resultado
Village	moveDown	setupScenary1	<i>ninguno</i>	se modifica el atributo del objeto PokemonTrainer que posee la clase, sumando movement a a su atributo en Y

Objetivo de la Prueba: Verificar el correcto funcionamiento del sistema de movimiento en dirección hacia la derecha para el personaje modificando de manera correcta las coordenadas del mismo

Clase	Método	Escenario	Valores de Entrada	Resultado
Village	moveRight	setupScenary1	<i>ninguno</i>	se modifica el atributo del objeto PokemonTrainer que posee la clase, sumando movement a a su atributo en X

Objetivo de la Prueba: Verificar el correcto funcionamiento del sistema de movimiento en dirección hacia la izquierda para el personaje modificando de manera correcta las coordenadas del mismo

Clase	Método	Escenario	Valores de Entrada	Resultado
Village	moveLeft	setupScenary1	<i>ninguno</i>	se modifica el atributo del objeto PokemonTrainer que posee la clase, restando movement a a su atributo en X

Objetivo de la Prueba: Verificar que el jugador no pueda atravesar diferentes objetos que hay por el mapa, por la misma razón no podrá salirse del mismo por zonas no permitidas

previamente,esto se consigue creando objetos los cuales ocupan un espacio en el espacio no permitiendo el paso

Clase	Método	Escenario	Valores de Entrada	Resultado
Game Object	isNear	setupScenary2	x = 50 y = 50	true El jugador no podrá continuar moviéndose en la dirección hacia la derecha ya que está “colisionando” con un objeto el cual no puede atravesar