

1.1 Alcance del videojuego

"Average Day at Nueva" es un videojuego tipo Roguelike desarrollado en Java, ambientado en un día normal para el argentino promedio. El jugador se enfrenta a diversos desafíos y enemigos mientras intenta llegar a la facultad. Será disponible para PC, está dirigido a un público mayor de 13 años debido a la presencia de violencia, alcohol, tabaco y referencias socio-políticas.

1.2 Definiciones, acrónimos y abreviaturas

- **Spawn:** Zona donde se generan enemigos.
- **Tile:** Casilla que almacena información como datos o imágenes.
- **Mov Speed:** Velocidad de movimiento del personaje.
- **NPC:** Personaje no jugable (aliado o enemigo).
- **Display:** Interfaz gráfica mostrada en pantalla.
- **Melee:** Ataque cuerpo a cuerpo.

1.3 Referencias

- Vampire Survivors.
- Juegos Roguelike varios.
- Videos de YouTube y repositorios de juegos 2D.

1.4 Visión general

Este documento contiene la descripción general del videojuego, los requerimientos funcionales y no funcionales, así como los de jugabilidad, anexos y materiales de apoyo al desarrollo.

Diagrama de clases

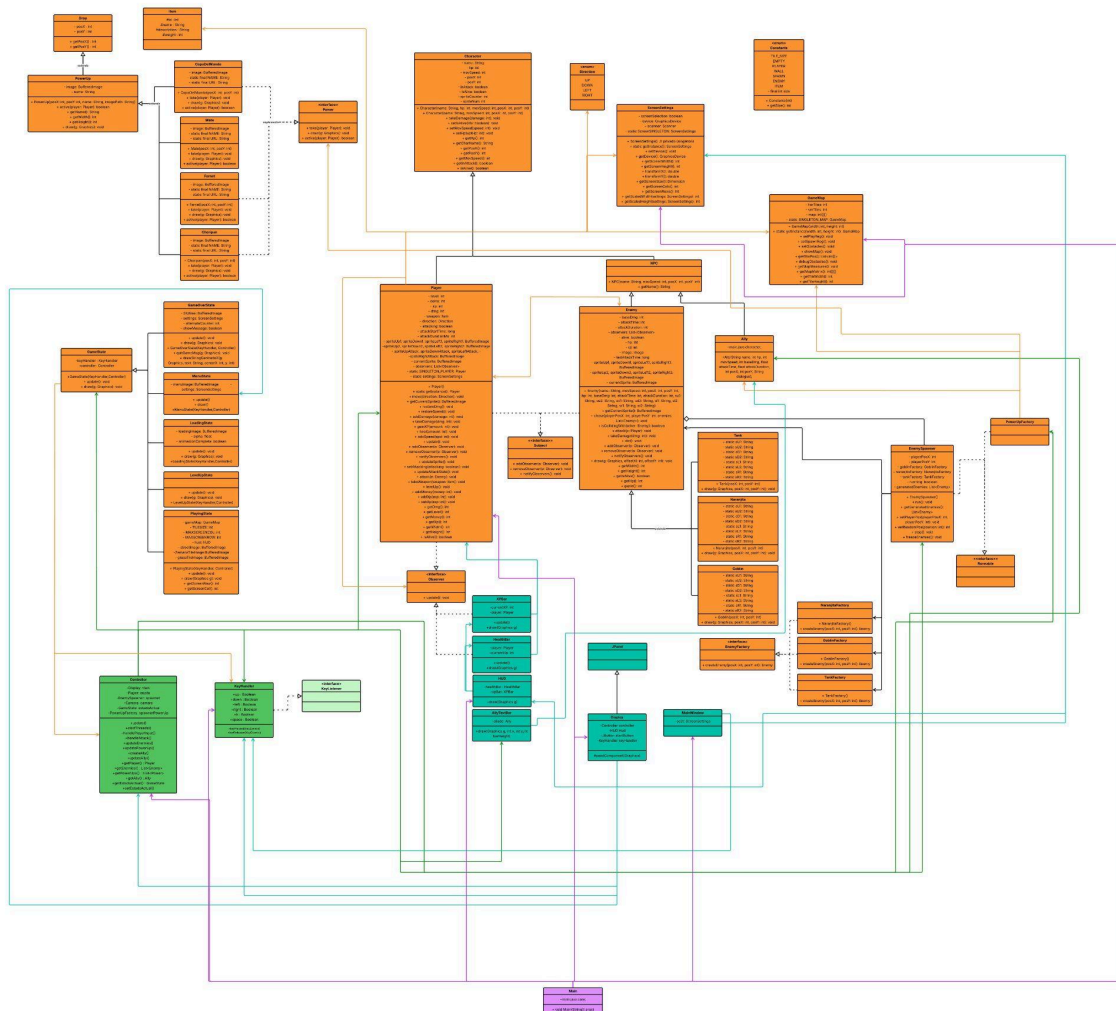


diagrama de clases

```
graph LR
    Actor[Actor  
Player]
    subgraph Videojuego
        I1(Iniciar partida)
        I2(Mover personaje)
        I3(Atacar)
        I4(Interacción con aliados)
        I5(Recolectar XP)
        I6(Subir de nivel)
        I7(Finalizar partida)
    end
    Actor --- I1
    Actor --- I2
    Actor --- I3
    Actor --- I4
    Actor --- I5
    Actor --- I6
    Actor --- I7
```

[illegible]

diagrama de secuencias

2. Descripción General del Videojuego

2.1 Perspectiva del producto

El videojuego está basado en la franquicia Vampire Survivors. El jugador debe sobrevivir a oleadas de enemigos en un entorno ambientado en Argentina. El personaje, el cual será controlado por el usuario, aparece en una calle remota de Nueva Córdoba, a partir de ese momento podrá desplazarse por un mapa finito que tendrá obstáculos que bloquean su camino. Una cámara centrada en el personaje irá siguiéndolo mientras se desplaza en cualquier dirección con tal de recoger diferentes drops e interactuar con los NPCs aliados que vayan apareciendo durante la partida.

A medida que pasa el tiempo se irán invocando diferentes tipos de enemigos con armas, estadísticas y habilidades únicas; los enemigos aparecerán por oleadas y cada oleada se volverá más poderosa. Por eso nuestro personaje deberá adquirir nuevas armas y subir de nivel.

Por último, con el dinero recolectado tras cada partida, el jugador podrá adquirir mejoras en la tienda disponible desde el menú principal, las cuales facilitarán el progreso en futuras sesiones de juego.

El progreso del jugador —incluyendo la puntuación récord y el dinero acumulado— se almacenará en un archivo de texto (.txt). Si bien este archivo podría ser modificado manualmente, se contempla la posibilidad de implementar un algoritmo de encriptación con el fin de proteger la integridad de los datos almacenados.

2.2 Funciones del producto

- Movimiento libre del personaje en un mapa de dimensiones fijas.
- Oleadas de enemigos.
- Recolección de ítems, armas, dinero y experiencia.
- Progresión en la dificultad del entorno.

2.3 Características de los jugadores

- Edad: 13+
- Perfil: Jugadores casuales o con poca experiencia.
- Accesibilidad: Diseñado para jugarse con una sola mano (teclas ASDW).

2.4 Restricciones

- Limitado a plataformas de escritorio (PC) con Windows.
- Requerimientos de Hardware mínimos: 4 GB RAM / Intel Pentium G3250 / 500 MB ROM.
- Uso exclusivo de Java y Java Swing.
- Contenido restringido por calificaciones de edad.

2.5 Suposiciones y dependencias

- El juego será desarrollado completamente en Java.
 - Se utilizará Java Swing para la interfaz.
 - Los assets serán propios o de uso libre.
-

3. Requerimientos Funcionales

- El sistema debe permitir guardar el dinero ganado en partidas.
 - Controlar el movimiento del jugador mediante teclas.
 - Detectar colisiones entre personajes y objetos.
 - Implementar un sistema de puntuación con la puntuación actual de cada partida y la puntuación récord.
 - Diseñar un menú principal con opciones.
 - Gestionar niveles y progreso del jugador.
-

4. Requerimientos No Funcionales

- Rendimiento mínimo de 60 FPS estables.
 - Tiempo de carga mínimo (<10 seg) tanto al inicio como al cargar una partida.
 - Código modular para permitir la extensión futura (armas, enemigos, mapas).
 - Interfaz intuitiva, con curva de aprendizaje menor a 5 minutos. Que contenga iconos y colores llamativos para que el usuario comprenda y se oriente en el software.
 - Capacidad de manejar al menos 500 enemigos simultáneos sin caídas críticas de rendimiento. (Mantener los FPS por encima de 50)
 - Escalabilidad con capacidad de implementación de nuevas actualizaciones que agreguen armas, características, personajes, enemigos, etc.
-

5. Requerimientos de Jugabilidad (Gameplay)

- Dificultad progresiva (enemigos ganan daño y vida).
- Sistema de recompensas por progreso.
- Reglas claras de victoria y derrota.
- IA básica para enemigos y NPCs con distintos comportamientos.

6. Épicas e historias de usuario

- Épica: Mapa.

<input type="checkbox"/> Backlog	(4 of 32 work items visible)	0	0	0	Create sprint
SCRUM-23	Como un jugador, quiero que la cámara del personaje se mueva junto con el personaje	MAPA	TO DO	-	
SCRUM-32	Como jugador, quiero que el mapa tenga obstáculos para limitar el movimiento.	MAPA	TO DO	-	
SCRUM-33	Como jugador, quiero ver efectos visuales que me indiquen zonas peligrosas	MAPA	TO DO	-	
SCRUM-54	Como jugador, quiero que tenga limites mi mapa y no pueda cruzarlos	MAPA	TO DO	-	
+ Create					

- Épica: Desarrollo de los NPC

<input type="checkbox"/> Backlog	(6 of 32 work items visible)	0	0	0	Create sprint
SCRUM-29	Como jugador, quiero que los enemigos se muevan siempre en dirección al personaje principal.	DESARROLLO DE LOS PE...	TO DO	-	
SCRUM-30	Como jugador, quiero que los enemigos se generen aleatoriamente por fuera del mapa	DESARROLLO DE LOS PE...	TO DO	-	
SCRUM-31	Como jugador, quiero que todos los enemigos inflingan daño por contacto al personaje principal.	DESARROLLO DE LOS PE...	TO DO	-	
SCRUM-35	Como jugador, quiero un tipo de enemigo que ataque a distancia, que infliga daño en área	DESARROLLO DE LOS PE...	TO DO	-	
SCRUM-38	Como jugador, quiero un tipo de enemigo que ataque a melee	DESARROLLO DE LOS PE...	TO DO	-	
SCRUM-62	Como jugador, quiero que los NPC's tengan diálogos para sentirme más inmerso en la trama	DESARROLLO DE LOS PE...	TO DO	-	

- Épica: Armas del jugador

<input type="checkbox"/> Backlog	(6 of 32 work items visible)	0	0	0	Create sprint
SCRUM-45	Como jugador, quiero poder elegir una nueva arma al subir de nivel para aumentar mi poder.	ARMAS DEL JUGADOR	TO DO	-	
SCRUM-50	Como jugador, quiero un arma que lanza una daga al enemigo más cercano	ARMAS DEL JUGADOR	TO DO	-	
SCRUM-51	Como jugador, quiero un arma que cae del cielo, provocando daño en área	ARMAS DEL JUGADOR	TO DO	-	
SCRUM-52	Como jugador, quiero un arma que apunte a un enemigo aleatorio y lo congele, inmovilizándolo ...	ARMAS DEL JUGADOR	TO DO	-	
SCRUM-53	Como jugador, quiero un arma de tipo orbital, que gire alrededor del personaje e infliga daño de ...	ARMAS DEL JUGADOR	TO DO	-	
SCRUM-59	Como jugador, quiero tener un limite de armas	ARMAS DEL JUGADOR	TO DO	-	

- Épica: Desarrollo de la jugabilidad del personaje

<input type="checkbox"/> Backlog	(3 of 33 work items visible)	0	0	0	Create sprint
SCRUM-39	Como jugador, quiero mover a mi personaje con el teclado para explorar el mapa.	DESARROLLO DE LA JU...	TO DO	-	
SCRUM-55	Como jugador, quiero que mi pj ataque a melee en la direccion que observo	DESARROLLO DE LA JU...	TO DO	-	
SCRUM-63	Como jugador, quiero que mi pj tenga armas a distancia que ataquen de forma aleatoria	DESARROLLO DE LA JU...	TO DO	-	
+ Create					

- Épica: Sistema de economía y experiencia

☐ Backlog (5 of 33 work items visible)

000Create sprint

SCRUM-40	Como jugador, quiero que el jugador gane XP para subir de nivel	SISTEMA DE ECONOMÍ...	TO DO	-	
SCRUM-48	Como jugador, quiero recoger monedas para gastar en mejoras fuera de la partida	SISTEMA DE ECONOMÍ...	TO DO	-	
SCRUM-56	Como jugador, quiero ganar dinero para comprar buff e items	SISTEMA DE ECONOMÍ...	TO DO	-	
SCRUM-57	Como jugador, quiero un sistema de dinero que se mantenga en el tiempo	SISTEMA DE ECONOMÍ...	TO DO	-	
SCRUM-58	Como jugador, quiero que enemigos o la inflacion me quite dinero	SISTEMA DE ECONOMÍ...	TO DO	-	

- Épica: Items y PU's

☐ Backlog (3 of 33 work items visible)

000Create sprint

SCRUM-46	Como jugador, quiero recoger cofres que me otorguen mejoras aleatorias.	ITEMS Y PU'S	TO DO	-	
SCRUM-47	Como jugador, quiero recoger comida para recuperar salud	ITEMS Y PU'S	TO DO	-	
SCRUM-61	Como jugador, quiero que lo items se clasifiquen por tipos, nivel y calidad	ITEMS Y PU'S	TO DO	-	

+ Create

- Épica: Interfaz de usuario

☐ Backlog (4 of 33 work items visible)

000Create sprint

SCRUM-34	Como jugador, quiero ver nivel y armas en uso	INTERFAZ DE USUARIO	TO DO	-	
SCRUM-36	Como jugador, quiero una pantalla de pausa para revisar mis estadísticas.	INTERFAZ DE USUARIO	TO DO	-	
SCRUM-37	Como jugador, quiero una pantalla de game over para ver mi puntuación	INTERFAZ DE USUARIO	TO DO	-	
SCRUM-44	Como jugador, quiero ver la barra de vida de mi pj para saber cuanto daño puedo recibir.	INTERFAZ DE USUARIO	TO DO	-	

+ Create