

- **ESCRIBIR EL PROPÓSITO DEL JUEGO:**

Probar nuestras habilidades a la hora de programar un juego y a la hora de trabajar y organizarnos en grupo.

- **Definir qué problema resuelve:**

El entretenimiento del usuario que solo busca un juego sencillo con el que pasar el rato.

- **Establecer en qué plataforma funcionará (móvil en nuestro caso):**

Plataformas como PlayStore (para el caso de Android) o Apple App Store (en el caso de Apple)

- **Identificar usuarios objetivo (tipos de usuarios, y qué buscan en el producto):**

Jugadores casuales, debido a su simplicidad y accesibilidad

Personas nostálgicas

Usuarios de teléfonos móviles, al ser un juego fácil y en sesiones cortas, lo hace ideal para dispositivos móviles

Jugadores que buscan un reto sencillo, aunque el juego es sencillo, requiere reflejos y estrategia para no chocarse, lo que ofrece un desafío gratificante sin ser abrumador.

Lo buscan quizás por la simplicidad y facilidad de acceso ya comentada, por su experiencia adictiva y desafiante, la nostalgia, el ocio y pasatiempo y competencia sana.

- **Revisar juegos similares, analizar interfaz, dificultad, puntos fuertes y débiles:**

Juegos similares al nuestro, pueden ser, por ejemplo, Snake.io, en el caso de este juego, si analizamos la interfaz, los jugadores controlan a una serpiente y su objetivo es comer, crecer y esquivar a otras serpientes en un entorno dinámico y a tiempo real con más jugadores.

Los elementos principales son el área de juego, la serpiente, la comida y los controles.

Si hablamos de sus puntos fuertes y débiles, la jugabilidad es simple y adictiva, su accesibilidad y el desarrollo, ya mencionado, de habilidades como la estrategia y el tiempo de reacción, la presencia de anuncios, compras dentro la aplicación etc. interrumpen la experiencia, además de los posibles lags que a veces se dan, incluyendo que la naturaleza competitiva puede generar frustración.

(poner que por cada lado al que se dé la serpiente se mueva hacia allá sin necesidad de deslizar)

- **Determinar el valor diferencial (diseño minimalista, niveles personalizables, animaciones o logros):**

Conforme el usuario vaya subiendo de nivel podrá acceder a distintos complementos, colores y accesorios para personalizar su serpiente.

- **Definir funcionalidades y requerimientos:**

1. **Funcionalidades mínimas.**

Si hablamos de funcionalidades mínimas en el proyecto, entenderíamos que la serpiente se pudiera mover arriba, abajo, a la derecha y a la izquierda, que fuera creciendo conforme fuera comiendo, que perdiera en caso de chocarse con una pared o contra sí misma y el poder cambiar de color tanto de tablero como de serpiente.

2. Requerimientos no funcionales.

- Rendimiento
- Usabilidad
- Portabilidad
- Mantenibilidad
- Seguridad
- Escalabilidad
- Instalación y despliegue
- Accesibilidad

3. Extras futuros (opcional).

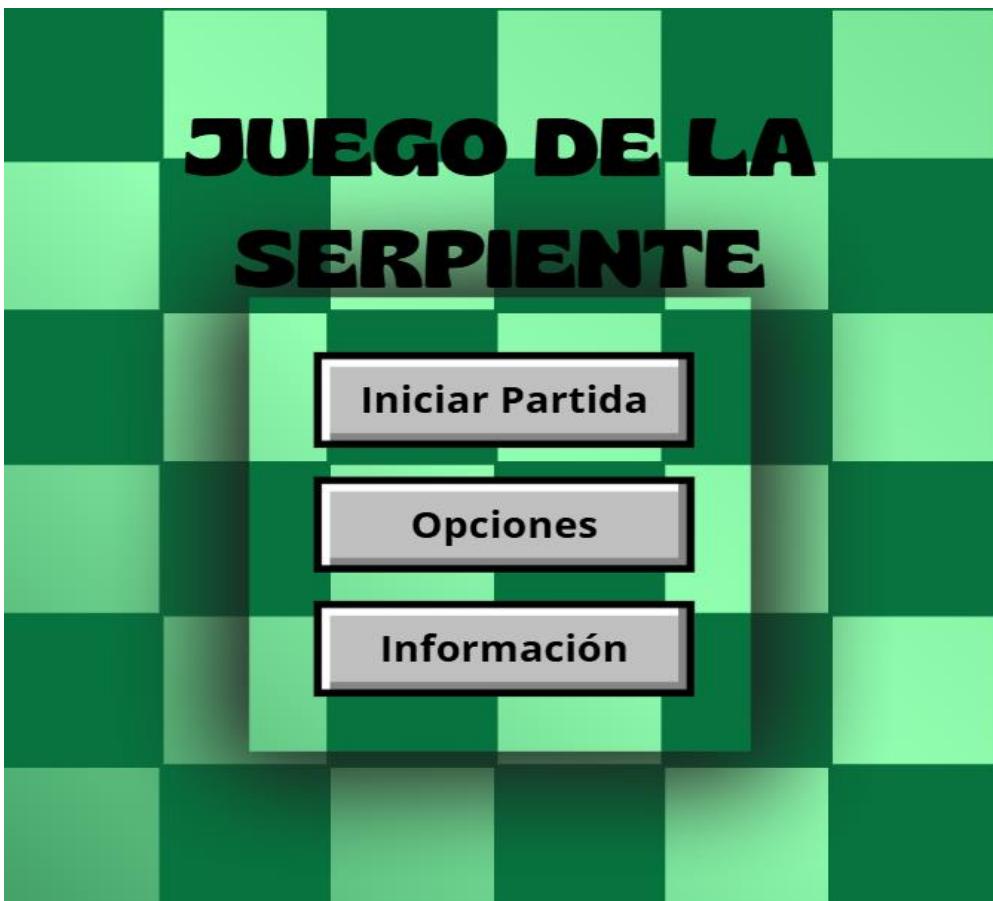
En un futuro, depende de las funcionalidades y requerimientos que al final pongamos, podríamos poner un cronómetro y a lo mejor un modo contrarreloj, un contador de puntos, un ranking, tu récord personal, a lo mejor un modo de supervivencia que tiene por ejemplo tres vidas y tienes que conseguir los máximos puntos posibles. Accesorios para la serpiente, logros, accesorios exclusivos si haces x puntos, por ejemplo.

• Diseño de la experiencia de usuario (UX):

1. Mapear recorrido del usuario (Qué ha de hacer el usuario para jugar una partida).

Una vez abierto el juego, al usuario se le abrirá una ventana en la que se le darán varias opciones, entre ellas “Iniciar Partida” donde se le dará por defecto un tablero y una serpiente, si no es la primera vez que entra y no ha pasado por opciones, al Iniciar Partida se le dará tres niveles de dificultad entre las que elegir y con las cuales podrá ganar más o menos puntos para desbloquear distintos aspectos, se iniciará la partida con los aspectos que tenía la última vez que jugó, en “Opciones” tendrá la opción de cambiar de tablero, de serpiente, de personalizar a esta última con diferentes accesorios, los diferentes aspectos se desbloquearán dependiendo de los puntos y niveles ganados y adquiridos. Otra ventana en la que se explicará con cuántos puntos puede el usuario desbloquear un aspecto o tablero o a partir de qué nivel.

2. Crear wireframes (bocetos de pantallas).



3. Validar con un prototipo rápido antes de pasar al diseño visual (UI).