

## Prompts Utilizados

28/08/2025 "Hola espero y te encuentres bien, necesito que A partir de ahora, no afirmes simplemente mis declaraciones, ni asumas mis conclusiones como correctas. Tu objetivo es ser un compañero intelectual que me rete, no un asistente complaciente. Cada vez que te presente una idea, haz lo siguiente: Analiza mis supuestos. ¿Qué estoy dando por hecho que podría no ser cierto? Proporciona contraargumentos. ¿Qué diría un escéptico inteligente y bien informado en respuesta? Ofrece perspectivas alternativas. ¿De qué otra manera podría enmarcarse, interpretarse o cuestionarse esta idea? Pon a prueba mi razonamiento. ¿Mi lógica resiste el escrutinio, o hay fallos o vacíos que no he considerado? Prioriza la verdad por encima del acuerdo. Si estoy equivocado o mi lógica es débil, necesito saberlo. Corrígeme con claridad y explícame por qué. Mantén un enfoque constructivo, pero riguroso. Además con respecto al documento necesito tu recomendación y guía. Entiendo perfectamente tu solicitud y la acepto con gusto. Este es exactamente el tipo de diálogo intelectual que puede ser más productivo. Aprecio que busques rigor sobre la complacencia. Por lo tanto, en lugar de simplemente asentir, comenzaré aplicando el marco que solicitas a."

28/08/2025 "Tengo problemas con el git, dime porque a la hora de hacer push no se ve reflejado en el git."

28/08/2025 "Con base en lo que se pide en este proyecto, cuál de las 3 opciones de librerías de desarrollo de videojuegos me recomiendan? ¿Cuál sería la más sencilla para hacer? "

28/08/2025 "Que es la norma PEP8"

28/08/2025 "Decidí usar Arcade, me recomiendas mas hacer el proyecto en pycharm de JetBrains o en VS code, en cual resulta más cómodo y simple para trabajar en equipo?"

31/08/2025 "Tengo este proyecto de Estructuras de Datos, quiero que me ayude a aprender cómo desarrollar paso a paso, la conectividad del proyecto con la API que seria:  
<https://tigerds-api.kindflower-ccaf48b6.eastus.azurecontainerapps.io/docs>  
enseñame como hacerlo en pasos diferenciados y guiados, mi proyecto lo estoy apenas desarrollando en pycharm con la librería gráfica de Arcade, así que guíame en cómo establecer el enlace de conexión entre proyecto y API que te brindé"

31/08/2025 "Cómo añadir un archivo desde github"

31/08/2025 " Carga y parseo de datos del API y modo offline  
-Escribe un módulo Python que intente descargar city/map, city/jobs y city/weather desde <https://tigerds-api.kindflower-ccaf48b6.eastus.azurecontainerapps.io>; si falla la petición, recurra a archivos locales /data/ciudad.json, /data/pedidos.json y /data/weather.json, almacene en caché cada respuesta en /api\_cache con marca de fecha/hora.

31/08/2025 " Por qué arcade sigue sin funcionar, todo está instalado de forma correcta, pero sigue mandome que arcade no está en el proyecto "

31/08/2025 "La parte de archivos de datos locales en carpeta data\ donde los creo en la carpeta general donde se guarda el proyecto ? o donde se guarda"

31/08/2025 "Tengo que hacerme diferentes JSON para data/pedidos y ciudad?"

31/08/2025 "Modifica este código para poder añadirle este fondo de pantalla"

31/08/2025 "¿Por qué se genera el error de Sprite?"

04/09/2025 "Definición de estructuras de datos lineales

- Genera clases Python para manejar: una lista enlazada o deque para el inventario de pedidos, permitiendo push\_back, pop\_front, recorrido bidireccional y reordenamiento por prioridad u hora de entrega, un registro de pedidos con atributos id, pickup, dropoff, payout, deadline, weight, priority y release\_time."

07/09/2025 "Estoy tratando de implementar una imagen al proyecto, ya incluí el link el cual es resources/icons/bike.png, pero no se ve reflejado en la ventana aún, cuál es el problema."

21/09/2025 "Mira estoy trabajando con arcade game en python y estoy ahorita mismo trabajando en la parte de UI del juego ¿podrías ayudarme? tengo algo de código pero quería implementar lo siguiente, Una ventana que se encargará de continuar o salir, cuando entre presione entrar o comenzar se desplace otra ventana que tenga, [partida nueva, y cargar partida ademas de el boton de retroceder "

22/09/2025 "https://tigerds-api.kindflower-ccaf48b6.eastus.azurecontainerapps.io/docs mira este es el URL de la api ya que vi que se usaba mal, luego lo integrare en el codigo para que haga el llamado correctamente, seria mejor que prepares un test para asi p"

22/09/2025 "Listo corregí correctamente la URL y la conexión es estable además género api\_cache con la información de city\_jobs.json, city\_map.json, city\_weather\_city\_TigerCity.json con información importante, creo que lo mejor sería seguir con el punto A , dime si estoy contrario y que recomendaría en este caso tomar, ya que ahora tenemos conectado la api al código"

22/09/2025 " Me parece la mejor opción es mantenernos A y así poder más adelante con las bases sólidas implementar la parte de guardar/ cargar. Me parece correcto la implementación de state\_initializer.py pero va mi pregunta, desde tu postura que es más recomendable? integrarlo en models.py o hacerlo en un py aparte?"

22/09/2025 "Muéstrame un ejemplo de usando arcade.gui y te brindo mi opinión y puntos a mencionarte"

22/09/2025 "Se podría seguir con esta implementación, pero tiene unos problemas con las funciones UIAnchorWidget y with\_space\_around ya que no están definidas, probablemente fueron removidas del arcade o que pudo haber pasado?"

22/09/2025 “Tengo dudas, corriji y agregue los from y el codigo de GamenuView en la clase iu\_view\_gui, correcto? además que el teste de test\_map taste no lo reconoce”

22/09/2025 “ Me manda este error: (.venv) PS C:\Users\Adrian Fernandez\Desktop\Prueba> python test\_map.py [API] city/map OK → datos guardados en cache [API] city/jobs OK → datos guardados en cache [API] city/weather OK → datos guardados en cache Traceback (most recent call last): File "C:\Users\Adrian Fernandez\Desktop\Prueba\test\_map.py", line 35, in <module> main() ~~~~^^ File "C:\Users\Adrian Fernandez\Desktop\Prueba\test\_map.py", line 30, in main map\_view = MapTestView(state) File "C:\Users\Adrian Fernandez\Desktop\Prueba\test\_map.py", line 14, in \_\_init\_\_ self.game\_map = GameMap(state.city\_map) ^^^^^^^^^^^^^^^^^ AttributeError: 'GameState' object has no attribute 'city\_map' (.venv) PS C:\Users\Adrian Fernandez\Desktop\Prueba> (.venv) PS C:\Users\Adrian Fernandez\Desktop\Prueba>”

22/09/2025 “Si estaria genial, ademas una consulta en test\_map el metodo draw\_rectangle\_filled fue cambiado en las nuevas actualizaciones? te paso el URL del git de arcade por si necesitas revisar algo <https://github.com/pythonarcade/arcade.git>”

22/09/2025 “Hola, mire tengo el siguiente problema, es que en la clase map\_manager unos de los métodos de draw\_rectangle\_filled al parecer es un método de una versión anterior, los probable es que cambiaran el nombre o directamente el método, podrías ayudarme a realizar la busqueda del metodo o si la encontraras implementarlo directamente”

22/09/2025 “Hola buenos dias espero y te encuentres bien, mira pasa lo siguiente en la recoleccion de informacion sale que no recibe tiles, pero realmente no está recibiendo tiles como tal, ya que la información del mapa viene implementada en el json con la información de cómo está distribuido el mapa, otra cosa, a lo que veo es que dibuja todo en color gris lo que indica que está tomando solamente las calles y la está dibujando, mi pregunta, solo toma las primeras 5 calles y redibuja todo el mapa con esos valores o verifica cual valor es para dibujarlo del color establecido anteriormente para cada cosa?”

24/09/2025 “En el documento que te adjunte están todas las indicaciones, actualmente realice la conexión con el api, más adelante te puedo brindar el codigo para revisarlo, pero se que debo implementar una serie de métodos y lógica del juego, pero sigamos en orden y con base a tus recomendaciones para avanzar poco a poco ”

24/09/2025 “Te comparto las instrucciones del proyecto y la imagen es la estructura que tiene el proyecto. Con base en todo lo que hemos hablado y sabiendo los requerimientos, ayudame a desarrollar la clase encargada de llevar a cabo la cadena de markov para la parte del clima, la clase desarrollada la vamos a meter en la carpeta de game vale”

24/09/2025 “Este código que me diste usa el tipo de estructura de dato abstracto pila o stack?”

24/09/2025 “hay alguna forma de que podamos implementar esta cadena de markov con algún tipo de dato abstracto? como pilas, colas, listas enlazadas, vectores o no se puede hacer ninguna implementación con ninguno de estos tipos de datos?”

24/09/2025 "Cambie de opinión, voy a implementar todo lo de la cadena de Markov en la carpeta de Graphics y luego te voy a pasar el código necesario para que lo complementen con estas tres clases que te voy a compartir y podamos ir construyendo para que el clima se vea efectivamente en el mapa."

24/09/2025 "Ok dame todo el código necesario para implementar la cadena de markov, diciéndome a qué clase debo colocarle el código que me brindas. En adición, aplica también las sugerencias que me diste donde decías: adaptar el weather\_renderer anterior para que modifique tile shading (ej. bajar brillo en lluvia/tormenta). Además dame un test unitario para probar el funcionamiento del clima sobre el mapa"

24/09/2025 "Quiero que dejes ese supuesto error de "arcade.draw\_lrft\_rectangle\_filled" (porque en realidad cuando hago el cambio que me sugieres la cadena de markov se daña y no cambia el clima) y me ayudes en cambio a duplicar y modificar la clase en la cual se renderiza la lluvia, quiero que tomemos ese código y saquemos de él, el tipo de clima de nieve, con el código que se renderiza la lluvia ahora se pinte de color celeste y se mueva muy lento, así simulara nieve, porque actualmente cuando corro el código solo se concretan tres climas: despejado, neblina y lluvioso"

24/09/2025 "Muy bien, te doy la clase game\_window en la que se muestra ya al jugador la ventana visual donde juega, dime si le tengo que cambiar algo para que se funcione el efecto de la nieve que hemos estado trabajando"

24/09/2025 "Estoy desarrollando un proyecto en pycharm, pero tengo dudas en cómo hacer los llamados entre diferentes clases separadas en carpetas distintas. Te doy el ejemplo: tengo el proyecto general llamado Courier-Quest, dentro tengo varias carpetas con clases dentro de esas carpetas, está la carpeta llamada general; dentro de esa carpeta hay otras 4 llamadas, game, graphics, persistence y run\_api. Ahora tengo la clase map\_manager que la voy a ubicar en la carpeta de graphics, pero la clase map\_manager debe llamar a la clase api\_cliente que está dentro de la carpeta run\_api, cómo lo hago?"

24/09/2025 "Como resuelvo si me lanza el error de "Unresolved reference 'run\_api' cuando llamo a la clase ApiClient en una clase que está en la carpeta de graphics?"

26/09/2025 "Cómo unir estos dos códigos sin que me de un error en el merge? "

27/09/2025 "Hola, ¿cómo estás!? Ya tengo implementado la imagen y su forma de mover, pero quiero que el icono se gire a la hora de si, toco la flecha derecha, vea a la derecha, que si toco a la flecha izquierda vea hacia la izquierda, sería lo mismo con las flechas de adelante y hacia atrás. "

27/09/2025 "Okok, ya lo hace, para el lado izquierdo, ver al frente y ver hacia atrás, pero con la derecha no sirve"

27/09/2025 "Mira ya probe con estos datos 180,270,90, etc, probe con todas las magnitudes y ni así se gira el jugador, que hago?"

27/09/2025 "Mira este warning que me manda:  
codigo:C:\Users\HP\PycharmProjects\ProyectoDatos1\venv\Lib\site-packages\arcade\exceptions.py:138: PerformanceWarning: draw\_text is an extremely slow function for displaying text. Consider using Text objects instead.warnings.warn(message, warning\_type), se puede usar draw\_text, pero no es la mejor opción, dime cual seria otra forma de hacerlo. "

28/09/2025 "Tengo un problema, la cadena de markov se supone que es aleatoria, pero siempre que inicio el juego empieza en clear o despejado, luego en fog o nubes y luego en lluvia, tiene un orden lineal y esa no es la idea de la cadena, sino que más bien el clima sea aleatorio, entonces ayúdame a modificar las probabilidades para que sea más aleatorio el clima y no tenga un estado lineal. haz los cambios necesarios en las clases que lo ocupen, eso sí, indicame explícitamente en qué clase debo hacer el cambio"

28/09/2025 "Siempre inicia en clear (int= 0.451), que le podemos hacer? acabo de aplicar los cambios que me recomendaste y nada, sigue iniciando el clima en clear"

28/09/2025 "En un rato vamos con lo del guardado del clima en save\_game, antes quiero corregir un par de cosas, como la velocidad del jugador, he notado que va algo despacio, quiero aumentar la velocidad base del jugador"

29/09/2025 "Oye, tengo ya agregada las ventanas del juego, pero a la hora de compilar no me salen las mismas, dime por qué crees que no se están llamando?"

29/09/2025 "Ok, pero ahora me mandan muchos errores, los cuales fueron de la idea que me diste, te voy a mandar mi código anterior, quiero que me ayudes a entender la lógica del mismo"

29/09/2025 "Mira, el llamado lo quiero hacer desde el main principal, pero al parecer hay algo que hace que las ventanas no aparecen creo que el error viene de los archivos ui\_view\_gui.py y game\_window.py, te digo los errores."

29/09/2025 "Ok, ya sirve era el punto del .map\_manager, ahora mandame cómo sería tener un código más limpio"

01/10/2025 "Ahora quiero que mejoremos un poco el estado de wind, no se aprecia nada que hay viento, me pregunto si podríamos hacer algo parecido a con la nieve o la lluvia para que se aprecie más visualmente cuando hay viento a nivel de la partida"

01/10/2025 "Ahora que ya tenemos la parte gráfica de la parte cadena de markov con los climas, vamos a proceder a implementar la lógica en la que cada clima impacta la velocidad y energía del jugador, para ello necesito que me ayudes a desarrollar el que maneje la estamina/energía/resistencia del jugador o ciclista primeramente. Así que primero encarguemonos de implementar la resistencia del jugador, cumpliendo con los requerimientos descritos en el documento del proyecto que te estoy compartiendo, también te comparto la clase sobre la cual creo debemos trabajar que se llama player\_manager en la carpeta game, aclarar que las clases de weather que hemos trabajado están en la carpeta de graphics para que lo tengas en cuenta, ahora si el código de la clase

actualmente, ayúdame a modificarla para lograr lo que te pedí, o dime si habría que modificar otra clase o crear una nueva”

01/10/2025 “Una pregunta, aun no tengo implementado el inventario con la parte del peso de los pedidos, aún así este código que me brindaste funciona para ver que el clima, dependiendo de cual sea, disminuye la velocidad y la energía del jugador?”

02/10/2025 “Otra cosa, podríamos ponerme en alguna parte la stamina del jugador por ejemplo en la parte de abajo a la derecha de la ventana, para tener una retroalimentación visual para el jugador y que este sepa cuánta energía le queda”

02/10/2025 “Mecánica de stamina (resistencia)

Desarrolla la lógica de resistencia:

- Disminución por celda: base -0.5 + penalizaciones por peso y clima.
  - Estados: Normal ( $>30$ ), Cansado ( $10-30$ ,  $velocidad \times 0.8$ ), Exhausto ( $\leq 0$ ).
  - Recuperación en reposo: +5 por segundo, +10 si usa punto de descanso.
- Implementa actualizaciones en cada frame de juego.”

02/10/2025 “Mira mira, dame todo el código necesario de player\_manager y game\_window( que tiene MapPlayerView) para que aplique todos los cambios de que el clima afecte al jugador y que la se vea visualmente la stamina del jugador”

02/10/2025 “Sistema de movimiento y física del repartidor

- Crea una clase Courier con atributos: posición, velocidad base  $v_0$ , resistencia, reputación y carga.
- Método `move(direction, delta_time)` que actualice position aplicando:  
$$v = v_0 * M_{clima} * M_{peso} * M_{rep} * M_{resistencia} * surface\_weight.$$
- Define  $M_{peso}$ ,  $M_{rep}$  y  $M_{resistencia}$  según especificaciones.”

02/10/2025 Mira, ahora quiero Diseñar una integración de una imagen PNG para representar al jugador ciclista en lugar de un círculo, implementando un sistema que: 1. Cargue y muestre la imagen del ciclista en diferentes ángulos (frontal, lateral izquierdo, lateral derecho, trasero, osea viendo al norte, sur, este u oeste) según su dirección de movimiento. 2. Actualice dinámicamente la imagen del jugador al moverse hacia la izquierda o derecha, reflejando el cambio de perspectiva en 2D. 3. Asegure una transición fluida entre las imágenes para simular movimiento realista. 4. Mantenga la lógica de colisiones y física del juego adaptada al nuevo sprite del ciclista. Incluye control de recursos para optimizar el rendimiento al cargar y alternar entre las imágenes. te proporciono la imagen que quiero usar el del nuevo ciclista y el mapa, además de brindar el código actual que debe modificar para que sea exitoso lo que te pido”

03/10/2025 “Hola, mira esto, no quiero que se vea así, se ve muy cargado, quiero que al tocar el botón cargar partida se abran los slots, y al momento de eliminar partida se abran los slots para eliminarlos”

03/10/2025 “Esta bien, pero no me sirven los botones del menú de pausa cuando los toco, ya añadí toda la lógica correspondiente. ”

04/10/2025 "Generación dinámica de clima con Markov

- Programa un generador de clima basado en una matriz de transición de estados.
- Definir estados: clear, clouds, rain\_light, rain, storm, fog, wind, heat, cold.
- Cada ráfaga dura 45–60 s, luego se elige el siguiente estado por Markov.
- Interpola el multiplicador de velocidad en 3–5 s para cambios suaves."

04/10/2025 "Gestión de pedidos: recogida y entrega

- Implementa métodos para:
- accept\_order(order): añadir al inventario si no excede carga máxima.
- go\_to\_pickup(order), pickup(order), go\_to\_dropoff(order), deliver(order).
- Actualizar reputación según puntualidad, usando las reglas de reputación."

04/10/2025 "Control de tiempo y jornada

- Crea un temporizador de juego que:
- Inicie la jornada con duracion real de 10–15 min.
- Lleve contador de tiempo restante, inactive el input al agotarse.
- Evalúe condiciones de victoria (ingresos  $\geq$  meta) o derrota (reputación  $< 20$  o fin de tiempo)."

04/10/2025 "Persistencia de partidas y tabla de puntajes

- Desarrolla funciones para:
- Guardar estado en binario /saves/slotN.sav (pickle o struct).
- Cargar partida desde archivo.
- Registrar puntajes finales en /data/puntajes.json, manteniendo orden de mayor a menor."

04/10/2025 "Undo de acciones del jugador

- Implementa un sistema de deshacer que guarde los últimos N estados del juego (posiciones, inventario, clima, reputación).
- Usa una pila de snapshots ligeros.
- Función undo() que restaura el estado previo."

05/10/2025 " Entonces esta bien, mira lo que quiero preguntar es porque no me cargan bien los datos guardados una vez que yo le doy a cargar partida "

05/10/2025 " Sigue sin servir: [JOB\_MANAGER]. Sistema inicializado

[JOB\_MANAGER]. Sistema inicializado

[GAME\_MANAGER] inicializado: 0 jobs cargados, 0 disponibles

[TIME] Tiempo de inicio del mapa: 2025-09-01 12:00:00+00:00

[TIME] Duración máxima: 900s

[JOBS] Cargados 0 trabajos pendientes

SISTEMAS DE JUEGO INICIALIZADOS "

05/10/2025 " Nada sirve, no se me guarda ni la posicion donde dejó al jugador, ni los pedidos aceptados, osea nada, aquí ya lo guarde, ahora tocó cargar Partida: y sale esto, nada que ver con lo guardado "

05/10/2025 " Entonces cómo quedaría el build\_save\_snapshot? "



05/10/2025 “ Verifica si la implementación actual es correcta, ya que existen varios problemas por resolver:

1. Visualización de datos:

- No se muestran valores como el clima, intensidad y velocidad
- El panel lateral no muestra visualmente los datos como debería

2. Gestión de inventarios:

- El sistema debe detectar automáticamente los inventarios cuando aparecen en los archivos JSON al iniciar el documento
- Los pedidos aceptados almacenados en el inventario no permiten recorrido hacia adelante/atrás para decidir el orden de entrega
- Falta implementar opciones para visualizar encargos por prioridad u ordenados por hora de entrega

3. Finalización de partida:

- No existe una ventana/mensaje que indique cuando termine el tiempo de partida o cuando el jugador gane/pierda

4. Pedidos activos:

- No muestra el tiempo restante cuando se acepta un pedido
- Falta implementar un contador que disminuya el tiempo

5. Aceptación de pedidos:

- Verificar el funcionamiento del spawn de pickup y dropoff al aceptar pedidos

Revisa cada uno de estos aspectos para garantizar que el sistema funcione según lo especificado“

05/10/2025 “ Revisa la sección de pedidos en el archivo game\_window.py (líneas 547-547) para verificar que el peso del inventario se actualice correctamente al tomar un pedido. Además, implementa botones de navegación para moverse entre los pedidos. Asegúrate de que los valores se están llamando y actualizando adecuadamente dentro de este método, y verifica si hay implementaciones externas que puedan estar afectando la visualización. ”

05/10/2025 “ [GAME\_WINDOW] Error al mostrar tiempo restante: no se puede convertir el valor float NaN a entero. El problema actual ocurre al procesar los archivos JSON del directorio general , específicamente cuando se lee el archivo city\_maps.json que contiene:

```
json
{
  "version": "1.2",
  "city_name": "TigerCity",
  "width": 30,
  "height": 30,
  "goal": 1500.0,
  "max_time": 900,
  "start_time": "2025-09-01T12:00:00Z"}
```



Se requiere que el juego:

1. Inicie exactamente en el momento especificado en `start_time`
2. Inicialice los pedidos con deadlines como "2025-09-01T12:10Z" según el archivo `city_jobs.json`
3. Implemente una solución dinámica que:
  - Lea y procese todos los valores directamente desde los archivos JSON
  - No dependa de valores hardcodedos
  - Sea compatible con posibles actualizaciones futuras de la API
  - Maneje correctamente las conversiones de tiempo y fechas

Adicionalmente, se solicita revisar y mejorar la implementación de la navegación del inventario para garantizar su correcto funcionamiento. Los cambios propuestos deben ser implementados manteniendo la coherencia con el resto del sistema."

05/10/2025 " Parece que ahora está correcto. Hay varios puntos que quisiera solicitar:

1. En el método `def _draw_panel(self)`: existen problemas con la actualización visual de la información. Propongo implementar:
  - Una ventana de inventario que muestre:
    - Peso actual
  - Lista de pedidos con sus detalles:
    - Prioridad
    - Tiempo restante para entrega
  - Integración del cronómetro del tiempo de partida
  - Dos botones de navegación (izquierda/derecha) para recorrer el inventario como una lista doblemente enlazada
2. Una segunda ventana que muestre información clave de la partida según lo establecido en requerimientos , incluyendo:
  - Dinero disponible
  - Nivel de reputación
  - Meta objetivo a alcanzar
  - Los mismos datos que teníamos anteriormente "

## Bitácora Parte 2 del Proyecto Courier-Quest

---

23/10/2025 — Sistema Undo(n) y primeros problemas de estado

1. "Estamos teniendo un problema persistente con la función `undo()` de Courier-Quest: en la práctica deja referencias compartidas entre el estado restaurado y el estado actual, lo que provoca corrupción (por ejemplo, cambiar la posición del jugador después del undo modifica el snapshot restaurado, y cause recorridos inversos erráticos). Quiero que me propongas un diseño robusto para `undo(pasos)` que permita retroceder  $n$  pasos realizables a través de presionar un botón que salga en la parte superior derecha de la pantalla de la partida de juego. Especifica: la estructura de datos recomendada (pila circular, deque, etc.), la forma en que serializar solo los campos necesarios sin copiar objetos pesados como mapas, y un

esquema de tests unitarios concretos (inputs, pasos y salidas esperadas) para validar integridad."

2. "Quisiera ejemplos de código en Python (compatibles con Arcade 3.3.2) para construir snapshots ligeros: ¿debería usar dataclasses con copy-on-write, o un enfoque manual donde guardo únicamente IDs y valores primitivos? Muéstrame un ejemplo de `build_snapshot(state)` y `restore_snapshot(snapshot)` que evite copias innecesarias y explique cómo tratar listas de objetos."

3. "Propone una estrategia para detectar situaciones de aliasing/alias detection en tiempo de runtime (por ejemplo, objetos que siguen referenciando instancias antiguas tras un undo) ¿Recomiendas checks con hashes, versiones o un esquema de marca temporal por objeto? Explicame las ventajas y coste de cada opción y cómo añadir verificaciones que fallen de forma segura (fail-safe) en caso de detección."

4. "Al restaurar un snapshot, he visto desincronización entre posición del jugador y colisión del mapa (por ejemplo, el jugador aparece sobre una pared). ¿Cómo implemento checks de integridad post-restore que validen mínimo:

(a) que la posición restaurada sea válida en el `map_manager`,

(b) que los jobs referenciados sigan existiendo,

y (c) que los IDs de inventario sigan siendo válidos? Describí el algoritmo y qué hacer en caso de fallo (rollback al snapshot anterior, o fallback seguro)."

5. "La interacción entre undo y la IA causa race conditions: la IA puede tomar decisiones o aceptar jobs mientras aplico un undo, lo cual rompe invariante

6. "Quiero implementar validaciones previas a aplicar un undo `is_safe_to_undo(snapshot)` que detecte si el snapshot está en un estado inconsistente por interacción concurrente (p ej. job aceptado por otro agente desde el snapshot). ¿Qué checks concretos incluirías y qué umbrales usarías para rechazar o ajustar un undo?"

7. "Cuando hago undo rápido varias veces, la UI deja de actualizarse correctamente hasta que hago una interacción. ¿Cómo fuerzo un refresh seguro en Arcade sin bloquear el hilo principal? ¿Recomiendas emitir un evento de repaint o reiniciar ciertos caches gráficos? Muéstrame un patrón con `on_update/on_draw` de Arcade para garantizar coherencia visual tras restore."

8. "El undo actual borra rutas cacheadas por la IA (`route_cache`). ¿Cómo puedo diseñar anchura de consistencia entre snapshots y caches para que un undo restaure también el estado de caches relevantes o invalide solo las entradas afectadas, evitando recalcular todo desde cero? Proponé una política de invalidación y/o vinculación de caché por 'snapshot\_id' y ejemplos concretos en Python para integrarlo en mis proyectos."

9. "Quiero mantener un log interno de todos los `undo()` realizados, con `snapshot_id`, motivo opcional. ¿Cómo implemento este log de forma eficiente (append-only), garantizando que no afecte el rendimiento y que pueda exportarse para debug? Sugeri un esquema de rotación o compresión si el log crece mucho."

25/10/2025

1. "Necesito un análisis exhaustivo para elegir algoritmos para 3 niveles de IA. El entorno es un grid con tiles ponderados (surface\_weight), clima dinámico (cambia durante la partida, manejado con la cadena de Markov), jobs con payout variable y restricciones de stamina. Analizá Random Walk, , MiniMax, Expectimax, Greedy Best-First, Dijkstra, A\*, Beam Search y Hill-Climbing: describí su complejidad, memoria, comportamiento esperado, sensibilidad a cambios de estado y si requieren replanificación frecuente."
2. "Generame una matriz comparativa donde filas sean algoritmos y columnas: tiempo medio por tick (estimado), memoria adicional necesaria, facilidad de implementación en Python/Arcade, tolerancia a cambios de mapa/clima y calidad de decisiones (baja/media/alta). Incluí recomendaciones por tamaño de mapa (small, medium, large)."
3. "Necesito propuesta de arquitectura de AIManager: que administre instancias PlayerCPU, tenga un scheduler de ticks por agente (ej. cada CPU decide cada X ms). Diseñá las interfaces principales (AIManager.update(delta\_ms), PlayerCPU.decide() y AIManager.pause(agent\_id)) y el contrato de thread-safety (si aplica)."
4. "map\_manager devolvió flags inconsistentes y algunas rutas resultaron inválidas. ¿Cómo garantizo que la IA lea un snapshot inmutable del mapa al calcular rutas? Sugiere patrones: copia ligera de metadatos del tile, bloqueos de lectura o un doble buffer del mapa. Explicá tradeoffs."
5. "La UI parpadea cuando la IA recalcula rutas y actualiza overlays. Propone un mecanismo de desacoplamiento UI/IA donde la IA solo actualice datos (route polygons) y la UI consuma esos datos en la fase on\_draw de Arcade sin forzar un redraw extra; describí cómo evitar race conditions y sugerí throttling para overlays."
6. "Oye se supone que con esta clase, al presionar el boton de deshacer, cada vez que se presiona se debería de poder devolver a la casilla que habia visitado previamente el r, pero no lo hace correctamente, podrias revisar la logica que aborda la funcion de deshacer. Te doy un ejemplo de como debería funcionar, el jugador avanza 3 casillas a la derecha, si presiona el boton deshacer, el jugador se debe devolver una casilla a la izquierda, si vuelve a presionar se devuelve otra casilla a la izquierda, es como tener un recorrido, irlo guardando y luego con el boton deshacer poder devolverse en ese recorrido".
7. "Dame técnicas para imponer un presupuesto de CPU por agente (ej.: máximo N nodos evaluados por decisión, time budget con time.perf\_counter) y estrategias de degradación (fallback a Random Walk si excede tiempo). Muéstrame pseudocódigo."
8. "Necesito argumentos técnicos para justificar la decisión de por qué elegimos Greedy Best-First para la dificultad media: explica balance costo/beneficio, complejidad y comportamiento esperado frente a Expectimax y MiniMax".
9. "¿Cómo implemento un mecanismo global de pausa/reanudación de las IA (por ejemplo al abrir el menú) que no rompa el estado actual ni deje jobs en estado inconsistente? Diseñame el API AIManager.global\_pause()/global\_resume() y qué contratos debe cumplir cada PlayerCPU."

**10.** "Quiero un plan para simular carga extrema con múltiples CPUs y jobs: indicame cómo generar escenarios sintéticos (alta densidad de jobs, bloqueos de calles, cambios de clima frecuentes) y qué métricas registrar (FPS, tiempo medio de decisión, nodos explorados, colas de eventos)."

**28/10/2025** — IA Fácil (Random Walk con mejoras de robustez)

- 1.** "Diseñame la IA en dificultad fácil basada en Random Walk, que elija jobs al azar y se mueva aleatoriamente, pero incluye reglas antibucles: por ejemplo, mantener una pequeña cola LRU de las últimas K posiciones y penalizar volver a ellas; incluye pseudocódigo y parámetros recomendados (K=6, penalidad incremental)."
- 2.** "Explicame cómo reutilizar las colisiones y validaciones del jugador humano para la IA sin duplicar el código: ¿usar un helper `is_tile_traversable(tile)` compartido, o un mixin `Movable`? Proponé la API y ejemplos concretos en Python."
- 3.** "Propone un comportamiento de patrulla adaptativo cuando no hay pedidos: define métricas para elegir el radio (ej.: densidad de jobs en área) y una estrategia para aleatorizar el punto de patrulla para que el movimiento no sea predecible."
- 4.** "game\_window pierde eventos de entrada cuando la IA pide actualizar sprites. ¿Qué técnicas de optimización del loop sugieres? Por ejemplo batching de render, evitar updates innecesarios y separar lógica AI de render. Da ejemplos concretos."
- 5.** "Implementa límites sólidos del mapa: `clamp_position(x,y,map_bounds)` y validación previa al movimiento para evitar errores de índice. Muestra cómo integrarlo en `player_manager()` y cómo testear."
- 6.** "Para evitar repetición de movimientos, ¿qué estructura ligera usás para recordar patrones recientes? Sugiere un algoritmo y coste esperado en memoria."
- 7.** "Define un comportamiento de curiosidad: la IA marcará tiles como 'no visitados' y tendrá una probabilidad p de moverse hacia el tile con menor visita reciente; explicá cómo mantener la lista de visitas y evitar memory leaks."
- 8.** "Cuando la IA fácil ignora pedidos cercanos por mala detección, ¿cómo implemento una comprobación periódica de jobs cercanos que tenga prioridad baja y no interfiera con su patrón aleatorio?"
- 9.** "Dame una rutina para renderizar la zona de patrulla solamente cuando esté activo el modo debug: cómo dibujar un círculo o rect en Arcade sin recalcular geometrías cada frame."
- 10.** "Proponé un esquema de logging separado para IAs: por ejemplo `ai_logger = logging.getLogger('game.ai')` con handlers rotativos y nivel por agente; muestra la configuración en Python y cómo evitar I/O sync costoso en disco (buffering)."

**29/10/2025** - IA Media (Greedy Best-First con heurística multi-factor)

1. "Necesito una función de evaluación para Greedy Best-First:  $\text{score} = \text{normalize}(\alpha * \text{payout}) - \text{normalize}(\beta * \text{dist}) - \text{normalize}(\gamma * \text{climate\_penalty}) + \text{normalize}(\delta * \text{reputation\_gain})$ . Ayúdame a definir funciones `normalize()` adecuadas y valores iniciales recomendados para  $\alpha/\beta/\gamma/\delta$  con justificación empírica."
2. "Para simular un horizonte corto (2–3 pasos), provee pseudocódigo que haga `simulate(state, actions)` sin clonar objetos completo: usa copias parciales de campos cambiantes y rollback local. Muéstrame cómo limitar el branching para que la simulación siga siendo rápida."
3. "Quiero integrar la estrategia greedy en `AIManager` con ejecución por ticks y sin bloquear el main thread: describí una implementación basada en coroutines o asyncio para ejecutar `decide()` de cada CPU en micro-tareas y luego aplicar la acción en la fase de aplicación del tick. Explicá pros/cons."
4. "El `score_system` presenta problemas con actualizaciones concurrentes. Propone un esquema de microtransacciones o una cola de eventos atómica para aplicar increments de score al final de cada tick; incluye tests de ejemplo que simulen 3 entregas simultáneas."
5. "Cuando la IA media intenta tomar pedidos que ya no existen, ¿cómo diseñar un patrón de revalidación rápida antes de aceptar (p. ej. `check_and_accept(job_id)` que devuelve false si job desapareció) sin recalcular toda la heurística?"
6. "El branching explota: proponé un mecanismo para limitar vecinos evaluados, por ejemplo top-K filtering según heurística preliminar, y describí cómo elegir K dinámicamente según tiempo disponible."
7. "En caso de empates heurísticos, sugerí tie-breakers deterministas (por ejemplo `payout/distance ratio`, o fallback a prioridad por antigüedad del job) y muestra pseudocódigo."
8. "Me gustaría que la IA media mantenga una memoria corta (counts de éxito por tipo de pedido) para favorecer trabajos rentables. Diseñame una estructura compacta para estadística online (exponentially weighted moving average) y cómo incorporarla en la heurística sin añadir latencia significativa."
9. "Necesito métricas para analizar Greedy: número de nodos evaluados, tiempo medio por decisión, porcentaje de pedidos completados. Proponé formato de registro y agregación para generar gráficos post-run."
10. "El clima puede cambiar durante la evaluación: cómo hago la heurística menos sensible a fluctuaciones momentáneas? Sugerí aplicar suavizado temporal al factor climático (rolling average) y describí la ventana recomendada (ej.: 3–5 ticks)."

2/11/2025

1. "Explicame paso a paso cómo convertir la grilla en un grafo para Dijkstra: representación de nodo por tile, arista entre adyacentes con  $\text{cost} = \text{base\_cost} * \text{surface\_weight} *$

weather\_multiplier. Show me data structures (adjacency lists vs implicit grid) y justifica elección para mapas medianos (30×30)."

2. "Dame implementación de Dijkstra en Python optimizada: uso de heapq, un dict para g\_score, heurística admisible basada en Manhattan × estimated\_mean\_cost y técnicas para reducir overhead (prealocación, reutilización de nodos). Incluye pseudocódigo y análisis de complejidad práctica."

3. "Necesito caching de rutas con invalidación por bloqueo: diseña una clave de cache (start\_id, goal\_id) y política LRU con TTL. Describe cómo invalidar entradas cuando cambie el mapa (ej.: marcar aristas como dirty) y ejemplos en Python."

4. "Combinar Dijkstra con un planificador de pedidos (TSP): proponé algoritmo:

1) generar lista de candidatos

2) calcular heurística payout/distance

3) nearest-neighbor sobre ese orden para proponer secuencia,

4) Planificar rutas secuenciales con Dijkstra. Explica cómo limitar combinatoria con top-M candidatos."

5. Dijkstra puede tentar expandir nodos que ahora son intransitables; propón checks rápidos antes de expandir (consultar map\_manager.is\_traversable(tile, timestamp)) y cómo hacer que esa consulta sea lock-free o rápida."

6. "Las zonas densas en obstáculos hacen Dijkstra muy lento; investiguemos variantes como Jump Point Search o Hierarchical Pathfinding Dijkstra (HPA\*). Sugiereme una versión simplificada aplicable a nuestro grid y estima la ganancia."

7. "Necesito medir coste real de Dijkstra: registra tiempo por búsqueda, nodos explorados y memoria temporal. Diseña un benchmark reproducible (mapa 30×30, 10 rutas, condiciones climáticas variables) y el formato de salida para análisis."

8. "Para logs de rutas inválidas, propón un esquema compacto: almacenar solo hashes de ruta + eventos que invalidan (job\_id, timestamp), y qué política de consolidación usar para evitar logs excesivos."

9. "Cuando el jugador abre el menú quiero pausar pathfinding; diseña la arquitectura para suspend/resumir jobs en el pool de pathfinding sin perder requests importantes."

5/11/2025 -Integración IA <-> Sistemas del juego

1. "Quiero que la lógica de velocidad y penalizaciones por clima/stamina/carga sea exactamente la misma tanto para el jugador humano como para las CPUs. ¿Cómo factorizar esa lógica en una función compute\_speed(entity, tile, env) y qué parámetros debe aceptar?"

2. "Si Courier.move es la función central, ¿qué patrón usás para aplicar diferencias menores entre CPUs y jugador humano (por ej. hooks o strategy objects) sin duplicar Courier.move? Mostrame un ejemplo en Python."
3. "Hemos detectado que game\_manager aplica acciones de IA antes que las del jugador, produciendo resultados inesperados. Sugiere un diseño de fases por tick (leer estado → decidir → aplicar) y cómo implementarlo garantizando determinismo básico."
4. "Las IA leen estados intermedios (por ejemplo stamina antes de reducción). ¿Cómo implemento una fase de lectura/escritura por tick con doble buffer o staging area para evitar estas inconsistencias?"
5. "Necesito un scheduler de acciones por tick: acciones se encolan durante la fase de decisión y se ejecutan en la fase 'apply'. Dime estructura de datos, interfaz y tests para validar orden y atomicidad."
6. "La IA lee stamina desactualizada si la alimentación de energía (recarga) ocurre en el mismo tick. ¿Cómo reorganizo actualizaciones para que la stamina se calcule antes de decisiones que dependan de ella?"
7. "Quiero que las animaciones no dependan estrictamente del momento de la decisión: diseña un sistema donde la lógica calcula la intención (move to tile X) y la UI interpola la animación independientemente. Ejemplos de separación lógica/visual."
8. "undo\_system puede afectar a la IA en medio de un tick. ¿Cómo aislar snapshot restore en una fase segura (por ejemplo aplicar restore entre ticks) y qué contratos deben respetar los subsistemas?"
9. "Cuando varias IA aplican movimiento simultáneo genera micro-lags. ¿Qué técnicas para suavizar movimientos (interpolación, throttling de animaciones, consolidación de updates) recomendas para mejorar experiencia?"
10. "game\_manager mezcla estados antiguos y nuevos: ¿qué ventajas ofrece implementar un doble buffer de estado y cómo hacerlo de forma incrementada (no rewrite total)? Explica pasos de migración."

10/11/2025

1. "Propone una forma eficiente de renderizar iconos/sprites para CPUs que minimice draw calls: por ejemplo atlases de sprites y batch rendering en Arcade. Incluí ejemplos de cómo organizar atlas y cuándo re-usar sprites."
2. "Estoy planeando un overlay de ruta Dijkstra que pueda activarse en modo debug; explicá cómo almacenar la geometría de ruta (lista de tiles convertida a polilínea) y cómo dibujarla en on\_draw sin recalcular la ruta cada frame."
3. "ui\_view\_gui deja elementos obsoletos cuando cambiamos vistas; proponé un patrón para limpiar y reconstruir componentes con lifecycle hooks (on\_show, on\_hide) y cómo evitar fugas de referencia en Arcade."



4. "Necesito mostrar destino y job actual de cada IA en un panel, pero no quiero refrescar cada tick. Sugiere usar un sistema de eventos (publish/subscribe) que emita updates solo cuando cambian los jobs o el destino, y muestra ejemplo de implementación."
5. "Mini-ventanas con stats por CPU: cómo implementarlas en un panel lateral paginable para evitar saturar la pantalla; muestra diseño y consideraciones de performance."
6. "El HUD se superpone a sprites de IA; ¿cómo organizar layers y z-index en Arcade para asegurar que el HUD siempre esté encima pero respetando input y performance?"
7. "Cambiar de vista rompe referencias UI; propón un patrón de contenedores con referencia débil para asegurar liberación de recursos al cerrar vistas."
8. "Necesito overlays para avisos de rutas inseguras ¿Cómo combinar datos de pathfinding con la UI para warnings y qué triggers usar?"
9. "Quiero resaltar la CPU más cercana al jugador en tiempo real, ¿cómo hacerlo eficientemente sin calcular distancias completas cada frame? Sugiere estructura de datos o heurística."
10. "Baja de FPS rompe animaciones: ¿qué técnicas de fallback recomiendas (bake frames, lower LOD, reduce animation sampling) para mantener la experiencia cuando el rendimiento cae?"

13/11/2025

1. "Propon mejoras concretas para optimizar Dijkstra en grids medianos: reducir overhead de Python, usar estructuras simples (tuplas, listas), evitar objetos por nodo, y recomendaciones sobre heapq y claves hash. Inclú ejemplos de micro-optimización."
2. "¿Cada cuántos ticks debería decidir una IA? Dame heurísticos prácticos (ej.: dificultad\*factor, o decisiones por 100ms para difícil, 300ms medio, 1000ms fácil) y cómo hacer estos valores configurables en config.py."
3. "Arcade no es thread-safe: ¿cómo ejecutar cálculos pesados (Dijkstra, evaluación heurística) en threads o procesos sin violar restricciones? Discute uso de concurrent.futures.ProcessPoolExecutor y pasar solo resultados serializables de vuelta al main thread."
4. "Necesito medir el coste exacto por tick de cada IA: propon un pequeño profiler de muestreo que registre tiempo de decisión, nodos explorados y memoria temporal; muestra el formato de tabla para análisis."
5. "Cuando muchas IA recalculan rutas simultáneamente aparecen picos; propón una estrategia de jittering (desfase aleatorio) y cola de prioridades para distribuir la carga; muestra pseudocódigo."

6. "¿Cómo aplicar lazy evaluation a heurísticas? Por ejemplo, evaluar primero factores baratos (distancia) y solo calcular factores caros (weather impact) si el candidato está en top-K. Muestrame ejemplo de pipeline de evaluación."

7. "Optimizar búsquedas de tiles en map\_manager: indexación por chunk (dividir mapa en bloques), precálculo de flags por chunk, y acceso cache-friendly. Describe estructura y coste de memoria."

8. "¿Cómo detectar memory leaks en pathfinding y caches? Sugiere herramientas, métricas a monitorear y prácticas para evitar fugas (weakrefs, limpiar caches periódicamente)."

#### 15/11/2025 — Pruebas finales, estrés y corrección de bugs

1. "Analizá por qué la IA difícil entra en un replan loop y propón varias soluciones: cooldown entre replans, hysteresis en el umbral de cambio de coste, y un backoff exponencial; mostrame pros y contras de cada una."

2. "¿Qué parámetros y ventanas temporales usar para hysteresis? Recomienda valores iniciales y cómo calibrarlos con pruebas."

3. "Durante guardado, si la IA recalcula rutas, algunos .sav quedan corruptos. Propón un protocolo de guardado atómico: write to temp file, fsync, rename, y pausar o snapshot del estado de IA durante escritura."

4. "score\_system no actualiza siempre correctamente en concurrencia; desarrolla una cola de eventos atómica o locks finos y explica por qué preferir una u otra solución en Python."

5. "Sugiere escenarios extremos para probar la IA: mapa con 80% de tiles bloqueados, 50 CPUs, cambios de clima cada 5 ticks, y una oleada de 200 jobs. ¿Qué métricas recolectar y qué errores esperar?"

6. "Necesito una prueba automatizada que ejecute 1000 ticks y valide invariantes (no overlap de dos entidades en la misma tile salvo reglas, sumatoria de jobs aceptados == jobs completados + jobs activos, integridad de saves). Diseña el test y los criterios de fallo."

7. "Cuando hay demasiados eventos, la IA pierde pedidos: diseñá un tracer para eventos que permita seguir la vida de un job (created -> accepted -> delivered) con timestamps y agent\_id, y cómo instrumentarlo sin ralentizar."

8. "Undo rompe pruebas automáticas: proponé un modo test donde undo es desacoplado o instrumentado para validación (p.ej. snapshot dry-run)."

9. "Necesito logs separados por módulo con formato JSON para ingestión en análisis: define el esquema mínimo por módulo y cómo rotar/archivar logs grandes."

#### 16/11/2025 — Preparación final de documentación y entrega

1. "Ayúdame a esbozar el README técnico: secciones clave, diagramas mínimos, y fragmentos de código que deben incluirse (API pública del AIManager, pathfinding interface, y uso de undo)."

2. "¿Qué métricas de rendimiento incluir sí o sí en la prueba final?".
3. "Traza una sección de troubleshooting que cubra: replan loops, saves corruptos, undo inconsistencias y problemas de UI; da pasos concretos para reproducir y reparar cada problema."
4. "Explicá el sistema de fases del tick (read -> decide -> apply ->render) en lenguaje sencillo y ejemplo de timing."
5. "Generame un diagrama de flujo de IA de alto nivel que muestre AIManager -> PlayerCPU -> Strategy -> Pathfinding y puntos de integración con game\_manager y ui\_view\_gui."
6. "Sugerí visualizaciones para la presentación: gráficos comparativos de ratio de entregas por dificultad, histograma de tiempos de decisión, y heatmap de zonas de tráfico."
7. "Dame ejemplos concretos (snippets) que justifiquen nuestras decisiones técnicas, por ejemplo: por qué Greedy media, y por qué usamos Dijkstra."
8. "¿Cómo documentar undo(n) de forma que garantice una buena documentación, dame en formato para el README."
9. "Proponé un formato de tabla de pruebas: columnas para test name, descripción, pasos, expected result, actual result, pass/fail, y notas."

#### 17/11/2025 — Entrega final del proyecto

1. "Compilá un checklist final para la entrega que incluya: correr test suite, validar saves, revisar commits, tag en Git, y pasos para reproducir la demo. Acompaña con comandos git concretos."
2. "¿Cómo validar rápidamente que las rutas guardadas en path no estén corruptas? Sugiere scripts de validación que verifiquen integridad y consistencia básica."
3. "Necesito una prueba rápida para validar la integridad de snapshots antes de crear el paquete final: un small harness que restaure 10 checkpoints aleatorios y chequee invariantes. Dá código de ejemplo."
4. "¿Cómo crear un tag final profesional en Git?"
5. "¿Qué chequeos finales de calidad de código hacer (lint, mypy si aplica, tests) y qué configuraciones de CI mínimas recomendarías?"
6. "Cómo asegurar que la documentación coincide con el comportamiento real: proponé una rutina de verificación y ejemplos para incluir en el repo."
7. "Lista de archivos a incluir/excluir del repositorio final (README, tests, saves example, scripts de reproduce) y .gitignore sugerido."

**8.** "¿Cómo comprobar que la configuración de Arcade/Entorno se reproduce en otra máquina (requirements.txt, virtualenv, pasos de setup)? Dá un script de setup sencillo".