
This is the **published version** of the bachelor thesis:

Turu Pérez, Damià. *Under My Rules : un videojoc d'estratègia on el jugador programa la seva baralla*. Treball de Final de Grau (Universitat Autònoma de Barcelona), 2026 (Enginyeria Informàtica)

This version is available at <https://ddd.uab.cat/record/334434>

under the terms of the  license.

Under My Rules: un videojoc d'estratègia on el jugador programa la seva baralla

Damià Turu Pérez

28 de juny de 2026

Resum– Aquest treball consisteix en el disseny, desenvolupament i test d'un videojoc del gènere d'estratègia per torns competitiu amb l'objectiu de reforçar l'aprenentatge de conceptes bàsics de programació mitjançant una experiència interactiva. El projecte s'inspira en un joc de taula amb idees similars, però amb unes mecàniques massa complexes per a persones que s'inicien en la programació. El desenvolupament del videojoc es realitza amb el motor Unity i s'organitza seguint la metodologia àgil Kanban per tal de garantir una evolució controlada del producte. El joc planteja partides competitives on dos o més jugadors controlen un cotxe de joguina. L'objectiu principal és situar el cotxe sobre un objecte situat al centre del mapa i escapar. L'element diferencial del projecte rau en la possibilitat de modificar el terreny mitjançant fitxes configurables amb un sistema d'events que permet definir comportaments com canvis del terreny o interaccions amb els jugadors.

Paraules clau– Videojoc, Unity, Kanban, Programació, videojoc d'estratègia, torns simultanis

Abstract– This project consists of the design, development, and testing of a competitive turn-based strategy video game with the aim of reinforcing the learning of basic programming concepts through an interactive experience. The project is inspired by a board game with similar ideas, but whose mechanics are too complex for people who are new to programming. The development is carried out using the Unity engine and is organized following the Kanban agile methodology in order to ensure a controlled evolution of the product. The game features competitive matches in which two or more players control a toy car. The main objective is to obtain an object located at the center of the map and escape. The distinctive element of the project lies in the possibility of modifying the terrain through configurable tiles with an event system that allows behaviors such as terrain changes or interactions with players to be defined.

Keywords– Video game, Unity, Kanban, Programming, strategy video game, simultaneous turns

1 INTRODUCCIÓ

AQUEST treball sorgeix d'un joc de taula que vaig crear fa uns anys en el qual s'utilitzaven quadrats de paper dibuixats com a tauler on s'escrivien tota mena de normes i guanyava qui travessés el tauler amb el seu personatge. La gràcia del joc de taula és que tothom pot dibuixar la seva peça del tauler i escriure-hi normes.

- E-mail de contacte: 1666458@uab.cat
- Menció: Enginyeria de Computació
- Treball tutoritzat per: Enric Martí Gòdia (Departament de Ciències de la Computació)
- Curs 2025/26

A partir d'aquesta idea, es va pensar a desenvolupar un videojoc amb Unity amb una versió més senzilla del joc. La plataforma digital en simplifica la distribució i possibilita la implementació de funcionalitats addicionals com el multijugador en línia. A més, el videojoc té un clar component educatiu: pretén ser una eina per iniciar-se en la programació, doncs el joc permet escriure normes en les fitxes amb una interfície de text basada en paraules clau i permet veure com aquestes regles interactuen en el joc. Pel que fa al gènere, el projecte s'emmarca dins dels videojocs d'estratègia per torns, específicament en un subgènere anomenat "joc d'estratègia per torns simultanis" o també anomenat "WeGo". Aquest gènere es caracteritza per la planificació prèvia de moviments, l'anticipació de

les accions del rival i la presa de decisions basada en informació parcial o total de l'estat del joc. A diferència dels jocs en temps real, el sistema per torns afavoreix la reflexió i l'anàlisi. A més, l'ús de torns simultanis redueix el temps d'espera i afavoreix més la predicció de moviments de l'enemic. Finalment, el fet que existeixin pocs videojocs que combinin la modificació del terreny, estratègia per torns simultanis i un enfocament centrat en la programació reforça la proposta del projecte. Desenvolupar-lo com a Treball Final de Grau permetrà avaluar si el concepte és viable, atractiu i potencialment ampliable en futures versions.

Per tot això, el principal objectiu del projecte és el disseny, implementació i validació de *Under My Rules*, videojoc del gènere d'estratègia per torns simultanis. Les funcionalitats o objectius més concrets són:

- **Dissenyar i implementar un sistema de programació de fitxes:** Desenvolupar una interfície d'usuari i un sistema de lògica que permeti als jugadors modificar el comportament de les seves fitxes amb un mínim de cinc regles diferents.
- **Optimitzar el rendiment del sistema de físiques i moviment:** Aconseguir una jugabilitat fluida mitjançant un desplaçament d'objectes basat en físiques, garantint un rendiment estable d'un mínim de 60 FPS (fotogrames per segon) en la plataforma objectiu.
- **Desenvolupar una IA capaç d'interpretar i jugar al joc:** Programar un algoritme de decisió que sigui capaç d'avaluar l'estat de la partida i executar jugades totalment legals amb un temps de resposta inferior a 1 segon per torn.

La metodologia de treball Kanban és una manera d'organitzar visualment el treball amb taulers Kanban (figura 1). Aquests taulers s'organitzen per columnes que representen etapes de treball. Per a aquest projecte, el tauler s'ha estructurat en quatre etapes: "Per fer" (tasques pendents), "En procés" (tasques en curs), "Testejant" (proves de funcionament del codi) i "Fet" (tasques finalitzades). Kanban permet visualitzar la càrrega de treball i fomenta la millora contínua, doncs limita el nombre de tasques pendents abans de començar-ne altres, evitant la dispersió i ajudant a identificar colls d'ampolla. Això resulta ideal per a un desenvolupament individual, on la sobrecàrrega de tasques pot estancar el treball. D'altra banda, la definició i

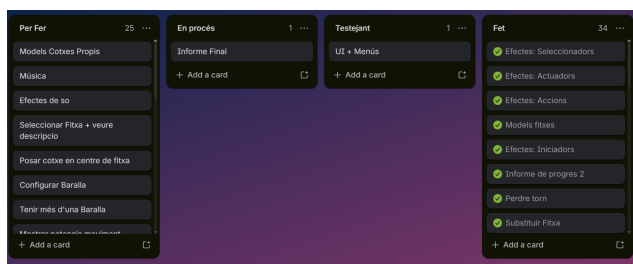


Figura. 1: Tauler Kanban fet amb Trello [1].

granularitat de les tasques s'ha alineat amb una estratègia

de desenvolupament iteratiu i incremental, basat en la creació de Prototips Mínims Viables (MVP) [18]. En el desenvolupament de videojocs, aquest enfocament permet validar mecàniques de joc ràpidament sense invertir temps en detalls secundaris. Per exemple, en lloc de planificar una tasca abstracta com "Dissenyar i programar la Classe Fitxa" (que requeriria preveure tots els mètodes futurs), s'ha optat per definir tasques orientades a la funcionalitat com "Posar Fitxa". Això permet desenvolupar simultàniament la lògica mínima del tauler i de la fitxa per fer funcionar la mecànica, fent que el progrés del videojoc sigui tangible des de les primeres fases.

Les eines de desenvolupament que hem utilitzat són:

- **Unity** [2] (la versió més recent, Unity 6.3) per al desenvolupament del gameplay.
- **Unity asset store** [3] per obtenir els models 3D.
- **Unity Version Control** [4] per mantenir un historial de versions.
- **Trello** [1] per organització de tasques amb metodologia Kanban.

El document s'estructura començant pels referents de l'estudi, per continuar amb el disseny i les mecàniques de joc (GDD), la seva implementació tècnica i la intel·ligència artificial (Gameplay), i la posterior validació mitjançant proves amb usuaris (Resultats i tests). El treball es completa amb les conclusions, la justificació de l'ús de la IA generativa, els agraïments i les referències bibliogràfiques.

2 REFERENTS

El gènere d'estratègia per torns simultanis es caracteritza per separar el joc en fases o torns. Això permet als jugadors planificar les seves decisions sense tenir pressió pel temps, de manera que la rapidesa o els reflexos tenen menys importància en aquest gènere. Sol predominar l'ús de caselles quadrades o hexagonals en un tauler. Amb l'ús de torns simultanis (sistema *WeGo*) tots els jugadors han d'establir el seu ordre i llavors es passa a una fase de resolució, cosa que dona més èmfasi a la predicció dels moviments dels enemics. A més, els torns simultanis són molt utilitzats en jocs de taula, ja que redueixen dràsticament el temps d'espera.

A continuació, s'analitzen els tres referents principals que han influït directament en el disseny i les mecàniques de *Under My Rules*: *Hextraction*, *Magic the Gathering: Arena* i *Ultimate Chicken Horse*.

2.1 Hextraction

Hextraction [5] és un joc de taula de codi obert creat l'any 2016 pel youtuber Zack Freedman i la seva companyia Voidstar Lab LLC [10]. El joc està basat en fitxes hexagonals impreses en 3D que es col·loquen sobre un tauler inclinat. L'objectiu dels jugadors és guiar una bala des de la part superior fins a la meta, modificant el camí a través de les propietats físiques de cada fitxa col·locada (figura 2). Algunes d'aquestes fitxes inclouen codis QR que afegixen regles lògiques o excepcions al flux del joc.

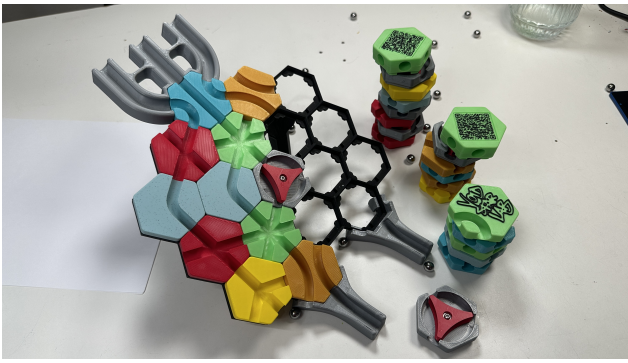


Figura. 2: Tauler d'Hextraction amb fitxes posades al tauler i d'altres apilades al costat.

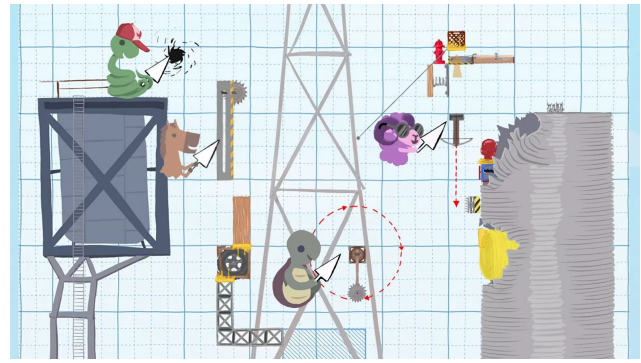


Figura. 4: Partida de Ultimate Chicken Horse en fase de construcció, on els 4 jugadors decideixen on posar el seu obstacle.

2.2 Magic: The Gathering Arena

Magic: The Gathering Arena [6] de Wizards of the coast [11] llançat el novembre de l'any 2017 com l'adaptació digital del conegut joc de cartes col·leccionables (figura 3). Destaca especialment pel seu complex sistema de regles automatitzades i, sobretot, per la fase prèvia de personalització d'estratègies a través de la creació de baralles (*deckbuilding*). Els jugadors seleccionen un conjunt limitat de cartes que defineixen el seu estil de joc abans d'entrar a la partida.



Figura. 3: Partida de Magic: the gathering Arena, on el jugador activa una de les seves cartes contra el rival.

mourà el seu personatge, el qual avança fins a col·lidir amb una paret o un altre jugador. L'objectiu final és capturar un objecte central equidistant i retornar a la posició inicial sense ser eliminat.

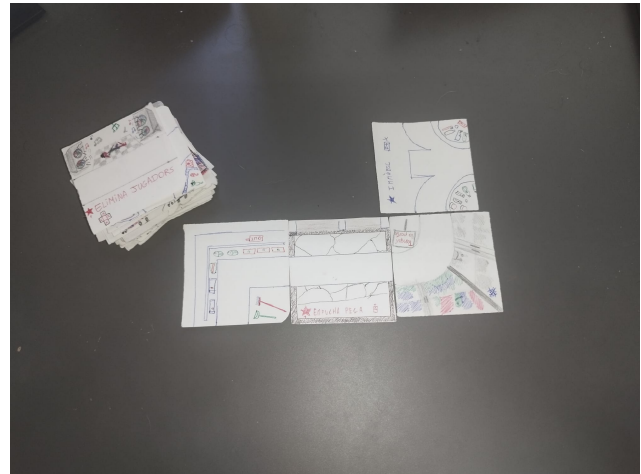


Figura. 5: Set de fitxes de Magmualitats que comparteixen un camí per on poder passar. Algunes fitxes tenen una estrella vermella que marquen que la fitxa té un efecte especial.

2.3 Ultimate Chicken Horse

Desenvolupat per Clever Endeavour Games [12] i publicat al març de l'any 2016, *Ultimate Chicken Horse* [7] és un joc competitiu de plataformes 2D que utilitza un bucle de joc dividit en dues fases col·laboratives/competitives (figura 4). A cada ronda, els jugadors passen per una fase de construcció simultània on col·loquen obstacles o plataformes en un mapa, i posteriorment una fase de resolució on intenten arribar a la meta superant el recorregut resultant.

2.4 Magmualitats

Magmualitats és un joc de taula en fase de prototip dissenyat per mi mateix a l'agost de l'any 2024 per a partides d'entre dos i sis jugadors. El joc es desenvolupa sobre una graella dinàmica on cada participant disposa del seu propi conjunt de fitxes quadrades amb camins i efectes especials (figura 5). A cada ronda, els jugadors decideixen simultàniament quina fitxa col·locar i la direcció en què es

3 GDD

Per al disseny del videojoc s'ha redactat un GDD (Game Design Document) que resumim en aquesta secció a través dels punts següents:

1. Visió general
2. Comparativa
3. Bucle principal del joc
4. Sistema de fitxes programables
5. Controls
6. Art
7. Interfície d'usuari

3.1 Visió general

Under My Rules repta els jugadors a pilotar un cotxe de joguina per tal d'aconseguir un objecte central i tornar a la casella de sortida, evitant que els rivals li prenguin i portin l'objecte cap a la seva pròpia base. L'escenari consisteix en un terreny incomplet que s'ha d'anar construint modularment mitjançant fitxes de terreny dels participants. La figura 6 mostra un moment del joc on es distingeix el vehicle del jugador (cotxe verd), l'objectiu (una moneda groga) situat sobre una plataforma, quatre fitxes ja col·locades al mapa (dues de planes i bàsiques, una rampa i una estructura amb tres parets) i tres fitxes a la part inferior que componen la mà del jugador (un cub, una rampa i una peça bàsica). Així

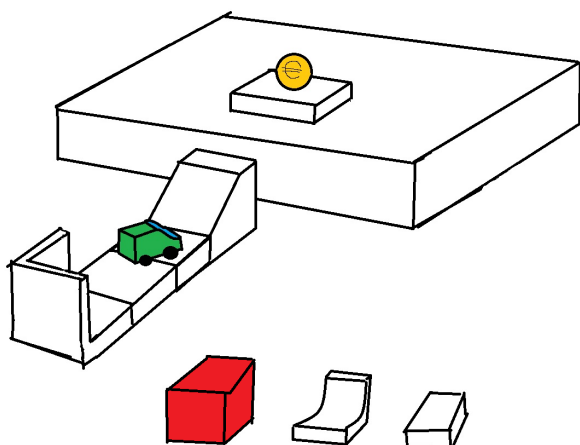


Figura. 6: Moment del joc abans del torn dels jugadors

mateix, la figura 7 il·lustra el desenvolupament del torn: el jugador ha incorporat una nova fitxa (un cub darrere de la moneda per tal d'evitar la caiguda) i ha desplaçat el cotxe, el qual es disposa a saltar a través de la rampa.

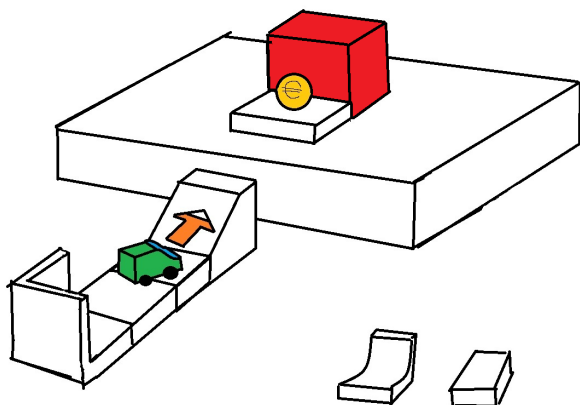


Figura. 7: Moment del joc després del torn del jugador, posant la fitxa sobre la plataforma i movent el cotxe

3.2 Comparativa

A partir dels referents analitzats, es detallen els aspectes clau extrets per a aquest treball i els elements diferenciadors del projecte:

- **Hextraction:** S'evoluciona la idea del joc de taula original substituint la gravetat per un motor de físiques

digital i transformant els codis QR en un sistema de programació d'events integrat a les fitxes. A més, es modifica el tauler per permetre una col·locació tridimensional (3D) de les peces aprofitant l'alçada.

- **Magic the Gathering: Arena:** S'adopta el concepte de preparació de la "baralla" de fitxes. A *Under My Rules*, la personalització inicial restringeix i defineix quines fitxes utilitzarà el jugador durant la partida i quines normes hi haurà establertes, evitant l'accés il·limitat al repertori complet.
- **Ultimate Chicken Horse:** És el referent principal per al disseny del bucle de joc. S'adapta l'estructura dividida en dues fases: una de col·locació de peces de manera pausada i sense pressió temporal, i una posterior d'execució on el vehicle del jugador interacciona mitjançant simulació amb els obstacles col·locats.
- **Magmualitats:** Com a antecedent directe, inspira les mecàniques base. En el nostre projecte es manté el nucli competitiu de resolució simultània de moviments i la quadrícula modular. No obstant això, *Under My Rules* fa el salt al medi digital substituint el moviment lineal simple per una simulació física en 3D i automatitzant la detecció de condicions de les fitxes programables.

3.3 Bucle principal del joc

El desenvolupament d'una partida de *Under My Rules* s'estructura al voltant d'un bucle de joc basat en torns de planificació simultània. Cada jugador comença la partida amb una mà de 3 fitxes de terreny agafades aleatòriament de la baralla (bossa de fitxes) i llavors en cada ronda, el jugador planifica les dues accions següents, que s'executaran posteriorment:

1. **Planificació:** Posar una de les tres fitxes de la seva mà en qualsevol lloc del terreny de joc i amb qualsevol orientació. També, si no és el seu primer torn pot substituir una fitxa del terreny per una de les seves quedant-se així amb la peça substituïda.
2. **Moviment:** Moure el cotxe. Es pot escollir qualsevol de les 4 direccions cardinals (N, S, E, O) i es pot modular la força en què el cotxe surt disparat.

Quan tots els jugadors han decidit aquestes dues accions, es passa a la fase de moviment on es resolen aquestes accions simultàniament, el que significa que les fitxes dels jugadors es col·loquen en les seves posicions en el mateix moment (en cas que dos jugadors la posin en el mateix lloc no es posa cap) i els cotxes dels jugadors es mouen a la vegada amb possibilitat que xoquin entre ells. Un cop els cotxes s'aturen es posaran en la fitxa on han quedat i començarà el següent torn.

El joc anirà canviant entre fase de planificació i de moviment i els jugadors han d'aconseguir portar el seu cotxe fins a un lloc objectiu on hi ha un objecte únic en el centre del terreny i portar-lo fins a la casella d'inici.

3.4 Sistema de fitxes programables

Abans de cada partida el jugador pot modificar les normes de les seves fitxes. Mitjançant una interfície de text amb autocompletat amb tabulació es poden afegir diferents paraules anomenades efectes que defineixen el comportament de la fitxa davant certs esdeveniments.

Els efectes estan separats en diferents categories: actuadors, seleccionadors, iniciadors i accions.

- Els **actuadors** fan referència a qui o què rep l'acció. Serien com les variables en programació on totes tenen el seu tipus de dades.
 - **Fitxa:** Una fitxa en el terreny. Es pot obtenir amb els seleccionadors **Aquesta:** Referència a ella mateixa.
 - **Cotxe:** El cotxe d'un jugador. Es pot obtenir amb els seleccionadors:
 - * **Amb objecte:** El cotxe que tingui l'objecte en aquell moment.
 - * **Usuari:** El cotxe del jugador que ha tirat la fitxa.
- Els **iniciadors** permeten diferenciar quan comença l'efecte de la fitxa i iniciar l'acció que tenen descrita a dins. Seria equivalent a definir qui crida el codi en programació. Hi ha els següents iniciadors:
 - **En acabar torn:** Un cop ha acabat la fase de moviment.
 - **En Xocar:** Quan un cotxe xoca amb la fitxa.
- Les **accions** modifiquen l'actuador assignat. Hi ha l'acció **Destrueix:** Elimina una fitxa del terreny o un cotxe. Els cotxes eliminats perden un torn i tornen a la seva fitxa inicial.

3.5 Controls de joc

Es prioritza un esquema de controls senzill que optimitzi la visualització, prenent el teclat de joc i el ratolí com a dispositius principals. Les interaccions es resumeixen a la taula 1.

Perifèric	Acció associada
Clic esquerre	Selecció, col·locació de fitxes i moviment de càmera.
Clic dret	Rotació de càmera i cancel·lació d'accions.
Roda ratolí	Zoom de la càmera.
WASD / Fletxes	Conducció del vehicle.
Shift / Espai	Control de l'alçada vertical de la càmera.

TAULA 1: ESQUEMA DE CONTROLS DE *Under my Rules*.

3.6 Art i història

L'ambientació del videojoc trasllada el jugador a un entorn quotidià: una habitació infantil o sala de jocs integrada per elements domèstics i joguines disperses. Aquest context justifica la premissa narrativa i el disseny dels escenaris, concebuts des d'una perspectiva a petita escala.

Pel que fa a l'apartat visual, es persegueix un estil artístic basat en el realisme estilitzat. L'objectiu és transmetre la sensació d'interactuar amb objectes tangibles i joguines físiques. Per aconseguir-ho, predomina una paleta de colors saturats i vius, combinada amb la definició de materials clars i reconeixibles com el plàstic injectat o la fusta polida (figures 8 i 9). La il·luminació i els acabats de les textures juguen un paper clau per emfatitzar les propietats físiques dels components del terreny i del propi vehicle. Els diferents models tridimensionals desenvolupats per a les fitxes es poden consultar a l'apèndix A.1.

3.7 Interfície d'usuari

La UI del joc està dividida en dues interfícies: la de menú i la de partida.

1. **Menú:** En la interfície de menú (figura 8) es poden programar les fitxes amb el sistema d'events (rectangle negre). També permet començar una partida i sortir del joc.



Figura. 8: Interfície de menú de *Under My Rules*

2. **Partida:** Mostra la interfície de la partida (figura 9), on hi ha el terreny de joc amb el tauler, a sota hi ha les fitxes del jugador i un text que explica els controls i a dalt a l'esquerra hi ha un botó (Sortir) per tornar a la interfície de menú.



Figura. 9: Interfície de partida de *Under My Rules*

4 GAMEPLAY

El desenvolupament d'un videojoc d'estratègia i lògica programable com *Under My Rules* requereix una infraestructura de programari sòlida, capaç de gestionar de manera eficient la concurrència d'esdeveniments, la simulació física en temps real i la interacció dinàmica de l'usuari. L'arquitectura del programari esdevé, per tant, el pilar fonamental que garanteix la viabilitat tècnica del projecte, definint com s'estructuren, es comuniquen i coexisteixen els diferents components del sistema.

Per a explicar l'arquitectura del gameplay s'ha fet un diagrama de mòduls de l'aplicació (figura 10). Hi ha els següents 8 mòduls: Interfície, IA, Jugadors, Inventari, Compilador d'events, Tauler, Entitats i Lògica Partida.

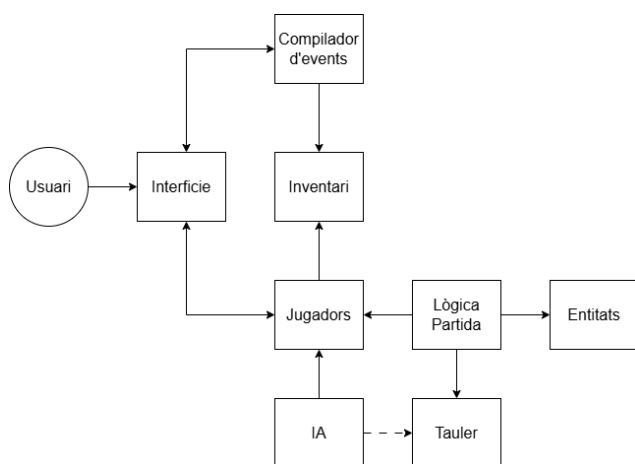


Figura. 10: Diagrama de mòduls de *Under My Rules*. Les fletxes representen les dependències i canals de comunicació entre els diferents mòduls (rectangles) durant l'execució del joc. L'usuari (cercle) es comunica directament amb la interfície del joc.

4.1 Interfície

Aquest mòdul actua com el canal de comunicació bidireccional entre l'usuari i el sistema; d'una banda, s'encarrega de la representació visual del joc en pantalla i, de l'altra, captura les interaccions del jugador mitjançant els dispositius perifèrics d'entrada (teclat i ratolí). A nivell d'arquitectura, aquest mòdul es comunica directament amb el mòdul de Jugadors. Per optimitzar l'experiència d'usuari i la captura d'events, s'ha dissenyat un flux d'entrada basat en una seqüència d'estats tancada. Des del punt de vista de la implementació tècnica, a cada fotograma del joc es processen seqüencialment el flux de dades i les entrades de l'usuari.

Mitjançant una màquina d'estats local i l'ús d'estructures de control (com ara vectors de rotació tridimensional, definició de màscares de capa del tauler per al llançament de rajos, punts de clic inicial i un llindar de seguretat per discernir l'arrossegament del ratolí), el jugador determina ordenadament les accions següents:

1. Selecció d'una fitxa de la mà activa.
2. Determinació de la posició de la fitxa sobre la capa corresponent del tauler.

3. Rotació de la fitxa en l'eix corresponent calculada a partir del vector d'arrossegament de l'usuari.
4. Selecció de la direcció inicial de l'impuls del vehicle.
5. Configuració de la força d'impuls del cotxe.

Quan s'ha completat satisfactòriament aquesta seqüència, el sistema activa un indicador booleà de control que té la funció d'avisar asíncronament la lògica de la partida que la planificació d'aquest jugador ha finalitzat. Així mateix, aquest mòdul disposa d'una referència directa al controlador de la càmera per adaptar el focus i el camp visual de l'usuari durant el procés interactiu de col·locació de peces.

4.2 Jugadors

El mòdul de Jugadors defineix l'estructura conceptual i l'encapsulament de dades dels participants de la partida. Per aconseguir un disseny desacoblant i adherent als principis de la programació orientada a objectes, s'utilitza una arquitectura polimòrfica basada en una classe abstracta o base comuna. Això permet tractar qualsevol participant de forma equivalent independentment de la seva naturalesa.

La classe base actua com un contenidor d'informació estructural i dades persistents amb un identificador numèric únic, el nom del jugador i una referència al seu inventari privat.

D'aquesta classe es generen dues subclasses:

- **Persona:** No conté lògica de decisió interna; la seva única funció és connectar-se amb el mòdul d'Interfície per recollir i emmagatzemar les dades introduïdes per l'usuari.
- **Intel·ligència Artificial (IA):** Sobreescriu les rutines de planificació buides de la classe mare per delegar de manera directa l'elecció de fitxes i moviments al mòdul autònom d'intel·ligència artificial, registrant la decisió òptima obtinguda en l'estructura compartida de la partida.

4.3 Inventari

Aquest component s'encarrega exclusivament de la persistència, emmagatzematge i distribució de les fitxes durant la sessió de joc. L'arquitectura separa clarament la informació estàtica immutable de la gestió dinàmica de memòria. La informació base es defineix en una estructura de magatzem o bossa global que conté un catàleg preconfigurat d'objectes.

En temps d'execució, el control operatiu recau en una classe d'inventari dinàmic vinculada a cada jugador. Aquest objecte gestiona els recursos mitjançant dues estructures de dades ben diferenciades:

- **La mà activa:** Una llista dinàmica de capacitat limitada que allotja les entitats de fitxa que el jugador té disponibles per col·locar de forma immediata.
- **La bossa de partida:** Una pila estructural de tipus LIFO (Last In, First Out) que emmagatzema les peces restants de la seva baralla privada.

El mòdul exposa procediments d'inicialització encarregats de rebre la bossa estàtica, instanciar els elements estructurals dins de la pila de memòria i realitzar les primeres extraccions aleatòries per poblar la mà al començament de la partida.

4.4 Tauler

El mòdul del Tauler és el responsable de mapejar, indexar i controlar el mapa de joc. Tradueix l'espai continu tridimensional del motor gràfic a una matriu lògica integrada per cel·les conceptuais. La seva funció principal és servir de registre centralitzat, emmagatzemant referències creuades de quins cotxes i quines fitxes ocupen cada coordenada de la quadrícula en un instant donat.

4.5 Entitats

Aquest mòdul comprèn les representacions físiques i objectes dinàmics que coexisteixen i interactuen dins l'escena. Es troba altament vinculat al sistema de col·lisió i mecànica de cossos rígids, actuant com un captador d'esdeveniments que registra els xocs i interaccions per transmetre'ls posteriorment al controlador general. Les entitats es divideixen en dues categories:

- **Fitxa en Partida:** És la manifestació física d'una peça col·locada sobre la quadrícula. Enllaça la informació estàtica del catàleg amb dades temporals de joc, com ara l'identificador del jugador propietari (que pot variar mitjançant la substitució de fitxes). Disposa de mètodes per projectar caixes de col·lisió cúbiques colineals amb les dimensions d'una cel·la lògica, facilitant-ne la correcta orientació.
- **Cotxe en Partida:** Gestiona el comportament dinàmic del cotxe de cada jugador a través de components mecànics i detectors de col·lisió per contacte. Integra mecanismes de captura d'objectes; quan el vehicle interactua amb la corona objectiu central, el sistema emparenta jeràrquicament l'objecte recollit al cotxe, incloent també les rutines inverses per a l'alliberament de l'element quan es produeix una pèrdua o un robatori.

4.6 Lògica Partida

El mòdul de la lògica de la partida coordina tots els altres mòduls. Controla el transcurs de la partida amb un cicle seqüencial cridant als mòduls de Jugadors, Tauler i Entitats. Aquest mòdul s'estructura en una màquina d'estats finits dins del bucle principal del joc. El seu flux passa per quatre fases:

1. **Planificació:** Es demana de manera asíncrona a tots els jugadors que determinin la seva següent acció.
2. **Preparació:** Es comprova que no hi hagi decisions conflictives dels jugadors com voler posar una fitxa en la mateixa casella i prepara els mòduls de Tauler i Entitats per a la següent fase.
3. **Moviment:** S'aplica el moviment a tots els jugadors i s'espera que tots els cotxes s'aturin.

4. **Manteniment:** Es demana als jugadors que robin noves fitxes i fa un control dels jugadors que s'han quedat sense torn.

4.7 Compilador d'events

Per a la compilació del codi que s'escriu en les fitxes s'ha utilitzat una arquitectura inspirada en el compilador de LOOS [9], desenvolupat a l'assignatura de Compiladors, tot i que simplificada per als requeriments del videojoc. El sistema fa servir el mateix principi de regles terminals i no terminals per definir la gramàtica del llenguatge visual de *Under My Rules*, articulant el procés en tres fases clàssiques:

1. **L'anàlisi sintàctica:** Valida que la seqüència de tokens segueixi l'estructura gramatical establerta. Inicialment, el disseny teòric del llenguatge contemplava una sintaxi més complexa que permetia condicionals i accions amb actuadors encadenats, definida per la següent especificació EBNF:

```
Iniciador {Condicional "("paràmetres
              ")"acció actuador "."selector } / (
              acció actuador "."selector {"","actuador
              "."selector} )
```

No obstant això, amb l'objectiu d'acotar l'abast del projecte i garantir una experiència de joc fluida i comprensible, la gramàtica final s'ha simplificat cap a una estructura purament lineal:

```
Iniciador Acció Actuador.Selector
```

A diferència del compilador de LOOS, que gestiona estructures molt més complexes com bucles, condicionals niuats o classes, la nostra gramàtica actual és completament lineal, fet que redueix dràsticament la complexitat computacional del parsing.

2. **L'anàlisi semàntica:** S'assegura que els tipus de dades siguin coherents; per exemple, valida que l'actuador "Cotxe" no pugui rebre un selector incorrecte com "Aquesta".
3. **La generació de codi** funciona de manera diferent de com es fa a LOOS, ja que el compilador de Under My Rules actua com un intèrpret en temps d'execució. En superar l'anàlisi semàntica, no es genera codi executable intermedi, sinó que es construeix una estructura de dades en forma d'arbre de sintaxi abstracta (*Abstract Syntax Tree*, AST). Durant la partida, quan es dispara un iniciador d'una fitxa, Unity avalua de manera seqüencial l'arbre generat i executa les funcions del joc corresponents. Per exemple, en l'arbre representat a la figura 11, l'intèrpret avalua primer que es compleixi l'iniciador i, posteriorment, executa l'acció sobre l'actuador seleccionat.

4.8 Intel·ligència Artificial (IA)

El desenvolupament d'un algorisme capaç de predir i planificar rutes en un entorn basat en físiques tridimensionals dinàmiques presenta una gran complexitat. Inicialment, es va considerar la implementació d'algorismes clàssics de

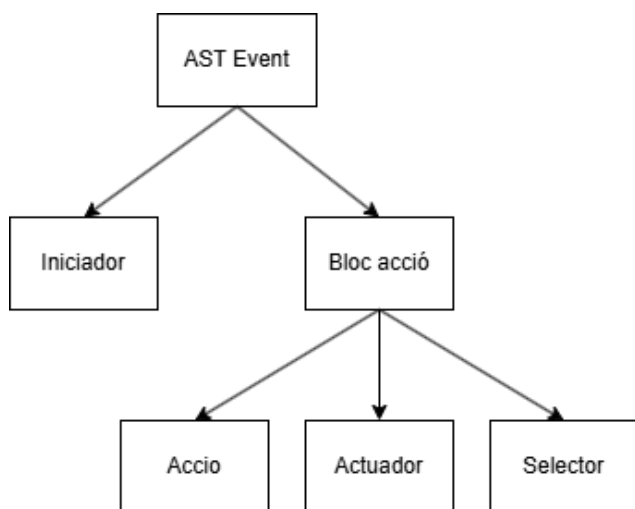


Figura. 11: AST generat per la instrucció: EnXocar Destrueix Fitxa.Aquesta. On EnXocar és l'iniciador, i el bloc d'acció el formen l'acció Destrueix, l'actuador Fitxa i el selector Aquesta.

cerca de camins, com A^* [14] o *Dijkstra* [13]. No obstant això, es van descartar ràpidament a causa de la impossibilitat de mapejar de forma eficient el tauler com un graf rígid, atès que cada fitxa pot tenir orientacions variables i els moviments del vehicle generen resultats caòtics i indeterministes per la fricció i les col·lisions. Per solucionar aquest problema, s'ha optat per un enfocament de simulació predictiva en segon pla aprofitant les capacitats de l'API de físiques de Unity. En lloc de calcular analíticament la trajectòria, la IA executa simulacions virtuals i invisibles per a l'usuari abans de prendre una decisió. A cada torn, el ventall de decisions del cotxe de la IA es redueix a un espai d'estats de 16 opcions possibles (4 velocitats * 4 direccions). El procés de decisió es divideix en dues fases mitjançant funcions heurístiques:

1. **Fase de Col·locació de la Fitxa:** Com que l'espai de combinacions de tauler seria massa gran per simular-lo completament, s'utilitza una funció d'avaluació més simple per a la col·locació de fitxes. La IA selecciona la casella buida que minimitza la distància euclidiana cap a l'objectiu, aplicant un criteri de prioritització per a aquelles caselles que siguin directament adjacents a la posició actual del seu cotxe.
2. **Fase de Moviment del Cotxe:** Un cop s'ha decidit la fitxa, per a cadascuna de les 16 opcions de moviment disponibles, la IA duu a terme una simulació predictiva invisible per al jugador. Això s'aconsegueix forçant l'avanç del motor de físiques de manera manual mitjançant el mètode *Physics.Simulate* de Unity en un sol fotograma. D'aquesta manera, es calcula a l'instant la posició vàlida final de repòs del vehicle (assegurant que no hagi caigut del tauler) i s'avalua mitjançant la distància euclidiana respecte a l'objectiu actual (la corona central o la seva fitxa inicial). Finalment, s'escull l'opció que minimitza aquesta distància.

Aquest enfocament híbrid entre simulació física en temps real i heurístiques geomètriques ha permès desenvolupar una IA competent, reactiva i amb un cost computacional

perfectament assumible per al rendiment del joc.

5 RESULTATS I TEST

En la versió final, el jugador disposa d'una interfície gràfica (UI) intuïtiva que li permet gestionar el seu inventari de fitxes, planificar el torn de manera asíncrona i simular de forma tridimensional les col·lisions i trajectòries del vehicle mitjançant el motor de físiques de Unity. Així mateix, s'ha implementat amb èxit el sistema d'Intel·ligència Artificial capaç de simular trajectòries virtuals per competir autònomament contra l'usuari.

Les figures 12, 13 i 14 mostren el desenvolupament d'un torn complet del videojoc i exemplifiquen el funcionament del bucle principal de joc. Inicialment, el jugador disposa d'una mà formada per tres fitxes de terreny (un cub, una tira i unes escales) i d'un terreny parcialment construït sobre el qual ha de planificar la seva estratègia. A continuació, el sistema permet seleccionar una de les fitxes i previsualitzar-ne tant la posició com l'orientació sobre el tauler abans de confirmar l'acció, proporcionant a l'usuari una representació clara de l'estat futur del terreny, en aquest cas el cub vora al centre del tauler (figura 13).

Finalment, la fase de resolució executa simultàniament les accions planificades per tots els jugadors i actualitza l'estat del tauler. En l'exemple mostrat, el vehicle del jugador és impulsat per una fitxa amb forma de rampa, salta sobre el terreny i col·lisiona amb el cub que s'acaba de col·locar, quedant finalment situat més a prop de la corona, que constitueix l'objectiu de la partida. Addicionalment, les fitxes ressaltades en color verd corresponen a les peces amb les quals el vehicle ha interactuat durant el moviment i que, d'acord amb les regles del joc, passen a ser candidates per ser substituïdes per noves fitxes del jugador en torns posteriors. Simultàniament, s'aprecia com els vehicles de la policia (controlats per la IA) completen el seu torn col·locant fitxes bàsiques i progressant cap endavant.



Figura. 12: Estat inicial d'un torn de partida.

D'altra banda, la figura 15 il·lustra la interfície de programació de fitxes integrada al menú principal. El sistema proporciona una entrada de text amb autocompletat que permet definir el comportament de les fitxes mitjançant el llenguatge específic desenvolupat per al projecte. Aquesta funcionalitat constitueix l'element diferencial principal de *Under My Rules*, ja que ofereix als jugadors la possibilitat de personalitzar les regles de joc i experimentar de manera interactiva amb conceptes bàsics de programació. En l'exemple es mostra la instrucció EnAcabarTorn

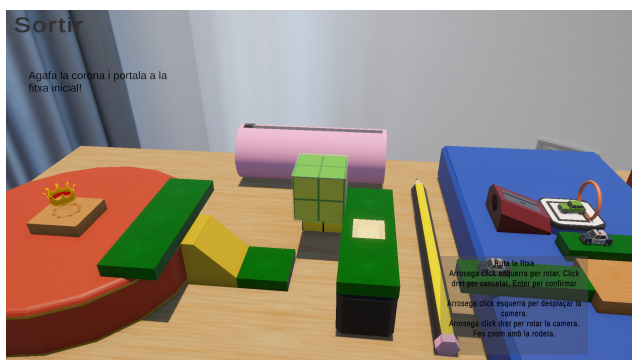


Figura. 13: Previsualització de la col·locació d'una fitxa sobre el tauler.



Figura. 14: Resultat de la resolució simultània del torn.

Destruïx Cotxe.AmbOb, mentre el sistema suggereix automàticament el selector AmbObjecte, evidenciant el funcionament del mecanisme d'autocompletat i d'assistència a l'escriptura de regles.



Figura. 15: Interfície de programació de fitxes amb autocompletat de regles.

Per tal de garantir l'estabilitat del programari i validar-ne l'experiència d'usuari i la corba d'aprenentatge, s'ha seguit un doble procés de validació: proves de desenvolupament internes i un estudi de test d'usabilitat extern.

5.1 Proves de desenvolupament internes

El procés de verificació del programari s'ha dut a terme de manera iterativa i incremental, estretament vinculat al taulell Kanban de gestió del projecte. En lloc d'executar una fase de llançament de proves tancada al final del projecte, cada funcionalitat o mòdul implementat s'ha sotmès a un cicle de proves de caixa negra i d'integració abans de ser mogut a la columna de tasques finalitzades. Durant aquesta fase, s'han utilitzat les eines de depuració natives de l'entorn de desenvolupament, com la consola d'esdeveniments de Unity i la inserció de punts de ruptura (breakpoints) a l'IDE. Això ha permès monitoritzar en temps real l'estat de les variables, el flux de l'execució de les instruccions i la correcta comunicació dels esdeveniments de joc, corregint errors de programació (bugs) de forma immediata.

5.2 Test d'usabilitat extern

Una vegada obtinguda una versió estable (*Release Candidate*), s'ha avaluat l'experiència de joc i l'accessibilitat mitjançant una mostra de control d'usuaris que han testejat el programari de forma autònoma. Posteriorment, han respost un formulari basat en les metodologies estàndard *System Usability Scale* (SUS) [15] i *Game Experience Questionnaire* (GEQ) [16]. Els resultats clau obtinguts es resumeixen en els següents punts:

- **Accessibilitat i controls:** Els usuaris han comprès ràpidament la navegació en l'espai tridimensional i el control de la càmera. El sistema de gestió d'elements del tauler (*drag and move*) ha mostrat resultats sòlids i homogenis, validant la implementació feta amb l'Input System de Unity.
- **Complexitat mecànica i disseny:** El joc es percep com a mecànicament complex i amb una dificultat elevada, fet que s'alinea amb l'objectiu de ser una eina orientada a la resolució de problemes lògics. Com a punt crític, el sistema de torn simultani (WeGo) genera una certa sobrecàrrega cognitiva en seguir les accions dels rivals, aspecte a solucionar en futures iteracions.
- **Experiència de joc i retenció:** Tot i la fricció inicial provocada per la manca d'un tutorial integrat i la necessitat d'explicacions prèvies, l'indicador de fidelització ha estat el més alt de l'estudi. El factor d'entreteniment i la intenció de tornar a jugar han obtingut una valoració excel·lent (4.67/5.00).

A partir del feedback qualitatiu rebut, s'han detectat dues deficiències inicials crítiques: una dificultat excessiva en la composició del tauler i problemes d'escalat en la font de la interfície. Ambdós problemes s'han resolt i refinat per a la versió final d'aquesta memòria.

6 CONCLUSIONS

Com a conclusions cal destacar:

- La realització d'aquest Treball de Final de Grau ha permès abordar el cicle complet de disseny i arquitectura del videojoc del gènere d'estratègia per torns simultanis, assolint una versió final estable, jugable i

completament funcional que valida la viabilitat tècnica del projecte.

- S'ha assolit l'objectiu de dissenyar un sistema de programació de fitxes mitjançant un compilador propi que genera un Arbre de Sintaxi Abstracta, permetent als usuaris modificar el comportament de les fitxes de manera intuïtiva amb les regles establertes.
- S'ha assolit l'objectiu d'optimitzar el rendiment del videojoc gràcies a la integració amb el motor de físiques natiu de Unity, aconseguint un comportament d'objectes fluid, previsible i orgànic que manté una taxa estable de 60 FPS a la plataforma objectiu.
- S'ha assolit l'objectiu de desenvolupar una intel·ligència artificial per al videojoc, basada en simulacions predictives del motor de físiques de Unity i heurístiques de distància, capaç d'interpretar l'estat de la partida i executar moviments vàlids dins del temps de resposta establert.

Els resultats obtinguts suggereixen que l'ús de fitxes programables constitueix un mecanisme viable per introduir conceptes bàsics de programació dins d'un entorn lúdic i competitiu.

Com a incidències principals, cal destacar:

- **Arquitectura i IA:** La necessitat d'integrar la IA com un usuari real va requerir refactoritzar la classe Jugador en una classe base abstracta. Tot i el temps invertit, s'ha aconseguit una arquitectura modular, robusta i fàcil de mantenir.
- **Nou Input System d'Unity 6:** L'adopció obligatòria d'aquest sistema va suposar una dificultat inicial a causa de la manca de documentació actualitzada de la comunitat, però la seva implementació ha permès resoldre de forma neta i nativa la dualitat de controls.

Com a millores futures per escalar el videojoc cap a un producte comercial, destacaria:

- **Implementació de multijugador en línia.** Desplegar una arquitectura servidor-client al núvol per permetre partides competitives en temps real, fomentant així la comunitat de jugadors.
- **Optimització de la IA.** Implementar un algorisme d'arbre de cerca Minimax perquè la intel·ligència artificial faci previsió de moviments a llarg termini i ofereixi una experiència monojugador més desafiant.
- **Expansió de contingut.** Dissenyar nous taulers amb obstacles dinàmics, nous tipus efectes per a les fitxes i modalitats de joc alternatives per enriquir la varietat del bucle de joc.

ÚS DE LA IA GENERATIVA

En aquest projecte la IA generativa només s'ha usat per:

- Millorar la redacció dels informes
- Consultes molt puntuals en aspectes tècnics de mètodes de llibreries de Unity/C# (com ara `List` o `Stack`)

AGRAÏMENTS

Primer de tot, vull agrair al meu tutor, Enric Martí Gòdia, per la seva ajuda i consells per a la definició del projecte i els seus objectius, així com per l'orientació que m'ha proporcionat per millorar la redacció i la claredat de cadascun dels informes com aquest.

Vull fer una menció especial als meus pares pel seu suport incondicional des del primer dia, i per haver estat els primers a testear cada versió del videojoc, aportant-me consells i la motivació necessària per portar aquest projecte fins a on és ara.

També vull agrair al meu germà, Dídac, el fet d'haver-me acompanyat en cada etapa del desenvolupament, respectant el meu espai de treball per garantir la meua concentració i aguantar les meves queixes en els moments de més estrès.

Finalment, vull donar les gràcies a totes les persones que han dedicat el seu temps a provar el videojoc, el seu feedback i entusiasme han estat l'empenta definitiva per fer realitat aquest treball.

REFERÈNCIES

- [1] Atlassian. (n.d.). Pàgina oficial de Trello. <https://trello.com> (Data darrer accés: Juny 2026)
- [2] Unity Technologies. (n.d.). pàgina oficial del motor de joc Unity. <https://unity.com> (Data darrer accés: Maig 2026)
- [3] Unity Technologies. (n.d.). pàgina oficial de la botiga de recursos Unity Asset Store. <https://assetstore.unity.com> (Data darrer accés: Març 2026)
- [4] Unity Technologies. (n.d.). pàgina oficial del controlador de versions Unity Version Control. <https://unity.com/features/version-control> (Data darrer accés: Maig 2026)
- [5] Hextraction. (2016). Hextraction. <https://www.playhextraction.com> (Data darrer accés: Febrer 2026)
- [6] Wizards of the Coast. (2019). Magic: The Gathering Arena [Videojoc]. Steam. https://store.steampowered.com/app/2141910/Magic_The_Gathering_Arena (Data darrer accés: Febrer 2026)
- [7] Clever Endeavour Games. (2016). Ultimate Chicken Horse [Videojoc]. Steam. https://store.steampowered.com/app/386940/Ultimate_Chicken_Horse (Data darrer accés: Juny 2026)
- [8] Unity Technologies. (n.d.). Unity Manual. <https://docs.unity3d.com/Manual/index.html> (Data darrer accés: Juny 2026)
- [9] Universitat Autònoma de Barcelona. (n.d.). Compilador Loos [Programari docent utilitzat a l'assignatura de Compiladors]. (Data darrer accés: Juny 2026)

- [10] Void Star Labs LLC. (n.d.). BoardGameGeek: Void Star Labs LC. <https://boardgamegeek.com/boardgamepublisher/54386/void-star-labs> (Data darrer accés: Maig 2026)
- [11] Wizards of the Coast. (n.d.). Wizards of the Coast. <https://company.wizards.com> (Data darrer accés: Maig 2026)
- [12] Clever Endeavour Games. (n.d.). Clever Endeavour. <https://www.cleverendeavourgames.com> (Data darrer accés: Maig 2026)
- [13] Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. *Numerische Mathematik*, 1(1), 269–271. (Data darrer accés: Juny 2026)
- [14] Hart, P. E., Nilsson, N. J., i Raphael, B. (1968). A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2), 100–107. (Data darrer accés: Juny 2026)
- [15] Brooke, J. (1996). SUS: A “quick and dirty” usability scale. En P. W. Jordan, B. Thomas, I. L. McClelland, i B. Weerdmeester (Eds.), *Usability evaluation in industry* (pp. 189–194). Taylor & Francis. https://www.researchgate.net/publication/228593520_SUS_A_quick_and_dirty_usability_scale (Data darrer accés: Juny 2026)
- [16] IJsselsteijn, W. A., de Kort, Y. A. W., i Poels, K. (2013). The Game Experience Questionnaire. Technische Universiteit Eindhoven. <https://research.tue.nl/en/publications/the-game-experience-questionnaire> (Data darrer accés: Juny 2026)
- [17] Unity Technologies. (n.d.). ProBuilder Manual. <https://docs.unity3d.com/Packages/com.unity.probuilder@6.1/manual/index.html> (Data darrer accés: Juny 2026)
- [18] Ries, E. (2011). *The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses*. Crown Business. (Data darrer accés: Juny 2026)

APÈNDIX

A.1 Models de Fitxes

En aquest apartat es presenten els models tridimensionals de les fitxes utilitzades en el projecte. El disseny i la modelització d'aquests elements s'han dut a terme mitjançant el paquet d'eines ProBuilder [17], integrat en el motor de desenvolupament Unity [2]. Totes les fitxes s'han dissenyat mantenint una mateixa altura, de manera que puguin encaixar correctament entre elles. A continuació, es detallen els diferents models generats.

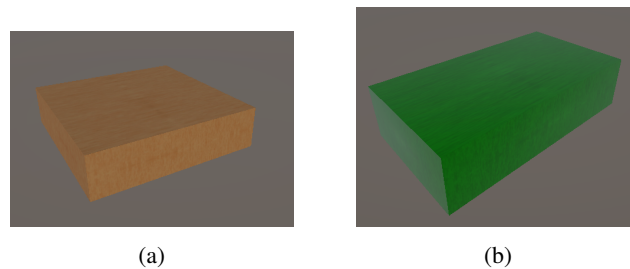


Figura. 16: Models de les fitxes bàsica (a) i tira (b), inspirades en peces dels jocs de construcció de fusta. La fitxa bàsica és el model més simple i permet el moviment en les quatre direccions. La fitxa tira, en canvi, només permet el moviment en dues direccions.

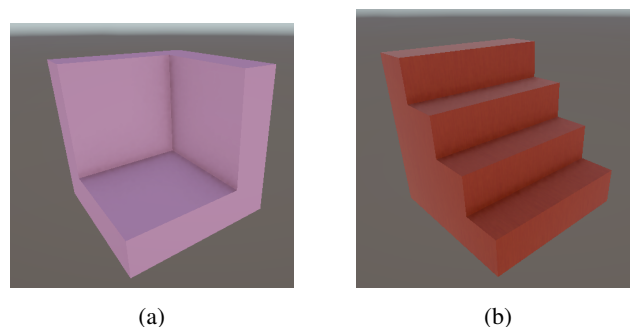


Figura. 17: Models de les fitxes tres parets (a) i escales (b), inspirades en peces dels jocs de construcció de fusta. La fitxa tres parets obliga el vehicle a aturar-se i canviar de direcció, mentre que la fitxa escales actua com un recorregut d'un únic sentit.

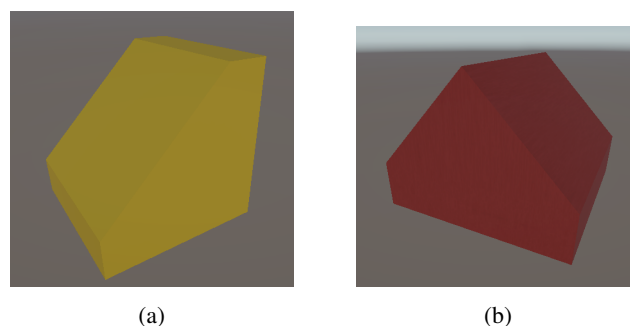


Figura. 18: Models de les fitxes rampa (a) i caseta (b), inspirades, respectivament, en peces de construcció i en les fitxes d'hotel dels jocs de taula. Ambdues permeten enlairar el vehicle, tot i que amb trajectòries diferents.

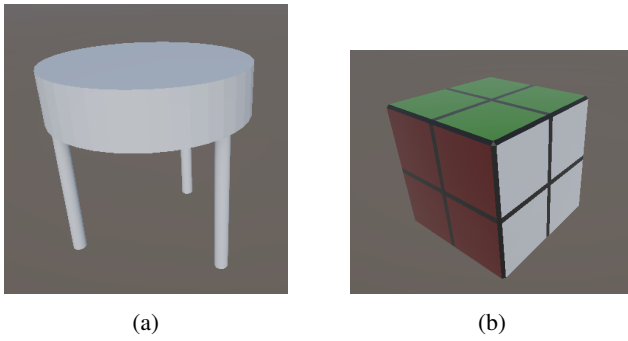


Figura. 19: Models de les fitxes tauleta de pizzas (a) i cub de Rubik (b). La fitxa tauleta actua com una intersecció en forma de T, bloquejant una direcció i permetent el pas per les altres, mentre que el cub de Rubik funciona com un obstacle que impedeix el pas.

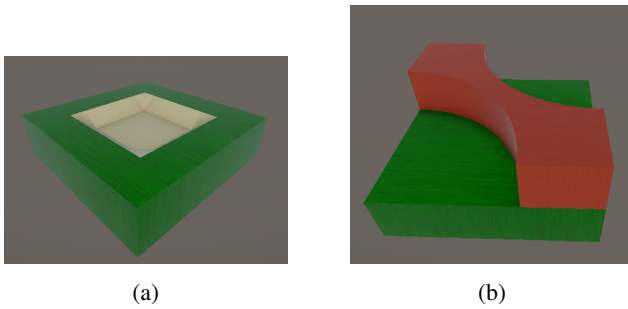


Figura. 20: Models de les fitxes búnquer (a) i gir (b), inspirades, respectivament, en els búnquers dels camps de golf i en peces de circuits de bales. La fitxa búnquer trava el pas del vehicle, mentre que la fitxa gir permet canviar de direcció sense necessitat d'aturar-se.

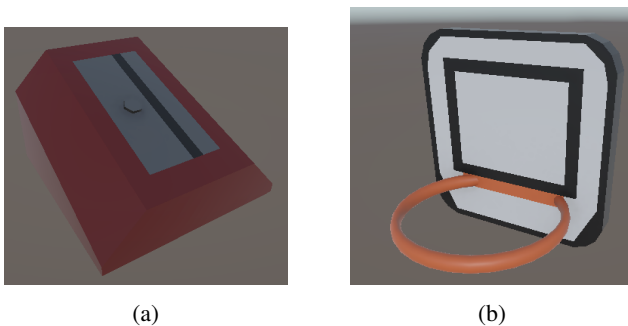


Figura. 21: Models de les fitxes maquineta (a) i cistella (b). La fitxa maquineta incorpora una rampa amb una inclinació més suau, que afavoreix salts més horitzontals, així com un forat que pot atrapar els vehicles. La fitxa cistella pot utilitzar-se com a barrera per facilitar determinats salts o, girant-la, actua com una fitxa bàsica.