

---

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

Lopez Peña, Pol; Cerdà Company, Xim, dir. Desarrollo de un juego didáctico 2D con Unity. 2023. (Enginyeria Informàtica)

---

This version is available at <https://ddd.uab.cat/record/298948>

under the terms of the  license

# Desenvolupament d'un joc didàctic 2D amb Unity

Pol López Peña

**Resum**—Aquest projecte presenta el desenvolupament d'un videojoc educatiu en 2D, dissenyat per a ensenyar nocions bàsiques de programació a principiants. Utilitzant el motor de videojocs **Unity[1]**, s'ha creat una activitat interactiva que combina elements de l'entreteniment i l'educació, facilitant la comprensió dels conceptes més bàsics de programació de manera lúdica i accessible. El videojoc s'estructura en diversos nivells progressius, cadascun enfocat en un concepte específic de programació, com poden ser variables, estructures de control, bucles, etc. Basat en la metodologia d'aprenentatge gamificació, aquest projecte utilitza petits exercicis en el joc, per tal que els jugadors apliquin els conceptes de programació en contextos pràctics, permetent un aprenentatge més lleuger i interactiu. El desenvolupament del joc ha implicat la integració de diverses tècniques i eines dins de Unity, per a poder aconseguir que el codi escrit per l'usuari s'executés correctament. Amb aquest projecte s'ha intentat demostrar el potencial dels videojocs com a eines complementàries en l'ensenyament, oferint una alternativa innovadora i efectiva per a l'aprenentatge de conceptes tècnics en un entorn interactiu i motivador.

**Paraules clau**— educació, programació, Unity, videojoc didàctic

**Abstract**—This project presents the development of a 2D educational video game designed to teach basic programming concepts to beginners. Using the Unity game engine, an interactive activity that combines elements of entertainment and education has been created, facilitating the understanding of fundamental programming concepts in an engaging and accessible manner. The game is structured in several progressive levels, each focused on a specific programming concept, such as variables, control structures or loops. Through simple in-game exercises, players apply these concepts in practical contexts, allowing for a more lighthearted and interactive learning experience. The development of the game involved integrating various techniques and tools within Unity to ensure that the code written by the user executed correctly. This project aims to demonstrate the potential of video games as complementary tools in education, offering an innovative and effective alternative for learning technical concepts in an interactive and motivating environment.

**Index Terms**— education, programming, Unity, didactic video game

---

## 1 INTRODUCCIÓ - CONTEXT DEL TREBALL

EN un inici, la proposta inicial del projecte era la creació d'un **joc de plataformes[2]** 2D des de zero utilitzant el motor Unity. El problema és que actualment hi ha milers de jocs com aquest, pel que era molt complicat afegir un factor innovador a la jugabilitat. Per aquest motiu, s'ha optat per una petita redefinició del projecte.

En aquesta redefinició, s'ha pensat en crear un sistema de joc on el jugador haurà d'escriure **snippets[3]** de codi per a poder superar cadascun dels nivells. D'aquesta manera, alhora que juga i avança en el joc, l'usuari també adquireix nous coneixements sobre programació.

Aquests tipus de jocs són una molt bona forma de començar a aprendre i sobretot despertar la curiositat per la programació als més joves. Aquesta tècnica d'utilitzar mecàniques de videojocs per a l'àmbit educatiu s'anomena

gamificació i va començar-se a popularitzar al voltant del 2008. El nombre de docents que decideixen aplicar la tècnica creix cada cop més, rebent resultats positius per part dels alumnes en forma de més motivació a les activitats a l'aula i per tant un major aprenentatge. Mecàniques extremes dels videojocs com rankings i rejugabilitat fan que els alumnes s'interessin en tornar a realitzar les activitats per posteriorment consultar tant els seus resultats com els dels seus companys impulsant així també un component social en forma tant de competitivitat com de treball en equip. La tendència de l'ús de jocs digitals com a forma d'adquirir coneixements ha sigut tal que fins i tot algunes empreses han començat a aplicar-lo com a millora de la experiència dels seus empleats [4]

Pel que fa a l'àmbit de la programació, jocs com el famós Scratch són la primera aproximació que tenen moltes persones al món de la programació i la raó per la qual s'han s'hi han interessat.

Els propis creadors d' Scratch diuen que veien la programació com a una nova literatura que havia de ser accessible per a tothom ja que la codificació és una pràctica que ajuda a organitzar i expressar idees igual que qualsevol altre

- 
- E-mail de contacte: 1603460@uab.cat
  - Menció realitzada: *Enginyeria del Software*
  - Treball tutoritzat per: *Xim Cerdà Company (CVC)*
  - Curs 2023/24

llenguatge. En el seu sistema es programa utilitzant per una banda uns objectes, anomenats sprites, i per l'altre una sèrie d'accions representades en peces similars a un puzzle que es poden combinar per a aconseguir que els sprites realitzin comportaments determinats creant així petits jocs que ajuden a desenvolupar el pensament lògic en els més petits [5].

Tot i així, podríem dir que en aquest joc no s'escriu codi com a tal, sinó que és més una aproximació als conceptes bàsics de la programació com condicionals, bucles, etc. Uns exemples diferents són **CodeCombat**[6] o **CodinGame**[7], on els conceptes són més avançats i més propers a la programació pròpiament dita.

Personalment aquests últims jocs són el motiu que em va portar a interessar-me pel món de la programació i, posteriorment, per la creació de software a més de ser la inspiració darrera d'aquest projecte.

## 2 OBJECTIUS

### 2.1 Dissenyar un entorn d'aprenentatge interactiu:

Crear una plataforma de videojoc que combini elements d'entreteniment i educació, fent l'aprenentatge de la programació més atractiu i accessible per a principiants.

### 2.2 Ensenyar conceptes bàsics de programació:

Aconseguir explicar els conceptes més fonamentals de la programació de manera clara i comprensible.

### 2.3 Crear un balanç en la experiència de joc

Trobar un balanç entre la part lúdica del joc i la part educativa, de manera que no sigui ni massa tediós i serveixi per a aprendre.

### 2.4 Desenvolupar habilitats tècniques de Unity:

Aprendre a utilitzar el motor de Unity a mesura que es va desenvolupant el projecte.

### 2.5 Garantir la usabilitat i accessibilitat:

Assegurar que el videojoc sigui fàcil d'usar i accessible per a una àmplia audiència, incloent a aquells sense experiència prèvia en programació.

## 3 METODOLOGIA

El desenvolupament del projecte ha estat dividit en diferents fases, cada fase dividida en **sprints**[8] de llargada variable i amb unes tasques específiques a realitzar (veure Figura 1 i Figura 1 de l'Apèndix per a més detalls). A continuació es detallen aquestes fases:

Anàlisi previ:

- Investigació i recopilació d'informació: Realitzar una anàlisi del concepte a tractar i estudiar existents que l'apliquen.
- Estudi tècnic: Informar-se sobre Unity per a confirmar que es tracta d'una plataforma correcta per al desenvolupament del projecte

Planificació:

- Definició de conceptes: definició dels conceptes clau de programació que el joc vol transmetre als jugadors, així com els objectius educatius que s'han de complir.
- Disseny conceptual de nivells: Petita descripció conceptual que expliqui que tractarà cada nivell.
- Planificació de continguts: Recercar els elements gràfics i de so del joc, incloent-hi els sprites dels personatges, entorns i menús.

Implementació:

- Implementació de mecàniques: Programar la lògica de les funcionalitats bàsiques del joc. Això inclou elements com el moviment del personatge o els menús.
- Desenvolupament del sistema d'entrada de codi.
- Detecció de col·lisions: Implementar la detecció de col·lisions mitjançant les **hitboxes**[9] de Unity.
- Construcció de Nivells: Construir els nivells a Unity segons el disseny conceptual planificat.
- Configuració dels puzzles: Escriure la lògica de control de cada un dels nivells.
- Disseny d'elements visuals: Adaptar o crear elements gràfics necessaris del joc.
- Integració de gràfics i so: Incorporar els elements visuals trobats i la música al joc.
- Desenvolupament de la UI: Implementar la interfície d'usuari dissenyada, es a dir, menús i altres elements del joc com el desplegable d'informació del nivell.

Test:

- Probes de jugabilitat: Enviar el joc a diversos usuaris per a rebre feedback sobre la seva experiència.
- Optimització: Aplicar canvis que es considerin apropiats a partir de la informació rebuda.

Per a gestionar aquesta planificació s’ha fet servir **Microsoft Project**[10]. Una eina que facilita el seguiment del projecte i la gestió del calendari.

Desenvolupament d'un joc didàctic amb Unity	82 días
Fase de planificació	7 días
Sprint 1	7 días
Definició de conceptes	2 días
Disseny conceptual de nivells	4 días
Planificació del contingut	1 día
Fase de desenvolupament	63 días
Sprint 2 i 3	32 días
Configuració del projecte	2 días
Desenvolupament de mecàniques	30 días
Implementar mecàniques bàsiques	14 días
Desenvolupament de sistema d'entrada de codi	14 días
Detecció de col·lisions i events	2 días
Sprint 4	17 días
Implementació de nivells	17 días
Construcció dels nivells segons disseny planificat	10 días
Configuració dels puzzles	7 días
Sprint 5	14 días
Disseny d'elements visuals	7 días
Desenvolupament de la UI	3 días
Integració de so i gràfics	4 días
Fase de test	12 días
Sprint 6	12 días
Probes de jugabilitat	5 días
Optimització	7 días

Figura 1: planificació inicial del projecte a Microsoft Project

Aproximadament a la meitat del desenvolupament es van avançar les tasques de l’sprint 5 i els sprints 3 i 4 es van allargar més de l’esperat pel que es va canviar la planificació (Figura 2).

Desenvolupament d'un joc didàctic amb Unity	82 días
Fase de planificació	8 días
Sprint 1	8 días
Definició de conceptes	2 días
Disseny conceptual de nivells	4 días
Planificació del contingut	2 días
Fase de desenvolupament	65 días
Sprint 2	20 días
Configuració del projecte	3 días
Disseny d'elements visuals	7 días
Desenvolupament de la UI	10 días
Sprint 3	28 días
Desenvolupament de mecàniques	28 días
Desenvolupament de sistema d'entrada de codi	14 días
Implementar mecàniques bàsiques	14 días
Sprint 4	14 días
Implementació de nivells	14 días
Construcció dels nivells segons disseny planificat	10 días
Configuració dels puzzles	4 días
Sprint 5	3 días
Refinament d'elements visuals	3 días
Fase de test	7 días
Sprint 6	7 días
Probes de jugabilitat	2 días
Optimització	5 días

Figura 2: planificació revisada del projecte a Microsoft Project

Tot i així al final el projecte s’ha quedat a la fase de desenvolupament, no s’han arribat a fer proves del joc amb usuaris reals.

## 4 DESENVOLUPAMENT

A continuació s’explica què s’ha desenvolupat a cada sprint amb els seus punts més rellevants.

### 4.1 Sprint 0: Documentació

Abans de començar amb el propi del desenvolupament del projecte, es va dedicar un temps exclusivament a estudiar la documentació de Unity per a familiaritzar-se amb el seu entorn. Aquesta fase inicial va ser essencial per garantir que es comptés amb una base sòlida de coneixements sobre les eines i funcionalitats que Unity ofereix, facilitant així el treball en la resta d’etapes del projecte.

Es va revisar la documentació de Unity amb l’objectiu d’entendre en profunditat els principis bàsics del motor. Aquesta lectura va incloure des dels conceptes bàsics de la interfície i l’editor fins a aspectes més complexos com el funcionament i la interacció de la programació de scripts amb C#.

Paral·lelament, es van visualitzar una sèrie de vídeos tutorialis d’iniciació de la plataforma. Aquests vídeos van ser una forma d’il·lustrar de manera més pràctica els conceptes teòrics. Els tutorialis van cobrir una àmplia gamma de temes, des de la creació i configuració de projectes fins a la implementació de mecàniques bàsiques de joc.

### 4.2 Sprint 1: Disseny i planificació dels continguts

El primer sprint va ser una setmana dedicada a planificar l’estructura del joc i la planificació dels reptes de cada nivell. En un principi s’ha pensat en 15 nivells entre els que es repartirien els diferents temes de programació a tractar però després de la reunió amb el tutor es va parlar que desenvolupar 8-10 nivells amb un tema bàsic a tractar cada un era un objectiu més realista (veure Taula 1). Encara que, com es podrà veure més endavant, es va infravalorar la complexitat de les funcionalitats base del joc.

La idea del joc es que el protagonista se li demanin una sèrie de tasques. Cada tasca correspondrà a un nivell del joc i cada nivell del joc a un concepte de programació.

Nivell	Concepte	Descripció
1	Variables Numèriques	El jugador haurà de recol·lectar recursos (nombres) que estaran pel nivell, guardar-los en variables i aplicar operacions matemàtiques per a obtenir un resultat demanat.
2	Strings	El jugador haurà de preguntar a membres del campament parts d’una contrasenya, guardar cada part com a un string, ajuntar-les i imprimir la contrasenya per a entrar a una sala.
3	Condicionals (if-else)	El jugador haurà d’utilitzar els coneixements previs per a obtenir

		un recurs i realitzar diferents accions depenent del tipus de recurs obtingut
4	Switch Case	El jugador haurà d'obrir un cofre per a obtenir un recurs (nombre) que podrà ser de 3 tipus diferents. Depenent de quin tipus s'obtingui, el jugador haurà de fer una sèrie d'accions diferents.
5	While	El jugador, mentre encara quedi tronc a la pila, haurà de partir el tronc en llenya, portar-la a una foguera i deixar-la.
6	For	El jugador haurà de repetir una acció 5 vegades utilitzant els coneixements previs.
7	Arrays	El jugador haurà de demanar l'edat a 5 membres del campament i després printar l'edat més gran i la més petita.
8	Llistas	El jugador haurà de fer una llista amb els noms de tots els membres presents al nivell i compararla amb la llista completa dels membres del campament. Haurà de printar el nom dels membres que faltin.
9	Diccionaris	El jugador haurà de crear un diccionari que guardi l'edat per a cada membre del campament (key: nom string, value: edat int)
10	Funcions	El jugador haurà de escriure el codi de la funció del moviment del jugador.

Taula 1: Llista dels nivells del joc amb la seva descripció.

Les funcionalitats que representen accions del personatge com el moviment, demanar un número o un string a un NPC[11] o obtenir la quantitat de recursos dins una caixa sera funcions que es donaran al jugador i que simularan el que als exercicis bàsics de programació serien els inputs de teclat.

### 4.3 Sprint 2: Disseny de l'estil visual i programació de menús

El segon sprint ha estat dedicat a buscar assets gràfics per a concretar l'estil visual que tindrà el joc i a desenvolupar el menú principal i la configuració d'opcions.

El menú principal consta de quatre funcionalitats (Figura 3). A dalt a la dreta tenim el botó d'opcions, que obre el menú de configuració d'opcions, el botó que canvia de mode de finestra a pantalla completa i el botó de sortir que tanca el joc i al centre trobem el botó de play que porta al menú de selecció de nivells.

Al menú d'opcions (Figura 4) hi ha dos sliders, un que controla el volum i un altre per a la brillantor del joc, i un desplegable per a escollir la resolució de la pantalla que obté les resolucions possibles del monitor de l'usuari i les llista per a escollir una.

Pel que fa al disseny visual del joc, es va optar per un

estil pixel art, una estètica que combina senzillesa i una nostàlgia personal.

S'han utilitzat **assets gratuïts**[12] disponibles al Marketplace de Unity. Això ha permès accelerar el procés de desenvolupament, proporcionant elements visuals de qualitat sense costos de temps addicionals. A més, es van extreure alguns assets del joc Mother 3 de Nintendo, aprofitant la seva riquesa visual i el seu estil distintiu que encaixava perfectament amb la visió del projecte.

S'han creat també una sèrie d'assets propis senzills per a alguns menús i botons. Aquests elements s'han dissenyat tenint en compte la coherència amb l'estètica general i per això s'han modificat lleugerament alguns dels assets existents.



Figura 3: Menú principal del joc



Figura 4: Menú d'opcions del joc

### 4.4 Sprint 3: Sistema d'introducció de codi

Degut al plantejament inicial de crear un joc de plataformes, la vista que en un principi anava a tenir el joc és la típica d'aquest estil de jocs: una vista d'scroll lateral on els nivells es desplacen d'esquerra a dreta. Aquesta perspectiva es caracteritza per oferir una experiència visual i jugable clàssica, on el jugador avança principalment en una direcció horitzontal, superant obstacles i assolint objectius a mesura que avança. Un cop es van començar a planificar i crear els nivells per provar el sistema d'introducció de codi, es va arribar a la conclusió que una visió zenital, similar a l'estil dels jocs **RPG**[13] com Pokémon, era més apropiada per al nou joc proposat. Aquesta nova perspectiva, en la qual el jugador veu el món des d'una vista

superior, proporciona un major camp de visió i una millor gestió de l'espai en el qual els jugadors poden interactuar amb els elements del joc, com ara els objectes, els personatges i els entorns.

Durant l'sprint s'ha desenvolupat el sistema amb el qual l'usuari introduirà els seus snippets de codi per a fer funcionar el joc. En aquesta tasca del desenvolupament és on s'ha trobat el principal inconvenient que ha afectat significativament a la planificació original. El problema és que el llenguatge de programació utilitzat per Unity, C#[14], no permet la compilació i execució de codi de forma dinàmica. Per tant, s'ha agut de buscar una solució alternativa externa a Unity per integrar les instruccions escrites per l'usuar temps real durant l'execució del joc.

Finalment, la solució trobada ha estat fer servir una API de sintaxi externa creada per Microsoft anomenada **Roslyn**[15]. Roslyn és una plataforma de compilació i anàlisi de codi que proporciona eines per compilar codi de C# i **Visual Basic** [16] en temps d'execució, facilitant així la creació d'un sistema dinàmic per la gestió del codi de l'usuari. El repte tècnic que ha suposat la integració de Roslyn al projecte ha resultat ser un obstacle considerable amb el que no es comptava, enrederint el desenvolupament i afectant al nombre final de nivells que s'han pogut implementar. Cal comentar que amb aquesta solució no es possible per al jugador crear les seves pròpies funcions i, per tant, el nivell 10 proposat quedava descartat.

S'ha creat un script anomenat `DynamicCodeExecutor`, el qual té el doble propòsit de gestionar l'execució del codi introduït per l'usuari i de proporcionar funcions bàsiques de moviment i interacció per al personatge del joc que l'usuari no pot solucionar amb lògica bàsica i per tant son donades per a que pugui fer la crida a accions com el moviment, la interacció amb objectes i altres personatges de l'escena, etc.

En aquest sprint s'han implementat les funcions de moviment en les quatre direccions `MoveRight()`, `MoveLeft()`, `MoveUp()` i `MoveDown()` a les quals s'ha de passar el nombre de passes a realitzar dins la quadrícula del joc.

#### 4.5 Sprint 4: Primer Nivell

En aquest quart sprint s'ha completat el nivell 1, centrat en el tractament de variables, que es va començar a l'anterior etapa, incloent la col·locació d'obstacles amb hitbox, la creació del repte i la configuració de les condicions necessàries per a superar-lo. En aquest primer nivell el jugador ha de moure al personatge fins a cadascun dels NPC (personatges no jugadors) preguntar la seva edat amb la funció `askAge()` implementada y emmagatzemar-la a una variable (Figura 5). Finalment ha de calcular la mitja d'edat dels personatges i comprovar la resposta amb la funció `respond()`.

Durant el desenvolupament, s'ha detectat un inconvenient significatiu relacionat amb l'execució de les línies de codi introduïdes pel jugador. Cada línia es processa de manera seqüencial sense esperar que la instrucció anterior es completi abans de passar a la següent. Aquest comportament provoca diversos errors, principalment en el moviment del personatge, que no respon de manera adequada

a les ordres emeses ja que d'aquesta forma els moviments es superposen i es realitzen en diagonal. Això també provoca que la resta d'accions que depenen de la ubicació del protagonista s'executin abans que arribi a la seva destinació.

Per a solucionar-ho, s'ha implementat una nova estratègia de processament del codi introduint una gestió més eficient i ordenada de les instruccions mitjançant tasques asíncrones. Les `async Tasks` permeten que cada línia de codi esperi la finalització de l'anterior abans de continuar amb la següent. Això s'aconsegueix utilitzant la paraula clau `await` davant de les crides a funcions asincròniques. Aquesta estratègia garanteix que les accions es realitzin en l'ordre correcte.

Aquest canvi implica una reestructuració de l'actual `DynamicCodeExecutor` per incloure una cua d'execució de les instruccions. En lloc de processar les ordres de manera immediata, el nou sistema posa cada instrucció en una cua d'execució i les processa una per una, assegurant que el moviment i les altres accions es compleixin sense conflictes.

S'ha utilitzat la classe `Task` de C# per gestionar les operacions asincròniques, permetent que les noves funcions com `MoveCharacterToPoint`, que ara realitza els moviments del jugador, i `QueueMove`, que s'encarrega de afegir a la cua d'execució els moviments, es comportin de manera no bloquejant, però esperant que cada moviment es completi abans de passar al següent.

S'ha creat també una nova funció per a tractar el codi escrit per l'usuari i afegir aquesta paraula clau `await` davant les funcions de moviment (`MoveRight`, `MoveLeft`, `MoveUp`, `MoveDown`) per a aconseguir aquest comportament desitjat sense exigir al jugador un coneixement tant avançat.

Pel que fa al tema de la compleció del nivell, quan es crida a la funció `respond()` passant la teva resposta, si aquesta es correcta, s'amaguen els elements d'introducció i es mostra un nou panell creat que indica que la solució donada es correcta i hi mostra un botó que et porta al següent nivell.



Figura 5: Nivell 1 del joc. A dalt a l'esquerra es veu el botó d'informació del nivell i a la dreta el panell d'introducció de codi.

#### 4.6 Sprint 5: Segon i Tercer Nivell

L'últim sprint del desenvolupament s'ha dedicat a la implementació dels nivells 2 i 3, representant les nocions de condicionals i de bucles for. Entre aquesta etapa i la anterior es van reconsiderar la planificació tenint en compte el temps restant. Inicialment es preveia la creació dels 8 nivells restants però després d'avaluar-ho, es va determinar que completar aquests dos nivells era l'objectiu més realista que es podia assolir dins del termini previst.

El segon nivell consisteix en demanar a l'NPC de color taronja un objecte. Aquest objecte es un string i pot tenir dos valors. Depenent del valor el jugador haurà de portar aquest objecte a l'NPC de color vermell o al verd, fent servir if-else.

S'han creat dos noves funcions de jugador per a aquest nou nivell, `give()` i `getItem()`.

La funció `give` verifica si la posició actual del personatge coincideix amb la casella específica d'un dels NPC. En cas afirmatiu, si l'objecte donat és l'esperat per aquest personatge, es dona per completat el nivell activant el panell de victòria. En cas negatiu o si l'objecte donat no és l'esperat s'activarà un nou panell creat en aquest sprint, el panell de derrota. Aquest mostra un botó per a tornar a intentar el nivell que quan es prem torna tots els elements a la seva posició inicial però copia el codi que l'usuari havia escrit al clipboard. Aquest panell de derrota s'ha agregat també al primer nivell.

La funció `getItem` comprova que el jugador està davant de l'NPC taronja i en cas afirmatiu retorna un dels dos possibles strings.

El tercer nivell, que tracta el bucle for, reutilitza molts elements del segon nivell ja que del que es tracta es de fer el mateix exercici però cinc vegades. Només es canvia un dels NPC amb el que s'interactua. Per tant, el jugador haurà de completar el mateix objectiu que al nivell anterior però utilitzant un for. Les funcions utilitzades son també `getItem` i `give`, que han sigut modificades per a actuar de forma diferent segons el nivell.

## 5 CONCLUSIÓ

El desenvolupament del joc ha suposat una interessant combinació d'entreteniment i aprenentatge, precisament les sensacions que es volien transmetre amb el projecte. Al llarg del projecte, han sorgit diversos desafiaments tècnics que han requerit solucions innovadores i adaptació i que han fet que no es pogués arribar al producte perfecte que s'havia planificat originalment, però sí a una bona aproximació de la que estar orgullós.

A l'objectiu d'aportar un factor educatiu i innovador, es va decidir redissenyar el projecte inicial per incloure un sistema on el jugador hagués d'escriure snippets de codi per superar els nivells, una jugabilitat més interessant i digna d'un treball de final de grau. Això ha transformat el joc en una eina didàctica que utilitza la gamificació per fer més atractiu l'aprenentatge de la programació.

A través dels diversos sprints, s'ha anat construint un joc que progressivament introdueix conceptes de programació com variables, condicionals o bucles. Encara que no

s'hagin implementat tots els nivells planificats, cada un està dissenyat per proporcionar al jugador un repte que requereix l'aplicació pràctica d'aquests conceptes, facilitant un aprenentatge interactiu i pràctic.

Definitivament, el repte tècnic més significatiu ha estat la integració del sistema d'introducció de codi dins de Unity degut a les característiques del llenguatge C#. Tot i això, la solució trobada i el temps dedicat a aquesta, ha permès superar aquesta limitació i ha estat una de les tasques més enriquidores del projecte.

Pel que fa a l'experiència de joc, es considera que s'ha aconseguit trobar un balanç entre la part lúdica i l'educativa, garantint que el joc sigui atractiu sense perdre el seu valor didàctic. El disseny visual, amb un estil pixel art, i la interfície d'usuari, han estat creats pensant en la usabilitat i l'accessibilitat per a una àmplia audiència.

Tot i que el projecte inicialment preveia la creació de 10 nivells, s'han completat amb èxit els primers 3 nivells centrats en variables, condicionals i bucles. S'ha considerat que aquests tres conceptes eren els que més representaven la programació més bàsica.

En conclusió, es considera que el projecte ha demostrat el potencial dels videojocs educatius com a eines complementàries en l'ensenyament de la programació. S'ha arribat l'objectiu d'adquisició d'habilitats tècniques amb Unity ja que utilitzant aquest motor com a plataforma, s'ha creat una experiència educativa innovadora i motivadora que pot ajudar a despertar l'interès per la programació en els més joves, així com proporcionar una base sòlida de coneixements bàsics en aquest camp. La gamificació aplicada en aquest projecte pot ser una inspiració per a futurs desenvolupaments educatius que busquin combinar aprenentatge i entreteniment de manera efectiva.

## 6 LÍNIES FUTURES

Per parlar de la continuïtat del projecte, els passos a seguir podrien consistir en implementar la resta de nivells proposats o fins i tot trobar una solució alternativa al sistema d'execució de codi per a permetre la creació de funcions i així ampliar les possibilitats del joc.

Degut al termini d'entrega no s'ha pogut dedicar el temps que un equip de desenvolupament de videojocs hi dedicaria a l'apartat artístic i d'experiència de usuari. Per millorar l'estat actual, potser es podrien afegir elements com petits tutorials a l'inici de cada nivell per mostrar un senzill exemple de el concepte que es tracta per aquells usuaris més nous o milloraria considerablement l'apartat visual amb il·lustracions, dissenys propis i animacions. També seria interessant afegir respostes visuals a les funcions que crida l'usuari per a poder seguir més clarament el que està passant a la pantalla.

## 7 AGRAÏMENTS

Voldria expressar el meu més sincer agraïment a totes les persones que han fet possible la realització d'aquest treball de final de grau.

En primer lloc, vull agrair profundament al meu tutor, Xim Cerdà Company, per la seva ajuda i disponibilitat. La seva paciència davant totes les preguntes i els problemes m'han ajudat a mantenir la calma durant el desenvolupament.

També a la meua família, gràcies per ser el meu pilar incondicional. El seu suport constant m'han ajudat a seguir endavant en els moments més baixos.

Finalment gracies també als meus amics i companys per confiar més que jo en les meves capacitats.

## BIBLIOGRAFIA

- [1] Unity és un motor de videojoc multiplataforma creat per Unity Technologies, anunciat i llançat per primera vegada al jurý de 2005 a la Conferència Mundial de Desenvolupadors d'Apple com a motor de jocs per a Mac OS X. Unity està disponible com a plataforma de desenvolupament per a Microsoft Windows, Mac OS, Linux. <https://unity.com/es> [https://es.wikipedia.org/wiki/Unity\\_\(motor\\_de\\_videojuego\)](https://es.wikipedia.org/wiki/Unity_(motor_de_videojuego))
- [2] Joc de plataformes. Els videojocs de plataformes, o simplement plataformes, són un gènere de videojocs que es caracteritzen per haver de caminar, córrer, saltar o escalar sobre una sèrie de plataformes i penya-segats, amb enemics, mentre es recullen objectes per poder completar el joc. [https://es.wikipedia.org/wiki/Videojuego\\_de\\_plataformas](https://es.wikipedia.org/wiki/Videojuego_de_plataformas)
- [3] Snippet. Un snippet, usualment traduït com a fragment de codi o simplement fragment, és un terme utilitzat en programació per referir-se a petites parts reusables de codi font, codi màquina o text. <https://es.wikipedia.org/wiki/Snippet>
- [4] Parra-González, M.E., Segura-Robles, A. (2019) Producción científica sobre gamificación en educación: un análisis cuantitativo, 113. <https://www.educacionyfp.gob.es/dam/jcr:ab346c1a-db2e-4b30-9946-c3fd31d49268/05parraesp-engl.pdf>
- [5] Bers, M. U., & Resnick, M. (2015). The official Scratch Jr book Help your kids learn to code. No Starch Press.
- [6] CodeCombat. <https://codecombat.com/>
- [7] CodinGame. <https://www.codingame.com/home>
- [8] Sprint és el nom que rep cadascun dels cicles o iteracions que hi ha dins d'un projecte.
- [9] Hitboxes. És una tècnica invisible comunament utilitzada en els videojocs per a la detecció de col·lisions en temps real [https://es.wikipedia.org/wiki/Caja\\_de\\_colisi%C3%B3n](https://es.wikipedia.org/wiki/Caja_de_colisi%C3%B3n)
- [10] Microsoft Project. és un programari d'administració de projectes i programes de projectes desenvolupat i comercialitzat per Microsoft per assistir a administradors de projectes en el desenvolupament de plans, assignació de recursos a tasques, fer seguiment al progrés, administrar pressupost i analitzar càrregues de treball.
- [11] NPC. un NPC, o PNJ, és generalment un personatge que forma part del programa, és a dir controlat per l'ordinador i no controlat per un humà [https://es.wikipedia.org/wiki/Personaje\\_no\\_jugador](https://es.wikipedia.org/wiki/Personaje_no_jugador)
- [12] Assets Gratuïts. Sunny Land. <https://assetstore.unity.com/packages/2d/characters/sunny-land-103349>
- [13] RPG. Un videojoc de rol o joc de rol per ordinador/ordinador o referit amb la sigla anglesa RPG (role-playing game) és un gènere de videojocs on el jugador controla les accions d'un personatge (o de diversos membres d'un grup) immers en algun detallat món.
- [14] C#. és un llenguatge de programació multiparadigma desenvolupat i estandaritzat per l'empresa Microsoft com a part de la plataforma .NET. La sintaxi bàsica deriva de C/C++ i utilitza el model d'objectes de la plataforma .NET, similar al de Java. [https://es.wikipedia.org/wiki/C\\_Sharp](https://es.wikipedia.org/wiki/C_Sharp)
- [15] Roslyn. .NET Compiler Platform, també coneguda pel seu nom en codi Roslyn, és un conjunt de compiladors de codi obert i API d'anàlisi de codi per a C# i Visual Basic [https://en.wikipedia.org/wiki/Roslyn\\_\(compiler\)](https://en.wikipedia.org/wiki/Roslyn_(compiler))
- [16] Visual Basic. Visual Basic (VB), originalment anomenat Visual Basic .NET (VB.NET), és un llenguatge de programació orientat a objectes i multiparadigma, implementat en .NET, Mono i .NET Framework. [https://en.wikipedia.org/wiki/Visual\\_Basic\\_\(.NET\)](https://en.wikipedia.org/wiki/Visual_Basic_(.NET))



## APÈNDIX

### A. DIAGRAMA DE GANTT

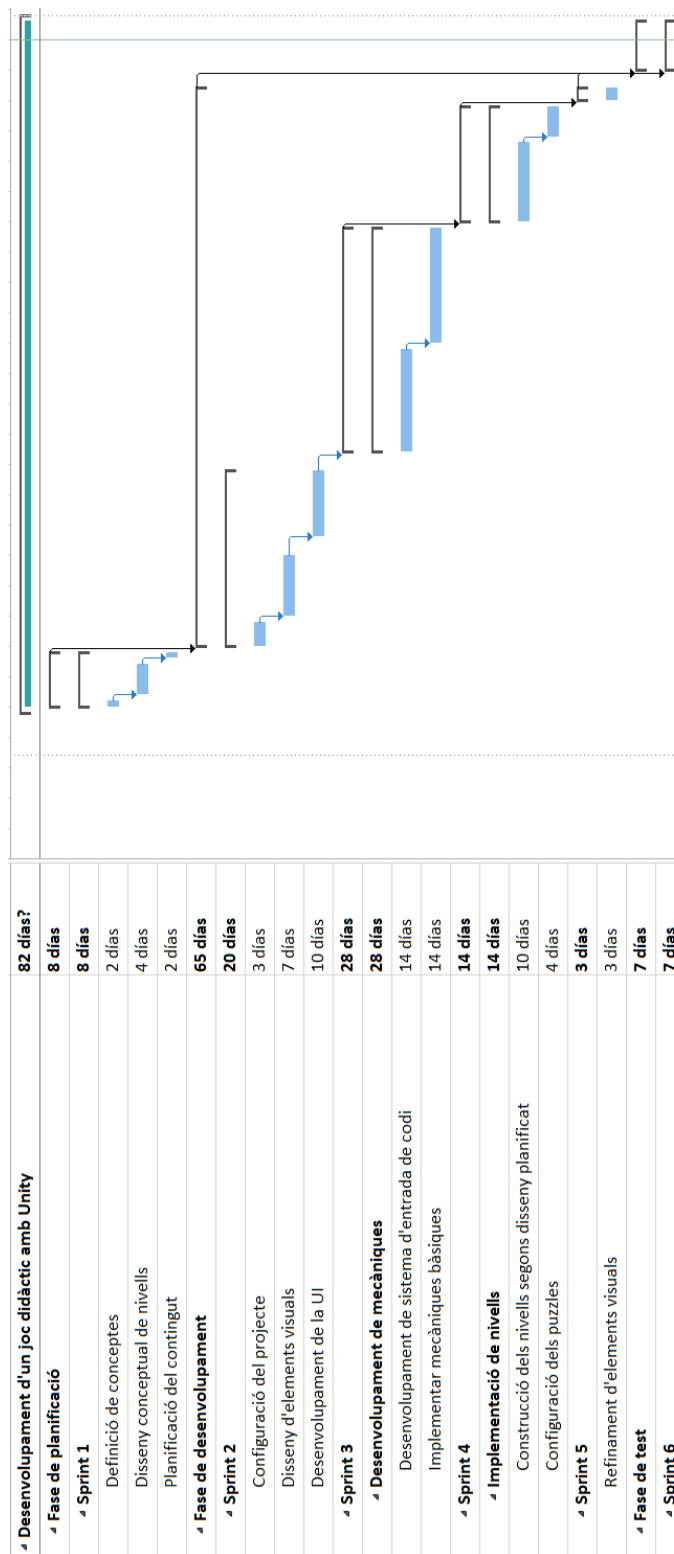


Figura 1: Diagrama de Gantt del projecte