



UNIVERSITAT DE
BARCELONA

Treball final de grau
**GRAU D'ENGINYERIA
INFORMÀTICA**

Facultat de Matemàtiques
Universitat de Barcelona

**Geopieces: Joc seriós per
l'aprenentatge de la geometria.
Part 2D-3D**

Autor: Sergi Cebrián Gres

Director: Dr. Anna Puig

Realitzat a: Departament de Matemàtiques i Informàtica

Barcelona, 30 de juny de 2016

Abstract

Learning by doing. Many studies claim that the best way of teaching is by making the students interact with what they want to learn. Accordingly, this Bachelor's Degree Final Project aims to provide teachers with an aiding tool, specifically in the field of geometry teaching during the last years of primary school. Thus, the main goal is to design and create a game using Unity with a series of geometry learning activities allowing students in primary and first years of secondary school to progress in an entertaining and engaging way, having them overcome several challenges.

Each student will have their own user and the game will be flexible enough to enable the teacher to analyze their evolution and to identify their difficulties. The teacher will also be able to customize the game, choosing which skills to train, and adding new activities and new 2D and 3D shapes among other geometric concepts.

Thus, the game will consist of a series of campaigns that the students will have to finish in order to obtain complex 3D objects from simpler three-dimensional shapes. To gain them, it will be necessary to get first some 2D shapes through a series of level-adjusted missions, chosen according to the student's acquired competences and which, in practice, will become ever more complex. Once this is accomplished, the game enters a second phase in which these flat shapes must be combined to get, through unfoldings, basic 3D objects. The performance speed and the mistakes made will be taken into account in order to achieve the campaign goals.

With this in mind, it has been necessary to split the project into two separate End of Bachelor Degree Projects, developed and coordinated by two students (D. Rausell and S. Cebrián) who worked as a team in the integration phase of the different parts of the game, and individually in the parts related to the 2D game and the unfolding game (2D-3D), respectively. The final project, therefore, has been divided in three sections:

- Common game parts (designed and developed by D. Rausell and S. Cebrián¹): game design, specification of the game campaigns and their attributes, working mechanics, design of user interaction, menu management and data persistence for users and campaigns.
- 2D game (designed and developed by D. Rausell). This section is devoted to the competences of identification of 2D shapes and their properties, such as symmetries, angles, perimeters and areas. Each player has a competence level which will increase as the game progresses and missions get accomplished, taking also into account the degree of accuracy and the completion time.
- 2D-3D game and polyhedron unfolding (designed and developed by S. Cebrián). This section deals mainly with spatial intelligence and different polyhedron nets. It also deals with the identification of 3D objects and their relationship to 2D shapes. The player must present a valid net for a given

¹ The parts of this document that both BDFP have in common have been painted blue

elemental polyhedron and is rewarded according to the completion time and the amount of mistakes made.

Resum

Aprendre fent. Són molts els estudis que afirmen que la millor manera d'ensenyar és fent que l'estudiant interactuï amb allò que vol aprendre. En aquest sentit, aquest TFG pretén donar una eina al professorat d'ajut en la docència, concretament en l'àmbit de l'aprenentatge de geometria durant el Cicle Superior de Primària. Així, l'objectiu principal és dissenyar i construir un joc amb Unity amb una sèrie d'activitats didàctiques de geometria que permetin als alumnes de primària i primers cursos de secundària aprendre la geometria d'una manera entretinguda, havent de superar diferents reptes.

Cada alumne tindrà el seu usuari i el joc serà prou flexible perquè el professor pugui analitzar l'evolució de l'estudiant i detectar en quins punts l'alumne té més dificultats. El joc serà modelable pel professor, de forma que podrà escollir les competències a treballar i afegir noves activitats, incorporant noves figures 2D i 3D a estudiar, entre altres conceptes geomètrics.

Així, el joc consistirà en un seguit de campanyes que hauran de completar els alumnes per tal d'aconseguir objectes 3D complexos a partir d'unes figures tridimensionals més simples. Per aconseguir aquests objectes, serà necessari aconseguir abans objectes 2D a través d'una sèrie de missions ajustades al nivell de competències adquirides per l'alumne i que amb la pràctica seran cada vegada més complexos. Un cop fet això, s'entrarà en una segona part on s'hauran d'unir aquestes figures planes per aconseguir, mitjançant desplegaments, figures bàsiques de tres dimensions. La velocitat d'execució i els errors cometuts seran factors a tenir molt en compte per tal d'assolir els objectius de la campanya.

Amb aquesta fita en ment, ha estat necessari separar el projecte en dos Treballs de Fi de Grau, coordinats i desenvolupats conjuntament per dos estudiants (D. Rausell i S. Cebrián), que han realitzat una part comuna d'integració de les diferents parts del joc i una part individual específica de la part del joc 2D i dels desplegaments en el pla, respectivament. Així, el projecte global s'ha dividit, en tres grans blocs:

- Parts comunes del joc (dissenyades i desenvolupades per D. Rausell i S. Cebrián²): disseny del joc, especificació de les campanyes i els seus atributs, mecàniques de funcionament, disseny de la interacció amb el jugador, gestió de menús i persistència de dades d'usuaris i campanyes.
- Joc de figures 2D (dissenyat i desenvolupat per D. Rausell). En aquesta part es treballen les competències d'identificació de figures 2D i de les seves propietats, com són la simetria, els angles, els perímetres i les àrees. Cada jugador té un nivell de competències que anirà augmentant a mesura que

² S'ha pintat de blau les parts de la memòria comunes als dos treballs

avanci el joc i es vagin complint les missions, tenint en compte també amb quin grau d'encert les completa i el temps que utilitza per resoldre-les.

- Joc 2D-3D i desplegaments de poliedres: (dissenyat i desenvolupat per S. Cebrián). En aquesta part es treballa principalment la intel·ligència espacial i els desplegaments plans de poliedres. També s'hi treballa la identificació de figures 3D i la seva relació amb les figures 2D. El jugador ha de presentar un desplegament al pla vàlid d'un poliedre elemental donat i rep recompenses segons el temps invertit i els errors comesos.

Agraïments

Vull agrair a l'Anna Puig per tota la feina i hores que ha dedicat a aquest projecte. Ha anat més enllà de les seves obligacions i no puc posar preu a la seva ajuda. Hem tingut molta sort de trobar tenir-la com a tutora.

Vull agrair al David Rausell, company de treball i amic, per haver fet tots aquests mesos de dura feina no només tolerables, sinó divertits

Vull agrair a tothom que ha tret temps durant exams per donar idees per millorar el projecte i fer proves del joc.

Vull agrair, finalment, a la meva família i amics per tolerar-me durant tot aquest dur curs, que no ha estat fàcil!

Moltes gràcies!

Índex

1	Introducció	1
1.1	Àmbit del projecte	2
1.2	Motivació	2
1.3	Objectius generals	3
1.4	Objectius específics	3
1.5	Planificació temporal	5
1.6	Estructura de la memòria	5
2	Antecedents	6
2.1	Teoria docent: Model de Van Hiele	6
2.2	Teoria de les intel·ligències múltiples	7
2.3	Aplicacions digitals i videojocs relacionats	9
3	Anàlisi i disseny del videojoc	11
3.1	Idea inicial del joc	11
3.1.1	El joc 2D-3D: Primera visió	12
3.2	Game Design Canvas	12
3.3	Setup del joc 2D-3D	13
3.3.1	Objectius d'aprenentatge en el joc 2D-3D	13
3.3.1.1	Identificació de figures 3D	13
3.3.1.2	Construcció espacial i desplegaments	14
3.3.2	Usuaris objectiu i tipus de jugadors	14
3.4	Game Design	15
3.4.1	Escenes y storyboard	15
3.4.2	Màquina d'estats del joc 2D-3D	16
3.4.3	Nivells de dificultat en el joc 2D-3D	18
3.5	Requeriments addicionals	18
4	Desenvolupament	20
4.1	Justificació de la tecnologia utilitzada	20
4.2	Arquitectura del sistema	21
4.2.1	Introducció a l'arquitectura MVC	21
4.2.2	Model	21

4.2.3	Controladors	23
4.2.4	Vista	28
4.2.5	Disseny artístic	30
4.2.6	Persistència de dades amb JSONs i servidor extern	31
4.3	Diagrama de classes	31
4.4	Classes i entitats importants	31
4.4.1	Inventory i InventoryItem	31
4.4.2	User	32
4.4.3	Campaign	33
4.4.4	NoDestroy	34
5	Resultats i simulacions	35
5.1	Planificació final	35
5.2	Resultats tècnics del joc	35
5.3	Resultats visuals del joc	36
5.4	Test del grau de reforç de les competències	37
5.5	Test de jugabilitat	39
6	Valoració econòmica	42
6.1	Anàlisi del temps de realització del projecte	42
6.2	Valoració econòmica	42
7	Conclusions	44
7.1	Treball futur	45
A	Manual tècnic	
A.1	Instal·lació: requeriments mínims i passos a seguir	
B	Manual d'usuari	
B.1	Manual del professor	
B.2	Manual de l'estudiant: Aprèn a jugar	
C	Test de competències	

Índex de figures

1	Planificació diagrama de Gantt den S. Cebrián	5
2	Van Hiele theory	7
3	Theory of Multiple Intelligences	8
4	GeoPieces 2D-3D: Game Design Canvas	13
5	Exemples de desplegaments d'un prisma triangular	14
6	Relacions entre escenes.	15
7	Diagrama de la màquina d'estats del joc 2D-3D	17
8	Exemple del funcionament del bloqueig d'arestes en el cas del cub. En les posicions verdes el jugador pot col·locar les cares de desplegament. Els bloquejos vénen representats per les posicions vermelles.	19
9	Exemple d'objectius adicionals	23
10	Controladors del joc 2D-3D	24
11	Exemple de solapament de previews	25
12	Exemple d'un desplegament que únicament causa col·lisions a arestes	28
13	Visió del HUD, sense el panell de tutorial, amb càmera perspectiva	29
14	Diagrama de classes	32
15	Instància classe Inventory	33
16	Planning real diagrama de Gantt vs previsió (S. Cebrián)	35
17	Pantalla de selecció de campanya. Els dos edificis inferiors de color taronja són clicable, mentre que el jugador ha completat la campanya de la font, que ja no és clicable i apareix al poblat amb textura i animació.	36
18	Exemple menú de campanya d'una campanya iniciada.	36
19	Exemples visuals d'errors i èxits als plegaments dels joc 2D-3D	37
20	Test de usabilitat: preguntes genèriques GeoPieces	40
21	Test de usabilitat: preguntes sobre el joc 2D-3D	41
22	Pantalla de configuració prèvia a GeoPieces.	
23	Pantalla de sign up i log in.	
24	Menú principal.	
25	Select Campaign.	
26	Menú de campanya	
27	Món 2D	
28	Món desplegaments	
29	Pregunta 1 Test de Competències	

30	Pregunta 2 Test de Competències	
31	Pregunta 3 Test de Competències	
32	Pregunta 4 Test de Competències	
33	Pregunta 5 Test de Competències	
34	Pregunta 6 Test de Competències	

Índex de taules

1	Test de competències abans de jugar a GeoPieces	38
2	Test de competències després de jugar a GeoPieces	38
3	Temps usat en el projecte (en hores i setmanes) per S. Cebrián	42
4	Cost per tasca i hores de S Cebrián	43
5	Pressupost del projecte	43

1 Introducció

L'educació és la base perquè un país funcioni. No és estrany veure a Espanya al final de les llistes de la Unió Europea en abandonament escolar i qualitat educativa. Segons l'estudi de la UNESCO "Educación para todos" de l'any 2012 [1], un de cada tres joves van deixar els estudis a Espanya abans d'acabar l'educació secundària enfront de l'1 de cada 5 de la mitjana europea. Tot i que als darrers anys ha anat disminuint aquest percentatge, segons l'Oficina d'Estadística de la Unió Europea (Eurostat), l'any 2015 l'abandonament escolar a Espanya encara se situava a gairebé el doble de la mitjana europea (21,9% estatal front l'11,1% europeu) convertint-se en el líder de la Unió Europea en aquest aspecte.

Per altra banda, l'informe PISA de 2012 [2] situava l'Estat Espanyol a la cua dels 34 països de l'Organització per a la Cooperació i Desenvolupament Econòmic (OCDE) en el que respecta a matemàtiques, lectura i ciència, significativament per sota de la mitjana de l'OCDE. A més, l'informe destaca una important dificultat en l'assimilació de conceptes relacionats amb la geometria, què és el tema que ens ocupa. Si s'analitzen les 4 subàrees avaluades ('quantitat', 'incertesa i dades', 'canvi i relacions' i 'espai i forma'), els resultats són clarament inferiors en el que respecta a les dues últimes. La persistència d'aquests resultats en el transcurs dels anys, deixa patent la importància estratègica de millora en l'ensenyança de la geometria (especialment als blocs curriculars que fan referència a espai, forma i mesura). Aquests resultats fan referència a competències de 4t d'ESO, fet que evidencia la necessitat de millorar l'educació en primària i principis de secundària, moment en el qual s'imparteixen els fonaments bàsics per després comprendre conceptes més avançats.

Tot i que les raons d'aquest fracàs escolar són moltes i diverses, una de les causes és aquest sistema rígid on el professor és l'agent actiu, parlant i escrivint a la pissarra, mentre que l'alumne es converteix en agent passiu, escoltant i com a molt, prenent apunts. Tot i que pot ser relativament assequible detectar l'origen del problema, és complicat trobar una solució transversal i, a més, fàcilment aplicable en l'àmbit econòmic i logístic.

Diferents ànalisis proposen incorporar a l'aula activitats geomètriques competencialment més riques, experimentals i constructives associades, a ser possible, a àmbits de la vida quotidiana. Generalment, aquestes activitats segueixen les següents etapes d'aprenentatge:

1. Experimentació: comparar, mesurar, assajar, sentir, percebre.
2. Descobriment: de relacions i patrons singulars.
3. Conceptualització: integració de l'experiència en primera persona o col·lectivament.
4. Abstracció: de la propietat deduïda.

En aquest projecte ens centrarem en les etapes d'experimentació, descobriment i abstracció, deixant la conceptualització per ser treballada per l'alumne amb l'ajuda del professor.

Estudis recents evidencien un increment de la motivació i rendiment si s'utilitzen experiències virtuals a l'aula. La virtualització obre la porta a nombroses possibilitats per l'aprenentatge en les etapes d'experimentació i descobriment permetent, per exemple, realitzar simulacions, treballar amb laboratoris虚拟s, personalitzar l'experiència segons el perfil de l'estudiant o gamificar l'aprenentatge fomentant la col·laboració, la competència entre equips o usant videojocs.

1.1 Àmbit del projecte

Aquest projecte sorgeix de la necessitat a les aules de treballar amb més profunditat activitats experimentals i/o de manipulació per a reforçar aprenentatge de la geometria, a través dels anomenats jocs seriosos. Els jocs seriosos -o serious games- són aquells jocs que tenen una finalitat afegida al simple fet de divertir-se o entretenir-se. El creixement d'aquests jocs seriosos com a eines d'aprenentatge són una realitat i el seu àmbit d'aplicació en el món de l'educació és immens.

En concret, el projecte consisteix a crear un joc per a ordinador i/o tablet perquè els alumnes de Cicle Superior de Primària i Cicle Inicial de Secundària exercitin i millorin les seves competències en geometria. Així doncs, es treballaran les competències d'identificació de figures, angles, simetries, perímetres i àrees, a més d'exercitar la seva capacitat d'abstracció per construir els desplegables de figures 3D.

Per la realització d'aquest videojoc s'utilitzarà el motor de joc Unity [3] i es programarà amb el llenguatge orientat a objectes C#. Tot i que el desenvolupament d'aquest videojoc implica la utilització de conceptes que s'han après a moltes assignatures de les cursades durant el grau en Enginyeria Informàtica, les més útils per la realització del projecte són les que citem a continuació:

- Programació II: Iniciació a la programació orientada a objectes i a l'arquitectura de software MVC (Model, Vista, Controlador).
- Factors humans i computació: Usabilitat i experiència d'usuari. Ús de storyboards.
- Software distribuït: Sistema de comunicació amb servidors a través de JSONs.
- Gràfics i visualització de dades: Objectes i transformacions geomètriques en mons 3D, càmeres i shaders.
- Enginyeria del Software: Gestió de projectes grans i ús de Unity com a motor de videojoc i Blender per modelar objectes 3D.

1.2 Motivació

Les possibilitats que donen les evolucions tecnològiques en forma de tablets o ordinadors i la baixada dels preus d'aquests aparells, cada vegada més accessibles a més

població, han permès als videojocs situar-se com una alternativa real als sistemes d'aprenentatge tradicionals integrats en les aules.

Aquest projecte pretén unir les noves possibilitats tecnològiques amb la necessitat d'un canvi en el model educatiu. El projecte també aspira que el software donat no sigui necessàriament utilitzat en hores lectives, desdibuixant així la idea d'una separació tan rígida entre món escolar i món familiar. A l'escola no només s'aprèn i a casa no només s'educa. Trencar amb aquesta dualitat no és només necessari, sinó cada vegada més indispensable.

La motivació principal, doncs, és poder ajudar a potenciar aquest tipus d'aprenentatge, utilitzant algoritmes perquè el joc s'adapti al màxim al nivell de l'alumne en qüestió. En aquest aspecte, és fonamental donar al professor una eina metodològica fàcil d'utilitzar, modificar i a la vegada proporcionar una experiència divertida, usable i motivadora pels estudiants perquè l'aprenentatge de geometria sigui molt més vivencial i enriquidor.

1.3 Objectius generals

L'objectiu principal del projecte és dissenyar i desenvolupar un videojoc que s'adapti al nivell i la progressió de l'alumne per ajudar-lo a aprendre competències relacionades amb l'àmbit de la geometria. Així mateix, es pretén dissenyar un joc prou flexible i modular per tal que sigui una eina modificable pel professor d'una manera intuïtiva i ràpida de forma que pugui definir nous reptes per als alumnes, analitzar la progressió dels estudiants i reforçar conceptes geomètrics a vegades difícils d'assimilar.

1.4 Objectius específics

L'objectiu general descrit a la secció anterior es pot descompondre en diferents objectius específics. Aquests objectius específics es detallen a continuació:

1. Disseny de joc seriós escalable, divertit, motivador, addictiu i visualment atractiu:
 - (a) Disseny del storyboard del joc. Disseny pantalla a pantalla de les escenes principals del joc, explicant el flux del mateix i com interactuen unes escenes amb les altres.
 - (b) Disseny dels escenaris. Disseny d'interfícies fàcils de navegar
 - (c) Disseny dels components del joc. Models de figures 2D i 3D, textures i sprites.
 - (d) Disseny de les mecàniques i dinàmiques del joc. Disseny del funcionament de cadascuna de les escenes.
 - (e) Disseny de la interacció del joc amb el jugador. Disseny d'interfícies fàcilment navegables amb pantalla tàctil o ratolí.

- (f) Disseny de la monitoratge del joc. Disseny de sistemes de monitoratge de les accions dels usuaris.
2. Disseny i desenvolupament del joc 2D
- Disseny d'itineraris adaptatius segons les competències adquirides pel jugador. El joc s'ha de poder adaptar al nivell acadèmic de l'alumne.
 - Disseny d'itineraris adaptatius segons les directrius definides a la campanya. El joc s'ha d'adaptar a la campanya en curs.
 - Desenvolupament del joc 2D. Desenvolupament de la dinàmica del joc 2D a partir del disseny.
3. Disseny i implementació del joc 2D-3D de desplegaments de figures en el pla
- Disseny d'interfície ajustable a l'objectiu actual de la campanya.
 - Disseny i desenvolupament d'un sistema d'animació dels plegaments per ajudar a la visualització de resultats
 - Desenvolupament del joc de desplegament. Desenvolupament de la dinàmica del joc a partir del disseny.
4. Disseny integrat en un joc de construcció d'objectes 3D: Joc consistent en construir figures 3D complexes a partir dels poliedres elementals aconseguits al joc 2D-3D.
5. Disseny i desenvolupament de la tecnologia que permeti la creació de noves campanyes:
- Disseny i desenvolupament de campanyes que permetin escollir les competències a reforçar per l'alumne
 - Disseny i desenvolupament de campanyes creades per defecte
 - Disseny i desenvolupament de sistemes per afegir noves campanyes personalitzades pel professor
6. Desenvolupament d'un sistema de connexió amb servidor extern per:
- Gestió d'usuaris. Donar un software que permeti gestionar els usuaris creats i guardar els seus avenços.
 - Gestió de campanyes. Donar un software que permeti gestionar les campanyes creades i les noves campanyes.

Aquests objectius s'han dividit de la següent manera entre els dos alumnes encarregats de dur a terme aquest projecte en dos TFGs separats:

- Objectius comuns: apartats 1, 4, 5 i 6.
- Objectius D. Rausell: apartat 2.
- Objectius S. Cebrián: apartat 3.

1.5 Planificació temporal

Per a la planificació temporal s'ha utilitzat un diagrama de Gantt (figura 1). Com el temps estimat de projecte és de 20 setmanes i en el diagrama es mostren fins a 40 unitats de temps, cada unitat de temps representa, doncs, mitja setmana o unes 7.5 hores. Els temps d'inici del projecte va ser mitjans de febrer tot i que abans ja es van tenir les primeres reunions per enfocar el Treball de Fi de Grau.

S'ha separat la planificació en 5 parts: formació, disseny/anàlisi, desenvolupament, testing i documentació.

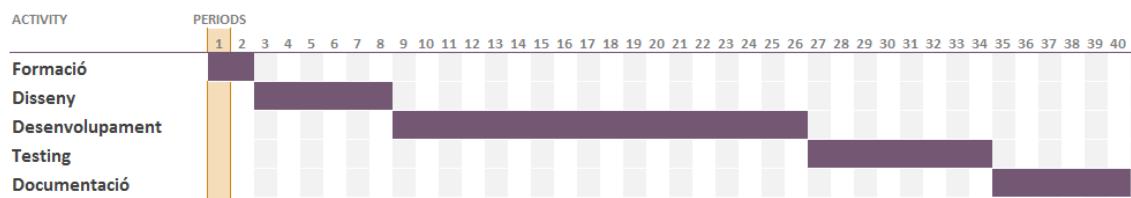


Figura 1: Planificació diagrama de Gantt den S. Cebrián

1.6 Estructura de la memòria

La memòria d'aquest treball de fi de grau està estructurada de la manera següent. Al capítol 2 es detallaran els treballs relacionats al projecte dut a terme, fent especial èmfasi a la teoria docent que hi ha al darrere i fent referència a la teoria d'intel·ligències múltiples proposada per H. Gardner [6]. Al capítol 3 es fa una petita introducció al joc i s'utilitza el Game Design Canvas per realitzar el disseny del joc. Al capítol 4 es justifica la tecnologia i es detalla l'arquitectura del joc, els components del mateix i algunes de les classes més importants. Al capítol 5 es parla dels resultats del joc, tant en l'àmbit visual i tècnic, com a partir d'un test de competències i d'usabilitat fet a diferents persones. Al capítol 6 s'explica l'abast econòmic del projecte, tenint en compte que ha estat fet per dos dissenyadors. Finalment, al capítol 7 es detallen les conclusions i es dóna peu a possibles treballs futurs relacionats.

En els apèndixs A, B i C es descriuen els manuals tècnics, manuals d'usuari i el test de competències, respectivament.

Cal destacar que s'han marcat en color blau els títols de les seccions i els textos comuns amb el projecte que s'ha desenvolupat en paral·lel per obtenir el joc final [4].

2 Antecedents

2.1 Teoria docent: Model de Van Hiele

El model de Raonament Matemàtic de Van Hiele és una teoria d'ensenyament i aprenentatge de la geometria, dissenyat pel matrimoni holandès Van Hiele [5], amb origen l'any 1957. La idea bàsica d'aquest model és el fet que l'aprenentatge de la geometria es construeix passant per nivells de pensament, que es resumeixen a continuació de forma breu i es detallen en un posterior esquema (figura 2):

- Nivell 0: Visualització. Els estudiants perceben figures geomètriques globalment, com objectes físics. Les descripcions d'aquests objectes es basen en característiques visuals o tàctils. No es reconeixen explícitament propietats matemàtiques. És el nivell típic d'alumnes d'educació infantil i primers cursos de primària. Aquest nivell fa referència a la competència d'identificació de figures, com veurem més endavant.
- Nivell 1: Anàlisis. Els alumnes perceben les figures geomètriques com conjunts de propietats i elements matemàtics, sense fer encara relacions lògiques entre ells. Poden realitzar raonaments empírics i generalitzar aquestes propietats. És típic en alumnes de finals de Primària o principis de Secundària. Aquest nivell seria en el que es focalitza aquest projecte.
- Nivell 2. Deducions informals. Els estudiants entenen les relacions de dependència lògica entre propietats matemàtiques i comencen a tenir capacitat de raonament abstracte, tot i que encara de forma informal. Típic dels últims cursos de Secundària obligatòria i, en menor mesura, de Batxillerats humanístics.
- Nivell 3. Deducions formals. Els estudiants són capaços de realitzar raonaments lògics i deductius en contexts de matemàtica formal. Típic en alumnes de Batxillerat Científics i principis de cursos universitaris.
- Nivell 4. Rigor. Els alumnes són capaços de raonar correctament en diferents sistemes axiomàtics. Propi de matemàtics avançats.

La idea d'aquest model és que per arribar al nivell n s'ha d'haver passat i superat el nivell $n - 1$ i el fet que el que és complicat en un nivell, és gairebé explícit en el següent. Per avançar d'un nivell al següent, el model explicita que a cada nivell es pot procedir de la següent forma:

- Fase 1: Informació. Davant un nou tema, el professor planteja activitats per fer una presa de contacte entre estudiant i tema tractat.
- Fase 2: Orientació dirigida. Els estudiants comencen a explorar el tema mitjançant activitats tancades que els guiaran en l'aprenentatge dels continguts bàsics del tema. El professor ha de comprovar que els estudiants arriben als resultats previstos.

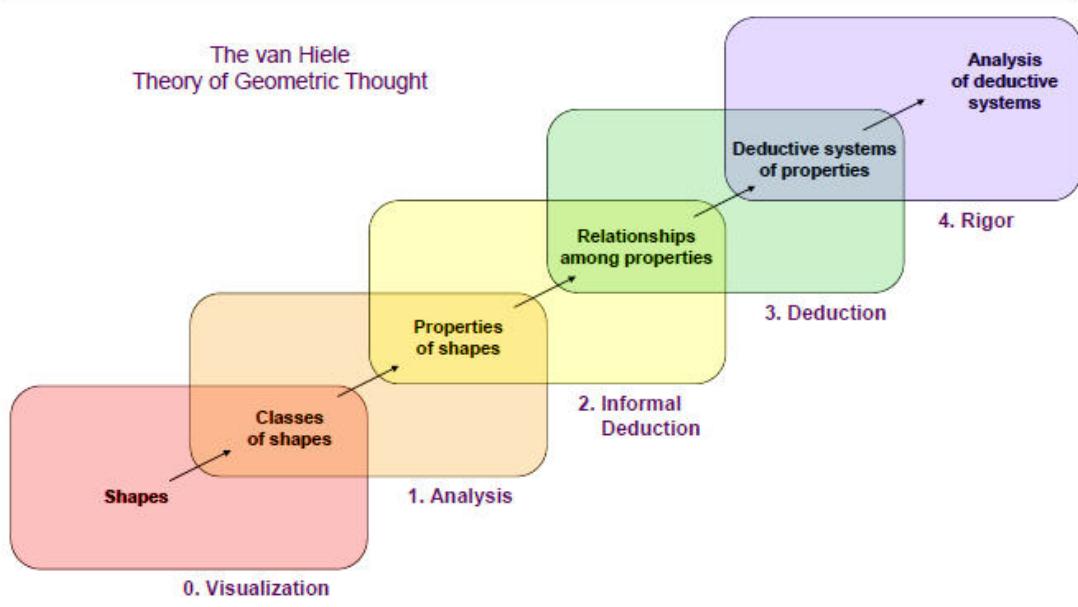


Figura 2: Van Hiele theory

- Fase 3: Explicació. S'expliquen els resultats obtinguts i les operacions realitzades, es justifiquen les afirmacions realitzades, es debaten amb el professor o altres estudiants.
- Fase 4: Orientació lliure. Els estudiants resolen activitats on han de posar en joc els nous coneixements i formes de raonament en situacions més creatives i obertes.
- Fase 5. Integració. La finalitat d'aquesta fase no és aprendre nous conceptes, sinó la integració de tot el que han après.

Aquest projecte pretén estar dins del model de Hiele. En aquest sentit, s'ha posat èmfasi en les fases més pràctiques d'aquest model (sobretot en les fases 1, 2, tot i que també en la fase 4 i parcialment en la fase 5).

2.2 Teoria de les intel·ligències múltiples

La teoria de les intel·ligències múltiples és un model de la intel·ligència proposat per primera vegada per Howard Gardner al seu *Frames of Mind: The Theory of Multiple Intelligences* [6]. En aquest llibre, Gardner articula diferents criteris per jutjar quins comportaments poden ser considerats "intel·ligències", com ara la seva rellevància a la història evolutiva o potencial d'afectació per lesions cerebrals. Gardner acaba definint les següents 8 (figura 3):

- Intel·ligència musical: Composició, interpretació, valoració de música i sons...
- Intel·ligència lingüística: Lectura, escriptura, parla...

- Intel·ligència lògica-matemàtica: Resolució de problemes, cerca de relacions i causalitats entre fenòmens, predicció...
- Intel·ligència espacial: Visió tridimensional del món, pensament visual...
- Intel·ligència corporal-cinestèsica: ús del cos per activitats o solució de problemes, sensacions somàtiques (pressió, dolor, etc.)...
- Intel·ligència interpersonal: Comprendre en vers a altres persones, capacitat d'entendre'ls i treballar amb ells...
- Intel·ligència intrapersonal: Comprendre en vers a un mateix, anàlisi de la situació pròpia, de forces i debilitats...
- Intel·ligència naturalista: Classificació, distinció i utilització d'elements naturals.



Figura 3: Theory of Multiple Intelligences

Tot i la rebuda força crítica d'aquesta teoria, deguda principalment a la manca de proves empíriques; és un model que serveix, des de la valoració subjectiva, per millorar l'ensenyament a alumnes. Segons Gardner, el sistema d'educació vigent

es centra, majoritàriament, en les intel·ligències lingüística i lògica-matemàtica, ignorant quasi per complet la resta.

Aquest projecte vol tractar la intel·ligència espacial, en el joc 2D identificant diferents figures i, principalment al joc 2D-3D, on l'alumne ha de visualitzar el plegament i desplegament de diferents figures al seu cap.

2.3 Aplicacions digitals i videojocs relacionats

En aquest apartat, es veuen diferents aplicacions i videojocs amb relació a aquest projecte. S'han classificat segons la plataforma on es poden jugar o fer servir:

- Aplicacions per mòbils i tablets:
 - DragonBox Algebra/DragonBox2 [7]: Joc mòbil per l'aprenentatge d'àlgebra. Hi ha una versió per a usuaris de 5+ anys i una per usuaris més grans que 12. Joc de pagament. Enfocat al joc individual.
 - DragonBox Elemens [8]: Joc mòbil per l'aprenentatge de propietats de figures geomètriques amb el mateix esperit que DragonBox Algebra. Joc de pagament. Enfocat al joc individual.
 - MathToons Geometry Pro [9]: Aplicació de tests de matemàtiques. Gratuïta. Ús individual.
- Java Applets en línia:
 - Java-Applets zur Mathematik [10]: Una col·lecció d'aplicacions java, en alemany, en línia per tots els temes, des de l'aritmètica fins a la trigonometria. Ús per classes o individual.
 - WisWeb applets [11]: Col·lecció d'applets de java i html5 de *Freudenthal Instituut* per l'aprenentatge de matemàtiques. Molt centrat en geometria.
- Aplicacions Flash en línia:
 - Math Playground [12]: Una col·lecció d'una gran quantitat de jocs Flash per a nens per a moltes branques de matemàtiques. Joc individual i gratuït.

També, com a aplicació d'ordinador per l'aprenentatge de matemàtiques, s'ha de destacar GeoGebra [13]. Aquesta es fa servir en molts instituts i consisteix, bàsicament, en un processador geomètric i algebraic.

En general, de les aplicacions digitals vistes, les utilitzades a l'aula solen estar enfocades directament a activitats d'aprenentatge sense ser pròpiament jocs. Les que s'han llistat que sí que són jocs, però, no són fàcilment traslladables a l'aula per manca d'adaptabilitat a les competències de cada alumne en cada moment ni disposen d'eines de seguiment pel professor. En aquest projecte es pretén cobrir

aquestes mancances amb el disseny d'un joc seriós que ja internament tingui en compte les competències, pugui adaptar-se a diferents nivells de dificultat i que sigui prou flexible per a suportar possibles ampliacions d'activitats d'aprenentatge.

3 Anàlisi i disseny del videojoc

3.1 Idea inicial del joc

Com ja s'ha explicat, l'objectiu principal del joc és el d'aprendre geometria d'una manera adaptativa i divertida. Es parteix de la base que cada nen o nena té el seu procés d'aprenentatge i per tant, el joc s'ha d'adaptar a la progressió de l'alumne. El joc es desenvolupa amb la idea de ser usat en plataformes tàctils, com tablets. Tot i així, també podrà ser utilitzats amb un cursor sigui via ratolí o via touchpad. L'idioma de tot el joc és l'anglès.

El joc portarà a l'alumne a un univers 3D, on es demanarà reconstruir un poblat amb diferents monuments i estructures. Per tant, serà necessari reconstruir un o diversos objectes per cadascun dels monuments. Per aconseguir-los, però, serà necessari viatjar a un món 2D i completar allà un conjunt de reptes i missions per aconseguir figures planes. Una vegada fet això, el pas al 3D tampoc serà immediat. Noves missions apareixeran al camí dels jugadors, que hauran d'aconseguir ajuntar de forma correcta les figures 2D per tal de crear objectes tridimensionals.

Amb aquesta idea en ment, i no perdent de vista mai què és un videojoc d'aprenentatge, existeixen dos modes de joc:

- Mode individual o autònom: on hi han unes campañas predefinidas que l'alumne pot triar i realitzar en un entorn acadèmic o domèstic. Els seus progrésos són guardats i les seves competències millorades en conseqüència.
- Mode classe: El professor pot crear rooms amb un codi d'accés. Els alumnes, a classe, poden connectar-se a aquestes sales, on el professor ha creat prèviament una sèrie de campañas a realitzar.

En tots dos casos, es pretén tenir en compte el progrés de l'alumne. Així, el professor podrà utilitzar aquest videojoc per posar reptes geomètrics als alumnes. Aquests reptes seran tant d'identificar de relacions i característiques de figures 2D com de treball de la capacitat d'abstracció mitjançant desplegaments de figures 3D. La idea genèrica és que un estudiant que realitza les missions ràpidament i sense errors o que ha practicat a casa, ha de resoldre cada vegada panells més complexos que el que té un procés d'aprenentatge més lent o té menys pràctica.

Tot i que per motius logístics no s'ha completat el segon mode de joc amb la connexió a un servidor on el professor pot configurar la seva classe ("mode classe"), ja que requeria un altre treball final de grau [23] que s'estava desenvolupant en paral·lel, sí que hem seguit la idea de joc adaptat al nivell d'aprenentatge de l'alumne i d'una programació escalable per poder, en un futur treball, realitzar la connexió amb el servidor..

Per donar-li un sentit als reptes que es van proposant a l'alumne, el videojoc té com a objectiu completar campañas. Envers el concepte "Campanya" es mou tot el joc. La idea és que una campanya tingui una figura objectiu tridimensional (el

monument), composta de figures 3D simples (cubs, tetraedres, etc). Per a aconseguir completar una campanya amb èxit és necessari aconseguir totes les figures 2D i 3D necessàries per, al ajuntar-les, aconseguir la figura tridimensional objectiu de la campanya. Així, si una campanya té com a figura objectiu una casa simple (un cub amb una piràmide a sobre), serà necessari aconseguir 7 quadrats (els 6 del cub i 1 de la piràmide) i 4 triangles (de la piràmide) i després aconseguir els desplegaments per aconseguir el cub i la piràmide.

3.1.1 El joc 2D-3D: Primera visió

Per a aconseguir figures tridimensionals, l'usuari haurà de tenir al seu inventari les cares que la componen i seleccionar la figura que es vol obtenir al menú de la campanya. Un cop dins del joc 2D-3D, l'usuari haurà de presentar un desplegament vàlid de la figura objectiu, enganxant les cares aresta amb aresta, podent tenir certes restriccions en la col·locació de les cares per evitar o dificultar la repetició de formes planes ja fetes. Tot seguit, es veurà una simulació animada del plegament, i s'hi farà la detecció d'errors. Aquests errors es veuran en la mateixa simulació de forma que el jugador podrà conèixer on s'ha equivocat i podrà modificar la seva proposta de desplegament vàlid.

3.2 Game Design Canvas

Dissenyar jocs entretinguts, divertits i que a més suposin un repte pel jugador és força complicat. I més quan s'ha d'intentar que el joc vagi més enllà d'ofrir pur entreteniment, com és el cas. És per això que el disseny del joc és una tasca crucial en el desenvolupament del mateix. Per aquest motiu, d'ara endavant i fins al final del capítol es parlarà del disseny del joc ajudant-nos de l'anomenat *Game Design Canvas* (GDC) [16].

El *Game Design Canvas* és un conjunt d'eines i processos que fan el disseny d'un joc més ràpid i més previsible. El resultat és un Canvas estructurat fàcil de compartir, entendre i modificar durant el procés de desenvolupament.

El GDC està dividit en tres seccions:

- Setup: Defineix la intenció del joc i el tipus de jugadors a qui va dirigit.
- Game Design: Defineix les interaccions, mecàniques i feedback de l'experiència.
- Consideracions addicionals: Defineix consideracions que tindran impacte sobre el disseny, desenvolupament i marketing del joc.

A continuació es mostra el *Game Design Canvas* del joc 2D-3D (figura 4). En les següents seccions es desenvolupa cadascuna de les tres parts del GDC. Així, la secció 3.3 parla sobre el setup del joc, 3.4 sobre el disseny del joc i, a l'apartat 3.5, es parlarà sobre els requeriments addicionals pel joc.

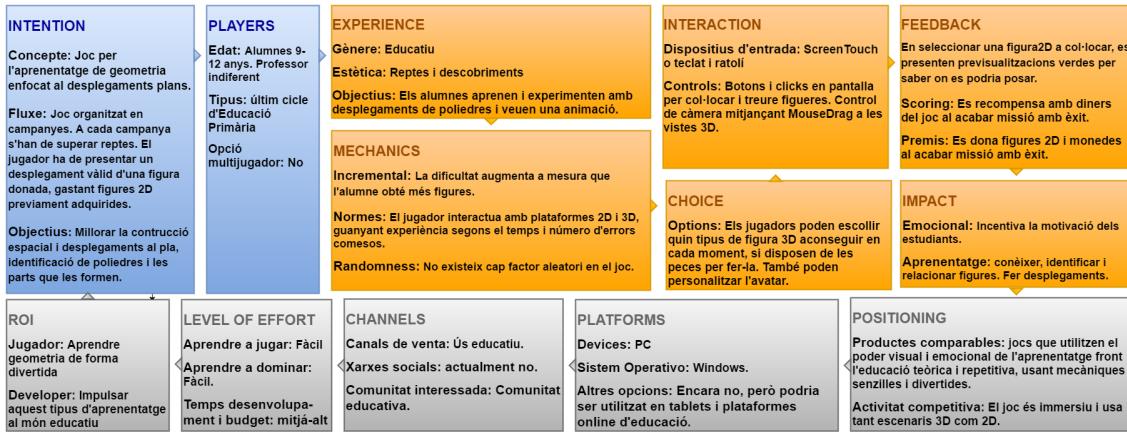


Figura 4: GeoPieces 2D-3D: Game Design Canvas

3.3 Setup del joc 2D-3D

En aquest apartat, es veurà amb més detall els apartats del GDC corresponents a *Intention* i *players*. Del primer, s'estudiarà amb detall els objectius del joc i, de la part de *players*, s'exploraran els tipus d'usuaris.

3.3.1 Objectius d'aprenentatge en el joc 2D-3D

El joc 2D-3D té com a objectius d'aprenentatge:

- La identificació de figures 3D i les figueres 2D que les formen, la relació entre elles i les seves propietats.
- La construcció espacial de poliedres elementals a partir de les seves cares i l'exploració dels seus desplegaments en el pla.

3.3.1.1 Identificació de figures 3D

Aquest objectiu es correspon al nivell 0 del Model de Van Hiele. Els alumnes hi han de reconèixer les figures 3D o poliedres i relacionar-les amb les cares que les formen. Les figures 3D han de ser elementals; és a dir, no han de poder-se subdividir en altres figures 3D de costats 1. En aquest joc, les cares seran sempre polígons regulars: triangles, quadrats i pentàgons.

Donada la immensa quantitat de figures elementals generables, el joc disposarà inicialment de les següents figures:

- Cub/Hexaedre: Figura de 6 costats formada per 6 quadrats amb un angle diedre de 90° entre ells.
- Tetraedre: Figura de 4 costats formada per 4 triangles amb un angle diedre de $\arctan(2\sqrt{2}) \approx 70.529^\circ$ entre ells.

- Piràmide de base quadrada: Figura de 5 costats formada per 1 quadrat i 4 triangles amb un angle diedre de $\arccos(-\frac{1}{3}) \approx 109.471^\circ$ entre els triangles i un angle diedre de $\frac{1}{2} \arccos(-\frac{1}{3}) \approx 54.736^\circ$ entre els triangles i la base.
- Octaedre: Figura de 8 costats formada per 8 triangles amb un angle diedre de $\arccos(-\frac{1}{3}) \approx 109.471^\circ$ entre ells.
- Prisma triangular: Figura de 5 costats formada per 3 quadrat i 2 triangles amb un angle diedre de 60° entre els quadrats i un angle de 90° entre els triangles i els quadrats. Els dos triangles no tenenarestes en comú i, per tant, no formen angle entre ells.
- Prisma pentagonal: Figura de 7 costats formada per 5 quadrat i 2 pentàgons amb un angle diedre de 108° entre els quadrats i un angle diedre de 90° entre els pentàgons i els quadrats. Els dos pentàgons no tenenarestes en comú i, per tant, no formen angle entre ells.

Cal destacar que tot i que s'han definit aquestes figures 3D base, se'n podrien afegir fàcilment de noves mitjançant una serialització JSON.

3.3.1.2 Contrucció espacial i desplegaments

Aquest objectiu es correspon directament amb el concepte d'intel·ligència espacial i la intel·ligència lògico-matemàtica de la *Theory of multiple intelligences* de Howard Gardner, ja que es treballen tan conceptes d'espai i forma com de prediccions i cerques de relacions entre cares en el desplegament d'una figura 3D.

S'entén per desplegament pla d'un poliedre un dibuix format per polígons units perarestes que, en plegar-lo, esdevenen les cares de la figura. Qualsevol desplegament que formi la figura objectiu el considerarem vàlid.



(a) Exemple de desplegament vàlid



(b) Exemple de desplegament invàlid

Figura 5: Exemples de desplegaments d'un prisma triangular

3.3.2 Usuaris objectiu i tipus de jugadors

Aquest joc està pensat perquè hi juguin alumnes de Cicle Superior de Primària. Tot i així, la idea és que serveixi d'eina també als professors. Així, hi ha dos tipus d'usuaris:

- Alumnes: Juguen amb el joc per millorar les seves competències geomètriques.

- Professors: Preparen el joc per ajudar als alumnes a aprendre geometria. Escullen competències a aprendre i sistema adaptatiu al nen o no, com veurem més endavant.

3.4 Game Design

En aquesta secció es veuran les diferents escenes i se'n presentarà un storyboard, es veurà la màquina d'estats que segueix el joc 2D-3D amb un diagrama, i s'estudiaran els diferents nivells de dificultat i els factors dels quals depenen.

3.4.1 Escenes y storyboard

A continuació (figura 6) es mostra el flux del joc complet que estarà compartit tant per la part 2D i la part 2D-3D. És important remarcar que menys el "Log In - Menú Principal", totes les altres escenes donen l'opció també d'anar enrere. El flux indicat en la figura és bidireccional, és a dir, el jugador sempre pot anar cap enrere en les accions triades.



Figura 6: Relacions entre escenes.

Explicació de les escenes:

- Log In. És la primera escena del joc. Dóna la possibilitat d'accedir al joc en el cas que el jugador ja tingui un usuari creat i també dóna la possibilitat de crear un nou usuari. Tot i que es posa per donar realisme al joc, no es fa gestió de contrasenyes.
- Menú principal. És la primera escena un cop el jugador ha entrat en el joc. Des d'aquí es pot anar a tres llocs diferents mitjançant els seus respectius botons: selecció de campanyes, estadístiques i selector d'avatar. Hem deixat com a botó no clicable l'opció de jugar en el "mode classe".
- Selecció de campanyes. Aquí el jugador arriba a un escenari de models 3D representant els objectius de cadascuna de les campanyes. Mentre la campanya no hagi estat completada, fent clic sobre el model 3D s'accedirà al menú de campanya corresponent.

- Estadístiques: Aquí es poden veure totes les estadístiques de l'usuari on es detallen els diners acumulats, informació detallada per competències (experiència, nivell, partides jugades, partides completades amb èxit i partides no iniciades després de veure objectiu) i la relació entre campanyes iniciades i campanyes acabades.
- Selecció d'avatar: És una escena poc rellevant en la dinàmica del joc, però és un component de joc apreciat pels tipus de jugadors a qui va adreçat. És extra i només pretén donar més incentius al jugador, podent comprar avatars diferents i equipar de forma personalitzada a l'avatar del jugador, seleccionant un avatar d'entre els que s'han comprat.
- Menú de campanya: En aquesta pantalla es poden iniciar missions per aconseguir figures 2D (Joc 2D) o missions per, a partir de les figures 2D aconseguides, construir figures 3D simples necessàries per completar el monument objectiu, o campanya. A més d'aquestes dues funcionalitats, també permet comprar figures sempre que es tingui els diners necessaris en un format de botiga. L'última opció que permet és accedir a la informació corresponent a la campanya en curs.
- Joc 2D: és el lloc on s'accedeix al polsar sobre un objecte 2D en la pantalla de menú de campanya. Aquí el jugador haurà de seleccionar unes figures objectiu sobre una graella 2D amb un temps determinat. Si ho aconsegueix, tindrà un premi (major o menor segons com de bé l'hagi realitzat) tant en l'àmbit de monedes com d'experiència. Aquest joc és el que ocupa la part central del treball de fi de grau del D. Rausell [4].
- Joc 2D-3D: és al lloc on s'accedeix en polsar sobre un objecte 3D en la pantalla de menú de campanya. Aquí el jugador haurà d'aconseguir fer el desplegament d'una figura 3D. Si ho aconsegueix, tindrà un premi econòmic i la figura 3D en qüestió.
- Informació de la campanya: En aquesta pantalla es podrà consultar tota la informació de la campanya en qüestió. Veient el nom de la campanya, els percentatges d'importància de cada competència en el cas que la campanya sigui per pesos o les competències que entren en joc, en el cas que els pesos no influeixin. També es pot veure el percentatge de missió completada fins al moment, que no és més que el nombre de figures aconseguides (ja siguin 2D o 3D) dividit entre el nombre total de figures que s'han d'aconseguir en acabar la campanya.

3.4.2 Màquina d'estats del joc 2D-3D

El joc 2D-3D disposa d'una màquina d'estats amb els següents estats (figura 7):

1. Inici: Estat inicial. L'usuari veu el centre de la pantalla buida i disposa de totes les peces necessàries per posar. Salt directe a l'estat Esperant.

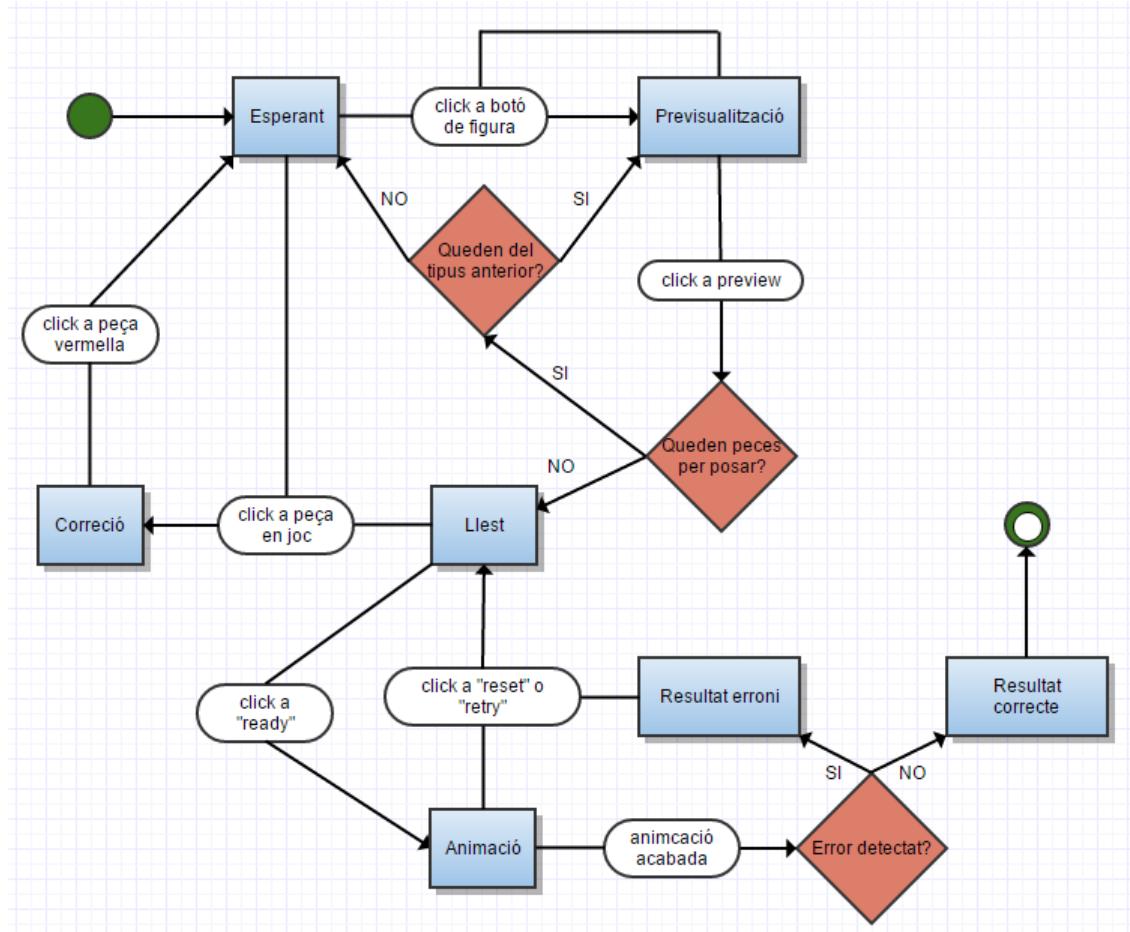


Figura 7: Diagrama de la màquina d'estats del joc 2D-3D

2. Previsualització: En seleccionar una figura 2D (si en té a l'inventari) es presenta una previsualització de totes les possibles posicions d'aquella peça, controlant superposicions internes. Si es fa clic en una altra tipus de peça i queden peces d'aquell tipus en l'inventari, es torna a aquest mateix estat; en cas contrari, el sistema s'espera per una acció de l'usuari.
3. Esperant: Estat sense previsualitzacions on encara queden figures per posar.
4. Llest: Estat sense previsualitzacions on ja no queden figures per posar. S'activa el botó de Ready. En prémer el botó Ready es passa a l'estat d'Animació/Prova.
5. Correcció: En fer clic a una cara del dibuix, aquesta s'il·lumina de color vermell i pot eliminar-se tornant-hi a clicar, junt amb tota la seva branca (vegeu 4.2.2) i es torna a l'estat Esperant. Fent clic a qualsevol altre lloc, retorna totes les figures al color original i torna a l'estat anterior.
6. Animació/Prova: En prémer el botó de Ready des de l'estat Llest, comença l'animació de doblegament que només pot parar-se mitjançant el botó de

Reset, que torna a Llest; o esperant al final de l'animació, que porta a un Resultat.

7. Resultat Erroni: Si hi ha hagut alguna col·lisió durant l'animació, es pinten les cares involucrades de color vermell per detecció d'errors i s'ha de clicar a Reset per tornar a Llest i corregir.
8. Resultat Correcte: Si no s'han detectat col·lisions i ha acabat l'animació, es mostra la pantalla d'enhorabona amb els corresponents objectius o estrelles aconseguides.

Aquests són doncs els estats interns i les seves transicions del joc 2D-3D. A més de tots ells, en qualsevol moment l'usuari pot marxar al menú de campanya. Aquest estat no es contempla a la màquina d'estats, ja que és extern al mateix joc, però representa una última transició possible.

3.4.3 Nivells de dificultat en el joc 2D-3D

El nivell de dificultat del joc 2D-3D és determinat per 4 factors principals:

- El primer factor és el mateix disseny de la mecànica de construcció del desplegament. Donat que únicament es poden col·locar noves peces a cares prèviament col·locades pel jugador, es limiten les accions possibles i facilita així la presentació del desplegament.
- El següent és la complexitat del poliedre en qüestió. Per exemple, donats 4 triangles, només existeixen 3 desplegaments possibles llevat de simetries i rotacions, i únicament un d'ells és un desplegament invàlid d'un tetraedre.
- El tercer factor ve donat pels objectius addicionals de no cometre errors i completar el dibuix en el menor temps possible. Recompensar a l'usuari per tardar menys d'una certa quantitat de temps (calculada automàticament a partir de la quantitat de cares) facilita que cometin errors.
- L'últim factor a tenir en compte ha estat dissenyat per forçar a l'usuari a canviar el tipus de desplegament un cop ja ha aconseguit la mateixa peça 3D. Amb aquest objectiu en ment, s'ha decidit bloquejar certa quantitat d'arestes a la figura inicial i no permetre així a l'usuari col·locar peces a segons quin costat per obligar-lo a canviar l'estructura (figura 8).

3.5 Requeriments addicionals

A continuació es mostren els requeriments tecnològics i addicionals al joc (part grisa de la figura 4):

- Plataforma. El joc es podrà jugar finalment al PC, tot i també estar previst per ser utilitzat en plataformes tablet.



(a) 1er cub, cap ares- (b) 2n cub, 1 aresta (c) 3er cub, 2 arestas
ta bloquejada bloquejada bloquejades bloquejades

Figura 8: Exemple del funcionament del bloqueig d'arestes en el cas del cub. En les posicions verdes el jugador pot col·locar les cares de desplegament. Els bloquejos vénen representats per les posicions vermelles.

- Sistema operatiu. El sistema operatiu on es fa el testeig del joc és Windows. El joc funcionarà sempre que la versió de Windows estigui entre Windows 7 fins a windows 10. En plataformes tablet, s'utilitzarà exclusivament Android.
- Comunitat interessada. La comunitat més interessada en aquesta aplicació és l'educativa.
- Dificultat per aprendre a jugar. La dificultat d'aprenentatge del joc és mitja-baixa.
- Temps de desenvolupament previst. El temps de desenvolupament previst és d'uns 4-5 mesos a mitja jornada.
- Resolució de pantalla. Com una de les plataformes objectiu és tablet, s'ha optat per una resolució 16:10, la més comuna entre elles.

4 Desenvolupament

4.1 Justificació de la tecnologia utilitzada

Un cop decidit que es vol desenvolupar un joc d'ordinador s'ha d'escol·lir un motor, i donades les característiques d'aquest projecte, ha de ser un motor capaç de tractar 3D i gratuït. Això deixa les següents opcions principals:

- Unity3D: Unity és un motor de videojocs multiplataforma desenvolupat per Unity Technologies. La versió Unity personal és gratuïta. Permet desenvolupament per a Windows, OS X, Linux, Xbox 360, PlayStation 3, Playstation Vita, Wii, Wii U, iPad, iPhone, Android i Windows Phone. [3]
- CryEngine: CryEngine és un motor de videojocs creat per l'empresa CryTek per a ús propi. La cinquena i última iteració, però, va canviar el model de negoci a *Pay what you want*. Permet desenvolupament per a Windows, Linux, Xbox360/One, Wii, WiiU i PlayStation3/4. [14]
- Unreal Engine 4: UE4 és la quarta iteració del motor de videojocs creat per Epic Games i la primera amb llicència gratuïta. Permet desenvolupament per Windows, Linux, OS X, Xbox One, PlayStation 4, HTML5, iOS i Android. [15]

Tots ells tenen avantatges i desavantatges, però, finalment es va optar per Unity3D peles següents motius:

- Experiència prèvia. Els programadors ja havien fet servir aquest entorn a l'assignatura d'Enginyeria del Software de tercer any d'Enginyeria Informàtica. Com a desenvolupador del projecte, la meva experiència en 3 gamejams prèvies i el desenvolupament de diferents projectes en Unity anteriors, avalaven fortement aquesta decisió.
- Integració *Cross-Platform*. Unity3D permet portar un projecte a diferents plataformes amb molt baixa dificultat.
- Facilitat de canvi de mecàniques 2D a 3D. Donat que el projecte involucra parts purament 2D i d'altres a mitges, la flexibilitat de Unity ha estat un factor important a l'hora de decidir.
- El tractament de geometria vectorial que ofereix Unity3D és idoni pels requeriments del Joc és idoni pels requeriments del Joc 2D-3D.
- Documentació. Unity3D disposa d'un web amb exemples de totes les classes i funcions que han creat, i com es programa amb c#, es pot trobar documentació externa al MSDN de Microsoft. A més disposa d'una comunitat molt activa en fòrums amb una col·lecció enorme de dubtes ja resolts.

Tot i les dificultats que presenta el treball en equip per Unity per les tasques d'integració, al ser un equip de només dues persones, es va optar per la creació de dos projectes en paral·lel i per una quedada setmanal per integrar-los.

4.2 Arquitectura del sistema

4.2.1 Introducció a l'arquitectura MVC

Per aquest projecte hem seguit el patró d'arquitectura Model Vista Controlador (MVC), freqüentment utilitzat en sistemes on es requereix una interfície d'usuari. Aquest patró sorgeix de la necessitat de crear software més robust, potenciant la facilitat de manteniment, reutilització del codi i separació de conceptes.

Així, es fonamenta en la separació del codi en tres capes:

- **Model:** Capa de dades. Té mecanismes per accedir a informació externa i actualitzar els seus valors. És el cas de les classes de campanya, usuaris, missions, etc.
- **Vista:** Capa visual. Conté el codi que ens permet veure la visualització de les capes d'usuari. En són exemples les classes que controlen la HUD.
- **Controladors:** És el cervell. Gestiona la comunicació entre la vista i el model de dades. En són exemples el controlador del joc 2D i 3D, el controlador de les missions, etc.

Es desenvolupen a continuació cadascuna de les parts.

4.2.2 Model

Objectes 3D

Es considera un desplegament en el pla d'un poliedre com a un graf planar connex dirigit sense bucles $G(V,E)$ on V són els nodes (cares) i E arestes (costats de contacte entre cares). Per a realitzar el plegament, cal guardar la següent informació als elements de E (ie, les arestes):

- Dos *Figura2Desplegament*: els controladors dels elements de V que uneix aquesta aresta. Com és un graf dirigit, guardem cap a quin node assenyala l'aresta i quin és el node original.
- Un Vector3D unitari assenyalant, inicialment, la direcció del centre de la cara al punt mitjà de l'aresta. Aquest és necessari per a l'instanciació de les previ-visualitzacions i per l'animació del plegament que es comentarà en profunditat a l'apartat 4.2.3
- Un booleà que marca si l'aresta està bloquejada, és a dir, si no es deixen col·locar peces per aquest costat. 3.4.3.

Pel que fa als nodes, tenen uns controladors per gestionar l'animació de plegament i guardar les seves propietats que s'explicaran amb detall a l'apartat 4.2.3

Missió 2D-3D

Una missió del joc 2D-3D té per objectiu, com ja s'ha detallat prèviament, presentar un desplegament pla vàlid d'un poliedre elemental objectiu. Per generar la missió i fer la comprovació posterior del resultat, doncs, cal tenir en compte tres característiques:

1. El nombre de peces de cada tipus que conformen el poliedre en qüestió.
2. Les normes de rotació de les cares per realitzar l'animació del plegament i la detecció d'errors (4.2.3).
3. El nombre de figures iguals a la figura objectiu que ja s'han realitzat per determinar el nombre d'arestes que es bloquegen per augmentar la dificultat de forma autonòmica 3.4.3.

Els apartats 1 i 2 es guarden a un objecte de la classe *Objective* i el nombre de vegades que l'usuari ha superat ja aquesta missió s'obté directament del seu inventari.

La classe Objective és alhora la que guarda l'estat actual de la missió un cop es carrega. Les normes de plegament s'hi guarden en un diccionari el següent format:

- Claus: Strings de 3 caràcters de la Forma "X-X", marcant X el tipus de cara a cada costat de l'aresta (S = Square, T = Triangle i P = Pentagon).
- Valors: Els angles que han de ser girades les cares internament per a obtenir la figura 3D objectiu.

Una missió té associats uns objectius addicionals en forma d'estrelles (figura 9) que s'obtenen en acabar. La primera estrella correspon a la completació de la missió, mentre que la segona i la tercera involucren el temps utilitzat i els errors cometuts respectivament. El temps per obtenir la 2a estrella és directament proporcional al nombre de cares del poliedre i el calcula automàticament. El nombre d'errors permesos per obtenir l'última estrella, ara mateix, és 0 independentment de la figura: no es permet fer errors sense penalització.

Com a premi per completar la missió correctament, el jugador rep la figura 3D i una quantitat de monedes multiplicada pel nombre d'estrelles aconseguides.

Botiga

Amb les monedes aconseguides en les missions 2D o 2D-3D es pot:

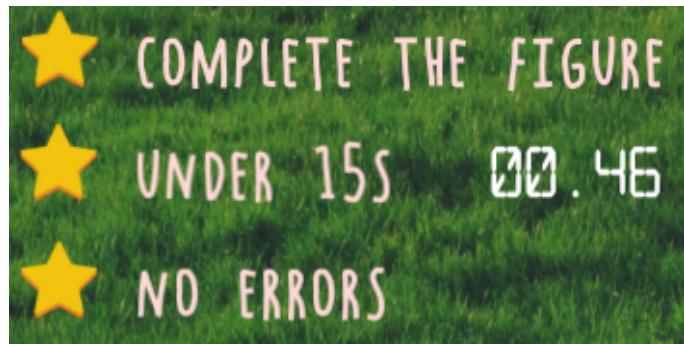


Figura 9: Exemple d'objectius adicionals

- Comprar figures objectiu de la campanya (2D i 3D). Com millor ho faci l'alumne en les missions, més diners tindrà i abans podrà acabar la missió. El preu que s'estableix per cada ítem (figura 2D o 3D) depèn només del nombre d'ítems a aconseguir. Com més ítems s'hagin d'aconseguir, més econòmic serà comprar-ne un.
- Comprar avatars. És una opció auxiliar, que va a part de l'aprenentatge de la geometria. És una forma d'incentivar els alumnes per aconseguir més monedes. Així, amb els diners obtinguts, es podrà comprar i equipar una sèrie d'avatars 2D que trobarem a l'escena “Selecció d'avatar” (vegeu secció 3.4.1).

4.2.3 Controladors

El controlador principal del joc és l'objecte `GameController` i, més en concret, el seu script `DesdoblamientoSceneController`. Aquest controla tots els inputs i accions del jugador i prepara l'escena segons la peça objectiu, que lleixa del `NoDestroy` 4.4.4. Delega el control d'animació i control d'errors als scripts dels nodes del graf, `Figura2Desdoblamiento`, i la gestió de la HUD i interfície gràfica al `UnfoldingHUD`.

L'estat actual de la missió és gestionat per aquest controlador principal, que segueix la màquina d'estats dissenyada a l'apartat 3.4.2. El següent diagrama (figura 10) ensenya la interacció i responsabilitats dels diferents controladors:

Guiaje de la col·locació de cares 2D en el pla

En aquest apartat s'expliquen les previsualitzacions o previews de les figures mig transparents que s'ensenyen per a guiar a l'usuari en la col·locació de peces, i la visualització de peces en referència al `GameObject` final de la cara.

Per facilitar la col·locació de cares al desplegament, en seleccionar l'usuari quina cara vol, s'instancien previsualitzacions a tots els costats lliures del graf. Cadascun d'aquests objectes guarda la informació següent:

- L'aresta de la qual ve. Guardant-la aquí és immediatament atribuïble a la figura definitiva que la substitueixi.

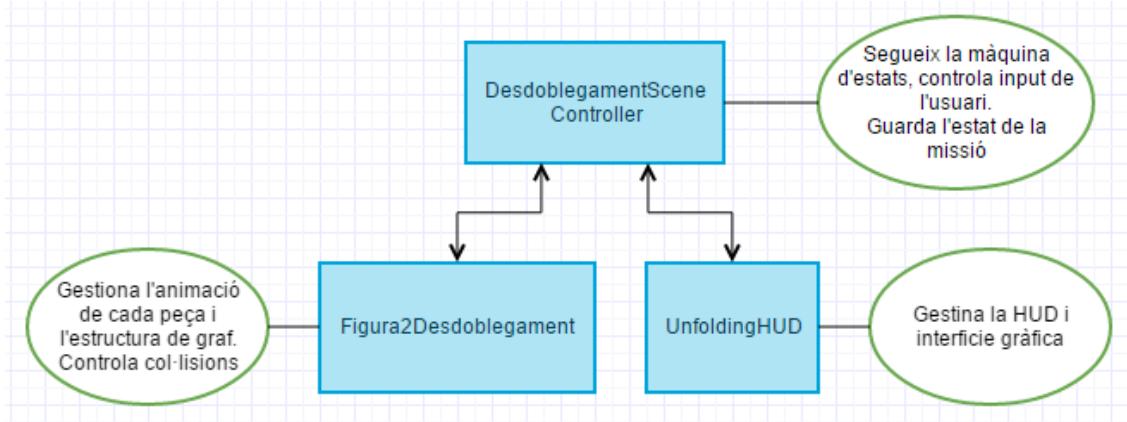


Figura 10: Controladors del joc 2D-3D

- El *GameObject* a instanciar en el cas que se seleccione aquest preview. Aquest és atribuït directament al Prefab, o objecte predefinit, aprofitant la potència de l'editor de Unity3D.
- El nombre de costats de la figura en qüestió. Aquest és necessari per al càlcul de l'apotema i la instanciació de les seves arestes amb el vector corresponent.
- L'apotema de la figura (ie. la longitud del segment ortogonal a un costat fins al centre). Aquest pot calcular-se mitjançant la fórmula:

$$Apotema = \frac{L}{2 \times \tan\left(\frac{\pi}{costats}\right)}$$

on L és la longitud del costat; en el nostre cas, sempre 1.

- Un indicador que controla si és un preview associat a una aresta bloquejada.

La primera peça només pot col·locar-se al centre de la pantalla i ve d'una aresta nul·la. Per escollir on instanciar la resta de previsualitzacions es recorre el graf des de la *RootPiece*. Per cada peça i per a cada aresta que es troba amb destí lliure s'instancia el prefab de preview de la peça a col·locar. Pel que fa a la posició i rotació es calculen de la següent manera:

- Posició: Siguin P_f = posició del preview, P_0 = posició de la peça original, v = vector de l'aresta (unitari), A_0 = apótema de la peça original, A_f = apotema del preview. Aleshores:

$$P_f = P_0 + v * (A_0 + A_f)$$

- Rotació: S'instancia inicialment sense rotació i es fa un SET del seu *transform.forward* (vector local que assenyala en front d'un *GameObject*) i se'l fa igual al vector de l'aresta.

Tots aquests passos es realitzen a la funció *RecursiveTreePreview* del *DesdoblamientoSceneController*, que rep de paràmetres l'arrel del subarbre (inicialment la *RootPiece*) i el *GameObject* del preview a instanciar.

Un cop instanciades les previsualitzacions es fa detecció de col·lisions:

- Si un preview troba que està en superposició amb alguna peça final, és eliminat immediatament, ja que assenyala una posició on no s'hi podria posar peces.
- Si, per altra banda, detecta superposició amb un altre preview, únicament són eliminats si hi ha superposició exacte (amb molt petit marge d'error). Aquest pas és necessari per evitar que es pogués tancar un bucle amb, per exemple, 4 quadrats (figura 11). Un bucle d'aquest tipus causaria una figura no plegable.

Per acabar, si l'aresta estava marcada com a bloquejada, el preview que s'instancia és de color vermell i no és interactuable.

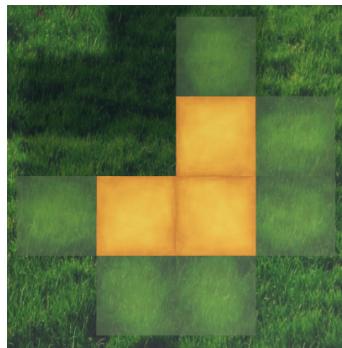


Figura 11: Exemple de solapament de previews

Si el jugador fa aleshores clic a un preview, aquest es destrueix i s'instancia en el seu lloc la peça final amb igual posició i rotació. En instanciar una peça, s'atribueix com a peça destí de l'aresta de la qual sortia el preview i s'instancien tantes arestes com els seus costats menys 1 i amb vectors $v_i = \text{transform.forward}$ girat $i * 360/costats$ graus en l'eix vertical.

Modificacions sobre del desplegament actual

En qualsevol moment durant la missió, l'usuari pot voler canviar la distribució de les cares que ha posat. Tenint en compte l'estructura de graf i que aquest ha de ser connex, si s'elimina una peça, ha d'eliminar-se tot el seu subarbre també. Com això pot causar esborrar cares que no es pretenia o bé simple confusió, es demana confirmació. Amb el primer clic a una cara, es pinta de vermell junt amb tot el seu subarbre. Si es fa clic en una figura en vermell, s'elimina junt amb tot el seu subarbre. Fer un clic a qualsevol altre lloc, torna les peces al seu estat inicial.

Aquestes accions les fan les funcions *readyToDelete()* i *deleteRecursive()* respectivament.

Control automàtic del plegament

Per a la presentació i control del resultat final, s'han estudiat tres possibilitats:

- Tenir guardats tots els desplegaments possibles, llevat de simetries i rotacions, de cada poliedre elemental i mostrar únicament la figura resultant si no hi ha errors.
 - Pros: Impossibilitat d'error.
 - Contres: No hi ha animació i dificulta el feedback d'errors. No hi ha manera immediata de comparar el graf amb els desplegaments possibles.
- Tenir guardats tots els desplegaments possibles, llevat de simetries i rotacions, de cada poliedre elemental i les animacions corresponents a cadascun d'ells.
 - Pros: Impossibilitat d'error. El jugador veu la seva feina animada.
 - Contres: No hi ha manera immediata de comparar el graf amb els desplegaments possibles. Hi ha massa animacions per guardar (en el cas del cub, ja hi ha 11 desplegaments possibles).
- Animar directament seguint unes normes de plegament i detectar errors mitjançant col·lisions.
 - Pros: Fàcilment ampliable: només cal donar les normes. En cas d'error pot veure's exactament on ha succeït.
 - Contres: Possibilitat d'error en el resultat.

De bon principi se sabia que es volia fer una animació del plegament per ensenyar de manera visual com s'ajuntarien les peces. Així, es va descartar ràpidament la primera opció. L'elegància de la tercera opció i la possibilitat d'ensenyar dinàmicament on falla el desplegament presentat pel jugador són el que va decantar la balança. Calia, però, fer bona gestió dels errors i les maneres de detectar errors s'explica amb profunditat a l'apartat 4.2.3.

Tenint aquest sistema d'animació en ment, un cop col·locades totes les cares disponibles, el controlador activa el botó *Ready*. En fer-hi clic el jugador, passen les següents coses:

1. Es para el cronòmetre del temps.
2. S'assegura el controlador que totes les peces són del color que toca, és a dir, no n'hi ha de vermelles.
3. Es preparen les variables de control del plegament.
4. La *MainCamera*, que inicialment era ortogràfica i amb vista aèria, passa a perspectiva i se situa en cert angle, tot mirant la *RootPiece* mitjançant la funció *transform.LookAt(Vector3 position)*

5. S'activa el control de la càmera per part de l'usuari arrossegant el ratolí amb clic esquerre.
6. Es dóna l'ordre al controlador de la *RootPiece* de *foldRecursive(obj)*, tot passant-li l'objectiu actual per saber les normes de plegament.

Quan es crida la funció *foldRecursive(obj)* d'una peça del desplegament, aquesta recorre totes les seves arestes i, si estan ocupades, crida a cadascuna de les peces adjacents la funció *foldSubTree()*, passant-los l'aresta que les ajunta i l'angle que han de girar segons les normes.

La funció *foldSubTree* comença la rotació de la figura i tot el seu subarbre al voltant del punt mitjà de l'aresta, prenent per eix el vector de l'aresta girat 90° a la dreta. D'aquesta manera, assegura que la rotació és sempre interna. Alhora, prepara a l'arrel del subarbre perquè, en acabar de rotar, executi el seu propi *foldRecursive(obj)*

La rotació en si de cada figura es gestiona al *Update()* del seu controlador. Cada frame rota una mica i, quan arriba a l'angle final de rotació, mira si s'han de realitzar més accions.

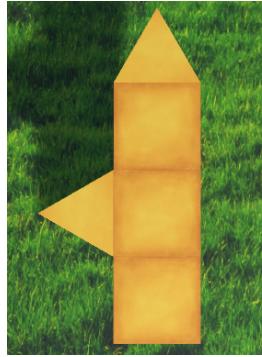
Anàlisi automàtica de la correctesa del doblegament

Per la detecció d'errors d'un desplegament pla donat d'un poliedre, va optar-se per buscar col·lisions entre les cares durant la rotació. És a dir, si en qualsevol moment durant la rotació, la cara detecta una col·lisió amb una altra cara, ambdues es pinten vermelles i s'envia un missatge de "error trobat" al controlador principal, que deté la resta de rotacions i prepara la pantalla de resultat (vegeu 4.2.4). Aquest enfoc, però, té un problema prou evident: Donat que les cares són objectes 3D (tot i que molt primis) i comparteixen arestes entre elles, es poden detectar col·lisions amb les veïnes que haurien de ser valides.

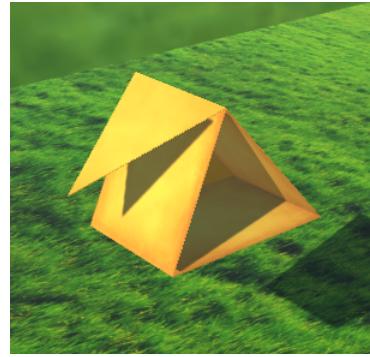
Per resoldre aquest problema, es va canviar el grandària del *Collider* de les figures per ocupar lleugerament menys i ignorar així col·lisions d'aresta amb aresta. En concret, s'ha reduït la grossor en una èpsilon del 5%. No obstant, això causa un nou problema: Pot haver-hi desplegaments erronis que només causin col·lisions a arestes (exemple: figura 12)

Aquest problema ha pogut solucionar-se augmentant únicament la grossor dels *colliders*, de manera que si dues cares que comparteixen aresta però formen un angle dihedral molt petit, es detectarà col·lisió. Ara, però, desplegaments vàlids de poliedres molt baixos com, per exemple, la piràmide pentagonal, es detectaran falsament, com invàlids.

Per solucionar aquest últim problema, trobat durant el testing (5.5), s'ha optat per augmentar la grossor dels colliders únicament per la part inferior. Això funciona, ja que les figures s'instancien sempre amb la mateixa cara amunt i les rotacions sempre es realitzen cap al centre del poliedre. Així s'han solucionat els falsos positius del tipus de la figura 12 i els falsos negatius dels poliedres baixos.



(a) Desplegament pla



(b) Plegament

Figura 12: Exemple d'un desplegament que únicament causa col·lisions a arestes

4.2.4 Vista

HUD i interfícies

El script *UnfoldingHUD* és el controlador de la interfície gràfica del joc 2D-3D. Aquest gestiona l'activació dels diferents botons i panells que es mostren seguit ordres delegades del *SceneController*. Per assegurar l'adaptació a tot tipus de pantalla, al component de CanvasScaler del hud s'hi ha seleccionat l'opció "Scale with screen size".

Aquests són els diferents panells i els seus components (figura 13):

- ”Tutorial Panel”: Aquest panell s’activa únicament la primera vegada que un usuari juga el joc 2D-3D. Aquest té un text explicant el comportament i els controls del joc, i un botó per tancar el panell i començar a jugar.
- ”Top Panel”: Aquest és el panell que conté els elements de la part superior següents:
 - El títol de la figura per recordar l’objectiu.
 - El panell d’objectiu principal. Invariant.
 - El panell de l’objectiu addicional 1: S’hi carrega el temps assignat a l’objectiu, l’estat de l’estrella i el timer.
 - El panell de l’objectiu addicional 2: S’hi carrega l’estat de l’estrella i el nombre d’errors que es pot realitzar.
- ”Bottom Panel”: En aquest panell hi són tots els botons de l’escena i controla l’estat de cadascun d’ells.
- ”Middle Panel”: Aquest panell presenta el resultat del plegament un cop acabat. Fins aleshores està inactiu. Els resultats possibles són:
 - Èxit en el plegament: Es posa el text Success en verd. S’ensenyen les estrelles aconseguides i s’activen els botons de retorn a la campanya i, si encara té peces, el de ”Get More”.

- Error en el plegament: Es posa el text Error en vermell. S’activa el botó de retorn a la campanya i el botó de Reset per tornar a editar el desplegament.

Càmera

Durant el joc 2D-3D, la càmera principal té dos estats:

- Ortogràfica fixa. Durant la presentació del desplegament al pla del poliedre, la càmera és fixa i mira el desplegament en vertical.
- Perspectiva mòbil. Durant l’animació i la vista del resultat, la càmera passa a ser perspectiva i rotatable al voltant de la figura per poder apreciar millor tant el plegament com el resultat. Està centrada durant l’animació i lleugerament desplaçada quan apareix el panell de resultat, per fer-li espai.

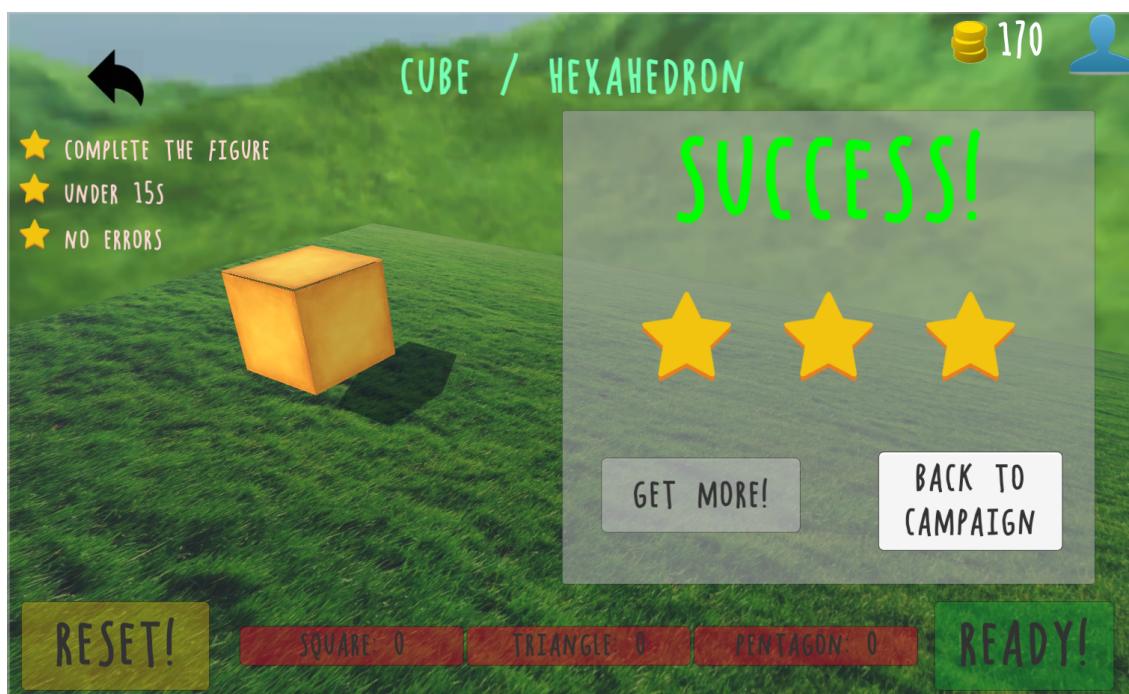


Figura 13: Visió del HUD, sense el panell de tutorial, amb càmera perspectiva

Renderitzat dels objectes 3D

Per cada figura 3D al joc 2D-3D s’han creat diferents materials per donar feedback visual a l’usuari. Per canviar-los es fa accés a seu component *MeshRendererEn* i canviant-hi l’atribut material. En concret els diferents materials per a cada objecte són:

- Previews: Transparent verd per marcar un preview vàlid i llest per instanciar-se i Transparent vermell per marcar un preview bloquejat.

- Peces 2D: Groc durant el desplegament i l'animació de plegament, i vermell quan es detecta un error o se selecciona per esborrar.

4.2.5 Disseny artístic

Més enllà de la programació, el joc també conté una sèrie de components visuals i auditives que val la pena comentar. S'ha separat de l'apartat de vista per donar-li importància afegida i perquè no està directament relacionat amb el model de la programació MVC.

- **Textures i sprites:** Han estat necessaris l'edició o creació de més de 40 sprites i textures diferents, agrupats en:

- Figures 2D: S'ha creat com a mínim una figura 2D per cada objectiu d'identificació, de color blanc i fons transparent, que durant el joc modifiquem (canviem de color).
- Figures 3D: S'ha creat una imatge de totes les possibles figures 3D objectius, siguin figures tridimensionals simples o figures tridimensionals més complexes (objectius finals de campanya). Ambdues s'usen en el menú de campanya.
- Avatar: S'han descarregat png dels 8 possibles avatars des de Google, tenint en compte el Copyright.
- Backgrounds i Skybox: Igual que l'Avatar, descarregats tenint en compte els criteris de Copyright.

- **Prefabs:** És comú a Unity la creació de *GameObjects* a la mateixa escena, afegint components i modificant-los com convingui. Són els anomenats Prefabs. Tot i així, és habitual trobar-te amb la situació de crear components que després requereixen ser referenciades des del codi, utilitzades en altres escenes, etc. En lloc d'haver de fer còpies, es pot guardar el *GameObject* en una estructura anomenada Prefab.

Hem creat Prefabs per les figures 2D, la imatge de background, les càmeres, estructura d'inventari i campanyes, bombes, semàfors o fins i tot la HUD, entre d'altres.

- **Models 3D:** S'ha creat un model 3D per a cada Campanya. Tot i que la primera idea era dissenyar-los utilitzant Blender, finalment s'ha optat per fer els models directament des de Unity.
- **Fonts:** S'han utilitzat diferents fons en tot el joc. Totes les fonts han estat extretes de la pàgina web www.1001fonts.com, d'on només s'han utilitzat algunes d'ús lliure. S'ha utilitzat una font digital especial pel timer del joc 2D, extreta del mateix lloc web [17].
- **So:** S'ha posat so en tot el joc. El so es controla des de la classe `noDestroy` (apartat 4.4.4), ja que aquesta classe està present en totes les escenes del joc.

Com els aparells electrònics actuals tots donen opció de apujar o abaixar el volum, no s'ha donat l'opció de fer-ho mitjançant l'aplicació en si. La música s'ha extret de www.bensounds.com, on es comparteix gratuïtament [18].

4.2.6 Persistència de dades amb JSONs i servidor extern

El treball està fet de manera que la gestió d'usuaris i companyes es realitza a partir de la lectura i escriptura de JSONs en fitxers de text. Això s'ha decidit fer-ho així per tal de facilitar la connexió amb el servidor una vegada estigui llest (vegeu secció 7.1). El servidor només haurà de llegir i escriure sobre aquests fitxers per tal que la connexió sigui completa.

La impossibilitat de disposar del servidor ha fet que el professor no tingui l'opció de crear i modificar companyes d'una manera simple. Tot i així, el professor podria modificar el fitxer de text per tal d'aconseguir aquesta demanda.

4.3 Diagrama de classes

Amb tot l'explicat fins ara, es presenta el diagrama de classes corresponent al joc 2D-3D (figura 14). S'ha optat per fer una visió reduïda d'algunes classes i només s'ensenyen atributs de control o essencials per entendre el funcionament intern del joc 2D-3D. Tampoc s'ensenyen mètodes, els més importants ja han estat comentats en els apartats anteriors.

En el següent apartat (secció 4.4) es desglossen algunes de les classes més importants.

4.4 Classes i entitats importants

Tot i que en tot el joc hi ha una gran quantitat de classes, creiem que val la pena explicar amb més detall les següents: User, Inventory i InventoryItem, Camapign i NoDestroy.

4.4.1 Inventory i InventoryItem

La classe Inventory és la que indica el progrés d'una campanya. Consta d'un identificador que sempre ha de coincidir amb el de la campanya a la qual fa referència, un booleà que indica si l'inventari està complet (s'usa per veure si la campanya està finalitzada) i finalment una llista de InventoryItems.

Un InventoryItem és una classe que representa un tipus de figura 2D o 3D (triangle, quadrat, pentàgon, però també tetraedre, prisme triangular, piràmide, etc). No només referència el tipus de figura sinó que també ens dóna informació de quina quantitat d'aquesta figura és necessària per completar la campanya amb èxit, la quantitat que tenim fins al moment, el path relatiu de la seva textura i, en el cas de ser un tipus de figura 3D, un atribut "cost" que ens indica quantes figures 2D

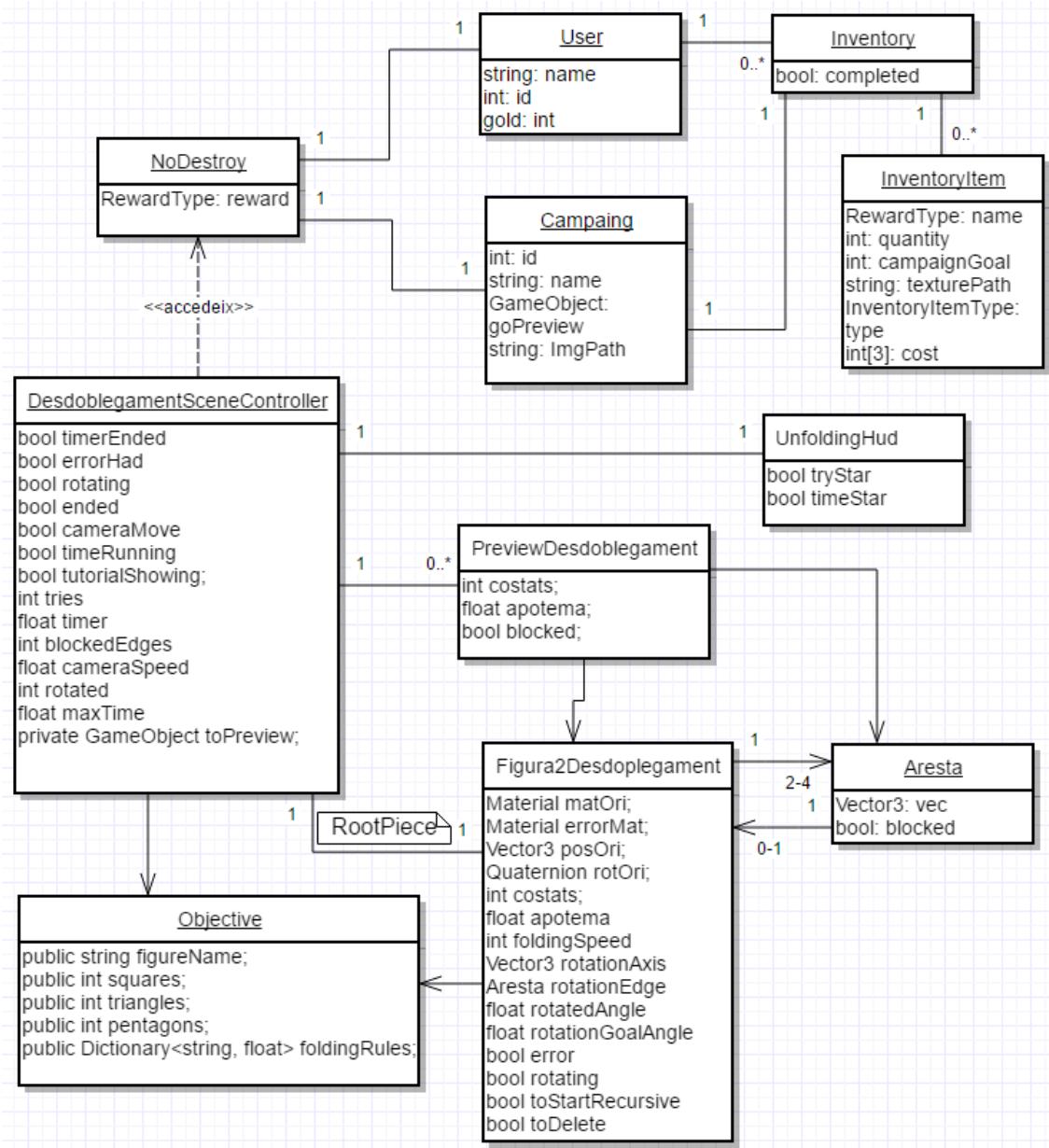


Figura 14: Diagrama de classes

són necessàries per crear-la. A mode d'exemple mostrem a continuació un inventari (figura 15) vist des de l'Inspector de Unity.

4.4.2 User

La classe User guarda tota la informació relacionada amb l'usuari. Per un cantó tenim atributs que fan referència al nom, identificador de l'usuari o a paràmetres de configuració, com l'avatar. També tindrà una llista d'inventaris que representarà el progrés de l'usuari en cada campanya. En començar una nova campanya, s'afegeix l'inventari de la campanya a la llista d'inventaris de l'usuari, que s'anirà actualitzant a mesura que vagi acabant missions. També guarda les monedes que s'aconsegueixen

Inventory	
Campaign ID	0
Completed	<input type="checkbox"/>
Items	
Size	4
▶ Element 0	
▶ Element 1	
Element 2	
Name	Cube
Quantity	0
Campaign Goal	4
Initial Goal	4
Type	Type 3D
Cost	
Size	3
Element 0	0
Element 1	6
Element 2	0
▶ Element 3	

Figura 15: Instància classe Inventory

al acabar una missió (2D i 2D-3D) i que poden ser utilitzades per comprar figures objectiu de la campanya o icones de personalització. A més d'aquests atributs, el més interessant són els que defineixen l'usuari des d'un punt de vista acadèmic. Aquí l'usuari tindrà:

- Un diccionari que té per índex el tipus de competència (recordem que es guarda en l'enumerador Competències, i són "Identificació", "Simetria", "Anglès", "Àrea i Perímetre") i per valor l'experiència adquirida d'aquella competència. Al completar amb èxit una missió 2D, i com s'ha explicat amb anterioritat, es donarà una experiència de la competència treballada. Aquesta experiència es traduirà en un nivell i el nivell posarà més o menys dificultats a les missions, com també s'explica més endavant en l'apartat 3.4.3.
- Diferents llistes de figures i objectius vistos. Aquestes llistes permetran gestionar quina missió pot fer cada usuari. La idea de fons és que els objectius de les missions no són de la mateixa dificultat i tenen, en molts casos, un cert ordre (per exemple, no demanarem trobar rectangles si no ha aconseguit acabar amb èxit com a mínim una missió de trobar quadrats o no demanarem trobar figures amb simetria rotacional, si no ha vist el que és la simetria axial). Així, aquestes llistes serviran per gestionar el fet que no faci missions no adequades per les coses que ha vist o après.

4.4.3 Campaign

La classe Campaign guarda tota la informació relacionada amb la campanya. Per un cantó té una sèrie d'atributs que fan referència al nom de la campanya i al seu

id, a un model 3D que serà la figura objectiu i una imatge que representarà la figura tridimensional en dues dimensions. A més d'aquests atributs, es consideren els següents elements:

- Inventari. Aquest inventari descriurà la missió en termes d'objectius a aconseguir per completar-la. L'inventari tindrà una llista de InventoryItems, que tindran quantitat igual a 0. Al iniciar una campanya, si l'usuari té un inventari amb id igual al id de la campanya, no es farà res. Si no té aquest id, s'afegeixà aquest inventari a la llista d'inventaris de l'usuari, fent una còpia. Aquest inventari no es modifica en cap moment.
- Tipus de campanya. La campanya té una variable booleana (weightMatter) que indica el tipus de campanya o, dit d'un altra manera, com s'escollliran les missions a fer durant aquella campanya. Per complementar aquest atribut es creen dos atributs més, un atribut per cadascun dels valors del booleà. Aquests atributs s'expliquen a continuació amb els dos tipus de mode de joc:
 - Pel mode lliure (weightMatter = false), s'afegeix una llista que guarda les competències a tractar.
 - Pel mode amb pesos (weightMatter = true) s'afegeix una llista de quatre floats que indica el pes que se li dóna a cada competència (primera posició el pes que se li dóna a identificació, segona a simetria, tercera a angles i quarta a perímetres i àrees).

Cal recordar que tot i que es pretenia que existís un servidor extern (vegis secció 4.2.6) per facilitar la feina al professor a l'hora de crear campanyes, això no ha estat possible i la manera de crear campanyes és, actualment, a través de la modificació d'un arxiu de text.

4.4.4 NoDestroy

Aquesta classe és una classe interna i auxiliar, però té importància cabdal en el funcionament del joc. S'encarrega de permanència de dades entre els salts d'escena. Així, aquesta classe guarda l'usuari que ha accedit al joc, mostrant sempre que sigui necessari els diners actuals i l'avatar escollit, i guarda la campanya que s'està cursant, sempre que se n'estigui realitzant alguna. Aquest fet permet consultar des de qualsevol escena els atributs de l'usuari i de la campanya actual, sempre que existeixin. També s'encarrega de guardar en cada moment l'objectiu actual de l'usuari perquè ho llegeixin els controladors de cada joc.

Aquesta classe també és l'encarregada de controlar la música mentre l'aplicació estigui oberta.

5 Resultats i simulacions

En aquest capítol es mostren els resultats finals del joc, tant en l'àmbit gràfic com tècnic. També es mostren els resultats d'un test d'usabilitat i d'un test de competències dissenyats específicament per aquest projecte.

5.1 Planificació final

A la figura 1 s'ha vist la planificació inicial del projecte en termes de formació, disseny, desenvolupament, testing i documentació. A continuació (figura 16) mostrem com s'ha transformat aquella previsió a la pràctica. Recordem que una unitat de temps correspon aproximadament a mitja setmana.

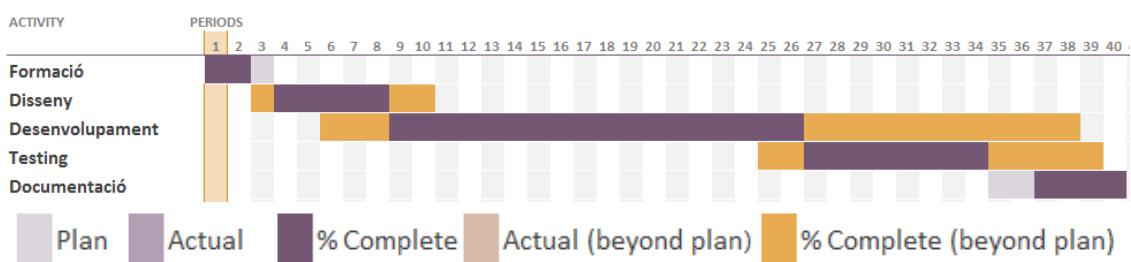


Figura 16: Planning real diagrama de Gantt vs previsió (S. Cebrián)

Com es pot observar a la figura, els plans no han acabat d'encaixar i s'ha superposat feina. La formació va requerir menys temps del que s'esperava i, per tant es va poder començar a dissenyar abans d'hora. El disseny i el desenvolupament van compartir les primeres setmanes del projecte, ja que s'havia de fer ànalisi dels problemes que anaven apareixent i redissenyat corresponentment.

Per factors externs, s'ha hagut d'ajornar el principi de la documentació fins a les darreres 2 setmanes i s'ha modificat codi fins a l'última setmana.

5.2 Resultats tècnics del joc

A continuació es mostren algunes característiques tècniques del joc:

- El joc funciona en sistema operatiu Windows. S'ha testejat en Windows 10.
- El joc ocupa gairebé 160MB i consta d'un executable (16MB), una carpeta on es guarden diferents dades (140MB) i un README.
- El joc està preparat per veure's en pantalles de dimensió 16 : 10. En la release, existeixen les següents opcions vàlides de resolució: 1260x768 és la recomenada, mentre que 640x400, 320x200 també mantenen el aspect ratio de 16 : 10. Es demana no utilitzar l'opció 1260x600, ja que s'han detectat errors durant el testing.
- Les targetes gràfiques on ha estat testejat el joc són de gamma baixa-mitja.

5.3 Resultats visuals del joc

En aquest apartat es vol mostrar els resultats finals obtinguts a GeoPieces, posant rellevància en la part visual del joc.

Com es pretenia, s'ha aconseguit dissenyar un joc centrat en la construcció d'edificis, englobats en el concepte campanya. A més, s'ha acabat donant un objectiu encara més ampli al joc, fent possible la construcció d'un poblat a mesura que es van aconseguint els seus edificis. A la figura 17 es mostra un exemple de l'escena 3D de selecció de campanyes

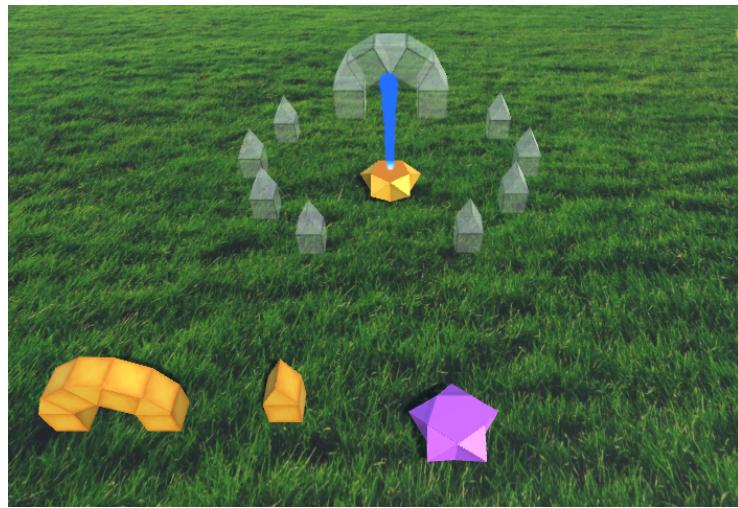


Figura 17: Pantalla de selecció de campanya. Els dos edificis inferiors de color taronja són clicable, mentre que el jugador ha completat la campanya de la font, que ja no és clicable i apareix al poblat amb textura i animació.

En el que respecta a una campanya en sí, s'ha aconseguit associar figures 2D i 3D, fent que els nens i nenes disposin de dues modalitats molt diferenciades del joc. A la figura 18 podem veure'n un exemple.



Figura 18: Exemple menú de campanya d'una campanya iniciada.

Quant al joc 2D-3D, s'ha aplicat un mètode dinàmic de creació de missions per a diferents figures 3D. S'ha aconseguit un sistema visual de detecció d'errors mitjançant les animacions dels plegaments. La figura 19, a continuació, en mostren diferents exemples.

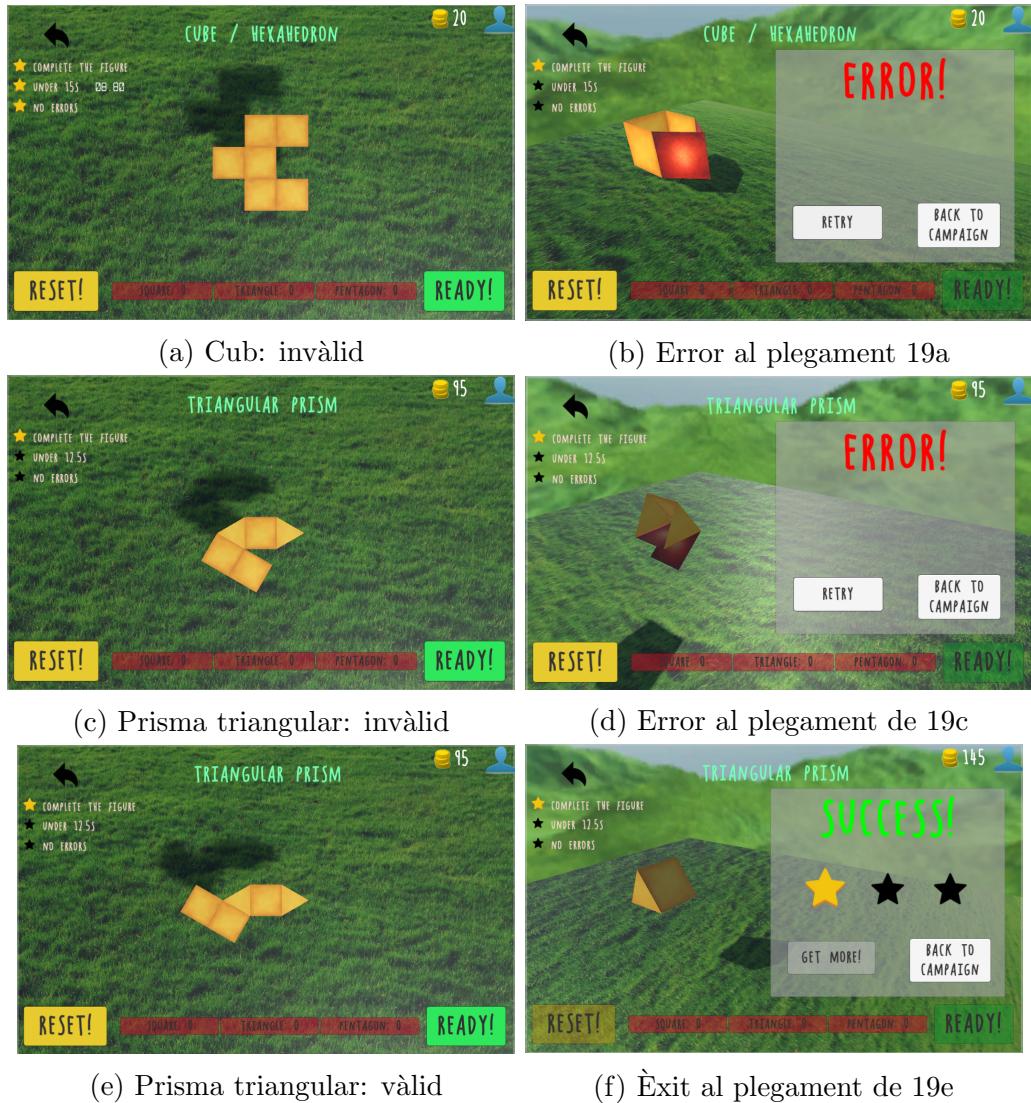


Figura 19: Exemples visuals d'errors i èxits als plegaments dels joc 2D-3D

5.4 Test del grau de reforç de les competències

Amb la idea d'intentar saber fins a quin punt serveix el joc per millorar les competències s'ha realitzat un test de reforç de competències.

El test consisteix en un primer qüestionari amb 6 preguntes multiresposta de conceptes geomètrics. Tot seguit, es deixa al nen o nena jugar al joc durant uns 15-20 minuts aproximadament. Una vegada acabat aquest procés, es repeteix el mateix qüestionari inicial per veure si ha après algun concepte nou.

El test s'ha realitzat a 4 nois i noies d'entre 10 i 12 anys el dia 22 de juny de 2016. Les preguntes del test es poden veure a l'apèndix C. Les respostes s'han agrupat en 4 categories:

- Correcte: Totes les opcions multi resposta correctes marcades.
- Parcialment correcte: S'han marcat respostes correctes, però no totes.
- Incorrectes: Alguna de les respostes marcades és incorrecte, independentment de si ha marcat alguna correcta.
- En blanc: No s'ha marcat cap opció.

Del total de sis preguntes multi respostes, quatre fan referència a la part 2D mentre que dos fan referència a la part de desplegament. Aquí ens hem centrat en les quatre primeres, tot i que citem els resultats de les altres.

Els resultats del test abans de començar a jugar han estat els que es veuen a la taula següent (taula 1).

Nº	Correctes	Parc. Correctes	Incorrectes	En blanc
1	0	4	0	0
2	0	3	0	1
3	2	1	1	0
4	1	0	3	0
5	0	3	1	0
6	0	2	1	1

Taula 1: Test de competències abans de jugar a GeoPieces

Els resultats del test després de jugar a GeoPieces es poden veure a la taula següent (taula 2).

Nº	Correctes	Parc. Correctes	Incorrectes	En blanc
1	2	2	0	0
2	1	0	2	1
3	3	1	0	0
4	2	0	2	0
5	2	2	0	0
6	0	3	1	0

Taula 2: Test de competències després de jugar a GeoPieces

Analitzem ara els resultats de les dues darreres preguntes, referents a la part 2D-3D.

- En començar, no hi ha cap resposta correcta al 100%. Després de la correcció, però, passa a haver-n'hi 2.
- El nombre de preguntes parcialment correctes es manté igual, ja que 2 passen a ser correctes i passa una pregunta d'incorrecta a parcialment correcta i una altra de en blanc a parcialment correcta.
- Hi ha ja només una resposta incorrecta, de dues inicials.

S'ha pogut comprovar durant el temps de joc, com els usuaris provaven diferents desplegaments dels vists en el test per comprovar i entendre la seva validesa. Cap dels alumnes del test ha deixat el test igual i tots ells han millorat els resultats.

Comentaris addicionals:

- Com el joc està en anglès al fer el testing s'ha requerit una persona al costat dels alumnes per tal d'explicar el que no entenen per la barrera de l'idioma.
- Tot i que el nombre d'alumnes enquestats no és prou alt per treure conclusions estadísticament rellevants, es fan patents algunes tendències positives.

5.5 Test de jugabilitat

Amb la idea d'intentar conèixer fins a quin punt és jugable el joc, si té una estructura fàcil de seguir, si és o no fàcil entendre els objectius, si enganxa o es fa pesat, etc. S'ha dissenyat un test d'usabilitat on hi ha participat una vintena d'usuaris de diferents edats. Per dissenyar-lo, hem utilitzat Google Forms.

El test conté preguntes quantitatives (valorar de l'1 al 5 una sèrie de característiques) de GeoPieces, tant de la part comuna, com de la part 2D i de desplegaments. A més a més, el test també conté preguntes quantitatives on els usuaris poden valorar les coses que més i menys els hi han agradat, entre d'altres.

A continuació mostrem les preguntes quantitatives i els seus resultats tant de la part comuna, com de la part 2D. Per fer-ho, utilitzem gràfics de barres. Les preguntes es valoren de l'1 al 5 on 1 és estar completament en desacord i 5 completament d'acord. Les preguntes generals de GeoPieces realitzades són les que citem a continuació:

1. L'aparença del joc és agradable (figura 20a).
2. El joc es controla de manera intuitiva (figura 20b).
3. La navegació entre menús resulta clara (figura 20c).
4. És fàcil entendre l'objectiu del joc (figura 20d).
5. És fàcil entendre l'objectiu d'una campanya (figura 20e).
6. Recomanaries aquesta aplicació per alumnes de primària (figura 20f).

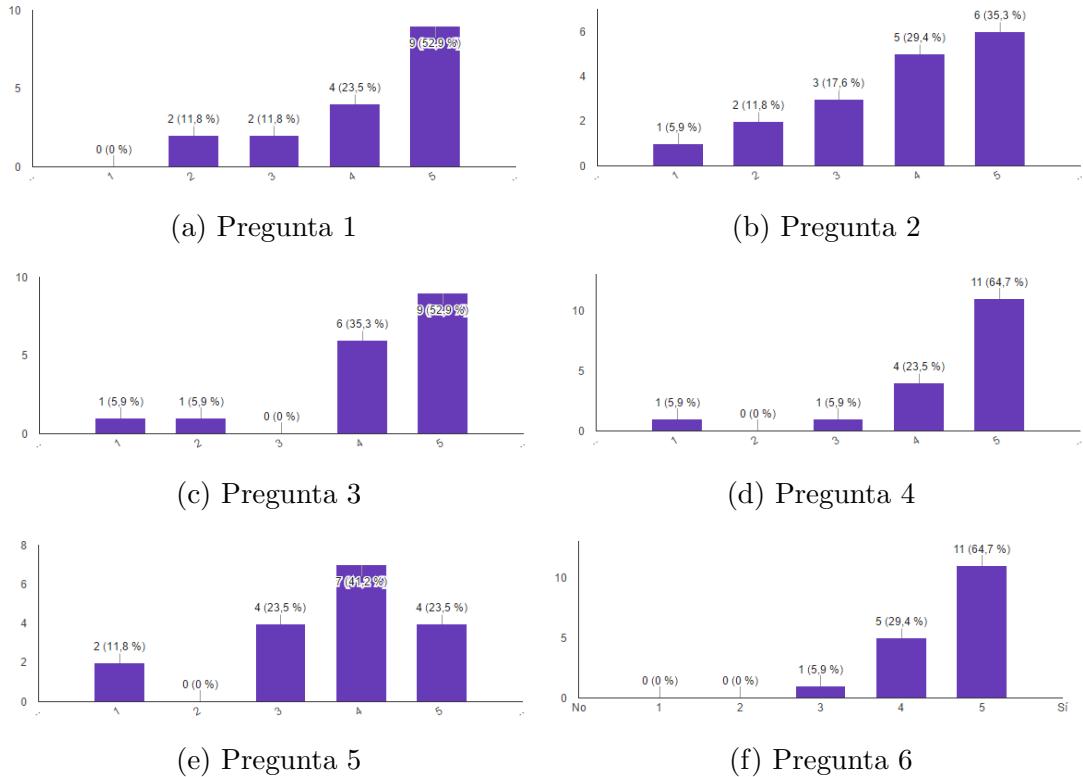


Figura 20: Test de usabilitat: preguntes genèriques GeoPieces

En general les preguntes han estat respostes de manera força positiva. Potser cal destacar la pregunta 2, que fa referència a la intuitivitat del joc i la pregunta 5, on es parla de la fàcil comprensió de l'objectiu final de la campanya. Cap resultat, per això, és realment alarmant.

Les preguntes concretes de la part 2D-3D realitzades es veuen a continuació i els seus resultats poden veure's a la figura 21.

1. La mecànica del joc és intuïtiva (figura 21a).
2. Les animacions ajuden a la detecció d'errors (figura 21b).
3. En cas d'error en una partida, s'entén on s'ha produït (figura 21c).

Com podem veure, els resultats de la primera pregunta són pocs satisfactoris pel que fa a la facilitat d'entendre la mecànica del joc. Tot i que inicialment s'havia plantejat disposar de petits videotutorials, en la versió beta implementada no s'han inclòs. Aquests podrien facilitar la comprensió de la mecànica significativament. Els resultats de les altres preguntes són molt positius.

El test iclou també preguntes genèriques de resposta redactada. D'aquestes es destaquen aquí les respostes més comuns i les que més afecten el joc 2D-3D per a ser analitzades:

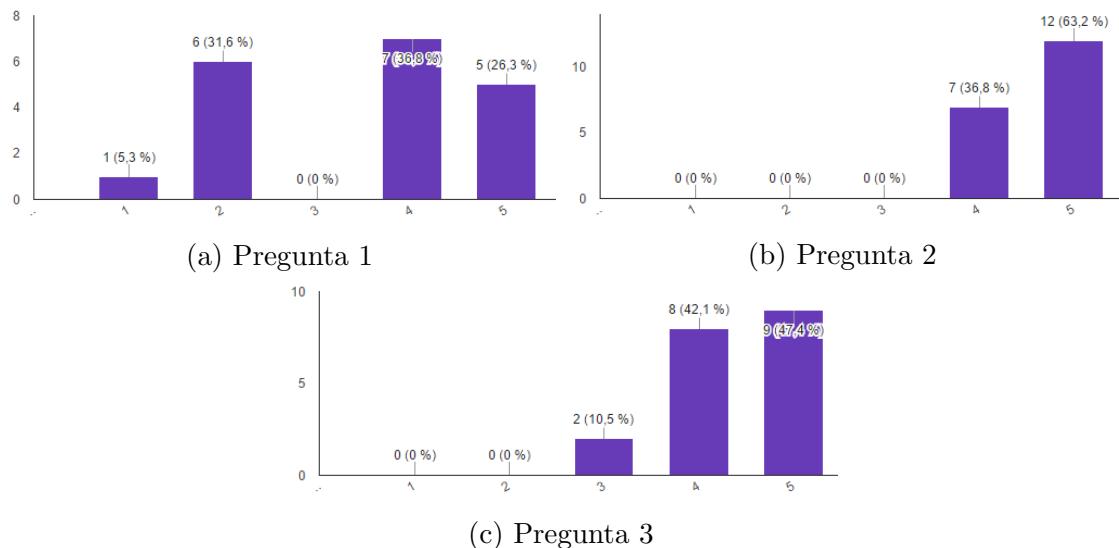


Figura 21: Test de usabilitat: preguntes sobre el joc 2D-3D

- Milloraries alguna cosa? En cas afirmatiu, quina o quines?
 - Millorar música
 - No he entès quina és la funció del botó reset quan estic al món 3d.
 - Al món 3d, posaria a algun lloc de la pantalla la figura que s'ha de fer.
 - Obligar a llegir les definicions dels objectius almenys una vegada.
- Què és el que més t'ha agradat?
 - La música.
 - L'animació de les figures 3D (reiteradament)
 - Trobo el joc molt intuïtiu.
 - Com van desapareixent les estrelles
- Què és el que menys t'ha agradat?
 - Els menús.
 - Els textos d'objectius.
 - Es pot arribar a fer una mica pesat per nens petits.

Després de la realització del test, es va decidir prendre una sèrie de mesures. A continuació se'n mostren algunes:

- Posar la música en loop. Això ja s'ha inclòs a la versió final del projecte.
- Afegir un fil argumental al joc.
- Afegir un temps o panell d'espera abans de començar les missions 2D-3D.

6 Valoració econòmica

Per conèixer el cost aproximat del projecte, s'ha de tenir en compte el temps utilitzat pel programador, en què s'ha utilitzat, i els components de hardware i software necessaris per dur-lo a terme.

6.1 Anàlisi del temps de realització del projecte

El temps utilitzat en la realització d'aquest projecte es pot dividir en cinc parts:

- Hores de formació. Aprendre a utilitzar el motor Unity3D i estàndards de programació en c#. Modelatge 3D.
- Hores d'anàlisi. Conèixer competències bàsiques de geometria a primària, veure jocs d'aprenentatge i fer disseny del joc GeoPieces.
- Hores d'implementació. Desenvolupament de codi i disseny gràfics (models, materials, figures2D, escenaris...). Inclou hores de proves realitzades pel programador.
- Hores de testing. Realització dels tests d'usabilitat i test de competències. Inclou el temps de preparació dels tests, el temps de duració dels tests presencials amb els alumnes i el temps corresponent a l'anàlisi dels resultats.
- Hores de documentació. Memòria i comentar codi.

A la taula següent (taula 3) es mostra el temps aproximat utilitzat en cadascuna dels grups d'hores citades. S'ha posat també el temps en setmanes per ajudar a valorar la quantitat d'hores, considerant que una setmana són 15h setmanals (3h per dia durant 5 dies).

Tipus	Temps en hores	Temps en setmanes
Formació	15	1
Anàlisi	37.5	2.5
Implementació	195	13
Testing	7.5	0.5
Documentació	45	3
Total	300	20

Taula 3: Temps usat en el projecte (en hores i setmanes) per S. Cebrián

6.2 Valoració econòmica

El primer que s'ha de tenir en compte és que no a totes les hores es pot imputar un cost. S'assumeix que el programador és un programador expert i no requereix

hores de formació. La part de documentació no és tampoc totalment atribuïble al programador. Així, només posarem com a cost el temps de documentació dedicat al manual d'usuari i manual tècnic, aproximadament 3h del temps de documentació.

El cost per hora depèn de la tasca a realitzar. A continuació (figura 4) es mostra el cost per hora en cadascuna de les tasques abans citades i el preu total del programador 2D-3D en hores per tot el projecte.

Els preus per hora establerts són els equivalents als valors mitjans de cost per hores per consultors en el món empresarial.

Tipus	Temps en hores	Cost per hora	Total en €
Anàlisis	37.5	80	3000
Implementació	195	60	11700
Testing	7.5	50	375
Documentació	3	50	150
Total			15225

Taula 4: Cost per tasca i hores de S Cebrián

Tenint en compte els dos treballs que participen en el projecte, tenint en compte que la llicència de Unity és gratuïta i que ambdós necessiten un ordinador de gamma mitjana-alta, que valorem en 800€, el cost final del projecte és el següent (taula 5)

Tipus de cost	Cost en €
Programador 2D	14750
Programador 2D-3D	15225
Ordinador x 2	1600
Total	31575

Taula 5: Pressupost del projecte

7 Conclusions

En aquest projecte s'ha desenvolupat un joc per l'aprenentatge de la geometria d'alumnes d'últims cursos de primària.

El joc consisteix en tres parts ben diferenciades que expliquem a continuació.

- Part comuna. Aquí s'ha basat tot el desenvolupament en el concepte Campanya i el concepte Usuari.

La campanya ha servit per donar un objectiu final al joc amb la construcció d'un edifici, a més de ser utilitzada com a eina per escollir quins tipus de competències geomètriques s'aprenen. Les campanyes són configurables i tot i permetre dos modes de joc, sempre intenta adaptar-se al nivell del nen o nena.

La informació acadèmica de l'alumne està encapsulada en la figura de l'usuari. Aquí es guarda tota la informació de competències apreses i objectius per aprendre.

A més, ha estat necessari crear un conjunt d'escenaris per tal de poder navegar pel joc, donar la possibilitat d'aconseguir noves figures a través de reptes i inclús donar alguna opció de customització.

- Part 2D (correspondent al treball de fi de grau del D. Rausell). És la part del joc que serveix per aconseguir les cares 2D a través d'una sèrie de reptes depenent de les competències de l'alumne en cada moment. Per més informació, veure el treball del D. Rausell [4].
- Part 2D-3D. És la part del joc que serveix per aconseguir poliedres elementals necessàries per a completar les campanyes.

L'objectiu del joc 2D-3D és, gastant les peces aconseguides a la part 2D, presentar un desplegament pla vàlid d'un poliedre donat, obtenint-lo com a premi. Així s'entrena la intel·ligència espacial i s'apren a relacionar les figures 3D amb les cares que les conformen.

Depenent dels cops que l'usuari hagi guanyat el mateix poliedre objectiu i de la complexitat del mateix, la dificultat varia de manera adequada. S'han vist els diferents factors a tenir en compte a l'hora d'assignar una dificultat a una missió i, conseqüentment, la seva recompensa.

Previ a començar el desenvolupament, es va quedar en fer dues reunions setmanals per coordinar la feina feta: Una reunió els dilluns d'integració entre els dos desenvolupadors i una els dijous amb la tutora del treball per enfocar la feina de cada setmana en una estructura semblant a la que segueix la filosofia SCRUM. Aquestes reunions s'han realitzat cada setmana amb molt poques excepcions al llarg de tota la duració del projecte.

De tots els objectius marcats a l'inici del projecte, l'únic que no s'ha pogut arribar a dur a terme ha estat el joc 3D per manca de temps. Aquest joc hauria

funcionat de manera similar al joc 2D-3D, sobretot en l'estructura de graf i l'eina d'ajut per la col·locació de peces. Es va abandonar la idea de desenvolupar-lo en les primeres fases del disseny per la seva complexitat i manca de temps.

El resultat del projecte final és un joc que s'ajusta adequadament a les diferents teories didàctiques que s'han comentat al llarg d'aquesta memòria. Per valorar el resultat, tant com a eina d'aprenentatge com a joc, s'han realitzat a més dos tests: El primer, de jugabilitat on hi han participat una vintena d'usuaris de diferents edats; El segon, de competències amb quatre alumnes de les edats objectiu, analitzant els seus coneixements abans i després de jugar. Les valoracions extretes han estat molt positives, però han deixat clar que hi ha espai per millorar.

Les principals complicacions que s'han hagut de superar han estat la integració entre els desenvolupadors i el disseny en forma de graf dels desplegaments al joc 2D-3D, que va tardar unes quantes iteracions a perfeccionar-se. Hi havia poca bibliografia sobre el tema de plegaments automàtics i d'allà va sorgir la idea de basar-los en una heurística de regles entre cares i els seus angles possibles. Tot i així, l'experiència en general del desenvolupament del joc 2D-3D ha estat molt bona i s'han après coses noves sobre el disseny de jocs i de l'entorn de Unity.

La conclusió final del projecte és, doncs, molt positiva, i s'espera que aquest pugui, en properes iteracions, arribar a ser emprat a classes o impulsar l'ús de noves tecnologies a l'aula. L'objectiu final del projecte era, com ja s'ha dit al principi d'aquesta memòria, aprendre fent.

7.1 Treball futur

Tot i que hi ha sempre molts camins per on anar per tal de seguir amb el projecte, a continuació citem alguns dels que més presents s'han tingut durant la realització d'aquest treball de fi de grau:

- Joc 3D. Una vegada es tenen els objectes 3D més simples, es podrien modelar escenaris 3D amb missions per tal d'ajuntar aquests objectes simples per formar l'objectiu tridimensional de la campanya.
- Aconseguir la connexió amb el servidor per tal que el professor o professora pugui crear noves campanyes de manera ràpida. A més, es podria donar al docent la possibilitat de realitzar el seguiment del progrés dels estudiants durant l'execució de les missions.
- Lligat amb el punt anterior, donar l'opció de jugar en mode “Classroom”, on el professor pot haver preparat una sèrie de campanyes que han de completar els seus alumnes i dóna a classe un nom i una contrasenya per tal que tots s'hi puguin connectar.
- Donar la possibilitat de crear noves missions 2D i 3D i afegir noves figures de manera senzilla a través d'un portal web d'ajut al professorat.

Referències

- [1] UNESCO. *La educación para todos, 2000 - 2015: Logros y desafíos*, 2015.
- [2] Informe PISA 2012. *Programa para la evolución internacional de los alumnos*, 2012.
- [3] Jackson, S.; *Unity 3D UI Essentials*, ISBN 1783553618, Packt Publishing, 2015.
- [4] Rausell, D.; *Geopieces: Joc seriòs per l'aprenentatge de la geometria. Part 2D*, TFG, 2016.
- [5] Mason, M. *The van Hiele levels of geometric understanding. The professional handbook for teachers: Geometry*, vol. 4, p. 4-8, 1998.
- [6] Gardner, H. *Frames of Mind: The Theory of Multiple Intelligences*, 1983.
- [7] DragonBox Algebra/DragonBox2, WEB, Last access June 2016.
- [8] DragonBox Elements WEB, Last access June 2016.
- [9] Mathtoons Media Inc. WEB, Last access June 2016.
- [10] Fendt, W. Java Applets zur Mathematik WEB, Last acces June 2016.
- [11] WisWeb applets WEB, Last access June 2016.
- [12] Math playground WEB, Last access June 2016.
- [13] Geogebra WEB, Last access June 2016.
- [14] CryEngine WEB, Last access June 2016.
- [15] Unreal Engine 4 WEB, Last access June 2016.
- [16] Carey, R. Game Design Canvas WEB, Last access May 2016.
- [17] 1001 fonts WEB, Last acces April 2016.
- [18] BenSounds WEB, Last acces June 2016.
- [19] Horacheck, D. *Creating E-Learning Games with Unity*, Març 2014.
- [20] Norton, T. *Learning C# by Developing Games with Unity 3D*, Setembre 2013.
- [21] Doran, J. P. *Unity Game Development Blueprints*, Novembre 2014.
- [22] Rodriguez, I.; Puig, A.; Cortada, V.; Grau, S.; Escayola, M.: *Designing a Math Game for Children Using a Participatory Design Experience*, Novembre 2015.
- [23] Muriel, C: *Fracsland: joc serós per aprendre fraccions*, TFG, 2016.

A Manual tècnic

A.1 Instal·lació: requeriments mínims i passos a seguir

A continuació s'explicarà el procés d'instal·lació del joc GeoPieces. Abans d'això, són requeriments necessaris:

- Sistema Operatiu Windows. S'ha testejat en Windows 10.
- Es requereixen gairebé 160MB: consta d'un executable (16MB), una carpeta on es guarden diferents dades (140MB) i un README.
- Prestacions stàndards, com poden ser per exemple un iCore5 a 5GHz o un iCore3 a 3,7GHz, amb targeta gràfica integrada.

Si no es compleixen aquestes condicions no es garanteix el bon funcionament del joc.

Dit això, els passos per a poder jugar a GeoPieces són:

1. Descarrega l'arxiu zip de l'enllaç: <http://tinyurl.com/j5yko77>
2. Exporta l'arxiu zip.
3. En aquest punt haures de tenir un arxiu executable GeoPieces.exe, una carpeta amb arxius i un README.txt.
4. Executa l'arxiu GeoPieces.exe.
5. Desmarca l'opció “Windowed” i selecciona la resolució de pantalla 1260x768 d'entre les diferents opcions. Si no dona aquesta opció, escull la resolució 640x400 o, en el seu defecte, 320x200.
6. Aquesta informació també la pots trobar al README.txt.

B Manual d'usuari

B.1 Manual del professor

Els paràmetres modificables actualment en aquest projecte són:

- Estil de missió
- Pes de cada competència si s'ha escollit estil per pesos.
- Competències a treballar si s'ha escollit estil sense pesos.

A continuació s'explica com fer-ho.

Modificar paràmetres de campanya

IMPORTANT: Els canvis que s'expliquen a continuació afecten al projecte en Unity, no a la release. Per tenir una nova release amb els canvis s'hauria de tornar a fer un build des de Unity. Per tant, la manera més simple de provar-ho és directament al playmode de Unity.

Aquest manual serveix per modificar paràmetres de les campanyes. Es pot modificar les campanyes "Catalan arch", "Simple House" o "Fountain". En el projecte Unity, a la carpeta Assets/Resources/Campaigns modifcar l'arxiu json i txt (els dos) de la manera següent:

1. Obrim arxiu amb editor de text (Notepad, Sublime o similars).
2. Si busquem (podem utilitzar CTRL+F) la paraula weightsMatter, veurem que al seu costat apareix la paraula true o false. Modifica aquesta paraula si vols canviar el mode de joc de la pantalla. Fes-ho així:

“weightsMatter” : *word*

on *word* pot ser la paraula *true* o *false*

3. Si escrius *true*, l'atribut immediatament anterior dóna la possibilitat de decidir els pesos (weights) de cadascuna de les competències. El format d'escriptura de weights ha de ser:

“weights” : [number1, number2, number3, number4]

on *numberX* ha de ser un número que pot contenir decimals utilitzant el punt per separar-la de la part entera. Exemples vàlids: 5, 3.27, 71, 42.

4. Si escrius *false*, l'atribut immediatament posterior permet decidir quin tipus de competències es volen treballar. El format d'escriptura d'aquest atribut és:

“compToLearn” : [number1, ..., numberN]

on $numberX$ ha de ser un número del 0 al 3. Com a molt N pot ser igual a 4. Cada número ha de ser inferior al següent.

Exemples vàlids:

- “*compToLearn*” : [1]
- “*compToLearn*” : [0, 1]
- “*compToLearn*” : [0, 3]
- “*compToLearn*” : [1, 2, 3]

Exemples no vàlids:

- “*compToLearn*” : [4]
- “*compToLearn*” : [0, 2, 1]
- “*compToLearn*” : [0, 2, 2]
- “*compToLearn*” : [0, 1, 2, 3, 3]

5. Guarda i tanca els fitxers.
6. A Unity, entra a playmode i crea un nou usuari. Les campanyes han canviat els seus objectius.

B.2 Manual de l'estudiant: Aprèn a jugar

Primers pasos per jugar al joc GeoPieces:

1. Descarrega l'arxiu .zip i extreu les dades en una carpeta.
 2. Executa l'arxiu GeoPieces.exe.
 3. Desmarca l'opció “Windowed” i selecciona la resolució de pantalla $1260x768$ d'entre les diferents opcions. Si no dona aquesta opció, escull la resolució $640x400$ o, en el seu defecte, $320x200$.
 4. Prem el botó “Play”.
 5. Si és el primer cop que jugues, escull un nom de mínim 3 lletres i escull una contrasenya de mínim 8 lletres i després fes clic sobre “Sign Up” i a continuació “Log in” (figura 23).
 6. Si no entres per primera vegada, escriu el teu nom i la teva contrasenya i prem directament el botó “Log in”.
- NOTA: Si cliques sobre el botó sign up amb un nom ja utilitzat es borrarán les dades d'aquell usuari i s'inicialitzarà un de nou.
7. Ja ets dins el joc (figura 24).
 8. Ves a “Campaign Mode”

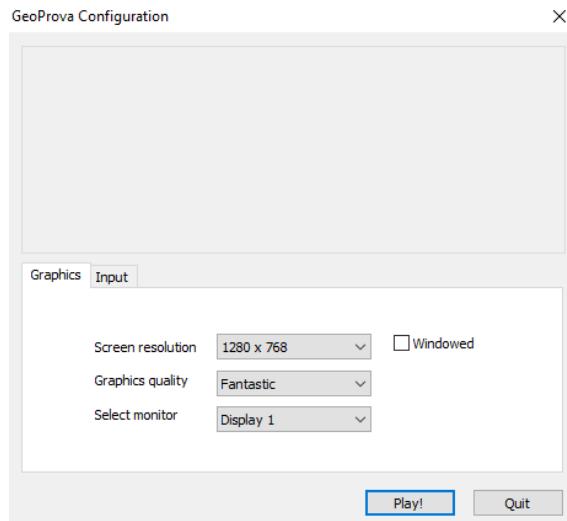


Figura 22: Pantalla de configuració prèvia a GeoPieces.



Figura 23: Pantalla de sign up i log in.



Figura 24: Menú principal.

9. Selecciona una campanya fent clic sobre una de les figures tridimensionals (figura 25).
10. Has entrat al menú de campanya (figura 26).
11. Aconsegueix figures 2D clicant sobre “Get Square”, “Get triangle” o “Get Pentagon” (figura 26).



Figura 25: Select Campaign.



Figura 26: Menú de campanya

12. Clica sobre les figures objectius i prem el botó “Check figures” quan creus que has acabat.

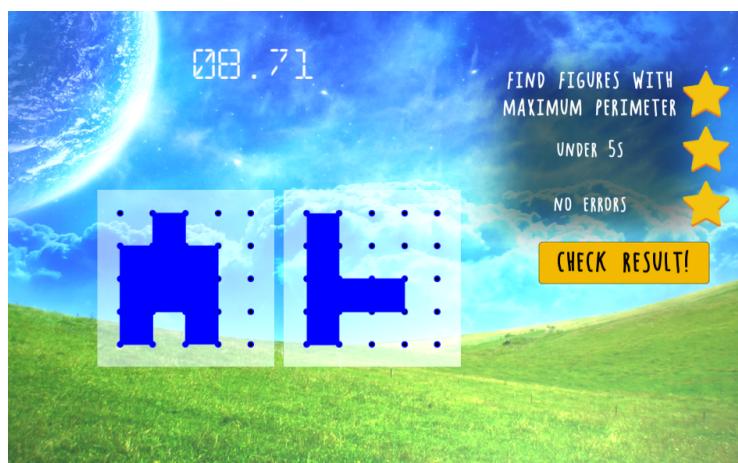


Figura 27: Món 2D

13. Quan tinguis prous figures 2D per poder fer el desplegament, aconsegueix les

figures 3D anant a “Get Cube”, “Get Pentagonal Prism” o variants.

14. Per fer les figures 3D, escull quin tipus de figura vols posar al desplegament i després sobre les caselles verdes per col·locar figures (figura 28).
15. Per comprovar el resultat, fes click al botó de Ready un cop ja hagis posat totes les peces en joc.



Figura 28: Món desplegaments

16. Fes el mateix amb les altres campanyes i aconsegueix reconstruir el poblat.

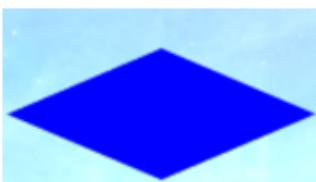
C Test de competències

Selecciona els rombe:

a)



b)



c)



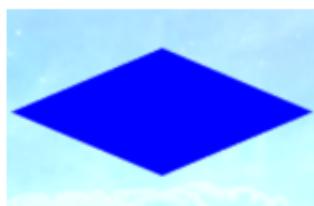
d)



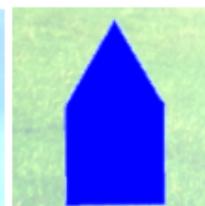
Figura 29: Pregunta 1 Test de Competències

Selecciona les figures amb simetria rotacional:

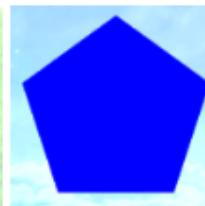
a)



b)



c)



d)



Figura 30: Pregunta 2 Test de Competències

Selecciona les figures d'àrea màxima:

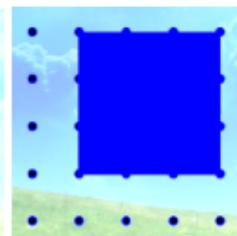
a)



b)



c)



d)



Figura 31: Pregunta 3 Test de Competències

Selecciona les figures de perímetre mínim:

a)

b)

c)

d)

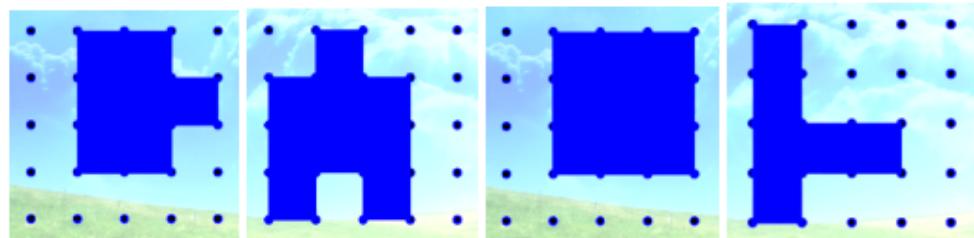
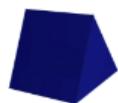


Figura 32: Pregunta 4 Test de Competències

Encercla els desplegaments que formen la figura següent:



a)

b)

c)

d)

e)

f)

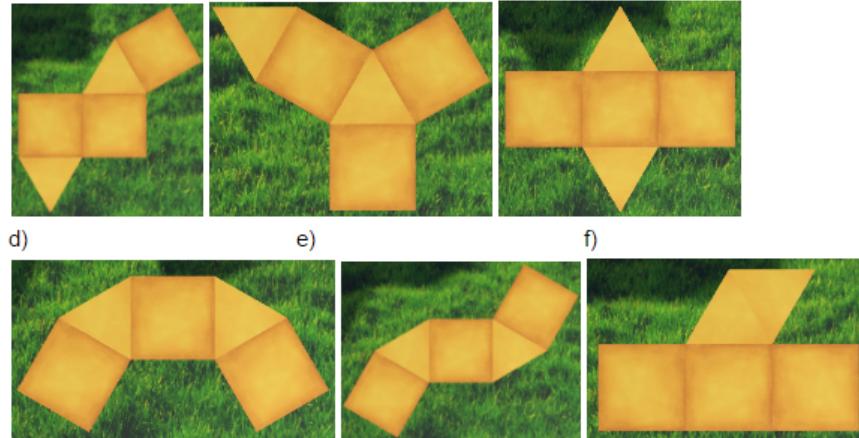
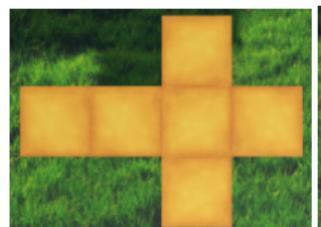


Figura 33: Pregunta 5 Test de Competències

Encercla els desplegaments que formen la figura següent:



a)



b)



c)



d)



e)



f)

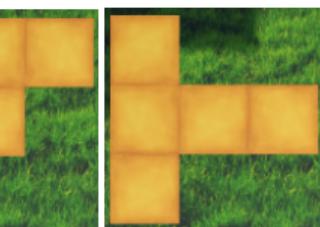


Figura 34: Pregunta 6 Test de Competències