



UNIVERSITAT POLITÈCNICA DE CATALUNYA  
BARCELONATECH  
Escola d'Enginyeria de Barcelona Est

TREBALL FI DE GRAU

**Grau en Enginyeria Mecànica**

**APLICACIÓ DE REALITAT VIRTUAL NO INVASIVA PER A  
EXTREMITATS SUPERIORS**



**Memòria i Annexes**

**Autors:** Arnau Doñate i Rivilla  
Oriol Jiménez Bellido

**Director:** Jordi Torner Ribé

**Departament** Departament d'Enginyeria Gràfica i de Disseny (DEGD)

**Co-Director:** Francesc Alpiste Penalba  
Gil Serranoli Masferrer

**Convocatòria:** Juny 2020



## Resum

Com en tots els camps de la medicina, la fisioteràpia, ha experimentat diferents formes de tractaments segons el moment històric i tecnològic. Actualment, hi ha una tendència a l'alça a la implementació de dispositius de realitat virtual a les teràpies. Aquests avenços suposen una personalització dels tractaments i una experiència més lleugera per a les persones. Els jocs formen part d'una sèrie d'aplicacions que s'utilitzen per a finalitats de recerca en l'àmbit de la rehabilitació a distància.

L'objectiu d'aquest projecte és crear una sèrie de mini-jocs per a l'Associació de Diversitat Funcional d'Osona (ADFO), associació que es planteja utilitzar les tecnologies esmentades en els seus processos de rehabilitació. Aquests jocs, han estat dissenyats sota la tutela de les persones especialistes del centre, i seguint les seves directrius, s'han creat quatre jocs.

Aquests jocs serveixen per a exercitar les extremitats superiors en totes aquells o aquelles pacients amb seqüeles degudes a malalties que puguin interferir en el seu dia a dia. La plataforma de joc és en dues dimensions i mitjançant l'ús del ratolí d'un ordinador s'exerciten les parts esmentades, segons una sèrie de moviments en un pla horitzontal. Aquests moviments al no ser molt exagerats i realitzar-se en un entorn dinàmic, ajuden a les persones d'una forma molt poc invasiva.

Per a la realització d'aquesta tasca s'ha disposat d'una sèrie d'eines informàtiques que permeten el desenvolupament d'aquestes aplicacions, concretament, el software Unity. Gràcies a aquest entorn de treball s'han generat els espais virtuals amb els quals la persona interactua, i mitjançant programació informàtica, s'han establert els funcionaments bàsics dels jocs.

Tot i trobar-se en una fase inicial, aquest projecte ha rebut bones crítiques per part dels i les especialistes. S'ha decidit realitzar modificacions dels jocs en funció de les proves de funcionalitat que s'han fet.

## Resumen

Como en todos los campos de la medicina, la fisioterapia, ha experimentado diferentes formas de tratamientos según el momento histórico y tecnológico. Actualmente, hay una tendencia a la alza con la implementación de dispositivos de realidad virtual en las terapias. Estos adelantos suponen una personalización de los tratamientos y una experiencia más ligera para las personas. Los juegos forman parte de una serie de aplicaciones que se utilizan para finalidades de investigación en el ámbito de la rehabilitación a distancia.

El objetivo de este proyecto es crear una serie de mini-juegos para la *l'Associació de Diversitat Funcional d'Osona* (ADFO), asociación que se plantea utilizar las tecnologías mencionadas en sus procesos de rehabilitación. Estos juegos, han sido diseñados bajo la tutela de las personas especialistas del centro, y siguiendo sus directrices, se han creado cuatro juegos.

Estos juegos sirven para ejercitar las extremidades superiores en todas aquellos o aquellas pacientes con secuelas debidas a dolencias que puedan interferir en su día a día. La plataforma de juego es en dos dimensiones y mediante el uso del ratón de un ordenador se ejercitan las partes mencionadas, según una serie de movimientos en un plano horizontal. Estos movimientos al no ser muy exagerados y realizarse en un entorno dinámico, ayudan a las personas de una forma muy poco invasiva.

Para la realización de esta tarea se ha dispuesto de una serie de herramientas informáticas que permiten el desarrollo de estas aplicaciones, concretamente, el software Unity. Gracias a este entorno de trabajo se han generado los espacios virtuales con los cuales la persona interactúa, y mediante programación informática, se han establecido los funcionamientos básicos de los juegos.

A pesar de encontrarse en una fase inicial, este proyecto ha recibido buenas críticas por parte de los y las especialistas. Se ha decidido realizar modificaciones de los juegos en función de las pruebas de funcionalidad que se han hecho.

## Abstract

As in all fields of medicine, the physiotherapy has experienced different forms of treatment according to the historical and technological moment. Currently, there is an upward trend with the implementation of virtual reality devices in therapies. These advances represent a personalization of the treatments and a lighter experience for people. The games are part of a series of applications that are used for research purposes in the field of distance rehabilitation.

The aim of this project is to create a series of mini-games for the Associació de Diversitat Funcional d'Osona (ADFO), an association that plans to use the aforementioned technologies in its rehabilitation processes. These games have been designed under the supervision of the specialists of the center, and following their guidelines, four games have been created.

These games serve to exercise the upper extremities in all those or those patients with sequelae due to injuries that may interfere with their daily activities. The game platform is in two dimensions and through the use of a computer mouse the mentioned body parts are exercised, according to a series of movements in a horizontal plane. These movements, not being exaggerated and performed in a dynamic environment, help people in a very non-invasive way.

To carry out this task, a series of computer tools have been made available in Unity that allow the development of these Applications. Thanks to this environment, the virtual spaces with which the person interacts have been generated, and through computer programming, the basic operations of the games have been established.

Despite being in an initial phase, this project has received good reviews from specialists. It has been decided to make modifications to the games based on the functionality tests that have been done.

## **Agraïments**

Tot i la situació excepcional en la que s'ha dut a terme aquest projecte volem agrair el suport incondicional de les nostres famílies, i dels que ja no hi són , tant en l'esforç econòmic que ha suposat la realització dels nostres estudis com en el suport emocional.

També volem agrair a la Universitat Politècnica de Catalunya les eines de les que disposem per a poder afrontar el món laboral. Aquesta tasca no hauria estat possible sense el professorat de l'escola pública, especialment agrair el paper que han dut a terme en Jordi Torné Ribé, Francesc Alpiste Penalba i Gil Serranoli Masferrer.

Gràcies a tots els companys i companyes del dia a dia que han fet aquesta experiència més enriquidora.

-Arnau Doñate i Oriol Jiménez-

## Glossari

-**Asset**: Nom que rep qualsevol arxiu emprat en el software Unity.

-**Audio source**: Component de Unity que controla l'Audio Clip en una escena.

-**Backup**: Es refereix a la còpia i arxivament de dades de la computadora de manera que es poden utilitzar per restaurar la informació original després d'una eventual pèrdua de dades.

-**Bug**: Un error de software o fallada en un programa d'ordinador o sistema de software que desencadena un resultat no desitjat.

-**Build**: Nom que rep la construcció d'un executable per a poder fer córrer l'arxiu sense necessitat de Unity.

-**Collider**: Defineix el contorn d'un objecte. No necessàriament ha de ser igual de gran que la forma d'aquest i li atorga un espai tridimensional amb el que es pot interactuar.

-**IE enumerator**: Funció que s'utilitza per a aturar iteracions o fragmentar llargues llistes d'arxius.

-**Machine learning**: Disciplina científica del àmbit de la intel·ligència artificial. Aquesta crea sistemes basats en algoritmes que permeten "aprendre" a les màquines com resoldre problemes.

-**On Click()**: S'utilitza per a assignar un comportament de botó a un objecte. Se li assigna un codi que controla el botó i l'escena que carregarà aquest click.

-**Particle system**: Conjunt de partícules que es generen en una posició concreta amb un comportament aleatori. Es sol utilitzar per a la generació d'efectes visuals com focs o explosions.

-**Rigid body**: Component que pot rebre i transmetre forces i parells. S'aplica a objectes per a que aquests experimentin magnituds físiques i puguin interactuar amb altres objectes mitjançant el motor físic de Unity.

-**Royalty free**: Dret d'utilitzar un material d'autor o propietat intel·lectual sense la necessitat de pagar.

-**Scripts**: Codi de programació que s'executa de forma seqüencial.

-**Start up**: També conegut com a empresa emergent, són empreses de creació recent.

-**Sprites**: Són objectes 2D simples que porten imatges gràfiques (anomenades textures).

-**Standalone**: Versió d'un projecte que es pot executar per si sol, sense necessitat de programes intermediaris.

# Índex

<b>1. PREFACI</b> .....	<b>1</b>
1.1. ORIGEN DEL TREBALL .....	1
1.2. MOTIVACIÓ .....	2
1.3. REQUISITS PREVIS .....	2
1.4. ADFO .....	3
<b>2. INTRODUCCIÓ</b> .....	<b>7</b>
2.1. OBJECTIUS DEL TREBALL .....	8
2.1.1. <i>Objectiu general</i> .....	8
2.1.2. <i>Objectius específics</i> .....	8
2.2. ABAST DEL PROJECTE .....	9
<b>3. ESTAT DE L'ART</b> .....	<b>11</b>
3.1. TIPUS DE REALITAT VIRTUAL.....	14
3.1.1. <i>Realitat no immersiva</i> .....	15
3.1.2. <i>Realitat semi immersiva</i> .....	15
3.1.3. <i>Realitat immersiva</i> .....	16
3.2. REALITAT VIRTUAL APLICADA A LA REHABILITACIÓ.....	17
3.3. REALITAT VIRTUAL IMMERSIVA A LA REHABILITACIÓ .....	20
3.4. REALITAT VIRTUAL NO IMMERSIVA A LA REHABILITACIÓ.....	21
<b>4. METODOLOGIA</b> .....	<b>24</b>
4.1. PLANIFICACIÓ DEL PROJECTE .....	24
4.2. DIAGRAMA DE GANTT .....	26
<b>5. EINES EMPRADES</b> .....	<b>27</b>
5.1. UNITY.....	27
5.1.2. <i>Unity Asset Store</i> .....	28
5.2. MICROSOFT VISUAL STUDIO.....	29
5.3. ADOBE PHOTOSHOP.....	30
<b>6. DISSENY DEL JOC</b> .....	<b>31</b>
6.1. DIAGRAMA DE FLUX .....	31
6.2. ESTIL VISUAL.....	32
6.3. DISSENY DE LES SEQÜÈNCIES.....	33
6.4. MENÚ PRINCIPAL.....	34
.....	36
6.5. JOC 1 .....	36
6.5.1. <i>Scripts Joc 1</i> .....	38
6.6. JOC 2 .....	44
6.6.1. <i>Scripts Joc 2</i> .....	45
6.7. JOC 3 .....	49
6.7.1. <i>Scripts Joc 3</i> .....	52
<b>7. RESULTATS</b> .....	<b>53</b>
7.1. ESTAT DEL JOC.....	53

---

7.2. MODIFICACIONS VERSIÓ D'ENTREGA .....	53
7.2.1. <i>Ratolí</i> .....	53
7.2.2. <i>Música de fons</i> .....	55
7.2.3. <i>Botó menú durant la partida</i> .....	56
7.3. MODIFICACIONS JOC 1 .....	57
7.4. MODIFICACIONS JOC 3 .....	57
7.5. INSTRUCCIONS .....	59
7.6. TRACTAMENTS DE DADES.....	60
<b>8. CONCLUSIONS .....</b>	<b>65</b>
8.1. LÍNIES FUTURES.....	66
<b>9. PRESSUPOST I ANÀLISIS ECONÒMICA .....</b>	<b>69</b>
9.1 DESPESA PERSONAL.....	69
9.2 DESPESES DE SOFTWARE .....	70
9.3 DESPESES DE HARDWARE .....	70
9.4 DESPESES ELÈCTRIQUES .....	71
9.5 DESPESES TOTALS .....	72
<b>10. BIBLIOGRAFIA.....</b>	<b>73</b>
10.1 BIBLIOGRAFIA MEMÒRIA.....	73
10.2 BIBLIOGRAFIA FOROS.....	74
10.3 BIBLIOGRAFIA VÍDEOS YOUTUBE .....	76
<b>11. ANNEXES. ....</b>	<b>81</b>
11.1 CARTA FEEDBACK ADFO .....	81
11.2. DADES EXTRETES DE LES PARTIDES <i>BETA</i> .....	83
11.3 CODIS DEL JOC. ....	89
11.3.1 <i>Codis Joc 1</i> . ....	89
11.3.2 <i>Codis Jocs 2 i 3</i> . ....	96
11.3.3 <i>Codis Joc 4</i> . ....	105
11.3.4 <i>Codis exportació de dades</i> .....	108
11.3.5 <i>Codis varis</i> . ....	111

## Índex de taules i il·lustracions

IL·LUSTRACIÓ 1 LOGOTIP ADFO.....	3
IL·LUSTRACIÓ 2 TIPUS DE PARÈSIA. ....	4
IL·LUSTRACIÓ 3 ESPATLLA CONGELADA. ....	4
IL·LUSTRACIÓ 4 MANEGOT DELS ROTATORIS. ....	5
IL·LUSTRACIÓ 5 DESVIACIÓ CUBITAL I RADIAL. ....	6
IL·LUSTRACIÓ 6 SIMULACIÓ ENTORN RV. ....	7
IL·LUSTRACIÓ 7 MOVIMENTS FLEXIÓ ESPATLLA COLZE.....	8
IL·LUSTRACIÓ 8 MOVIMENTS DE ROTACIÓ. ....	9
IL·LUSTRACIÓ 9 MOVIMENTS CANELL. ....	9
IL·LUSTRACIÓ 10 OCULUS RIFT DEV. KIT.....	11
IL·LUSTRACIÓ 11 OCULUS HALF DOME. ....	12
IL·LUSTRACIÓ 12 TECNOLOGIA DEEP FOCUS. ....	12
IL·LUSTRACIÓ 13 EPIDERMAL VR. ....	13
IL·LUSTRACIÓ 14 CAS PRÀCTIC DEL DISPOSITIU EPIDERMAL VR. ....	14
IL·LUSTRACIÓ 15 NINTENDO SWITCH. ....	15
IL·LUSTRACIÓ 16 EXEMPLE DE SIMULADOR DE VOL. ....	16
IL·LUSTRACIÓ 17 PERIFÈRICS PROPIS DE LA REALITAT IMMERSIVA. ....	16
IL·LUSTRACIÓ 18 NINTENDO WII. ....	17
IL·LUSTRACIÓ 19 DISPOSITIU BALANCE BOARD.....	17
IL·LUSTRACIÓ 20 RÀNQUING DE GE .....	19
IL·LUSTRACIÓ 21 CAS PRÀCTIC DE REHABILITACIÓ MITJANÇANT RV.....	21
IL·LUSTRACIÓ 22 HAND TUTOR DE L'EMPRESA MEDI TOUCH. ....	22
IL·LUSTRACIÓ 23 PACIENT D'ESPECIAL EFFECT .....	23
IL·LUSTRACIÓ 24 PROCÉS DE RECERCA.....	25
IL·LUSTRACIÓ 25 ORGANITZACIÓ DEL PROJECTE.....	26
IL·LUSTRACIÓ 26 INTERFÍCIE UNITY. ....	28
IL·LUSTRACIÓ 27 WEB UNITY ASSET STORE.....	29
IL·LUSTRACIÓ 28 EXEMPLE DE DESCÀRREGA.....	29
IL·LUSTRACIÓ 29 SPRITE EMPRAT AL JOC. ....	30
IL·LUSTRACIÓ 30 ESQUEMA FUNCIONAL DE L'APLICACIÓ. ....	32
IL·LUSTRACIÓ 31 TRANSFORMACIÓ D'IMATGE A LOW POLY. ....	32
IL·LUSTRACIÓ 32 PAISATGE LOW POLY. ....	33
IL·LUSTRACIÓ 33 ORGANITZACIÓ D'ESCENES A UNITY.....	34
IL·LUSTRACIÓ 34 MENÚ PRINCIPAL.....	35
IL·LUSTRACIÓ 35 RESPONSIVE BUTTON.....	35
IL·LUSTRACIÓ 36 NAVEGACIÓ JOCS.....	36
IL·LUSTRACIÓ 37 ESCENA JOC 1.....	37
IL·LUSTRACIÓ 38 EXEMPLES D'ELEMENTS DESCARREGATS.....	37
IL·LUSTRACIÓ 39 SEQÜENCIACIÓ JOC 1. ....	38

---

IL·LUSTRACIÓ 40 SPRITE COMPTE ENRERE. ....	39
IL·LUSTRACIÓ 41 ANIMACIÓ CORRESPONENT AL COMPTE ENRERE.....	39
IL·LUSTRACIÓ 42 ESDEVENIMENT FI D'ANIMACIÓ. ....	39
IL·LUSTRACIÓ 43 OBJECTE GENERADOR DE PILOTES. ....	40
IL·LUSTRACIÓ 44 CONTROLADOR DE L'OBJECTE A INSTANCIAR. ....	40
IL·LUSTRACIÓ 45 CONTROL DE FORÇA ENTRE ALTRES.....	41
IL·LUSTRACIÓ 46 PLA QUE DEST RUEIX LES PILOTES NO DESITJADES. ....	41
IL·LUSTRACIÓ 47 MARCATGE D'OBJECTES A DESTRUIR. ....	41
IL·LUSTRACIÓ 48 COLLIDER BEC. ....	42
IL·LUSTRACIÓ 49 CÀRREGA D'ANIMACIÓ DE DESTRUCCIÓ I PUNTUACIÓ. ....	42
IL·LUSTRACIÓ 50 ANIMACIÓ AL DESTRUIR LA PILOTA. ....	43
IL·LUSTRACIÓ 51 ESCENA DE FI DE JOC.....	44
IL·LUSTRACIÓ 52 CARREGAMENT DE PUNTUACIÓ OBTINGUDA. ....	44
IL·LUSTRACIÓ 53 ESCENA JOC 2.....	44
IL·LUSTRACIÓ 54 EDIFICIS DESCARREGATS. ....	45
IL·LUSTRACIÓ 55 SEQÜENCIACIÓ JOC 2. ....	45
IL·LUSTRACIÓ 56 ETIQUETATGE D'OBJECTES "FUEGO". ....	46
IL·LUSTRACIÓ 57 DESTRUCCIÓ I CÀRREGA D'ANIMACIÓ. ....	46
IL·LUSTRACIÓ 58 CONTROLADOR D'OBJECTES A CARREGAR. ....	47
IL·LUSTRACIÓ 59 FUNCIÓ DE CONTROL DE TEMPS.....	47
IL·LUSTRACIÓ 60 ASSIGNACIÓ DE PUNTS. ....	47
IL·LUSTRACIÓ 61 MIDES DE FOCs EMPRATS. ....	48
IL·LUSTRACIÓ 62 ANIMACIÓ D'EXPLOSIÓ.....	48
IL·LUSTRACIÓ 63 TEXT D'INFORMACIÓ DE PUNTS. ....	48
IL·LUSTRACIÓ 64 ESCENA JOC 3.....	50
IL·LUSTRACIÓ 65 VEHICLES DESCARREGATS DE L'ASSET STORE.....	50
IL·LUSTRACIÓ 66 ASSET FOC PETIT.....	51
IL·LUSTRACIÓ 67 ASSET FOC GRAN. ....	51
IL·LUSTRACIÓ 68 ASSET OBJECTE INVISIBLE. ....	51
IL·LUSTRACIÓ 69 ASSET GALLINA. ....	52
IL·LUSTRACIÓ 70 SEQÜENCIACIÓ JOC 3. ....	52
IL·LUSTRACIÓ 71 SUBSTITUCIÓ DEL MOUSE A UNITY. ....	54
IL·LUSTRACIÓ 72 MOUSE ACTUAL VS. PREDETERMINAT. ....	54
IL·LUSTRACIÓ 73 EXEMPLES DE CANÇONS DE YOUTUBE. ....	55
IL·LUSTRACIÓ 74 COMANDA QUE PERMET SEGUIR REPRODUINT LA MÚSICA. ....	55
IL·LUSTRACIÓ 75 COMPONENT AUDIO SOURCE DE UNITY. ....	56
IL·LUSTRACIÓ 76 BOTÓ MENÚ CANTONADA DRETA INFERIOR. ....	56
IL·LUSTRACIÓ 77 ESTELA QUE INDICA LA DIRECCIÓ I POTÈNCIA. ....	57
IL·LUSTRACIÓ 78 CONTROL DE L'AMPLADA DE L'ESTELA. ....	57
IL·LUSTRACIÓ 79 MENÚ INICIAL AMB JOC 4. ....	58
IL·LUSTRACIÓ 80 JOC 4 .....	58

IL·LUSTRACIÓ 81 ICONA D'INSTRUCCIONS.....	59
IL·LUSTRACIÓ 82 EXEMPLE D'INSTRUCCIONS EN EL JOC.....	60
IL·LUSTRACIÓ 83 DIFERÈNCIES ENTRE EXACTITUD I PRECISIÓ.....	60
IL·LUSTRACIÓ 84 FUNCÍO QUE CARREGA A LA CONSOLA LES COORDENADES.....	61
IL·LUSTRACIÓ 85 MISSATGES IMPRESOS A LA CONSOLA DE UNITY.....	61
IL·LUSTRACIÓ 86 MOSTRA DE REPRESENTACIÓ DE DADES.....	62
IL·LUSTRACIÓ 87 FORMAT D'ARXIU GENERAT.....	62
IL·LUSTRACIÓ 88 COMPROVACIÓ DE COORDENADES.....	63
IL·LUSTRACIÓ 89 RESOLUCIÓ ÒPTIMA PER A REALITZAR LA PARTIDA.....	63
IL·LUSTRACIÓ 90 FULL DE CÀLCUL PER A REPRESENTAR LES DADES.....	64
IL·LUSTRACIÓ 91 MENÚ AJUSTOS DE LA SENSIBILITAT.....	67
IL·LUSTRACIÓ 92 MISSATGE D'ERROR A LA COMPILACIÓ DEL PROJECTE.....	67
IL·LUSTRACIÓ 93 MIGRACIÓ A VERSIÓ ANTERIOR.....	68
TAULA 1 SALARI DEL PERSONAL.....	69
TAULA 2 COST FORMACIÓ DEL PERSONAL.....	69
TAULA 3 DESPESES LLICÈNCIES.....	70
TAULA 4 COST MATERIAL INFORMÀTIC.....	71
TAULA 5 CONSUMS DELS COMPONENTS.....	71
TAULA 6 COST TOTAL DEL PROJECTE.....	72





## **1.Prefaci**

Durant els anys, la medicina ha anat evolucionant a passos agegantats, des del descobriment de la penicil·lina fins als últims tractaments exitosos en pacients de VIH. Tal ha sigut l'evolució, que aquesta s'ha vist desglossada en una infinitat de branques a causa dels continus descobriments científics i tecnològics. Entre elles, la rehabilitació, que s'ha vist molt influenciada en aquesta evolució, i cada cop s'utilitzen recursos més diversos i innovadors per a aconseguir el seu màxim rendiment o possibilitat de tenir experiències telemàtiques.

L'avantatge d'aquests sistemes és que permeten realitzar tractaments no invasius, i a l'hora, transmetre sensacions al pacient a través d'un monitor o altres dispositius. El màxim exponent d'aquesta tecnologia és el que es coneix com a realitat virtual. Tot i ser evident que aquests són mètodes eficaços, el seu ús a gran escala pot trigar temps a desenvolupar-se, doncs al ser una tecnologia nova el marge de millora és elevat .

### **1.1 Origen del treball**

L'origen d'aquest treball era continuar certs Treballs de Final de Grau, basats en el disseny d'un programa de realitat virtual enfocada a la rehabilitació. En aquest cas es disposava d'un nou model de perifèric (Cybershoes), que permetia interactuar amb el joc mitjançant els peus, i així, poder afegir un nou mòdul als jocs existents. Per tant la idea principal era realitzar un joc en el qual, l'usuari o usuària hagués de moure els seus peus per a així fer una rehabilitació de les extremitats inferiors.

A causa de les condicions viscudes pel Covid-19 i no poder accedir al recinte universitari, es va haver d'adoptar una solució alternativa. Tenint en ment la realització d'un treball enfocat a la rehabilitació i seguint les recomanacions del professorat, es va re enfocar el treball a l'elaboració d'una plataforma de realitat virtual no immersiva, o dit d'una altra manera un joc interactiu mitjançant l'ordinador.

Per a la realització d'aquest projecte es va comptar amb la col·laboració de l'Associació Diversitat Funcional d'Osona (ADFO), que va dictar les directrius del projecte conjuntament amb el professorat.

## 1.2. Motivació

La motivació per aquest treball és principalment, l'impacte social que aquests tipus de solucions tenen sobre la societat. Tothom coneix persones (o nosaltres mateixos) que han tingut problemes per la rehabilitació, o que no han obtingut la millora que s'esperava; ja sigui per la dificultat dels exercicis, per la falta de motivació, pel mal assessorament o altres factors.

Donat aquest fet, es va pensar que es podria realitzar un programa que pogués complementar altres mètodes utilitzats actualment, poder obtenir bons resultats a l'hora de tractar pacients i assentar la base per a futures línies de treball. Així doncs si es realitzés un software de rehabilitació ben dissenyat, facilitaria la feina del pacient per rehabilitar-se així com als o les fisioterapeutes.

D'altra banda, a l'hora de crear un videojoc/programa des de zero, permetia treballar amb diversos software i llenguatges de programació que complementés la nostra experiència com a enginyers, doncs els entorns 3D sempre han estat d'interès i es va pensar que potser es resoldria satisfactòriament aquest projecte.

## 1.3. Requisits previs

Per al desenvolupament de qualsevol projecte és essencial saber cercar informació en fonts de confiança. Unity presenta un manual online on els usuaris hi poden acudir en cas de necessitar consultar informació i ha estat de molta utilitat durant el procés, juntament amb el fòrum de preguntes online d'usuaris del software.

Per a poder treballar amb el programa de forma fluida, es va disposar d'un curs de realitat virtual que va ajudar a fer els primers passos dins de la plataforma, com la presentació de l'entorn, elements més emprats o una iniciació al llenguatge de programació C++ , essencial a l'hora de programar els objectes i accions del joc. Novament, a internet s'han trobat moltes de les solucions als nostres problemes, com errors al compilar el codi o buscar solucions concretes a problemes específics.

Quan es va començar a realitzar el videojoc, també va ser de gran ajuda a l'fet de tenir bons coneixements de l'eina Photoshop, per a poder crear tots els decorats i fons necessaris pels videojocs.

Per començar a dissenyar l'aplicació va ser necessari els requisits previs de l'associació ADFO per a poder dissenyar un model que s'adaptés a les seves necessitats.

## **1.4. ADFO**

L'ADFO, veure il·lustració 1, són les sigles de l'Associació de Diversitat Funcional de la comarca d'Osona. És una associació sense ànim de lucre que lluita per a la plena integració social de les persones amb disminució física de la comarca. El seu treball l'ha convertit en l'entitat referent de la comarca i principalment s'enfoquen en potenciar la igualtat d'oportunitats en tots els àmbits, garantir una vida autònoma i inserció laboral, acompanyament personal i familiar durant el procés, orientació en tràmits i fomentar la participació social.



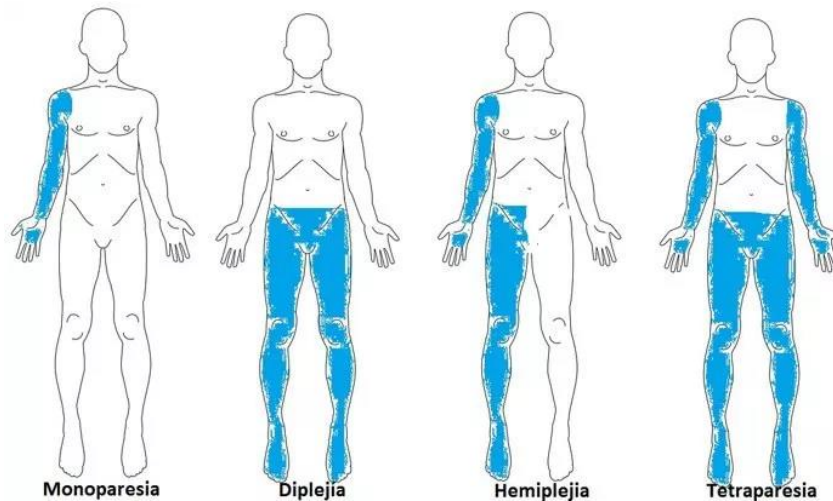
*Il·lustració 1 Logotip ADFO. ( Font: <https://incapacitatsfisiques.wordpress.com/entrevista-a-adfo/> )*

L'associació va compartir tres tipus de lesions que es treballen a les seves instal·lacions i per a les quals, un videojoc podia resultar d'utilitat a l'hora d'exercitar les parts desitjades. Els exercicis proposats van ser:

### 1- Flexió d'espatlla colze :

Aquest tipus d'activitats solen estar relacionades amb la parèsia, paràlisi d'alguna part del cos, ictus o amb l'espatlla congelada (o també coneguda professionalment com a capsulitis retràctil).

Per una banda la mono parèsia és un tipus de parèsia a la qual només una de les extremitats es veu afectada. Això repercuteix a la sensació anormal d'entumiment i/o formigueig en una extremitat del cos. Veure il·lustració 2.



Il·lustració 2 Tipus de parèsia. (Font: <https://www.aspacenavarra.org/noticias/que-es-la-paralisis-cerebral-2> )

D'altra banda tenim el ictus o també conegut com a embòlia. Aquesta, és una malaltia cerebrovascular que afecta els vasos sanguinis que s'encarreguen de subministrar la sang al cervell, fet que provoca la pèrdua d'alguna capacitat associada a parts del cos.

Per últim l'espatlla congelada, il·lustració 3, és una disminució progressiva del moviment passiu de l'espatlla a causa de la retracció i inflamació dels lligaments de l'articulació. Es desconeix la causa d'aquesta malaltia però s'ha estudiat que és més freqüent en el sexe femení.



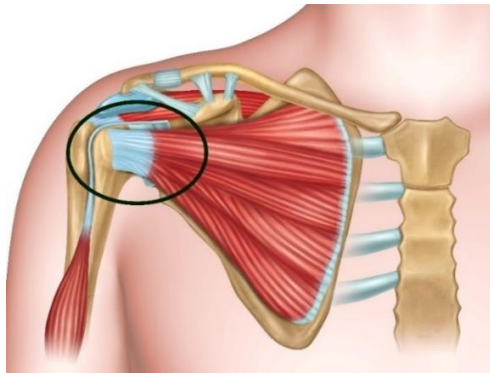
Il·lustració 3 Espatlla congelada. (Font: <https://orthoinfo.aaos.org/es/diseases--conditions/hombro-congelado-frozen-shoulder/> )

## 2- Rotació:

Aquest treball de rehabilitació pot ser molt útil sobretot per treballar rehabilitacions del manegot dels rotatoris, que són les més freqüents de l'espatlla. L'espatlla és una articulació de tipus enartrosis (l'os va encaixat a la cavitat per on es mou). La part superior (cap) de l'os del braç (húmer) va encaixada a la cavitat de l'escàpula (omòplat). Per tant aquesta articulació es pot considerar una ròtula esfèrica que permet que el braç roti en totes direccions.

El manegot dels rotatoris es compon de músculs que uneixen l'escàpula al capdavant de l'húmer, il·lustració 4, i és l'encarregat de reforçar l'espatlla i alhora ajuda a girar la part superior del braç. Aquesta lesió pot ser fruit d'una altre lesió d'espatlla, una degeneració progressiva o un desgast del teixit del tendó.

L'activitat repetitiva per sobre el cap o aixecar objectes pesats durant un temps prolongat poden irritar o fer mal al tendó. També és molt freqüent en l'àmbit dels esports on es realitzen moviments repetitius amb el braç, com llançadors de beisbol, jugadors de tennis i d'altres. La majoria d'aquestes lesions es recuperen amb repòs i rehabilitació, tan sols en casos excepcionals requereix cirurgia.



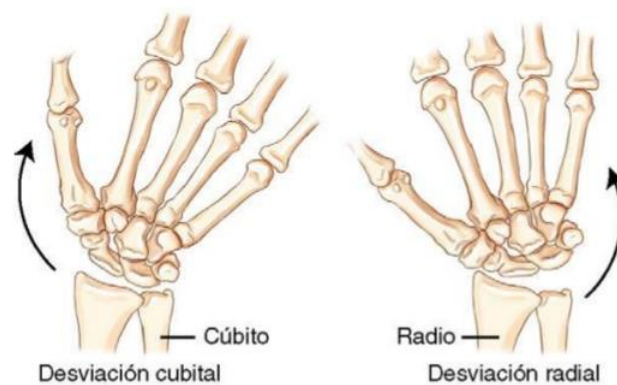
Il·lustració 4 Manegot dels rotatoris. (Font: <https://salud.asepevo.es/profesionales/rehabilitacion/protocolo-de-rehabilitacion-post-reparacion-quirurgica-del-manguito-rotador-de-hombro/> )

### 3- Desviació cubital o radial:

Aquest exercici de rehabilitació és molt útil per treballar la lesió de desviació cubital o bé desviació radial. Aquesta lesió característica del canell es produeix quan la mà perd l'alineament amb l'avantbraç desviant-se cap a la dreta o a l'esquerra, més enllà del rang mitjà de moviment.

Tal com el nom de la lesió indica, la desviació cubital consisteix en un desviament cap al cúbit (desviació cap al cinquè dit) i és la més comuna de les dues. Per altra banda la desviació radial és la provocada pel desviament cap al polze, com es pot veure a la il·lustració 5.

El rang lateral del canell no és gaire ample i, per tant, ràpidament s'arriba a l'extrem d'aquest camp. Un s'ha d'imaginar el canell com un coll d'ampolla per on passen tots els nervis i tendons, que, en desviar la mà se'ls obliga a fer passar entre ossos i músculs, fet que pot generar forces de contacte i fregaments després de moltes repeticions. Les persones poden desenvolupar aquestes desviacions com a resultat d'una inflamació crònica, anomalies congènites estructurals o de problemes amb els lligaments o músculs. És una lesió molt freqüent en pianistes.



*Il·lustració 5 Desviació cubital i radial. (Font: N/A)*

## 2. Introducció

Les lesions són un tema cada com més concurrent en la societat, normalment provocades per sobre entrenament, falta d'un correcte escalfament, exercicis mal realitzats o accidents de qualsevol mena.

Com molts camps de la medicina, la fisioteràpia s'està veient molt afavorida per l'aparició de noves tècniques relacionades amb la tecnologia, com pot ser la crioteràpia, electroteràpia, realitat virtual o ordinadors d'ús personal.

La realitat virtual va aparèixer com a un camp experimental que posteriorment es va transformar en oci i gràcies a les seves propietats, alguns sectors l'estan incorporant a la seva forma de treballar. Com per exemple arquitectura o enginyeria, veure il·lustració 6.



*Il·lustració 6 Simulació entorn RV. (Font: <https://www.ffkr.com/virtual-reality-in-the-design-process/> )*

En el cas de la fisioteràpia proporciona una experiència immersiva i no invasiva que permet enregistrar dades del o la pacient per a fer un correcte seguiment del seu progrés, creació d'activitats d'ús personalitzat o rehabilitació de forma telemàtica si es disposa del hardware.

## 2.1. Objectius del treball

### 2.1.1. Objectiu general

L'objectiu d'aquest projecte és crear un videojoc amè, amb tres nivells enfocats a diferents tipus de lesió neuromuscular, per a poder realitzar tele rehabilitació en entitats que volen incorporar aquest tipus de tecnologia mitjançant l'ús d'un ratolí i ordinador.

### 2.1.2. Objectius específics

Crear 3 jocs funcionals interactius per mirar de rehabilitar cadascuna de les següents necessitats:

**1- Flexió d'espatlla i colze:** Tenint com a referència el pla de la taula, crear un joc que permeti poder moure el ratolí endavant per acabar estenent tot el colze i endarrere, fins acostar-se la mà al cos. Veure il·lustració 7.



*Il·lustració 7 Moviments flexió espatlla colze.*

**2- Rotació:** L'objectiu del joc és que la mà i l'avantbraç ( i en alguns casos tot el braç) vagi des de la línia mitja fins a l'extrem de la taula o a on arribi el pacient, seguint una línia horitzontal en el pla de la taula, com es pot veure a la il·lustració 8.



*Il·lustració 8 Moviments de rotació.*

**3- Desviació radial-cubital canell:** Tenint l'avantbraç recolzat sobre la taula, moure el ratolí cap a la dreta i cap a l'esquerra. Veure il·lustració 9.



*Il·lustració 9 Moviments canell.*

## 2.2. Abast del projecte

El grup de recerca del Laboratori d'Aplicacions Multimèdia (LAM) té antecedents amb projectes similars relacionats amb la rehabilitació. Tots ells presenten una forma de rehabilitació més amena i visualment gratificant mitjançant un joc, on aquest s'adapta al pacient i els professionals poden modificar paràmetres de forma ràpida i senzilla.

Donada la situació d'excepcionalitat i l'aïllament social viscut, aquest joc té com a finalitat assentar les bases per al que en un o dos projectes més endavant, sigui un joc completament funcional, amb diferents nivells de dificultat, menús o música per a que es pugui utilitzar amb pacients reals i obtenir el feedback necessari, que actualment no s'ha pogut obtenir.

Per tant, amb el poc temps que s'ha disposat, aquest projecte pretén crear el codi base de funcionament del joc, amb els diferents nivells segons el tipus d'exercici i la interacció entre nivells de forma adequada. Tots els nivells presentaran un recompte de punts i temps que en un futur es podria millorar mitjançant l'ús de perfils en els que, segons l'usuari o usuària, els jocs es puguin personalitzar d'acord als aspectes a treballar de cada persona.

En aquesta versió *beta* es pretén crear una recollida de dades per a que els i les professionals puguin analitzar l'evolució del pacient a l'hora de jugar al joc. La recollida de dades es durà a terme de forma automàtica i enregistrarà els moviments del ratolí en pantalla.

### 3. Estat de l'art

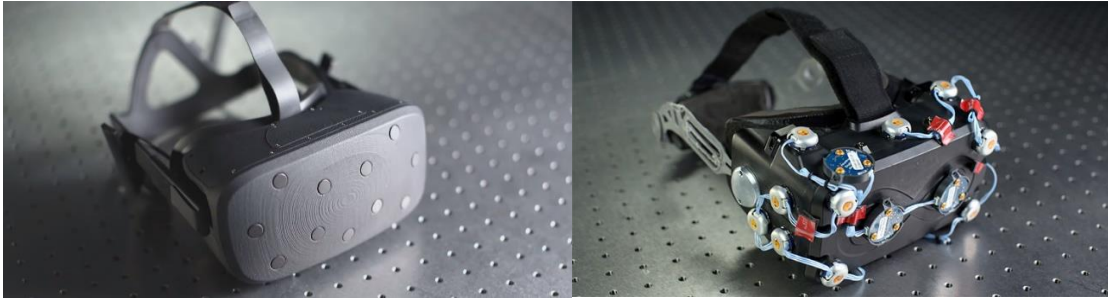
La realitat virtual ha experimentat una gran evolució en els últims anys, tant en format ulleres, *head mounted display*, com en altres dispositius que permeten experimentar sensacions virtuals.



*Il·lustració 10 Oculus Rift Dev. Kit. (Font: <https://www.makeuseof.com/tag/oculus-rift-development-kit-review-and-giveaway/>)*

Al 2012 Oculus va presentar un model d'ulleres per a realitat virtual, veure il·lustració 10. L'empresa, creada per un jove de 20 anys va recaptar 2.4 milions de dòlars gràcies a donacions a la web Kickstarter. Dos anys més tard Facebook va comprar la marca Oculus VR per un preu de 2 bilions de dòlars. Durant els següents anys un gran nombre d'empreses han invertit en desenvolupar nous dispositius, com HTC, Samsung, Sony, Amazon, entre moltes altres.

Actualment el casc de realitat virtual que més novetats presenta és el Half Dome, il·lustració 11, de Facebook que presenta la tecnologia DeepFocus, com es pot veure a la il·lustració 12. Aquest dispositiu enregistra constantment els ulls i permet ajustar la imatge projectada en les pantalles generant així profunditat en la imatge. En resum, permet gaudir d'una experiència més realista mitjançant l'enfocament que realitzaria l'ull humà en condicions normals.



Il·lustració 11 Oculus Half Dome.( Font: <https://computerhoy.com/noticias/zona-gaming/facebook-presenta-half-dome-futuro-oculus-realidad-virtual-79999> )



Il·lustració 12 Tecnologia Deep Focus.(Font : <https://tech.fb.com/introducing-deepfocus/> )

Actualment aquest tipus de tecnologia enfocada a l'oci ha trobat una molt bona acceptació entre el públic consumidor de videojocs i gran part de la investigació s'està enfocant en aquests productes. Tot i això hi ha institucions que porten un pas més enllà la interacció entre màquina i persona, com és el cas del projecte Epidermal VR.

El dispositiu es basa en la idea d'afegir un nou sentit a les experiències virtuals, el tacte. Ha estat desenvolupat per científics de la *Illinois' Northwestern University* i *City University of Hong Kong*. Aquest "teixit" permet adaptar-se als relleus propis del cos humà i en el seu interior disposa d'uns petits vibradors que s'encenen quan es vol simular pressió sobre la pell, com es veu a la il·lustració 13.



Il·lustració 13 Epidermal VR. (Font: <https://www.mccormick.northwestern.edu/news/articles/2019/11/epidermal-vr-gives-technology-a-human-touch-john-rogers.html> )

El dispositiu presenta moltes avantatges respecte als sistemes actuals, i es que aquest funciona sense cables, bateria, permet adaptar-se a qualsevol superfície i és lleuger. La freqüència de vibració d'aquests discs és de 200 cicles per segons, la freqüència en la que la pell humana presenta major sensibilitat.

Actualment aquest prototipus s'està utilitzant per a que persones mutilades puguin rebre un *feedback* de la pressió exercida per la seva pròtesis, com es veu a la il·lustració 14. Quan aquesta arriba a un punt d'elevada pressió, per exemple, suficient per a trencar un got, el pegat del braç vibra per a indicar-ho.



*Il·lustració 14 Cas pràctic del dispositiu Epidermal VR. (Font: <https://www.printedelectronicsworld.com/articles/18831/epidermal-vr-gives-technology-a-human-touch> )*

Aquests avenços tecnològics han trobat camps d'aplicació diferents als que inicialment s'havien dissenyat. Les finalitats d'aquests dispositius, normalment són experimentals, tot i que molts centres de rehabilitació han trobat usos concrets per a tractar dolències quotidianes.

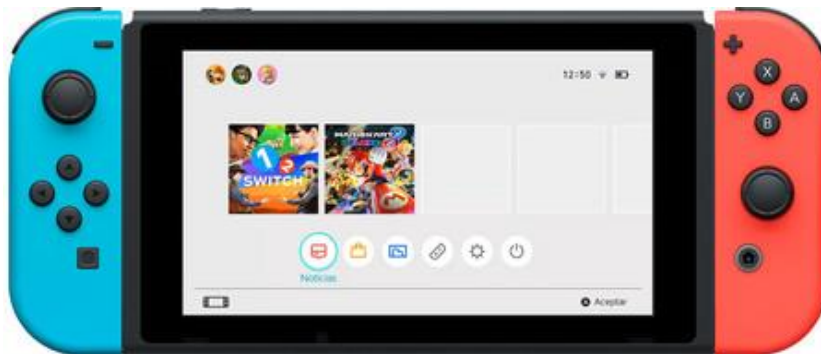
Arrel d'aquest fet les institucions estan creant col·laboracions amb personal de branques tècniques i sanitàries per a que amb els seus coneixements es puguin desenvolupar noves línies de recerca en les que el *feedback* rebut esculpeixi noves aplicacions.

### 3.1. Tipus de realitat virtual

Tot i que el concepte realitat virtual suggereix la utilització de cascos de realitat virtual, no és així. Qualsevol interacció entre persona i màquina es pot considerar realitat virtual, per tant la forma de distingir entre un tipus o un altre, el criteri per a fer-ho és mitjançant el grau d'immersió sensorial que pateix la persona que utilitza els dispositius, i es classifiquen en no immersiva, semi-immersiva o immersiva.

### 3.1.1. Realitat no immersiva

És el tipus de realitat en la que l'usuari o usuària interactua de forma usual, és a dir na experiència d'ús normal d'ordinador o videoconsola, com a la il·lustració 15. La forma d'interacció en aquest cas es limita a un entorn imprès en pantalla amb el que es pot comunicar mitjançant perifèrics com el ratolí, teclat o fins i tot comandaments, en el cas de videojocs. Aquest tipus de realitat porta molts anys a nivell comercial, doncs és el més assequible econòmicament i amb un ús més extens que la resta.



Il·lustració 15 Nintendo Switch. (Font: <https://www.xataka.com/analisis/nintendo-switch-analisis-48-horas-y-el-zelda-breath-of-the-wild-bastan-para-enamorarte-pero-casate-solo-cuando-haya-mas-juegos> )

### 3.1.2. Realitat semi immersiva

Aquest tipus d'experiència és una combinació de realitat no immersiva i immersiva. Es caracteritza per poder fer gaudir a les persones d'escenaris tridimensionals mentre es continua estant connectat a aspectes hàptics del món real com l'espai que t'envolta. Aquest tipus d'experiència permet a l'usuari o usuària poder observar la projecció d'imatges a través de pantalles.

Es sol utilitzar en activitats en les que la persona, recomanablement, pugui interactuar amb objectes reals, com és el cas de pilots de fórmula 1, pilots d'avions o projeccions d'entorns interiors o exteriors, com es veu a la il·lustració 16.

El principal inconvenient d'aquests dispositius és el seu elevat cost d'infraestructura i components. Es solen utilitzar per a casos molt concrets i amb fins lúdics.



Il·lustració 16 Exemple de simulador de vol. (Font : <https://www.aeroexpo.online/es/prod/elite-simulation-solutions/product-180342-36841.html> )

### 3.1.3. Realitat immersiva

Aquesta és el tipus de realitat que normalment s'associa amb realitat virtual, aquella en la que la persona té una sensació d'immersió sensorial completa, ja que li permet explorar i interactuar virtualment amb l'entorn gràcies a l'ús de cascos de realitat virtual, guants o auriculars, com es pot veure a la il·lustració 17. Aquest tipus d'experiència solen ser tan reals que fins i tot es pot arribar a experimentar vertigen o por al jugar a segons quines experiències.

L'ús d'aquests dispositius és encara limitat doncs alguns dispositius són realment cars i la compatibilitat d'aquests amb els jocs actual és bastant limitada, tot i que poc a poc cada cop hi ha més desenvolupament respecte aquests terminals.



Il·lustració 17 Perifèrics propis de la realitat immersiva. (Font : <https://webtoolsdepot.com/vr-and-gaming-facts-you-should-know/> )

### 3.2. Realitat virtual aplicada a la rehabilitació

Com la majoria de camps tecnològics, aquests evolucionen ràpidament i solen afectar a sectors molt diversos de la societat.

A l'any 2006 la companyia Nintendo, que anteriorment va fracassar amb la *Virtual Boy*, va llençar al mercat la seva videoconsola Wii, il·lustració 18, que va suposar una autèntica revolució, doncs fins al moment l'única forma de jugar a videojocs era de forma estàtica, sense necessitat de moure's. Wii va presentar un comandament sense fil que movent-ho amb la mà interactuaves amb la pantalla.



Il·lustració 18 Nintendo Wii.(Font: <https://es.wikipedia.org/wiki/Wii> )

Dos anys més tard la mateixa companyia va presentar una sèrie de videojocs anomenats *Wii fit*. Aquests videojocs permetien exercitar-se mitjançant un nou comandament, *Balance board*, il·lustració 19, sense la necessitat de sortir de casa i amb un tutor virtual per a tal de poder seguir l'activitat.



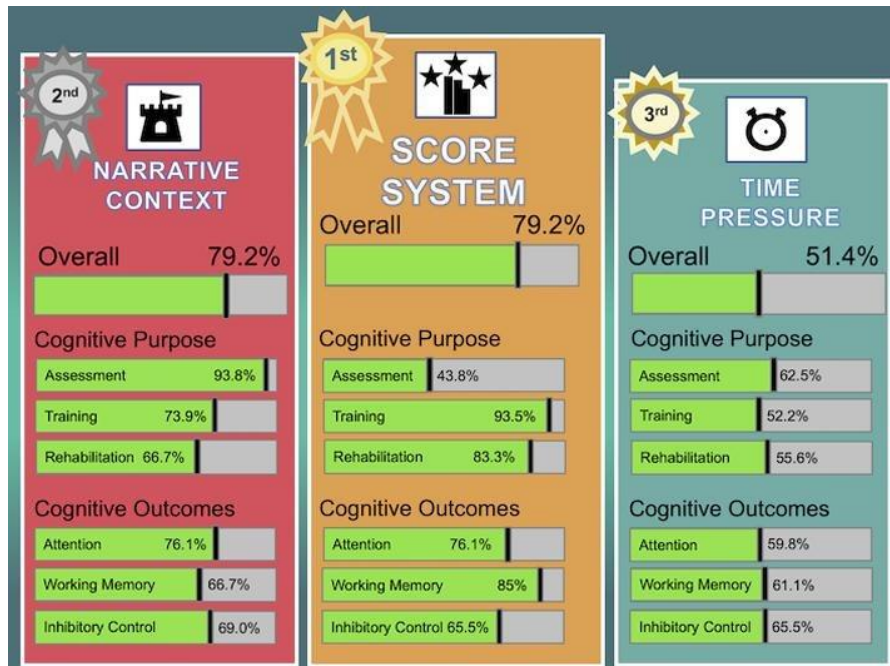
Il·lustració 19 Dispositiu Balance Board.(Font: <https://m.europapress.es/portaltic/videojuegos/noticia-fisioterapeuta-jubilada-demanda-nintendo-plagiar-wii-balance-board-20130902161935.html> )

Aquests dispositius van obrir les portes a un nou camp d'investigació com és el de la interacció entre pacients i dispositius electrònics per a tal de millorar o obtenir informació sobre una dolència en concret. Alguns articles com: -*“Virtual reality gaming in the rehabilitation of the upper extremities post-stroke”* - , publicat al 2016 per Michael Yates, Arpad Kelemen & Cecilia Sik Lanyi, demostren que en persones que han patit un infart cerebral l'ús d'aplicacions de realitat virtual per a tal de poder exercitar les seves extremitats superiors, han estat molt satisfactòries i la tendència de futur és que aquests nous mètodes puguin substituir les teràpies tradicionals.

També es va corroborar a :- *“Virtual Activities of Daily Living for Recovery of Upper Extremity Motor Function”*, publicat al 2018 per Richard J. Adams , Matthew D. Lichter, Allison Ellington, Marga White, Kate Armstead, James T. Patrie, and Paul T. Diamond, a un article on novament les persones que havien patit un infart cerebral presentaven notables millores amb sessions de jocs virtuals 3 cops per setmana.

L'eficàcia d'aquests estudis ha fet que s'investigui sobre les característiques que ha de presentar un joc enfocat a la rehabilitació. A l'article :- *“Game-based interventions for neurophysiological assessment, training and rehabilitation: Which game-elements to use? A systematic review”*, publicat a l'Octubre de 2019 per Filipa Ferreira-Brito, Mónica Fialho, Ana Virgolino, Inês Neves, Ana Cristina Miranda, Nuno Sousa-Santos, Cátia Caneiras, Luís Carriço, Ana Verdelho, Osvaldo Santos, es desenvolupen una sèrie d'aspectes a considerar a l'hora de dissenyar un joc amb finalitats mèdiques o *Game-based interventions (GBI)*.

Els criteris de disseny o *game-elements (GE)* amb més impacte sobre els pacients són l'ús d'un sistema de puntuació i l'existència d'un context narratiu. Com es pot veure a la il·lustració 20, aquest estudi ha tingut en compte diferents per a elaborar el rànquing com poden ser l'atenció presentada a l'hora de jugar, la memòria o el control inhibitori.



Il·lustració 20 Rànquing de GE (Font :

<https://www.sciencedirect.com/science/article/abs/pii/S1532046419302060?via%3Dihub#ab010> )

Les avantatges que presenten en comú tots aquests experiments és el fet de crear una experiència amena per a la persona que hi juga. Molts cops aquestes persones han de realitzar la rehabilitació des de l'hospital o en centres especialitzats i sovint, aquests espais es relacionen amb experiències traumàtiques o monòtones. L'ús d'aquestes tecnologies ajuda a que les persones puguin sentir-se evadides de l'entorn on hi eren i poder involucrar-se més en l'experiència, agilitant així el procés de millora, i fent que el transcurs del procés sigui atractiu.

Els beneficis d'aquestes tecnologies es poden resumir en una notable motivació per part dels pacients, tractaments no invasius, presentació de tasques de rehabilitació agradables i desafiants, beneficis psicològics, basats en el paradigma de psicologia positiva, apoderament i autoeficàcia de les persones que hi juguen, optimisme, esperança de canvi, motivació basada en l'ús de jocs i un augment de participació.

També suposen una millora en la recopilació de dades per a ser tractades per part dels professionals, teràpies de baix cost i una notable validesa ecològica.

### 3.3. Realitat virtual immersiva a la rehabilitació

Aquest tipus d'experiència potser és la més completa i gratificant visualment, doncs la persona es troba completament immersa en el joc.

Els camps d'aplicació d'aquesta tecnologia solen ser aquells on els estímuls aplicats puguin representar una notable millora a la patologia a tractar, per exemple totes les lesions relacionades amb dany neuronal. La plasticitat neuronal del cervell és una propietat intrínseca d'aquest, però s'ha demostrat que el fet de realitzar tasques gratificants amb una finalitat concreta agilitza molt el procés de recuperació.

Algunes associacions especialitzades han aconseguit crear jocs altament personalitzats mitjançant *machine learning*, d'aquesta forma poden obtenir una teràpia altament eficient segons les seves necessitats.

Aquest fet presenta una gran revolució, doncs fins a les hores les persones especialistes eren les encarregades de dissenyar o programar les teràpies, fet que degut a la subjectivitat o la complexitat de la feina, a vegades podien esdevenir en exercicis poc personalitzats i monòtons.

La principal desavantatge que presenta aquest tipus d'aplicació és la despesa econòmica que suposa i l'ús pràcticament limitat a centres especialitzats. Un exemple d'institucions que utilitzen aquesta tecnologia és l'associació *Neuro Rehab VR*, il·lustració 21, situada a Fort Worth, Texas.

Aquest col·lectiu es dediquen a desenvolupar solucions virtuals per a usuaris o usuàries que han sofert algun tipus de dany neuronal i conjuntament amb la supervisió d'especialistes i les seves instal·lacions posen en pràctica les teràpies dissenyades.



*Il·lustració 21 Cas pràctic de rehabilitació mitjançant RV. (Font : <https://www.neurorehabvr.com/> )*

### **3.4. Realitat virtual no immersiva a la rehabilitació**

Algunes dolències o lesions no tan complexes no requereixen d'una experiència completament immersiva. Normalment el tipus de lesió que es tracta amb experiències no immersives són aquelles relacionades amb lesions musculars o de lligaments.

També es poden utilitzar per a tractar aspectes cognitius, com els trastorns relacionats amb pèrdues de memòria. Un dels principals promotors d'aquests camps d'estudi és l'Institut Guttmann, hospital especialitzat en rehabilitació, conjuntament amb NeuroLleida, que han desenvolupat una aplicació per a poder tractar aquests tipus de casos clínics.

El principal avantatge que presenta aquesta forma d'interacció és la senzillesa dels sistemes i la possibilitat de disposar del material per a poder realitzar els exercicis de forma remota mitjançant un ordinador personal.

Un clar exemple d'aquests dispositius és l'empresa MediTouch, companyia especialitzada en disseny de solucions per a hospitals, clíniques i ús domèstic. Un dels seus productes, entre molts altres, és el *Hand Tutor*, il·lustració 22, un guant dissenyat per a exercitar tots aquells aspectes que poden afectar a la mobilitat de la mà i presenta un control total dels dits. Actualment aquest dispositiu s'utilitza a l'Hospital Sant Joan de Déu.



*Il·lustració 22 Hand Tutor de l'empresa MediTouch. (Font :*

*<https://www.institutocharbel.es/departamentos/fisioterapia-neurologica/handtutor/> )*

Aquests dispositius, al no ser especialment complexos d'utilitzar i necessitar unes prestacions tècniques bastant reduïdes, un ordinador, són una opció útil per a usos domèstics o institucions que no requereixen de sistemes molt complexes.

Aquesta modalitat de realitat virtual va ser l'escollida per a la realització del projecte, doncs amb un ordinador de nivell usuari es pot realitzar una aplicació amb jocs concrets per a poder exercitar les extremitats superior.

Tot i que molts dels dispositius estan pensats per a poder jugar amb plataformes dissenyades específicament per a rehabilitació motriu, hi ha moltes persones que pateixen una discriminació a l'hora de poder utilitzar els perifèrics, doncs les seves condicions físiques no els hi permeten gaudir de l'experiència de jugar.

L'empresa anglesa SpecialEffect és l'encarregada d'escurçar l'esclatxa entre persones discapacitades i videojocs gracies al disseny i adaptació de controls específics a les seves necessitats, com es pot veure a la il·lustració 23.



*Il·lustració 23 Pacient d'EspacialEffect (Font: <https://www.specialeffect.org.uk/activities/ceyda> )*

Sovint es sol pensar en la rehabilitació com a una tècnica per a millorar una dolència física, però aquesta també representa la recuperació de drets i dignitats de les persones per a poder realitzar un estil de vida amb normalitat i sense cap barrera que els hi impedeixi fer-ho.

## 4. Metodologia

### 4.1. Planificació del projecte

Per a poder desenvolupar aquest projecte, primer de s'havia d'obtenir coneixements i domini en el programa de Unity 3d. Per tal d'aconseguir-ho es va començar realitzant el curs "Unity VR Fundamentals" on es fa un repàs de tots els fonaments bàsics que cal saber per a començar amb el desenvolupament d'entorns de realitat virtual, com poden ser:

- Treballar estructuradament per directoris.
- Dimensions dels entorns.
- Ajustos del projecte.
- Creació d'*assets*, entorns 360, UI i sons.

Un cop acabat, per a tal de seguir profunditzant en l'aprenentatge es va realitzar un segon curs, "Unity C# Scripting Fundamentals". Que contenia:

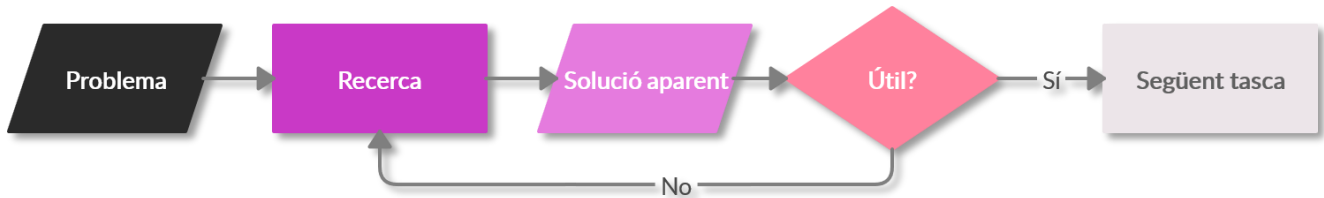
- Fonaments bàsics de *script*.
- Control del flux de codi.
- Manipulació bàsica d'objectes del joc.
- Interaccions d'objectes del joc.

Un cop finalitzats els cursos i disposar d'unes nocions bàsiques sobre el programa a treballar, es va concretar una sèrie de reunions amb els tutors per establir les tasques prèvies del projecte i la idea principal. En aquestes reunions va participar el doctor Stavros Skouras, responsable del conveni entre la UPC i la *start up* Senopi. Segons les directius d'Stavros es va anar cercant informació de la viabilitat del projecte i com poder dur-ho a terme implementant nous perifèrics a jocs existents de la universitat.

La idea inicial del projecte es va veure modificada pel COVID-19, ja que per una banda era impossible accedir a l'escola i poder utilitzar el laboratori, per l'altre banda, no es disposava dels perifèrics necessaris (Cybershoes) per a poder treballar amb ells.

Un cop començat el confinament, es van concretar una sèrie de reunions amb els tutors on es va fer una pluja d'idees sobre com es podia elaborar un projecte conjunt amb l'ADFO que englobés la rehabilitació mitjançant un joc per a ordinador.

D'aquestes reunions es van decidir tres lesions que tracta l'ADFO. Per a poder desenvolupar jocs que fossin adequats per al tipus de lesió es va cercar informació per a comprendre el grau i complexitat d'aquestes. La metodologia aplicada a l'hora de treballar amb el programa en concret s'ha basat en la recerca i solució de problemes concrets, com es pot veure a la il·lustració 24. Tot i haver realitzat els cursos de programació, molts aspectes es desconeixien del programa.



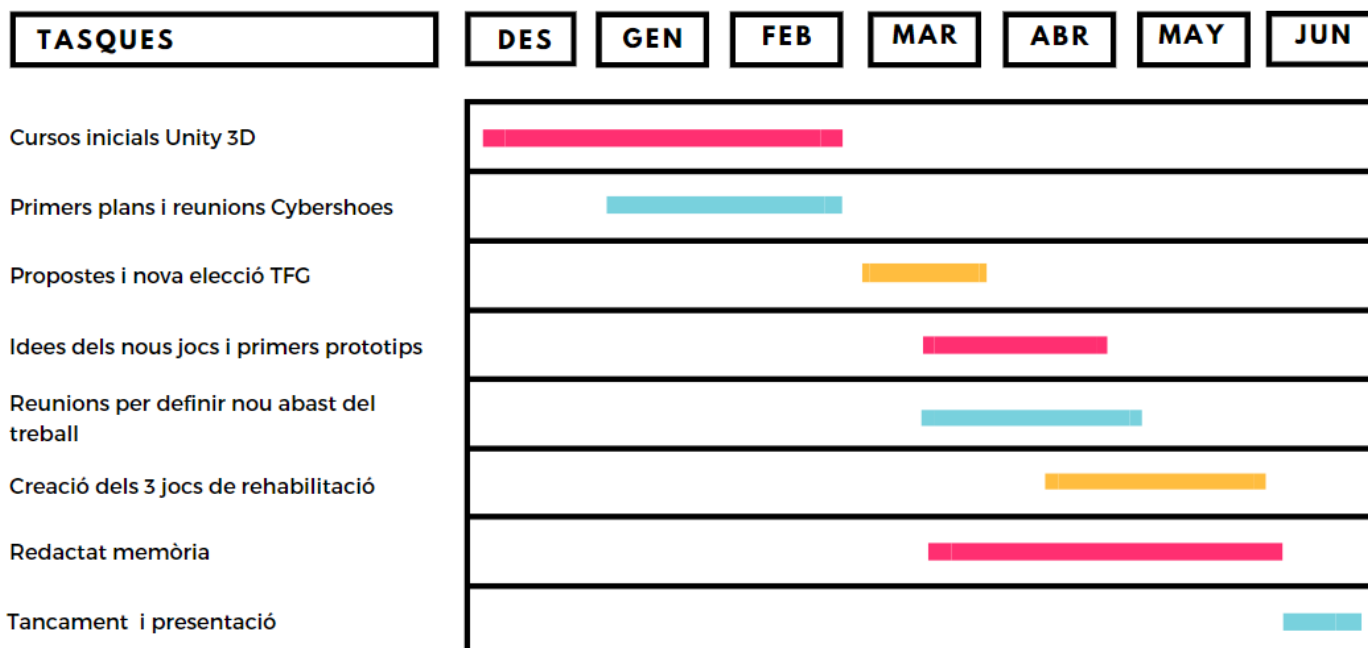
Il·lustració 24 Procés de recerca.

Al ser el primer cop programant, un dels principals problemes que es tenia era el com buscar les solucions als problemes que sorgien. Es tenia una intuïció d'on es volia arribar però no com fer-ho, i al no saber sobre les limitacions i comandes del programa, la recerca d'informació era molt farragosa.

Gran part del temps invertit en la recerca i solució de problemes s'ha degut a la manca de coneixement a l'hora de programar, pràcticament nul. Un cop s'arribava a un objectiu acordat amb el professorat, aquests rebien verificacions i propostes de millora per part dels tutors, que ens feien de nexa amb l'ADFO.

## 4.2. Diagrama de Gantt

Per a tal de poder seguir una planificació del projecte, s'ha realitzat un diagrama de Gantt. Aquest diagrama consisteix en representar mitjançant barres la durada de certes activitats, veure il·lustració 25. Tot i no estar compreses dintre d'aquest gràfic, les reunions de seguiment amb el professorats s'han dut a terme de forma freqüent, com s'ha comentat anteriorment.



*Il·lustració 25 Organització del projecte.*

## 5. Eines emprades

### 5.1. Unity

Unity és un producte dels desenvolupadors Danesos-Americans “Unity Technologies”, i com a tal, és una eina utilitzada per a la creació de videojocs, curtmetratges i aplicacions varies. El seu llançament va ser una revolució ja que va democratitzar la indústria del videojoc.

Aquesta plataforma permet a l'usuari disposar d'una sèrie de motors capaços de poder generar renderitzats d'imatges, aplicacions físiques 2D i 3D, treballar amb so o fer animacions entre moltes altres. La plataforma disposa d'una de les millors documentacions online on consultar dubtes sobre programació, fòrums de dubtes i llibreries que disposen material apte per a descàrrega. És una plataforma àmpliament utilitzada a la indústria de videojocs i disposa de diferents subscripcions segons els beneficis que es puguin generar mitjançant la comercialització del producte creat emprant Unity.

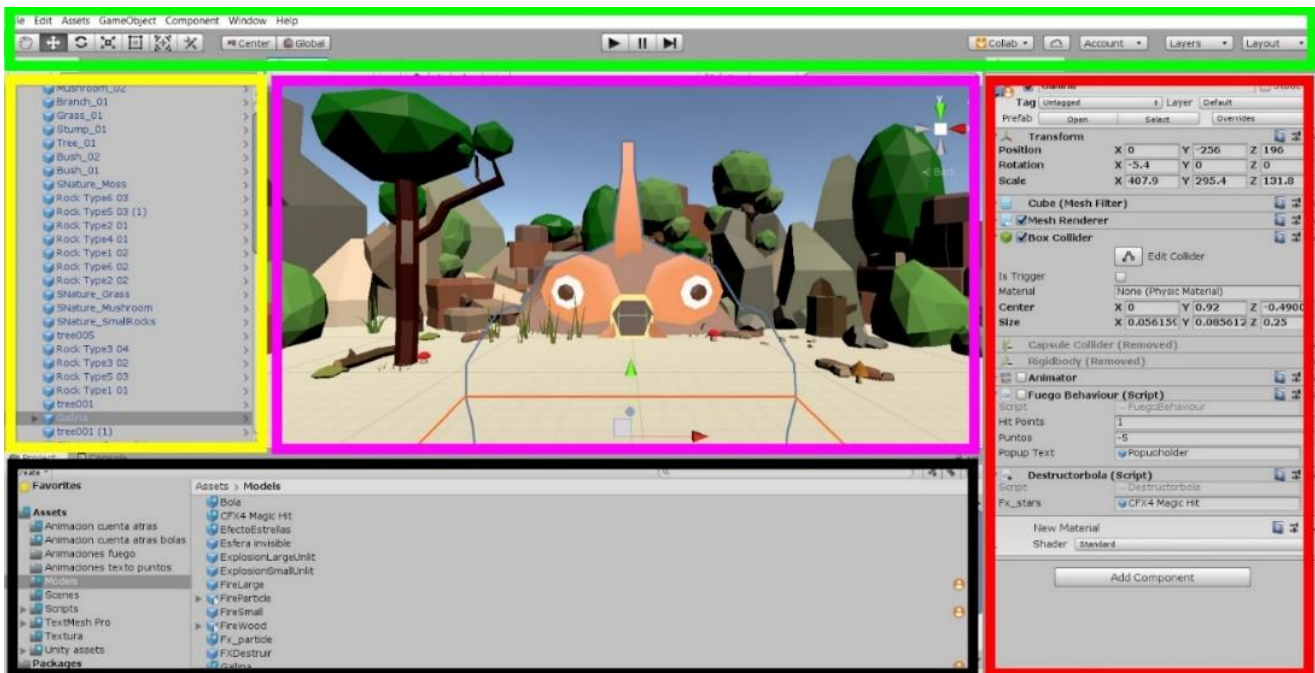
La interfície d'aquest software és molt semblant a la de molts programes i consta de diferents finestres modificables segons les necessitats de l'usuari o usuària. A la part superior de la finestra es troba la barra d'eines del joc (requadre verd figura 26) on podem executar el joc, desplaçar-nos per l'entorn virtual que s'està creant o modificar les mides dels objectes.

La columna esquerra ( requadre groc figura 26) és la jerarquia del joc. Aquí es troben tots els objectes que formen part de l'escena i també es poden crear vincles de dependència entre ells, com pares i fills.

L'escena es troba situada al centre de la pantalla ( requadre lila figura 26) i a ella és on es pot veure els objectes que hem incorporat a la jerarquia i també el joc en funcionament.

La columna de la dreta ( requadre vermell figura 31) és l'inspector. Aquesta columna mostra informació dels objectes seleccionats. Es pot modificar la seva mida, veure si hi ha cap *script* vinculat o si es tracta d'un objecte sòlid-rígid.

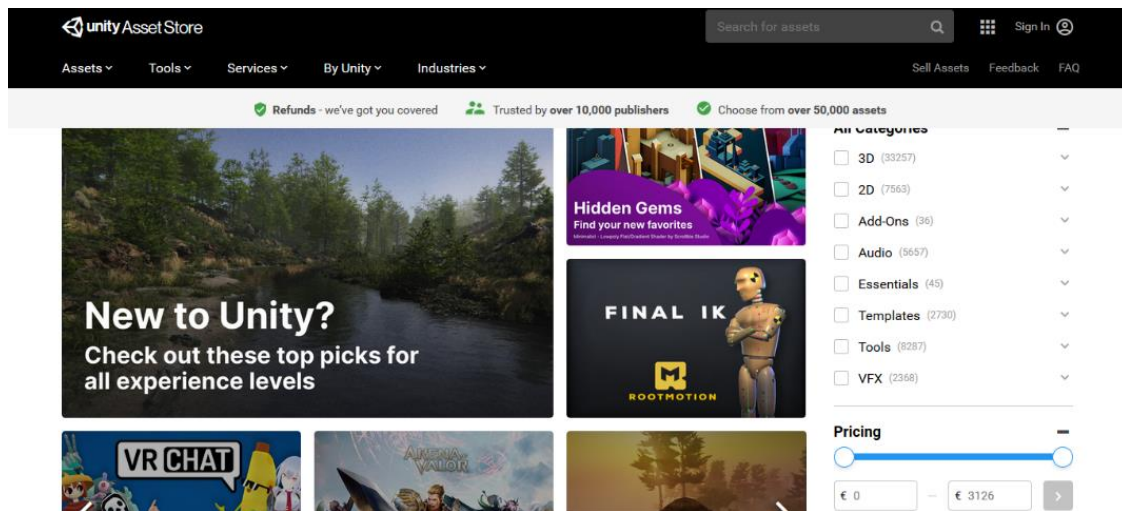
A la barra inferior (requadre negre figura 26) es troba la secció de projecte, on es situen les carpetes que contenen la majoria d'elements emprats del videojoc, tant models 3D com *scripts*. El seu ús, bàsicament, és organitzador doncs es poden crear carpetes i col·locar-hi elements segons tipus. Els elements que s'utilitzen dins d'aquest entorn s'anomenen *assets* i són qualsevol arxiu que pugui ser utilitzat en el videojoc, tant música, models 3D o codi.



Il·lustració 26 Interfície Unity.

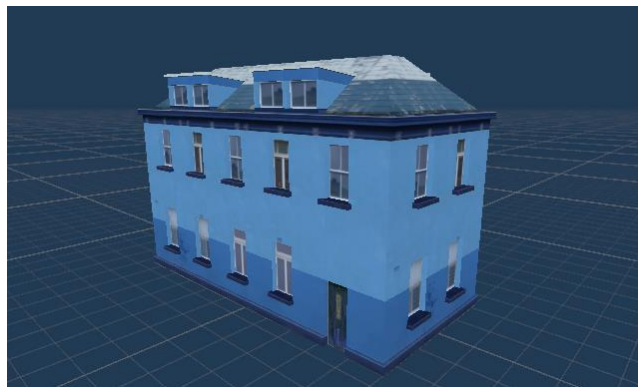
### 5.1.2. Unity Asset Store

Sovint, es pot necessitar un *asset* concret per a poder incorporar-ho al projecte, però no es disposa de les eines o coneixements per a dur-ho a terme. Unity disposa d'una llibreria, veure il·lustració 27, de recursos online on els usuaris poden descarregar elements de forma gratuïta o de pagament, fet que facilita molt la tasca. El domini és molt senzill d'utilitzar i es poden cercar elements segons tipus, versions compatibles de software, valoració d'usuaris i usaries, etc.



*Il·lustració 27 Web Unity Asset Store.*

La gran majoria d'elements del treball s'han obtingut de forma gratuïta d'aquest repositori, excepte tots els elements d'imatges o sons, com es pot veure a la il·lustració 28.



*Il·lustració 28 Exemple de descàrrega.*

## 5.2. Microsoft Visual Studio

Aquest software és un entorn de desenvolupament integrat (*IDE*) compatible amb diferents llenguatges de programació com C++, C#, Visual Basic, entre altres.

Hi ha varis programes que realitzen les mateixes funcions, com MonoDevelop, però en aquest cas s'ha treballat amb Microsoft Visual Studio. Unity per si sol és un motor funcional, pot realitzar tasques sense dependre de cap altre programa, ja que segons el resultat que es vulgui obtenir no fa falta.

No és així quan es vol certa interacció entre objectes i unes ordres de joc, com per exemple assignar a un botó un canvi d'escena o pantalla. En aquests casos disposar d'un programa per a poder fer els codis que determinaran el funcionament dels elements és essencial.

El funcionament de la interfície és bastant senzill, doncs es resumeix en escriure línies de codi amb funcions o variables específiques del llenguatge de programació, doncs segons el tipus de llenguatge, existeixen certes variacions entre ells.

### 5.3. Adobe Photoshop

Com s'ha esmentat, els *assets* poden ser qualsevol tipus d'arxiu que puguin tenir una funció en el joc. Per a generar alguns dels elements del videojoc, com els títols o el compte enrere, s'ha utilitzat Adobe Photoshop per a poder crear imatges d'acord a les necessitats i en format PNG doncs així poden aplicar-se mantenint un fons d'imatge transparent o conservar una bona resolució al escalar els arxius.

El funcionament d'aquest software es basa en el treball mitjançant capes de treball, així doncs la posició d'aquestes modifica el resultat final. Unity, a part de treballar amb models en tres dimensions també permet carregar models en dues dimensions, *sprites*, veure il·lustració 29. L'avantatge d'aquests models és que no requereixen de gran capacitat de processament i bàsicament amb qualsevol imatge d'un cercador, i treballant-la posteriorment amb Photoshop, es pot afegir al joc. Aquest programa també ha estat de gran utilitat per l'elaboració de la memòria, ja que al treballar amb moltes imatges sovint s'ha necessitat d'un software d'edició.



Il·lustració 29 Sprite emprat al joc.

## 6. Disseny del joc

El disseny d'aquesta aplicació s'ha dut a terme tenint en compte algunes característiques destacables, com són el recolliment de dades i activitats centrades en el control del moviment.

El sistema pretén informar dels avenços en la recuperació del moviment entre sessions successives. Aquesta informació pot ser útil per al terapeuta per avaluar i influir en el progrés dels moviments i és una característica cercada en alguns sistemes de rehabilitació VR. [1]

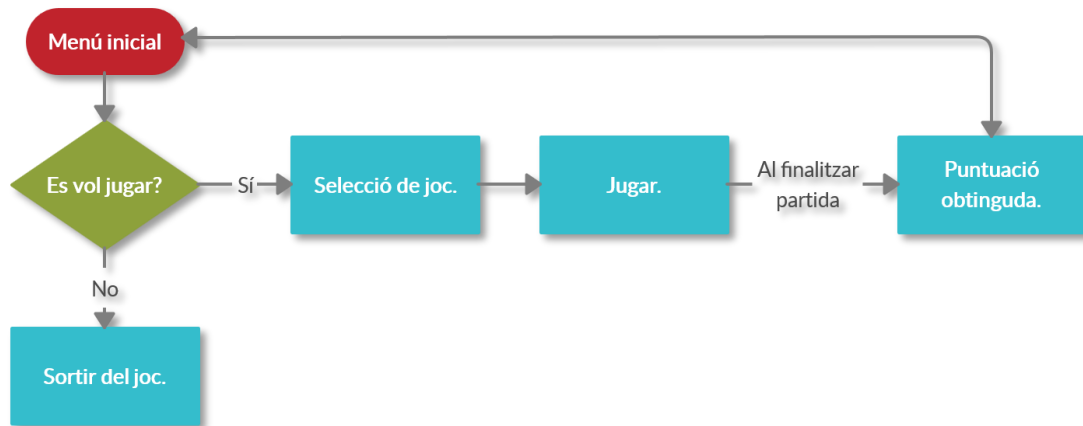
També té una gran adaptabilitat en la presentació dels moviments com a reproducció del moviment produït al món real pel pacient. Aquesta característica permet personalitzar l'experiència i pot ajudar a millorar la funció terapèutica del sistema proporcionant activitats de rehabilitació graduades que es poden individualitzar. [34]

[1] Lo et al. "Prospective clinical study of rehabilitation interventions with multisensory interactive training in patients with cerebral infarction: study protocol for a randomised controlled trial". *Trials*, vol. 18, no 173, 2017.

[2] M. Yates, A. Kelemen and C. Sik Lanyi, "Virtual reality gaming in the rehabilitation of the upper extremities post-stroke". *Brain Injury*, vol 30, no7, pp. 855-863, 2016.

### 6.1. Diagrama de flux

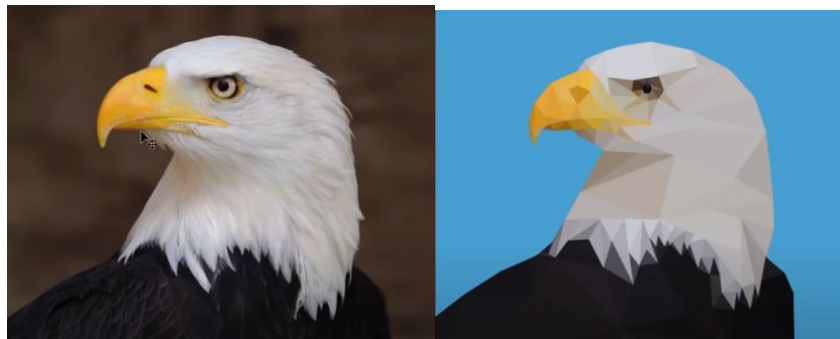
L'aplicació presenta un comportament seqüencial i incorpora un menú inicial on es pot escollir el tipus de joc que es vol realitzar segons el nivell de lesió o sortir de la plataforma. Un cop es realitza el joc i un es finalitza la partida apareix un menú de fi de joc on la puntuació aconseguida per la persona es mostra en pantalla. Després de finalitzar la partida es torna al menú principal i allà es decideix si es vol tornar a jugar o tancar el joc. El funcionament, simplificat es pot veure a la il·lustració 30.



Il·lustració 30 Esquema funcional de l'aplicació.

## 6.2. Estil visual

Per a l'ambientació dels videojocs s'ha seguit un estil molt conegut dins de la indústria; el *Low Poly* o *polygonal*, veure il·lustració 31. Aquesta tècnica tracta d'aconseguir objectes tridimensionals amb pocs polígons, acabats visualment atractius i senzills.



Il·lustració 31 Transformació d'imatge a Low Poly. (Font: N/A)

S'ha escollit aquest mètode visual ja que evita distreure a la persona que està jugant. Cal recordar que la majoria d'usuaris serà gent gran, normalment pateixen dificultat de visió, per tant com menys carregat sigui l'entorn més atenció i millor resultats es poden obtenir en l'objectiu del joc.

D'altra banda com els objectes estan formats per poca densitat de polígons, permeten poder utilitzar el joc en ordinadors que no disposen de grans especificacions tècniques, com a la il·lustració 32. Aquest fet també ha estat un avantatge a l'hora de treballar, doncs el disseny s'ha dut a terme mitjançant els nostres ordinadors d'ús personal i aquests no disposen de grans especificacions tècniques.

A la indústria, els principals programes per a la realització del *Low Poly* són Cinema4D i Blender. Com no tenim coneixements d'aquests programes en concret s'ha decidit obtenir els models que utilitzarem en el joc a través del Asset Store.



Il·lustració 32 Paisatge Low Poly. (Font : <https://www.invergestudios.com/blog/el-arte-low-poly/> )

### 6.3. Disseny de les seqüències

Per a tal que el joc pugui seguir un ordre, s'ha construït mitjançant diferents parts anomenades escenes, veure il·lustració 33. Cada escena presenta la seva jerarquia i els *assets* que s'hi carreguin són d'ús exclusiu per a aquesta vista. No succeeix el mateix amb els scripts, doncs es poden utilitzar en qualsevol nivell però tota modificació que es realitzi sobre ell afectarà a totes les altres escenes que també depenen d'aquell codi.

Per a evitar problemes, algunes vistes són les mateixes però els seus codis són diferents, com les de fi de joc. Altres escenes són diferents però presenten el mateix codi, per exemple, la forma de comptar punts.

Un cop s'han definit tots els nivells i els codis són independents o estan ben definits, s'ha de construir el projecte mitjançant els ajustos del software. Per a fer-ho s'han desat tots els nivells individualment i carregat en l'ordre lògic de funcionament.

Scenes In Build	
<input checked="" type="checkbox"/> Scenes/Mainmenu	0
<input checked="" type="checkbox"/> Scenes/Tragabolas	1
<input checked="" type="checkbox"/> Scenes/Fire fighter X	2
<input checked="" type="checkbox"/> Scenes/Fire fighter	3
<input checked="" type="checkbox"/> Scenes/Findeljuego	4
<input checked="" type="checkbox"/> Scenes/Finjuegobolas	5

*Il·lustració 33 Organització d'escenes a Unity.*

Per tant, la primera escena en carregar-se és el menú inicial, i el seu índex es 0, a partir d'aquí tots els altres nivells o vistes es poden carregar en l'ordre que es desitgi. La forma de fer-ho és mitjançant l'ordre "LoadScene", que carrega l'escena que se li indiqui.

Per exemple, al final de cada nivell s'hauria de carregar la pantalla de fi de joc, per a fer ho podem:

```
SceneManager.LoadScene("Findeljuego");
```

O bé,

```
SceneManager.LoadScene("4");
```

doncs tant el nom de l'escena com l'índex corresponen a l'ordre establert. Un cop s'ha construït el projecte, aquest funciona com a un sistema tancat i es pot generar un document executable o *standalone* que permet jugar sense necessitat de softwares addicionals.

## 6.4. Menú principal

La majoria de softwares disposen d'un menú principal. Aquest menú és la primera escena que es veu al executar el programa i té principalment dues funcions. La primera d'elles és la funcionalitat. El menú ha de ser senzill i concís, així doncs els usuaris o usuàries no poden confondre ni mal interpretar cap element de la pantalla i entenen quina es la funció de cada botó o acció.

La segona, especialment a la indústria dels videojocs, és tenir cert component artístic. Es sol escollir un leitmotiv que es trobi present durant tot el desenvolupament del joc o una estètica acord a la temàtica global. Aquest aspecte atorga una sensació de context i aporta personalitat al joc. En el cas d'aquest, s'ha decidit escollir un animal molt conegut i de fàcil reconeixement, una gallina, veure il·lustració 34.



Il·lustració 34 Menú principal

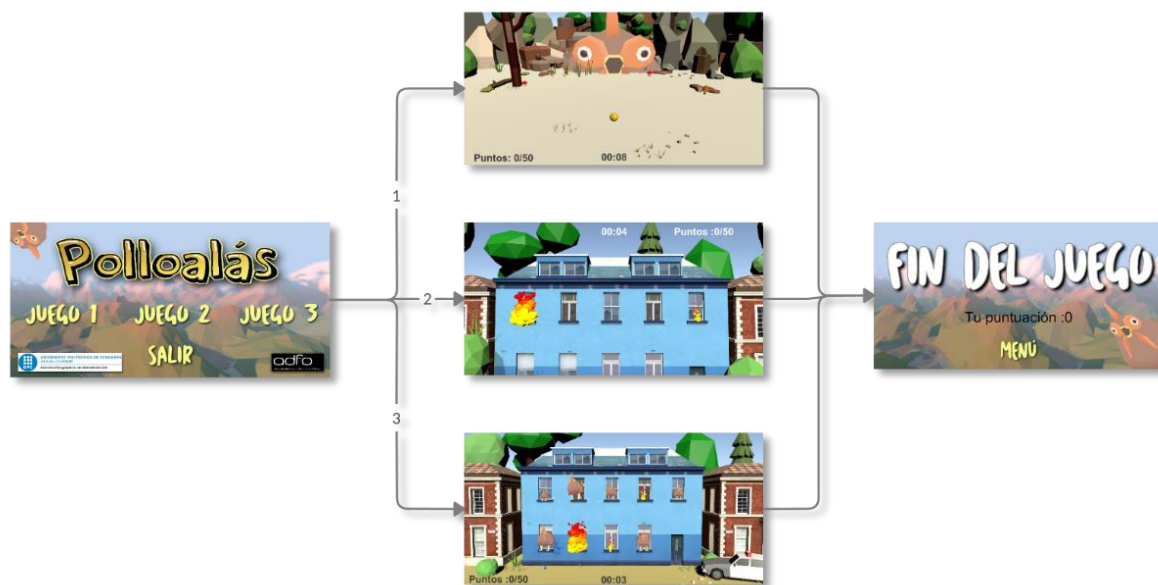
Aquesta escena consta de diferents elements entre els quals es poden trobar botons, imatges de recolzament (com el fons de pantalla o el títol) i els *scripts* de cada botó. Tots ells s'han creat mitjançant el programa d'edició gràfica Photoshop.

Els botons són interactius, veure il·lustració 35, així la persona que hi col·loca el cursor a sobre pot saber si hi està a sobre o no. Aquesta interactivitat s'ha aconseguit modificant el color de les lletres un cop el *mouse* s'hi troba a sobre, també anomenat *responsive button*.



Il·lustració 35 Responsive button.

Tots els botons estan connectats amb un *script* general des d'on es controla les escenes que s'han de carregar un cop es prem el botó, gràcies a la funció "On Click ()" de l'objecte en sí. Per tant, cada botó carrega la seva escena corresponent i el botó de sortida tanca l'aplicació, com es pot veure a la il·lustració 36.



*Il·lustració 36 Navegació jocs.*

## 6.5. Joc 1

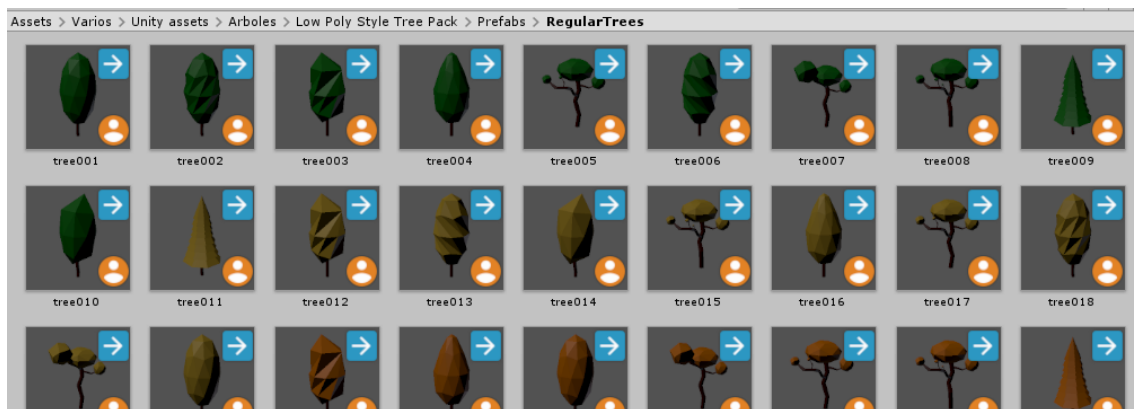
El joc 1 és un joc basat en el funcionament del conegut Angry Birds. Aquest consisteix en generar una pilota mitjançant la barra espaciadora, es clica i s'arrossega el ratolí per a generar força, i un cop el ratolí no es mou més, es deixa anar per a impactar a la gallina gegant de l'escena i aconseguir punts, si la pilota no impacta a la boca de la gallina, es destrueix per a tal d'evitar l'acumulació de pilotes en escena.

La distribució d'aquest nivell està pensada per a que la persona que hi juga només pugui realitzar el moviment del ratolí en un sentit, veure il·lustració 37, doncs l'exercici que es vol realitzar presenta aquestes característiques. Per a fer-ho s'ha generat un pla que simula el terra i la disposició dels elements al voltant, juntament amb la perspectiva de la càmera, genera una sensació de convergència cap a la gallina.



*Il·lustració 37 Escena Joc 1.*

Els elements per a generar l'escena s'han descarregat de l'Asset Store, veure il·lustració 38, i tots ells han estat de temàtica natural, com l'herba, pedres, etc. Aquests elements no afecten al funcionament del joc, tan sols serveixen com a decoració.

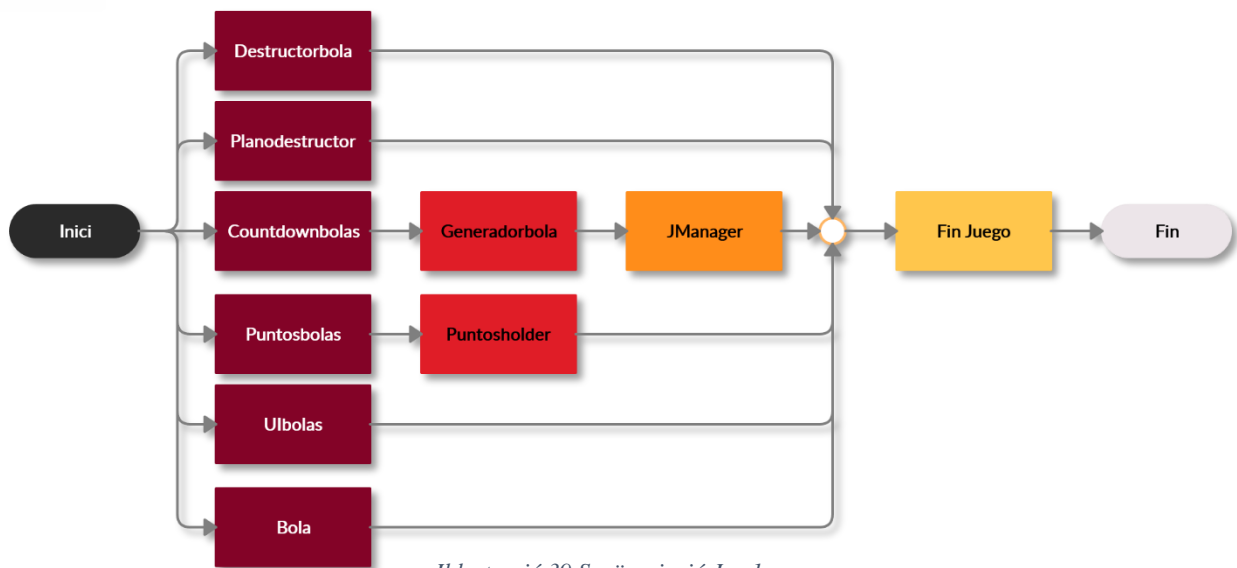


*Il·lustració 38 Exemples d'elements descarregats.*

Altres elements com la pilota, s'han generat mitjançant les funcions de generació d'objectes de Unity, ja que permet generar diferents volums com esferes, cubs o càpsules entre d'altres.

### 6.5.1. Scripts Joc 1

L'estructura general d'aquest joc és una seqüenciació de codis que estan connectats entre sí. Cada un s'encarrega d'un aspecte o particularitat concreta del joc i tots ells finalitzen quan el joc arriba al seu final. Com es pot veure a la il·lustració 39, un cop comença el joc automàticament s'inicien tots els codis que romanen invariables durant la partida i alguns d'ells amb el pas del temps s'enllacen amb nous codis. Un exemple seria l'inici d'un compte enrere que determina l'inici de la partida, doncs fins que aquest *timer* no finalitza no es pot jugar.



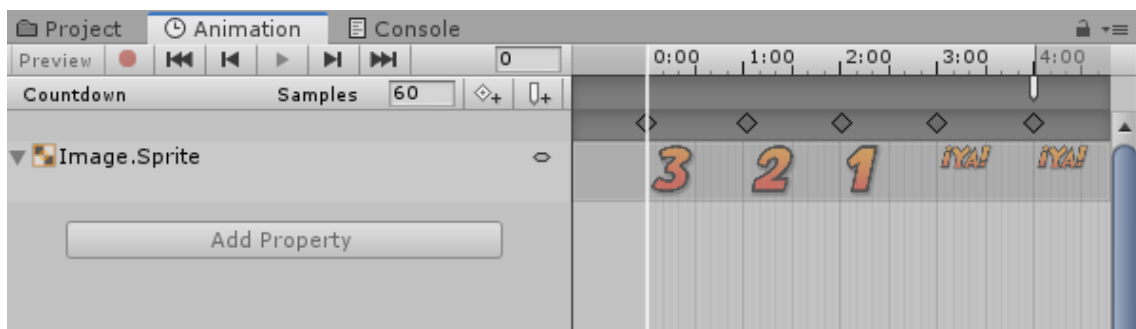
Per a cada joc es descriurà la funció de cada codi dins de la partida, així la seva relació amb els altres resulta més fàcil. Als annexes es poden trobar els codis classificats per jocs. Per tant, els noms dels codis i les seves accions són:

-**Countdownbolas**: Serveix per a controlar l'inici del joc. Un cop l'animació de compte enrere finalitza, aquest codi li comunica al "JManager" que pot començar a comptar el temps de joc. Les imatges emprades per al compte enrere es poden veure a la il·lustració 40.

**3 2 1 ¡YA!**

*Il·lustració 40 Sprite compte enrere.*

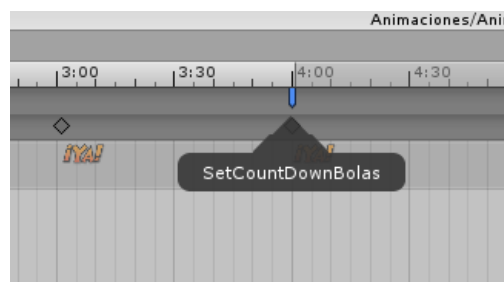
Per a que aquest codi funcioni correctament s'ha de crear una animació amb les imatges de compte enrere, com es pot veure a la il·lustració 41.



*Il·lustració 41 Animació corresponent al compte enrere.*

A l'instant de 3 segons, aquest hauria de representar el final del compte enrere, però al programa no li dona temps a mostrar la imatge i s'ha de duplicar per a que res pugui visualitzar durant un segon. Aquest fet no afecta al funcionament real del compte enrere, i per a que existeixi una interacció entre l'animació i el codi, s'ha creat un esdeveniment a l'instant de temps 4 segons, veure il·lustració 42.

Aquest esdeveniment (SetCountDownBolas) `public void SetCountDownBolas()` és amb el que es comunica el codi per a saber si aquesta animació o clip de vídeo ja ha arribat al seu fi i així poder carregar els següents codis o accions, en aquest cas, el generador de bola.



*Il·lustració 42 Esdeveniment fi d'animació.*

-**JManager**: És el controlador general i permet determinar el temps de partida, així com el seu inici i instanciar una funció que resta temps. La funció de temps funciona mitjançant un bucle `while` que va comprovant si el temps de partida és més gran que 0 `while (playtime > 0)`, i quan el temps s'esgota carrega la pantalla de fi de joc.

-**Generadorbola**: Per a tal de poder fer aparèixer la pilota s'ha creat un objecte invisible, així no es veu en escena, veure il·lustració 43, que ens determina la posició concreta on s'ha de generar la bola, i cada cop que es prem la barra espaiadora el codi la genera.



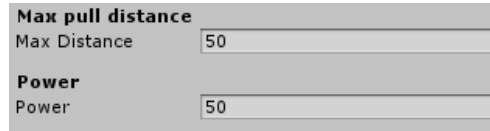
*Il·lustració 43 Objecte generador de pilotes.*

Per a generar la pilota s'ha utilitzat la comanda `Instantiate(bola);` que genera una còpia d'un objecte per a carregar-ho en pantalla. L'assignació de l'objecte es realitza mitjançant l'inspector, on s'arrossega l'objecte a generar dins del requadre anomenat "Bola", com es pot veure a la il·lustració 44.



*Il·lustració 44 Controlador de l'objecte a instanciar.*

-**Bola**: Controla la distància màxima d'estirada de la pilota, la potència amb la que sortirà disparada o el fregament de l'objecte amb el terra, veure il·lustració 45. Tots els valors es poden modificar des de l'inspector del joc. Aquest codi ha estat extret de llibreria donada la seva complexitat.



*Il·lustració 45 Control de força entre altres.*

-**Planodestructor**: Aquest codi va adjuntat a un pla invisible situat just darrere del bec de la gallina, il·lustració 46, per a que cada cop que detecti una col·lisió entre el pla i la pilota, aquesta es destrueixi.

La destrucció es realitza mitjançant un marcatge, és a dir, tots els objectes amb l'etiqueta o *tag* corresponent, es destrueixen. En aquest cas l'etiqueta és "Bola", com es pot veure il·lustració 46. Per a que es detecti l'impacte, tant la pilota com aquest pla contenen un component anomenat *collider* i *rigidbody* que li confereixen de propietats "físiques" a l'objecte.



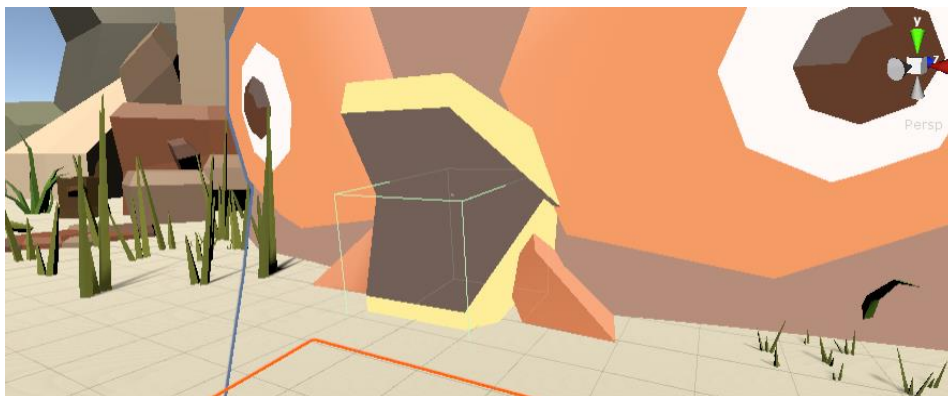
*Il·lustració 46 Pla que destrueix les pilotes no desitjades.*

```
if(collision.gameObject.tag == "Bola")
{
    Destroy(collision.gameObject);
}
```

*Il·lustració 47 Marcatge d'objectes a destruir.*

-**Destructorbola**: Per a poder diferenciar la destrucció de la bola també s'ha creat un altre codi que destrueix la pilota quan impacta amb el *collider* situat a la boca de la gallina, il·lustració 48, així podem controlar la destrucció concreta de la pilota i alhora poder sumar punts al joc.

Dins del codi s'ha definit una variable anomenada "scorevalue" que és l'encarregada d'afegir 10 punts cada cop que es destrueix la pilota. Quan es detecta la col·lisió s'instancia amb l'ordre *Instantiate*, veure il·lustració 49, un objecte animat que representa la destrucció, il·lustració 50, així, juntament amb l'adquisició de punts l'usuari o usuària sap que l'objecte s'ha destruït.



*Il·lustració 48 Collider bec.*

```
if (collision.gameObject.tag == "Bola")
{
    Instantiate(fx_stars);
    Puntosbolas.scorevalue += 10;
    Destroy(collision.gameObject);
}
```

*Il·lustració 49 Càrrega d'animació de destrucció i puntuació.*



*Il·lustració 50 Animació al destruir la pilota.*

-**Puntosbolas:** Aquest script és l'encarregat de definir un objectiu a assolir en quant a puntuació i també actualitza el valor de la funció "scorevalue" en pantalla. Per a que aquest codi es pugui veure en funcionament s'ha vinculat a un text en *on screen* situat a la cantonada inferior esquerra.

-**Uibolas:** Per a poder representar el temps que falta per a completar la partida es necessita aquest codi ja que el representa en pantalla, al igual que "Puntosbolas". S'ha connectat amb un text visible a la part inferior central de la pantalla.

-**Puntosholder:** Actua com a memòria, així doncs un cop s'arriba a la pantalla de fi de joc, es crida aquest codi per a carregar la puntuació obtinguda.

-**Finjuegobola:** Un cop es completa el nivell, es carrega la pantalla de fi de joc, veure il·lustració 51, mitjançant el "JManager" en aquesta ha d'aparèixer la puntuació. Es defineix un codi en el qual es recorda la puntuació obtinguda del joc mitjançant "Puntosholder", veure il·lustració 52, i es mostra en pantalla.



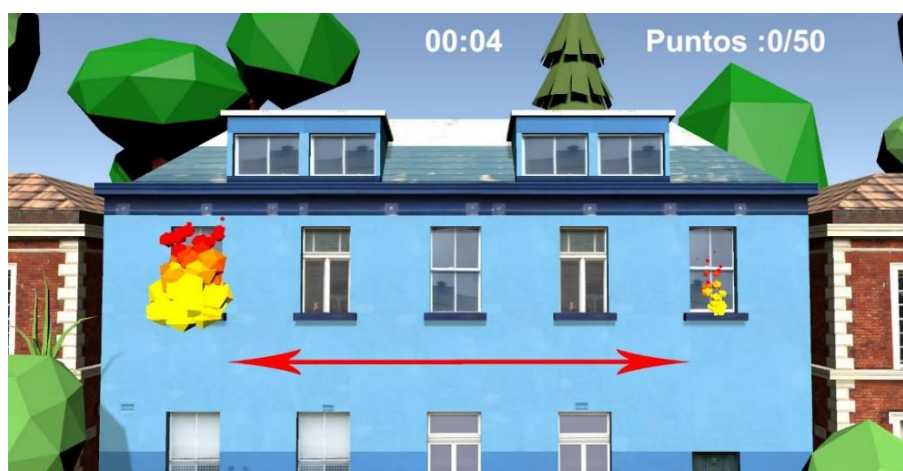
*Il·lustració 51 Escena de fi de joc.*

```
{
  puntosText.text = "Tu puntuación : " + Puntosholder.scorevalue;
}
```

*Il·lustració 52 Carregament de puntuació obtinguda.*

## 6.6. Joc 2

El funcionament del segon joc es basa en una escena amb dos punts on es generen aleatòriament focs, i l'objectiu és extingir-los clicant amb el ratolí a sobre d'ells, veure il·lustració 53. Per a poder generar aparicions aleatòries s'ha utilitzat la funció *spawn* que permet definir la quantitat d'objectes que es volen invocar (*Length*) i el temps d'aparició (*Range*), mitjançant un objecte invisible que actua de generador, com en el Joc 1.



*Il·lustració 53 Escena Joc 2.*

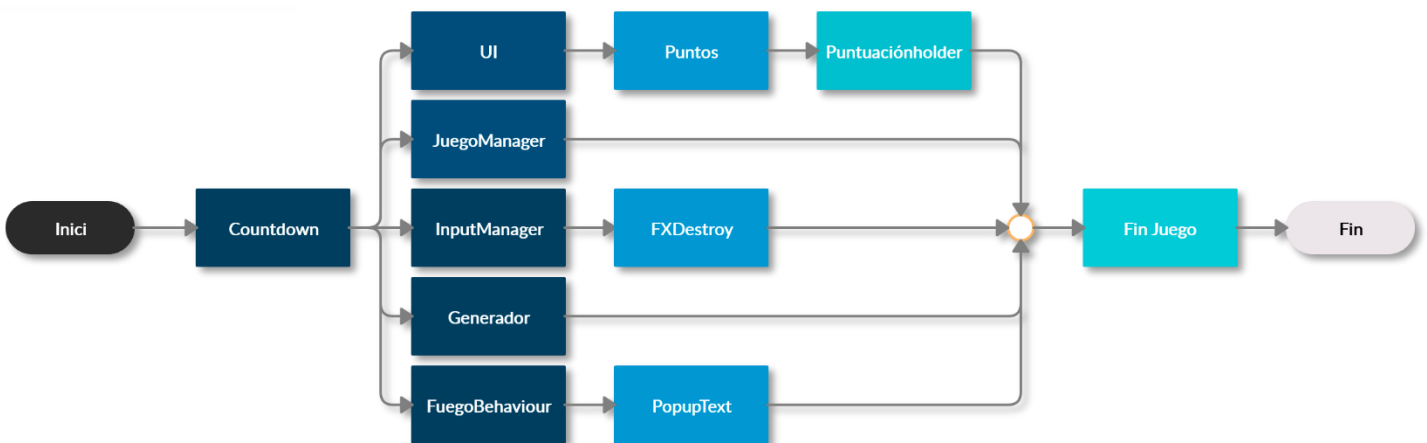
L'escena del joc ha estat composta mitjançant edificis obtinguts a l'Asset Store, veure il·lustració 54, i el conjunt d'elements naturals utilitzats al primer joc. La col·locació dels elements i càmera en aquest joc estan pensats per a que la persona que hi juga només s'hagi de moure en l'eix horitzontal de coordenades, fent així que només es realitzi moviment en el sentit que pertoca per a l'exercici proposat pels i les fisioterapeutes.



Il·lustració 54 Edificis descarregats.

### 6.6.1. Scripts Joc 2

La seqüenciació dels codis d'aquest videojoc, il·lustració 55, comparteix el mateix principi que el Joc 1 i el Joc 3. Al iniciar-se la partida també s'executen els codis segons l'ordre corresponent.



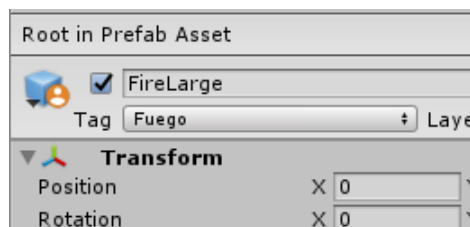
Il·lustració 55 Seqüenciació Joc 2.

Novament la funció de cada codi s'explica a continuació:

**-Countdown:** És el mateix codi que controla el compte enrere que al Joc 1, però amb un altre nom. També s'utilitza el mateix sistema d'animació i esdeveniment que al primer joc.

**-JuegoManager :** Novament, aquest *script* controla el temps de partida i la puntuació ideal a assolir i carrega el fi de joc.

**-InputManager:** Quan aquest codi es troba en funcionament i es clica durant el joc, es projecta un raig imaginari anomenat *ray* des del punt de vista de la càmera. Si aquest detecta un objecte etiquetat com a "Fuego", veure il·lustració 56, executa diferents accions. La primera d'elles es destruir l'objecte i executar l'animació pròpia de l'objecte, que és un *Particle System* assignat a l'inspector.



Il·lustració 56 Etiquetatge d'objectes "Fuego".

Després es comunica amb el codi "FuegoBehaviour" per a que afegixi els punts al recompte. El fet d'etiquetar els objectes és útil doncs permet dir-li al programa que discrimini o no objectes segons la seva etiqueta. Així doncs, tots els objectes "Fuego" són els que es destrueixen, la resta no, com podria ser l'edifici o altres *assets*. La instrucció es pot veure a il·lustració 57.

```
if(Physics.Raycast(ray,out hit))
{
    if(hit.collider.tag == "Fuego")
    {
        Instantiate(fx_destroy, hit.point, Quaternion.identity);
    }
}
```

Il·lustració 57 Destrucció i càrrega d'animació.

**-Generador:** El generador controla el nombre d'objectes a carregar i les posicions on ho faran, el temps d'aparició dels focs "Range" i la quantitat d'objectes a mostrar "Length", com es pot veure a la il·lustració 58. En aquets cas *Length* és igual a dos objectes, dos focs, un petit i un gran, mentre que *Range* és un interval de temps i s'ha definit entre 7 i 10 segons, com es pot veure a la il·lustració 59. Dintre d'aquest interval, el programa genera aleatòriament els temps d'aparició, això fa que el joc no tingui un comportament previsible.



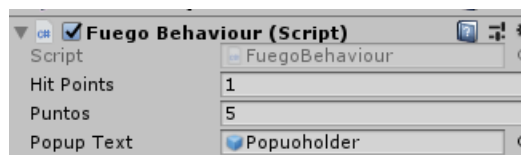
Il·lustració 58 Controlador d'objectes a carregar.

```
Invoke("Spawn", Random.Range(0f, 7f));
```

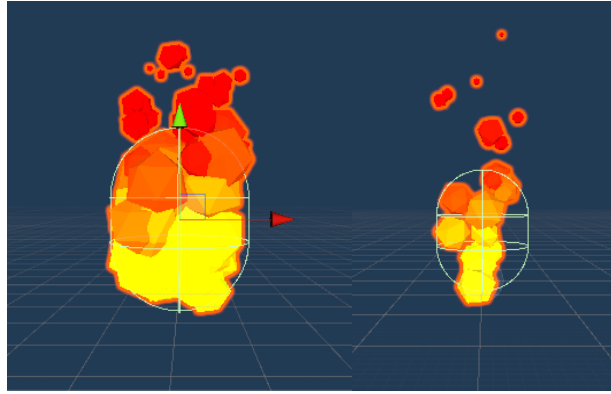
Il·lustració 59 Funció de control de temps.

Aquest codi va adherit a un objecte invisible anomenat "Generador" que actua com a font generadora de focs, similar al del primer joc. L'avantatge de anomenar una variable com a pública és que ens permet modificar des de la consola la mida del vector d'objectes a carregar, com s'observarà al joc 3.

**-FuegoBehaviour:** Aquest codi és l'encarregat de definir la puntuació dels focs, carregar la seva animació corresponent i destrucció dels objectes. Quan es destrueix un foc i es rep la informació de "InputManager" es suma punts a la puntuació general. Per a que resulti clar aquesta addició de punts també es defineix un *GameObject* anomenat "Popuholder" i conté un *script* anomenat "Popup text". Els focs, al tenir mides diferents també se'ls ha atorgat diferents puntuacions, il·lustració 60. El petit, al ser més difícil de clicar, té una puntuació de 10 punts, mentre que el gran 5. Els focs, per tant, tenen *colliders* de diferents mides, com es pot veure a la il·lustració 61.

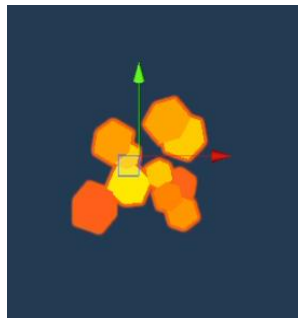


Il·lustració 60 Assignació de punts.



*Il·lustració 61 Mides de focs emprats.*

**-FXDestroy:** Aquest codi s'encarrega de controlar el temps de durada de l'explosió que representa la destrucció del foc, il·lustració 62. S'ha definit una variable pública que permet modificar el temps des de la consola.



*Il·lustració 62 Animació d'explosió.*

**-Popup Text:** Un cop es destrueix l'objecte es carrega una animació de text per a indicar els punts que afegeix el foc destruït, il·lustració 63, i aquest codi va connectat a un objecte de text per a que hi aparegui un símbol de “+” conjuntament amb la puntuació assignada al “FuegoBehaviour”.



*Il·lustració 63 Text d'informació de punts.*

-**Puntos:** Aquest codi va connectat a l'objecte *GameManager* i fa menció al *UserInterface*, on es mostra en pantalla el temps de joc i puntuació. Cada cop que s'afegeixen punts aquest codi li comunica al "UI" que s'actualitzi.

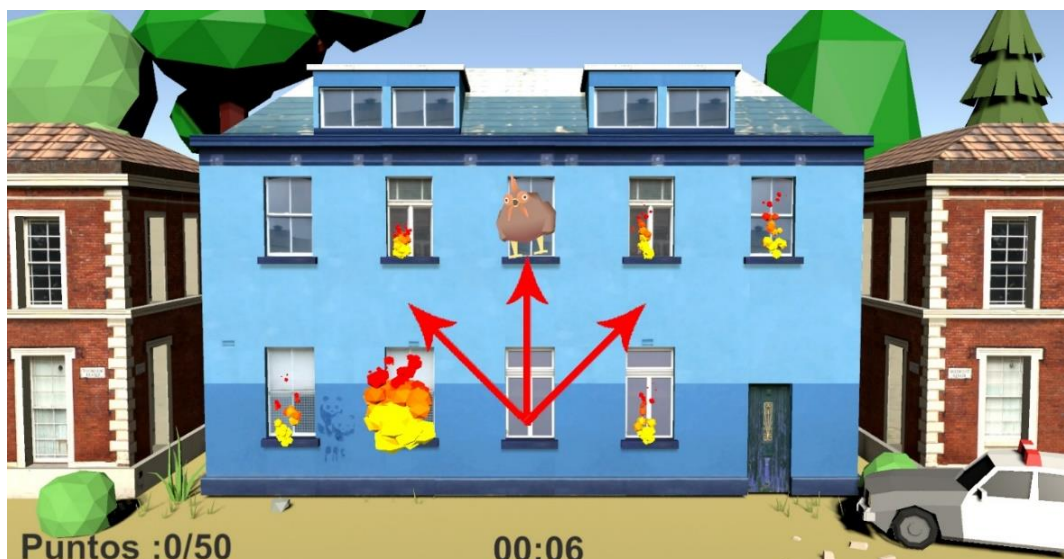
-**UI:** El UI es comunica amb "Puntos" i "JuegoManager". El temps es va actualitzant constantment per a mostrar-se en pantalla, els punts en canvi, s'actualitzen cada cop que es destrueix un objecte i mitjançant "Puntos" es refresca la informació.

-**Puntuaciónholder / Findeljuegopantalla :** Actuen de la mateixa forma que al Joc 1.

## 6.7. Joc 3

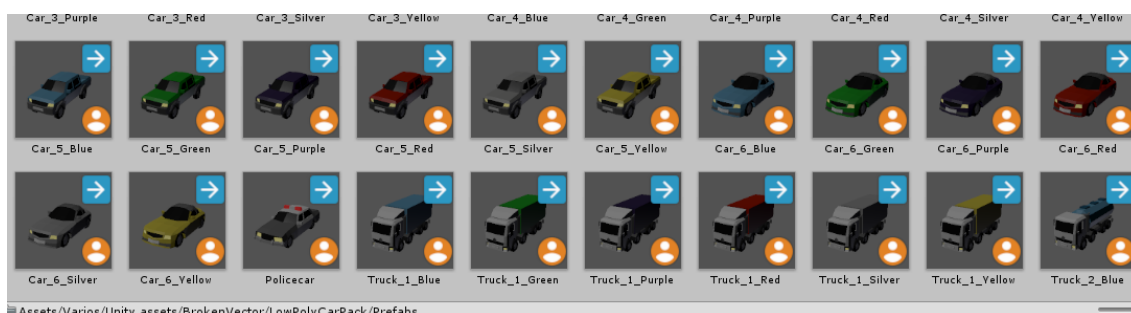
L'últim joc és una variant del Joc 2 ja que s'utilitza bàsicament el mateix sistema però amb algunes modificacions. La primera d'elles és la disposició de l'escena, mitjançant una col·locació diferent de la càmera per a que pugui abastar tot l'edifici, com s'aprecia a la il·lustració 64. Aquesta està pensada per a que al jugar amb el canell es pugui arribar a les finestres on hi apareixen diferents objectes fent que l'objectiu d'aquesta pantalla sigui exercitar el canell o l'avant braç.

En aquesta modalitat de joc els tipus d'objectes a carregar per l'objecte generador són 4, i tots ells tenen puntuacions diferents. S'ha pensat que un de els objectes penalitzi si hi cliques, així s'obliga de forma intencionada que la persona hagi d'esquivar aquell punt i vagi a buscar un altre de proper.



*Il·lustració 64 Escena Joc 3.*

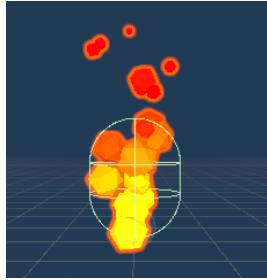
Un dels objectes nous a escena és el cotxe de policia, que novament s'ha extret de la llibreria Asset Store, il·lustració 65. La descàrrega contenia molts altres vehicles però només s'ha utilitzat el de policia.



*Il·lustració 65 Vehicles descarregats de l'Asset Store.*

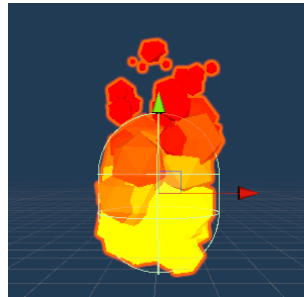
Com s'ha esmentat, la diferència principal d'aquest joc respecte al segon, és la quantitat d'objectes a carregar per l'objecte generador, i aquests són:

**-Foc petit:** Foc amb un *collider* de mida petita per a que sigui més difícil de clicar-hi, il·lustració 66. La seva puntuació és de +10 punts i té assignat com en el foc gran un objecte per a simular la destrucció.



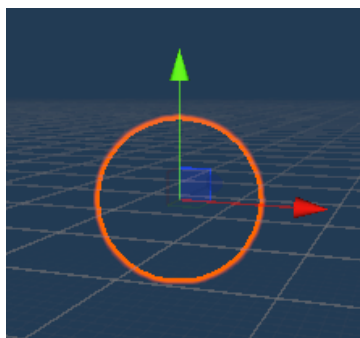
*Il·lustració 66 Asset foc petit.*

**-Foc gran:** Foc amb una mida considerablement més gran que el petit, per tant és més fàcil de poder destruir, il·lustració 67. La seva puntuació és de +5 punts.



*Il·lustració 67 Asset foc gran.*

**-Objecte invisible:** L'objecte invisible és un *asset* pensat per a que l'usuari o usuària no es posicionin a sobre d'una finestra i esperin a que aparegui un foc. Al generar-se un objecte invisible, il·lustració 68, no es pot interactuar amb ell i s'ha de canviar d'objectiu.



*Il·lustració 68 Asset objecte invisible.*

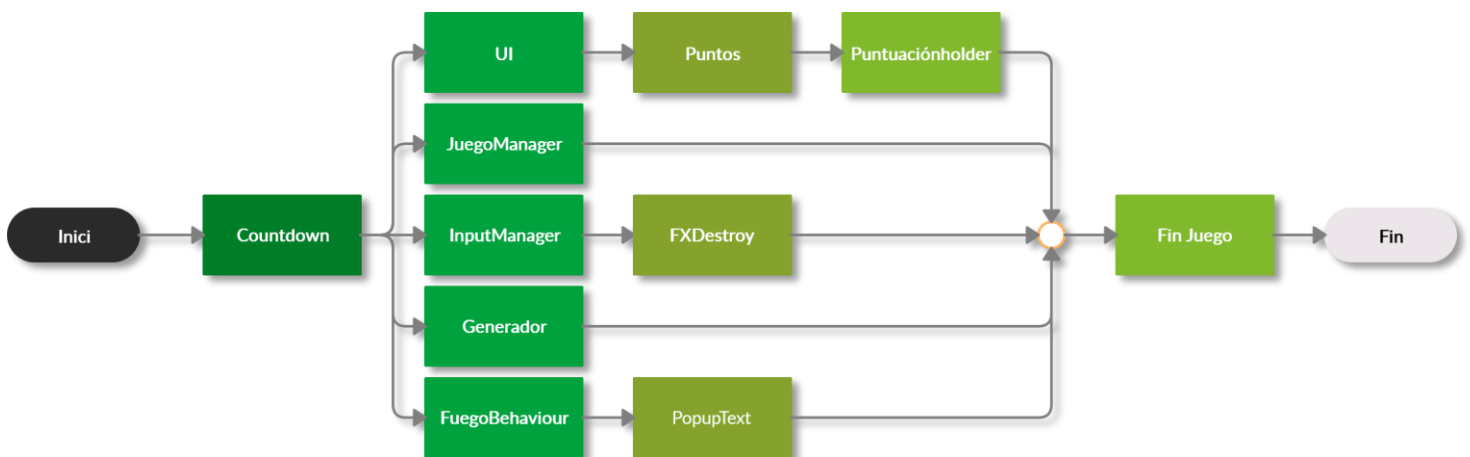
-**Gallina**: La gallina segueix el mateix principi que l'objecte invisible, en aquest cas però, l'objecte sí que es pot destruir i aquest té una puntuació de -5 punts, il·lustració 69. S'ha pensat per tal que un estigui atent o atenta per a no clicar-hi, fet que dinamitza el joc.



Il·lustració 69 Asset gallina.

### 6.7.1. Scripts Joc 3

Els *scripts* utilitzats en aquest joc són els mateixos que al Joc 2, il·lustració 70, ja que s'ha aprofitat la mateixa base de joc però afegint-hi algunes modificacions pel que fa a objectes i escena, el funcionament però, és exactament el mateix.



Il·lustració 70 Seqüenciació Joc 3.

## 7.Resultats

### 7.1.Estat del joc

Un cop finalitzat el disseny dels tres jocs, l'ADFO va demanar una versió del joc per a que es pogués utilitzar amb pacients i així provar si el joc és funcional o no. La versió que es va enviar funcionava sense errors de compilació i amb un entorn de Unity ben organitzat per a identificar ràpidament els directoris dels objectes en cas de millora o modificació.

Al cap d'unes setmanes es va rebre *feedback* per part de l'associació i el resultat va ser satisfactori, ja que la majoria d'usuaris o usuàries els hi havia agradat el disseny dels jocs i volien poder continuar jugant a aquests. (10.1. Carta *feedback* ADFO). L'associació va proposar una sèrie de millores per a acabar d'adaptar els jocs a les persones que hi jugaran:

-El cursor del ratolí hauria de ser més gran, ja que algunes persones tenen dificultat per a poder identificar-lo a la pantalla.

-Al primer joc es va demanar poder intuir la trajectòria de la pilota als nivells inicials i augmentar el grau de complexitat mitjançant diferents nivells.

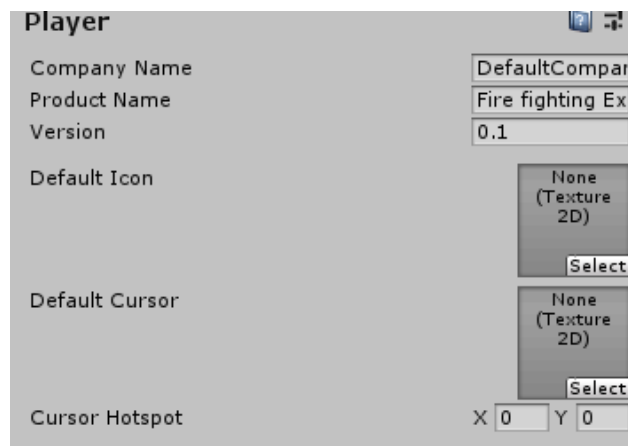
-Incorporar algun aspecte cognitiu al tercer joc. A part de ser un joc entretingut, volien afegir imatges per a que el jugador o jugadora les hagués de relacionar amb altres objectes de la pantalla.

### 7.2.Modificacions versió d'entrega

#### 7.2.1.Ratolí

Aquest requisit tenia com a finalitat poder identificar amb major facilitat el cursor a la pantalla, ja que alguns o algunes pacients si han patit dany cerebral poden tenir afectacions a nivell visual. Per a poder aplicar aquest canvi s'ha hagut de crear un *script* que controla el cursor del ratolí. (Veure 10.3.5. Codis varis).

Aquest codi permet carregar una imatge que substitueix el *mouse*. Per a que no hi hagi interferència entre l'antic i el nou, també es controla la visibilitat de l'antic, passant a estar en mode no visible: `Cursor.visible = false;` Unity, als ajustos del joc, permet a l'usuari o usuària poder substituir el cursor mitjançant la consola del programa, com es pot veure a la il·lustració 71.



*Il·lustració 71 Substitució del mouse a Unity.*

Aquesta funcionalitat no és útil ja que la mida màxima del cursor que es pot carregar mitjançant els ajustos és molt petita, i precisament es volia afegir una de més gran. Es van fer proves amb diferents imatges i es va arribar a la conclusió que la millor d'elles és un cercle, il·lustració 72. Amb el *mouse* antic el punt on s'aplica la interacció a l'hora de clicar és a la punta, i al carregar imatges de mides diferents aquest punt patia un *offset*, fent que fos poc intuïtiu i confús.

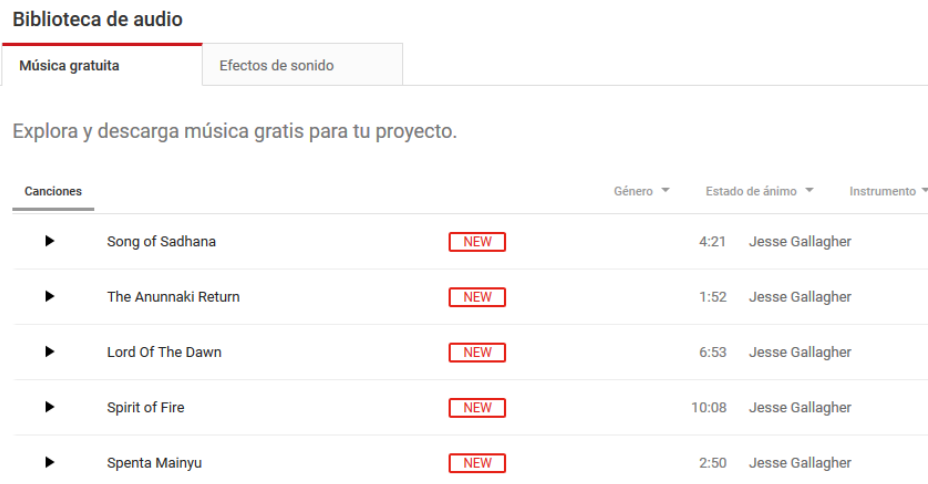


*Il·lustració 72 Mouse actual vs. predeterminat.*

## 7.2.2. Música de fons

Intentant fer del joc una experiència més amena, s'ha afegit una melodia a un volum baix per a evitar distreure a la persona a nivell d'estímul. Com la majoria de cançons tenen drets d'autor, s'ha hagut de recórrer a dominis on hi ha música lliure de drets o *royalty free*.

Una d'elles, i amb bon contingut, és el repositori de YouTube, il·lustració 73, que conté música i efectes de so aptes de descàrrega gratuïta.



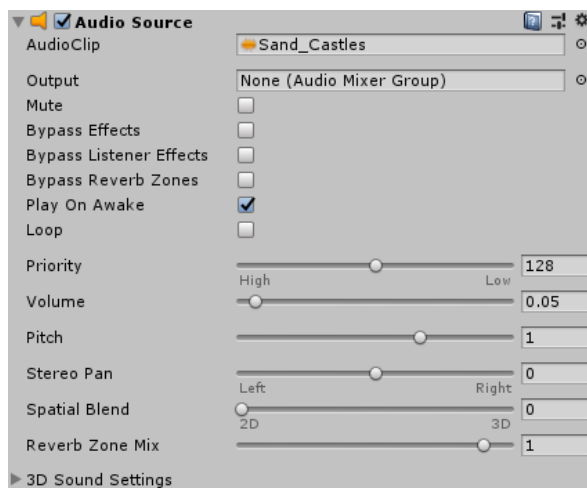
Il·lustració 73 Exemples de cançons de YouTube. (Font: <https://www.youtube.com/audiolibrary/music?nv=1>)

Un dels principals inconvenients que presenta Unity a l'hora de treballar amb clips de so i voler integrar-los a la partida és el control d'aquests. Quan es produeix un canvi d'escena la música es talla, ja que està associada a aquella pantalla en concret. Per a evitar aquest fenomen s'ha generat un petit codi que controla aquesta funció, il·lustració 74, i s'ha aplicat a un *Audio Source*, il·lustració 75. (Veure 10.3.5. Codis varis).

```
DontDestroyOnLoad(this.gameObject);
```

Il·lustració 74 Comanda que permet seguir reproduint la música.

El tractament de so a Unity va molt més enllà del que a simple vista presenta el programa en sí, ja que aquest està bastant limitat, i en general, la comunitat de desenvolupadors se les ha d'enginyar per a poder portar a terme els seus casos concrets.



*Il·lustració 75 Component Audio Source de Unity.*

### 7.2.3. Botó menú durant la partida

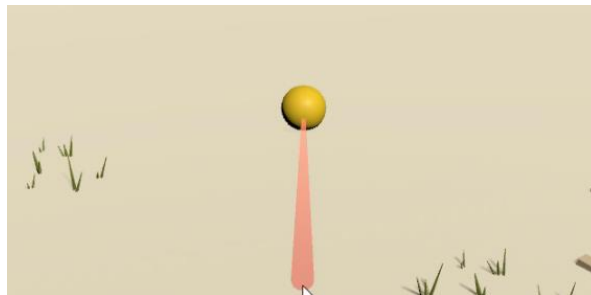
S'ha decidit afegir un botó de menú durant la partida, il·lustració 76, així en cas de voler abandonar-la no s'ha d'esperar a que s'esgoti el temps, ja que pot resultar molest. Al ser el temps de partida variable, alguns cops era excessiu. El control d'aquest es du a terme amb el codi que controla els altres botons del joc.



*Il·lustració 76 Botó menú cantonada dreta inferior.*

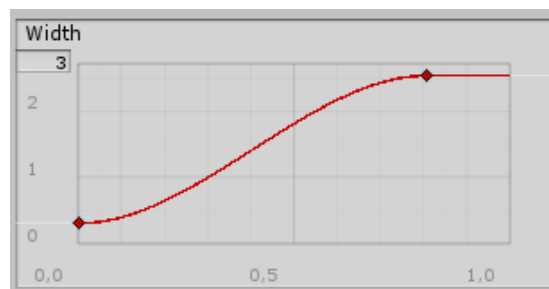
### 7.3. Modificacions Joc 1

Per a que es pogués intuir la trajectòria de la pilota, s'ha afegit al codi "Bola" una opció que permet generar una estela que indica la quantitat de força aplicada (amb unitats indeterminades) sobre l'objecte i la direcció cap on anirà al deixar-la anar, il·lustració 77. Es pot seleccionar el color de l'estela i també si es vol que la càmera del joc segueixi a la pilota durant el seu desplaçament.



Il·lustració 77 Estela que indica la direcció i potència.

El color de l'estela es pot ajustar, així com la forma d'aquesta. S'ha decidit generar-la en forma de llàgrima, així quan més gran és la força més accentuada és la forma. Com es pot veure a la il·lustració 78, la gràfica representa la variació d'amplada respecte a l'amplitud en píxels. Al punt d'origen o distància 0 l'amplada és pràcticament nul·la, però a mesura que s'allarga, aquesta creix. (Veure 10.3.1. Codis Joc 1).



Il·lustració 78 Control de l'amplada de l'estela.

### 7.4. Modificacions Joc 3

Les primeres revisions del joc van estar realitzades pel professorat. Aquests van donar el vist i plau però no va ser fins la revisió de l'ADFO que es van proposar una millora o modificació del joc.

Aquesta consistia en fer aparèixer una imatge en pantalla i anar generant objectes per tal que la persona cliqui sobre l'objecte que correspon a la imatge. Aquesta proposta es fonamenta en exercitar aspectes cognitius del o la pacient. Es va valorar l'opció d'incorporar les modificacions al joc existent i es va concloure que segurament comportaria problemes a causa de la manca de temps que es disposava. L'alternativa per a poder satisfer el suggeriment va ser crear un quart joc basat en el funcionament *Matching game* o *Memory* i es va afegir al menú inicial l'accés al joc, il·lustració 79.



Il·lustració 79 Menú inicial amb Joc 4.

Aquest tipus de joc consisteix en la col·locació aleatòria de cartes cap per avall i a l'anar clicant en elles es revela una imatge que la persona ha d'emparellar. Les imatges que es mostren a les cartes corresponen a elements coneguts i familiars per a tothom. Presenten formes, mides i colors diferents que afavoreixen la retenció o identificació cognitiva., com es pot veure a la il·lustració 80.



Il·lustració 80 Joc 4

Bàsicament el joc funciona generant una matriu amb imatges carregades amb noms diferents entre elles. Quan es detecta una coincidència entre imatges les congela, en el cas contrari les torna a girar. (Veure 10.3.3. Codis Joc 4).

## 7.5. Instruccions

S'ha decidit realitzar un menú d'instruccions, il·lustració 81, per a que els jugadors o jugadores coneguin el funcionament d'aquests. (Veure 10.3.5. Codis varis). Per a generar les imatges de text s'ha utilitzat Photoshop, il·lustració 82.



*Il·lustració 81 Icona d'instruccions.*

**JOC 1:** L'objectiu d'aquest joc és alimentar amb una bola de pinso la gallina. Per a fer ho un cop finalitzat el compte enrere es prem la barra espaciadora per a fer aparèixer en pantalla una pilota. Al clicar sobre aquesta i arrossegar enrere apareix una barra de color que indica la direcció i força aplicada. S'ha d'apuntar i deixar anar per a que la pilota es mogui.

**JOC 2:** En aquest altre joc, s'ha d'intentar apagar els focs que es generen a les finestres de la forma més ràpida possible. Per a fer-ho s'ha de clicar sobre aquests.

**JOC 3:** En aquest altre joc, s'ha d'intentar apagar els focs que es generen a les finestres de la forma més ràpida possible. Per a fer-ho s'ha de clicar sobre aquests. S'ha d'evitar clicar a les gallines, ja que aquestes penalitzen i resten punts del marcador.

**JOC 4:** El funcionament d'aquests es basa en anar clicant les cartes, fer-les girar i veure quina imatge conté. L'objectiu és emparellar totes les cartes. Un cop es finalitza es pot tornar a jugar clicant al botó *RESET* o sortir al menú principal.

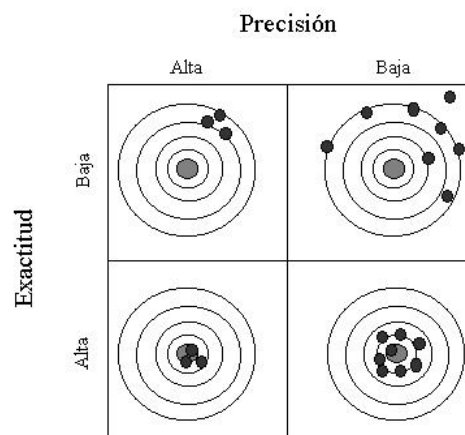


Il·lustració 82 Exemple d'instruccions en el joc.

## 7.6. Tractaments de dades

Com s'ha esmentat, el projecte va presentar una bona acollida per part de les persones del centre. Es va pensar que tot i que el joc s'hagués dissenyat tenint en compte les especificacions dels i les especialistes, juntament amb *game-elements*, seria convenient poder realitzar un seguiment o extracció de dades per a poder veure de forma gràfica si la persona està realitzant el joc correctament.

Un dels principals estimulants a l'hora de realitzar el joc és la puntuació, però que s'hagi assolit una bona qualificació no és indicatiu d'una realització correcte del joc, ja que la persona pot presentar poca exactitud i precisió a l'hora de jugar, il·lustració 83, per això s'ha decidit poder extreure les coordenades del ratolí, i amb un programa de full de càlcul representar els resultats.



Il·lustració 83 Diferències entre exactitud i precisió.

Per a l'obtenció de les dades es van realitzar dos mètodes inicials. El primer d'ells consistia en una mostra de les coordenades a través de la consola de Unity. Per tal que es mostressin, es va afegir un codi a la càmera principal del joc, i amb la funció: `print(Input.mousePosition);`, es mostraven les coordenades en temps real del ratolí.

Per a que hi hagués un refrescament i mostratge continu de les dades es va afegir la funció al mètode `Update()`. Al executar el codi, aquest executa l'`Update()` d'acord als *frames per second (FPS)* del videojoc, és a dir, la quantitat de fotogrames que componen un segon de vídeo del joc, i aquests solen ser de 60 FPS. La quantitat de coordenades generades era molt elevada i no hi havia manera de saber exactament quan s'estava clicant sobre els objectes d'interès del joc.

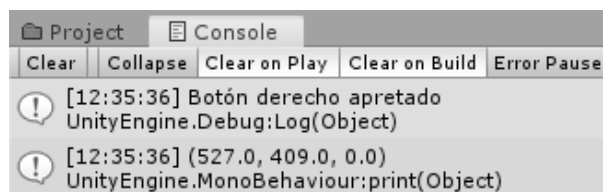
Com que aquesta aproximació era molt complicada i poc útil, es va decidir generar les coordenades seguint un altre criteri. Aquest consistia en mostrar només les coordenades del ratolí quan es clicava a la pantalla, mostrant la seva posició en píxels. D'aquesta forma no existia el problema de la gran quantitat de dades generades. ( Veure 10.3.4. Codis exportació de dades).

Es va decidir afegir al mètode `Update()` una condició `if` i imposar al programa que cada cop que es complís la condició mostrés a la consola un missatge per a saber que s'havia clicat i també les coordenades del ratolí, il·lustració 84. Per a fer-li saber al programa que es vol obtenir informació del botó dret del ratolí, aquest correspon al número 0 dins dels controls de Unity.

```
void Update()
{
    if (Input.GetMouseButtonDown(0))
    {
        Debug.Log("Botón derecho apretado");
        print(Input.mousePosition);
    }
}
```

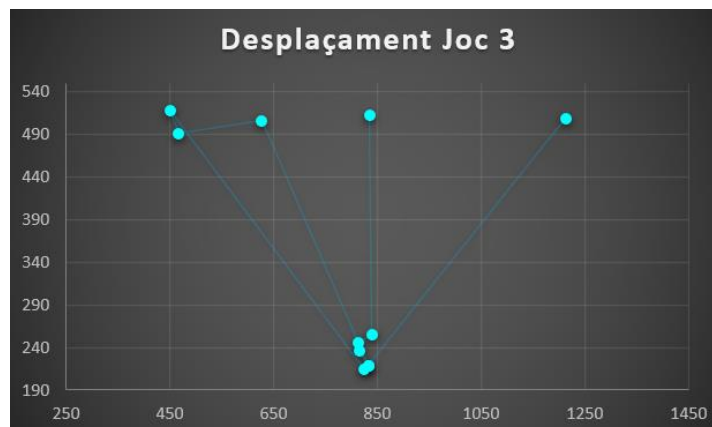
Il·lustració 84 Funció que carrega a la consola les coordenades.

Per tant, a la consola de Unity els missatges que apareixien eren els de la il·lustració 85.



Il·lustració 85 Missatges impresos a la consola de Unity.

Tractant les coordenades individualment i generant un gràfic es podien veure els punts on s'havia realitzat el clic i la trajectòria del ratolí, il·lustració 86. Tot i això, el tractament de les coordenades s'havia de fer d'una en una.

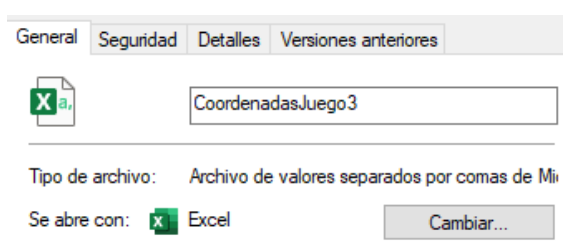


*Il·lustració 86 Mostra de representació de dades.*

L'única forma d'obtenir les dades era mitjançant la consola de Unity, i per a poder-les obtenir s'ha de tenir el programa instal·lat, fet que va obligar a descartar aquestes alternatives i haver de trobar-ne una de més pràctica.

L'alternativa a les dues propostes inicials ha estat treballar amb fitxers .csv. Aquests tipus de fitxers es poden executar amb programes de fulls de càlcul i emmagatzemar dades separant-les per comes.

El funcionament d'aquest codi es fonamenta en el mateix principi que els anteriors, carregant les coordenades del lloc on s'ha clicat. Quan l'escena en concret, o joc finalitza, es genera un arxiu "CoordenadasJuegoX.csv", il·lustració 87, a l'escriptori amb una columna de dades separades per comes, que corresponen als valors de X i Y. Aquests arxius es generen directament des de la versió *standalone*, sense la necessitat de disposar del motor Unity. (Veure 10.3.4. Codis exportació de dades).



*Il·lustració 87 Format d'arxiu generat.*

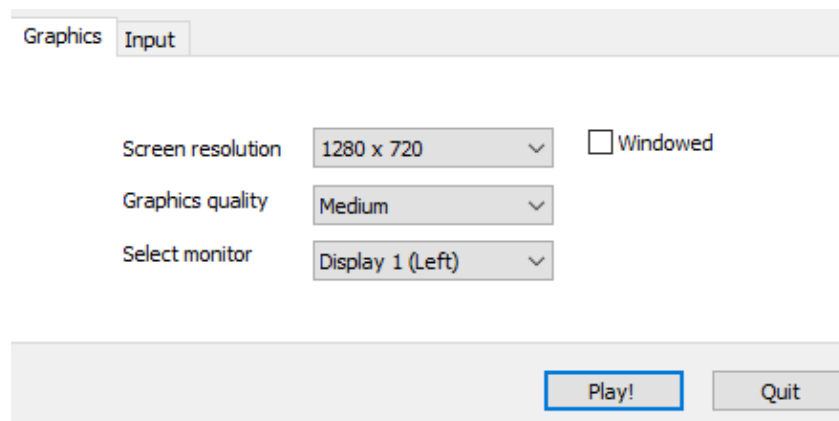
Aquesta forma de tractaments de dades és molt més pràctica i fàcil de treballar. Per a comprovar que les coordenades eren correctes s'ha realitzat comparacions entre els mètodes inicials i l'extracció de .csv per a corroborar que no hi ha errors, com es pot veure a la il·lustració 88 a la columna 1A i el primer missatge de la consola.

	A	
1	444,213	[13:17:52] (444.0, 213.0, 0.0) UnityEngine.MonoBehaviour:print(Object)
2	998,505	[13:17:52] Botón derecho levantado UnityEngine.Debug:Log(Object)
3	998,505	[13:17:52] (443.0, 212.0, 0.0)
4	822,51	UnityEngine.MonoBehaviour:print(Object)

Il·lustració 88 Comprovació de coordenades.

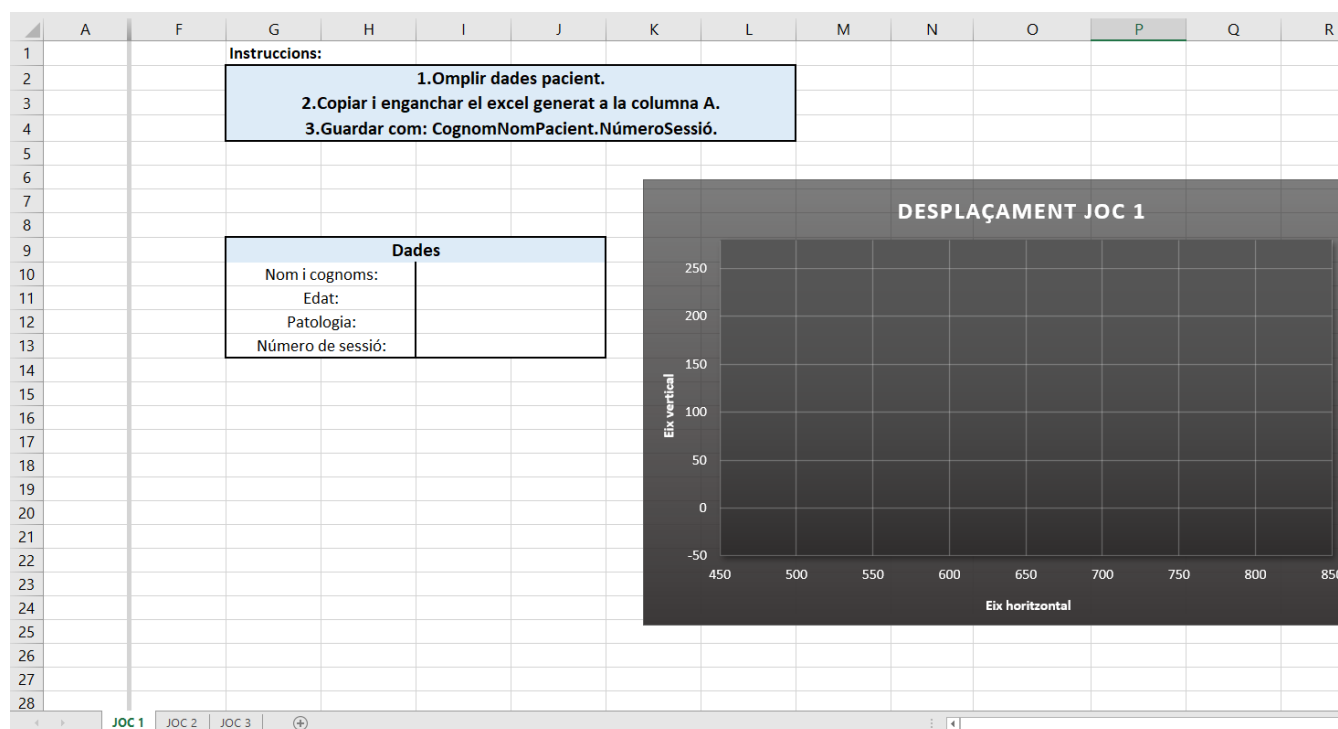
S'ha decidit realitzar una sèrie de proves *beta* amb persones properes per a poder veure el comportament d'aquestes dades. Per a facilitar la tasca s'ha generat un arxiu d'Excel en el qual enganxant la columna de dades extretes del joc, les representa gràficament, il·lustració 90.

Per a que l'adquisició de dades concordi amb el material proporcionat i les representacions siguin correctes es recomana realitzar la partida amb la configuració de resolució mostrada a la il·lustració 89.



Il·lustració 89 Resolució òptima per a realitzar la partida.

Algunes de les persones que han realitzat les primeres proves del joc han aportat les representacions incloses als annexes (10.2. Dades extretes de les partides beta).



*Il·lustració 90 Full de càlcul per a representar les dades.*

Aquest tractament de dades és útil a l'hora de poder saber si s'està movent el ratolí correctament, ja que al clicar en pantalla, es representa un punt en el gràfic d'acord a la coordenada on s'ha dut a terme el clic. A mesura que es juga es van acumulant punts i aquests mostren el comportament durant el joc.

El primer joc al estar dissenyat per a moviments verticals s'hauria d'esperar una representació similar a una línia recta, amb possibles desviacions en l'eix horitzontal depenent de la persona. Durant el segon joc es treballa en l'eix horitzontal i el resultat hauria de ser una línia majoritàriament plana. Per últim, el joc 3 al tractar-se d'un exercici per a canell, s'espera una dispersió que ocupi una gran part del gràfic, fet que mostra el moviment de canell per a arribar a tots els objectes distribuïts al llarg de la pantalla.

La validesa d'aquestes dades recau en poder identificar si la persona presenta dificultats a l'hora de jugar. Per exemple, a la pàgina 80, la persona té artrosi i presenta dificultats a l'hora de poder arribar a les parts més baixes del joc 3. Es pot intuir que la pacient hauria d'exercitar-se amb més sessions focalitzades en millorar aquest aspecte i poder fer un seguiment al llarg del temps.

## 8. Conclusions

Al començament d'aquest projecte es van voler dissenyar conjuntament amb l'ADFO tres aplicacions que exercitessin les extremitats superiors per a que aquestes es puguin utilitzar com a teràpies alternatives a les actuals, podent-se realitzar de forma telemàtica mitjançant un ordinador.

Com s'ha comprovat, aquestes teràpies alternatives cada cop es troben més presents a l'actualitat, doncs suposa una millora tant en aspectes lúdics com terapèutics, ja que les persones que juguen a aquestes aplicacions pensades per a poder rehabilitar certes lesions, presenten una molt bona resposta a entorns virtuals que siguin entretinguts i a l'hora els ajudin a exercitar-se. Sota aquesta premissa els quatre jocs han adoptat una estètica i dinàmiques que afavoreixen la participació dels pacients, com es va comprovar amb el comunicat per part del personal de l'ADFO.

Malgrat que els coneixements sobre programació són molt limitats o nuls, la recerca d'informació ha estat una part fonamental en el desenvolupament de les propostes per part del professorat i professionals. Aquest fet s'ha vist agreujat per la crisi sanitària, que ha fet que la comunicació entre els participants d'aquest projecte hagués de ser de forma telemàtica i les reparticions de tasques es veiessin condicionades segons el rendiment dels equips de treball informàtic.

A l'haver de realitzar el treball durant un confinament, els resultats obtinguts d'aquests jocs s'han hagut de dur a terme amb persones de proximitat. Els resultats, tot i ser orientatius sobre el comportament general de la persona al jugar-hi, s'ha pogut comprovar que el disseny dels jocs ha complert les expectatives inicials de disseny, ja que els resultats han corroborat que les persones juguen seguint les directrius proposades pels i les especialistes per a tal d'exercitar uns músculs en concret. També, es pot apreciar si la persona hi juga d'una forma precisa o si el nombre de clics al jugar-hi és elevat o no.

El resultat final ha estat la creació de quatre jocs d'acord als criteris inicials i posterior *feedback* per part de l'associació, presentant entorns virtuals amens i motivadors per a la persona que hi juga. Els resultats obtinguts amb el mostreig de dades ha corroborat que les dinàmiques de joc que un moment es van proposar com a teràpia alternativa per a realitzar amb ordinador s'han realitzat correctament i aquestes es podran implementar en un futur exitosament.

## 8.1. Línies futures.

A causa l'excelsionalitat d'aquest quadrimestre, aquest projecte es troba en fase "beta" i hi han molts aspectes a seguir desenvolupant per a treure-li encara més rendiment. S'anima a futurs grups a col·laborar-hi ja que tindran una bona base per a seguir treballant.

Les propostes de cara a futurs grups o projectes són:

-Poder escollir nivells de dificultat per a cada joc. Una de les propostes seria que al Joc 1 (tirador) s'incorporés un nivell on hi hagués diverses boques amb diferents mides de bola i s'hagués d'introduir en la boca corresponent. Això faria a la vegada que també es treballés l'aspecte cognitiu. Un tercer nivell podria ser un de similar al segon però on aquests es moguessin lleugerament per afegir precisió addicional.

-En el cas de que es volgués conservar el Joc 4, es creu que el nombre de cartes i imatges es pot veure ampliat segons el nivell de dificultat. També es poden afegir nous nivells mitjançant objectes del mateix color, mides o siluetes semblants, així la persona ha de presentar més atenció a l'hora de jugar-hi.

-Possibilitat de regular aspectes dels jocs, com aconseguir fer anar la sensibilitat del ratolí (que no s'ha pogut aconseguir) fins a poder modificar el temps o la freqüència d'aparició d'objectes mitjançant un nivell d'ajustos.

A la primera versió del joc es va intentar treballar en l'opció de poder ajustar la sensibilitat del ratolí, per a poder fer que els o les fisioterapeutes poguessin ajustar-la i afegir un grau de dificultat a l'hora de realitzar els jocs.

Es va investigar sobre aquesta opció, i es va veure que era bastant més complicat del que inicialment es pensava. La raó, és que Unity, tot i ser un dels programes més potents a l'hora de dissenyar entorns virtuals no té aquesta opció incorporada de forma intuïtiva, i molts usuaris l'han proposat per a versions futures.

El programa, però, sí que permet ajustar la sensibilitat del cursor quan per exemple es clica i s'arrossega un objecte, ja que al realitza-la, el programa permet variar la velocitat amb la qual l'objecte es mourà.

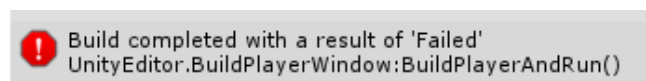
Aquest escenari no és representatiu, ja que en aquest joc, la interacció que es realitza és mitjançant *clicks*. La única forma que existeix per a poder variar la velocitat del *mouse* en aquesta situació, és generar un nou cursor, dir-li al programa que ignori el cursor de Windows, i redirigir tots els controls del ratolí predeterminat al nou.

Casualment es va trobar una aplicació que permetia modificar la sensibilitat del ratolí mitjançant el control general de Windows sense haver de sortir del programa Unity o del joc. Per tant es va crear un menú d'ajustos on es pot modificar aquest paràmetre, com es pot veure a la il·lustració 91.



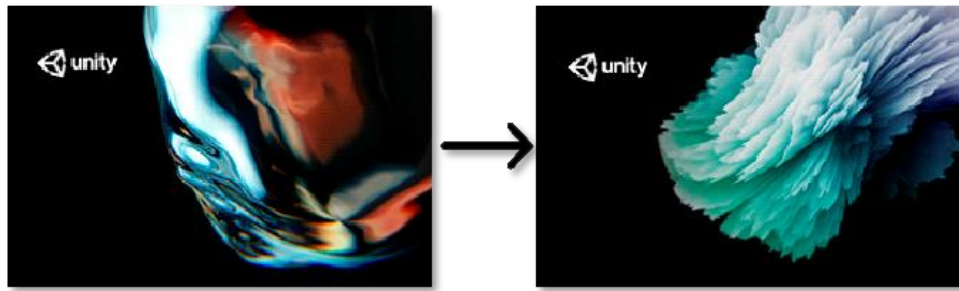
*Il·lustració 91 Menú ajustos de la sensibilitat.*

Durant el disseny del joc i al comprovar-lo tot funcionava bé. Es va decidir fer un *Build* i malauradament aquest codi interferia amb els controladors de Windows, il·lustració 92, concretament arxius .NET, i no permetia construir el joc per a poder executar-se en *standalone*.



*Il·lustració 92 Missatge d'error a la compilació del projecte.*

Es va decidir fer una migració de projecte, de la versió 2019 a la 2017, il·lustració 93, ja que s'havia llegit en fòrums que existia la possibilitat que es solucionés, però no va ser així.



*Il·lustració 93 Migració a versió anterior.*

Un dels principals inconvenients que té aquest *software* és la quantitat d'errors o *bugs* que presenta i això fa que cada poc temps es presenti una nova versió on es milloren o corregeixen els problemes de les versions anteriors. Aquest fet fa que existeixin molts problemes de compatibilitat entre versions i pot arribar a originar molts problemes.

Tota aquesta sèrie de fets, juntament amb la recuperació de *backups* va fer que es descartés aquesta opció ja que estava molt lluny dels coneixements de programació actuals.

-Millora de l'extracció de dades per a que el procés sigui més fluid, com per exemple crear un menú on ingressar el nom del pacient, guardar en directoris les partides de les persones i emmagatzemar-les a un servidor.

## 9. Pressupost i anàlisi econòmica

En aquest apartat s'especifica en tot detall els costos totals per a la realització d'aquest projecte. Per fer-ho s'ha dividit en quatre grans grups: Despesa personal, despesa de software, despesa de hardware i despesa de consum elèctric.

### 9.1 Despesa personal

El primer gran bloc i més important és el de despesa personal, que s'entén com el sou dels treballadors que han participat en aquest projecte això com la formació que han necessitat per a la realització.

En els sous, taula 1, s'ha tingut en compte la seguretat social que se li reté al treballador de la seva nòmina (corresponent al 6,35% del sou brut), així com la seguretat social de l'empresa (corresponent al 25%). Aquests costos es poden veure a la taula X.

Tipus de personal	Personal	Temps	Cost	Cost total
Enginyer Júnior	2	500 hores	35€/hora	35.000,00 €
Lloguer		3 mesos	980€/mes	2.940,00 €
			<b>Total</b>	<b>37.940,00 €</b>

Taula 1 Salari del personal.

A la taula 2, es pot veure els costos corresponents a la inversió en formació dels treballadors:

Curs	Preu	Nombre de cursos	Cost Total
Unity VR Fundamentals	- €	2	- €
Unity C# Scripting Fundamental	9,00 €	2	18,00 €
Classes Programació Online	12,00 €	6	72,00 €
		<b>Total</b>	<b>90,00 €</b>

Taula 2 Cost formació del personal.

Per a tant la part de despesa personal ascendeix fins a **38.030,00 €**.

## 9.2 Despeses de Software

El segon bloc correspon a les despeses de Software, taula 3. S'entén com a Software les instruccions per a comunicar-se amb el ordinador i que fan possible el seu ús. El software a la informàtica són els programes. Unity presenta una particularitat, i és que si els ingressos generats per el joc són inferior a 100.000\$ és gratuït.

Adobe Photoshop permet fer pagaments de forma mensual ( 24,19 euros/mes), per tant els dies emprats per a la realització del projecte han estat de 20 dies. S'ha decidit abonar dues mensualitats.

Per al que fa a Microsoft Visual Studio, el seu preu ha estat nul, doncs al ser estudiants, es disposa d'accés gratuït amb certes limitacions.

Així doncs en la taula a continuació es poden veure els programes utilitzats.

Software	Preu
Unity 3D	- €
Microsoft Visual Studio	- €
Adobe Photoshop	48,38 €
<b>Total</b>	<b>48,38 €</b>

*Taula 3 Despeses llicències.*

Per a tant les despeses de Software han estat de **48,38 €**.

## 9.3 Despeses de Hardware

El tercer bloc de despeses és el de Hardware, taula 4. S'entén com a Hardware aquelles parts físiques (tangibles) d'un sistema informàtic, els seus components elèctrics, electrònics i mecànics.

En el cas d'aquest projecte només s'ha utilitzat el ordinador així com els seus components. Per a aquest projecte cadascú dels 2 treballadors ha utilitzat el seu ordinador personal, així que s'ha fet un preu estimat del conjunt ordinador + perifèrics, que es pot veure en la taula a continuació.

Per obtenir el cost real primer s'ha estimat una vida útil de 36 mesos i un valor de 1200 pel conjunt ordinador. Aleshores per trobar el cost real s'ha fet ús de la següent formula :

$$\frac{\text{Mesos de ús (5)}}{\text{Vida útil en mesos (36)}} * 100 = \text{Depreciació} * \text{Cost real}$$

Producte	Preu (€/u)	Quantitat	Vida útil (mesos)	Depreciació (%)	Entitat	Cost real
Conjunt ordinador (Compost per 2 pantalles, teclat, ratolí i torre)	1200	2	36	13,9%	Personal	333,33 €

Taula 4 Cost material informàtic.

Per tant les despeses de Hardware són de fins a **333,33 €**.

## 9.4 Despeses elèctriques

L'últim bloc de despeses correspon al consum elèctric de tots els components utilitzats en el projecte, taula 5. S'ha agafat com a referència totes les hores del projecte (500), realitzades amb ordinador, i que el preu mitjà d'electricitat de la tarifa general de consum és de 0,077 €/KW\*hora. Aquestes despeses es poden veure reflectides en la taula a continuació.

Material	Consum (W/ud.)	Quantitat (ud.)	Cost
Ordinador	250	2	9,62 €
Pantalla	45	4	1,73 €
Ratolí	2	2	0,08 €
Teclat	5	2	0,19 €
		<b>Total</b>	<b>11,61 €</b>

Taula 5 Consums dels components.

Les despeses totals elèctriques són de **11,61 €**.

## 9.5 Despeses totals

A la taula a continuació, taula 6, s'hi troben especificades cada una de les despeses així com el total:

Despesa	Quantitat
Personal	38.030,00 €
Software	48,38 €
Hardware	333,33 €
Elèctrica	11,61 €
<b>Total</b>	<b>38.423,32 €</b>

*Taula 6 Cost total del projecte.*

El cost total del projecte és de **38.423,32 €**.

## 10. Bibliografia.

### 10.1 Bibliografia memòria

- [1]. “Flujo de trabajo de los Assets (Asset Workflow)” [Online] Manual de Unity 2016 Disponible: <https://docs.unity3d.com/es/530/Manual/AssetWorkflow.html>
- [2]. “Importando desde la Asset Store” [Online] Manual de Unity 2016 Disponible: <https://docs.unity3d.com/es/530/Manual/AssetStore.html>
- [3]. “Microsoft Visual Studio” [Online] Disponible : [https://es.wikipedia.org/wiki/Microsoft\\_Visual\\_Studio](https://es.wikipedia.org/wiki/Microsoft_Visual_Studio)
- [4]. “Link Trainer” [Online] Disponible : [https://en.wikipedia.org/wiki/Link\\_Trainer](https://en.wikipedia.org/wiki/Link_Trainer)
- [5]. “Edwin Albert Link” [Online] Disponible : [https://en.wikipedia.org/wiki/Edwin\\_Albert\\_Link](https://en.wikipedia.org/wiki/Edwin_Albert_Link)
- [6]. Alejandro Polanco Masa “Ed Link y el primer simulador de vuelo” [Online] 2015 Disponible: <https://alpoma.net/tecob/?p=10357>
- [7]. “Pygmalion’s Spectacles, Probably the First Comprehensive and Specific Fictional Model for Virtual Reality” [Online] Disponible: <http://www.historyofinformation.com/detail.php?entryid=4543>
- [8]. “Sensorama” [Online] Disponible: <https://es.wikipedia.org/wiki/Sensorama>
- [9]. “The Sword of Damocles (virtual reality)” [Online] Disponible: [https://en.wikipedia.org/wiki/The\\_Sword\\_of\\_Damocles\\_\(virtual\\_reality\)](https://en.wikipedia.org/wiki/The_Sword_of_Damocles_(virtual_reality))
- [10]. “History of VR - Timeline of Events and Tech Development” Disponible : <https://virtualspeech.com/blog/history-of-vr>
- [11]. “Realidad no inmersiva” [Online] Disponible: <https://proyectovrunicyt.wordpress.com/2017/04/19/realidad-no-inmersiva/>
- [12]. Alejandro Alcolea Huertos “Facebook presenta Half Dome, el futuro de Oculus y la realidad virtual” [Online] Maig 2018 Disponible : <https://computerhoy.com/noticias/zona-gaming/facebook-presenta-half-dome-futuro-oculus-realidad-virtual-79999>
- [13]. Jeremy Hortwitz “Facebook teases Oculus Half Dome 2 and 3 prototypes” [Online] Setembre 2019 Disponible : <https://venturebeat.com/2019/09/25/facebook-teases-oculus-half-dome-2-and-3-prototypes/>
- [14]. Ben Coxworth “ Epidermal VR brings a new sense of Touch to virtual reality” [Online] Novembre 2019 Disponible : <https://newatlas.com/vr/epidermal-vr-touch-virtual-reality/>
- [15]. “Skin-integrated wireless haptic interfases for virtual and augmented reality” [Online] Novembre 2019 Disponible: <https://www.nature.com/articles/s41586-019-1687-0>
- [16]. Dave Muoio “In-Depth therapeutic VR in 2018 is no longer just a distraction (therapy)” [Online] Maig 2018 Disponible: <https://www.mobihealthnews.com/content/depth-therapeutic-vr-2018-no-longer-just-distraction-therapy>
- [17]. “Neuro Rehab VR” [Online] Disponible: <https://www.neurorehabvr.com/>

- [18]. NEUROLleida “La rehabilitació de la memòria a través de l’ordinador: La plataforma “Guttmann NeuroPersonal Trainer” [Online] Maig 2017 Disponible: <https://www.neurolleida.cat/ca/associacio/neurolleida/blog/la-rehabilitacio-de-la-memoria-a-traves-de-lordinador-la-plataforma-guttmann-neuropersonal-trainer/17999.html>
- [19]. “Rehabilitación y Medicina Física: innovación” [Online] Disponible: <https://www.sjdhospitalbarcelona.org/es/rehabilitacion-medicina-fisica-innovacion>
- [20]. Paul L. Liebert “Lesiones de hombro” [Online] Juny 2018 Disponible : <https://www.msmanuals.com/es/hogar/traumatismos-y-envenenamientos/lesiones-deportivas/lesiones-de-hombro>
- [21]. Ana Martínez “Low Poly El Arte de lo Minimalista” [Online] Abril 2017 Disponible: <https://www.invergestudios.com/blog/el-arte-low-poly/>
- [22]. Martín Julio Vazquez “Low poly: el arte poligonal que triunfa en los videojuegos y más allá” [Online] Disponible: <https://www.domestika.org/es/blog/2232-low-poly-el-arte-poligonal-que-triunfa-en-los-videojuegos-y-mas-alla>
- [23]. Pàgina oficial ADFO Disponible: <https://www.adfo.cat/lentitat/lassociacio/>
- [24]. “La desviación de la mano: una de las principales causas de lesión de pianistas” [Online] Disponible: <http://pianoacoeur.com/es/desviacion-cubital-lesion-pianistas/>

## 10.2 Bibliografia fòrums

- [25]. “Using a custom rotation in Instantiate” Answers Unity [Online] Disponible: <https://answers.unity.com/questions/399924/using-a-custom-rotation-in-instantiate.html>
- [26]. “Instantiated Object loses prefab’s rotation” Answers Unity [Online] Disponible: <https://answers.unity.com/questions/146754/object-rotation-not-working-as-such.html>
- [27]. “The script needs to derive from MonoBehaviour” Answers Unity [Online] Disponible: <https://answers.unity.com/questions/406899/the-script-needs-to-derive-from-monobehaviour.html>
- [28]. “Animation Event won’t show the function I need” Answers Unity [Online] Disponible: <https://answers.unity.com/questions/624074/animation-event-wont-show-the-function-i-need.html>
- [29]. “Duplicate Full Scene With New Copies of Scripts?” Forum Unity [Online] Disponible: <https://forum.unity.com/threads/duplicate-full-scene-with-new-copies-of-scripts.70750/>
- [30]. “How to Reference the Spacebar Key in C# (Unity)” Stack Overflow [Online] Disponible: <https://stackoverflow.com/questions/47956273/how-to-reference-the-spacebar-key-in-c-sharp-unity>
- [31]. “Destroy OTHER objects on Collision Unity” Answers Unity [Online] Disponible: <https://answers.unity.com/questions/1077612/destroy-other-objects-on-collision-unity.html>

- [32]. “Solve error CS0029: Cannot implicitly convert type ‘UnityEngine.Transform’ to ‘UnityEngine.Vector3’ “ Answers Unity [Online] Disponible: <https://answers.unity.com/questions/1300794/how-do-i-solve-error-cs0029-cannot-implicitly-conv.html>
- [33]. “Why don’t animation events allow you to call functions that take a bool parameter?” Answers Unity [Online] Disponible: [https://answers.unity.com/questions/786404/why-dont-animation-events-allow-you-to-call-functi.html?\\_ga=2.203007189.570365954.1586790495-698460940.1573759271](https://answers.unity.com/questions/786404/why-dont-animation-events-allow-you-to-call-functi.html?_ga=2.203007189.570365954.1586790495-698460940.1573759271)
- [34]. “Material crash in 2.8” Blender [Online] Disponible: <https://developer.blender.org/T64108>
- [35]. “Remove one particular mesh of FBX models at runtime” Answers Unity [Online] Disponible: <https://answers.unity.com/questions/57735/i-have-a-bunch-of-fbx-models-and-i-want-to-remove.html>
- [36]. “Cursor.visible not working” Answers Unity [Online] Disponible: <https://answers.unity.com/questions/1252395/screenvisible-not-working.html>
- [37]. “How to display a flat background-2D image, not a skybox-behind everything in the camera” Unity Answers [Online] Disponible : <https://answers.unity.com/questions/9729/how-can-i-display-a-flat-background-2d-image-not-a.html>
- [38]. “The name ‘SceneManager’ does not exist in the current context” Answers Unity [Online] Disponible : <https://answers.unity.com/questions/1166190/assetsloadlevelcs917-error-cs0103-the-name-scenema.html>
- [39]. “Is there a way to duplicate a scene?” Answers Unity [Online] Disponible : <https://answers.unity.com/questions/313228/is-there-a-way-to-duplicate-a-scene.html>
- [40]. “Object don’t show in a game mode” Answers Unity [Online] Disponible : <https://answers.unity.com/questions/1205035/object-dont-show-in-game-mode.html>
- [41]. “Sprite showing up in Scene but not in game” Answers Unity [Online] Disponible : <https://answers.unity.com/questions/1057863/help-sprite-showing-up-in-scene-but-not-in-game.html>
- [42]. “Input.GetAxis” Unity Documentation [Online] Disponible : <https://docs.unity3d.com/es/530/ScriptReference/Input.GetAxis.html>
- [43]. “Select object, drag mouse to aim, release mouse to shoot?” Answers Unity [Online] Disponible: <https://answers.unity.com/questions/1470951/select-object-drag-mouse-to-aim-release-mouse-to-s.html>
- [44]. “Unity 3D Drag Rigidbody With a Mouse Cursor” Sharp Coder [Online] Disponible: <https://sharpcoderblog.com/blog/drag-rigidbody-with-mouse-cursor-unity-3d-tutorial>
- [45]. “Changing mouse sensitivity” Forum Unity [Online] Disponible: <https://forum.unity.com/threads/changing-mouse-sensitivity.126042/>
- [46]. “Changing the mouse sensitivity of FPSController” Forum Unity [Online] Disponible: <https://forum.unity.com/threads/solved-changing-the-mouse-sensitivity-of-fpscontroller.310978/>

- [47]. “How to change mouse sensitivity with slider” Forum Unity [Online] Disponible: <https://forum.unity.com/threads/unity-how-to-change-mouse-sensivity-with-slider.376428/>
- [48]. “Collision detection/preventing objects from moving through each other” Forum Unity [Online] Disponible : <https://forum.unity.com/threads/collision-detection-preventing-objects-from-moving-through-each-other.296349/>
- [49]. “Get Axis” Learn Unity [Online] Disponible: <https://learn.unity.com/tutorial/getaxis-e/?projectId=5dc4a877edbc2a00205e1124&tab=overview#5dc4a879edbc2a00205e1169>
- [50]. “Object.Instantiate” Unity Documentation [Online] Disponible: <https://docs.unity3d.com/ScriptReference/Object.Instantiate.html>
- [51]. “Cursor size with SetCursor or alternative?” Answers Unity [Online] Disponible: <https://answers.unity.com/questions/780190/cursor-size-with-setcursor-or-alternative.html>
- [52]. “Playing audio depending on scene” Answers Unity [Online] Disponible: <https://answers.unity.com/questions/1041174/help-playing-audio-depending-on-scene.html>

### 10.3 Bibliografia vídeos Youtube

- [53]. How To Make Gameobject Pulsing When You Hover Mouse Pointer Above It In Unity Software Game : [https://www.youtube.com/watch?v=0qlxrnD\\_8DQ&list=PLjQD3sdVY17IrF00mWUFfrLQsXKjiZJQD&index=2&t=0s](https://www.youtube.com/watch?v=0qlxrnD_8DQ&list=PLjQD3sdVY17IrF00mWUFfrLQsXKjiZJQD&index=2&t=0s)
- [54]. How To Make Gameobject Pulsing When You Hover Mouse Pointer Above It In Unity Software Game : [https://www.youtube.com/watch?v=0qlxrnD\\_8DQ&list=PLjQD3sdVY17IrF00mWUFfrLQsXKjiZJQD&index=2&t=0s](https://www.youtube.com/watch?v=0qlxrnD_8DQ&list=PLjQD3sdVY17IrF00mWUFfrLQsXKjiZJQD&index=2&t=0s)
- [55]. How To Make Gameobject Pulsing When You Hover Mouse Pointer Above It In Unity Software Game : [https://www.youtube.com/watch?v=0qlxrnD\\_8DQ&list=PLjQD3sdVY17IrF00mWUFfrLQsXKjiZJQD&index=2&t=0s](https://www.youtube.com/watch?v=0qlxrnD_8DQ&list=PLjQD3sdVY17IrF00mWUFfrLQsXKjiZJQD&index=2&t=0s)
- [56]. CLICK TO MOVE IN 3D - EASY UNITY / C# TUTORIAL: <https://www.youtube.com/watch?v=KU2CKBICAxQ&list=PLjQD3sdVY17IrF00mWUFfrLQsXKjiZJQD&index=3&t=0s>
- [57]. Unity - Keeping The Player Within Screen Boundaries: [https://www.youtube.com/watch?v=ailbszpt\\_AI&list=PLjQD3sdVY17IrF00mWUFfrLQsXKjiZJQD&index=4&t=0s](https://www.youtube.com/watch?v=ailbszpt_AI&list=PLjQD3sdVY17IrF00mWUFfrLQsXKjiZJQD&index=4&t=0s)

- [58]. How to Position Cameras - Unity Tips:  
<https://www.youtube.com/watch?v=DXri5QRC3HU&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=5&t=0s>
- [59]. Unity Course: Whack a Mole 3D:  
<https://www.youtube.com/watch?v=Z1GKfOEJPgQ&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=6&t=0s>
- [60]. How to get started with Unity - Using the Animator:  
[https://www.youtube.com/watch?v=gON\\_hhhvheI&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=7&t=0s](https://www.youtube.com/watch?v=gON_hhhvheI&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=7&t=0s)
- [61]. 32.Unity Mouse Input & Mouse Position - Unity C# Scripting Tutorial:  
<https://www.youtube.com/watch?v=riX1wv0JYv0&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=8&t=0s>
- [62]. 33.Unity Taking Input with Input.GetAxis - Unity C# Scripting Tutorial:  
<https://www.youtube.com/watch?v=B8OraV1-Z1I&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=9&t=0s>
- [63]. Unity Tutorial 21 - Mouse input:  
<https://www.youtube.com/watch?v=L6Z03BOIT8&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=10&t=0s>
- [64]. Get it straight - Using Angle in Unity:  
<https://www.youtube.com/watch?v=8jXubo53-jA&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=11&t=0s>
- [65]. Unity Raycasts and Crosshairs - GameDevHQ:  
<https://www.youtube.com/watch?v=xP1Ex32VY2k&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=12&t=0s>
- [66]. #7: Mira, arma y disparos (raycast, triggers, mouse en pantalla) [Unity, tutorial completo]:  
<https://www.youtube.com/watch?v=PLPV3DJX8NI&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=13&t=0s>
- [67]. Unity Spawn Prefab at Position Tutorial C# Easy:  
[https://www.youtube.com/watch?v=M\\_xXmplIOGys&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=14&t=0s](https://www.youtube.com/watch?v=M_xXmplIOGys&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=14&t=0s)
- [68]. Unity C# Creating and Deleting Objects:  
<https://www.youtube.com/watch?v=XO-E6QaTniQ&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=15&t=0s>
- [69]. Low Poly Racing - Making Of - Ep 2 - Unity & Blender:  
[https://www.youtube.com/watch?v=yoWS\\_G-9Ntw&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=16&t=0s](https://www.youtube.com/watch?v=yoWS_G-9Ntw&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=16&t=0s)
- [70]. Tutorial Blender español para principiantes / Árboles LowPoly de una manera muy, muy fácil:  
<https://www.youtube.com/watch?v=3w-4HOB9VyU&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=17&t=0s>

[71]. Unity 5 - From Beginner to Pro #44 - UI Image:

<https://www.youtube.com/watch?v=uew0zYUh3d0&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=18&t=0s>

[72]. HOW TO MAKE A GREAT MAIN MENU IN UNITY - TUTORIAL:

<https://www.youtube.com/watch?v=VXK1KMDAldo&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=19&t=0s>

[73]. Creating a Game Over Scene in Unity:

<https://www.youtube.com/watch?v=YdxYdHidCkE&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=20&t=0s>

[74]. 5 Minute MAIN MENU Unity Tutorial:

<https://www.youtube.com/watch?v=-GWjA6dixV4&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=21&t=0s>

[75]. How to Calculate Angles in Unity - A Unity Math Tutorial:

<https://www.youtube.com/watch?v=s-Ho5hF2Yww&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=22&t=0s>

[76]. Raycasting & Disparo - Tutorial Unity3d:

<https://www.youtube.com/watch?v=VLslOhPdNrs&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=23&t=0s>

[77]. Unity - Point and Shoot Tutorial

[https://www.youtube.com/watch?v=7-8nE9\\_FwWs&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=24&t=0s](https://www.youtube.com/watch?v=7-8nE9_FwWs&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=24&t=0s)

[78]. How to Export your Unity Game and Play it on Your Desktop (Official Unity Tutorial)

<https://www.youtube.com/watch?v=dx52hMxn0FY&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=25&t=0s>

[79]. DRAG & SHOOT in Unity | 2D Game Dev Tutorial:

<https://www.youtube.com/watch?v=Tsha7rp58LI&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=26&t=0s>

[80]. **Unity force Metter - Pull release add force:**

<https://www.youtube.com/watch?v=viJxlpBMgB8&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=27&t=0s>

[81]. How to show a drag line - [ Unity Tutorial ]:

<https://www.youtube.com/watch?v=UliUxEYK1js&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=28&t=0s>

[82]. How to Add Force To Gameobject with Swipe to Throw It In 3D Android Unity Game? Simple Tutorial:

<https://www.youtube.com/watch?v=7O9bAFyGvH8&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=29&t=0s>

[83]. Unity Creating a scoreboard and adding scores on collisions:

<https://www.youtube.com/watch?v=fMd3B0T5ow0&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=30&t=0s>

- [84]. HOW TO CUSTOMIZE YOUR MOUSE CURSOR - UNITY TUTORIAL:  
<https://www.youtube.com/watch?v=cCKlMAwvQcl&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=31&t=0s>
- [85]. Adding Custom Cursors in Unity:  
<https://www.youtube.com/watch?v=LxzHJgMrHyA&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=32&t=0s>
- [86]. Unity Soccer Star Tutorial Part-1:  
<https://www.youtube.com/watch?v=XkcbSescKHQ&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=33&t=0s>
- [87]. Make a Game like Golf Battle | Unity force Metter - Pull release add force in Unity 3D  
<https://www.youtube.com/watch?v=XkcbSescKHQ&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=33&t=0s>
- [88]. Unity3D Rigidbody Linear Drag:  
<https://www.youtube.com/watch?v=SrsZC0XMqNk&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=35&t=0s>
- [89]. Unity 101: 5 ways to move Unity3D Objects:  
<https://www.youtube.com/watch?v=tNtOcDryKv4&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=36&t=0s>
- [90]. How to control the ball like golf battle in unity 3d:  
<https://www.youtube.com/watch?v=xZ0pPnm3tDI&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=37&t=0s>
- [91]. Destroy Object On Collision In Unity:  
<https://www.youtube.com/watch?v=QsqAz5ohODO&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=38&t=0s>
- [92]. C# Instantiate in Unity! - Beginner Scripting Tutorial:  
<https://www.youtube.com/watch?v=Q3u0x8VRJS4&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=39&t=0s>
- [93]. unity 5 C# press key to instantiate and destroy GameObject:  
<https://www.youtube.com/watch?v=K6KjhrIFOWI&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=40&t=0s>
- [94]. Destroy Object in Unity 5 (Collision Event):  
<https://www.youtube.com/watch?v=H6GeDEF2Myw&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=41&t=0s>
- [95]. Destroy Object On Collision In Unity:  
<https://www.youtube.com/watch?v=DzvxP6kR2KU&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=42&t=0s>
- [96]. Smooth Mouse Follow in Unity | 2D Game Dev Tutorial  
<https://www.youtube.com/watch?v=Ehk9fKBwS3Y&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=43&t=0s>

- [97]. In-Game Virtual Cursor in Unity3D:  
<https://www.youtube.com/watch?v=AUGFokfNuT8&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=44&t=0s>
- [98]. How to add a score counter into your Unity 2D game | Easy Unity 2D Tutorial:  
<https://www.youtube.com/watch?v=QbqnDbexrCw&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=45&t=0s>
- [99]. Moving Player in unity | Unity Tutorials | CG AURA:  
[https://www.youtube.com/watch?v=Y\\_clgZ3ngx8&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=46&t=0s](https://www.youtube.com/watch?v=Y_clgZ3ngx8&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=46&t=0s)
- [100]. How To Change The Cursor In Unity! [Unity3D 2019 Beginner Tutorial]:  
[https://www.youtube.com/watch?v=W4SE0\\_cfAqc&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=47&t=0s](https://www.youtube.com/watch?v=W4SE0_cfAqc&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=47&t=0s)
- [101]. Background Music / Audio in Unity  
<https://www.youtube.com/watch?v=BKCsH8mQ-IM&list=PLjQD3sdVY17rF00mWUFfrLQsXKjiZJQD&index=48&t=0s>
- [102]. Countdown Timer In Unity - Easy Beginners Tutorial/ Guide  
<https://www.youtube.com/watch?v=o0j7PdU88a4>

## 11. Annexes.

### 11.1 Carta feedback ADFO

La reacció o resposta a la proposta del joc la van comunicar en forma d'escrit el personal de l'ADFO. En el text de a continuació es pot veure el missatge enviat.

#### VALORACIÓ GLOBAL PROPOSTA D'EXERCICIS (Maig 2020)

Reunides les persones participants en l'assaig clínic del 30 de maig de 2019, l'educadora social i terapeuta ocupacional d'ADFO, hem valorat diferents aspectes de les aplicacions proposades. La intenció ha estat conèixer les dinàmiques, valorar les habilitats a treballar i aportar possibles aspectes a tenir en compte per afavorir les propostes, tenint en compte les necessitats de les persones a qui van dirigides:

- Tots els participants coincideixen en que són de gran motivació aquest perfil d'aplicacions: la realitat virtual afegeix aspectes motivadors a la rehabilitació física i ajuda a mantenir el procés terapèutic en el temps i al domicili.
- Tanta és la motivació que tothom té ganes de provar les propostes i començar a portar a terme el projecte
- Des d'ADFO volem mostrar el nostre agraïment a tots els/les alumnes participants al projecte, per la vostra implicació en aquest projecte, unint TIC, social i salut. Creiem que s'està fent un gran esforç i estem convençuts que en sortirà un gran projecte, amb les aportacions de tothom.
- Estem molt contents de la gran varietat d'aplicacions que han sortit. Aquesta varietat fa que es pugui adaptar la rehabilitació als interessos i a les necessitats de cada participant, podent així pautar la rehabilitació personalitzada.

Aplicacions:

- Collir pomes: és una aplicació que agrada i que es veu viable de seguir. A més, el fet que la tècnica pugui decidir i que quedi registrat el recorregut, és un aspecte molt positiu perquè permet que el pacient el pugui utilitzar des de l'inici fins al final del tractament.

Es podria tenir en compte de cara a augmentar el grau de dificultat: podrien caure pomes que s'hagin de recollir i altres productes que no. Aquests a l'inici haurien de ser de diferent categoria semàntica (ex: pomes i cotxes)

- Pollo a l'ast: la dinàmica de l'aplicació i els moviments utilitzats estan molt bé. Caldria valorar la possibilitat de canviar els pollastres i el foc per icones més relacionades amb la realitat (Per exemple: icones d'elements habituals o imatges de personatges). Aquesta imatge podria sortir en una zona de la pantalla i que s'hagués de cercar a les finestres o que es veiés, desapareixes i s'hagués d'encertar a quina finestra es trobava.
- Disparar amb el tirador: la proposta és motivadora i els moviments que es treballen també. Animem a seguir treballant en aquest joc i a presentar el vídeo on es pugui observar la dinàmica i l'augment en el grau de complexitat. Proponem valorar si és possible marcar el recorregut en els nivells inicials.
- Exercicis AVD (forn, tallar verdures): aquest conjunt d'aplicacions han agradat molt, ja que fan referència a activitats que realitzen diàriament i que per tan, més enllà del joc, es poden aplicar.  
Proponem que l'entorn no estigui tan carregat d'estímul visual, de manera que permeti visualitzar més concretament quin és l'aspecte a treballar. També proposem que si es treballa amb extremitat dreta, el participant visualitzi a la pantalla la imatge en aquest cantó i no en l'altre, per les dificultats de reconeixement que podria suposar.

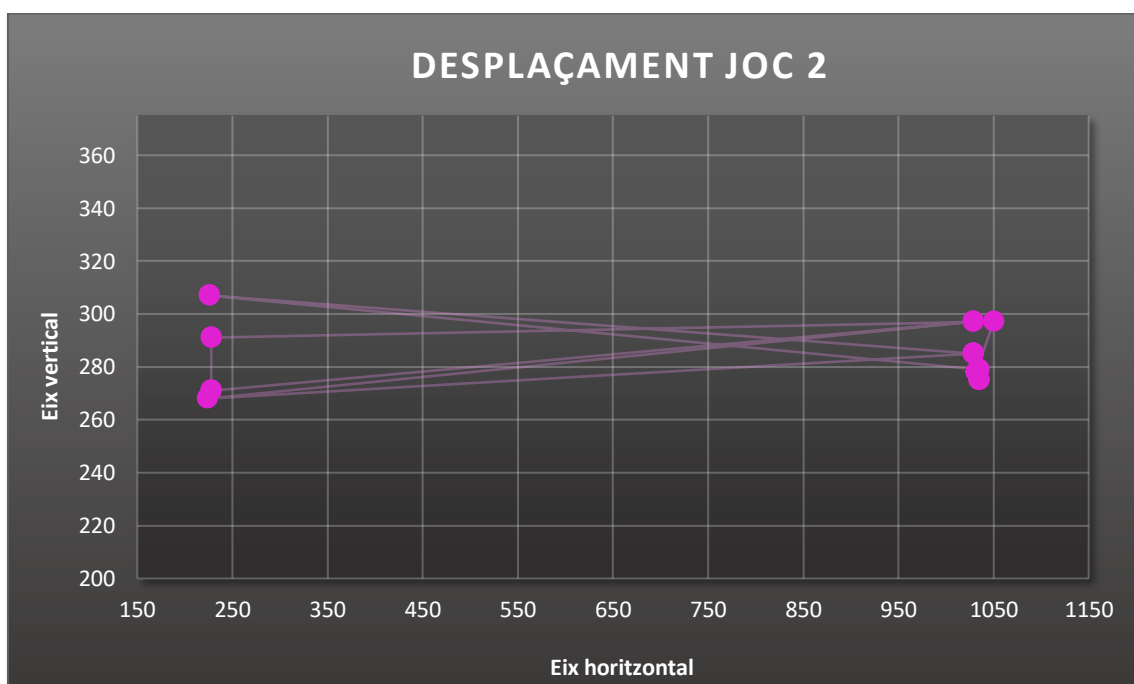
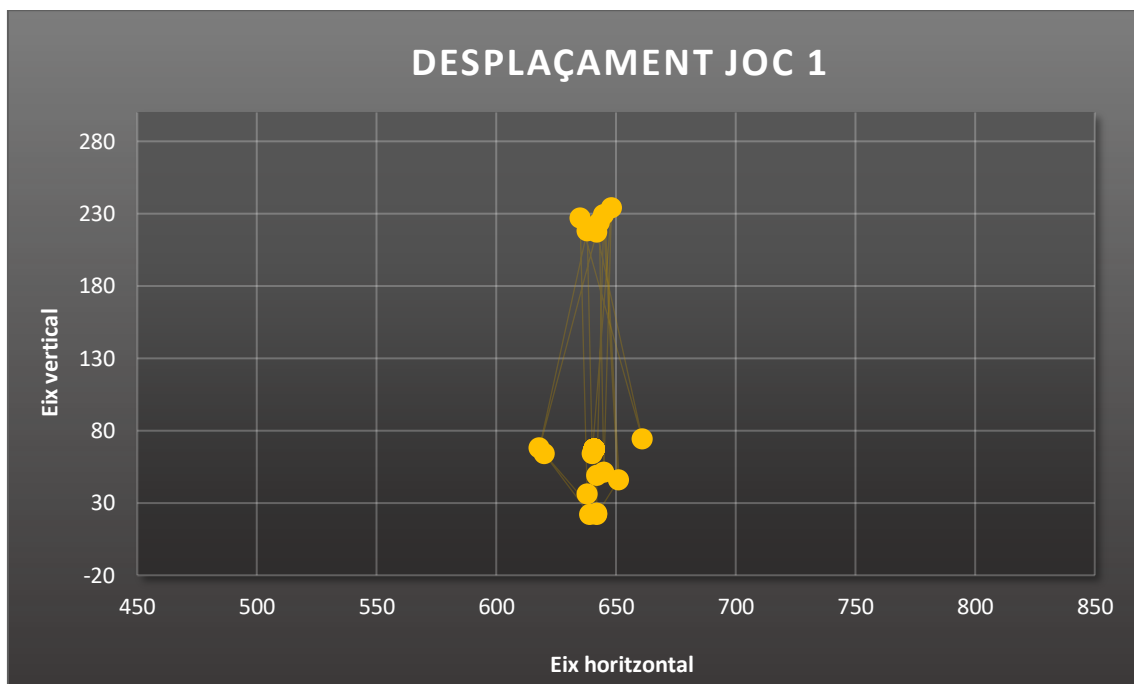
Finalment, animem a tots/es a continuar treballant en aquestes aplicacions tenint en compte que quan més es puguin adaptar l'aplicació en funció dels moviments que interressi treballar en aquell moment millor, per poder realitzar un procés terapèutic més personalitzat.

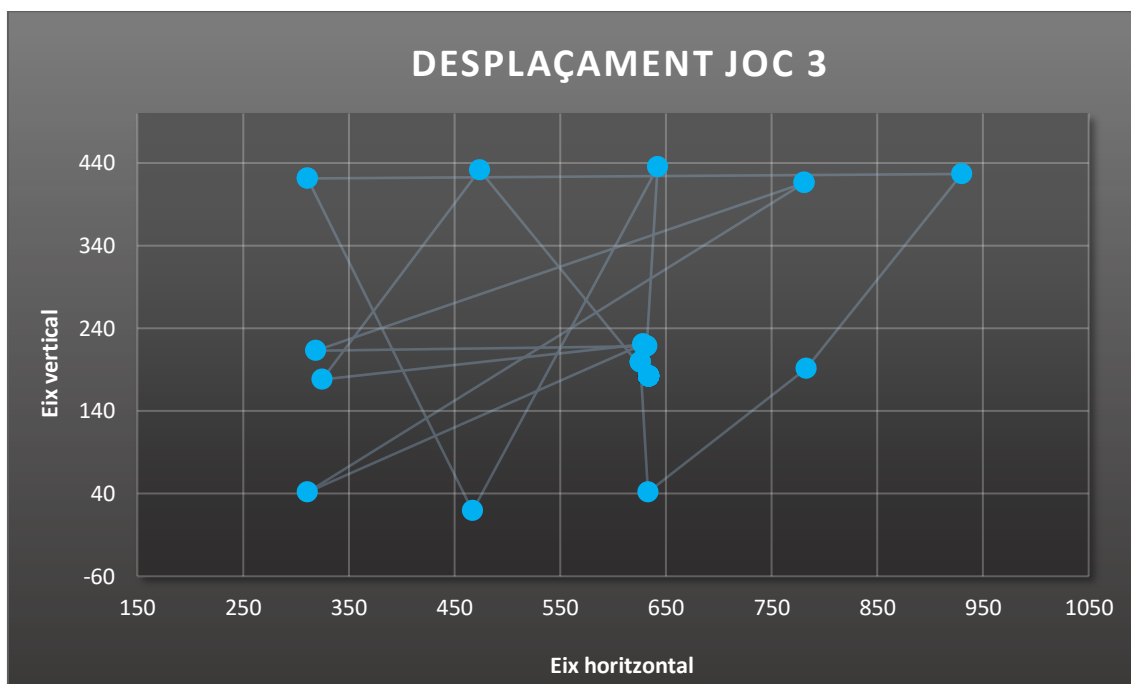
També comentar que a nivell visual, és important que no hi hagi una sobrecàrrega d'estímul i que aquests es puguin distingir icona/fons, degut a que quan hi ha un dany cerebral hi pot haver afectació a nivell visual.

Desitgem que tots els participants estiguin aprenent i gaudint elaborant aquestes aplicacions i que els siguin útils les presents propostes de millora.

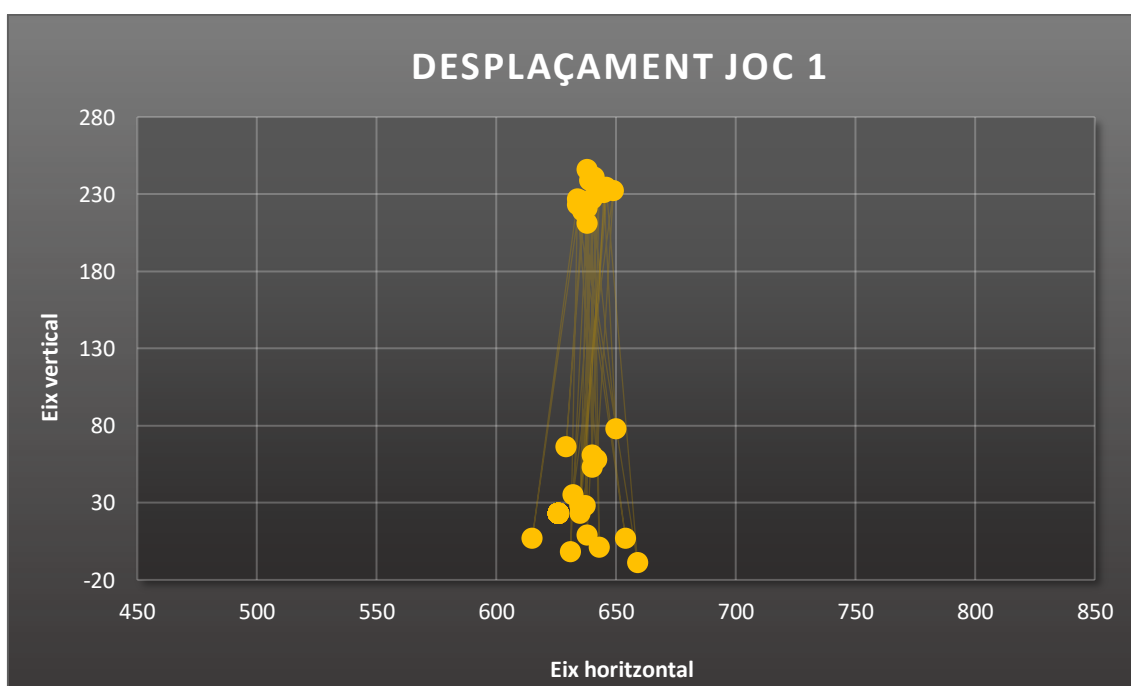
## 11.2. Dades extretes de les partides *beta*.

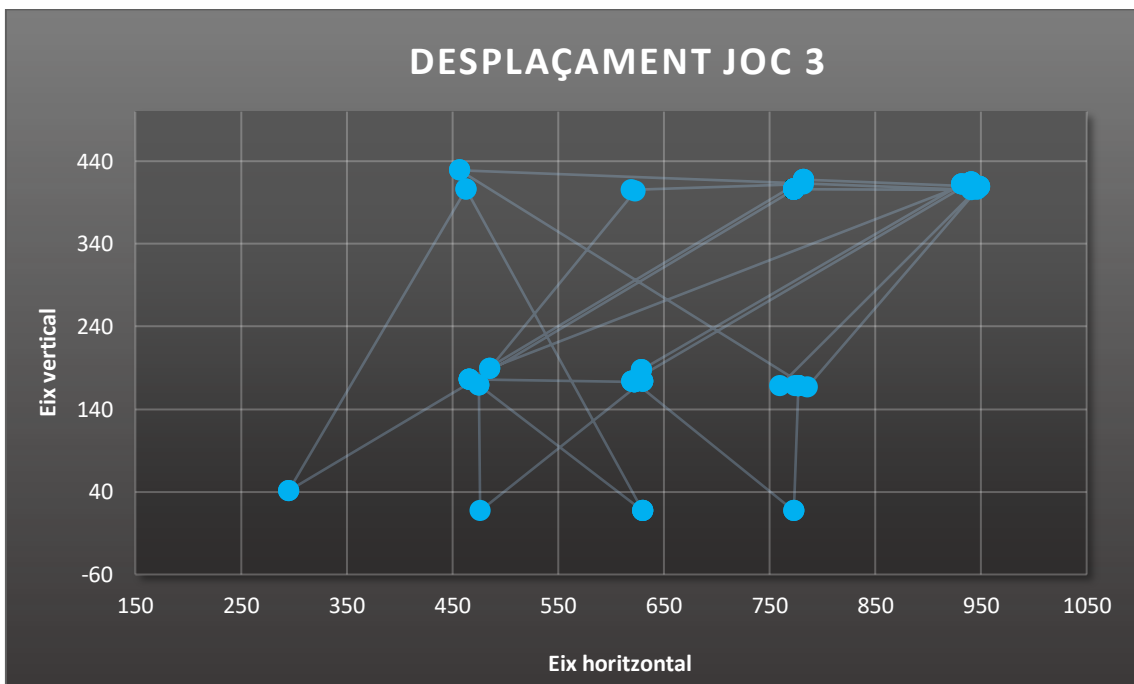
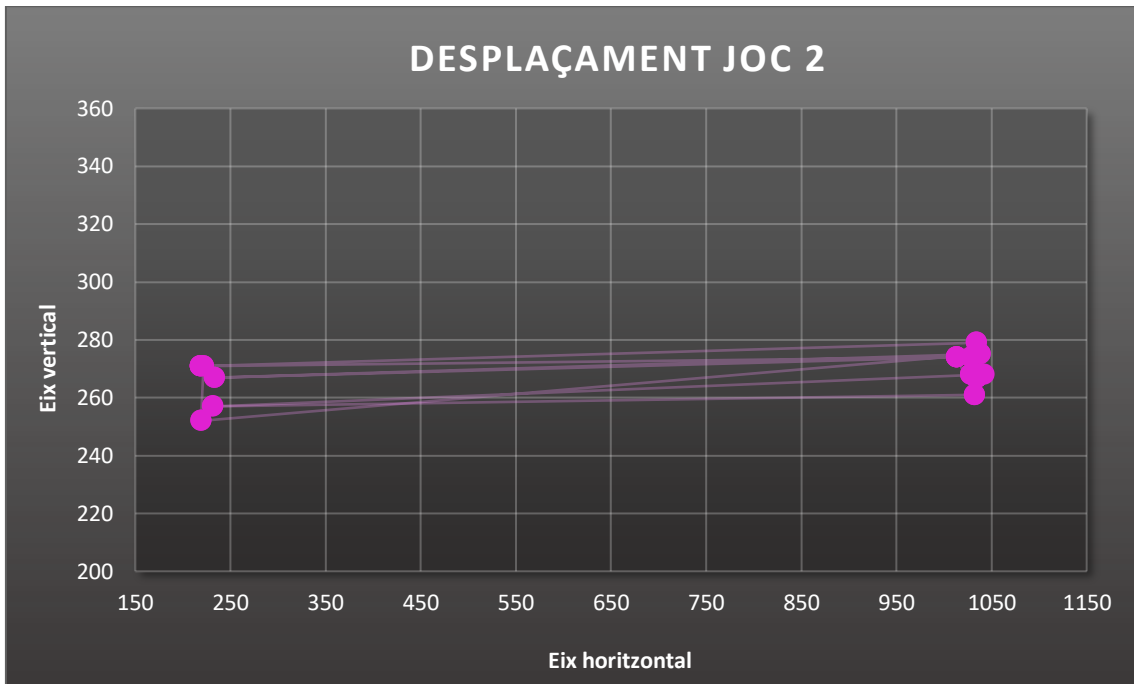
Dades	
Nom i cognoms:	Oriol Jiménez Bellido
Edat:	26
Patologia:	---
Número de sessió:	---



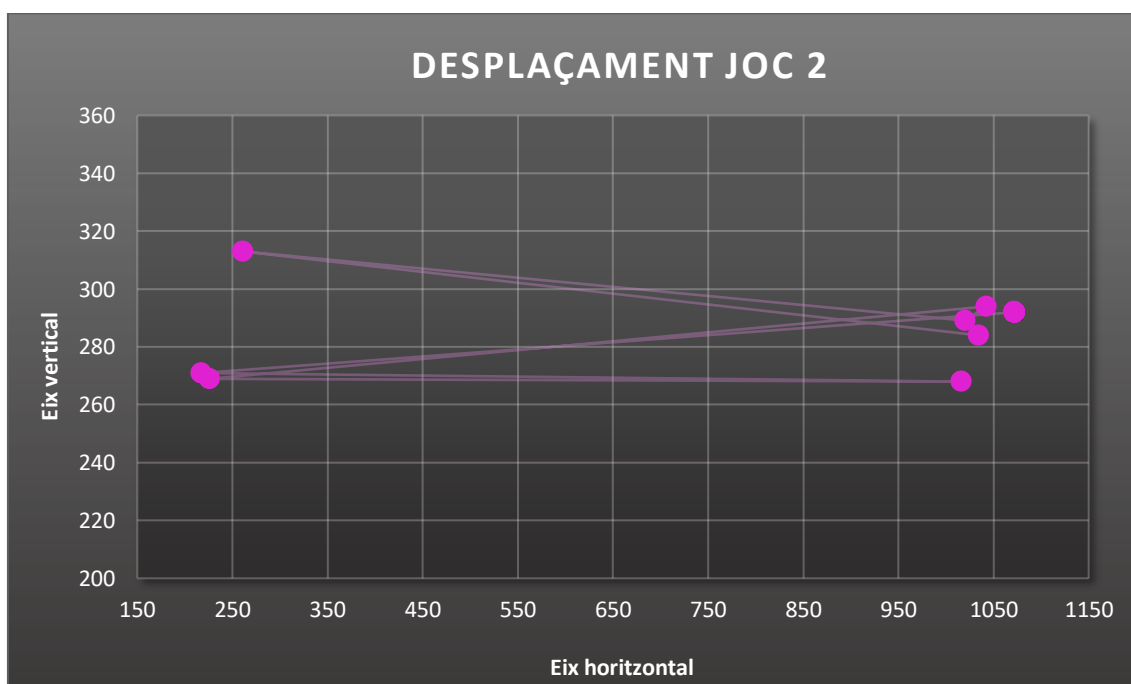
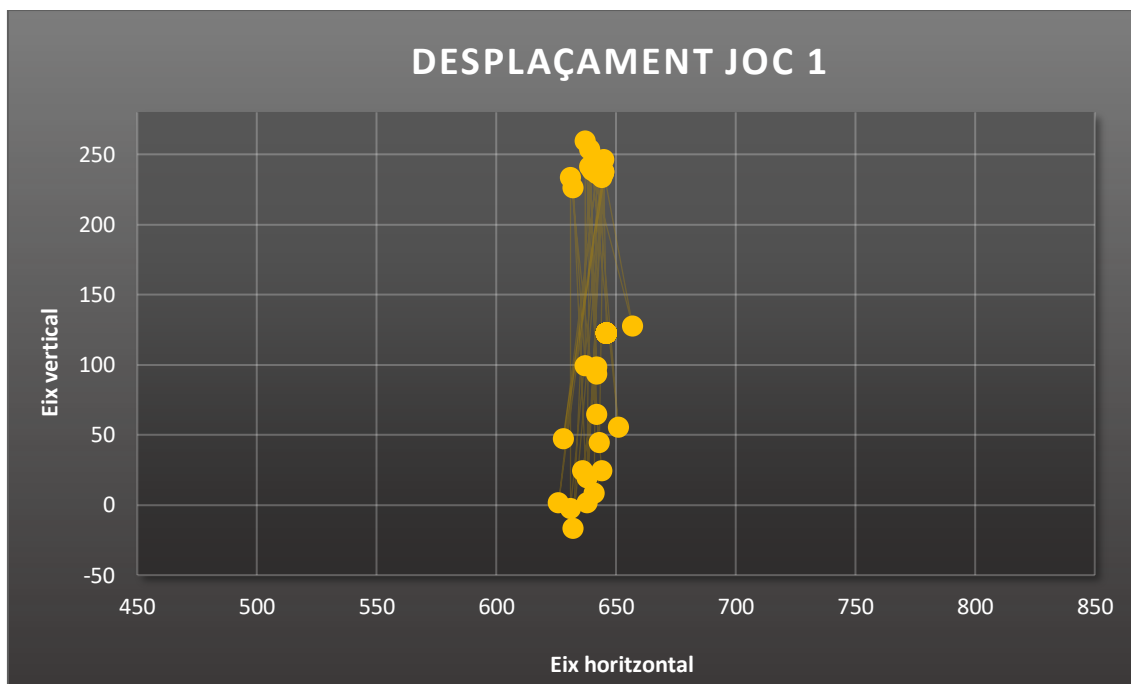


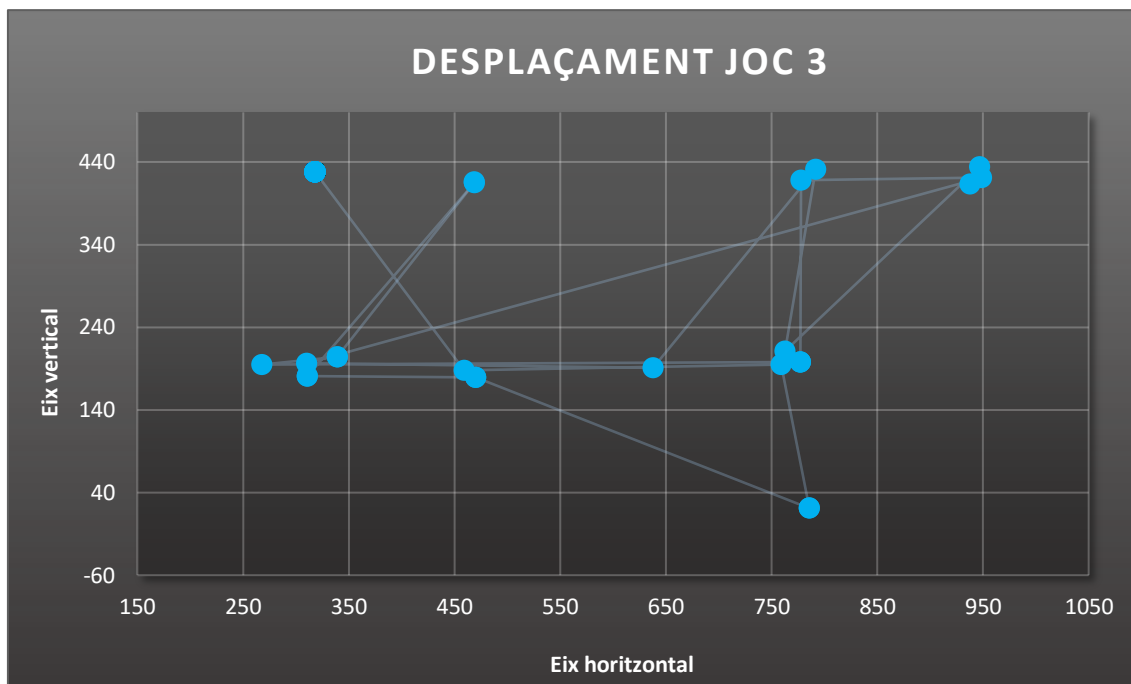
Dades	
Nom i cognoms:	Arnau Doñate i Rivilla
Edat:	23
Patologia:	---
Número de sessió:	---



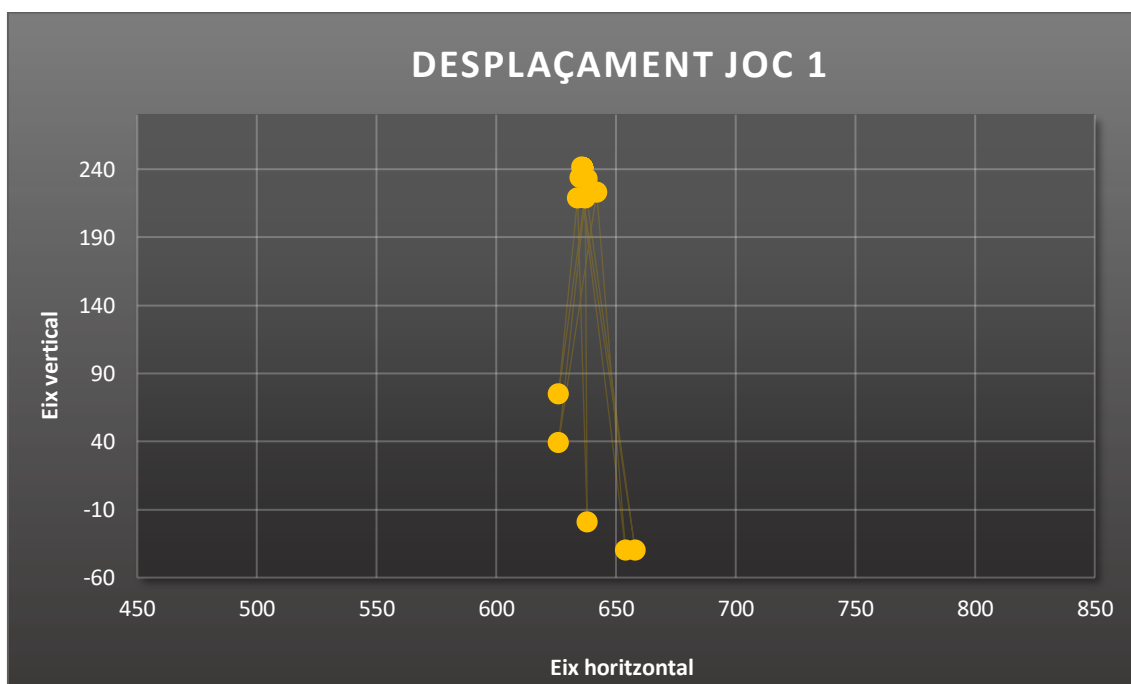


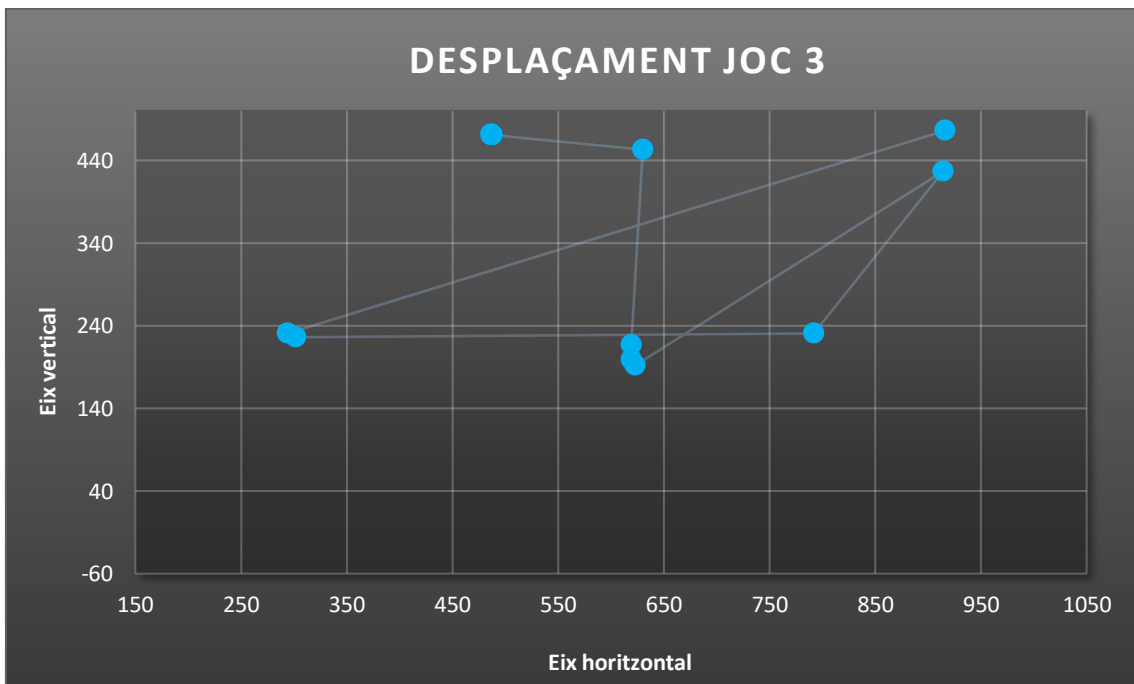
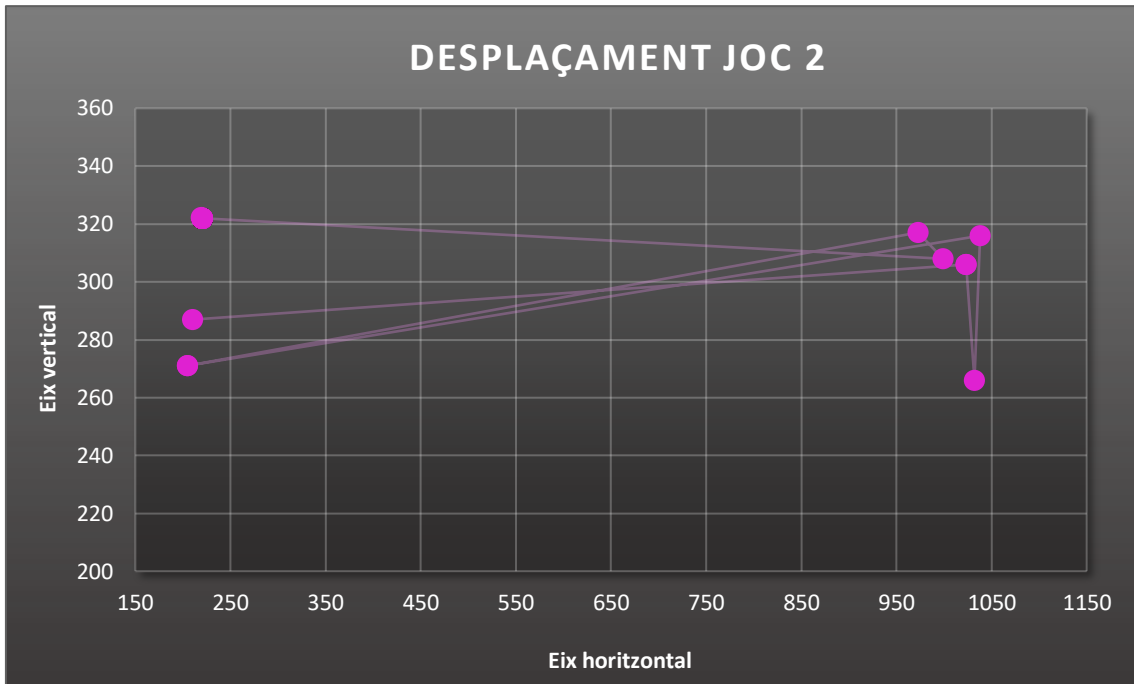
Dades	
Nom i cognoms:	Judith Repullo Bellido
Edat:	12
Patologia:	---
Número de sessió:	---





Dades	
Nom i cognoms:	Francisca Bellido García
Edat:	53
Patologia:	Artrosi
Número de sessió:	---





## 11.3 Codis del joc.

### 11.3.1 Codis Joc 1.

#### Countdownbolas

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class Countdownbolas : MonoBehaviour
{
    void Start()
    {
        this.gameObject.SetActive(true);
    }
    public void SetCountDownBolas()
    {
        JManager.instance.countDownDone = true;
        JManager.instance.StartCoroutine("Timer");
        this.gameObject.SetActive(false);
    }
}
```

#### Jmanager

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;

public class JManager : MonoBehaviour
{
    public static JManager instance;
    // timer
    public int playtime = 60;
    int seconds, minutes;
    public bool countDownDone;

    void Awake()
    {
        instance = this;
    }

    void Start()
    {

```

```

    //scoreToReach = baseScore;
    //Puntos.scoreToReach = scoreToReach;
    // UI.instance.UpdateUI(0, scoreToReach);
    //StartCoroutine("Timer");
}
IEnumerator Timer()
{
    while (playtime > 0)
    {
        yield return new WaitForSeconds(1);
        playtime--;
        seconds = playtime % 60;
        minutes = playtime / 60 % 60;
        // Deberiamos actualizar el UI
        UIbolos.instance.UpdateTime(minutes, seconds);
    }
    Debug.Log("El tiempo se ha acabado");
    //Condicion para ganar
    //CheckForWin();
    Puntosholder.scorevalue = Puntosbolos.scorevalue;
    SceneManager.LoadScene(5);
}
}

```

### Generadorbola

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class Generadorbola : MonoBehaviour
{
    public GameObject bola;
    // Start is called before the first frame update
    void Start()
    {
    }
    // Update is called once per frame
    void Update()
    {
        if (Input.GetKeyDown(KeyCode.Space))
        {
            Instantiate(bola);
        }
    }
}

```

## **Bola**

```
using System;
using UnityEngine;

//these are required for the script to work.
[RequireComponent(typeof(Rigidbody))]
[RequireComponent(typeof(LineRenderer))]

public class Bola : MonoBehaviour
{
    //privates
    private float currentDistance; //the distance from the mouse and ball
    private float goodSpace; //the right amount of space between 0 - max distance
    private float shootPower; //the power when release mouse click
    private Vector3 shootDirection; //derection to shoot
    private LineRenderer line; //use to generate line
    private Transform MainCam; //use to make camera follow ball
    private RaycastHit hitInfo; //for raycasting. enable mouse position to be 3D and not 2D
    private Vector3 currentMousePosition; //the current mouse position
    private Vector3 temp;

    //publics
    [Header("The Layer/layers the floors are on")]
    public LayerMask groundLayers;
    [Header("Max pull distance")]
    public float maxDistance = 3f;
    [Header("Power")]
    public float power;
    [Header("The colors for your Line Renderer")]
    public Color StartColor;
    public Color EndColor;
    [Header("Custom camera offset")]
    public Vector3 CamOffset = new Vector3(0,7,-15);
    [Header("Should the camera follow the ball?")]
    public bool CameraFollowBall;
    [Header("Camera follow speed")]
    public float smoothSpeed = 10.0f;
    public float speed = 1f;

    private void Awake()
    {
        //lets find what we need in the game and components
        line = GetComponent<LineRenderer>(); //set line renderer
    }
}
```

```

//get the main camera with tag if follow ball
MainCam = GameObject.FindGameObjectWithTag("MainCamera").transform;
}

private void OnMouseDown()
{
    //enable toe first point of the line
    line.enabled = true;
    //the line begins at this target position
    line.SetPosition(0, transform.position);
}

private void OnMouseDrag()
{
    currentDistance = Vector3.Distance(currentMousePosition * speed, transform.position);
//update the current distance
    //lets make sure we dont go pass max distance
    if (currentDistance <= maxDistance)
    {
        temp = currentMousePosition * speed; //saving the current possion while dragin is
allowed
        goodSpace = currentDistance;
        line.startColor = StartColor; //set the starting color of the line
    }
    else
    {
        temp = new Vector3(currentMousePosition.x*speed, currentMousePosition.y*speed,
temp.z); // dont go any further
        goodSpace = maxDistance;
    }
    //assign the shoot power and times it by your desired power
    shootPower = Mathf.Abs(goodSpace) * power;

    //get mouse position over the floor - when we drag the mouse position will be allow the x
y and Z in 3D :) Yay!
    Ray ray = Camera.main.ScreenPointToRay(Input.mousePosition);
    if(Physics.Raycast(ray, out hitInfo, Mathf.Infinity, groundLayers))
    {
        currentMousePosition = new Vector3(hitInfo.point.x, transform.position.y,
hitInfo.point.z);
    }

    //calculate the shoot Direction
    shootDirection = Vector3.Normalize(currentMousePosition - transform.position);
    //handle line drawing and colors

```

```

    ///update the line while we drag
    line.SetPosition(1, temp);
}

private void OnMouseUp()
{
    Vector3 push = shootDirection * shootPower * -1; //force in the correct direction
    GetComponent<Rigidbody>().AddForce(push, ForceMode.Impulse);
    line.enabled = false; //remove the line
}

private void LateUpdate()
{
    if(CameraFollowBall)
    {
        //position for camera to follow with offset.
        Vector3 desiredPosition = transform.position + CamOffset;
        // camera follow balll
        MainCam.position = desiredPosition;
    }
    //change color gradually base of how far you drag
    line.endColor = Color.Lerp(StartColor, EndColor, currentDistance / maxDistance);
}
}

```

### **Planodestructor**

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class Planodestructor : MonoBehaviour
{
    private void OnCollisionEnter(Collision collision)
    {
        if(collision.gameObject.tag == "Bola")
        {
            Destroy(collision.gameObject);
        }
    }
}

```

**Destructorbola**

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class Destructorbola : MonoBehaviour
{
    public GameObject fx_stars;
    private void OnCollisionEnter(Collision collision)
    {
        if (collision.gameObject.tag == "Bola")
        {
            Instantiate(fx_stars);
            Puntosbolas.scorevalue += 10;
            Destroy(collision.gameObject);
        }
    }
}
```

**Puntosbolas**

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;

public class Puntosbolas : MonoBehaviour
{
    public static int scorevalue = 0;
    Text score;
    public float Puntosreach = 50;
    // Start is called before the first frame update
    void Start()
    {
        score = GetComponent<Text>();
    }
    // Update is called once per frame
    void Update()
    {
        score.text = "Puntos: " + scorevalue + "/" + Puntosreach;
    }
}
```

### UIbolas

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;

public class UIbolas : MonoBehaviour
{
    public static UIbolas instance;
    public Text timeText;
    private void Awake()
    {
        instance = this;
        UpdateTime(1, 0);
    }
    public void UpdateTime(int mins, int secs)
    {
        timeText.text = mins.ToString("D2") + ":" + secs.ToString("D2");
    }
}
```

### Puntosholder

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
public class Puntosholder
{
    public static int scorevalue { get; set; }
}
```

### Finjuegobola

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;
using UnityEngine.SceneManagement;

public class Finjuegobola : MonoBehaviour
{
    public Text puntosText;
    void Start()
    {
        puntosText.text = "Tu puntuación :" + Puntosholder.scorevalue;
    }
    void Update()
    {
    }
}
```

### 11.3.2 Codis Jocs 2 i 3.

#### Countdown

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class Countdown : MonoBehaviour
{
    void Start()
    {
        this.gameObject.SetActive(true);
    }
    public void SetCountDown()
    {
        JuegoManager.instance.countDownDone = true;
        JuegoManager.instance.StartCoroutine("PlayTimer");
        this.gameObject.SetActive(false);
    }
}
```

#### JuegoManager

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;

public class JuegoManager : MonoBehaviour

{
    public static JuegoManager instance;

    // timer
    public int playtime = 60;
    int seconds, minutes;

    //puntuacion para superar el nivel
    public int baseScore = 50;
    int scoreToReach;

    public bool countDownDone;

    void Awake()
    {
```

```
instance = this;
}

void Start()

{
    scoreToReach = baseScore;
    Puntos.scoreToReach = scoreToReach;
    UI.instance.UpdateUI(0, scoreToReach);
    //StartCoroutine("PlayTimer");
}
IEnumerator PlayTimer()
{
    while(playtime > 0)
    {
        yield return new WaitForSeconds(1);
        playtime--;
        seconds = playtime % 60;
        minutes = playtime / 60 % 60;
        // Deberiamos actualizar el UI
        UI.instance.UpdateTime(minutes, seconds);
    }
    Debug.Log("El tiempo se ha acabado");
    //Condicion para ganar
    CheckForWin();
}

void CheckForWin()
{
    if(Puntos.ReadScore() >= scoreToReach)
    {
        Debug.Log("Has ganado!");
    }
    else
    {
        //Game over
        //Debug.Log("Prueba otra vez!");
    }
    Puntuaciónholder.puntos = Puntos.ReadScore();
    SceneManager.LoadScene("Findeljuego");
}
}
```

**InputManager**

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class InputManager : MonoBehaviour
{
    public GameObject fx_destroy;

    void Update()
    {
        if (Input.GetMouseButtonDown(0))
        {
            //Traza un rayo imaginario desde la camara al puntero, para detectar colision o no.
            Ray ray = Camera.main.ScreenPointToRay(Input.mousePosition);
            RaycastHit hit;

            if(Physics.Raycast(ray,out hit))
            {
                if(hit.collider.tag == "Fuego")
                {
                    Instantiate(fx_destroy, hit.point, Quaternion.identity);

                    FuegoBehaviour molen =
hit.collider.gameObject.GetComponent<FuegoBehaviour>();
                    molen.SwitchCollider(0);
                    molen.anim.SetTrigger("hit");

                    Debug.Log(hit.collider.gameObject + "got hit");
                }
            }
        }
    }
}
```

### Generador

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class Generador : MonoBehaviour {

    public GameObject[] fuegos;
    public bool hasFire;

    void Start()
    {
        if(!hasFire) // Controlamos el tiempo de reaparición
        {
            Invoke("Spawn", Random.Range(0f, 7f));
        }
    }

    void Spawn()
    { // Aqui se genera un numero al azar para que aparezcan los fuegos, el rango va de 0 a n fuegos.
        //Length es igual a 3. Fuego pequeño, grande y un objeto vacio ( transparente ).

        if(!hasFire && JuegoManager.instance.countDownDone == true)
        {
            int num = Random.Range(0, fuegos.Length);

            GameObject fuego = Instantiate(fuegos[num], transform.position, Quaternion.identity)
as GameObject;
            fuego.GetComponent<FuegoBehaviour>().myParent = this.gameObject;
            hasFire = true;
        }

        Invoke("Spawn", Random.Range(0f, 7f));
    }
}
```

**FuehoBehaviour**

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class FuegoBehaviour : MonoBehaviour
{
    Collider col;
    public int hitPoints = 1;
    public int puntos = 1;
    [HideInInspector]public GameObject myParent;

    [HideInInspector] public Animator anim;

    public GameObject popupText;

    void Start()
    {
        anim = GetComponent<Animator>();
        col = GetComponent<Collider>();
        col.enabled = false;
    }
    public void DesruirF()
    {
        myParent.GetComponent<Generador>().hasFire = false;
        Destroy(gameObject);
    }

    public void SwitchCollider(int num)
    {
        col.enabled = (num == 0) ? false : true;
    }
    // Para contar puntos
    public void GotHit()
    {
        hitPoints--;

        if(hitPoints > 0)
        {
            col.enabled = true;
        }

        else
        {
            myParent.GetComponent<Generador>().hasFire = false;
        }
    }
}
```

```
Puntos.AddScore(puntos);

GameObject pop = Instantiate(popupText) as GameObject;

pop.transform.SetParent(UI.instance.transform, false);
pop.transform.position = Camera.main.WorldToScreenPoint(transform.position);

PopupText popText = pop.GetComponent<PopupText>();
popText.ShowText(puntos);

Destroy(gameObject);
}
}
}
```

### **FXDestroy**

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class FXDestroy : MonoBehaviour
{
    public float destroyTime = 1.2f;
    // Start is called before the first frame update
    void Start()
    {
        Destroy(gameObject, destroyTime);
    }
}
```

**PopupText**

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

using UnityEngine.UI;

public class PopupText : MonoBehaviour
{
    public Text popupText;
    Animator anim;
    void Start()
    {
        anim = GetComponentInChildren<Animator>();
        AnimatorClipInfo[] info = anim.GetCurrentAnimatorClipInfo(0);
        Destroy(gameObject, info[0].clip.length);
    }
    public void ShowText(int amount)
    {
        popupText.text = (amount > 0) ? "+" + amount : amount.ToString();
    }
}
```

**Puntos**

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class Puntos : MonoBehaviour
{
    static int puntos;
    public static int scoreToReach;

    public static void AddScore(int amount)
    {
        puntos += amount;
        Debug.Log("Puntos actuales:" + puntos);
        UI.instance.UpdateUI(puntos, scoreToReach);
    }
    public static int ReadScore()
    {
        return puntos;
    }
}
```

## UI

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

using UnityEngine.UI;

public class UI : MonoBehaviour
{
    public static UI instance;

    public Text scoreText;
    public Text timeText;

    private void Awake()
    {
        instance = this;
        UpdateTime(1, 0);
    }
    public void UpdateTime(int mins,int secs)
    {
        timeText.text = mins.ToString("D2") + ":" + secs.ToString("D2");
    }
    public void UpdateUI(int score, int scoreToReach)
    {
        scoreText.text = "Puntos :" + score + "/" +scoreToReach;
    }
}
```

## Puntuaciónholder

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class Puntuaciónholder : MonoBehaviour
{
    public static int puntos { get; set; }
}
```

**Findeljuegopantalla**

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

using UnityEngine.UI;
using UnityEngine.SceneManagement;

public class Findeljuegopantalla : MonoBehaviour

{
    public Text scoreText;
    // Start is called before the first frame update
    void Start()
    {
        scoreText.text = "Tu puntuación : " + Puntuaciónholder.puntos;
    }

    // Update is called once per frame
    void Update()
    {

    }
}
```

**Volvermenu**

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine.SceneManagement;
using UnityEngine;

public class Volvermenu : MonoBehaviour
{
    public void Menu()
    {
        SceneManager.LoadScene(0);
    }
}
```

### 11.3.3 Codis Joc 4.

#### SceneController

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;

public class SceneController : MonoBehaviour {

    public const int gridRows = 2;
    public const int gridCols = 4;
    public const float offsetX = 4f;
    public const float offsetY = 5f;
    [SerializeField] private MainCard originalCard;
    [SerializeField] private Sprite[] images;
    private void Start()
    {
        Vector3 startPos = originalCard.transform.position; //The position of the first card. All
other cards are offset from here.

        int[] numbers = { 0, 0, 1, 1, 2, 2, 3, 3};
        numbers = ShuffleArray(numbers); //This is a function we will create in a minute!

        for(int i = 0; i < gridCols; i++)
        {
            for(int j = 0; j < gridRows; j++)
            {
                MainCard card;
                if(i == 0 && j == 0)
                {
                    card = originalCard;
                }
                else
                {
                    card = Instantiate(originalCard) as MainCard;
                }
                int index = j * gridCols + i;
                int id = numbers[index];
                card.ChangeSprite(id, images[id]);
                float posX = (offsetX * i) + startPos.x;
                float posY = (offsetY * j) + startPos.y;
                card.transform.position = new Vector3(posX, posY, startPos.z);
            }
        }
    }
    private int[] ShuffleArray(int[] numbers)
    {
        int[] newArray = numbers.Clone() as int[];
        for(int i = 0; i < newArray.Length; i++)
    
```

```
{
    int tmp = newArray[i];
    int r = Random.Range(i, newArray.Length);
    newArray[i] = newArray[r];
    newArray[r] = tmp;
}
return newArray;
}
private MainCard _firstRevealed;
private MainCard _secondRevealed;
private int _score = 0;
[SerializeField] private TextMesh scoreLabel;

public bool canReveal
{
    get { return _secondRevealed == null; }
}
public void CardRevealed(MainCard card)
{
    if(_firstRevealed == null)
    {
        _firstRevealed = card;
    }
    else
    {
        _secondRevealed = card;
        StartCoroutine(CheckMatch());
    }
}
private IEnumerator CheckMatch()
{
    if(_firstRevealed.id == _secondRevealed.id)
    {
        _score++;
        scoreLabel.text = "Puntuación : " + _score;
    }
    else
    {
        yield return new WaitForSeconds(0.5f);

        _firstRevealed.Unreveal();
        _secondRevealed.Unreveal();
    }
    _firstRevealed = null;
    _secondRevealed = null;
}
public void Restart()
{
    SceneManager.LoadScene("Scene_001");
}
}
```

## MainCard

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class MainCard : MonoBehaviour {

    [SerializeField] private SceneController controller;
    [SerializeField] private GameObject Card_Back;

    public void OnMouseDown()
    {
        if(Card_Back.activeSelf && controller.canReveal)
        {
            Card_Back.SetActive(false);
            controller.CardRevealed(this);
        }
    }

    private int _id;
    public int id
    {
        get { return _id; }
    }

    public void ChangeSprite(int id, Sprite image)
    {
        _id = id;
        GetComponent<SpriteRenderer>().sprite = image; //This gets the sprite renderer
component and changes the property of it's sprite!
    }

    public void Unreveal()
    {
        Card_Back.SetActive(true);
    }
}
```

## 11.3.4 Codis exportació de dades.

### Input Mouse

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class InputMouse : MonoBehaviour
{
    // Update is called once per frame
    void Start()
    {
    }
    // Update is called once per frame
    void Update()
    {
        if (Input.GetMouseButtonDown(0))
        {
            Debug.Log("Botón derecho apretado");
            print(Input.mousePosition);
        }
        if (Input.GetMouseButtonUp(0))
        {
            Debug.Log("Botón derecho levantado");
            print(Input.mousePosition);
        }
    }
}
```

### **DataExport1**

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using System.Text;
using System.IO;
using System;

public class DataExport1 : MonoBehaviour
{
    private List<Vector2> mousePositions = new List<Vector2>();

    void Update()
    {
        if (Input.GetMouseButtonDown(0))
            mousePositions.Add(Input.mousePosition);

        if (Input.GetMouseButtonUp(0))
            mousePositions.Add(Input.mousePosition);
    }

    void OnDisable()
    {
        ExportMousePositions();
    }

    void ExportMousePositions()
    {
        string path =
System.IO.Path.Combine(System.Environment.GetFolderPath(System.Environment.SpecialFolder.Desktop), "CoordenadasJuego1.csv");
        System.IO.File.WriteAllLines(path, mousePositions.ConvertAll(position => position.x + "," +
position.y));
    }
}
```

**DataExport2/3**

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using System.Text;
using System.IO;
using System;

public class DataExport2 : MonoBehaviour
{
    private List<Vector2> mousePositions = new List<Vector2>();

    void Update()
    {
        if (Input.GetMouseButtonDown(0))
            mousePositions.Add(Input.mousePosition);
    }

    void OnDisable()
    {
        ExportMousePositions();
    }

    void ExportMousePositions()
    {
        string path =
System.IO.Path.Combine(System.Environment.GetFolderPath(System.Environment.SpecialFolder.Desktop), "CoordenadasJuego2.csv");
        System.IO.File.WriteAllLines(path, mousePositions.ConvertAll(position => position.x + "," + position.y));
    }
}
```

### 11.3.5 Codis varis.

#### Menu

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;
public class Menu : MonoBehaviour
{
    public void Juego1()
    {
        SceneManager.LoadScene(1);
    }
    public void Juego2()
    {
        SceneManager.LoadScene(2);
    }
    public void Juego3()
    {
        SceneManager.LoadScene(3);
    }
    public void QuitGame()
    {
        Application.Quit();
    }
}
```

#### Volvermenu

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine.SceneManagement;
using UnityEngine;
public class Volvermenu : MonoBehaviour
{
    public void Menu()
    {
        SceneManager.LoadScene(0);
    }
    public void Reset()
    {
        SceneManager.LoadScene(6);
    }
}
```

**CursorN**

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class CursorN : MonoBehaviour
{
    public Texture2D cursorTex;
    public int cursorSize = 63;
    int sizeX;
    int sizeY;

    void Awake()
    {
        sizeX = cursorSize;
        sizeY = cursorSize;
    }

    void Start()
    {
        Cursor.visible = false;
    }

    void Update()
    {
        sizeX = cursorSize;
        sizeY = cursorSize;
    }

    void OnGUI()
    {
        GUI.DrawTexture(new Rect(Event.current.mousePosition.x - (cursorSize / 2),
Event.current.mousePosition.y - (cursorSize / 2 ), sizeX, sizeY), cursorTex);
    }
}
```

## BGSoundScript

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class BGSoundScript : MonoBehaviour {

    // Use this for initialization
    void Start () {

    }

    //Play Global
    private static BGSoundScript instance = null;
    public static BGSoundScript Instance
    {
        get { return instance; }
    }

    void Awake()
    {
        if (instance != null && instance != this)
        {
            Destroy(this.gameObject);
            return;
        }
        else
        {
            instance = this;
        }

        DontDestroyOnLoad(this.gameObject);
    }
    //Play Gobal End

    // Update is called once per frame
    void Update () {

    }
}
```

**Controlinstrucciones**

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;

public class Controlinstrucciones : MonoBehaviour
{
    public void Instrucciones1()
    {
        SceneManager.LoadScene(7);
    }
    public void Instrucciones2()
    {
        SceneManager.LoadScene(8);
    }
    public void Instrucciones3()
    {
        SceneManager.LoadScene(9);
    }
    public void Instrucciones4()
    {
        SceneManager.LoadScene(10);
    }
}
```