

Chessly

Joan Segovia Millan

Grau en Enginyeria Informàtica

TFG - Videojocs

Consultor: Nwdd Garcia Romero

Professor responsable de l'assignatura: Joan Arnedo Moreno

06 de Juny del 2021



Aquesta obra està subjecta a una llicència de [Reconeixement-NoComercial-SenseObraDerivada 3.0 Espanya de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

FITXA DEL TREBALL FINAL

Títol del treball:	<i>Chessly</i>
Nom de l'autor:	<i>Joan Segovia Millan</i>
Nom del consultor/a:	<i>Nwdd Garcia Romero</i>
Nom del PRA:	<i>Joan Arnedo Moreno</i>
Data de lliurament (mm/aaaa):	<i>06/2021</i>
Titulació o programa:	<i>Grau en Enginyeria Informàtica</i>
Àrea del Treball Final:	<i>TFG - Videojocs</i>
Idioma del treball:	<i>Català</i>
Paraules clau	<i>Videojoc, Escacs, Online</i>
<p>Resum del Treball (màxim 250 paraules): <i>Amb la finalitat, context d'aplicació, metodologia, resultats i conclusions del treball</i></p>	
<p>L'objectiu d'aquest treball és canviar la fórmula clàssica dels jocs d'escacs dotant al resultat d'una variabilitat que el joc centenari no té. Per aconseguir això, el joc inclou un editor de taulers que permet la redistribució de les peces així com un canvi en les dimensions del tauler.</p> <p>A més a més, el joc disposa de diferents modes de joc: individual, a dos jugadors en local i mode on-line. En el joc individual, el jugador podrà provar la seva habilitat contra una IA, amb una dificultat molt reduïda, mentre que en la resta de modes el jugador ho farà contra altres persones, ja sigui en local o per internet.</p> <p>L'editor de partides permet al jugador deixar anar la imaginació i generar un plantejament dels escacs totalment nou i personalitzat. S'han afegit tres peces de jocs d'escacs de diferents llocs del món. El jugador també disposa del joc més clàssic: un tauler 8x8 amb la col·locació de peces tradicional.</p> <p>Per últim remarcar que el joc té una funcionalitat on-line on l'aplicació es comunica amb un servidor AWS d'Amazon configurat amb una API Rest que gestiona els usuaris, les partides i cada moviment que es realitza.</p>	
<p>Abstract (in English, 250 words or less):</p>	
<p>The aim of this work is to change the classic formula of chess games, giving the result a variability that the centenary game does not have. To achieve this, the game includes a board editor that allows the redistribution of the pieces as</p>	

well as a change in the dimensions of the board.

In addition, the game has different game modes: individual, two-player on-premises and online mode. In the individual game, the player will be able to test his ability against an AI, with a very reduced difficulty, whereas in the rest of the modes the player will do it against other people, or in local or by Internet.

The game editor allows the player to unleash their imagination and generate a completely new and personalized approach to chess. Three pieces of chess from different parts of the world have been added. The player also has the most classic game: an 8x8 board with the placement of traditional pieces.

Finally, note that the game has online functionality where the application communicates with an Amazon AWS server configured with a Rest API that manages users, games, and every move that is made.

Índex

1. Introducció.....	2
1.1 Context i justificació del Treball	2
1.2 Objectius del Treball	3
1.3 Enfocament i mètode seguit	3
1.4 Planificació del Treball.....	4
1.5 Breu sumari de productes obtinguts	8
1.6 Breu descripció dels altres capítols de la memòria	8
2. Estat de l'art	10
3. Definició del videojoc.....	11
3.1 Història, ambientació i/o trama	11
3.2 Definició dels personatges/elements	11
3.3 Interacció entre actors del joc.....	11
3.4 Concept Art.....	12
4. Disseny tècnic	14
4.1 Avaluació del <i>engine</i>	14
4.2 Eines utilitzades.....	15
4.3 Diagrama de casos d'ús	15
4.4 Descripció de les classes.....	16
4.5 Diagrama UML de classes.....	18
4.6 Diagrames de flux i de seqüència.....	20
5. Disseny de nivells.....	25
5.1 Editor de taulells.....	25
5.2 Partida off-line (local).....	26
5.2 Partida on-line.....	27
6. Definició de la IA (dummy)	29
7. API Rest	31
8. Requisits.....	33
9. Conclusions.....	34
9.1 Coneixements adquirits	34
9.2 Assoliment d'objectius i seguiment de planificació	34
9.3 Futur del projecte	35
10. Glossari	36
11. Bibliografia.....	37

1. Introducció

1.1 Context i justificació del Treball

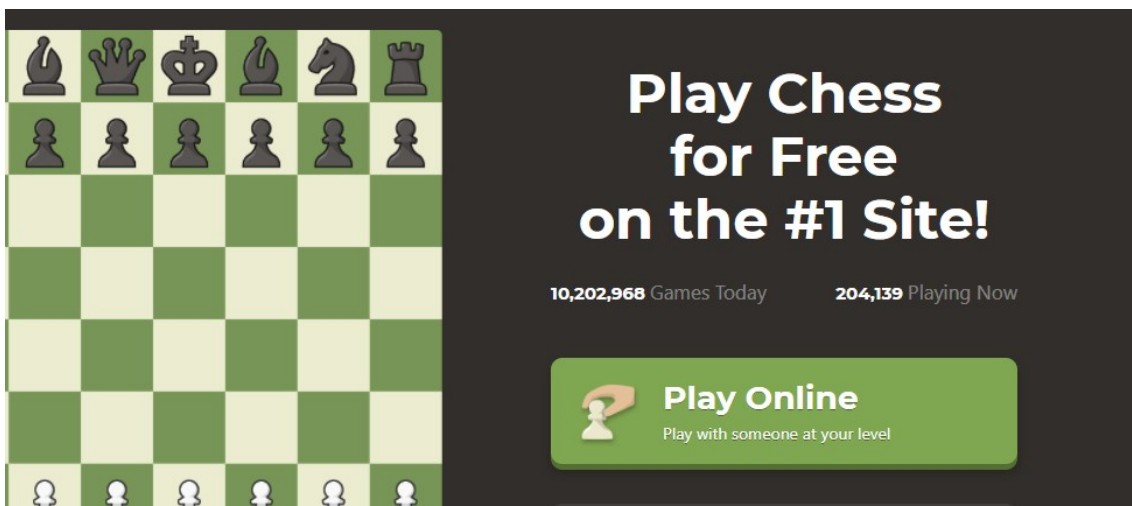
La idea bàsica de **Chessly** és fer un volta al clàssic joc dels escacs. Aquesta idea neix del creixent augment en el interès en aquest joc produït tant per els streamers[1] en Twitch que publiquen partides en directe com per l'aparició d'una sèrie de Netflix dedicada a ell com és "*The Queen's Gambit*"[2].

Fent una mirada al propi Twitch i veient la categoria de Chess, que identifica els streams on es juga a aquest joc amb independència de l'aplicació que s'utilitzi per fer-ho, es pot veure que hi ha un públic que mostra interès en aquestes partides.



Font: Plataforma d'stream Twitch (01/06/2021) [1]

A més, també es veu com l'afició per aquest joc és gran si es para atenció en una de les plataformes per a jugar als escacs on-line més importants que hi ha, Chess.com, organitzadors del campionat d'streamers més important de l'any: Pogchamps; on aquest any ha participat el Rubius entre altres [3]. En aquesta web es pot veure el total de partides diàries que han fet. Fixant-se en els resultats, es veu que es tracta d'un joc molt viu en els temps actuals.



Font: Pantalla d'inici de Chess.com (01/06/2021)

Així doncs, l'objectiu del treball és aprofitar la bona imatge actual del joc per plantejar una proposta més moderna al joc clàssic del segle XV [4]. Per modernitzar el joc tradicional la proposta del projecte és crear un editor de partides que permeti al jugador crear la seva pròpia disposició de tauler i peces així com afegir noves peces que el joc tradicional no disposa.

1.2 Objectius del Treball

El principal objectiu d'aquest treball és crear un joc divertit per a poder jugar amb algun amic als escacs i, a la vegada, poder personalitzar-te la partida al teu gust per sortir del joc clàssic si es vol.

Per aconseguir això, s'han de complir una sèrie d'objectius secundaris:

- Poder jugar a partides d'escacs tradicionals.
- Poder configurar partides amb un tauler i disposició de peces personalitzada.
- Generar partides a dos jugadors.

A més a més, també s'han plantejat altres objectius, en aquest cas no tècnics, pensats més per atraure al públic juvenil del joc:

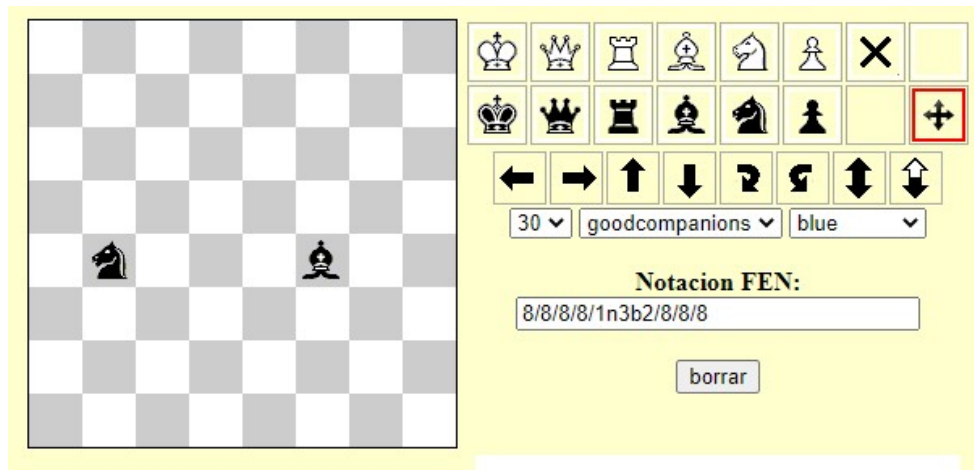
- A de ser un joc minimalista sense gran complicacions en el seu ús.
- Amb una estètica seguint l'estil i la senzillesa del joc.

A tot això s'han de tenir en compte els propis objectius de realitzar aquest treball:

- Aprendre l'ús de l'eina Unity 2D per a la realització de jocs.
- Realitzar la planificació del projecte adaptant-la als inconvenients que van sorgint durant el desenvolupament del mateix.
- Aplicar de forma efectiva els coneixements obtinguts al llarg del grau cursat.

1.3 Enfocament i mètode seguit

Per aquest projecte es va decidir crear un producte nou des de zero basat en les idees i el funcionament del joc físic clàssic, d'algunes aplicacions o webs. En concret per al que fa al joc en si mateix i les partides es va decidir seguir l'estil de la pròpia web Chess.com [3] i del joc Simply Chess [5] mentre que per l'editor de partides es basaria en la web d'Ajedrez Eureka [6].



Font: Editor de taulers d'Ajedrez Eureka

L'enfocament inicial es basava només en la generació de partides amb la configuració tradicional o personalitzada per a partides a dos jugadors en local. La idea era que l'editor tingués més profunditat en els camps editables podent modificar el tauler i les pròpies peces de qualsevol manera.

Al llarg del projecte, en base al feedback amb el consultor, es va decidir limitar més l'editor per donar temps en la planificació al desenvolupament d'un mode de joc individual contra una IA i un mode multijugador on-line.

Com a metodologia en el projecte es va definir en un inici, en la PAC 1, una planificació molt general sense entrar en detall en cada desenvolupament i com a resultat el projecte va quedar definit per una metodologia de cascada on es realitzaven poques iteracions sobre els desenvolupaments i no es permetia gaire flexibilitat els temps.

A partir de la següent entrega, i sobretot després de la introducció en la planificació del mode individual i del on-line, es va optar per a canviar a un mètode més pròxim a *l'agile*. Aquest canvi en la metodologia ha permès adaptar millor el desenvolupament als inconvenients que han anat sorgint al llarg del projecte.

Com a conclusió, es pot dir que el canvi a una metodologia menys estricte ha sigut de molta ajuda per duu a terme el projecte i aconseguir el resultat presentat.

1.4 Planificació del Treball

La planificació del treball s'ha plantejat en quatre parts corresponents cadascuna d'elles amb l'entrega d'una PAC o lliurement de UOC. Així doncs, les dates finals de cada fase del projecte són:

- Final de la Fase 1: 28/02/2021
- Final de la Fase 2: 04/04/2021

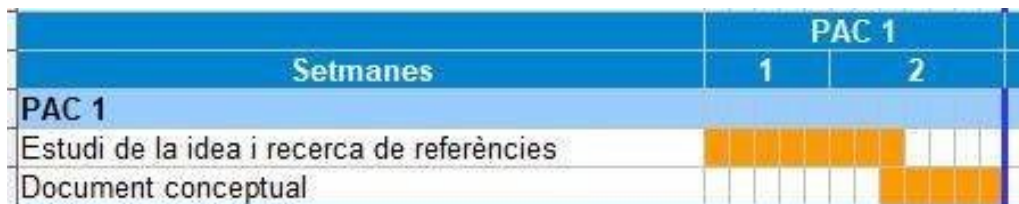
- Final de la Fase 3: 09/05/2021
- Final de la Fase 4 (final del projecte): 06/06/2021

S'ha de tenir en compte que en base el feedback de la entrega de la Fase 1, es va modificar la planificació de les entregues posteriors. A més a més, degut als imprevistos que s'han anat produït al llarg del projecte, l'execució de les tasques ha variat en ordre i en duració. Així doncs, la planificació inicial no correspon amb la final pel que respecte a les Fases 2, 3 i 4.

S'afegeix també com a comparativa un diagrama amb els temps reals dedicats a cada tasca per poder realitzar un estudi entre expectativa i realitat.

1.4.1 Fase 1

Aquesta Fase es va dedicar en la recerca d'un joc a realitzar, la motivació inicial del projecte i la definició d'objectius a complir. Totes aquestes dades són les descrites en els punts 1.1 i 1.2. També es va crear el primer document amb la definició inicial del projecte. No es tractava del Game Design Document (GDD) doncs no contenia una definició dels continguts del joc completa però si assentava les bases del que seria el producte resultant del projecte.



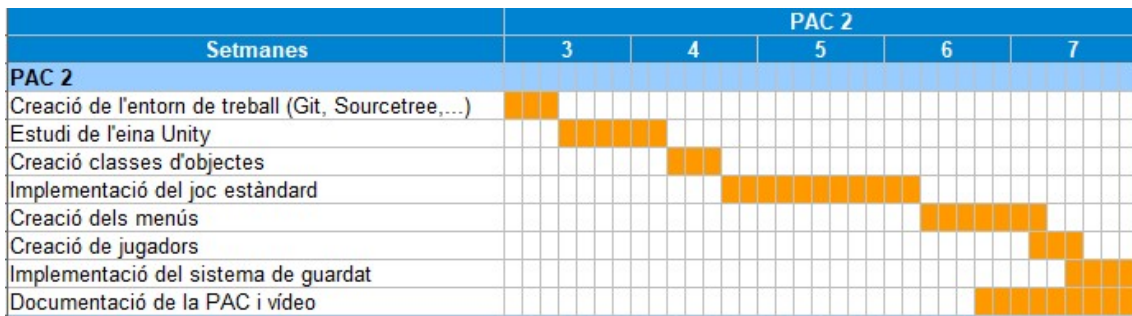
Planificació Fase 1 (PAC1)

1.4.2 Fase 2

Aquesta fase es va dedicar a la creació de l'aplicació com a tal. Aquesta contenia un menú bàsic on es podia seleccionar el mètode de joc i la possibilitat de jugar partides amb la configuració clàssica de tauler i peces.

A més a més, com les eines utilitzades per a la programació del joc no les havia usat mai, es va dedicar part del temps a l'estudi del Unity i de les seves funcionalitats.

En una comparativa entre la planificació inicial i els temps finals de desenvolupament es pot veure com els càlculs de temps a priori no estaven ben ajustats.



Planificació inicial de la Fase 2

Aquest és un diagrama de Gantt amb els temps reals dedicats a cada tasca. A més a més, es pot veure com hi ha tasques que en un inici no es van detectar o que estaven poc definides.

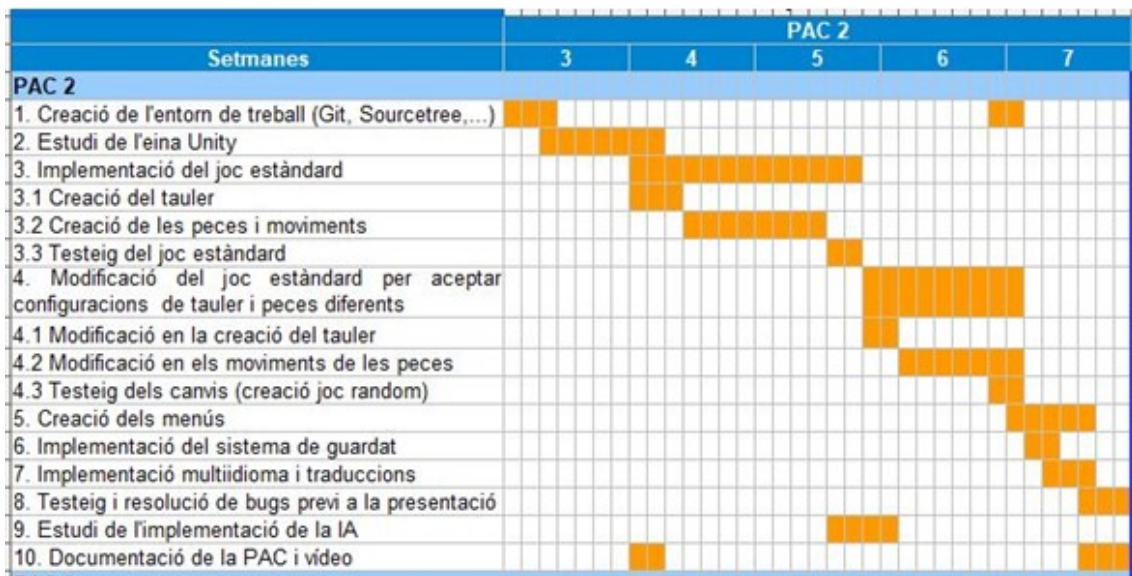
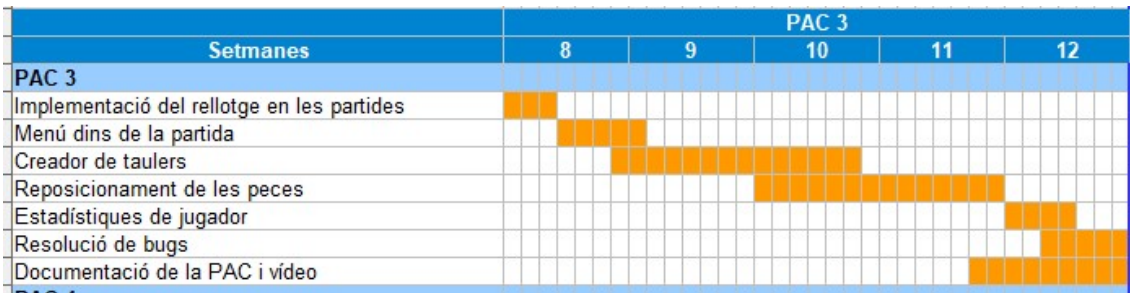


Diagrama de Gantt amb els temps dedicats a cada tasca de la Fase 2

El resultat final de la fase 2 inclou també la creació d'un mode de joc anomenat 'partida a l'atzar', o *random game*. Aquest mode de joc no estava planificat inicialment si no que es va crear per testejar que el funcionament de les peces. Finalment vaig decidir deixar aquesta nova manera de generar partides com un *feature* més del joc com s'explica més endavant.

1.4.1 Fase 3

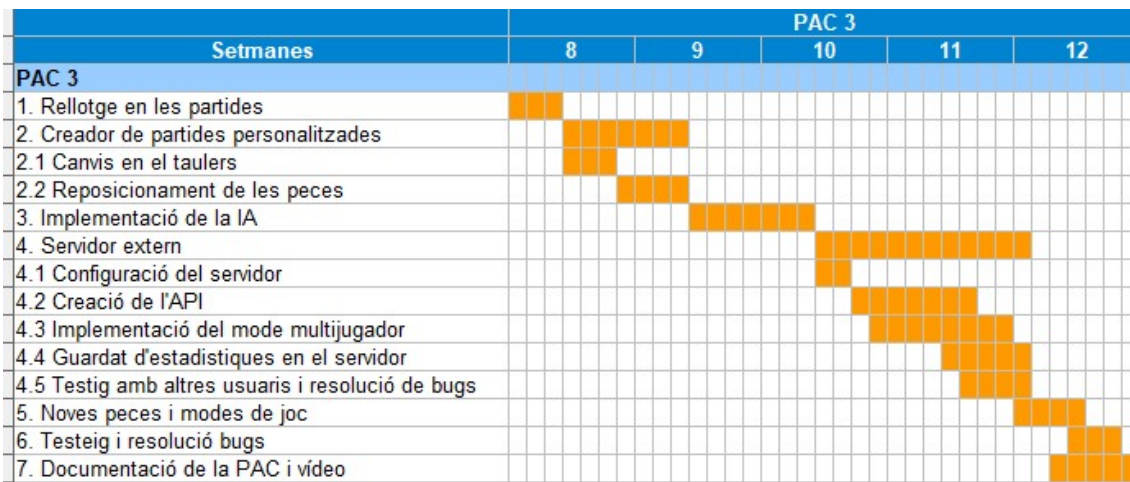
Inicialment aquesta fase s'anava a dedicar al creador de partides personalitzades on un jugador podria generar la configuració de tauler i peces que volgués. Com es pot veure en el diagrama que hi ha a continuació, també es volien afegir un seguit de estadístiques de les partides jugades.



Planificació inicial de la Fase 3

Degut als canvis que es van decidir aplicar amb la introducció del mode de joc individual i les partides on-line, la planificació es va veure afectada. Es van retallar temps de desenvolupament de l'editor per poder dedicar-ho a altres tasques. A més, algunes característiques del joc com són les estadístiques de les partides es van veure modificades. En concret aquestes només contindrien informació sobre les partides on-line.

Així doncs, el diagrama de Gantt actualitzat va quedar de la següent manera.



Planificació de la Fase 3 actualitzada durant la Fase 2

Durant el desenvolupament d'aquestes tasques es van produir alguns imprevistos amb la tasca del joc on-line i els temps dedicats a cada tasca en van veure afectats.

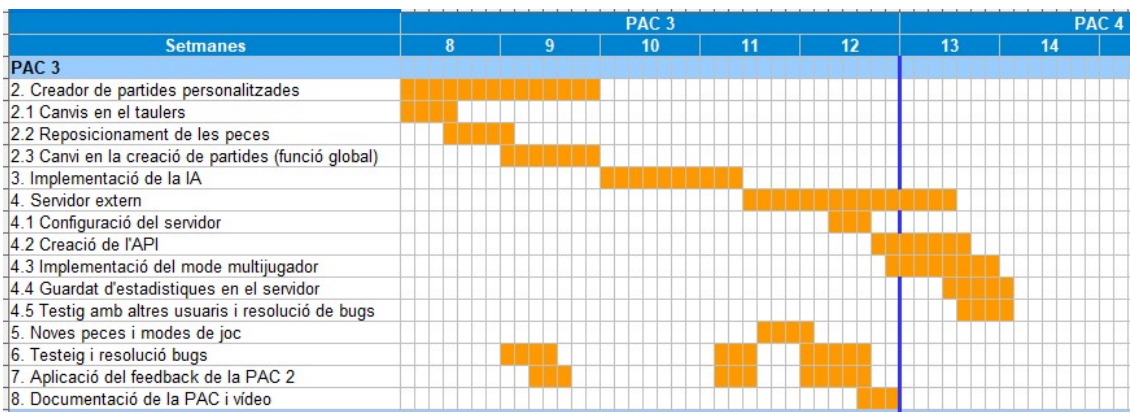
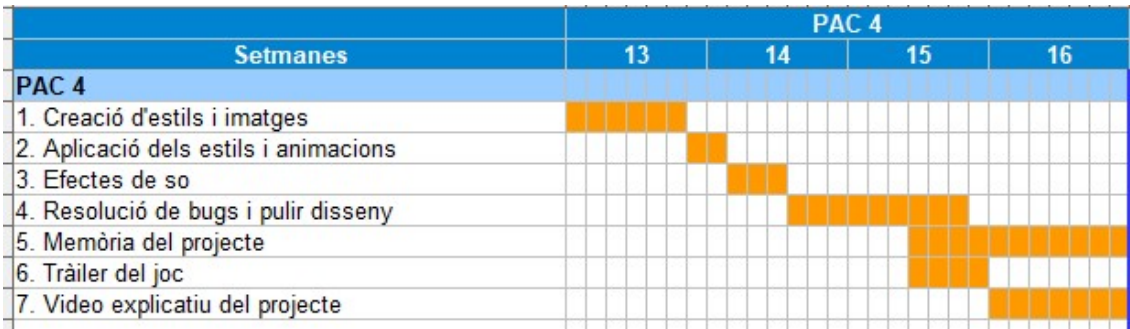


Diagrama de Gantt amb els temps dedicats a cada tasca de la Fase 3

1.4.4 Fase 4

Aquesta última fase es va planificar inicialment com un temps per a realitzar les tasques de prova i polir del joc així com per millorar els estils visuals del joc i el *gamefeel*. Aquesta Fase no es va veure modificada per les tasques afegides en la Fase 2 i els temps de desenvolupament s'han respectat.



Planificació de la Fase 4

1.5 Breu sumari de productes obtinguts

Els productes obtinguts en aquest projecte són els nombrats en el següent llistat:

- Un executable amb el joc operatiu que funciona en Windows.
- Una API Rest creada amb Laravel vinculada a una base de dades SQL Server sobre un Windows Server 2019 en un AWS d'Amazon.
- Aquesta memòria documental
- Un vídeo promocional (tràiler) del joc.
- Un vídeo de presentació del joc on es mostra una explicació del projecte així com *gameplay*.

1.6 Breu descripció dels altres capítols de la memòria

Seguidament s'exposa un llistat de cada capítol d'aquest document amb una breu explicació sobre el seu contingut.

- Capítol 2: Estat de l'art. En aquest capítol es descriu el tipus de joc i gènere al que pertany així com les idees i estils sobre els que es basa.
- Capítol 3: Definició del videojoc. En aquest apartat es defineix el videojoc. Quines característiques té i quines funcionalitats conté.
- Capítol 4: Disseny tècnic. En aquest punt es defineix l'entorn escollit per al projecte i la seva justificació. També s'inclou aquí una definició

de cada classe un diagrama UML amb les relacions així com cassos d'ús del joc.

- Capítol 5: Disseny de nivells. Aquí es descriuen les diverses pantalles on els jugadors podran jugar.
- Capítol 6: Definició de la IA (dummy). Capítol dedicat a la descripció del IA i les seves funcions.
- Capítol 7: API Rest. En aquest capítol s'expliquen alguns punts sobre la creació d'aquesta API i la configuració del servidor que la executa.
- Capítol 8: Conclusions. Breu descripció de l'opinió sobre els resultats del projecte i el desenvolupament del mateix.

2. Estat de l'art

La idea bàsica de **Chessly** és fer un volta al clàssic joc dels escacs. Es tracta d'un joc pensat per a ser jugat per dos jugadors. Els jugadors crearan una partida d'escacs en la que podran jugar per torns com si es tractés del joc físic de taula. El canvi sobre el joc original ve en les possibilitats durant la creació de la partida. Els jugadors podran decidir la mida i forma del tauler i el nombre de peces, tipus i col·locació de cada jugador. Aquests canvis generaran mecàniques i estratègies que en el joc estàndard no són possibles dotant a aquest joc del segle XV de noves perspectives.

Pel que respecte al tipus de videojoc, és difícil classificar els jocs d'escacs com un gènere de videojoc concret doncs alguns ho classifiquen com a jocs d'estratègia mentre que hi ha que creuen que els jocs d'escacs són un gènere independent de la resta. Per altra banda, als jocs multi jugador local per torns basats en partides ràpides se'ls sòl classificar com jocs *party* i als videojocs que reproduïxen jocs de taula com a *tabletop games*.

Així doncs, degut a la idea d'afegir noves regles que surten del joc bàsic dels escacs, es pot dir que la idea de **Chessly** és fer un *party game*.

3. Definició del videojoc

3.1 Historia, ambientació i/o trama

Al tractar-se d'un joc d'escacs realment no hi ha cap historia o ambientació més enllà que la del joc original: dos grans exercits igualats es disputen en una guerra el control del territori i només les capacitats estratègiques dels seus reis podran portar-los a la victòria. El joc en si mateix manca de trama i el videojoc proposat, pensat per a fer partides ràpides tampoc aporta un espai on crear una trama coherent o integrada.

A més a més, l'editor de partides permet trencar la simetria entre jugadors fent que el plantejament inicial dels dos exercits en igualtat de condicions es perdi.

3.2 Definició dels personatges/elements

Realment el joc dels escacs ha canviat molt des de la seva idea original. El joc de taula està basat en el xaturanga [7], un joc molt antic de la India del que no es té constància de tots els moviments originals de les seves peces. Amb el pas dels segles es van definir les peces del joc de taula fins que en el segle XV es van descriure les actuals [4].

Com s'ha descrit anteriorment, a més del joc clàssic, aquest videojoc permet la creació de taulers personalitzats, sortint així del 8x8 estàndard, i també dona la possibilitat de canviar l'ordre i el nombre de peces per a cada jugador. Aprofitant la pròpia evolució de les peces que hi ha hagut durant la historia, aquesta opció de personalització també disposa de la possibilitat d'introduir en la partida algunes antigues peces i els seus moviments així com peces d'altres jocs derivats del mateix antecessor dels escacs com el Xiangqi [8] o Makruk [9].

Pel que fa al joc on-line, el jugador pot crear un perfil des del propi joc o iniciar sessió amb un perfil creat anteriorment. Un cop connectat, pot cercar partides on jugar, crear-ne de noves o consultar les estadístiques de les partides jugades i els resultats obtinguts.

3.3 Interacció entre actors del joc

Com ja ha quedat descrit, el joc serà per torns on cada jugador tindrà la possibilitat de fer un moviment d'una de les seves peces. En aquest aspecte, el joc manca d'interacció real entre actors si no tenim en compte el fet de que una peça mati una altra. Podem dir que aquest és l'únic moment que una acció d'un jugador afecta al joc de l'altre pel que fa al joc local.

Per altra banda, el videojoc al disposar d'un mode on-line, es crea la interacció entre dos jugadors on un pot veure els moviments de l'altre. També

s'estableix un control de duració de torns per a evitar que un jugador quedi a l'espera d'un altre que ja no està connectat.

3.4 Concept Art

La idea del disseny de Chessly és que sigui en 2D, amb un estil simple i minimalista. Es busca dotar al joc d'una imatge no tan seriosa com normalment es veu aquest joc i atraure així un públic més juvenil. Per a l'art del joc s'ha tingut en compte el joc online nombrat anteriorment: Chess.com [3]. D'aquest joc s'ha volgut incorporar la senzillesa del disseny, la vista zenital del tauler i el desplaçament de les peces.



Font: tauler de joc de Chess.com

El disseny de peces està basat en el del joc Simply Chess [5], actualment disponible gratuïtament en Steam, quedant de forma similar a les següents:



Font: Tauler del joc Simply Chess

4. Disseny tècnic

4.1 Avaluació del *engine*

Inicialment, es van cercar eines per poder generar un joc executable que es pogués jugar de forma off-line a més de la resta de funcionalitats de les que disposa Chessly. Per aquest motiu, es va descartar de primeres fer el joc en JQuery, la elecció que personalment hagués pres si el joc fos web ja que és un llenguatge que domino bé, o NodeJs ja que els dos llenguatges estan pensats per un entorn web.

L'objectiu era trobar quines eren les eines més utilitzades doncs al no haver realitzat mai cap projecte així, crec que el fet que l'eina tingui una gran comunitat d'usuaris ajuda a poder trobar solucions als problemes que pugui anar trobant durant el desenvolupament.

De entre les primeres llistes d'eines al Google [10], les més nombrades són:

- GameMaker
- GameSalad
- Stencyl
- Unity
- Unreal Engine

Analitzant-les es pot veure que GameMaker només disposa d'un Free Trial de 30 dies [11] i, per tant, va quedar descartada la idea d'utilitzar-lo.

L'eina GameSalad és també de pagament, sense opció gratuïta per desenvolupadors. A més, tal i com ells es defineixen en la seva *landing page: no-code* [12], doncs està pensat per generar jocs sense haver de programar i, per aquest grau, crec que és indispensable que el projecte contingui alguna part de programació.

L'*engine* Stencyl també està fet amb la idea de crear videojocs sense haver de programar [13] i per aquest motiu també es va descartar.

De les altres dues opcions, l'Unreal Engine semblava massa complex per el joc que es volia fer.

Així doncs, finalment es va decidir fer el projecte amb Unity tot i que potser l'ús de GameMaker seria una millor opció per al tipus de joc que estic realitzant si fos una eina gratuïta.

Per últim, una altra opció hagués sigut crear el joc de zero sense cap eina, només a partir codi amb Visual o altres llenguatges. Es va descartar aquesta idea per inexperiència en aquests llenguatges i en la creació d'aplicacions similars. A més, tot i no fer servir totes les funcionalitats del Unity, la gestió dels objectes i *sprites*, funcionalitats com la gestió d'inputs, tractament del efectes de so, així com l'entorn visual per a la creació de menús i entorn em faciliten el

desenvolupament i permetrien ajustar millor els temps a la planificació del projecte.

4.2 Eines utilitzades

Per desenvolupar aquest videojoc s'ha utilitzat l'eina Unity, més concretament la versió **Unity 2019.4.15f1**. Una versió gratuïta de l'eina. Unity és àmpliament coneguda en el sector del videojocs, especialment en el àmbit dels jocs *indies*. Amb aquesta eina es pot crear un videojoc sense haver de comprar una llicència però, aleshores, apareixerà el logotip de Unity a inici de joc. Per que fa a aquest projecte, no vaig veure la necessitat d'adquirir una llicència per remoure aquest efecte inicial.

Per la programació de codi del joc s'ha usat l'eina **Visual Studio Community 2017**. Mentre que per a programar l'API REST en php s'ha fet servir **Sublime text 3**.

En concret, l'API està programada en **Laravel** sobre un **Windows Server 2019** i connectada a un base de dades **SQL Server**.

Per al disseny dels elements gràfics s'ha fet servir el **Adobe Illustrator** y el **Photoshop CS6**. i pel desenvolupament del tràiler del joc i dels vídeos de cada entrega s'ha usat l'eina **Adobe Premiere**.

Per portar un control de les versions de codi s'ha utilitzat el **Git** amb el programa **Sourcetree**.

4.3 Diagrama de casos d'ús

En base a les necessitats generades pels requisits definits en el joc, s'han creat un seguit de casos d'ús. Aquestes són les accions que un jugador pot duu a terme en videojoc. algunes d'elles es poden executar sobre premisses o amb característiques diferents.



Diagrama de casos d'ús

Tot seguit es defineix cada cas d'ús a alt nivell, és a dir, sense especificar cada detall que el forma, només per donar una visió més general del que comporta el cas concret.

- Jugar una partida contra un altre jugador: El jugador podrà jugar una partida ja sigui en el format tradicional o no i, també, sigui off-line o on-line contra un altre jugador.
- Jugar una partida contra la IA: El jugador podrà jugar una partida ja sigui en el format tradicional o no contra la IA.
- Iniciar sessió on-line: El jugador es podrà connectar al servidor amb l'ús del seu nom d'usuari i la seva contrasenya.
- Crear compte on-line: El jugador podrà crear una compte d'usuari per a jugar on-line.
- Tancar sessió on-line: El jugador es podrà desconnectar del servidor.
- Crear una partida on-line: El jugador podrà crear una partida on-line ja sigui personalitzada o amb un format tradicional.
- Unir-se a una partida on-line: El jugador podrà unir-se a una partida, ja sigui personalitzada o tradicional, per jugar contra un altre jugador a través d'internet.
- Personalitzar partides: El jugador podrà crear partides personalitzades on podrà modificar les mesures del tauler i la distribució de les peces de cada jugador.
- Canviar idioma: El jugador podrà canviar el idioma del joc en funció de les seves preferències.
- Canviar color de peces i del tauler: El jugador podrà canviar el color de les peces i de les caselles del tauler en funció de les seves preferències.

4.4 Descripció de les classes

Per donar una idea general sobre les classes que conformen el codi del videojoc, tot seguit es descriuen cada una d'elles. Les classes usades per l'API en el servidor es detallen el Capítol 7.

- GameButton: aquesta classe conté les funcions relacionades amb els botons del menú del joc.

- Login: conté les funcions per iniciar sessió, crear un usuari per al joc on-line, les consultes a l'API relacionades amb l'obtenció de les estadístiques del jugador i les funcions de control sobre els botons del menú realcionats amb el joc on-line.
- LanguageManager: controla l'obtenció i aplicació de les traduccions en el joc.
- SaveLoadData: controla la càrrega i el guardat d'informació en fitxers.
- EditorManager: conté les funcions per a la creació personalitzada de partides.
- BoardEditor: té les funcions per a gestionar el tauler en l'editor.
- CellEditor: té les funcions per a gestionar les cel·les en l'editor.
- PieceEditorManager: aquesta classe té les funcions que permeten la creació d'una distribució personalitzada de les peces en el tauler del editor.
- CellActionPiece: funcions per a la gestió de les accions que es poden realitzar en una cel·la dins l'editor com per exemple afegir un peça o treure'n una.
- CellVoid: conté la informació sobre la peça que s'ha assignat en una cel·la en l'editor, la seva posició en la graella, el color i el tipus de peça.
- GameManager: conté totes les funcions per executar una partida a partir de les opcions seleccionades en el menú i la informació sobre el tauler i les peces amb que es vol jugar.
- Board: similarment a BoardEditor, té les funcions per a gestionar el tauler de joc durant una partida.
- Cell: de forma similar a CellEditor, té les funcions per a gestionar les cel·les durant una partida: posició, color mida,...
- PieceManager: conté totes les funcions relacionades amb les peces i els jugadors. En concret, aquesta classe controla el torns del jugadors, les peces que segueixen en joc i si la partida ha acabat o no.
- BasePiece: classe abstracta que defineix les funcions de cada peça: moure-la, la interacció amb les altres peces, les caselles a les que es pot moure, etc.
- Classes de cada peça: cada tipus de peça té una classe pròpia on s'especifiquen les particularitats que té: el moviment, les accions especials com l'enderroc, etc.

4.5 Diagrama UML de classes

Aquest és el diagrama de classes del joc Chessly, en concret de l'aplicació d'escriptori. Per simplificar-ne la lectura i la comprensió de les relacions, no es mostra en aquests diagrames ni els atributs i les funcions de cada Class. També per millorar la comprensió, s'han afegit requadres de color que engloben les classes usades en cada Scene.

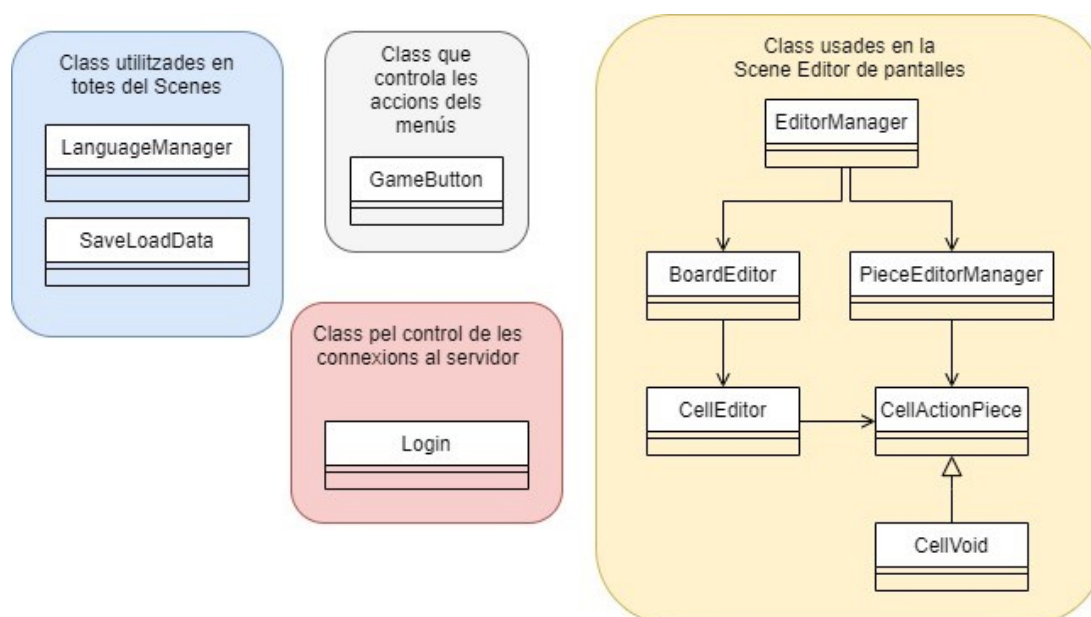


Diagrama 1 UML de classes

A l'estar l'aplicació de Unity dividida en Scenes, és a dir, escenes, cada una d'elles té un conjunt de Scripts que s'executen en funció de les necessitats d'aquesta. Per tant, a l'hora de mostrar la informació en el UML es pot veure com gran part de les Classes no tenen relacions entre elles ja que simplement executen un seguit de funcions independents.

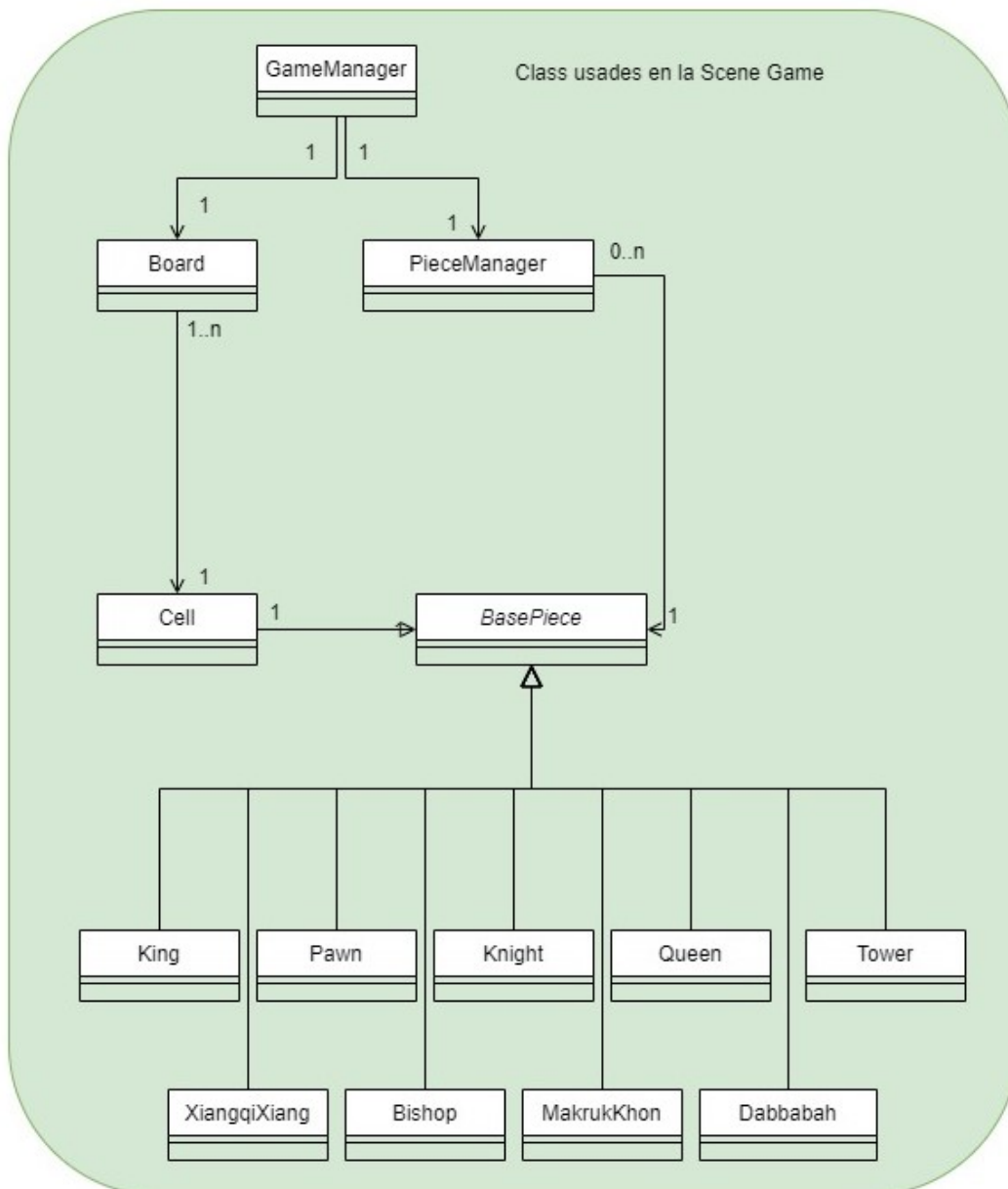


Diagrama 2 UML de classes de la Scene Game

A part d'aquestes classes, per també s'han generat unes estructures de dades per a la gestió de la informació. Aquestes estructures són Class usades per guardar, entre altres, la informació rebuda en algunes crides de la API Rest o les traduccions dels múltiples idiomes. En concret s'han creat les següents:

- OptionsData: Estructura la informació guardada en les opcions per part del jugador. Aquesta informació es conserva al tancar el joc.
- LanguageData: Classifica les traduccions obtingudes del json per aplicar el text traduït.
- VoidPiece: Peça buida creada com a substitució de les peces en l'editor. S'explica el seu ús en el Capítol 5.

- OnlineGameData: Emmagatzema la informació del joc, tauler i peces d'una partida on-line.
- OnlineMoveData: Guarda la informació sobre un moviment en una partida on-line.
- OnlineStatisticsData: Estructura la informació rebuda del servidor amb les estadístiques de joc d'un jugador.

A més de les Class del joc creat amb Unity, també s'han hagut de crear unes per l'API Rest del servidor. Tot i no estar especificat en el propi UML, totes les classes tenen uns atributs timestamp que permeten saber la data de creació, última modificació i eliminació, en concret de soft-delete.

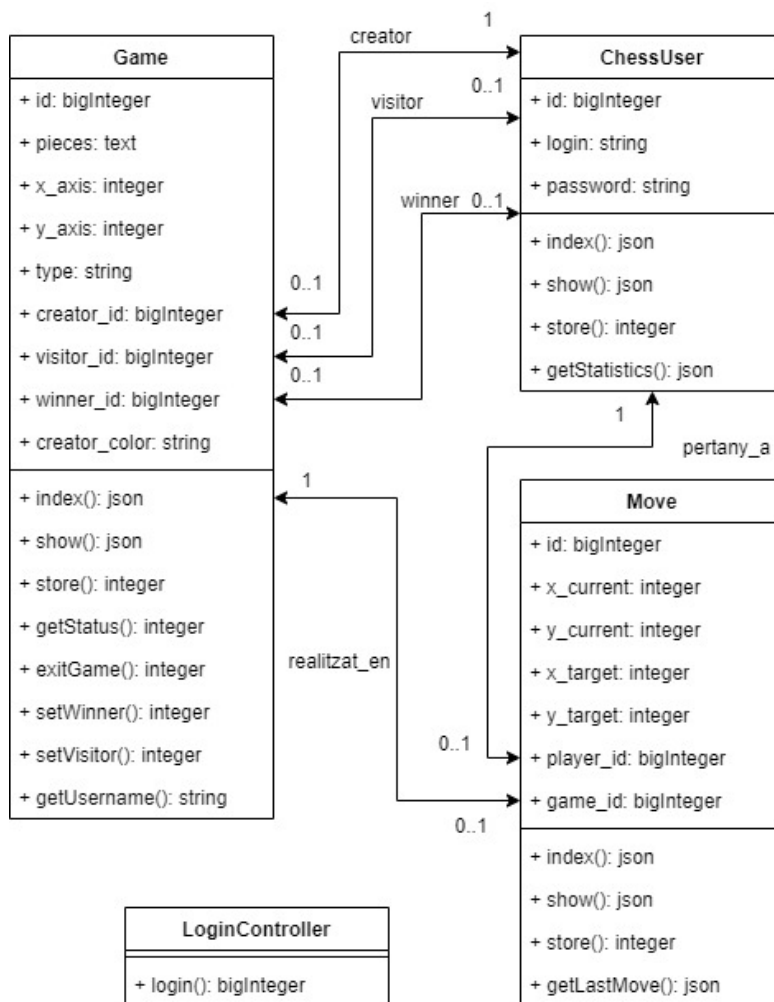


Diagrama 3 UML classes de l'API Rest

4.6 Diagrames de flux i de seqüència

Aquest són els diagrames de flux d'una partida off-line i una on-line. A simple vista es pot veure que la gestió de torns és més complex doncs el joc per

internet requereix de moments d'espera on l'aplicació realitza consultes contínuament al servidor per saber si li toca actualitzar dades o no.

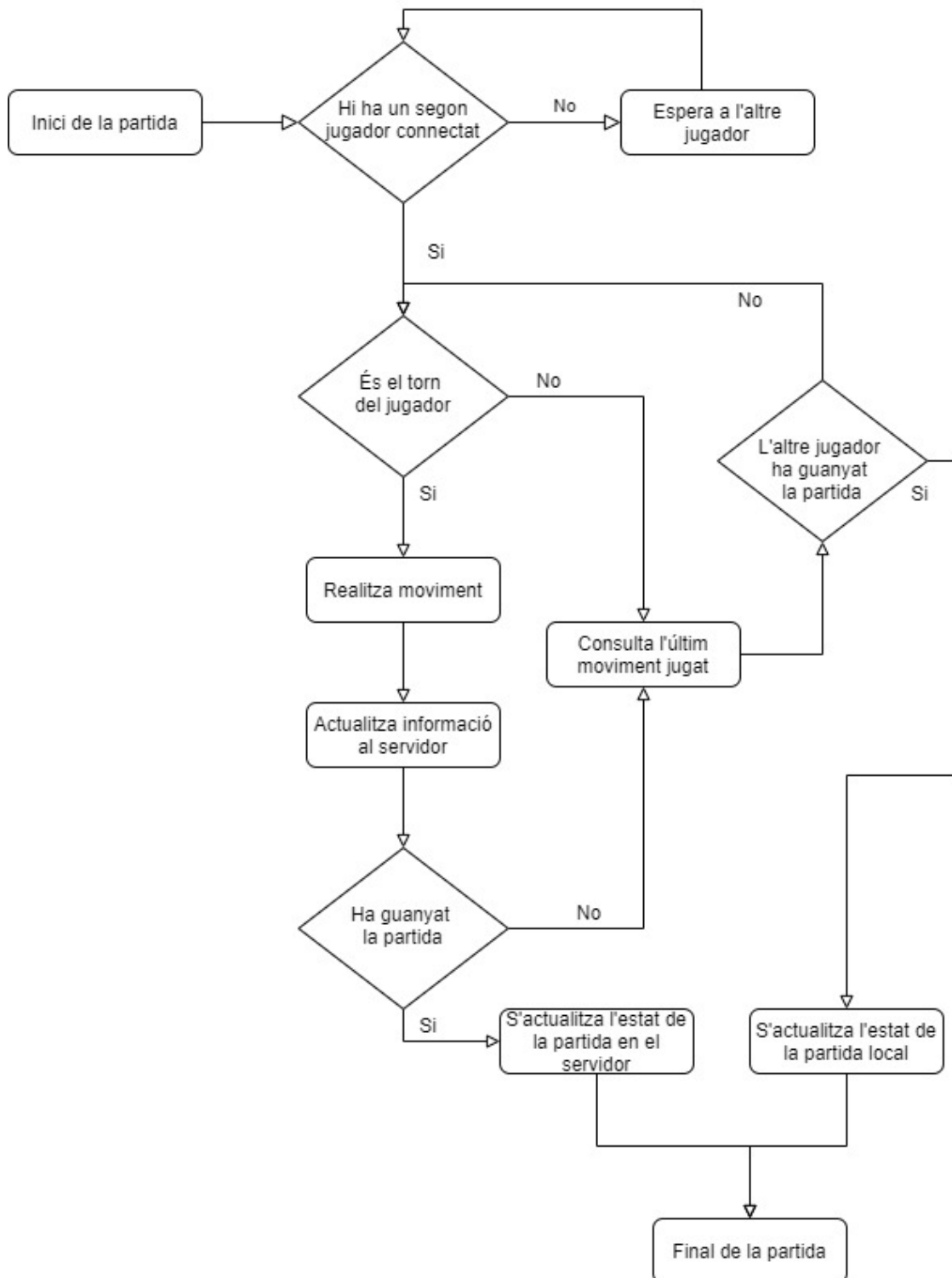


Diagrama de flux d'una partida on-line

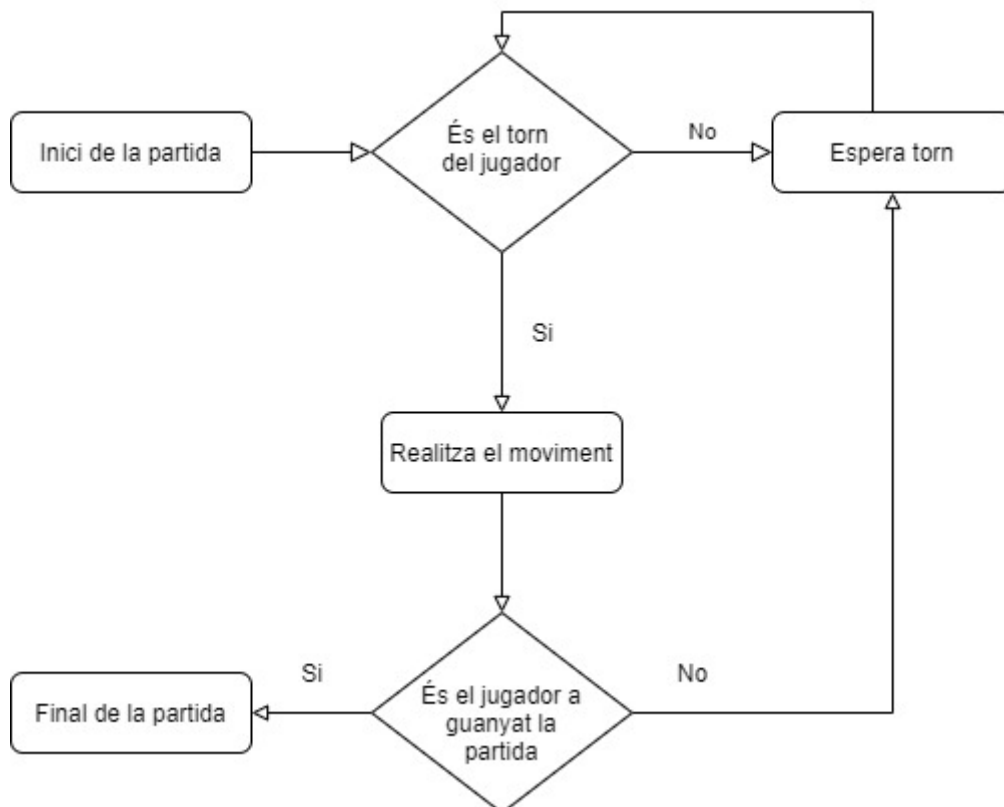


Diagrama de flux d'una partida off-line (en local)

Per entendre millor la seqüència d'accions en una partida on-line, s'adjunta també un diagrama de seqüència simplificat. Primer s'inicialitzaria amb un inici de sessió del jugador per després passar al propi joc. En el segon diagrama es pressuposa que la partida ja està creada i que els jugadors han iniciat sessió.

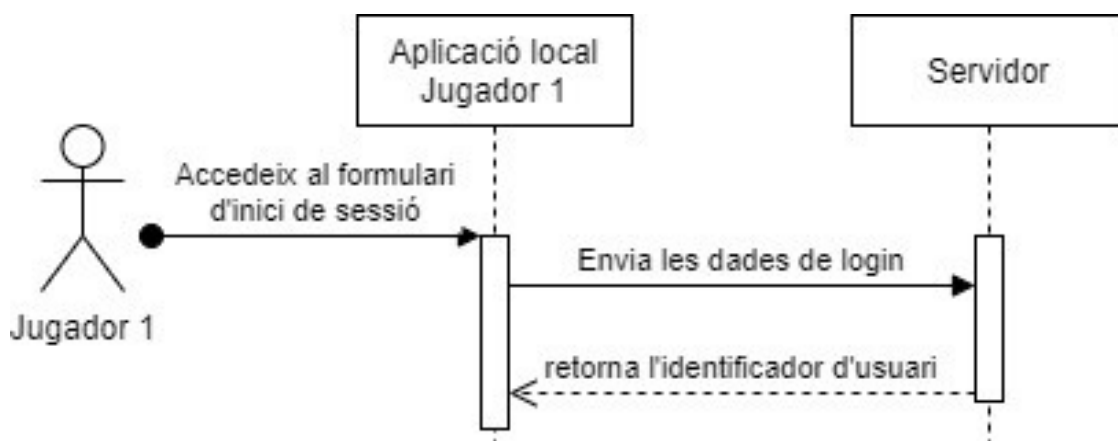


Diagrama de seqüència d'inici de sessió on-line

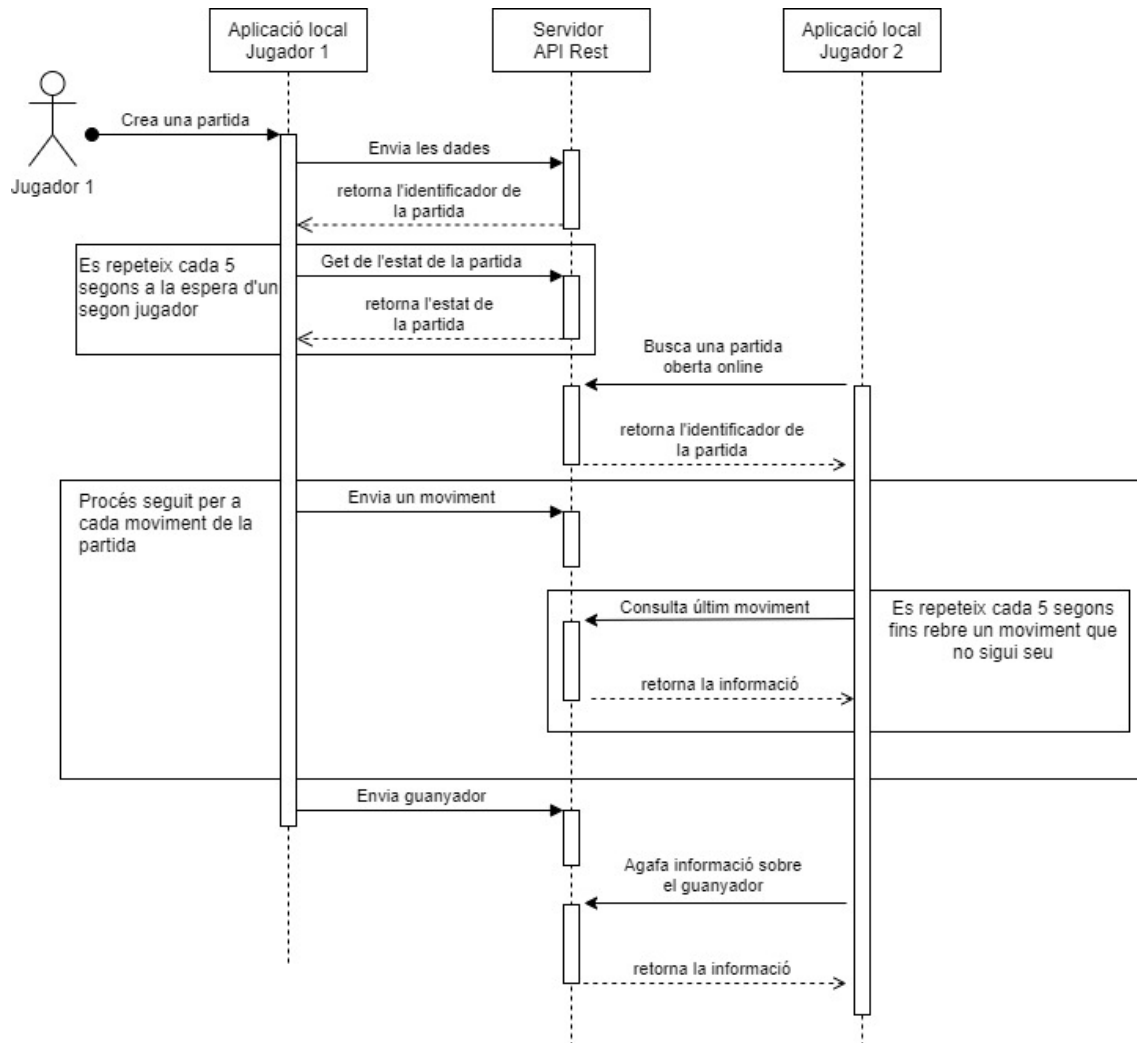


Diagrama de seqüència d'una partida on-line

4.7 Gràfics i so

Com gairebé tots els jocs, Chessly utilitza imatges per a representar els seus elements. Les peces dels escacs així com el *background* dels menús o el logotip del joc són *Sprites* de dins del joc.



Logotip de Chessly

Tant el logotip del joc com el seu nom i les peces tenen d'intenció de semblar modernes mentre que l'estic minimalista del tauler pretén dotar al joc d'un aire juvenil.



Sprites de les peces del videojoc

Pel que fa als efectes de so, aquests són de Freesound [21] i a la música de la web Epidemicsound [22]. En aquesta web es pot trobar música de tots els estils lliure de drets d'autor mentre es pagui la quota mensual. Com el primer mes és gratuït, s'han aprofitat els seus àudios per al joc.

El joc només utilitza quatre cançons i 'ha fet servir les mateixes per al vídeo de presentació i per al tràiler.

5. Disseny de nivells

En Chessly no hi ha nivells pròpiament dit però si podem tractar les partides com si ho fossin. Així doncs, podem definir que l'editor de taulers és en si mateix un editor de nivells del propi joc.

Per altra banda, les partides on-line disposen de variàncies respecte les partides off-line, en local, om ja s'ha vist en els diagrames anteriors i també s'haurien de definir com una varietat de nivells diferent.

5.1 Editor de taulers

Des del menú principal o en el menú que apareix un cop connectat al on-line, el jugador pot triar entrar al editor de taulers. En ell, el jugador selecciona primer de tot les mesures del tauler. Aquestes estan acotades entre un màxim i mínim tan en altura com en amplada essent una entre 5 i 8 i l'altre entre 5 i 14. Un cop seleccionades les mides, es carrega el tauler disposant les cel·les de forma dinàmica. Internament, en el moment de crear el tauler, el joc crea un objecte anomenat CellVoid posicionat sobre cada cel·la per a poder gestionar millor l'acció de posicionar una peça en ella.

ornar al Menú Jugar Partida Zona del jugador negre

Tria Peça:

- Peó
- Torre
- Knight
- Alfil
- Reina
- Rei
- Makruk Khon
- Kiangqi Xiang
- Dabbāba

Tria Color:

- Blanc
- Negre

Zona del jugador blanc

Tria les dimensions del tauler

Amplada (Introduïex un valor entre 5 i 14)

Alçada (Introduïex un valor entre 5 i 8)

Editor de taulers

A partir d'aquest moment, el jugador pot afegir peces triant-ne el color i el tipus de peça simplement fet clic en la cel·la on vol posar-la. En cas d'error, tornant a posar la mateixa peça que ja hi ha en una cel·la, aquesta es treu deixant la casella buida. Si es disposa una peça en una casella ocupada per una altra

diferent, aquesta queda substituïda. Entre les peces que hi ha per triar, s'han afegit tres que no formen part dels escacs tradicionals: el Paov dels escacs xinesos [8], el Khon del Makruk [9] i el Gaja del xaturanga [7]. Cadascun d'ell amb els seus moviments diferents. Part de la gràcia del joc és descobrir per un mateix els seus moviments.



Tauler personalitzat

Cal tenir en compte que hi ha peces direccionals com els peons, que no poden retrocedir i, per tant, és important saber cap on avançaran a l'hora de posar-les al tauler. Per aquest motiu s'especifica la zona inicial del jugador negre i blanc.

El procés és simple i sense masses opcions complexes que dificultin a un jugador inexpert crear-ne un com ell desitja.

Un cop acabada la edició, el jugador pot jugar la partida en el mode que ha seleccionat prèviament. En el moment de passar al joc, es comprova que tots dos jugadors tenen mínim un rei en el tauler doncs es tracta d'una peça obligatòria.

5.2 Partida off-line (local)

Aquestes partides estan pensades per a jugar sense internet. El jugador pot seleccionar jugar contra la IA o contra un altre jugador que se suposa que té

accés al seu ordinador, ja sigui física o remotament, per a poder realitzar el seus moviments.

Després d'aquesta tria, el jugador pot escollir entre una partida clàssica, una random (a l'atzar) i entrar en l'editor de taulers descrit anteriorment.

Les partides clàssiques es juguen en un tauler 8x8 amb la distribució de peces tradicional. Les partides a l'atzar, o random, es generen com diu el seu nom: per atzar. En el moment d'entrar en la partida, el joc genera uns càlculs i modifica l'amplada i alçada del tauler i distribueix les peces de forma aleatòria entre les dues files superiors i inferiors. Les peces usades són les del joc clàssic. Aquestes partides són una modificació de la variant de joc Fisher [14].



Tauler Random

El mode en que es desenvolupen les partides és el descrit en els diagrames de flux anteriors.

5.2 Partida on-line

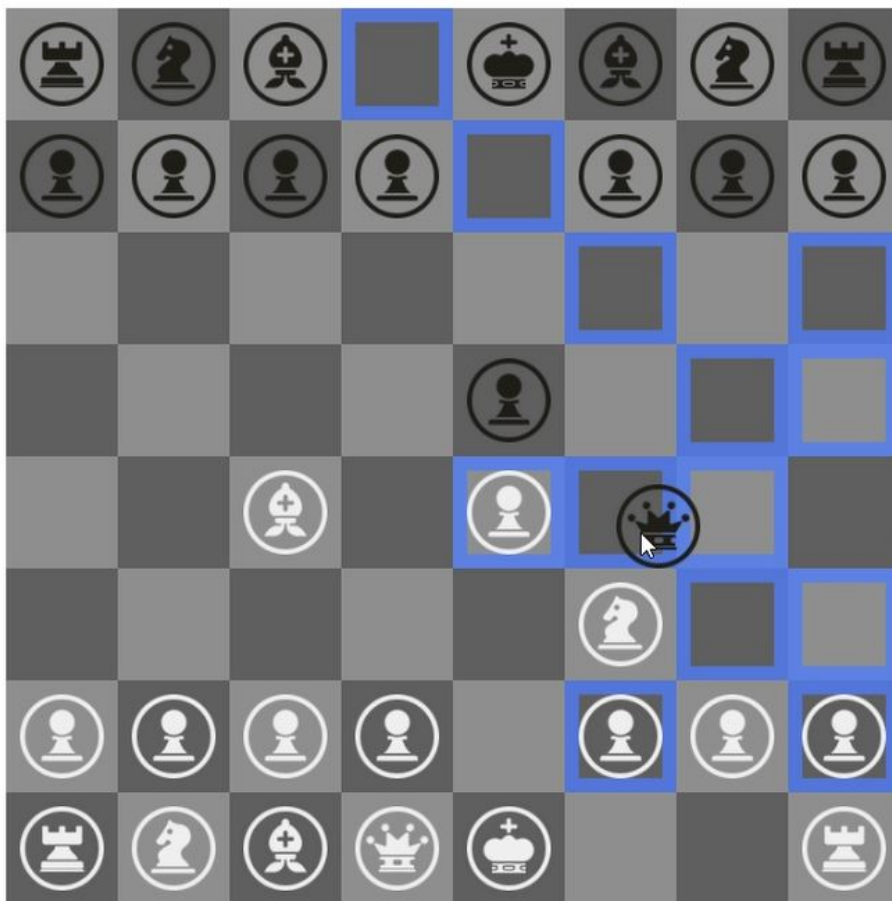
En el cas de les partides on-line, el jugador pot seleccionar, després de iniciar sessió o registrar-se, entre entrar en una partida oberta o crear una ell mateix. En qualsevol dels dos casos, el jugador pot triar entre partida clàssica o personalitzada. En el cas de crear una d'aquest últim tipus, s'obrirà l'editor descrit anteriorment per després, al acabar la personalització, iniciar la partida.

En començar un joc per internet, el primer pas que s'executa és la descarrega de les dades de la partida del servidor. Aquestes dades contenen un string en base 64 amb la informació de cada peça: la seva localització i el tipus i color que té. També en rep les dimensions del tauler. El joc amb aquesta informació presenta la partida al usuari.

El següent pas és la espera d'un contrincant. Cada 5 segons es realitza una consulta al servidor sobre l'estat de la partida. En detectar-ne un segon jugador és passa al torn del jugador blanc. Els torns consisteixen en el flux representat en diagrama del punt anterior. Un jugador realitza el seu moviment mentre l'altre realitza consultes a l'API del servidor preguntant per l'últim moviment

enregistrat de la partida. Si el moviment és d'ell mateix, es que encara no s'ha guardat un moviment del seu oponent i per tant ha de seguir esperant. El temps màxim d'espera és de 300 segons i es pot visualitzar en un *timer* per pantalla. Si passat aquest temps no s'ha realitzat moviment, el jugador de qui és e torn perd el joc.

Si algun moviment mata una peça de rei, el jugador que l'ha realitzat envia al servidor una petició avisant que ha guanyat la partida. En el moment en que es realitza el moviment en el joc de l'altre jugador, el joc consulta al servidor qui ha guanyat per assegurar la informació. En acabar la partida els jugadors tornen al menú de selecció anterior.



Partida on-line entre els jugadors test i test2

6. Definició de la IA (dummy)

Abans d'entrar en detall, cal deixar clar que la IA del joc no està pensada per a ser un repte per a cap jugador si no més aviat al contrari, una distracció lleugera. El moviment són bastant previsible i és senzill guanyar-la. Per la descripció més detallada del procés de creació que hi ha a continuació, es va decidir no invertir més temps en acabar generant una IA no massa difícil de guanyar i convertir la seva senzillesa en una *feature* del joc.

Per a poder jugar de forma individual he introduït una IA en el joc per moure les peces del color contrari al del jugador. En una primera instància aquesta funcionalitat l'havia descartat degut a la complexitat de fer una Intel·ligència Artificial prou potent per a realment ser un repte a superar. Finalment es va decidir buscar una IA ja existent i adaptar-la al codi del videojoc.

Primer es va consultar una de les més conegudes IA que hi ha [15]: Stockfish [16]. Es tracta d'un software amb llicència GPL v3 i, per tant, el vaig poder descarregar gratuïtament per mirar d'aplicar-lo al projecte. Després de revisar el codi al Github, no em vaig veure capaç d'adequar-lo a les necessitats del codi i he buscat altres opcions. Finalment, vaig trobar dues IA [17][18] en Github amb llicència *opensource* que podia provar de modificar per poder introduir-ne una en el joc.

Després d'avaluar-les, vaig veure decidir utilitzar de base el codi de Chess_IA_Unity [17] per fer una IA que estigués adaptada al meu videojoc. Havia de modificar el codi de tal manera que la IA fos capaç d'adaptar-se a taulers de diverses dimensions i a peces noves. Estudiant el codi de la IA, vaig poder veure que el càlcul del moviment de cada torn el dividia en tres parts:

- Control sobre l'estat del rei
- Avaluació del millor moviment de peça.
- Estudi del següent moviment del jugador.

Els dos últims punts els repetia dues vegades cadascun.

Per tant, per adaptar el codi, primer havia de revisar si el rei estava en perill, és a dir, en escac. En cas de trobar que el rei està en perill, ha de moure'l o tancar l'accés a ell a la peça enemiga.

El següent pas era saber quin moviment era millor executar i en aquest cas el codi font del programa que feia servir de base generava un seguit de valors a cada casella on podia moure una peça. Si al moure-la matava una peça enemiga, la casella tenia més valor. També dotava a cada casella d'un valor base segons la peça que es pogués moure a ell. Per exemple un peó proper a arribar al final de tauler per a promocionar tenia preferència a un que encara no havia mogut.

Per últim, es realitzava el mateix càlcul per a les peces del jugador considerant que s'havia mogut la peça a aquella casella. Si això feia que la matessin, la jugada perdia valor respecte les altres.

Degut a la naturalesa dinàmica en les mesures del tauler i el posicionament de les peces que representa el punt principal del joc, el càlcul per a decidir el millor moviment de cada peça no podria replicar-lo de la mateixa manera així que només tenia en compte el fet d'eliminar una peça del jugador i no el posicionament dins del tauler.

Aquest canvi de càlcul feia perdre molta fiabilitat a la tria de moviments de les peces i per tant tornava els punts 2 i 3 en un estudi molt superficial de la partida per part de la IA. Veient que la inversió en temps per a desenvolupar un estudi millor del tauler tenint en compte la variabilitat del mateix no garantia que pogués acabar a temps el projecte, es va decidir deixar la IA com a *dummy* deixant clar al jugador que no trobaria un repte en guanyar-la.

7. API Rest

Per al joc on-line es van analitzar les diverses opcions que hi ha per a connectar dos jugadors a una mateixa partida. La primera opció tracta de que un dels dos jugadors actua com a servidor on genera la partida i l'altre jugador s'hi connecta. La segona opció, és crear un servidor públic que gestiona les partides a on els jugador es connecten per crear-les o per jugar. Finalment es va optar per la segona opció.

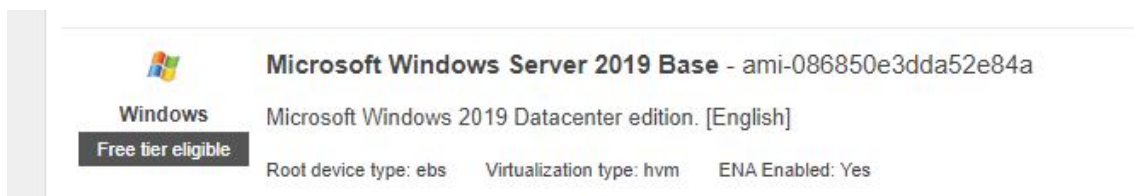
S'ha triat aquest sistema per evitar els problemes generats per IP's dinàmiques, problemes de connexió deguts a la xarxa d'un jugador en concret i també per tenir un millor control sobre les partides creades i els jugadors.

Per a gestionar les partides i usuaris jugadors es va decidir crear una API Rest a la que els videojocs enviessin les seves peticions GET o POST. Amb l'ús d'aquesta API un jugador pot crear un usuari per a les partides on-line, es podrà connectar amb l'ús d'un nom i una contrasenya, crear partides dels tipus descrits en el Capítol 5 o unir-se a alguna partida ja creada. A més a més tindrà accés a unes estadístiques bàsiques sobre ell.

Aquest servidor públic es va adquirir creant un compte gratuït en AWS d'Amazon [19]. Amb la creació d'un usuari en aquest sistema i vinculant una targeta de crèdit, Amazon et proporciona un servidor de forma gratuïta durant un any [20]. Aquest servidor s'escull d'un llistat de servidors i a partir de certes característiques s'aplica un preu. Inicialment, la falta d'una targeta bancària va produir un endarreriment en el desenvolupament d'aquesta part de gairebé una setmana però finalment s'ha pogut acabar fent viable aquest mode de joc. Aquests servidors són anomenats instàncies.



Font: Pàgina d'inici de AWS



Font: Selecció del servidor

<input type="checkbox"/>	t2	t2.nano	1	0.5
<input checked="" type="checkbox"/>	t2	t2.micro Free tier eligible	1	1
<input type="checkbox"/>	t2	t2.small	1	2
<input type="checkbox"/>	t2	t2.medium	2	4
<input type="checkbox"/>	t2	t2.large	2	8
<input type="checkbox"/>	t2	t2.xlarge	4	16

Font: Selecció del tipus de servidor.

En aquestes imatges es pot veure com un màquina Microsoft Windows Server 2019 i t2.micro es totalment gratuïta el primer any. Per iniciar el desenvolupament aquesta va ser l'elecció. Com aquesta màquina només disposava d'una CPU i 1 Gb de RAM, al inici de les proves del joc on-line, amb dos videojocs fent enviaments continus al servidor, aquest no suportava les tasques i la CPU pujava al 100% fent que no es pogués jugar.

Per solucionar aquest problema es va ampliar la màquina a una t2.medium amb 2 CPUs i 4 Gb de RAM. Amb aquest augment de les característiques es va solucionar el problema de rendiment del joc on-line.

Per últim, per poder connectar el joc al servidor, era necessari que aquest disposés d'una IP fixa que no variés en el temps. D'aquesta manera, mentre el servidor estigués encès, el joc disposaria de connexió a ell. Per fer això, Amazon AWS disposa d'una eina anomenada *Elastic IP* que assigna una IP fixa a una instància amb independència de la seva configuració o els canvis que es realitzin a ella.

8. Requisites

Chessly és un videojoc que no té masses requisits però és important tenir en compte certs factors abans de jugar. Degut al seu desenvolupament, les següents condicions són necessàries per a poder jugar correctament:

- Chessly està programat per a funcionar en un Windows i només ha sigut testejat en Windows 10.
- La resolució mínima per a poder jugar és de 1600x900 però la recomanada és 1920x1080. Amb resolucions més baixes de la mínima, el joc és jugable però els botons i el tauler queden totalment descol·locats i no fa agradable la experiència de joc
- Com és de suposar, per poder jugar on-line és necessari tenir internet.

Tenint clars aquests punts, Chessly ofereix l'experiència de joc esperada.

9. Conclusions

9.1 Coneixements adquirits

Durant la realització d'aquest treball hi ha hagut un conjunt de coneixements nous que he anat adquirint. Sense ells no hagués estat capaç de realitzar el producte final. També he adquirit noves experiències amb certes parts de la gestió de projectes i de càrregues de treball. En resum podria dir que duu a terme aquest projecte no només m'ha donat com a resultat el producte final del mateix sinó també un conjunt de coneixements nous que em podran ser útils en un futur.

L'ús de l'eina Unity, més en concret el 2D, així com la seva vinculació amb el llenguatge de programació C# és un exemple d'aquests coneixements adquirits. Prèviament al projecte mai havia treballat amb aquesta eina i desconeixia com s'executaven els scripts que configuren el joc.

L'adquisició d'un servidor AWS també és part d'aquestes coses que he après durant el desenvolupament del projecte. Com crear una compte en la seva web, contractar el servidor i configurar-lo. La gestió de la IP fixa també és un procés nou que no coneixia prèviament.

Per últim també crec que en la gestió tasques i el càlcul de temps d'aquestes he guanyat experiència. En la primera llista de tasques es pot veure que no estan ben definides i que els temps no són massa realistes o estan mal ajustats. En canvi en els revisions posteriors de la planificació es comprova que coneixia millor les tasques a fer i podia aproximar de forma més correcte els temps de desenvolupament d'elles.

9.2 Assoliment d'objectius i seguiment de planificació

Pel que respecte als objectius estic bastant satisfet amb el resultat obtingut. El joc permet partides a dos jugadors tan en local com per internet i l'edició personalitzada del tauler i la col·locació de les peces. Per altra banda és cert que pel que respecta a la IA no estic dels tot content amb el resultat. No esperava poder tenir una gran intel·ligència artificial que suposés un repte per a molt jugadors ni habilitat un sistema de dificultats en funció de les eleccions del jugador però crec que li falta profunditat. Tot i això, crec que és una funcionalitat addicional que proporciona un valor afegit al producte respecte a una versió on no hi fos.

Per altra banda, s'ha introduït un multidioma al joc que inicialment no estava contemplat així com el mode de joc Random que tampoc constava en els objectius d'inici. Crec que aquests dos afegits tenen més valor que una IA més ben programa. En concret, jocs d'escacs amb una bona IA que puguin suposar un repte ja hi ha en el mercat mentre que joc amb la opció de 'joc a l'atzar' no són habituals.

Pel que respecte a la planificació, crec que tot i no respectar estrictament la plantejada al inici, he desenvolupat el projecte dins els terminis establerts en la mateixa. No només les entregues de cada PAC tenien el contingut que esperava que tinguessin sinó que, com en el cas del 'joc a l'atzar', algunes entregues tenien més contingut. Per altra banda, és cert que pel que respecte al desenvolupament del joc on-line ha tingut un endarreriment però no ha suposat la pèrdua de funcionalitats en el joc.

Considero doncs que tan els objectius com la planificació s'han aconseguit i seguit de forma positiva i, tot i no se un producte perfecte i haver tingut endarreriments en algun desenvolupament, el resultat final compleix correctament el producte desitjat inicialment.

9.3 Futur del projecte

Mirant a un futur, aquest projecte encara té moltes possibilitats i funcionalitats que es podrien afegir. La IA podria millorar-se per oferir el mateix nivell de jugabilitat que gran part del jocs d'escacs del mercat ofereixen. Una altra millora podria estar en l'editor i les seves opcions de personalització. Es podria afegir l'opció de canviar regles de joc, inventar peces noves des de zero o definir caselles amb condicions personalitzades. I pel que respecte al online també hi ha possibles millores: un buscador de partides que permetés filtrar la cerca, un funció d'afegir amics o altres funcions socials com un xat dins la partida o avatars per a jugadors.

En definitiva, les opcions que hi ha per a seguir millorant el projecte són diverses. Gairebé a cada component que forma el joc se li poden afegir funcionalitats noves sense perdre la idea base que el defineix. En certa manera, això mostra que el producte Chessly és una bona base que pot créixer fins ser un videojoc molt versàtil sense canviar la idea original que el defineix: poder actualitzar els escacs tradicionals.

10. Glossari

- API Rest: Sistema de transmissió d'informació entre servidor i client. Consisteix en crides GET i POST, entre altres, a direccions web configurades en el servidor per donar o emmagatzemar informació.
- Background: fons d'una pantalla o menú. Usualment una imatge o color pla.
- Dummy: Terme anglès usat per definir la falta de intel·ligència o perícia d'alguna persona o programa.
- Feature: Característica o funcionalitat d'un programa o producte.
- Game Engine: Programa o tecnologia usada per a fer el joc.
- Game feel: Sensació de joc percebuda pel jugador.
- IA (intel·ligència artificial): Programació que permet a l'aplicació poder donar resposta a diferents inputs d'entrada. En aquest cas, realitzar un moviment d'una peça en funció a les que hi ha en el tauler i la seva col·locació.
- Off-line: Terminologia usada per definir la capacitat de prescindir de connexió a internet per funcionar.
- On-line: Terminologia usada per definir la necessitat d'una connexió via Internet.
- Scene (Escena): Seqüència de joc o pantalla. El joc consta d'un seguit d'elles. Cadascuna d'elles estan formades per imatges (sprites), textos i scripts, entre altres.
- Scripts: Codis de programació que s'executen en les Scenes del joc. Amb ells es controlen les accions del jugador i dels elements del videojocs.
- Sprites: Imatges usades dins del joc.
- Streamers: Creadors de contingut audiovisual que emeten en directe partides de videojocs entre altres continguts.
- Soft-delete: Mètode per marcar com a esborrat contingut de la base de dades sense borrar realment la informació.
- Timestamps: Marca de temps emmagatzemada en la base de dades.

11. Bibliografia

- 1- Streamers de Twitch que es dediquen als escacs:
<https://www.twitch.tv/directory/game/Chess> (01/06/2021)
- 2- Augment en la venda d'escacs produït per la sèrie "The Queen's Gambit":
<https://seriepolis.com/2020/12/the-queens-gambit-ajedrez/> (01/06/2021)
- 3- Article de PogChamps3 en Chess.com:
<https://www.chess.com/article/view/pogchamps-3-all-the-information>
(01/06/2021)
- 4- Pàgina de la Wikipedia dels escacs:
<https://en.wikipedia.org/wiki/Chess> (01/06/2021)
- 5- Joc d'escacs Simply Chess:
https://store.steampowered.com/app/312280/Simply_Chess/?l=spanish
(01/06/2021)
- 6- Editor de taulers Ajedrez Eureka:
<https://www.ajedrezeureka.com/editor-fen-diagrama-de-ajedrez-estatico-que-crea-imagen-grafica/> (01/06/2021)
- 7- Pàgina de la Wikipedia del xaturanga:
<https://en.wikipedia.org/wiki/Chaturanga> (01/06/2021)
- 8- Pàgina de la Wikipedia del Xiangqi:
<https://en.wikipedia.org/wiki/Xiangqi> (01/06/2021)
- 9- Pàgina de la Wikipedia del Makruk:
<https://en.wikipedia.org/wiki/Makruk> (01/06/2021)
- 10- Webs consultades sobre els *engines* de programació de jocs:
<https://revistadigital.inesem.es/informatica-y-tics/herramientas-para-desarrollo-de-videojuegos/> (03/04/2021)
<https://www.aprendegamemaker.com/programas-para-crear-videojuegos-gratis/> (03/04/2021)
<https://www.hobbyconsolas.com/reportajes/mejores-programas-gratis-crear-videojuegos-94320> (03/04/2021)
<https://www.softzone.es/2018/10/29/motores-aplicaciones-crear-juegos-windows/> (03/04/2021)
https://www.youtube.com/watch?v=YPOPp_BWgJM&ab_channel=ATMAN (03/04/2021)
<https://www.industriaanimacion.com/2020/03/programas-para-crear-videojuegos/> (03/04/2021)
- 11- Pàgina web de descarrega de GameMaker:
<https://www.yoyogames.com/en/get> (03/04/2021)

- 12-Pàgina web de Game Salad:
<https://gamesalad.com/> (03/04/2021)
- 13 - Pàgina web de Stencyl:
<http://www.stencyl.com/> (03/04/2021)
- 14 – Wikipedia sobre la variant de Fischer del escacs:
https://ca.wikipedia.org/wiki/Escacs_aleatoris_de_Fischer (01/06/2021)
- 15 – Article de chess.com sobre IA en els escacs:
<https://www.chess.com/es/article/view/inteligencia-artificial-ajedrez>
(03/04/2021)
- 16-Pàgina oficial de Stockfish, una IA d'escacs:
<https://stockfishchess.org/> (03/04/2021)
- 17- Github de la IA d'escacs Chess_IA_Unity:
<https://github.com/SkylerAlvarez/Chess-AI-Unity> (03/04/2021)
- 18- Github de la IA d'escacs Chess_IA:
<https://github.com/SebLaque/Chess-AI> (03/04/2021)
- 19- Pàgina de AWS d'Amazon
<https://aws.amazon.com/es/> (03/06/2021)
- 20- Anunci d'Amazon AWS sobre la capa gratuïta de servidors
https://aws.amazon.com/es/free/?all-free-tier.sort-by=item.additionalFields.SortRank&all-free-tier.sort-order=asc&awsf.Free%20Tier%20Types=*all&awsf.Free%20Tier%20Categories=*all (03/04/2021)
- 21 – Pàgina web de FreeSound
<https://freesound.org/> (01/06/2021)
- 22 – Pàgina web de Epidemic Sound
<https://www.epidemicsound.com/music/featured/> (01/06/2021)