



Treball de Fi de Grau

GRAU D'ENGINYERIA INFORMÀTICA

**Facultat de Matemàtiques
Universitat de Barcelona**

**Fracsland: joc seriós per aprendre
fraccions**

Cristian Muriel Ordoñez

Director: Anna Puig Puig
Realitzat a: Departament de
Matemàtica Aplicada i
Anàlisi. UB

“Digues i ho oblido, ensenya’m i ho recordo, involucra’m i ho aprenc.”

Benjamin Franklin

Abstract

Fracsland is an educative project oriented to reinforce the concept of fraction to the students of the “Cicle Superior de Primària” and how this knowledge can be related with real world concepts.

In order to achieve this goal a platform which consists in two parts is presented in this project.

On the one hand, there is a part designed especially for the students which consists in a serious game of fractions with beautiful 3D graphics powered by Unity3D which will allow them to have fun on an island based environment while learning about concepts related with fractions.

On the other hand, there is a web platform designed for teachers, powered by Django which will allow them to adapt the game to each student according to their skills. This platform also provides teachers with the ability of monitor the progress their students, useful to know which ones require more help.

Resum

Fracsland és un projecte de caràcter educatiu que té com a objectiu ajudar als alumnes del Cicle Superior de Primària a consolidar el concepte de Fracció i com aquest es pot relacionar amb elements del món real mitjançant “Serious Games”.

Per portar a terme aquest objectiu en aquest projecte es presenta una nova plataforma anomenada Fracsland que disposa de dues parts.

Per una banda els alumnes disposen d'un joc seriós de fraccions amb bonics gràfics 3D desenvolupat en Unity3D ambientat en una illa on es divertiran vivint aventures mentre aprenen sobre conceptes relacionats amb fraccions.

Per altra banda el personal docent disposa d'una plataforma web desenvolupada amb Django des de la es pot adaptar el joc individualment per cada estudiant ajustant-lo a les necessitats d'aprenentatge de cada un d'un mode senzill. També permet monitoritzar quin és el progrés dels estudiants, ajudant així a mostrar quins són els estudiants que precisen de més ajuda.

Agraïments

La realització d'aquest Treball de Final de Grau no hagués estat possible sense les persones que han estat allà dia a dia, motivant-me quan em sentia desmotivats, animant-me quan em sentia trist i sobretot fent-me tocar de peus a terra quan m'exigia massa.

Per aquest motiu m'agradaria agrair a la meua tutora que sempre hagi estat allà quan l'he necessitada disposada a aportar tots els coneixements i recursos que he necessitat.

També m'agradaria agrair als meus companys de la facultat la seva presència, ja que la seva actitud crítica típica de científics i enginyers i les llargues converses que hem tingut en els bonics jardins de la UB m'han ajudat a millorar com a enginyer informàtic, cosa que m'ha resultat molt útil per la realització d'aquest projecte.

Per últim m'agradaria agrair a les persones més properes a mi que m'hagin donat suport durant aquest camí d'alts i baixos emocionals que he viscut durant el temps que he estat treballant en Fracslan.

A tots, gràcies de tot cor.

Taula de continguts

Capítol 1. Introducció	7
1.1. Àmbit del projecte.....	10
1.2. Motivació.....	11
1.3. Objectiu General	11
1.4. Objectius Específics.....	12
1.5. Tasques.....	12
1.6. Planificació temporal	13
1.7. Organització de la memòria.....	13
Capítol 2. Anàlisi del problema.....	15
2.1. Introducció.....	15
2.2. Tipus d'usuari i principals casos d'ús (administrador, professor, jugador)	15
2.3. Arquitectura	20
2.3.1. Servidor Web.....	20
2.3.2. Joc patró	21
2.3.3. Disseny del joc de Fracland.....	22
2.4. Conclusions.....	23
Capítol 3. Disseny del joc Fracland	24
3.1. Disseny del joc seriós.....	24
3.1.1. Experiència.....	24
3.1.2. Mecàniques	28
3.1.3. Choice	33
3.1.4. Interacció.....	33
3.1.5. Feedback	34
3.1.6. Impact	35
3.2. Desenvolupament tecnològic del joc	35
3.2.1. Tecnologia utilitzada	35
3.2.2. Desenvolupament.....	37
Capítol 4. Disseny del servidor.....	61
4.1. Disseny del panell d'Administració Fracland.....	61
4.2. Disseny del panell d'Administració Fracland.....	62
4.2.1. Disseny del configurador de Quests (rol administrador).....	62
4.3. Disseny del panell d'Administració del professor.....	64
4.3.2. Tutorial pel professor	75
4.4. Disseny de la Base de Dades.....	76
4.4.1. Components del joc.....	76
4.4.2. Joc configurable	77
4.4.3. Organització de classes i sessions de joc.....	78

4.4.4. Comunicació joc Servidor	79
4.4.5. Utilitats	79
4.5. Interfície de Comunicació amb el Joc	80
4.6. Justificació de les tecnologies utilitzades	84
Capítol 5. Resultats i Simulacions	87
5.1. Plataforma Resultant	87
5.2. Tests amb usuaris:	92
5.3. Temps de resposta al servir una pàgina utilitzant PingDom	93
5.4. Proves d'estrès servidor	94
5.5. Anàlisi del temps de realització del projecte	96
Capítol 6. Conclusions (i feines futures)	98
Capítol 7. Referències bibliogràfiques	100
Apèndix A: Manual tècnic	102
A.1. Instal·lació entorn de desenvolupament: Requeriments mínims i passos a seguir .	102
A.1.1. Requeriments mínims	102
A.1.2. Instal·lació entorn de desenvolupament Joc	102
A.1.3. Instal·lació entorn de desenvolupament Servidor	103
A.2. Manual del desenvolupador	105
A.2.1. Modificar les configuracions del joc	105
A.2.2. Crear una nova Quest	105
A.2.3. Creació d'una nova escena	106
A.2.4. Modificació IP on el joc realitza les peticions:	106
Apèndix B: Resultats tests d'usuari	107
B.1. Resultats tests plataforma professor	107
B.2. Resultats tests joc	109

Capítol 1. Introducció

Fracsland és un projecte de caràcter educatiu que té com a objectiu ajudar als alumnes del Cicle Superior de Primària consolidar el concepte de Fracció i com aquest concepte es pot relacionar amb coses del món i de la vida quotidiana, ja que a priori sembla que és un concepte que en general costa d'assumir.

En aquests últims anys, els nens i nenes s'han acostumat a la tecnologia, amb l'aparició dels videojocs per tabletas, mòbils i videoconsoles aquests s'han convertit en quelcom quotidià per ells.

Els videojocs semblen tenir un poder d'atracció molt potent tant per els joves com per els adults, ja sigui per els atractius gràfics i colors dels que molts d'ells presenten o per el fet que permeten que l'usuari sigui el protagonista d'una infinitat d'històries diferents de forma interactiva.

Fracsland, el joc que es proposa en aquest projecte, igual que molts altres jocs que fa anys que estan apareixent, aprofitant el potencial d'atracció que aquestes tecnologies presenten, pretén divertir i entretenir als alumnes i al mateix temps ajudar a assumir conceptes acadèmics d'un mode suau i desenfadat, servint com a complement a les seves classes de matemàtiques.

Convertir l'aprenentatge en un joc implica que aprendre i divertir-se no tenen per què ser conceptes oposats, fet que desgraciadament en general molts alumnes assumeixen.

Fracsland intenta motivar i recompensar l'esforç del estudiant proporcionant-li algun tipus de benefici immediat dins del joc on ell es troba immers, més enllà de la recompensa acadèmica que representa treure una bona nota per haver fet un exercici correctament. A part, també estalvia al professor el procés de correcció, ja que quan un alumne resol un repte, el sistema és capaç de determinar si el repte ha estat superat correctament o no permetent saber al professor quants errors ha comés l'estudiant i si ha pogut o no resoldre l'exercici satisfactòriament.

Fracsland també té com a avantatge el fet que es pot fer servir per crear experiències personalitzades d'un mode relativament més confortable que els exercicis tradicionals, ja que només canviant alguns paràmetres en una pàgina web per part del professorat, un alumne pot accedir a una experiència única i adaptada al seu nivell acadèmic actual.

El sistema de personalització de nivell de Fracsland, a part de fer servir les configuracions que crea el professor mitjançant la web, també té en compte l'experiència del jugador dins del joc, assignant un nivell superior dins del joc als alumnes que hagin jugat més i això farà que els jugadors més experimentats comencin a treballar conceptes més avançats en el tema com el concepte d'equivalència de fracció i pel que fa el la part de joc cada cop serà una mica més difícil per tal d'intentar estimular a l'alumne a jugar, ja que sinó podria caure en la monotonia i l'avorriment.

Cal esmentar que el joc pot ser jugat tant en el centre docent com a casa, ja que per tal de seguir el joc del nen/a es fa ús d'internet per rebre i enviar dades, tant acadèmiques com del propi joc. L'alumne no té perquè estar en el centre durant la realització de l'activitat, sinó que pot jugar a casa i automàticament el professor quan es connecti a la plataforma podrà veure els progressos dels estudiants.

Per realitzar aquest propòsit Fracsland disposa de dues parts diferenciades, una destinada al professor i l'altra al alumnat.

D'una banda, el personal docent disposa d'una pàgina web, on es poden registrar, crear classes i registrar als seus estudiants dins d'elles i llavors decidir quines fraccions treballaran els seus alumnes en una sessió de joc determinada (podríem dir que cada sessió es correspon a una sessió de classe).

A mesura que els alumnes juguin i vagin progressant en el joc, els professors podran consultar quin és el progrés dels alumnes, des del mateix panell web on han configurat el joc d'un mode transparent a l'estudiant.

L'alumne, per altra banda, disposa d'un videojoc en tres dimensions on juga el paper d'un nàufrag al que se li ha espatllat el vaixell i ha anat a parar a una illa anomenada Fracsland.

En aquesta illa, resulta que les fraccions no son només una branca de les matemàtiques, sinó que son màgiques i permeten reconstruir ponts, derrotar enemics, fer que les plantes creixin més ràpid ...

Quan un estudiant entra en el joc, no té la sensació d'estar realitzant una tasca acadèmica, ja que el joc presenta l'estètica d'un joc d'aventures i de fet ho és, Fracsland, a part de ser un joc destinat a l'aprenentatge de fraccions, és un joc de rol on el jugador és lliure per recórrer el món, lluitar contra enemics, plantar plantes i comprar-se objectes per ser més fort o bé roba per tenir un aspecte diferent.

Quan l'alumne entra en el joc apareix en un poble anomenat Fracstown on ha de parlar amb el governador d'aquest poble perquè li proporcioni algunes peces per reparar el vaixell que té varat a la platja.

La Figura 1 és la imatge inicial del que veu un alumne al entrar al joc. La pantalla té varies parts diferenciades: l'estat del jugador, la zona de missions (Goals Zone) i la part central, on una fletxa indica la direcció cap a on pot anar el jugador per a obtenir l'objectiu actual del joc.



Figura 1: Situació inicial del jugador a la illa de Fracstland

El governador, accepta donar-li les peces sempre i quan l'ajudi amb alguns problemes que hi ha en l'illa. Així, el jugador ha d'anar a buscar el governador, Lord Barus, tal i com indica la interfície "Zone Goals" que surt a la figura. La fletxa verda del centre, indica cap a on es troba en Lord Barus.

Durant el desenvolupament de l'activitat, a part de resoldre els reptes acadèmics, el jugador haurà d'explorar Fracstland per tal d'obtenir fraccions i diners per poder millorar el seu personatge, en la Figura 2 es poden veure algunes de les zones que pot que recórrer el jugador.

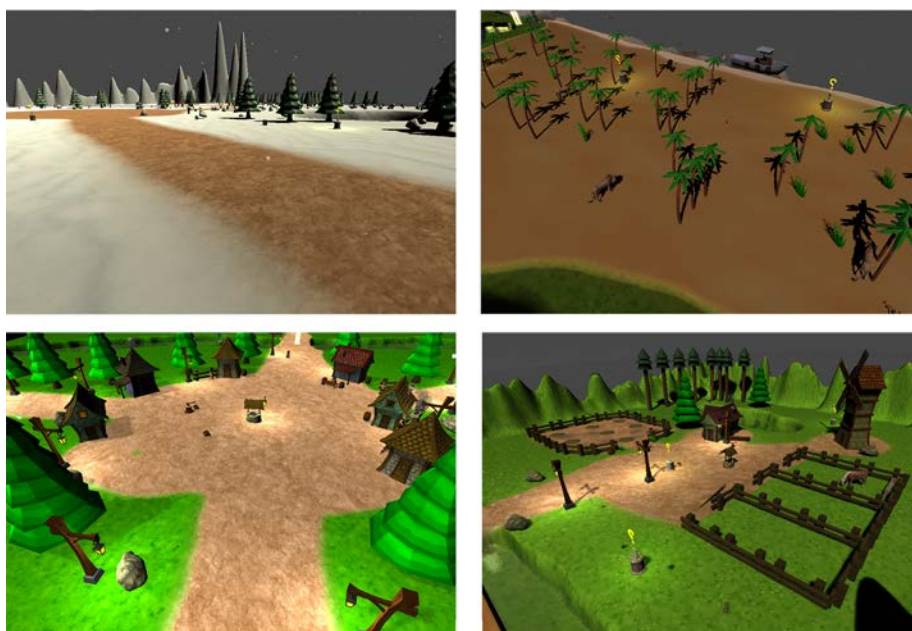


Figura 2: D'esquerra de dreta: Frozen Woods, Small Beach, Fracstown i Fracfarm.

Fracsland s'ha creat amb la filosofia de que l'estudiant és lliure de decidir com vol jugar, és a dir, un estudiant que tingui un caràcter més bèl·lic, podrà decidir realitzar les tasques proposades per el professor en zones del joc on el combat és lo més important, mentre que un alumne que prefereixi jugar sense gaire violència pot decidir anar a zones més tranquil·les i realitzar les seves activitats allà. D'aquesta manera dos alumnes podrien tenir exactament les mateixes fraccions per treballar i viure dues experiències de joc totalment diferents.

1.1. Àmbit del projecte

Fracsland és un projecte amb finalitats educatives que sorgeix de la necessitat en les aules on activitats matemàtiques s'han de treballar més, per tal de consolidar els conceptes introduïts prèviament en classe.

També ha sorgit de la necessitat dels professors de poder configurar activitats adequades al nivell de cada estudiant.

Fracsland és un videojoc seriós, del anglès “serious games”. Els jocs seriosos són jocs que tenen com a objectiu formar al jugador en algun camp, ja sigui en l'àmbit acadèmic o el professional (Abt, 1970).

Fracsland, tal i com s'ha explicat prèviament en la introducció, intenta servir com a complement a l'educació en fraccions, ajudant als estudiants a consolidar el concepte de fraccions mentre es troben immersos en una aventura digital.

Dins del pla d'estudis d'Enginyeria Informàtica de la Universitat de Barcelona, Fracsland es pot relacionar amb una gran quantitat d'assignatures.

Es pot veure Fracsland com un servidor web amb una base de dades, una interfície de comunicacions i un joc, per tant el nombre de matèries que avarca dins d'aquest pla d'estudis és molt gran.

Per desenvolupar tot software, es precisa de coneixements de programació, algorísmica i estructures de dades per tal de poder portar a terme el projecte, ja que els coneixements que aporten aquests camps permeten convertir un disseny en quelcom que un sistema informàtic és capaç de representar, convertint així el que és un disseny en una realitat.

També, tenint en compte que ens trobem davant d'un videojoc que es porta a terme en un món virtual en tres dimensions es precisen coneixements de física i matemàtiques, ja que aquests jocs intenten emular d'alguna manera la realitat. . Dins d'aquests camps anteriorment esmentats, les matemàtiques, a part del càlcul i l'estadística dos components necessaris per realitzar un joc, també ens aporten l'estudi de les matrius i els vectors que resulten ser fonamentals ja que barrejats amb coneixements físics permeten simular com es mouen els cossos en aquest món virtual, gestionar col·lisions entre cossos, simular la gravetat, etc.

També cal tenir en compte que tot i saber simular el comportament dels cossos en un món 3D no es podria realitzar un joc sense poder visualitzar el que està passant, aquí es on entren en joc els coneixements de Gràfics per Computador, que a part de donar-nos una

representació visual del món, ens permeten il·luminar-lo, donar-li color i convertir l'experiència en quelcom comprensible per l'ull humà.

Com Fracsland no només és un joc sinó també un editor de jocs que requereix components web i utilitzar la xarxa, per la realització de Fracsland també són molt necessaris els coneixements que aporten camps com les Xarxes i el Software Distribuït, ja que sense ells no es podria portar a terme la comunicació entre el joc que està jugant l'alumne amb el panell d'administració del professor.

La plataforma de Fracsland com tot projecte informàtic, està feta per als usuaris, per això mateix s'ha de dedicar temps i esforç en entendre el que necessiten, ja que una plataforma dissenyada ignorant això pot ser que, tot i que funcioni correctament, mai no arribi a ser útil, degut a que no s'entén com funciona, resulti avorrida, angoixant o senzillament no sigui pràctica. El camp dels Factors Humans ens ajuda a dissenyar interfícies usables i amenes, conscienciant als desenvolupadors en pensar en l'usuari durant el desenvolupament del projecte i aportant regles que serveixen com a guia per dissenyar els mecanismes d'interacció amb l'usuari final de la plataforma.

En aquesta plataforma també es recullen dades, dades que s'han d'emmagatzemar i llavors estar disponibles per a ser consultades. El camp de les Bases de Dades ofereix el que es necessita per tal de poder realitzar això, aportant coneixements de com s'ha de dissenyar una base de dades i com utilitzar-les.

Tampoc cal oblidar que abans de desenvolupar, s'ha de dissenyar el que serà desenvolupat, aquí és on Disseny de Software resulta útil, ja que precisament aporta els coneixements necessaris, per tal de saber captar com la informàtica podrà ajudar a resoldre un problema. Realment serveix com a nexa d'unió entre el problema real i què pot aportar la informàtica per resoldre'l.

Finalment en el dia a dia del projecte els coneixements obtinguts en Enginyeria del Software han servit per tal de mantenir el curs del desenvolupament en bona direcció en tot moment, gràcies als coneixements que aporta sobre com s'ha de gestionar un projecte mantenint-se àgil, ja que una de les coses més importants que s'ensenya en aquesta assignatura és a gestionar el recurs més important del que disposem, el temps.

1.2. Motivació

La motivació principal del projecte és ajudar als estudiants del Cicle Superior de Primària a reforçar les competències en fraccions, ja sigui en l'entorn de l'aula com fora d'ella.

Fracsland ha de ser un videojoc divertit, fàcilment configurable per el professor, tenint en compte les limitacions dels equips informàtics dels centres docents disposen.

1.3. Objectiu General

L'objectiu del projecte Fracsland és crear una plataforma tecnològica que permeti als docents ajudar als alumnes a reforçar les competències en fraccions d'un mode personalitzat al nivell de cada alumne usant videojocs modificables. Així els alumnes reforcen aquestes

competències mentre juguen a un joc entretingut que els manté motivats i els impulsa a millorar.

1.4. Objectius Específics

L'objectiu general es pot refinar en els següents objectius específics:

- **Anàlisi i Disseny d'un patró de un joc flexible i configurable:** S'ha de crear un patró d'un sistema que permeti la creació de jocs que puguin rebre una sèrie de configuracions.
- **Anàlisi, disseny i creació d'un joc de fraccions (creació).**
- Anàlisi disseny i desenvolupament d'una plataforma Web tecnològica de configuració de jocs que segueixen el patró que s'ha definit en el primer punt.
- Validació de les eines proposades.

1.5. Tasques

Donats els objectius anteriors, s'han definit les següents tasques a desenvolupar en el projecte:

- **T1: Organització del projecte:** Planificar com es durà a terme el projecte i quines metodologies es seguiran, així com avaluar a cada iteració del procés les només tasques a desenvolupar i a refinar.
- **T2: Joc patró:** Anàlisi i disseny d'un joc configurable, en aquesta tasca és on es pensa quins paràmetres podrà modificar el professor i quins s'enviaran entre el joc i el webservice i per tal de crear l'experiència personalitzada de l'alumne i la capacitat de monitorització del professor.
 - **T2.1. Anàlisi**
 - **T2.2. Disseny**
- **T3: Joc:** Procés de creació d'un videojoc de fraccions que a partir dels paràmetres definits en la T2 sigui capaç de rebre unes configuracions i crear una experiència personalitzada per el jugador i enviï les dades que es considerin oportunes per tal d'avaluar el progrés d'aprenentatge de l'alumne.
 - **T3.1. Anàlisi**
 - **T3.2. Disseny**
 - **T3.3. Formació**
 - **T3.4. Implementació**
 - **T3.5. Testing**
- **T4: Plataforma Web:** Creació d'una plataforma web on els administradors de Fracslan i els professors es puguin connectar per tal de gestionar les dades definides en la Tasca 2.
 - **T4.1. Anàlisi**
 - **T4.2. Disseny**
 - **T4.3. Formació**
 - **T4.4. Implementació**
 - **T4.5. Testing**
- **T5: Interfície comunicació:** Creació dels sistemes de comunicació entre el client del alumne i el servidor.
 - **T5.1. Anàlisi**
 - **T5.2. Disseny**
 - **T5.3. Formació**
 - **T5.4. Implementació**
 - **T5.5. Testing**

1.6. Planificació temporal

Les primeres tasques que s'han de portar a terme són la organització del projecte, i l'anàlisi i el disseny de joc patró, ja que sense aquestes no és pot començar la resta de tasques per manca de les especificacions concretes que necessiten en el seu anàlisi.

Un cop acabades aquestes tasques és s'analitzen i es dissenyen els components que conformen la plataforma Fracsland.

Un cop realitzat tots els anàlisis i dissenys es pot començar amb el procés de implementació del joc, tenint en compte que en algun moment aquest s'haurà de connectar amb el webservice.

Un cop creat el joc, s'ha de crear les plataformes del professor i de l'administrador de Fracsland, per tal de poder crear classes, registrar alumnes, configurar jocs...

Un cop realitzat això s'han de connectar els dos components a través de el webservice, per tant la planificació temporal es mostra en el Diagrama de Gantt de la Figura 3.

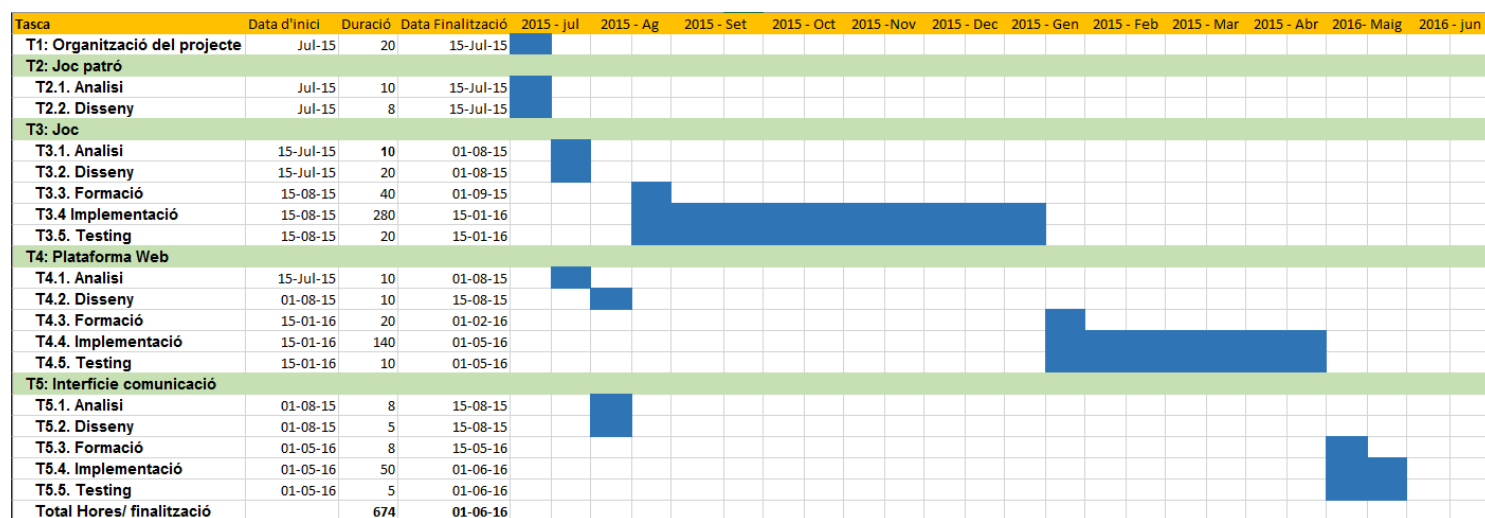


Figura 3: Diagrama de Gantt previst per el projecte

Aquest projecte es va començar l'estiu del 2015, donat el gran abast d'objectius que es pretenia aconseguir. Tal i com es pot veure el temps total previst és d'unes 674 hores i la data de finalització prevista és el dia 1 de juny de 2016.

1.7. Organització de la memòria

- **Capítol 1. Introducció:** En aquest capítol es realitza una breu introducció al projecte presentant que és, quins son els seus objectius i quines son les tasques que es realitzaran per portar a terme aquests objectius.

- **Capítol 2. Anàlisi del problema:** Em aquest capítol s'analitza quin és el problema que vol resoldre aquest projecte i com s'ha plantejat resoldre'l.

- **Capítol 3. Disseny del joc Fracsland:** En aquest capítol s'explica com s'ha dissenyat el joc des del punt de vista d'un dissenyador de videojocs i també com s'ha acabat implementat.

- **Capítol 4. Disseny del servidor:** Al igual que en el capítol anterior primer s'explica com s'ha dissenyat la plataforma web i tot el que aquesta conté i com s'ha acabat implementant tot.
- **Capítol 5. Resultats i Simulacions:** En aquest capítol es presenta la plataforma resultant, els resultats dels testos d'usuari i els resultats de diverses simulacions en el servidor.
- **Capítol 6. Conclusions (i feines futures):** En aquest capítol es recorda quin es l'objectiu principal del projecte i quines aportacions s'han fet per tal de que sigui possible. A part també conté les feines que poden sorgir partint d'aquest projecte com a base.
- **Capítol 7. Referències Bibliogràfiques:** En aquest capítol s'enumeren les referencies que s'han utilitzat per escriure la memòria i desenvolupar la plataforma.

Capítol 2. Anàlisi del problema

2.1. Introducció

Fracsland ha de permetre als educadors poder crear una experiència de joc personalitzada per els seus alumnes sense necessitat de tenir coneixements de programació informàtica ni tan sols uns coneixements avançats a nivell d'usuari. Així doncs s'ha de crear un sistema que els permeti crear aquest joc personalitzat d'un mode senzill.

A part de poder oferir la possibilitat de configurar un joc, la plataforma ha de permetre als professors monitoritzar el progrés dels alumnes.

Per el que fa la part de l'alumne s'ha de dissenyar un joc seriós de fraccions, que sigui atractiu i entretingut que tingui la capacitat de rebre aquestes configuracions remotament i adaptar-se a elles d'un mode transparent als usuaris de la plataforma.

Tal i com s'ha esmentat anteriorment en el capítol 1 aquests sistemes han de poder ser utilitzats mitjançant el material del que es disposa a les aules, que actualment solen ser ordinadors i tablets.

2.2. Tipus d'usuari i principals casos d'ús (administrador, professor, jugador)

La plataforma de Fracsland esta destinada a 3 tipus d'usuaris:

Administrador: Aquest usuari és el que decideix i gestiona quins són els atributs configurables d'un joc, des del seu panell. L'usuari administrador permet obrir la plataforma a nous possibles jocs configurables seguint estratègies similars a Fracsland.

L'administrador té els següents cassos d'ús:

- UC1: Gestionar les configuracions de les quests: Com a administrador ha de poder decidir quins son els paràmetres configurables d'un joc des de la mateixa plataforma web.
- UC2: Afegir i borrar tipus de fraccions: Sempre que es creí una nova representació de fracció a dins del joc l'administrador la pot afegir i d'aquesta manera els professor la tindran disponibles en el configurador de quests.
- UC3: Visió general de la plataforma: L'administrador ha de poder-se fer una idea de quants usuaris hi ha registrats a la plataforma, quantes classes hi ha registrades i quants jocs hi ha sense necessitat d'accedir al servidor.

En la figura 4 es pot veure el diagrama de casos d'ús explicats anteriorment per l'administrador:

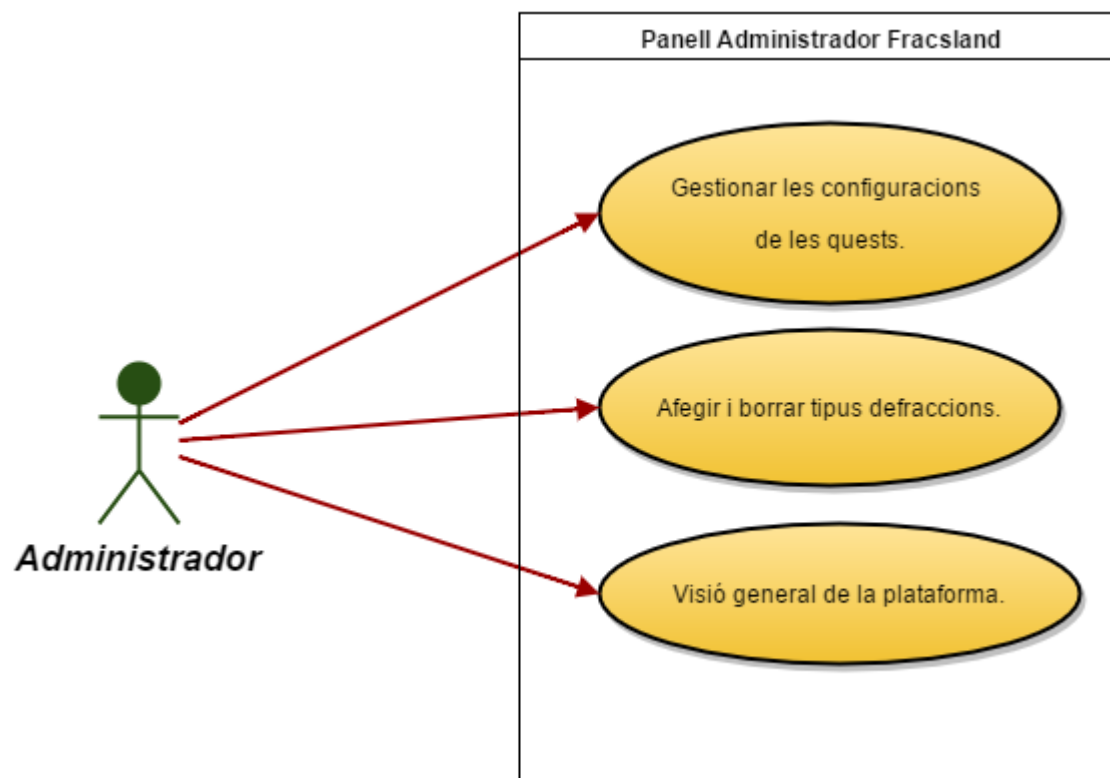


Figura 4: Diagrama casos d'ús administrador

Els professors: Són els encarregats de crear les classes d'estudiants, registrar alumnes en elles, configurar sessions de joc i monitoritzar el progrés dels alumnes dins de l'àmbit acadèmic de Fracslan.

El professor té els següents cassos d'ús:

- UC1: Registrar-se: Un professor s'ha de poder registrar com a tal des de la plataforma.
- UC2: Login i Logout: Un professor ha de poder accedir o tancar la sessió de la compta creada durant el procés de registre.
- UC3: Afegir una nova classe: Un professor ha de ser capaç de crear una nova classe on agrupar els seus alumnes, crear sessions i jocs.
- UC4: Afegir estudiants via fitxer: Un professor ha de ser capaç de afegir els seus estudiants utilitzant un excel, d'aquesta manera pot afegir molts estudiants de cop sense necessitat d'anar un per un des de la plataforma.
- UC5: Afegir estudiants manualment: Afegir estudiants manualment des de la mateixa llista d'estudiants.
- UC6: Editar estudiant: Un professor ha de poder modificar paràmetres de l'alumne, tals com el seu nom, cognom, email, password des de la llista d'estudiants.
- UC7: Esborrar estudiant: Un professor ha de poder donar de baixa a un estudiant de la seva classe esborrant-lo des de la llista.
- UC8: Veure la llista de sessions de joc: Un professor ha de poder veure quines sessions de joc té dins d'una classe.

- UC9: Creació sessió de joc: Un professor ha de poder crear una nova sessió de joc des de la llista de sessions.
- UC10: Veure els detalls d'una sessió de joc: Un professor ha de poder veure quins són els progressos dels seus alumnes dins de la sessió de joc que ha configurat.
- UC11: Veure quines classes té disponibles: Un professor ha de disposar d'una llista on poder veure quines classes té disponibles, per tal d'accedir a una o a una altra segons ho necessiti.

En la figura 5 es poden veure el diagrama de casos d'ús del professor:



Figura 5: Diagrama casos d'ús professor

Els estudiants: Són els jugadors de la plataforma, aquests han de ser capaços de entrar en una sessió de joc configurada per tal de poder jugar i ser avaluats.

Aquests són els casos d'ús que representen el que pot fer un estudiant:

- UC1: Login: Un alumne ha de poder accedir al joc mitjançant el compte creat per el professor.

- UC2: Entrar en un joc: Un alumne ha de poder accedir a un joc creat configurat prèviament per un professor.
- UC3: Gestionar inventari: Un alumne ha de poder gestionar el seu inventari des del joc per així poder organitzar les seves fraccions i objectes com desitgi.
- UC4: Comprar objectes: Un alumne ha de ser capaç de comprar objectes a la botiga.
- UC5: Equipar-se amb objectes: Un alumne ha de ser capaç d'equipar-se els objectes que ha comprat.
- UC6: Visualitzar el seu nivell i estadístiques: Un alumne ha de ser capaç de veure quines son les millores (stats) que li donen els objectes que té equipats.
- UC7: Visualitzar vida actual: Visualitzar la vida que té actualment.
- UC8: Curar-se: Un alumne ha de ser capaç de curar-se per tal de poder continuar lluitant contra els seus enemics.
- UC9: Veure la seva localització en el món: Un alumne ha de veure en tot moment en quina zona del joc es troba i en quin punt d'aquesta.
- UC10: Acceptar quest: Un alumne ha de ser capaç d'acceptar una quest parlant amb el Lord Barus.
- UC11: Lluitar: Un alumne ha de ser capaç de lluitar contra les amenaces que es troba en Fracslan amb les armes que te equipades.
- UC12: Obtenir fraccions: El alumne ha de ser capaç d'obtenir fraccions per les seves quests.
- UC13: Resoldre quest: Un alumne ha de ser capaç de resoldre quests utilitzant les fraccions que ha aconseguit prèviament.
- UC14: Moure's o navegar pel món: Un alumne ha de ser capaç de moure's per el món per tal d'explorar-lo i passar-se el joc.
- UC15: Visualitzar recursos econòmics disponibles: Un alumne ha de veure de quins recursos disposa per tal de poder comprar objectes a la botiga.

En la figura 6 es pot veure el diagrama de casos d'us d'un estudiant en el joc:

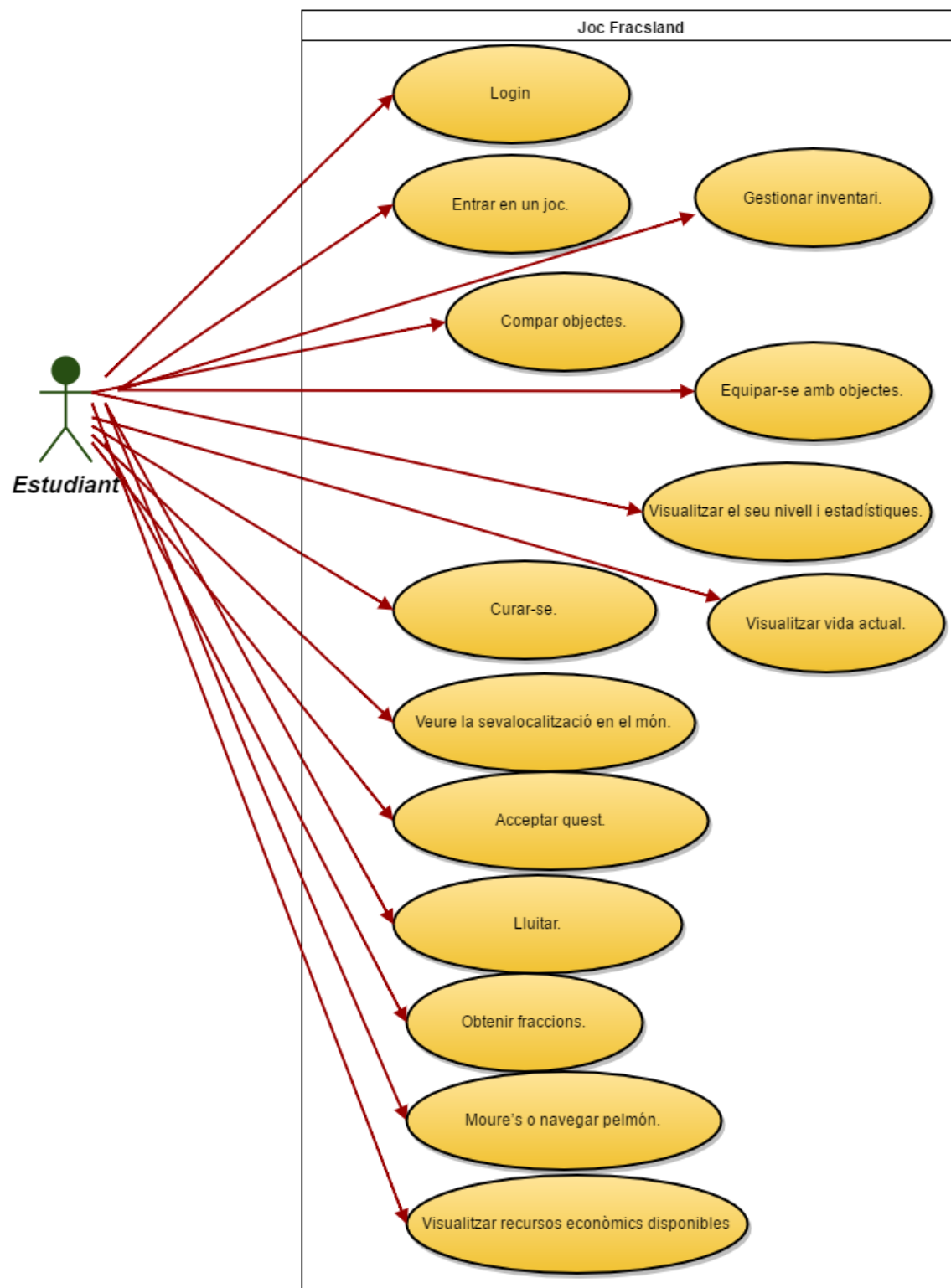


Figura 6: Diagrama casos d'ús alumne

2.3. Arquitectura

La plataforma que conforma Fracsland està estructurada en varis sistemes visibles (veure Figura 7).

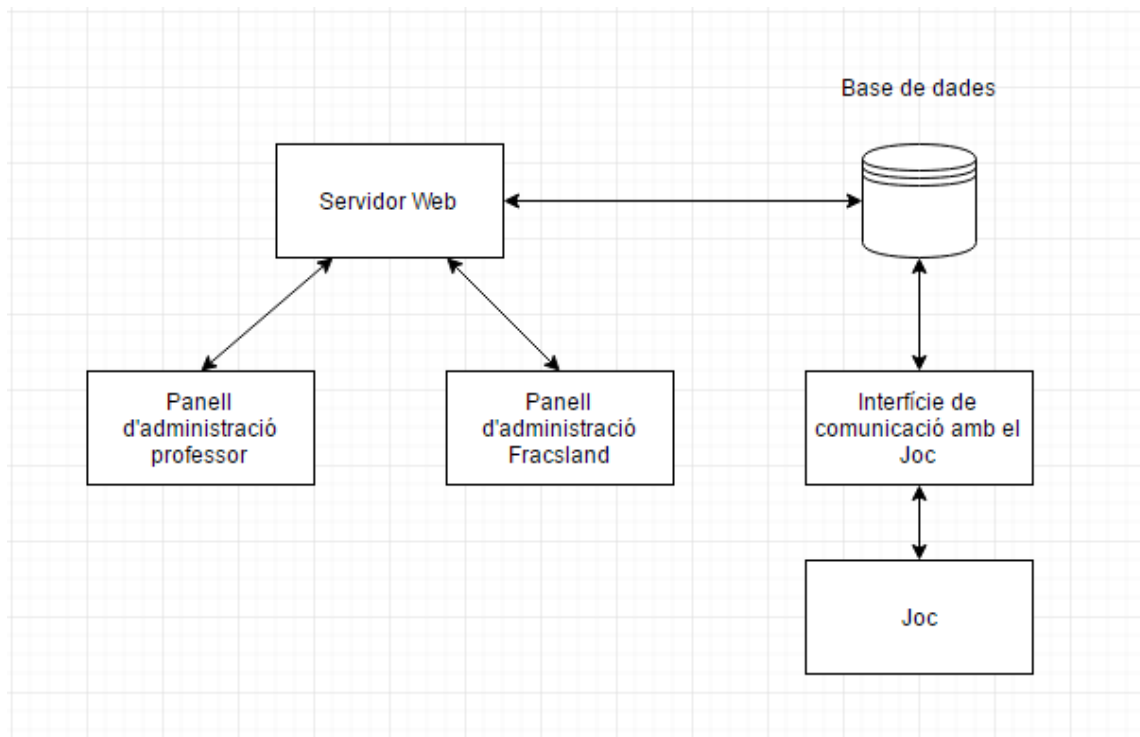


Figura 7: Arquitectura del sistema.

L'aplicació web de Fracsland, permet la generació de configuracions dinàmiques per les tasques que haurà de realitzar l'estudiant dins del joc (quests). Per aquesta raó, s'ha dissenyat un patró de joc que serà configurable des de l'aplicació web. La instanciació del patró de joc en diferents valors en els seus paràmetres definirà el joc final a jugar per l'estudiant.

2.3.1. Servidor Web

Des del punt de vista del professorat, un joc seriós té definits uns **objectius d'aprenentatge** determinats per les competències curriculars del programa docent a cada etapa. Cada objectiu o conjunt d'objectius d'aprenentatge es pot assolir realitzant un conjunt d'**activitats docents** que permeten avaluar el grau en el que l'estudiant ha adquirit la competència associada a l'objectiu. Es pretén dissenyar un panell d'administració pel professor, que des de l'aplicació Web permeti manegar un grup classe, tal que els seus estudiants es puguin guiar de forma col·lectiva o individual i que es puguin veure els progressos durant el joc.

Des del punt de vista de l'administrador, un administrador podrà modificar usant el panell d'administració quins paràmetres rebrà el joc per tal de personalitzar l'experiència de l'estudiant, sense necessitat de modificar el codi del servidor.

Aquesta característica converteix aquest panell d'administració en una base perfecte per poder realitzar altres projectes que requereixin la creació de jocs configurables.

2.3.2. Joc patró

Per a tenir en compte aquests components d'aprenentatge, en aquest projecte s'ha dissenyat un patró per a jocs de rol, o RPG (*Role Playing Game*), en el què els jugadors assumeixen el paper de personatges en un escenari fictici on han de prendre un paper actiu i realitzar accions dins d'una narrativa. Les accions preses tenen èxit o fracàs segons un sistema formal de regles i directrius, normalment amb la presència d'un controlador del joc. Concretament, s'han estudiat els casos per a definir un joc d'aventura gràfica, en la que la forma de jugar (o *gameplay*) és resoldre reptes relacionats amb el context. En el patró de joc proposat, el controlador de joc és el motor del joc que és el responsable de presentar l'escenari fictici, arbitrar els resultats de les accions dels personatges segons les regles del joc, i mantenir el flux narratiu per obtenir l'objectiu del joc. El conjunt d'activitats docents corresponen a **campanyes** en el joc que el jugador ha de passar. Per a passar cada campanya, cal realitzar un conjunt de **missions** que està formada per un o més **reptes**. Els reptes són les accions més bàsiques que el jugador fa en el joc i s'utilitzen per a reforçar els seus continguts didàctics.

Donada una certa narrativa (un viatge espacial, un naufragi en una illa), el patró de joc seriós que es proposa conté els següents components:

- l'objectiu del joc: la narrativa del joc proposa al jugador una fita a assolir, encara que poden haver objectius secundaris (o missions secundàries) que també es poden jugar.
- l'entorn virtual o escenari on es desenvolupa el joc i els personatges ficticis del joc. L'escenari està format per diferents àrees que es poden activar o desactivar per professor.
- les mecàniques i les dinàmiques del joc: venen determinades pel fet de dissenyar un joc d'aventura gràfica on el *gameplay* és resoldre reptes relacionats amb l'entorn fictici del joc i els propis reptes matemàtics. Cada repte es caracteritza per una sèrie d'atributs que podran tenir diferents valors, definits pel professor.
- el jugador i el tipus d'interacció amb l'entorn virtual. El tipus d'interacció està caracteritzada per la navegació i exploració de l'entorn (2D o 3D).
- la interfície del joc i l'aspecte visual i sonor del feedback pel jugador. El tipus d'interacció està caracteritzada per la navegació i exploració de l'entorn (2D o 3D).
- els punts de seguiment del progrés del jugador en el joc i la monitorització de cadascun dels jugadors. Per a realitzar el seguiment del progrés de forma personalitzada, el jugador s'ha d'identificar en el joc.

Donat un patró d'un joc concret, des del panell de l'administració es podran definir les campanyes que han de dur a terme els estudiants per a aconseguir l'objectiu del joc, amb tots els atributs que siguin característics de la campanya en concret. Les campanyes seran un conjunt de missions i reptes tals que el professor podrà configurar els diferents valors dels atributs des del seu panell i que més tard apareixeran en el joc. Aquests atributs són característiques com el valor del denominador a treballar en les fraccions, el grau de dificultat del problema, el tipus de figura geomètrica a construir, etc.

2.3.3. Disseny del joc de Fracland

Per tal de realitzar l'anàlisi i el disseny del joc s'ha utilitzat el següent framework, com a guia de les diferents etapes implicades en el seu desenvolupament (Carey, 2015) (veure Figura 8):



Figura 8: Game design canvas.

A continuació es detallen les dues primeres etapes del framework (setup del joc en blocs de color blau), on es detallen la intenció del joc, amb els seus objectius d'aprenentatge i els tipus de jugadors a qui va dirigit. En el capítol 3 es detallarà el disseny específic del joc (veure blocs de color taronja). En el capítol 3 també es definiran els requisits de la plataforma i els sistemes operatius on caldrà utilitzar el joc (veure segon bloc gris en la Figura 8).

L'objectiu d'aprenentatge d'aquest joc és reforçar el concepte de fraccions aplicat a situacions reals i obrir la possibilitat de reforçar el concepte d'equivalència de fraccions i operacions entre elles. El joc està dirigit a nens i nenes de Cicle Superior de Primària, on el concepte de fracció és conegut però molts vegades no s'ha assolit completament.

El disseny d'aquest joc ha estat partir d'un disseny participatiu amb una classe de sisè de primària de l'Escola del Mar de Barcelona (Rodríguez 2015). En aquest disseny participatiu, els nens i nenes representaven activament el paper de dissenyadors del videojoc, paper que els va permetre reforçar el concepte de fracció tot creant i pensant la història, la mecànica i reptes que definirien el joc seriós final. A partir d'aquest disseny base, el joc s'ha acabat transformant i depurant en un joc d'aventura gràfica situat en un món obert (*open world*) on el jugador pot anar a diferents àrees del joc per a fer els reptes de fraccions que li ha planificat el professor. En aquestes àrees s'han associat diferents mecàniques de joc i escenaris o situacions diverses, de forma que el nen o nena pot realitzar la seva missió en la regió del joc que més li agradi.

El joc es situa en una illa perduda enmig del no res, on el jugador ha naufragat i el seu objectiu principal és construir un motor per a sortir de l'illa. El jugador va obtenint les peces del motor durant la seva progressió del joc. La **campanya** doncs és resoldre un conjunt de missions per a sortir de l'illa. Cada **missió** es correspon a una sessió programada pel professor i està formada per un conjunt de **reptes** (o Quests), que estan relacionats amb fraccions. Els reptes estan repartits per les diferents àrees o regions de l'illa. Alguns dels reptes del joc no estan associats a missions didàctiques sinó que són missions que afegeixen components de diversió i aleatorietat pròpies d'un joc, com pot ser salvar un personatge o lliurar al poblat d'un lleó perillós.

El seguiment del joc es farà durant el mateix joc, i al final de cada sessió, el professor podrà analitzar els progressos dels seus estudiants.

Què podrà configurar el professor?

- Les missions: el professor pot configurar el conjunt de reptes que un jugador ha de resoldre durant el joc, ja sigui en una sessió de classe o a casa. Aquestes missions poden ser diferents per a cada jugador.
- Els reptes (o Quests): el professor pot configurar el valor dels denominadors a treballar en un repte d'un determinat tipus (sigui taules o rodones).
- La personalització dels reptes a cada nen o nena de la classe: el professor pot definir el nivell de joc que cada jugador podrà fer i els diferents reptes associats a cada perfil. Addicionalment, un jugador pot guanyar experiència en el propi joc si segueix practicant fora de la sessió programada pel professor, fet que li proporcionarà més objectes i guanys dins del joc i el permetrà finalitzar més fàcilment les missions futures.

2.4. Conclusions

En aquest capítol s'han exposat els principals requeriments de la plataforma tecnològica a desenvolupar i s'han definit els principals components de la plataforma: el joc de Fracsland i el disseny de l'aplicació Web que permet la seva configuració.

Capítol 3. Disseny del joc Fracsland

3.1. Disseny del joc seriós

Tal i com s'ha explicat en l'apartat d'anàlisi s'està utilitzant un framework per tal de dissenyar el joc d'un mode estructurat, en aquest apartat ens centrarem en la part dedicada al disseny intern del joc seriós (veure Figura 9) i s'ha estructurat el capítol seguint els blocs de cadascuna de les etapes definides pel framework.



Figura 9: Disseny del joc segons el Game Design Canvas

3.1.1. Experiència

Fracslan és un joc "Open-World" (de món obert) amb bonics gràfics 3D on el jugador és lliure de moure's per qualsevol part del mapa en qualsevol moment, aquest disseny s'ha tobat molt oportú ja que la llibertat del jugador és una prioritat.

La decisió de crear un joc amb aquest disseny es fonamenta en el fet de trencar paradigmes que caracteritzen la majoria de jocs educatius, on a un estudiant se li imposa una tasca i només pot fer allò que s'espera d'ell. Fracslan intenta premiar la cooperació, no imposar-la.

El joc comença en Fracstown la capital de Fracslan, on el nostre heroi ha anat després d'arribar a l'illa per accident a causa d'una averia en el motor del seu vaixell.

L'objectiu principal del jugador és sortir de l'illa, però per poder fer-ho necessita un conjunt de peces que li falten per reparar el seu medi de transport, la quantitat d'aquestes peces depèn del nombre de quests el professor hagi configurat.

Resulta que només hi ha una persona que li pot proporcionar aquestes peces, el governador de Fracslan conegut com Lord Barus.

Lord Barus no és conegut entre els habitants de Fracslan per donar les coses franc, així que per tal de donar-li les peces demana a l'heroi que realitzi treballs per ell, però com és una persona honesta i sap que una simple peça d'un vaixell no és una recompensa suficient,

també valora el treball que realitza l'heroi proporcionant-li diamants i or cada cop que aquest resol una tasca. Això li permet al jugador obtenir noves armes i objectes que l'ajudaran, a poder combatre als enemics que hi ha Fracsland més efectivament i lluir un aspecte diferent a la resta.



Figura 10: Lord Barus

Zones:

Fracsland disposa de 4 zones, d'esquerra a dreta, Frozen Woods, Little Beach, FracsTown y FracFarm (veure Figura 11).

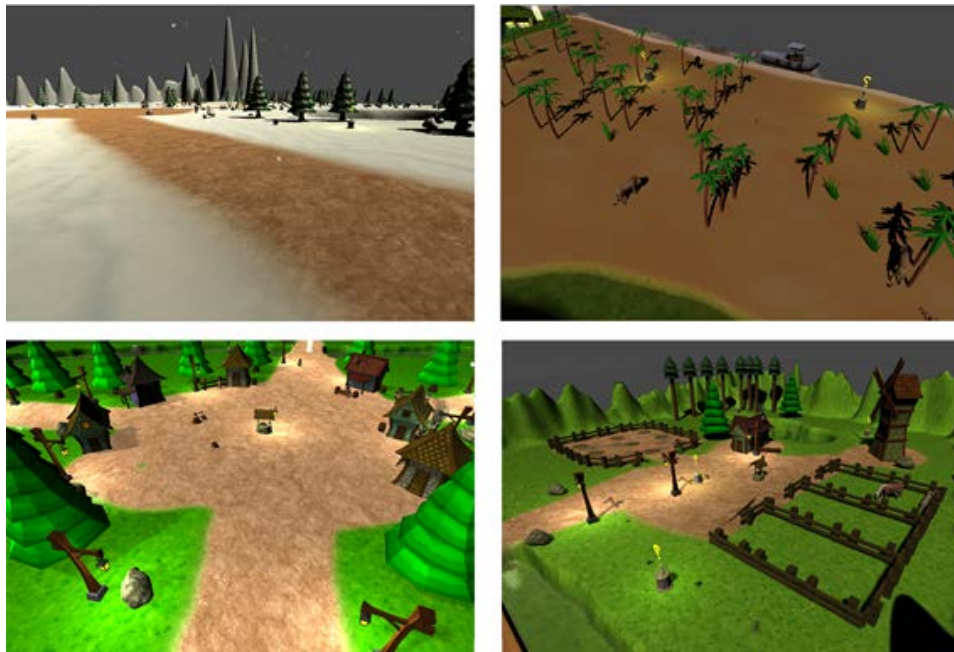


Figura 11: Frozen Woods, Little Beach, FracsTown i FracFarm .

A continuació es detallen les 3 zones dissenyades, segons els tipus de jugadors que s'han considerat (més bèl·lics, més tranquils o jugadors intermitjos). La quarta zona dissenyada correspon a la zona de poble i botiga on el jugador podar canviar i vendre les seves pertinences.

1. Frozen Woods:

Aquesta zona ha estat dissenyada explícitament per el jugadors més bèl·lics de Fracsland.

És el lloc més perillós de Fracsland, aquí es on viuen ferotges llops i en Capità John un ogre amb poders i la seva banda.

És el mapa més extens del joc, amb enormes explanades plenes perills ambientada en un bosc invernal.



Figura 12: Heroi en primer pla i darrere en Capità John rodejat de dos membres de la seva banda i un ferotge llop

2. Little Beach:

Aquesta zona és un lloc intermedi a nivell de violència, el jugador hi ha d'anar cada cop que el pont és destruït per les fortes mareas que hi ha a l'illa. El jugador ha d'anar amb compte amb els lleons, ja que son éssers pacífics fins que t'acostes a ells, però pot triar no apropar-se a diferència dels llops i la banda d'en Capità John que sempre es mouen en busca d'alguna víctima.



Figura 13: Heroi envoltat de lleons

3. FracFarm:

Aquí es d'on prové la majoria de menjar que consumeix la població de Fracstown, consta d'un fèrtil hort, conills, pollastres i una vaca.

És la zona que preferiran els jugadors més pacífics, ja que les missions d'aquesta zona es basen en cultivar la terra i alimentar als animals de la granja.



Figura 14: Heroi rodejat dels animals de la granja davant de l'hort.

4. FracsTown

És on viuen els pacífics habitants de Fracsland, sempre que el jugador vulgui obtenir noves qüestions o fer negocis a la botiga, haurà de venir fins aquí.



Figura 15: Lord Barus (barret cinta blava), al costat del nostre heroi rodejats de dos simpàtics veïns que treballen a FracsTown, posant davant de la botiga.

3.1.2. Mecàniques

En relació a les mecàniques del joc, es descriuran a continuació els personatges que no són jugadors i que tenen comportament “intel·ligent” dins del joc (NPCs) i el funcionament de les Qüestions (o missions) que haurà de realitzar el jugador.

3.1.2.1. NPC:

Fracsland disposa de personatges i criatures que interaccionaran amb el jugador per tal de crear una experiència entretinguda i dinàmica.

Aquests personatges es divideixen en varis grups: amistosos, neutrals i hostils.

Dins dels amistosos, personatges que poden ajudar al jugador, es troben:

- **Lord Barus:** És un personatge que viu a Fracstown, està quiet a l'espera de que el jugador parli amb ell, quan un jugador interacciona amb ell és o bé per acceptar una missió o bé per cobrar una recompensa.

- **Venedor:** Al igual que Lord Barus aquest es troba a Fracstown a l'espera de parlar amb el jugador. Quan un jugador interacciona amb ell aquest ha de permetre al jugador accedir a una botiga virtual on podrà comprar objectes sempre i quan el jugador disposi de recursos econòmics suficients.

Dins dels neutrals, el jugador pot interactuar amb:

- **Animals de la granja:** Aquests també són personatges passius (a priori no es mouen) que poden ser alimentats per el jugador quan aquest es troba en la qüest adequada. Normalment esperen passivament en la granja i si el jugador no té cap qüest no es mouen, però si el jugador accedeix a la granja i el jugador està realitzant una qüest on ha d'alimentar un animal, aquest s'aproparà a ell.

Si el jugador encerta podrà donar de menjar a l'animal que estigui a prop d'ell i si no encerta aquest li donarà una petita empenta.

- **Lleons:** Els lleons es troben en l'interior de la zona de platja de Fracsland, ja que és el lloc més càlid de l'illa. Aquests animals es mantenen immòbils i no fan res a no ser que el jugador s'acosti a ells. En aquest moment es tornaran agressius i lluitaran contra ell.

- **Capità John:** Aquest personatge només apareix si el jugador està fent la qüest adequada, és un personatge que a priori es comporta com els membres de la seva banda buscant al jugador.

Un cop es trobi amb el jugador el seguirà, si el jugador l'ataca no li podrà fer mal a no ser que li llenci la fracció adequada per desactivar els poders que el fan immortal.

Si el jugador intenta tirar una fracció i comet un error, el capità John atacarà al jugador fent-li una quantitat considerable de mal i tirant-lo al terra. En canvi, si el jugador encerta, aquest perdrà els seus poders i es posarà a lluitar contra ell a la desesperada fent-li molt poc mal.

Finalment, els personatges hostils que es trobarà el jugador són:

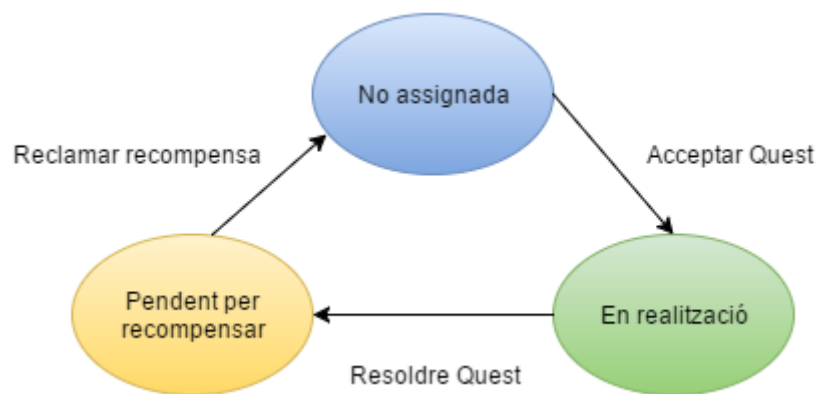
- **Llops:** Aquests personatges es troben en la zona de Froozen Woods, aquests es mouen tota l'estona a localitzacions aleatòries per tal de trobar al jugador i caçar-lo.

- **Membres de la banda d'en capità John:** Aquests personatges es troben en les immediacions d'una espècie de assentament en l'interior de Frozen Woods, al igual que els llops es mouen per la zona per tal de buscar intrusos i eliminar-los.

Adicionalment, s'han dissenyat personatges que no interaccionen amb el jugador però que fan de figurants, donant ambient en el joc, com poden ser els habitants del poble, els ocells a la part de la platja. Aquests personatges tindran animacions pròpies però sense cap efecte en el joc.

3.1.2.2. Quests:

Les quests o missions són la part bàsica d'interacció amb el jugador. Les quests poden ser de diferents formes, associades o no a la resolució de fraccions, es poden tenir missions com trobar a algú, cultivar algun aliment, matar algun personatge, construir ponts, etc. Així, mentre un jugador realitza quests en Fracsland sempre es troba en algun dels següents estats:



Quan un jugador comença a jugar, no té cap quest assignada, així que la primera cosa que haurà de fer és acceptar-ne una.

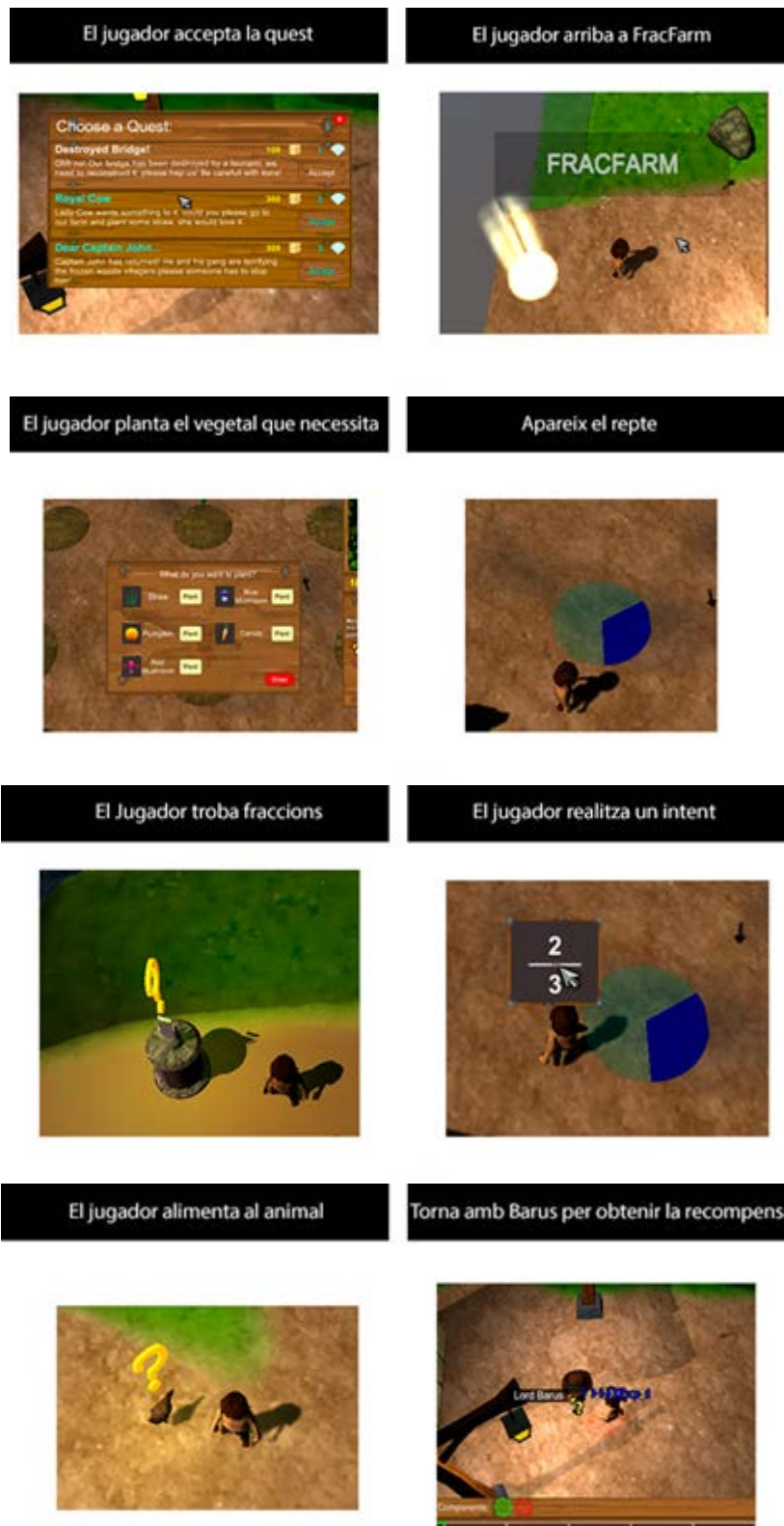
Un cop hagi acceptat una quest, el jugador es trobarà dins de la aventura que hagi triat i el seu objectiu serà resoldre aquesta situació.

Fins que aquest no resolgui la quest que s'ha compromès a realitzar amb el Lord Barus es trobarà en realització, pot equivocar-se tants cops com vulgui el joc tornarà a preparar la quest perquè el jugador pugui resoldre-la.

Finalment quan el jugador resolgui la quest, podrà anar a reclamar la seva recompensa, un cop ho faci es tornarà al primer estat mentre li quedin quests.

Tot seguit es poden veure exemples de quests (en forma de storyboards) que haurà de resoldre el jugador:

1. Alimentar Animals:



2. Reconstruir el Pont:

El jugador accepta la quest



El jugador arriba a Small Beach



El Jugador es pot trobar amb lleons



El Jugador obte una mica de fusta



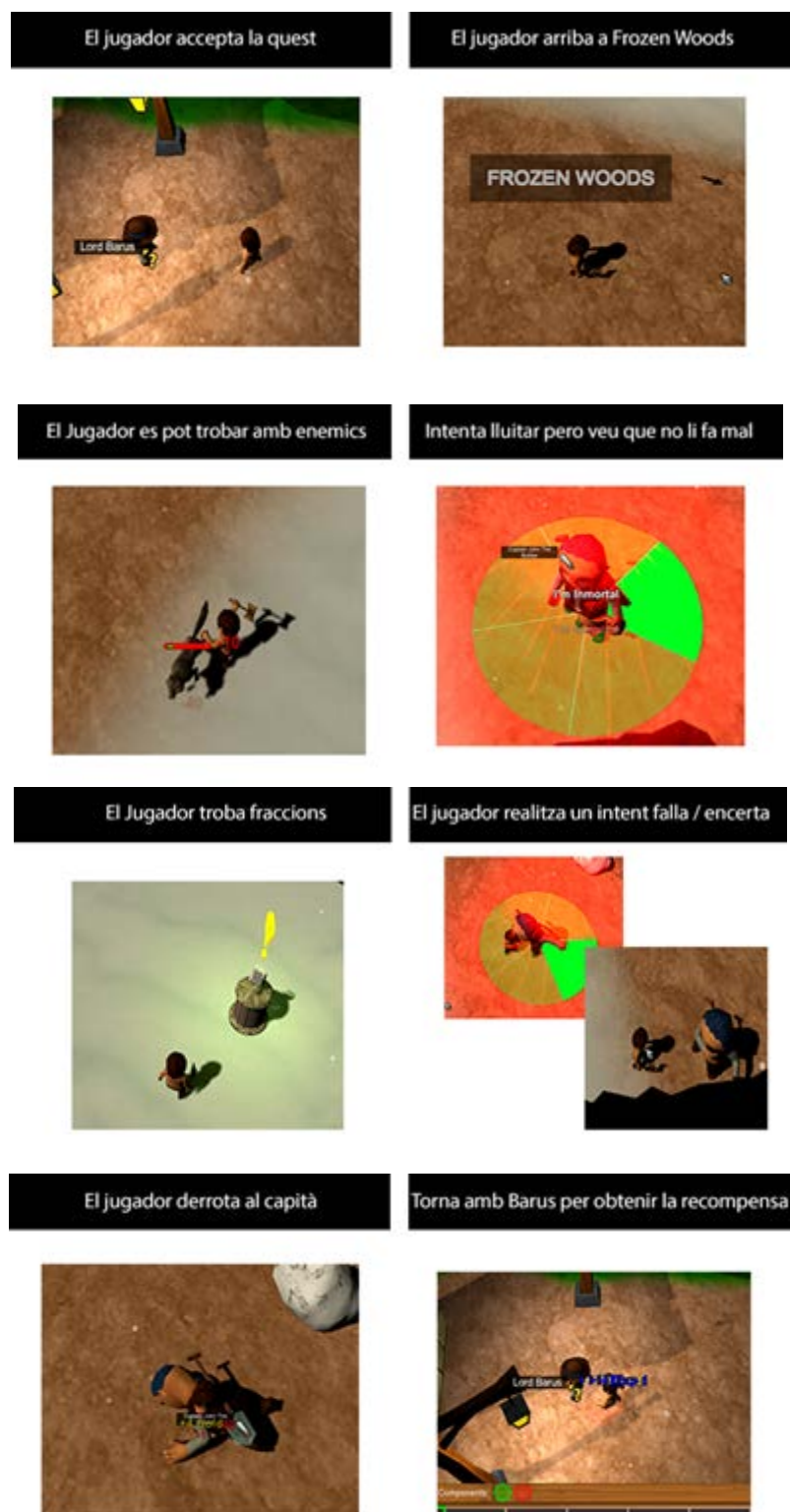
El jugador reb feedback de la seva acció



Torna amb Barus per obtenir la recompensa



3. Lluitar contra el capità John:



3.1.3. Choice

El jugador podrà decidir quines missions fer, triant sempre entre una de cada zona.

Quan el jugador interaccioni amb Lord Barus aquest ha de seleccionar 3 quests a l'atzar (una de cada zona) per tal de que el jugador pugui decidir a quina quest vol fer.

Per tal de que l'alumne faci quests que tinguin la representació que el professor prefereix Lord Barus ha de recompensar amb una mica més d'or i diamants aquelles que treballin la representació de fracció seleccionada pel professor.

El joc ha serà d'un sol jugador, tot i que pot estar obert a la possibilitat de fer-lo multiplayer en futures versions del joc.

3.1.4. Interacció

Per interaccionar amb el joc el jugador disposarà d'una interfície d'usuari per a que pugui accedir al joc, gestionar el seu inventari, acceptar missions, comprar objectes i veure l'estat del sistema i podrà moure el seu personatge amb el ratolí fent click en el lloc del terreny on vol anar o sobre un objecte interactuable.

Més detalladament l'interfície del joc ha de disposar de:

1- Elements per accedir al joc:

- 1.1- Una pantalla de login on el jugador pugui accedir al seu compte o bé recuperar la seva contrasenya i una pantalla on pugui seleccionar a quina sessió de joc vol entrar.

2- Elements dins del joc sempre visibles:

Un cop dins del joc el jugador ha de disposar del següents elements sempre visibles que li mostrin:

2.1- Ell seu nivell, la seva vida actual i quanta experiència li falta per pujar de nivell.

2.2- Un minimapa per poder-se situar, juntament amb el nom de la zona on es troba.

2.3- Quins son els seus recursos econòmics.

2.4- Quins son els seus objectius actuals i on ha d'anar per poder progressar en la seva quest actual.

2.5- Quantes peces del vaixell li falten per aconseguir.

2.6- Un inventari d'accés ràpid on el jugador podrà posar les fraccions que creu que li seran més útils juntament amb 3 botons que li permetin curar-se (si li queden pocions), obrir l'inventari i veure el seu equipament actual.

2.7- Un botó per sortir del joc.

3- Elements dins del joc només visibles durant la seva utilització:

- 3.1- Un inventari en forma de quadricula, on el jugador podrà guardar objectes i fraccions disposant-los al seu gust tant en l'inventari com en l'inventari d'accés ràpid (el que sempre és visible).

3.2- Un lloc on el jugador pugui veure quin aspecte té vist de front que li mostri algunes estadístiques sobre quina força té, quina és la seva vida màxima, nivell ...

3.3- Una botiga on el jugador pugui comprar i equipar-se objectes, on es vegi clarament si el jugador disposa de recursos econòmics suficients per comprar-los.

3.4- Un element que permeti al jugador acceptar una quest de entre una llista amb una breu descripció de la quests un títol i quina serà la recompensa econòmica d'aquestes.

3.5- Si el jugador ha de realitzar alguna interacció específica amb algun element del joc, com pot ser plantar una planta s'ha d'obrir un diàleg on el jugador pugui veure quines opcions té i realitzar-ne una.

4- Elements de notificació:

4.1- Quan un jugador entri en una zona, ha d'existir un element que li mostri clarament a quina zona del joc està accedint que desaparegui al cap d'uns segons.

4.2- Quan una entitat del joc, guanyi vida, en perdi, obtingui or... S'ha de mostrar un text a sobre d'aquesta que ho indiqui i al cap d'un temps desaparegui, per tal de que el jugador pugui estar informat de qui ha guanyat o perdut alguna cosa. Aquesta notificació no ha de ser gaire gran ni interrompre l'interacció amb el joc ja que poden passar bastantes accions cada segon.

4.3- Quan el jugador obtingui una nova fracció, en ser un esdeveniment important dins del joc, se li ha de notificar amb algun tipus de element que aparegui fent una animació i el felicitati. Aquesta notificació en ser significativament més important que les descrites en 4.2 ha de ser també més visible.

4.4- Per tal que el jugador sàpiga si pot interactuar amb una entitat, s'ha de mostrar el nom de l'entitat quan aquest passi el ratolí per sobre i canviar el ratolí per alguna altra icona que resulti representativa de l'acció que es pot portar a terme amb l'entitat en qüestió.

4.5- Quan un jugador pugui de nivell ha de ser visible clarament mostrant a part d'un text que indiqui que ha pujat de nivell mostrant algunes partícules o llums al voltant del jugador.

3.1.5. Feedback

El jugador comença en el nivell 1 i pot pujar de nivell fins a un nivell màxim a determinar (veure l'apartat 3.2.2.3.)

El joc ha de disposar de 2 tipus de recompenses: econòmiques i experiència, aquestes es s'otorguen quan un jugador venç a un enemic, cultiva una planta, alimenta a un animal i/o parla amb Lord Barus al acabar una quest satisfactòriament.

3.1.6. Impact

El jugador ha de tenir la sensació d'estar immers en una aventura plena de d'emocions i diversió. Al ser un joc destinat a alumnes de primària tot i haver-hi combat, han de ser més aviat divertits, mai no resultar massa violents o produir por.

El jugador a part de divertir-se ha de sortir d'una sessió de joc amb una millor comprensió de què representa una fracció i com es pot aplicar en elements del món real.

A mesura que el jugador pugui de nivell, el joc ha de canviar el seu comportament per tal de fer-se una mica més difícil d'un mode progressiu per tal de continuar ser estimulant i desafiant, per evitar caure en la monotonia.

3.2. Desenvolupament tecnològic del joc

En aquest apartat s'explicarà com s'ha implementat la part del joc corresponent a la part remarcada en blau en l'arquitectura que es mostra en Figura 16, tenint en compte les decisions de disseny de joc que s'han explicat en l'apartat anterior (3.1).

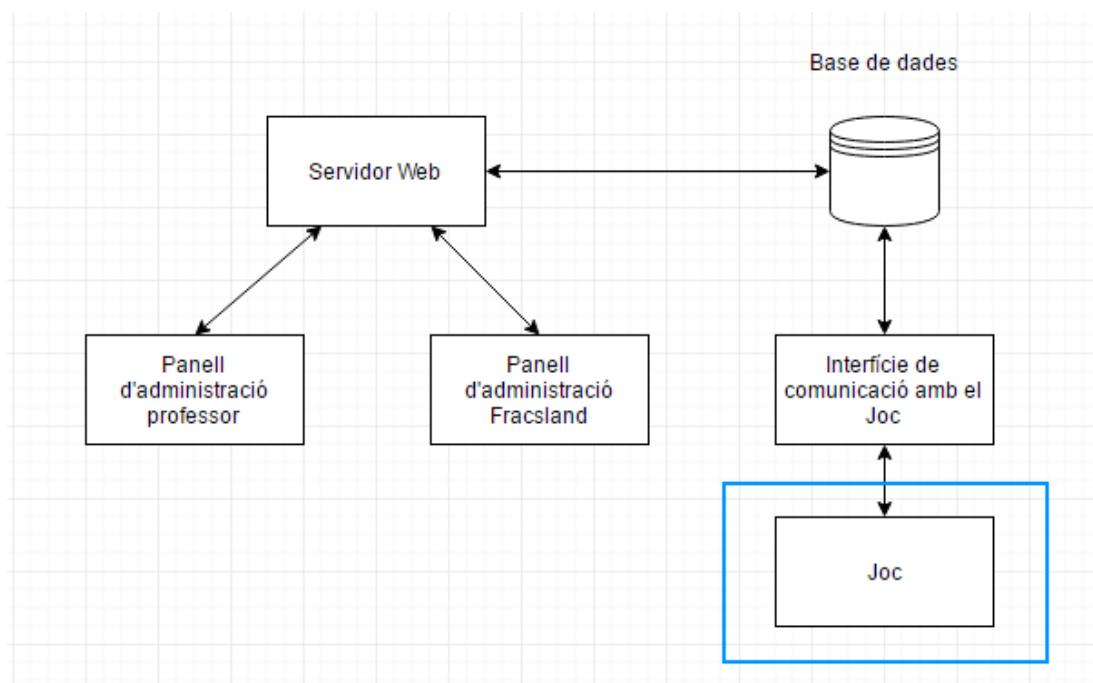


Figura 16: Part dedicada al joc dins de l'arquitectura del sistema.

3.2.1. Tecnologia utilitzada

Unity3D

El client de Fracsland utilitza Unity3D que és un motor de videojocs multi-plataforma molt popular, creat per Unity Technologies. (Unity, 2016)

S'han analitzat altres alternatives com utilitzar OpenGL (Khronos, 2016) i C++ per implementar el motor de renderització i usar C++ i llibreries matemàtiques com GLM (OpenGL, 2016) per desenvolupar les físiques i mecàniques del joc. Per una banda el gran

avantatge d'aquestes tecnologies respecte Unity3D, és que proporcionen un control a més baix nivell del motor del joc, permetent així realitzar un motor a mida per Fracslan. Unity3D, en ser un motor de propòsit general¹ inclou moltes funcionalitats que no són necessàries per al desenvolupament del joc. A part com ha estat pensat per treballar amb múltiples plataformes, els seus temps de compilació són lents² degut a les operacions intermèdies que ha de realitzar per generar l'executable. El gran inconvenient d'utilitzar directament OpenGL i C++ és el seu lent procés de desenvolupament, ja que cal desenvolupar cadascuna de les prestacions a molt baix nivell.

D'altra banda Unity3D permet centrar-se en el desenvolupament de la lògica del joc i les comunicacions amb el webservice, fet que resulta beneficiós quan l'objectiu de la feina requereix desenvolupar una plataforma sencera des del frontend al backend en un temps relativament curt.

Concretament, Unity3D inclou moltes funcions d'alt nivell que s'han utilitzat i no s'ha hagut d'implementar, algunes d'elles es llisten a continuació:

- 1- Importació i visualització de models 3D de forma totalment transparent al desenvolupador.
- 2- Sistemes d'il·luminació.
- 3- Portabilitat per Windows, Mac i Linux.
- 4- Físiques.
- 5- Pathfindig (navegació)
- 6- Interfícies d'usuari.
- 7- Escenes.

Una part molt potent de Unity3D és el seu editor, que permet al desenvolupador treballar en un entorn visual, que ajuda a poder comprovar les noves funcionalitats implementades àgilment.

Unity3D també disposa d'una comunitat de desenvolupadors molt activa que ha generat molta documentació, fet que resulta molt útil quan sorgeixen problemes o dubtes en la implementació del treball a realitzar.

Sobre C#, el llenguatge que s'ha utilitzat per codificar les mecàniques del joc, interfícies d'usuari i les comunicacions amb el webservice, ha estat triat perquè permet treballar usant la potència d'abstracció de la programació orientada a objectes permetent realitzar herències, polimorfismes... i simplificant el codi font i fent-lo més concís fàcil de mantenir.

És evident que existeixen motors que ofereixen prestacions similars a Unity3D com CryEngine de Crytek (Crytek, 2016) o UnrealEngine de Epic Games (Epic, 2016). Aquí el factor més important per triar Unity3D sobre aquestes alternatives ha estat els coneixements ja adquirits sobre Unity3D, decremant així el temps d'aprenentatge i de desenvolupament.

¹ Unity3D ha estat desenvolupat per creat tot tipus de jocs i simulacions, per un ventall molt gran de plataformes.

² En comparació a un motor escrit a mida en C++ i OpenGL

3.2.2. Desenvolupament

Un cop realitzat l'anàlisi i el disseny del joc en aquest apartat es detalla com s'ha dissenyat i s'ha implementat el joc.

El joc tal i com s'explica més endavant conté unes 110 classes que serveixen per controlar components específics, ja sigui de la GUI, com NPCs, comunicacions amb el servidor, etc...

Però hi ha 5 classes principals que governen el joc relacionades tal i com es veu en la Figura 17, aquestes es demanen informació entre elles per poder gestionar tot lo referent al joc:

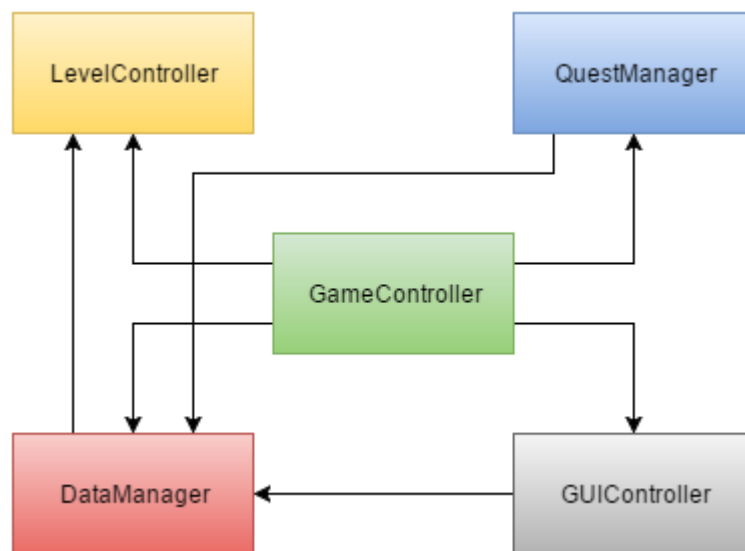


Figura 17: Diagrama Controladors

A continuació es detallen el disseny i aspectes d'implementació dels diferents blocs del disseny del joc presentats en l'apartat 3.1.

3.2.2.1. Mecàniques

1. Qüests

Totes les qüests que hi ha en Fracslan, segueixen la mateixa lògica. Això te l'avantatge que un cop el jugador n'hagi fet una, podrà intuir com és fan la resta.

Quan una qüest o missió és acceptada pel jugador es seguirà la lògica que es mostra en la Figura 18, tan bon punt el jugador estigui dins de l'àrea on s'ha de realitzar la qüest

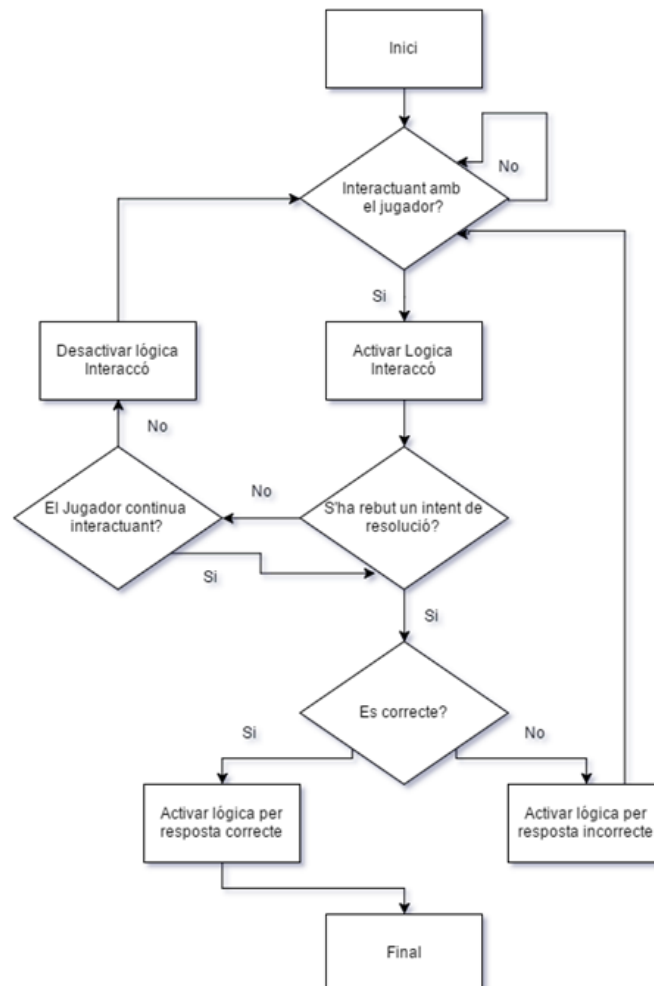


Figura 18: Diagrama de flux quest

El primer que fa una qüest es comprovar si està interactuant amb un jugador, si no ho està es manté a l'espera fins que ho estigui.

Un cop el jugador comença a interactuar amb el jugador la qüest activa la seva lògica d'interacció.

Un cop el jugador comença a interactuar amb la qüest es queda a l'espera per tal de poder gestionar els seus intents de resoldre la qüest, en cas de que aquest ja no es trobi interactuant amb la qüest tornarà al estat inicial.

Si es rep un intent de resolució, el sistema comprovarà si aquest es correcte o no, si es correcte s'activarà la lògica corresponent si no ho és s'activarà la lògica per gestionar un intent fallit i tornarà a l'estat inicial.

2. Arquitecte Virtual

Es vol configurar l'escenari per a que no es vegi sempre de la mateixa manera, per exemple la distribució d'arbres, enemics, altars de fraccions...

Aquest sistema permet el posicionament d'elements pseudo-aleatòriament en temps d'execució definint prèviament en l'editor zones on es volen posicionar entitats en concret, en el moment de creació de l'escenari d'una quest.

Un nivell disposa de varies zones corresponents definides per l'espai físic del que es disposa.

Utilitzant aquest arquitecte pots dividir aquestes zones en rectangles, on cada rectangle correspon a una possible zona d'aparició d'una entitat en concret.

D'aquesta manera es poden definir varis rectangles superposats per tal de poder crear més d'una entitat per cada zona (ja que cada rectangle correspon a una entitat).

En la imatge de la Figura 19 es pot veure (a) l'escenari del nivell i (b) el nivell amb les àrees designades per posar-hi elements.

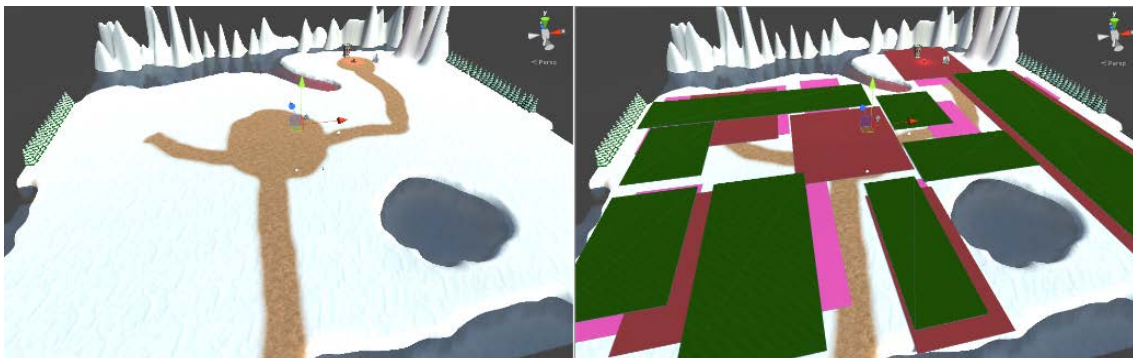


Figura 19: A l'esquerra (a) es pot veure el mapa amb els components del arquitecte amagats i a la dreta (b) el disseny de les zones d'aparició d'entitats.

A part s'ha definit una estructura de tipus enum que permet canviar el tipus de entitat que es vol posicionar des del mateix editor de Unity (d'aquesta manera si es vol canviar l'entitat que posiciona un rectangle en concret no cal crear-ne un de nou, només cal canviar el valor ta i com es veu en la figura 20):

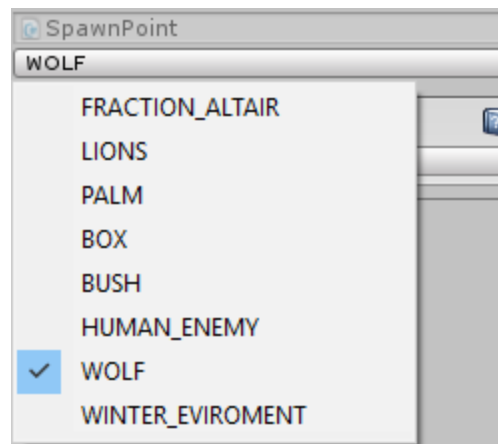


Figura 20: Selector d'entitats disponibles per ser posicionades en una zona.

Per realitzar aquest posicionament, es segueix el següent algorisme, aquest algorisme es divideix en 2 parts, temps de creació del nivell i temps d'execució).

Temps de creació del nivell:

- 1- Primer es defineixen la densitat i la probabilitat en la que apareixeran les entitats en la zona marcada.
- 2- Es calculen els punts on poden aparèixer les entitats basant-se en la densitat definida en la primera etapa del algorisme i es barreja la llista perquè sigui més difícil de predir on apareixeran les entitats, ja que sinó podrien aparer alineades i agrupades, ja que hi ha casos en que les entitats estan limitades (els enemics i els altars de fraccions tenen un límit).

Temps d'execució:

- 3- Es comprova quina entitat es vol fer aparèixer.
- 4- Per cada possible localització es comprova aleatòriament si posar o no una entitat (si la entitat te límit, es comprova també si encara en queden per posar).
- 5- Hi ha entitats que requereixen un mínim d'elles per assegurar la jugabilitat, sinó s'hagués assolit aquest límit, es torna al punt 4.

Aquesta construcció semi-automàtica permet donar visualitzacions molt diferents del mateix escenari del joc.

En les següent figura (figura 21) es poden veure varies distribucions generades per l'arquitecte virtual:

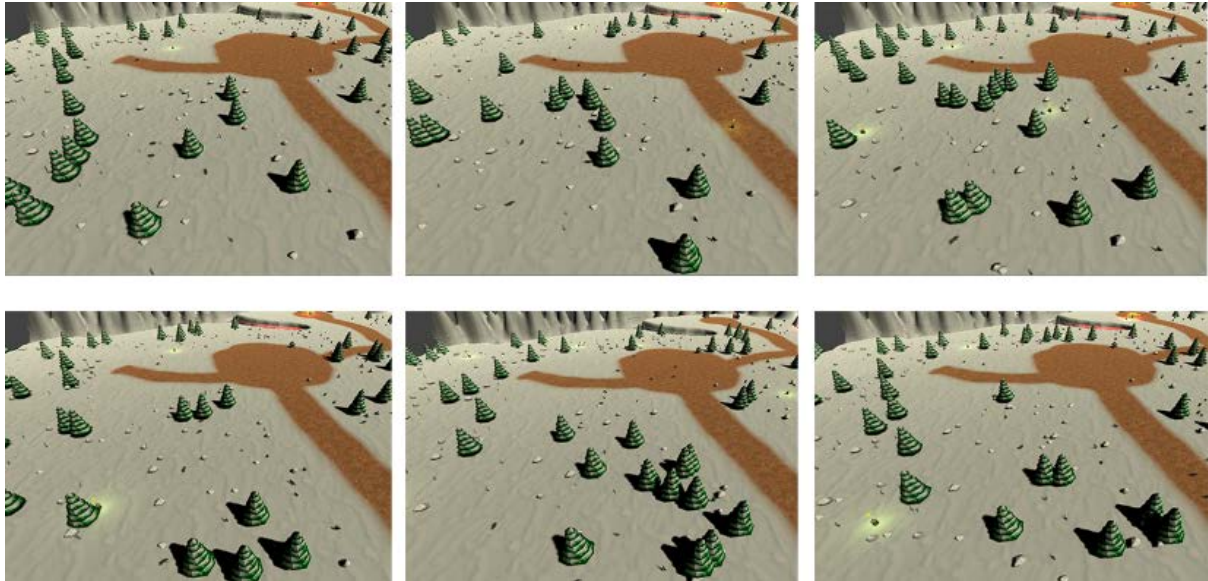


Figura 21: Diverses generacions de Frozen Woods generades amb l'arquitecte virtual

3. Moviment de les entitats

Per moure les entitats s'ha dissenyat una classe anomenada MovementManager, d'aquesta manera totes les entitats que es poden desplaçar tenen un mecanisme equivalent.

El MovementManager pot desplaçar els objectes de dues maneres:

- La primera d'elles és navegar sobre els navmesh integrats en Unity3D (s'ha de demanar que es calculi explícitament quan es crea el terreny). D'aquesta manera l'entitat pot navegar per el nivell fent càlculs per tal de triar camins que no passin a traves d'altres elements.

En la figura 22 es pot veure la navmesh (zona blava) per on podrà navegar una entitat en Froozen Woods, tal i com es pot veure aquesta engloba totes les parts on el mapa no te grans desnivells deixant els llacs com a zona no navegable:

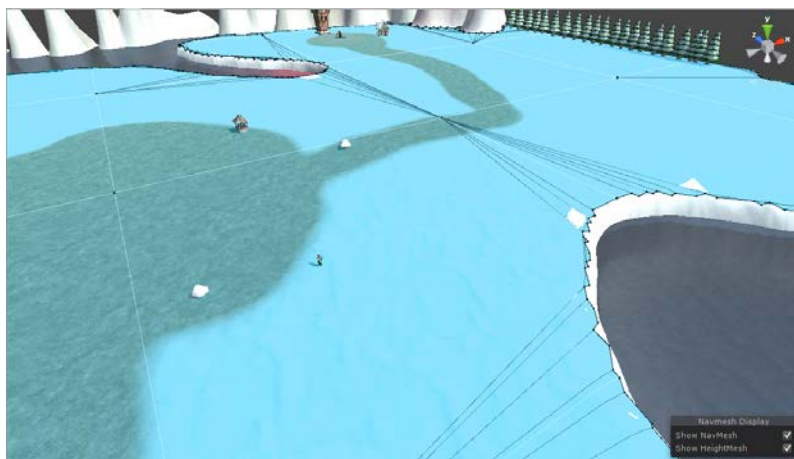


Figura 22: Navmesh generat per recorre Frozen Woods

- La segona és moure un objecte ignorant la navegació d'un punt A a un punt B interpolant la posició del objecte entre els punts a mesura que el temps avança (útil per exemple per moure ocells), d'aquesta manera es pot forçar a un objecte a travessar camins que utilitzant el navmesh no podrien a recórrer.

4. Esdeveniments del joc

Com s'ha explicat prèviament, Fracsland utilitza un sistema d'esdeveniments per tal de controlar els progressos del jugador dins del joc.

El funcionament d'aquest sistema és el següent:

1-Una entitat crea un esdeveniment d'un tipus, actualment hi ha la següent llista d'esdeveniments disponibles:

1.1- PALM_EVENT.

1.2- FRACTION_EVENT.

1.3- FRACTION_ALTAIR.

1.4- SOLVE_EVENT.

1.5- FARM_EVENT.

1.6- PLANT_EVENT.

1.7- ANIMAL_FEED_EVENT.

1.8- OPEN_QUEST_DIALOG.

1.9- OPEN_SHOP_DIALOG.

2- Una sèrie d'entitats es registren en els esdeveniments d'un tipus definint un mètode que s'ha de cridar quan l'esdeveniment es dispari.

3- Quan un esdeveniment es dispara, totes les entitats registrades en l'esdeveniment del tipus que s'ha disparat, es criden tots els mètodes de les entitats registrades a l'esdeveniment.

4- Una entitat pot decidir si deixar d'estar registrada en un esdeveniment quan ho desitgi.

Aquest sistema és avantatjós perquè permet reaprofitar els events per diverses entitats sense necessitat de programar-los explícitament per una entitat en concret (desacoblament), ja que només assignant un event a una entitat aquesta a partir d'aquest moment serà capaç d'utilitzar-lo.

Un exemple molt clar d'això es el SOLVE_EVENT, quan es crea una quest només cal assignar amb quin objecte del joc s'haurà d'interactuar i a partir d'aquest moment aquest ja acceptarà fraccions i permetrà avisar a la seva quest de que l'usuari ha realitzat un intent de resolució.

5. Element de la interfície

Tots els elements que es poden amagar o mostrar en la interfície d'usuari hereten o utilitzen de la mateixa entitat (UIElement).

Aquesta classe associa a l'entitat una propietat anomenada "canvas group", que permet definir l'alfa d'un element (transparència) i tots els seus fills, decidir si bloqueja raycasts o no i definir si es poden interactuar amb els objectes de la GUI associats a aquell element.



Figura 23: Canvas Group associat a un element de la GUI.

D'aquesta manera es poden mostrar o amagar elements sense necessitat de desactivar els objectes, això té l'avantatge que l'element continua actualitzant-se tot i no ser visible, evitant així haver d'utilitzar la filosofia MVC per els elements de la GUI, ja que una sola classe ho pot controlar tot.

6. Elements interactius

Dins del joc hi ha elements que poden interactuar amb el jugador, ja sigui fent clic en ells o bé deixant-hi anar una fracció a sobre.

Per poder realitzar aquesta funcionalitat s'ha definit una entitat que des de l'editor permet definir quin tipus d'interacció acceptarà un element.

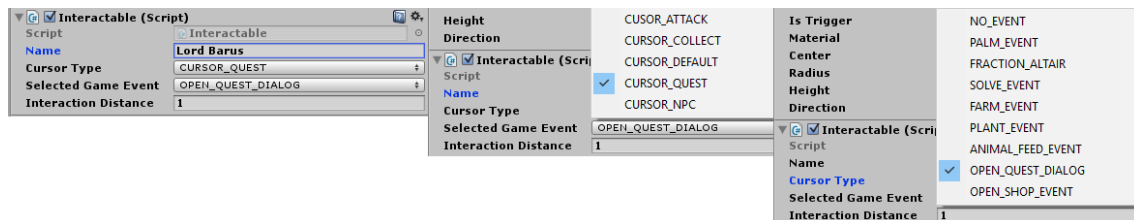


Figura 24: Diferents tipus de paràmetres que son modificables en un objecte de joc que tingui associat el codi "Interactable"

Des de l'editor es poden definir varies coses:

- 1- Quin és el nom que mostrarà l'inspector quan el jugador posicioni el ratolí a sobre d'ell.
- 2- Quin cursor es mostrarà.
- 3- Quin esdeveniment es dispararà quan el jugador faci clic a sobre de l'entitat.
- 4- Quina és la distancia mínima on ha d'estar l'heroi, perquè la interacció es porti a terme.

Si l'heroi no es troba dins del rang definit per la distancia mínima, caminarà fins estar dins d'ella mitjançant el MovementManager.

7. Pie Fraction

Aquesta entitat permet generar la representació d'una fracció en forma circular, com es veu en la figura 25.

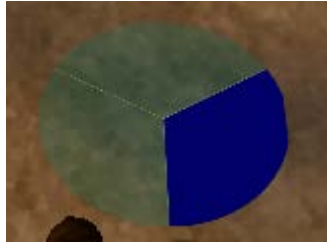


Figura 25: Pie Fraction generada dinàmicament

Per realitzar aquesta tasca es crea una textura dinàmicament treballant píxel a píxel seguint els següents procediments:

1- Es crea una textura de 512x512 píxels.

2- Es defineix una variable "r" que representa el radi del cercle definit de la següent manera:

$$r = \frac{\text{textureWidth}}{2}$$

3- Es dibuixa el cercle mitjançant el següent algorisme:

Per cada píxel "i" i "j":

$$d = \sqrt{(i - r)^2 + (j - r)^2}$$

Si d és inferior al radi → s'associa color al píxel sinó es deixa transparent.

4- Si es vol generar un cercle amb una divisió hem acabat, sinó continua.

5- Es calcula quin serà l'angle entre dues línies en el cercle mitjançant el següent càlcul:

$$\alpha = \frac{360}{\text{numDivisions}}$$

6- Es defineix la variable que guardarà el centre del cercle:

$$\text{center} = \left(\frac{\text{textureWidth}}{2}, \frac{\text{textureWidth}}{2} \right)$$

7- S'inicialitza una llista on es guarden els punts centrals de les rectes que es dibuixaran aquests punts representen el centre d'una recta no confondre amb el centre de la "Pie Fraction" (centralPoints).

8- Des per cada divisió que es vol fer ($i < \text{numDivisions}$) començant per $i=0$:

8.1- Es calcula el punt on acabarà una recta que es comenci a dibuixar des del centre tenint com informació el radi i l'angle:

8.1.1- Es calcula l'angle actual:

$$\text{angle} = i * \alpha$$

8.1.2- Es converteix l'angle a radians:

$$angle = angle * \frac{\pi * 2}{360}$$

8.1.3- El punt on acabarà la recta es calcula de la següent manera:

$$endPoint = (centerX + radius * \cos(angle), centerY + radius * \sin(angle))$$

8.2- S'aplica l'algorisme "Bresenham-based supercover line" (<http://lfc.univ-fcomte.fr/home/~ededu/projects/bresenham/>) per obtenir una llista de posicions que defineix la recta mitjançant els dos punts calculats anteriorment.

8.3- Es pinta cada punt de la recta obtinguda en 8.2, el color que es vol per les divisions.

9- Es pinten les regions que es vol visualitzar plenes:

9.1- S'inicialitza una llista on es guardaran els punts de difusió que són els punts des d'on es començarà a difondre el color per cada regió (diffusePoints).

9.2- Si el nombre de divisions és major que 2 (es necessita que l'angle entre les rectes dels punts que estem tractant no sigui de 180°):

9.2.1- Per cada punt de "centralPoints":

9.2.1.1: S'obté el punt central de la regió (el punt de difusió), mitjançant l'algorisme usat en 8.2 entre el centralPoint actual i el següent i es guarda en diffusePoints.

9.2.2- Si el nombre és 2, es guarden 2 punts en la llista diffusePoints usant l'algorisme vist en 8.1 usant el primer punt de central points com origen, un angle de 90° per el primer punt, un angle de 270° per el segon i un radi de 20, així obtenim punts a banda i banda de la recta.

9.3- Mentre el nombre de zones colorejades sigui inferior al nombre de zones que es volen pintar:

9.3.1- S'utilitzen els punts de difusió que s'han guardat en diffusePoints com a entrada per el realitzar el següent.

9.3.2- S'inicialitzen dos diccionaris visitedNodes i nodesToVisit (aquests guardaran quins punts s'han visitat i quins no).

9.3.3- S'inicialitza una llista que guardarà quins punts s'han de pintar al final del procediment anomenada nodesToPaint.

9.3.4- Mentre quedin elements en NodesToVisit.

9.3.4.1- S'obté i s'esborra el primer punt de nodesToVisit.

9.3.4.2- S'obté el color de la textura en el punt on s'està treballant.

9.3.4.3- Si el color del punt on es treballa és igual al color que s'ha definit com "color buit" de la textura.

9.3.4.3.1- Es guarda el punt en nodesToPaint.

9.3.4.3.2- Per els punts que hi ha a sobre, a dreta, a esquerra i a sota del punt actual, si aquest no està en visitedNodes ni en nodesToVisit, es guarda en nodesToVisit.

9.3.5- Es pinten tots els píxels que hi ha en nodesToPaint del color ple (una altre aproximació seria pintar els punts en el moment de trobar-los, en aquest cas s'ha fet així perquè Unity3D facilita pintar molts punts de cop).

10- Es canvien les mides de la textura generada per a què s'adapti a les mides que s'ha de mostrar en el joc.

7. API

Aquesta entitat permet als components del joc crear peticions al servidor.

Quan un element del joc vol fer una petició es crea un objecte "petició" (Request) que conté tot lo necessari per gestionar el procés, com la URL on es realitzarà la sol·licitud, paràmetres que s'enviaran i els mètodes que s'executaran quan es rebí la resposta depenent del tipus de resposta rebuda, en el diagrama de flux de la Figura 26 es veu el funcionament d'aquest sistema.

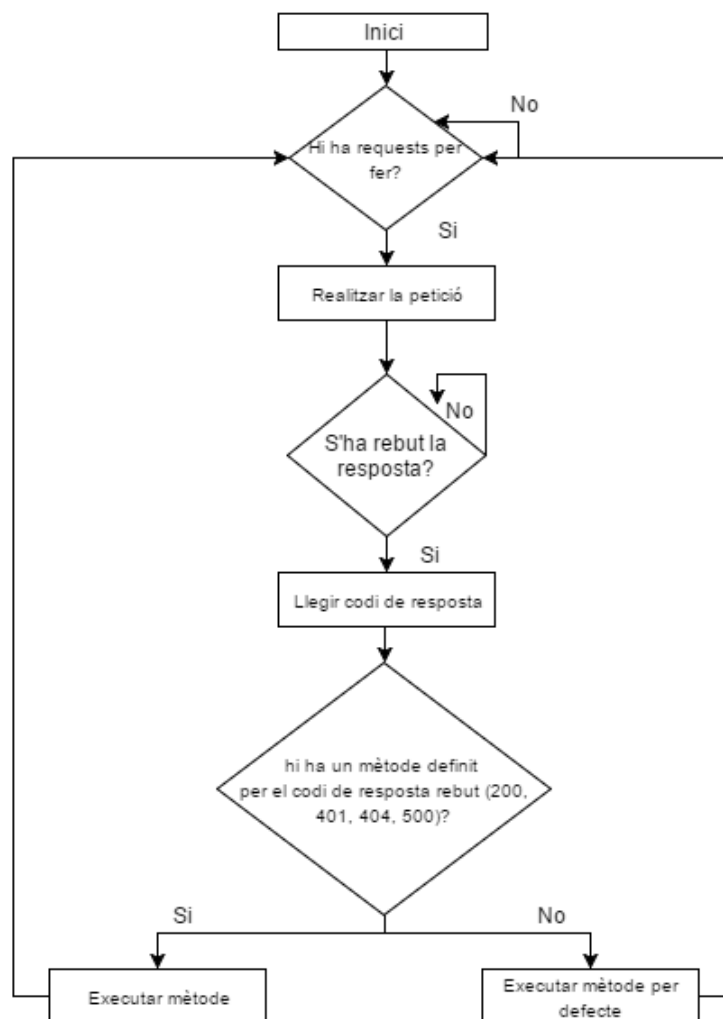


Figura 26: Diagrama de flux del sistema de requests.

Aquest sistema el que fa és el següent:

- 1- Comprova si hi ha request per fer, si no és així es queda a l'espera.
- 2- Un cop hi ha una o més request per fer realitza la primera que te disponible.
- 3- Mentre el servidor no contesta aquest espera, fins el moment de la recepció de la resposta.
- 4- Un cop ha rebut la resposta comprova quin codi de resposta s'ha rebut (200 vol dir que tot ha anat bé, 401 vol dir que l'usuari no esta autoritzat sol aparèixer en els errors d'autenticació, 404 vol dir que la URL on l'usuari ha intentat accedir no existeix i per últim 500 vol dir que hi ha hagut un error intern en el servidor).
- 5- Un cop llegit el codi el s'executa el mètode associat a la resposta (per exemple en un login si es rep 200 s'accedirà al joc, si es rep un 401 mostrarà al usuari que hi ha hagut un error d'autenticació).

3.2.2.2. Interacció

Interfície d'usuari

La interfície d'usuari s'ha dissenyat perquè l'estudiant pugui gestionar els objectes, introduir dades i poder accedir al joc d'un mode intuïtiu:

Les interfícies del joc han estat dissenyades per 3 tipus de pantalles diferents.

Aquí es poden veure uns exemples de les interfícies que s'han dissenyat perquè el jugador pugui accedir a un joc.

S'han dissenyat aquestes dues interfícies (veure figura 27) perquè un jugador pugui fer login (esquerra) i triar en quina partida vol entrar (dreta) s'ha decidit crear un diàleg flotant que saluda al jugador creuant la pantalla per tal de fer sentir al usuari que el joc es refereix a ell:

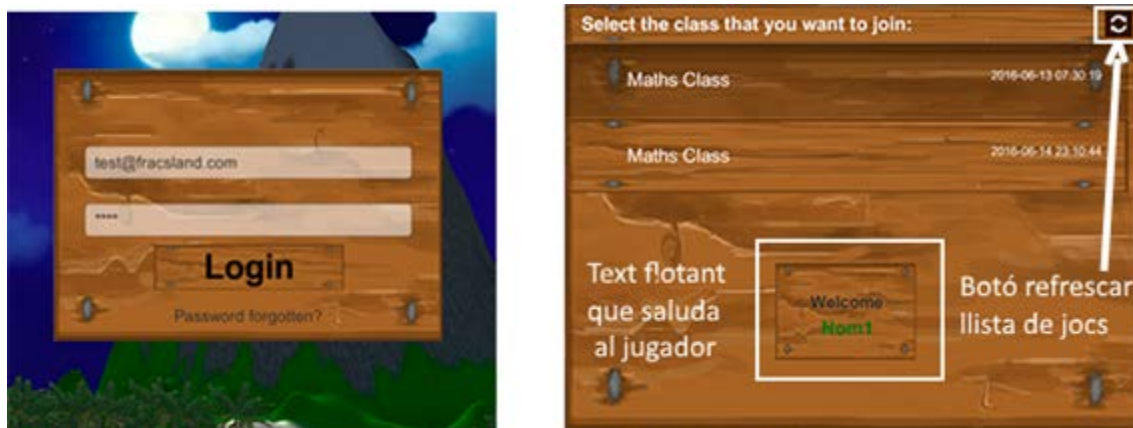


Figura 27: Pagina de login (esquerra) i pantalla de per accedir als jocs disponibles (dreta)

Aquesta (veure figura 28) és la interfície d'usuari que veu l'usuari mentre està en el joc, tot seguit s'expliquen els components d'un en un incloent els no visibles en la següent imatge.



Figura 28: Interfície alumne dins del joc

Les següents parts de la interfície són sempre visibles, i mostren al jugador informació sobre l'estat del joc que li resultarà útil per veure els seus progressos i opcions:

Aquesta primera part (veure figura 29) està dedicada a la vida del jugador (en forma de fracció circular), el seu nivell actual i l'experiència que li falta per pujar de nivell en forma de barra, quan la part blava arribi al final el jugador pujarà de nivell.



Figura 29: Part de l'interfície dedicada a la vida, nivell i expència del jugador

La següent part conté varies dades, en la barra superior es poden veure quantes peces li falten al jugador per aconseguir, a mesura que el jugador vagi fent qüestions, aquestes peces s'aniran posant de color verd.

També es pot veure l'inventari d'aquest ràpid, que no és més que una part de l'inventari que sempre està visible, d'aquesta manera el jugador pot tenir una manera d'accedir ràpid als objectes que creu que li resultaran útils a curt termini.

Per últim, a la dreta de tot es pot veure quantes pocions li queden (per beure'n una ho pot fer aparentant la tecla q o bé fent click en la icona (pensat per pantalles tàctils), un accés ràpid al inventari i un accés ràpid a el seu equipament actual.

Tot seguit es pot veure la part descrita anteriorment (veure figura 30):



Figura 30: Inventari d'accés ràpid, pocions, botó d'accés a l'inventari i botó d'accés als stats del jugador.

En la següent figura (figura 31), es pot veure la una fletxa que indica al jugador cap a quina direcció ha d'anar per complir els objectius que tingui actualment.



Figura 31: Fletxa que indica al jugador on ha d'anar.

En la següent imatge (figura 32) es pot veure el mini-mapa que serveix al jugador per tenir una idea de quina forma té el mapa on es troba i on està ell en relació a tota la illa.



Figura 32: Minimapa

El jugador també disposa d'aquesta part de la interfície (figura 33) dedicada a mostrar quins són els seus estalvis, i quins són els seus objectius actuals:



Figura 33: Objectius del jugador i recursos econòmics

Fins aquí les parts visibles sempre, ara tot seguit s'explica les parts que són visibles només quan el jugador n'ha de fer ús, com l'inventari, el selector d'accions, o la botiga.

Inventari

L'inventari de Fracsland és una graella on el jugador veurà quins objectes té disponibles i en quina quantitat sempre que sigui un objecte acumulable, les fraccions en aquest cas no ho són.

S'ha dissenyat perquè que l'usuari pugui arrossegat i deixar anar els continguts de l'inventari, per tal de poder-los ordenar al seu gust o inclús passar-los a l'inventari d'accés ràpid o viceversa, en la següent imatge (figura 34) es pot veure com és aquesta interfície:



Figura 34: Inventari

Selector d'accions:

Fracsland també disposa de menús de selecció d'accions, les següent figura (figura 35) mostra a l'esquerra com el jugador pot triar la missió que li vingui de gust tot fent click sobre "Accept" (les blaves son missions que aporten més guany al usuari, ja que són les que treballen les fraccions amb la forma que prefereix el professor) i a la dreta es mostra el diàleg que li permet plantar un vegetal fent click en "Plant".

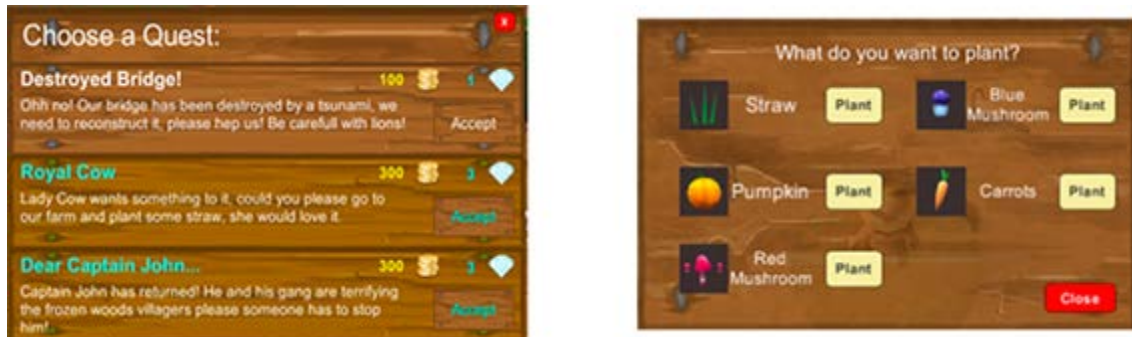


Figura 35: Menú de selecció de qüestions (esquerra) i menú de selecció de plantes per plantar a l'hort (dreta).

Fins aquí s'han explicat les parts de la interfície que el jugador pot controlar, ja que reaccionen a esdeveniments de l'usuari.

Botiga

Una acció molt important que pot realitzar el jugador, és comprar objectes, ja que són la motivació d'aconseguir or i resoldre quests.

Per tal de realitzar això s'ha dissenyat una interfície (veure figura 36) que permet al jugador visualitzar quines seccions de la botiga hi ha (eines, cap i cos), quin és el valor econòmic dels objectes en les dues monedes disponibles en Fracsland i quina bonificació li aporten.

També s'ha dissenyat un codi de colors perquè el jugador pugui veure si té efectiu suficient per comprar un objecte o no ràpidament (vermell vol dir que no pot, verd que sí).

A part s'ha dissenyat una interfície que mostra el que capta una càmera que està situada davant de l'heroi, però que pugui veure com llueix el seu jugador amb els objectes que té equipats i què li aporten (aquesta tot i que s'obre al parlar amb el venedor en la botiga també la pot obrir en qualsevol moment prement la tecla "k" o fent clic sobre la icona que té en la part inferior de la interfície).

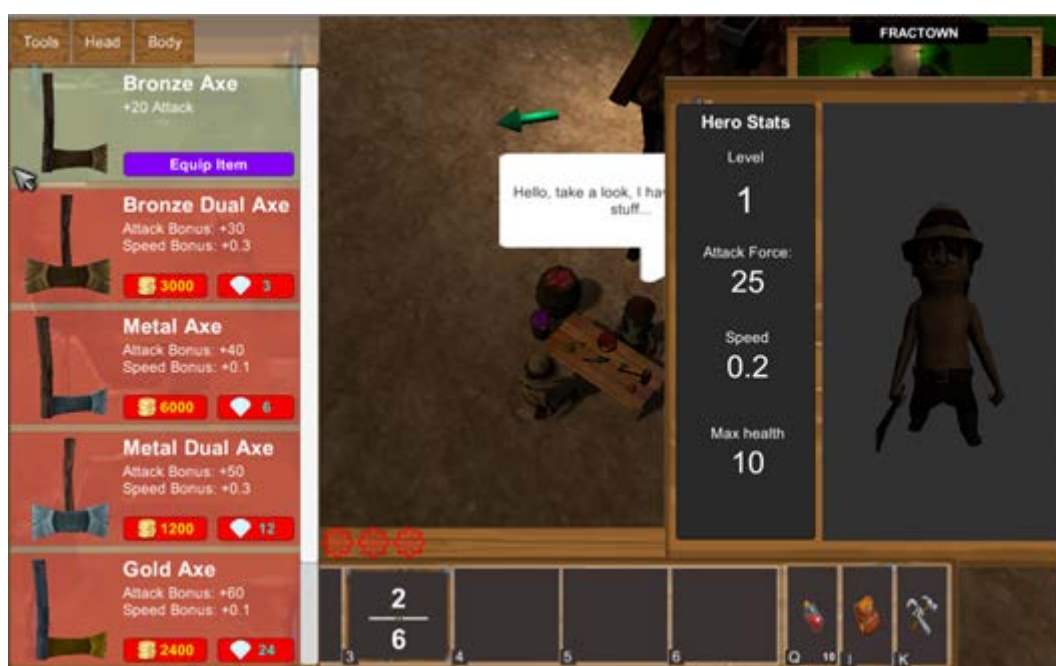


Figura 36: Botiga (esquerra) i estadístiques del jugador (stats en anglès)

Notificacions

Ara tot seguit es passa a mostrar la part de la interfície dedicada a les notificacions.

La primera d'aquestes interfícies el panell que apareix i desapareix suaument quan l'usuari canvia de zona (figura 37), d'aquesta manera pot saber a simple vista on es troba.

Per tal de no sobrecarregar la interfície, s'ha dissenyat un sistema de textos flotants que mostren al jugador dades relacionades amb objectes, diamants, or, vida.



Figura 37: Text que apareix per presentar el nom de la zona on el jugador està accedint

Aquests textos (figura 38) apareixen per sobre de l'usuari o l'entitat en qüestió amb la que estiguin relacionats i desapareixen suaument mentre recorren una petita porció de la pantalla de dalt a baix.



Figura 38: Textos flotants a sobre del jugador.

Si el que aconseguix l'usuari té una rellevància important en el desenvolupament de la seva tasca, com és el cas de l'adquisició d'una nova fracció, el sistema mostra un petit panell en la part superior que desapareix en pocs segons tot realitzant una animació per tal de separar el que són notificacions de les quals és important que el jugador en sigui conscient de les que són menys importants (com les mencionades en els textos flotants) aquest panell es pot veure en la següent figura (figura 39):



Figura 39: Notificació de nova fracció adquirida.

També es disposa del inspector (veure figura 40), que permet a l'usuari saber si pot interactuar amb un objecte o no i quin és el seu nom només posant el cursor a sobre d'aquest. Cal recalcar que s'han dissenyat també uns certs cursors (atac, quest, recol·lectar fusta, parlar) per tal de donar-li a l'usuari una idea de com serà la interacció amb l'objecte:

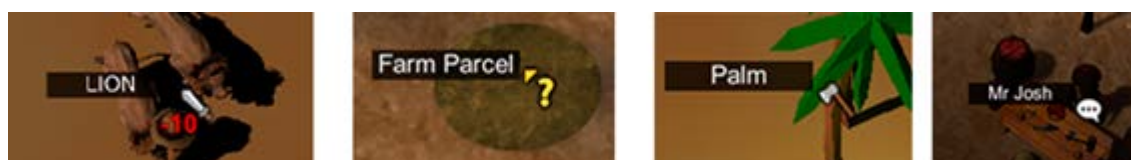


Figura 40: Inspector d'objectes mostrant varies entitats intractables dins del joc.

Així mateix, quan un jugador puja de nivell, el sistema mostrarà un panell indicant-li això juntament amb unes partícules sobre l'heroi perquè pugui ser conscient de que a partir d'aquest moment el seu nivell es superior (veure figura 41). S'ha intentat fer agradable de veure per tal de motivar al jugador perquè vulgui que això passi més cops, incitant-lo així a intentar pujar de nivell.



Figura 41: Feedback que rep l'usuari al pujar de nivell.

3.2.2.3. Feedback

Nivell

Sobre la dinàmica del joc, Fracslancl inclou una part RPG (RoI Playing Game) basada en el concepte de nivell.

Un jugador quan entra a Fracslancl per primer cop és nivell 1 i per tant s'assumeix que el jugador no està familiaritzat amb el joc. Aquest fet provoca que hi haurà menys enemics i no es treballarà amb l'equivalència de fraccions (proves de més dificultat que no pas el concepte de fraccions).

La fórmula per calcular l'experiència que ha de guanyar un jugador per pujar de nivell es calcula de la següent manera. Es comença amb 500 d'experiència de base per pujar al nivell 2, a partir d'aquí l'experiència necessària es calcula seguint la següent progressió:

exp = experiència actual.

expAnt= experiència nivell anterior.

$$exp = Round(1.1 * expAnt)$$

D'aquesta manera cada cop costa un 10% més pujar de nivell.

També cal recalcar que les recompenses que deixen anar els mostres en ser vençuts també s'adapten al nivell del jugador, quan un usuari té un nivell baix, rep una quantitat d'or inferior que quan té un nivell superior.

Quan un enemic deixa anar or, utilitza la següent fórmula:

$$1.1^{nivell} + Max(Random(0, nivell), 3)$$

D'aquesta manera, com més alt sigui el nivell del jugador, més or rebrà quan venci enemics i podrà accedir a un nivell de vida superior dins del joc que li permetrà comprar objectes més cars de la botiga.

S'ha afegit també una component aleatòria a l'equació per tal de generar una certa curiositat i expectativa per part del jugador per comprovar si la quantitat d'or li donarà el següent monstre serà superior, d'aquesta manera s'evita que el joc sigui previsible en aquest sentit. A part d'or, els enemics tenen una probabilitat del 30% de deixar anar pocions per tal que el jugador es pugui curar després d'una lluita.

I per últim amb una probabilitat del 10% un enemic pot deixar anar una fracció que li pot resultar útil per resoldre la quest que estigui realitzant.

Les fraccions es poden aconseguir de 3 maneres:

1-Lluitant i guanyant a enemics (amb una probabilitat del 10%).

2-Recol·lectant vegetals en l'hort de Fracsland amb una probabilitat del 100%. Aquesta alternativa s'ha afegit per tal que un jugador no estigui obligat a lluitar per tal de passar-se el joc.

3-Troba altars, que deixen anar sempre varies fraccions, els valors de les quals es s'escullen aleatòriament internament en el joc.

Tal i com s'ha explicat el nivell també serveix per determinar si el jugador s'haurà d'enfrontar al concepte d'equivalència de fracció.

Cada cop que es dona una fracció al jugador, s'aplica la següent algorisme per tal de decidir si la fracció serà multiplicada per algun factor:

1-Es miren quines possibilitats hi ha en base a quina fracció s'està treballant i s'introdueix dins d'una llista.

2-Es tria una fracció aleatòria de dins la llista.

3-Es calcula un factor per la fracció resultat d'aplicar la següent fórmula:

$$Random(1, Round(nivell/5)) + 1)$$

D'aquesta manera quan més alt és el nivell d'un jugador, més possibilitats hi ha que la fracció que rebí estigui multiplicada per algun factor i el nombre de factors disponibles sigui més alt.

4- Per últim, es multiplica la fracció per el factor calculat anteriorment.

3.2.2.4. Diagrama de classes

El disseny de Fracsland conté aproximadament unes 110 classes, algunes coordinadores i altres implementen funcions específiques com control de personatges, elements de l'interfície d'usuari, emmagatzematge de dades i transmissió de dades cap al servidor.

Dins d'aquestes classes n'hi ha 5 que són les que coordinen els aspectes més importants del joc.

3.2.2.4.1. Principals classes: Controladors

En la figura 42 es poden veure els diferents controladors que gestionen la lògica principal del joc i com es referencien entre ells per tal d'obtenir dades necessàries per la realització del joc o bé per demanar la utilització de funcions d'un altre.

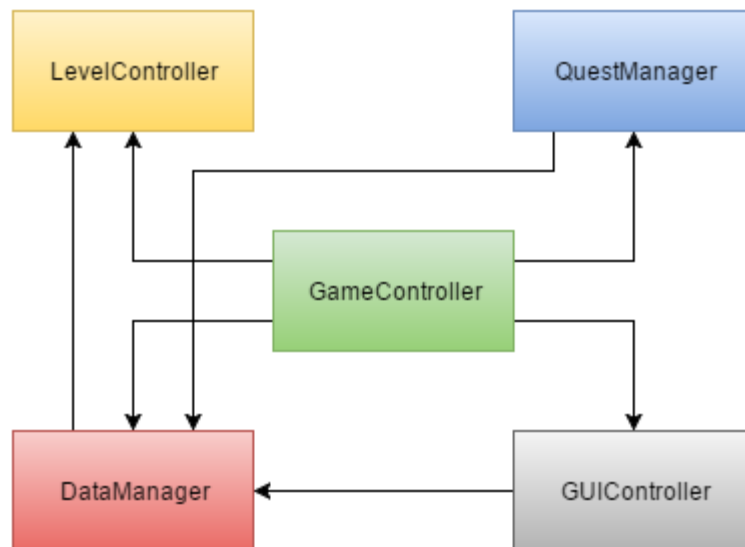


Figura 42: Diagrama controladors principals (les fletxes que volen dir que la entitat a referencia a l'entitat b, a->b)

1. GameController

Aquesta classe sap quins objectes s'han de destruir al sortir de les escenes de joc, avisa als altres controladors quan es carrega una nova escena perquè facin les seves funcions, surt del joc quan el jugador ho demana i s'encarrega de gestionar coses importants com l'economia del jugador, donar-li recompenses, fraccions aleatòries...

Es podria dir que en una espècie de jerarquia de controladors ell és el que estaria en la punta de la piràmide.

2. LevelController

Aquesta classe assegura que quan una escena es carregada, tot el seu contingut es posicioni on toca i contingui els elements necessaris per un correcte desenvolupament del joc.

Algunes de les seves tasques són:

1- Posicionar l'heroi on ha d'estar, tant quan entra en escena com quan es ressuscitat després de morir.

2- La seva tasca més important, crear i posicionar elements pseudo-aleatòriament (seguint l'algorisme descrit en la secció 2 de l'apartat mecàniques) per a que les escenes on es desenvolupen les quests no tinguin sempre la mateixa distribució, d'aquesta manera sempre és interessant que el jugador explori el joc, ja que no pot saber mai si es trobarà alguna cosa nova en algun lloc.

3. QuestManager

Aquesta classe és la més important per gestionar la lògica que seguirà el joc per poder portar a terme les quests. Les seves funcions més importants son:

- 1- Saber quina és la missió actual i on es portarà a terme.
- 2- Crear events que seran activats quan el jugador realitzi alguna acció com: Tallar una palmera, obtenir fraccions, plantar, alimentar animals...
- 3- Un cop activats aquests events és la que sap a quines entitats del joc s'ha d'avisar de que un event s'ha portat a terme.
- 4- Guiar al jugador durant una quest.
- 5- Saber si el jugador està a l'escena que li toca per fer una quest.
- 6- Assignar una recompensa a les quests.
- 7- Mostrar els detalls d'una quest.
- 8- Configurar una quest quan el jugador entra en el joc segons els paràmetres que ha configurat el professor.
- 9- Donar la recompensa al jugador quan aquest acaba una quest.
- 10- Canviar l'estat actual d'una missió quan sigui necessari, actualment n'hi ha 3 (DOING_QUESTION, PENDING_TO_REWARD, UNASSIGNED).
- 11- Acceptar una quest.

4. GUIController

Aquesta classe és la que controla la interfície d'usuari que hi ha durant el transcurs del joc, per tal de poder informar al jugador de l'estat del sistema. Les seves tasques més importants son:

- 1-Canviar el cursor del joc segons on esta posicionat (sobre un enemic, una palmera, un NPC...)
- 2- Mostrar la vida del jugador, nivell, experiència...
- 3- Mostrar i amagar menús, ja sigui des dels botons de la GUI o des de events del joc.
- 4- Mostrar elements inspeccionats.
- 5- Mostrar barres de vida sobre els enemics.
- 6- Crear els textos flotants que mostren, vida perduda o guanyada, or, objectes...
- 7- Mostrar objectius de les quests en pantalla.
- 8- Mostrar el nom d'una zona quan s'entra i en el minimapa.
- 9- Mostrar quants diners i diamants té el jugador.

5. DataManager

Aquesta classe és la que s'inicia en entrar en el joc (obté i guarda les dades que obté el jugador del servidor al entrar) i manté tota la informació que s'ha de sincronitzar mitjançant els scripts destinats a la comunicació per la xarxa, com els objectes que té el jugador equipats, els diamants...

3.2.2.4.2. Diagrames de les principals classes

Com mostrar el diagrama sencer del joc, no és possible donat que implica moltes classe, tot seguit es veuran una sèrie de diagrames simplificats relacionats amb cada controlador.

1. GameController:

El GameController al igual que els altres controladors, manté una referència estàtica a ell mateix, d'aquesta manera els altres components del joc poden referenciar-lo a ell d'un mode còmode.

En el següent diagrama (figura 43) es poden veure les referències que fa el GameController i quines referències es fan cap al game controller.

Les classes blaves treballen juntament amb el GameController per realitzar tasques de la lògica global del joc.

Les classes grogues referencien el GameController per realitzar peticions i tasques de caràcter econòmic.

I les verdes per altres motius, com preguntar si queden missions disponibles per fer o saber amb quina entitat esta interactuant el jugador ara mateix.

Una fletxa que surt d'una entitat i va cap a una altre vol dir que la entitat d'on surt, referencia a l'entitat que apunta.

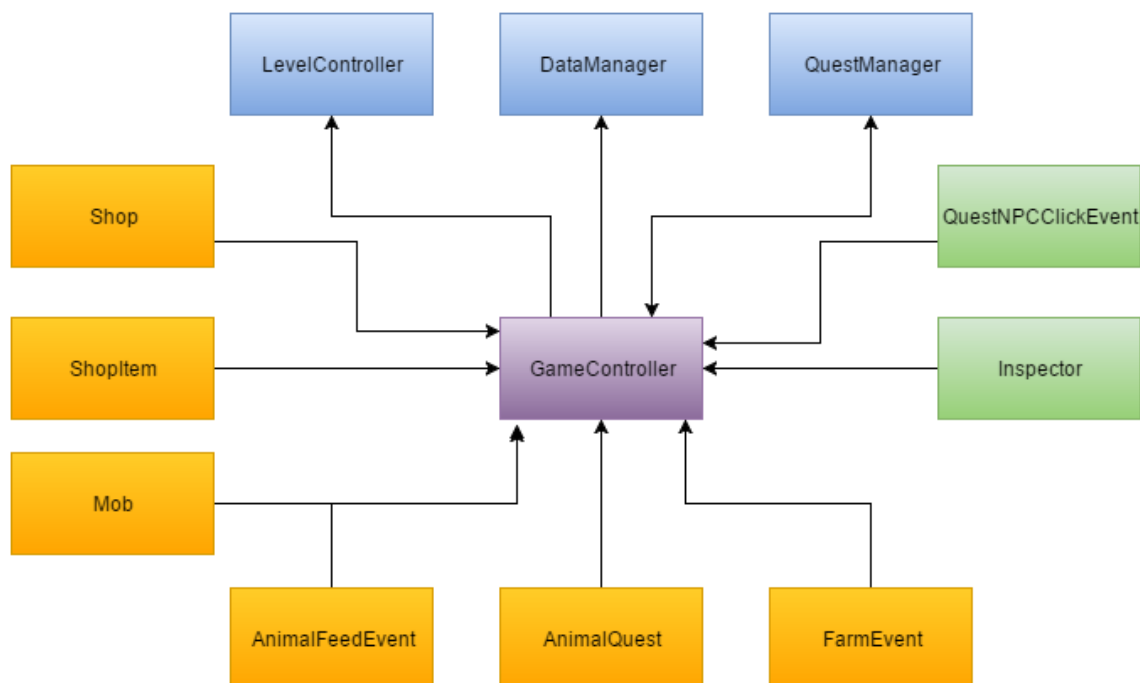


Figura 43: Diagrama referències GameController

2. LevelController:

En el següent diagrama (figura 44) les classes blaves, referencien al LevelController per temes de lògica del joc, per exemple DataManager després de fer una primera sincronització, li demana al level controller que prepari el primer nivell, un cop fet això aquesta responsabilitat se l'atribueix el GameController cada cop que es canvia de escena.

El LevelController guarda una llista de SpawnPoints (classe que decideix com entitats d'un nivell aleatòriament) i també guarda una referencia a l'heroi, ja que ell és l'encarregat de crear-lo i posicionar-lo en una escena. Per aquesta ultima raó, totes les classes que estan en groc referencien al LevelController, ja que necessiten referenciar a l'heroi, ja sigui per moure'l, mostrar quelcom a sobre d'ell...

Per últim es té una classe en color verd que referencia a LevelController per saber en quina escena es troba.

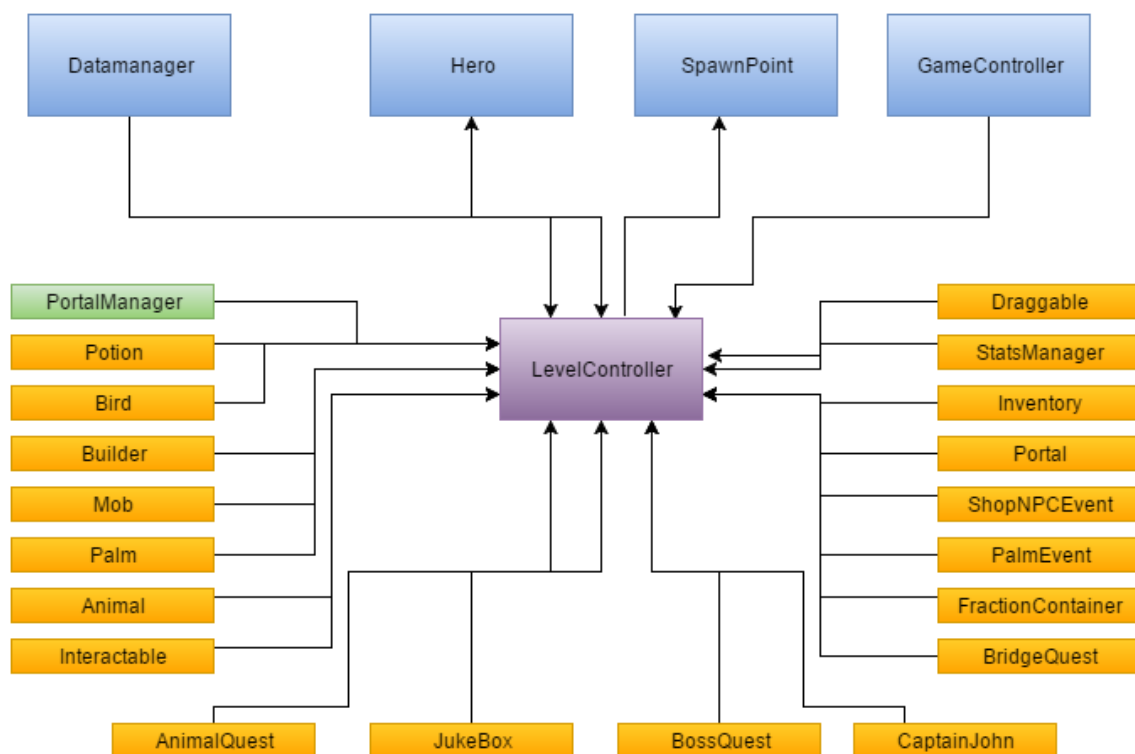


Figura 44: Diagrama referencies LevelController.

3. GUIController

Aquesta entitat s'encarrega de mostrar al jugador informació sobre l'estat del sistema, de forma que cada vegada que una entitat necessita mostrar informació la utilitza.

Els elements de color blau (visibles en el diagrama de la figura 45) , són aquells que són prefabs, útils per mostrar elements en la interfície que són una mica més elaborats, ja que es poden clonar, modificar i mostrar.

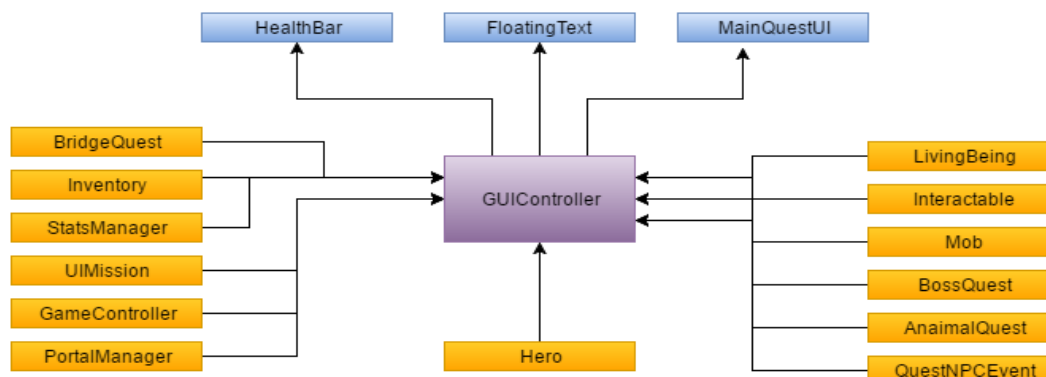


Figura 45: Diagrama GUIController

4. DataManager

El DataManager conté la informació important del joc que ha de ser sincronitzada amb el servidor. Sempre que una entitat referencia al DataManager és per llegir informació o bé per sincronitzar una nova informació amb el servidor per mitjà de l'API, per veure aquestes referències veure figura 46).

DataManager conté una entitat anomenada PlayerWarehouse que conté tots els objectes i diners que s'han de sincronitzar amb el servidor, que pertanyen al jugador i una llista de les quests o missions que el professor a configurat per l'alumne.

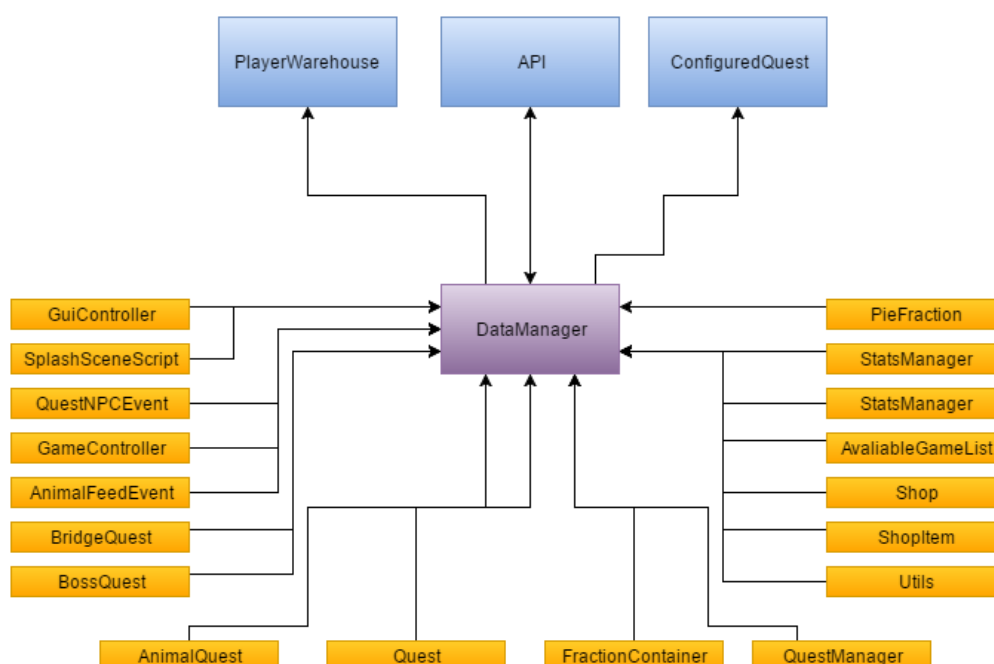


Figura 46: Diagrama relacions Datamanager

Capítol 4. Disseny del servidor

4.1. Disseny del panell d'Administració Fracsland

En aquest capítol s'explicaran les parts referents a Servidor Web, Base de dades, els panells d'administració i la interfície de comunicacions amb el joc, deixant per el proper capítol l'arquitectura i el desenvolupament del joc pròpiament dit.

En aquest capítol s'explicaran les parts referents a Servidor Web, Base de dades, els panells d'administració i la interfície de comunicacions amb el joc. En primer lloc es detallen les components dels panells d'administració del professor i de la plataforma. Després es detalla el disseny de la interfície de comunicació amb el Joc. Finalment es defineix el disseny de la Base de Dades que permet establir i mantenir la persistència de les dades del cadascun dels jugadors en els jocs. En la Figura 47 es mostren requadrat en blau les components de l'arquitectura que s'explicaran en aquest capítol.

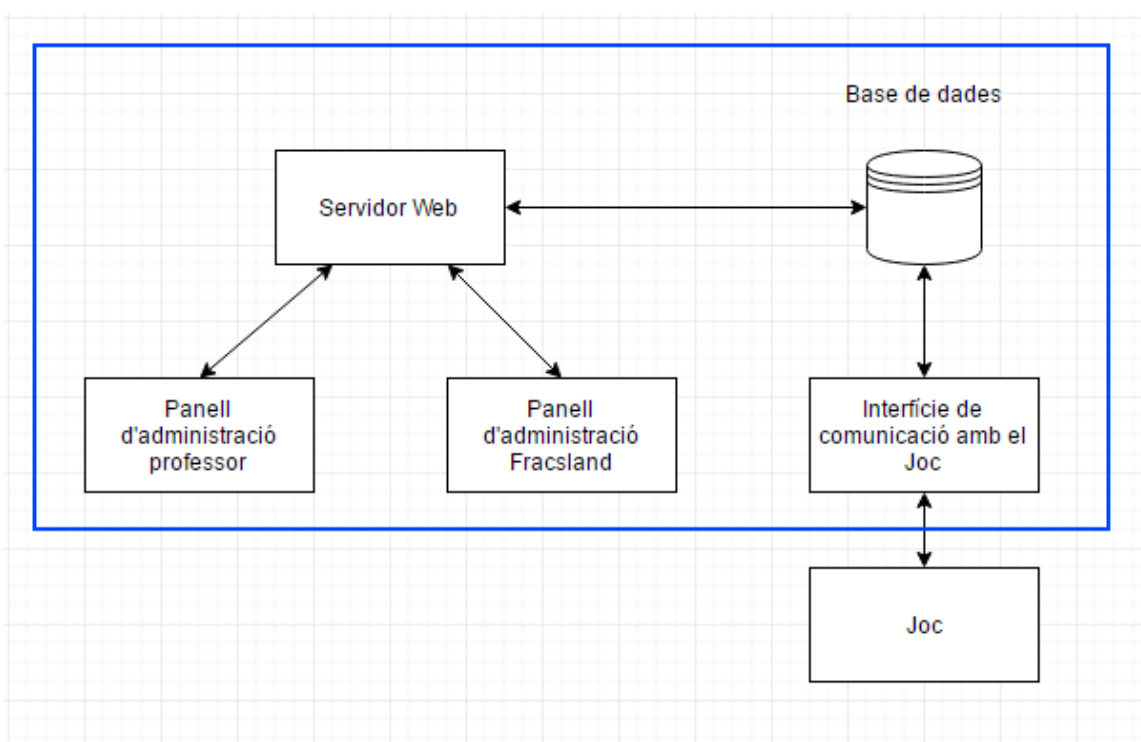


Figura 47: Arquitectura del sistema amb les parts referents al servidor web, base de dades i webservice (interfície de comunicació amb el joc) ressaltades.

4.2. Disseny del panell d'Administració Fracsland

El panell d'administració de Fracsland està dividit en diferents apartats visibles en el següent diagrama (figura 48):

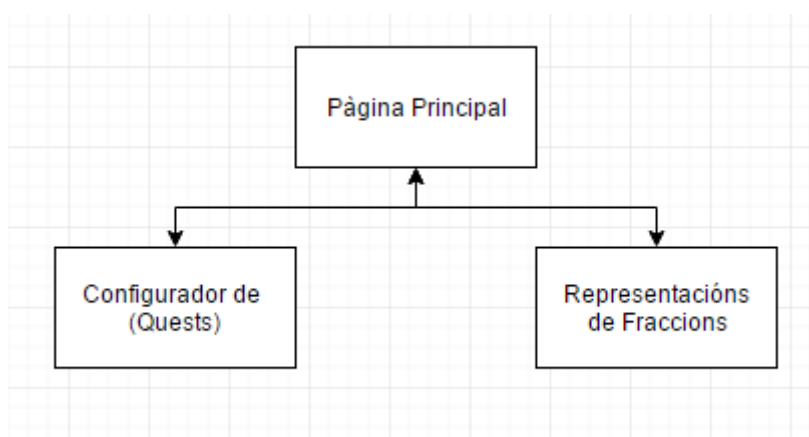


Figura 48: Diagrama de d'apartats visibles en el panell d'administració de Fracsland.

4.2.1. Disseny del configurador de Quests (rol administrador)

Aquest apartat permet al Administrador decidir quins atributs seran personalitzables dins del joc:

Quest configurator

Actual Configuration			
Int			
Name	Is required	In Game ID	Help text
Working Fraction	<input type="checkbox"/>	working_fraction	The fraction that the student
Fraction Representation			
Name	Is required	In Game ID	Help text
Preferred Fraction Representation	<input type="checkbox"/>	frac_rep	Gives an extra reward to the
Fraction			
Name	Is required	In Game ID	Help text
Solution (Optional)	<input type="checkbox"/>	solution	This is an optional field, if

Widgets

- Fraction
- Bool
- Int
- Fraction Representation

Save Configuration

Figura 49: Quest configurator del panel administrador.

Això, tal i com es pot veure en la Figura 49, s'ha dissenyat una interfície d'usuari fàcil d'usar i el més intuïtiva possible, que permet arrossegar el tipus de camp que es vol afegir des de l'apartat "widgets" fins a l'apartat "Actual Configuration" i un cop allà assignar el nom de la configuració, si és requerit o no, la seva id dins del joc i un petit text d'ajuda per indicar al professor de què es tracta.

També es poden esborrar les configuracions i ordenar-les per tal de triar l'ordre en el què vol que es mostrin en el panell de configuració del professor.

En les següents imatges (figures 50 i 51) es pot veure un exemple de configuració creada en el Panell d'Administració de Fracsland i com la percebrà el professor des del seu panell (aquesta segona part es detalla en l'apartat "Panell d'Administració Professor").

Actual Configuration			
Int			
Name	Is required	In Game ID	Help text
Working Fraction	<input checked="" type="checkbox"/>	working_fraction	The fraction that the student
Fraction Representation			
Name	Is required	In Game ID	Help text
Preferred Fraction Representation	<input checked="" type="checkbox"/>	frac_rep	Gives an extra reward to th
Fraction			
Name	Is required	In Game ID	Help text
Solution (Optional)	<input type="checkbox"/>	solution	This is an optional field, if

Figura 50: Configuracions seleccionades per l'administrador de Fracsland.

Quest Configuration	
Working Fraction: ?	<input type="text" value="2"/>
Preferred Fraction Representation: ?	<input type="text" value="Pie"/>
Solution (Optional): ?	<input type="text" value="Numerator"/>
	<input type="text" value="2"/>

Figura 51: Representació de les configuracions de la figura 50 en el panell del professor.

Representació de Fraccions:

En aquest menú l'administrador del joc pot anar afegint noves representacions de les fraccions a mesura que es vagin implementant, actualment el joc suporta la representació en "Rectangle" i la presentació de tipus "Pie".

Gràcies a això, en el cas que s'hagi afegit un camp del tipus "Fraction Representation" dins del apartat "Configurador de Quests", el professor podrà veure totes les representacions de fraccions disponibles que s'hagin enregistrat en la base de dades usant aquest apartat.

Pàgina principal:

Permet a l'administrador del joc veure quants usuaris hi ha registrats, quantes classes s'han fet i quants jocs s'han configurat, sense necessitat d'accedir a la base de dades.

4.3. Disseny del panell d'Administració del professor

El panell d'administrador del professor es divideix en varies pàgines per tal de poder realitzar el que s'espera del sistema d'un mode el més ordenat i còmode per al professor.

En la Figura 52 es mostra esquema on es poden veure les pàgines més importants del sistema.

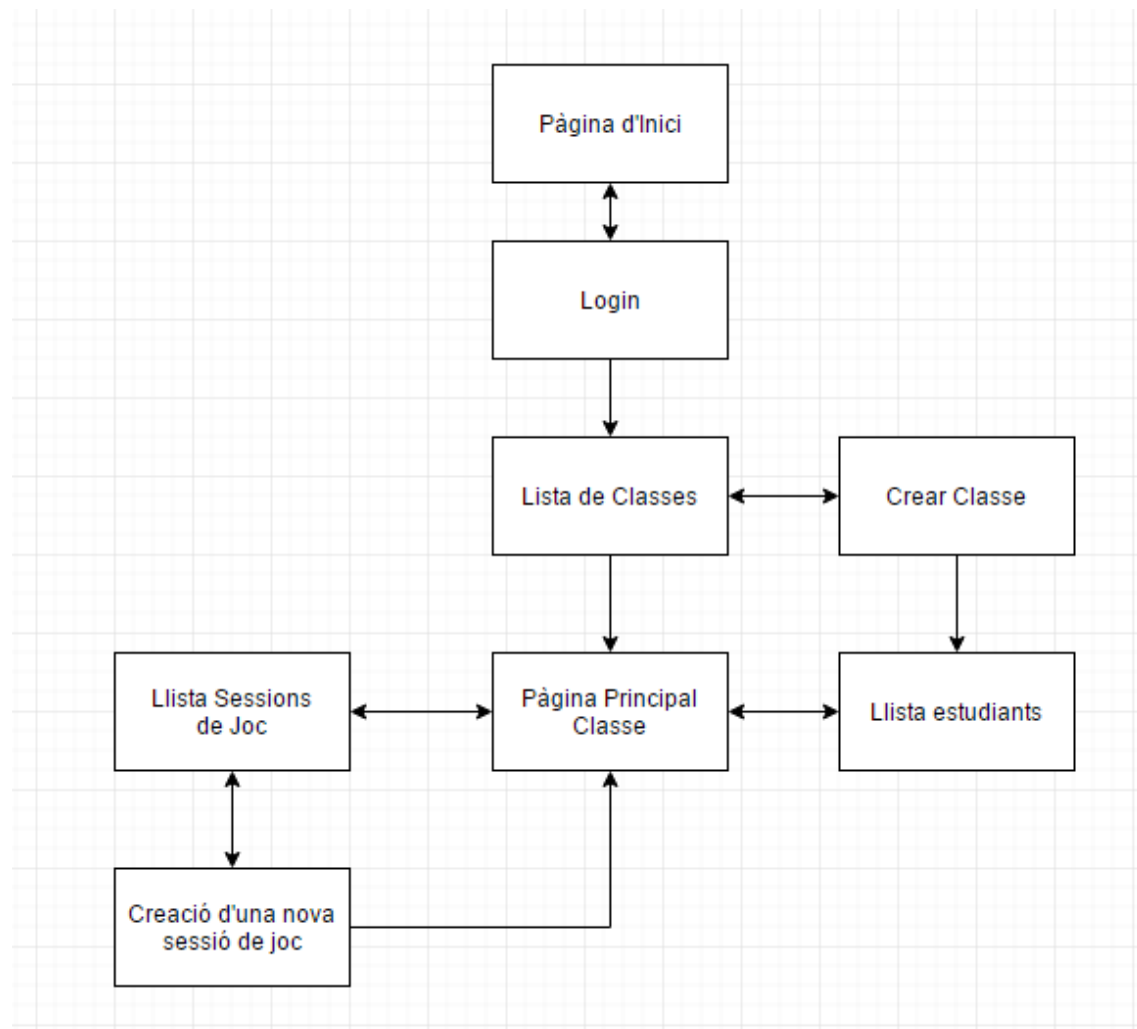


Figura 52: Diagrama de les pàgines accessibles en el panel d'administració del professor.

Les fletxes dins del diagrama indiquen, quin és el recorregut que pot fer un usuari entre aquestes pàgines, tant mitjançant enllaços (són les bidireccionals) com després de realitzar alguna petició.

A continuació es detallen cadascun dels blocs descrits a la Figura 52:

1. Pagina d'Inici

Aquesta pàgina conté varies seccions, una a sota de l'altra, que poden ser accedides ràpidament usant els enllaços de la part superior (veure figura 53):



Figura 53: Enllaços disponibles en la pàgina d'inici.

La primera secció es veu només entrar és un formulari de registre, i permet als professors enregistrar-se, introduint el seu correu electrònic i una contrasenya.

Un cop enregistrat, el professor accedirà al tutorial (detallat en l'apartat 3.2.2) que li explicarà com crear la seva primera classe i configurar un nou joc.

La següent secció, explica que és Fracslan, per tal d'informar als visitants de la pàgina, de què es tracta aquest projecte.

I per últim tenim l'apartat de descàrregues el qual conté enllaços per poder obtenir les diferents versions del Joc (Windows, Mac i Linux).

A part, en la barra superior també es pot trobar un enllaç per anar a la pàgina d'autenticació.

2. Login:

Aquesta pàgina utilitza el mecanisme d'autenticació de Django per tal d'iniciar la sessió d'un usuari, si les credencials d'un usuari determinat són correctes podrà accedir al seu panell d'administració pertinent, un professor accedirà a la seva llista de classes i l'administrador de Fracslan anirà al panell d'administració global de la plataforma.

3. Llista de Classes:

En aquesta pàgina un professor, pot veure totes les classes que ha creat, per tal d'accedir a elles o crear una nova classe si ho considera oportú.

4. Crear Classe:

En el moment de la creació d'una classe, el professor accedirà al tutorial, que va fer al principi obviant la part de la pàgina de benvinguda, ja que ell ja és membre de Fracslan.

Quant el docent, faci click en l'opció "Afegir classe" veurà una pàgina amb un formulari que li demana que bategi la nova classe amb un nom.

Un cop la classe hagi estat batejada el sistema crearà una nova entrada en la taula "Classroom" de la base de dades.

Llavors accedirà a la pàgina que li permetrà afegir estudiants a la seva classe i seguirà el tutorial per tal de poder configurar un nou joc per aquesta (tot aquest procés serà explicat més endavant en l'apartat "Tutorial Professor").

La decisió d'utilitzar la plataforma en mode tutorial quan es vol crear una nova classe està motivada per el fet de d'agilitzar el procés, ja que si no es fes així un usuari hauria d'accedir

explícitament als apartats pertinents de la web per tal de tenir tots els ingredients necessaris per poder configurar una sessió. D'aquesta manera només ha de seguir els passos indicats i en pocs minuts pot disposar de tot funcionant correctament.

5. Llista d'Estudiants:

Aquesta pàgina ha estat dissenyada perquè un docent pugui observar fàcilment quins estudiants pertanyen a la classe en qüestió, en el cas de que hi hagi alguna dada incorrecte o precisi ser canviada, el docent pot fer click en editar i canviar les dades que necessiti.

Tot aquest procés es fa amb AJAX, per tal de que sigui directe i entenedor (ja que no cal recarregar la pàgina de nou per visualitzar els canvis).

En les següents imatges es pot veure un exemple d'aquest procés.

En la imatge de la Figura 54, es pot veure una llista amb dos estudiants, i resulta que durant la introducció de les dades, ja sigui via Excel o manualment. En l'exemple el docent ha comès un error, resulta que ha escrit en el camp email nom@centre.ed (s'ha descuidat una "u"):

Student List

Student list					
Search...					Q
First Name	Surname	Email	Level	Add Student	
Nom1	Cognom1	nom1@centre.ed	1	Edit	Delete
Nom2	Cognom2	nom2@centre.edu	1	Edit	Delete

Figura 54: Llista estudiants de una classe amb 2 estudiants registrats.

Per realitzar aquest canvi només ha de fer click en "Edit" i la fila referent a l'estudiant es posarà en mode edició (veure figura 55):

Student List

Student list					
Search...					Q
First Name	Surname	Email	Level	Add Student	
Nom1	Cognom1	nom1@centre.ed	1	Change Password	Save Delete
Nom2	Cognom2	nom2@centre.edu	1	Edit	Delete

Figura 55: Llista d'estudiants amb el primer estudiant en mode edició.

Com es pot veure en la imatge de la Figura 55 el docent pot canviar les dades de l'alumne i la seva contrasenya. Un cop hagi acabat d'editar les dades pot fer click en "Save" i es farà una petició AJAX al servidor per tal de canviar les dades.

El format en que s'envien les noves dades al servidor es JSON.

En el cas de que el professor hagués volgut esborrar l'estudiant, només hagués tingut que fer click en "delete" i confirmar que és realment aquesta l'acció que es vol portar a terme.

Una altre funcionalitat de la llista d'estudiants, és la possibilitat d'afegir estudiants. En aquest cas quan un professor faci click en "Add Student" apareixerà el següent diàleg (figura 56):

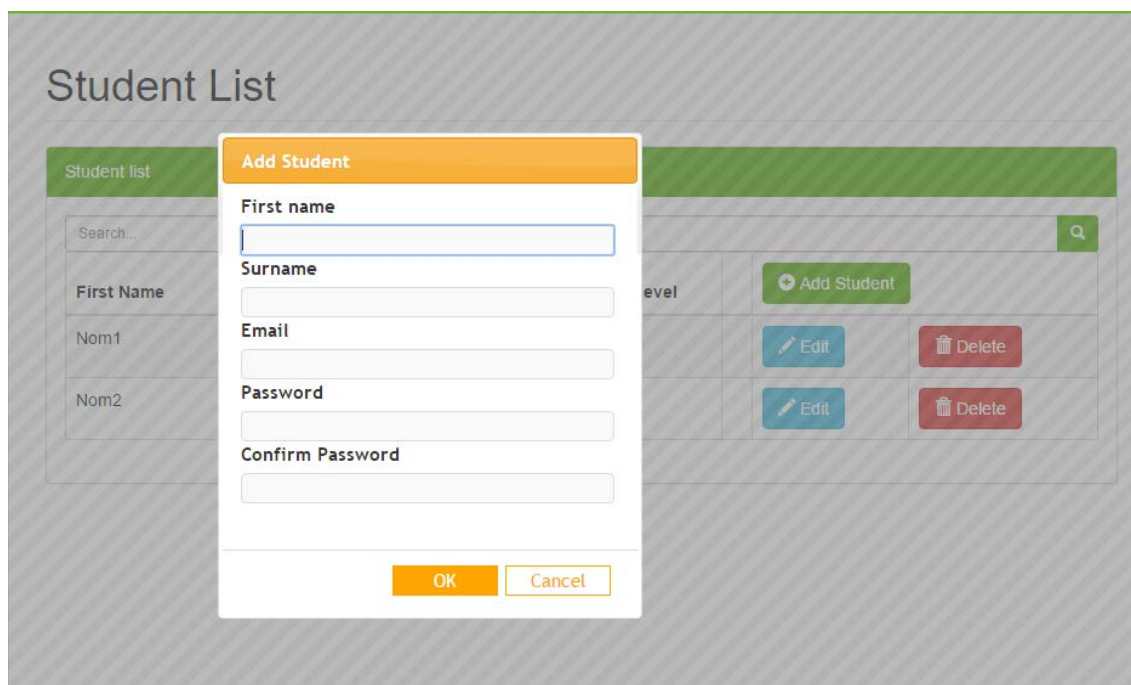
The image shows a web application interface titled "Student List". In the foreground, there is a modal dialog box titled "Add Student" with an orange header. It contains five text input fields: "First name", "Surname", "Email", "Password", and "Confirm Password". At the bottom of the dialog are two buttons: "OK" (orange) and "Cancel" (white with orange border). In the background, a table of students is visible. The table has columns for "First Name", "Surname", "Email", "Level", and actions. The "Add Student" button is highlighted in green. The "Edit" and "Delete" buttons are blue and red respectively.

Figura 56: Diàleg registre nou estudiant.

Aquest diàleg conté un formulari, que serà enviat per POST al servidor per tal de crear les dades associades a la creació d'un nou estudiant.

6. Pàgina Principal de la Classe:

En aquesta pàgina es mostren quants estudiants hi ha registrats a la classe, quantes sessions de joc hi ha acabades i quantes d'aquestes encara no s'han acabat.

D'aquesta manera el docent pot observar quin és l'estat de la classe sense necessitat d'accedir als apartats corresponents.

7. Llista de sessions de Joc i Detalls de Sessió:

La llista de sessions de joc, permet veure quines sessions té configurades una classe en concret, permetent accedir als detalls d'aquestes (Detalls de Sessió), per tal de monitoritzar d'un mode transparent a l'alumne com està progressant en els reptes que li han estat assignats.

També conté l'opció de crear una nova sessió de joc en la part superior de la llista.

També té al costat de cada sessió un botó que li permet tancar o obrir una sessió de joc. Això resulta útil per tal de que els alumnes que no varen acabar els reptes en una sessió anterior ja no rebin més aquella sessió com activa.

En la imatge de la Figura 57 es pot veure una sessió en la llista identificada per la seva data de creació juntament amb el seu botó de tancar.

Creation Date	+ Add Class Session
June 12, 2016, 3:33 p.m.	Close

Figura 57: Llista de sessions configurades amb una sessió disponible.

Quan un professor accedeix en una sessió de la llista, fent click en la data d'aquesta, accedeix a la pàgina Detalls de sessió (veure figura 58).

Session Details

1
Games Finished

2
Games to finish

Students Progress Update Data						
First Name	Surname	Quests Done	Errors	Success Ratio	Has finished	Quests todo
Nom1	Cognom1	0	0	1	False	3
Nom2	Cognom2	1	0	1	True	0
Nom3	Cognom3	1	2	0.3333333333	False	2

Figura 58: Detalls d'una sessió amb 3 participants.

En la part superior d'aquesta pàgina es poden veure quants jocs han estat finalitzats i quants jocs falten per finalitzar, així el professor pot veure ràpidament quin es l'estat general de la classe.

En la part superior dreta de la llista de progressos dels estudiants, el professor disposa d'un botó que li permet actualitzar les dades de la classe, per tal de que el sistema li entregui les noves dades sobre els progressos del alumnes, tot fent una petició GET al servidor.

L'últim element d'aquesta pàgina és la llista de progressos dels estudiants, en la qual el docent pot veure, d'esquerra a dreta, el nom de l'estudiant, el seu cognom, quantes tasques a acabat, quants errors a comés, quin es el seu percentatge d'èxit, si ha acabat o no i quantes missions li queden per fer.

Una funcionalitat molt important d'aquesta pàgina és utilitza un codi de colors per els estudiants. Quan un estudiant està cometent molts errors, la seva fila tendeix a posar-se de color vermell.

D'aquesta manera el professor, pot identificar ràpidament quins son els alumnes que necessiten ajuda amb les tasques, sense necessitat de llegir tota la taula.

Un cop un estudiant acaba totes les seves tasques la seva fila es posa de color blau.

8. Creació d'una nova sessió de joc:

Aquesta pàgina és la que dona el poder de configuració per part del professor a la plataforma Fracsland, ja que és la que determinarà quines seran les tasques que haurà de fer cada alumne durant una sessió de joc.

Cal recordar, tal i com s'ha explicat en l'apartat de l'Administrador de Fracsland, que els paràmetres que es poden configurar del joc, son dinàmics, el que vol dir que és creen en base als paràmetres que ha afegit l'administrador en la pàgina de Game Configuration (veure apartat 3.3.1).

Els paràmetres configurables de Fracsland actualment són els següents:

- 1- Working Fraction: Aquest paràmetre és de tipus sencer i especifica quina fracció treballarà l'alumne durant la realització de la tasca.
- 2- Preferred Fraction Representation: Poden ser de dos tipus Rectangle i Pie.
En les següent figura (figura 59) es poden veure dos proves dins del joc configurades treballant fraccions del 8, a l'esquerra un exemple de representació de fracció "Rectangle" i a la dreta "Pie".



Figura 59: Representació de fracció del tipus rectangle en el joc (Esquerra) i representació del tipus "Pie" (dreta).

Aquesta configuració fa que l'alumne rebi una recompensa més alta si tria una missió on la representació de la fracció que s'està treballant coincideix amb la que ha indicat el professor. D'aquesta manera no s'obliga mai a l'estudiant a realitzar una quest en concret però si que se'l premia amb més diamants i or si la fa, fet que li permetrà comprar-se més objectes.

En la imatge del joc de la Figura 60 es pot veure com hi ha dues quests que aporten més or i diamants a l'alumne, ja que, en aquest cas, el professor ha triat la representació de la fracció "Pie", representació que treballen aquestes quests.



Figura 60: Diferents quests que disponibles l'estudiant dins del joc (en blau les que tenen un tipus de representació que coincideix amb la preferència configurada per el professor).

- 3- Solution: Aquest paràmetre és opcional i permet al professor especificar quina serà la solució exacte que l'alumne haurà de donar per resoldre la quest. En cas de deixar aquest paràmetre buit, el servidor la generarà de forma aleatòria.

El procés que es porta a terme en aquesta pàgina es realitza en dos passos:

- 1- Configuració Global.
- 2- Configuració individual.

Aquesta manera de procedir és beneficiosa per el professor, ja que pot crear una configuració que es copiarà per cadascun dels seus estudiants i arribats a aquest punt només haurà de modificar els paràmetres que consideri oportuns per cada estudiant en qüestió, en comptes d'anar creant les configuracions una a una.

Configuració Global:

Game Configuration



First of all let's create a global configuration for all your students (you will be able to modify it for each student in the next step).

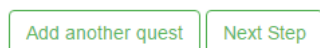
The image shows three 'Quest Configuration' cards. Each card has a green header and contains the following fields: 'Working Fraction' (with a question mark icon), 'Preferred Fraction Representation' (with a question mark icon and a dropdown menu), and 'Solution (Optional)' (with a question mark icon and a text input field). The first card has values: Working Fraction: 2, Preferred Fraction Representation: Pie, Solution (Optional): Numerator 2. The second card has values: Working Fraction: 3, Preferred Fraction Representation: Rectangle, Solution (Optional): Numerator 3. The third card has values: Working Fraction: 5, Preferred Fraction Representation: Pie, Solution (Optional): Numerator 5.

Figura 61: Exemple de configuració global per una sessió de joc.

En la imatge de la Figura 61 es pot apreciar que en la part superior hi ha 2 botons: (1) el primer permet afegir una nova tasca, que es materialitza com una de les capsas que es poden veure en la part inferior, i (2) el segon botó permet anar a l'apartat de Configuració individual.

Si es passa el punter per sobre d'una de les configuracions apareixerà el següent:

En la imatge de l'esquerra (figura 62), el punter està situat sobre l'icona que permet esborrar una configuració, i en la de la dreta el punter està situat sobre l'opció de "copiar una configuració a la dreta".

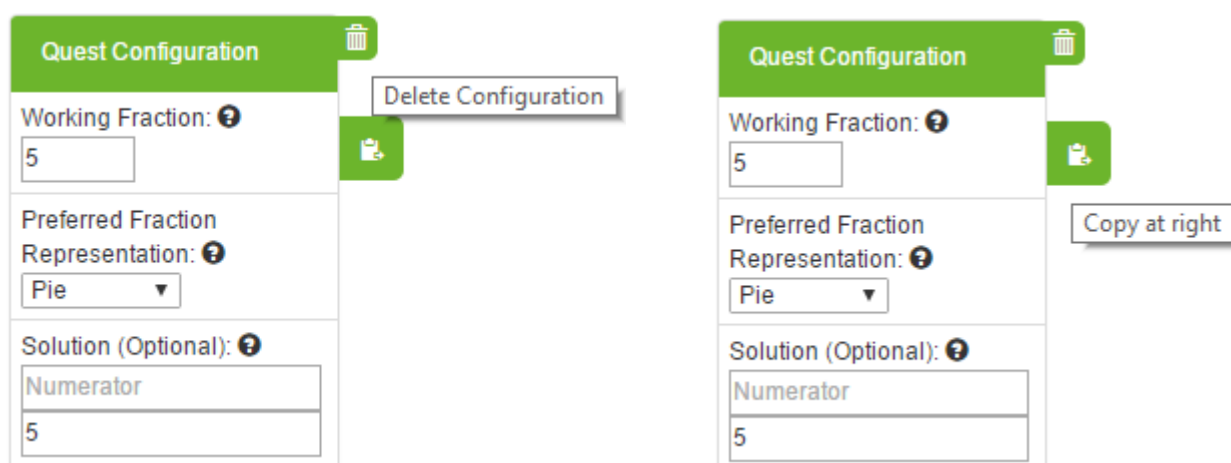


Figura 62: Punter sobre borrar configuració (esquerra) i punter sobre copiar a la dreta (imatge dreta).

En cas de fer click sobre borrar una configuració de tasca, aquesta desapareix tot realitzant una animació (figura 63):

Figura 63: Animació de destrucció d'una configuració.

En cas de fer click en “copiar una configuració a la dreta” apareixerà una nova capsa amb les dades de la configuració a copiar a la dreta del element desitjat. Aquesta opció és útil si ja s'ha configurat tot el circuit de tasques i es voldria replicar una tasca justament després d'una altre.

Configuració individual:

Un cop acabada la configuració global, el professor podrà accedir a aquest apartat, on se li mostrarà l'opció d'acabar la configuració del joc o bé modificar una a una les configuracions individuals dels estudiants creades a partir de la configuració global, tal i com es pot veure en la imatge de la Figura 64:

Game Configuration

First Name	Name	
Nom2	Cognom2	Configuration
Nom3	Cognom3	Configuration

Figura 64: Llista d'estudiants que tenen associats una configuració per una sessió de joc.

En cas de que el professor volgués canviar una configuració per algun estudiant en concret, només ha de fer click en el botó “Configuració” de la fila de l'alumne per el qual desitgi modificar la configuració i modificar-la seguint el mateix procediment que va seguir durant la configuració global.

Un exemple d'això es pot veure en la imatge de la Figura 65, on s'ha fet click sobre el botó configuració del primer alumne:

First Name	Name	
Nom1	Congnom1	Configuration

Quest Configuration

Working Fraction: ⓘ
2

Preferred Fraction Representation: ⓘ
Pie

Solution (Optional): ⓘ
Numerator
2

Quest Configuration

Working Fraction: ⓘ
3

Preferred Fraction Representation: ⓘ
Rectangle

Solution (Optional): ⓘ
Numerator
3

Quest Configuration

Working Fraction: ⓘ
5

Preferred Fraction Representation: ⓘ
Pie

Solution (Optional): ⓘ
Numerator
5

Figura 65: Configuració d'un estudiant en concret desplegada.

Aquesta darrera funcionalitat ens permet crear sessions de joc a mida per cada un dels estudiants de la classe.

Per enviar les dades d'aquest apartat al servidor i poder emmagatzemar les configuracions en la base de dades, s'ha optat per crear un JSON a partir de les dades que figurin en el la pàgina un cop el professor hagi fet click en "Acabar Configuració".

Aquest JSON es genera utilitzant JQuery per obtenir totes les "capses de configuració" de cadascuna de les files dels estudiants per tal de crear un JSON amb la següent estructura (veure figura 66):

```

1 {
2   "data": [{
3     "student": "<id_estudiant>",
4     "configuration": [{
5       "quest_conf": [{
6         "field_id": " < idcamp > ",
7         "field_values": [{
8           "value": " < valor_camp > "
9         }]
10      }]
11    }]
12  }]
13 }
```

Figura 66: Estructura del JSON resultant de la configuració de qüests.

Com es pot veure en la figura 66, data conté una llista amb objectes, on cada objecte representa la configuració de joc d'un estudiant en concret.

El primer que conté la configuració es un camp "student", el qual té com a valor la id de l'estudiant en qüestió.

Es disposa d'un altre camp anomenat "configuration", que conté una llista de tasques que s'han configurat per l'alumne.

Cada “quest_conf” conté una llista amb els objectes que es corresponen als camps configurables creats dinàmicament des del panell del administrador de Fracsland.

El JSON que es pot veure en l'imatge de la Figura *** és una simplificació del què s'envia al servidor, ja que les llistes d'objectes de la figura només en contenen un, quan en la realitat se'n contenen varis.

En la següent imatge (figura 67) es pot veure com es veuria el contingut de la configuració d'un estudiant, extret d'una petició real:

```
{
  "student": "13",
  "configuration": [
    //Configuració estudiant concret
    {
      "quest_conf": [
        {
          "field_id": "1", "field_values": [{"value": "2"}], //Working Fraction
        },
        {
          "field_id": "2", "field_values": [{"value": "0"}], //Prefered Fraction Type
        },
        {
          "field_id": "3", "field_values": [{"value": ""}, {"value": "2"}]}, //Solution (optional)
        }
      ],
      "quest_conf": [
        {
          "field_id": "1", "field_values": [{"value": "4"}],
        },
        {
          "field_id": "2", "field_values": [{"value": "0"}],
        },
        {
          "field_id": "3", "field_values": [{"value": "2"}, {"value": "4"}]},
        }
      ],
      "quest_conf": [
        {
          "field_id": "1", "field_values": [{"value": "6"}],
        },
        {
          "field_id": "2", "field_values": [{"value": "0"}],
        },
        {
          "field_id": "3", "field_values": [{"value": ""}, {"value": "6"}]},
        }
      ]
    }
  ],
}
```

Figura 67: Exemple de configuració de quest per un estudiant obtinguda del servidor.

Es pot observar que els camps “field_id” estan identificats per ID i no per noms, fet que seria més llegible per l'ull humà, això produeix perquè aquest sistema de generació de JSON és dinàmic. Per tant, si l'administrador de Fracsland modifica els paràmetres configurables del joc no s'haurà de re-programar, senzillament apareixan o desapareixan “field_id”.

Quan el servidor obté aquestes dades, s'interpreten i es guarden en la base de dades per la seva posterior utilització en el joc i en “Detalls de Sessió”.

4.3.2. Tutorial pel professor

Quan un professor es registra, la plataforma web es posa en mode tutorial, el que vol dir que haurà de passar per un seguit de pàgines de la plataforma per tal de ensenyar-li com crear totes les dades necessàries perquè els seus alumnes puguin començar a jugar.

En la imatge de la Figura 68 es mostra quin és el recorregut que ha de seguir un professor per tal d'acabar el tutorial.

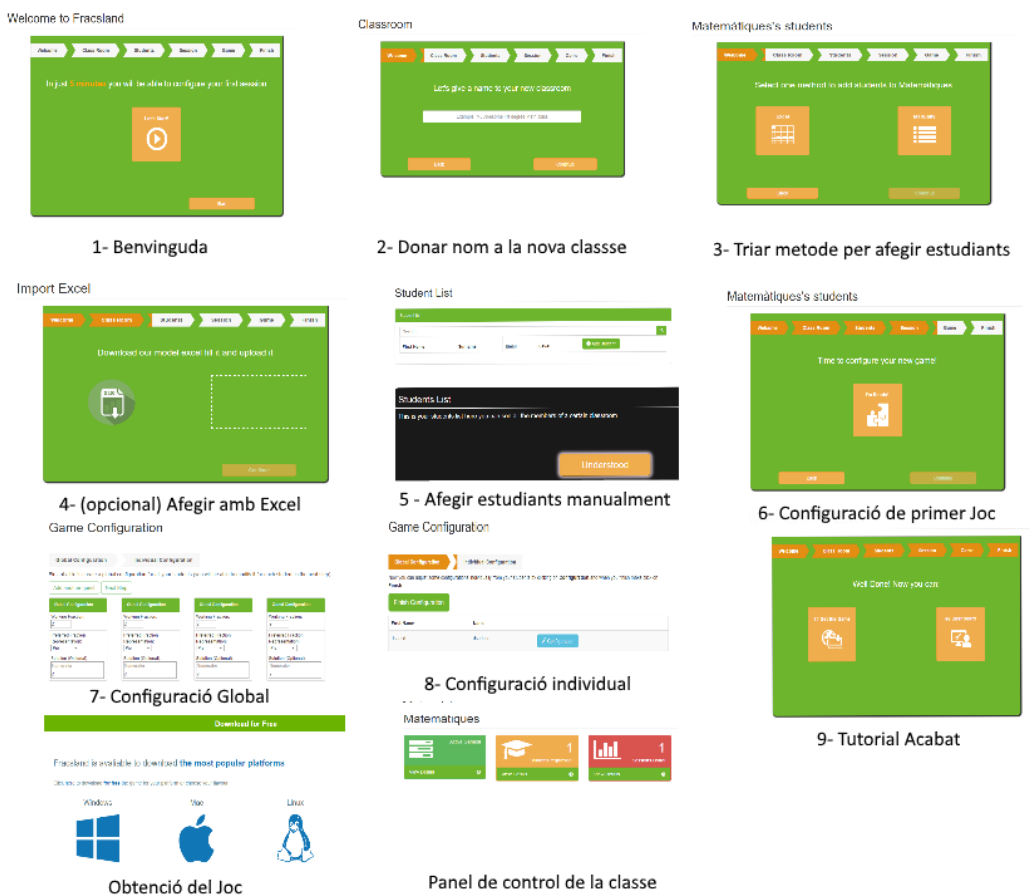


Figura 68: Tutorial professor.

En aquest tutorial es realitzen diferents etapes:

- Benvinguda: Aquesta pàgina dona la benvinguda al professor i li explica que en uns 5 minuts serà capaç de configurar la seva primera sessió de joc.
- Donar nom a la nova classe: A partir d'aquest moment el procés és exactament el mateix que s'utilitza per la creació d'una nova classe, d'aquesta manera a partir del moment en que el professor acabi el tutorial, la creació d'una nova classe li resultarà familiar.
- L'apartat 4 opcional i permet al professor introduir els seus alumnes tots de cop descarregant-se un Excel model, omplint-lo i arrossegant-lo al quadrat amb línies discontinues.

Un cop acabat el tutorial, en la l'apartat 9 s'indica al professor, on pot obtenir el joc o bé se li ofereix anar al seu panell de control. Quan el professor faci click en qualsevol de les dues

opcions la pàgina que triï s'obrirà en una nova pestanya, ja que així li resultarà més fàcil poder realitzar les dues opcions en cas de que sigui necessari.

4.4. Disseny de la Base de Dades

La base de dades de Fracsland permet a totes les parts del sistema tenir un lloc on poder obtenir i desar les dades perquè la plataforma funcioni i permeti la persistència de dades. El seu esquema es detalla en la Figura 69, que serveix al servidor mantenir la persistència de les dades, ja sigui de les classe, els estudiants o de l'estat actual de la planificació que està fent el professor.

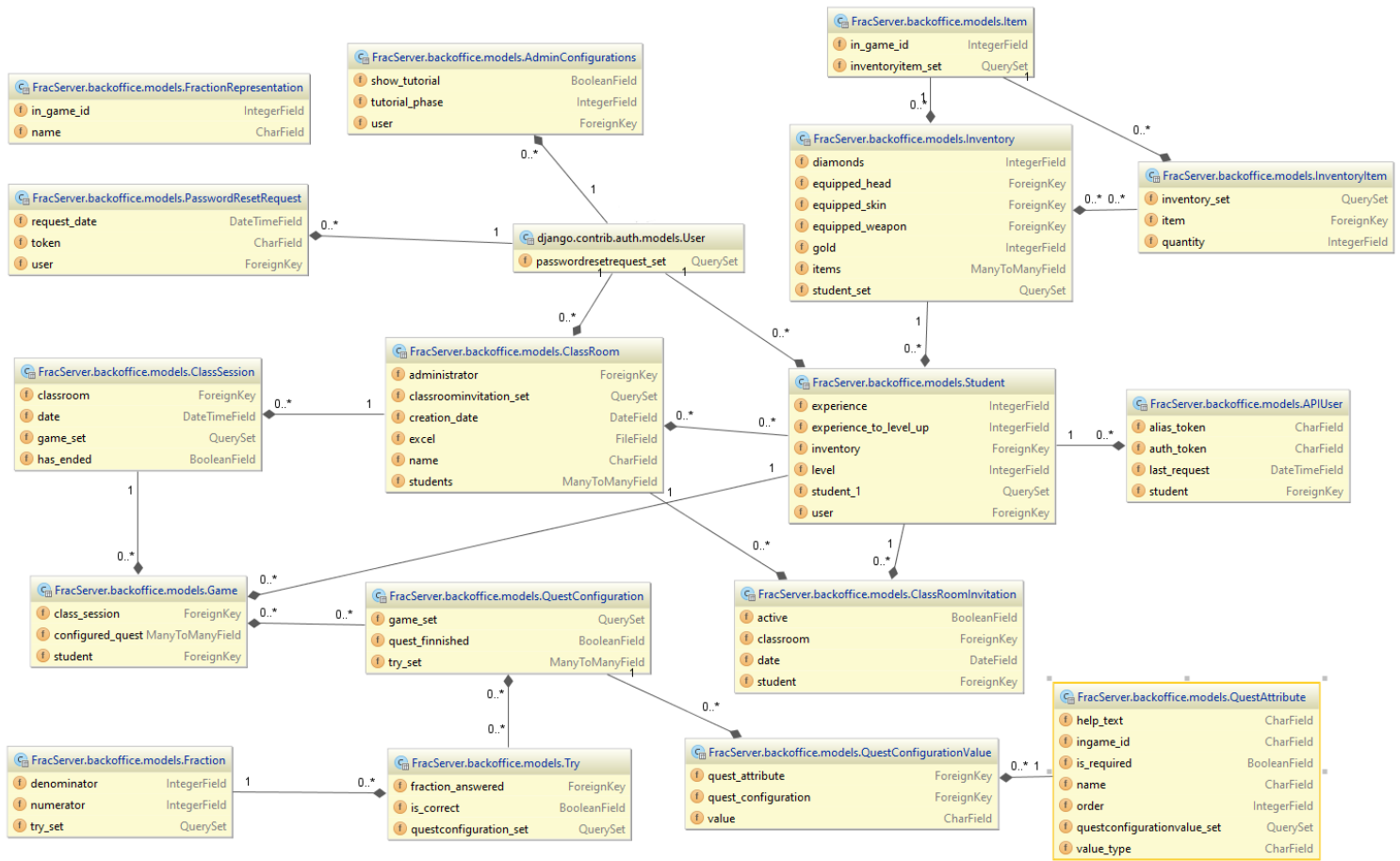


Figura 69: Diagrama de la base de dades de Fracsland.

A continuació s'expliquen les entitats organitzades segons la funció que aporten a la plataforma:

4.4.1. Components del joc

Aquestes entitats son les encarregades de proveir al joc de persistència de dades.

Item:

Aquesta entitat només guarda la ID d'un "item" per tal de distingir-los entre ells. Aquests items es corresponen objectes en el joc, com poden ser armes, barrets, vestits, etc.

InventoryItem:

Aquesta entitat conté una clau forana amb un objecte del tipus item i indica també quina quantitat d'aquest objecte té l'estudiant en qüestió.

Inventory:

Aquesta entitat representa els objectes del joc que un determinat estudiant té en un cert moment, ja que dins del joc podrà comprar-ne i es precisa d'algun suport en la base de dades on poder donar permanència al objectes que el jugador va adquirint durant les seves sessions de joc.

Aquesta taula guarda els items que té l'usuari en l'inventari, juntament amb la quantitat de diamants i or de la que disposa.

A part també indica quina arma, barret i/o aparença té equipada el jugador en la actualitat, mitjançant claus foranes cap a el "Item" en qüestió.

Fraction:

Aquesta entitat representa una fracció i conté el seu numerador i denominador, s'utilitza per emmagatzemar les solucions de les tasques assignades al estudiants i els intents que han realitzat durant el joc.

4.4.2. Joc configurable

Aquestes entitats són les encarregades de permetre que un joc sigui configurable.

QuestAttribute:

Aquesta entitat conté quins paràmetres seran configurables del joc, això ho fa utilitzant els següents camps:

- 1- Name: Nom de la configuració en qüestió (per exemple: solució, fracció a treballar...).
- 2- Is_required: Serveix per dir de forma explícita si un professor està obligat a donar un valor a aquest atribut (en Fracsland per exemple la Solució és opcional, però quina fracció es treballarà en aquest nivell és un atribut obligatori).
- 3- Order: En quin ordre s'ha de mostrar dins l'apartat de configurar un nou joc (veure apartat "Creació d'una nova sessió de joc").
- 4- Value_type: Això és necessari, ja que depenent del tipus de dades que es vulguin emmagatzemar, s'haurà de mostrar un widget o un altre en el panell d'administrador del professor. Actualment es suporten: fracció, número, text, booleà i representació de la fracció.
- 5- ingame_id: Defineix de forma explícita quina és la id d'aquesta configuració en el joc, d'aquesta manera quan el joc rebí les dades del webservice sabrà de quina configuració es tracta i com ha d'interpretar-la, ja que el programador l'haurà introduït prèviament en els dos llocs (servidor i joc).

QuestConfiguration:

Aquesta entitat representa una tasca configurada i permet guardar si una tasca s'ha acabat o no i està relacionada amb tots els intents que ha realitzat l'estudiant per tal d'acabar-la.

QuestConfigurationValue:

Aquesta entitat representa el valor que un professor ha donat a una determinada configuració d'una tasca. Conté una clau forana a la tasca a la que pertany i una altra a quina configuració pertany i per últim conté el valor d'aquesta.

FractionRepresentation:

Aquesta entitat serveix per a que l'administrador de Fracsland pugui afegir nous tipus de representacions de fraccions dinàmicament (a mesura que el desenvolupament del joc avanci) dins de la configuració d'una tasca.

Game:

Aquesta entitat representa un "joc" configurat per una determinada sessió. Conté a quin estudiant i a quina sessió pertany. Està vinculat amb totes les tasques (QuestConfiguration) que se li han assignat al estudiant per la sessió en qüestió.

4.4.3. Organització de classes i sessions de joc

Aquestes entitats permeten al professor agrupar els seus alumnes en classes i organitzar i gestionar sessions de joc.

Student:

L'entitat estudiant, conté una clau forana cap a la taula "user" de Django, per a establir la relació amb els usuaris de Django i el seu manegament intern.

A part també conté el nivell del jugador (per la part de RPG del joc), quanta experiència ha acumulat en aquest nivell, quanta experiència li falta per pujar de nivell i una clau forana cap al seu inventari.

ClassRoom:

Aquesta L'entitat representa una classe d'una escola o qualsevol altre tipus de centre docent. Conté el nom de la classe, una clau forana com al seu administrador (és a dir el professor), la seva data de creació, amb quin Excel va ser generada (opcional) i està relacionada amb tots els estudiants que són membres de la classe.

ClassSession:

Aquesta entitat representa una sessió de joc i conté la data en la que va ser creada, a quina classe pertany i si ha acabat o no (aquest paràmetre canvia quan el professor apreta tancar la sessió, en la llista de sessions disponibles (veure secció 7 de l'apartat 3.2).

Try:

Aquesta entitat emmagatzema tots els intents que ha realitzat un estudiant en una determinada tasca, també ens indica si aquest intent ha estat exitós o no (així no s'ha de calcular cada vegada que es vulguin explotar les dades més endavant).

4.4.4. Comunicació joc Servidor

Aquestes son les entitats que s'utilitzen en el webservice per tal de poder gestionar la comunicació client servidor.

APIUser

Aquesta entitat serveix perquè el webservice pugui comunicar-se amb el Joc, ja que conté a quin estudiant representa, quina va ser la seva última petició (útil per veure si el joc està encara actiu i encara s'està jugant) i dos "tokens" (testimonis) que serveixen per a què l'usuari només envii un cop el seu correu electrònic i la seva contrasenya per la xarxa en una sessió del joc. Per a comunicar el joc amb la plataforma durant el joc, els missatges s'enviaran amb aquests dos tokens i no amb el correu i la contrasenya, fet que proporciona una mica més de seguretat a la plataforma, ja que aquests testimonis canvien cada cop que l'usuari torna a entrar accedir al joc.

Aquesta entitat tot i que podria ser integrada a l'entitat "Student", s'ha considerat de forma separada per diferenciar en diferents capes les entitats purament d'emmagatzematge de dades de les taules de lògica de la plataforma.

4.4.5. Utilitats

Admin Configurations:

Aquesta entitat està relacionada amb un usuari, que sigui professor, i serveix per controlar si s'ha de mostrar o no el tutorial inicial i en quina fase del tutorial es troba l'usuari, en el cas de que tanqui sessió o s'interrompi el tutorial per alguna raó desconeguda.

Això ho pot fer gràcies a un middleware que comprova si l'usuari hauria d'estar en el tutorial i en cas afirmatiu el porta a la fase del tutorial on pot continuar la seva feina.

ClassRoomInvitation:

Aquesta entitat protegeix als estudiants del fet que qualsevol usuari pugui afegir-los a la seva classe i modificar els seus paràmetres (amb intencions malicioses). És útil ja que si un professor agrega a un estudiant que pertany a una altra classe d'un altre professor, primer se li pregunta si vol accedir o no a la classe, donat que en el moment en què l'estudiant figura en la classe, el professor podrà canviar les seves dades d'accés.

Això es realitza utilitzant una clau forana cap a la classe on es vol convidar al estudiant, una altra a l'estudiant en qüestió, la data en què va ser realitzada la petició i si la petició segueix activa. En el cas que l'estudiant declini la proposta es desactivarà aquesta petició i el professor podrà veure que la seva petició ha estat declinada i li permetrà tornar-li a demanar si ho considera oportú.

PasswordResetRequest:

Aquesta entitat permet a un usuari, recuperar la seva contrasenya via correu electrònic si ho sol·licita. Conté l'usuari que ha fet la petició d'un testimoni aleatori i la data en què s'ha realitzat la petició. Si la petició caduca en el cas que faci més d'una hora de la seva petició. En el cas que hagi expirat el termini, s'ha de tornar a fer.

4.5. Interfície de Comunicació amb el Joc

La interfície de comunicació amb el joc consta de les següents funcionalitats, visibles en la taula 70 juntament amb la URL que el joc ha de demanar per tal d'accedir a les dades de l'apartat en concret.

Nom recurs	Url
Autenticar	api/v1/auth/
Realitzar Intent	api/v1/try_quest/
Obtenir dades estudiant	api/v1/player_stats/
Obtenir jocs actius	api/v1/active_games/
Obtenir configuració de joc	api/v1/game_config/<id_del_joc>
Obtenir Inventari	api/v1/get_inventory/
Equipar objecte	api/v1/equip_item/
Actualitzar Inventari	api/v1/update_item/
Actualitzar Quantitat Or	api/v1/update_gold/
Actualitzar Quantitat Diamants	api/v1/update_diamonds/
Afegir Experiència	api/v1/add_experience/

Figura 70: Funcions del webservice juntament amb la URL per invocar-les.

Cal recalcar que aquest webservice ha estat dissenyat purament per assegurar la permanència de dades del joc, no per evitar que els jugadors puguin fer trampes, ja que resulta evident que un atacant podria afegir-se objectes, or i experiència a ell mateix realitzant les peticions adequades.

La part positiva és que si arribés el cas, aquest jugador només es podria modificar els seus propis atributs, ja que per accedir als atributs d'un altre jugador primer hauria de trobar la manera de saber les credencials d'aquest jugador.

A continuació es detallen cadascuna de les funcionalitats de la Taula 70:

1. Autenticar:

Un procés essencial del webservice és la part d'autenticació. Aquesta part del webservice és la que es crida quan un usuari vol fer login al webservice, i requereix que la capçalera de la petició contingui una HTTP_AUTHORIZATION amb el correu electrònic del alumne i la seva contrasenya.

Si aquestes dades són correctes envia al client un JSON amb els següents atributs:

- 1- Auth_token: Aquest testimoni permet al client enviar una "contrasenya temporal", per així evitar haver d'enviar la seva contrasenya varies vegades. Aquest testimoni canvia cada cop que l'usuari accedeix al joc.
- 2- Alias_token: Aquest altre testimoni temporal (i únic en la base de dades) serveix per substituir la funció del email en la autenticació evitant així, al igual que amb la contrasenya, l'enviament repetit d'aquesta dada sensible. Aquest testimoni també canvia cada cop que el jugador accedeix al joc.

D'aquesta manera si per algun motiu les dades fossin capturades, l'atacant només disposaria d'aquestes dades temporals.

Tots els apartats següents han de rebre els dos testimonis explicats anteriorment per poder funcionar, ja que a partir d'ells obtenen l'alumne que està realitzant la petició.

Retorn de l'apartat (JSON):

```
{
  "at": "<alias_token>",
  "ht": "<auth_token>",
  "requestID": "<id_request>"
}
```

2. Realitzar intent:

Un altre procés important és quan el jugador realitza un intent. Aquest apartat rep el numerador i el denominador del intent de l'usuari, comprova si és correcte i actualitza la base de dades introduint-hi el nou intent. D'aquesta manera el professor podrà veure els progressos del alumne tal i com s'ha explicat anteriorment en l'apartat "Panel d'Administració Professor". Aquest apartat retorna true si l'alumne ha encertat la resposta i false si no ho ha fet.

3. Obtenir dades estudiant:

Per a obtenir dades de l'estudiant, es retornen les dades de l'estudiant que el joc necessita formatades d'aquesta manera:

```
{
  "gold": "<quantitat>",
  "level": "<nivell_actual>",
  "experience": "<quantitat>",
  "nick": "<nom_del_alumne>",
  "requestID": "<id_request>",
  "diamonds": "<quantitat>",
  "experience_to_level_up": "<experiencia_per_pujar_de_nivell>"
}
```

4. Obtenir jocs actius:

Quan el servidor vol consultar els jocs actius, el webservice fa una petició als jocs enviant el següent (observar que games_data és una llista, per tant contindrà tants elements com jocs tingui actiu l'alumne):

```
{
  "games_data": [{
    "classroom": "<clase_a_la_que_pertany_el_joc>",
    "date": "<data_creació_sessió>",
    "game_id": "<id_del_joc>"
  }],
  "requestID": "<id_request>",
}
```

5. Obtenir configuració de joc:

Aquesta funcionalitat envia totes les quests configurades per un joc en concret identificat per la id del joc en la URL. Així, el joc disposa de les dades necessàries per crear una experiència personalitzada per l'estudiant. Les dades s'envien en el següent format: (quest_config és una llista, podria contenir tantes dades com quests tingui configurades l'usuari)

```
{
  "quests_config": [{
    "frac_rep": "<representació_fracció>",
    "working_fraction": "<quina_fracció_es_treballa>",
    "conf_id": "<id_de_la_configuració>",
    "solution": "<solució_quest>"
  }],
  "requestID": "<id_de_la_request>",
}
```

6. Equipar objecte:

Aquesta secció del webservice rep la id del objecte que el jugador s'ha equipat per tal que si es desconnecta del joc, continuï tenint equipat aquest objecte el pròxim cop que es connecti.

Valor de retorn: 200 OK (el protocol HTTP retorna com a codi de resposta el valor 200 si tot ha anat bé tal i com s'especifica més endavant).

7. Obtenir Inventari:

Quan el joc fa una petició a aquesta URL el webservice retorna els continguts de la taula inventari relacionada amb l'alumne amb el següent format:

```
{
  "equipped_items": [{
    "item_id": "<id_del_item>",
    "place": "<id_cap>"
  }, {
    "item_id": "<id_del_item>",
    "place": "<id_posició_arma>"
  }, {
    "item_id": "<id_del_item>",
    "place": "<id_posició_skin>"
  }],
  "requestID": "id_de_la_request",
  "inventory": [{
    "item_id": "<id_del_item>",
    "quantity": "<quantitat>"
  }]
}
```

Si l'usuari no té cap objecte equipat en una de les posicions, cap, arma o "skin" "item_id" continuarà un -1.

8. Actualitzar Inventari:

Aquest apartat rep la id del objecte a afegir o actualitzar dins de l'inventari i la quantitat que es vol afegir o restar d'aquest objecte, serveix per quan un jugador aconsegueix o compra un objecte continuï disposant d'ell un cop es desconnecti del joc.

Valor de retorn: 200 OK (si tot ha anat bé)

9. Actualitzar Quantitat Or:

Afegeix o disminueix la quantitat d'or que pertany a l'estudiant, això passa quan mata a un enemic, aconsegueix una missió o compra un objecte.

Valor de retorn: 200 OK (si tot ha anat bé)

10. Actualitzar Quantitat Diamants:

Similar a afegir or, només que aquest mètode es crida quan un jugador rep una recompensa per una missió o gasta diamants en la botiga, ja que els diamants no es poden obtenir dels monstres.

Valor de retorn: 200 OK (si tot ha anat bé)

11. Afegir Experiència:

Aquesta secció es crida quan un jugador obté experiència dins del joc ja sigui acabant missions o acabant amb enemics.

Valor de retorn: 200 OK (si tot ha anat bé)

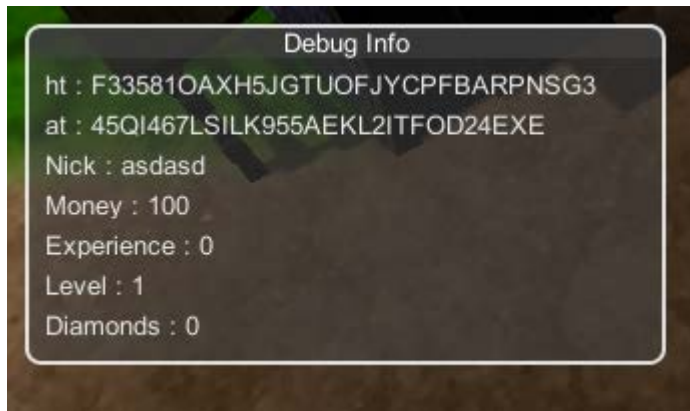
Cal destacar, que totes aquestes funcionalitats envien codis de retron per expressar l'estat de la petició. A part dels JSON abans esmentat el webservice pot retornar els següents codis, els quals són interpretats per el joc:

- 1- 200: La petició ha resultat exitosa.
- 2- 401: Error d'autenticació.
- 3- 500: Error intern del servidor.
- 4- 404: Recurs no trobat.

Per veure quines respostes esta tornant el webservice es pot activar un flag anomenat "DEBUGGING_TRANSACTIONS" i el servidor imprimirà per la sortida estàndard de la consola quines respostes esta tornant.

Anàlogament ,si en el joc s'activa el flag "DEVELOPMENT" situat a constants, imprimirà per la consola de l'editor de Unity3D quines peticions s'estan fent. Aquestes utilitats s'han implementat per a realitzar el testing de les comunicacions de forma més ràpida i còmode.

En cas de estar treballant amb una versió compilada del joc on no es disposa d'accés a la consola del editor, si es té aquest flag activat, es pot prémer la tecla F9 i és mostrarà una finestra flotant amb informació de les dades que el joc rep per part de el webservice:



Per la realització d'aquesta part del projecte s'han utilitzat vèries tecnologies:

- 1- Django i Python tant per el servidor web com per la API.
- 2- HTML5 usant Bootstrap, JQuery i JQueryUI per el panel de control del professorat.
- 3- SQLite i PostgreSQL com a sistemes gestors de bases de dades.

4.6. Justificació de les tecnologies utilitzades

Django

Django és un “framework” (Django, 2016) que permet desenvolupar aplicacions web a alt nivell, fet que permet un desenvolupament centrat en les característiques del producte a realitzar i no en de les parts comunes que tota plataforma web. Així, els requeriments de Fracsland són:

- 1- Sistemes de autenticació. Tant per part del professor com de l'administrador
- 2- Sessions.
- 3- Comunicacions amb la base de dades.
- 4- Middlewares.
- 5- Validació de formularis.

L'ús de Django, a part de proporcionar velocitat de desenvolupament, és molt beneficiós a nivell de seguretat, ja que evita problemes i vulnerabilitats degudes a errors comuns d'implementació³, ja que els sistemes utilitzats per Django han estat provats i mantinguts durant anys.

Sobre l'escalabilitat també s'ha tingut en compte que productes com Instagram (Instagram, 2012) i la web del Washinton Post (Holovaty, 2005) confien en ell. Això demostra que pot ser una solució escalable que permetrà créixer al projecte a mesura que el transit creixi.

³ Els Hackers tenen mètodes automatitzats que ataquen a vulnerabilitats conegudes i catalogades com freqüents en el disseny de plataformes web.

Django compta amb una gran comunitat de desenvolupadors i una documentació molt rica, fet que resulta ser un punt a favor davant de dubtes i problemes durant el desenvolupament.

Un punt a destacar sobre Django que li ha donat notorietat per sobre de qualsevol framework disponible en PHP (Php Group, 2016) (l'altre opció considerada) és el fet que té una corba d'aprenentatge molt suau i que utilitza Python. Així, aquesta primera versió de Fracslan pot servir de base per treballs finals de grau d'Intel·ligència Artificial i Machine Learning. A més a més, el fet que Python (Python Software Foundation, 2016) és un llenguatge que en la Universitat de Barcelona s'ensenya per aquests propòsits, permetrà una fàcil integració d'aquests sistemes dins de la plataforma.

HTML5, Bootstrap Admin, JQuery i JQueryUI

Per la implementació del client web s'ha utilitzat HTML5 (Google, 2016) que és l'opció més popular actualment⁴.

També s'ha utilitzat Bootstrap Admin (Miller, 20016), que és un conjunt de plantilles web i widgets pensats exclusivament per la creació de panells de control web.

Això ens proporciona un conjunt d'eines de visualització molt bona de cara a mostrar dades en la plataforma i segueix una estètica que l'usuari pot relacionar fàcilment amb l'estètica de qualsevol panell d'administració web, millorant així la usabilitat d'aquesta.

A part com utilitza Bootstrap (Otto, 2016) de base, ens permet crear interfícies responsives amb facilitat, fet que permet a l'usuari poder usar la plataforma en dispositius amb grandàries de pantalles variades.

El panell d'administració tant per el professor, com per l'administrador de Fracslan és un panell que necessita dinamisme i que ha de poder interaccionar amb l'usuari, aquí és on resulten útils les següents eines:

JQuery (jQuery Foundation, jquery.com, 2016) i JQueryUI (jQuery Foundation, 2016) , ens faciliten funcions com "drag and drop", elements que canvien d'aparença al llarg del temps i manipulació del DOM, sense necessitat d'implementar aquestes funcions des de zero. A part també compten amb el beneficis de una gran documentació i una comunitat activa.

SQLite i PostgreSQL

Durant les primeres etapes del desenvolupament, quan el model de la base de dades és més sensible a canviar i la base de dades es esborrada amb freqüència, SQLite (SQLite Consortium, 2016) resulta ser una opció més còmode que PostgreSQL (PostgreSQL Global Development Group, 2016), ja que tot el que es necessita és un arxiu on s'emmagatzema la base de dades. Per tant, esborrar-lo és fàcil i ,com Django el torna a crear cada cop que es migra la base de dades, és totalment transparent al desenvolupador.

En etapes posteriors del desenvolupament, en les quals el model de la base de dades roman inalterat, es comença a utilitzar PostgreSQL ja que és el Sistema Gestor de bases de dades que s'utilitzarà en producció. PostgreSQL és conegut per ser un sistema gestor de bases de

⁴ Fa uns anys també es feia servir Flash per aques tipus de interfícies webs que requerien coses com drag and drop o efectes.

dades relacionals robust, que suporta transaccions atòmiques, afavoreix la consistència de dades i la seva durabilitat.

A part, a l'igual que passava amb Python, PostgreSQL és un sistema que s'ensenyava en la Universitat de Barcelona, fet que afavoreix possibles contribucions i ampliacions a aquest projecte per part dels estudiants de forma fàcil, sense necessitat d'invertir hores en formació. Aquesta ha estat una de les raons de pes davant de Mysql (Oracle Corporation, 2016), que hagués estat un molt bon candidat també.

Capítol 5. Resultats i Simulacions

En aquest capítol es presenta la plataforma resultant, els tests d'usuari i algunes proves referents al servidor.

5.1. Plataforma Resultant

Per poder il·lustrar que s'ha aconseguit ara es mostrarà el procés que es segueix des del moment que un professor es registra fins que un alumne resol una missió mitjançant un exemple concret en la plataforma real.

Primer de tot el professor es registra mitjançant el formulari de registre situat a l'índex de la plataforma web tal i com es pot veure en la figura 71:

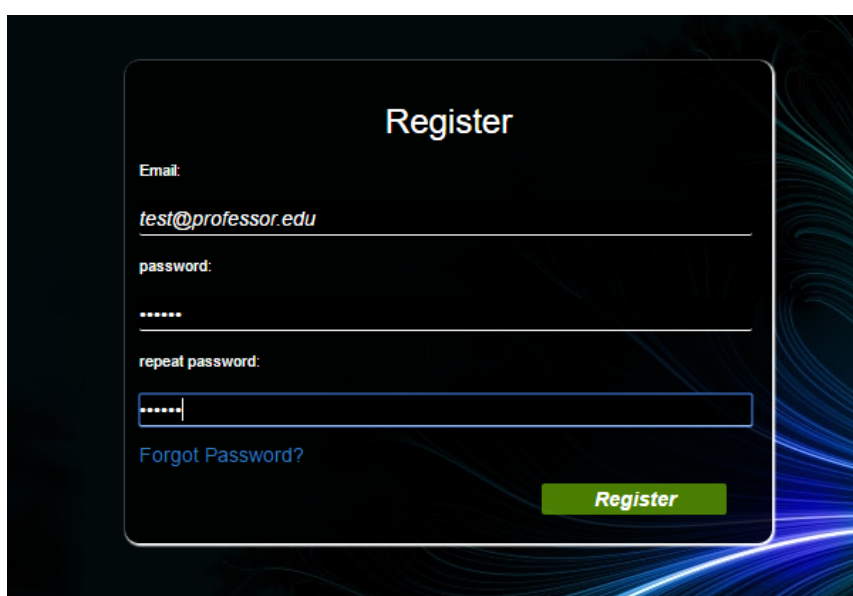
A screenshot of a web registration form titled "Register". The form is set against a dark background with a blue and green abstract pattern on the right. It contains three input fields: "Email:" with the text "test@professor.edu", "password:" with masked characters "*****", and "repeat password:" also with masked characters. Below the password fields is a link "Forgot Password?". At the bottom right of the form is a green button labeled "Register".

Figura 71: Registre professor

Crea una nova classe (Figura 72):

A screenshot of a web interface for creating a new classroom. At the top is a horizontal navigation bar with six steps: "Welcome", "Class Room", "Students", "Session", "Game", and "Finish". The "Class Room" step is currently active. The main area has a green background and the text "Let's give a name to your new classroom". Below this is a white input field containing the text "Matemàtiques 2016 (primavera)". At the bottom are two orange buttons: "Back" on the left and "Continue" on the right.

Figura 72: Exemple de creació classe

En la següent figura (figura 73) es pot veure com el professor ha decidit crear la seva classe mitjançant un Excel el que s'ha fet es descarregar un model d'Excel que proporciona la plataforma, s'ha omplert (els noms dels alumnes son totalment arbitraris, s'han creat noms i cognoms realistes per tal de donar una millor estètica al exemple) i s'ha arrossegat a la capsa amb línies discontinues per tal d'afegir tots els estudiants de la classe de cop:

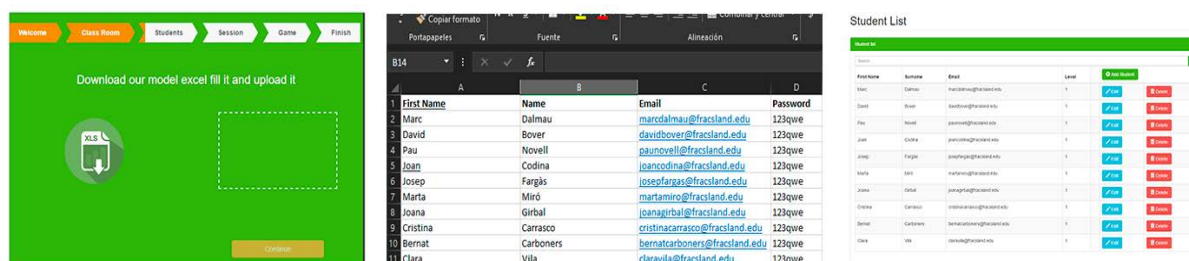


Figura 73: Plana de l'administrador on descarregar el model d'Excel i pujar-lo després (esquerra), Excel plé d'estudiants (centre) i llista d'estudiants pujats en la plataforma via Excel.

En la següent figura (figura 74) es mostra com el professor configura 3 qüestions per als seus estudiants triant com a fraccions a treballar les del 6, 7 i 10, i com a preferències per la representació fracció: "rectangle", "pie" i "rectangle", deixant al sistema triar les solucions aleatòriament.

Global Configuration

Individual Configuration

First of all let's create a global configuration for all your students (you will be able to modify it for each student)

Add another quest

Next Step

Quest Configuration

Working Fraction: 6

Preferred Fraction Representation: Rectangle

Solution (Optional): Numerator 6

Quest Configuration

Working Fraction: 7

Preferred Fraction Representation: Pie

Solution (Optional): Numerator 7

Quest Configuration

Working Fraction: 12

Preferred Fraction Representation: Rectangle

Solution (Optional): Numerator 12

Figura 74: Exemple de configuració d'un joc amb 3 qüestions.

Per tal de fer aquesta explicació més curta anem a utilitzar el sistema de personalització que ens ofereix fracslan per fer que en Pau (per exemple) faci només una quest del 7, tal i com es pot veure en la figura 75 a l'esquerra es veu la llista amb tots els estudiants i a la dreta la nova configuració específica per en Pau, la configuració d'en David també està visible per poder il·lustrar el fet de que varis estudiants poden tenir configuracions diferents en una mateixa sessió de joc:

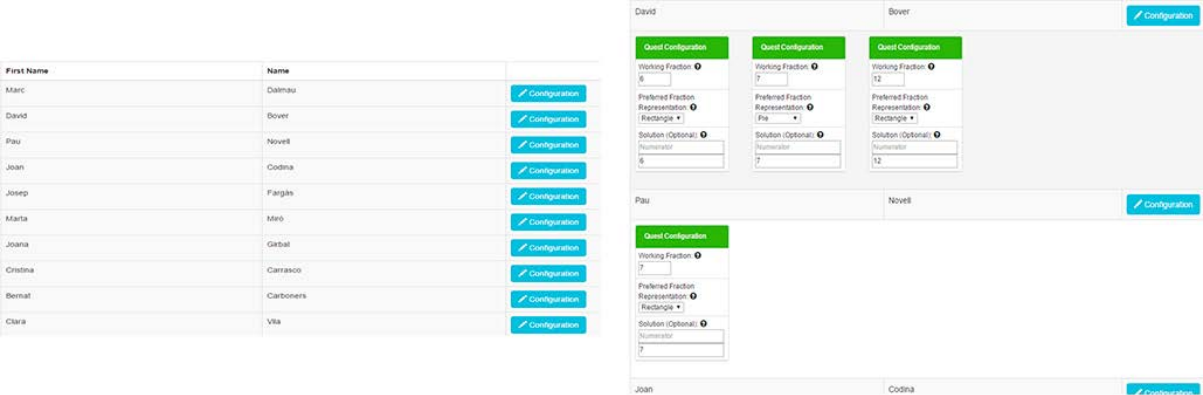


Figura 75: Llista estudiants en pantalla de configuració individual de quests (esquerra) i configuracions diferents per 2 estudiants desplegadas en la mateixa llista (dreta).

En la següent figura (figura 76) es pot veure el panell del que disposa el professor per poder observar quin és el progrés dels seus estudiants. En aquest cas encara no ha començat la sessió:

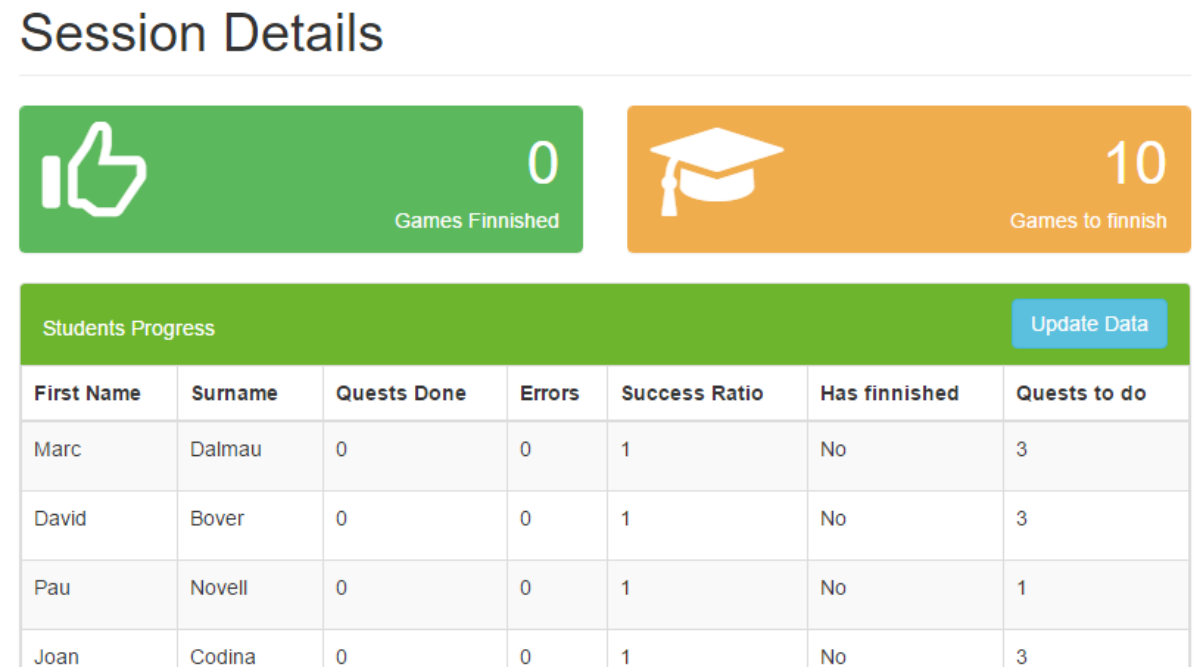


Figura 76: Panell de control sessió.

A partir d'aquest moment deixem a part un moment el paper del professor i anem a centrar-nos en la trajectòria que seguiria el suposat alumne al que hem anomenat Pau. El primer que es farà es accedir al joc mitjançant els credencials que ha creat el professor per ell, un cop dins apareixerà la pantalla de selecció de classe, com en Pau només disposa d'una sessió de joc disponible només en disposa d'una a la llista tal i com es pot apreciar en la figura 77:

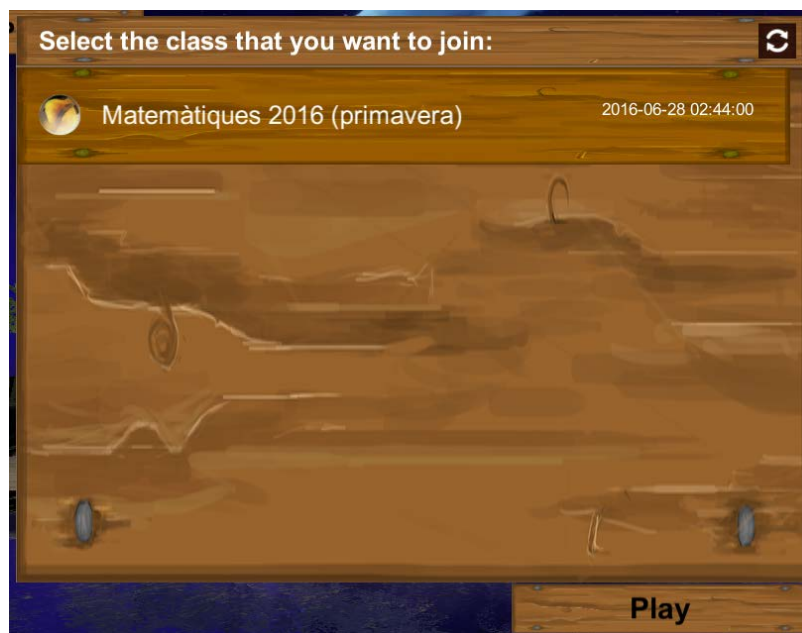


Figura 77: Llista sessions a les que pot accedir un estudiant amb una sessió configurada.

En la següent figura (figura 78) podem veure com el jugador a trobat a Lord Barus i té el menú amb 3 quests disponibles per fer entre elles una destacada (on la remuneració és més alta), la del pont ja que és la única que coincideix amb la representació que fraccions que el professor ha triat:



Figura 78: Llista de quests disponibles que té el jugador amb la quest "Destroyed Bridge!" com a destacada.

Després de superar perills i obtenir les fraccions i els materials necessaris per portar a terme la missió del pont en Pau es troba davant del pont i realitza un intent, falla i després de reviure torna al pont i aconsegueix reconstruir el pont correctament. Aquesta situació descrita anteriorment es pot veure en la figura 79:



Figura 79: En la primera imatge es pot veure el jugador contestant incorrectament, en la imatge de la dreta es pot veure la conseqüència (ofegar-se), en la imatge inferior esquerra es pot veure al jugador contestant correctament i a la dreta com creua el pont.

Per últim com en Pau només hauria de fer una missió al anar a cobrar la recompensa pot marxar de Fracsland (veure figura 80):

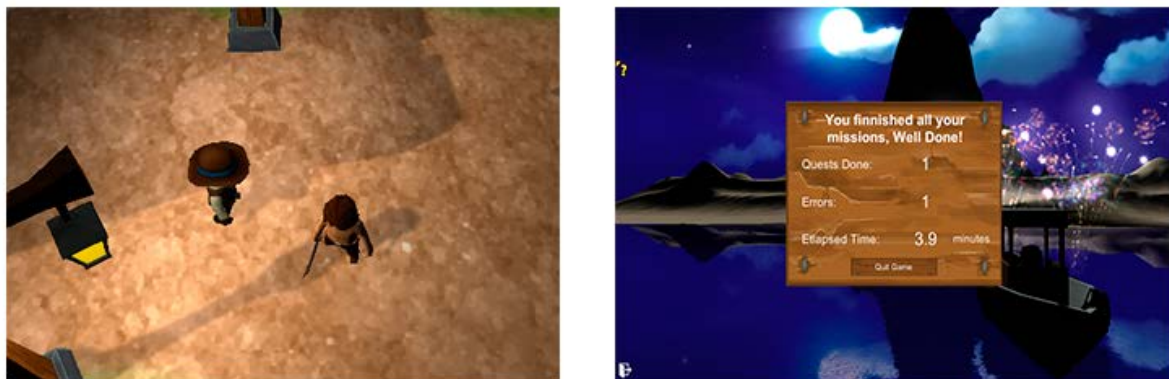


Figura 80: A l'esquerra el jugador a punt de parlar amb Lord Barus per poder obtenir la seva recompensa, a la dreta el jugador marxant de Fracsland amb un panel que li indica quants errors ha fet i el temps que ha trigat en acabar totes les quests.

Per últim només cal veure què el professor ha rebut la informació de la partida d'en Pau cada cop que realitzava un intent, en la figura 81 es pot veure la informació que pot veure el professor després d'acabar la partida d'en Pau:

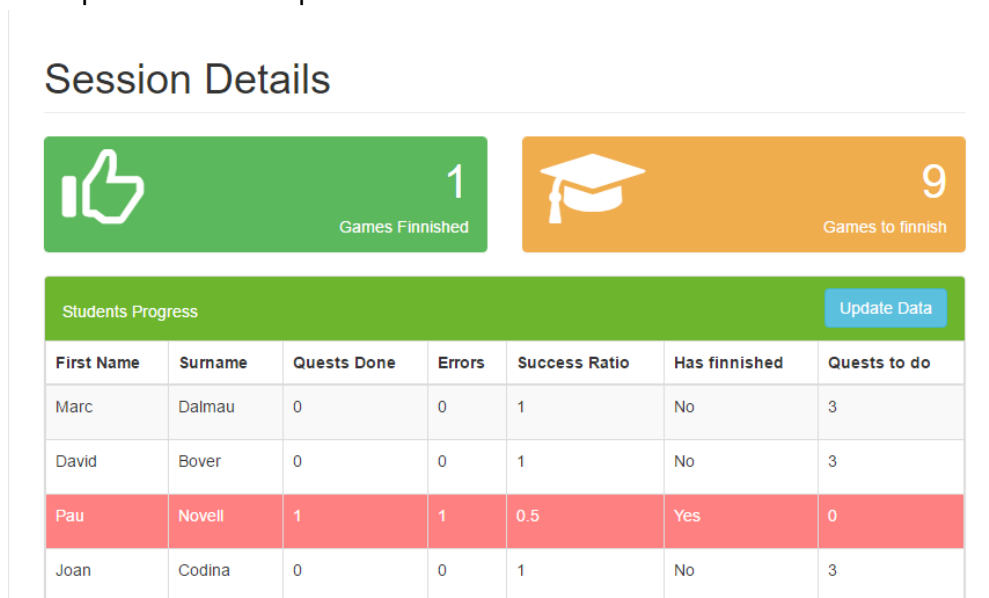


Figura 81: Llista d'estudiants de la sessió amb un d'ells que ha acabat el joc amb un success ratio baix.

L'exemple explicat ajuda a il·lustrar d'un mode simplificat el funcionament dels mecanismes que s'han implementat (s'han omès aspectes com l'exploració de les zones, combats contra enemics, obtenció de fraccions...), cal recalcar que ara per ara el joc disposa de 3 quests, fet que no dona una excessiva llibertat al jugador sobre quina quest realitzar per cada tasca, ja que per exemple si es vol realitzar una quest que tingui una representació de fracció rectangular només n'hi ha una implantada, actualment.

La part positiva és que Fracsland ha estat dissenyat des del principi per ser fàcil d'ampliar, així que aquest problema es solucionarà quan s'afegeixin més quests al joc per cada tipus de representació de fracció.

5.2. Tests amb usuaris:

Per realitzar els tests d'usuari s'ha demanat als usuaris que facin el següent:

Com a professor:

- 1 – Registrar-se.
- 2 – Crear una Classe.
- 3 – Registrar 2 estudiants.
- 4 – Crear 2 quests.
- 5 – Anar al panell d'administració i entrar en la sessió creada.
- 6 – Actualitzar dades de les quests.
- 7 – Fer logout.

Com estudiant:

- 1 – Fer login.
- 2 – Accedir a la classe creada.
- 3 – Acceptar una quest.
- 4 – Realitzar la quest.

- 5 – Cobrar la quest.
- 6 – Acceptar una altre quest i realitzar-la.
- 7 – Cobrar la quest i acabar el joc.

Els resultats dels tests s'han recollit fent una petita enquesta als usuaris que han realitzat el test, els resultats de l'enquesta es poden veure en l'apèndix B.

A partir dels tests, podem veure que en general els usuaris han gaudit d'una bona experiència, tot i que hi ha certs punts tant del servidor com del joc que es poden millorar.

Tot i que el temps del projecte és limitat s'ha pogut realitzar una petita iteració després dels tests d'usuari que ha permès resoldre problemes crítics detectats i algunes de les queixes més generals, sobretot relacionades amb problemes d'usabilitat.

Tot i que no s'ha pogut arreglar tot el que el que ha estat suggerit per els beta testers, aquestes dades resultaran útils per la següent iteració del projecte, ja que ens permeten saber que és el primer que hauríem de millorar de les plataformes i que és el que s'ha de mantenir i/o potenciar.

5.3. Temps de resposta al servir una pàgina utilitzant PingDom

S'han realitzat proves de rendiment per el servidor.

En el primer test s'ha valorat el temps de resposta del servidor per servir una pàgina utilitzant l'eina Pingdom (PingDom AB, 2016). En la figura 82 es pot veure un resum del resultat del test:

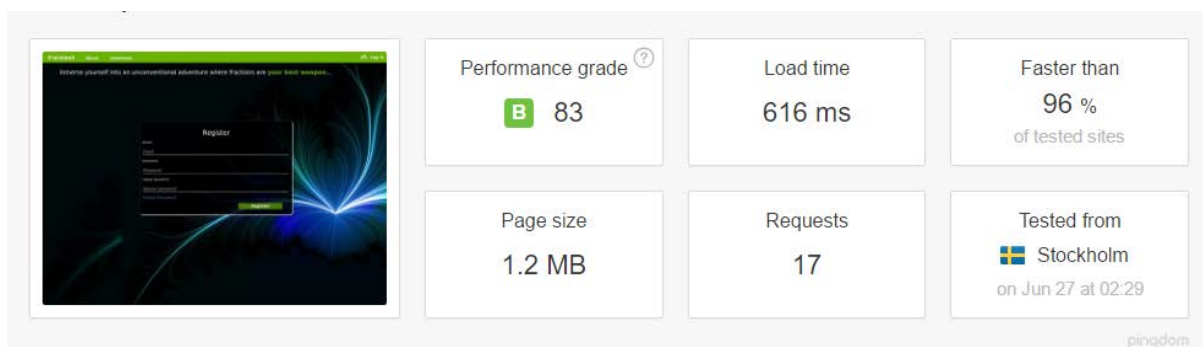


Figura 82: Resultats tests realitzats a <https://tools.pingdom.com>.

Els resultats mostren les següents característiques:

- 1 – Performance grade (Google Developers, 2016): És el grau de rendiment obtingut de valorar factors com el temps que el navegador pot guardar en caché els recursos obtinguts en la pàgina, com es serveixen els continguts estàtics, si es realitzen requests dolentes (es a dir requests que no porten enlloc), si hi ha redireccions o no, si la mida de la request és massa gran. (per més informació consultar: <https://developers.google.com/speed/pagespeed/>)
- 2 – El temps de càrrega de la pàgina des de que es fa la request (616 ms).
- 3 – La velocitat de descarrega de la pàgina en relació a altres pàgines analitzades (96%).
- 4 – La mida de la pàgina (pàgines massa grans penalitzen els temps de càrrega).
- 5 – Quantes requests han calgut per carregar la pàgina, és a dir per aconseguir el html, el css, javascripts...

6 – On esta el servidor que ha realitzat la prova (Estocolm, Suècia).

Analitzades les dades obtingudes, es pot extreure que el servidor tot i ser un Kimsufi - 4G Atom 1x1TB Server que és un servidor de gamma baixa proporcionat per l'empresa Kimsufi (Kimsufi, 2016), respon bé a les peticions per obtenir pàgines.

5.4. Proves d'estrès servidor

Un cop realitzat el primer test es pot veure que el servidor respon bé a una petició per a obtenir una pàgina. En el següent test s'ha comprovat el temps de resposta del servidor en realitzar múltiples peticions i si aquest és capaç de mantenir-se en funcionament quan el nombre de peticions per segon és elevat.

L'objectiu d'aquest test és obtenir una prova que el servidor actual serà capaç d'escalar amb el projecte.

Per realitzar les peticions s'ha utilitzat la comanda "ab" (Apache Software Foundation, 2016) que és una eina per comprovar quantes peticions per segon pot afrontar una determinada instal·lació del servidor Apache, sistema utilitzat per servir la plataforma web de Fracsland.

Aquesta comanda, entre d'altres, permet triar 2 valors importants utilitzats en les proves:

- 1- La concurrència, és a dir quantes peticions simultànies rebrà el servidor.
- 2- El nombre de peticions que es faran per a la prova.

Les dades dels tests contenen els següents atributs:

- **Concurrency Level: / Complete Requests:** Aquestes dades es corresponen al nombre de peticions simultànies que es fan i quantes peticions es faran durant la prova.
- **Time taken for tests (seconds):** Temps total transcorregut per acabar la prova en segons.
- **Failed requests:** Quantes peticions han fallat durant la prova.
- **Total transferred (bytes):** Quants bytes s'han transferit durant la prova.
- **Requests per second [requests/sec] (mean):** Quantes peticions per segon s'han fet de mitja.
- **Time per request [ms] (mean):** Quin es el temps mig per petició sense tenir en compte que s'estan realitzant varies peticions en paral·lel.
- **Time per request: [ms] (mean, across all concurrent requests):** Quin es el temps mig de resposta tenint en compte que s'estan realitzen varies peticions a la vegada.
- **Transfer rate: [Kbytes/sec] received:** Quants kbytes per segon s'han rebut.

En la figura 83 es poden veure els resultats de les proves:

Concurrency Level: / Complete Requests:	1/100	2/100	10/100	50/100	100/100	100/1000
Time taken for tests (seconds):	16.147	9.467	8.329	7.384	7.375	73.147
Failed requests:	0	0	0	0	0	0
Total transferred (bytes):	724200	724200	724200	724200	724200	7242000
Requests per second [requests/sec] (mean):	6.19	10.56	12.01	13.54	13.56	13.67
Time per request [ms] (mean):	161.466	189.333	832.948	3692	7375.089	7,314,713
Time per request: [ms] (mean, across all concurrent requests):	161.466	94.667	83.295	73.842	73.751	73,147
Transfer rate: [Kbytes/sec] received:	43.8	74.71	84.91	95.78	95.89	96.69

Figura 83: Proves rendiment servidor.

A partir dels resultats obtinguts en les proves es poden extreure les següents conclusions per cada dada, obtinguda en els tests:

- **Time taken for tests (seconds):** El que es conclou observant les dades és que com més peticions fem en paral·lel més ràpid es realitzen totes fins arribar a unes 50 peticions, en què la millora ja no és significativa.
- **Failed requests:** Com es pot veure les peticions poden trigar més o menys segons el transit que hi hagi al servidor, però no es perden peticions per molt que el nombre augmenti.
- **Total transferred (bytes):** Evidentment en les 5 primeres proves roman igual, ja que el nombre de peticions és el mateix i en la última es 10 vegades més ja que fem 10 vegades més peticions.
- **Requests per second [requests/sec] (mitja):** Com es pot comprovar en les dades a mesura que es realitzen més peticions simultànies el nombre de peticions per segon que el sistema es capaç de processar augmenta fins arribar a un cert punt (50 peticions a la vegada) en què pràcticament la millora és mínima tot i que encara millora en menor magnitud.
- **Time per request [ms] (mitja):** Com es pot comprovar en aquesta sèrie, a mesura que anem paral·lelitzant les peticions el temps que triga una petició augmenta, ja que el servidor i el client han de processar més peticions alhora.
- **Time per request: [ms] (mitja, across all concurrent requests):** Aquestes dades mostren que tot i que el temps que triga una petició és superior a mesura que se'n fan més, la mitja tenint en compte que les peticions es fan paral·lelament millora, fins a convergir aproximadament en 73 ms.
- **Transfer rate: [Kbytes/sec] received:** Aquí es pot confirmar que el servidor ens pot dedicar una ha pogut transferir uns 97kbyte per segon aproximadament, ja que el

sistema mai no ha superat aquest valor per molt que es facin més peticions concurrents.

Com a conclusions finals de les proves, es pot observar que el sistema hagut de transferir moltes dades degut a que se li han demanat paquets de 7.2kb. Tot i així ha pogut contestar a una bona velocitat sense deixar de respondre en cap moment.

Per una banda es pot deduir que probablement el motiu que farà que s'hagi que migrar a una màquina més potent serà el moment en que desenes de professors es connectin simultàniament, Un símptoma que apareixerà llavors serà que les pàgines trigaran molt a carregar-se, fent la navegació per la plataforma lenta.

D'altra banda les proves han mostrat que gràcies al fet de demanar paquets relativament grans (demanàvem una plana HTML de 7.2kb) en comptes de petits JSON denota que probablement el sistema pot suportar la càrrega produïda per centenars de jocs simultanis. Aquests jocs només envien dades uns quants cops per minut i, a part, les dades que envien són petits JSON de l'ordre de 60 bytes que no tenen res a veure amb els paquets transferits en les proves que s'han enviat fins a 13 vegades per segon.

Per tant el problema no serà tenir classes amb molts alumnes sinó tenir moltes classes amb molts professors connectats.

5.5. Anàlisi del temps de realització del projecte

En la següent taula (figura 84) es mostra el temps total dedicat al projecte :

Tipus d'activitat	Hores dedicades	Percentatge (temps/ total) *100
Organització del projecte	23	2.99%
Analisi	38	4.95%
Disseny	43	5.6%
Formació	70	9.11%
Implementació	556	72.4%
Testing	38	4.95%
Total:	768	

Figura 84: Temps dedicat als diferents tipus de tasques i percentatge sobre el total.

Com es pot veure (en la figura 75) Fracsland en ser un projecte que conté una quantitat significativa de software el temps d'implementació ha estat el més significatiu, contrastant les hores reals dedicades al projecte amb les que es va preveure, en començar en l'apartat 1.6 (674 hores) (veure apartat 1.6), es pot veure que s'ha trigat 94 hores més de lo previst.

Això es degut a que durant l'anàlisi del projecte es va pensar que es faria una iteració fet que al final no ha estat així ja que a mesura que es materialitzava el projecte es varen fer evidents

milliores a fer tant en disseny com implementació que es van anar afegint, Tot i que ha incrementat el temps de desenvolupament del projecte, també s'ha incrementat la qualitat del mateix.

Capítol 6. Conclusions (i feines futures)

Tal i com es va esmentar en el capítol d'objectiu principal, l'objectiu del projecte Fracsland era crear una plataforma tecnològica que permetés als docents ajudar als alumnes a reforçar les competències en fraccions d'un mode personalitzat al nivell de cada alumne usant videojocs modificables. Així els alumnes reforçarien aquestes competències mentre juguen a un joc entretingut que els mantindria motivats i els impulsaria a millorar.

En el present projecte s'han aconseguit la major part dels objectius plantejats en un començament:

- 1- S'ha creat un patró per un joc flexible i configurable.
- 2- S'ha aconseguit crear una plataforma tecnològica per poder crear el patró flexible amb tots els components que aquesta requereix:
 - 2.1- Panells d'administració web (administrador i professor).
 - 2.2- Base de dades.
 - 2.3- Interfície de comunicacions amb el joc (webservice).
 - 2.4- Protocol de comunicació entre el client i el servidor.
- 3- S'ha creat un joc seriós de fraccions en 3D capaç de comunicar-se amb la plataforma tecnològica mencionada en el punt 2, capaç d'interpretar les configuracions enviades per el servidor i adaptar-se a elles.
- 4- S'ha aconseguit crear un servidor que permet general configuracions per jocs. Un punt molt important d'aquest servidor és que no només per serveix per el joc presentat en aquest treballs, sinó que pot ser utilitzat per crear altres tipus de jocs, degut a la seva naturalesa flexible a l'hora de generar configuracions.

El desenvolupament realitzat s'ha fet en tot moment pensant en l'escalabilitat del sistema, proporcionant utilitats en el joc i en el webservice que poden ser utilitzades en altres situacions per a configurar des de l'escenari del joc, fins a les pròpies qüestions.

En general els resultats obtinguts, després de fer alguns test d'usuaris, han estat molt satisfactoris, tant des del punt de vista del professor com del jugador. Fracsland ara per ara és un projecte que acaba de néixer, i en un futur es podria fer un projecte basat els test d'usabilitat que es dissenyen en Factors Humans i Computació amb l'objectiu en tornar a iterar sobre el que es té per tal de poder ajustar-se millor al que els jugadors esperen del joc i solucionar problemes d'usabilitat i jugabilitat que segur que apareixeran durant els tests més exhaustius. També es van plantejar fer avaluacions en escoles per tal de comprovar l'efectivitat del joc i si suposava una millora real en el coneixement del concepte de fracció per tal de poder millorar la plataforma en aquest aspecte. La manca de temps al final del projecte no ha permès fet aquests tipus de test, encara que es preveuen fer durant el curs vinent.

En relació a l'efectivitat del joc en el reforç de les competències sobre fraccions caldria també analitzar-la i estudiar-la per par de professors i didàctes.

Tot i haver aconseguit tots els objectius concrets que aquest projecte plantejava, encara hi ha molt lloc per feines futures.

De cara a següents projectes es podrien realitzar les següents feines:

1 - Fracsland ara per ara treballa els conceptes de fracció i equivalència de fraccions. En un futur es podrien afegir mecàniques que permetin treballar més conceptes relacionats amb fraccions:

- 1.1- Operacions sobre fraccions (suma, resta, multiplicació i divisió)
- 1.2- Simplificació de fraccions.
- 1.3- Noves representacions de fraccions.

2 - Fracsland és un joc d'un sol jugador, en un futur es podria afegir la possibilitat de jugar en mode multijugador, mitjançant la xarxa. En un inici es va plantejar el disseny del joc per a multijugador, però donat l'abast del projecte, va ser un dels requeriments que es va rebutjar en una de les fases del projecte.

3 - Ampliació de continguts: Fracsland ara disposa de 6 quests i 3 tipus d'objectes diferents (eines, barrets i aparences). Una bona feina futura podria ser crear més continguts per tal de fer el joc més extens. En aquest sentit, es podria pensar en configurar les quests dels alumnes de forma que s'adaptessin a les necessitats de cada alumne sense supervisió humana.

4 - La plataforma web de Fracsland ha estat construïda per tal de poder realitzar l'objectiu principal en el temps que es disposava encara queda molta feina que es podria realitzar en un altre treball per tal d'afegir més funcionalitats a la plataforma (millor visualització del progrés d'una classe, poder modificar les sessions de joc mentres aquestes es porten a terme, donar més poder d'administració al administrador...). Aquest fet es podria enriquir explotant les dades recollides en la plataforma per tal de poder realitzar estadístiques i obtenir conclusions de quines són les fraccions més encertades, més errades, quines són les representacions de fraccions que resulten més fàcils per als alumnes...

Capítol 7. Referències bibliogràfiques

Abt, C. C. (1970). Serious Games. A C. C. ABT, *Serious Games*.

Apache Software Foundation. (2016). *apache.org*. Recollit de <https://httpd.apache.org/docs/2.4/programs/ab.html>

Carey, R. (2015). <http://richardcarey.net/>. Recollit de [http://richardcarey.net/game-design-canvas/](http://richardcarey.net/http://richardcarey.net/game-design-canvas/)

Crytek. (2016). <https://www.cryengine.com/>. Recollit de [https://www.cryengine.com/](https://www.cryengine.com/https://www.cryengine.com/)

Django, D. S. (2016). *Django*. Recollit de <https://www.djangoproject.com/>

Epic, G. (2016). <https://www.unrealengine.com/>. Recollit de [https://www.unrealengine.com/](https://www.unrealengine.com/https://www.unrealengine.com/)

Google. (2016). *html5rocks*. Recollit de <http://www.html5rocks.com/es/>

Google Developers. (2016). *Developers Google*. Recollit de <https://developers.google.com/speed/pagespeed/>

Holovaty, A. (2005). *djangoproject*. Recollit de <https://www.djangoproject.com/weblog/2005/dec/08/congvotes/>

Instagram. (2012). *Instagram-engineering*. Recollit de <http://instagram-engineering.tumblr.com/post/13649370142/what-powers-instagram-hundreds-of-instances>

JQuery Foundation. (2016). Recollit de <https://jqueryui.com/>

JQuery Foundation. (2016). *jquery.com*. Recollit de <https://jquery.com/>

Khronos. (2016). <https://www.opengl.org/>. Recollit de <https://www.opengl.org/>

Kimsufi. (2016). *kimsufi.com*. Recollit de <https://www.kimsufi.com/es/servidores.xml>

Miller, D. (20016). *startbootstrap.com*. Recollit de <http://startbootstrap.com/>

OpenGL, M. (2016). <http://glm.g-truc.net/0.9.7/index.html>. Recollit de <http://glm.g-truc.net/0.9.7/index.html>

Oracle Corporation. (2016). *www.mysql.com*. Recollit de <https://www.mysql.com/>

Otto, M. (2016). *getbootstrap.com/*. Recollit de <http://getbootstrap.com/>

Php Group. (2016). *Php*. Recollit de <https://secure.php.net/>

PingDom AB. (2016). *tools.pingdom.com*. Recollit de <https://tools.pingdom.com/>

PostgreSQL Global Development Group. (2016). *postgresql.org*. Recollit de <https://www.postgresql.org/>

Python Software Foundation. (2016). *python.org*. Recollit de <https://www.python.org/>

SQLite Consortium. (2016). *www.sqlite.org*. Recollit de <https://www.sqlite.org/>

Unity, T. (2016). *Unity3D*. Recollit de <https://unity3d.com/es/company>

Apèndix A: Manual tècnic

A.1. Instal·lació entorn de desenvolupament: Requeriments mínims i passos a seguir

A.1.1. Requeriments mínims

Per desenvolupar el joc de Fracsland els requeriments mínims de Unity3D ja que és l'entorn de desenvolupament que demanda de més recursos, actualment són els següents (veure figura 85 informació extreta directament de la web oficial de Unity (Unity, 2016)):

PARA DESARROLLO

OS: Windows 7 SP1+, 8, 10; Mac OS X 10.8+.

Windows XP y Vista no son compatibles; y las versiones de servidor de Windows & OS X no se han probado.

GPU: Capacidades de tarjeta de vídeo con DX9 (modelo de shader 2.0). Todo lo que se haya lanzado desde 2004 debería funcionar.

El resto depende principalmente de la complejidad de sus proyectos.

Requisitos adicionales para el desarrollo de plataformas:

- iOS: Computadora Mac que ejecuta como mínimo la versión OS X 10.9.4 y Xcode 6.x.
- Android: Android SDK y Java Development Kit (JDK).
- Windows 8.1 Store Apps / Windows Phone 8.1: 64 bit Windows 8.1 Pro y Visual Studio 2013 Update 2+.
- WebGL: Mac OS X 10.8+ o Windows 7 SP1+ (solo editor de 64 bits)

Figura 85: Requeriments mínims Unity3D

Per obtenir les dades més actuals sobre els requisits mínims de Unity3D accedir a: <https://unity3d.com/es/unity/system-requirements>.

A.1.2. Instal·lació entorn de desenvolupament Joc

Per poder desenvolupar el joc de Fracsland, es necessita instal·lar Unity3D i Visual Studio o Monodevelop (per editar el codi c#).

Abans de començar però consultar apartat A.1.1 per comprovar si el sistema del què es disposa satisfà els requisits mínims de Unity.

Per obtenir els programes s'ha d'accedir a: <https://unity3d.com/es/get-unity> i descarregar-se la versió personal, tot i que es pot treballar amb la PRO sense problemes.

Un cop obtingut el programa, només cal seguir el procés d'instal·lació fins al final i ja es disposarà de tot el que sigui necessari per a començar a desenvolupar la part del joc de Fracsland.

Per obrir unity amb el joc descarregar el codi del següent repositori:

<https://gitlab.com/muriel-maths/Fracsland.git>

Després, cal obrir qualsevol escena situada en `fracsland_game/Game/Assets/Resources/Scenes` fent click en una de les icones amb extensió “.unity”, en la figura 86 es pot veure una d'aquestes escenes:



Figura 86: Arxiu .unity del joc de Fracsland

Un cop fet això s'obrirà l'editor de Unity3D i es podrà començar a treballar.

A.1.3. Instal·lació entorn de desenvolupament Servidor

Per poder desenvolupar i modificar el servidor es necessiten els següents elements:

- 1- Eclipse: Anar a <https://www.eclipse.org/downloads/> descarregar-se la última versió i instal·lar-la.
- 2- SDK de Java: Es pot obtenir d'aquí: <http://www.oracle.com/technetwork/es/java/javase/downloads/index.html>
- 3- Pydev (plugin per desenvolupar python en Eclipse): <http://www.pydev.org/download.html>
- 4- PostgreSQL: <http://www.enterprisedb.com/products-services-training/pgdownload>
- 5- Pip (per obtenir els mòduls de python utilitzats en el servidor): <https://pip.pypa.io/en/stable/installing/>
- 6- Obtenir el codi del servidor del següent enllaç: <https://gitlab.com/muriel-maths/Fracsland.git>
- 7- Accedir a la carpeta `conf` de dins del codi del servidor mitjançant una terminal i executar la següent instrucció:
`pip install req.txt` (això instal·larà Django i algunes dependències del projecte, si sorgeixen problemes al instal·lar `psycpg2` buscar informació de com fer-ho per la plataforma en la que s'està treballant)

Un cop instal·lats tots els elements i obtingut el codi hem de importar-lo a Eclipse per tal de poder desenvolupar:

- 1- Fer click en `file > import > General > Existing Projects into de Workspace`.
- 2- On posa `Select root directory` fer click en `browse` i anar fins la carpeta `FracServer`.
- 3- Fer click en `finish`.
- 4- En aquest moment apareixerà un diàleg dient que no es disposa d'un intèrpret de Python configurat, fer click en `ManualConfig`.
- 5- En la llista de `Python Interpreters` fer click en `New...`
- 6- Donar un nom al intèrpret (això és útil si s'utilitzen varies versions de Python en la mateixa màquina).

- 7- Fer click en Browse... i navegar fins on es té el executable de Python, en Windows sol ser "C:\Python27" i seleccionar el python.exe.
- 8- Fer click en OK.
- 9- En el diàleg que te com a títol "Selection needed" deixar tot tal i com està i fer click en OK.
- 10- Fer click en Apply i fer click en OK.

Un cop tinguem l'entorn de desenvolupament configurat, cal modificar settings.py del servidor ja que per seguretat no s'han fet públiques certes dades de la plataforma:

- 1- Crear una SECRET_KEY o demanar la existent al coordinador/a del projecte Fracslan.
- 2- En l'apartat DATABASES:
Omplir el nom de la base de dades, l'usuari i la contrasenya o bé demanant les dades al coordinador/a del projecte.
- 3- Omplir les dades del EMAIL_HOST, EMAIL_PORT... perquè la plataforma pugui enviar emails o bé demanar aquestes dades al coordinador/a del projecte.

També accedir a backoffice>migrations>0002_auto_20160605_1222.py i omplir la crida: `User.objects.create_superuser("<nom usuari>", "<email>", "<password>")` així es crearà una compte d'usuari per defecte per l'administrador en el moment de la migració de la base de dades. (com sempre aquestes dades poden ser demanades al coordinador/a del projecte o inventar-se).

Un cop tenim totes les dades, accedim a pgAdmin i creem una base de dades amb el nom, usuari i password definits en l'apartat DATABASES del settings.py.

Per últim ja només cal obrir una terminal accedir on hem descarregat el codi del servidor i utilitzar la següent comanda:

`python manage.py migrate` (Aquesta comanda omplirà la base de dades creada prèviament amb les taules definides en Fracslan)

A partir d'aquest moment, per executar el projecte només cal fer click dret en Fracserver (situat en el Package explorer de eclipse) i fer click en Run As>1: Pydev: Django.

A.2. Manual del desarrollador

En aquest apartat es mostren algunes guies sobre què s'ha de tenir en compte per poder modificar alguns aspectes importants de la plataforma, per llegir aquesta guia es pressuposa que el desenvolupador ja disposa de coneixements de Unity3D, aquesta guia servirà de cara a orientar a futurs desenvolupadors vulguin ampliar el joc.

A.2.1. Modificar les configuracions del joc

Per poder modificar les configuracions que rebrà el joc, és a dir els paràmetres que s'enviaran mitjançant la interfície de comunicacions s'han de tenir el compte els següents punts:

- 1- Accedir com a administrador a la plataforma de Fracslan.
- 2- Accedir al apartat "Quest Configurator".
- 3- Esborrar o afegir nous widgets (paràmetres configurables).
- 4- En cas d'afegir nous widgets recordar el "In Game ID" ja que serà usat en el joc.
- 5- Accedir al codi del joc.
- 6- Accedir al codi DataManager.cs i modificar el mètode "ParseGameConfiguration" i la classe ConfiguredQuest (ja que segons el que s'hagi fet es precisaran de més atributs, menys, canviar els tipus algun...).

Un cop fet això el joc ja serà capaç de rebre i emmagatzemar noves configuracions.

A.2.2. Crear una nova Quest

Per afegir una nova quest al joc s'han de realitzar els següents passos:

- 1- Decidir en quina escena es portarà a terme la quest.
- 2- Crear un nou script C# que heredi de Quest i associar-lo a un objecte de l'escena, no té perquè ser l'objecte visible on es deixaran anar les fraccions tot i que en alguns casos pot tenir avantatges (com calcular la distància de l'objecte al jugador només accedint al transform del propi objecte).
- 3- Implementar els mètodes abstractes (s'aconsella mirar altres quests per entendre el funcionament del sistema).
- 4- Un cop acabada d'implementar la quest es necessitarà d'un objecte que rebí la fracció, per realitzar això s'ha d'associar el codi "Interactable" a l'objecte en qüestió i seleccionar SOLVE_EVENT com a "Selected Game Event" també cal associar un "Collider" a l'objecte perquè pugui rebre les fraccions.
- 5- Per tal de que Lord Barus pugui donar la quest a un jugador informant-lo dels detalls d'aquesta és necessita registrar-la com a tal. Per fer això accedir al la classe QuestDetailArchive per tal de crear una nova instància de QuestInformation per la quest i guardar-la juntament amb les altres tal i com es veu en mètode Init del QuestDetailArchive (usant alguna altre quest registrada com exemple es dedueix fàcilment com registrar una quest).

A.2.3. Creació d'una nova escena

Per tal de crear una nova escena on es porti a terme el joc de Fracslan es precisa que aquesta contingui varies coses:

- 1- Un portal per on poder accedir a l'escena tant en l'escena nova com en una altre des de la qual es vol accedir a la nova.
- 2- Una spawn position (associat al portal, veure exemples en altres escenes)
- 3- El prefab "Camera" és la que seguirà l'heroi.
- 4- Una càmera per el minimapa.
- 5- El prefab _DESTROYABLE_SCRIPTS. Utilitzat per col·locar l'heroi quan canvia d'escena.
- 6- El game object HeroSpawnPoint (usat per reviuere l'heroi).

A.2.4. Modificació IP on el joc realitza les peticions:

Per modificar la IP del servidor on el joc farà les peticions només s'ha d'accedir al codi Constants.cs i modificar la variable API_PRODUCTION_IP. Com es pot deduir al observar algunes de les constants, Fracslan disposa de dues IP on fer peticions, una és la API_PRODUCTION_IP i l'altre és API_DEVELOPMENT_IP (que correspon a la màquina local). Per fer que el joc enviï les peticions a la IP local només cal canviar el valor de la variable DEVELOPMENT a "True" d'aquesta manera sense alterar la IP de producció el joc realitzarà les peticions al "localhost", això resulta útil quan es vol treballar amb el servidor situat en la mateixa màquina on s'està executant el joc.

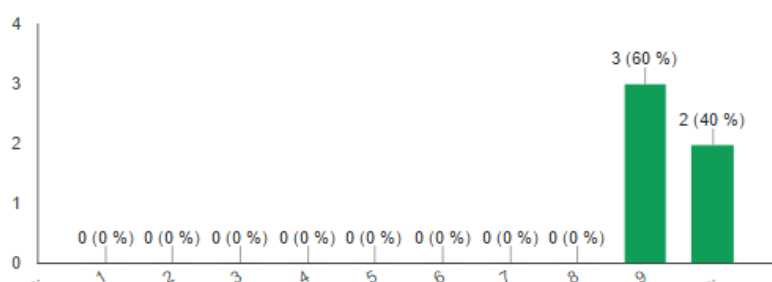
Apèndix B: Resultats tests d'usuari

En aquest capítol s'inclouen els resultats complets dels testos realitzats amb usuaris, tant per la plataforma del professor com per el joc.

B.1. Resultats tests plataforma professor

En les figures 87, 88 i 89 es poden veure els resultats de les respostes obtingudes en els tests d'usuari realitzats per el panell d'administració del professor:

En general, estic satisfet amb com de fàcil ha estat utilitzar el sistema
(5 respuestas)



Ha resultat fàcil seguir el tutorial per configurar un joc (5 respuestas)

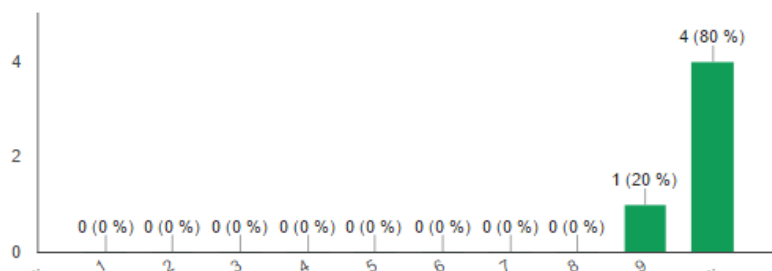
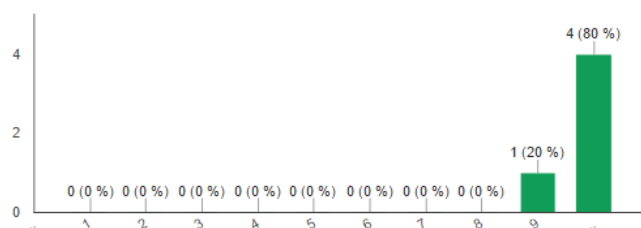


Figura 87: Resultats de les preguntes 1 i 2 dels tests d'usuari de la plataforma del professor.

Crec Fracsland (Joc i Servidor) és útil com a eina d'aprenentatge de fraccions
(5 respuestas)



Recomanaria aquest sistema a una altre persona (5 respuestas)

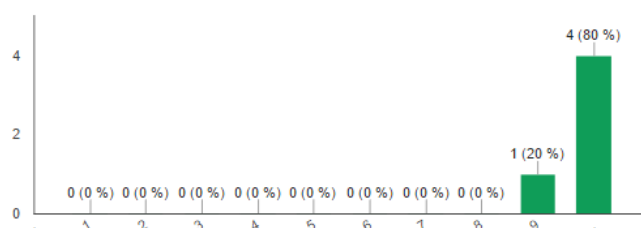


Figura 88: Resultats de les preguntes 3 i 4 dels tests d'usuari de la plataforma del professor.

Crec que s'ha de millorar: (4 respuestas)

- L'últim pas del tutorial de com afegir alumnes, no surt botó Understood (no cal que surti, simplement l'esperava).
- En el panel, es pot fer scroll perquè apareix una línia en blanc abaix. Potser seria millor si fos simplement la pàgina sencera sense scroll.
- Els temps de boto d'understood surten massa lents.
- El detalls de sessio m'ha faltat alguna indicacio de que es podia entrar.
- S'hauria de decrementar el temps que es dona per llegir la frase del tutorial o bé posar el butó de seguir endavant, per si ja se sap com fer el pas que s'indica
- Al nivel visual, los mensajes de abajo (de color gris) eran un poco grandes y me sorprendian cuando aparecian.

El que més m'ha agradat és: (4 respuestas)

- La interfície es intuïtiva i el procés molt guiat i senzill de seguir
- No pantalles excessives i ràpid de configurar.
- Afegir estudiants i coses similars en la mateixa pantalla.
- Que en 5 min reals es disposa d'un joc personalitzat a cada estudiant
- Que todo va muy rápido. No tienes a penas que esperar para que cargue la pagina siguiente. Además, todo es bastante intuitivo !

Figura 89: Respostes a les preguntes 5 i 6 dels tests d'usuari de la plataforma del professor.

B.2. Resultats tests joc

En les figures 90, 91, 92, 93, 94 i 95 es poden veure el resultats de les respostes obtingudes en els tests d'usuari realitzats per el joc:

Edat (7 respuestas)

10
10
11
11
12
12
23

Sexe (7 respuestas)

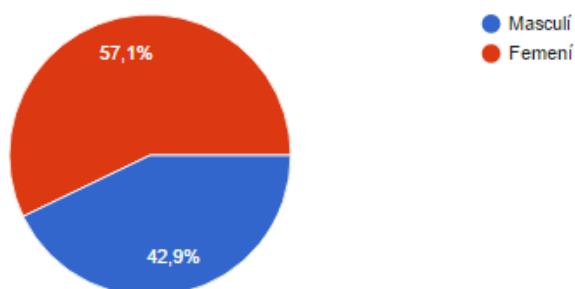
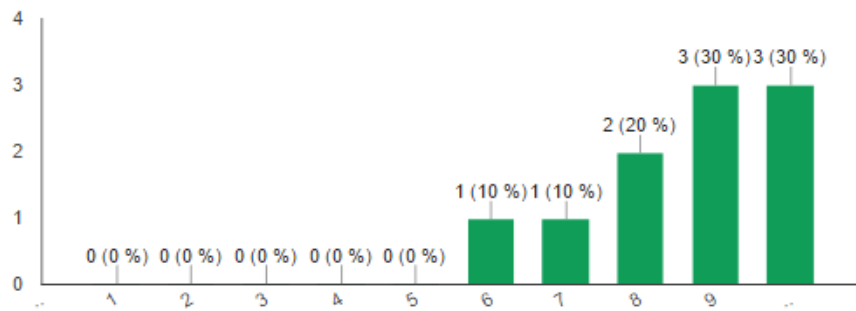


Figura 90: Resultats de les qüestions 1 i 2 dels tests d'usuari del joc.

En general, estic satisfet amb com de fàcil ha estat utilitzar el sistema
(10 respuestas)



Ha resultat senzill saber que fer en cada quest (10 respuestas)

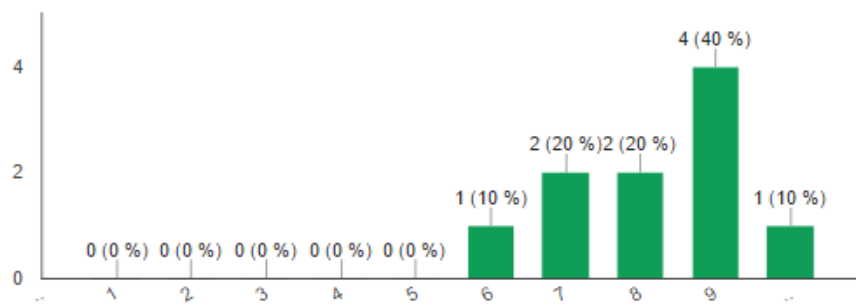
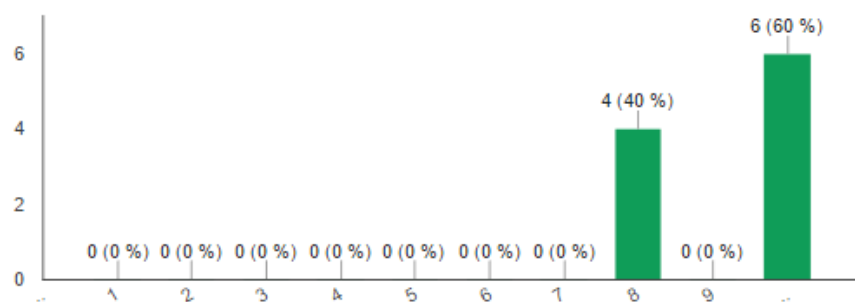


Figura 91: Resultats de les qüestions 3 i 4 dels tests d'usuari del joc.

M'he divertit jugant (10 respuestas)



Crec que és útil per aprendre sobre fraccions (10 respuestas)

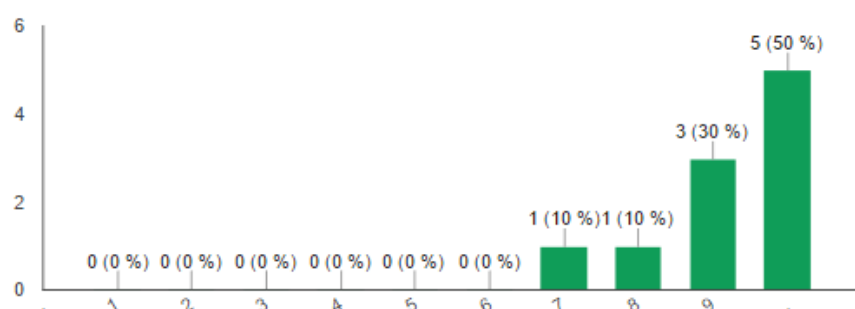


Figura 92: Resultats de les qüestions 5 i 6 dels tests d'usuari del joc.

Recomanaria aquest joc a una altre persona (10 respuestas)

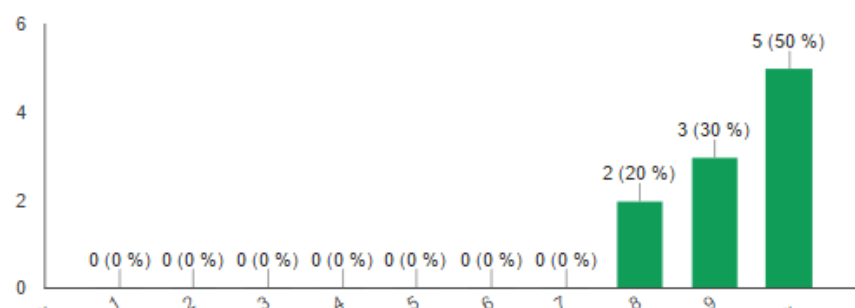


Figura 93: Resultats de la resposta a la qüestió 7 dels tests d'usuari del joc.

Crec que aquest joc ha de millorar: (9 respuestas)

Potser indicar com obtenir la destal, ja que es requereix per talar fusta. Si bé el pont diu que fa falta fusta, pot no quedar clar com obtenir-la (tot i que si has passat pels arbres, que no era el meu cas, es veu fàcilment).

El minimapa està mal orientat i no correspon amb el moviments reals en el joc.

Com a afegit, estaria bé poder cancel·lar missions, tot i que tampoc es un impediment pel normal funcionament del joc.

La botiga es podria tancar automàticament quan estàs més lluny d'una distància X. O incloure un botó per tancar-la.

M'ha costat entendre si el capità volia la fracció $3/4$ o $1/4$.

Al clicar a un enemic estaria bé que ataqués més d'un cop.

No estava del tot clar que s'havia de comprar una destal.

S'han de millorar petits detalls com que al sortir de les àrees no recuperis la vida perduda o que no et treguin les fraccions un cop acabat el quest. Gràficament només s'hauria d'intentar trobar alguna manera que la transició entre aigua i terra no fós tant brusca.

En el puente, a veces al personaje le cuesta seguir el ratón. Además, no es evidente como empezar la misión del puente. Parece que tienes que pulsar sobre el y tienes que pulsar la tecla "e".

els lags i transpassar alguna casa

que et puguis moure amb les fletxes

si, porque en la prueba de la isla no encontraba fracciones para matar al ogro

les indicacions a l'hora de realitzar les proves (era muy raro:(no se encontraban las fracciones en la isla helada y en el puente debería haber indicado mejor como conseguir madera)

que no se cogian muy bien las cosas como plantas...

Figura 94 : Respostes dels usuaris a l'apartat 8 dels tests d'usuari del joc.

El que més m'ha agradat d'aquest joc és: (10 respuestas)

Crec que pot ser útil per aprendre les fraccions, a la par que els gràfics són sencills però sofisticats, permeten una agradable experiència. El joc es ràpid en el sentit que veus resultats d'hora i queda clar si són positius o negatius.

Els jocs en si demanaven les fraccions de manera divertida.
Els mapes eran variats i macos.
El tweening sempre queda bé.

M'ha agradat que s'hagin tingut en compte animacions que altres jocs no inclouen, com la caiguda dels arbres o el creixement de les plantes.

Todo alrededor te vuelve a tu infancia : la música, los dibujos, el tipo de misión, ... Da ganas de jugar más !
Además, es agradable poder elegir el tipo de misión !

matar els enemics

es pot comprar i no només es fan les missions i pots matar i millorar les armes

que pots comprar noves destrals i armadures

es original y muy divertido, me encantó el huerto (está muy currado)

como crecían las setitas en el huerto:)

cuando tenias que ir buscando cosas

Figura 95: Respostes dels usuaris a l'apartat 9 dels tests d'usuari del joc.