

Indice

1	Introduzione	5
1.1	Piattaforme	6
1.2	Processo di sviluppo	7
1.3	Multiplayer	8
2	Design	15
2.1	Titoli sul mercato	15
2.1.1	Crash Team Racing	15
2.1.2	Rocket League	18
2.1.3	Geometry Wars 3: Dimensions Evolved	20
2.2	OrbTail	23
2.2.1	Modalità di gioco	24
2.2.2	Veicoli	26
2.2.3	Level design	28
2.2.4	Potenziamenti	29
2.2.5	Interazione utente	32
2.2.6	Interfaccia utente	32
2.2.7	Flusso di gioco	33
3	Direzione artistica	35
4	Sviluppo	36

Elenco delle figure

1	Overcooked - Party game a visuale condivisa	9
2	Heroes of Might & Magic - Multiplayer locale a turni	10
3	Mario Karts - Multiplayer in split-screen	10
4	Battlerite - Multiplayer online battle arena.	11
5	EVE Online - Massive multiplayer online game	12
6	PlayerUnknown's Battleground - Battle royale	12
7	Battlefield 4 - Multiplayer con interoperabilità pc\mobile . . .	14
8	Crash Team Racing	16
9	Crash Team Racing - Modalità «battaglia»	17
10	Rocket League - Personalizzazione del veicolo	18
11	Rocket League	19
12	Geometry Wars 3: Dimensions Evolved - Arena sferica	20
13	Geometry Wars 3: Dimensions Evolved - Arena	21
14	Flusso di gioco	33

Ringraziamenti

Sommario

In questo elaborato di tesi si descrive il processo di sviluppo di Orbtail, un gioco multiplayer multiplatforma che coniuga elementi *racing* ed *arena* e diverse modalità di gioco in un gameplay rapido e competitivo. Il prodotto, nato nel 2013 come progetto per il corso di Videogame Design and Programming, é stato rielaborato ed esteso, andando ad aggiornare l'intero comparto tecnico, il design originale e la direzione artistica. Il pregio principale di Orbtail risiede nella sua capacità di poter essere giocato da piú giocatori contemporaneamente, online o offline, in una qualsiasi combinazione di piattaforme e/o numero di giocatori locali. Questo documento ne descrive il processo produttivo, a partire dal *concept* fino alle varie scelte tecniche, facendo numerosi paralleli con la versione originale al fine di individuare carenze e spunti per migliorie.

Questa tesi é strutturata in capitoli.

Nel primo capitolo viene dato un contesto al progetto, fornendo una panoramica generale ed andando ad analizzare i requisiti dello sviluppo multiplatforma e di varie modalità multigiocatore.

Nel secondo capitolo vengono delineate le meccaniche del gioco e definiti i requisiti di *level design*. Vengono inoltre analizzati alcuni titoli concorrenti al fine di individuare meccaniche simili ed analizzarne i vari aspetti e criticità.

Nel capitolo successivo verrà inquadrata la direzione artistica del progetto usando diversi titoli esistenti come riferimento. Viene inoltre analizzata la produzione dei vari *asset* utilizzati all'interno del gioco considerando i requisiti tecnici e di design.

Il quarto capitolo verte sulle scelte tecnologiche, a partire dalla scelta dell'engine, delle piattaforme e dell'architettura del gioco. Ogni scelta verrà contestualizzata rispetto a varie alternative e opportunamente giustificata, considerando necessariamente i requisiti di design.

Il capitolo finale é dedicato alle conclusioni nonché ad eventuali sviluppi futuri.

1 Introduzione

I videogame rappresentano una delle forme di intrattenimento moderne più affermate e diffuse. A differenza di quanto avviene con i mezzi classici quali *film* e *libri*, l'utente ha pieno controllo dell'esperienza: alcuni giochi richiedono concentrazione, altri una buona dose di coordinazione e reattività, altri ancora pazienza e pianificazione. Sebbene vi siano videogiochi che si concentrano principalmente sulla componente *single-player*, i giochi, intesi anche nel senso classico del termine, nascono come mezzo di *condivisione* dell'esperienza con altri partecipanti.

In questa tesi si descrive il processo di sviluppo di *Orbtail*, un videogioco *multiplayer* con elementi *racing* disponibile per dispositivi mobile e desktop quali Android, iOS, Windows e OSX. Il gioco prevede tre modalità di gioco differenti e consente fino a quattro giocatori di sfidarsi online in modalità *cross-platform play*¹ su una delle tre arene disponibili. Vi è inoltre la possibilità di condividere uno stesso dispositivo desktop tra due o più giocatori locali mediante un'apposita modalità *splitscreen*².

La necessità di supportare più piattaforme, impedendo che ciascuna di esse potesse risultare avvantaggiata rispetto alle altre durante le sessioni *cross-platform play*, ha portato allo sviluppo di un *gameplay*³ essenziale ed immediato. La durata limitata delle sessioni, unita ad una curva d'apprendimento semplice, rende il prodotto particolarmente adatto ad un pubblico di *casual gamer*⁴.

¹Il termine *cross-platform play* identifica una modalità di gioco online in cui giocatori possono giocare tra loro indipendentemente dalla piattaforma utilizzata.

²Il termine *splitscreen* identifica una modalità di gioco in cui lo schermo viene suddiviso in più quadranti in modo da consentire a più giocatori di usare uno stesso dispositivo contemporaneamente.

³Il *gameplay* (in italiano traducibile come “*esperienza utente*”) comprende gli elementi di gioco quali storia, regole, obiettivi, progressioni, interazione utente, ecc.

⁴Si definisce *casual gamer* un giocatore saltuario, non particolarmente interessato alla cultura dei videogiochi e, solitamente, alla tecnologia in generale.

1.1 Piattaforme

L'esperienza videoludica ha il grande vantaggio di poter essere studiata per soddisfare utenze dai gusti profondamente diversi su un vasto numero di piattaforme. Ad oggi esistono decine di tipologie di dispositivi di intrattenimento differenziati per costo, performance e modalità d'interazione.

Le piattaforme *mobile*, pur essendo nate solo di recente, si sono subito affermate come una delle principali piattaforme da gaming. Il successo senza precedenti di *smartphone* e *tablet* ha consentito all'industria mobile di generare un volume d'affari pari a quello di *pc* e *console* combinati [1].

Pur non godendo delle stesse performance e manifestando un ciclo di vita più breve rispetto alle controparti classiche, i dispositivi mobile godono di una più vasta diffusione e si rivolgono ad un pubblico decisamente più ampio.

L'interazione primaria affidata all'*input touch* è affiancata da un grande numero di sensori accessori quali *GPS*, *accelerometri*, *giroscopi*, *bussole* e *videocamere*. Recenti sviluppi tecnologici quali *augmented reality*⁵ e *virtual reality*⁶, unito all'uso sapiente di queste nuove modalità d'interazione, ha permesso la nascita di nuovi paradigmi di *gameplay* rimasti finora inesplorati.

I *pc* e le *console* sono le piattaforme storiche su cui sono nati e diffusi i videogame. Accomunate da paradigmi d'interazione e performance paragonabili, queste due famiglie di dispositivi sono nate per scopi diversi. Laddove le *console* sono pensate per essere un dispositivo d'intrattenimento *dedicato*, che fa della *facilità d'uso* il suo cavallo di battaglia, i *pc* hanno una natura più generica e garantiscono performance e flessibilità maggiori al costo di una richiesta più elevata di competenza da parte dell'utenza.

Il ciclo di vita e il costo delle due piattaforme è inoltre molto diverso. Le *console* sono dei *sistemi embedded* precostruiti e limitatamente aggiornabili; il loro ciclo di vita è generalmente superiore al quinquennio e sono caratterizzati da un costo relativamente basso. I *pc*, d'altro canto, possono essere assemblati scegliendo i componenti che meglio si adattano alle esigenze dell'utente ed aggiornati quando ritenuto necessario. Il costo di un *pc* è generalmente molto più alto rispetto a quello di una console, anche a parità di specifiche tecniche.

⁵La *augmented reality* (in italiano «realtà aumentata»), consiste nell'arricchire il mondo circostante attraverso contenuti di tipo visivo, aptico o uditivo generati da un elaboratore.

⁶La *virtual reality* (in italiano «realtà virtuale») consiste nel simulare un ambiente tridimensionale tramite un elaboratore e lasciare che l'utente vi interagisca attraverso periferiche di input specializzate quali visori, cuffie e controller.

L'ampio parco di periferiche di input supportate, dalle piú comuni quali tastiere, mouse e gamepad a quelle piú specializzate quali volanti, pedaliera e *HOTAS* ⁷, rendono queste piattaforme adatte a qualsiasi tipologia di gioco ed interazione utente.

1.2 Processo di sviluppo

Il processo di sviluppo di un videogioco cambia radicalmente in funzione della piattaforma e dell'utenza a cui viene destinato il prodotto. Un design di successo deve essere in grado di sfruttare le peculiarit  di ciascun dispositivo e considerarne le limitazioni tecniche, senza pregiudicare l'esperienza utente.

Lo sviluppo risulta solitamente tanto pi  avvantaggiato quanto minori sono le limitazioni delle piattaforme o variet  di specifiche tra dispositivi. La presenza di dispositivi *identici* tra *console* appartenenti ad uno stesso ecosistema garantisce un'elevata consistenza dell'esperienza utente e consente interventi di ottimizzazione ad una granularit  molto fine. Laddove questo processo   solitamente molto *efficiente* su console, lo stesso non pu  essere detto per piattaforme *mobile* e *desktop* per via dell'elevata variet  di specifiche tecniche o combinazioni di componenti. I *pc* risultano avvantaggiati per via delle elevate performance e assenza di grosse limitazioni e ci  consente loro di mantenere un'elevata fedelt  e fruibilit  del contenuto. Lo sviluppo su dispositivi quali *smartphone* e *tablet*, d'altro canto, deve scontrarsi con la presenza di dispositivi con capacit  profondamente diverse e comparti tecnici non sempre in equilibrio tra loro (non   raro assistere a dispositivi che associano elevate risoluzioni a performance mediocri). In questo caso   richiesto uno sforzo maggiore affin  l'applicazione scali in funzione del dispositivo per garantire una buona esperienza utente.

Sebbene esistono titoli sviluppati in *esclusiva* per alcune piattaforme, la necessit  di aumentare il bacino d'utenza, e di conseguenza i ricavi, solitamente richiede che un prodotto venga distribuito su pi  ecosistemi. Un tempo processo lungo ed oneroso per via delle marcate differenze tra le architetture e carenza di *tool*, ad oggi lo sviluppo *cross-platform* risulta molto avvantaggiato. Il rilascio di *engine third-party*, unito alla convergenza delle piattaforme verso architetture simili, permette agli sviluppatori di lavorare

⁷L'*HOTAS* (acronimo di «*hands on throttle and stick*»), usato solitamente per giocare a simulatori di volo, consiste di un joystick a 4 o pi  assi e una leva d'accelerazione.

ad un livello di astrazione piú elevato evitando di creare supporti di basso livello specifici per dispositivo.

Laddove sviluppare un videogioco per piattaforme simili quali potrebbero essere *pc* e *console* oppure *smartphone* e *tablet*, costituisce piú un problema implementativo che non a livello di *gameplay*, sviluppare per piattaforme profondamente diverse richiede importanti considerazioni a livello di *design* e produzione *asset*. In primo luogo il differente grado di apprezzamento dell'utenza delle varie piattaforme può precludere il successo a certe tipologie di gioco di nicchia (quali potrebbero essere simulatori o strategici) e favorire i design che si rivolgono un pubblico piú *mainstream*. Le differenti modalità d'interazione potrebbero inoltre richiedere la rivisitazione dell'interfaccia grafica e l'eventuale eliminazione degli input disponibili solo su certe tipologie di dispositivi. Il comparto tecnico deve inoltre consentire al prodotto di poter scalare in funzione delle performance a disposizione. Possibili interventi consistono nella riduzione del dettaglio delle *texture* o della complessità poligonale, riduzione o rimozione dell'effettistica, limitazione degli oggetti a schermo. Ove ciò non fosse possibile o insufficiente potrebbe essere necessario rivisitare il *gameplay* attraverso la riduzione di giocatori o avversari, semplificazione della *IA* o rimozione di feature particolarmente onerose.

1.3 Multiplayer

L'interazione sociale come mezzo per aumentare il coinvolgimento videoludico é il motivo principale per cui i videogiochi multiplayer hanno da sempre riscosso un grande successo. Sebbene esistono infinite variazioni sul tema, le modalità di gioco multiplayer possono essere classificate in due famiglie: *cooperative* e *competitive*. Alla prima categoria appartengono quelle modalità in cui due o piú giocatori *collaborano* tra loro al fine di raggiungere un obiettivo comune. In queste modalità l'elemento di sfida é rappresentato dal gioco stesso e governato da *intelligenze artificiali* piú o meno sofisticate. Nelle modalità *competitive* l'elemento di sfida é invece rappresentato dai giocatori stessi: gli utenti sono portati a confrontarsi gli uni con gli altri al fine di raggiungere un obiettivo impedendo al contempo che gli altri giocatori possano fare altrettanto. Il livello di sfida offerto da alcuni titoli é tale per cui, pur di eccellere, alcuni giocatori hanno deciso di farne una carriera, sottoponendosi costantemente a veri e propri allenamenti. Questo fenomeno ha di recente portato alla nascita di competizioni a livello agonistico, organizzate e regolate da entità terze, in cui i partecipanti non sono semplici giocatori ma



Figura 1: Overcooked - Party game a visuale condivisa

veri e propri atleti. Queste competizioni prendono il nome di *e-sports* (sport elettronici).

Le modalità multiplayer hanno accompagnato lo sviluppo dei videogiochi fin dagli albori, in un'epoca in cui la diffusione di Internet era molto limitata. Le realizzazioni più semplici consistono nello sfruttare uno stesso dispositivo col quale tutti i giocatori possono interagirvi. La presenza fisica di tutti i partecipanti coinvolti favorisce un'interazione sociale più immediata e rende queste modalità di gioco particolarmente adatte ai *party-game*⁸. Le implementazioni più semplici consistono nell'utilizzare una visuale condivisa tra tutti i partecipanti (Fig.1) oppure, laddove l'interazione in contemporanea non fosse necessaria, sfruttare la meccanica dei *turni* affinché ciascun giocatore goda di un punto di vista unico sul mondo di gioco (Fig.2). Un'implementazione più sofisticata, nota col termine *split-screen*, consiste nel suddividere lo schermo in quadranti, solitamente da due a quattro, e mostrare in ciascuno di essi il punto di vista di uno dei giocatori in maniera indipendente (Fig.3). L'elevato impatto sulle performance, tale da rendere necessarie ottimizzazioni particolarmente aggressive, unita al ridotto bacino d'utenza cui queste modalità si rivolgono, ha di recente portato ad un calo di titoli che offrono questo tipo di esperienza.

⁸I *party-game* sono l'equivalente videoludico dei giochi di società.



Figura 2: Heroes of Might & Magic - Multiplayer locale a turni



Figura 3: Mario Karts - Multiplayer in split-screen



Figura 4: Battlerite - Multiplayer online battle arena.

La rapida diffusione di Internet, unita ai suoi sviluppi degli ultimi decenni, ha consentito la nascita di nuove modalità multigiocatore *online*. Rispetto all'approccio classico, un multiplayer online permette agli utenti di usare il proprio dispositivo per accedere a sessioni di gioco con un numero di partecipanti che varia tra la decina e il migliaio, indipendentemente dalla distanza fisica che li separa. Il rinnovato successo dei giochi multiplayer ha consentito la nascita di nuovi generi, dai *MOBA* in cui due team dal numero ristretto di giocatori si sfidano all'interno di un'arena (Fig.4) , agli *MMO* i cui mondi persistenti vantano migliaia di giocatori attivi contemporaneamente (Fig.5) , passando per i più recenti *Battle Royale* caratterizzati da centinaia di giocatori che lottano per la sopravvivenza (Fig.6).

Le implementazioni più comuni sono riducibili a due diverse *architetture*, in funzione del ruolo che i diversi dispositivi hanno all'interno della sessione di gioco. Nell'architettura *client-server* i dispositivi dei giocatori, detti *client*, sono connessi ad un dispositivo centrale, detto *server*, il cui ruolo consiste nel coordinare tutti i partecipanti, gestire l'evoluzione della partita, propagare lo stato condiviso di gioco ed eventualmente validare le azioni dei singoli giocatori per evitare *cheat* ⁹. I *client* comunicano esclusi-

⁹Il *cheat*, in italiano «imbrogliare», rappresenta una qualsiasi tecnica atta a sovvertire le regole di gioco affinché un giocatore ne ottenga un beneficio immeritato.



Figura 5: EVE Online - Massive multiplayer online game



Figura 6: PlayerUnknown's Battleground - Battle royale

vamente col *server* limitandosi a dichiarare le azioni effettuate dall'utente. Il ruolo di *server* può essere ricoperto sia da una macchina esterna che non prende parte alla partita detta *server dedicato*, oppure direttamente da uno dei dispositivi dei giocatori, detto *host*. L'approccio con *server dedicato* permette di sfruttare macchine remote caratterizzate dalla elevate performance al fine di gestire efficientemente un numero anche elevato di giocatori. Il secondo di questi, invece, consente di evitare i costi associati alla gestione o noleggio del server al costo di una minore scalabilità. L'esperienza multigiocatore con questo tipo di approccio è inoltre soggetta a fattori difficilmente prevedibili quali prestazioni del dispositivo che fa da *host*, qualità della connessione verso i *client* e la possibilità di disconnessione del *server* durante una partita che potrebbe richiedere il *trasferimento* del ruolo di server ad uno degli altri dispositivi (processo noto col termine *host migration*).

Nelle architetture *peer-to-peer*, a differenza di quanto avviene con l'approccio *client-server*, lo stato di gioco è condiviso su tutti i dispositivi e gestito in maniera distribuita. Sebbene questa architettura impedisce che il carico computazionale si concentri su un unico dispositivo, evitando inoltre i costi associati a macchine server esterne, la condivisione dell'*autorità* sullo stato di gioco favorisce il proliferare del fenomeno del *cheating*, oltre che a complicare la sincronizzazione tra i vari dispositivi e la risoluzione di eventuali discrepanze nello stato di gioco.

Per far fronte al numero crescente di giocatori all'interno di ciascuna sessione di gioco, le implementazioni più moderne si basano su un approccio *misto client-server* in cui parte della computazione viene delegata ai vari *client* che, pertanto, possiedono un certo grado di autorità sulla partita.

Sebbene esistano numerosi paradigmi consolidati per la gestione dei giochi multiplayer, la quasi totalità delle implementazioni *segmentano* il bacino d'utenza in funzione della piattaforma d'appartenenza. Questo approccio semplifica enormemente la gestione dei servizi online accessori, specifici per piattaforma e molto diversi tra loro, evitando i costi associati alla gestione dell'*interoperabilità* tra di essi, tuttavia impedisce che giocatori su piattaforme diverse possano giocare gli uni contro gli altri. Esistono alcuni esempi di giochi multipiattaforma che consentono a giocatori su *smartphone* e *tablet* di interagire ed influenzare partite in corso su altre piattaforme (quali *pc* e *console*), tuttavia essi rappresentano una nicchia ristretta che non ha mai riscosso un vero e proprio successo per via dei ruoli particolarmente asimmetrici e gameplay profondamente diversi tra i vari dispositivi (Fig.7).



(a) PC - First person shooter. Il *soldati* prendono controllo di punti strategici facendo guadagnare risorse al *comandante*.



(b) Mobile - Strategico. Il *comandante* gestisce la conquista dispiegando soldati e risorse, lanciando attacchi missilistici e proteggendo i propri soldati dagli avversari.

Figura 7: Battlefield 4 - Multiplayer con interoperabilità pc\mobile

2 Design

Questo capitolo é dedicato al design di OrbTail, partendo dai titoli esistenti che hanno ispirato il gioco e proseguendo con le meccaniche di base, le modalità di gioco, i livelli e la descrizione dettagliata di tutti gli elementi di gioco. I contenuti si concentreranno principalmente sugli aspetti *concettuali* del gioco, in maniera agnostica rispetto alla direzione artistica e dalle soluzioni tecnologiche che verranno adottate (sebbene ne verranno considerate le eventuali limitazioni).

2.1 Titoli sul mercato

In questo capitolo vengono presentati alcuni titoli esistenti al fine di analizzarne le meccaniche di base e trovare spunti interessanti da integrare all'interno del *concept* di gioco. I titoli sono stati selezionati tra diverse piattaforme in modo da individuare un sottoinsieme di caratteristiche desiderate e facilmente adattabili alla grande diversità di dispositivi per cui verrà sviluppato OrbTail. La data di pubblicazione dei titoli scelti copre inoltre un orizzonte temporale molto vasto: l'intento é quello di coniugare tra loro caratteristiche dei giochi recenti con vecchi paradigmi di gameplay da rivisitare in chiave moderna.

2.1.1 Crash Team Racing

Crash Team Racing, sviluppato da *Naughty Dogs* e pubblicato da *Sony Computer Entertainment* nel 1999 per *PlayStation*, é un videogioco di guida tratto dalla fortunata serie platform *Crash Bandicoot*. Il gioco ruota attorno alle vicende di un team di piloti di *kart* intenti a difendere la Terra da un alieno, *Nitros Oxide*, il quale vuole trasformare il pianeta in un parcheggio e rendere schiavi i suoi abitanti.

Nel gioco l'utente controlla uno tra quindici personaggi disponibili, contraddistinti ciascuno da un proprio kart dalle caratteristiche uniche. I veicoli sono in grado di accelerare, sterzare, frenare e *saltare*; la meccanica del *power slide* garantisce inoltre un'accelerazione temporanea durante i *drift*. I veicoli sono caratterizzati da tre parametri fondamentali, *velocità*, *accelerazione* e *manovrabilità*, i quali influenzano lo stile di guida e la difficoltà del personaggio.



Figura 8: Crash Team Racing

All'interno di ogni circuito sono distribuite delle *casse* speciali che, raccolte, garantiscono al giocatore un potenziamento casuale, tanto più potente quanto più svantaggiata la sua posizione nella gara corrente. Questi oggetti collezionabili comprendono un ricco assortimento di armi usate per intralciare gli avversari, barriere difensive e bonus temporanei alla velocità del veicolo. Speciali frutti «wumpa» consentono ai veicoli di procedere più velocemente e migliorare i suddetti potenziamenti.

Il gioco offre cinque modalità di gioco, di cui tre a giocatore singolo (Fig.8) e due multigiocatore locale in splitscreen. Nella modalità «avventura» il giocatore seleziona uno dei personaggi a disposizione, gareggiando su sedici circuiti diversi, al fine di collezionare *trofei*, *reliquie* ed altri oggetti necessari per poter proseguire con la trama. I circuiti sono suddivisi in diversi *mondi*, ciascuno contenente quattro tracciati, al termine dei quali viene richiesto di fronteggiare un *boss* in una gara testa a testa. All'interno della modalità avventura sono inoltre presenti diverse *sotto-modalità* che cambiano leggermente le regole delle varie sfide: collezionare oggetti nascosti all'interno del circuito entro la fine della gara, vincere più gare in successione o terminare la gara entro un tempo limite usando un singolo potenziamento in grado di congelare il tempo per pochi secondi. Nella modalità «sfida a tempo», il giocatore corre da solo al fine di registrare il miglior tempo su tutti i tracciati disponibili. Questa modalità è caratterizzata dall'assenza di avversari e



Figura 9: Crash Team Racing - Modalità «battaglia»

potenziamenti di alcun genere. Nella modalità «sala giochi» il giocatore può gareggiare su circuito a scelta o organizzare *campionati* formati da quattro tracciati. In questa modalità è prevista la presenza di otto partecipanti e l'intero arsenale di armi e potenziamenti. Quest'ultima modalità è anche fruibile in formato multiplayer locale con numero di giocatori compreso tra due e quattro. L'ultima modalità «battaglia» stravolge le regole del gioco: i partecipanti si sfidano all'interno di un'arena chiusa collezionando armi e potenziamenti al fine di colpire e danneggiare gli avversari (Fig.9). Esistono diverse condizioni di vittoria configurabili: una richiede il raggiungimento di un certo numero di punti guadagnabili colpendo gli avversari, un'altra ha un tempo limite scaduto il quale il giocatore col punteggio più alto viene proclamato vincitore, nell'ultima ogni giocatore inizia la sfida con un numero predefinito di vite che vengono decrementate ogni volta che il kart viene colpito da un'arma. Quando il numero di vite scende a zero il giocatore viene eliminato: l'ultimo rimasto in partita è proclamato vincitore. Questa modalità è caratterizzata dall'assenza di partecipanti governati dall'*intelligenza artificiale* ed è pertanto preclusa dall'esperienza *singleplayer*.



Figura 10: Rocket League - Personalizzazione del veicolo

2.1.2 Rocket League

Rocket League é un gioco multiplatforma sviluppato e pubblicato da *Psyonix* nel 2015 per *Windows* e *PlayStation 4* e successivamente rilasciato per *OSX*, *Linux* e *Nintendo Switch*. Il gioco combina due generi, quello *sportivo* e quello *racing*, in un gameplay inedito in cui due team formati da potenti veicoli si sfidano ad una partita di *calcio* all'interno di uno *stadio*.

Ogni giocatore ha a disposizione decine di veicoli differenti, identici nelle prestazioni e modelli di guida, ma riccamente personalizzabili tramite *livree*, *decalcomanie*, *pneumatici*, *razzi* e molto altro ancora (Fig.10). I veicoli sono inoltre in grado di saltare e roteare in volo al fine di colpire la palla, effettuare salvataggi o portarsi in vantaggio rispetto agli avversari. Un potente razzo posteriore consente di aumentare drasticamente la propria velocità, planare o spiccare il volo per brevi periodi. É inoltre possibile usare il proprio veicolo come un ariete contro i veicoli avversari in modo da farli esplodere e rimuoverli dalla partita per pochi secondi.

Ogni *stadio* é caratterizzato da uno stile diverso. La presenza di zone speciali all'interno di essi consente ai veicoli di ottenere un'accelerazione aggiuntiva quando attraversate; particolari oggetti distribuiti nello stadio consentono al veicolo di recuperare parte dell'energia necessaria per azionare i razzi posteriori.



Figura 11: Rocket League

Il gioco offre diverse modalità a *giocatore singolo* e *multigiocatore locale* tramite *splitscreen*, *online* o *misto*. Le modalità *online* sono inoltre fruibili nel formato *cross-platform play* tra *Windows* e *PlayStation 4* oppure tra *Windows*, *XBoxOne* e *Nintendo Switch*. La modalità di gioco principale «Carlcio» (gioco di parole originato dalle parole «car» e «calcio») prevede la sfida tra due team di dimensione variabile tra uno e quattro giocatori (Fig.11) . I partecipanti usano i propri veicoli per colpire una grossa palla al fine di spingerla nella porta avversaria ed ottenere punti. Allo scadere del tempo il team col maggior numero di punti é dichiarato vincitore. In caso di parità il primo team a segnare oltre lo scadere del tempo vince. Ogni giocatore ha inoltre un proprio punteggio personale che può essere incrementato tirando la palla in porta, segnando, effettuando salvataggi, distruggendo i veicoli avversari e così via. Aggiornamenti successivi del gioco hanno introdotto nuove modalità di gioco quali «Snowday» ispirata ad una partita di *hockey su ghiaccio* in cui la palla é sostituita da un *disco* caratterizzato da una fisica diversa, «Hoops» ispirata al *basket* in cui la porta é sostituita con un *canestro* ed infine «Rissa» in cui i giocatori possono usare diversi potenziamenti casuali al fine di congelare la palla, agganciarla con una fune e così via. L'ultima modalità di gioco introdotta nel 2017 «Dropshot» stravolge le regole del gioco, rimuovendo completamente le porte. In questa modalità il pavimento é suddiviso in piastrelle esagonali che, colpite dalla palla due volte

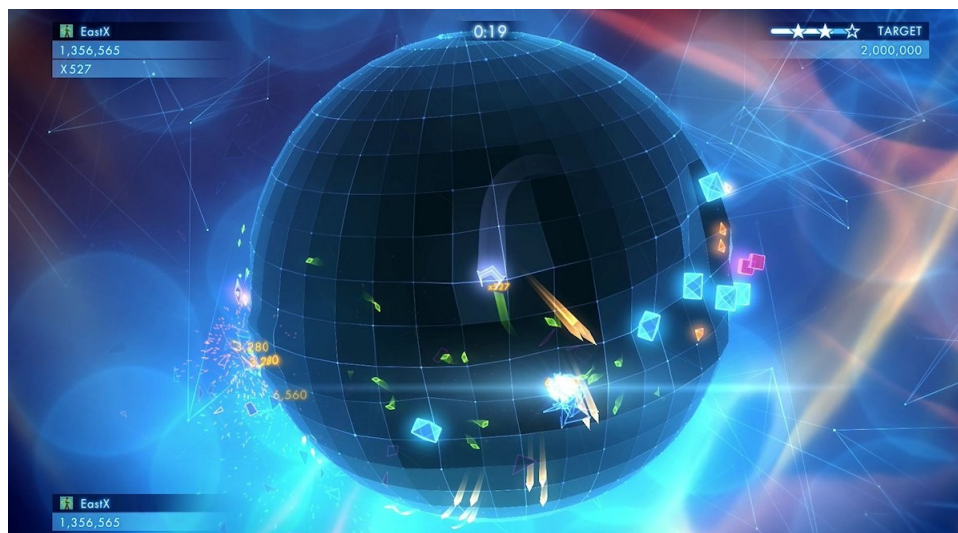


Figura 12: Geometry Wars 3: Dimensions Evolved - Arena sferica

in successione, crollano lasciando una voragine nel pavimento che funge da porta per il team a cui appartiene la mezzeria di campo.

Il gioco presenta inoltre una modalità *competitiva online* basata sulle regole principali le cui *stagioni* hanno durata pari ad alcuni mesi. I giocatori sono organizzati in *categorie*, in funzione del proprio livello di abilità, e suddivise a loro volta in quattro *divisioni*; i risultati del team consentono allo stesso di cambiare di divisione all'interno della *categoria* d'appartenenza.

2.1.3 Geometry Wars 3: Dimensions Evolved

Geometry Wars 3: Dimensions Evolved è un gioco *shoot 'em up* sviluppato da *Lucid Games* e pubblicato da *Sierra Entertainment* nel 2014 per tutte le maggiori piattaforme *desktop*, *mobile* e *console*. Terzo capitolo della saga *Geometry Wars* è il primo gioco della serie ad introdurre una modalità «avventura» ed un ambiente di gioco totalmente 3D (Fig.12).

Nel gioco l'utente controlla una piccola astronave in grado di muoversi rapidamente e sparare in qualsiasi direzione al fine di evitare e distruggere ondate di astronavi nemiche. Il giocatore ha a disposizione un *drone* che lo accompagnerà in battaglia aumentandone la potenza di fuoco ed aiutandolo a collezionare gli oggetti sparsi per la mappa e una *super* abilità dagli effetti devastanti utilizzabile un numero limitato di volte. Esistono diversi *droni*

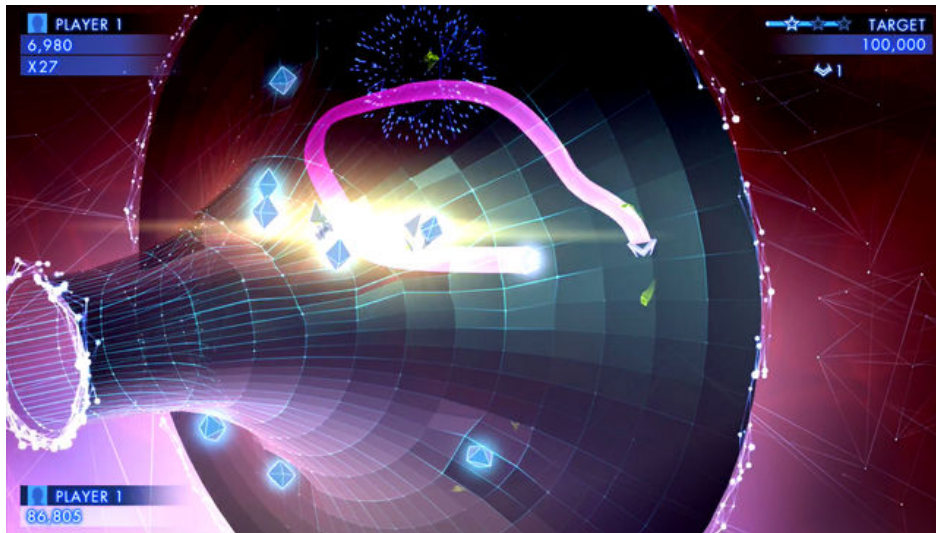


Figura 13: Geometry Wars 3: Dimensions Evolved - Arena

e *super* abilità, ciascuno caratterizzato da diversi stili di gioco ed effetti. Durante le partite è inoltre possibile collezionare *super stati*, potenziamenti generati casualmente nel livello che forniscono *barriere*, *armi potenziate* e *magneti* per brevi periodi di tempo, e *geom*, oggetti che fungono da *moltiplicatore di punteggio* ottenibili abbattendo le navi avversarie. Collezionare *geom* è una delle meccaniche fondamentali del gioco in quanto consente al giocatore di aumentare il proprio punteggio di svariati ordini di grandezza. Al raggiungimento di certe soglie di punti, il giocatore viene inoltre premiato con *vite extra* o potenti *bombe* in grado di distruggere tutte le astronavi nemiche.

I livelli sono caratterizzati da diverse *topologie* e *dimensioni* e si sviluppano formando solidi tridimensionali quali *sfere*, *cubi* e *nastri* percorribili lungo la superficie (Fig.13) . I nemici vengono generati ad *ondate* in punti specifici di ciascuna mappa e sono caratterizzati da diversi schemi di movimento e strategie: alcuni tendono ad essere più aggressivi, altri più rapidi, altri ancora prediligono una strategia più elusiva. La successione delle ondate in ciascun livello è prestabilita ed è pertanto possibile ricordarla a memoria al fine di massimizzare il punteggio a fine gara.

Il gioco prevede *dodici* modalità di gioco *single-player* organizzate in una modalità «classica» che comprende *sei* delle modalità storiche della saga e giocabili su qualsiasi livello ed una modalità «avventura» in cui il giocatore

affronta *cinquanta livelli* diversi in sequenza, ciascuno dei quali caratterizzato da regole proprie. Le modalità di gioco consistono nel ottenere il maggior numero di punti avendo a disposizione solo un numero limitato di vite ed oggetti speciali oppure entro un tempo limite prestabilito. Alcune modalità inoltre introducono diverse regole, quali numero limitato di munizioni, assenza di armi, presenza di nemici in grado di dividersi, aree di gioco che diventano via via sempre più ristrette, e così via. Durante la modalità «avventura» il giocatore sarà costretto ad affrontare dei *boss*, potenti nemici in grado di sostenere molteplici danni e in grado di usare speciali scudi per rigenerare la propria *salute*. Sono inoltre disponibili due modalità *multiplayer*: una *locale* cooperativa a visuale condivisa, caratterizzata dalla sola presenza di arene bidimensionali e un numero di giocatori compreso tra due e quattro, ed una *online competitiva* fino ad un massimo di otto giocatori.

2.2 OrbTail

OrbTail é un gioco competitivo con elementi *racing* in cui quattro partecipanti si sfidano all'interno di un'arena. Il titolo offre diverse modalità di gioco a tempo caratterizzate da condizioni di vittoria differenti e molteplici arene con un diverso grado di difficoltà. Il titolo é stato sviluppato per le principali piattaforme *desktop* e *mobile* quali Windows, OSX, Android e iOS e consente agli utenti di giocare tra loro indipendentemente dalla piattaforma utilizzata in modalità *cross-platform play*.

In OrbTail ogni giocatore controlla un veicolo in grado di muoversi agilmente in un'arena, collezionando elementi di gioco sferici detti *orb* che si agganciano ad esso formando una lunga coda. É possibile utilizzare il proprio veicolo per scontrarsi contro quelli avversari al fine di sottrarre *orb* in misura proporzionale alla direzione e alla forza d'impatto. Questi elementi sono inizialmente distribuiti lungo la superficie dell'arena e la loro acquisizione costituisce la meccanica principale attorno alla quale ruotano le varie modalità di gioco descritte nei capitoli successivi. Gli *orb* rappresentano la *risorsa limitata* su cui si concentra *l'elemento di sfida* tra i vari giocatori. A seguito di valutazioni empiriche é stato deciso che il numero di *orb* per arena é pari a *ventotto*: tale quantitativo garantisce un buon bilanciamento tra fasi di scontro e fasi di raccolta.

All'interno dell'arena sono inoltre distribuiti degli elementi di gioco rari, detti *core*, che garantiscono al giocatore un potenziamento casuale temporaneo. Lo scopo di questi potenziamenti é quello di aggiungere *profondità* al *gameplay* e scuotere i normali equilibri di gioco.

Per garantire la massima fruizione del videogioco in varie casistiche, tutte le modalità di gioco ed arene sono disponibili in configurazione a *giocatore singolo*, *multigiocatore locale* e, qualora dovesse essere disponibile una connessione ad Internet, *multigiocatore online*. É inoltre possibile giocare in *modalità online mista* in cui piú giocatori condividono una stessa postazione e giocano con altri giocatori su altri dispositivi. La configurazione *multigiocatore locale* é gestita tramite *splitscreen* con numero di partecipanti compreso tra due a quattro. Questa configurazione é disponibile per le sole piattaforme *desktop* essendo queste le uniche in grado di supportare piú periferiche di input contemporaneamente. I dispositivi *mobile* ne sono esclusi sia per ragioni di performance e sia perché queste piattaforme sono solitamente ottimizzate per l'interazione da parte di un unico utente.

Tutte le modalità di gioco sono state concepite considerando un numero di partecipanti fisso pari a *quattro* e bilanciate di conseguenza: qualora il numero di giocatori dovesse essere inferiore, i veicoli rimanenti verranno scelti e controllati da un'opportuna *intelligenza artificiale*. Questi veicoli sono in grado di eseguire semplici compiti quali collezionare gli elementi di gioco, esplorare l'ambiente e scontrarsi con gli avversari. Per ragioni tecniche il livello di difficoltà dell'*intelligenza artificiale* è unico e non è possibile configurarlo. Nel design originale di gioco l'*intelligenza artificiale* era disponibile nella sola modalità a giocatore singolo: era pertanto possibile giocare sessioni *online* con un numero inferiore di giocatori. Questa limitazione è stata *rimossa* al fine di garantire una più elevata consistenza tra le esperienze *singleplayer* e *multiplayer* e, allo stesso tempo, aumentare la difficoltà delle sessioni online.

Le meccaniche di base sono pensate per essere semplici, al fine di adattarsi ad un vasto pubblico di giocatori su un gran numero di piattaforme differenti e bilanciate in modo da non avvantaggiare nessuna di queste durante le sessioni *cross-platform play*. La durata ridotta delle sessioni di gioco, solitamente inferiore ai *tre minuti*, unita al gran numero di configurazioni disponibili, garantisce una *versatilità* del prodotto unica: il gioco è tanto adatto al *casual gamer* in cerca di un'esperienza a giocatore singolo, quanto ad un gruppo di amici in cerca di un *party-game* rapido e competitivo.

2.2.1 Modalità di gioco

Il titolo offre diverse modalità di gioco accomunate dalle medesime meccaniche di base ma con condizioni di vittoria differenti. Le modalità di gioco sono rimaste largamente immutate rispetto al *concept* originale del 2014.

Arcade In questa modalità l'obiettivo è quello di fare più punti possibile entro un tempo limite prestabilito di *centoventi secondi*. Ogni *orb* collezionato aumenta di *dieci* il punteggio del giocatore. Questa modalità di gioco è pensata per favorire scontri tra i veicoli dei partecipanti ed è caratterizzata da un *ritmo* frenetico per l'intera durata della sfida. Il numero limitato di *orb* porta i giocatori allo scontro continuo in modo che ve ne sia sempre un discreto numero da collezionare al fine di garantire un flusso costante di punti. In questa modalità gli scontri

tra i veicoli possono risultare tanto positivi per l'attaccante quanto per il difensore: una volta collezionati tutti gli *orb* a disposizione, l'unico modo per fare piú punti é liberarne di nuovi, anche a discapito dei propri. Questa meccanica consente di mantenere un elevato grado di sfida in quanto, al termine di ciascuno scontro, ogni giocatore dovrà fare del suo meglio per raccogliere gli *orb* staccati prima che gli avversari possano fare altrettanto. Nelle fasi finali i giocatori dovranno inoltre essere piú cauti, valutando gli esiti di ciascun scontro ed evitandone di inutili in caso di vantaggio.

Longest Tail In questa modalità l'obiettivo é quello di terminare la partita col maggior numero di *orb* attaccati alla propria coda. Qualora non dovesse essere possibile determinare un vincitore unico al termine dei *centoventi secondi*, la partita termina in *parit  *. Questa modalit   é caratterizzata da un *ritmo* crescente che culmina in una fase finale particolarmente *frenetica* ed *imprevedibile*. L'esito della partita determinato solo dalle condizioni finali di gioco, favorisce la nascita di *alleanze implicite* e *mutevoli* tra i partecipanti e finalizzate ad intralciare il giocatore attualmente in vantaggio ed impedire che questi vinca. Secondo questa dinamica di gioco, il giocatore con la coda pi   lunga sar   portato a *fuggire* dagli avversari, evitando di perdere il vantaggio, laddove tutti gli altri saranno portati a intraprendere azioni pi   spericolate, sperando di ribaltare l'esito della partita. Questi due *ruoli* cambiano rapidamente nel tempo, specialmente nelle fasi finali in cui sar   richiesta un'elevata *concentrazione* e *reattivit  *.

Eliminazione L'obiettivo di questa modalit   é *eliminare* tutti gli altri partecipanti prima che questi possano fare altrettanto. All'inizio della partita gli *orb* presenti nel livello vengono equamente ripartiti fra tutti i giocatori, formandone la *coda* iniziale. Un giocatore rimane in partita fintanto che la sua coda contiene almeno un elemento, altrimenti viene *eliminato*. Questa modalit   é caratterizzata da un elevato grado di *competizione* e un livello di difficolt   che cresce al diminuire del numero di partecipanti in gara. Ogni volta che un giocatore viene eliminato, i veicoli rimanenti si spartiranno i suoi *orb*, aumentando le loro possibilit   di sopravvivenza. Come per la precedente modalit  , i partecipanti saranno portati a concentrare i propri sforzi per eliminare il giocatore col maggior numero di *orb* e ci   consente di bilanciare il *game-*

play in presenza di giocatori particolarmente forti. I giocatori eliminati possono assistere al resto della sfida attraverso un'opportuna modalità *spettatore* che consente loro di inquadrare i partecipanti ancora in vita. Questa modalità ha inoltre un tempo limite di *centottanta secondi*, scaduti i quali la partita termina in parità. Questo limite temporale è necessario per incentivare lo scontro tra i partecipanti rimanenti, impedendo che la partita si protragga indefinitamente. La durata è più elevata rispetto alle modalità precedenti in quanto si vuole aumentare la probabilità che la partita finisca a causa dell'eliminazione dei partecipanti invece che per esaurimento del tempo di gara.

2.2.2 Veicoli

Il giocatore ha a disposizione *sei* veicoli caratterizzati da aspetto e stili di guida diversi. I veicoli possono *accelerare*, *frenare* e *sterzare*; l'uso prolungato del freno consente al veicolo di *accelerare* in *retromarcia*. I diversi stili di guida consentono ai veicoli di adattarsi a diverse tipologie di giocatori o strategie e sono determinati da tre *parametri* fondamentali:

Velocità Rappresenta la massima velocità raggiungibile dal veicolo, influenzandone il *potenziale offensivo, difensivo e tattico*. Un valore elevato di *velocità* consente di staccare più *orb* dagli avversari a seguito di uno scontro, raggiungere più velocemente obiettivi di gioco e allontanarsi rapidamente dai veicoli avversari vanificando eventuali tentativi offensivi.

Accelerazione Rappresenta la massima accelerazione del veicolo. Un valore più elevato consente al veicolo di raggiungere velocità maggiori sulle brevi distanze, causando danni maggiori in ambienti ristretti e minimizzando le circostanze in cui il veicolo si trova fermo e quindi facilmente bersagliabile. Questo parametro ha valenza principalmente *tattica* e, più limitatamente, *offensiva*.

Manovrabilità Rappresenta la massima velocità a cui il veicolo può sterzare. Veicoli caratterizzati da elevata *manovrabilità* sono facilitati nel cambio di direzione e pertanto sono più efficienti durante le fasi di raccolta di elementi di gioco quali *orb*. La possibilità di cambiare rapidamente traiettoria consente inoltre una maggiore precisione durante gli scontri o le schivate.

Veicolo	Velocitá	Accelerazione	Manovrabilitá
Flare	• • • • • • •	• • • • • • •	• • • • • • •
Flash	• • • • • • • • •	• • • • •	• •
Glow	• • • • •	• • • • • • •	• • • • • • • • • •
Radiance	• • • • • •	• • • • • • • • • •	• • • • •
Shine	• • • • • • • • •	• • • • • • • •	• • • • •
Sparkle	• • • • •	• • • • • • • • •	• • • • • • • • •

Tabella 1: Parametri dei veicoli

Il design originale di OrbTail prevedeva due parametri aggiuntivi, *attacco* e *difesa*, i quali influivano direttamente sull’esito degli scontri e il numero di *orb* staccati. Dopo varie iterazioni é stato deciso di rimuovere questi due parametri in quanto il loro contributo risultava impercettibile durante le sessioni e perché l’interazione con il parametro *velocitá* rendeva difficile differenziare e bilanciare i vari veicoli. Questi due parametri sono utilizzati internamente per determinare l’esito degli scontri tra i veicoli ma sono configurati allo stesso modo su ciascuno di essi.

Di seguito sono riportate le descrizioni dettagliate di tutti i veicoli disponibili. I parametri sono stati configurati in maniera *euristica* al fine di esaltare diversi stili di gioco e bilanciati in modo da non avvantaggiare nessun veicolo (Tab.1) .

Flare Caratterizzato da uno stile di guida *bilanciato*, questo veicolo é pensato per adattarsi ad ogni tipo di strategia. Indicato per i nuovi giocatori.

Flash Dotato di un’*elevata velocitá*, questo veicolo esprime tutto il suo *potenziale offensivo* sulla lunga distanza sacrificando gran parte della manovrabilitá. Particolarmente adatto per i giocatori piú esperti che prediligono strategie ad alto rischio ed alto guadagno.

Glow Questo veicolo é dotato di un’elevata mobilitá e ciò lo rende particolarmente adatto per collezionare *orb* e *core*. La bassa velocitá é compensata dalla facilitá di accesso ai potenziamenti e ciò lo rende adatto ai giocatori che prediligono uno stile di gioco piú strategico.

Radiance L'elevata *accelerazione* e *velocità* di questo veicolo lo rendono particolarmente efficace in *mischia*. Adatto per strategie aggressive sulle medie e brevi distanze.

Shine Veicolo bilanciato, sacrifica parte della manovrabilità per un più elevato *potenziale offensivo*. Adatto a giocatori intermedi.

Sparkle Veicolo ottimizzato per la *mobilità* e caratterizzato dall'elevato *potenziale difensivo*. La reattività di questo veicolo lo rendono perfetto per *schivare* tentativi offensivi da parte degli avversari.

Al fine di facilitare il riconoscimento dei partecipanti in gara, ciascun veicolo prevede quattro differenti *livree*, caratterizzate da un aspetto e colore diverso. La livrea associata al veicolo viene automaticamente selezionata dal sistema in funzione dell'indice del giocatore all'interno della partita: il primo giocatore avrà una *livrea rossa*, il secondo una *blu* e gli altri due rispettivamente una *livrea verde* e una *gialla*.

2.2.3 Level design

Il titolo offre *tre* livelli totali, caratterizzati da diverso aspetto e *topologia*. Le arene si sviluppano in solidi tridimensionali: i veicoli possono muoversi solamente lungo loro superficie ma non possono mai passarvi attraverso o volare al loro interno. Sebbene le arene si differenziano per dimensione, l'ambiente di gioco rimane sempre ragionevolmente limitato. Le motivazioni di questa scelta di design sono molteplici. In primo luogo la durata limitata delle partite richiede che i giocatori debbano interagire tra di loro più frequentemente possibile nel breve tempo a disposizione, si vuole evitare quindi il dover attraversare grandi distanze prima di potersi scontrare. In secondo luogo si vogliono evitare strategie particolarmente *elusivo* in cui un giocatore continua a fuggire dagli avversari senza che questi possano raggiungerlo o *accerchiarlo*. Questa proprietà è fondamentale per garantire un grado di sfida adeguato nelle modalità *longest tail* ed *elimination*. Nel *concept* originale del gioco era prevista la presenza di elementi attivi ed ostacoli all'interno dei livelli: sebbene questa funzionalità avrebbe certamente arricchito l'esperienza di gioco, ciò avrebbe richiesto lo sviluppo di un'*intelligenza artificiale* più sofisticata. Questa motivazione, unita ai tempi di sviluppo ridotti ha portato alla rimozione dei suddetti.

Di seguito sono riportate le descrizioni delle arene disponibili.

Stadium Il livello é costituito da un'arena *emisferica* in cui i veicoli possono correre sulla sua base ma non possono superarne i bordi. Questo livello é caratterizzato da una topologia *piana* e pertanto tutti gli elementi di gioco sono sempre ben visibili. Questa arena é adatta per i nuovi giocatori oppure per gli utenti che preferiscono concentrarsi sull'azione di gioco evitando la difficoltà introdotta da topologie diverse.

Planetarium Questa arena é ricavata all'interno di una *sfera* ed é caratterizzata da una forza di gravità radiale che spinge i veicoli e tutti gli elementi di gioco sulla *superficie interna* della stessa. La topologia di questa arena richiede un livello di abilità maggiore da parte dei partecipanti, in quanto la curvatura limita la visibilità degli elementi di gioco. La grande estensione superficiale del livello rende inoltre più complicato individuare la posizione degli avversari e capirne le traiettorie.

Torus Arena caratterizzata da una topologia *toroidale* e campo gravitazionale ad anello in cui gli elementi di gioco sono distribuiti lungo la *superficie interna*. Questo livello costituisce il massimo grado di sfida offerto dal gioco. La topologia complicata rende particolarmente arduo valutare correttamente le distanze degli oggetti e prevedere le traiettorie degli avversari. La guida dei veicoli all'interno di quest'arena richiede inoltre un più elevato grado di precisione in quanto non é sempre semplice individuare il percorso migliore per raggiungere un determinato obiettivo.

Durante lo sviluppo sono state inoltre valutate topologie *convesse* (come ad esempio la superficie esterna di una *sfera* o di un *toro*), tuttavia a parità di dimensione con la controparte *concava*, il raggio di curvatura risultava tale da ridurre drasticamente il numero di elementi di gioco a schermo e rendere ancora più complicato individuare avversari ed oggetti. A seguito di queste considerazioni é stato deciso di evitare l'aggiunta di suddette topologie.

2.2.4 Potenzamenti

All'interno delle arene sono distribuiti degli speciali elementi di gioco detti *core* che, una volta raccolti, garantiscono al giocatore un *potenziamento* o un'arma casuale attivabile a comando. Questi elementi di gioco sono stati introdotti al fine di smuovere l'*equilibrio* della partita, introducendo una

maggiore profondità e varietà al *gameplay*. Il *sapiente* uso di questi potenziamenti al momento giusto può ribaltare completamente le sorti di una partita o consolidare la propria posizione di vantaggio. Per evitare eventuali abusi i potenziamenti *non sono cumulabili* e vengono *consumati* una volta utilizzati.

Nel *design originale* questi potenziamenti erano ottenibili raccogliendo degli *orb* che venivano «imbevuti» casualmente una volta ogni *dieci* secondi e iniziavano a brillare vistosamente. Durante le iterazioni ci si è resi conto che sfruttare gli *orb* stessi per fornire potenziamenti impediva ai giocatori di opporsi agli avversari in posizione di estremo vantaggio ed in generale rendeva molto difficile ottenerne di nuovi: basta considerare che durante la partita il numero di *orb* non collezionati è sempre molto basso (e quindi nel caso generico non ne venivano «imbevuti» abbastanza). Per questo motivo è stato deciso di introdurre un elemento di gioco collezionabile apposito, il *core*, il cui numero per arena è fissato a *quattro*. A differenza degli *orb*, questi elementi scompaiono una volta collezionati e vengono rigenerati nello stesso punto dopo un breve periodo di *cinque secondi*.

Di seguito sono riportati i potenziamenti ottenibili attraverso i *core*.

Turbo Una volta attivato, questo potenziamento imprime una forza costante che spinge il veicolo in avanti per *tre* secondi a grande velocità. Questo potenziamento è pensato per essere utilizzato in ambito *offensivo* per aumentare il numero di *orb* staccati o *difensivo*, quando è necessario distanziare gli avversari rapidamente.

Gravità Una volta attivato, permette al veicolo di attrarre a sé tutti gli *orb* nel raggio di *tre* metri per *sette* secondi. Grazie a questo effetto è possibile raccogliere un gran numero di *orb* in poco tempo e ciò rende questo potenziamento uno dei più apprezzati e versatili.

Dirottamento Il veicolo lascia cadere un elemento di gioco visivamente identico ad un *orb*. Gli avversari che tentano di raccogliere questo oggetto perdono il controllo del proprio veicolo per *due* secondi. L'uso di questo potere è principalmente difensivo e permette di neutralizzare gli avversari temporaneamente per allontanarsene.

Missile Lancia un missile a ricerca verso l'avversario più vicino. Il missile esplode a contatto oppure automaticamente dopo 5 secondi: i veicoli

coinvolti nell'esplosione perdono *due orb*. Quest'arma ha effetti particolarmente devastanti nella modalità *elimination* o nelle fasi finali di *longest tail* in quanto consente di ribaltare le sorti della partita. Il raggio di curvatura limitato di quest'arma consente ai giocatori piú abili di schivare il missile, in attesa che questi esploda automaticamente.

Invincibilitá Una volta attivato, il veicolo del giocatore non può perdere *orb* per *due* secondi. Sebbene rappresenti il potere col piú alto potenziale *difensivo*, in grado di proteggere il veicolo da armi avversarie o scontri, la sua breve durata richiede un'adeguata temporizzazione per ottenerne il massimo effetto.

Sovraccarico Raddoppia la velocità massima del veicolo per *sette* secondi. A differenza del «turbo», garantisce una manovrabilità maggiore al costo di un aumento di velocità inferiore.

Proiettile Il veicolo spara un proiettile in linea retta che esplode a contatto oppure automaticamente dopo *sette* secondi. I veicoli coinvolti nell'esplosione perdono 4 *orb*. Quest'arma risulta particolarmente efficace sulle brevi distanze dove vi é una piú elevata probabilità di colpire gli avversari. Il gran numero di *orb* staccati consente di ridurre il vantaggio degli avversari oppure di *neutralizzarli* definitivamente qualora dovessero averne in numero esiguo.

Scudo Il veicolo del giocatore é immune da tutte le armi nemiche per *sette* secondi. A differenza dell'«invincibilitá», questo potere garantisce una protezione piú estesa da tutte le armi avversarie.

Nella versione originale erano previsti alcuni poteri che avevano dagli effetti *negativi* sul veicolo del giocatore. L'intento era quello di introdurre una meccanica simile ad una *scommessa*, secondo la quale il giocatore che collezionava un potenziamento non era sempre sicuro di ottenerne un beneficio. Nella versione finale queste meccaniche sono state rimosse al fine di promuovere ancor di piú l'uso di potenziamenti ed in generale perché risultavano ingiustificatamente punitive.

2.2.5 Interazione utente

OrbTail é caratterizzato da un numero molto limitato di controlli ed azioni che l'utente può effettuare e ciò lo rende facilmente fruibile su tutte le piattaforme supportate.

Per le piattaforme *mobile* l'interazione utente all'interno del menú é affidata al *touch*, a differenza della gara vera e propria in cui, per evitare che l'utente possa coprire gran parte dello schermo con le proprie mani, i controlli sono affidati all'*accelerometro*. L'inclinazione del dispositivo consente al veicolo di *accelerare* oppure di andare in *retromarcia*. Piegando il dispositivo a destra o sinistra é inoltre possibile *sterzare* in misura tanto maggiore quanto piú grande l'angolo d'inclinazione. L'attivazione dei potenziamenti e delle armi é invece affidata ad un semplice *tocco* dello schermo in qualsiasi punto.

La versione *desktop*, a differenza di quella *mobile*, deve essere inoltre interagibile da piú partecipanti contemporaneamente ed é pertanto previsto il supporto per *tastiere*, *mouse* e fino a *quattro pad*. Tramite *tastiera* é sufficiente usare le *freccie direzionali* per muovere il veicolo e la *barra spaziatrice* per attivare i potenziamenti. Questi tasti sono liberamente configurabili da parte dell'utente. L'interazione tramite *pad* prevede l'uso dei *grilletti* posteriori per accelerare, frenare o attivare la retromarcia, lo *stick sinistro* per sterzare e un *tasto* apposito per attivare i potenziamenti. L'assenza di gradi di libertà da parte della tastiera impedisce all'utente di poter guidare con precisione il veicolo e ciò rende il titolo piú adatto all'interazione tramite *pad*. É inoltre possibile interagire con tutti gli elementi dell'interfaccia grafica (in menú e in gara) attraverso il *mouse*.

2.2.6 Interfaccia utente

L'interfaccia utente é pensata per essere poco intrusiva: gli elementi dell'*HUD* sono ridotti al minimo e consentono di avere sempre una buona visione del campo di gioco, ciò é particolarmente importante sui dispositivi *mobile* caratterizzati solitamente da schermi dalle dimensioni ridotte.

I menu all'interno del gioco sono caratterizzati da elementi sempre ben visibili, sono escluse pertanto liste a scorrimento o menú con un gran numero di voci. Questa tipologia di menú é necessaria per facilitare l'interazione tramite *touchscreen*. Laddove ci si aspetta che l'utente faccia piú scelte (ad esempio modalitá di gioco, arena e veicolo), le singole scelte sono suddivise su piú pagine.

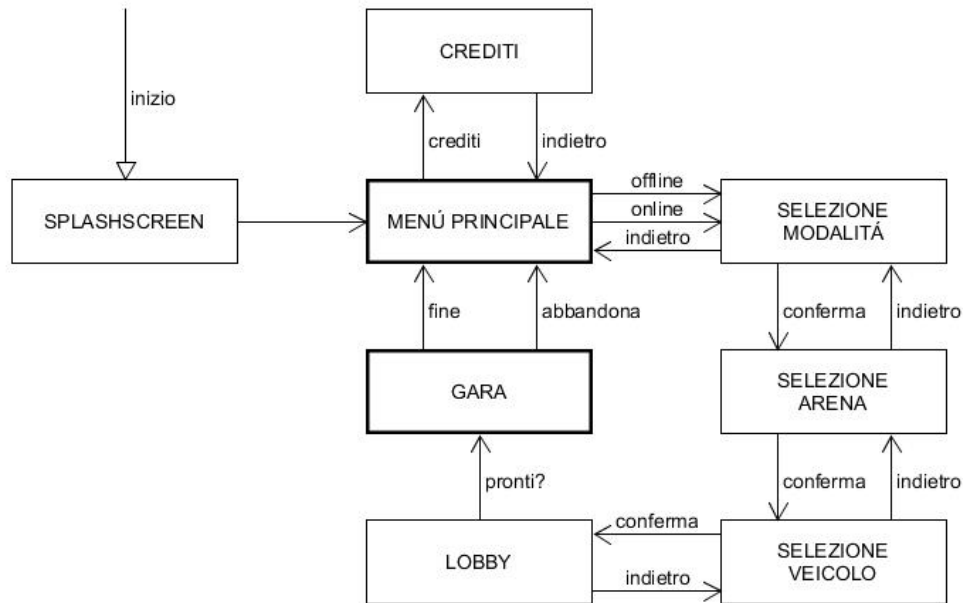


Figura 14: Flusso di gioco

La *HUD* all'interno di ogni sessione consiste di un *timer di gara* che mostra il tempo rimanente prima della conclusione del match e altri *quattro* elementi colorati finalizzati a mostrare il punteggio o la lunghezza della coda di ciascun partecipante a seconda della modalità di gioco. Un elemento grafico apposito consente di visualizzare il *potenziamento* o l'*arma* a disposizione del partecipante. Un tasto consentirà al giocatore di abbandonare la partita e ritornare al menù principale. Per evitare pressioni accidentali sarà necessario interagire due volte in rapida successione al fine di confermare l'azione.

Il gioco prevede infine un elemento tridimensionale finalizzato ad aiutare il giocatore ad orientarsi meglio. Questo elemento ha funzione di *bussola* e consente di visualizzare la direzione relativa dei veicoli avversari rispetto a quella del giocatore.

2.2.7 Flusso di gioco

Il flusso di gioco é pensato per portare rapidamente l'utente in partita, riducendo al minimo il numero di interazioni necessarie (Fig.14) . Il flusso

prevede la selezione della *modalità di gioco*, dell'*arena* e del *veicolo*, dopodiché l'utente sarà portato in un'apposita schermata di *matchmaking* in cui il sistema provvederà a creare la *lobby*. La selezione della modalità di gioco e dell'*arena* è *opzionale*: qualora l'utente non dovesse esprimere alcuna preferenza, il sistema provvederà automaticamente a riempire una delle *lobby* esistenti. Durante la selezione del veicolo sarà possibile aggiungere più giocatori locali alla partita (attraverso un opportuno tasto sul *pad*) ed unirsi ad una *lobby* in gruppo. Qualora non dovesse esserci nessuna *lobby* disponibile, il sistema provvederà a crearne una nuova a cui altri utenti online potranno liberamente accedervi. Questo processo non è necessario per la modalità a *giocatore singolo*.

Affinché la partita possa iniziare, *tutti* i giocatori in *lobby* devono esplicitamente dichiarare di essere *pronti*: il sistema provvederà ad aggiungere *intelligenze artificiali* finché il numero di partecipanti non diventa esattamente *quattro*. Un breve conto alla rovescia porterà i giocatori in gara. All'inizio di ogni partita è prevista una schermata di *tutorial* che spiega le regole della modalità di gioco selezionata ed i potenziamenti disponibili. Questa schermata ha la duplice funzione di *tutorial* e di punto di sincronizzazione, in attesa che tutti gli altri giocatori online abbiano caricato il livello. La partita inizia una volta che tutti i giocatori hanno congedato la schermata.

Gli utenti possono tornare al menù principale una volta terminata la partita oppure *abbandonarla* in qualsiasi momento.

3 Direzione artistica

3D, Reference, Art Bible

4 Sviluppo

Conclusioni

Riferimenti bibliografici

- [1] http://www.theesa.com/wp-content/uploads/2018/05/EF2018_FI-NAL.pdf