



**Politecnico
di Torino**

POLITECNICO DI TORINO

**Corso di Laurea Magistrale
in Ingegneria Informatica**

Anno Accademico 2021-2022

**IL METAVERSO:
UN'ANALISI SULLO STATO DELL'ARTE**

Relatore

Prof. Danilo Bazzanella

Correlatore

Jacopo Romagnoli

Candidato

Carmelo Proetto



**Politecnico
di Torino**

Prefazione

Nell'ultimo anno abbiamo assistito a una vera e propria rivoluzione nel mondo tecnologico che punta alla costruzione di una nuova era digitale basata sul Metaverso. Seppure non ancora chiaro di cosa si stia parlando, il concetto di Metaverso ha preso sempre più piede ai giorni nostri suscitando curiosità anche e soprattutto nel mondo aziendale.

Il Metaverso rappresenta un canale di comunicazione non ancora totalmente esplorato e chiaro, è per questo che l'obiettivo di questa tesi è quello di approfondire il concetto di Metaverso, partendo da una definizione formale, passando alle sue origini, allo stato dell'arte attuale e alle prospettive future, per giungere fino ad un'analisi tecnica, provando ad analizzare ed implementare i vari blocchi che ne costituiscono l'architettura.

In questo lavoro di tesi, dunque, si partirà introducendo il termine Metaverso, analizzandone le caratteristiche, la sua storia e quali sono i principali attori che ne permettono il funzionamento.

Successivamente verranno introdotti i concetti di Realtà Virtuale e Realtà Aumentata, per poi passare alla vera spina dorsale del Metaverso ovvero la Blockchain e di come, anche quest'ultima, possa essere rivoluzionaria nel contesto aziendale, portando con sé tutti i vantaggi di un sistema decentralizzato. Verranno analizzati gli Smart Contract, le Criptovalute e gli NFT, fondamentali nel contesto Metaverso e verranno valutati anche in questo caso i benefit che potrebbero portare se utilizzati.

Nel quarto capitolo sarà introdotta una lista dei principali Metaversi attualmente attivi e se ne valuteranno le caratteristiche tecniche, i pro e contro che portano con sé e a quale tipologia di utenti sono indirizzati. Si parlerà quindi dei due principali Metaversi: Decentraland e Sandbox, oltre a citare ed analizzare una più piccola realtà come OVR, particolare e differente dalle altre analizzate in quanto basata non sulla Realtà Virtuale, bensì su quella Aumentata.



Infine, nel quinto capitolo sarà analizzata l'architettura di un Metaverso, valutando tutti i blocchi necessari per la sua costruzione, verrà quindi realizzato un prototipo di Metaverso nella quale l'utente potrà acquistare, tramite l'utilizzo di Smart Contract su blockchain Ethereum, degli appezzamenti di terreno fra quelli disponibili. Infine, parlando del progetto realizzato all'interno del Metaverso Spatial per Var Group, sarà analizzato come attualmente si presenta il Client di un Metaverso. Tale progetto è stato realizzato per essere presentato durante il Convegno dell'azienda tenutosi a Riccione il 20-21 ottobre 2022. Ho provato a ricostruire, tramite tecniche di modellazione e l'utilizzo di Blender, un ambiente accogliente ed efficace, che espliciti la realtà aziendale e che offra inoltre degli spazi virtuali, come un auditorium, dove poter accogliere utenti interessati al mondo VarGroup e potenziali clienti.

Ringraziamenti

Non mi sembra neanche vero che per la seconda volta nella mia vita mi ritrovo qui a scrivere dei ringraziamenti. Questi due anni sono stati molto intensi per me, pieni di alti ma anche di profondi bassi.

Parto col dire che sono orgoglioso di aver terminato il mio percorso di studi: l'esperienza a Torino è stata un qualcosa che mi ha profondamente segnato in positivo e mi ha fatto crescere non solo dal punto di vista formativo ma anche da quello personale. Un'esperienza fatta di studio, sacrifici, ansie ma anche di momenti di svago e pura felicità. La vita però purtroppo non è sempre rose e fiori e quindi ho dovuto passare anche delle esperienze di profonda malinconia.

Due anni fa dedicavo la mia tesi triennale a te nonna, che oggi sei la stella più bella del cielo. Questo traguardo lo dedico soprattutto a te. So che aspettavi felicissima questo momento, ma so anche che da lontano stai festeggiando comunque.

A festeggiare da lassù ci sarai anche tu piccola altra stella. Saresti stata molto contenta di essere qui oggi, come due anni fa. Non si può dimenticare il tuo sorriso nei momenti di festa.

Il ringraziamento più grande va a mio padre, mia madre e Martina, grazie davvero per esserci sempre stati, nei momenti di gioia e di dolore, nelle ansie e nelle vittorie. Grazie per aver sempre sostenuto le mie scelte, spero che un giorno riuscirò a ripagare tutti i vostri sacrifici.

Ringrazio il prof. Danilo Bazzanella, il mio relatore Jacopo e Pierpaolo per avermi aiutato nello svolgimento di questo percorso di tesi.

Ringrazio la mia numerosa famiglia per il sostegno che mi ha sempre regalato anche da lontano. Anche un semplice messaggio a volte dà i giusti stimoli per affrontare al meglio le sfide che questo percorso mette davanti.

Ringrazio i miei amici di Nicosia, una certezza nella mia vita. Anche se le nostre strade pian piano vanno a separarsi, ci siete sempre stati e spero ci siate sempre.

Ringrazio i miei colleghi ed i miei amici di Torino. In quest'ultimo anno siete stati parte fondamentale della mia vita. Abbiamo vissuto insieme tantissime esperienze e ci siamo sempre dati forza nei momenti di sconforto. Non potevo desiderare di trovare compagnia migliore.

Infine, ringrazio me stesso per essere oggi qui. I miei sacrifici, la mia determinazione e le mie ambizioni mi hanno portato ad affrontare e terminare questo percorso di studi ma anche e soprattutto di vita.

Sommario

1	Introduzione generale	1
1.1	Definizione di Metaverso	1
1.2	La Storia del Metaverso	3
1.2.1	Dove tutto è iniziato	3
1.2.2	Dalle prime applicazioni ai giorni nostri	4
1.2.3	Attualità	5
1.2.4	Principali competitor e prospettive future	6
1.3	Gli utenti nel Metaverso	7
2	La tecnologia alla base del Metaverso.....	9
2.1	Realtà Aumentata	9
2.1.1	Definizione	9
2.1.2	Storia	11
2.1.3	Tipologie di dispositivi	14
2.1.4	Sistemi di tracciamento	19
2.1.5	Il futuro della realtà aumentata	20
2.2	Realtà Virtuale	21
2.2.1	Definizione	21
2.2.2	Storia	23
2.2.3	La tecnologia alla base della VR	25
2.2.4	Il futuro della realtà virtuale	29
3	Il cuore del Metaverso: la Blockchain	31
3.1	Storia	32
3.2	Bitcoin	34
3.3	Blockchain: funzionamento e caratteristiche	35
3.3.1	Distributed ledger technology (DLT)	36
3.3.2	Livello di Consenso	37
3.3.3	Interfaccia di Calcolo	40
3.3.4	Governance	41
3.4	Classificazione delle blockchain	41
3.5	Ethereum e gli smart contract	42
3.5.1	Standard ERC	44

3.6	NFT	45
4	I principali esponenti del Metaverso	47
4.1	The Sandbox	47
4.1.1	Overview	47
4.1.2	Ecosistema	49
4.1.3	SAND token e LAND	50
4.2	Decentraland	53
4.2.1	Overview	53
4.2.2	Ecosistema	57
4.2.3	MANA token e LAND	59
4.3	OVR	60
4.3.1	Overview	60
4.3.2	Ecosistema	62
4.3.3	OVR Token e OVR Lands	67
5	Costruire il Metaverso	70
5.1	Architettura di un Metaverso	70
5.2	Creazione di una Mappa	75
5.2.1	Overview	76
5.2.2	Struttura del Progetto ed Analisi Tecnica	78
5.3	Var Group sul Metaverso	90
5.3.1	Overview	91
5.3.2	Strumenti utilizzati	93
6	Risultati e sviluppi futuri	96
7	Figure e altro	98
8	Bibliografia e Sitografia	100

1 Introduzione generale

“Il Metaverso permetterà alle persone di replicare o migliorare le loro attività. Questo potrebbe accadere trasportando o estendendo le attività fisiche in un mondo virtuale o trasformando quello fisico”, dice Marty Resnick, analista e vicepresidente di Gartner¹. L’azienda prevede un incremento esponenziale degli utenti sul Metaverso, affermando come secondo alcune analisi entro il 2026, circa il 25% degli esseri umani sfrutteranno tale tecnologia per svariati motivi, quali lavoro, shopping, socializzazione, intrattenimento e molto altro. Ma che cos’è un Metaverso?

1.1 Definizione di Metaverso

Il Metaverso è uno spazio aperto virtuale collettivo, creato dalla convergenza della realtà fisica e digitale potenziata virtualmente, è così che il filosofo Luciano Floridi ha definito tale tecnologia. In altre parole, è indipendente dal dispositivo e non è di proprietà di una singola entità centralizzata. Presenta un’economia virtuale indipendente, abilitata da valute digitali e token non fungibili, gli NFT. Un Metaverso, per poter funzionare, richiede un’innovazione combinatoria, nella quale varie tecnologie devono coesistere e collaborare fra di loro. Le tecnologie che contribuiscono a tale processo includono la realtà virtuale (VR) ed aumentata (AR), gli HMD², i sistemi IoT, il 5g, l’intelligenza artificiale e molto altro. Possiamo immaginare un Metaverso come una versione successiva di internet, dove ogni utente può condividere qualcosa con gli altri, come i vari siti web fanno attualmente. Le potenzialità di un Metaverso sono dunque illimitate, per fare degli esempi potremmo comprare abiti ed accessori per il nostro avatar virtuale, acquistare terreni digitali, partecipare ad eventi virtuali, come concerti o mostre, potremmo fare shopping all’interno di centri commerciali virtuali o utilizzare delle aule virtuali per scopi lavorativi, di studio o ludici, potremmo acquistare dei beni digitali come quadri, oggetti di collezioni, o potremmo interagire con gli altri utenti e divertirci insieme a loro.

¹ Società per azioni multinazionale che si occupa di consulenze strategiche

² Head Mounted Display, casco per la visione di contenuti in VR o AR

Ci si aspetta quindi che un Metaverso fornisca modelli e opportunità di business persistenti, decentralizzati, collaborativi ed interoperabili che permetteranno alle organizzazioni di espandere il business digitale.

Matthew Ball definisce i mondi paralleli generati nel Metaverso come *“una rete di esperienze ad applicazioni interconnesse tramite dispositivi e prodotti, strumenti ed infrastrutture”* e proprio come questa rete, che utilizza una varietà di tecnologie per funzionare, anche il crescente spostamento verso il web 3.0 e i sistemi decentralizzati come la Blockchain, stanno contribuendo a plasmare il Metaverso.

Spesso il concetto di Metaverso viene confuso con quello di Realtà Virtuale. La sostanziale differenza fra i due concetti è che, se nelle applicazioni in Realtà Virtuale il principale obiettivo è quello di guardare o manipolare gli oggetti all'interno dell'esperienza virtuale, all'interno del Metaverso le persone possono esplorare, connettersi ed interagire fra di loro. Inoltre, il Metaverso non si limita ad un'esperienza solamente immersiva, ovvero tramite un caschetto di realtà virtuale, bensì è possibile accedervi anche da altri tipi di dispositivi quali pc o smartphone.

Un'altra differenza sostanziale sta nel concetto di persistenza, fondamentale quando parliamo di Metaverso. Un Metaverso esiste indipendentemente dalla presenza di qualcuno al suo interno. Supponendo di interagire con un'esperienza metaverse, se modificassimo, per esempio, la posizione di un oggetto, al successivo accesso riavremo quell'oggetto nella stessa posizione in cui l'avevamo lasciato a meno di modifiche fatte da altri utenti. In altre parole, il mondo virtuale del Metaverso è persistente, ovvero uguale per tutti i suoi utenti e modificabile in conseguenza alle azioni di questi ultimi. Questo concetto non è presente nelle applicazioni in Realtà Virtuale in quanto comporterebbe un mantenimento delle informazioni in memoria che andrebbero poi distribuite in real-time a tutti gli utenti che accedono all'esperienza.

Negli ultimi anni la parola Metaverso è diventata molto di tendenza sia nel mondo delle criptovalute che nell'industria tecnologica in generale. Un fattore

fondamentale che ha creato molto hype intorno al Metaverso è sicuramente stato il rebranding di Facebook a Meta. Molte aziende hanno così deciso di lanciarsi all'interno di questo nuovo mondo virtuale per cavalcare l'onda del progresso. Proviamo quindi a capire le origini di questo concetto.

1.2 La Storia del Metaverso

1.2.1 Dove tutto è iniziato

La storia del Metaverso risale incredibilmente ai primi anni di Internet. Già nel 1932, Antonin Artaud ne *"Il teatro della crudeltà (primo manifesto)"* introdusse il concetto di "realtà virtuale". La parola Metaverso invece fu utilizzata per la prima volta dallo scrittore Neal Stephenson nel 1992 nel suo romanzo di fantascienza *Snow Crash*, in questo caso la parola Metaverso risultava essere un acronimo fra le parole "meta" e "universo". Stephenson immaginava il Metaverso come un mondo, basato sulla realtà virtuale, che avrebbe sostituito internet per come si conosceva. In questo mondo virtuale gli utenti avrebbero potuto, tramite l'utilizzo di avatar, esplorare tutto il regno digitale a loro disposizione. L'idea era nata dalla necessità di alienarsi dal mondo reale, dominato da forze mercenarie, da mafie aziendali e da guardie private che proteggevano la comunità dei sovrani dopo l'avvento di un collasso economico globale. Molte delle visioni del Metaverso trattate nel romanzo da Stephenson sono già diventate (o lo diventeranno presto) realtà. Sicuramente questa visione futuristica di Stephenson è stata anche condizionata dall'avvento del *World Wide Web*, inventato da Tim Berners-Lee nel 1989.

Già nel 1992 si iniziò a parlare di *Proof of Work*, questo veniva utilizzato nel contesto della sicurezza informatica come metodo per prevenire lo spam nelle e-mail. Ai giorni nostri questa tecnica viene utilizzata per verificare e legittimare le transazioni su blockchain.

Nel 1998 l'ingegnere informatico Wei Dai introduce per la prima volta *b-money*, una criptovaluta distribuita e decentralizzata. Anche Satoshi Nakamoto, coniatore di *Bitcoin* ha costruito il suo lavoro su progetti precedenti, come appunto b-money.

Qualche anno dopo, nel 2011, Ernest Cline pubblicò *Ready Player One*, un romanzo, da cui fu tratto l'omonimo film di Stephen Spielberg nel 2018, dove ci si immergeva in una realtà distopica, simile a *Snow Crash*. Il lettore veniva immerso in un mondo in cui gli esseri umani giocavano un gioco in realtà virtuale basato sul Metaverso dal nome OASIS, dove l'obiettivo principale era quello di riuscire a scappare dal caos presente sulla terra.

Nell'ultimo decennio, a causa della sempre più crescente popolarità, è comune trovare il concetto di Metaverso nella cultura pop, basti pensare a film come *Matrix* o l'episodio *Fifteen Million Merits* di *Black Mirror*, dove i cittadini vanno su una bicicletta fissa per guadagnare crediti utili a partecipare successivamente ad un talent show, attraverso i loro avatar personali.

1.2.2 Dalle prime applicazioni ai giorni nostri

Nei primi anni 2000 il boom del gioco multiplayer *World of Warcraft* ha fornito ai programmatori una grossa opportunità per iniziare a familiarizzare con un mondo virtuale dove più utenti potevano interagire fra di loro, siamo però ancora ben lontani dal concetto di Metaverso, dato che stiamo parlando di un sistema centralizzato. Ma il vero precursore degli attuali Metaversi risale al 2003, quando Philip Rosedale pubblicò *Second Life*, ritenuto la vera prima applicazione del concetto di Metaverso. In questo mondo virtuale non vi è una storia da completare o degli obiettivi da raggiungere, i giocatori hanno invece la libertà di poter creare i propri contenuti, interagire con gli altri utenti attraverso i propri avatar digitali, così come esplorare tutto il mondo che *Second Life* offre, oltre che partecipare alle diverse attività che gli altri utenti sviluppano e propongono. Negli anni successivi sono stati sviluppati diversi videogiochi che presentavano il loro mondo virtuale, per fare degli esempi possiamo pensare a *There.com* che presentava un mondo virtuale molto simile a quello di *Second Life*.

Una delle principali novità che questa piattaforma ha introdotto è stata l'inserimento di una propria valuta digitale, chiamata *Therebucks*. Successivamente un altro chiaro esempio di applicazione della tecnologia del

Metaverso è Roblox, lanciato nel 2006, permetteva di sviluppare e giocare a giochi creati dagli altri utenti presenti sulla piattaforma.

Un altro esempio molto significativo che ci introduce agli attuali Metaversi è sicuramente *Minecraft*, videogioco ancora oggi molto popolare in tutto il mondo. In questo caso i giocatori possono esplorare mondi 3D basati su una grafica voxel, dalla quale possono estrarre materie prime, creare elementi di gioco e costruire virtualmente tutto, è possibile, per esempio, realizzare ambientazioni simili alla Terra di Mezzo del Signore degli Anelli o un processore a 8bit in grado di eseguire giochi.

Uno dei giochi più moderni che si avvicina molto al concetto di Metaverso è invece *Fortnite*, sviluppato da Epic Games nel 2017, è tra i giochi multiplayer Free-to-Play di maggior successo fino ad oggi. Di base la modalità di gioco principale di Fortnite è quella del Battle Royale³, ma Epic Games ha ampliato il suo ecosistema con molte altre attività e caratteristiche. Due degli eventi sicuramente degni di nota avvenuti all'interno di Fortnite sono i concerti virtuali in collaborazione con Travis Scott e Marshmello che in totale sono stati seguiti da 23 milioni di utenti.

1.2.3 Attualità

Negli ultimi anni il clamore intorno ai Metaversi è aumentato a dismisura, questo ha fatto sì che anche lo sviluppo e le applicazioni inerenti a questa nuova tecnologia hanno subito un grosso aumento. Uno dei fattori fondamentali che ha portato a ciò è sicuramente l'ascesa della tecnologia decentralizzata che è diventata fondamentale in questo campo. Oltre agli smart contract, dApps, DAOs e DeFi e naturalmente le criptovalute stesse, i token non fungibili (NFT) sono diventati un'innovazione chiave all'interno dello spazio crypto. Dato che sempre più persone si rendono conto dei benefici degli NFT, che permettono di identificare in modo univoco e non replicabile un asset digitale o fisico, i progetti sul Metaverso e sulla blockchain hanno conferito agli NFT un maggior numero di casi d'uso. Convertendo la maggior parte degli elementi di gioco in NFT sulla blockchain, gli sviluppatori possono

³ Modalità di videogioco con obiettivo la sopravvivenza in un'arena player versus player

dare agli utenti la possibilità di possedere, in modo esclusivo, tutti i loro beni. Inoltre, l'implementazione di asset digitali nativi e il modello Play-to-Earn (P2E) hanno permesso ai vari team di creare vere e proprie economie complete all'interno di mondi virtuali realizzati. Di conseguenza gli utenti non solo possono comprare e vendere gli asset digitali presenti all'interno del Metaverso alla quale partecipano, ma possono anche guadagnare partecipando attivamente all'ecosistema. Alcuni progetti si sono spinti oltre, combinando il modello Play-to-Earn e gli NFT con le attività popolari della DeFi (ad esempio il performance farming e la partecipazione attiva) per espandere le opportunità di guadagno per gli utenti, quindi inevitabilmente creare molta più audience e rendere le loro piattaforme più sostenibili a lungo termine. Molte aziende hanno quindi approfittato delle tecnologie già presenti sul mercato, della popolarità del Play-to-Earn e del grosso hype che si è creato intorno ai mondi virtuali, per creare i loro Metaversi anche prima che lo facessero le grandi aziende nell'ambito tecnologico, come ad esempio Facebook. I due grossi pionieri che hanno cavalcato, nel miglior modo quest'onda, sono sicuramente Decentraland e Sandbox. A differenza delle soluzioni proposte dalle grandi aziende, progetti come quelli sopra citati, sfruttano la decentralizzazione e la tecnologia Permissionless per creare i loro mondi virtuali. Dunque, ogni utente presente all'interno del mondo è responsabile della governance del mondo stesso e non esiste una società centralizzata. Approfondiremo successivamente questi due Metaversi.

Un altro aspetto fondamentale, che ha sicuramente contribuito al grosso boom di questa tecnologia, è stata la pandemia COVID-19, in quanto quest'ultima ha reso l'interazione di persona molto più difficile rispetto a prima, questo ha invogliato le persone a cercare modi alternativi per comunicare con gli altri, fare il loro lavoro e gestire le loro finanze. Per questo motivo i Metaversi hanno assunto un ruolo fondamentale nell'ospitare le versioni virtuali di eventi e attività come concerti, workshop e conferenze.

1.2.4 Principali competitor e prospettive future

Attualmente la principale azienda nello spazio del Metaverso è Meta, basti pensare a come la notizia del rebranding di Facebook ha alimentato

tantissimo hype intorno ai mondi virtuali. Non conosciamo esattamente quali siano i piani esatti della società per il suo prossimo mondo digitale, ne abbiamo solo un concetto vago attualmente, ma Meta ha sicuramente stabilito una buona infrastruttura negli ultimi anni che potrebbe servire come quadro generale in futuro. Oltre ai miliardi di utenti presenti sui vari social, come Facebook ed Instagram, Meta possiede Oculus, che produce dispositivi VR che sicuramente potranno aiutare nel migliorare l'esperienza degli utenti nel Metaverso. Nel dicembre 2021, la società ha lanciato Horizon Worlds, un videogioco in realtà virtuale per Oculus Rift S e Oculus Quest dove i giocatori possono costruire, interagire ed esplorare insieme.

Anche Microsoft si è interessata molto a questa nuova tecnologia, attualmente sappiamo che la prima offerta a tema Metaverso dell'azienda sarà Microsoft Mesh, una piattaforma di comunicazione e collaborazione virtuale. Un'altra grossa azienda che sta investendo molto su questa nuova tecnologia è Epic Games che, con un investimento da circa 1 miliardo di dollari, è pronta a sviluppare un Metaverso completo introno a Fortnite tramite il supporto VR e AR. Infine, anche Google si sta molto interessando all'argomento, più nel dettaglio l'azienda si sta concentrando principalmente sulla connessione fra mondo fisico e digitali attraverso la realtà aumentata, utilizzando i propri dispositivi quali Google Glass.

Attualmente, anche se sono già stati lanciati diversi Metaversi, siamo ancora in una fase iniziale, ci si aspetta quindi che lo sviluppo in questo settore acceleri in modo significativo nei prossimi mesi, per rendere questa nuova tecnologia accessibile a tutti, sempre migliore e con nuove interessanti opportunità.

1.3 Gli utenti nel Metaverso

Abbiamo quindi capito cos'è un Metaverso e quale è la sua storia. Possiamo quindi riassumere il concetto di Metaverso come un mondo virtuale aperto, persistente ed in real-time costruito attraverso le tecnologie del web 3.0, come la blockchain, gli smart contract, gli NFT e le criptovalute. È molto importante sottolineare come i concetti sopracitati sono necessari per poter

parlare di Metaverso, difatti vi è molta confusione su quale tipologia di applicazione in VR o AR rispecchiano il concetto di Metaverso. Spesso, infatti, le applicazioni che sfruttano queste tecnologie vengono definiti dei Metaversi, ma non è così in quanto queste applicazioni non sono permanenti; quindi, una volta chiusi, i dati della sessione non vengono conservati e soprattutto spesso non presentano interoperabilità, poiché sono isolate e non è possibile in alcun modo connetterli ad altre applicazioni esterne. Attualmente quindi, come sottolineato quando abbiamo parlato della storia, siamo in un periodo di forte impulso, dove la tecnologia sta prendendo sempre più piede, ma è comunque ancora una fase embrionale. Allo stato attuale il principale motore di business dei videogiochi online è l'acquisto e la vendita di assets di gioco, in particolare di skin. Anche nel Metaverso il business principale si basa sulla vendita di oggetti virtuali, questo è favorito dall'interoperabilità. Rendendo ogni asset un NFT, i beni digitali possono essere utilizzati su più piattaforme, vendute e affittate in modo semplice fra i vari utenti. Oltre alle esperienze ludiche, un altro aspetto da tenere in considerazione è il lato della produttività che il Metaverso potrebbe offrire, si stanno sempre più creando applicazioni di supporto all'apprendimento, allo smart working e ai meeting, che migliorano di molto gli attuali sistemi flat come Zoom o Teams. Si immagina infatti di potersi connettere con altre persone magari geograficamente molto distanti e per esempio entrare in una fedele riproduzione della casa di un nostro amico che si trova a 10000 km di distanza.

Accedere ad un Metaverso è semplice, non sono necessarie particolari competenze o conoscenze informatiche, basta avere un dispositivo come uno smartphone, un laptop, un visore e scaricare o connettersi al corretto applicativo. Ogni applicazione basata sul Metaverso ha le proprie regole d'accesso, per esempio l'idea di Zuckerberg in Meta è quello di avere un avatar che ti permetta di muoversi infinitamente attraverso un teletrasporto. L'idea rivoluzionaria di Meta si basa sul concetto di ricreare un social basato su questa tecnologia che renda l'esperienza utente e le interazioni umane migliori rispetto a quelle che siamo abituati attualmente ad utilizzare. È possibile ottenere questo grazie alla sensazione di Presenza, con la quale

l'utente si sente effettivamente immerso nel Metaverso. Per fare un altro esempio pratico di accesso al Metaverso possiamo far riferimento a Decentraland, in questo caso per poter accedere all'esperienza è necessario avere un wallet, ovvero un portafoglio digitale che serve per contenere i token MANA, criptovaluta utilizzata nell'economia di questo mondo virtuale. Se consideriamo gli utenti come Community, quello che ricercano è sicuramente un'esperienza innovativa che dia nuovi stimoli. Il ruolo della Community è importantissimo nell'ecosistema in quanto le persone, pur non essendo fisicamente insieme, possono confrontarsi online in modo semplice. È per questo che la sfida principale del Metaverso e dei brand, che decidono di prenderne parte, è quello di riuscire a catturare la Community.

2 La tecnologia alla base del Metaverso

In questo capitolo verranno analizzate due delle tecnologie principali che stanno alla base del Metaverso ovvero la Realtà Aumentata e la Realtà Virtuale, analizzeremo la loro definizione formale, parleremo della loro storia e di come sul lato pratico queste funzionano.

2.1 Realtà Aumentata

Riuscite ad immaginare una tecnologia talmente potente da offuscare il confine fra il mondo fisico nella quale viviamo e il mondo digitale? Tale tecnologia è la realtà aumentata, in grado sicuramente di aumentare il livello di produttività ed innovazione in ogni campo. Già tantissime aziende hanno iniziato ad utilizzare tale tecnologia, basti pensare ad Apple che l'ha introdotta come caratteristica fondamentale sugli iPhone, a Hyundai e Toyota che la utilizzano nelle automobili o ai vari brand di abbigliamento come Gucci o Adidas che la utilizzano per permettere di provare virtualmente un vestito. Ma che cosa è la Realtà Aumentata?

2.1.1 Definizione

Possiamo definire la realtà aumentata come una vera e propria tecnologia. Risulta difficile quindi dare una definizione ben precisa di essa. Possiamo definire l'AR come una tecnologia immersiva che sovrappone un'immagine generata dal computer su delle superfici fisiche del nostro mondo reale,

visualizzabili tramite appositi strumenti quali visori, smartphone, tablet ecc. Questa tecnologia ha sicuramente delle potenzialità incredibili in quanto le informazioni che possiamo generare arricchiscono il mondo .

Una prima definizione che descrive tale tecnologia è quella data da Paul Miligram e Fumio Kishino, che definirono il termine "realtà mista" come "*tutto ciò che è contenuto tra gli estremi del continuo della virtualità*" [1], gli estremi alla quale si riferivano erano la realtà fisica nella quale viviamo, senza la presenza di elementi virtuali aggiuntivi, e la realtà virtuale, ovvero un mondo completamente modellato tramite software 3D che estranea l'utente dal mondo reale. In questa definizione l'AR si posiziona più verso il mondo reale che quello virtuale, in quanto si ha una sovrapposizione di oggetti virtuali nel mondo fisico.

Se volgiamo però una definizione più precisa, ci possiamo riferire a quella data da Azuma, uno dei principali ricercatori nel campo AR, descrive questa tecnologia come: "*. . . a technology that allows computer-generated virtual imagery information to be overlaid onto a live direct or indirect real-world environment in real time*". [2]. L'AR quindi funziona come una sorta di ponte tra il reale ed il virtuale in modo da creare una sorta di continuità fra di essi. Nelle applicazioni realizzate tramite questa tecnologia il mondo reale prevale su quello virtuale, gli oggetti virtuali sono posizionati coerentemente con il mondo fisico nella quale sono inseriti e l'applicazione gira in tempo reale oltre ad essere interattiva, sarà dunque possibile interagire con i modelli 3D al suo interno.

Ma quali sono gli elementi virtuali alla quale ci riferiamo? Tali elementi possono essere sia delle riproduzioni fedeli di oggetti reali, sia dei modelli totalmente immaginari, dei testi o dei menù che contribuiscono a migliorare o rendere più efficaci ciò che ci circonda.

Uno dei principali vantaggi delle applicazioni in AR è che serve poca potenza di calcolo rispetto a quella presente negli attuali dispositivi tecnologici, questo è sicuramente dovuto al fatto che è necessaria la presenza di molti meno

modelli digitali rispetto a quelli necessari in una classica applicazione per pc o smartphone.

2.1.2 Storia

L'AR non è una tecnologia creata di recente, infatti già nel 1968 Ivan Sutherland, ricercatore e professore presso l'università di Harvard, creava il primo head-mounted display (HMD) con ottiche see-through, denominandolo come Spada di Damocle [3]. Questo dispositivo non era altro che un caschetto che presentava un display ottico per ogni occhio. Tale display permetteva di visualizzare solamente semplici disegni in wireframe, ovvero un particolare tipo di disegno dove vengono visualizzati solamente i vertici e le linee dell'oggetto disegnato. Uno dei principali problemi che questo dispositivo aveva era il grosso peso, era infatti impossibile da mantenere in testa senza un sostegno, veniva dunque fissato sul soffitto attraverso appositi bracci. **(figura 1)**



Figura 1 Spada di Damocle - Ivan Sutherland, 1968

Nel 1975 Myron Kruger realizzava un laboratorio, dal nome Videoplace, nel quale più utenti potevano interagire con oggetti virtuali senza l'utilizzo di dispositivi esterni, quali occhiali o HMD, tale tecnologia venne definita realtà artificiale e fu realizzata grazie all'utilizzo di video-proiettori.

Ma la vera coniazione del termine "Realtà Aumentata" può essere attribuita a Tom Caudell, ricercatore presso la Boeing. Nel progetto dell'azienda veniva

utilizzato un casco che presentava due visori, i quali permettevano la visualizzazione di dati virtuali relativi alle procedure di volo e di atterraggio. La grossa novità era appunto che questa tecnologia non ostruiva in alcun modo la visione del mondo reale, ma consentiva di "aumentare" la visione del mondo fisico con informazioni aggiuntive. Questa tecnologia fu poi portata avanti da Tom Caudell e David Mizell che nel 1992 svilupparono un nuovo head-mounted display per applicazioni industriali; in particolare veniva utilizzato nel supporto all'assemblaggio delle parti di cavetteria [4]. Nel 1993 fu realizzato Chamelon il primo dispositivo hand-held per AR, realizzato da George Fitzmaurice, questo dispositivo permetteva di visualizzare specifiche informazioni in base alla posizione dell'utente. In quegli anni la potenza di calcolo dei dispositivi non era ancora abbastanza per avere un hardware proprio; quindi, il dispositivo si appoggiava ad una workstation. Nel 1995 fu progettato il primo dispositivo che funzionava tramite l'utilizzo di marker. Il dispositivo, chiamato NaviCam, progettato da Jun Rekimoto e Nagao, presentava una fotocamera integrata che riusciva a rilevare i marker presenti nell'ambiente e mostrava delle informazioni a loro associate [5]. Tale tecnologia fu portata avanti da Rekimoto che nel 1996 realizza il primo dispositivo AR in grado di utilizzare marker 2D. Nel 1996 Schmalstieg ed i suoi collaboratori realizzarono la prima esperienza multiutente in AR, nella quale grazie all'utilizzo di un head-mounted display le immagini venivano renderizzate tenendo conto della prospettiva di visualizzazione dell'utente che lo indossava. Nel 1997 Azuma introducesse l'attuale definizione di AR, inoltre nello stesso anno la Columbia University creò la Touring Machine [6], un dispositivo che mostrava all'utente informazioni addizionali, basate sugli oggetti reali presenti nell'ambiente circostante, che riguardavano ciò che l'utente stesso stava osservando, ad esempio il nome degli edifici, delle strade o cenni storici legati ad essi. Già a ridosso del 2000 venne creato l'ARToolKit [7], una libreria open-source, realizzata da Hirokazu Kato, che permette la creazione di applicazioni in AR, questa include una libreria di tracciamento di fiducial marker che permettono la sovrapposizione di oggetti virtuali e reali. Tale libreria è tutt'ora disponibile su molte piattaforme ed ancora molto utilizzata. Nel 2003 Wagner e Schmalstieg realizzò il primo dispositivo

hand-held per AR totalmente integrato, dunque senza il bisogno di una workstation esterna di supporto. Negli ultimi anni l'industria videoludica ha mostrato molto interesse nell'utilizzo della tecnologia AR per la realizzazione di videogiochi. Il primo esempio è stato realizzato nel 2003 con *Mozzies*, il primo videogioco per dispositivi mobili in AR. Questo gioco consisteva nel colpire delle zanzare, modellate digitalmente, che venivano sovrapposte al mondo reale catturato tramite fotocamera del dispositivo. Per molti anni questa tecnologia è rimasta in sordina a causa dell'elevato costo computazionale necessario per il suo utilizzo e dunque per gli elevatissimi costi che ne conseguivano. Tuttavia, grazie soprattutto all'incremento esponenziale della potenza di calcolo presente sui nostri dispositivi tecnologici, l'AR è diventata una tecnologia di consumo. Basti pensare che nel 2014 Google ha rilasciato la prima versione beta di Google Glass, nel 2016 anche Microsoft ha presentato il suo primo dispositivo AR chiamato HoloLens, prodotto destinato soprattutto all'industria, nello stesso anno Meta rilascia Meta 2 Developer Kit. Ma l'esempio di gran successo delle applicazioni AR è sicuramente quello ottenuto da *Pokemon Go* (**figura 2**), rilasciato nel 2016, ha totalizzato un bacino di utenza di circa 500 milioni di utenti generando fatturati che superano i 5 miliardi di dollari. Nel 2017, Google lancia ARCore, ed Apple annuncia ARKit, SDK utili per la realizzazione di applicazioni in AR sui propri dispositivi mobili.



Figura 2 - Pokemon GO - Ninantic, 2016

2.1.3 Tipologie di dispositivi

I dispositivi di realtà aumentata possono essere classificati in quattro tipi [8]. Ognuno di queste categorie di dispositivi presenta dei vantaggi e degli svantaggi.

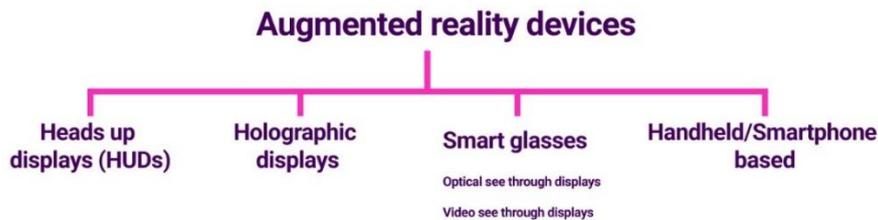


Figura 3 - Tipi di sistemi AR in commercio

2.1.3.1 Heads up displays (HUDs)

Un HUD è un display trasparente simile ad una pellicola su cui un'applicazione AR trasmette i suoi dati. Il vantaggio degli HUD è che non richiedono all'utente di cambiare il suo focus su un mondo totalmente virtuale, ma si integrano, nell'ambiente fisico, gli oggetti virtuali. Il display alla base degli HUD è pressoché trasparente, essendo le informazioni proiettate con un contrasto visivo e cromatico che si adatta all'ambiente circostante. L'informazione inoltre è proiettata con fuoco all'infinito, in modo tale da non dover, ogni volta che si leggono le informazioni presenti sul display, rimettere a fuoco la propria vista. Il funzionamento è quindi incentrato sul proiettare l'immagine in un display trasparente, che non ostruisce la visuale dell'utilizzatore, definito come Combiner.

Un HUD è costituito da tre elementi fondamentali:

- **Unità di proiezione:** che tipicamente è un collimatore ottico, ovvero una lente convessa o uno specchio concavo ed un display collocato nel suo fuoco.
- **Combiner:** in genere un pezzo di vetro piatto inclinato e semitrasparente che si posiziona direttamente davanti il visore e che renderizza l'immagine

proiettata dal proiettore in modo da combinare tale immagine e il mondo fisico che circonda l'utente.

- **Computer:** fornisce l'interfaccia fra l'unità di proiezione e i dati da visualizzare.

Ad oggi esistono due tipi differenti di HUD, quelli fissi, in cui l'utilizzatore guarda attraverso uno schermo trasparente montato su un supporto fisso, che può essere per esempio il pannello strumenti dell'aereo o il cruscotto di un veicolo; e quelli integrati nel casco in cui il display viene installato direttamente sulla visiera o su un mirino ottico che si muove insieme alla testa del pilota.

Le prime applicazioni di questa tecnologia sono state realizzate per gli aeromobili in quanto si permetteva ai piloti di visualizzare in modo semplice e non intrusivo i dati di volo. Successivamente tale tecnologia è stata implementata sulle moderne automobili che trasmettono sul parabrezza delle informazioni utili al guidatore. Molti produttori di auto come BMW, AUDI, Volvo ecc. hanno HUD sui cruscotti del veicolo. Questi forniscono informazioni all'utente come la navigazione, la lettura del tachimetro, la distanza della destinazione, gli ostacoli sulla carreggiata, ecc.

2.1.3.2 Display Olografici

Un display Olografico è un tipo di display 3D che utilizza la diffrazione della luce per poter visualizzare un'immagine tridimensionale. A differenza degli HUD, non hanno bisogno di alcuna superficie per proiettare le loro immagini, ma fanno uso di proiettori in grado di ricreare le immagini direttamente sulle superfici degli oggetti. La maggior parte dei dispositivi attualmente in commercio utilizza un laser come fonte di luce, questo laser viene proiettato su una scena che viene poi riflessa su un apparecchio di registrazione. Un'altra tecnologia utilizzata si basa sull'elettrografia, i display elettrografici sono display che trasmettono i dati delle immagini utilizzando un risonatore elettromagnetico. Questi segnali vengono elaborati da un modulatore ottico-acustico e convertiti in un'immagine visualizzata su display RGB. Un'altra tecnologia è quella dell'olografia a parallasse completa, attraverso questo

processo l'immagine risultante fornirà la stessa prospettiva di una scena a tutti gli spettatori indipendentemente dall'angolo di visione. Infine, la tecnologia MEMS consente ai display olografici di incorporare parti mobili molto piccole di design, questa è sicuramente la miglior tecnologia fra quelle discusse in quanto l'immagine risultante risulta nitida.

Questa tecnologia ha sicuramente dei vantaggi rispetto alle altre tecnologie AR, in quanto non è necessario indossare o supportare alcun tipo di dispositivo, che può inevitabilmente portare ad affaticamento dell'utilizzatore dopo sessioni lunghe. Un altro vantaggio è il campo visivo in quanto questo risulta molto più ampio. Ovviamente questi dispositivi presentano anche degli svantaggi, innanzitutto la calibrazione è molto macchinosa e va effettuata ogni qualvolta ci si sposta in un nuovo ambiente, inoltre, il più grosso problema è invece la bassa luminosità di questi dispositivi che limita il loro utilizzo solamente all'interno di ambienti chiusi.

2.1.3.3 Wearable

Quando si parla di dispositivi wearable, definiti delle volte come Smart glasses (**Figura 3**), ci riferiamo a quei dispositivi indossabili come gli head-mounted display, utilizzabili come normali occhiali con il vantaggio di avere le mani libere. Questa tecnologia però ha lo svantaggio di essere abbastanza scomoda, in quanto gli occhiali in questione devono contenere al loro interno tutta la circuiteria necessaria per funzionare e risultano stancanti da utilizzare se indossati per sessioni di parecchie ore. Il concetto di smart glasses è stato introdotto nel 2012 da Oakley. Colin Baden, CEO dell'azienda, sta lavorando per proiettare le informazioni direttamente sui vetri degli occhiali senza bisogno di alcuna pellicola aggiuntiva. Questa tecnologia utilizza la proiezione della retina per proiettare immagini o dati sulla lente di vetro. Sono forse la forma di realtà aumentata più avanzata attualmente sul mercato.

Nel caso dei Wearable abbiamo due differenti tipologie: gli Optical See-through e i Video See-through.

Per Optical see-through intendiamo un display trasparente o semitrasparente che permette all'utente di vedere il mondo reale come attraverso un

banalissimo vetro, anche se filtrato dagli strati che compongono lo schermo. Considerando la qualità dell'immagine risultante, poiché l'utente vede direttamente il mondo fisico, non vi è alcuna perdita di definizione. Gli svantaggi principali di tale tecnologia sono invece il campo visivo limitato e la scarsa luminosità e contrasto, che rendono tali dispositivi inadeguati per un utilizzo all'aperto o in ambienti troppo illuminati. La tecnologia basata su Video see-through presenta due fondamentali vantaggi, in primis il costo di realizzazione è molto inferiore, inoltre è molto più semplice realizzare dispositivi basati su questa. Il mondo fisico in questo caso viene catturato tramite delle videocamere e combinato con l'immagine degli oggetti virtuali, il render della fusione delle due componenti viene poi visualizzato dall'utente tramite dei display. Il vantaggio principale è sicuramente che questi dispositivi sono utilizzabili in qualunque situazione di luce in quanto l'utente non ha una visione diretta del mondo esterno, è quindi facile ottenere un'immagine ricalibrata per risultare nitida e chiara all'occhio dell'utilizzatore. Di contro però la qualità dell'immagine visualizzata è imparagonabile rispetto all'altra soluzione, in quanto verrà visualizzata su dei display che stanno a pochi centimetri dagli occhi dell'utente che potrebbe notare i pixel presenti. Oltre a ciò, l'utilizzo di tale tecnologia potrebbe portare nausea ad alcuni utenti a causa dell'effetto di parallasse, infatti, per limiti fisici, la videocamera non può essere posizionata in corrispondenza degli occhi e quindi tende a differire rispetto alla visione normale, si sta comunque lavorando per cercare di limitare tale problema.

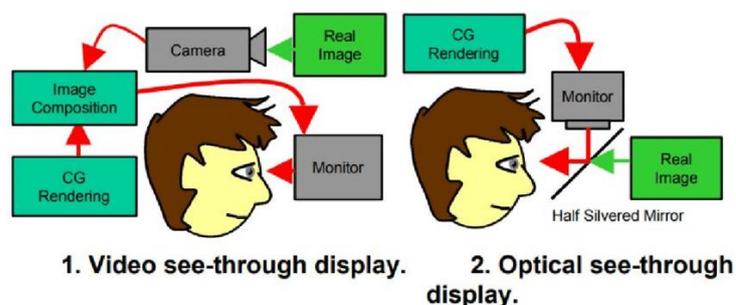


Figura 4 - Differenza di funzionamento fra le tecnologie Video see-through ed Optical see-through

2.1.3.4 Hand-held

Con hand-held si intende un dispositivo che deve essere tenuto in mano, per esempio un tablet o uno smartphone. Questo è sicuramente il supporto tecnologico per l'AR alla quale tutti hanno accesso. Lo smartphone presenta tutti i componenti hardware necessari per realizzare un'esperienza in realtà aumentata: la videocamera, necessaria per catturare il mondo fisico che ci circonda, i sensori come giroscopio ed accelerometro, utili per tracciare la posizione e l'orientamento del dispositivo, un display dove renderizzare l'immagine catturata dalla fotocamera e mixarla con gli oggetti modellati digitalmente ed infine un processore ed una memoria con abbastanza potenza da far girare un'applicazione in AR senza particolari problemi. Analizzando il processo necessario per trasformare uno smartphone in un dispositivo AR partiamo dall'utilizzo della videocamera che cattura in real-time la scena nella quale ci troviamo, allo stesso tempo i vari sensori catturano la posizione dell'utente e la rotazione dello smartphone. Infine, il processore si occupa di renderizzare gli oggetti virtuali e sovrapporle con le immagini catturate dalla videocamera. Una volta mixate, l'immagine finale risultante verrà visualizzata in real-time sul display del dispositivo (**figura 5**).

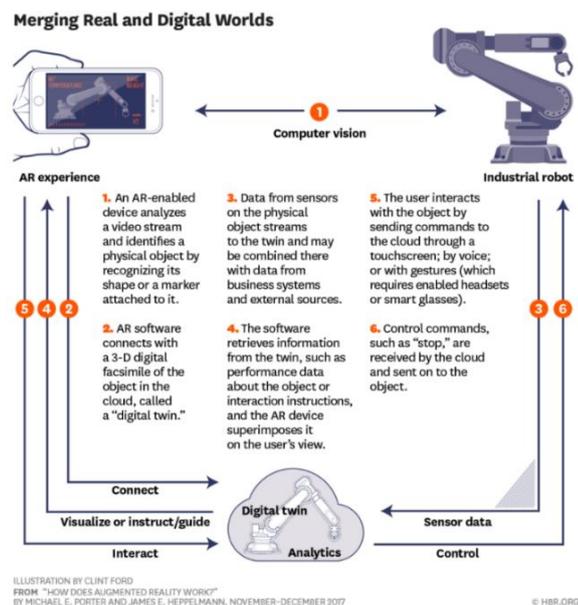


Figura 5 - Funzionamento sistema AR

2.1.4 Sistemi di tracciamento

Alla base della realtà aumentata abbiamo capito che ci deve necessariamente essere una armonia fra mondo fisico e mondo virtuale. Tale armonia richiede che tutto il sistema AR, includendo sia software che hardware, soddisfi determinati standard di accuratezza e robustezza, sia nei sistemi di tracciamento del mondo fisico, sia nella riproduzione e nel posizionamento degli oggetti virtuali e sia nella velocità di risposta agli input dell'utente.

Partiamo quindi dall'analisi del sistema di tracciamento. Da questo sistema dipende il corretto posizionamento degli oggetti virtuali nel mondo reale, un allineamento non ottimale può rendere l'applicazione a tutti gli effetti inutilizzabili. Ma come è possibile posizionare correttamente un oggetto? I sensori che entrano in gioco in questo caso devono misurare con molta accuratezza gli spazi reali, le misure degli oggetti presenti e le rotazioni che il dispositivo che si sta utilizzando subisce. Tali parametri vengono riportati sulla camera della scena 3D, a questo punto il motore grafico esegue il rendering dell'oggetto da inserire, l'immagine viene quindi processata, ridimensionata e sovrapposta alla realtà. Per quanto riguarda il posizionamento degli oggetti virtuali, le soluzioni più comuni sono basate su tracker ottici, nei quali le immagini catturate dalle camere sono processate in riferimento ad oggetti reali presenti nella scena, ad esempio marker, che permettono di determinare la posizione e la dimensione rispetto all'ambiente nella quale si trovano. Altre tecnologie si basano invece su misure inerziali, magnetiche o meccaniche. Si parla di sistema di tracciamento tramite Marker-based quando la localizzazione del dispositivo avviene tramite appositi marker posizionati nell'area di utilizzo del sistema AR. I vantaggi principali di tale tecnologia sono sicuramente la semplicità di sviluppo e i costi ridotti, a discapito del fatto che sono necessari dei marker fisici da utilizzare per questo tipo di funzionamento. Un altro tipo di tecnologia è il Simultaneous localization and mapping, detto anche SLAM, ovvero una localizzazione e mappatura simultanea. SLAM funziona grazie all'aiuto di sensori di posizione come un giroscopio o un accelerometro. Questo hardware viene utilizzato per mappare uno spazio o un oggetto. Una volta terminata la fase di mappatura

dell'immagine, il software si occuperà di eseguire una simulazione dell'immagine AR che viene proiettata nello spazio fisico con la corretta dimensione. La maggior parte degli attuali SDK AR, come quello fornito da Apple ARKit3 ed Android ARCore, sono dati della tecnologia SLAM, questo è fondamentale in quanto i vari programmatori non devono sviluppare simulazioni AR da zero.

2.1.5 Il futuro della realtà aumentata

Stando ai dati, l'AR influenzerà positivamente l'economia di diversi fattori, quali la produzione, la sanità, la vendita al dettaglio, l'istruzione, l'intrattenimento e altro ancora. Secondo uno studio di PWC l'AR aumenterà il PIL per i paesi in USA, Asia e il resto del mondo [\[9\]](#).

Ma come questa tecnologia può inserirsi nei settori sopra citati? Per quanto riguarda la sanità questa tecnologia è sicuramente una scelta eccellente in quanto potrebbe risultare un ottimo supporto ai professionisti della salute per accedere in modo rapido alle informazioni necessarie, per esempio durante un intervento, senza il rischio di dover distogliere lo sguardo da quello che si sta facendo. Un altro punto sicuramente a favore di questa tecnologia è per quanto riguarda la formazione, in quanto faciliterebbe di molto l'apprendimento. Con l'AR si può godere di un apprendimento coinvolgente. Si può provare ad interagire con modelli 3D che simulano perfettamente gli organi dei pazienti, prendere note durante la simulazione d'intervento e quindi mettere a tutti gli effetti l'utilizzatore sotto le condizioni che poi realmente dovrà affrontare, annullando così ogni tipo di rischio. Un altro ambito nella quale l'AR sarà molto di aiuto è quello dell'educazione. In passato gli psicologi hanno stabilito che gli studenti mantengono l'attenzione nell'ordine dei 10-15 minuti. Tuttavia, andando avanti questa soglia sta drasticamente scendendo. La realtà aumentata con l'esperienza immersiva che offre, può essere l'antidoto giusto per questo grosso problema in quanto, utilizzando il concetto di gamification, si può rendere il momento di apprendimento anche un momento ludico.

Per fare un esempio utilizzando dei marker all'interno dei musei sarebbe possibile visualizzare dei modelli 3d, anche animati, di ciò che si sta osservando per coinvolgere molto di più lo spettatore.

Un altro settore che sicuramente può trarre grandi vantaggi dall'utilizzo della realtà aumentata è quello del marketing. Alcuni passi sotto questo aspetto li ha per esempi fatti Coca Cola, il brand di bevande ha infatti integrato all'interno delle sue lattine un codice che, se puntato con la fotocamera, mostrerà sullo smartphone delle storie esclusive, invogliando quindi i consumatori ad acquistare. Zara invece ha creato la sua app basta su AR che permette di puntare il proprio smartphone verso una vetrina e vedere dei modelli 3d che indossano l'abbigliamento proposto. Infine, come ultimo esempio possiamo parlare di IKEA, per aiutare i propri clienti nella scelta e nell'assemblaggio dei propri mobili, Ikea ha realizzato la propria app di supporto in AR. L'app permette ai consumatori di vedere il mobile selezionato all'interno della propria casa in tempo reale ed in una scala perfetta.

Un altro settore dove l'AR è ormai diventata fondamentale è quello del turismo, sono infatti presenti una moltitudine di applicazioni che permettono dei tour autoguidati, dove un avatar o un bot ci accompagna nella visita, rendendo l'esperienza molto più piacevole ed istruttiva rispetto a farla da soli. Infine, il settore che sicuramente più ha subito un upgrade grazie all'AR è quello ludico, si prevede che i giochi in realtà aumentata portino ad un mercato superiore ai 285 miliardi di dollari entro il 2023.

2.2 Realtà Virtuale

2.2.1 Definizione

Con il termine realtà virtuale (VR) si identifica una simulazione all'elaboratore di una situazione reale con la quale il soggetto umano può interagire, a volte per mezzo di interfacce non convenzionali, estremamente sofisticate, quali occhiali e caschi su cui viene rappresentata la scena [\[10\]](#). In altre parole, l'utente è completamente immerso in un mondo virtuale. L'espressione Realtà Virtuale definisce quindi un concetto complesso ed astratto.

Diverse sono state le definizioni di questo concetto in passato.

“Virtual Reality is an alternate world filled with computer-generated images that respond to human movements. These simulated environments are usually visited with the aid of an expensive data suit which features stereophonic video goggles and fiber-optic data gloves” [11].

“It is a simulation in which computer graphics is used to create a realistic-looking world. Moreover, the synthetic world is not static, but responds to the user’s input (gesture, verbal command, etc.). This defines a key feature of virtual reality, which is real-time interactivity” [12].

La coniazione del termine Realtà Virtuale può essere attribuita a Jaron Lanier, egli precisò come il concetto nacque nel settore IT e non da istruzione accademiche, inoltre affermò come la definizione di VR focalizzata solo sulla tecnologia e non sull’esperienza utente fosse riduttiva. Sugerì invece una definizione più completa in cui la VR veniva descritta come un particolare tipo di esperienza coadiuvata da una struttura IT.

Un’altra importante definizione di Realtà Virtuale arriva da Steuer che sottolinea l’importanza dell’utente e della sua esperienza attraverso i concetti di presenza e telepresenza [13]. La differenza sostanziale è che il termine Presenza si riferisce alla naturale percezione dell’ambiente, mentre il termine Telepresenza si riferisce alla percezione mediata di questo mondo. Attraverso il secondo termine quindi Steuer diede una propria definizione di Vr: *“A “virtual reality” is defined as a real or simulated environment in which a perceiver experiences telepresence”*.

Micheal Heim pone invece l’attenzione su come le potenzialità della VR possano superare i limiti presenti nel mondo fisico nella quale viviamo, difatti nell’ambiente virtuale l’utente è in grado di sperimentare infinite opportunità d’interazione con dati ed informazioni che vanno ben oltre i limiti fisici del reale [14]. Infine, un’ultima definizione molto importante è quella data da Rob Shields, che si focalizza sulle capacità immersive della realtà virtuale. Egli riflette sulle varie tecniche di interazione e comunicazione dell’utente tramite il suo avatar all’interno dei mondi virtuali [15]. L’utente riesce a vivere pienamente l’esperienza con un senso di presenza e coinvolgimento.

Il termine immersivo può essere quindi definito come l'illusione dell'utente di trovarsi all'interno del mondo virtuale, estraniato da quello fisico.

Abbiamo capito come sia importante far chiarezza sul concetto di "presenza", che possiamo definire come la sensazione dell'utente di essere totalmente immerso in un ambiente, reale o virtuale. Questo concetto è fondamentale in quanto la maggior parte degli aspetti della VR, sia negativi che positivi, sono fortemente legati ad esso. Un sistema VR, infatti è progettato per aumentare il senso di presenza dell'utente all'interno del mondo digitale.

2.2.2 Storia

Si iniziò a parlare di Realtà Virtuale già nel 1960 quando fu realizzato da Morton Heilig, il primo sistema funzionante che utilizzava un see-through display posizionato sulla testa dell'utente [16], il movimento del display poteva essere tracciato in due differenti modi: attraverso un tracker di tipo meccanico o di tipo ultrasonico. Tale display permetteva di visualizzare solamente semplici disegni in *wireframe*. Nel 1962, lo stesso Heilig brevettava un simulatore dal nome *Sensorama* (Figura 6). Questo dispositivo meccanico permetteva all'utente di vedere, ascoltare ed addirittura odorare cinque cortometraggi. Lo spettatore veniva stimolato tramite odori, vento generato da ventole integrate sul dispositivo, suoni generati da speaker e vibrazioni generate da una sedia vibrante.

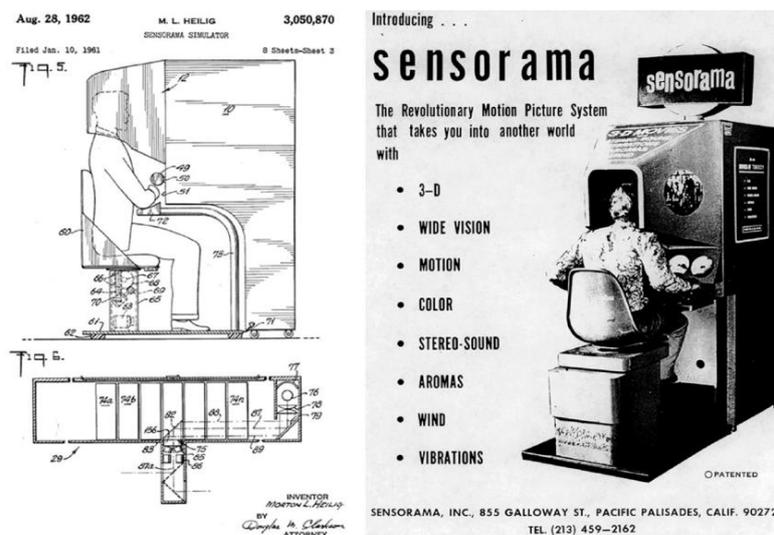


Figura 6 - Sensorama - Morton Heilig, 1962

L'idea alla base di tale sperimentazione era quella di realizzare un mondo sintetico nella quale l'utente poteva interagire in modo realistico con gli oggetti virtuali attraverso i propri sensi; per fare ciò veniva utilizzato un HMD che immergeva l'utilizzatore in questo mondo virtuale. Nel 1968, come già citato nella storia della realtà aumentata venne realizzato il primo HMD, denominato Spada di Damocle. Nel 1978 venne realizzato il primo vero simulatore per HMD, parliamo di *Visually Coupled Airbone Systems Simulator*, un avanzato simulatore di volo in cui il pilota indossava un visore nel quale venivano proiettate immagini radar e dati di volo. Nel 1984 la NASA realizza Virtual Visual Environment Display (VIVED), un HMD basato sulle tecnologie commerciali. Il primo prodotto commercializzato risale invece al 1985, quanto il produttore VPL inizia a vendere *Data Glove*, un guanto smart in grado di tracciare i movimenti della mano dell'utente. Dopo qualche anno, nel 1987, anche Nintendo rilascia il proprio guanto chiamato *Power Glove*. L'idea di Nintendo era quella di realizzare un guanto da gioco per il Nintendo Entertainment System. Il dispositivo presentava un tracker e sensori di piegatura per tracciare il movimento delle dita, oltre a dei pulsanti sul retro.

Il termine Realtà Virtuale, risale al 1989, quando Jaron Lanier, il fondatore di Visual Programming Lab (VPL), lo coniò per la prima volta. La stessa VPL nel 1988 realizza il primo HMD appositamente studiato per VR dal nome *EyePhone*. Nel 1989 fu commercializzato BOOM dalla casa di produzione Fake Space Labs, si trattava di un box realizzato con due schermi CRT, l'utente poteva muoversi all'interno dell'ambiente virtuale grazie ad un sistema meccanico che permetteva di tracciarne lo spostamento.

Nel 1992 venne rilasciato CAVE, un sistema in VR utilizzabile senza il supporto di un HMD, questo dispositivo infatti utilizzava dei proiettori per generare le immagini sulle pareti di una stanza, l'utente in ogni caso doveva utilizzare speciali occhiali LCD di tipo shutter per ottenere una vista stereoscopica e per riuscire a percepire la profondità. Negli anni successivi la tecnologia, a causa degli elevati costi di produzione e alla poca potenza di calcolo che i dispositivi presentavano, passò abbastanza in sordina.

Una svolta epocale nel mondo della VR avviene nel 2012 quando Oculus lancia la propria campagna kickstarter per la realizzazione di un nuovo visore VR, la campagna ebbe un enorme successo totalizzando 2.5M\$ e riaccese inevitabilmente l'interesse da parte di tutti verso la Realtà Virtuale. Nel 2014 Google rilascia i Cardboard, un visore low-cost realizzato in cartone utilizzabile tramite smartphone: in pratica il Cardboard funziona solo da supporto per lo smartphone, il quale aveva incorporati tutti i sensori necessari per far girare un'applicazione in VR, come il giroscopio, l'hardware performante ed il display. Nel 2016 Oculus rilascia la prima versione di visore chiamata Oculus Rift, nello stesso anno anche HTC entra nel mondo della VR rilasciano HTC Vive con la particolarità di avere i primi motion controller basati sulla tecnologia Lighthouse sviluppata in esclusiva da Valve. Negli ultimi anni la realtà Virtuale ha subito un'esplosione incredibile in quanto sono stati realizzati i primi dispositivi accessibili a tutti, come Oculus Quest 2, rilasciato nel 2020 che ha un costo di circa 400\$, inoltre l'introduzione del concetto di Metaverso e il rebranding di Facebook in Meta hanno contribuito ad aumentare esponenzialmente l'utilizzo di questa tecnologia e l'interesse delle grandi aziende e degli sviluppatori di produrre nuovi contenuti su questa.

2.2.3 La tecnologia alla base della VR

Il punto fondamentale nella realizzazione di un'applicazione in realtà virtuale è sicuramente il flusso video. Come abbiamo già spiegato alla base del funzionamento della Realtà Virtuale vi è il concetto di "presenza", per garantire un elevato grado di immersività bisogna dunque che gli strumenti utilizzati appaghino a pieno l'ambito sensoriale dell'utente. L'unione dei contributi forniti da ogni componente hardware e software determina l'entità della sensazione di presenza nell'ambiente virtuale.

Nella realtà virtuale le immagini non sono calcolate in precedenza bensì è necessario generarle in tempo reale a seconda del posizionamento dell'utente nella scena. Per ottenere una sensazione di fluidità, è necessario mantenersi su un flusso video di almeno 25 frame per secondo, altrimenti i movimenti dell'utente risulteranno imprecisi e a scatti, oltre a poter causare sensazioni di nausea all'utilizzatore.

2.2.3.1 Flusso Video

L'illusione visiva della tridimensionalità dell'ambiente virtuale nella quale l'utente viene immerso viene dato dal meccanismo di visione stereoscopica, che sfrutta i meccanismi di funzionamento della vista umana. L'occhio umano è un sistema complessissimo, composto da varie parti ed assimilabile ad una macchina fotografica. La pupilla è la parte dell'occhio che serve a mettere a fuoco l'immagine, l'immagine passando dal cristallino viene poi proiettata sulla retina. Il nervo ottico a questo punto si occupa di prendere l'immagine dalla retina trasferirla al cervello. All'interno della struttura dell'occhio sono poi presenti i coni e i bastoncelli. I coni si occupano della visione diurna e colgono i dettagli di ciò che si osserva, i bastoncelli invece veicolano le immagini meno nitide, ma sono molto più sensibili alla luce. Essendo gli occhi distanti fra di loro di circa 8 cm, le immagini prodotte dai due occhi risultano con prospettive differenti, il cervello si occupa di fondere le due immagini e dare la sensazione di profondità che percepiamo, questo processo viene definito come stereopsi. La stereopsi è quindi la capacità percettiva che consente di unire le immagini che provengono dai due occhi. I meccanismi presenti all'interno dei visori di Realtà Aumentata sfruttano questo meccanismo, ovvero le due lenti presenti negli HMD fanno visualizzare all'utente due immagini differenti che vengono elaborate dal nostro cervello che ne ritorna una, dando appunto la sensazione di essere all'interno di un ambiente tridimensionale.

2.2.3.2 Flusso Audio

Un altro concetto importante è quello dell'audio spazializzato. Questo permette di posizionare all'interno dell'ambiente i suoni in un qualsiasi punto, di direzionarli a piacere, di generare effetti di assorbimento o eco, esattamente come avviene nel mondo reale. Per rendere possibile ciò è necessario un sistema audio stereofonico che sia in grado di individuare la posizione delle sorgenti sonore all'interno della scena e calibrare l'output audio di conseguenza. Ancora una volta è utile capire come funziona il sistema sensoriale umano per comprendere questa tecnologia. Un suono generato nello spazio crea un'onda sonora che si propaga fino alle nostre orecchie.

Supponiamo di essere in presenza di un suono proveniente dalla nostra destra, le onde generate da questo suono raggiungeranno prima l'orecchio destro e poi quello sinistro, inoltre il suono percepito dall'orecchio sinistro sarà più attenuato rispetto a quello ricevuto dall'orecchio destro. Inoltre, i vari suoni che ascoltiamo sono filtrati a causa dell'interazione con il torso, la testa ed il padiglione auricolare, la struttura del padiglione difatti modifica di molto le onde sonore, rinforzandone alcune e attenuandone altre. Il nostro cervello, quindi, utilizza questo ritardo proveniente da una delle due orecchie per determinare la posizione della sorgente del suono. L'audio 3D riesce a ricostruire tale meccanismo livellando l'audio ascoltato dalle due orecchie.

2.2.3.3 Interazione con l'ambiente virtuale

Uno dei principali aspetti che garantiscono un gran senso di immersività è dato dalla possibilità di interazione con l'ambiente virtuale. I tradizionali sistemi di input che conosciamo, come la tastiera, il mouse o i joystick possono essere sostituiti da strumenti che offrono un'interfaccia più naturale, per esempio l'utilizzo direttamente delle mani attraverso delle camere che ne tracciano il movimento. Il tipo di interazione che si può avere con il mondo virtuale può essere di due tipi, bidirezionale, ovvero se l'utente muove una parte del corpo, l'input genererà un output che per esempio ritorna una forza, ovvero una sensazione tattile coerente con il movimento effettuato. Supponiamo per esempio di star afferrando un oggetto, la forza di output che il sistema deve generare deve essere coerente con la forma dell'oggetto che stiamo afferrando. Tale interfaccia è definita come aptica, e può coinvolgere qualunque parte del corpo. Esistono svariati dispositivi aptici fra i quali i più diffusi si limitano alle mani o alle braccia dell'utente, ma si stanno sviluppando sempre più dispositivi che coinvolgono l'intero corpo. Per ottenere l'effetto del ritorno della forza si possono utilizzare vari meccanismi, come per esempio dei supporti meccanici per consentire o meno alcuni movimenti o rallentarli. Oppur utilizzano piccole scariche elettriche che vanno a stimolare il corpo umano.

2.2.3.4 Sincronizzazione della posizione dell'utente

Fondamentale in un sistema di Realtà Virtuale risulta anche il giusto posizionamento dell'utente all'interno del mondo reale. Senza un corretto posizionamento sarebbe impossibile generare il corretto output visivo, sonoro e tattile, venendo meno la sensazione di presenza. Sono pertanto necessari dei dispositivi di tracciamento della posizione dell'utente molto accurati e precisi, quali accelerometri e giroscopi. Una delle principali difficoltà sta sicuramente nella sincronizzazione fra i vari dispositivi hardware che gestiscono tutto ciò sopra descritti, difatti è necessario una scheda grafica per il flusso video, una scheda audio per i suoni, i dispositivi aptici per l'interazione, oltre che un processore in grado di renderizzare tutte le immagini in real-time ed unire tutti i dispositivi sopra elencati senza cali di prestazione.

2.2.3.5 Dispositivi per il tracciamento

I dispositivi 3D possono essere classificati a seconda della tecnologia che utilizzano per la misurazione. I dispositivi meccanici utilizzano potenziometri o encoder ottici per misurare rotazione dei perni di vincolo delle aste di collegamento. Noti gli angoli di ciascun giunto e note le lunghezze delle aste presenti nella catena cinematica, è possibile con facilità calcolare la posizione dell'oggetto tracciato. I dispositivi elettromagnetici sono costituiti da un trasmettitore e da un ricevitore. Il trasmettitore genera un campo magnetico mediante delle spire che, attraverso altre spire viene, rilevato dal ricevitore. La variazione del segnale elettromagnetico permette di calcolare la variazione di posizione e di orientamento. I dispositivi ottici sono caratterizzati da una elevata accuratezza, ma anche da elevati costi ed elevata complessità di calcolo. L'oggetto da tracciare viene riempito da segnali luminosi, definiti marker e delle telecamere rilevano la posizione e l'orientamento in real-time. I dispositivi inerziali utilizzano giroscopi per misurare cambiamenti di rotazione attorno ad uno o più assi. Questa tecnologia è quella che viene più utilizzata negli ultimi anni, basti pensare agli stabilizzatori delle videocamere.

2.2.4 Il futuro della realtà virtuale

Negli ultimi anni tutte le aziende tech stanno ormai puntando sull'Extended Reality, che comprende sia la VR che l'AR. Meta ha venduto circa 10 milioni di visori nell'ultimo anno e mezzo. Microsoft sta lanciando HoloLens2 ed Apple entro il 2023 lancerà il proprio dispositivo AR e VR. Secondo le previsioni Idc, saranno spediti quest'anno circa 16 milioni di visori ed entro un decennio le vendite potrebbero raggiungere quelli degli smartphone. La qualità dei visori per la realtà virtuale è migliorata in modo esponenziale dagli anni Novanta **(figura 7)**, c'è chi dice che entro il 2040 la realtà virtuale sarà in uno stadio così avanzato che i mondi fisici saranno indistinguibili da quelli virtuali.

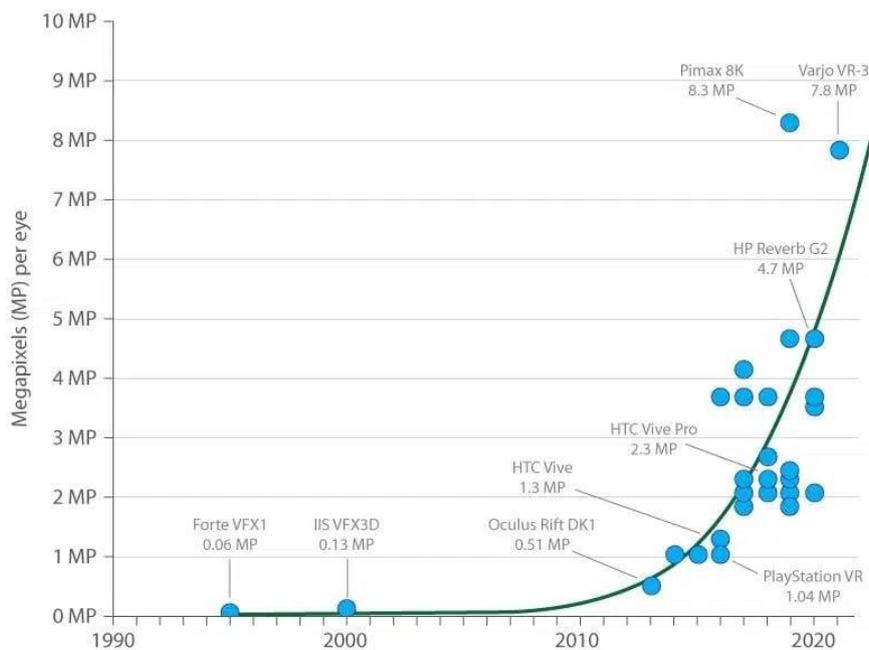


Figura 7 - Miglioramenti dei Visori negli anni

Tuttavia, oggi lo stato dell'arte è il 4k, una risoluzione in grado di fornire circa 8 megapixel per occhio, ma siamo ancora bene lontani dall'essere indistinguibile dalla realtà. I miglioramenti in ogni caso sono all'ordine del giorno, sia per quanto riguarda il campo visivo (FOV) sia per quanto riguarda le latenze e i sistemi di tracking sempre migliori.

La previsione è dunque che i visori VR combinati con sensori audio, sensori tattili e altri miglioramenti ai dispositivi associati, consentiranno di vivere esperienze VR davvero indistinguibili dal reale fra non molto tempo.

Secondo Raja Koduri, Chief Architect, Senior Vicepresidente della divisione grafica di Intel, la VR con 16K a 240Hz consentirebbe una visione indistinguibile dal mondo reale. Anche Michael Abrash, Chief Scientist di Oculus, ha affermato che la risoluzione del display a 16K è il punto in cui l'occhio umano non sarebbe in grado di distinguere tra realtà virtuale e realtà fisica, inoltre si sta cercando di migliorare anche il campo visivo che potrebbe in brevissimo tempo raggiungere gli stessi 220 gradi che costituiscono il normale FOV della visione umana.

Insomma, nuove applicazioni verranno costantemente sviluppate con dettagli e realismo sempre migliore, specialmente se combinate con l'AI. In campo didattico, ad esempio, le rievocazioni storiche potrebbero consentire a un utente di interagire con personaggi famosi e osservare eventi chiave del passato. In una sorta di "viaggio virtuale" sarebbe possibile "passeggiare" nelle città del mondo con estremo realismo, con una sorta di Google Street View in realtà virtuale. Questo sicuramente favorito dal sempre più grande interesse da parte sia degli utenti che delle aziende nel mondo del Metaverso, che spinge i programmatori a creare sempre nuovi software per ogni genere di utenza. Si prospetta quindi un futuro roseo per questa tecnologia.

3 Il cuore del Metaverso: la Blockchain

Dopo aver parlato di Realtà virtuale e Realtà aumentata, sicuramente fondamentali per l'utilizzo in modo rivoluzionario e produttivo del Metaverso, andiamo a capire come il Metaverso garantisce ai propri utenti affidabilità e trasparenza. Come ampiamente descritto nel primo capitolo, possiamo immaginare un Metaverso come un'applicazione in real-time dove più utenti possono collegarsi, anche in parallelo, ad altri utenti e svolgere diverse attività come fare shopping, partecipare ad eventi o concerti, svagarsi utilizzando dei giochi e molto altro. Lo schema classico di un Metaverso online può essere dunque immaginato come un'applicazione locale che si basa su un unico database ed un unico server centralizzato. **(figura 8)**



Figura 8 - Sistema Centralizzato

Questa schematica però presenta una grossa problematica, ovvero, chiunque è in grado di accedere al database potrebbe alterare le caratteristiche dell'applicazione, quindi per esempio eliminare utenti, cambiare permessi o proprietà o inserire nuove monete all'interno dell'economia del gioco. Per questo motivo i vari Metaversi vengono realizzati tramite l'utilizzo di una Blockchain. **(figura 9)**

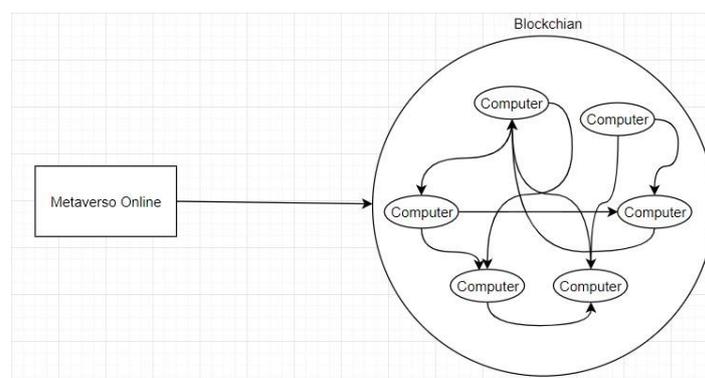


Figura 9 - Sistema Decentralizzato

Attraverso questa tecnologia, non esiste più un database unico nella quale sono contenute tutte le informazioni degli utenti e dell'applicazione, bensì le varie informazioni sono appunto decentralizzate e contenute all'interno di tutti i computer appartenenti alla rete blockchain. Ogni qualvolta bisogna validare un'operazione, tutti i computer comunicano fra di loro per concordare lo stato della blockchain. Questo è sicuramente il principale motivo per la quale la blockchain è diventata la spina dorsale dei Metaversi. Ma analizziamo più nello specifico che cos'è una blockchain e come funziona.

3.1 Storia

"A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution." [\[18\]](#)

La vera introduzione del concetto di blockchain può essere attribuita a Satoshi Nakamoto⁴ che nell'ottobre del 2008 all'interno del suo white-paper intitolato "Bitcoin: A Peer-to-Peer Electronic Cash System" propose un meccanismo di consenso decentralizzato, ovvero con la possibilità di validare delle transazioni senza la necessità di avere un intermediario⁵. L'idea era quella di avere una moneta virtuale a chiave crittografica peer-to-peer senza un'entità centralizzata che la gestisse, dove l'unico validatore era l'algoritmo, stiamo parlando del Bitcoin. Nel 2009 la tecnologia di Satoshi inizia a funzionare, viene utilizzato il primo Bitcoin per acquistare una pizza. Nel 2012 Bitcoin raggiunge la capitalizzazione di 1 miliardo di dollari.

Specialmente nei primi anni dalla sua idealizzazione l'idea di un sistema decentralizzato ha suscitato molta curiosità, specialmente nel mondo della finanza, successivamente, negli anni più recenti, è aumentata la conoscenza e la consapevolezza di questa tecnologia e la blockchain si è trasformata in una base tecnologica in grado di soddisfare diverse esigenze e risolvere diversi problemi. In parole povere con la blockchain l'utente che la utilizza si connette alla rete e completa la propria transazione esclusivamente

⁴ Pseudonimo dell'autore del progetto, la cui identità rimane ancora un mistero

⁵ un'entità centralizzata

affidandosi al programma ed al suo protocollo, non esistono più quindi mediatori che validano la transazione. Per questo motivo, nel 2010, il concetto di bitcoin viene associato al mercato illegale a causa dell'assenza di un'autorità di controllo, tale tecnologia passò così molto in sordina in quegli anni e creò disinteresse nel mondo del business.

Nel 2014 però l'interesse inizia a traslare dal Bitcoin in sé alla tecnologia che vi è dietro, la blockchain. Iniziano quindi a nascere nuove piattaforme che sfruttano tale tecnologia come Ethereum⁶, Ripple ed altri. Nel 2015 nasce anche R3, consorzio composto dalle più importanti banche mondiali, che sviluppa Corda⁷. Anche la fondazione Linux si dà da fare ed inizia a lavorare al progetto Hyperledger, una piattaforma collaborativa utilizzabile dalle aziende basta su blockchain. Sono questi gli anni nella quale vi è molto scetticismo, se da una parte c'è chi crede al progetto ed alla tecnologia, dall'altra c'è chi credere che il boom mediatico introno all'argomento abbia creato solo una bolla, che è diventata pura speculazione a vantaggio delle criptovalute e delle azioni illegali ad esse associate. Gli anni successivi furono un consolidamento delle tecnologie ed un interesse sempre maggiore da parte sia degli utenti che delle aziende di comprenderne le potenzialità. Nel 2017 si entra però nella cosiddetta fase di disillusione dell'hype cycle di Gartner, i concetti che stanno alla base della tecnologia non riescono a concretizzare quanto promesso negli anni precedenti, le criptovalute continuano ad oscillare di valore e soprattutto il processo di validazione della rete di Bitcoin⁸ presentava dei limiti nell'effettuare le transazioni, consumava elevata energia e vi era comunque il rischio di centralizzare la tecnologia, a causa dei pochi computer connessi alla rete. [\[19\]](#) Negli ultimi anni sta nuovamente prendendo piede l'utilizzo e lo sviluppo di tale tecnologia. Molte più aziende si stanno interessando a questo nuovo paradigma. Attualmente questa tecnologia viene infatti utilizzata in vari settori quali la pubblica amministrazione, il settore alimentare, la logistica, le compagnie assicurative,

⁶ Piattaforma orientata alla creazione di smart contract

⁷ Piattaforma blockchain, che consente di effettuare transazioni utilizzando smart contract, garantendo privacy e sicurezza.

⁸ Mining

la sanità, le applicazioni come il Metaverso e molto altro. Per fare un esempio pratico nel 2019 durante la Mostra internazionale dell'artigianato è stato presentato il primo manufatto artigianale Made in Italy realizzato dalla Savio Firminio che ha tracciato in tutte le sue fasi di realizzazione il vassoio in questione tramite blockchain.

3.2 Bitcoin

Come abbiamo già visto l'applicazione più famosa e diffusa che si basa su tecnologia blockchain è il Bitcoin. Sviluppato da Satoshi Nakamoto nel 2008, questa tecnologia proponeva di fare una vera e propria guerra contro le banche ed il sistema centralizzato che queste proponevano. Nel suo white-paper [18] fu lo stesso ideatore ad introdurre il funzionamento. Il codice utilizzato dalla valuta digitale è XBT secondo lo standard ISO-4217. Bitcoin non avendo un'amministrazione centralizzata, viene classificato come valuta virtuale decentralizzata dal Tesoro degli Stati Uniti. Inoltre, viene anche definita come criptovaluta, ovvero che si basa su protocolli software crittografici per generare la valuta e convalidare le transazioni [20]. I punti di forza del protocollo servono a garantire in primis integrità e sicurezza dei dati durante una transazione. La rinuncia ad un sistema centralizzato permette una più semplice ed efficace gestione delle transazioni finanziarie, oltre a permettere a chiunque di poter accedere a questi ultimi, limitando le disparità sociali. Nel white paper di Bitcoin viene spiegato come un sistema del genere risolve il problema del double-spending⁹.

Si può leggere nel documento "*The problem of course is the payee can't verify that one of the owners did not double-spend the coin.*", il problema è quindi che in assenza di un intermediario non si può avere la certezza nelle transazioni. Supponiamo che Alice voglia inviare cento monete a Bob, come può Bob essere sicuro che Alice non abbia inviato quelle monete prima ad un altro utente senza la presenza di un garante? L'adozione di un database distribuito consente di evitare tale problematica in quando la transazione viene validata dall'algoritmo stesso e non esiste più un single point of failure.

⁹ Doppia spesa

Nel tecnico la soluzione proposta nel *libro mastro*¹⁰ è quella di utilizzare un server timestamp. Un server timestamp funziona prendendo un hash da un blocco di elementi, questo viene contrassegnato con il timestamp e l'hash risultante. La forza del timestamp in questo caso è quella di dimostrare che i dati devono essere esistiti al tempo. Ogni timestamp include inoltre il timestamp precedente nel suo hash, questo crea una catena che rafforza i dati successivi.

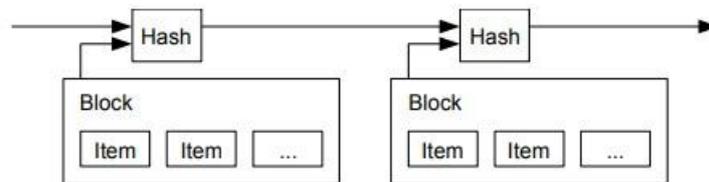


Figura 10 - Timestamp Server

3.3 Blockchain: funzionamento e caratteristiche

Una volta capito di cosa stiamo parlando, dobbiamo capire su quali caratteristiche si basa tale tecnologia e come effettivamente funziona. Possiamo individuare alcune caratteristiche fondamentali che sono: decentralizzazione, trasparenza, sicurezza, consenso, immutabilità, responsabilità e programmabilità. Per parlare della tecnologia che sta alla base della blockchain possiamo riferirci a quella presentata da Satoshi Nakamoto in bitcoin, in quanto esistono anche altre varianti ma abbastanza simili fra di loro se non per alcuni aspetti. Possiamo definire una blockchain come un database pubblico decentralizzato che tiene traccia di ogni transazione e dei beni posseduti dagli utenti appartenenti alla rete in modo trasparente e pubblico attraverso una rete peer-to-peer. I dati presenti all'interno della blockchain sono protetti tramite crittografia e raccolti cronologicamente tramite dei blocchi di dati. In linea di principio quindi possiamo considerare la blockchain come una struttura dati con timestamp di sola aggiunta, non è infatti possibile rimuovere o mutare degli elementi una volta inseriti nella rete. Per raggiungere questo obiettivo si può

¹⁰ Così venne definito il White Paper di Bitcoin

considerare la blockchain come un insieme di meccanismi interconnessi che forniscono caratteristiche specifiche all'infrastruttura [\[22\]](#) (**figura 11**).

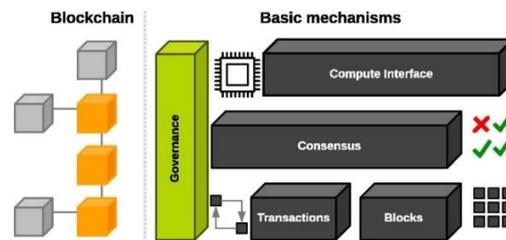


Figura 11 - Struttura di una Blockchain

3.3.1 Distributed ledger technology (DLT)

La tecnologia di un registro distribuito (DLT) è un sistema digitale per la registrazione delle transazioni, in cui le transazioni e i loro dettagli sono registrati in più luoghi contemporaneamente. A differenza dei database tradizionali, i registri distribuiti non dispongono di un archivio dati centrale o di funzionalità di amministrazione [\[21\]](#). In un registro distribuito, ogni nodo della rete elabora e verifica ogni elemento, generando così un record e creando un consenso sulla sua veridicità. La tecnologia DLT si riferisce specificamente all'infrastruttura tecnologica e ai protocolli che consentono l'accesso simultaneo, la convalida e l'aggiornamento dei record presenti nei registri distribuiti. Viene utilizzata la crittografia per archiviare i dati in modo sicuro e vengono usate chiavi crittografiche per consentire l'accesso solo agli utenti autorizzati. Oltre a ciò, si crea un database che risulta essere immutabile, il che significa che le informazioni, una volta memorizzate, non possono essere eliminate e gli eventuali aggiornamenti vengono salvati in modo permanente, dunque per sempre visibile; è quindi possibile tracciare tutte le transazioni fatte per esempio da un portfolio connesso alla blockchain, senza in alcun modo poter compromettere i dati. La trasparenza ottenuta

dalla tecnologia DLT fornisce un alto livello di fiducia tra i partecipanti alla rete ed elimina la possibilità che si verifichino attività fraudolente.

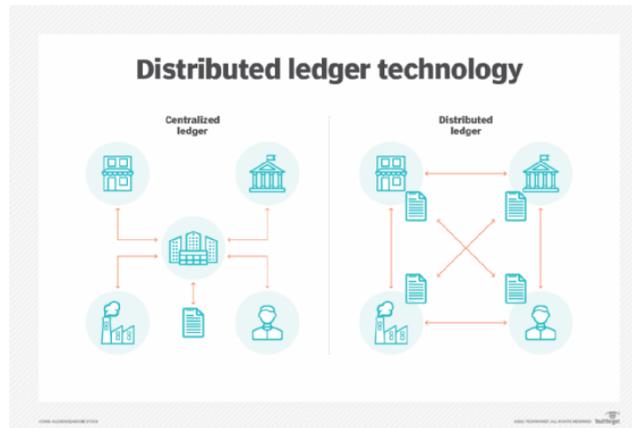


Figura 12 - Centralized ledger VS Distributed ledger

3.3.2 Livello di Consenso

Nel più basso livello dell'infrastruttura blockchain abbiamo le transazioni firmate tra peer. Tali transazioni denotano un accordo fra due partecipanti, che può comportare per esempio il trasferimento fisico o digitale di un bene. Almeno uno dei partecipanti firma la transazione, questa successivamente viene inoltrata ai nodi vicini per essere validata. In generale definiamo con il termine nodo ogni entità che si connette alla rete blockchain. I nodi completi sono invece quelli che verificano tutte le regole blockchain, questi si occupano di raggruppare le transazioni in blocchi e sono responsabili di determinare se tali transazioni sono valide o meno. Per fare un esempio di transazione valida possiamo supporre che Bob debba mandare un bitcoin ad Alice. Tuttavia, Alice potrebbe aver tentato in precedenza di inviare lo stesso bitcoin a Carol, pertanto i vari nodi della rete devono raggiungere un accordo sulle transazioni che devono essere mantenute sulla blockchain. Questo è l'obiettivo del secondo livello di consenso. È possibile registrare nuove transazioni soltanto quando la maggioranza dei partecipanti alla rete dà il proprio consenso. A seconda del tipo di blockchain, esistono diversi meccanismi di consenso. I più noti sono il Proof-Of-Work (PoW) ed il Proof-Of-Stake (PoS), ne esistono poi altri come Byzantine Fault Tolerance (BFT) e le sue varianti. [\[22\]](#)

Il Proof-of-Work è un algoritmo di consenso che richiede elevate computazione, i miners¹¹ devono risolvere dei problemi matematici complessi per poter aggiungere dei blocchi alla blockchain, questo comporta un elevato dispendio elettrico. Il vantaggio di utilizzare tale tecnologia è che, seppur il calcolo risulta complesso da risolvere, è invece molto semplice da verificare. I miners competono fra di loro per trovare la soluzione al problema e validare il blocco. Il miner vincitore, ovvero che risolverà prima il problema, sarà ricompensato tramite una reward. Una volta trovata la soluzione, questa viene passata agli altri nodi che prima di aggiungere il blocco ne controllano la correttezza, processo molto facile come già detto. Un altro vantaggio di questo sistema è che dato che la complessità del calcolo è molto elevata eventuali attaccanti sono scoraggiati. Se la soluzione proposta non è corretta il blocco non sarà inserito sulla rete e si è solo sprecato risorse. Il processo di creazione di una transazione si articola in questo modo [\[23\]](#):

1. Tutte le transazioni sono memorizzate in un blocco.
2. I minatori convalidano le transazioni in ogni blocco.
3. I minatori risolvono un problema matematico chiamato proof-of-work.
4. Il minatore che risolve la transazione viene premiato.
5. Infine, viene creata la blockchain pubblica con transazioni validate.

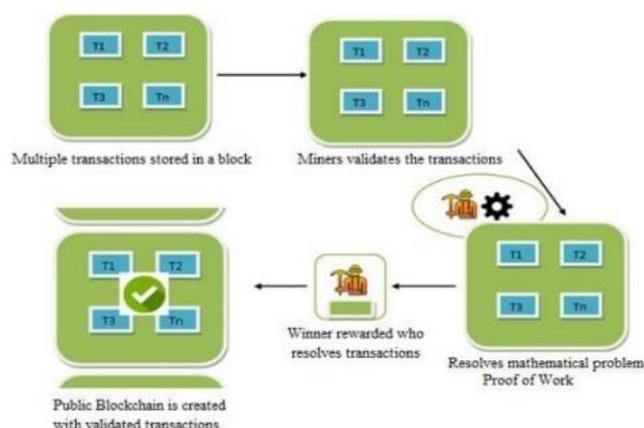


Figura 13 - Processo di validazione di una transazione tramite PoW

¹¹ Persone che registrano e verificano transazioni su una blockchain

Il PoW viene utilizzato da diverse criptovalute in primis Bitcoin. Bitcoin utilizza Hashcash (SHA-256) come PoW. Hashcash è un sistema di Proof-Of-Work creato nel 1997 da Adam Back, anche se il White Paper fu pubblicato nel 2002. L'idea è nata con lo scopo di limitare l'abuso sistematico di risorse Internet come le e-mail. [\[24\]](#)

Nel White Paper di Bitcoin possiamo leggere come Hashcash prevede la scansione di un valore che, se sottoposto ad hash, come con SHA-256, inizia con un numero di zero bit. Il lavoro medio richiesto è esponenziale nel numero di zeri bits richiesti e può essere verificato usando un singolo hash [\[18\]](#). In altre parole, venivano usate delle cifre esadecimali: 8 per il nonce, 64 per i valori di hash e 4 zeri iniziali per confermare il lavoro. Il nonce, generalmente di 32 bit, è un contatore aggiuntivo che serve da input della funzione di hash per aumentarne la complessità. Per provare il lavoro di hashing, il nonce viene incrementato di 1 bit ad ogni calcolo dell'hash finché il digest da 256 bit, SHA-256 come detto, non sarà costituito da 16 bit zeri iniziali. Il primo nodo che riuscirà a trovare la soluzione potrà creare un nuovo blocco sulla rete. Questo processo sopra descritto viene definito come mining ed i nodi coinvolti come miner. La complessità del problema da svolgere è proporzionale al numero di miner ed alla potenza di calcolo, possono essere richiesti problemi come l'individuare un input partendo dal digest della funzione hash o la scomposizione di numeri primi. Fondamentale risulta essere proporre un problema bilanciato perché se un problema risultasse troppo complicato si rallenterebbe troppo la creazione di blocchi, mentre se il problema risultasse troppo semplice, la rete sarebbe facilmente attaccabile.

La Proof-of-Stake è indirettamente proporzionale alla dimensione della rete e al numero di persone in staking sulla valuta digitale. Se ci sono molti utenti in staking, ovvero che stanno scommettendo sulla moneta in questione, ci saranno meno ricompense. Inoltre, se gli utenti hanno il possesso di più criptovalute per un periodo più lungo, otterranno più ricompense, tuttavia, il processo dovrebbe essere condiviso sulla rete in modo che il controllo della moneta non sia limitato ad una sola persona. Questo concetto funziona in modo simile al deposito fisso delle banche in cui il cliente guadagna più

interessi se mantiene una quantità di denaro significativa o per tanto tempo. In questo sistema l'utente che convalida la transazione mette in gioco le sue monete per creare un nuovo blocco. I nodi possono anche perdere la loro partecipazione per le transazioni future se il validatore si accorge di un comportamento fraudolento. Per verificare il validatore del blocco successivo possono essere utilizzati diversi modi [\[23\]](#):

1. **Selezione Randomica:** l'utente con il valore hash più basso e con la dimensione di puntata più bassa avrà l'opportunità di selezionare il blocco successivo.
2. **Selezione basata sull'età della moneta:** l'età della moneta seleziona il validatore.

Sia il PoW che il PoS possiedono dei punti a favore e contro. Sicuramente il punto principale che verte verso il Proof-Of-Stake è il rispetto dell'ambiente, difatti l'idea di base risiede nella necessità di evitare sprechi di energia dovuti all'elevato costo computazionale necessario nel PoW. Il PoW di contro, oltre all'ingente consumo energetico, presenta il rischio di andare verso un sistema sempre più centralizzato, questo è dovuto al fatto che se prima era possibile entrare a far parte della rete con un normale computer di casa, adesso la rete è costituita da super computer organizzati in sistemi aggregati ed appartenenti a pochi nodi. Anche sotto questo aspetto il PoS tenta di risolvere il problema utilizzando un sistema randomizzato che non aiuta semplicemente il validatore con più potenza di calcolo. Sono anche questi i motivi per la quale il PoS sta prendendo sempre più piede, anche Ethereum da poco tempo ha deciso di cambiare il suo paradigma passando ad un Proof-Of-Stake.

3.3.3 Interfaccia di Calcolo

Un livello aggiuntivo è definito come Compute Interface, questo layer consente alle blockchain di offrire più funzionalità. Una blockchain memorizza uno stato che consente agli utenti di visualizzare tutte le transazioni che sono state effettuate sulla rete, questo permette il calcolo del saldo di ciascun utente, abbiamo quindi trasparenza. Tuttavia, nel caso di applicazioni più avanzate, abbiamo bisogno di memorizzare stati complessi che vengono

aggiornati dinamicamente, per esempio degli stati che passano da un utente ad un altro solo quando alcuni criteri sono soddisfatti. Questa necessità ha dato origine agli Smart Contracts, che utilizzano i nodi della blockchain per eseguire i termini di un contratto [22]. Analizzeremo gli Smart Contracts successivamente.

3.3.4 Governance

L'ultimo blocco nella struttura è quello di Governance. Il livello di Governance estende la tecnologia Blockchain per coprire le interazioni umane che avvengono nel mondo fisico. Difatti, seppure i protocolli siano bene definiti, questi sono influenzati dagli input degli umani che integrano nuovi metodi, creano degli upgrade e risolvono i problemi esistenti. Sebbene questi step siano fondamentali per ottimizzare il funzionamento della blockchain, costituiscono processi sociali che vanno al di fuori della catena. Dunque, la governance blockchain si occupa di gestire ed accertare come questi diversi attori si uniscono per produrre, mantenere o modificare gli input che compongono la rete.

3.4 Classificazione delle blockchain

Sebbene l'idea iniziale di Satoshi Nakamoto fosse quella di una rete che fosse totalmente pubblica, esistono vari tipi di Blockchain. Quando parliamo della blockchain Bitcoin stiamo parlando in un tipo di rete pubblica.

Nelle blockchain pubbliche ogni utente può partecipare alla rete come nodo e svolgere operazioni di mining, mettendo a disposizione la propria quota di partecipazione, che può essere computazione o staking. In questo caso tutti possono effettuare, validare transazioni ed accedere ai contenuti del database. Stiamo parlando di una rete totalmente decentralizzata che garantisce la sicurezza dei dati da potenziali attacchi esterni.

Nella blockchain privata solamente ad alcuni utenti, con particolari permessi, è consentito validare o modificare lo stato della catena, questo ovviamente non fa rinunciare alle caratteristiche di autenticità e decentralizzazione, fondamentali quando parliamo di blockchain. In questo caso non tutti possono scrivere sulla rete ma tutti possono leggerne il contenuto.

Esiste un'altra tipologia di blockchain, la permissioned detta anche "decentrata" o Consorzio, questa risulta essere un misto fra le due precedentemente descritte; il meccanismo di consenso è gestito da nodi predefiniti che sono noti apriori, questo garantisce fiducia pregressa ed una maggiore efficienza di validazione. Le autorizzazioni di lettura invece possono essere rilasciate a tutti gli utenti della rete oppure solamente ad alcuni utenti, per garantire privacy.

3.5 Ethereum e gli smart contract

Gli smart contract¹² sono stati ideati da Nick Szabo nella sua opera Smart Contract Glossary [25] già nel 1995. Nell'anno successivo Szabo approfondì l'argomento nell'opera Smart Contracts: Building Blocks for Digital Markets. [26] Il lavoro svolto ha gettato le basi per il funzionamento attuale degli smart contract che tutt'ora vengono utilizzati all'interno delle piattaforme digitali. Nella sua prima opera Nick parlava dei concetti fondamentali per avere uno Smart Contract. Gli attori principali sono:

1. **Agenti**: una persona, un'organizzazione o un programma controllato da un computer e che agisce per conto di un agente.
2. **Contratto**: una serie di accordi o promesse tra agenti.
3. **Parti**: Agenti che hanno accettato il contratto in questione.
4. **Terze Parti**: agenti che non hanno accettato il contratto.
5. **Sicurezza Contrattuale**: paradigma per prendere accordi di sicurezza tra le organizzazioni.
6. **Distribuzione delle chiavi del contratto**: un paradigma per la distribuzione delle chiavi tra individui ed organizzatori.
7. **Protocollo**: sequenza di messaggi fra più agenti
8. **Contratto intelligente**: una serie di promesse, inclusi i protocolli, all'interno dei quali le parti mantengono le altre promesse.

Più che di veri contratti stiamo parlando di codici if-this-then-that che in autonomia leggono le clausole e le condizioni contrattuali poste quando vengono stipulati e sono quindi in grado di autogestirsi nel tempo. Questo è

¹² In italiano contratti intelligenti

appunto fondamentale in una rete decentralizzata in quanto non esiste più la figura di intermediario che fa da garante per la transizione, bensì la responsabilità decisionale ricade solo sul software, garantendo oggettività di giudizio. Il rischio alla quale si va in contro è quello di non prevedere qualche scenario possibili durante la stipula, creando delle potenziali falle. Possiamo quindi definire gli Smart Contracts autonomi e distribuiti.

La vera rivoluzione in quest'ambito avvenne con l'avvento di Ethereum. Possiamo dire che uno Smart Contract è semplicemente un programma eseguito sulla blockchain di Ethereum. È una raccolta di codice (le funzioni) e dati (lo stato) che risiede a un indirizzo specifico sulla blockchain di Ethereum. Gli Smart Contracts sono dunque un tipo di account Ethereum. Questo vuol dire che hanno un proprio saldo e possono essere oggetti di transizioni. Questi non sono controllati da un utente, ma sono distribuiti in rete ed eseguiti come pre-programmato. Gli utenti possono quindi interagire con gli Smart Contract inviando transazioni che vanno ad eseguire una funzione scritta all'interno del contratto. Gli Smart Contract possono definire regole, come un normale contratto, e utilizzarle in modo automatico tramite il codice. Gli SC non sono eliminabili, una volta inseriti all'interno della rete blockchain rimarranno lì per sempre e non sono nemmeno reversibili, una volta che si interagisce con essi, l'interazione rimarrà sulla rete per sempre visibile [\[27\]](#).

Ethereum è una blockchain sviluppata nel 2013 da Vitalik Buterin e nasce dalla necessità di un linguaggio di scripting per la creazione di applicazioni decentralizzate. Questa tecnologia segna il passaggio al Distributed Computing¹³, possiamo quindi considerarla come un grosso computer distribuito, sempre accessibile da ovunque e con una potenza di calcolo elevatissima. La rete Ethereum è definita anche programmabile, in quanto dà, agli utenti che la utilizzano, la possibilità di creare delle applicazioni decentralizzate. Ethereum è la tecnologia gestita dalla community che alimenta la criptovaluta ether (ETH) e migliaia di applicazioni decentralizzate [\[28\]](#).

¹³ Più componenti software che si trovano su più computer, ma eseguiti come un unico sistema

Ethereum è considerabile come una piattaforma decentralizzata che si basa sul Web 3.0 per la creazione e pubblicazione di Smart Contracts. La criptovaluta collegata alla rete è l'Ether ed è attualmente la seconda in capitalizzazione, preceduta solamente da Bitcoin. Per poter girare sulla rete, i contratti di Ethereum pagano l'utilizzo della sua potenza computazionale tramite Ether, che svolge quindi un duplice compito, criptovaluta e carburante¹⁴.

A differenza di Bitcoin, Ethereum opera seguendo le cosiddette transizioni di stato, che si basano sui saldi correnti di tutti i conti. L'informazione relativa allo stato non è memorizzata sulla rete blockchain, bensì viene archiviata in un albero di Merkle¹⁵. Il sistema di scrittura anche in questo caso si basa su chiave asimmetrica, attraverso la chiave privata sarà possibile effettuare una transazione in Ether. Per indirizzare degli Ether ad un conto si deve essere in possesso dell'hash calcolato dalla chiave pubblica del conto di destinazione, calcolato tramite l'algoritmo Keccak-256. Importante notare come i conti Ether non sono personali ma identificano uno o più specifici indirizzi.

I contratti intelligenti adottati in Ethereum vengono generati separatamente uno dall'altro all'interno della Ethereum Virtual Machine (EVM). L'istanza fisica dell'EVM non può essere descritta con una nuvola, ma esiste come un'unica entità gestita da migliaia di computer collegati fra di loro che eseguono un client Ethereum. Lo stesso protocollo Ethereum esiste esclusivamente allo scopo di mantenere il funzionamento continuo, ininterrotto ed immutabile di questa speciale macchina a stati [\[29\]](#).

3.5.1 Standard ERC

Gli smart contract sulla rete Ethereum, o su reti compatibili alla EVM, possono seguire delle linee guida durante la loro scrittura. Queste linee guida sono definite come standard Ethereum Request for Comments (ERC). Se le caratteristiche specificate nello Standard vengono rispettate sullo smart

¹⁴ Definito anche come GAS

¹⁵ Albero binario nel quale ogni nodo è padre di due figli e il suo hash è dato ricorsivamente dalla loro concatenazione

contract, allora quest'ultimo può interoperare con altri smart contract della blockchain. Gli SC creano dunque una contabilità secondaria che non va a mischiarsi con quella di Ether degli indirizzi, viene in ogni caso garantita tracciabilità, trasparenza e sicurezza. Sebbene nascano sempre nuovi standard i principali sul mercato sono sicuramente due ERC20 ed ERC721.

ERC20 è il più noto ed utilizzato, questo rappresenta lo standard usato per gli Smart Contract nella blockchain Ethereum per lo sviluppo di token. Un contratto ERC20 tiene traccia dei cosiddetti token fungibili. Quindi un token è identico a qualsiasi altro della stessa tipologia ed a nessun token possono essere associati attributi o diritti specifici e differenti dagli altri. Queste caratteristiche rendono questo standard utile per lo scambio di monete e le convalide delle transizioni.

ERC721 è invece la soluzione per i cosiddetti NFT. Un NFT, acronimo di token non fungibile, rappresenta la proprietà di un asset virtuale o fisico. Questi asset possono per esempio includere elementi collezionabili, come oggetti d'arte, asset fisici come immobili o asset di valore negativo come per esempio un prestito. Ogni token in questo caso è univoco ed ha le proprie informazioni su proprietà e stato di cui deve tenere traccia. Questo standard, dunque, presenta un ID Token per ogni token, che viene associato allo smart contract in modo univoco. Un'alternativa allo standard ERC721 è ERC1155 che risulta essere più flessibile e permette il trasferimento e il minting¹⁶ di token in gruppi.

3.6 NFT

Un non-fungible token¹⁷ (NFT) è uno particolare tipo di token, che rappresenta l'atto di proprietà ed il certificato di autenticità di un bene fisico o digitale. Gli NFT a differenza delle criptovalute per definizione non sono intercambiabili; mentre le criptovalute per definizione sono fungibili, ovvero possono essere duplicati infinite volte in copie esatte ed interscambiabili, gli NFT non possono essere scambiati fra di loro perché presentato delle

¹⁶ Processo di generazione di nuove monete, autenticando i dati

¹⁷ In italiano token non fungibile

caratteristiche uniche. La storia degli NFT risale a quando CryptoKitties [\[30\]](#) divenne virale. Da lì diverse campagne riuscirono ad ottenere diversi finanziamenti, basti pensare a RareBits che raccolse 6 milioni di dollari e Decentraland, un mondo virtuale basato su catene di blocchi, che ha raccolto ben 26 milioni di dollari solo inizialmente, per poi crescere in modo esponenziale nel tempo.

Sicuramente il campo con più applicabilità della tecnologia NFT è quello artistico, in quanto si potrebbe facilmente garantire l'autenticità di un'opera, eliminando le riproduzioni.

Gli NFT, tuttavia, vengono utilizzati in diverse applicazioni specifiche che hanno la necessità di avere degli asset digitali univoci. Questi quindi all'intento di un gioco possono essere controllati direttamente dall'utente che li possiede e non dal programmatore del gioco stesso, consentendo anche lo scambio di questi asset su mercati terziari senza l'autorizzazione dello sviluppatore del gioco; è per questo che ricoprono un ruolo fondamentale all'interno del Metaverso.

4 I principali esponenti del Metaverso

Una volta capito cos'è un Metaverso, quali sono le sue caratteristiche chiave e quali sono le tecnologie necessarie che ne permettono il suo utilizzo, in questo capitolo analizzeremo i principali Metaversi presenti sul mercato e cosa questi offrono ai loro utenti.

4.1 The Sandbox

"The Sandbox team is building a unique virtual world where players can build, own, and monetize their gaming experiences using SAND, the main utility token of the platform." [\[31\]](#)

4.1.1 Overview

Il progetto Sandbox nasce da due titoli differenti The Sandbox, pubblicato nel 2011 per piattaforma mobile, che ha ricevuto oltre 10 milioni di download sui dispositivi Android, e The Sandbox Evolution, prodotto nel 2019 anche esso per smartphone e anch'esso con download che superano i 10 milioni. L'obiettivo dei due videogiochi è quello di permettere all'utente di sbizzarrirsi con la creazione di un mondo 2D in pixel-art, potendo esplorare liberamente il mondo ma anche completando specifici obiettivi. Questa esperienza ha portato i creatori a ridisegnare i videogiochi realizzati dando la possibilità ad ogni utente di creare il proprio mondo e condividerlo con la community, garantendo ai creatori la proprietà delle opere realizzate tramite NFT e premiando la loro partecipazione all'ecosistema the Sandbox.

Nasce così il nuovo mondo The Sandbox con una nuova grafica basata su un mondo 3D in voxel¹⁸. Come precisato nel White Paper i possessori di Sand, la valuta del Metaverso in questione, potranno partecipare alla governance della piattaforma tramite una *Decentralized Autonomus Organization* (DAO), dove possono essere esercitati i diritti di voto sulle decisioni chiave dell'ecosistema The Sandbox. Ogni giocatore potrà creare nuove risorse digitali (NFT), caricarle sul mercato e costruire dei giochi grazie al tool The Sandbox Game

¹⁸ Controparte tridimensionale del pixel bidimensionale

Maker. The Sandbox mira a portare il concetto di blockchain all'interno dei giochi tradizionali, offrendo tutti i vantaggi che ne conseguono.



Figura 14 - The Sandbox

In the Sandbox è quindi possibile costruire, possedere e monetizzare le proprie esperienze di gioco sulla blockchain di Ethereum utilizzando SAND, l'utility token della piattaforma. L'obiettivo principale è quello di rivoluzionare il sistema centralizzato dei videogiochi come Minecraft e Roblox, fornendo ai creatori la vera proprietà di quanto realizzato. Questo sarà possibile grazie all'utilizzo degli NFT, ogni oggetto in game avrà quindi un unico immutabile identificatore blockchain. I vantaggi che ne conseguono sono identificabili in quattro punti principali:

1. **Reale possesso digitale:** grazie all'utilizzo della blockchain ogni oggetto può essere tokenizzato, facendo decidere al giocatore che ne possiede la proprietà come scambiare, vendere o regalare i propri oggetti.
2. **Sicurezza ed Immutabilità:** gli oggetti possono essere facilmente scambiabili grazie alla tecnologia blockchain, anche attraverso marketplace secondari, il possesso dell'asset risulterà pubblico sul ledger distribuito, eliminando le possibilità di falsificazioni.
3. **Trading:** la possibilità di acquistare e vendere Asset permette ai creatori di poter generare profitto dalle loro opere; oltre a ciò, anche se una piattaforma dovesse chiudere o fallire il valore degli Asset sarà mantenuto.
4. **Interoperabilità cross-platform:** gli asset realizzati potranno essere utilizzati anche al di fuori del mondo The Sandbox.



Figura 15 - Benefit dell'utilizzo della blockchain

Anche gli utenti che non creano Asset o giochi potranno partecipare all'economia di the Sandbox, questo è favorito dal fatto che, partecipando alle attività all'interno delle terre virtuali, si potranno ottenere delle ricompense. Inoltre, i giocatori possono utilizzare il sistema play-to-earn implementato raccogliendo ricompense durante il gioco.

4.1.2 Ecosistema

L'ecosistema di The Sandbox è costituito da tre sezioni differenti, integrate fra di loro, che forniscono un'esperienza completa per l'utente. È possibile realizzare nuovi oggetti, nuovi mondi o nuovi giochi, oltre che partecipare alle esperienze pubbliche presenti sulla piattaforma che quotidianamente vengono realizzate dagli altri utenti.



Figura 16 - Ecosistema The Sandbox

-**VOXEDIT**: è un tool di modellazione in voxel 3D semplice e intuitivo da utilizzare, consente sia la creazione di nuovi asset, sia la loro animazione. Attraverso questo tool è possibile realizzare qualunque cosa reale o

immaginario, per esempio persone, animali, vegetali, edifici, ecc. Una volta esportati e caricati sul Marketplace l'oggetto creato diventerà un vero e proprio asset di gioco.

-**MARKETPLACE**: il mercato di the Sandbox è web-based e consente agli utenti di caricare, pubblicare e vendere le loro creazioni, chiamati ASSET realizzati tramite il VOXEDIT ed esportate come token non fungibili di tipo ERC-721 e ERC-1155.

-**GAME MAKER**: chiunque possiede degli assets, sia realizzati in autonomia, sia acquistati sul marketplace potrà utilizzarli per realizzare un gioco attraverso il Game Maker. Questo consente agli utenti di posizionare ed utilizzare i loro Assets all'interno di una LAND (anch'essa un token ERC-721) che l'utente deve possedere nel mondo virtuale. Questa Land può essere decorata con gli Assets e possono essere implementate meccaniche di gioco attraverso un sistema a nodi.

4.1.3 SAND token e LAND

SAND è una parte fondamentale della piattaforma. Analizzandone il suo lato tecnico è un token ERC-20 costruito sulla rete Ethereum e funge da base per ogni tipo di transazione all'interno dell'ecosistema. Viene utilizzato per:

1. **Accesso alla piattaforma Sandbox**: gli utenti spendono SAND per giocare, acquistare oggetti di gioco, personalizzare il proprio avatar. Gli stessi utenti partecipando alle attività all'interno dell'ecosistema potranno guadagnare SAND. I creatori e gli artisti utilizzano i SAND per acquistare Assets, terreni e tramite staking.
2. **Governance**: chi possiede dei SAND potrà partecipare alle decisioni sulla piattaforma, utilizzando una struttura DAO. Gli utenti che ne hanno il diritto possono per esempio prendere decisione sulle nuove funzionalità da inserire, sulla creazione di nuovi giochi e sulla prioritizzazione della Roadmap.
3. **Staking**: SAND consente lo staking, questo permette entrate passive ai possessori di terreni che decidono di metterli in staking.

4. **Modello Fee Capture:** il 5% di tutto il volume delle transazioni effettuate tramite il token SAND sarà assegnato allo Staking Pool e suddiviso fra i possessori di token e la "fondazione".
5. **Fondazione:** il ruolo della Fondazione è quello di supportare l'ecosistema di The Sandbox, offrendo ricompense che servono ad incentivare lo sviluppo della piattaforma.

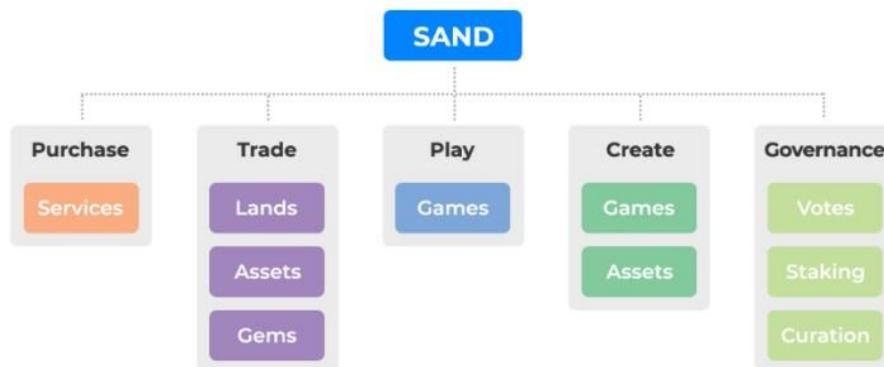


Figura 17 - SAND casi d'uso

Per garantire il successo di the Sandbox è stato sviluppato un approccio a quattro Stakeholders. I ricavi generati attraverso la piattaforma vengono distribuiti a quattro differenti attori:

1. **Foundation Pool:** progetto per garantire che le entrate generate attraverso il gioco siano ridistribuite per sostenere la crescita dell'ecosistema.
2. **Staking Pool:** progettato per fornire incentivi ai possessori di token che stanno bloccando i loro fondi attraverso Smart Contract.
3. **Tesoreria aziendale:** rappresenta il SAND di proprietà della società da cui proviene la vendita di beni. I SAND generati in questo modo saranno rivenduti al mercato per pagare le spese operative.

4. **Riserva Aziendale:** costituita dal 20% di token totali, tale riserva sarà alimentata dalla cessione di beni di proprietà della società con 6 mesi di blocco.



Figura 18 - Il flow di SAND

Le LAND, definibili anche come terreni, sono token virtuali supportati tramite blockchain, che utilizzano lo standard ERC-721 per NFT. Questi terreni consentono ai giocatori di possedere una parte del Metaverso e quindi di essere in grado di ospitare all'interno di essa contenuti come attività o giochi. Il Metaverso Sandbox si basa su una mappa costituita da 166464 Lands (408*408). Queste Lands possono essere utilizzate in prima persona dall'utente che li possiede per costruire una propria attività, ma possono anche essere affittate ai creatori di contenuti. La Land pur presentando degli edifici prefabbricati è completamente editabile.

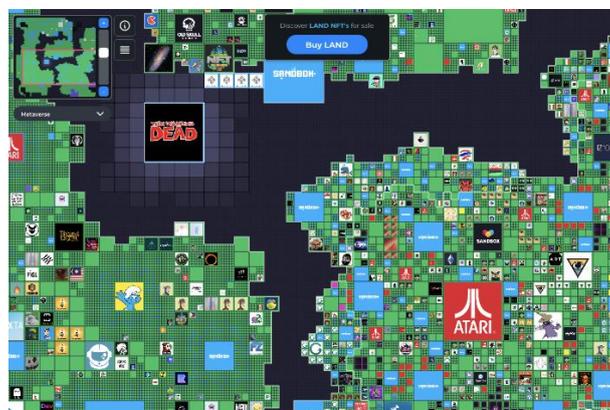


Figura 19 - Lands Map, the Sandbox

4.2 Decentraland

"Decentraland provides an infrastructure to support a shared virtual world, also known as a metaverse. It consists of a decentralized ledger for land ownership, a protocol for describing the content of each land parcel, and a peer-to-peer network for user interactions." [\[32\]](#)

4.2.1 Overview

Decentraland è una piattaforma di realtà virtuale che si basa sulla blockchain di Ethereum. All'interno di questo mondo virtuale gli utenti possono creare le proprie esperienze e monetizzare i propri contenuti. L'ecosistema Decentraland non ha un proprietario unico ma è posseduto da tutta la community, che ha quindi il pieno controllo sulla governance. I proprietari possono controllare il contenuto che viene pubblicato sulla loro porzione di terreno che viene identificata da un insieme di coordinate (x, y).

I contenuti in Decentraland possono variare da modelli 3D statici, a modelli animati o interattivi, fino ad arrivare a vere e proprie esperienze videoludiche. La Land è una NFT trasferibile e conservato tramite smart contract. Questo può essere acquistato tramite il token ERC20 di Decentraland ovvero MANA. MANA può essere utilizzato anche per fare acquisti nel mondo di bene e servizi digitali, essendo una criptovaluta a tutti gli effetti. Ancora una volta la particolarità, come già visto quando abbiamo parlato di the Sandbox, è che l'ecosistema non è controllato da un'organizzazione centralizzata ma da tutti gli utenti che ne fanno parte.

In altre parole, Decentraland consiste in un registro decentralizzato per la proprietà terriera, un protocollo che descrive il contenuto di ogni appezzamento di terreno ed una rete peer-to-peer per la gestione delle interazioni fra utenti. Decentraland mira a stabilire una rete che consente ai suoi creatori di contenuti di possedere e catturare tutti il valore dei contributi dati.

Il team ha iniziato a lavorare sul progetto dal 2015, all'epoca le tecnologie utilizzate erano in una fase embrionale, sia la tecnologia blockchain sia l'adozione di *cryptoassets*.



Figura 20 – Decentraland

Decentraland nasce come un sistema utile per l'allocazione delle proprietà immobili digitali su una blockchain. Ogni immobile digitale, inizialmente, è stato progettato come un pixel su un'infinita griglia 2D. Ogni pixel di questa griglia conteneva metadati che identificavano il proprietario e descrivevano il colore del pixel.

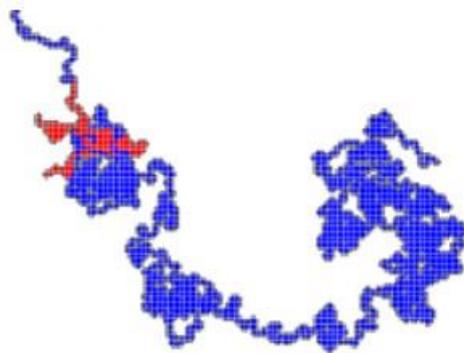


Figura 21 - Mappa primordiale Decentraland

Alla fine del 2016, il team ha iniziato a sviluppare la cosiddetta *Bronze Age*, un mondo virtuale 3D suddiviso in appezzamenti di terreno. Ogni proprietario di un appezzamento poteva associare la terra con un hash di riferimento ad

un file, utilizzando una blockchain Bitcoin modificata ad-hoc. Attraverso tale riferimento gli utenti potevano utilizzare una *Distributed Hash Table (DHT)* e tramite *BitTorrent* scaricare un file che conteneva i modelli e le texture da visualizzare nell'appezzamento in questione. La versione successiva di Decentraland, *the Iron Age*, permette di creare applicazioni su Decentraland, distribuirli ad altri utenti e monetizzarli.

L'idea è quella di avere una comunicazione peer-to-peer, un sistema intuitivo di scripting che permette l'implementazione di contenuti interattivi ed un sistema di pagamenti veloci in criptovalute per le transazioni all'interno del Metaverso. Strato fondamentale risulta essere la comunicazione per le esperienze sociali, tramite animazioni, posture, chat vocale ed altro.

I vari appezzamenti di terreno all'interno dell'ecosistema sono adiacenti fra di loro, e le nuove terre che man mano vanno a crearsi saranno sempre contigue a quelle esistenti. Questa adiacenza consente agli utenti di scoprire nuovi contenuti o di creare nuovi distretti dedicati a temi ben precisi. Mentre un classico dominio web può avere infiniti collegamenti ipertestuali, un terreno su Decentraland ha un numero fisso di adiacenze. Gli utenti, quindi, possono viaggiare fra le terre adiacenti ed interagire con le applicazioni che queste presentano.

La proposta di valore di Decentraland per gli sviluppatori di applicazioni è che possono farlo pienamente sfruttando le interazioni economiche fra gli utenti. Per permettere queste interazioni economiche, la piattaforma deve consentire lo scambio di valuta, beni e servizi. Per fare ciò Decentraland presenta un sistema di base che consente un sistema globale ed intuitivo di pagamenti tra due utenti qualsiasi su internet.

Il sistema di scripting a disposizione degli sviluppatori consente ad questi di sviluppare le interazioni fra gli utenti e l'applicazione; l'applicazione sarà eseguita esclusivamente sul lato client ma consente flussi di dati diversi che vanno dagli effetti sull'applicazione in locale, passando per una tradizionale architettura client-server, fino al P2P.

Agli sviluppatori che programmano nuove applicazioni verranno dati dei benefici come micropagamenti veloci ed economici, archiviazione decentralizzata ed altre funzioni speciali abilitate grazie all'utilizzo di tecniche crittografiche sviluppate tramite Smart Contract su blockchain.

Per favorire lo scambio di beni virtuali, devono essere previsti incentivi economici, inoltre un contenuto statico potrebbe essere copiato, viene quindi implementato un sistema di identità che stabilisce il vero autore di un'applicazione, parliamo ancora una volta di NFT; gli utenti saranno in grado di tracciare e verificare il consenso di un autore tramite firme crittografiche. Per quanto riguarda il linguaggio di scripting di Decentraland, consentirà agli sviluppatori di poter interagire con ogni parte del sistema, sarà possibile creare scene 3D statiche e dinamiche, creare trame di gioco ed implementarle, gestire la fisica ed anche i pagamenti e molto altro.

Per quanto riguarda la strategia di marketing, i marchi che decidono di approcciarsi a Decentraland possono pubblicizzare i propri prodotti, servizi o eventi tramite i cartelloni pubblicitari sparsi nei vari appezzamenti di terreno, oltre a questo potrebbero creare delle vere e proprie esperienze virtuali all'interno di Decentraland, come eventi o giochi.

Non essendoci specifiche tecniche da rispettare, qualunque idea potrebbe essere realizzata all'interno dell'ecosistema, se pensiamo a dei casi d'uso potremmo pensare alla formazione, all'istruzione, alla progettazione 3D, al turismo e molto altro.

4.2.2 Ecosistema

Il protocollo di Decentraland è costituito da tre livelli fondamentali: Livello di Consenso, di Contenuto della Terra e Real-Time.

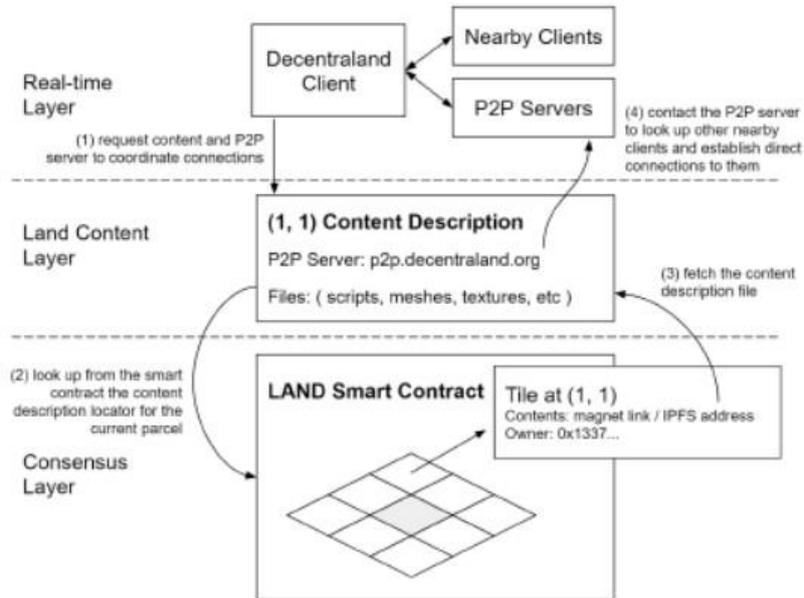


Figura 22 - Protocollo Decentraland

1. **Livello di Consenso:** Decentraland utilizza degli Smart Contract su blockchain Ethereum. Questo livello serve per mantenere il registro che contiene le caratteristiche di ogni appezzamento di terreno pubblico, garantendo inoltre trasparenza. Gli appezzamenti di terreno, anche in questo caso, vengono definiti come LAND; ognuna di esse presenta delle coordinate (x,y) che risultano univoche. Ogni qualvolta un utente si collega, questo sarà connesso sulla rete Ethereum e sarà in grado di recuperare gli aggiornamenti all'interno dell'ecosistema grazie alla lettura dello stato degli Smart Contract. Per acquistare dei terreni viene utilizzata la moneta di Decentraland ovvero Mana¹⁹, quello che accade è che una volta acquistata una terra, i MANA utilizzati saranno distrutti e comparirà una nuova voce nel registro distribuito che descrive la LAND. Importante notare come le nuove terre devono necessariamente essere adiacenti a quelle già presenti.

¹⁹ Token ERC20 fungibile di fornitura fissa

- 2. Livello di Contenuto della terra:** Decentraland utilizza un sistema decentralizzato per la distribuzione dei vari contenuti nell'ecosistema. Ogni qualvolta deve essere recuperata un pacco, viene recuperato il file ad esso associato tramite lo smart contract, questo attraverso l'utilizzo di BitTorrent permetterà di scaricarlo il contenuto. Il vantaggio di aver utilizzato un sistema decentralizzato è sicuramente quello di far esistere tutto l'ecosistema finché esisteranno utenti che distribuiscono contenuti, spostando i costi di gestione sugli utenti ed eliminando ogni forma di sistema centralizzato. Un altro vantaggio di questo sistema è una forte resistenza alla censura, in quanto non esiste un'entità centrale che possa modificare il contenuto o le regole sviluppate dagli utenti.

La descrizione dei pacchi sopra citati deve contenere l'elenco dei servizi ospitati, il posizionamento degli oggetti all'interno della LAND, il loro comportamento, i riferimenti alle mesh 3D, le loro texture, eventuali file multimediali come audio e video ed infine deve contenere anche le informazioni sull'interazione P2P, ciò consente al client di connettersi ad un server che esegue il bootstrap e permette la connessione fra utenti.

- 3. Livello Real-Time:** come sopra citato la connessione fra utenti all'interno dell'ecosistema avviene tramite una connessione P2P con l'ausilio di server ospitati dai possessori di LAND o da terzi parti. Ancora una volta viene quindi eliminato il concetto di server centralizzato. Uno dei problemi principali di questa soluzione è la gestione e la manutenzione dei server, l'idea è quella di utilizzare protocolli sempre più leggeri come STUN13. Per quanto riguarda i contenuti più difficili a livello computazionale da gestire, per esempio la chat vocale, l'incentivo a mantenere attivo un server potrebbe essere quello dei micropagamenti.

Esistono infine altri due livelli importanti all'interno dell'architettura di Decentraland:

4. **Canale di Pagamento:** un possibile metodo di gestione dei pagamenti potrebbe essere quello di reti HTLC come Lightning, ma l'idea è quella di utilizzare dei sistemi *low-trust hub-and-spoke* che consentirebbero transazioni rapide e a basso costo. Il sistema di pagamento risulta fondamentale in Decentraland per migliorare la qualità del servizio offerto e la gestione dei server P2P. Inoltre, un sistema di pagamento decentralizzato risulta essere rivoluzionario, in quanto i più comuni sistemi tramite carta di credito, hanno alla base la fiducia che l'utente dà all'ente centralizzato che ne garantisce il pagamento. Con un sistema decentralizzato il pagamento risulta diretto fra utente e sviluppatore, evitando qualunque rischio di truffa o falsificazione dell'asset. Oltre a ciò, alcuni servizi su Decentraland, come già ampiamente descritto, hanno necessità di micropagamenti che servono ancora una volta a coprire i costi di gestione ed incentivare più sviluppatori e utenti ad entrare nel Metaverso.
5. **Sistema di Identità:** Decentraland utilizza sistemi di identità decentralizzati per costruire un livello di proprietà. Questo paradigma deve consentire a tutti gli utenti di verificare facilmente il possesso ed il consenso di un autore, per fare ciò vengono utilizzate chiavi pubbliche e firme digitali di tipo *human-readable*²⁰, questo grazie a sistemi come *uPort* o *Ethereum*.

4.2.3 MANA token e LAND

Sia il concetto di MANA che il concetto di LAND sono stati lanciati in Decentraland con l'avvento della *'the Iron Age'*. MANA è un token fungibile che rispecchia gli standard ERC20, questo viene sia utilizzato per l'acquisto di LAND, sia per effettuare acquisti all'interno dell'ecosistema che possono essere sia beni che servizi.

Per quanto riguarda le LAND, uno dei loro punti di forza si basa sul fatto di essere adiacenti ad altre LAND, oltre a ciò, i possessori di LAND possono inserire all'interno di queste delle esperienze o dei giochi con la quale gli

²⁰ In italiano leggibili dall'uomo

utenti, che visitano la terra, possono interagire. Ogni terra non ancora acquistata è tutt'ora acquistabile tramite l'utilizzo di MANA, di base la terra ha un prezzo ma questo può variare se l'acquisto viene effettuato su marketplace secondari; quindi, MANA funge da proxy per valutare il prezzo di un appezzamento di terreno. L'utilizzo di MANA per l'acquisto di beni e servizi all'interno di Decentraland ne fa oscillare il prezzo. Verranno premiati tutti gli sviluppatori che decidono di investire tempo e risorse sul progetto, inoltre la fondazione di Decentraland organizza eventi, giochi ed applicazioni per invogliare più utenti a connettersi. Ad ogni nuovo utente verrà assegnata un'indennità, in modo tale da consentirgli di partecipare attivamente all'economia del mondo.



Figura 23 - Mana token, token di Decentraland

4.3 OVR

"OVR makes it possible for users provided with a mobile device or smart glasses to live interactive augmented reality experiences customised in the real world." [\[33\]](#)

4.3.1 Overview

OVR è una piattaforma AR open source basata sulla blockchain Ethereum. L'idea alla base di OVR è quella di concedere a tutti gli utenti che possiedono uno smartphone o qualunque tipo di supporto alla realtà aumentata, di vivere delle esperienze virtuali in funzione della posizione geografica nella quale si trovano. La blockchain Ethereum viene utilizzata per decentralizzare tutte le

dinamiche relative allo scambio di token. Anche il token di OVR è ovviamente di tipo ERC20.



Figura 24 - OVR Metaverse

Le OVRLand sono pacchi archiviati sulla blockchain che costituiscono lo strato digitale della suddivisione del mondo reale in esagoni. Anche OVRLand è un token, in questo caso stiamo parlando di un ERC721 quindi un token non fungibile che permette di ottenere il possesso digitale di beni come le OVR Lands o le OVRExperiences²¹.

Le varie OVR Lands sono scambiabili fra utenti utilizzando il mercato proprietario della piattaforma OVROwner. I possessori delle varie terre possono decidere il contenuto di queste, e possono creare delle esperienze di tutti i tipi che spaziano da modelli 3D statici ad animazioni complesse.

Attualmente è possibile accedere alle OVRExperiences tramite dispositivi Android ed IOS o con Hololens e Magic Leap. Le OVRExperiences possono essere costruite tramite SDK basato su Unity3D.

OVR implementa un sistema di pubblicità basato su editore e inserzionista. Ogni possessore di una Land può inserire all'interno della sua terra dei contenuti sponsorizzati e guadagnare dei token da questo.

²¹ Esperienze in realtà aumentata nel mondo OVR

4.3.2 Ecosistema

L'ecosistema OVR è costituito da diverse parti che si uniscono e collaborano fra di loro. Per quanto riguarda gli utenti ne esistono di vari tipi all'interno dell'ecosistema:

In primis abbiamo gli OVR Owner che utilizzano il token OVR per acquistare le OVR LAND ed inserire al loro interno le esperienze in realtà aumentata. Gli OVR Creator invece si occupano di creare le OVR Experiences. Gli OVR Miners attivano i nodi che permettono la decentralizzazione di OVR. Gli OVR User sono tutti gli utenti che entrano all'interno della piattaforma per provare le esperienze presenti. Gli OVR Advertiser utilizzano i token per pubblicare sponsorizzazioni all'interno delle terre. Infine, gli OVR Staker mettono in stake gli OVR token e votano per gli IPFS OVRNodes.

Come già detto gli utenti della community possono sviluppare le proprie applicazioni AR tramite SDK OVR su Unity3D e possono quindi o pubblicarlo su una terra, se ne possiedono una, oppure vendere quanto realizzato. L'archiviazione di tale esperienza avviene all'interno di IPFS OVRNode. Per garantire la correttezza della transazione vengono utilizzati gli Smart Contract.



Figura 25 - Processo di vendita di OVR Experiences

Un concetto simile è ritrovabile negli OVR Adv: gli strumenti presenti nell'SDK permettono la gestione di articoli promozionali che possono essere per esempio banner in 3D o, più genericamente, planimetrie con texture assegnabili per sponsorizzare un qualche servizio, prodotto o evento. Il mercato gestisce l'acquisto e la vendita di questi spazi sia su base temporale che su base delle prestazioni ottenute.

Come abbiamo già detto in precedenza, l'esperienza offerta da OVR è in AR, l'ambiente circostante viene quindi tracciato tramite delle telecamere ed il software inserisce in tempo reale delle immagini digitali, visualizzabili tramite il display. Più precisamente viene utilizzata una tecnologia AR persistente, ovvero un utente può vivere la stessa esperienza virtuale più volte e in qualsiasi tempo e luogo. Questo è possibile grazie al salvataggio della scansione ambientale.

In OVR il mondo viene setacciato tramite degli esagoni, per fare ciò l'applicazione ha necessità di accedere in real-time alla posizione dell'utente. Un sistema che si basa solo su GPS non è però abbastanza affidabile in quanto la precisione del dispositivo non è sempre perfetta e potrebbe crearsi un errore anche di decine di metri fra la posizione rilevata e la reale posizione. È per questo che si è cercata un'alternativa più efficace. Il concetto è quello di unire due tecnologie differenti per il posizionamento: lo SLAM, funzionante a livello locale, ed il GPS, funzionante a livello globale, in modo da confrontare i dati rilevati, conservare quelli simili e scartare quelli anomali. Come abbiamo visto nel secondo capitolo lo SLAM è una tecnologia che ci permette di riconoscere gli elementi nell'ambiente grazie ad una videocamera e ci permette di rilevare in modo molto dettagliato la posizione della videocamera stessa. Tale sistema porta ad un'innovazione concettuale nell'utilizzo del GPS in quanto, unendo le due tecnologie, otteniamo una precisione molto più accurata, grazie all'aggiunta della variabile temporale al GPS.

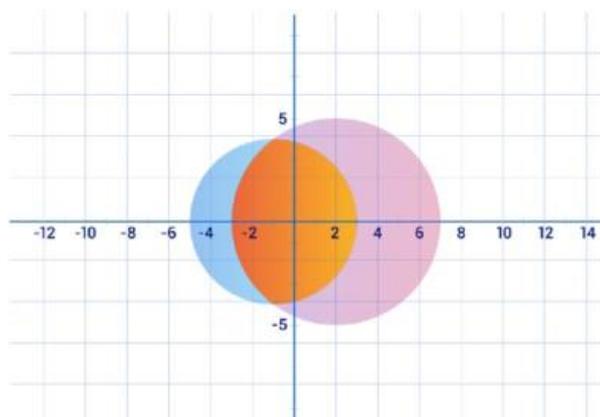


Figura 26 - Esempio pratico unione SLAM e GPS

Come possiamo notare in **figura 26** sopra, l'area viola rappresenta i dati GPS attesi, mentre quella blu quelli registrati. Analizzandoli separatamente non riusciremmo a capire la posizione corretta dell'utente e quale delle due contenga l'informazione corretta, ma intersecandoli otteniamo un'area molto curata, la zona arancione, zona nella quale l'utente si trova. La registrazione continua consente di migliorare il segnale e rendere il calcolo sempre più accurato.

Per contribuire a migliorare ancora la precisione vengono utilizzati altri due attori, il target e lo scanning. Questi permettono di compensare meglio l'imprecisione dovuta al GPS, posizionando correttamente gli elementi virtuali nella scena. Utilizzando il target, viene preso come riferimento un oggetto presente nella scena e a seconda delle sue dimensioni, della sua posizione e della sua inclinazione, definite manualmente dall'utente, vengono fornite le corrette coordinate geografiche. Inserendo le dimensioni del target si può calcolare in maniera esatta la distanza tra il target ed il dispositivo dell'utente. Otteniamo così il vettore distanza. Una volta calcolato ciò, è compito dello SLAM ancorare gli oggetti virtuali al mondo reale. La seconda tecnologia invece è quella basata sullo *scanning* preventivo del luogo di interesse. Per utilizzare questa metodologia è necessaria che un umano, definito come OVRLandscanner, vada fisicamente sul luogo per scansionarlo. Le informazioni raccolte forniranno la mappa dell'ambiente nella quale inserire gli assets digitali. L'SDK permette agli sviluppatori di decidere la modalità di tracciamento che si vuole utilizzare.

Le risorse OVR sono ospitate dagli OVRNodes che utilizzano un protocollo P2P IPFS per garantire la decentralizzazione. Nella prima fase erano presenti dieci nodi master dislocati in aree geografiche differenti. Al lancio della piattaforma questi nodi sono stati sovvenzionati da OVR. Viene così inserito un sistema di remunerazione automatica in base allo storage ed alla banda utilizzata dagli utenti privati. Ogni proprietario di OVRLand paga una quota di hosting per ospitare il contenuto IPFS, tale quota verrà suddivisa a tutti i nodi che ne permettono il funzionamento in modo bilanciato rispetto alle risorse impiegate.

Come precedentemente introdotto l'applicazione gira su rete blockchain Ethereum. A causa degli elevati costi di gas su rete Ethereum, OVR utilizza una tecnologia sidechain collegata alla rete principale Ethereum. Gli NFT presenti nell'ecosistema, che rappresentano le OVR Lands, saranno scambiate su XDai²² eliminando così i costi del gas. Sarà sempre possibile trasferire NFT conosciuti sulla rete mainchain Ethereum e venderli su OpenSea²³. La sicurezza è garantita dagli Smart Contract che permettono gli scambi di token dalla rete principale alla sidechain e viceversa.

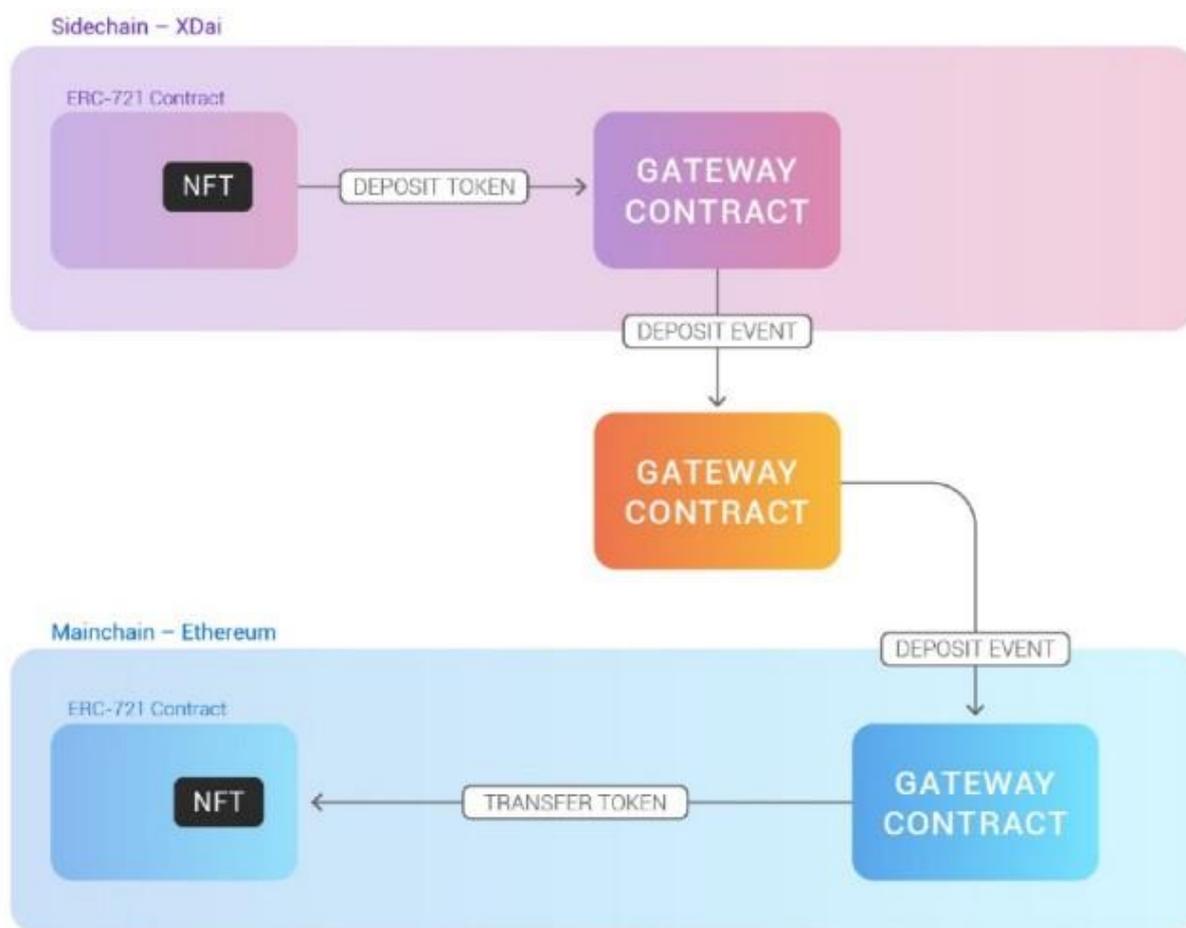


Figura 27 - Infrastruttura blockchain OVR

²² Stable coin dalle basse commissioni

²³ Piattaforma specializzata nella compravendita di criptovalute e NFT

Grazie all'utilizzo di una sidechain come XDai l'esperienza mobile è migliore, rimuovendo le varie meccaniche di commissione necessarie sulla mainchain Ethereum. L'esperienza sarà quindi molto simile a quella del Web 2.0, migliorando la familiarità dell'applicazione e agevolando l'adozione di massa.

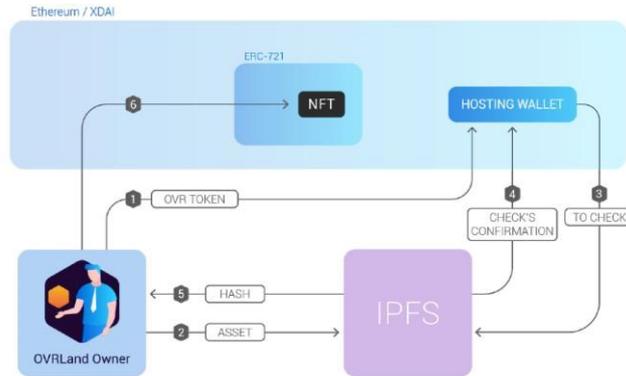


Figura 28 - Infrastruttura blockchain - OVR Owner

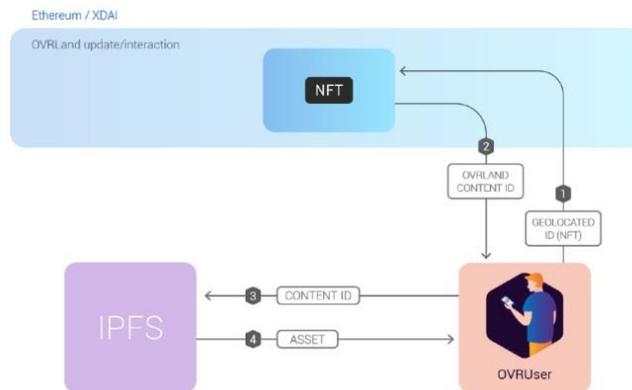


Figura 29 - Infrastruttura Blockchain - OVRUser (privata)

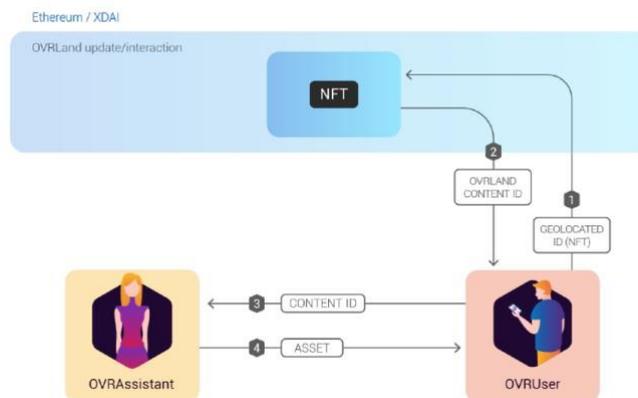


Figura 30 - Infrastruttura blockchain - OVRUser (pubblica)

Infine, per terminare il discorso sull'ecosistema introduciamo l'OVRAssistant. Questo è un avatar umano modellato in 3D che si basa su tecnologie di intelligenza artificiale e sintesi vocale. L'assistente è utile per dare delle informazioni sul luogo nella quale ci troviamo; infatti, è stato addestrato con una rete neurale su Wikipedia ed ha una conoscenza programmabile dal proprietario della terra. Si può interagire con l'avatar attraverso la voce.



Figura 31 - OVRAssistant

L'intelligenza dell'assistente sarà alimentata da un insieme di diverse tecnologie PNL che possono essere utilizzate anche congiuntamente, che sono:

1. **Modello di ricerca ad albero:** domande e risposte sono strutturate all'interno di una struttura ad albero nidificato.
2. **Riconoscimento di modelli locali:** utilizza NLP basati su BERT*, una rete neurale sviluppata da Google. Questa è una rete non supervisionata ma bidirezionale addestrata su Wikipedia.

4.3.3 OVR Token e OVR Lands

Il token OVR è un token ERC20 fungibile ed è utile in ogni scambio all'interno del mondo OVR. Il token consente di sfruttare appieno la blockchain basta su Ethereum ed anche integrarsi ottimamente con le infrastrutture esterne.

L'OVR token si basa su due diverse tipologie di token. OVRToken ed OVRLand. OVRToken viene utilizzato per interagire con la piattaforma in svariati modi. Per fare degli esempi serve per la governance, per l'acquisto di OVRLand, per

lo staking dei nodi e il liquidity mining, per acquistare asset sullo store o per acquistare un'esperienza sulla piattaforma e quindi pagare un OVRCreator, per farsi pagare per provare la propria esperienza, per affittare uno spazio virtuale e per molto altro. OVRLand è invece un token non fungibile di tipo ERC721 che rappresenta la proprietà di una LAND. La sostenibilità del progetto sarà garantita da tre differenti flussi di token non correlati, ovvero:

1. **Commissioni di cambi:** Ogni volta che viene scambiato un OVRLand vi sarà una commissione pari al 5% della transazione. Anche lo scambio di assets avrà una commissione pari al 5% della transazione. Verrà infine generato un altro 5% di commissione dall'affitto di OVRLand a editori ed inserzionisti.
2. **Acquisti in app:** Gli utenti potranno acquistare skin ed abiti per il proprio avatar, potranno acquistare speciali pass per eventi virtuali privati o crediti per i giochi in AR/VR.
3. **Commissioni IPFS per l'hosting di asset:** le risorse digitali saranno archiviate su IPFS. Per mantenere gli assets bisognerà quindi pagare una cifra. I token raccolti saranno suddivisi in due parti, la prima spetta ai fornitori di servizi IPFS, la seconda sarà utilizzata per la remunerazione dei gestori della rete, per lo sviluppo futuro dell'ecosistema e per promozioni mirate ad ampliare il bacino di utenti.

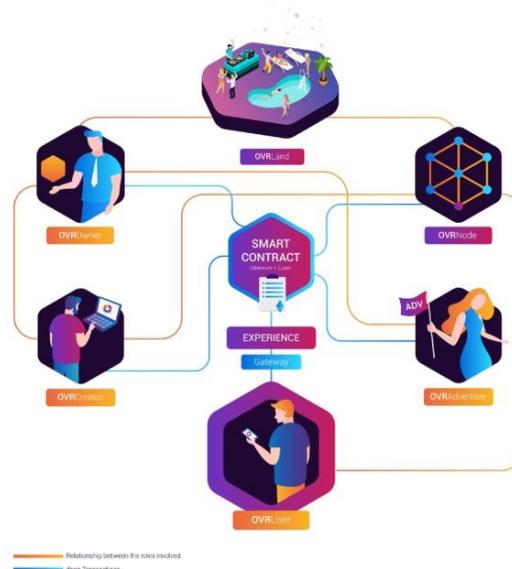


Figura 32 - Funzionamento OVRToken

Come anticipato la struttura delle OVR Lands è esagonale. La somma di tutti gli esagoni copre l'intero globo terrestre. La dimensione standard di questi esagoni è di 300 metri quadrati, quindi in totale avremmo 1660954464112 OVR Lands. Ognuno di questi esagoni può a sua volta essere diviso in sette esagoni, rendendo la localizzazione ancora più precisa. Oltre a dare un riferimento spaziale per posizionare i contenuti digitali, le OVR Lands come detto, sono dei veri e propri Asset, degli NFT di tipo ERC721, questo garantisce decentralizzazione e diritto di proprietà, oltre che resistenza alla censura. Uno dei principali problemi dell'utilizzo degli Smart Contract è quello della perdita di chiave privata, perdendo la chiave privata una terra sarebbe persa per sempre. Per ovviare a questo problema OVR ha implementato una soluzione basata sugli affitti che non necessita dell'accettazione del proprietario della terra. In ogni caso se un possessore di Land possiede la chiave privata può rifiutare l'affitto ed utilizzare la propria terra come meglio crede. La scelta delle terre esagonali non è casuale, questo deriva dal fatto che gli esagoni vicini, innanzitutto sono sempre e solo posizionati sui lati e mai sugli angoli, inoltre la distanza fra i vicini ed il centro risulta sempre uguale, cosa non permessa utilizzando triangoli o quadrati.

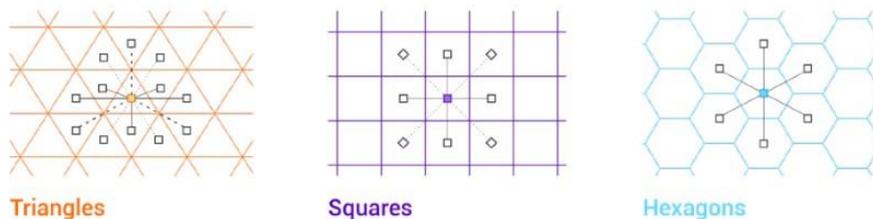


Figura 33 - OVR Lands

Gli esseri umani non sono in grado di ricordare i numeri come le parole, difatti un adulto riesce tranquillamente a ricordare più di 20000 parole, cosa impossibile per combinazione di numeri. Per questo motivo OVR ha brevettato un algoritmo che associa la posizione geografica di ogni Land, definita da longitudine e latitudine, ad una tripletta di parole facilmente memorizzabili. Per ottenere questo risultato è stato utilizzato un elenco di parole da Google N-Grams, che contiene 20000 parole più comunemente utilizzate nella lingua inglese.

5 Costruire il Metaverso

Nei capitoli precedenti abbiamo analizzato tutto quello che gira intorno al mondo del Metaverso. Siamo partiti dandone una definizione, passando per la sua storia e analizzandone le tecnologie che vengono mixate fra di loro, quali Realtà Virtuale, Realtà Aumentata e Blockchain. Abbiamo anche approfondito quali sono i Metaversi attualmente più interessanti nel panorama tech. Quello che ci resta da vedere è come costruire passo per passo un Metaverso, quali sono in generale i blocchi principali che ne costituiscono l'architettura e come renderlo un sistema decentralizzato attraverso l'utilizzo di Smart Contract su blockchain. In questo capitolo, quindi, partiremo analizzando l'architettura di un Metaverso per capire quali sono i blocchi principali che lo costituiscono. Lavoreremo in seguito con gli Smart Contract su rete blockchain per costruire un prototipo di mappa di Metaverso da 0, dove sarà possibile acquistare le varie proprietà presenti. Infine, analizzeremo un esempio di utilizzo di cliente del Metaverso, realizzando uno spazio virtuale che introduca la realtà aziendale di Var Group in un'esperienza immersiva sul Metaverso.

5.1 Architettura di un Metaverso

Possiamo suddividere l'architettura di un Metaverso in vari blocchi: l'immagine sottostante ce ne può dare un'idea.

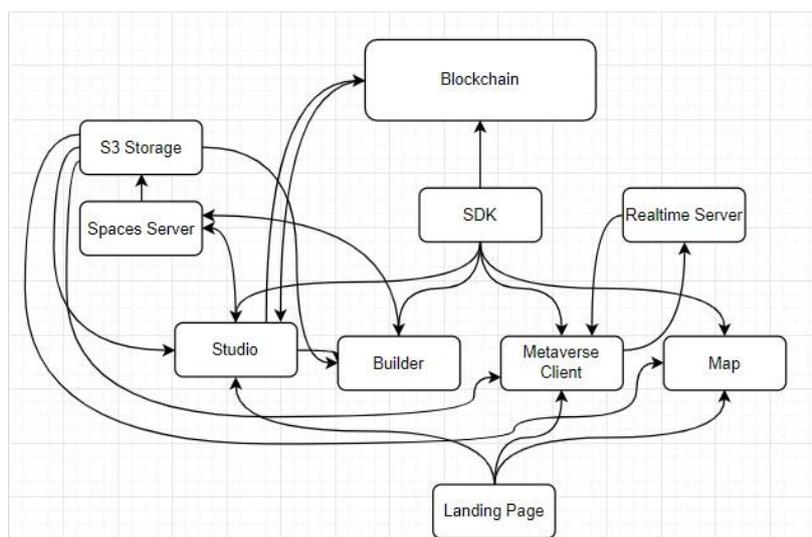


Figura 34 - Architettura Metaverso

Di base ogni Metaverso può avere un'architettura differente. Quella che andiamo ad analizzare in questo capitolo viene utilizzata attualmente sul Metaverso Open Source chiamato Edenlans [34] e può essere un buon esempio.

In questo caso abbiamo alla base cinque differenti applicazioni; notiamo come queste applicazioni sono tutte separate fra di loro, seppure comunicanti.

La prima applicazione è la Landing Page; una Landing Page è semplicemente una pagina web dove le persone possono connettersi per trovare i link e le informazioni sul progetto. È da questa pagina che si può accedere, direttamente o scaricando i software necessari, nelle altre applicazioni che compongono il Metaverso, quali lo studio, il Client del Metaverso e la Mappa. Il Builder nel nostro caso non è direttamente accessibile dalla Landing Page, bensì bisogna accedere allo Studio per utilizzarlo. Di solito le Landing Page sono delle normalissime pagine web statiche costruite tramite React o NextJS.

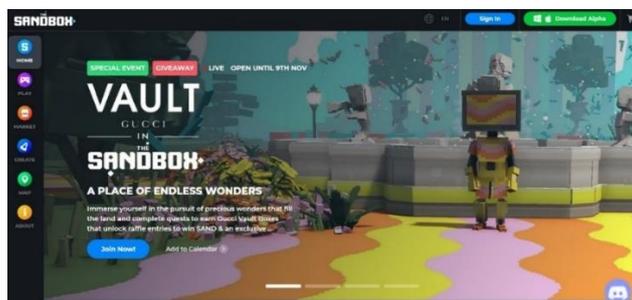


Figura 35 - Landing Page The Sandbox

Tutte le altre applicazioni che costituiscono un Metaverso utilizzano invece software come Metamask come provider, next.js, React, three.js come linguaggi di programmazione. In questo caso quindi non abbiamo più un'applicazione statica bensì delle applicazioni reali che comunicano direttamente con l'SDK e a volte con la blockchain.

Un altro blocco molto importante che è presente nella maggior parte dei Metaversi è lo Studio; questo offre all'utente la possibilità di gestire i propri token, come gli NFT e consente di connettersi alla rete blockchain per sincronizzare i dati.

Oltre a ciò, permette di accedere al Builder, in cui gli utenti possono costruire nuovi modelli, ambienti e in alcuni casi anche applicazioni. Per fare un esempio basti pensare al Game Maker di The Sandbox.



Figura 36 - VoxEdit - The Sandbox

Il Client del Metaverso è quella parte utile puramente per visualizzare i dati che vengono realizzati o manipolati tramite lo Studio ed il Builder. Di base l'utente accede in questa zona tramite il proprio avatar e può muoversi liberamente per esplorare le varie terre, dove i proprietari hanno costruito i propri edifici o le proprie esperienze.



Figura 37 - Metaverse Client - Decentraland

Infine, abbiamo la Mappa, la Mappa si presenta con una vista solitamente bidimensionale dall'alto che contiene tutti gli appezzamenti di terreno che costituiscono il Metaverso; attraverso un clic su un determinato appezzamento sarà possibile visualizzare le informazioni di quella specifica terra e se libera sarà possibile acquistarla.

Per verificare la proprietà di un appezzamento di terreno entra in gioco la blockchain, questa conterrà le informazioni di ogni LAND che verranno convertiti quindi in Smart Contract di tipo ERC721.



Figura 38 - Mappa OVR

Ogni token non fungibile sulla blockchain presenta un ID univoco. Quello che dobbiamo fare, è quindi riuscire ad accedere a questo ID per verificarne le proprietà. Per fare ciò abbiamo bisogno di un SDK che vivrà su npm²⁴ e che gli sviluppatori possono inserire all'interno del loro progetto e collegare con le varie applicazioni. Questo, quindi, è un punto centrale che contiene delle informazioni, delle variabili o delle funzioni alla quale tutte le applicazioni sopra descritte necessitano di accedere. Bisogna dunque creare un SDK che verifichi lo smart contract e che permetta a tutte le applicazioni sottostanti di creare un'istanza di quel contratto. L'SDK seppur non necessario è molto utile per non creare un codice ridondante, in quanto tutte e 4 le applicazioni sottostanti (studio, builder, client e map) necessitano di poter accedere alla stessa istanza; quindi, se non utilizzassimo l'SDK per ognuna di queste applicazioni andrebbe creato il codice e se qualcosa cambia in una, dovrebbe essere cambiata anche nelle altre 3. Dobbiamo quindi avere un SDK pubblico, al quale ogni utente può accedere in modo semplice. Pur avendo creato un SDK comune, non tutte le applicazioni necessitano delle stesse informazioni,

²⁴ Gestore di pacchetti per il linguaggio di programmazione JavaScript

per esempio nello studio dobbiamo mostrare il numero di token che l'utente possiede; per far ciò possiamo fare un check attraverso l'SDK, creando un'istanza sulla blockchain e leggendo i dati che ci interessano. Per renderlo più efficiente non andiamo a leggere sempre i dati che sono archiviati sull'IPFS, ovvero il meccanismo di archiviazione decentralizzato, in quanto risulterebbe un processo molto lento e dispendioso, quindi ogni volta che aggiorniamo il token NFT, per esempio cambiando il nome, la descrizione o anche i dati di build, questo sarà archiviato in un server di cache tramite node.js e quindi i dati saranno aggiornati su un bucket di archiviazione S3; non abbiamo quindi bisogno di un vero e proprio database, bensì di file che vengono modificati nel bucket S3. Questi file, quindi, possono essere letti quando servono in maniera rapida dallo studio, rendendo il processo molto più veloce. Per rendere il tutto sicuro abbiamo bisogno di una firma dell'utente sul provider; dunque, ogni richiesta verrà inviata con una firma, il server si occuperà di verificare che l'utente che ha richiesto la modifica ha il permesso per farlo e solo allora modificherà lo stato del token. Lo stesso principio è applicabile anche per il builder, in quanto anche da lì è possibile modificare il nostro token. A questo punto, per rendere l'esperienza utente ottimale, sia il Client del Metaverso, sia l'applicazione che gestisce la Mappa potranno leggere direttamente dal bucket in modo da rendere il processo di generazione dell'ambiente molto veloce, cosa infattibile, come già detto, se si leggesse ogni volta dall'IPFS. Né il Client, né la Mappa possono modificare il contenuto presente nel bucket.

Quello che abbiamo creato, attraverso quanto descritto sopra, è un'applicazione completamente centralizzata, la blockchain però serve a verificare se l'utente possiede il token che gli permette di utilizzare tale applicazione. Per rendere decentralizzata totalmente l'applicazione, pur mantenendo la velocità che un server locale può offrirci, diamo la possibilità allo studio di accedere direttamente alla blockchain e quindi poter sia modificare lo smart contract presente nella rete, sia di leggere i dati presenti sulla blockchain. Questo passaggio rende il nostro Metaverso interoperabile, in quanto è possibile scrivere anche da altre applicazioni sulla blockchain e gli

aggiornamenti saranno direttamente letti dal nostro studio che andrà ad aggiornare la versione locale del nostro token. In più, il file viene conservato come Json, quindi i dati possono essere facilmente formattati e governati da schemi in qualsiasi modo o forma.

L'ultimo pezzo del puzzle che serve per completare l'architettura del nostro Metaverso è un server Real-Time. Questo interagisce solamente con il Client, in quanto è necessario sapere quanti utenti sono collegati al suo interno per visualizzarne tutti gli avatar e il loro movimenti.

5.2 Creazione di una Mappa

Abbiamo visto come uno dei blocchi fondamentali che costituiscono un Metaverso è la Mappa. Una Mappa è quell'applicativo che ci permette di acquistare un terreno virtuale all'interno di un Metaverso o di poter accedere alle esperienze associate ad ogni terreno già acquistato da noi o da altri utenti. Possiamo assimilare l'acquisto di un terreno virtuale a quello di un acquisto di un terreno nel mondo reale con la differenza che la proprietà non sarà fisica bensì sarà permanente sulla blockchain. Gli utenti che accedono potranno prendere parte a questa terra virtuale acquistata, all'interno della quale il possessore potrà creare le proprie esperienze.

L'obiettivo principale di possedere una terra virtuale è quindi molto simile a quello di possedere un appezzamento di terra reale, ovvero quello di creare un business che giri intorno a quello spazio. Per fare alcuni esempi potremmo utilizzare quello spazio per permettere l'acquisto di qualcosa, per mostrare le nostre opere d'arte o per organizzare un evento o un concerto.

Quello che analizzeremo quindi è come poter creare un'applicazione metaverse in 3D all'interno del browser utilizzando React, three.js e gli Smart Contract su blockchain grazie all'utilizzo di Solidity.

5.2.1 Overview

Lo scopo di questa parte progettuale è stata quella di creare un'applicazione metaverse che permetta all'utente che accede alla pagina web di poter acquistare e possedere terreni virtuali. L'obiettivo principale è quello di creare una mappa molto simile a quelle presente negli attuali Metaversi, come quelli analizzati nel capitolo 4, ovvero The Sandbox, Decentraland e OVR.

Possiamo schematizzare ad alto livello il nostro Metaverso come in **figura 39**.

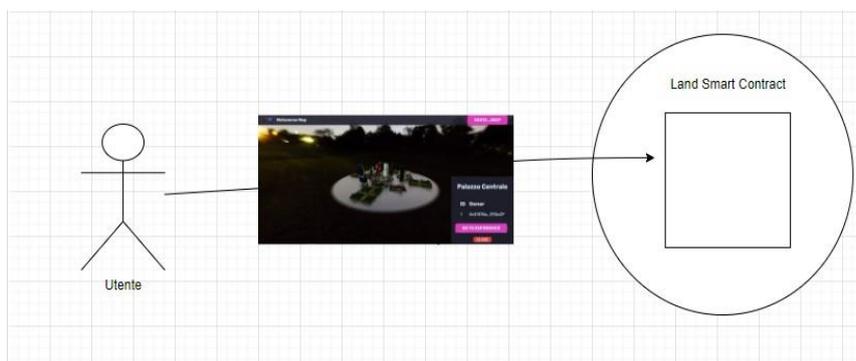


Figura 39 - Overview Architettura Mappa

Tale applicazione web è stata realizzata tramite l'utilizzo di React.js [35] ed il supporto di three.js [36]. React è una libreria basata su JavaScript utile per creare interfacce utente. Il meccanismo principale di funzionamento di React si basa sull'unione di varie componenti UI. React rende semplice la creazione di UI interattive. Ad ogni cambio stato le varie componenti, che dipendono da quello stato, saranno aggiornate. Per utilizzare dei modelli 3D all'interno dell'applicazione web, viene usata la libreria three.js in particolare le componenti messe a disposizione da Three Fiber [37].

Inizialmente la mappa sarà costituita da un piano circolare vuoto che presenta 13 porzioni quadrate, rappresentanti le terre acquistabili. Una volta acquistato uno di questi appezzamenti, al prezzo di 1 Ether, verrà caricato un modello 3D associato a quella terra virtuale. È possibile visualizzare la mappa con tutte le 13 terre acquistate in **figura 40**. In questa immagine possiamo notare come si presenta l'interfaccia utente.

Questa presenta in alto una Navbar contenente il logo ed il nome dell'applicazione e l'identificativo del wallet collegato all'applicazione. Sotto la

Navbar è presente la nostra mappa nella quale è possibile, cliccando su uno degli edifici presenti, far comparire una modale che ci dà le indicazioni sul proprietario dell'edificio, se la terra è stata acquistata in precedenza da qualcuno, oppure indicazioni sul fatto che quell'appezzamento sia ancora disponibile. Nella stessa modale, tramite un apposito pulsante, se la terra non ha ancora un proprietario può essere acquistata, se ha già un proprietario è possibile collegarsi all'esperienza virtuale ad essa associata. Tali esperienze sono state realizzate basandosi sul Metaverso Spatial [38] che analizzeremo in seguito quando parleremo del progetto realizzato per VarGroup.

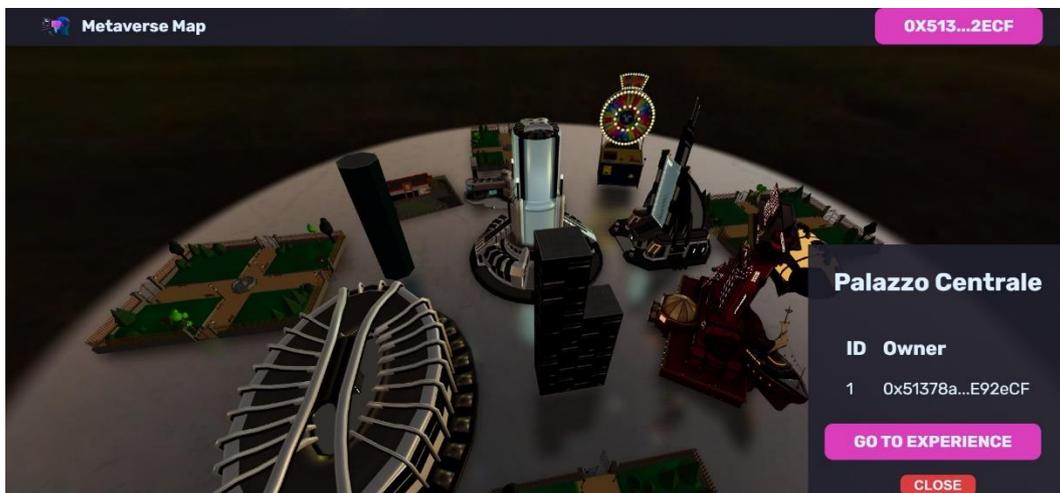


Figura 40 - Mappa con tutte le terre acquistate

Rianalizzando la **Figura 39** notiamo come la nostra applicazione web comunica direttamente con i nostri *Land Smart Contract*. Una *Land Smart Contract* è fondamentalmente un'applicazione sulla blockchain, scritta utilizzando il linguaggio di programmazione Solidity [39]. Per simulare una blockchain all'interno del mio pc ho utilizzato una *local development blockchain* chiamata Ganache [40]. Ganache è un software che permette di avviare rapidamente una blockchain Ethereum personale, utile in fase di test. Per sviluppare gli Smart Contract invece viene utilizzato Truffle [41], suite di strumenti utili per lo sviluppo e la distribuzione di Smart Contract su blockchain.

Per permettere il funzionamento di React e Truffle è stato utilizzato Node.js [42], ambiente di sviluppo runtime JavaScript che ci permette la gestione di pacchetti npm.

5.2.2 Struttura del Progetto ed Analisi Tecnica

La prima cosa necessaria per lo sviluppo della progettualità è stata quella di installare tutti i pacchetti necessari sopra citati. Una volta installato Truffle tramite il comando sulla Power Shell `npm install -g truffle`, ho creato la struttura del progetto in React tramite il comando `npx create-react-app metaverse`. Una volta creato il progetto React, è stato necessario inizializzare la dipendenza di truffle, utilizzando il comando `truffle init`; questo creerà delle nuove cartelle e dei nuovi file utili per la gestione degli Smart Contract.

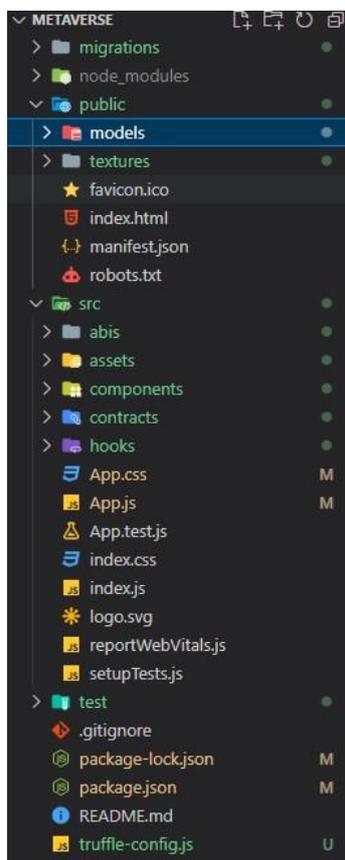


Figura 41 - Struttura del Progetto

In **figura 41** possiamo vedere come è suddiviso il nostro progetto. Un primo file importante è `truffle-config.js`. Questo permette di configurare host e porta di accesso per una rete di sviluppo dove testare il nostro applicativo.

Oltre a ciò, nello stesso file vengono definite le corrette directory dei contratti e la versione del compilatore necessaria per supportare quanto sviluppato. Per quanto riguarda le cartelle, la cartella *migrations* contiene dei file JavaScript utili per la distribuzione dei nostri Smart Contracts sulla rete blockchain. In particolare, sono presenti due file: il primo è un esempio creato in automatico da Truffle mentre il secondo, dal nome *2_deploy_contract.js*, è quello interessante in quanto permette di distribuire gli smart contract da noi definiti. Esso, infatti, contiene i valori Name, Symbol e Cost che il costruttore richiede come vedremo nel file *Land.sol*.

```
2_deploy_contracts.js U ●
migrations > 2_deploy_contracts.js > ...
1  |const Land = artifacts.require("Land")
2
3  |module.exports = async function (deployer) {
4  |    const NAME = 'Metaverse Map'
5  |    const SYMBOL = 'CAR'
6  |    const COST = web3.utils.toWei('1', 'ether')
7
8  |    await deployer.deploy(Land, NAME, SYMBOL, COST)
9  |  }
10
11
```

Figura 42 - File di migrazione dei nostri smart contracts su blockchain

La cartella *node_modules* è una cartella auto-riempita che contiene tutti i pacchetti necessari per il funzionamento delle varie dipendenze del progetto. Possiamo immaginare questa cartella come una cache per i moduli esterni da cui il progetto dipende.

La cartella *public* contiene al suo interno altre due cartelle molto importanti, *models* e *textures*, che contengono rispettivamente i modelli 3D che vengono generati quando viene acquistato un terreno virtuale e le textures presenti nelle scene. Per quanto riguarda i modelli 3D questi sono modelli scaricati da *Sketchfab* [43] e sono in formato *.glb*, in quanto *three.js* ha un'ottima gestione di tale formato, ed inoltre contengono direttamente sia le animazioni che le textures associate al modello 3D. Per quanto riguarda le textures queste sono principalmente necessarie per lo sfondo HDR e per le textures relative al pavimento della mappa stessa.

Anche la cartella *src* contiene al suo interno diverse cartelle e file utili. La cartella *abis* è utile in quanto viene riempita di file in formato json da truffle, una volta distribuiti i contratti sulla blockchain. La cartella *assets* contiene il logo del nostro Metaverso, visibile all'interno della Navbar. La cartella *components* presenta le componenti che costituiscono la nostra interfaccia grafica. È buona norma infatti suddividere in file separati le componenti che costituiscono la nostra web app. Una delle principali componenti presenti è la Navbar, descritta nel file *Navbar.js*. Questa presenta a sinistra il logo ed il nome dell'applicazione ed a destra un *Button* che mostrerà l'indirizzo del wallet se la connessione è avvenuta con successo. Nel caso in cui l'utente non ha ancora effettuato l'accesso il pulsante a destra permetterà di connettersi al proprio wallet tramite Metamask²⁵ grazie al richiamo della funzione *web3Handler* definita in *app.js*. Tale funzione fa una chiamata in modo asincrono tramite il metodo *eth_requestAccounts* a Metamask, che ritorna le informazioni dell'account attivo e le salva nel vettore *accounts*.

```
Navbar.js X
src > components > Navbar.js > @Navbar
1 import logo from '../assets/logo_meta.png'
2
3 const Navbar = ({ web3Handler, account }) => {
4   return (
5     <nav className="flex-between">
6       <div
7         className="flex"
8         href="https://www.linkedin.com/in/carmelo-proetto-265549232/"
9         target="_blank"
10        rel="noopener noreferrer"
11      >
12        <img src={logo} className="App-logo" alt="logo" />
13        Metaverse Map
14      </div>
15      <div
16        {account ? (
17          <a
18            href="https://www.linkedin.com/in/carmelo-proetto-265549232/"
19            target="_blank"
20            rel="noopener noreferrer"
21            className="button"
22            {account.slice(0, 5) + '...' + account.slice(38, 42)}
23          >
24            <button onClick={web3Handler} className="button">Connect Wallet</button>
25          </a>
26        ) : (
27          <button onClick={web3Handler} className="button">Connect Wallet</button>
28        )}
29      </div>
30    </nav>
31  )
32
33 export default Navbar;
```

Figura 43 - Implementazione Navbar

```
// MetaMask Login/Connect
const web3Handler = async () => {
  if (web3) {
    const accounts = await window.ethereum.request({ method: 'eth_requestAccounts' });
    setAccount(accounts[0])
  }
}
```

Figura 44 - Funzione web3Handler

²⁵ Portafoglio di criptovalute che permette all'utente di interagire con blockchain Ethereum

La componente Ground è definita in Ground.js. Tale componente definisce il piano circolare che rappresenta la nostra mappa. Per dare un effetto reflective è stata utilizzata la componente *MeshReflectorMaterial* messa a disposizione dalla libreria *drei* [44].

La cartella contracts contiene il file Land.sol, dove sono stati definiti i nostri smart contract. L'estensione di questo file è Solidity. Come già accennato, Solidity è un linguaggio di programmazione orientato agli oggetti utile per l'implementazione di smart contracts su piattaforme blockchain, in particolare quelle basate su rete Ethereum. Per creare uno smart contract in Solidity utilizziamo la parola chiave *contract*. Vogliamo rendere questo contratto un NFT, in quanto ogni Land all'interno della nostra mappa deve avere al massimo un unico proprietario e deve essere univoca rispetto alle altre Lands. È necessario quindi rispettare lo standard 721, descritto al capitolo 3. Per fare ciò è stata utilizzata una libreria di supporto OpenZeppelin [45]. OpenZeppelin è una libreria di supporto che permette in sicurezza di creare, automatizzare e gestire applicazioni decentralizzate. Una volta importata la libreria sul nostro progetto possiamo quindi definire il nostro smart contract di tipo ERC721.

All'interno dello smart contract è necessario definire alcune variabili, in primis il costo. In questo caso ogni contratto avrà un costo pari ad 1 ether. Oltre a questo, abbiamo una variabile che tiene traccia della quantità massima di appezzamenti di terreno disponibili, ed una variabile che conta quelli acquistati. È importante per queste variabili essere definiti come *public*, in quanto ogni utente che accede alla blockchain potrà avere libero accesso a tali variabili.

```
import "@openzeppelin/contracts/token/ERC721/ERC721.sol";

contract Land is ERC721 {

    uint256 public cost = 1 ether;
    uint256 public maxSupply = 13;
    uint256 public totalSupply = 0;
}
```

Figura 45 - Definizione Smart Contract e sue variabili pubbliche

Una volta definite le variabili, devono essere definiti anche gli edifici, ognuno di questi è una *struct* contenente tutte le informazioni sul proprietario, sulla posizione nella mappa e sulle dimensioni. Definita tale struttura, è necessario definire un array che contiene tutti e 13 i nostri edifici.

Successivamente è stato definito il costruttore, questo verrà chiamato ogni qual volta il nostro smart contract viene distribuito sulla blockchain. Il costruttore prende in input tre parametri: nome, simbolo e costo. Al suo interno vengono inseriti nell'array *buildings* i 13 edifici che compongono la mappa. Per far ciò viene utilizzato il metodo *push*. Ogni *Building* avrà come parametro il nome dell'edificio, l'indirizzo del wallet dalla quale è stato acquistato e le sue coordinate spaziali e scalari. Il wallet, in fase di distribuzione, è settato a 0x0 in quanto nessuna Land ha inizialmente un proprietario.

```
struct Building {
    string name;
    address owner;
    int256 posX;
    int256 posY;
    int256 posZ;
    uint256 sizeX;
    uint256 sizeY;
    uint256 sizeZ;
}

Building[] public buildings;

constructor(
    string memory _name,
    string memory _symbol,
    uint256 _cost
) ERC721(_name, _symbol) {
    cost = _cost;

    buildings.push(
        Building("Palazzo Centrale", address(0x0), 0, 0, 0, 60, 60, 60)
    );
    buildings.push(
        Building("Parco Nord", address(0x0), 0, 0, -180, 60, 60, 60)
    );
    buildings.push(
        Building("Parco Ovest", address(0x0), -180, 0, 0, 60, 60, 60)
    );
    buildings.push(
        Building("Parco Sud", address(0x0), 0, 0, 180, 60, 60, 60)
    );
    buildings.push(
        Building("Parco Est", address(0x0), 180, 0, 0, 60, 60, 60)
    );
    buildings.push(
        Building("Stadio", address(0x0), 90, 0, 90, 70, 70, 70)
    );
    buildings.push(
        Building("Ristorante", address(0x0), -90, 0, 90, 70, 70, 70)
    );
    buildings.push(
        Building("Sala Giochi", address(0x0), -90, 0, -90, 70, 70, 70)
    );
    buildings.push(
        Building("Albergo", address(0x0), 90, 0, -90, 70, 70, 70)
    );
    buildings.push(
        Building("Palazzo Di Giustizia", address(0x0), 0, 0, 90, 40, 40, 40)
    );
    buildings.push(
        Building("Palazzo Reale", address(0x0), 90, 0, 0, 40, 40, 40)
    );
    buildings.push(
        Building("Palazzo Generale", address(0x0), 0, 0, -90, 40, 40, 40)
    );
    buildings.push(
        Building("Palazzo tecnologico", address(0x0), -90, 0, 0, 40, 40, 40)
    );
}
```

Figura 46 - Definizione edifici e costruttore

Successivamente al costruttore, vengono definite le varie funzioni. La prima è la funzione *mint*, questa è utile per coniare ed acquistare le varie *land*.

Tale funzione deve essere pubblica, in quanto tutti gli utenti possono evocarla e deve essere di tipo payable, in quanto serve per poter inviare denaro. Come unico parametro presenta un Id univoco, in quanto l'acquisto di una land non deve essere ambiguo. Prima che un terreno venga acquistato ci si deve assicurare che la fornitura non sia superiore a quella massima sopra definita, che il terreno che si sta provando ad acquistare non abbia già un proprietario e che il conto utilizzato abbia fondi a sufficienza per l'acquisto. Per fare questi controlli viene utilizzata la funzione *require* che, in caso di esito negativo, ripristinerà la transazione. Se i controlli sono andati a buon fine bisogna quindi riassegnare il proprietario con l'indirizzo del wallet che ha acquistato la land ed aumentare il numero totale di supply possedute. Infine, richiamare la funzione *_safeMint* definita nella libreria OpenZeppelin che gestirà l'acquisto seguendo lo standard ERC721.

```
function mint(uint256 _id) public payable {
    uint256 supply = totalSupply;
    require(supply <= maxSupply);
    require(buildings[_id - 1].owner == address(0x0));
    require(msg.value >= cost);

    buildings[_id - 1].owner = msg.sender;
    totalSupply = totalSupply + 1;

    _safeMint(msg.sender, _id);
}
```

Figura 47 - Funzione mint di Land.sol

Oltre a questa funzione, nello stesso file Land.sol sono presenti altre funzioni necessarie, in quanto bisogna rispettare lo standard come sopra descritto. Tali funzioni sono *transferFrom* e *safeTransferFrom* per definire il trasferimento e *getBuildings* e *getBuilding* per visualizzare i vari edifici. Da notare in queste ultime due funzioni la parola chiave *view*, utile per risparmiare i costi di gas in quanto non dobbiamo scrivere sulla blockchain ma solo leggere.

```
function transferFrom(
    address from,
    address to,
    uint256 tokenId
) public override {
    require(
        _isApprovedOrOwner(_msgSender(), tokenId),
        "ERC721: transfer caller is not owner nor approved"
    );

    buildings[tokenId - 1].owner = to;

    _transfer(from, to, tokenId);
}

function safeTransferFrom(
    address from,
    address to,
    uint256 tokenId,
    bytes memory _data
) public override {
    require(
        _isApprovedOrOwner(_msgSender(), tokenId),
        "ERC721: transfer caller is not owner nor approved"
    );

    buildings[tokenId - 1].owner = to;

    _safeTransfer(from, to, tokenId, _data);
}

function getBuildings() public view returns (Building[] memory) {
    return buildings;
}

function getBuilding(uint256 _id) public view returns (Building memory) {
    return buildings[_id - 1];
}
```

Figura 48 - Altre funzioni in *Land.sol*

Una volta definiti i contratti, è buona norma fare dei test locali in quanto, una volta distribuiti sulla blockchain, tali contratti diventano immutabili ed in caso di errori questi diverrebbero permanenti sulla rete. Tali test sono stati sviluppati nel file *land.test.js* definito nella cartella *test*. Quello che viene fatto inizialmente, prima della creazione di ogni blocco, è salvare il contratto in una variabile temporanea chiamata *land*. In altre parole, vengono eseguiti dei controlli sul contratto prima di essere distribuito. Per ogni test effettuato si avrà un risultato; è utile quindi definire anche una variabile *result*.

I vari test vengono definiti all'interno della funzione *describe*, contenente il singolo processo. Nel primo caso il processo sotto esame è quello di distribuzione. Ogni test viene eseguito attraverso i blocchi *it*. Per quanto riguarda la parte di deployment si verifica la correttezza del nome, del simbolo, del prezzo e la massima disponibilità di edifici. Per fare questi controlli viene utilizzata la libreria *chai* [\[46\]](#).

```
describe('Deployment', () => {  
  
  it('Returns the contract name', async () => {  
    | result = await land.name()  
    | result.should.equal(NAME)  
  })  
  
  it('Returns the symbol', async () => {  
    | result = await land.symbol()  
    | result.should.equal(SYMBOL)  
  })  
  
  it('Returns the cost to mint', async () => {  
    | result = await land.cost()  
    | result.toString().should.equal(COST)  
  })  
  
  it('Returns the max supply', async () => {  
    | result = await land.maxSupply()  
    | result.toString().should.equal('13')  
  })  
  
  it('Returns the number of buildings/land available', async () => {  
    | result = await land.getBuildings()  
    | result.length.should.equal(13)  
  })  
  
})
```

Figura 49 - Test sul deploy degli smart contract

Successivamente sono effettuati altri test sia sui processi di acquisto, sia su quelli di trasferimento. In questi casi vengono eseguiti dei test differenti nel caso di successo e fallimento. Nel caso di successo in entrambi i processi viene verificata la correttezza di aggiornamento dei dati del proprietario e dell'edificio. Nel caso di fallimento, per il processo di acquisto viene controllato se si sta inviando il corretto quantitativo di denaro, se l'Id dell'acquirente è valido e se il terreno non ha già un proprietario; mentre per il processo di trasferimento si controlla se tale trasferimento è effettuato senza avere il possesso della land o senza l'approvazione del proprietario della land. Una volta scritti i test, è possibile eseguire il processo di testing. Per fare ciò bisogna avviare la nostra blockchain. Come citato all'inizio di questo capitolo si utilizza una blockchain locale chiamata Ganache, per avviarla viene utilizzato il comando su terminale *ganache-cli*. Questo avvierà una blockchain emulata e metterà a nostra disposizione dieci account contenenti 1000 eth l'uno. Ogni account presenta una chiave pubblica ed una chiave privata. La chiave privata sarà successivamente utile nell'import su MetaMask. È possibile adesso distribuire i nostri smart contract sulla blockchain tramite Truffle eseguendo il comando *truffle migrate --reset*. Una volta che sono stati distribuiti è possibile valutare la correttezza utilizzando i test scritti grazie al comando *truffle test*.

Se tutto è andato a buon fine si otterrà una stringa che dice che tutti i 14 test effettuati sono stati correttamente superati. A questo punto sono stati implementati correttamente i nostri Smart Contract, sono stati distribuiti su una blockchain e sono stati verificati attraverso i vari test. Il risultato sono degli smart contracts sicuri ed affidabili che possono essere pubblicati senza problemi su una blockchain reale.

```
Contract: Land
Deployment
  ✓ Returns the contract name
  ✓ Returns the symbol
  ✓ Returns the cost to mint
  ✓ Returns the max supply
  ✓ Returns the number of buildings/land available (193ms)
Minting
  Success
    ✓ Updates the owner address (49ms)
    ✓ Updates building details
  Failure
    ✓ Prevents mint with 0 value (468ms)
    ✓ Prevents mint with invalid ID (52ms)
    ✓ Prevents minting if already owned (114ms)
Transfers
  success
    ✓ Updates the owner address
    ✓ Updates building details
  failure
    ✓ Prevents transfers without ownership
    ✓ Prevents transfers without approval (74ms)

14 passing (11s)
```

Figura 50 - Test Smart Contracts

Lo step successivo è quello di collegare i nostri smart contracts all'applicazione web. La quasi totalità di interazione fra smart contracts e applicazione web è implementata in *App.js*, posizionato nella cartella *src*. Per far interagire le varie componenti ed il codice React con gli smart contracts viene utilizzata la libreria Web3. Una volta aperta l'applicazione è possibile tramite apposito pulsante accedere al proprio account tramite MetaMask.

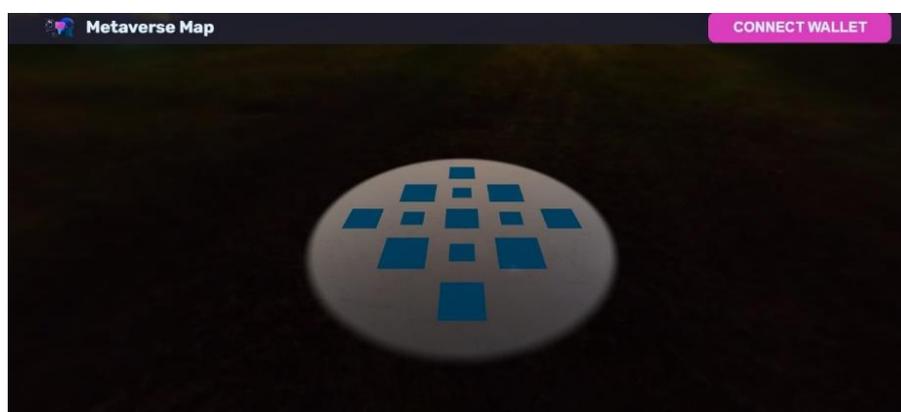


Figura 51 - Schermata iniziale applicazione web realizzata

Cliccato il pulsante verrà eseguito un tentativo di caricamento dell'account attraverso la funzione *web3Handler*. Se MetaMask darà una risposta positiva, ovvero se viene trovato un account collegato, questo sarà inserito in una variabile locale chiamata *account* attraverso *setAccount*. Ogni qualvolta la variabile *account* viene modificata viene evocato lo *useEffect* ad esso associato che richiama la funzione *loadBlockchain()*. Tale funzione controlla la presenza di MetaMask all'interno del pc, successivamente verifica che vi sia un account collegato ed in caso affermativo setta l'account recuperato da MetaMask. A questo punto vengono recuperate tutte le informazioni dei contratti presenti sulla blockchain utilizzata, sempre grazie a Web3 e MetaMask e vengono settate le relative variabili locali definite nel file javascript. L'applicazione web è stata correttamente collegata con gli Smart Contract distribuiti.

```
function App() {
  const [web3, setWeb3] = useState(null)
  const [account, setAccount] = useState(null)

  // Contract & Contract States
  const [landContract, setLandContract] = useState(null)

  const [cost, setCost] = useState(0)
  const [buildings, setBuildings] = useState(null)
  const [landId, setLandId] = useState(null)
  const [landName, setLandName] = useState(null)
  const [landOwner, setLandOwner] = useState(null)
  const [hasOwner, setHasOwner] = useState(false)

  const loadBlockchainData = async () => {
    if (typeof window.ethereum !== 'undefined') {
      const web3 = new Web3(window.ethereum)
      setWeb3(web3)

      const accounts = await web3.eth.getAccounts()

      if (accounts.length > 0) {
        setAccount(accounts[0])
      }

      const networkId = await web3.eth.net.getId()

      const land = new web3.eth.Contract(Land.abi, Land.networks[networkId].address)
      setLandContract(land)

      const cost = await land.methods.cost().call()
      setCost(web3.utils.fromWei(cost.toString(), 'ether'))

      const buildings = await land.methods.getBuildings().call()
      setBuildings(buildings)

      window.ethereum.on('accountsChanged', function (accounts) {
        setAccount(accounts[0])
      })

      window.ethereum.on('chainChanged', (chainId) => {
        window.location.reload();
      })
    }
  }

  const web3Handler = async () => {
    if (web3) {
      const accounts = await window.ethereum.request({ method: 'eth_requestAccounts' });
      setAccount(accounts[0])
    }
  }

  useEffect(() => {
    loadBlockchainData()
  }, [account])
}
```

Figura 52 - Implementazione funzioni collegamento a MetaMask

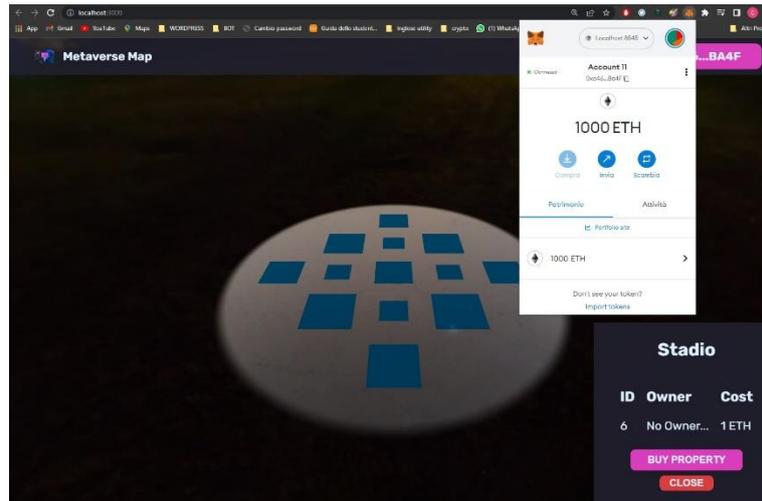


Figura 53 - Collegamento Account Metamask

Un'altra funzione importante definita in `App.js` è `buyHandler`. Questa è utile per permettere il processo di acquisto di uno degli appezzamenti di terreni disponibili. Cliccando su un appezzamento libero comparirà, in basso a destra dello schermo, una modale che permette l'acquisto di quella proprietà. La funzione `buyHandler` è collegata a quel bottone. Analizzandone l'aspetto implementativo, tale funzione chiama in modo asincrono la funzione `mint` definita sullo smart contract e se l'acquisto va a buon fine setta le variabili associate a quell'edificio, ovvero il nome della land ed il wallet proprietario.

```
const buyHandler = async (_id) => {
  try {
    await landContract.methods.mint(_id).send({ from: account, value: '1000000000000000000' })

    const buildings = await landContract.methods.getBuildings().call()
    setBuildings(buildings)

    setLandName(buildings[_id - 1].name)
    setLandOwner(buildings[_id - 1].owner)
    setHasOwner(true)
  } catch (error) {
    window.alert('Error occurred when buying')
  }
}
```

Figura 54 - Funzione per acquisto proprietà

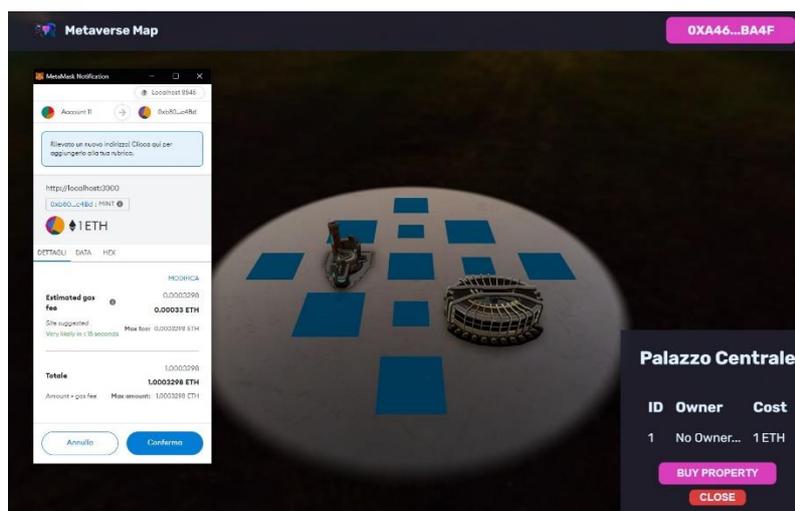


Figura 55 - Processo di Acquisto sul Metaverso

La visualizzazione delle terre, sia non ancora acquistate da nessuno, sia in possesso di qualcuno, avviene tramite una funzione di map sul vettore *buildings* che contiene le informazioni di tutte le terre disponibili nella mappa. Se la terra non appartiene a nessuno avrà il valore di *owner* settato a $0x0..0$, in questo caso viene generato un piano con la componente *planeGemoetry*, in posizione e scala recuperata dal vettore stesso.

Se la Land invece appartiene a qualcuno, viene controllato il nome dell'edificio ed a seconda di questo, viene caricato nella giusta posizione il corretto modello, sempre basandosi sui valori posizionali dell'item contenuto in *buildings*.

Dunque, ogni qual volta un acquisto va a buon fine, come già descritto, viene renderizzato un modello 3D associato alla Land. Per far ciò i file *.glb* vengono caricati attraverso la funzione *useLoader* e conservati in apposite variabili che verranno visualizzate se e quando necessario.

```

(buildings && buildings.map((building, index) => {

const clickHandler = () => {
  setLandName(building.name)
  setLandId(index + 1)

  if (building.owner === '0x00000000000000000000000000000000') {
    setLandOwner("No Owner")
    setHasOwner(false)
  } else {
    setLandOwner(building.owner)
    setHasOwner(true)
  }
}

if (building.owner === '0x00000000000000000000000000000000') {

return (
  <mesh
    onClick={clickHandler}
    rotation-x={Math.PI * -0.5}
    scale={[building.sizeX, building.sizeY, building.sizeZ]}
    key={index}
    position={[building.posX, 0.7, building.posZ]}
    landId={index + 1}
    landInfo={building}
    setLandName={setLandName}
    setLandOwner={setLandOwner}
    setHasOwner={setHasOwner}
    setLandId={setLandId}
  >
    <planeGeometry />
    <MeshReflectorMaterial color={"#25AEEC"} />
  </mesh>
)
} else {
  if (building.name === "Palazzo Centrale"){
    return (
      <object3D>
        <mesh
          onClick={clickHandler}

          scale={[0.025, 0.025, 0.025]}
          key={index}
          position={[building.posX, 0, building.posZ]}
          landId={index + 1}
          landInfo={building}
          setLandName={setLandName}
          setLandOwner={setLandOwner}
          setHasOwner={setHasOwner}
          setLandId={setLandId}
        >
          <primitive object={palazzoCentrale.scene.clone()} />
        </mesh>
      </object3D>
    )
  }
  else if (building.name === "Parco Nord"){

```

Figura 56 - Visualizzazione Land ed Edifici

Un'ultima funzione importante, visibile in **Figura 56**, è *clickHandler*. Questa serve per far comparire la modale che permette di visualizzare le informazioni sulla land o sull'edificio. Inoltre, se la land è disponibile tale modale permette di acquistarla, se è già in possesso di qualcuno consente di collegarsi all'esperienza associata.

5.3 Var Group sul Metaverso

Come ampiamente discusso in questo capitolo e nei capitoli precedenti, il vero cuore del Metaverso si concentra sulle esperienze utente realizzabili all'interno dei vari spazi virtuali. È per questo che ho voluto provare con mano a costruire un'esperienza, realizzando un ambiente utile all'azienda che mi ha aiutato a svolgere questa tesi. Tale progetto è stato visionato tramite visori Meta Quest 2 da diversi membri della realtà VarGroup, nella convention che si è tenuta il 20 ed il 21 ottobre 2022 al Palariccione. Ho integrato questa

esperienza con il progetto di mappa precedentemente descritto attraverso il link collegato al Palazzo centrale della mappa. È comunque possibile accedere direttamente all'esperienza dal link che lascio in sitografia [\[47\]](#), sia da web, che da mobile, ma soprattutto da visore per godere di un'esperienza ottimale.

5.3.1 Overview

L'idea del progetto è stata quella di realizzare una rivisitazione del mondo VarGroup, in un modo più immersivo rispetto al classico sito web [\[48\]](#). Ho quindi cercato di ricreare in modo fedele ma innovativo il sito dell'azienda in un'esperienza completamente virtuale per far scoprire ai visitatori o ai clienti la realtà aziendale. Il mondo realizzato si presenta suddiviso in otto differenti zolle.



Figura 57 - Metaverso VarGroup

Analizzando più nel dettaglio le varie zolle, queste sono state organizzate per mostrare le varie realtà aziendali, le varie business unit e per contenere anche zone più generiche utili per incontri o presentazioni.

La prima zolla nella quale ci si trova una volta entrati nell'esperienza virtuale è un ambiente introduttivo all'azienda dove si viene a conoscenza dei dati principali del gruppo e si è accolti da un video introduttivo che mostra tutti i punti chiave dell'azienda. Guardandosi intorno si notano tutte le altre zolle, ognuna con un proprio mood differente. La zolla VarDigitalArt presenta dei video introduttivi al progetto VarDigital dell'azienda, ovvero un progetto nel quale diversi artisti, tramite tecnologie basate sull'intelligenza artificiale,

hanno e stanno tutt'ora realizzando diverse opere d'arti. Oltre a questo, tale zolla presenta una riproduzione della struttura appositamente realizzata per l'evento avvenuto al Palariccione, che mostrava a chi interessato tali opere. Sono inoltre presenti diversi quadri che rappresentano le principali attività svolte, collegate con link diretti agli articoli sul sito dell'azienda. Un'altra zolla mostra tutti i principali servizi offerti dalla business unit di cyber security. Questa è molto particolare, in quanto per rinforzare l'idea, è stata progettata all'interno di una cassaforte. Oltre a presentare i quadri dei principali lavori svolti, con relativi link agli articoli, presenta differenti modelli 3D, come telecamere e guardie, per rendere la scena più dinamica.

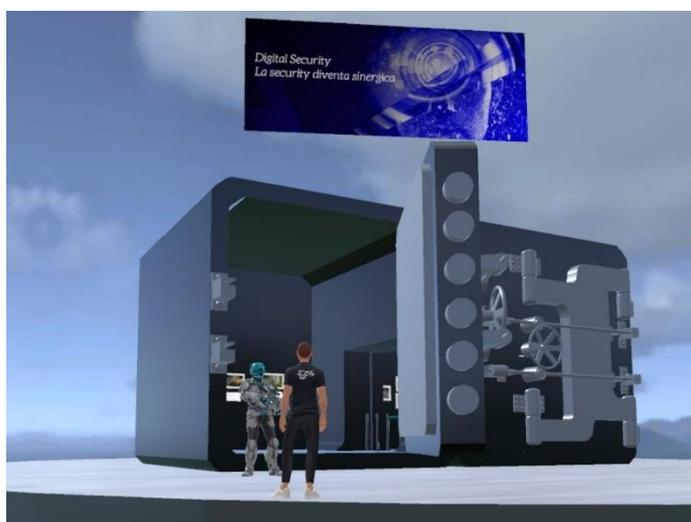


Figura 58 - Zolla Cyber Security

La successiva zolla presenta invece una duplice funzionalità: da un lato mostra i vari lavori svolti dal core di VarGroup che si occupa di Business Technology e Smart Services, dall'altro è un'area di meeting, dove sono presenti degli accoglienti divanetti, nella quale poter accogliere potenziali clienti per discutere di progetti o quant'altro. Un'altra zolla presente è quella relativa al reparto Cloud, che per rendere l'idea è sospesa tra le nuvole; questa è collegata tramite un ponte in vetro alla zolla principale di Spawn. Abbiamo poi una zolla relativa alla business unit di Digital Industries, dove troviamo un palazzo che accoglie dei modelli 3D come bracci robotici o ventilatori futuristici, oltre ai soliti quadri rappresentanti le informazioni sulla parte aziendale e i migliori progetti realizzati.

Inoltre, è presente una zolla relativa all'intelligenza artificiale, dove sono presenti molti modelli di prototipi futuristici quali robot dalle sembianze umane o di animali.



Figura 59 - Zolla Intelligenza Artificiale

Infine, abbiamo un auditorium molto futuristico nel quale è possibile fare delle presentazioni all'interno del Metaverso accogliendo fino a 50 utenti contemporaneamente.

L'ambiente realizzato, dunque, oltre a risultare introduttivo alle varie facce dell'azienda VarGroup può essere utile per introdurre i potenziali clienti ad una prima esperienza all'interno di un Metaverso e alle infinite potenzialità che questo offre.

5.3.2 Strumenti utilizzati

Per la realizzazione di questo progetto mi sono affidato ad un servizio online chiamato Spatial [\[38\]](#). Spatial è un Metaverso basato su piattaforma web che, come si legge sul loro sito, è dedicato ad aiutare creatori e brand a costruire i propri spazi nel metaverso, per condividere delle esperienze con il proprio pubblico.

L'idea è quella di consentire ai creatori la possibilità di costruire i propri spazi, partendo da ambienti già definiti o importandone di nuovi completamente da zero. Questi ambienti successivamente potranno essere conosciuti come NFT e

venduti/affittati ad altri utenti che desiderano ospitare delle esperienze virtuali.

Spatial quindi è un Metaverso dedicato principalmente ad eventi e mostre che permette di creare uno spazio accessibile contemporaneamente da 50 differenti utenti.

Nello specifico per la realizzazione del progetto sopra descritto, oltre a registrarmi e creare il mio avatar sulla piattaforma, ho deciso di importare un'ambientazione completamente da zero. Ho realizzato l'ambientazione utilizzando il software Blender (**figura 60**).

Blender è un software di modellazione ed animazione di oggetti 3D, che mi ha permesso di realizzare i vari modelli presenti all'interno del mondo virtuale.

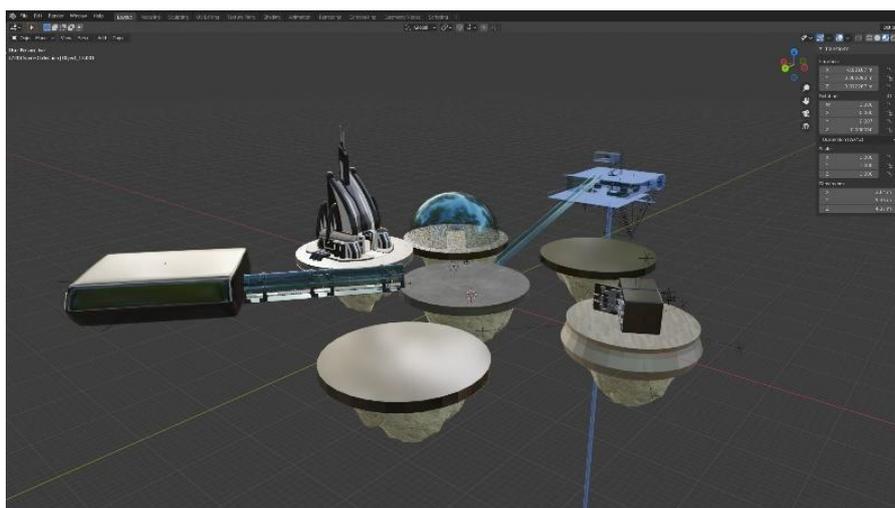


Figura 60 - Creazione ambientazione su Blender

Alcuni modelli sono stati realizzati da zero, come le varie piattaforme e le varie zolle, mentre alcuni dei modelli presenti sono stati scaricati da Internet ed importati sul software. Per fare un esempio pratico di un modello realizzato da 0, possiamo prendere in causa la struttura della zolla, la quale è stata realizzata partendo da un piano modellato tramite l'utilizzo del *proportional editing* e della modalità di *sculpt*. Una volta creato il modello su Blender questo è stato esportato con estensione glb, per essere facilmente caricato su Spatial. Un file di questo tipo è facilmente gestibile sul Metaverso Spatial

in quanto contiene in sé sia il modello 3D, che tutte le informazioni necessarie per i materiali e per le animazioni. Per la gestione degli oggetti non inclusi nell'ambientazione ho utilizzato altri modelli, alcuni realizzati sempre da 0 su Blender, altri scaricati da Sketchfab, ma entrambi importati tramite il tool di costruzione messo a disposizione da Spatial.

Anche per quanto riguarda la parte di foto e video presenti nell'esperienza, l'import è abbastanza semplice ed intuitivo, in quanto il sistema di costruzione dell'ambiente è facilmente accessibile ed utilizzabile anche grazie alle funzioni di drag and drop. Anche per quanto riguarda il posizionamento, lo scalamento e la rotazione degli oggetti, l'interfaccia mette a disposizione dei cursori che ne permettono una facile gestione.



Figura 61 - Gestione modelli su Spatial

6 Risultati e sviluppi futuri

Grazie all'analisi svolta all'interno di questo percorso di tesi ho potuto comprendere al meglio tutta l'innovazione e tutta la tecnologia che si cela dietro un applicativo come il Metaverso che all'apparenza potrebbe risultare banale e non innovativo. Ho capito da cosa è nata l'idea di Metaverso, quale problema va a risolvere, quali possono essere i suoi usi e le sue potenzialità, cosa attualmente offre ai propri utenti e cosa in futuro potrebbe offrire per rivoluzionare il mondo come lo conosciamo oggi.

Grazie a questa analisi ho potuto approfondire i concetti di Realtà Virtuale e Realtà Aumentata ed ho capito come, grazie alle loro caratteristiche di immersività e presenza, abbiano un'importanza chiave nella realizzazione di un progetto così ambizioso e futuristico come il Metaverso.

Ho potuto capire perché ed in che modo la Blockchain sia fondamentale per mantenere in piedi tutto il sistema, senza rinunciare ad affidabilità e sicurezza e con la garanzia di ottenere dei benefit utilizzando una piattaforma decentralizzata.

Ho potuto analizzare ed approfondire nel dettaglio i principali competitors presenti attualmente sul mercato, analizzandone vantaggi e svantaggi, peculiarità e potenzialità per gli utenti e soprattutto per le aziende.

Per quanto riguarda la parte progettuale, la realizzazione di una Mappa del Metaverso mi ha permesso di toccare con mano le effettive tecnologie che stanno alla base della blockchain e del Metaverso.

Ho imparato le basi di Solidity, ho familiarizzato con l'utilizzo di una Blockchain e di wallet digitali come MetaMask, ho imparato a collegare tutta la tecnologia decentralizzata con un'applicazione web ed ho consolidato le mie conoscenze di React imparando ad usare anche nuove librerie di supporto, come quelle per la gestione di modelli 3D sul web.

Valuto quindi questo esperimento fortemente costruttivo e stimolante, in quanto ho compreso in prima persona quanto sia importante la cura e l'attenzione nel dettaglio durante la progettazione di uno Smart Contract e

quanto potenzialmente una tecnologia decentralizzata, che per esempio in questo caso garantisce affidabilità ed univocità negli acquisti, permetta un'infinità di possibili utilizzi o impieghi. Seppur ho portato avanti in modo parziale questo progetto, caricando dei modelli non NFT ed utilizzando una piattaforma già esistente, ovvero Spatial, per la realizzazione delle esperienze associate alle varie Land, reputo che l'insegnamento appreso da questa parte progettuale possa essere applicabile anche nella creazione e nello sviluppo delle varie esperienze utente associate alle terre virtuali.

Per quanto riguarda la parte di esperienza realizzata tramite l'utilizzo di Spatial questa è stata molto utile per comprendere a pieno, mettendosi nei panni di un utente alle prime armi con un metaverso, le potenzialità di una piattaforma del genere, in quanto è molto semplice ed intuitivo accedervi, interagire con l'ambiente e incontrare altri utenti nello spazio virtuale. È quindi molto semplice organizzare un evento, creare una mostra o partecipare ad una riunione in una maniera più immersiva rispetto ai classici metodi utilizzati attualmente, quali i vari software di video call.

Se la semplicità di accesso ed interazione con il mondo virtuale è molto appagante, quello in cui Spatial pecca attualmente è il fattore sicurezza, in quanto non ci stiamo basando su un sistema decentralizzato. In futuro, quindi, modificherò l'esperienza realizzata su Spatial provando a ri-crearla da zero, grazie anche ai concetti appresi durante la realizzazione della mappa, senza nessuna piattaforma di supporto e rendendo ogni oggetto presente all'interno dell'esperienza un NFT da me realizzato, per portare ad uno step successivo, basato sui concetti di proprietà di beni, sicurezza dei contenuti e decentralizzazione.

7 Figure e altro

Figura 1 Spada di Damocle - Ivan Sutherland, 1968.....	11
Figura 2 - Pokemon GO - Ninantic, 2016.....	13
Figura 3 - Tipi di sistemi AR in commercio	14
Figura 4 - Differenza di funzionamento fra le tecnologie Video see-through ed Optical see-through.....	17
Figura 5 - Funzionamento sistema AR.....	18
Figura 6 - Sensorama - Morton Heilig, 1962	23
Figura 7 - Miglioramenti dei Visori negli anni	29
Figura 8 - Sistema Centralizzato	31
Figura 9 - Sistema Decentralizzato	31
Figura 10 - Timestamp Server	35
Figura 11 - Struttura di una Blockchain.....	36
Figura 12 - Centralized ledger VS Distributed ledger	37
Figura 13 - Processo di validazione di una transazione tramite PoW	38
Figura 14 - The Sandbox.....	48
Figura 15 - Benefit dell'utilizzo della blockchain	49
Figura 16 - Ecosistema The Sandbox.....	49
Figura 17 - SAND casi d'uso	51
Figura 18 - Il flow di SAND.....	52
Figura 19 - Lands Map, the Sandbox	52
Figura 20 – Decentraland	54
Figura 21 - Mappa primordiale Decentraland.....	54
Figura 22 - Protocollo Decentraland	57
Figura 23 - Mana token, token di Decentraland	60
Figura 24 - OVR Metaverse	61
Figura 25 - Processo di vendita di OVRExperiences	62
Figura 26 - Esempio pratico unione SLAM e GPS.....	63
Figura 27 - Infrastruttura blockchain OVR	65
Figura 28 - Infrastruttura blockchain - OVR Owner	66
Figura 29 - Infrastruttura Blockchain - OVRUser (privata)	66
Figura 30 - Infrastruttura blockchain - OVRUser (pubblica).....	66
Figura 31 - OVRAssistant	67
Figura 32 - Funzionamento OVRToken.....	68
Figura 33 - OVR Lands	69
Figura 34 - Architettura Metaverso.....	70
Figura 35 - Landing Page The Sandbox.....	71
Figura 36 - VoxEdit - The Sandbox.....	72
Figura 37 - Metaverse Client - Decentraland	72
Figura 38 - Mappa OVR	73
Figura 39 - Overview Architettura Mappa	76
Figura 40 - Mappa con tutte le terre acquistate	77
Figura 41 - Struttura del Progetto	78
Figura 42 - File di migrazione dei nostri smart contracts su blockchain	79

Figura 43 - Implementazione Navbar.....	80
Figura 44 - Funzione web3Handler	80
Figura 45 - Definizione Smart Contract e sue variabili pubbliche	81
Figura 46 - Definizione edifici e costruttore.....	82
Figura 47 - Funzione mint di Land.sol	83
Figura 48 - Altre funzioni in Land.sol.....	84
Figura 49 - Test sul deploy degli smart contract	85
Figura 50 - Test Smart Contracts.....	86
Figura 51 - Schermata iniziale applicazione web realizzata	86
Figura 52 - Implementazione funzioni collegamento a MetaMask	87
Figura 53 - Collegamento Account Metamask.....	88
Figura 54 - Funzione per acquisto proprietà	88
Figura 55 - Processo di Acquisto sul Metaverso.....	89
Figura 56 - Visualizzazione Land ed Edifici	90
Figura 57 - Metaverso VarGroup.....	91
Figura 58 - Zolla Cyber Security.....	92
Figura 59 - Zolla Intelligenza Artificiale	93
Figura 60 - Creazione ambientazione su Blender.....	94
Figura 61 - Gestione modelli su Spatial.....	95

8 Bibliografia e Sitografia

- [1] Paul Milgram, Haruo Takemura, Akira Utsumi, and Fumio Kishino. Augmented reality: A class of displays on the reality-virtuality continuum. *Telemanipulator and Telepresence Technologies*, 2351, 01 1994. doi: 10.1117/12.197321.
- [2] Ronald T Azuma. «A survey of augmented reality». In: *Presence: Teleoperators & Virtual Environments* 6.4 (1997), pp. 355–385.
- [3] Ivan E. Sutherland. A head-mounted three-dimensional display. In *AFIPS Conference Proceedings (1968)* 33, I, pages 757–764, 1968.
- [4] Thomas Caudell and David Mizell. Augmented reality: An application of heads-up display technology to manual manufacturing processes. volume 2, pages 659 – 669 vol.2, 02 1992. ISBN 0-8186-2420-5. doi: 10.1109/HICSS.1992.183317
- [5] Jun Rekimoto. A magnifying glass approach to augmented reality systems. *Presence*, 6:399–412, 01 1997.
- [6] Steven Feiner, Blair Macintyre, Tobias Höllerer, and Anthony Webster. A touring machine: Prototyping 3d augmented reality systems for exploring the urban environment. *Personal and Ubiquitous Computing*, 1:208–217, 01 1997.
- [7] <http://www.hitl.washington.edu/artoolkit> - Hirokazu Kato. Artoolkit, 1999.
- [8] <https://uxdesign.cc/augmented-reality-device-types-a7668b15bf7a> - Understanding the different types of AR devices.
- [9] <https://www.pwc.com/seeingisbelieving> - How VR and AR will transform business and the economy.
- [10] <https://www.treccani.it/enciclopedia/realta-virtuale>
- [11] GREENBAUM P., *The lawnmower man*. Film e video, 1992.
- [12] BURDEA G., COIFET P., *Virtual reality technology (2nd ed.)*, IEEE Press, 2003.
- [13] STEURER J., *Communication in the Age of Virtual Reality*, LEA Publishers, 1995.
- [14] HEIM M., *The Metaphysics of Virtual Reality*, OUP USA, 1994.
- [15] SHIELDS R., *The Virtual*, Routledge, 2003.
- [16] "U.S. Patent 2955156," (Oct. 4, 1960.).
- [17] Ivan E. Sutherland. The ultimate display. In *Proceedings of the IFIP Congress*, pages 506–508, 1965
- [18] <https://bitcoin.org/bitcoin.pdf>
- [19] https://blog.osservatori.net/it_it/bitcoin-storia-blockchain
- [20] <https://www.fincen.gov/news/testimony/statement-jennifer-shasky-calvery-director-financial-crimes-enforcement-network>
- [21] [https://www.techtarget.com/searchcio/definition/distributed-ledger#:~:text=Distributed%20ledger%20technology%20\(DLT\)%20is,data%20store%20or%20administration%20functionality](https://www.techtarget.com/searchcio/definition/distributed-ledger#:~:text=Distributed%20ledger%20technology%20(DLT)%20is,data%20store%20or%20administration%20functionality).

- [22] A systematic literature review of blockchain-based applications: Current status, classification and open issues
- [23] Proof-of-Work Vs Proof-of-Stake: A Comparative Analysis and an Approach to Blockchain Consensus Mechanism - Husneara Sheikh, Rahima Meer Azmathullah, Faiza Rizwan
- [24] Hashcash - A Denial of Service Counter-Measure - Adam Back, 2002
- [25] Smart Contracts Glossary by Nick Szabo, 1995
- [26] Smart Contracts: Building Blocks for Digital Markets by Nick Szabo, 1996
- [27] <https://ethereum.org/it/developers/docs/smart-contracts/#:~:text=Gli%20smart%20contract%20sono%20un,rete%20ed%20eseguiti%20come%20programmato.>
- [28] <https://ethereum.org/it/>
- [29] <https://ethereum.org/en/developers/docs/evm/>
- [30] <https://www.cryptokitties.co/>
- [31] The Sandbox - White Paper
- [32] Decentraland - White Paper
- [33] OVR - White Paper
- [34] <https://edenlans.com/>
- [35] <https://it.reactjs.org/>
- [36] <https://threejs.org/>
- [37] <https://docs.pmnd.rs/react-three-fiber/getting-started/introduction>
- [38] <https://www.spatial.io/>
- [39] <https://docs.soliditylang.org/en/v0.8.17/>
- [40] <https://trufflesuite.com/ganache/>
- [41] <https://trufflesuite.com/>
- [42] <https://nodejs.org/en/>
- [43] <https://sketchfab.com/feed>
- [44] <https://github.com/pmndrs/drei>
- [45] <https://www.openzeppelin.com/>
- [46] <https://www.npmjs.com/package/chai>
- [47] <https://www.spatial.io/s/VarGroup-633eb35756b230000156eca3?share=6728909407832214510>
- [48] <https://www.vargroup.it/>