

ALMA MATER STUDIORUM · UNIVERSITÀ DI BOLOGNA

---

SCUOLA DI SCIENZE

CORSO DI LAUREA IN  
INFORMATICA PER IL MANAGEMENT

**Pop Tools:  
Sviluppo e progettazione di  
una Web Directory per  
Freelance in ambito ICT**

RELATORE:  
**Prof. Angelo Di Iorio**

PRESENTATA DA:  
**Alessandro Pasi**

SESSIONE II  
ANNO ACCADEMICO 2022/2023



# Indice

<b>1</b>	<b>Introduzione</b>	<b>4</b>
<b>2</b>	<b>Concetti base</b>	<b>5</b>
2.1	Concetto di <i>web directory</i>	5
2.1.1	La nascita	6
2.1.2	Perché erano importanti	6
2.1.3	Motori di ricerca vs <i>web directory</i>	6
2.2	Evoluzione delle <i>web directory</i>	7
2.2.1	Yahoo! e l'ascesa delle <i>web directory</i>	7
2.2.2	Nuovi ruoli e obiettivi delle <i>web directory</i>	7
2.2.3	La nascita delle <i>community</i>	7
2.2.4	Adattamento e successo delle <i>web directory</i>	8
2.3	Importanza delle risorse digitali per professionisti e freelance	9
2.3.1	Impatto delle risorse digitali sui <i>freelance</i>	9
2.3.2	L'importanza dei <i>MOOCs</i>	9
2.3.3	Strumenti digitali innovativi	9
2.4	<i>Web directory</i> : stato dell'arte	11
2.4.1	Product Hunt	11
2.4.2	Futurepedia	12
2.4.3	Altre <i>directory</i> di rilievo	12
<b>3</b>	<b>Descrizione del progetto</b>	<b>13</b>
3.1	La piattaforma <i>Pop Tools</i>	13
3.1.1	Semplificare la ricerca	13
3.1.2	Qualità dei contenuti	13
3.1.3	L'importanza della <i>community</i>	14
3.2	Vantaggi del progetto	15
3.2.1	Accesso facilitato alle risorse digitali	15
3.2.2	Riduzione del tempo e dei costi	15
3.2.3	Collaborazione e condivisione	15
3.2.4	Supporto alla crescita professionale	15
3.2.5	Sicurezza e protezione dei dati	16
3.2.6	Semplicità di utilizzo	16
3.2.7	Considerazioni	16
3.3	Funzionalità chiave	17
3.3.1	Struttura generale	17
3.3.2	Aggiunta e gestione dei preferiti	17
3.3.3	Filtri di ricerca	18
3.3.4	Partecipazione attiva e condivisione	19
3.3.5	<i>Dashboard</i> per la gestione e l'amministrazione	20
3.4	Scenari d'uso	21
3.4.1	Attori	21
3.4.2	Casi d'uso	21
3.4.3	Casi d'uso rilevanti	23
<b>4</b>	<b>Progettazione dell'interfaccia utente</b>	<b>25</b>
4.1	<i>Layout e design</i> dell'interfaccia	25
4.1.1	Disposizione degli elementi	25
4.1.2	Gerarchia dei componenti	25
4.1.3	Navigazione	26
4.1.4	Colori e immagini	26
4.2	<i>User Experience</i> e usabilità	27
4.2.1	<i>Design System</i>	27
4.2.2	Accessibilità	27
4.2.3	<i>Responsive design</i>	28
4.3	Schermate principali	29
4.3.1	<i>Homepage</i>	29

4.3.2	<i>Search</i>	30
4.3.3	<i>Login &amp; Registration</i>	31
4.3.4	<i>Profile &amp; Site Submit</i>	32
4.3.5	<i>Tool</i>	33
4.3.6	<i>Dashboard</i>	34
4.4	<i>Testing</i>	35
4.4.1	Obiettivi e importanza	35
4.4.2	<i>Testing SUS</i>	35
4.4.3	Calcolare punteggio SUS	35
4.4.4	Risultati	36
<b>5</b>	<b>Tecnologie e strumenti utilizzati</b>	<b>37</b>
5.1	<i>Framework</i> di sviluppo	37
5.1.1	<i>SvelteKit</i>	37
5.1.2	<i>Tailwind CSS</i>	37
5.2	Autenticazione	39
5.2.1	<i>OAuth 2.0</i>	39
5.2.2	<i>OpenID Connect</i>	40
5.2.3	<i>Auth.js</i>	41
5.3	<i>Database</i>	42
5.3.1	<i>PostgreSQL</i>	42
5.3.2	<i>Prisma</i>	44
5.3.3	<i>Firebase</i>	46
<b>6</b>	<b>Scelte progettuali e implementative</b>	<b>47</b>
6.1	Struttura <i>BE</i>	49
6.1.1	<i>REST API</i>	49
6.1.2	<i>Query</i> e modelli	49
6.2	Struttura <i>FE</i>	51
6.2.1	Componenti <i>AppMyComponent</i>	51
6.2.2	Componenti <i>logici</i>	51
6.3	<i>Data fetching</i>	52
6.3.1	Funzioni di <i>Load</i>	52
6.3.2	<i>Client-Side data fetching</i>	53
6.3.3	<i>Form actions</i>	53
6.4	Ruoli e permessi	54
6.4.1	Implementazione nel <i>BE</i>	54
6.4.2	Implementazione nel <i>FE</i>	55
<b>7</b>	<b>Conclusioni</b>	<b>56</b>

## Elenco delle tabelle

1	Attori nel Sistema . . . . .	21
2	Caso d'Uso 1: Ricerca e visualizzazione dei siti web . . . . .	23
3	Caso d'Uso 2: Invio proposta di un nuovo sito web . . . . .	23
4	Caso d'Uso 3: Gestione dei preferiti . . . . .	24
5	Caso d'Uso 4: Inserimento di un nuovo sito web . . . . .	24
6	Valutazione dell'Usabilità tramite Questionario SUS . . . . .	36

## Elenco delle figure

1	Struttura di una web directory . . . . .	5
2	Homepage di PopTools . . . . .	13
3	Schermata Homepage e Tool che mostrano la struttura generale di Pop Tools . . . . .	17
4	Schermata Homepage e Profile che mostrano la funzionalità di aggiunta ai Preferiti . . . . .	18
5	Filtri di ricerca . . . . .	18
6	Schermata Tool e Tool Submit che mostrano le funzionalità di pubblicazione recensioni e invio proposte . . . . .	19
7	Schermata Dashboard che mostra le funzionalità di gestione siti web e proposte . . . . .	20
8	Diagramma dei casi d'uso della piattaforma Pop Tools . . . . .	22
9	Componenti <i>card</i> contenenti le informazioni dei siti web . . . . .	25
10	Schermata Homepage visualizzata su <i>Desktop e Mobile</i> . . . . .	29
11	Schermata Search visualizzata su <i>Desktop e Mobile</i> . . . . .	30
12	Schermate Login e Registrazione visualizzate su <i>Desktop e Mobile</i> . . . . .	31
13	Schermate Profile e Site Submit visualizzate su <i>Desktop e Mobile</i> . . . . .	32
14	Schermata Tool visualizzata su <i>Desktop e Mobile</i> . . . . .	33
15	Schermata Dashboard visualizzata su <i>Desktop</i> . . . . .	34
16	Processo di funzionamento di OAuth 2.0 . . . . .	39
17	Processo di funzionamento di OpenID Connect . . . . .	40
18	Modello ER del database . . . . .	43

# 1 Introduzione

Nell'attuale panorama digitale, si assiste ad una crescita senza precedenti delle informazioni e delle risorse disponibili in rete. Questa proliferazione, sebbene rappresenti una fonte inestimabile di opportunità, pone anche nuove sfide. La grande quantità di contenuti e strumenti può, infatti, risultare opprimente e disorientante per gli utenti, in particolare per *professionisti, freelance e organizzazioni* che cercano di navigare in questa mole di dati alla ricerca di risorse pertinenti e di alta qualità che possano migliorare o agevolare il loro quotidiano. In tale contesto, emerge la necessità di sistemi e piattaforme che possano facilitarne l'accesso, la valutazione e la condivisione.

Nasce qui l'idea di "*Pop Tools*", una web directory concepita per rispondere in modo efficace a queste sfide. Essa non è una semplice raccolta di link o una banca dati passiva, ma un ecosistema digitale progettato per fornire una soluzione completa e semplice da utilizzare ovunque e su qualsiasi dispositivo. La sua importanza risiede nel fatto che va oltre l'organizzazione efficiente delle risorse ma incoraggia attivamente la partecipazione degli utenti, promuovendo la condivisione di conoscenze ed esperienze.

Nel contesto di *Pop Tools*, l'attenzione principale è rivolta a professionisti e freelance attivi in ambito ICT, che contribuiscono a creare una comunità di utenti, con esigenze uniche e particolari, che attualmente manca all'interno del panorama digitale odierno. Questa comunità, ricca di competenze specializzate e spesso impegnata in progetti di ogni genere, necessita di strumenti mirati per navigare attraverso le risorse digitali disponibili in modo efficiente ed efficace, così da avere anche la possibilità di scoprire nuovi strumenti che possano agevolare in modo consistente il loro operato.

La directory *Pop Tools* è stata quindi progettata tenendo a mente le esigenze e i comportamenti di questi utenti specifici che ne hanno dettato l'architettura, le funzionalità e il design così da concepire una piattaforma che possa rispondere in modo mirato alle sfide che professionisti e freelance affrontano quotidianamente. Essa contiene infatti una raccolta di strumenti e siti web utili, suddivisi in categorie che spaziano dallo sviluppo come "*Web design*" o "*Software development*" al settore grafico come "*Art & Illustration*" al marketing come "*SEO*" o "*Customer care*". Grazie a tutte queste categorie l'utente potrà filtrare le sue ricerche in base alle sue necessità per trovare i tools migliori per il suo prossimo progetto.

Questo elaborato si articola dunque in diverse sezioni chiave. Inizialmente, si esplorano i concetti base delle web directory, tracciando un quadro storico e analizzando il loro ruolo nel contesto digitale. Successivamente, si delinea la progettazione e l'implementazione della piattaforma *Pop Tools*, mettendo in luce le scelte tecnologiche, di design e le funzionalità chiave. Infine, vengono presentati i risultati ottenuti e le riflessioni sulle potenziali implicazioni e sviluppi futuri del progetto.

## 2 Concetti base

Le web directory possono essere considerate come uno dei pilastri fondamentali per comprendere l'evoluzione del *web* specialmente nella sua fase iniziale. Essendo il *WWW (world wide web)* un mondo digitale in continua espansione dove regna l'informazione, è infatti sorta la necessità di creare una mappa da utilizzare per riuscire a muoversi tra le miriadi di risultati che esso ha da offrire. Ed è qui che emerge la figura della web directory.

### 2.1 Concetto di *web directory*

Una web directory[37] non è altro che una raccolta organizzata e curata di link a siti web esterni generalmente suddivisi per categorie tematiche, che possono spaziare da ampi temi come "arte" o "scienza", a sottocategorie più specifiche come "pittori italiani del 900" o "biologia marina". La peculiarità di una web directory rispetto ad altri strumenti come un motore di ricerca è la presenza del fattore umano.

Mentre questi ultimi utilizzano algoritmi per indicizzare e classificare le pagine web in autonomia, una web directory necessita di una redazione che si occupi di selezionare e categorizzare manualmente i link al suo interno. Questa selezione manuale ha sempre rappresentato un valore aggiunto in termini di qualità, precisione e affidabilità dei contenuti presenti nella raccolta.

Inoltre per comprendere al meglio il significato di queste directory, si consideri l'esempio seguente: si stanno svolgendo ricerche sui cambiamenti climatici e sul loro impatto sulla vita marina, si desidera perciò trovare fonti affidabili per la ricerca. Si è però sopraffatti dalla grande quantità di informazioni disponibili online finendo per rendere la ricerca inefficiente e dispendiosa in termini di tempo ed energie. È qui che una web directory può essere preziosa. Accedendo a una directory affidabile incentrata su argomenti ambientali, si potrebbe infatti navigare rapidamente attraverso siti Internet preselezionati relativi al cambiamento climatico e alla biologia marina senza dover vagliare risultati di ricerca irrilevanti.

Sul lato tecnico, gestire una web directory non è un compito da poco. L'architettura dell'informazione è fondamentale e la qualità dei contenuti presenti deve essere impeccabile. Sono necessari criteri chiari per la categorizzazione e la sottocategorizzazione dei siti così da poter essere implementati in modo efficiente da tutta la redazione.

L'usabilità e l'esperienza utente sono il fulcro del sistema e l'obiettivo della raccolta è quello di far trovare all'utente ciò che cerca in pochi click, onde evitare un suo abbandono della directory. Il problema maggiore da affrontare è la scalabilità, che con migliaia, se non milioni, di siti da valutare e catalogare rende dispendioso in termini di tempo e risorse il mantenimento della web directory per la quale è inoltre necessario un *back-end* solido ed efficiente.



Figura 1: Struttura di una web directory

### 2.1.1 La nascita

Riflettendo sulle origini del web, ci si trova di fronte a un paesaggio digitale completamente diverso da quello attuale. Gli anni '90 hanno visto la nascita e la crescita esponenziale di *Internet* che era molto diverso da come è conosciuto oggi. Come per ogni territorio ancora inesplorato non esistevano delle "mappe" ed è in questo contesto che nasce la necessità di creare una guida, un catalogo che contenesse liste di siti web suddivisi per categoria. *Yahoo!*, per esempio, fu uno dei pionieri in questo campo e inizialmente era proprio una web directory.

Ciò che oggi è riconosciuto come un gigante del web, iniziò come una semplice lista di siti web favoriti, organizzati per argomento, da due studenti universitari, *Jerry Yang* e *David Filo* nel 1994. La sua caratteristica distintiva era proprio la cura editoriale nell'inclusione dei siti web selezionati appunto manualmente da una redazione che ne garantiva la cura.

### 2.1.2 Perché erano importanti

Per rispondere a questa domanda bisogna tornare indietro a quei tempi dove internet era la novità e le persone avevano bisogno di strumenti per navigare in modo facile e intuitivo senza perdersi in un mare di contenuti non desiderati. In tutto questo le web directory erano perfette in quanto servivano e servono tuttora a categorizzare e offrire una panoramica generale sulle risorse disponibili al loro interno migliorando così l'esperienza dell'utente finale.

Grazie a questi "elenchi" concisi di siti, venivano presentate all'utente solamente le informazioni chiave in un formato chiaro e organizzato consentendogli di individuare rapidamente i contenuti pertinenti alla sua ricerca.

Essi servivano inoltre a far avvicinare l'utente ad altri contenuti correlati che potessero essere potenzialmente di suo interesse. Le web directory conferivano anche una sorta di certificazione sulla qualità del contenuto presente. Il fatto che un sito fosse in una directory significava infatti che era stato valutato precedentemente da qualcuno che lo aveva considerato degno di nota.

Perciò essere in una di queste raccolte era considerata una forma di garanzia e di tutela per il fruitore del sito web. Questo processo di selezione e categorizzazione era essenziale, soprattutto in un'epoca dove non esistevano ancora algoritmi di ricerca efficienti e precisi come al giorno d'oggi.

Oltre a ciò avevano anche un ultimo grande vantaggio, le web directory puntavano a creare delle vere e proprie comunità di persone, concentrando un pubblico di utenti con esigenze ed interessi simili sulla stessa piattaforma. Molti siti web erano infatti creati anche da appassionati di un particolare argomento o settore, ed essere inclusi in una directory era un grande riconoscimento del duro lavoro e della passione investiti nello sviluppo del progetto nonché una vetrina online per altri utenti interessati.

### 2.1.3 Motori di ricerca vs *web directory*

Non tutto era roseo nel mondo delle web directory. Con l'aumento della popolarità e dell'utilizzo di internet venivano pubblicati siti web ad un ritmo elevatissimo. Questa mole di nuovi contenuti in rete ha reso difficile per le web directory mantenere il passo poiché, necessitando di una redazione che controllasse, curasse e mantenesse il sistema, ha iniziato a mostrare i propri limiti.

In più l'ingresso nel panorama attuale di nuovi motori di ricerca come *Google*, con algoritmi sempre più sofisticati, precisi ed efficienti, ha iniziato a cambiare le regole del gioco, portando la ricerca sul web ad un livello successivo.

Tuttavia, le directory non hanno perso la loro rilevanza ma, al contrario, si sono evolute concentrandosi principalmente su nicchie specifiche o settori particolari. Questa specializzazione ha permesso loro di offrire valore in aree che i motori di ricerca generalisti non potevano raggiungere con la stessa efficienza e precisione.

## 2.2 Evoluzione delle *web directory*

Il ruolo delle web directory oggi è una tematica molto importante affrontata durante la fase di ideazione del progetto *Pop Tools* in quanto essendo il concetto di directory nato insieme a *Internet* è sempre stato in continua evoluzione. Grazie al paragrafo precedente è stata delineata la natura e l'origine delle directory, mentre in questo verranno affrontate le motivazioni che hanno spinto queste raccolte digitali al cambiamento.

### 2.2.1 Yahoo! e l'ascesa delle *web directory*

L'era digitale<sup>1</sup> è un periodo in cui il mondo è cambiato radicalmente grazie al forte sviluppo tecnologico, caratterizzato dalla diffusione e dall'utilizzo massivo delle nuove tecnologie digitali come computer con una potenza di calcolo sempre più elevata o la possibilità di collegarsi a *Internet* da qualsiasi luogo.

In quest'epoca c'è anche una enorme produzione costante di nuove risorse che in ogni istante vengono elaborate, archiviate, trasmesse e condivise in forma digitale da una parte all'altra del globo. Ciò ha generato un ecosistema online vasto e complesso formato da miliardi di pagine web, applicazioni, piattaforme *social* e servizi di ogni genere che necessita di una navigazione altrettanto efficiente e, in questo contesto, le web directory con la loro promessa di organizzare tutte le informazioni al loro interno con cura sembrerebbero la scelta ideale. Ma la realtà è ben più complessa.

Agli inizi di questa digitalizzazione, la crescita esponenziale dei contenuti online ha inizialmente amplificato il valore delle web directory dal momento in cui c'era un enorme bisogno di entità che potessero aiutare le persone a trovare ciò che cercavano all'interno di un *Internet* in costante espansione. Ecco perché aziende come *Yahoo!*, nei primi anni sono riuscite a guadagnare una larga fetta di mercato e di utenti offrendo loro la possibilità di reperire l'informazione in modo ordinato, categorizzato e valutato, offrendo un servizio di grande valore per quel periodo.

### 2.2.2 Nuovi ruoli e obiettivi delle *web directory*

Con il passare del tempo, l'ecosistema digitale ha iniziato a subire mutamenti radicali. Come anticipato l'avvento di *Google* ha iniziato a ridimensionare il ruolo delle web directory. L'introduzione del *PageRank (PR)*[8] da parte di *Google*, che è un algoritmo utilizzato per attribuire un punteggio alle pagine web basandosi sulla quantità e sulla qualità dei link in ingresso, ha rivoluzionato il modo in cui le informazioni venivano recuperate e mostrate agli utenti che, invece di utilizzare le categorie di una web directory, permettevano all'utilizzatore di svolgere una ricerca su *Internet* utilizzando il suo linguaggio naturale in maniera più semplice e intuitiva, e gli permettevano (e permettono tuttora) di ricevere risultati basati sulla rilevanza e popolarità delle pagine web.

Questo cambio di paradigma ha avuto ripercussioni profonde sul ruolo delle web directory. Mentre prima erano viste come una necessità, ora erano a rischio di diventare obsolete. Ma come spesso accade in un sistema in continua evoluzione, la funzione delle directory ha semplicemente iniziato a cambiare puntando ad una nicchia di riferimento differente.

Invece di competere direttamente con i motori di ricerca, molte web directory hanno iniziato a specializzarsi ulteriormente. Si sono concentrate su settori specifici, come l'arte, la scienza o l'industria, offrendo approfondimenti e risorse che i motori di ricerca generalisti potevano trascurare. Questa evoluzione ha portato alla nascita di directory di nicchia, altamente specializzate, che offrivano valore attraverso la loro expertise in settori specifici. Alcune di queste directory si sono concentrate su settori professionali, come legge o medicina, fornendo risorse preziose per specialisti e ricercatori.

### 2.2.3 La nascita delle *community*

Ma questa era digitale non solo ha cambiato il modo in cui le persone ricercano e accedono alle informazioni, ha anche dato la possibilità a tutti di crearne e condividerne costantemente di nuove. Con l'arrivo dei *social network* e delle piattaforme di condivisione, gli utenti hanno avuto la possibilità di diventare dei veri e propri *content creators* lasciando le web directory nella posizione di dover gestire un ulteriore flusso di contenuti generati dagli utenti.

Per far fronte a queste novità alcune web directory hanno dovuto reinventarsi dando la possibilità agli utenti di condividere dei propri *feedback* all'interno della comunità e permettendogli inoltre di votare e recensire le risorse presenti. In questo modo, la cura non era più solo nelle mani degli amministratori della

---

<sup>1</sup>Era digitale: L'arco di tempo iniziato con l'introduzione e l'uso esteso degli elaboratori elettronici e delle tecniche a essi connesse.[11]

directory, ma diventava un processo collettivo avente il duplice vantaggio sia di mantenere la directory qualitativamente aggiornata sia di instaurare un senso di comunità tra gli utenti a scopo di fidelizzazione.

#### **2.2.4 Adattamento e successo delle *web directory***

Nel mentre, con la crescente diffusione di dispositivi mobili all'interno del mercato, la semplice navigazione su internet ha iniziato a spostarsi verso di essi necessitando un adeguamento della visualizzazione dei contenuti che fino a quel momento erano stati sviluppati solamente per *desktop*.

Anche questo ha presentato nuove sfide e opportunità per le directory che si sono dovute adeguare per riuscire a mostrare le informazioni al loro interno in modo efficace anche su schermi più piccoli senza ridurre le funzionalità. L'opportunità era quella di integrarsi con app e servizi specifici per *mobile*, offrendo un valore aggiunto attraverso la cura e l'organizzazione delle informazioni.

In termini di *business*, l'era digitale ha portato anche a nuovi modelli di monetizzazione per le web directory. Mentre inizialmente molti si basavano su pubblicità o sponsorizzazioni, con l'avvento delle piattaforme di *e-commerce* e la crescita del marketing affiliato alcune web directory hanno iniziato a esplorare nuovi flussi di reddito.

Una web directory specializzata in prodotti elettronici potrebbe per esempio guadagnare commissioni sugli acquisti effettuati attraverso i suoi link, oppure potrebbe offrire la possibilità ai possessori di un sito di mostrarlo tra i primi risultati della directory per aumentare il traffico e la credibilità agli occhi degli utenti. In tutto ciò, anche l'analisi dei dati ha iniziato ad essere molto importante in ambito digitale. Con l'avvento di strumenti analitici avanzati, le web directory ora potevano monitorare e analizzare il comportamento degli utenti in tempo reale, ottimizzando la presentazione delle informazioni e offrendo esperienze personalizzate.

Da tutto ciò si può dunque evincere che le web directory non erano (e non sono) destinate a scomparire, ma piuttosto a evolvere e adattarsi all'evoluzione tecnologica. La loro capacità di organizzare, curare e presentare le informazioni rimaneva inestimabile, anche se il contesto e le modalità di consegna cambiavano di anno in anno. E questo è il segno distintivo di qualsiasi strumento o servizio di successo: la capacità di adattarsi, reinventarsi e continuare a fornire valore in un mondo in rapido cambiamento.

## 2.3 Importanza delle risorse digitali per professionisti e freelance

“Le tecnologie digitali rappresentano oggi non solo il luogo primario di incontro fra domanda e offerta di lavoro ma veri e propri strumenti lavorativi e professionali. In particolare, nelle industrie della comunicazione e della conoscenza le professioni sono sempre più «digitali», in quanto la Rete interviene in maniera determinante non solo nei processi di ricerca di lavoro, ma anche nella natura e nello svolgimento dello stesso.”[3]

Al giorno d’oggi infatti si è circondati da risorse digitali di ogni genere, che hanno portato professionisti e freelance ad adottarli e implementarli all’interno delle loro pratiche lavorative così da rendere il proprio operato più agile ed efficiente sviluppando anche nuove strategie di *marketing* ed evolvendo le modalità di interazione con i clienti ed il mercato in generale.

Con l’avvento di tutte queste nuove possibilità, la storia della professione ha subito una vera e propria rivoluzione paragonabile all’introduzione del telefono nella vita di tutti. Ciò che una volta richiedeva ore di ricerca in biblioteca o lunghi viaggi per consultazioni e *meeting* può ora essere eseguito in pochi secondi, con pochi click o tap su un dispositivo, non solo riducendo i tempi nella raccolta di informazioni, ma anche migliorandone la qualità e precisione che precedentemente erano impensabili.

### 2.3.1 Impatto delle risorse digitali sui *freelance*

I professionisti e freelance odierni di qualsiasi settore, oggi hanno a disposizione una varietà di risorse digitali che possono migliorare esponenzialmente le loro competenze, gestire le loro attività e servire al meglio i loro clienti portando vantaggi ad entrambe le parti. Questi benefici giovano in particolar modo per i freelance, che spesso devono fare affidamento su loro stessi per una moltitudine di compiti che vanno dalla gestione dei clienti alla contabilità e dalla promozione dei servizi alla formazione continua.

Per esempio, consideriamo un freelance che opera nel settore “*Graphic Design*”. In passato gran parte del suo lavoro sarebbe stato quello di utilizzare strumenti fisici per grafici come matite, carta e righelli. In questa era digitale, grazie a *software* avanzati come *Adobe Photoshop*[2], possono sviluppare e creare in modo più semplice e direttamente sul loro computer *design* complessi che prima avrebbero richiesto un’enorme mole di tempo e lavoro. In più grazie a piattaforme come *Dribbble*[13] o *Pinterest*[30] possono connettersi con grafici ed artisti da tutto il mondo per accrescere le loro conoscenze e prendere ispirazione per i prossimi lavori che potrebbero aver trovato su servizi online come *Fiverr*[15] o *Upwork*[43] che offrono l’opportunità di trovare nuovi clienti online facilmente.

### 2.3.2 L’importanza dei *MOOCs*

Le risorse digitali, oltre ad essere ottimi strumenti operativi, sono anche grandi risorse per l’apprendimento, la crescita personale e professionale. I *MOOCs* (Corsi massivi online aperti a tutti)[7] per esempio sono diventati un sistema di formazione online di crescente importanza a livello mondiale.

Questi corsi sono destinati a un vasto pubblico e sono accessibili solo via web. La maggior parte dei *MOOCs* è reperibile in inglese e in modo gratuito, ma spesso richiedono un pagamento per l’ottenimento di certificazioni. Non esistono linee guida standardizzate per le certificazioni ma ogni istituzione stabilisce le proprie modalità.

Il mercato dei *MOOCs* è dominato da grandi università o consorzi di università. Nonostante non ci siano standard internazionali per i *MOOCs*, l’interazione e la collaborazione online tra studenti, docenti e tutor sono centrali. Questi non sono semplicemente lezioni universitarie tradizionali trasmesse online ma sono progettati specificamente per l’ambiente digitale, con contenuti provenienti da diverse fonti e organizzati in modo flessibile.

Grazie ad essi l’apprendimento è esteso a tutti, consentendo a freelance e professionisti di rimanere aggiornati sulle ultime tendenze e tecniche del loro settore, indipendentemente dalla loro posizione geografica o dalle risorse economiche.

E non sono solo le università tradizionali a offrire queste risorse. Al giorno d’oggi esistono piattaforme molto famose come *Udemy*[42] e *Coursera*[10] che offrono corsi specifici per una vasta gamma di competenze professionali, dalla programmazione alla scrittura, dalla fotografia al *marketing*. Questi corsi *online* sono spesso tenuti da professionisti del settore e ampiamente riconosciuti nel loro campo, offrendo *insights* pratici e consigli basati sulla loro esperienza.

### 2.3.3 Strumenti digitali innovativi

Un altro aspetto fondamentale delle risorse digitali riguarda la gestione e l’organizzazione dell’attività professionale. Strumenti come *Trello*[41], *Asana*[4] o *Monday*[25] hanno rivoluzionato il modo in cui i

professionisti e i freelance gestiscono i progetti, monitorano i progressi e collaborano con i clienti o con altri professionisti.

Questi strumenti offrono una visione chiara del flusso di lavoro, permettendo una pianificazione efficace e facilitando la comunicazione che, in effetti, è un altro settore che ha beneficiato enormemente dall'avvento delle risorse digitali. *Skype*[36], *Zoom*[46] e *Teams*[23], per citarne solo alcuni, hanno reso possibili *meeting* virtuali, presentazioni e *workshop*, eliminando la necessità di viaggi costosi e lunghi e rendendo la collaborazione più semplice e immediata.

Nel contesto della promozione e del *marketing*, le risorse digitali hanno aperto orizzonti completamente nuovi. Grazie a piattaforme come *Facebook*[14], *Instagram*[20] e *LinkedIn*[21] non solo permettono ai professionisti di costruire un *brand* personale e di interagire con i clienti, ma offrono anche strumenti avanzati di *advertising* che permettono di raggiungere segmenti di pubblico specifici con precisione chirurgica e offrendo possibilità di una personalizzazione del pubblico target mai vista in precedenza.

Inoltre, gli *analytics* integrati in molte di queste piattaforme come *Google Analytics*[18] o *Meta Business Suite*[22] forniscono dati preziosi sull'efficacia delle campagne pubblicitarie, sulle preferenze dei clienti e sulle tendenze del mercato. Questa capacità di misurare e analizzare in tempo reale le *performance* è essenziale per adattare e affinare le strategie in un mercato che varia giornalmente.

Ma forse uno degli aspetti più importanti che hanno introdotto queste risorse digitali è il modo in cui hanno abbattuto le barriere geografiche. Oggi infatti un freelance può offrire con semplicità i suoi servizi a clienti sparsi in tutto il mondo, collaborare con professionisti in altri continenti e partecipare a conferenze virtuali con partecipanti da ogni angolo del pianeta.

Questa globalizzazione ha portato con sé nuove opportunità, ma anche nuove sfide. La concorrenza è ora globale, e la necessità di differenziarsi e di fornire valore aggiunto è più pressante che mai. In questo contesto, le risorse digitali non sono solo strumenti utili: sono essenziali, consentendo ai professionisti di rimanere competitivi, di adattarsi alle nuove tendenze e di navigare con successo in un mercato sempre più complesso e interconnesso.

## 2.4 Web directory: stato dell'arte

Nell'era digitale le web directory sono sempre state un punto di riferimento fondamentale per organizzare, classificare e accedere alle risorse online in modo semplice. Queste piattaforme che, come riportato in precedenza, erano inizialmente elenchi statici di siti web, sono diventate nel tempo strumenti sofisticati che offrono una vasta gamma di funzionalità che ampliano il concetto di directory. Esse hanno dovuto reinventarsi per rimanere pertinenti e competitive all'interno del mercato odierno, differenziandosi sia per sopperire alle differenti esigenze degli utenti sia per offrire vantaggi unici che le rendessero una novità.

Mentre le directory tradizionali come *DMOZ*[12] e *Yahoo! Directory*[45] hanno lasciato un segno indelebile nella storia del web, piattaforme come *Producthunt*, una vetrina per *start-up* e innovazioni tecnologiche, e *Futurepedia*, una directory specializzata in strumenti basati sull'intelligenza artificiale, stanno tracciando una nuova strada per il futuro, dimostrando che c'è ancora un posto per le web directory nell'era moderna.

Ad oggi entrambe le piattaforme hanno raggiunto una certa notorietà ed è particolarmente interessante osservare come pur facendo evolvere il concetto di directory verso una forma più interattiva e *community-driven* restino comunque fedeli al modello standard.

### 2.4.1 Product Hunt

Product Hunt[33] si è affermato come un incubatore virtuale per idee e innovazioni nel mondo delle *start-up* tecnologiche. Fondato da *Ryan Hoover* nel 2013, Product Hunt è un sito web che offre agli sviluppatori una piattaforma per lanciare il loro nuovo prodotto o servizio. Una volta registrato, lo sviluppatore dovrà caricare la *web-app* o l'applicazione mobile che ha portato a termine e sarà poi la redazione della piattaforma a stabilire se includerla nella directory o meno. Il loro pubblico di riferimento è piuttosto variegato e comprende:

- **Imprenditori e Startupper:** Questo gruppo utilizza Product Hunt come piattaforma per lanciare nuovi prodotti o servizi, ottenere *feedback* e attirare potenziali investitori.
- **Investitori:** Questi utenti cercano opportunità di investimento promettenti e utilizzano la piattaforma per scoprire *start-up* emergenti e innovative.
- **Blogger:** Product Hunt è una fonte costante di notizie e tendenze nel mondo tech, che possono essere utilizzate da questi ultimi per trovare materiale interessante su cui scrivere.
- **Professionisti e Freelance:** Sviluppatori, *designer*, *manager* e altri professionisti del settore possono utilizzare Product Hunt per rimanere costantemente aggiornati sulle ultime innovazioni e trovare strumenti che possano migliorare la qualità del loro lavoro.
- **Utenti alla ricerca di strumenti innovativi:** Questo gruppo è interessato a scoprire e provare nuovi prodotti, applicazioni o servizi e rappresenta la parte fondamentale della *community* grazie ai loro *feedback* e recensioni.

La parte migliore di Product Hunt è infatti l'elemento comunitario. Nella *homepage* della directory i nuovi prodotti vengono classificati in base alla loro popolarità che è determinata dai voti degli altri membri della *community*. Più voti positivi otterrà un prodotto e più in alto si troverà nella lista.

Su ogni pagina prodotto c'è anche una sezione commenti in cui i membri possono discutere e lasciare una recensione. Che si tratti di porre una domanda al creatore, mostrare il loro supporto o vedere cosa pensano gli altri la sezione commenti rimane il posto perfetto. Le discussioni possono inoltre continuare nella sezione "*Community*" del sito, dove i membri possono pubblicare mini post sul blog della piattaforma per informare o fare domande alla comunità.

Da un punto di vista di esperienza utente il design intuitivo di Product Hunt e la sua usabilità rendono facile la navigazione tra le diverse categorie di prodotti. In più le dinamiche di interazione tra gli sviluppatori dei prodotti, i consumatori e gli esperti del settore creano un terreno fertile per il *co-working*, la co-creazione e la co-innovazione, facendo di Product Hunt un modello di *business* quasi inedito nel suo genere. In un mondo in cui il capitale sociale è spesso tanto prezioso quanto il capitale finanziario, la piattaforma è diventata un catalizzatore indispensabile per l'innovazione collaborativa.

### 2.4.2 Futurepedia

Futurepedia[16] è la più grande web directory contenente strumenti relativi all'intelligenza artificiale disponibile sul mercato. Sviluppata appositamente per *data scientist*, ricercatori e sviluppatori, questa directory è progettata per aiutare gli utenti a navigare nel panorama in continua crescita dell'IA fornendo loro un unico punto in cui possono facilmente trovare e confrontare strumenti in varie categorie come *machine learning*, analisi dei dati e modelli di simulazione del linguaggio.

La directory aiuta inoltre gli utenti a confrontare e scoprire nuovi strumenti appropriati alle loro esigenze individuali e al loro *budget* fornendo descrizioni e recensioni approfondite per ciascuno strumento elencato, creando così un punto di raccolta centralizzato per tutti gli ultimi sviluppi riguardo l'intelligenza artificiale. La piattaforma fornisce infatti anche la possibilità di votare e recensire i prodotti presenti in essa per dare voce alla comunità e renderla più attiva e coesa.

Futurepedia è inoltre una piattaforma recente (inizio 2023), pubblicata durante lo sviluppo di questo progetto di tesi e in pochi mesi ha raggiunto oltre 150 mila utenti registrati e un traffico mensile di 400 mila visitatori singoli. Questi numeri impressionanti riflettono la crescente popolarità e la necessità di una piattaforma che cataloghi e recensisca gli strumenti di IA disponibili sul mercato. Inoltre ha rafforzato l'idea che è stata la base per lo sviluppo di *Pop Tools*, fornendo la prova che una web directory ben strutturata, che si rivolga ad una nicchia specifica e offra la possibilità di creare una comunità sia ritenuta ancora molto utile e possa ricoprire un ruolo significativo all'interno dell'era digitale.

### 2.4.3 Altre directory di rilievo

Nel panorama delle web directory, oltre a Product Hunt e Futurepedia, si celano altri strumenti che meritano considerazione. Queste piattaforme, ciascuna con le proprie peculiarità e approcci, hanno contribuito in modo significativo alla scoperta e alla condivisione di risorse digitali su Internet. Di seguito una breve descrizione delle più degne di nota:

- **Toools Design**[40]: Toools Design è una piattaforma che offre un vasto archivio di risorse di design di alta qualità. Questa directory è quindi rivolta ad un pubblico di grafici o appassionati, interessati a risorse come illustrazioni, icone, mockup, tipografia e altre che si allineano alle loro specifiche esigenze.

Un punto di forza di Toools Design è il suo impegno verso la comunità dei creatori dato che, oltre a condividere risorse, la piattaforma offre anche una newsletter settimanale che mette in evidenza i cinque migliori nuovi strumenti di design, garantendo agli utenti aggiornamenti regolari sulle ultime tendenze.

- **Insanely Useful Websites**[19]: Insanely Useful Websites è una piattaforma che presenta una collezione di siti web "incredibilmente utili". Questi siti sono stati selezionati per la loro unicità, utilità e innovazione, rendendoli risorse preziose per chiunque cerchi strumenti, servizi o informazioni online.

Anche questo strumento offre una newsletter settimanale in cui propone diverse soluzioni utili in base all'ambito, ma uno svantaggio rilevante è il fatto che i siti web non siano suddivisi per categorie ma solamente inseriti all'interno di una unica lista rendendo la ricerca di tools specifici molto complessa.

- **Ncode**[27]: Ncode è una startup italiana con lo scopo di avvicinare gli utenti agli strumenti No Code, ideali per chi desidera realizzare progetti digitali senza la necessità di saper scrivere codice. Questi strumenti sono stati selezionati in base alla loro efficacia, facilità d'uso e innovazione e sono suddivisi in base a differenti categorie incentrate per lo più sullo sviluppo web.

## 3 Descrizione del progetto

### 3.1 La piattaforma *Pop Tools*

Nel contesto digitale attuale sempre in evoluzione dove ogni 3 secondi[17] si assiste alla nascita di un nuovo sito web e dove la quantità delle informazioni è ritenuta migliore rispetto alla qualità di quest'ultime dai motori di ricerca, unita ad una società sempre più focalizzata e dipendente dalle risorse presenti su *Internet* è emersa la mancanza di una piattaforma che potesse rispondere alle esigenze specifiche di professionisti e freelance del settore digitale e li aiutasse a districarsi tra la moltitudine di servizi e informazioni superflue presenti in rete.

È nata quindi l'idea di sviluppare *Pop Tools*, una web directory che possa fornire a questa tipologia di utenti solamente risorse qualitative per la risoluzione di esigenze specifiche riguardo il loro settore, ottimizzando e semplificando il loro processo di ricerca e scoperta di nuovi strumenti e contenuti pertinenti.

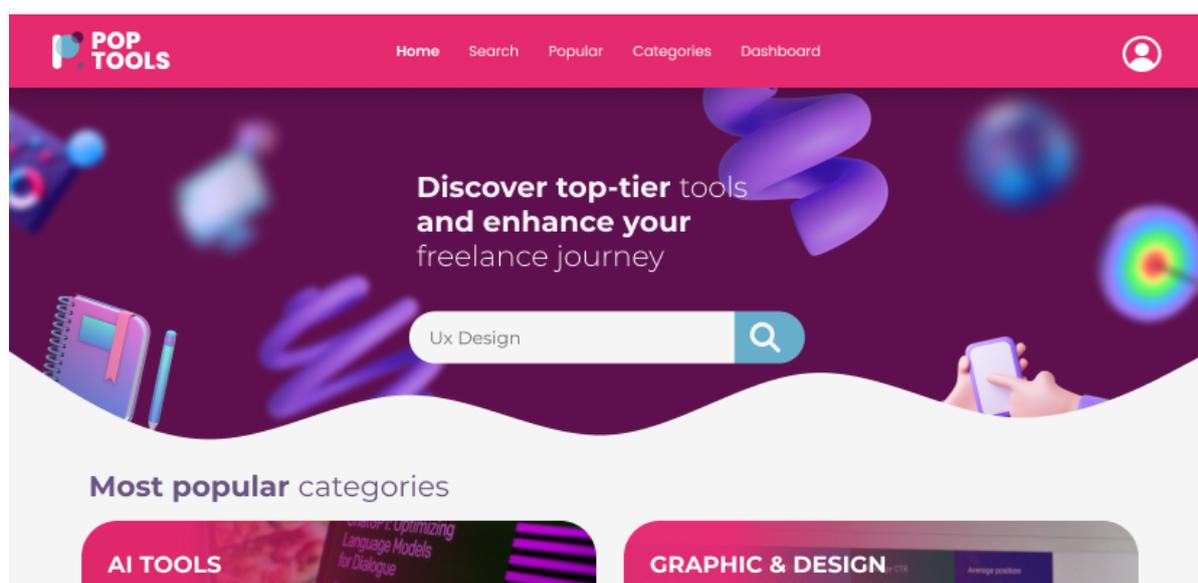


Figura 2: Homepage di PopTools

#### 3.1.1 Semplificare la ricerca

Come anticipato una delle principali sfide che i professionisti e i freelance affrontano quando si affacciano a *Internet* è la sovrabbondanza di informazioni e risorse. Questa vastità di opzioni risulta molto spesso travolgente per qualsiasi tipo di utente e, per individuare ciò che esattamente si sta ricercando, si rischia di spendere fin troppe energie.

*Pop Tools* non fa altro che proporsi di superare questa sfida affrontata da tutti, tutti i giorni, fornendo un ambiente semplice e organizzato, in cui gli utenti possono navigare attraverso categorie specifiche che rispecchiano le loro esigenze senza perdersi in lunghe ricerche senza meta. Questo approccio semplifica notevolmente il processo di ricerca e scoperta di una nuova risorsa e consente agli utenti di accedervi più rapidamente e in maniera più efficiente.

#### 3.1.2 Qualità dei contenuti

Il fulcro del servizio inizia dunque durante la fase di ricerca dei contenuti, che d'ora in poi chiameremo tools, da inserire all'interno della directory e suddivisi per categorie che riflettono tutte le aree chiave delle attività svolte da professionisti e freelance del mondo digitale.

Questo meticoloso processo di selezione è svolto dall'entità principale della piattaforma ovvero la redazione, che si occupa di valutare attentamente ogni sito web prima di includerlo all'interno della directory. Questo approccio di curatela consente di garantire solamente ai tools più utili e validi di entrare a far parte del servizio così da essere considerato dagli utenti finali una fonte affidabile che eroga contenuti di qualità e utili per la loro carriera professionale.

### 3.1.3 L'importanza della *community*

Un elemento distintivo della piattaforma è la componente di interazione e partecipazione della *community*. Gli utenti non solo possono navigare la directory alla ricerca di nuovi strumenti come descritto in precedenza ma gli è offerta anche la possibilità di contribuire attivamente suggerendo nuovi siti web da includere potenzialmente nella lista, fornendo una prospettiva più ampia e diversificata sulle risorse disponibili riducendo anche i costi in termini di tempo ed energie che la redazione avrebbe dovuto spendere per la ricerca autonoma.

Prima che le nuove richieste vengano messe online devono passare per un processo di selezione e controllo qualità da parte dell'amministrazione, adibito a garantire che vengano pubblicati solamente contenuti rilevanti e di alto livello. Questo processo di interazione tra utenti e redazione favorisce un ambiente collaborativo che arricchisce il rapporto tra le parti e garantisce che le risorse vengano costantemente aggiornate e ampliate.

Un altro modo in cui gli utenti possono contribuire alla *community* è tramite la votazione e pubblicazione di commenti e recensioni riguardo tools che hanno avuto modo di utilizzare, testare e valutare attentamente. La possibilità di esprimere opinioni attraverso recensioni e commenti offre un canale diretto per la condivisione di esperienze personali, opinioni e suggerimenti relativi a ciascun sito web presente nella directory producendo valore non solo per la *community* stessa e gli utenti, che grazie alle recensioni potranno valutare in modo più attento l'utilizzo di un servizio più che un altro, ma anche per la redazione, che grazie ad essi potrà trarre spunti per identificare le risorse più popolari e pertinenti e, allo stesso tempo, valutare le necessità emergenti degli utilizzatori. Ciò si traduce in una piattaforma sempre all'avanguardia e capace di adattarsi rapidamente alle tendenze e alle richieste in evoluzione del mercato digitale.

## 3.2 Vantaggi del progetto

L'implementazione del progetto *Pop Tools* mira ad offrire una serie di vantaggi chiave per professionisti e non, dell'ambito digitale, che possono sicuramente influenzare in modo positivo il loro quotidiano. Tutti i diversi aspetti del servizio sono stati progettati proprio per migliorare la produttività, l'efficienza e la qualità del lavoro dell'utente finale così da permettergli di superare tutte le sfide poste in questo periodo di forte digitalizzazione.

### 3.2.1 Accesso facilitato alle risorse digitali

L'accesso rapido e facilitato alle risorse digitali è un vantaggio fondamentale offerto dal progetto. Come anticipato nel capitolo precedente ci troviamo in un momento storico caratterizzato da una produzione massiva di informazioni online che rende un'impresa ardua e *time-consuming* il trovare risorse pertinenti e di alta qualità.

La piattaforma *Pop Tools* risolve in modo semplice questo problema andando ad offrire all'utilizzatore un accesso intuitivo ad una vasta gamma di informazioni selezionate ed organizzate in categorie specifiche sempre relative al mondo digitale come "*Graphic & Design*" o "*Business Growth*".

Questo sistema strutturato in categorie ad albero permette agli utenti di trovare rapidamente tools rilevanti per la loro attività e di scoprirne di nuovi evitando così un dispendio di tempo eccessivo causato da una ricerca "per tentativi". La grande possibilità di poter accedere ad una varietà di strumenti e contenuti in modo così chiaro e strutturato permette a *Pop Tools* di ottimizzare il processo decisionale dell'utilizzatore incrementando la sua efficienza nell'ottenimento di risultati concreti.

### 3.2.2 Riduzione del tempo e dei costi

Il risparmio di tempo e denaro è un obiettivo cruciale per qualsiasi imprenditore di ordine e grado. La piattaforma *Pop Tools* offre una soluzione diretta a questa esigenza andando ad eliminare la necessità di navigare attraverso innumerevoli siti web alla ricerca di risorse adeguate allo scopo.

La redazione infatti, composta da esperti del settore, svolge un ruolo di filtro, selezionando solo i siti web più validi e utili per ciascuna categoria. Questo meticoloso processo di selezione svolto a priori dalla piattaforma consente agli utenti di risparmiare una grande quantità di tempo prezioso, poiché possono essere sicuri che le risorse presenti nella directory siano di alta qualità e rilevanza. Inoltre per l'imprenditore, il risparmio di tempo è strettamente legato ad una riduzione dei costi operativi, poiché il tempo risparmiato può essere dedicato a compiti più produttivi e remunerativi, invece di essere speso in ricerche infruttuose.

### 3.2.3 Collaborazione e condivisione

La natura collaborativa della piattaforma rappresenta un altro vantaggio cruciale per i professionisti e i freelance. La possibilità di condividere nuovi siti web attraverso la funzione di invio proposte per la possibile inclusione offre un'opportunità unica di contribuire alla crescita della directory.

Questa funzionalità favorisce l'interazione tra gli utenti e promuove uno scambio dinamico di risorse e conoscenze. Ogni utente ha la possibilità di condividere risorse che ha trovato utili nel suo percorso professionale, consentendo ad altri membri della *community* di beneficiarne a loro volta dandogli inoltre la possibilità di commentarle e votarle nell'apposita sezione delle recensioni, rafforzando i due vantaggi precedenti e aggiungendo un'ulteriore selezione e raffinazione dei contenuti svolta però dalla *community* stessa. Questa sinergia tra utenti amplia costantemente l'offerta di risorse presenti sulla piattaforma, creando un ambiente in cui il valore della condivisione è alla base della crescita collettiva.

### 3.2.4 Supporto alla crescita professionale

Un elemento distintivo del progetto è il supporto dedicato alla crescita professionale degli utenti finali. La presenza di categorie specifiche per la formazione e l'apprendimento online rappresenta una risorsa inestimabile per coloro che desiderano migliorare le proprie competenze digitali.

*Pop Tools* non vuole essere infatti solamente un luogo dove trovare tools e risorse pratiche ma anche una piattaforma che possa offrire contenuti sia informativi che formativi. Dando per esempio la possibilità a tutti di accedere a guide, tutorial e corsi online specifici all'ambito digitale e creando un ambiente stimolante in cui professionisti e freelance possano costantemente accrescere le loro conoscenze sviluppando e apprendendo nuove abilità così da aumentare la loro competitività all'interno di un mercato in continua crescita.

### 3.2.5 Sicurezza e protezione dei dati

La sicurezza dei dati è un altro aspetto critico per qualsiasi professionista o freelance che opera nel mondo digitale. La piattaforma vuole distinguersi per l'impegno devoluto alla sicurezza e protezione dei dati degli utenti essendo questo tema molto importante in un contesto in cui è proprio l'utente finale, con le sue informazioni, a diventare merce di scambio per le multinazionali globali.

La redazione non si preoccupa solamente di selezionare siti web di alta qualità ma si pone come obiettivo di ricerca anche l'affidabilità e sicurezza dei tools. Per *Pop Tools* la protezione dei dati degli utenti è una priorità assoluta e presentargli risorse affidabili e sicure consente loro di operare in un ambiente protetto, preservandone la *privacy*.

### 3.2.6 Semplicità di utilizzo

L'interfaccia utente intuitiva e *user-friendly* è un altro vantaggio significativo del progetto. La piattaforma è stata progettata per essere facilmente accessibile e utilizzabile da utenti con diversi livelli di competenza digitale e su tutti i dispositivi.

L'esperienza utente è stata ottimizzata attraverso diversi fattori come l'implementazione di un *layout* contenente solamente le informazioni fondamentali e che verranno approfondite nel capitolo 4 o l'aggiunta di numerosi filtri per una selezione più accurata dei risultati, una barra di ricerca avanzata e categorie chiaramente definite e stabilite dalla redazione.

Questo consente agli utenti di ottenere risultati in modo rapido e senza sforzo e di navigare attraverso la piattaforma in modo fluido. Al giorno d'oggi ottimizzare l'esperienza utente è fondamentale per garantire che i vantaggi offerti dal servizio siano accessibili a tutti e non vengano oscurati da una piattaforma mal progettata indipendentemente dal livello di competenza tecnologica.

### 3.2.7 Considerazioni

Nonostante i numerosi vantaggi elencati precedentemente, è importante sottolineare che la piattaforma *Pop Tools* non è immune da alcune limitazioni e sfide. È fondamentale riconoscere che la mole di informazioni online rappresenta una sfida comune a tutte le web directory e che, nonostante gli sforzi per semplificare l'accesso alle risorse digitali, la vastità di Internet può ancora presentare difficoltà nell'individuare esattamente ciò che si cerca.

Inoltre, l'organizzazione dei contenuti da parte della redazione, anche se eseguita da esperti, è soggettiva e potrebbe non sempre rispecchiare appieno le esigenze specifiche di ciascun utente. Ciò significa che, in alcuni casi, gli utenti potrebbero trovare più adatti i propri metodi di ricerca e selezione delle risorse.

Un'altra considerazione importante riguarda il fatto che molte delle funzionalità offerte da *Pop Tools*, come la condivisione di nuove risorse e la valutazione dei tools, richiedano una comunità di utenti attivi per conferire quel valore aggiunto ricercato dalla piattaforma. Per il suo successo essa richiede dunque una massa critica di utenti che contribuiscano, condividano e valutino le risorse al suo interno. È quindi necessario imporsi in modo deciso sul mercato attuando strategie di marketing e pubblicità per portare un flusso iniziale di utenti potenzialmente interessanti sulla piattaforma così da gettare le basi per ottenere in futuro una solida e attiva community.

### 3.3 Funzionalità chiave

L'efficacia della piattaforma *Pop Tools* è strettamente collegata alle sue molteplici funzionalità che sono state delineate per sopperire alle richieste di un pubblico ampio e variegato e per poter offrire tutti i vantaggi precedentemente descritti nell'elaborato. In questo paragrafo verranno descritte alcune delle funzionalità più importanti inserite all'interno del progetto, che mirano ad offrire agli utenti un'esperienza completa e altamente funzionale indipendentemente dal settore digitale in cui operano. Queste funzioni verranno approfondite ulteriormente nel capitolo 4 dove verrà affrontata, nello specifico, la progettazione dell'interfaccia utente.

#### 3.3.1 Struttura generale

Pop Tools si presenta come una web directory community-driven semplice e funzionale strutturata attorno a "card" che rappresentano singoli siti web o strumenti digitali. Ogni card fornisce informazioni chiare sul sito, inclusi titolo, descrizione, categoria di appartenenza, un pulsante per l'aggiunta del sito ai preferiti e un altro per raggiungere il sito web vero e proprio a cui il tool fa riferimento.

Queste card vengono mostrate sia nella Homepage (descritta nel paragrafo 4.3.1) che nella pagina Search (descritta nel paragrafo 4.3.2) raggiungibile svolgendo una ricerca tramite l'apposita barra.

L'organizzazione dei contenuti è basata su categorie e macrocategorie specifiche, tutte relative al mondo digitale che possono essere utilizzate per filtrare i risultati e permettere all'utente di ricercare solamente tools relativi ai suoi interessi.

Per un utente registrato è anche possibile recensire i siti web, inviare nuove proposte per l'aggiunta di un nuovo tool all'interno della directory, accedere al proprio profilo in cui si possono visualizzare le proprie informazioni personali e visualizzare anche tutti i siti web preferiti come descritto nel paragrafo successivo.

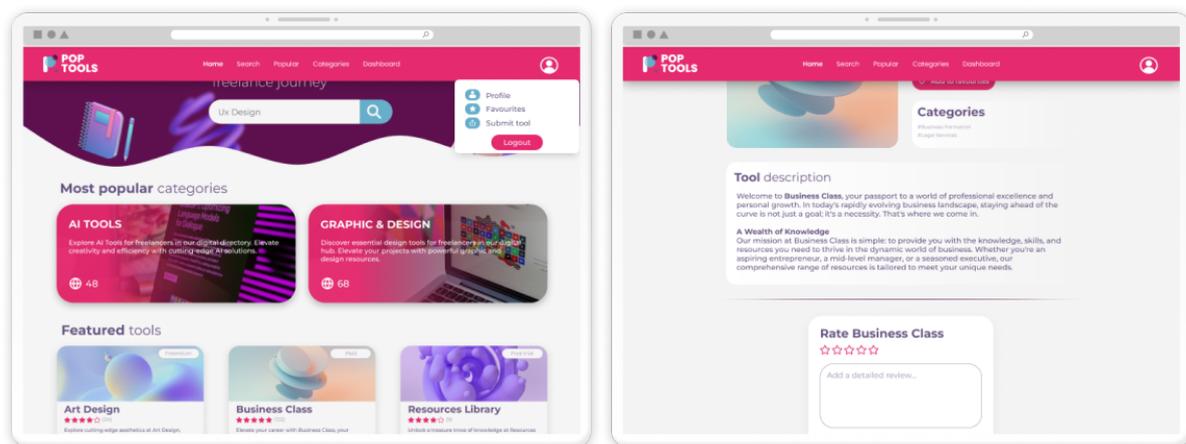


Figura 3: Schermata Homepage e Tool che mostrano la struttura generale di Pop Tools

#### 3.3.2 Aggiunta e gestione dei preferiti

La funzione più importante all'interno della piattaforma seppur banale è sicuramente l'implementazione dei "Preferiti". L'utente può infatti inserire un sito web all'interno dei suoi preferiti in modo da conservare le risorse digitali che ritiene più rilevanti durante la sua ricerca. Questa semplice funzionalità è alla base di tutta la web directory, creando uno spazio personalizzato e centralizzato in cui l'utente può consultare le risorse salvate in qualsiasi momento.

L'aggiunta di un tool ai preferiti è un processo semplice e intuitivo, reso possibile dall'icona a forma di cuore presente su tutte le card dei siti, che mostra inoltre il numero di *account* che hanno messo il servizio nei *favourites*, facendogli acquisire notorietà e sviluppando un potenziale interesse nei nuovi utenti che, vedendo un elevato numero di salvataggi, potrebbero essere incentivati ad utilizzarlo.

La gestione dei preferiti è un altro fattore importante che è stato considerato durante lo sviluppo. Essi possono essere visualizzati all'interno del profilo nella sezione "Favourites" e l'azione di visualizzare e trovare un sito preferito tra tutti i molteplici preferiti che un utilizzatore potrebbe avere è stato reso un

meccanismo semplice ed efficiente attraverso l'implementazione di filtri di ricerca. L'integrazione di questa funzionalità è inoltre affrontata nuovamente al paragrafo 4.3.4 dell'elaborato.

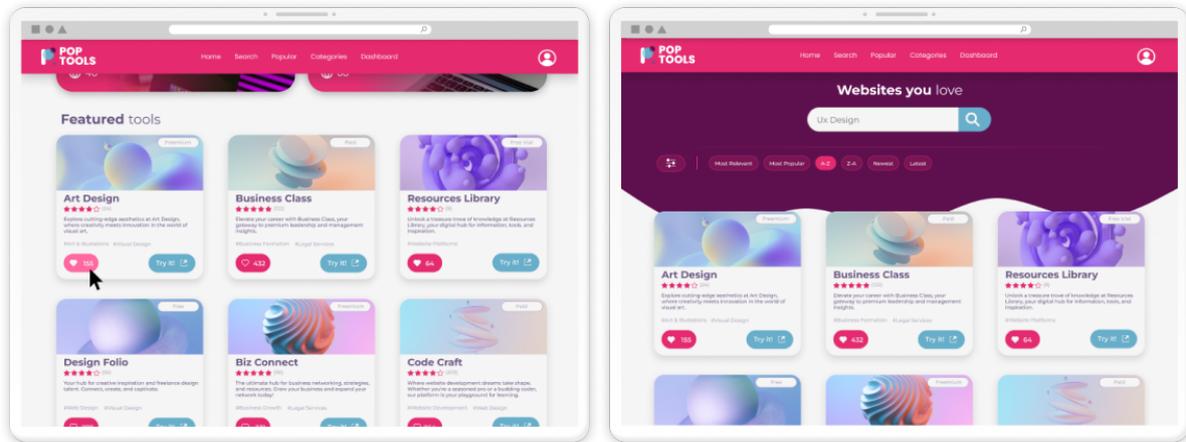


Figura 4: Schermata Homepage e Profile che mostrano la funzionalità di aggiunta ai Preferiti

### 3.3.3 Filtri di ricerca

I filtri di ricerca rappresentano infatti un altro elemento cruciale all'interno di *Pop Tools*. Anche se l'obiettivo delle directory odierne non è più fornire milioni di risultati come i motori di ricerca ma specializzarsi su determinate nicchie di utenti come professionisti e freelance del mondo digitale, avere la possibilità di filtrare ulteriormente i siti presenti sulla piattaforma può solo essere di aiuto per far ottenere risultati ancora più mirati e precisi agli utenti finali.

I filtri disponibili sulla piattaforma si trovano sia nella pagina di ricerca che nel profilo all'interno della sezione relativa ai preferiti e si dividono essenzialmente in due tipologie: filtri per categoria e filtri di ordinamento. La prima è indubbiamente la più utile in quanto permette all'utente di includere o escludere varie categorie all'interno della sua ricerca permettendogli di operare solamente con risultati che rispecchiano i filtri selezionati. Un utente potrebbe infatti voler ricercare tutti i tools che appartengono sia alla categoria *"Website Development"* che alla categoria *"Web & App Design"* così da affinare ulteriormente la quantità e qualità di risultati ottenuti.

I filtri di ordinamento al contrario non vanno ad escludere nessun risultato dalla ricerca ma, come da nome, vanno a riordinare i tools in base ad opzioni come l'ordine alfabetico, la data di pubblicazione o il punteggio che gli utenti hanno attribuito ai tools. Questa tipologia di filtri non fa altro che offrire una ulteriore personalizzazione ad un pubblico, al giorno d'oggi, sempre più esigente.

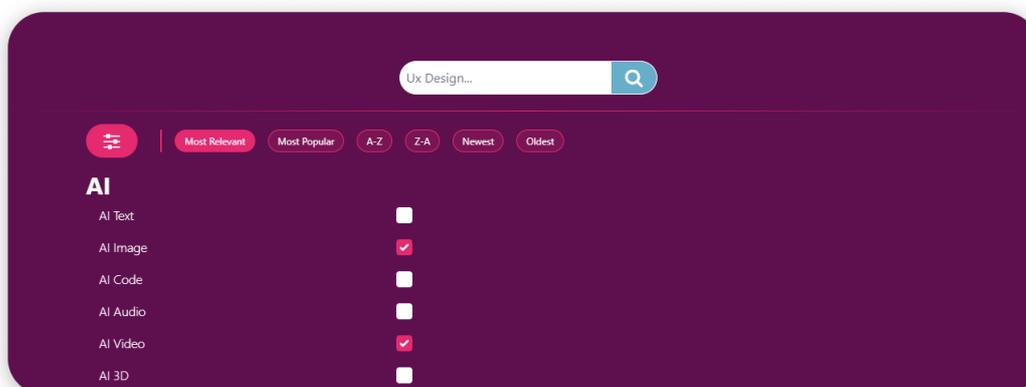


Figura 5: Filtri di ricerca

### 3.3.4 Partecipazione attiva e condivisione

Un altro elemento molto importante del sistema sono i meccanismi offerti per la partecipazione attiva e la condivisione di un nuovo sito web da parte della *community*. Nella piattaforma infatti, entrando all'interno delle card relative ai siti web, oltre a visualizzare le informazioni generali dei tools è stato reso disponibile per gli utenti un form in cui possono assegnare un punteggio da 1 a 5 stelle al servizio e dove possono scrivere una dettagliata recensione riguardo i loro pensieri ed esperienze durante l'utilizzo dello stesso.

Publicando la recensione essa diventa visualizzabile da tutti gli utenti e il *rating* conferito al sito web viene aggiunto agli altri per calcolare il punteggio medio finale, necessario a valutare se un determinato tool è apprezzato o meno dalla comunità. Il secondo metodo che *Pop Tools* offre ai suoi utenti per renderli parte della *community* è la possibilità di proporre ed inviare agli amministratori un nuovo potenziale sito web, tramite un form accessibile solamente da utenti loggati dal menu a tendina del profilo.

Una volta inviata la proposta, sarà poi compito della redazione stabilire se incorporare o meno il nuovo sito all'interno della directory aggiungendo tutte le informazioni necessarie. Si può affermare dunque che la partecipazione attiva e la condivisione definiscono l'essenza di *Pop Tools*, puntando a renderlo un ecosistema virtuale dinamico e collaborativo in cui ogni utente è fonte attiva del cambiamento e del progresso.

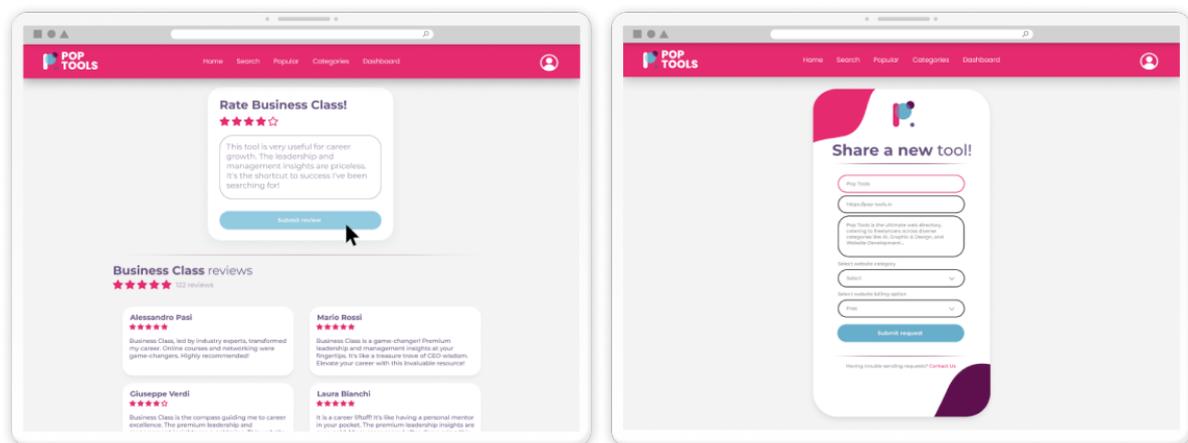


Figura 6: Schermata Tool e Tool Submit che mostrano le funzionalità di pubblicazione recensioni e invio proposte

### 3.3.5 Dashboard per la gestione e l'amministrazione

Infine vi è la Dashboard, che è il nucleo amministrativo della web directory ed offre agli amministratori un controllo completo sulla gestione e l'ottimizzazione di quest'ultima. Essa permette alla redazione di supervisionare ogni aspetto di *Pop Tools* garantendo una centralizzazione dei servizi amministrativi in un unico portale.

In esso un admin può per esempio visualizzare l'elenco di tutti i siti web presenti sulla piattaforma andando a modificarne i dati qualora ce ne fosse la necessità. Può andare a gestire la lista delle categorie e macrocategorie modificandole o inserendone di nuove in base alle possibili richieste della *community* rendendo la piattaforma dinamica e propensa all'evoluzione.

Infine vi è la parte relativa alla gestione delle richieste, in cui un redattore può muoversi all'interno della lista e decidere se approvare o rifiutare queste ultime dopo averle analizzate singolarmente in modo meticoloso. La Dashboard è senza alcun dubbio l'area della piattaforma più ricca di funzionalità e verrà descritta più accuratamente nel paragrafo 4.3.6.

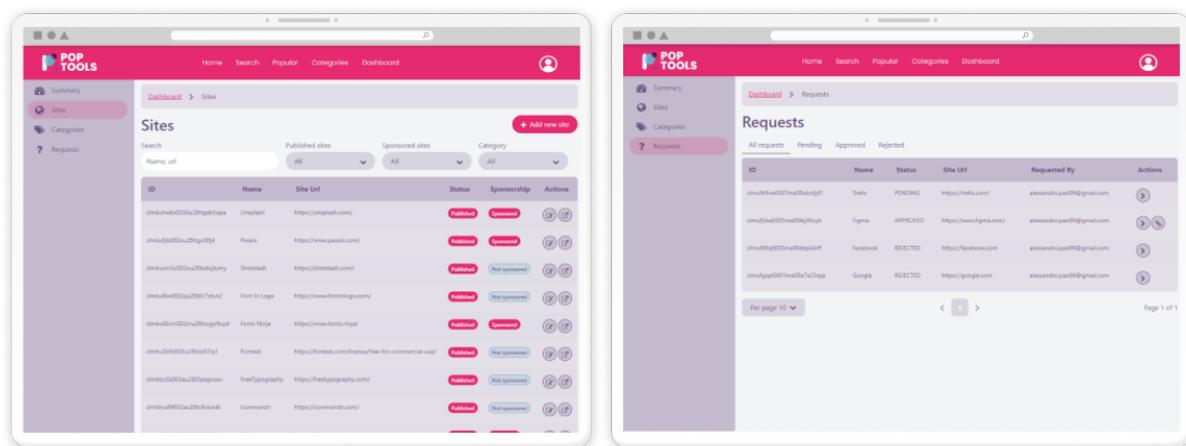


Figura 7: Schermata Dashboard che mostra le funzionalità di gestione siti web e proposte

### 3.4 Scenari d'uso

Per lo sviluppo e progettazione di un'applicazione web complessa come *Pop Tools* è necessario definire in fase progettuale gli scenari d'uso della web directory. Questi servono a delineare le diverse situazioni in cui i vari utenti possono interagire con il sistema, fornendo una panoramica completa dei comportamenti che potranno essere svolti all'interno della piattaforma.

#### 3.4.1 Attori

Un attore può essere definito come un'entità che scambia informazioni con il sistema fornendo in input dei dati o ricevendone in output dallo stesso. Gli attori fanno parte degli *stakeholders* coinvolti all'interno del progetto e servono a delineare un ruolo specifico che un utente o un'altra entità può assumere interagendo con il sistema attraverso il caso d'uso.

Nel contesto della web directory, identifichiamo tre attori principali:

Attore	Descrizione
Utente non Registrato	Questo è un utente che sta visitando la piattaforma senza aver effettuato l'accesso. Può visualizzare la Homepage, cercare siti web utilizzando la barra di ricerca e i filtri, ma non ha accesso alle funzionalità riservate agli utenti registrati.
Utente Registrato	Questo attore è un professionista, un freelance o un appassionato del digitale che ha creato un account sulla piattaforma. L'utente registrato può accedere alla web directory, cercare siti web, aggiungere siti ai preferiti, scrivere recensioni e inviare proposte per l'inserimento di nuovi siti.
Amministratore	L'amministratore è un utente con privilegi speciali che ha il compito di gestire la piattaforma. Può approvare o rifiutare le proposte degli utenti per l'inserimento di nuovi siti, modificare i dettagli dei siti esistenti, gestire le categorie e le macrocategorie, e monitorare l'attività globale sulla piattaforma.

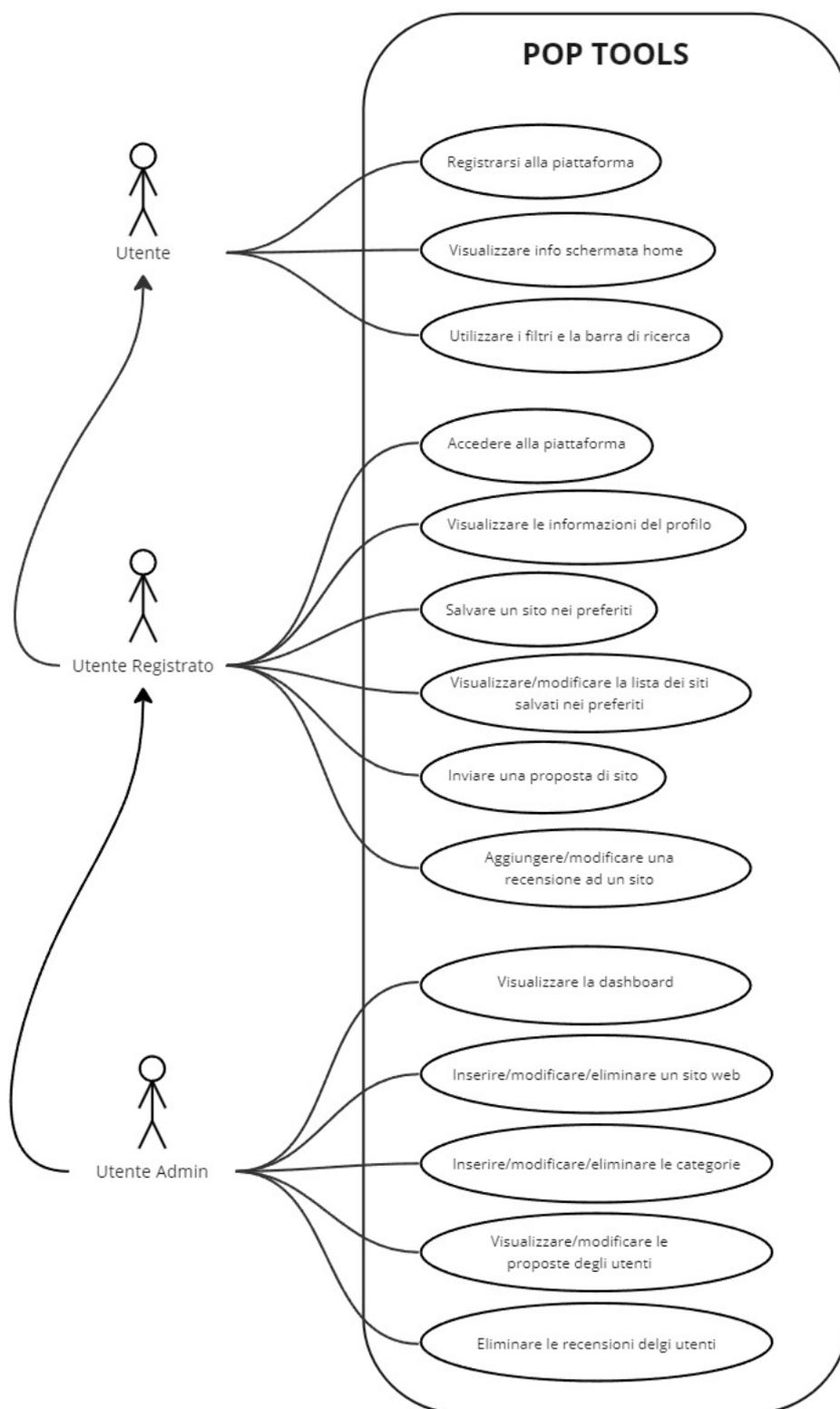
**Tabella 1:** Attori nel Sistema

#### 3.4.2 Casi d'uso

I casi d'uso sono uno strumento fondamentale per eseguire in maniera precisa e non ambigua, la raccolta dei requisiti per poter rilasciare un prodotto di qualità. Essi non sono altro che una descrizione dettagliata di come il sistema risponde a una specifica richiesta da parte di un attore.

I casi d'uso rappresentano quindi una sequenza di azioni o eventi che si verificano quando un utente interagisce con il sistema per raggiungere uno specifico obiettivo e sono utilizzati per modellare il comportamento del sistema dal punto di vista degli attori.

Per visualizzare tutti i possibili casi d'uso nella loro interezza viene sviluppato un diagramma grafico che descrive le diverse interazioni tra gli attori (utenti o sistemi esterni) e un sistema *software*. Esso viene utilizzato come mezzo di supporto e punto di partenza per la progettazione dal momento in cui servirà come mezzo di comunicazione tra gli *stakeholder*, inclusi sviluppatori, *designer* e clienti.



**Figura 8:** Diagramma dei casi d'uso della piattaforma Pop Tools

### 3.4.3 Casi d'uso rilevanti

Durante la fase di progettazione sono stati realizzati alcuni casi d'uso per descrivere le funzionalità principali della piattaforma, essi sono stati riportati di seguito:

<b>Nome caso d'uso</b>	Ricerca e visualizzazione dei siti web
<b>Attori coinvolti</b>	Utente Registrato, Utente non Registrato
<b>Precondizione</b>	L'utente è autenticato o sta visitando la homepage come visitatore non registrato.
<b>Scenario principale</b>	<ol style="list-style-type: none"> <li>1. L'utente accede alla homepage della web directory.</li> <li>2. L'utente utilizza la barra di ricerca nella homepage per inserire parole chiave o frasi relative al sito web che sta cercando.</li> <li>3. Il sistema esegue una ricerca tra i siti web presenti nella directory e restituisce una lista di risultati corrispondenti alla query di ricerca dell'utente.</li> <li>4. L'utente esamina i risultati della ricerca e visualizza le informazioni dei siti web.</li> </ol>
<b>Scenario alternativo</b>	<ol style="list-style-type: none"> <li>3.a. Il sistema esegue una ricerca tra i siti web presenti nella directory ma non trova risultati pertinenti.</li> <li>4.a. Il sistema mostra un messaggio che invita l'utente a modificare i parametri di ricerca e riprovare.</li> </ol>
<b>Postcondizione</b>	L'utente ha eseguito una ricerca e visualizzato le informazioni generali dei siti web.

**Tabella 2:** Caso d'Uso 1: Ricerca e visualizzazione dei siti web

<b>Nome caso d'uso</b>	Invio proposta di un nuovo sito Web
<b>Attori coinvolti</b>	Utente Registrato
<b>Precondizione</b>	L'utente è autenticato e ha accesso alla funzionalità di proposta di un nuovo sito.
<b>Scenario principale</b>	<ol style="list-style-type: none"> <li>1. L'utente accede alla pagina per inviare la proposta di un nuovo sito.</li> <li>2. L'utente compila il form inserendo i dettagli del sito web da proporre, come il nome, l'URL, una breve descrizione, la categoria di appartenenza e il costo del servizio.</li> <li>3. L'utente clicca sul pulsante di invio.</li> <li>4. Il sistema registra la proposta e mostra all'utente un messaggio di successo.</li> </ol>
<b>Scenario alternativo</b>	<ol style="list-style-type: none"> <li>4.a. Alcuni campi del form non sono stati compilati e il sistema mostra un messaggio di errore.</li> <li>4.b. L'invio della richiesta al sistema non va a buon fine e viene mostrato un messaggio di errore.</li> </ol>
<b>Postcondizione</b>	L'utente ha proposto l'inserimento di un nuovo sito web.

**Tabella 3:** Caso d'Uso 2: Invio proposta di un nuovo sito web

<b>Nome caso d'uso</b>	Gestione dei preferiti
<b>Attori coinvolti</b>	Utente Registrato
<b>Precondizione</b>	L'utente è autenticato e ha aggiunto almeno un sito web ai preferiti.
<b>Scenario principale</b>	<ol style="list-style-type: none"> <li>1. L'utente accede al proprio profilo utente dalla homepage.</li> <li>2. Il sistema mostra una lista dei siti web precedentemente aggiunti ai preferiti dall'utente.</li> <li>3. L'utente visualizza i risultati e decide di selezionare un filtro di ricerca per trovare tra i suoi preferiti solamente i siti web appartenenti ad una certa categoria.</li> <li>4. Il sistema esegue una ricerca tra i siti web preferiti dall'utente e restituisce una lista di risultati aventi la categoria selezionata.</li> </ol>
<b>Scenario alternativo</b>	<ol style="list-style-type: none"> <li>4.a. Il sistema esegue una ricerca tra i siti web preferiti dall'utente utilizzando il filtro selezionati ma non trova risultati pertinenti.</li> <li>5.a. Il sistema mostra un messaggio che invita l'utente a modificare i parametri di ricerca e riprovare.</li> </ol>
<b>Postcondizione</b>	L'utente ha gestito con successo i siti web nei propri preferiti.

**Tabella 4:** Caso d'Uso 3: Gestione dei preferiti

<b>Nome caso d'uso</b>	Inserimento di un nuovo sito web
<b>Attori coinvolti</b>	Utente Amministratore
<b>Precondizione</b>	L'utente è autenticato e ha accesso alla Dashboard di amministrazione.
<b>Scenario principale</b>	<ol style="list-style-type: none"> <li>1. L'utente accede alla Dashboard dal menu.</li> <li>2. Il sistema carica i dati della Dashboard e mostra la pagina principale contenente alcuni dati come il numero di utenti registrati, il numero di siti presenti e il numero di richieste inviate dagli utenti.</li> <li>3. L'utente clicca sul pulsante Sites dal menu della Dashboard.</li> <li>4. Il sistema mostra una pagina contenente una tabella con la lista di tutti i siti web presenti nella directory.</li> <li>5. L'utente clicca sul pulsante per l'aggiunta di un nuovo sito.</li> <li>6. Il sistema carica il form nel quale l'utente deve inserire tutte le informazioni relative al nuovo sito.</li> <li>7. L'utente clicca sul pulsante per il salvataggio del nuovo sito.</li> <li>8. Il sistema carica il nuovo sito nella directory e mostra un messaggio di successo.</li> </ol>
<b>Scenario alternativo</b>	<ol style="list-style-type: none"> <li>4.a. Alcuni campi del form non sono stati compilati e il sistema mostra un messaggio di errore.</li> <li>4.b. L'invio del nuovo sito al sistema non va a buon fine e viene mostrato un messaggio di errore.</li> </ol>
<b>Postcondizione</b>	L'utente Amministratore ha inserito un nuovo sito web nella directory.

**Tabella 5:** Caso d'Uso 4: Inserimento di un nuovo sito web

## 4 Progettazione dell'interfaccia utente

La progettazione dell'interfaccia utente (*UI*) è un fattore molto importante da prendere in considerazione quando ci si avvicina allo sviluppo di un'applicazione o un servizio digitale come la web directory *Pop Tools*. La *User Interface* è ciò che connette l'utente al servizio finale e, per questo motivo, deve essere progettata attentamente e con cura per fornire all'utilizzatore un'esperienza ottimale. Nelle prossime sezioni verranno elencate e motivate le scelte svolte in questo contesto fornendo una panoramica completa dell'interfaccia.

### 4.1 Layout e design dell'interfaccia

Il layout e il design di un'applicazione web svolgono un ruolo cruciale per l'esperienza utente. Esse danno un primo impatto all'utente influenzando significativamente la sua percezione rispetto alla qualità del servizio.

Il layout e il design però, vanno ben oltre un fattore estetico, sebbene questa sia molto importante per contribuire positivamente all'aspetto visivo, l'obiettivo principale di un'interfaccia funzionale è garantire che l'utente non abbia intoppi a comunicare con il sistema così da rendere il suo utilizzo piacevole e intuitivo.

#### 4.1.1 Disposizione degli elementi

Un elemento fondamentale quando si parla di *UI* è la disposizione degli elementi sulla pagina come bottoni, menu, campi di input e contenuti che devono essere ben pianificati per evitare che risultino invadenti e si oscurino a vicenda.

Nel caso di *Pop Tools* per esempio, gli elementi che hanno richiesto sicuramente più cura sono le *card* relative ai siti web. Questi piccoli elementi infatti sono i componenti presenti nel maggior numero all'interno dell'applicativo e devono essere presentati all'utente in modo armonioso, mostrandogli solamente le informazioni essenziali e più importanti senza tralasciare nulla. L'obiettivo di queste *card* è anche quello di attirare l'attenzione dell'utente mostrandogli una piccola immagine riassuntiva di ciò che il sito web ha da offrire.

Come si evince dalla grafica sottostante, le *card* contengono numerose informazioni utili come il nome della piattaforma, il punteggio e il numero di recensioni attribuite dagli utenti, una breve descrizione dell'applicativo, le categorie alle quali fa riferimento e i vari piani tariffari che il servizio può offrire. Inoltre, alla base delle *card*, sono presenti due pulsanti con funzionalità distinte. Il primo dal colore fucsia mostra il numero di utenti che hanno inserito la piattaforma all'interno dei preferiti e permette all'user stesso di aggiungerlo agli elementi salvati, al contrario il pulsante azzurro, come indica il testo presente su di esso, rimanda al sito web dell'applicativo stesso.

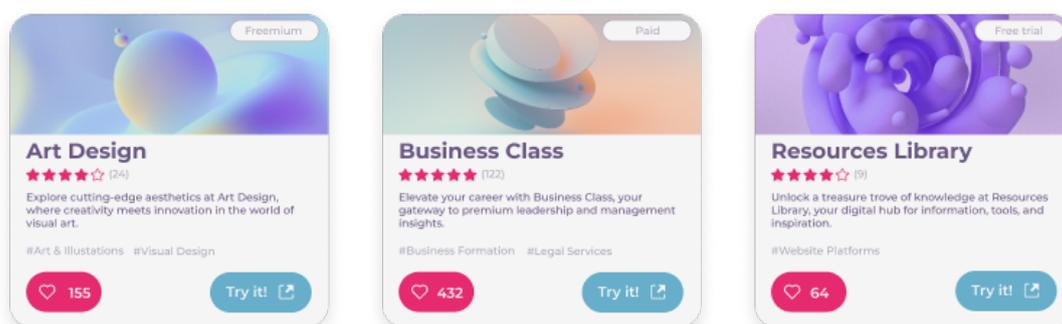


Figura 9: Componenti *card* contenenti le informazioni dei siti web

#### 4.1.2 Gerarchia dei componenti

Un'altro aspetto utilizzato nella progettazione delle *card* e del sito in generale è la gerarchia visuale. Con questo concetto si intende l'uso corretto di dimensioni, colori, contrasti, posizioni e caratteri per indicare l'importanza relativa ai differenti elementi, testi e informazioni presenti sulla pagina.

Ad esempio, ritornando all'elemento delle *card* notiamo che le componenti che devono essere più evidenti agli occhi degli utenti sono caratterizzate da font più spessi, colori vivaci e dimensioni più grandi,

mentre tutti quelli meno rilevanti e che servono per dare informazioni aggiuntive sono rappresentati in modo più discreto.

Questa gerarchia visuale serve quindi a guidare l'utente attraverso l'applicazione, aiutandolo a concentrarsi meglio sugli elementi chiave senza essere sopraffatto da un eccesso di informazioni fuorvianti.

#### 4.1.3 Navigazione

Un'altra componente essenziale e non banale presente sulla piattaforma è la navigazione. L'interfaccia deve fornire un metodo chiaro e intuitivo per l'utente di spostarsi tra le diverse sezioni e pagine dell'applicazione.

Ciò può essere svolto tramite una *navbar*, un menu a tendina, pulsanti di accesso rapido o persino scorciatoie da tastiera. All'interno del progetto sono state implementate tutte le possibilità appena descritte così da rendere la navigazione efficace e garantire che gli utenti riescano ad accedere in modo semplice, rapido e intuitivo a tutte le aree dell'applicativo.

È stata implementata una *navbar* superiore che permette di navigare tutte le pagine principali in comodità, una volta eseguito l'accesso inoltre cliccando sull'icona utente della barra di navigazione, si apre il menu a tendina contenente tutte le funzioni relative al profilo di un utente, come la visualizzazione dei suoi siti web preferiti o l'accesso al form per l'invio di una proposta per un nuovo sito web. Inoltre cliccando sulle categorie presenti all'interno delle *card* dei siti web sarà possibile svolgere una ricerca automatica per categoria che riporterà l'utente alla pagina *Search*. Grazie a funzionalità simili è possibile conferire più dinamicità all'applicativo rendendo una navigazione ben strutturata essenziale per l'utente.

#### 4.1.4 Colori e immagini

Infine come non menzionare l'utilizzo di immagini e colori che rendono il *design* di *Pop Tools* unico e distintivo.

La *palette* è un elemento molto potente nel *design* e, da sola, può influenzare le scelte e le sensazioni dell'utente. Oltre all'aspetto estetico, i colori vengono utilizzati per una moltitudine di ragioni, per esempio evidenziando le azioni che l'utilizzatore sta compiendo facendo variare per esempio il colore di un pulsante appena premuto rendendo l'esperienza più interattiva, o utilizzando colori differenti per evidenziare informazioni più importanti che devono risaltare all'interno del contesto.

Gli elementi grafici e le immagini invece servono principalmente per arricchire l'aspetto dell'interfaccia e comunicare concetti in modo più visuale ed efficace. Le componenti grafiche come le icone, d'altra parte, possono essere utilizzate, invece del testo, per rappresentare funzioni ben precise che l'utente, all'interno del contesto digitale attuale, sa riconoscere autonomamente.

È importante però ricordare che colori e immagini non vanno abusati, ma utilizzati in modo responsabile e pertinente al contesto per non creare confusione all'occhio dell'utente, ma al contrario, guidarlo visivamente verso le varie funzionalità della piattaforma.

## 4.2 *User Experience* e usabilità

L'aspetto relativo all'ottimizzazione della User Experience oltre ad essere un tema strettamente collegato al paragrafo precedente è anche di estrema rilevanza all'interno del panorama digitale attuale. Questi concetti sono fondamentali per il successo di qualsiasi piattaforma e la web directory *Pop Tools* non fa eccezione.

Sviluppare un'interfaccia *user-friendly* è stato uno degli obiettivi principali sia in fase di progettazione che implementazione vera e propria. Questa grande attenzione è dovuta dal fatto che statisticamente gli utenti trascorrono più tempo su piattaforme che offrono una navigazione ed esperienza generale più agevole e senza intoppi.

Per questo motivo è stato dedicato molto tempo in una fase di ricerca e studio così da creare un ambiente che mettesse al centro le esigenze e le priorità degli utenti.

### 4.2.1 *Design System*

Per ottimizzare al meglio l'usabilità del sistema si è deciso di utilizzare un design system che, in ambito informatico, non è altro che un insieme di linee guida, componenti, stili e principi di *design* che garantiscono una coerenza visiva e funzionale all'interno dell'ecosistema dell'applicazione. Non è altro che una serie di regole che definiscono come deve comportarsi e apparire l'interfaccia utente di un'applicazione.

Nel progetto *Pop Tools*, è stato utilizzato "*Skeleton*"[35], un design system *open source* basato su *Svelte*[38] e *Tailwind CSS*[39] che offre i vantaggi seguenti per quanto riguarda la progettazione dell'interfaccia utente:

- **Open Source:** Skeleton è un software completamente gratuito e open source senza piani o funzionalità a pagamento, il che lo rendono ottimo per sviluppare qualsiasi tipo di progetto.
- **Community:** Essendo un progetto open source Skeleton è costantemente aggiornato da una community di talentuosi contributori e appassionati che conferiscono un valore aggiunto al servizio.
- **Temi personalizzati:** Grazie ai molteplici temi già presenti nella piattaforma e al generatore offerto dal servizio, l'utente è in grado di personalizzare in modo autonomo e intuitivo il design system per adattarlo alle proprie esigenze.
- **Design Token:** Skeleton utilizza un semplice ma potente sistema di token progettuali utili ad implementare le impostazioni del tema negli elementi e nei componenti. Questi assegnano dei valori predefiniti alla maggior parte degli elementi e dei componenti, il che significa che si adattano automaticamente al tema e necessitano di una configurazione minima.

### 4.2.2 *Accessibilità*

Un altro tema fondamentale affrontato durante la progettazione e lo sviluppo dell'usabilità è l'accessibilità in quanto strettamente correlata all'esperienza dell'utente. L'obiettivo principale del servizio è offrire un'esperienza utente accessibile a tutti, indipendentemente dalle loro capacità fisiche o cognitive.

Durante il processo iniziale nel quale si è approcciata l'accessibilità sono state seguite le linee guida *WCAG (Web Content Accessibility Guidelines)*[1] essendo uno standard riconosciuto a livello internazionale per l'implementazione di questa nel web. Queste linee guida hanno fornito una base solida per identificare e risolvere diversi ostacoli che utenti con disabilità potrebbero incontrare durante l'utilizzo della piattaforma *Pop Tools*.

Uno dei principali aspetti dell'accessibilità riguarda la navigazione che deve permettere l'accesso a tutte le aree ed elementi interattivi dell'applicativo tramite l'utilizzo della sola tastiera. Ciò consente agli utenti che non possono utilizzare un mouse di riuscire comunque ad interagire nel modo migliore con il servizio garantendo un'esperienza fluida e senza impedimenti.

Per permettere queste funzionalità, il codice HTML è stato progettato in modo coerente e logico, utilizzando nel modo appropriato i vari elementi come paragrafi, link, liste e pulsanti che permettono una navigazione da tastiera utilizzando per esempio il pulsante "*tab*" per muoversi tra gli elementi e "*barra spaziatrice*" ed "*invio*" per selezionare elementi o premere su link e pulsanti. In questo modo qualsiasi utente potrà svolgere una ricerca, inserire un elemento tra i preferiti o recensire un sito web.

### 4.2.3 *Responsive design*

L'implementazione di un design responsive all'interno del progetto è stata una componente cruciale per garantire un'esperienza utente ottimale su qualsiasi dispositivo come desktop, tablet e mobile. Questo aspetto ha assunto un'enorme rilevanza all'interno del panorama digitale attuale dove gli utenti possono accedere ad un servizio online attraverso una moltitudine di dispositivi differenti aventi caratteristiche, dimensioni e risoluzioni da rispettare in fase di progettazione.

Un design responsive permette ad un sito web di adattarsi completamente e rispondere in modo efficiente e reattivo alle dimensioni dello schermo del dispositivo sul quale sta venendo visualizzato facendo variare il proprio *layout* e la propria struttura in base al dispositivo dell'utente.

Nel contesto del progetto *Pop Tools*, l'implementazione di un design responsive è stata una priorità sin dalle prime fasi di sviluppo in cui è stato scelto di sviluppare l'applicativo utilizzando un approccio *mobile-first* senza trascurare però le visualizzazioni *desktop* e *tablet* anch'esse ottimizzate.

Questa scelta progettuale è data dall'idea che la web directory vuole essere uno strumento di scoperta consultabile in ogni momento della giornata in modo rapido ed efficiente indipendentemente da dove si trovi l'utente consentendogli di salvare i tools ricercati comodamente anche da mobile per poi approfondirli e utilizzarli in un secondo momento tramite desktop.

Uno dei principali vantaggi di questo approccio *mobile-first* è che promuove la semplicità e la chiarezza dell'interfaccia utente concentrandosi sulle funzionalità essenziali per garantire che siano facilmente accessibili senza sovraccaricare lo schermo con elementi superflui o complicati. Questo ha contribuito notevolmente a migliorarne l'usabilità e la facilità di utilizzo garantendone una più semplice scalabilità per dispositivi *desktop* e *tablet*.

### 4.3 Schermate principali

Nell'ambito della progettazione dell'interfaccia utente del progetto, la scelta delle pagine da sviluppare è stato un argomento molto importante in quanto l'obiettivo iniziale, era quello di esprimere e fornire agli utenti solamente le informazioni utili e fondamentali che potessero essergli necessarie all'interno della web directory.

Sono state quindi sviluppate solamente le pagine ritenute utili e rilevanti ai fini di garantire un'esperienza ottimale sia agli utenti registrati che agli amministratori. Di seguito verranno descritte la struttura e le funzionalità delle schermate più importanti della directory.

#### 4.3.1 Homepage

La Homepage rappresenta la pagina iniziale mostrata agli utilizzatori dell'applicativo ed è visualizzabile da tutte le tipologie di utente.

Essa è suddivisa in tre sezioni principali:

- **Hero:** Questa sezione contiene la barra di ricerca utile agli utenti per navigare risorse digitali specifiche. Gli utenti possono inserire parole chiave, categorie o argomenti di loro interesse per ottenere in maniera rapida risultati di loro interesse.
- **Sezione categorie:** Essa mostra un carosello contenente le cards relative alle macro categorie presenti nel sistema. Ciascuna di esse si differenzia grazie all'immagine di sfondo, al titolo e alla breve descrizione di ciò che rappresenta. Questa sezione offre agli utenti un punto di partenza immediato per esplorare le risorse digitali disponibili senza la necessità di eseguire una ricerca specifica.
- **Sezione tools:** Contiene una lista paginata di siti web ordinati in base al punteggio conferito dagli utenti. Questi componenti sono l'elemento più importante del progetto e contengono tutte le informazioni rilevanti di un sito web come descritto al paragrafo 4.1.1.

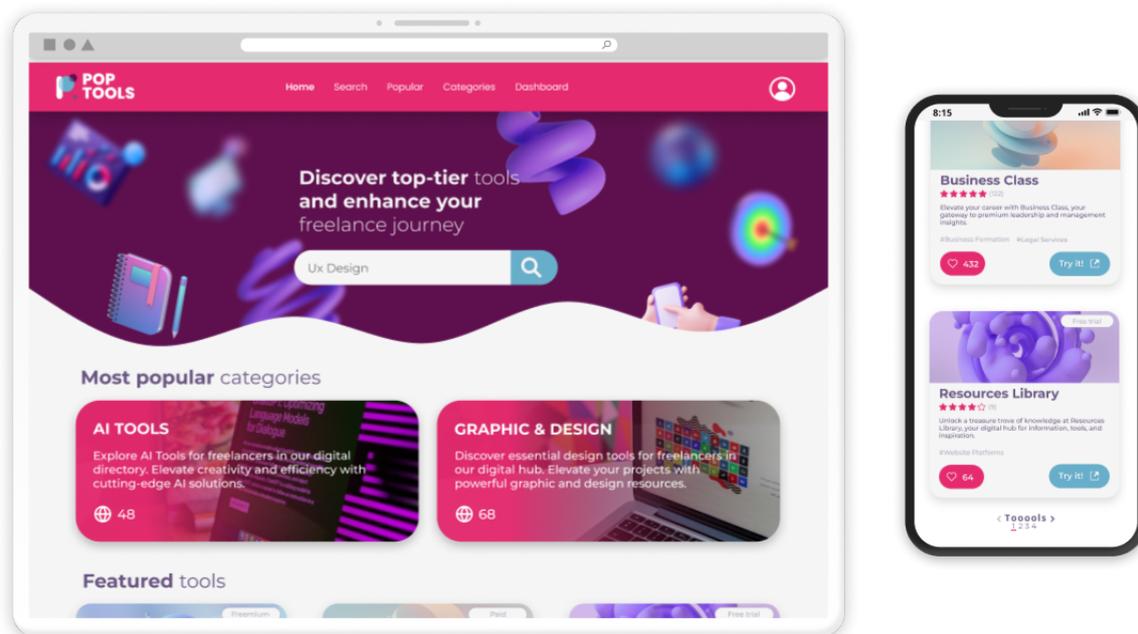


Figura 10: Schermata Homepage visualizzata su Desktop e Mobile

### 4.3.2 Search

La pagina di ricerca è raggiungibile tramite la barra presente in Homepage o cliccando sulla voce "Search" dal menù ed è visualizzabile da tutte le tipologie di utente.

Questa schermata implementa il filtro, un componente molto rilevante per la web directory che è stato descritto in maniera approfondita nel paragrafo seguente 3.3.3.

Grazie a questi filtri avanzati gli utenti possono raffinare la loro ricerca in base a criteri specifici, garantendo che possano trovare esattamente ciò che stanno cercando.

Un utente può quindi ricercare una keyword utilizzando la barra di ricerca e filtrare il risultato selezionando per esempio due categorie distinte così da ottenere solo tools che le contengono entrambe. Inoltre può ordinare i risultati come più ritiene opportuno, sia alfabeticamente, che per numero di voti e recensioni o ponendo i siti web inseriti più recentemente per primi o ultimi.

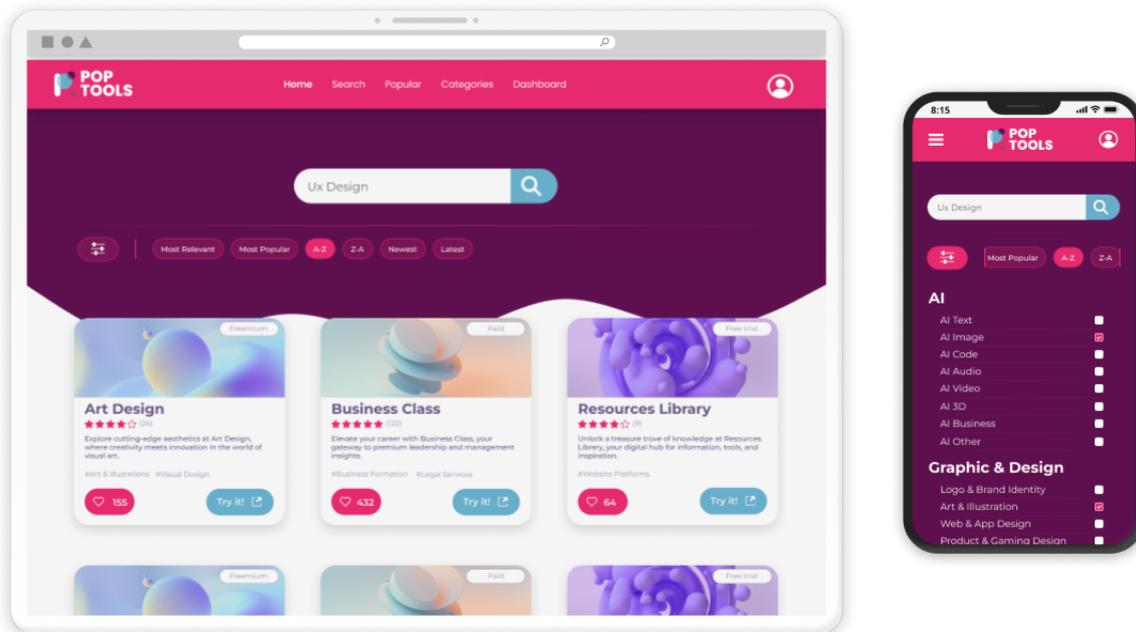


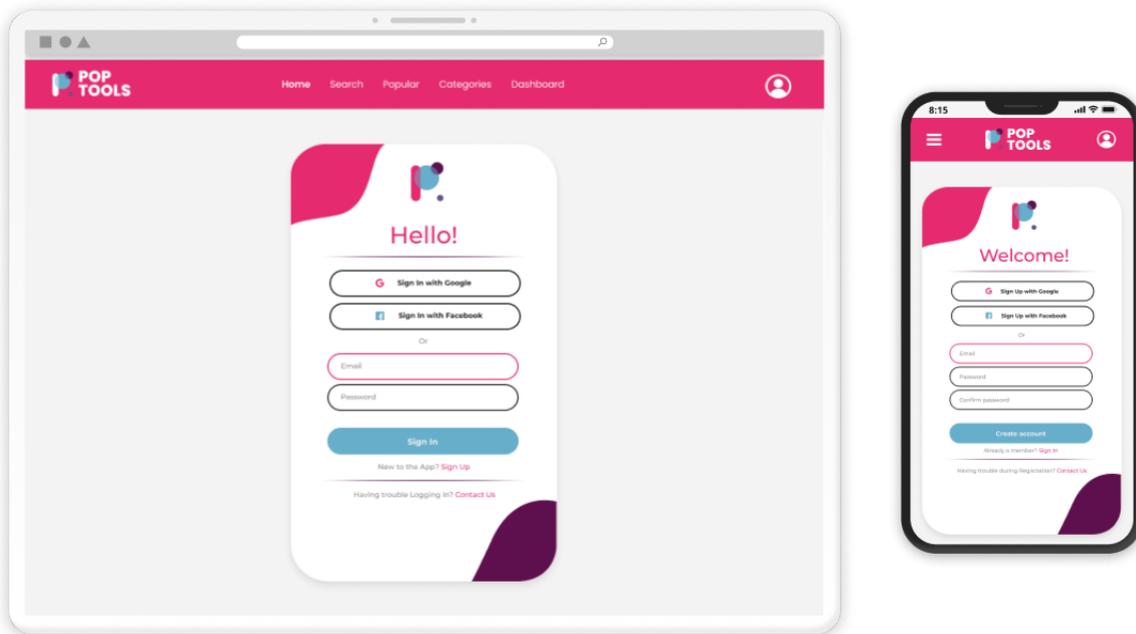
Figura 11: Schermata Search visualizzata su Desktop e Mobile

### 4.3.3 Login & Registration

Le pagine di Login e Registrazione sono raggiungibili tramite il collegamento presente nella *navbar* e sono accessibili solamente da utenti non ancora loggati.

Un utente loggato a differenza di un visitatore infatti può inserire un tool tra i preferiti che può visualizzare nella sezione “*Favourites*” del proprio profilo, può accedere alle pagine singole dei siti web che contengono tutte le informazioni di essi e permettono agli utenti di commentarli e votarli, può inviare agli amministratori una nuova proposta per l’inserimento di un nuovo tool e permette ai redattori di accedere alla *Dashboard* riservata.

In più l’accesso alla piattaforma può essere eseguito anche tramite provider come *Google e Facebook* che ne semplificano e velocizzano il meccanismo facendo risparmiare tempo agli utilizzatori che non saranno obbligati a creare un vero e proprio account.



**Figura 12:** Schermate Login e Registrazione visualizzate su *Desktop e Mobile*

#### 4.3.4 Profile & Site Submit

Le pagine Profile e Site Submit sono accessibili tramite il popup che viene mostrato cliccando sull'icona dell'utente presente sulla *navbar* e sono accessibili solamente da utenti loggati.

Il Profilo utente è simile alla pagina di ricerca ma con le differenze sostanziali che la hero contiene anche le informazioni relative all'account dell'utente e nella sezione relativa ai tools, sono presenti solamente quelli che l'utente ha inserito tra i suoi preferiti.

Tutti i filtri per categoria e di ordinamento infatti avranno effetto sempre e solo sui siti web preferiti da esso per permettergli di ricercarli e trovarli con semplicità anche qualora ne avesse un numero elevato. La sezione "*Favourites*" permette dunque all'utilizzatore di andare a gestire con semplicità e in modo intuitivo la propria lista preferiti.

La pagina per l'invio di una nuova proposta invece è una delle funzionalità chiave della web directory perché offre agli utenti la possibilità di prendere parte al vero ecosistema della piattaforma diventando un vero e proprio ingranaggio fondamentale al suo sviluppo.

Gli utenti registrati una volta eseguito l'accesso alla pagina, possono compilare i campi del form con le informazioni del sito web che vorrebbero venisse pubblicato come il nome, l'URL, una breve descrizione, il piano tariffario e la macrocategoria di riferimento per poi inviare la richiesta che sarà visualizzabile dagli amministratori nell'apposita sezione della loro *Dashboard*.

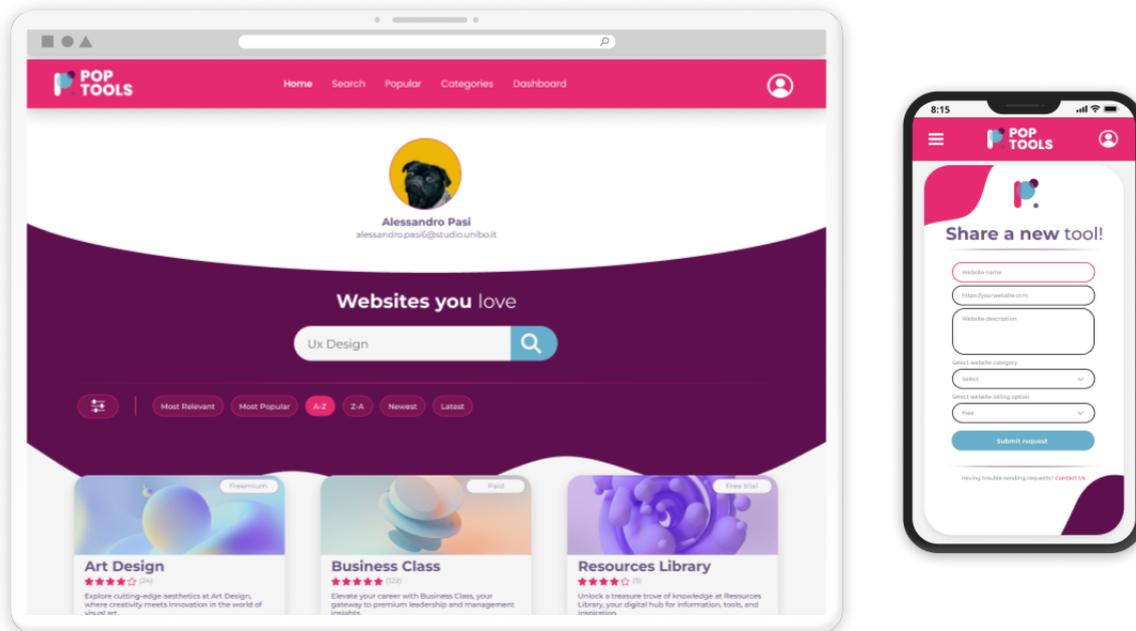


Figura 13: Schermate Profile e Site Submit visualizzate su *Desktop e Mobile*

#### 4.3.5 Tool

La pagina relativa al singolo tool è raggiungibile cliccando sulla *card* di un sito web ed è accessibile solamente da un utente loggato.

Questa pagina è fondamentale all'interno del progetto aggiungendo diverse funzionalità all'applicativo. Qui è infatti possibile visualizzare tutte le informazioni appartenenti ad un sito web compresa una dettagliata descrizione di esso ed un video dimostrativo riguardante il suo utilizzo se disponibile.

Una delle caratteristiche più interattive di questa pagina è la sezione delle recensioni degli utenti. Essi possono infatti leggere le reviews scritte da altri membri della *community* per ottenere una prospettiva più dettagliata sull'esperienza di utilizzo del tool.

Possono anche scrivere, pubblicare e modificare le proprie recensioni, assegnando un punteggio da 1 a 5 stelle e condividendo i loro commenti e *feedback*.

Inoltre è disponibile una sezione finale contenente i siti web correlati per spingere l'utente alla scoperta di nuove possibilità inerenti ai suoi interessi.

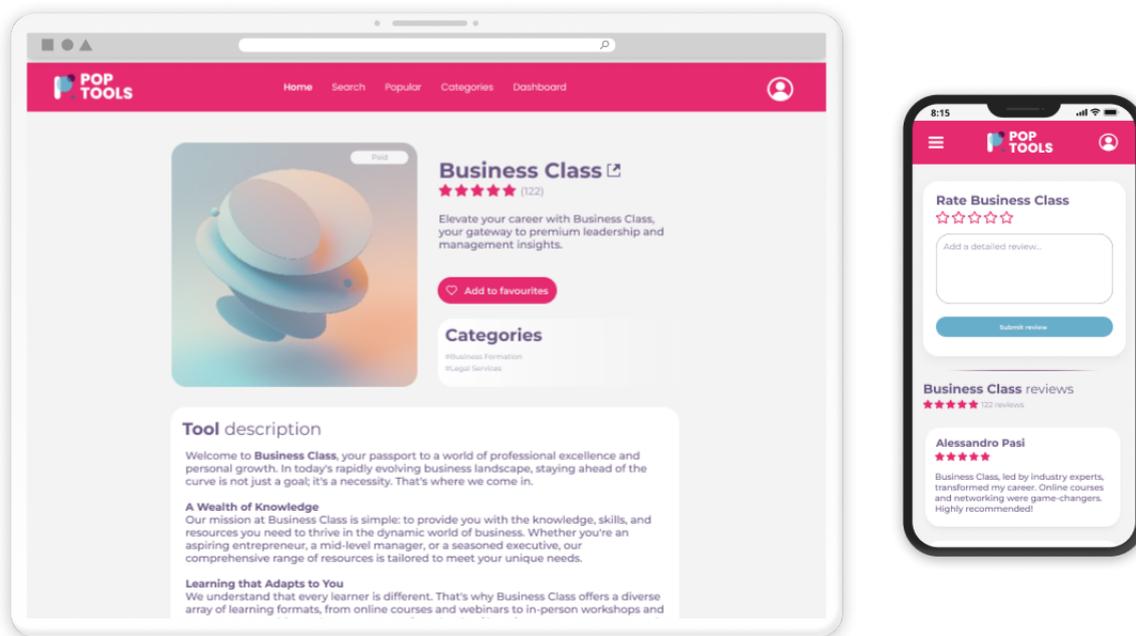


Figura 14: Schermata Tool visualizzata su *Desktop e Mobile*

#### 4.3.6 Dashboard

La pagina Dashboard è un'area riservata agli amministratori della piattaforma ed è accessibile tramite il link presente sulla *navbar*.

Al suo interno è possibile monitorare e gestire tutti gli aspetti della web directory ed è stata progettata per offrire un controllo completo e intuitivo sull'intera piattaforma.

La Dashboard è suddivisa a sua volta in 4 schermate principali:

- **Summary:** La pagina della dashboard inizia con una panoramica completa delle statistiche chiave della piattaforma. Gli amministratori possono vedere il numero totale di siti web presenti nella web directory, il numero di categorie e macrocategorie, e una visualizzazione grafica del numero di siti per categoria. Queste informazioni consentono agli amministratori di avere una visione immediata delle dimensioni e della distribuzione dei contenuti nella piattaforma.
- **Sites:** La sezione dei siti web consente agli amministratori di visualizzare tutti i siti web presenti nella directory. Questa tabella fornisce una visione completa delle informazioni sui siti web, compresi il nome, l'URL, la descrizione breve, la categoria di riferimento e il numero di utenti che lo hanno aggiunto ai preferiti. Gli amministratori possono eseguire una serie di azioni, tra cui la modifica delle informazioni del sito, la pubblicazione o l'archiviazione dei tools e la sponsorizzazione di questi per evidenziarli nella directory.
- **Categories:** La sezione delle categorie e delle macrocategorie consente agli amministratori di gestire l'organizzazione della web directory. Possono aggiungere nuove macrocategorie e categorie, modificarne le descrizioni e eliminarle se necessario. Questa flessibilità consente di adattare la struttura della directory alle esigenze in evoluzione degli utenti.
- **Requests:** La sezione delle richieste è un elemento critico della dashboard. Qui gli amministratori possono visualizzare tutte le richieste inviate dagli utenti per l'inclusione di nuovi siti web nella directory. Queste richieste possono trovarsi in uno stato *"Pending"* se non sono ancora state gestite dagli amministratori, *"Approved"* se sono state accettate, o *"Rejected"* se sono state rifiutate. Gli amministratori possono aprire ogni richiesta per esaminarla, modificarla e prendere una decisione sulla sua inclusione nella directory.

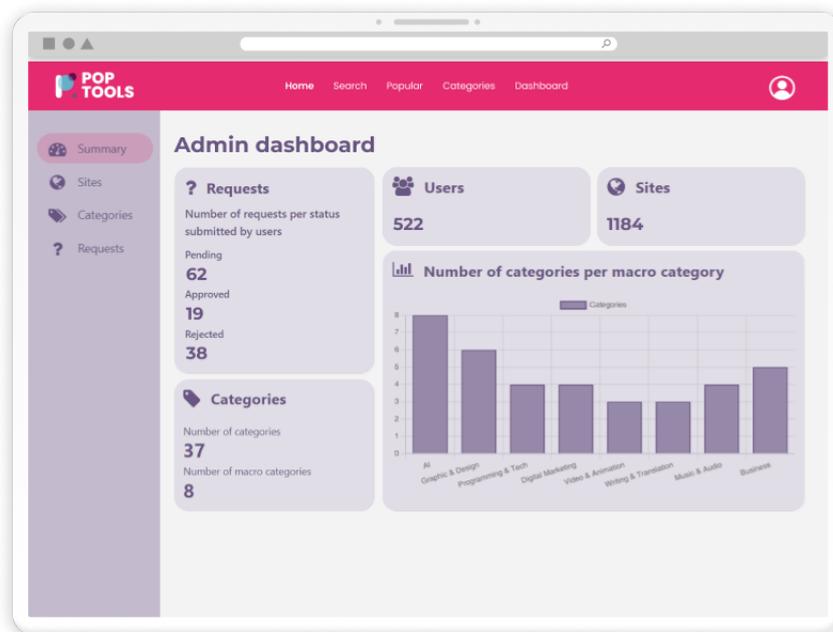


Figura 15: Schermata Dashboard visualizzata su Desktop

## 4.4 *Testing*

Il testing dell'interfaccia utente, dopo lo sviluppo, è una delle fasi più importanti per produrre un software di valore in quanto rappresenta il punto cruciale in cui la teoria si traduce in pratica. In questa fase infatti la piattaforma viene messa alla prova dai primi utenti che hanno il compito di testare l'interfaccia e le varie funzionalità.

### 4.4.1 **Obiettivi e importanza**

Prima di elencare le varie metodologie di testing è importante comprendere il valore intrinseco che questo processo offre all'applicazione web. La *UI* è infatti l'ambiente in cui gli utenti interagiscono col servizio stesso esplorando le funzionalità e svolgendo azioni specifiche, pertanto l'esperienza utente generale è fortemente influenzata dalla qualità di essa.

Il testing della piattaforma ha due obiettivi fondamentali. Il primo mira a identificare precocemente eventuali problemi di usabilità o errori che potrebbero ostacolare gli utenti durante l'utilizzo dell'applicativo per poi risolverli. Il secondo è quello di raccogliere *feedback* dagli utilizzatori così da scoprire eventuali possibili miglioramenti o modifiche per poter migliorare il design dell'interfaccia e le varie funzionalità del sistema.

In questo contesto infatti l'usabilità è un concetto chiave. Un'interfaccia utente poco intuitiva porta gli utilizzatori ad avere un'esperienza confusionaria e poco gratificante che può portarli ad abbandonare il servizio a favore di alternative più *user-friendly*. Al contrario, un'interfaccia ben progettata e testata può aumentare la soddisfazione degli utenti, fidelizzandoli e facendo aumentare il valore intrinseco della piattaforma.

### 4.4.2 **Testing SUS**

Il *System Usability Scale*, abbreviato in SUS, è uno strumento ampiamente utilizzato per valutare l'usabilità di un sistema, un'applicazione o un'interfaccia utente. È stato sviluppato da *John Brooke* nel 1986 ed è diventato uno degli standard dell'industria per la valutazione dell'usabilità. L'obiettivo principale del SUS è misurare la percezione degli utenti sulla facilità d'uso di un sistema in un contesto realistico.

Il metodo SUS si basa su un breve questionario standardizzato composto da 10 affermazioni riguardanti la *UX* del sistema. Gli utenti che partecipano al test leggono ciascuna affermazione e assegnano un punteggio da 1 a 5, dove 1 indica "fortemente in disaccordo" e 5 indica "fortemente d'accordo". Le affermazioni coprono vari aspetti dell'usabilità come la facilità di utilizzo e la chiarezza delle funzionalità.

Di seguito le 10 affermazioni del questionario SUS:

1. Penso che mi piacerebbe utilizzare questa piattaforma frequentemente
2. Ho trovato il sito inutilmente complesso
3. Ho trovato il sito molto semplice da usare
4. Penso che avrei bisogno del supporto di una persona già in grado di utilizzare la piattaforma
5. Ho trovato le varie funzionalità del sito bene integrate
6. Ho trovato incoerenze tra le varie funzionalità del sito
7. Penso che la maggior parte delle persone potrebbe imparare ad utilizzare la piattaforma facilmente
8. Ho trovato il sito molto macchinoso da utilizzare
9. Mi sono sentito a mio agio nell'utilizzare il sito
10. Ho avuto bisogno di imparare molti processi prima di riuscire ad utilizzare al meglio il sito

### 4.4.3 **Calcolare punteggio SUS**

Una volta che gli utenti hanno compilato tutte le risposte del questionario, si procede al calcolo dei veri e propri punteggi. Ogni affermazione ha un punteggio che va da 0 a 4, quindi il punteggio massimo totale per il questionario è 40 (10 affermazioni x 4 punti ciascuna).

Il punteggio viene calcolato in modo diverso in base alla domanda, le dispari vengono conteggiate normalmente da 0 a 4 mentre per quelle pari bisogna prendere 4 e sottrarre il punteggio assegnato dall'utente.

Inoltre la scala di valutazione dei punteggi SUS va da 0 a 100, quindi una volta ottenuto il risultato totale di un questionario, un punteggio da 0 a 40 punti, basterà moltiplicare quest'ultimo per 2.5.

In media si può affermare che:

- **Punteggio medio SUS < 68:** Un punteggio superiore a 68 indica un'usabilità sopra la media. Gli utenti percepiscono il sistema come relativamente facile da usare.
- **Punteggio medio SUS > 68:** Un punteggio inferiore a 68 indica un'usabilità sotto la media. Gli utenti potrebbero avere difficoltà nell'utilizzare il sistema e sono necessari miglioramenti.

#### 4.4.4 Risultati

Seguendo le linee guida del SUS si è deciso di presentare il questionario ad un campione di 30 persone con un'età compresa tra 18 e 44 anni. Questo campione è stato selezionato in modo da garantire una rappresentanza diversificata dei possibili utenti della piattaforma *Pop Tools*.

Ecco alcune informazioni aggiuntive sul campione dei tester:

- **Differenze nelle esigenze e negli interessi:** I partecipanti al test hanno rappresentato una varietà di background e interessi. Alcuni sono studenti universitari di discipline differenti, mentre altri sono lavoratori in settori come il design, lo sviluppo web, il marketing digitale e la consulenza. Questa diversità mira a valutare come la piattaforma possa soddisfare le esigenze di un pubblico eterogeneo.
- **Età e background:** L'età dei partecipanti varia da 18 a 44 anni, coprendo una vasta gamma di fasce di età così da verificare che la piattaforma sia accessibile ad utenti con diversi gradi di esperienza digitale.
- **Stato studentesco e lavorativo:** Alcuni partecipanti essendo studenti universitari hanno utilizzato la piattaforma per scopi legati agli studi, come la ricerca e lo sviluppo di risorse per progetti accademici. Altri partecipanti invece, essendo lavoratori attivi, hanno visto la piattaforma come uno strumento per semplificare il loro lavoro quotidiano e migliorare la produttività.

Di seguito, sono riportate le domande del questionario affiancate dalla media delle risposte ottenute dagli utilizzatori.

Domanda	Risposta Media degli Utenti
1. Penso che mi piacerebbe utilizzare questo sistema frequentemente.	3.7
2. Ho trovato il sistema complesso senza che ce ne fosse bisogno.	1.2
3. Ho trovato il sistema molto semplice da usare.	3.9
4. Penso che avrei bisogno del supporto di una persona già in grado di utilizzare il sistema.	1.1
5. Ho trovato le varie funzionalità del sistema bene integrate.	3.6
6. Ho trovato incoerenze tra le varie funzionalità del sistema.	1.3
7. Penso che la maggior parte delle persone potrebbe imparare ad utilizzare il sistema facilmente.	3.8
8. Ho trovato il sistema molto macchinoso da utilizzare.	1.0
9. Ho avuto molta confidenza con il sistema durante l'uso.	3.6
10. Ho avuto bisogno di imparare molti processi prima di riuscire ad utilizzare al meglio il sistema.	1.2

**Tabella 6:** Valutazione dell'Usabilità tramite Questionario SUS

Il punteggio totale attribuito dagli utenti alla piattaforma è di 32.8 che moltiplicato per 2.5 fa ottenere al servizio una **punteggio SUS di 82**.

Complessivamente, il questionario SUS indica un alto grado di soddisfazione degli utenti e un apprezzamento generale per l'usabilità della web directory.

Tuttavia, le risposte suggeriscono anche alcune aree in cui possono avvenire diversi miglioramenti, come l'ottimizzazione dell'integrazione delle funzionalità e l'eliminazione di eventuali incoerenze riscontrate dagli utenti durante l'utilizzo.

Il feedback degli utenti è stato sicuramente fondamentale per guidare le priorità dei futuri aggiornamenti e per garantire che la web directory continui a offrire un'esperienza utente eccezionale.

## 5 Tecnologie e strumenti utilizzati

Nella realizzazione del progetto *Pop Tools* rivolto al mondo digitale e ai freelance, la scelta delle tecnologie e dei *framework* di sviluppo è stata una delle fasi più importanti in quanto ogni componente tecnologica e strumento è stato scelto in modo ponderato in base ai vantaggi e svantaggi che avrebbe portato alla struttura del sistema.

### 5.1 *Framework* di sviluppo

Un *framework*[44] è definito come un insieme di strumenti, librerie e convenzioni che consentono agli sviluppatori di creare applicazioni in modo più rapido ed efficiente. I *framework* di sviluppo forniscono strutture e modelli per l'organizzazione del codice e dei file delle applicazioni, oltre a funzionalità comuni come la gestione dei *database*, la convalida degli input degli utenti, la gestione delle sessioni utente e la sicurezza.

I *framework* sono solitamente progettati per uno specifico linguaggio di programmazione, come *SvelteKit per Javascript*, *Django per Python* o *Laravel per PHP*, consentendo agli sviluppatori di concentrarsi sulla creazione della *business logic* dell'applicazione piuttosto che sui dettagli tecnici dell'infrastruttura sottostante, il che può accelerare notevolmente lo sviluppo e la manutenzione dell'applicativo.

#### 5.1.1 *SvelteKit*

Per lo sviluppo del progetto *Pop Tools* è stato utilizzato *SvelteKit*[38]. Esso è un *framework* per lo sviluppo web moderno e innovativo, progettato per semplificare il processo di costruzione di applicazioni web efficienti e ad alte prestazioni. A differenza dei *framework* frontend tradizionali, che utilizzano il DOM virtuale come React, *SvelteKit* adotta un approccio diverso, spostando gran parte del lavoro in fase di compilazione.

Il nucleo di *SvelteKit* è incentrato sul concetto di utilizzare "componenti Svelte". Questi incapsulano sia l'interfaccia utente che il comportamento associato, creando porzioni di codice autonome, riutilizzabili e facili da gestire e comprendere. Durante il processo di creazione, il compilatore *Svelte* ottimizza e compila questi componenti in codice JavaScript altamente efficiente, che si traduce in un invio di bundle di dimensioni ridotte e tempi di caricamento più rapidi.

Questo approccio offre diversi vantaggi che rendono *SvelteKit* una scelta interessante per gli sviluppatori che desiderano creare applicazioni web dinamiche e reattive.

- **Prestazioni migliorate:** Come scritto in precedenza, *SvelteKit* è noto per le sue prestazioni eccezionali, grazie al suo metodo di compilazione.
- **Sviluppo semplificato:** *SvelteKit* semplifica lo sviluppo di applicazioni web, fornendo agli sviluppatori una struttura solida per le loro applicazioni includendo funzioni integrate di routing, code splitting e server-side rendering, che facilitano la creazione di applicazioni complesse.
- **Facilità d'uso:** *SvelteKit* è adatto ai principianti e ha una curva di apprendimento ridotta rispetto ad altri *framework* come React e Angular. Offre una sintassi pulita e intuitiva che lo rende facile da capire e da utilizzare.
- **Flessibilità:** *SvelteKit* offre agli sviluppatori la possibilità di utilizzare le librerie e gli strumenti che preferiscono. Non impone una struttura specifica, il che rende facile l'integrazione con altri strumenti.
- **Scalabilità:** *SvelteKit* è progettato per gestire applicazioni grandi e complesse, il che lo rende un'ottima scelta per lo sviluppo di web app aziendali.

#### 5.1.2 *Tailwind CSS*

Un altro *framework* utilizzato per lo sviluppo dell'applicativo, anche se non rilevante quanto *SvelteKit*, è *Tailwind CSS*[39] che ha comunque dimostrato di essere una scelta utile al miglioramento della piattaforma sia in termini di coerenza grafica che di accessibilità.

*Tailwind* offre un approccio completamente nuovo e innovativo alla scrittura dello stile per le applicazioni web. A differenza dell'approccio tradizionale, in cui gli sviluppatori scrivono manualmente ogni regola CSS per personalizzare l'aspetto di un sito web, *Tailwind* offre un set predefinito di classi che possono essere applicate direttamente all'HTML.

Questo approccio, noto come *"utility-first"*, porta con sé una serie di vantaggi significativi come la modularità. Il framework è dunque suddiviso in piccoli moduli, ognuno dei quali si occupa di una specifica regola CSS come il padding, il margine, i colori, i tipi di carattere e così via, che consente agli sviluppatori di utilizzare solo le parti del framework che sono necessarie alle finalità del progetto.

La modularità si traduce anche in una maggiore manutenibilità dato che ogni classe CSS è *"auto-contenuta"* e indipendente dalle altre, il che significa che le modifiche possono essere apportate senza il timore di danneggiare altre aree della piattaforma sulle quali non si sta lavorando. Ciò è particolarmente utile in progetti mediamente complessi come il seguente, in cui diverse parti dell'interfaccia utente possono essere gestite da diversi operatori, semplificando notevolmente il processo di sviluppo e manutenzione.

Sebbene l'approccio iniziale possa risultare lento e non intuitivo, un altro grande vantaggio che Tailwind possiede è la sua documentazione completa e ben strutturata che aiuta gli sviluppatori proprio in questa fase iniziale e li segue per tutto il percorso di utilizzo. Essa infatti facilita l'adozione del framework e garantisce che tutti gli sviluppatori abbiano accesso alle informazioni di cui hanno bisogno in qualsiasi momento.

## 5.2 Autenticazione

Anche la gestione delle autenticazioni è stato un tema molto importante affrontato durante lo sviluppo di *Pop Tools*. L'obiettivo era infatti quello di dare la possibilità agli utenti di accedere e registrarsi alla piattaforma in modo semplice garantendo la sicurezza dei dati, la privacy e l'integrità complessiva della piattaforma.

Nei paragrafi successivi verranno quindi descritte tutte le tecnologie utilizzate e implementate nell'applicativo che permettono tutto ciò.

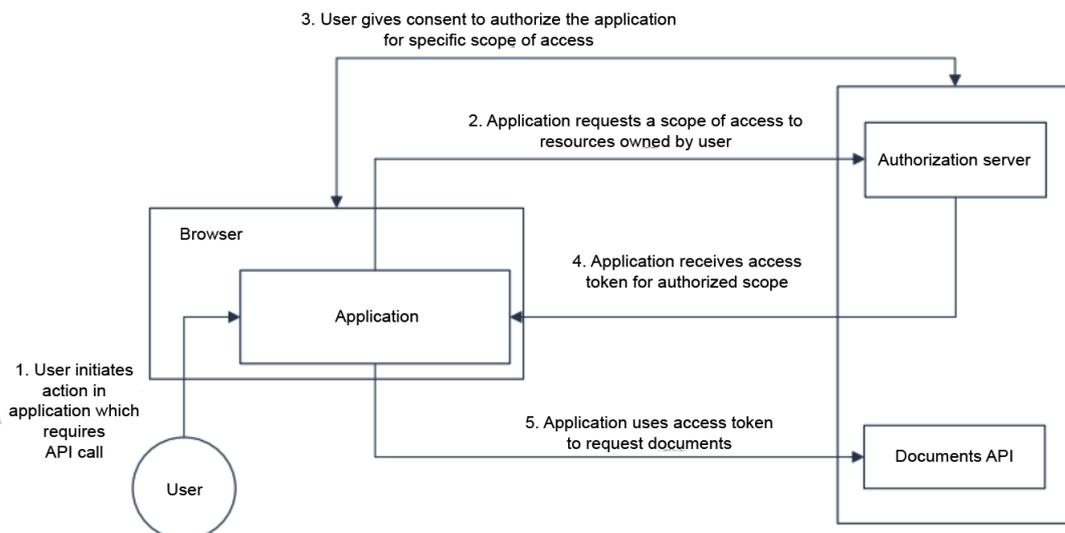
### 5.2.1 OAuth 2.0

OAuth 2.0[28] è un protocollo di autorizzazione, introdotto nel 2012, che consente a un'applicazione di accedere alle API di terze parti in modo sicuro. Questo framework offre all'applicazione la possibilità di ottenere l'autorizzazione per utilizzare le API esterne senza richiedere all'utente di condividere le proprie credenziali di accesso.

L'obiettivo principale di OAuth 2.0 è consentire agli utenti di autorizzare un'applicazione ad accedere alle proprie risorse protette su siti separati e fornire una soluzione a due importanti problematiche. Prima dell'introduzione di questo framework infatti, l'unico modo per consentire a un'applicazione di accedere a tali risorse era fornire le credenziali di accesso, il che poteva comportare un rischio significativo in termini di sicurezza.

Inoltre, una volta che l'applicazione aveva accesso, poteva effettuare operazioni su dati sensibili senza restrizioni, aumentando il potenziale per accessi non autorizzati, specialmente se l'applicazione veniva compromessa.

Inoltre, la revoca delle credenziali significava la loro modifica, il che poteva avere implicazioni negative per altre applicazioni che utilizzavano le stesse credenziali.



**Figura 16:** Processo di funzionamento di OAuth 2.0

Il funzionamento di OAuth 2.0 è suddiviso in diverse fasi:

1. All'interno dell'applicazione, l'utente inizia un'operazione che richiede l'accesso a risorse esterne attraverso API.
2. L'applicazione invia una richiesta di autorizzazione per l'uso delle API a un server dedicato noto come server di autorizzazione. Questa richiesta deve specificare quali risorse intende richiedere tramite le API, definite come "scope".
3. Se le risorse appartengono all'utente, il server di autorizzazione autentica l'utente e chiede il suo consenso per consentire all'applicazione di accedere ai dati specificati.
4. Una volta ottenuto il consenso dell'utente, l'applicazione riceve dall'API server un "access token". Questo token è una chiave di accesso unica che consente all'applicazione di effettuare richieste alle API nel limite delle autorizzazioni concesse in precedenza.

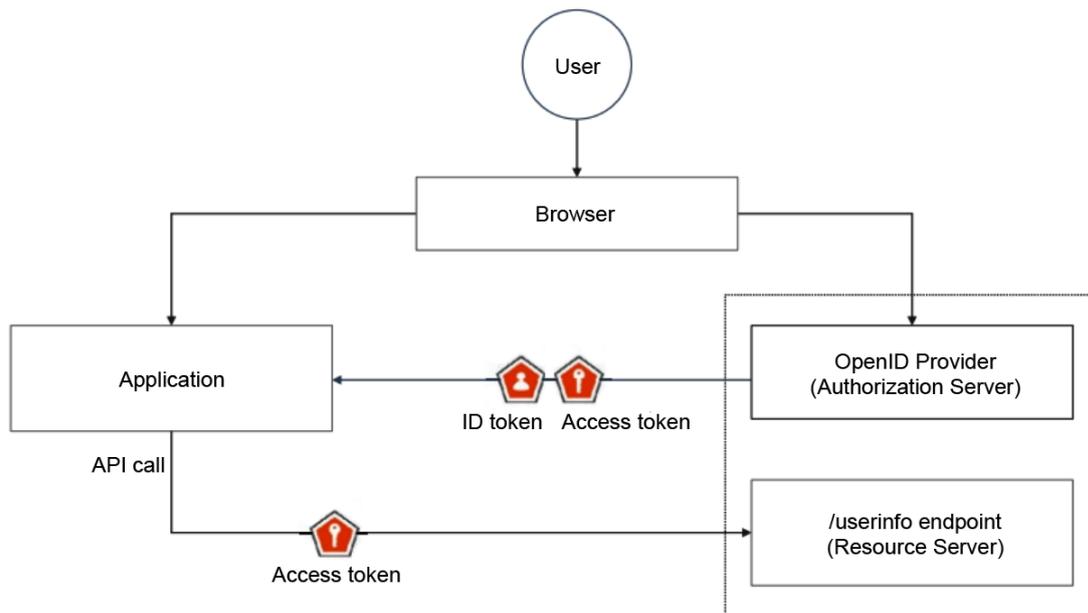
5. L'applicazione utilizza quindi l'access token per accedere e interagire con i dati necessari attraverso le API.

Va notato che OAuth 2.0 si concentra esclusivamente sull'autorizzazione e non gestisce l'autenticazione dell'utente quindi l'access token non trasporta informazioni relative all'identità dell'utente. Per l'autenticazione di esso infatti si utilizza un protocollo diverso che in questo caso è *OIDC (OpenID Connect)*.

### 5.2.2 OpenID Connect

Come anticipato OpenID Connect[29] rappresenta un protocollo di autenticazione basato su OAuth 2.0 che si focalizza principalmente sulla gestione delle autenticazioni degli utenti all'interno della piattaforma.

Il suo obiettivo è consentire agli authorization server di effettuare l'autenticazione degli utenti per conto delle applicazioni e di fornire queste informazioni alle applicazioni stesse.



**Figura 17:** Processo di funzionamento di OpenID Connect

Processo di funzionamento di OpenId Connect:

1. L'utente inizia il processo di accesso all'applicazione.
2. L'applicazione redireziona il browser dell'utente a un server di autorizzazione che implementa OIDC, noto come *OpenID Provider*.
3. L'OpenID Provider comunica con l'utente per verificarne l'identità, soprattutto se l'utente non ha già effettuato l'accesso.
4. Dopo un'autenticazione riuscita, il browser viene rimandato all'applicazione e con esso vengono consegnati i token di sicurezza richiesti.
5. Per ottenere le informazioni di autenticazione dell'utente, l'applicazione ha a disposizione due opzioni:
  - Può richiedere un ID token, che è un token utilizzato per trasmettere dati sull'evento di autenticazione e l'identità autenticata dell'utente alla *relying party*.
  - Oppure può richiedere un access token, che verrà successivamente utilizzato per ottenere le informazioni necessarie dall'endpoint UserInfo.

### 5.2.3 Auth.js

Auth.js[6] è una libreria di autenticazione e autorizzazione che fornisce un metodo sicuro per gestire sia l'accesso che i privilegi degli utenti all'interno della directory ed è un servizio che si integra in modo efficace con il framework SvelteKit utilizzato per lo sviluppo.

Oltre alla sicurezza, Auth.js offre una serie di funzionalità che semplificano notevolmente la gestione delle autenticazioni come la gestione delle sessioni utente, permette di eseguire l'accesso tramite differenti provider come *Google* e *Facebook* e offre la possibilità di definire ruoli e permessi personalizzati per gli utilizzatori.

Ad esempio, nell'applicativo, gli utenti registrati hanno il ruolo di *"utente standard"*, mentre gli amministratori hanno il ruolo di *"admin"* che gli permette di avere accesso a funzionalità autorizzate come la visualizzazione e l'utilizzo della Dashboard.

Per quanto riguarda le sessioni invece, Auth.js permette di gestirle a 360 gradi, garantendone sempre la sicurezza, occupandosi in automatico delle sessioni scadute e ripristinandole qualora l'utente si disconnettesse involontariamente.

Di seguito il codice che illustra come inizializzare Auth.js:

```
import {SvelteKitAuth, type SvelteKitAuthConfig} from '@auth/sveltekit';
import Google from '@auth/core/providers/google';
import {GOOGLE_ID, GOOGLE_SECRET} from '$env/static/private';

const authOptions: SvelteKitAuthConfig = {
  trustHost: true,
  adapter: PrismaAdapter(prisma),
  providers: [
    Google({
      clientId: GOOGLE_ID,
      clientSecret: GOOGLE_SECRET,
    }),
  ],
};
```

## 5.3 Database

All'interno del progetto il database riveste un ruolo cruciale nella gestione dei dati, nell'organizzazione delle informazioni e nella facilitazione delle operazioni di lettura e scrittura.

Nel contesto della web directory l'affidabilità dei dati è di grande importanza. Gli utenti si affidano alla piattaforma per accedere a informazioni accurate e aggiornate riguardo una grande lista di siti web per cui un database ben progettato svolge un ruolo centrale nell'assicurare che queste informazioni siano affidabili e coerenti.

Un *database relazionale*, come quello che è stato implementato nel progetto, è strutturato in modo da gestire dati in modo tabellare, organizzando le informazioni in tabelle con righe e colonne collegate le une alle altre tramite relazioni stabilite in fase progettuale.

Questo approccio permette di ottenere una struttura organizzativa chiara che ne semplifica la gestione e il mantenimento e, inoltre, grazie all'utilizzo di vincoli di integrità referenziale e chiavi primarie ci si assicura che i dati siano strutturati nel modo adeguato e che non ci siano duplicati o informazioni inconsistenti.

### 5.3.1 PostgreSQL

Per questo progetto, la scelta riguardo al sistema di gestione del database è ricaduta su PostgreSQL[31], un *DBMS open-source* gratuito e robusto che si è affermato come uno dei DBMS più affidabili disponibili sul mercato e ampiamente utilizzato in una varietà di applicazioni, da sistemi aziendali complessi a progetti open-source.

PostgreSQL è noto inoltre per la sua scalabilità che lo rende una scelta ideale per un'applicazione in crescita come *Pop Tools* dal momento che è previsto un aumento costante di siti web e utenti all'interno della piattaforma. Il DBMS supporta partizioni, clustering e replica, consentendo una gestione efficace del carico e una ridondanza dei dati garantendo che il sistema possa gestire senza problema una potenziale crescita futura.

PostgreSQL è un DBMS relazionale che utilizza il linguaggio *SQL* per gestirne i dati. Questa struttura relazionale è stata una scelta naturale per la progettazione della web directory poiché possiede i vantaggi seguenti alcuni dei quali già anticipati precedentemente:

- **Precisione:** In un database relazionale, le tabelle possono essere collegate tra loro utilizzando chiavi primarie e chiavi esterne così da evitare duplicati. Pertanto, i database relazionali offrono una maggior precisione rispetto a qualsiasi altro sistema.
- **Accessibilità:** In un DB relazionale non si ha bisogno di un pattern specifico per accedere ai dati diversamente da altre tipologie di database che al contrario necessitano per esempio di essere "navigati" tramite una struttura gerarchica o ad albero. Nel DB relazionale chiunque acceda ai dati può interrogare qualsiasi tabella del database e addirittura combinarle per recuperare dati richiesti utilizzando un join.
- **Integrità:** L'integrità dei dati è una caratteristica fondamentale dei database relazionali. L'inserimento di dati strutturati assicurano che il database contenga solamente dati coerenti e adeguati alla sua struttura relazionale. Queste relazioni tra tabelle aiutano ad evitare che i record siano imperfetti, isolati o non correlati.
- **Flessibilità:** La struttura flessibile di un database relazionale gli consente di adattarsi a requisiti in continuo mutamento, permettendogli di espandersi a piacimento. Ciò semplifica l'aggiunta di nuovi dati, gli aggiornamenti e le eliminazioni.

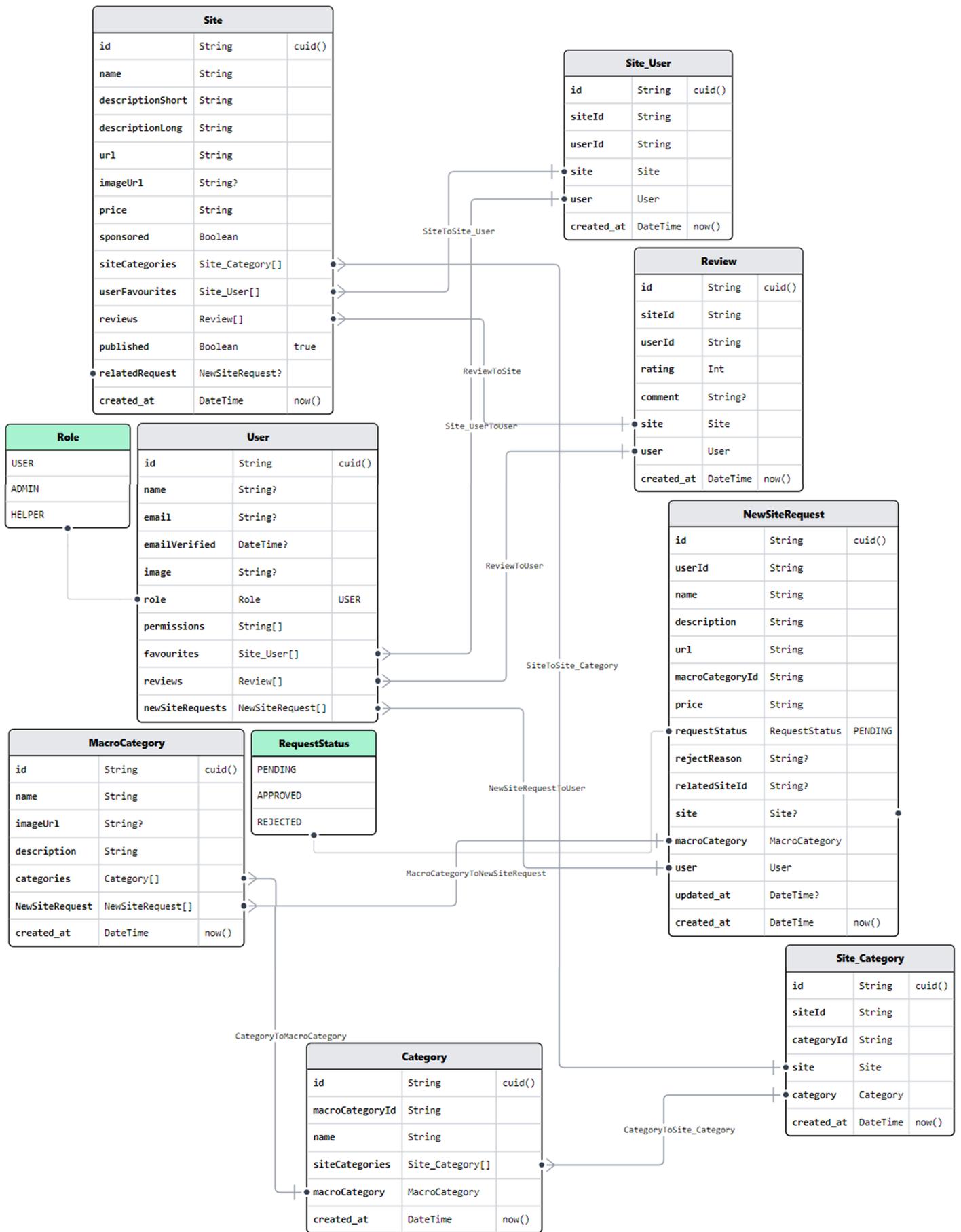


Figura 18: Modello ER del database

### 5.3.2 Prisma

Per gestire i dati di *Pop Tools* in modo efficiente e scalabile si è fatto affidamento su Prisma[32], un *ORM* (*Object-Relational Mapping*) rivoluzionario grazie alla sua natura open source e alla sua architettura di nuova generazione.

Esso si compone di diverse parti fondamentali:

- **Prisma Client:** Si tratta di un query builder per *Node.js* e *TypeScript* che consente agli sviluppatori di interagire con il database in modo efficiente e sicuro, garantendo al contempo la correttezza dei tipi.
- **Prisma Migrate:** Questo sistema di migrazione consente agli sviluppatori di effettuare modifiche strutturali al database in modo controllato.
- **Prisma Studio:** Una *GUI* (*Graphical User Interface*) che permette di visualizzare e modificare i dati nel database. Va notato che Prisma Studio è l'unica parte di Prisma ORM che non è open source. Tuttavia, può essere eseguito localmente.

Il cuore di Prisma è il suo schema, che rappresenta un punto di partenza per ogni progetto che utilizza gli strumenti di esso. Questo schema permette agli sviluppatori di definire i vari modelli utilizzando un linguaggio più semplice e intuitivo e offre anche diverse funzionalità per la modellazione dei dati che semplificano la gestione delle relazioni tra i dati.

Oltre a questo Prisma offre anche un supporto completo per le operazioni *CRUD* (*Create, Read, Update, Delete*) sui dati del database. Queste operazioni sono fondamentali per la gestione dei dati all'interno del progetto, consentendo agli utenti di aggiungere nuovi siti, leggere informazioni esistenti, aggiornare dettagli e, se necessario, eliminarli.

Di seguito alcuni esempi che mostrano la semplicità di utilizzo offerta da Prisma:

#### 1. Create (creazione di nuovi dati)

```
import { PrismaClient } from '@prisma/client';

const prisma = new PrismaClient();

async function createNewWebsite() {
  const newWebsite = await prisma.website.create({
    data: {
      name: 'Il Mio Sito Web',
      url: 'https://www.ilmiositoweb.com',
      description: 'Un sito web incredibile',
      // Altri campi dati del sito web
    },
  });
  console.log('Nuovo sito web creato:', newWebsite);
}

createNewWebsite()
  .catch((error) => {
    console.error('Errore durante la creazione del sito web:', error);
  })
  .finally(async () => {
    await prisma.$disconnect();
  });
```

## 2. Read (lettura di dati esistenti)

```
import { PrismaClient } from '@prisma/client';

const prisma = new PrismaClient();

async function readWebsite(websiteId: number) {
  const website = await prisma.website.findUnique({
    where: { id: websiteId },
  });
  console.log('Sito web letto:', website);
}

readWebsite(1)
  .catch((error) => {
    console.error('Errore durante la lettura del sito web:', error);
  })
  .finally(async () => {
    await prisma.$disconnect();
  });
```

## 3. Update (aggiornamento di dati esistenti)

```
import { PrismaClient } from '@prisma/client';

const prisma = new PrismaClient();

async function updateWebsite(websiteId: number) {
  const updatedWebsite = await prisma.website.update({
    where: { id: websiteId },
    data: {
      description: 'Nuova descrizione del sito web',
      // Altri campi dati da aggiornare
    },
  });
  console.log('Sito web aggiornato:', updatedWebsite);
}

updateWebsite(1)
  .catch((error) => {
    console.error("Errore durante l'aggiornamento del sito web:", error)
    ↵ ;
  })
  .finally(async () => {
    await prisma.$disconnect();
  });
```

### 5.3.3 *Firebase*

Nell'ambito dello sviluppo della web directory, si è scelto di adottare Firebase[26] come soluzione ottimale per la gestione delle immagini.

Firebase è infatti una piattaforma di servizi cloud gestiti e di proprietà di Google che offre una vasta gamma di funzionalità per lo sviluppo di applicazioni web e mobili. Inizialmente noto per il suo ampio utilizzo nello sviluppo di app Android, Firebase è diventato un'ottima soluzione anche per progetti web, offrendo una serie di vantaggi che si adattano perfettamente alle esigenze di *Pop Tools*.

Uno dei motivi principali per la scelta di Firebase è la sua facilità di integrazione. Firebase è già integrato con diverse tecnologie utilizzate all'interno del progetto, semplificando l'implementazione e l'utilizzo dei suoi servizi.

La funzione che è stata utilizzata è il *Firebase Cloud Storage* che permette di memorizzare e recuperare le immagini che, nel progetto, sono state utilizzate nelle pagine dei singoli siti web o come copertina nelle card relative alle categorie, in modo affidabile e scalabile garantendone l'accesso in qualsiasi momento, su qualsiasi dispositivo e senza ritardi.

Di seguito il codice per la configurazione di Firebase:

```
import serviceAccount from './admin-key.json';

export const storageBucket = 'looneytools-946fa.appspot.com';

export const getSiteImageUrl = (fileName: string) =>
  `https://firebasestorage.googleapis.com/v0/b/${storageBucket}/o/site
  ↪ %2F${fileName}?alt=media`;

export const getMacrocategoryImageUrl = (fileName: string) =>
  `https://firebasestorage.googleapis.com/v0/b/${storageBucket}/o/
  ↪ macrocategory%2F${fileName}?alt=media`;

if (getApps().length === 0) {
  initializeApp({
    credential: cert({
      projectId: serviceAccount.project_id,
      clientEmail: serviceAccount.client_email,
      privateKey: serviceAccount.private_key,
    }),
    storageBucket,
  });
}

export const storage = getStorage();
```

## 6 Scelte progettuali e implementative

Come anticipato nel paragrafo 5.1.1 il framework SvelteKit è stato una scelta fondamentale che ha gettato delle solide fondamenta per lo sviluppo della struttura dell'applicativo.

SvelteKit adotta un *sistema di routing*[34] che consente di gestire facilmente le pagine del sito web. Esso è basato su file e cartelle, il che significa che per creare una nuova pagina bisognerà spostarsi nella directory *“routes”* del progetto e creare una nuova cartella avente il nome della pagina scelta, come per esempio *“login”* al cui interno possono essere implementate diverse tipologie di file come:

- **+page.svelte**: Il componente **+page.svelte** definisce una pagina dell'applicazione. Di base le pagine vengono renderizzate sul server (*SSR*) durante la prima richiesta che viene eseguita e nel browser (*CSR*) per tutte le successive navigazioni.
- **+page.ts**: Spesso, una pagina deve caricare alcuni dati prima di poter essere renderizzata. Per questo, viene aggiunto un modulo **+page.js** che esporta una funzione di caricamento come chiamate alle API.
- **+page.server.ts**: Se la funzione di caricamento può essere eseguita solo sul server, ad esempio se è necessario recuperare dati da un database o accedere a variabili d'ambiente private come chiavi API, è possibile rinominare **+page.js** in **+page.server.js** e modificare il tipo di caricamento in *PageServerLoad*.
- **+layout.svelte**: Fino a questo punto, le pagine sono state trattate come componenti completamente autonomi. Ad ogni richiesta il componente **+page.svelte** esistente verrà distrutto e ne verrà creato uno nuovo al suo posto. Tuttavia, in molte applicazioni, ci sono elementi che dovrebbero essere visibili su ogni pagina, come la barra di navigazione o un piè di pagina e, invece di ripeterli in ogni **+page.svelte**, è possibile inserirli in **+layout.svelte**.

Tutte le pagine “pubbliche” del progetto sono accessibili direttamente e quindi utilizzano *Server-Side Rendering* per la visualizzazione delle informazioni come la Homepage, Search e Login permettendo un'ottimizzazione a livello di *SEO*.

Questo processo migliora la velocità di caricamento e l'indicizzazione di *Pop Tools* da parte dei motori di ricerca, contribuendo così a posizionare il sito web in modo vantaggioso nei risultati.

D'altra parte, le pagine amministrative come la Dashboard, non richiedono la stessa ottimizzazione. Pertanto, queste pagine sono state implementate utilizzando il rendering lato client, più adatto per pagine amministrative dove il contenuto dinamico e l'interattività sono più importanti di una rapida indicizzazione.

Un altro aspetto importante della progettazione delle pagine amministrative è il controllo degli accessi. Per garantire che solo gli utenti autorizzati possano accedere alle funzionalità della Dashboard è stato implementato un sistema di routing protetto. Questo permette che alcune routes siano accessibili solo da utenti con determinati permessi o ruoli, come gli amministratori del sistema.

Questo controllo sugli accessi viene gestito nel file *“hook.server.ts”*, che funge da *middleware* all'interno di SvelteKit. Questo file è responsabile di verificare i permessi dell'utente prima di consentirne l'accesso a una pagina. Nel caso in cui un utente non abbia i permessi necessari, il sistema di routing impedisce la navigazione verso la pagina protetta e reindirizza l'utente a una pagina di errore o a un'altra destinazione appropriata.

Di seguito il codice per l'implementazione del routing protetto:

```
const authorization: Handle = async ({ event, resolve }) => {
  const session = await event.locals.getSession();

  if (event.url.pathname.includes('login')) {
    if (session) {
      throw redirect(303, '/');
    }
  }

  if (event.route.id?.includes('(protected)')) {
    if (!session) {
      throw redirect(303, '/');
    }
  }

  if (event.route.id?.includes('(admin)')) {
    if (!session) {
      throw redirect(303, '/');
    }

    if (hasEveryPermission(event, ['can_view_admin_dashboard']) ===
    ↪ false) {
      throw redirect(303, '/');
    }
  }

  return resolve(event);
};
```

## 6.1 Struttura *BE*

La componente di back-end di *Pop Tools* è stata progettata con grande attenzione per garantire un funzionamento efficiente, scalabile e sicuro. In questa sezione, esploreremo i principali aspetti dell'architettura del BE e come sono stati implementati nel contesto del progetto.

### 6.1.1 *REST API*

Per comprendere appieno l'architettura del nostro back-end, è fondamentale avere una conoscenza chiara di cosa siano le REST API. Esse rappresentano un insieme di regole e convenzioni per la progettazione di servizi web che consentono la comunicazione e lo scambio di dati tra diverse applicazioni o sistemi. Questi servizi sono basati sul protocollo HTTP, fondamentale per la comunicazione su Internet, rendendoli efficienti e facili da utilizzare.

Le REST API seguono il principio fondamentale dell'architettura REST (*Representational State Transfer*), che implica che le risorse del sistema siano rappresentate in modo chiaro e uniforme attraverso l'URL e che le operazioni su queste risorse siano eseguite utilizzando i metodi standard di HTTP, come *GET* (per recuperare dati), *POST* (per creare nuove risorse), *PUT* (per aggiornare risorse esistenti) e *DELETE* (per rimuovere risorse).

Le API sviluppate nel progetto sono state strutturate in modo organizzato all'interno della cartella *src/routes/api*, dove ogni file rappresenta un endpoint specifico progettato per soddisfare le diverse esigenze del sistema e consentire l'interazione con il database e altre funzionalità cruciali.

Un aspetto importante della progettazione di queste API è stato il versionamento, che è stato implementato attraverso la definizione di un prefisso *"v1"*. Questo ci consente di gestire le future evoluzioni delle API senza interrompere il funzionamento delle versioni precedenti, garantendo la compatibilità all'indietro per gli sviluppatori che le utilizzano.

Le REST API sono state divise in due categorie principali: API per gli amministratori e API *"pubbliche"*. Questa suddivisione è stata fatta per garantire una gestione appropriata delle funzionalità amministrative del nostro sistema. Le API per gli amministratori sono state progettate specificamente per la parte di amministrazione della web directory e, di conseguenza, tutti gli *endpoint* dispongono di controlli specifici sui permessi. Questi controlli saranno dettagliatamente discussi nel capitolo 6.4.

Al contrario, le API pubbliche, sono utilizzabili da tutti gli utenti, con alcune eccezioni che prevedono controlli sugli utenti autenticati permettendogli per esempio di poter accedere ai propri tools preferiti, di poter accedere alla pagina per l'invio di una nuova proposta di un sito web o di poter recensire un sito web.

### 6.1.2 *Query e modelli*

Una delle sfide principali affrontate a lato back-end è stata la gestione delle query più complesse come per esempio la ricerca e il filtraggio dei siti web all'interno della directory. Per farlo si è scelto di non utilizzare Prisma (descritto nel paragrafo 5.3.2) nella sua forma più semplice ma di sviluppare appositamente dei query builder e modelli specifici.

Questi query builder consentono di avvolgere la complessità delle query e la logica di validazione in funzioni e modelli distinti, rendendo il codice più leggibile e semantico. Ad esempio, per la ricerca dei siti web, è stato creato un query builder che permette di facilitare la manutenzione del codice e la comprensione delle query.

Inoltre, alcuni dei modelli che rappresentano i dati nel back-end sono stati progettati seguendo il principio dei *"modelli non anemici"*[24]. Questo significa che questi modelli non sono semplici contenitori passivi di dati, ma includono logica di validazione e metodi che ne garantiscono la coerenza. Per esempio, quando un utente invia una richiesta per aggiungere un nuovo sito web alla web directory, il modello che rappresenta questa richiesta contiene la logica per validare i dati inseriti, garantendo che siano conformi alle specifiche del sistema riducendo il rischio di errori nei dati e migliorando la coerenza del database.

Per questi modelli si è scelto di non fare un uso estensivo dei tipi generati da Prisma, poiché a volte possono risultare complessi da comprendere a causa della struttura complicata dei dati JSON che viene generata da determinate query. Invece, sono state create classi TypeScript personalizzate seguendo uno specifico pattern di progettazione chiamato *"factory method"*. In questo modo si rende il costruttore privato, impedendo la creazione diretta di oggetti e garantendo che i dati siano sempre validi.

Questo metodo permette inoltre di mappare i risultati ottenuti dalla query di Prisma in modelli custom chiamati *"ViewModel"* in maniera ancora più semplice, utilizzando il metodo *MyModel.create(myModelDto)*:

*MyModelDTO*). Questi modelli sono progettati per soddisfare specificatamente le esigenze del front-end e vengono condivisi e utilizzati per garantire che tutte le informazioni richieste siano disponibili.

Anche per quanto riguarda la creazione di risorse come per esempio la creazione di un nuovo sito o di una review si è pensato di utilizzare modelli personalizzati identificati con la notazione *NewMyModel*. Questi infatti racchiudono come già specificato in precedenza la logica e le validazioni nell'apposito factory method *create()* e inoltre implementano un metodo *persist()* che permette di creare la nuova risorsa all'interno del database.

```
export const POST: RequestHandler = async (event) => {
  const { user } = event.locals;

  if (!user) {
    return new Response(null, { status: 401 });
  }

  if (hasEveryPermission(event, [Permissions.CAN_EDIT_SITE_SETTINGS])
    ↪ === false) {
    return new Response(null, { status: 403 });
  }

  try {
    const newSite = NewSite.create(await event.request.json());

    await newSite.persist();

    AppLogger.info(`New site created: ${newSite.name}`);

    return json(newSite);
  } catch (error) {
    AppLogger.error((error as Error).message, (error as Error).stack);
    return new Response(null, { status: 500 });
  }
};
```

Questo codice interno all'API *src/routes/api/v1/admin/sites/+server.ts* permette di creare un nuovo sito nel DB. Da notare che avendo creato *newSite* con il factory method il codice risulta pulito e sicuro in quanto le validazioni sono gestite all'interno di tale metodo.

## 6.2 Struttura *FE*

Come anticipato nell'introduzione del capitolo, il front-end di *Pop Tools* è costituito da varie routes, ovvero dalle pagine navigabili dall'utente che, al loro interno fanno largo uso di componenti Svelte.

L'obiettivo principale del progetto è stato infatti quello di strutturarli per la maggioranza da componenti singoli che inglobassero sia il layout che la business logic in modo da snellire il più possibile il codice presente nelle routes.

All'interno dell'applicativo sono state introdotte due tipologie di componenti differenti che verranno approfondite di seguito.

### 6.2.1 Componenti *AppMyComponent*

Questi componenti costituiscono la maggioranza all'interno della directory e la loro caratteristica distintiva è che sono progettati per essere riutilizzabili e generalmente non gestiscono stati o azioni specifiche. Invece, accettano input sotto forma di proprietà o di *ViewModel* e il loro ruolo principale è quello di far visualizzare all'utente dati o contenuti.

Un esempio di tale componente largamente utilizzato potrebbe essere *"AppSiteCard"* che viene richiamato per rappresentare e visualizzare le card contenenti le informazioni di un sito web all'interno della web directory.

Questo tipo di componente viene riutilizzato in diverse aree dell'applicazione per garantire una coerenza visiva e funzionale. Esso si trova infatti nella Homepage, in Search e all'interno dei Favourites dell'utente e questa riusabilità ne semplifica anche la manutenzione, poiché eventuali modifiche apportate al componente si rifletteranno automaticamente ovunque esso venga utilizzato.

### 6.2.2 Componenti *logici*

Oltre agli *AppMyComponent*, sono stati sviluppati anche un'altra tipologia di componenti denominati *"logici"*.

Questi componenti, anche se presenti in numero minore, hanno una responsabilità più specifica e funzionalità maggiori rispetto agli altri dato che, oltre a mostrare dei dati, si occupano anche di gestire stati, effettuare chiamate alle API e gestire azioni come l'invio di form da parte dell'utente.

Essi sono quindi più complessi rispetto agli *AppMyComponent* poiché devono gestire una vera e propria business logic.

Un esempio importante di componente logico è il *"Filters"*. Esso è infatti responsabile della gestione dei filtri di ricerca all'interno della pagina Search della web directory ed è anche riutilizzato per gestire meglio i siti preferiti dall'utente.

Esso comprende diverse funzionalità come mantenere lo stato dei filtri selezionati dall'utente, reagire dinamicamente alle modifiche apportate agli stessi o eseguire chiamate alle API per aggiornare i risultati visualizzati in base ai filtri selezionati. La struttura grafica del componente *Filters* è visualizzabile al paragrafo 3.3.3.

## 6.3 Data fetching

All'interno del progetto la gestione dei dati e la comunicazione con il server sono aspetti critici che influenzano direttamente l'esperienza dell'utente e necessitano di essere gestiti nel modo migliore.

Per affrontare questo compito in modo efficiente sono stati utilizzati diversi metodi che si differenziano in base al contesto specifico di riferimento.

### 6.3.1 Funzioni di *Load*

Una delle caratteristiche fondamentali di SvelteKit sfruttate all'interno di *Pop Tools* sono le "funzioni di load". Queste consentono di effettuare chiamate al server prima che la pagina venga renderizzata. Nelle pagine che richiedono Server-Side Rendering, come ad esempio i file con estensione "+page.ts", queste funzioni vengono eseguite prima di generare il contenuto HTML della pagina stessa mentre nel caso delle pagine "+page.server.ts", le funzioni di load vengono eseguite sempre in modalità SSR.

Queste funzioni di load sono state ampiamente sfruttate per recuperare dati nella parte "pubblica" del sito, escludendo l'area amministrativa. Questo approccio ha permesso di ottenere i benefici del Server-Side Rendering, che include una migliore performance e un supporto migliore per essere visibili sui motori di ricerca e ha consentito di separare la logica di recupero dei dati dai componenti stessi, contribuendo a mantenere il codice più pulito e manutenibile.

I dati ottenuti da queste funzioni di load sono resi disponibili nella proprietà "data" esportata nel file "+page.svelte". Grazie ad essa è possibile accedere a tutte quelle informazioni che dovranno poi essere utilizzate o mostrate all'utente. Per fare un esempio pratico, grazie al codice sottostante presente nel file /src/routes/+page.ts, richiamando la proprietà "sites" all'interno di +page.svelte, verrà restituito un array di siti web contenente per ogni sito le informazioni come il suo nome, l'URL, la descrizione, il rating e così via.

```
import type { SiteViewModel } from '$lib/types/SiteViewModel';
import type { PaginatedResponse } from '$lib/types/http/
  ↳ PaginatedResponse';
import type { Load } from '@sveltejs/kit';

export const load: Load = async ({ fetch, url }) => {
  const fetchSites = async () => {
    const data = await fetch(`api/v1/sites?page=${url.searchParams.get('
  ↳ page') ?? 1}`);
    const sites = (await data.json()) as PaginatedResponse<SiteViewModel
  ↳ >;
    return sites;
  }

  return {
    sites: fetchSites(),
  };
};
```

### 6.3.2 Client-Side data fetching

Per alcune chiamate, in particolare quelle effettuate nell'area amministrativa, sono state utilizzate funzioni TypeScript personalizzate per comunicare direttamente con il server in modo da migliorare la reattività dell'applicazione, essenziale nell'area della Dashboard.

Queste funzioni sono state chiamate "services" e sono state progettate per essere chiare e standardizzate.

Di seguito un esempio di servizio utile a recuperare i dati relativi ad un sito web all'interno del pannello amministrativo:

```
import type { NewSiteRequestInfoDTO } from '$lib/types/DTO/  
  ↳ NewSiteRequestInfoDTO';  
  
export const getRequestInfoData = async (requestId: string): Promise<  
  ↳ NewSiteRequestInfoDTO> => {  
  
  const response = await fetch(`/api/v1/admin/requests/${requestId}`);  
  const data = await response.json();  
  
  if (data === null) {  
    throw new Error(`Request with id ${requestId} not found`);  
  }  
  
  return data;  
}
```

Questa struttura comune dei services prevede una serie di dati come: eventuali parametri da passare come "query params" o all'interno del body (nel caso di operazioni PUT o POST), l'effettiva chiamata con fetch, il mapping della risposta, e il ritorno dei dati.

Per quanto riguarda gli errori, non sono gestiti all'interno del servizio ma da altri componenti che, a loro volta, mostrano messaggi di errore appropriati come i "toasts"; elementi grafici largamente utilizzati all'interno della piattaforma per la visualizzazione di messaggi che possono essere informativi, di errore o successo.

### 6.3.3 Form actions

Un'altra caratteristica chiave di SvelteKit che è stata sfruttata nel progetto sono le form actions. Queste permettono di utilizzare i form HTML in modo molto efficiente per l'invio di dati al DB.

Le form action sono state principalmente utilizzate nella parte pubblica del sito, dove gli utenti inseriscono o creano nuove risorse, come le proposte o le recensioni.

Questo approccio è stato scelto perché le form actions consentono di eseguire operazioni sul server e ottenere un "refresh" silenzioso della pagina.

In pratica, quando un utente completa un form e lo invia, l'operazione viene eseguita sul server e la pagina viene aggiornata con i nuovi dati senza un ricaricamento completo di essa.

Questo offre un'esperienza utente fluida anche nei casi in cui l'utente non abbia abilitato JavaScript all'interno del proprio browser.

## 6.4 Ruoli e permessi

La gestione dei ruoli e dei permessi all'interno del progetto è un elemento fondamentale per garantire un controllo efficace sulle azioni che gli utenti possono svolgere e a definire chi può fare cosa all'interno della web directory.

In effetti, i ruoli sono un concetto utilizzato principalmente per assegnare vari permessi a un utente. Questa struttura offre un alto grado di flessibilità, consentendo di adattare le autorizzazioni in modo specifico alle esigenze dell'applicazione.

Nel database, ogni utente ha una colonna dedicata al ruolo, che può assumere uno dei seguenti valori: *"NULL"* (quando l'utente non ha un ruolo specifico), *"USER"* (utente standard), *"HELPER"* (utente assistente), e *"ADMIN"* (amministratore con tutti i privilegi).

Oltre alla colonna del ruolo, c'è anche una colonna dedicata ai permessi, che contiene un array di stringhe rappresentanti vari permessi. Questi permessi sono utilizzati per definire in modo dettagliato ciò che un utente può o non può fare all'interno dell'applicazione come visualizzabile nel codice successivo rappresentante la lista di tutti i privilegi assegnabili:

```
export enum Permissions {
  CAN_VIEW_ADMIN_DASHBOARD = 'can_view_admin_dashboard',

  CAN_SEE_PROFILE_PAGE = 'can_see_profile_page',

  CAN_EDIT_SITE_SETTINGS = 'can_edit_site_settings',
  CAN_DELETE_SITE = 'can_delete_site',

  CAN_DELETE_CATEGORY = 'can_delete_category',
  CAN_EDIT_CATEGORY = 'can_edit_category',

  CAN_SETTLE_REQUEST = 'can_settle_request',

  CAN_EDIT_OWNED_REVIEW = 'can_delete_owned_review',

  CAN_DELETE_ALL_REVIEWS = 'can_delete_all_reviews',
}
```

Una buona gestione dei ruoli e dei permessi è perciò una condizione necessaria per garantire la sicurezza e la personalizzazione dell'applicazione. Ad esempio, un utente standard ha il permesso di visualizzare il proprio profilo ma non può visualizzare la Dashboard.

D'altra parte, un amministratore ha accesso completo, incluso il permesso di modificare i dettagli dei siti e gestire gli utenti.

Questo sistema consente di controllare attentamente chi ha accesso a quali funzionalità e di limitare le azioni non autorizzate, in più permette di conferire anche temporaneamente permessi straordinari ad un utente qualora fosse necessario.

### 6.4.1 Implementazione nel *BE*

Nel back-end, il file *"hook.server.ts"* svolge un ruolo fondamentale nella gestione delle rotte protette.

In particolare, la funzione *"handleAuth"* si occupa di configurare l'autenticazione utilizzando *Auth.js* (affrontato nel capitolo 5.2.3) e di recuperare le informazioni sull'utente che ha effettuato una richiesta.

Durante questo processo, vengono estratti il ruolo e i permessi dell'utente utilizzando il metodo di utility *"getPermissionsForUser"* che ha il compito di:

1. Recuperare i permessi dalla colonna del database se presenti.
2. Ottenere il ruolo dell'utente.
3. Assegnare permessi specifici in base al ruolo.
4. Restituire un array di permessi, che rappresenta l'insieme dei permessi dell'utente, combinando quelli presenti nel database con quelli assegnati al ruolo.

### 6.4.2 Implementazione nel *FE*

Anche nel frontend, i permessi sono utilizzati per determinare quali funzionalità o pulsanti devono essere resi disponibili agli utenti. Ad esempio, all'interno delle pagine viene effettuato un controllo per verificare se un utente ha un determinato permesso prima di mostrare o nascondere determinati elementi dall'interfaccia.

Un esempio concreto può essere osservato nella Dashboard, all'interno della sezione che consente agli amministratori di eliminare un sito web. In questa pagina, prima di visualizzare il pulsante di eliminazione, verifichiamo se l'utente ha il permesso specifico *"CAN\_DELETE\_SITE"*. Se l'utente non ha questo permesso, il pulsante non viene mostrato, impedendo così azioni non autorizzate.

Inoltre, i permessi dell'user sono reperibili all'interno di tutta l'applicazione, essendo recuperati globalmente durante il caricamento del layout principale *"layout.server.ts"*. Questo significa che i permessi possono essere facilmente verificati e utilizzati in qualsiasi parte dell'applicazione senza doverli richiedere separatamente per ogni pagina o componente.

## 7 Conclusioni

La realizzazione di *Pop Tools* è stata un processo intriso di sfide e scoperte significative emerse in risposta a una chiara necessità nel panorama digitale contemporaneo di semplificare la ricerca e l'accesso a risorse di alta qualità in un mare di informazioni e strumenti online.

Questa web directory si è differenziata dalle soluzioni esistenti per diversi motivi chiave. Prima di tutto, è stata progettata attentamente per adattarsi alle esigenze specifiche di professionisti e freelance, un pubblico notoriamente attivo ed esigente quando si tratta di strumenti e risorse digitali che possono semplificare il loro lavoro.

Secondariamente, ha incentivato attivamente la partecipazione degli utenti, promuovendo la condivisione di pensieri ed esperienze all'interno della sua comunità generando un valore significativo per la stessa. Grazie alle recensioni e alle proposte degli utenti la redazione è infatti capace di comprendere al meglio l'andamento del mercato digitale, concentrando le proprie risorse per ricercare e proporre strumenti sempre più utili e specifici.

Inoltre grazie ai risultati del questionario SUS è stata confermata la semplicità d'uso di *Pop Tools*. Con un punteggio medio di 82 su 100, la piattaforma ha dimostrato di offrire un'esperienza utente sopra la media e raggiungere uno dei suoi obiettivi principali ovvero quello di fornire un ambiente in cui gli utenti si sentono a proprio agio, non necessitano di supporto esterno e le funzionalità sono ben integrate.

Riguardo al confronto con altre directory rilevanti, come Product Hunt, Futurepedia, Toools Design, Insanely Useful Websites e Ncode, emerge la sua specificità nell'indirizzare su strumenti specifici per professionisti e freelance. Product Hunt si concentra infatti sul lancio di nuovi prodotti e strumenti online, Futurepedia è altamente specializzata nell'ambito dell'intelligenza artificiale, mentre le altre directory come Toools Design, Insanely Useful Websites e Ncode sono più specifiche o limitate a nicchie particolari. *Pop Tools* si distingue quindi prendendo tutti i vantaggi e le funzionalità più utili da ciascuna, riarrangiando e migliorandone i vari aspetti come filtri di ricerca, struttura delle card e informazioni fondamentali da mostrare, struttura delle pagine e componenti relative alla community creando un ecosistema interattivo e dinamico.

Una delle osservazioni più significative emerse durante lo sviluppo della directory è stata l'importanza della massa critica di utenti. È emerso chiaramente che *Pop Tools*, per raggiungere il suo massimo potenziale, deve attirare un numero sufficiente di utenti attivi. Questo concetto è in parte autoalimentante: più utenti contribuiscono, più ricca diventa la piattaforma, attirando ulteriori utilizzatori interessati. Tuttavia, raggiungere questa massa critica iniziale può essere una sfida, e richiede strategie di incentivazione, promozione e coinvolgimento degli utenti fin dall'inizio.

*Pop Tools* ha dimostrato però di avere un grande potenziale, ma il suo percorso è appena iniziato. Ci sono diverse prospettive future da esplorare per migliorare ulteriormente la piattaforma e massimizzare il suo impatto come:

1. **Espansione della Community:** Un obiettivo chiave per il futuro è espandere la comunità di utenti di *Pop Tools*. Aumentare il numero di utenti attivi contribuirà a creare una piattaforma ancora più ricca di risorse e a consolidare il suo valore per i professionisti e i freelance.
2. **Implementazione Newsletter settimanale:** Per mantenere gli utenti informati sulle ultime risorse e aggiornamenti, si prevede l'implementazione di una newsletter settimanale. Questa newsletter fornirà agli utenti contenuti selezionati, suggerimenti e notizie pertinenti direttamente nella loro casella di posta.
3. **Implementazione Forum:** Per facilitare ulteriormente la condivisione di conoscenze e discussioni tra gli utenti, si prevede di implementare un forum all'interno di *Pop Tools*. Gli utenti potranno aprire argomenti di discussione su una vasta gamma di temi legati al mondo digitale, favorendo maggiormente l'interazione e la collaborazione.

Per permettere questi sviluppi, durante la progettazione della struttura del codice, è stata posta una particolare attenzione alla manutenibilità e alla flessibilità riguardo aggiornamenti e integrazioni future. Tutte le componenti sono state progettate in modo modulare, consentendo modifiche e aggiunte senza compromettere l'integrità del sistema. Ciò significa che *Pop Tools* è pronta ad adattarsi a nuove esigenze, incorporando funzionalità e miglioramenti futuri con facilità.

## Bibliografia e Sitografia

- [1] *Accessibility Guidelines*. URL: [https://ysu.edu/sites/default/files/mathematics-achievement-center/Hawkes%20Learning\\_Web%20Content%20Accessibility%20Guidelines%20\(WCAG\)%2020.pdf](https://ysu.edu/sites/default/files/mathematics-achievement-center/Hawkes%20Learning_Web%20Content%20Accessibility%20Guidelines%20(WCAG)%2020.pdf).
- [2] *Adobe Photoshop*. URL: <https://www.adobe.com/it/products/photoshop.html>.
- [3] *Alessandro Gandini*, «Il lavoro freelance: reputazione e capitale sociale nell'era del "lavoro digitale"», *Quaderni di Sociologia*, 69 — 2015, 87-106. URL: <https://journals.openedition.org/qds/522#quotation>.
- [4] *Asana*. URL: <https://asana.com/it>.
- [5] *Attori e Casi d'uso*. URL: <https://webthesis.biblio.polito.it/18167/1/tesi.pdf>.
- [6] *Auth.js*. URL: <https://developer.okta.com/docs/guides/auth-js/main/>.
- [7] Meltem Huri Baturay. «An Overview of the World of MOOCs». In: *Procedia - Social and Behavioral Sciences* 174 (2015). International Conference on New Horizons in Education, INTE 2014, 25-27 June 2014, Paris, France, pp. 427-433. ISSN: 1877-0428. DOI: <https://doi.org/10.1016/j.sbspro.2015.01.685>. URL: <https://www.sciencedirect.com/science/article/pii/S1877042815007363>.
- [8] Sergey Brin e Lawrence Page. «The anatomy of a large-scale hypertextual Web search engine». In: *Computer Networks and ISDN Systems* 30.1 (1998). Proceedings of the Seventh International World Wide Web Conference, pp. 107-117. ISSN: 0169-7552. DOI: [https://doi.org/10.1016/S0169-7552\(98\)00110-X](https://doi.org/10.1016/S0169-7552(98)00110-X). URL: <https://www.sciencedirect.com/science/article/pii/S016975529800110X>.
- [9] John Brooke. «SUS: A quick and dirty usability scale». In: *Usability Eval. Ind.* 189 (nov. 1995).
- [10] *Coursera*. URL: <https://www.coursera.org/>.
- [11] *Definizione era digitale*. URL: <https://www.treccani.it/enciclopedia/era-digitale/#:~:text=L'arco%20di%20tempo%20iniziato,cui%20precisione%20pu%C3%B2%20essere%20qualsivoglia..>
- [12] *DMOZ*. URL: <https://web.archive.org/web/20170317083619/http://www.dmoz.org/>.
- [13] *Dribbble*. URL: <https://dribbble.com/>.
- [14] *Facebook*. URL: <https://www.facebook.com/>.
- [15] *Fiverr*. URL: <https://it.fiverr.com/>.
- [16] *Futurepedia*. URL: <https://www.futurepedia.io/>.
- [17] *General Website Statistics*. URL: <https://www.forbes.com/advisor/business/software/website-statistics/>.
- [18] *Google Analytics*. URL: <https://marketingplatform.google.com/intl/it/about/analytics/>.
- [19] *Insanely Useful Websites*. URL: <https://www.insanelyusefulwebsites.com/>.
- [20] *Instagram*. URL: <https://www.instagram.com/>.
- [21] *LinkedIn*. URL: <https://it.linkedin.com/>.
- [22] *Meta Business Suite*. URL: <https://it-it.facebook.com/business/tools/meta-business-suite>.
- [23] *Microsoft Teams*. URL: <https://www.microsoft.com/it-it/microsoft-teams/free?market=it>.
- [24] *Modelli non anemici*. URL: <https://www.baeldung.com/java-anemic-vs-rich-domain-objects>.
- [25] *Monday*. URL: <https://monday.com/lang/it>.
- [26] Laurence Moroney. «Cloud Storage for Firebase». In: *The Definitive Guide to Firebase: Build Android Apps on Google's Mobile Platform*. Berkeley, CA: Apress, 2017, pp. 73-92. ISBN: 978-1-4842-2943-9. DOI: 10.1007/978-1-4842-2943-9\_4. URL: [https://doi.org/10.1007/978-1-4842-2943-9\\_4](https://doi.org/10.1007/978-1-4842-2943-9_4).
- [27] *Ncode*. URL: <https://www.ncode.it/tool-directory>.
- [28] *OAuth 2.0*. URL: <https://oauth.net/2/>.
- [29] *OpenID Connect*. URL: <https://openid.net/developers/how-connect-works/>.

- [30] *Pinterest*. URL: <https://www.pinterest.it/>.
- [31] *PostgreSQL*. URL: <https://www.postgresql.org/about/>.
- [32] *Prisma*. URL: <https://www.prisma.io/docs/concepts/overview/what-is-prisma>.
- [33] *Product Hunt*. URL: <https://www.producthunt.com/>.
- [34] *Routing di SvelteKit*. URL: <https://kit.svelte.dev/docs/routing>.
- [35] *Skeleton UI*. URL: <https://www.skeleton.dev/docs/introduction>.
- [36] *Skype*. URL: <https://www.skype.com/it/>.
- [37] Sofia Stamou et al. «Classifying Web Data in Directory Structures». In: *Frontiers of WWW Research and Development - APWeb 2006*. A cura di Xiaofang Zhou et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 238–249. ISBN: 978-3-540-32437-9.
- [38] *SvelteKit*. URL: <https://kit.svelte.dev/docs/introduction>.
- [39] *Tailwind CSS*. URL: <https://tailwindcss.com/>.
- [40] *Toools Design*. URL: <https://www.toools.design/>.
- [41] *Trello*. URL: <https://trello.com/>.
- [42] *Udemy*. URL: <https://www.udemy.com/it/>.
- [43] *Upwork*. URL: <https://www.upwork.com/>.
- [44] *What is a framework?* URL: <https://www.bocasay.com/what-is-development-framework/#:~:text=A%20development%20framework%20is%20a,as%20database%20management%2C%20user%20input>.
- [45] *Yahoo! Directory*. URL: [https://en.wikipedia.org/wiki/Yahoo!\\_Directory](https://en.wikipedia.org/wiki/Yahoo!_Directory).
- [46] *Zoom*. URL: <https://zoom.us/it>.