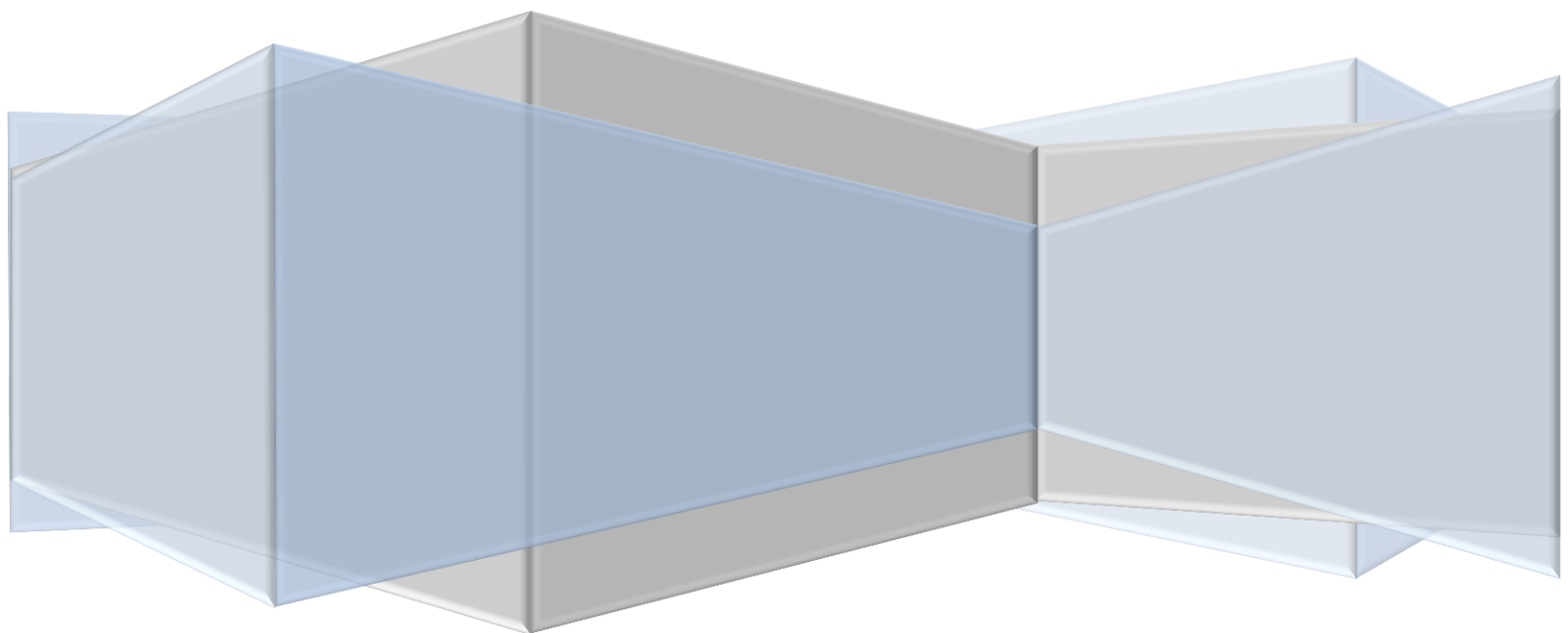


Aireem

Offrire prestazioni online

Andrea Altomare – 5^A I.T.I.S. "G. Marconi" Jesi

A.S. 2016/2017



"Nel mondo liquido-moderno la solidità delle cose, così come la solidità dei rapporti umani, tende a essere considerata male, come una minaccia: dopotutto, qualsiasi giuramento di fedeltà e ogni impegno a lungo termine (per non parlare di quelli a tempo indeterminato) sembrano annunciare un futuro gravato da obblighi che limitano la libertà di movimento e riducono la capacità di accettare le opportunità nuove e ancora sconosciute che (inevitabilmente) si presenteranno. La prospettiva di trovarsi invischiati per l'intera durata della vita in qualcosa o in un rapporto non rinegoziabile ci appare decisamente ripugnante e spaventosa."

- Zygmunt Bauman

INDICE

<u>INTRODUZIONE</u>	3
<u>GIG ECONOMY</u>	5
<u>AIREEM</u>	6
➤ Envisioning.....	6
• Mission.....	7
• Vision.....	7
• Valori.....	7
➤ Branding.....	8
• Naming.....	8
• Logo.....	9
• Colori.....	9
➤ Dettagli tecnici (Doumentazione).....	12
• Analisi delle funzionalità.....	12
• Schema E/R – Database.....	15
- Analisi schema E/R.....	16
• Schema logico – Database.....	17
• Ambiente di sviluppo.....	24
• Struttura del sito.....	25
• Mockup.....	26
- UX Design.....	26
- UI Design.....	26
• Linguaggi utilizzati.....	32
• Architettura.....	33
• Analisi parti di codice.....	34
• Hosting.....	40
➤ Manuale utente.....	41
<u>CONCLUSIONI</u>	48
<u>FONTI</u>	48

INTRODUZIONE

La scelta di esordire con una citazione dell'illustre sociologo da poco scomparso Zygmunt Bauman si è resa quanto mai necessaria, dal momento in cui i mutamenti che la nostra civiltà sta affrontando sono di una portata talmente imponente che solo grazie all'aiuto di grandi menti che ci guidano saremo in grado di apprezzarli. Proprio la sua concezione della società liquida può permettere di comprendere, se non pienamente, almeno le dinamiche principali che caratterizzano il nostro tempo: quello nel quale non esistono più strutture istituzionali efficaci né confini politici ben precisi, quello in cui ogni realtà assume i propri caratteri nella forma di ciò in cui è contenuta.

Tale idea trova oggi riscontro per molti ambiti che da sempre hanno avuto un ruolo fondamentale nella storia umana: dalla politica alla concezione della nazione, dalla religione al modo di porsi verso, in generale, quelli che sono stati i baluardi della civiltà fino ad oggi, la cui solidità è ora messa in discussione. Quello che però maggiormente interessa ai fini della prefazione a questo progetto di tesi è legato alla trasformazione dei modelli dell'economia classica, prevalenti dalla seconda metà del Settecento a gran parte del Novecento, che si trovano a doversi adattare non solo alle esigenze di un pubblico sempre più evoluto sul piano dei mezzi che ha a disposizione, ma anche a quelle di un mercato oramai non più dominato dalle vecchie e limitanti logiche di produzione e consumo industriale, bensì da altre iniziate proprio in seguito alla rivoluzione avvenuta negli ultimi decenni del XX secolo.

Dagli anni '80 la società ha conosciuto un'accelerazione dello sviluppo tecnologico che ha permeato le vite di ogni individuo, portando ai radicali cambiamenti nell'assetto economico e sociale di cui oggi possiamo osservare l'importanza.

Dall'affidamento alla semplice forza lavoro costituita da operai che svolgevano compiti ripetitivi si è passati ad un'organizzazione tecnocentrica nella quale automi e sistemi digitali di elaborazione giocano un ruolo fondamentale. Le società, perlomeno quelle più avanzate, hanno creato ed iniziato ad avvalersi di nuovi modelli produttivi e gestionali con i quali è stata possibile una notevole riduzione dei costi a fronte di un incremento del rendimento. L'organizzazione del lavoro poteva ora essere effettuata in automatico e in maniera efficace tramite l'ausilio di un sistema informatico, e molte delle operazioni in precedenza svolte dagli operai nelle catene di montaggio potevano dunque diventare appannaggio di efficienti automi costruiti su misura. Ciò ha sì comportato la fine di alcune occupazioni, ma anche la nascita di nuove posizioni aziendali per le quali l'ottenimento dei relativi incarichi avrebbe richiesto nuove competenze proprio nell'ambito digitale.

Dall'inizio del nuovo millennio si è assistito ad una repentina realizzazione dell'economia di scala riguardante i prodotti tecnologici: Internet e i principali mezzi informatici del tempo hanno via via visto una drastica riduzione dei costi ed un sensibile aumento della domanda, determinando una vasta diffusione degli stessi, negli anni a venire.

Ciò, congiuntamente alla crescente sostituzione della forza lavoro umana nelle industrie, ha condotto allo spostamento dell'asse di sviluppo dell'economia su una costante terziarizzazione: mercato del lavoro e della produzione sono pertanto maggiormente orientati verso il fornimento di servizi ai clienti piuttosto

che alla creazione di beni di consumo materiali. Avvalendosi delle nuove tecnologie, tale settore economico viene talvolta indicato come *"terziario avanzato"* o *"quaternario"*.

La diffusione, la facilità di utilizzo e le potenzialità offerte dai nuovi strumenti dell'era digitale sono tali che è possibile affermare che, ad oggi, in parte, **i lavoratori posseggano i mezzi di produzione**. Attualmente, la concezione ed il modello economico basato sulla storica dicotomia padrone/operaio (o, meno anacronisticamente rispetto alla nostra contemporaneità, imprenditore/impiegato) sono messi in discussione proprio dalla diffusione dei nuovi strumenti.

Venendo messa in crisi la dipendenza del lavoratore dai mezzi posseduti dagli industriali, crolla il sistema basato sulla staticità dei luoghi e dei tempi operativi, a favore di maggiore dinamicità, flessibilità e libertà: senza la proprietà esclusiva delle risorse, precisi luoghi dove adempiere ai propri compiti ed orari precisi e determinati, nella quasi totalità delle professioni, non possono essere imposti.

Le trasformazioni non solo economiche, ma anche sociali e culturali associate al rapido sviluppo tecnologico alla fine del XX secolo costituiscono il fenomeno della **new economy**.

Vari sono i modelli economici ed i mercati nati con la new economy, è doveroso soprattutto citare il *"telelavoro"* come nuovo paradigma che, tutt'oggi, sta mano a mano trovando una sempre più larga adozione da parte delle imprese, soprattutto quelle coinvolte proprio nel campo dell'innovazione. Questo nuovo modo di concepire il lavoro prevede infatti che l'impiegato non debba necessariamente recarsi nella sede eventualmente messa a disposizione dall'azienda nella quale tradizionalmente si svolgerebbero le mansioni, bensì che possa eseguire i propri compiti e collaborare con i colleghi dalla propria abitazione, tramite l'utilizzo di appositi dispositivi e mezzi di comunicazione (quali computer, rete Internet, smartphone, ecc...).

Le nuove forme di commercio che hanno visto la luce in questo ultimo decennio stanno scardinando i principi del capitalismo classico (sul quale sono state più o meno basate le economie degli stati maggiormente avanzati d'Occidente), i quali vogliono gli imprenditori focalizzati più sulla produzione e la vendita di beni (ed il loro conseguente possesso da parte dei singoli consumatori): ora l'asse si sta spostando sempre di più verso un contesto che predilige la condivisione di mezzi e servizi piuttosto che il diretto possesso di ciò di cui si ha bisogno, tale tipo di economia è nota come *"sharing economy"*. Essa prevede, appunto, che beni e strumenti di vario tipo possano essere usati dalle persone che non ne sono dotate direttamente ma che pagano il prezzo proprio alla fruizione degli stessi. Se, dunque, si ha bisogno di un'automobile per raggiungere una certa destinazione, senza acquistarne una, si può ricorrere ad uno dei molti servizi di *car sharing* esistenti e pagare solamente per il tragitto percorso, in base, come è ovvio, a precisi criteri dettati dall'azienda fornitrice del servizio (tempo impiegato, distanza, usura, ecc...).

La rivoluzione digitale non coinvolge, come accennato in precedenza, solo i beni materiali concretamente utilizzabili, ma anche i servizi del terziario la cui esistenza è determinante nella nostra avanzata società. *"Gig economy"* è proprio la denominazione di questo nuovo modello economico.

GIG ECONOMY

Il paradigma dell'*economy on demand* sta affermandosi in questi ultimi tempi grazie alla sua efficacia nel rispondere adeguatamente alla sempre più grande e rapida domanda di prestazioni da parte delle persone.

In una società basata sul settore terziario è inevitabile che il nucleo sul quale ha le proprie fondamenta il sistema economico vigente sia costituito proprio dal mercato dei servizi.

Il fulcro, in questo caso, risiede nell'offerta di prestazioni occasionali su richiesta (*on demand*), mirate ad assolvere gli specifici bisogni di una persona proprio quando ne ha la necessità. Anche in questo caso, essendo l'intero sistema decentralizzato ed i mezzi telematici, che si pongono come intermediari, a disposizione di tutti, è possibile maggiore flessibilità degli orari lavorativi i quali permettono non solo agli "impiegati" di operare quando meglio credono, ma soprattutto ai clienti di ricevere il servizio che desiderano potenzialmente in ogni momento della giornata, notte compresa.

L'*economia del lavoretto* non è però esente da problemi. Tali questioni sono poste in essere specialmente quando un'eccessiva deregolamentazione del mercato è causa di forte instabilità tra i lavoratori del settore. Essi infatti non solo sono attualmente privi delle tutele di cui potevano godere gli impiegati del "vecchio" modello, ma sono anche soggetti ad una serrata concorrenza che, sebbene abbia il potere di garantire idealmente il miglior servizio possibile ai clienti, costringe i "dipendenti" a doversi impegnare esclusivamente nelle proprie mansioni, togliendo di fatto il tempo a tutte le altre attività che, normalmente, riempiono la vita di ogni persona (quelle relative alla casa, alla famiglia, agli hobby, ecc...).

L'estrema flessibilità portata da questo nuovo paradigma è quindi in realtà da considerare come arma a doppio taglio: a beneficio di chi ha la possibilità di svolgere compiti legati maggiormente all'area intellettuale, ma a svantaggio di chi non ha altra scelta se non quella di cimentarsi nei nuovi "mestieri" senza la possibilità di affermare i propri diritti di lavoratore.

Le contraddizioni del sistema liberista non possono dunque dirsi superate nemmeno oggi, occorre pertanto un giusto compromesso fra interventismo statale e deregolamentazione del mercato in modo da garantire sì la libertà e flessibilità rese disponibili dalle nuove tecnologie, ma anche le giuste tutele ai chi si impegna nel lavoro in tali settori, così che a subirne le conseguenze non siano le persone senza i mezzi per potersi elevare ad una migliore condizione sociale.

AIREEM

Il mio proposito era quello di sviluppare uno strumento che potesse essere d'aiuto nel fornire un servizio la cui domanda di mercato non era stata ancora soddisfatta. Effettuando varie ricerche sono approdato alla conclusione che l'idea migliore sarebbe stata quella di permettere lo svolgimento non di una sola determinata prestazione utile, bensì di tutte quelle di cui le persone hanno bisogno. Contando sull'autoregolamentazione del mercato, ho riposto fiducia in chi dovrebbe *offrire* tali servizi, ponendo l'accento e il centro focale dello strumento da ideare non tanto sulle caratteristiche di una singola opera ma sulla possibilità di mettere a disposizione le proprie capacità per qualunque tipo di mansione, indipendentemente dalla qualifica degli offerenti, i quali potrebbero dunque candidarsi per più lavori venendo valutati direttamente dai clienti (in modo che il sistema di valutazione funga proprio da *mano invisibile* regolatrice del mercato).

La semplificazione delle logiche di richiesta-fornimento di servizi a vantaggio sia dei "produttori" che dei consumatori mi ha definitivamente spinto all'impegno nella creazione di uno strumento informatico che possa quindi porsi proprio come mediatore tra le due parti, anche nell'ottica di una società sempre più avanzata dal punto di vista tecnologico, veloce ed esigente.

Ho dunque cominciato a sviluppare l'idea partendo dalle fondamenta che definiscono un'impresa o una startup. Il progetto non è infatti collegato al solo prodotto finale ma è da collocarsi all'interno di un più ampio contesto che ha per motore una precisa *missione* ed una particolare *visione* del futuro.

In questo processo è stato molto utile lo stage formativo che ho svolto presso il **Gruppo Loccioni** durante l'estate di fine quarto anno scolastico (2015/2016). Vi ho partecipato come membro del progetto "*Classe Virtuale*" (precisamente ho ricoperto il ruolo di coordinatore del team di sviluppo informatico), edizione 2016: un'idea che vede alcuni ragazzi frequentanti le classi IV degli Istituti Tecnici della zona collaborare per realizzare una proposta progettuale dell'impresa. Durante le giornate formative sono state spiegate, a me e gli altri ragazzi, le principali nozioni sulla comunicazione aziendale, in particolare tutto ciò che riguarda l'immagine che un'impresa vuole dare all'esterno e all'interno di se stessa, in modo da promuovere la condivisione di una visione e di valori comuni così da coinvolgere maggiormente tutti i collaboratori: ciò permette di raggiungere risultati significativi sia in fase di produzione che di vendita, avendo ben chiaro in mente ciò che l'azienda produce, come effettua il proprio lavoro, e la qualità dei propri beni.

ENVISIONING

Il processo che porta a stabilire quali debbano essere le priorità sul lungo termine di un'impresa e lo scenario futuro che si vuole contribuire a creare è chiamato "*envisioning*".

Tale passo è di fondamentale importanza poiché definisce l'*anima* dell'azienda e ne orienterà, insieme al modello di business, gli obiettivi commerciali. Esso ha quindi un ruolo di estrema importanza specialmente per quanto riguarda i fini strategici, la cui scelta è prerogativa della dirigenza.

L'envisioning è composto da tre pilastri principali:

- **Mission:** è lo scopo ultimo dell'azienda, il motivo per il quale esiste. È anche il tratto distintivo che permette di differenziarla da tutti gli altri *competitor* presenti sul mercato.
- **Vision:** è la proiezione di uno scenario futuro, la prospettiva di ciò che sarà la propria impresa ed il contesto in cui sarà calata. La visione è coerente con gli ideali promossi dagli imprenditori ed incentiva all'azione.
- **Valori:** è l'insieme di scopi e principi che guidano l'agire della società, sono parte della "mentalità" aziendale. Essi definiscono l'approccio al business, all'ambiente e agli stakeholder, sono le fondamenta sulle quali deve poggiare l'operato dell'impresa.

Ho quindi determinato il **core purpose** di ciò che avrei realizzato in seguito, stabilendo gli elementi fondamentali riportati sopra.

MISSION

La "missione" di Aireem rispecchia proprio lo scopo per il quale è stata poi creata:

"Permettere a qualunque persona di usufruire di ogni tipo di servizio nei momenti e nei luoghi desiderati, tali prestazioni saranno offerte proprio dalle persone, in totale libertà."

VISION

La "visione" del futuro è congrua a quanto permesso dalla tecnologia sviluppata.

"Un mercato aperto e decentralizzato, nel quale lo scambio di prestazioni occasionali sia il motore dell'economia della società."

SLOGAN

Essendo comune fra le realtà aziendali formulare slogan universalmente riconosciuti e che siano anche in grado di trasmettere efficacemente il messaggio che tali imprese vogliono dare di loro stesse, ho riassunto missione e visione di Aireem in un unico motto:

"Any service at your fingertips"

VALORI

I valori di Aireem sono legati alla visione della tecnologia rispetto all'avvenire ed il suo ruolo nella società, nonché alle ormai necessarie caratteristiche che deve possedere un qualsiasi nuovo strumento immesso sul mercato digitale.

Sono:

- Semplicità
- Velocità
- Interconnessione
- Affidabilità
- Libertà

Lo sviluppo di Aireem è proprio stato effettuato secondo questi punti cardine. Dall'interfaccia utente, all'elaborazione delle informazioni, a come sono strutturate le pagine, fino alle possibilità offerte, tutto è stato realizzato cercando di aderirvi il più possibile.

BRANDING

Con *“branding”* si intende l'insieme di attività strategiche e operative relative alla costruzione e alla gestione della marca.

Il brand è, dunque, “l'identità di un prodotto o un'azienda, la descrizione sintetica di un pacchetto di valori su cui i consumatori possono contare”.

Concretamente, con il branding si costruisce l'immagine dell'impresa: ciò che le persone immagineranno quando penseranno ad una certa azienda dipende proprio da quale messaggio essa sia riuscita a veicolare tra il pubblico.

Tale processo si avvale di svariate metodologie di marketing, puntando talvolta anche sulla psicologia dei potenziali consumatori (o di quelli che lo sono già, nell'ottica della fidelizzazione dei clienti). Nel caso di grandi campagne di comunicazione è molto importante il ruolo svolto dallo storytelling: l'arte di “raccontare” la realtà del marchio in modo da suscitare determinate emozioni, le quali verranno quindi associate a brand stesso. È comprensibile dunque come tali tecniche possano risultare estremamente efficaci nel creare maggiore interesse intorno ad un prodotto (o un'azienda) e, conseguentemente, maggior profitto.

NAMING

Prima di lavorare sul marchio vero e proprio, ho dovuto trovare un nome allo strumento che stavo per creare. Questa operazione è chiamata *“naming”* ed è probabilmente la parte più importante di tutto il processo di branding: al nome ogni persona proietta determinate sensazioni e prospettive, inoltre un buon nome deve essere il più possibile riconoscibile e ben distinguibile dagli altri prodotti della concorrenza, e magari deve anche essere in grado di veicolare i valori che si sono stabiliti in precedenza.

Il nome non deve quindi banalmente descrivere il prodotto in sé, bensì deve stimolare psicologicamente i potenziali consumatori suscitando emozioni adeguate.

Per scegliere il nome ho tenuto in considerazione molteplici fattori, tra cui:

- **Mercato di riferimento:** essendo un prodotto informatico, mi sono focalizzato sull'elemento linguistico. L'inglese è la lingua della tecnologia nonché tra le più conosciute a livello globale, per cui la scelta è ricaduta proprio su essa.
- **Semplicità:** questo fattore è presente anche proprio fra i valori sopra descritti, per cui è stato fondamentale prenderlo in considerazione. Mi sono focalizzato quindi sulla semplicità della forma (leggibilità) e della pronuncia, in modo che fosse sì originale ma che potesse anche ricordarsi senza difficoltà.
- **Originalità:** ovviamente il nome scelto non poteva essere già esistente, per cui prima di trovare quello definitivo ho accuratamente controllato nel web che non ci fossero servizi e/o prodotti omonimi.

Tenendo in considerazione quanto scritto sopra, e i valori stabiliti in precedenza, ho optato per un nome di fantasia:

"Aireem"

Questo nome è compatto e di facile pronuncia, non contiene caratteri speciali ma ha allo stesso tempo una doppia "e" al suo interno che gli conferisce una certa originalità, in modo da non essere sterile. Comincia inoltre con la prima lettera dell'alfabeto: proprio come Apple, Atari, Amazon, sarebbe quindi fra le prime imprese a comparire in un elenco alfabetico. Infine, la prima parte: "Air" (*aria*), dovrebbe evocare un senso di leggerezza, semplicità e tranquillità, proprio ciò che chi dovesse usare lo strumento dovrebbe percepire.

LOGO

Successivamente ho progettato e realizzato il logo del prodotto. Anche questo, essendo parte fondamentale della comunicazione strategica di un'impresa, ha richiesto ricerche all'interno del mercato di riferimento e sulla psicologia riguardante determinate forme e colori.

Un logo deve innanzitutto raccontare qualcosa. La scelta è stata quella di veicolare, ancora una volta, i valori aziendali tramite un certo stile di scrittura e della geometria del disegno.

Ho utilizzato come logo proprio il nome del prodotto: *"aireem"* (con la prima lettera minuscola, per una migliore rendita grafica e per dare continuità allo stile delle altre lettere componenti il nome). Il font (lo stile del carattere) scelto è tale da consolidare le sensazioni, che si intendono suscitare già tramite il nome, legate ai principi aziendali.

Le versioni sono quattro:

- **Scritta con colori scelti:** la scritta *"aireem"* colorata secondo quanto deciso nella fase successiva (quella relativa ai colori).
- **Scritta in un quadrato con gli angoli arrotondati:** per rispondere alle esigenze di avere un logo utilizzabile come icona dell'eventuale applicazione.
- **Lettera "a" in un quadrato con gli angoli arrotondati:** per una migliore rendita nel caso di piccoli schermi o icone di ridotte dimensioni.

COLORI

La scelta dei colori del logo e, in generale, del prodotto è stata frutto di una ricerca sulla psicologia dei colori: quali sensazioni (certamente legate alla cultura occidentale e al nostro tempo) ogni colore suscita e quali messaggi veicola.

Dopo un'attenta analisi, ho optato per una tonalità di verde che tendesse anche al celeste. Le emozioni legate a queste due sfumature sono le seguenti:

- **Verde:** calma e sicurezza, può anche ispirare equilibrio e, in ultima analisi, è rilassante: esattamente come deve sentirsi chi utilizza Aireem. Talvolta è anche utilizzato proprio come colore della natura.

- **Celeste:** infonderebbe chiarezza, senso di apertura ed ambizione: proprio come dovrebbe essere chi offre un servizio tramite il prodotto. Inoltre, essendo vicino al blu, può suscitare un senso di affidabilità.

Qui sotto, uno schema riassuntivo dei colori e le sensazioni a cui sono legati, che non ha comunque la pretesa di essere esaustivo (tale disciplina non può infatti essere "ridotta" ad un semplice schema).



Figura 1 - Psicologia dei colori

I loghi, progettati secondo quanto scritto sopra, sono i seguenti:

aireem



Li ho realizzati utilizzando il programma di elaborazione grafica [Adobe Photoshop CC 2015](#). Di seguito una schermata del progetto in fase di lavorazione:

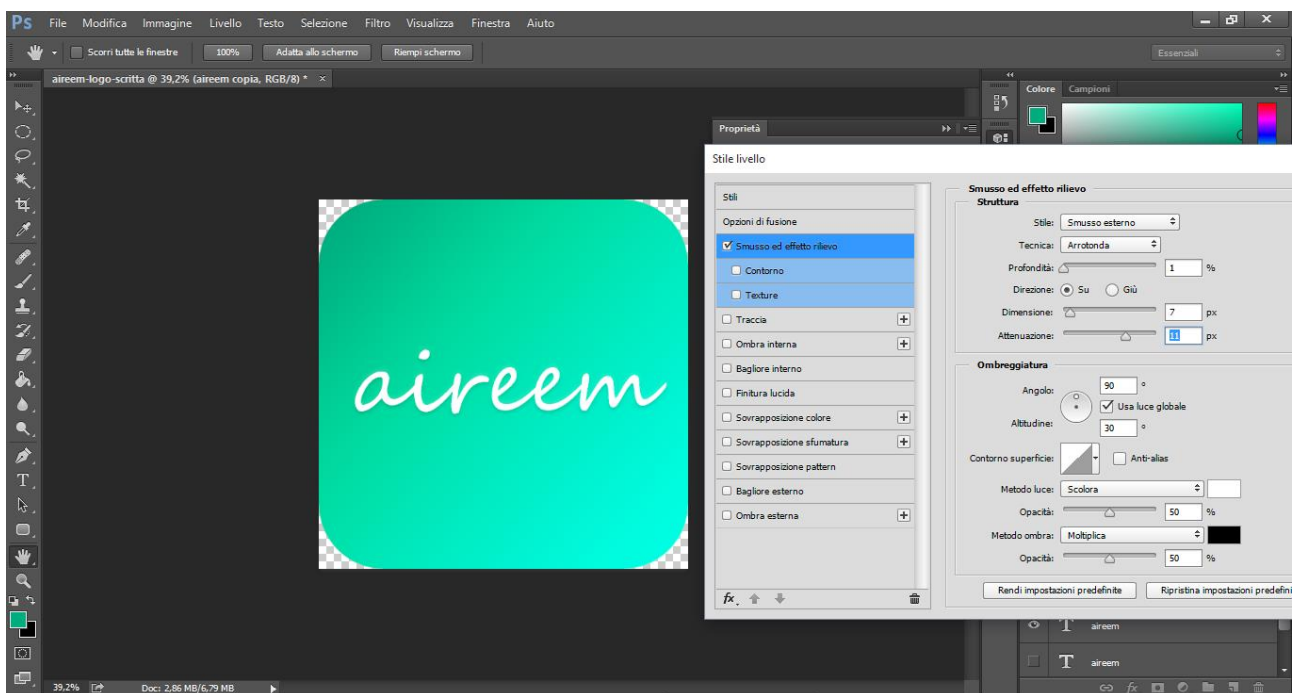


Figura 2 – Realizzazione del logo

DETTAGLI TECNICI (DOCUMENTAZIONE)

ANALISI DELLE FUNZIONALITÀ

Di seguito è riportata l'analisi effettuata nella fase iniziale di progettazione di Aireem. Le funzionalità approfondite di seguito non sono state tutte implementate: il progetto è infatti da intendersi in una più ampia ottica di sviluppo e miglioramento contando anche sulle competenze di altri esperti nei vari settori che l'applicazione coinvolge. Il prodotto finale è quindi un prototipo funzionante ma non maturo per un eventuale lancio sul mercato.

Generalità

La piattaforma dovrà permettere a varie persone di registrarsi e di **proporsi per dei lavori** che potrebbero servire alla gente comune (ad esempio: imbianchino, parrucchiere, elettricista, meccanico, ecc...) dunque una volta iscritto, un'utente, fondamentalmente, avrà due opzioni di utilizzo della piattaforma. Può proporsi per un servizio: per esempio imbianchino, **a domicilio o all'indirizzo immesso in fase di registrazione**, ed indicare il **costo della prestazione** offerta.

L'offerente prestazione può decidere di lavorare nelle **fasce orarie** che immetterà nel momento della creazione dell'offerta (vi possono essere fasce orarie differenti per lavori differenti, infatti **ogni utente può proporsi per più servizi**).

Oppure **l'utente può cercare chi offre un certo tipo di prestazione**: ad esempio "estetista" e scegliere in base al prezzo, alla vicinanza, all'affidabilità dell'offerente la prestazione (**il rating è relativo alla prestazione offerta**, NON all'utente stesso, perché una persona potrebbe eccellere come parrucchiere ma essere poco bravo come imbianchino, per cui bisogna votarla in base a ciò che direttamente sta compiendo per l'utente richiedente).

Rating

Il **rating** dovrà essere controllato e non si potrà votare liberamente: un utente potrebbe altrimenti imbrogliare e votare sempre a proprio favore o a sfavore di un altro. Quindi per dare la propria valutazione si dovrà avere un codice ottenibile dall'utente fruitore solo pagando l'offerente del servizio in questione; ciò sarà però possibile solo se si gestirà il pagamento tramite la piattaforma, altrimenti il rating sarà libero, con l'unica limitazione di non poter votare se stessi o più volte una stessa offerta (la modifica del voto sarà comunque permessa).

Metodi di pagamento

Ogni **metodo di pagamento** è relativo ad uno specifico utente: egli potrà scegliere con quale tipo di carta pagare, oppure, se offre servizi, il **conto corrente sul quale ricevere il denaro** (è permesso anche PayPal). I metodi di pagamento verranno memorizzati sul DB (come su Amazon) in modo che gli utenti non siano costretti a reinserirli ogni volta.

Segnalazioni

Un utente può anche segnalare (una volta, scegliendo più motivi) un altro utente (*Segnala abuso*) in modo che si limiti almeno il rischio di truffa da parte degli offerenti delle prestazioni.

Prestazioni

Le **prestazioni** potranno essere inserite in un'apposita casella di testo. Ciò che sarà inserito in modo **personalizzato dall'utente verrà registrato nel DB**.

Una prestazione creata da un utente *può in qualunque momento essere modificata o cancellata*.

Le prestazioni saranno **divise per macro categorie** (medica, casalinga, tecnica, legale, babysitting-dogsitting, ecc...) e poi scelte singolarmente. Ciò, sebbene in seguito si sia ritenuto più opportuno permettere all'utente di digitare il nome del servizio cercato in una casella di testo, in modo da semplificare le operazioni di ricerca.

Appuntamenti

Un utente può prenotare un appuntamento per una determinata offerta di prestazione da parte di un altro utente. L'appuntamento *può essere in qualunque momento modificato o cancellato dal cliente (prevedere, in caso si gestisca il pagamento, un acconto di garanzia da versare all'offerente della prestazione per il quale ci si è prenotati in modo da ripagare il danno procurato cancellando o modificando radicalmente, ad esempio cambiando la data dell'incontro, l'appuntamento, e da disincentivare comportamenti scorretti da parte dei clienti verso gli offerenti)*. **Colui che offre ha il diritto di cancellare una prestazione** (il cliente dovrà quindi essere avvisato).

Dato che il sistema di appuntamenti avrebbe richiesto, per una più efficace gestione degli impegni degli offerenti che avrebbero deciso di avvalersene, l'uso di un calendario elettronico generale (come "Calendar", offerto da Google), si è deciso per il momento di non permettere la prenotazione. Configurando di fatto l'applicazione come una sorta di bacheca elettronica.

Si è anche pensato ad un sistema di *matching* che permetta ai clienti, tramite un apposito pulsante, di mettersi in contatto con chi offre le prestazioni d'opera tramite software di messaggistica istantanea che hanno già installato nei propri dispositivi (WhatsApp, Telegram, ecc...). Tale funzionalità potrebbe quindi essere aggiunta in seguito.

Utenti

Gli utenti potranno essere **persone private** o **aziende**.

Tutti e due dovranno inserire un'**email identificativa** e una **password per accedere al proprio account**, ma se è un'azienda dovrà immettere anche la **Partita Iva** (è valida in tutta l'Unione Europea), la quale lo collegherà all'entità azienda.

Ogni **persona** dovrà **inserire le proprie generalità** e, opzionalmente, il link al proprio **profilo LinkedIn**. Potrà indicare anche **contatti** come il numero di **cellulare** e di **telefono**, e il proprio eventuale **sito web**.

Ogni **azienda** dovrà **immettere i dati di interesse per la propria identificazione** e i **contatti per il pubblico**; sarà disponibile anche lo spazio per inserire il relativo **profilo LinkedIn** e il proprio **sito web**.

Notifiche

Quando verranno **prenotati nuovi appuntamenti, modificati quelli esistenti, o cancellati**, l'utente che compirà una di queste azioni **genererà automaticamente una notifica**, la quale **verrà inviata all'altro utente di interesse** (coinvolto nell'appuntamento). Succederà la stessa cosa *quando un offerente cancellerà un appuntamento* che un cliente ha preso con lui.

Una **notifica** sarà strutturata in modo da poter vedere **nome e cognome** (o **username**, in alternativa) *dell'utente generatore della stessa*, la sua **foto profilo (propic)**, una **sintetica descrizione** dell'azione che ha compiuto, e la **data e ora** in cui la notifica è stata generata.

Messaggi

Si potrebbe rendere necessario un **sistema di messaggistica istantanea** fra due utenti. In questo caso il **messaggio** sarà caratterizzato (oltre che dagli **ID degli users coinvolti**) dal **corpo del testo** e da un'**immagine** (poiché un utente potrebbe inviare un'immagine con un messaggio), vi saranno poi anche

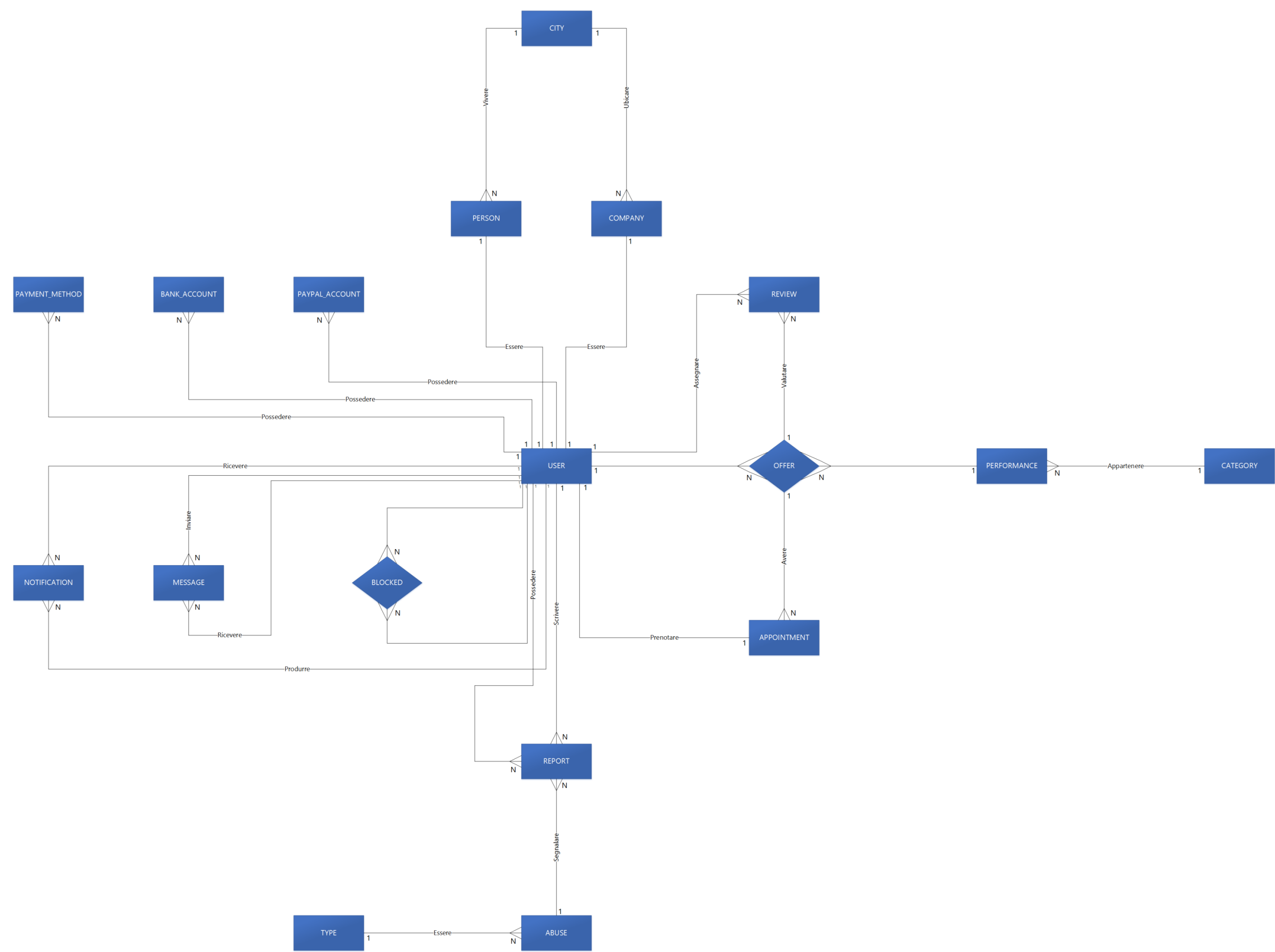
la data e l'ora di invio, consegna e lettura del messaggio (in modo da *informare anche l'utente mittente della comunicazione della consegna/lettura da parte del destinatario*).

Utenti bloccati

Sarà possibile bloccare più utenti semplicemente conoscendone l'username. (Una nuova entità si è resa necessaria a tale scopo).

SCHEMA E/R – Database

Dovendo realizzare un database, ho cominciato la progettazione partendo dallo **schema concettuale**, ricavato proprio dall'analisi della realtà di riferimento effettuata in precedenza.



Analisi schema E/R

Le entità individuate tramite lo schema E/R rispondono alle esigenze evidenziate dall'analisi del problema.

La parte principale del modello è costituita dalle entità **USER**, **OFFER**, **PERFORMANCE**, e **APPOINTMENT**. Un user può offrire più performance (es: parrucchiere ed estetista), ed una performance può generalmente essere offerta da più user (vi possono essere più persone che offrono il servizio di idraulico), dunque è presente un'associazione N:M che va semplificata creando una nuova entità (**OFFER**, appunto), la quale sarà costituita da chiavi esterne per riconoscere le due entità collegate, inoltre vi saranno altri attributi che descriveranno l'offerta di performance nel particolare per un determinato user (ad esempio, due user possono offrirsi come idraulici, ma uno al costo di 10€/h, mentre l'altro a 15€/h). Un user può inoltre prenotare un appuntamento per una certa offerta di prestazione, ad un certo orario. Tutte le caratteristiche inerenti l'appuntamento in sé, sono appunto comprese nell'entità **APPOINTMENT**. L'associazione fra user e appointment è di 1:1 poiché un certo appuntamento è relativo ad un solo user, ed un user può prenotare un certo appuntamento (con una data ed una ora precise) solo UNA volta. Mentre una certa offerta di performance (**OFFER**), può avere più appuntamenti (1:N) poiché un user può lavorare in più orari diversi in una sola giornata. Tutti gli appuntamenti presi da un singolo utente possono essere ricavati tramite una selezione di tutte le occorrenze della relazione **APPOINTMENT** che come chiave esterna che identifica l'utente prenotante ha proprio il valore dell'ID dell'utente per il quale li si stanno cercando.

PERFORMANCE ha un'associazione N:1 con l'entità **CATEGORY** poiché può appartenere ad un'unica categoria, mentre una categoria (ad esempio: Ristrutturazioni) può avere più performance.

Un'offerta può inoltre essere valutata. Tramite l'entità **REVIEW** si memorizzeranno i voti ed i commenti dei singoli utenti relativi ad una singola offerta. Dunque vi sarà un'associazione 1:N tra user e review poiché un utente può scrivere più recensioni, mentre una determinata recensione può essere scritta da un solo utente. Una recensione è relativa ad una sola offerta, mentre un'offerta può avere più recensioni (per cui vi è fra le due entità un'associazione 1:N).

È fondamentale distinguere gli user che sono persone da quelli che sono aziende. A tale proposito sono state create due entità: **PERSON** e **COMPANY**. C'è un'associazione 1:1 sia fra user e person che fra user e company poiché le due entità identificano rispettivamente UNA determinata persona e UNA determinata azienda, per cui un utente del sistema può essere UNA certa persona o UNA certa azienda, e una persona o un'azienda possono essere relative solo ad UN certo user.

Dato che fra i vari attributi figurano anche quelli relativi all'abitazione o all'ubicazione della persona o dell'azienda, si è resa necessaria la creazione dell'entità **CITY** in modo da evitare la ridondanza delle informazioni relative ad una singola città (ad esempio: CAP, provincia e stato di appartenenza). Dunque fra person e city, come fra company e city, vi è un'associazione di tipo N:1 perché una persona ha la residenza in un solo centro abitato, così come un'azienda (almeno, la sede principale) si può trovare in una sola città; mentre in una città si possono trovare più persone e più aziende.

Dato che si vogliono gestire anche i pagamenti fra gli utenti, sono state realizzate tre apposite entità che memorizzano i metodi di pagamento (con i quali pagare), i conti bancari (sui quali ricevere il denaro), ed i conti PayPal (con i quali inviare o ricevere soldi), chiamate rispettivamente **PAYMENT_METHOD**, **BANK_ACCOUNT**, e **PAYPAL_ACCOUNT**.

Tra queste entità ed user vi sono associazioni N:1 dato che un determinato metodo di pagamento (ad esempio, i dati di una certa carta prepagata), conto bancario o account PayPal sono relativi ad uno specifico utente, ed un utente può avere più metodi di pagamento (carte di credito, debito, prepagate, ecc...), conti bancari e account PayPal da utilizzare.

Per gestire le notifiche si è creata l'entità **NOTIFICATION**: un user può avere più notifiche, e, dato che queste sono generate proprio dagli utenti (un user può generare più notifiche ai diretti interessati, per esempio prenotando più appuntamenti), c'è una doppia associazione fra USER e NOTIFICATION di 1:N (una determinata notifica è relativa ad UN SOLO specifico utente ed è generata solo da un user).

Le conversazioni effettuate tramite l'eventuale sistema di messaggistica istantanea sono memorizzate nell'entità **MESSAGE**: analogamente alle notifiche, vi è una doppia associazione 1:N fra user e message. Un utente può infatti inviare o ricevere più messaggi, ma un messaggio è relativo solo a due specifici utenti (il mittente e il destinatario).

Gli utenti bloccati verranno memorizzati in un'entità creata semplificando l'associazione N:M fra USER e USER, ovvero un'associazione ricorsiva (un utente può bloccare più user, e un user può essere bloccato da più utenti). L'entità realizzata è **BLOCKED** e conterrà, a parte la chiave primaria, due chiavi esterne che identificano l'utente bloccante e l'utente bloccato).

Il sistema di segnalazione degli utenti è realizzato tramite tre entità: **REPORT**, **ABUSE**, e **TYPE**.

Un utente può segnalare più user, ed un user può essere segnalato da più utenti, dunque c'è, anche qui, una doppia associazione 1:N fra REPORT e USER (una singola segnalazione è infatti relativa a due soli utenti: chi segnala e chi viene segnalato).

Un report, essendo descritto principalmente da un commento (opzionale, che scrive chi segnala), è legato ad uno specifico abuso (ad esempio: costo dichiarato non veritiero, le foto ritraggono un'altra persona, contenuti inadatti, creazione di offerte false, ecc...), mentre ad un abuso possono essere collegati più report (più utenti infatti possono compiere lo stesso atto illecito, ad esempio una truffa), per cui è presente un'associazione 1:N fra ABUSE e REPORT.

Un abuso può essere di un certo tipo (ad esempio: il costo dichiarato non veritiero può essere del tipo truffa; la creazione di offerte false può essere del tipo spam, ecc...), e di un tipo possono essere più abusi specifici. Per questo motivo è stata creata l'entità TYPE, con un'associazione 1:N con ABUSE.

SCHEMA LOGICO – Database

Dal modello concettuale ho derivato le relazioni che avrebbero poi costituito le tabelle del database. Sono riportate di seguito.

Users(usrID, usrEmail, usrPassword, usrUsername, usrProPic, usrPremium, usrRegistrationDate, usrIPAddress, *usrPersonCode*, *usrP_Iva*)

Users		
Nome	Tipo	Uso
<u>usrID</u>	int	PK
usrEmail	varchar (100)	Obbligatorio
usrPassword	varchar (100)	Obbligatorio
usrUsername	varchar (100)	Obbligatorio
usrProPic	text (URL)	Opzionale
usrPremium	boolean	Obbligatorio
usrRegistrationDate	datetime	Obbligatorio
usrIPAddress	varchar(15)	Obbligatorio
<i>usrPersonCode</i>	int	FK
<i>usrP_Iva</i>	varchar (11)	FK

People(pplPersonCode, pplName, pplSurname, pplBirthdate, pplGender, pplQualification, pplOccupation, pplBio, pplAddress, pplHouseNumber, pplApartmentInfo, pplTel, pplCell, pplCV, pplLinkedin, pplWebsite, *pp/CyID*)

People		
Nome	Tipo	Uso
<u>pplPersonCode</u>	int	PK
pplName	text	Obbligatorio
pplSurname	text	Obbligatorio
pplBirthdate	date	Obbligatorio
pplGender	varchar (1)	Opzionale
pplQualification	text	Opzionale
pplOccupation	text	Opzionale
pplBio	text	Opzionale
pplAddress	text	Obbligatorio
pplHouseNumber	varchar (5)	Obbligatorio
pplApartmentInfo	text	Opzionale
pplTel	varchar (18)	Opzionale
pplCell	varchar (18)	Opzionale
pplCV	text (URL)	Opzionale
pplLinkedin	text	Opzionale
pplWebsite	text	Opzionale
<i>coCyID</i>	smallint	FK

Companies(coP_Iva, coName, coAddress, coHouseNumber, coApartmentInfo, coTel, coLinkedin, coWebsite, *coCyID*)

Companies		
Nome	Tipo	Uso
<u>coP_Iva</u>	varchar (11)	PK
coName	text	Obbligatorio
coAddres	text	Obbligatorio
coHouseNumber	varchar (5)	Obbligatorio
coApartmentInfo	text	Opzionale
coTel	varchar (18)	Opzionale
coLinkedin	text	Opzionale
coWebsite	text	Opzionale
<i>coCyID</i>	smallint	FK

Cities(cyID, cyZipCode, cyDistrict, cyState, cyCountry)

Cities		
Nome	Tipo	Uso
<u>cyID</u>	smallint	PK
cyName	text	Obbligatorio
cyZipCode	varchar (10)	Obbligatorio
cyDistrict	varchar (100)	Opzionale
cyState	varchar (100)	Opzionale
cyCountry	varchar (2)	Obbligatorio

Bank_accounts(bnkIBAN, bnkBankName, bnkHolder, *bnkUserID*)

Bank_accounts		
Nome	Tipo	Uso
<u>bnkIBAN</u>	varchar (50)	PK
bnkBankName	text	Obbligatorio
bnkHolder	text	Obbligatorio
<i>bnkUserID</i>	int	FK

PayPal_accounts(pypID, pypEmail, *pypUserID*)

PayPal_accounts		
Nome	Tipo	Uso
<u>pypID</u>	int	PK
pypEmail	varchar (100)	Obbligatorio
<i>pypUserID</i>	int	FK

Payment_methods(pmtID, pmtCardHolder, pmtCardType, pmtCardNumber, pmtCardExpDate, pmtCardSecurityCode, *pmtUserID*)

Payment_methods		
Nome	Tipo	Uso
<u>pmtID</u>	int	PK
pmtCardHolder	text	Obbligatorio
pmtCardType	text	Obbligatorio
pmtCardNumber	varchar (16)	Obbligatorio
pmtCardExpDate	date	Obbligatorio
pmtCardSecurityCode	varchar (4)	Obbligatorio
<i>pmtUserID</i>	int	FK

Currencies(curID, curCode, curDecimal, curName, *curCoID*)

Currencies		
Nome	Tipo	Uso
<u>curID</u>	smallint (3)	PK
curCode	varchar (3)	Obbligatorio
curDecimal	tinyint (1)	Opzionale
curName	varchar (50)	Obbligatorio
<i>curCoID</i>	varchar (2)	FK

Categories(catID, catName, catInfo)

Categories		
Nome	Tipo	Uso
<u>catID</u>	smallint	PK
catName	varchar (100)	Obbligatorio
catInfo	text	Opzionale

Performances(prfID, prfName, prfInfo, *prfCatID*)

Performances		
Nome	Tipo	Uso
<u>prfID</u>	smallint	PK
prfName	varchar (150)	Obbligatorio
prfInfo	text	Opzionale
<i>prfCatID</i>	smallint	FK

Offers(offID, offInfo, offWorkExperience, offSkills, offCost, offHourlyCost, offCostToBeDetermined, offHomeWorking, offMon, offTue, offWed, offThu, offFri, offSat, offSun, offStartTime, offEndTime, offCurID, offPrfID, offUserID)

Offers		
Nome	Tipo	Uso
<u>offID</u>	mediumint	PK
offInfo	text	Opzionale
offWorkExperience	tinyint	Opzionale
offSkills	text	Opzionale
offCost	decimal (12, 2)	Opzionale
offHourlyCost	boolean	Obbligatorio
offCostToBeDetermined	boolean	Obbligatorio
offHomeWorking	boolean	Obbligatorio
offMon	boolean	Obbligatorio
offTue	boolean	Obbligatorio
offWed	boolean	Obbligatorio
offThu	boolean	Obbligatorio
offFri	boolean	Obbligatorio
offSat	boolean	Obbligatorio
offSun	boolean	Obbligatorio
offStartTime	time	Obbligatorio
offEndTime	time	Obbligatorio
offCurID	smallint (3)	FK
offPrfID	smallint	FK
offUserID	int	FK

Appointments(appID, appDate, appTime, appBookingDateTime, appOffID, appUserID)

Appointments		
Nome	Tipo	Uso
<u>appID</u>	int	PK
appDate	date	Obbligatorio
appTime	time	Obbligatorio
appBookingDateTime	datetime	Obbligatorio
appOffID	mediumint	FK
appUserID	int	FK

Reviews(revID, revTitle, revRating, revComment, revDate, *revOffID*, *revUserID*)

Reviews		
Nome	Tipo	Uso
<u>revID</u>	int	PK
revTitle	text	Obbligatorio
revRating	tinyint	Obbligatorio
revComment	text	Opzionale
revDate	date	Obbligatorio
<i>revOffID</i>	mediumint	FK
<i>revUserID</i>	int	FK

Types(typID, typName, typInfo)

Types		
Nome	Tipo	Uso
<u>typID</u>	tinyint	PK
typName	varchar (50)	Obbligatorio
typInfo	text	Opzionale

Abuses(abuID, abuName, abuInfo, *abuTypID*)

Abuses		
Nome	Tipo	Uso
<u>abuID</u>	tinyint	PK
abuName	varchar (100)	Obbligatorio
abuInfo	text	Opzionale
<i>abuTypID</i>	tinyint	FK

Reports(repID, repComment, *repAbuID*, *repReportedUser*, *repReportingUser*)

Reports		
Nome	Tipo	Uso
<u>repID</u>	int	PK
repComment	text	Opzionale
<i>repAbuID</i>	tinyint	FK
<i>repReportedUser</i>	int	FK
<i>repReportingUser</i>	int	FK

Notifications(notID, notInfo, notDateTime, *notNotifierUserID*, *notUserID*)

Notifications		
Nome	Tipo	Uso
<u>notID</u>	int	PK
notInfo	text	Obbligatorio
notDateTime	datetime	Obbligatorio
<i>notNotifierUserID</i>	int	FK
<i>notUserID</i>	int	FK

Messages(msgID, msgText, msgImg, msgSeDateTime, msgDeDateTime, msgReDateTime, *msgSenderUserID*, *msgReceiverUserID*)

Messages		
Nome	Tipo	Uso
<u>msgID</u>	int	PK
msgText	text	Opzionale
msgImg	text (URL)	Opzionale
msgSeDatetime	datetime	Obbligatorio
msgDeDateTime	datetime	Opzionale
msgReDateTime	datetime	Opzionale
<i>msgSenderUserID</i>	int	FK
<i>msgReceiverUserID</i>	int	FK

Blocked(blkID, *blkBlockedUserID*, *blkUserID*)

Blocked		
Nome	Tipo	Uso
<u>blkID</u>	int	PK
<i>blkBlockedUserID</i>	int	FK
<i>blkUserID</i>	int	FK

La relazione **CURRENCIES** è stata aggiunta in un secondo momento dato che si è voluto gestire le valute dei prezzi delle offerte create dagli utenti, poiché l'utilizzo della web app deve essere possibile anche al di fuori della zona euro.

Le tabelle dello schema logico evidenziano i domini dei vari campi (ovvero il loro tipo di dato) ed i ruoli che gli attributi di ogni relazione svolgono all'interno delle stesse: **chiavi primarie**, **chiavi esterne**, **dati ad immissione obbligatoria** o **facoltativa**.

AMBIENTE DI SVILUPPO

Come ambiente sul quale sviluppare Aireem ho scelto la piattaforma web [Cloud9](#), ricca di numerose funzionalità utili ai programmatori che si cimentano nella scrittura di codice per le proprie applicazioni.

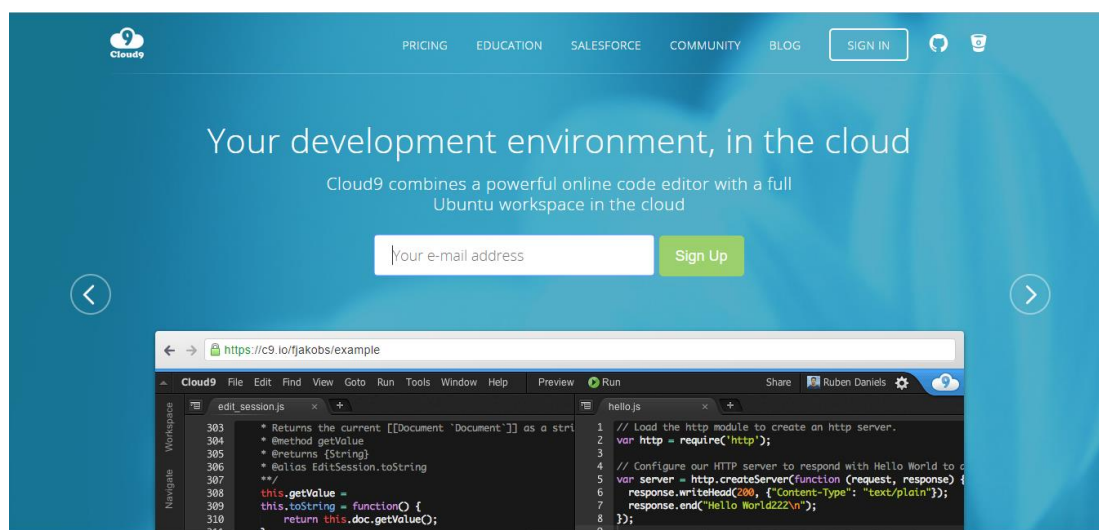


Figura 3 - Home Page Cloud9

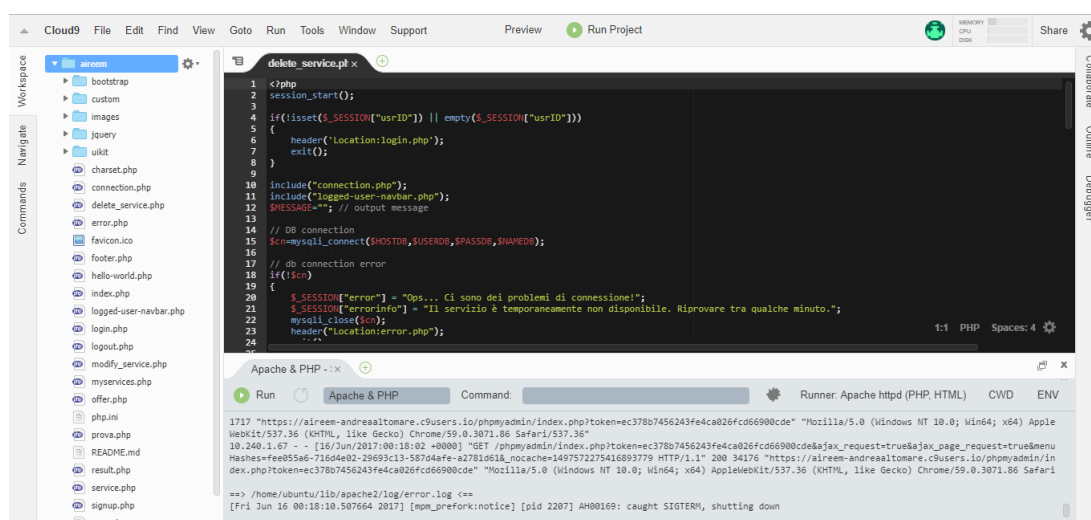


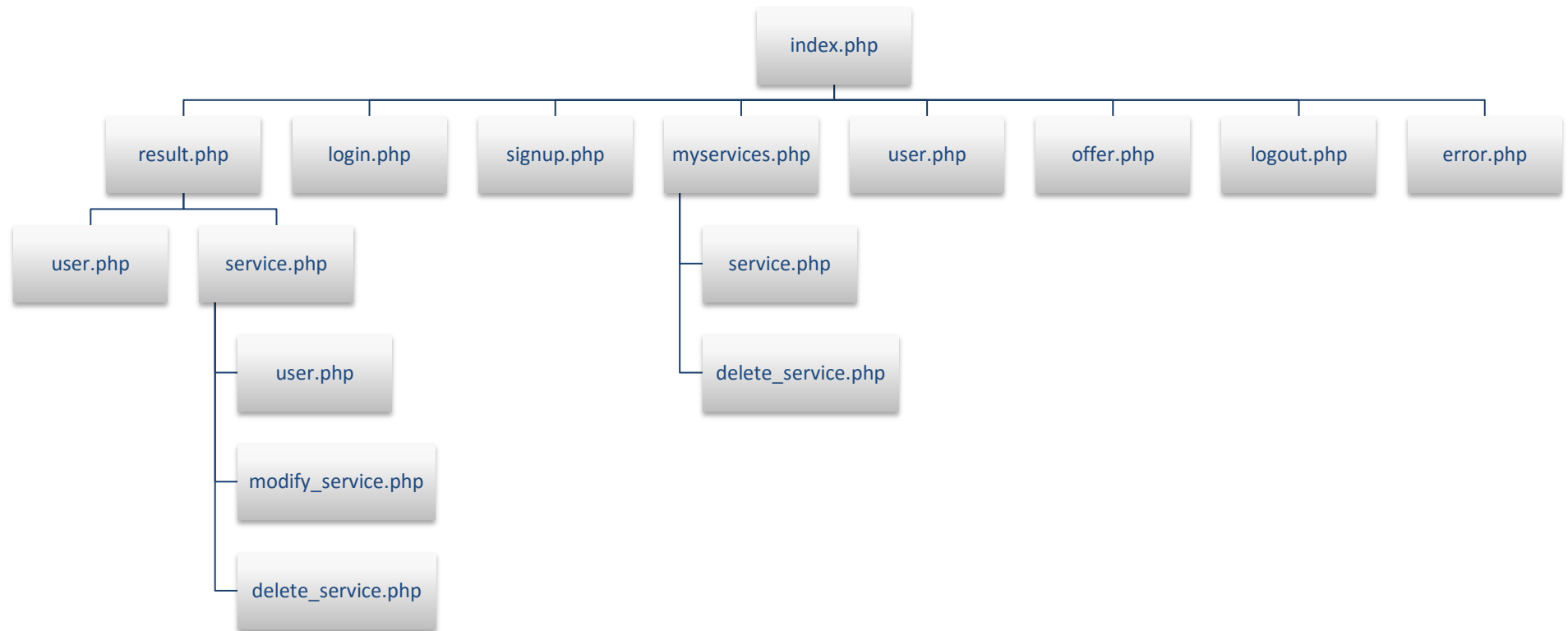
Figura 4 – Schermata ambiente di sviluppo

Questo software permette non solo di scegliere fra i maggiori linguaggi di programmazione, ma anche di personalizzare l'ambiente virtuale sul quale i propri prodotti dovranno funzionare, simulando quindi una situazione reale anche in fase di sviluppo.

Il sistema che ho configurato per Aireem, essendo una web application e dovendo essere eseguita su un server web, ha come **sistema operativo** [Ubuntu](#) (distribuzione Linux) sul quale è attivo un **web server** [Apache](#) (molto comune sul web). Il **DBMS** (*Database Management System*) impiegato per gestire il database è [MySQL](#), sempre messo a disposizione dalla piattaforma, il quale viene amministrato mediante il software [phpMyAdmin](#), scritto proprio a tale scopo. Lo **script engine**, ovvero l'interprete del codice del programma che permette il funzionamento *lato server* dell'applicazione creata, lavora con [PHP](#), che è proprio il linguaggio che ho utilizzato. Concretamente, Cloud9 si presenta quindi come una **pacchetto integrato di strumenti XAMP** ("x, Apache, MySQL, PHP", poiché come S.O. può essere scelto anche Microsoft Windows, ed è quindi multiplatforma).

STRUTTURA DEL SITO

Essendo, il prodotto finale, un sito web, è stato innanzitutto necessario definire la *struttura del sito*, ovvero una mappa che permette di capire come le pagine (visitabili dall'utente finale) che lo costituiscono siano legate tra loro e come possono essere raggiunte. Ciò si concretizza in una struttura ad albero che evidenzia, tramite i nodi, proprio i vari componenti.



MOCKUP

Una volta decisa la struttura del sito e, conseguentemente, le pagine che avrebbero dovuto comporre l'applicazione, ho progettato l'interfaccia utente.

Tale stadio è in realtà diviso in due fasi: il design dell'*esperienza utente* (UX Design) e, successivamente, quello dell'*interfaccia* (UI Design).

UX Design

Con "esperienza utente" (UX, "User eXperience") si intende *"la sommatoria delle emozioni, delle percezioni e delle reazioni che una persona prova quando si interfaccia con un prodotto o servizio."* Più semplicemente la si definisce come il grado di soddisfazione che ha l'utente nell'interagire con il sistema.

Mentre il relativo design è definito come *"l'insieme di processi atti ad accrescere la soddisfazione dell'utente migliorando l'usabilità di una pagina web, la sua facilità di utilizzo, l'intuitività e l'interazione."*

Dunque si tratta della progettazione e della cura dell'interazione che l'utente avrà con il prodotto, e lo scopo finale è quello di migliorarne l'esperienza.

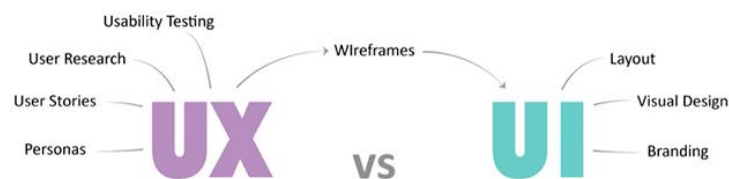
Per questo compito ho ancora una volta preso in considerazione i valori stabiliti all'inizio, in modo da effettuare una pianificazione coerente con ciò che un eventuale utente si aspetterebbe dal sito. Ho dunque puntato molto sulla semplicità e la rapidità di utilizzo, attraverso un posizionamento dei pulsanti chiaro e standard come in molti altri siti web.

UI Design

Con "interfaccia utente" (UI, "User Interface") si intende *"ciò che si frappona tra una macchina e un utente, consentendo l'interazione tra i due."* Quindi ogni software o componente hardware che media le azioni di una persona permettendole di "comunicare" con la macchina è un'interfaccia.

La UI Design quindi si occupa prettamente della presentazione del prodotto stesso.

Le due discipline sono strettamente collegate in quanto il progetto dell'esperienza utente guida quello dell'interfaccia grafica.



Quando mi sono cimentato nella creazione dei modelli per il sito, ho quindi tenuto in considerazione quella che avevo deciso sarebbe dovuta essere l'esperienza dell'utente, ed ho disegnato l'interfaccia grafica di conseguenza.

La UI è inoltre pensata per essere il più coerente possibile, anche questa volta, con i principi di Aireem.

Questa fase si è conclusa con la realizzazione dei **mockup** per le pagine principali. Un mockup è *"una rappresentazione statica del progetto finale dalla qualità medio-alta: di solito è un progetto grafico molto dettagliato, a volte coincide con il progetto finale vero e proprio, così come verrà presentato a lavoro finito."*

Sono visibili di seguito:



Figura 5 - Mockup home page visitatori



Figura 6 - Mockup home page utenti registrati

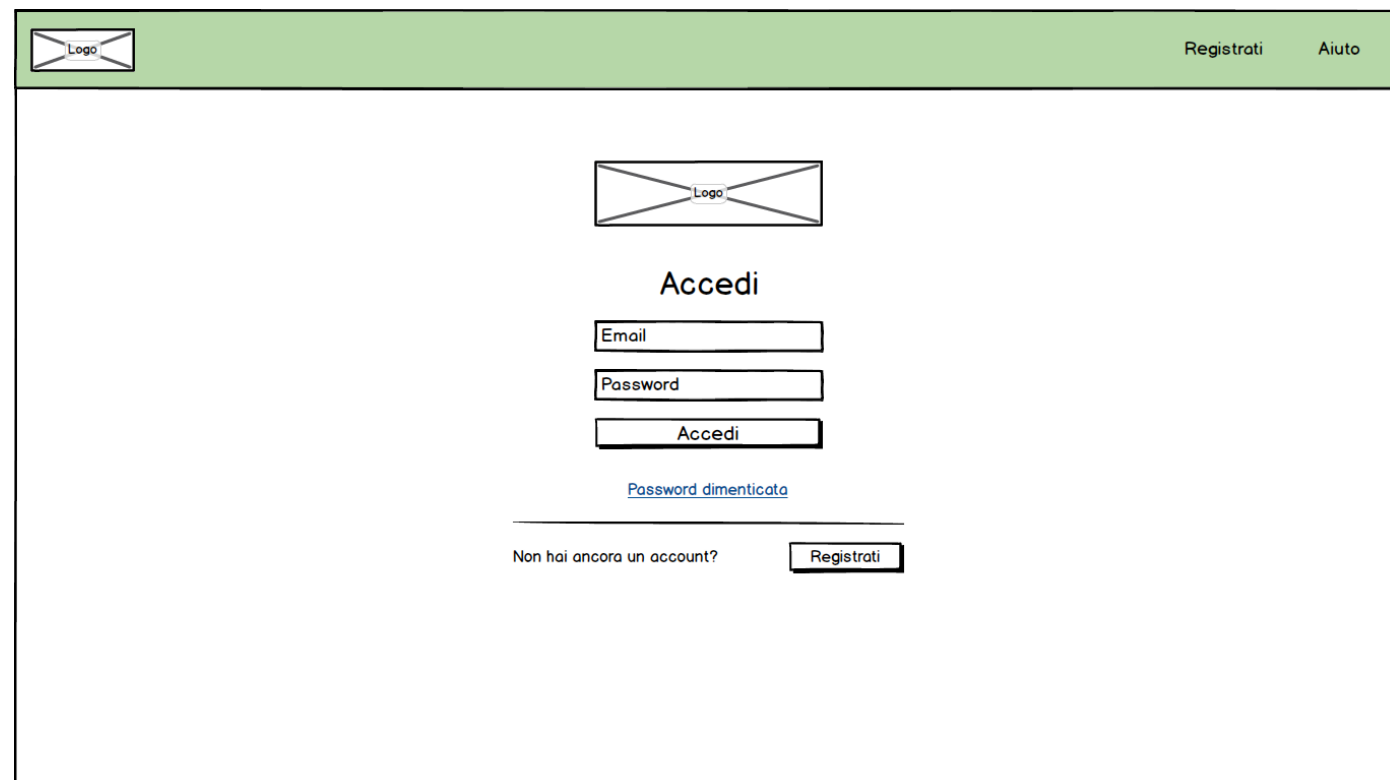


Figura 7 - Mockup pagina login

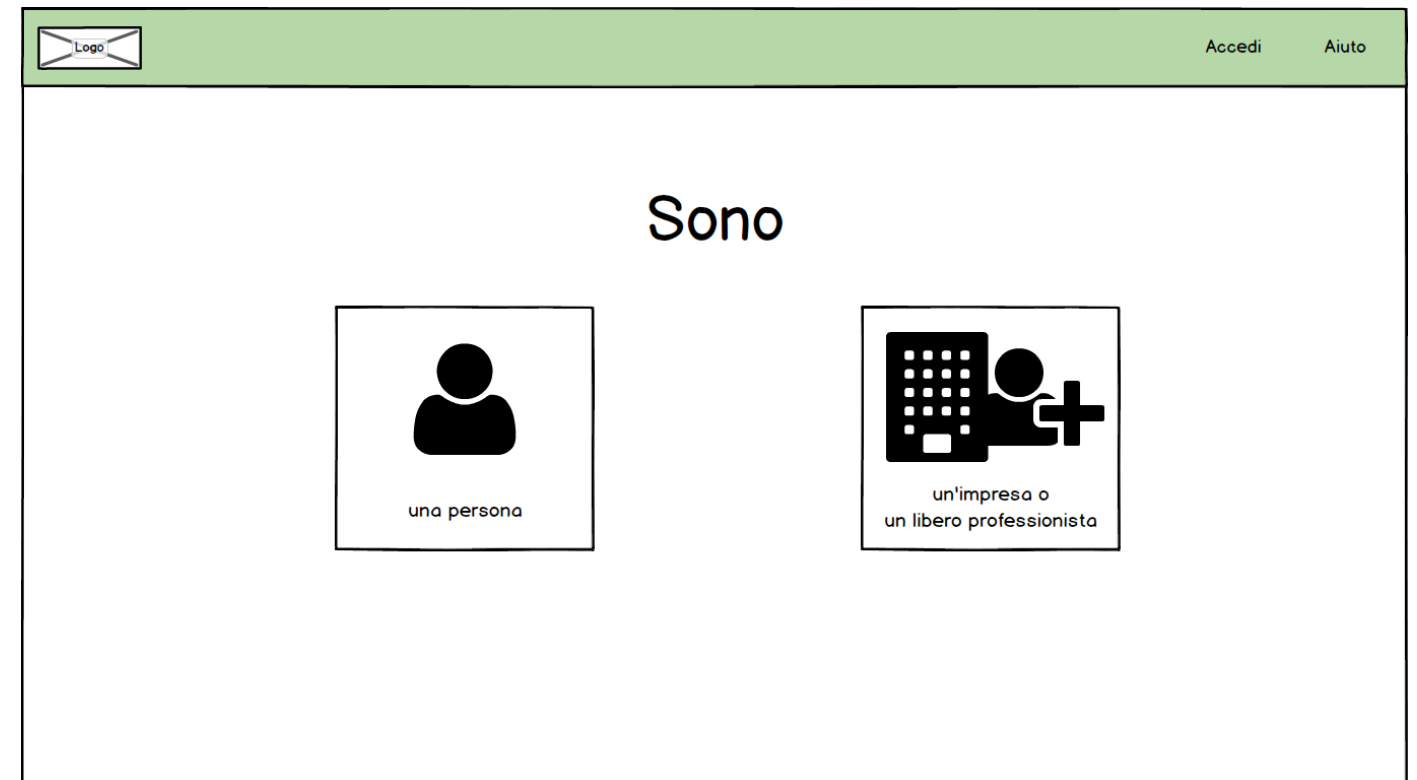



Figura 8 - Mockup registrazione: scelta tipo utente



Accedi
Aiuto

Registrati come persona

* - Campo obbligatorio

Nome *

Cognome *

Email *

Username *

Password *

Ripeti password *

* Data di nascita:

01/01/1990

Sono:

Seleziona...

Qualifica:

Seleziona...

Occupazione:

Es: impiegato, tecnico informatico...

Su di te:

Bio...

Domicilio

(servirà per prenotare appuntamenti o come luogo in cui ricevere i clienti nel caso offrirai dei servizi)

* Via, viale, piazza...

* N°

Piano, interno, scala...

* Città

* CAP

Stato, regione, provincia...

* Nazione...

Contatti

Telefono:

Es: +390001112233


Cellulare:

Es: +393442227771

Curriculum Vitae:

Seleziona file...

(max. 2MB)

 LinkedIn:


www.linkedin.com/in/my-profile

Sito web:




www.mywebsite.com

Completa registrazione!

Figura 9 - Mockup registrazione: form



Prenota
Offri

Offri un servizio

Seleziona la categoria...

Scegli il servizio: idraulico, babysitter...

Scrivi qualche informazione sul servizio che offri...

Anni di esperienza:

10

Competenze nel campo:

Es: Sviluppo front-end CSS, Full stack developer per applicazioni gestionali...

Prezzo:

10€

Costo orario

☒
☐

Prezzo trattabile

☒
☐

Servizio a domicilio

☒
☐

Disponibilità

Giorni:

☒ Lun
☒ Mar
☒ Mer
☒ Gio
☒ Ven
☒ Sab
☒ Dom

Fascia oraria

Da:


8:00




a:


20:00

Crea l'offerta

Figura 10 - Mockup form registrazione nuove offerte

 Prenota Offri



Modifica profilo

<Nome>

<Cognome>

<Username>

Info

Data di nascita:

01/01/1990

Titolo di studio o qualifica:

<qualifica>

Occupazione:

<occupation>

Curriculum Vitae

Bio:

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Contatti

Indirizzo:


Via Garibaldi, 9, Milano, 20100 - MI, Italia

Telefono:

+39000112233

Cellulare:

+393442227771




LinkedIn:




<profilo_LinkedIn>

Sito web:


<website>

Figura 11 - Mockup pagina profilo utente

 Prenota Offri

Servizio offerto: <nome prestazione>



<Nome> <Cognome>

<Username>

Anni di esperienza: <work_experience>

Competenze:

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Disponibilità: Lun - Mar - Mer - Gio - Ven

Orari: 8:00 - 20:00

Costo: 20€ all'ora trattabile

Presso: Il lavoro verrà svolto presso il domicilio del cliente che prenota

Contatti

Email:


<email>

Telefono:

<telefono>

Cellulare:

<cellulare>



LinkedIn:

<profilo_LinkedIn>

Sito web:

<website>

Prenota contattando l'offerente di questo servizio

Figura 12 - Mockup pagina dettaglio prestazione

29

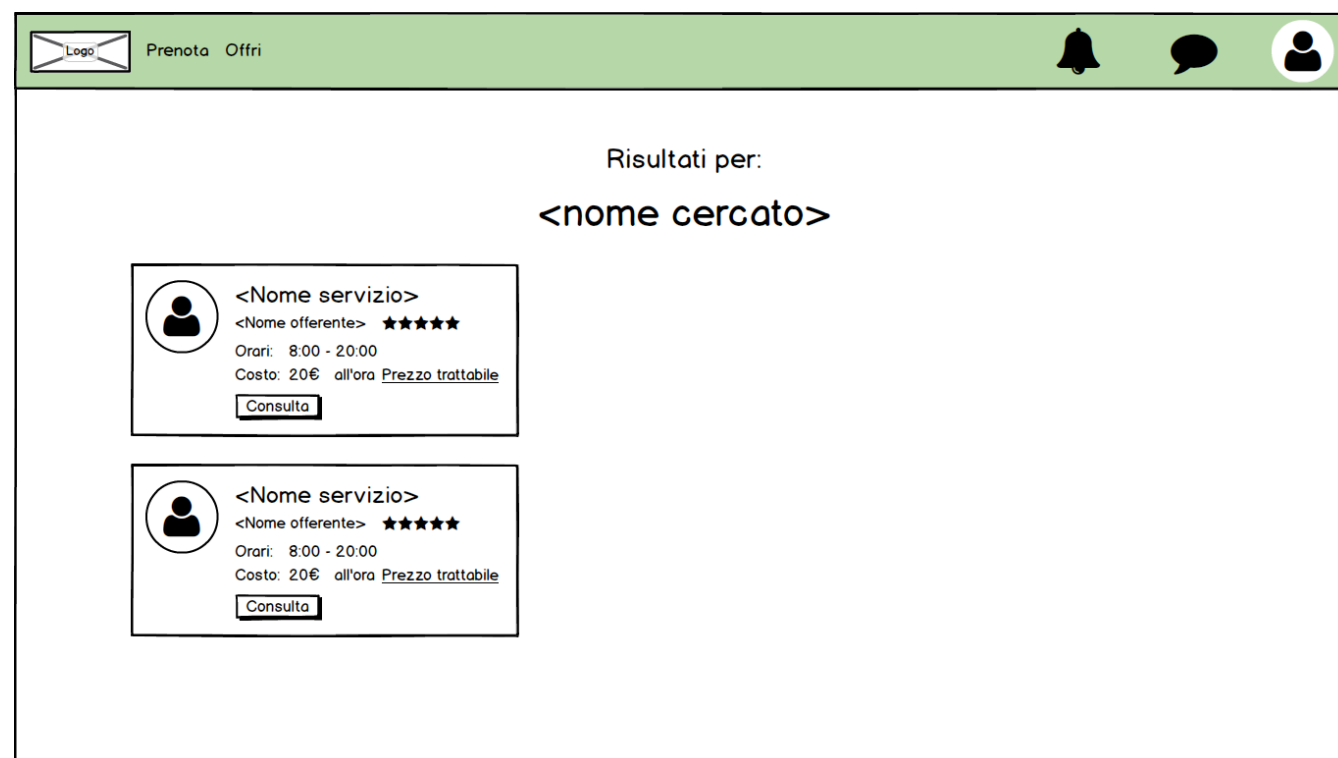


Figura 13 - Mockup pagina risultati



Figura 14 - Mockup menù utenti loggati



Figura 15 - Mockup pagina d'errore

Alcune interfacce sono relative a funzioni o ad aspetti del sito non implementati:



Figura 16 - Mockup commenti servizio

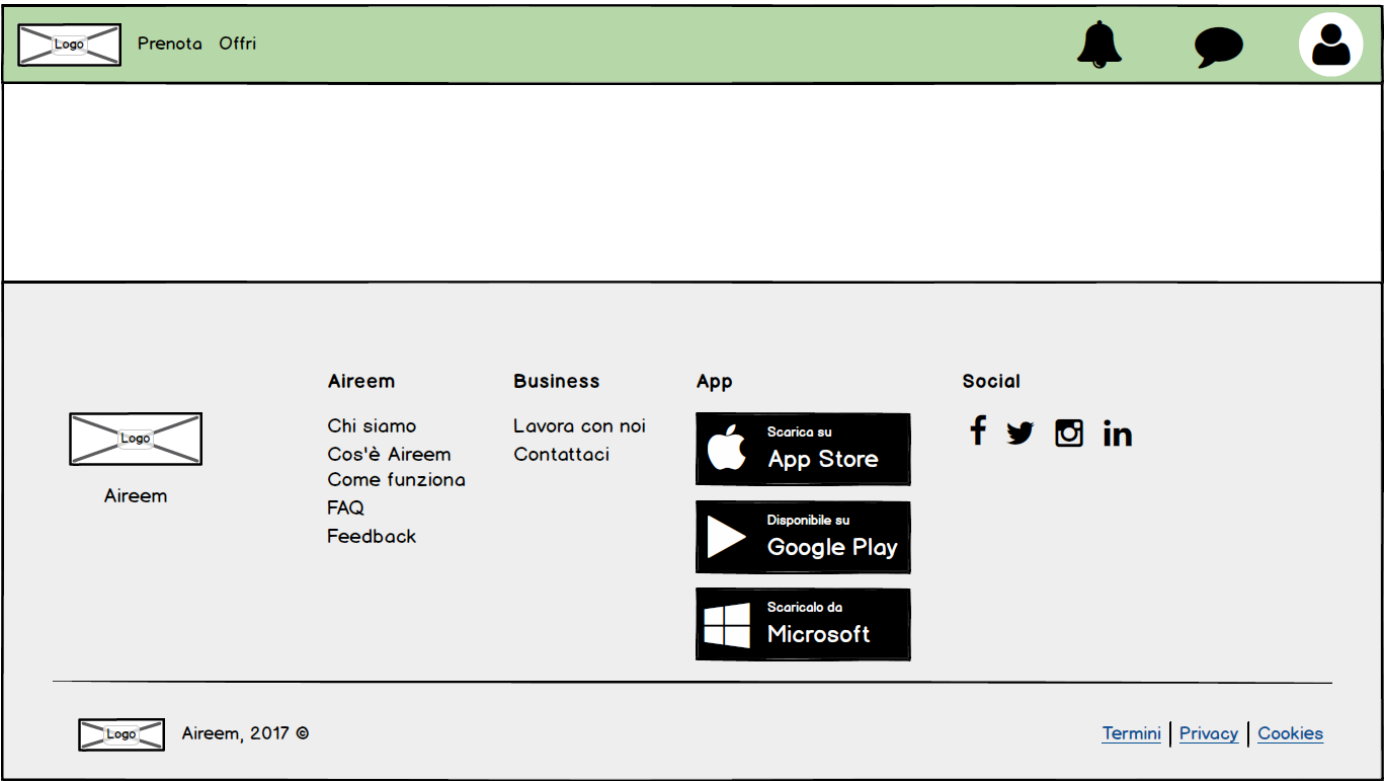


Figura 17 - Mockup footer

LINGUAGGI UTILIZZATI

I linguaggi adoperati riguardano la sfera della programmazione web *client* e *server*.

Per realizzare l'interfaccia, ovvero per il lato **front-end** della web app, mi sono avvalso di HTML, CSS e JavaScript.

L'**HTML** ("*HyperText Markup Language*") viene usato per definire la struttura delle pagine, la disposizione degli elementi al loro interno ed il relativo contenuto. Non è un linguaggio di programmazione propriamente detto, bensì è più vicino ad un modo di scrivere un documento che permetta, secondo certe regole, di mostrare ciò che si inserisce in determinati modi, a seconda del risultato voluto. Il codice HTML viene interpretato dal browser, il quale mostrerà conseguentemente la pagina all'utente.

I **CSS** ("*Cascading Style Sheets*"), o fogli di stile a cascata, sono dei file testuali, anche essi interpretati dal browser, che permettono di manipolare l'aspetto grafico della pagina web al quale li si applica. Tali fogli di stile sono scritti secondo un linguaggio (omonimo: CSS) che concede al programmatore di modificare le caratteristiche visuali proprie degli elementi presenti in un sito.

Il **JavaScript** è invece un vero e proprio linguaggio di programmazione (più precisamente, di *scripting*) ed è impiegato per gestire il comportamento delle pagine web. Può essere considerato ad oggetti poiché lavora con il **DOM** ("*Document Object Model*"), ovvero una rappresentazione ad albero degli elementi presenti in un sito. Tramite il DOM, JavaScript può quindi determinare le azioni che il browser (che ne interpreta le istruzioni) deve compiere quando un certo evento si verifica coinvolgendo uno specifico "oggetto" (ad esempio: un bottone, o un link). Gli script sono utilizzati concretamente per migliorare la navigazione e, in generale, l'esperienza dell'utente finale.

Relativamente ai CSS e al JavaScript, mi sono servito del framework **Bootstrap**: una sorta di libreria che raccoglie in alcuni file degli script e dei fogli di stile già pronti all'uso, in modo che al programmatore spetti il compito di disporre gli elementi della pagina, con l'HTML, per ottenere il risultato voluto, senza dover necessariamente cimentarsi nella modifica dei dettagli grafici o comportamentali dei contenuti.

Con alcuni oggetti ho impiegato anche **UIKit**: simile a Bootstrap, ma con un risultato grafico finale differente.

Insieme, questi linguaggi costituiscono il **DHTML** ("*Dynamic HTML*"): un insieme di tecnologie che permette di rendere più dinamica (a livello utente) una pagina web.

Per quanto riguarda, invece, il lato **back-end** dell'applicazione, la scelta è stata quella di impiegare il PHP.

PHP (acronimo di "*PHP: Hypertext Preprocessor*") è un linguaggio di scripting *general purpose* e *open source* molto utilizzato sul web.

Ha la caratteristica di essere interpretato, è perciò molto versatile poiché basta uno script engine apposito per eseguirne le istruzioni. Dunque molti siti adoperano tale tecnologia, in modo da far fronte ad eventuali cambiamenti delle caratteristiche software o hardware dei server web che garantiscono il servizio di hosting, giacché un programma scritto in un linguaggio compilato, come il C o il C++, potrebbe richiedere la ricompilazione o addirittura la modifica in certe parti a causa dell'incompatibilità delle operazioni interne le quali rispondevano alle peculiarità della macchina per le quali erano state impiegate.

L'interazione con il database MySQL ha ovviamente richiesto l'uso del linguaggio SQL.

Lo "*Structured Query Language*" permette, attraverso stringhe di testo denominate *query*, di interrogare un DB ed ottenere le informazioni desiderate. Le operazioni consentite da tale linguaggio prevedono anche la creazione di database e tabelle, l'alterazione (modifica strutturale) e l'eliminazione delle stesse, nonché l'inserimento, la modifica e l'eliminazione di dati al loro interno.

Le possibilità elencate sopra possono essere divise in tre categorie di linguaggi:

- **DDL (*Data Definition Language*)**: per la creazione di nuove basi di dati e degli archivi che le compongono.
- **DML (*Data Manipulation Language*)**: per l'inserimento, la variazione e la cancellazione dei dati.
- **QL (*Query Language*)**: per l'estrazione delle informazioni.

ARCHITETTURA

Di seguito è riportata, in maniera semplificata, l'architettura hardware di Aireem. La web app è hostata su un **server web** che, a sua volta, contiene un **database** (con il quale il software interno comunica); il dispositivo client (come un qualunque personal computer) effettua la richiesta della pagina web collegandosi attraverso la **rete Internet**. I dispositivi coinvolti sono ovviamente collegati ad **apparecchi di rete** che permettono l'instradamento del traffico di dati.

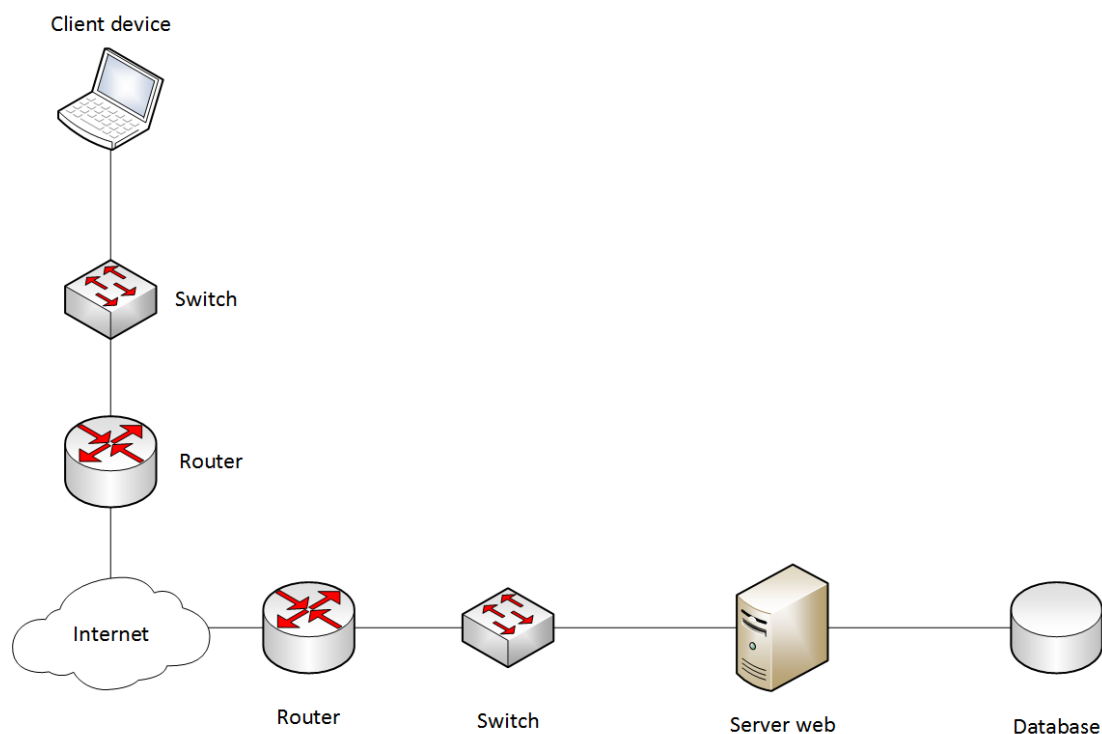


Figura 18 - Architettura hardware

Questa è nota come **architettura client/server** e gode di una diffusione che la rende particolarmente popolare rispetto ad altri tipi (come il *peer-to-peer*).

Essa permette appunto ad un client di richiedere un certo servizio o una risorsa ad un server tramite la rete. In questo specifico caso, essendo Aireem un sito web composto da pagine ipertestuali, il protocollo utilizzato durante la comunicazione tra i componenti è l'HTTP.

HTTP è l'acronimo di "HyperText Transfer Protocol" ed è un protocollo di livello applicazione (7, dello stack [ISO/OSI](#)). La comunicazione usando HTTP è sostanzialmente in due parti: richiesta e risposta. In una **richiesta** vengono specificati:

- **Metodo:** ovvero il tipo di operazione che il client richiede al server (*GET*, *POST*).
- **URL:** "*Uniform Resource Locator*", identifica la risorsa, *locale rispetto al server*, che si vuole ottenere.
- **Identificativo della versione del protocollo HTTP**
- **Informazioni aggiuntive:** data e ora di generazione della richiesta, user agent, ecc...

Una **risposta** invece comprende:

- **Identificativo della versione del protocollo http**
- **Codice di stato:** 200 OK, 404 Not Found, ecc...
- **Informazioni aggiuntive riguardanti la risposta**
- **Contenuto della risorsa richiesta dal client:** come una pagina HTML o una stringa di testo JSON.

L'architettura software di Aireem è composta fondamentalmente da un **web server** al quale vengono inoltrate le richieste HTTP dei **client** (a cui dovrà rispondere), dei **file PHP**, che sono le vere risorse richiamate, che elaborano il **documento HTML** finale tramite dati ottenuti all'occorrenza dal **database**. Tale documento è poi inviato al **browser** del client che, dopo averlo interpretato, lo mostrerà a video.

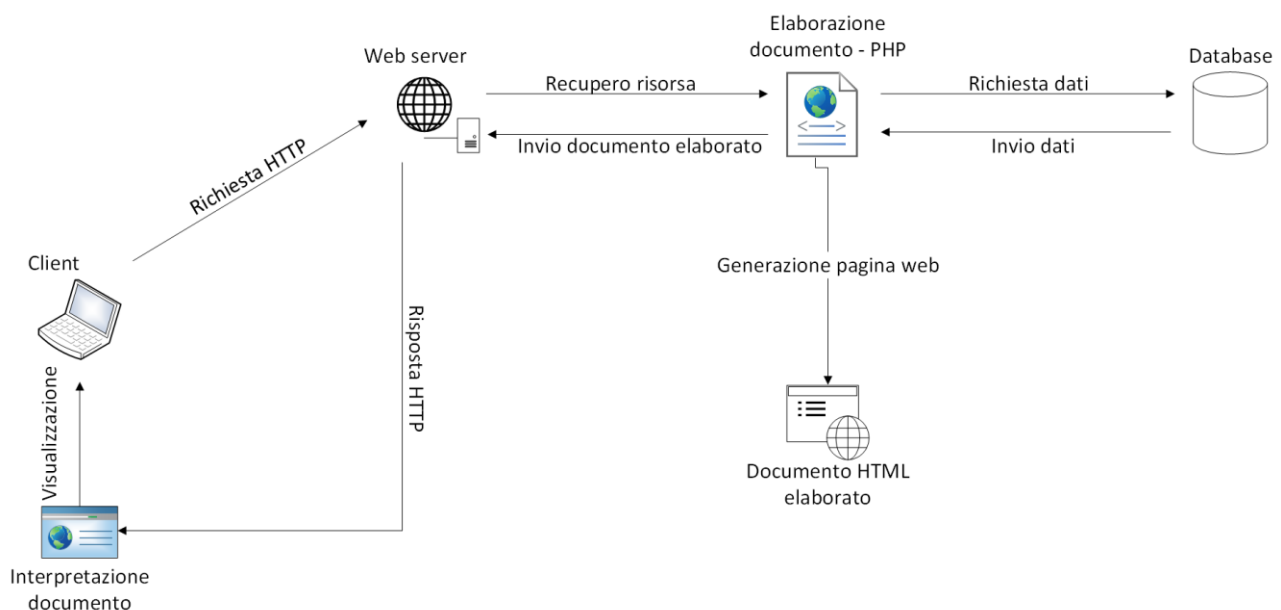


Figura 19 - Architettura software

ANALISI PARTI DI CODICE

A questo punto è necessaria la spiegazione riguardo alcune parti del codice degli script PHP scritti per Aireem.

Verranno presi in analisi i frammenti più significativi, attorno ai quali gravita l'esecuzione delle richieste di dati al DB e la preparazione degli stessi per la visualizzazione in un documento HTML.

Innanzitutto, nell'**header** di ogni pagina sono stati inclusi i fogli di stile e date informazioni come titolo e metadati.

```
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1">

  <link rel="icon" href="favicon.ico">
  <title>Aireem - Compra e vendi prestazioni online</title>
  <link href="bootstrap/css/bootstrap.min.css" rel="stylesheet">
  <link href="custom/css/custom-css.css" rel="stylesheet">

  <!-- HTML5 shim and Respond.js for IE8 support of HTML5 elements and media queries -->
  <!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
  <!--[if lt IE 9]>
    <script src="https://oss.maxcdn.com/html5shiv/3.7.3/html5shiv.min.js"></script>
    <script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
  <![endif]-->
</head>
```

Figura 20 - Codice: head HTML

Nelle pagine visitabili sia da utenti esterni che registrati, la scelta di quale navbar mostrare è data dalla presenza (o meno) in memoria di sessione dell'user ID, il quale viene caricato proprio al momento del login o della registrazione. La **sessione** è utilizzata dal PHP per effettuare operazioni in base all'utente che le richiede. Ogni sessione è identificata da alcuni dati tra cui l'user-agent e l'indirizzo IP.

```
if(isset($_SESSION["usrID"]) && !empty($_SESSION["usrID"]))
{
    include("logged-user-navbar.php");
}
else
{
    include("visitor-navbar.php");
}
```

Figura 21 - Codice: navbar

Ogni visitatore può effettuare il login tramite il form che viene stampato a video, generato dal seguente codice HTML:

```
<div class="container-fluid">
  <p></p>
  <p class="text-center"><h3 class="text-center">Accedi</h3></p>
  <div class="row">
    <div class="col-lg-4 col-lg-offset-4 col-md-4 col-md-offset-4 col-sm-6 col-sm-offset-3 col-xs-12">
      <?php echo $MESSAGE; ?>
    </div>
    <div class="col-lg-2 col-lg-offset-5 col-md-2 col-md-offset-5 col-sm-4 col-sm-offset-4 col-xs-12">
      <form id="login" name="login" action="login.php" method="post" enctype="multipart/form-data">
        <div class="form-group">
          <input type="email" class="form-control" name="email" id="email" placeholder="Email">
        </div>
        <div class="form-group">
          <input type="password" class="form-control" name="password" id="password" placeholder="Password">
        </div>
        <div class="form-group">
          <input type="hidden" name="logging" id="logging" value="y">
        </div>
        <button type="submit" class="btn btn-primary center-block" style="width:100%;">Accedi</button>
      </form>
      <form action="#" class="text-center">
        <button type="submit" class="btn btn-link">Password dimenticata</button>
      </form>
    </div>
    <div class="col-lg-4 col-lg-offset-4 col-md-4 col-md-offset-4 col-sm-6 col-sm-offset-3 col-xs-12">
      <hr>
      <form action="signup.php">
        <h5 style="float:left;>Non hai ancora un account?</h5> <button type="submit" class="btn btn-success btn-xs text-right" style="float:right;">
      </form>
    </div>
  </div>
</div>
```

Figura 22 - Codice: form login

Una volta immessi ed inviati i dati al server tramite il pulsante "Accedi", lo script PHP verrà chiamato nuovamente e svolgerà le dovute verifiche tramite la richiesta dei dati al DB.

```
if(isset($_POST["email"]) && isset($_POST["password"]) && !empty($_POST["email"]) && !empty($_POST["password"]))
{
    $email=$_POST["email"];
    $pw=$_POST["password"];

    // DB connection
    $cn=mysqli_connect($HOSTDB,$USERDB,$PASSDB,$NAMEDB);

    // db connection error
    if(!$cn)
    {
        $_SESSION["errorcodelogin"] = 2;
        $_SESSION["errorlogin"] = "Ci sono problemi di connessione. Riprovare tra qualche minuto.";
        mysqli_close($cn);
        header("Location:login.php");
        exit();
    }

    $sql="SELECT * FROM Users WHERE usrEmail='$email'";

    $res=mysqli_query($cn,$sql); // query execution
}
```

Figura 23 - Codice: connessione al DB

I dati vengono passati in POST, per cui PHP userà tali variabili per ricavarli. Successivamente stabilirà la connessione tramite gli appositi parametri definiti nel file "*connection.php*" (incluso in precedenza), sfruttando la funzione *mysqli_connect()*. Lo script provvederà anche a controllare se il collegamento sia avvenuto realmente con la struttura di selezione *if(!\$cn)*, e nel caso non sia "true", reindirizzerà l'utente su una pagina d'errore.

Dopo aver aperto un canale di comunicazione con il database, viene preparato l'SQL in una stringa e lo si esegue con la funzione *mysqli_query()*, la cui risposta va nella variabile *\$res*.

```
// reading from db error
if(!$res)
{
    $_SESSION["errorcodelogin"] = 3;
    $_SESSION["errorlogin"] = "Ci sono problemi di acquisizione dati. Riprovare tra qualche minuto.";
    mysqli_close($cn);
    header("Location:login.php");
    exit();
}

// number of rows
if (mysqli_num_rows($res) != 1)
{
    // user not found
    $_SESSION["errorcodelogin"] = 4;
    $_SESSION["errorlogin"] = "Email o password errata.";
    unset($_POST["password"]);
    mysqli_close($cn);
    header('Location:login.php');
    exit();
}

// reading from DB
$riga = mysqli_fetch_assoc($res);
```

Figura 24 - Codice: recupero dati dal DB

A questo punto viene controllato se la query ha generato dei risultati con *if(!\$res)*: il valore "true" eventualmente tornato dall'espressione, poiché *\$res* è negato ("!"), indica una risposta vuota, per cui un probabile errore nella query stessa, in quanto non è stato proprio possibile effettuare l'operazione richiesta sul DB.

Nel caso contrario, la funzione ha generato dei dati. Dato che la stringa eseguita prevedeva la selezione di un utente in base all'email, e dato che un'email è unica in tutta la tabella, se il numero dei record tornati non è uguale a 1, vuol dire che l'utente non esiste oppure che, anche questa volta, ci sia stato un errore di esecuzione. Questo controllo viene effettuato con la funzione `mysqli_num_rows()`, e se il risultato è diverso da 1 si procede dunque con la preparazione di un messaggio d'errore ed il reindirizzamento alla stessa pagina (`login.php`) in modo da mostrarlo.

Se il flusso di esecuzione procede senza errori, si prende ciò che è stato ritornato dalla richiesta e lo si memorizza in una variabile locale per utilizzare in seguito le informazioni. L'operazione descritta viene realizzata tramite la funzione `mysqli_fetch_assoc()`.

A questo punto si ricava la password tramite `$riga`, che è di fatto un **array associativo**, ovvero ad ogni elemento non è assegnata solo una posizione numerica ma anche una chiave alfanumerica grazie alla quale vi si può fare riferimento. Se la password ricavata è uguale a quella immessa dall'utente, il login è avvenuto con successo, per cui vengono messi in sessione tutti gli altri dati fondamentali dell'user e si esegue una redirect alla pagina iniziale, altrimenti si dà un messaggio d'errore.

```
if ($pw == $riga["usrPassword"]) // MD5($pw)
{
    // password is correct
    $_SESSION["usrID"] = $riga["usrID"];
    $_SESSION["usrEmail"] = $email;
    $_SESSION["usrUsername"] = $riga["usrUsername"];
    $_SESSION["usrProPic"] = $riga["usrProPic"];
    $_SESSION["usrPremium"] = $riga["usrPremium"];
    $_SESSION["usrPersonCode"] = $riga["usrPersonCode"];
    $_SESSION["usrP_Iva"] = $riga["usrP_Iva"];
    unset($_POST["password"]);
    mysqli_close($cn);
    header('Location: /');
    exit();
}
else
{
    // password is wrong
    $_SESSION["errorcodelogin"] = 4;
    $_SESSION["errorlogin"] = "Email o password errata.";
    unset($_POST["password"]);
    mysqli_close($cn);
    header('Location: login.php');
    exit();
}

mysqli_close($cn);
```

Figura 25 - Codice: controllo password

Va notato che prima di ogni reindirizzamento, e soprattutto alla fine, la connessione al DB va chiusa con la funzione `mysqli_close()`.

```
// print login error message if it is set
if(isset($_SESSION["errorlogin"]) && !empty($_SESSION["errorlogin"]) && isset($_SESSION["errorcodelogin"]) && !empty($_SESSION["errorcodelogin"]))
{
    $MESSAGE="<div class=\"alert alert-danger\" role=\"alert\"><span class=\"glyphicon glyphicon-exclamation-sign\" aria-hidden=\"true\"></span> <
    unset($_SESSION["errorcodelogin"]);
    unset($_SESSION["errorlogin"]);
}
else
{
    if(isset($_POST["email"]) && isset($_POST["password"]) && !empty($_POST["email"]) && !empty($_POST["password"]))
    {
```

Figura 26 - Codice: gestione errori login

Data la possibilità che *login.php* venga richiamato per dare in output un messaggio d'errore, all'inizio è presente un controllo proprio a tale scopo, in modo che se la relativa variabile in sessione è impostata e non è vuota, allora si preparerà la stringa in questione.

Quando si cerca un servizio in una certa zona, nella pagina iniziale, l'SQL eseguito effettuerà molteplici **join** per ricavare le informazioni necessarie. Di seguito il codice.

```
$sql="SELECT offID,offStartTime,offEndTime,offCost,offHourlycost,offCostToBeDetermined,offUserID,prfName,pplName,pplSurname,usrProPic,curCode,revOffID,AVG(revRating) AS 'Rating' FROM Offers,Performances,People,Users,Cities,Currencies,Reviews WHERE prfName LIKE '%".$_GET["service"]."% AND offPrfID=prfID AND offUserID=usrID AND usrPersonCode=pplPersonCode AND cyState='".$_GET["state"]."' AND pplCyID=cyID AND offCurID=curID AND revOffID=offID GROUP BY revOffID,offID,offStartTime,offEndTime,offCost,offHourlycost,offCostToBeDetermined,offUserID,prfName,pplName,pplSurname,usrProPic,curCode;";
```

Figura 27 - Codice: query SQL

La fase di recupero dati dall'esecuzione della query è analoga a quanto mostrato sopra. La pagina dei risultati verrà poi costruita concatenando i frammenti di codice riguardanti un'offerta alla volta e le relative informazioni ottenute, come è possibile vedere dall'immagine sottostante.

```

while($riga = mysqli_fetch_assoc($res))
{
    // create time
    $STARTTIME=date_create($riga["offStartTime"]);
    $ENDTIME=date_create($riga["offEndTime"]);

    // compose result card
    $RESULTS.="<div class='row'>
        <div class='col-lg-4 col-lg-offset-1 col-md-4 col-md-offset-1 col-sm-12 col-xs-12'>
            <div class='panel' style='width:100%;border:0px;'>
                <div class='uk-card uk-card-default uk-card-hover uk-card-body' style='padding-top:20px;padding-bottom:20px;padding-left:15px;'>
                    <!-- user's service propic -->
                    <div class='col-lg-4 col-md-4 col-sm-1 col-xs-0' style='margin-top:5px;padding-left:0px;'>
                        <img src='".$riga["usrProPic"]."' class='img-responsive img-circle center-block' alt='Profile pic' width='70px' height
                    </div>
                    <!-- service body -->
                    <div class='col-lg-8 col-md-8 col-sm-11 col-xs-12' style='margin-top:5px;padding-left:0px;'>
                        <table border='0' style='width:100%;>
                            <tr>
                                <td><h3 style='margin-top:0px;'>".$riga["prfName"]."</h3></td>
                            </tr>
                            <tr>
                                <td><div style='margin-bottom:10px;'><a href='user.php?id='".$riga["offUserID"]."' target='_blank'>".$riga["pp1
                            </tr>
                            <tr>
                                <td><div style='margin-bottom:10px;'>Orari:&nbsp;&nbsp;&nbsp;".$date_format($STARTTIME,'H:i').<div style='margin-bottom:10px;'>Costo:&nbsp;&nbsp;&nbsp;".$riga["offCost"]."&nbsp;&nbsp;&nbsp;".$riga["curCode"]."&nbsp;&nbsp;&nbsp;
                            </tr>
                            <tr>
                                <td><button class='btn btn-success btn-xs' onclick='\"window.location.href='service.98:45 PHP Spaces:4
                            </tr>
            </div>
        </div>
    </div>

```

Figura 28 - Codice: preparazione HTML dei risultati

Cliccando sul bottone "Consulta" di una specifica offerta si potrà accedere alla relativa pagina di dettaglio, costruita a partire dai dati che la riguardano stampati nel codice HTML dallo script PHP.

```

<td><div class='text-center'><h4 style='margin-top:0px;'>Contatti</h4></div></td>
</tr>
<tr>
<td style='padding:10px;'><b>Email:</b></td>
<td class='text-right' style='padding:10px;'><?php echo ((isset($data["usrEmail"]) && !empty($data["usrEmail"]))?$data["usrEmail"]:"<i>Non disponibile</i>") ?></td>
</tr>
<tr>
<td style='padding:10px;'><b>Telefono:</b></td>
<td class='text-right' style='padding:10px;'><?php echo ((isset($data["pplTel"]) && !empty($data["pplTel"]))?$data["pplTel"]:"<i>Non disponibile</i>") ?></td>
</tr>
<tr>
<td style='padding:10px;'><b>Cellulare:</b></td>
<td class='text-right' style='padding:10px;'><?php echo ((isset($data["pplCell"]) && !empty($data["pplCell"]))?$data["pplCell"]:"<i>Non disponibile</i>") ?></td>
</tr>
<tr>
<td style='padding:10px;'>&nbsp;&nbsp;&nbsp;<b>LinkedIn:</b></td>
<td class='text-right' style='padding:10px;'><?php echo ((isset($data["pplLinkedin"]) && !empty($data["pplLinkedin"]))?"<a href='{$data["pplLinkedin"]}' target='_blank'
</tr>
<tr>
<td style='padding:10px;'><b>Sito web:</b></td>
<td class='text-right' style='padding:10px;'><?php echo ((isset($data["pplWebsite"]) && !empty($data["pplWebsite"]))?"<a href='{$data["pplWebsite"]}' target='_blank'>Va
</tr>
</table>
</td>

```

Figura 29 - Codice: preparazione HTML dei contatti

Quando un utente che ha in precedenza eseguito l'accesso vorrà effettuare il **logout**, passerà per la pagina *Logout.php*, di seguito:

```

<?php
session_start();

// delete SESSION data
unset($_SESSION["usrID"]);
unset($_SESSION["usrEmail"]);
unset($_SESSION["usrUsername"]);
unset($_SESSION["usrProPic"]);
unset($_SESSION["usrPremium"]);
unset($_SESSION["usrPersonCode"]);
unset($_SESSION["usrP_Iva"]);
session_destroy(); // destroy session

header('Location: /');
exit();
?>

```

Figura 30 - Codice: operazioni di logout

Tramite la funzione *unset()* vengono cancellati i dati dalla memoria di sessione, e con *session_destroy()* la sessione stessa è eliminata. Infine vi è un redirect alla pagina principale con *header('Location: /')* ed *exit()*.

HOSTING

Una volta terminata, la web app è stata pubblicata sul sito di hosting gratuito [AlterVista](#). La piattaforma mette a disposizione diversi strumenti per il caricamento e la gestione dei file componenti la propria applicazione. Offre inoltre un database di tipo MySQL con il quale poter lavorare, anche tramite l'interfaccia di phpMyAdmin.

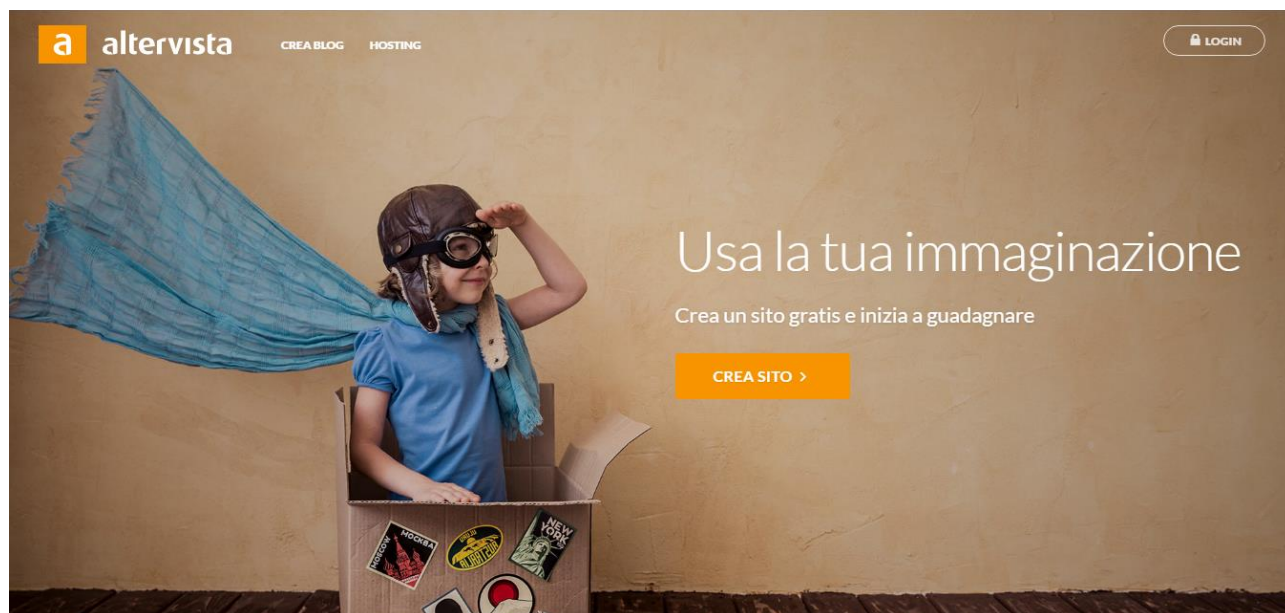


Figura 31 - Home Page AlterVista

	signup.php	47 KB	12/06/2017
	user.php	7 KB	12/06/2017
	visitor-navbar.php	1 KB	12/06/2017

NuovaCartella

NuovoFile

html

Crea File

Crea Pagina con editor

Crea Cartella

Risorse

SPAZIO UTILIZZATO

0,2%

3.140.146 KB

liberi

5.328 KB

usati per i file

254 KB

usati dal database

La tua quota di spazio ammonta a **3.072 Megabytes**, potrai aumentarla fino all'**infinito**.
Ricorda che ogni database mysql attivo è incluso nel tuo spazio web.
È anche possibile aumentare la banda a tua disposizione in modo analogo.

Aumenta lo Spazio

Ricalcola spazio occupato

Feedback

Figura 32 - Gestione pagine web AlterVista

MANUALE UTENTE

Il sito è raggiungibile online all'indirizzo aireem.altervista.org.

La home page (*index*) si presenta in maniera differente nel caso in cui un utente abbia effettuato l'accesso o meno.

Alla prima visita verrà visualizzato anche un pulsante che, se premuto, condurrà alla pagina di registrazione.

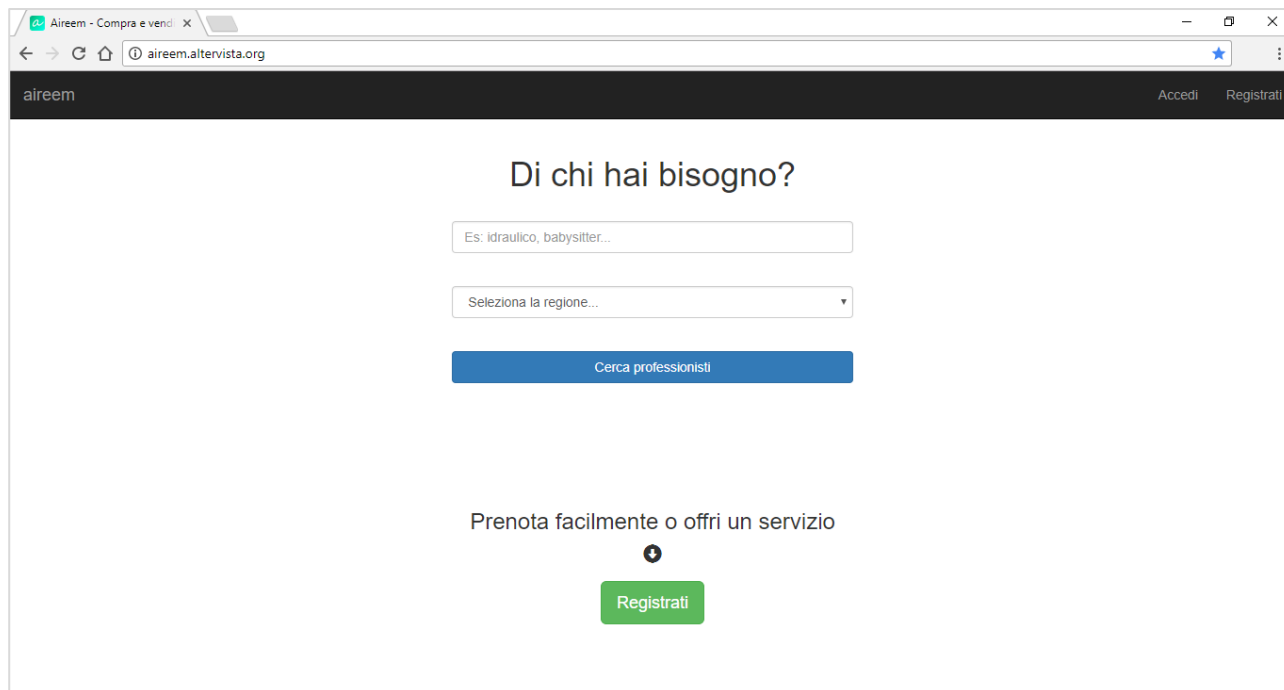
The screenshot shows a web browser window with the address bar displaying 'aireem.altervista.org'. The page has a dark header with the 'aireem' logo on the left and 'Accedi' and 'Registrati' links on the right. The main content area is white and features the heading 'Di chi hai bisogno?'. Below this is a search input field containing the text 'Es: idraulico, babysitter...'. Underneath the search field is a dropdown menu labeled 'Seleziona la regione...'. A blue button labeled 'Cerca professionisti' is positioned below the dropdown. Further down, the text 'Prenota facilmente o offri un servizio' is displayed above a green button labeled 'Registrati'.

Figura 33 - Aireem: Home Page visitatori

Cliccando su "Accedi" in alto a destra, si potrà eseguire il login.

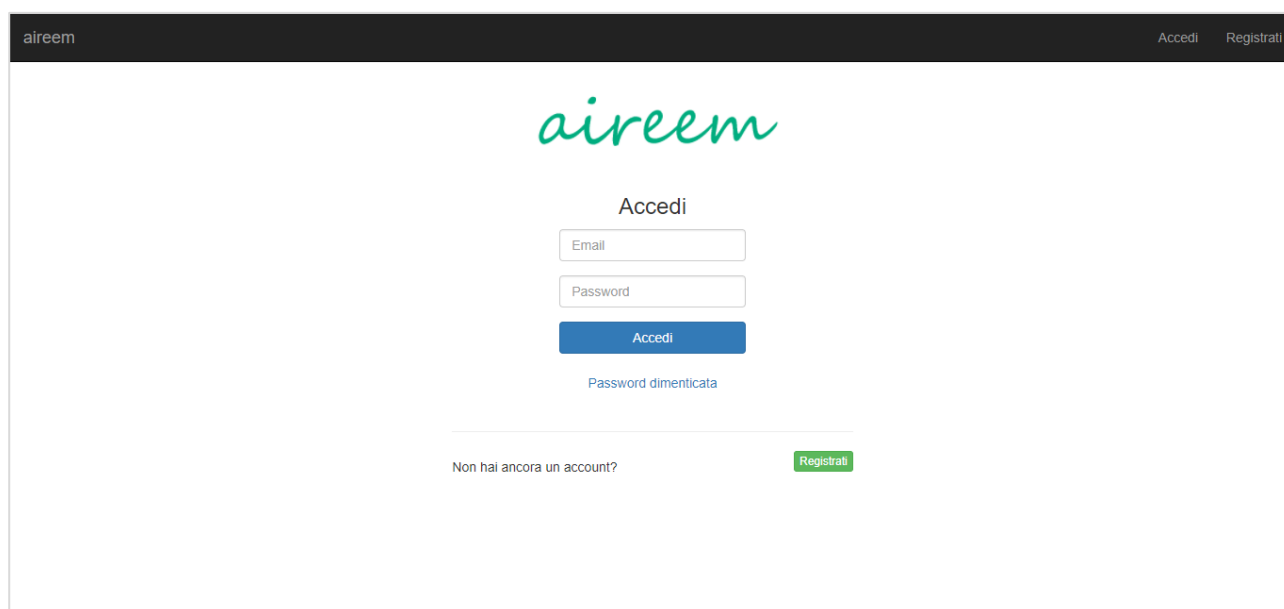
The screenshot shows the login page of the Aireem website. The dark header remains at the top with the 'aireem' logo and 'Accedi' and 'Registrati' links. The main content area is white and features the 'aireem' logo in a green script font. Below the logo is the heading 'Accedi'. There are two input fields: 'Email' and 'Password'. A blue button labeled 'Accedi' is located below the password field. Underneath the button is a link that says 'Password dimenticata'. At the bottom of the page, there is a line of text that reads 'Non hai ancora un account?' followed by a green button labeled 'Registrati'.

Figura 34 - Aireem: login

Oppure, se non si ha ancora un account, ci si può registrare tramite l'apposito link "Registrati". Bisognerà innanzitutto scegliere il tipo di utente, se privato o aziendale:

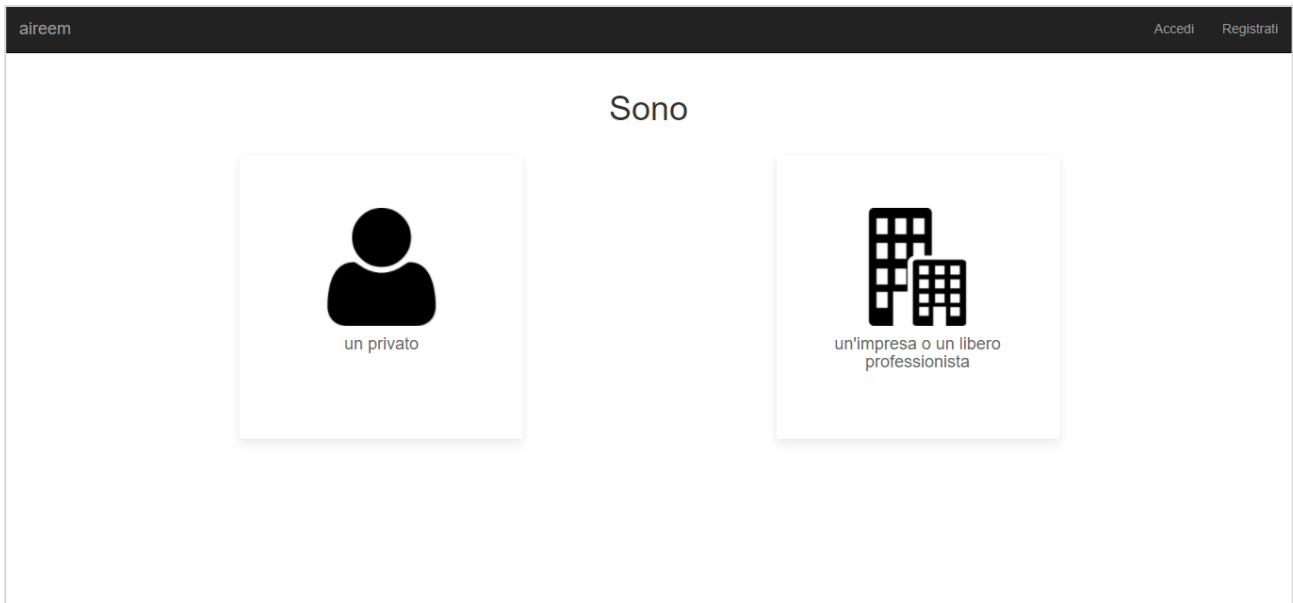


Figura 35 - Aireem: scelta tipo utente

Poi si dovrà compilare un modulo inserendo le informazioni richieste:

The screenshot shows the 'Registrati come privato' registration form. At the top, it has the 'aireem' logo and 'Accedi' / 'Registrati' links. The heading 'Registrati come privato' is centered. Below it, a legend indicates '* - Campo obbligatorio'. The form contains several input fields: '* Nome', '* Cognome', '* Email', '* Username', '* Password', and '* Ripeti password'. There is a date picker for '* Data di nascita' with the value '1990-12-31'. Below these are three dropdown menus: 'Sono:' with 'Seleziona...' selected, 'Qualifica:' with 'Seleziona...' selected, and 'Occupazione:' with 'Es: impiegato, tecnico informatico...' selected.

Figura 36 - Aireem: registrazione

A questo punto, alle visite successive si potrà accedere con le credenziali immesse in fase di registrazione.

La home page apparirà leggermente differente, e il menù in alto conterrà alcune opzioni per la navigazione all'interno del sito.

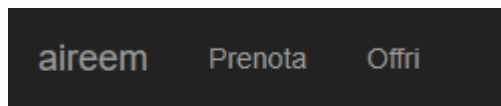


Figura 37 - Aireem: Prenota e Offri

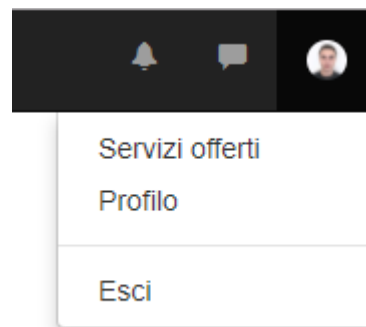


Figura 38 - Aireem: menù utente

Per consultare le offerte di prestazioni basterà scrivere nella prima casella di testo (della home page, raggiungibile anche tramite il link del menù in alto a sinistra "Prenota") ciò a cui si è interessati e scegliere la regione per circoscrivere la ricerca.

The image shows the main content area of the Aireem website. At the top, there is a dark navigation bar with the 'aireem' logo and links for 'Prenota' and 'Offri'. Below this, the heading 'Di chi hai bisogno?' is centered. Under the heading, there is a text input field containing the placeholder text 'Es: idraulico, babysitter...', a dropdown menu labeled 'Seleziona la regione...', and a blue button labeled 'Cerca professionisti'.

Figura 39 – Aireem: Home Page utenti loggati

The image shows a close-up of the search form. It features the heading 'Di chi hai bisogno?' followed by a text input field with the value 'computer', a dropdown menu with the value 'Lombardia', and a blue button labeled 'Cerca professionisti'.

Figura 40 - Aireem: ricerca offerte

Premuto il tasto "Cerca professionisti" si verrà poi indirizzati alla pagina dei risultati:

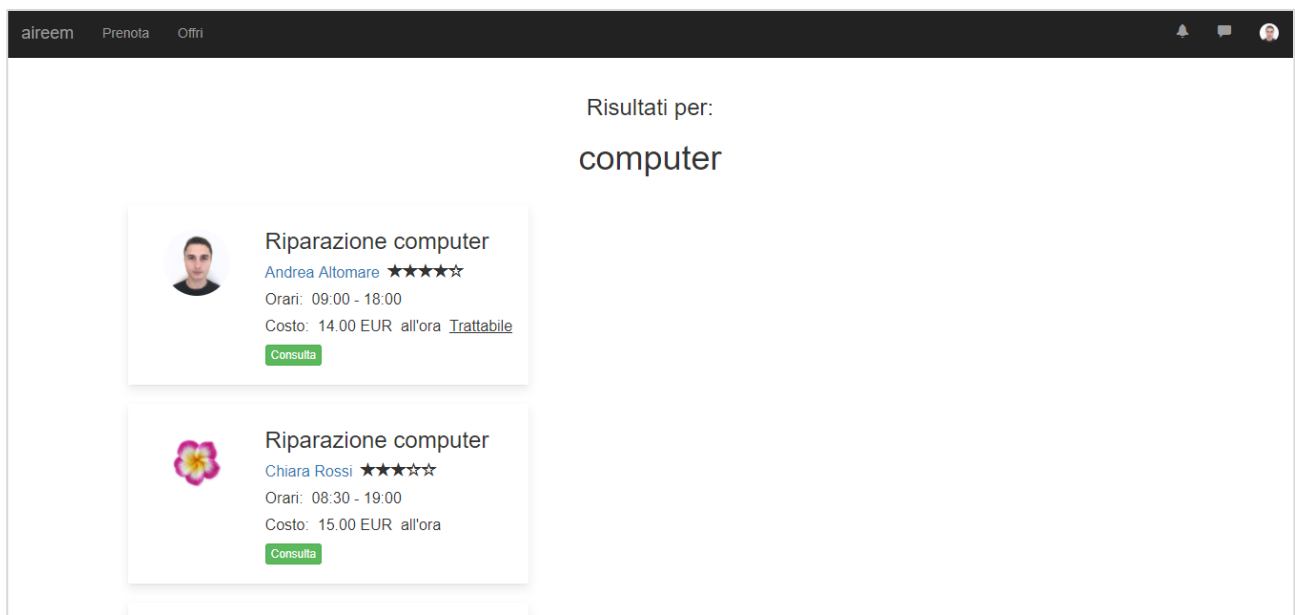


Figura 41 - Aireem: pagina risultati

Cliccando su "Consulta" si accederà alla pagina di dettaglio della prestazione, la quale conterrà maggiori dettagli nonché i contatti dell'utente offerente.



Figura 42 - Aireem: dettaglio prestazione

Gli utenti registrati potranno vedere sinteticamente tutte le proprie offerte grazie alla voce "Servizi offerti" del menù a tendina apribile cliccando sulla propria immagine di profilo in alto a destra.

aireem Prenota Offri					
Servizi offerti					
ID	Servizio	Info	Ora inizio	Ora fine	
1	Web Design	Web design for companies and people. Personal portfolio and lot more!	09:00	18:00	Visualizza Elimina
12	Riparazione computer	Riparazione componenti hardware PC	09:00	18:00	Visualizza Elimina
15	Video editing	Editing video professionale. Servizio freelance.	09:30	17:00	Visualizza Elimina

Figura 43 - Aireem: servizi offerti da utente loggato

Se si vuole creare un'offerta si dovrà cliccare sul link "Offri" in alto a sinistra. Verrà dunque aperta la pagina nella quale inserire i dati relativi alla propria prestazione.

aireem Prenota Offri					
Offri un servizio					
* - Campo obbligatorio					
* Seleziona la categoria...					
* Il servizio che offri, es: babysitter, idraulico, programmatore, ecc...					
Scrivi qualche informazione sul servizio che offri...					
Anni di esperienza:					
Competenze nel campo:					
Es: Public Speaking, Sviluppo front-end CSS, Full Stack Developer...					
Prezzo:					
Costo orario					

Figura 44 - Aireem: form nuove offerte

Dopo aver immesso le informazioni richieste, premendo il tasto "Crea l'offerta" l'operazione sarà ultimata e si verrà reindirizzati alla pagina di dettaglio del nuovo servizio.

Es: Public Speaking, Sviluppo front-end CSS, Full Stack Developer...

Prezzo: EUR ☐ Costo orario ?

Prezzo trattabile ☐ ?

Servizio a domicilio ☐ ?

Disponibilità

Giorni: ☐ Lun ☐ Mar ☐ Mer ☐ Gio ☐ Ven ☐ Sab ☐ Dom

Fascia oraria
L'orario entro cui si possono svolgere una o più prestazioni.

Dalle: alle:

[Crea l'offerta](#)

Figura 45 - Aireem: conferma creazione nuova offerta

Sempre dal menù espandibile, cliccando su *"Profilo"* si potranno vedere i dati della propria persona. Tale pagina sarà utile anche nel caso si vogliano consultare le informazioni riguardo uno specifico utente offerente.

aireem Prenota Offri

Andrea Altomare

andrea.altomare

Info:

Data di nascita: 1998-03-23

Titolo di studio: Diploma di istruzione tecnica - Informatica

Occupazione: Software analyst at Aireem

Bio:
Ragazzo europeo diplomato in un Istituto Tecnico. Ho competenze informatiche.

Contatti:

Indirizzo: Via Dante, 1 - Milano, 20121 - Lombardia, IT

Telefono: 0731431622

Cellulare: 3251997462

LinkedIn: www.linkedin.com/in/andrea-altomare/

Figura 46 - Aireem: profilo utente

Aireem è consultabile anche da dispositivi mobili. Di seguito alcune schermate delle pagine principali.

Di chi hai bisogno?

Es: idraulico, babysitter...

Seleziona la regione...

Cerca professionisti

Prenota facilmente o offri un servizio

Registrati

Di chi hai bisogno?

Es: idraulico, babysitter...

Seleziona la regione...

Cerca professionisti

Prenota facilmente o offri un servizio

Registrati

ID	Servizio	Info	Ora inizio	Ora fine		
1	Web Design	Web design for companies and people. Personal portfolio and lot more!	09:00	18:00	Visualizza	Elimina
12	Riparazione computer	Riparazione componenti hardware PC	09:00	18:00	Visualizza	Elimina

Accedi

Email

Password

Accedi

Password dimenticata

Non hai ancora un account? Registrati

Risultati per:
computer

Riparazione computer
Andrea Altomare ★★★★★
Orari: 09:00 - 18:00
Costo: 14.00 EUR all'ora [Trattabile](#)
[Consulta](#)

Riparazione computer
Chiara Rossi ★★★★★
Orari: 08:30 - 19:00
Costo: 15.00 EUR all'ora

CONCLUSIONI

Aireem è dunque una web application che, sebbene ancora immatura per il mercato, può essere utilizzata per cercare e offrire prestazioni gratuitamente e senza alcun intermediario. Ogni persona può quindi mettere a disposizione abilità e competenze ed essere retribuita in base al proprio lavoro, in totale libertà.

FONTI

Bibliografia

- E. Bonifazi, F. Franceschi, F. Ricciardelli, A. Quercioli, *Ecostoria*, Bulgarini
- S. Guarracino, *Storia degli ultimi sessant'anni*, Bruno Mondadori
- T. Detti, G. Gozzini, *Storia contemporanea 2*, Bruno Mondadori
- A. De Bernardi, S. Guarracino, *I saperi della storia 3*, Bruno Mondadori
- S. Vergine, *Più liberi ma più precari*, in «La vita condivisa» (Dossier), «L'Espresso», LXI, 2015, pp. 44-47

Sitografia

- [treccani.it/enciclopedia/new-economy_\(Dizionario-di-Economia-e-Finanza\)/](http://treccani.it/enciclopedia/new-economy_(Dizionario-di-Economia-e-Finanza)/)
- tpi.it/mondo/europa/sharing-economy-car-sharing-italia-europa
- adapt.nova100.ilsole24ore.com/2015/01/31/le-grandi-trasformazioni-del-lavoro-un-tentativo-di-periodizzazione/
- blog.pixell.it/come-si-crea-un-buon-marchio/
- yourinspirationweb.com/2012/10/08/i-principi-della-progettazione-grafica-la-sezione-aurea-nel-web-design/
- yourinspirationweb.com/2013/01/14/grafica-web-e-psicologia-del-colore-come-i-colori-comunicano-con-linconscio/
- businessplanvincente.com/modello-di-business-plan/gli-obiettivi-dellazienda
- masteruniversity.org/blog/marketing/piano-marketing-differenza-mission-vision/1425/
- impresaperta.it/cinque-fattori-chiave-per-incardinare-i-valori-aziendali/
- abitudinicreative.it/2017/02/definizione-di-branding.html
- cpiub.com/2013/10/cose-il-branding-e-perche-ti-serve/
- ciquattrostudio.it/naming-strategy/
- linkedin.com/pulse/web-design-ux-ui-qual-%C3%A8-la-differenza-valerio-foddai
- doctorbrand.it/2012/12/cose-la-user-experience/
- webhouseit.com/wireframe-prototype-e-mockup-dove-sta-la-differenza/
- www.ce.uniroma2.it/~lopresti/Didattica/RetiWeb/RetiWeb1213/HTTP_completo4.pdf