



Università degli Studi di Padova
Corso di laurea in Informatica



Nome del sito

Relazione sul progetto di Tecnologie Web
a.a. 2017/2018

Componenti del gruppo

Sara Feltrin	9999999
Lisa Parma	1121284
Silvia Bazzoato	6666666

eyrugveoruvgevgeqfuhvqer8hv

Indice

1	Introduzione	2
1.1	Abstract	2
1.2	Utenti desctinatari	2
2	Progettazione	2
2.1	Database	2
3	Struttura	2
3.1	scjdissd	2
3.1.1	Nor chiu iuhhiu	3
4	Presentazione	3
4.1	scjdissd	3
4.2	dihdshdsh	3
4.2.1	Nor chiu iuhhiu	3
5	Comportamento	3
5.1	PHP	3
5.1.1	Strutturazione pagine	3
5.1.2	Ricerca	4
5.1.3	Registrazione	4
5.1.4	Login	4
5.1.5	Area personale utente	4
5.1.6	Area personale aziende	4
5.1.7	Area personale amministratore	4
5.2	JavaScript	4
5.2.1	Validazione form registrazione utenti e aziende	4
5.2.2	Ridimensionamento header	4
5.2.3	Evidenziare nel menù la pagina in cui sono	5
5.2.4	Menù ad hamburger per mobile	5
6	Accessibilità	5
6.1	scjdissd	5
6.1.1	Nor chiu iuhhiu	5
7	Divisione del lavoro	5

1 Introduzione

1.1 Abstract

Il progetto Jomp si propone di implementare un sito internet per facilitare la ricerca di lavoro da parte degli utenti e la ricerca di personale da parte delle aziende. Il sito dispone di una semplice e intuitiva barra di ricerca in cui si possono filtrare le proposte di lavoro per titolo dell'annuncio, luogo di lavoro e tipologia di impiego, dando luogo all'elenco di annunci corrispondenti. Il sito è stato sviluppato rispettando gli standard W3C, la separazione tra struttura, presentazione, comportamento e le regole di accessibilità richieste.

1.2 Utenti destinatari

Come menzionato precedentemente, il sito è rivolto a chi cerca o offre un lavoro, ragion per cui il target è definito da soggetti adulti (la registrazione può avvenire solo se si è maggiorenni). Al sito possono accedere diverse categorie di utenti: -gli utenti registrati che possono salvare gli annunci per loro più interessanti e vedere informazioni più dettagliate riguardo all'annuncio; -le aziende registrate che possono pubblicare i loro annunci di lavoro; -gli utenti non registrati che posso ricercare annunci senza però poterli salvare o vedere i dettagli dell'offerta di lavoro (le aziende non registrate non possono aggiungere annunci di lavoro); -l'amministratore che gestisce il sito può rimuovere utenti, aziende e annunci.

2 Progettazione

2.1 Database

Il sito, per funzionare correttamente, è collegato(?) ad un database costruito in linguaggio SQL. È stato scelto questo linguaggio in quanto tutti i componenti del gruppo lo conoscono e ne hanno dimestichezza. All'interno del database vi sono diverse tabelle, le principali sono Utenti, Aziende e Annunci, e ognuna di esse si appoggia ad altre tabelle minori come Consultazioni, Tipo, OrarioLavoro, ContrattoLavoro. Le ultime tre, in particolare, raccolgono la tipologia di caratteristiche che l'annuncio può assumere e si è scelto di inserirle in una tabella per evitare che l'utente possa modificare la voce nel menù a tendina (?).

3 Struttura

3.1 scjdisd

rvaefvafdgvaergvaergvae

boooooo

3.1.1 Nor chiu iuhhiu

fgsbfgsb sfg sfv

4 Presentazione

4.1 scjdissd

rvaefvafdgvaergvaergvae

4.2 dihdshdsh

4.2.1 Nor chiu iuhhiu

fgsbfgsb sfg sfv

5 Comportamento

5.1 PHP

Dichiamo in generale che lo abbiamo utilizzato per tutto e che è figo. oppure non scriviamo nulla qui

5.1.1 Strutturazione pagine

Abbiamo usato funzioni comuni a tutte le pagine

5.1.2 Ricerca

5.1.3 Registrazione

5.1.4 Login

5.1.5 Area personale utente

5.1.6 Area personale aziende

5.1.7 Area personale amministratore

5.2 JavaScript

Il linguaggio di scripting JavaScript è stato utilizzato per migliorare l'usabilità del sito. Si è tenuto conto del fatto che JavaScript può non funzionare su tutti i dispositivi che possono accedere al sito web (per non compatibilità o per preferenze dell'utente) così sono state assicurate funzioni equivalente server side con PHP.

In particolare con JavaScript sono state aggiunte le seguenti funzionalità:

5.2.1 Validazione form registrazione utenti e aziende

Prima di inviare i dati raccolti al server per controllarli con PHP, sono stati aggiunti dei controlli campo per campo che se non soddisfatti non permettono di inviare i dati al server. I controlli sono eseguiti *onBlur* su ogni input e permettono di controllare lunghezza massima e minima, se un certo dato soddisfa una certa espressione regolare (definite per e-mail e sito internet aziende) e se la ripetizione della password corrisponde alla prima inserita. Questi controlli real-time sono di grande aiuto all'utente per sapere subito se i dati inseriti corrispondono alla richiesta o c'è la necessità di modificarli.

Inoltre è servito anche un controllo per la gestione dell'input type *data*, tipo di input di HTML5 non gestito correttamente in tutti i browser. Nei browser in cui si vede l'input come stringa preformata (*gg/mm/aaaa*) o come calendario viene effettuato un controllo solo per vedere se l'input è stato immesso, negli altri browser dove invece si vede un normale input di testo viene eseguito il controllo anche sul formato dei dati immessi, che deve essere del tipo *aaaa/mm/gg* per aderire al type *data*.

5.2.2 Ridimensionamento header

Quando avviene uno scroll nelle pagine del sito, l'*header* (che per proprietà CSS risulta fisso) viene ridimensionato così da occupare meno spazio nella pagina visibile.

Per fare ciò attraverso JavaScript viene aggiunta dinamicamente la classe *small* all'*header* quando l'offset verticale della pagina risulta maggiore dell'offset iniziale cambiandone le proprietà grafiche. Quando si torna ad inizio pagina questa classe viene tolta facendo tornare l'*header* con le proprietà iniziali.

5.2.3 Evidenziare nel menù la pagina in cui sono

Nel menù principale è stato scelto di evidenziare la pagina in cui ci si trova attraverso JavaScript così da evidenziare le differenze che ci sono state tra questa tecnica e quella che è stata fatta per i menù dentro le pagine personale in cui viene evidenziato ciò tramite PHP.

La funzione, molto semplice, prevede solamente un confronto tra l'attributo *href* dentro i tag *li* del menù e l'URL della pagina corrente. Quando viene trovata una corrispondenza viene aggiunta una classe che ne modifica le proprietà grafiche tramite CSS. Rispetto alla soluzione PHP di passare una variabile con la pagina corrente alla funzione che stampa il menù e di gestire molti sottocasi (uno per pagina) risulta una soluzione molto più veloce e semplice. Il suo limite però è nel fatto che JavaScript può essere disabilitato/non andare, mentre PHP no.

Dato che sotto l'*header* è presente il *breadcrumb* con la pagina corrente, anche in caso di disattivazione di JavaScript non si rischia che l'utente non capisca più in che pagina si trova.

5.2.4 Menù a tendina

Anche per il menù a tendina si è utilizzata la tecnica di aggiungere una classe ad un secondo menù che permettesse di cambiare l'attributo *display*. L'aggiunta di questa classe è data dall'evento *onClick* di un elemento (immagine menu ad hamburger) che è resa visibile da una media query solamente sotto gli 840px, quindi non solamente per mobile ma anche per finestre browser ristrette.

6 Accessibilità

6.1 scjdisd

rvaefvafdgvaergvaergvae

6.1.1 Nor chiu iuhhiu

fgsbfgsb sfg sfv

7 Divisione del lavoro

Lisa Parma:

- Struttura e comportamento pagine che non richiedono una sessione attiva
- Pagine personali utente
 - UtDashboard.php

booooooo

- UtCercaAnnuncio.php
- UtAnnunciSalvati.php
- UtModificaDati.php
- CSS (file: desktop.css)
- JavaScript (file: javascript.js)

Sara Feltrin:

- Struttura e comportamento pagine che non richiedono una sessione attiva
- Pagine personali azienda
 - AzDashboard.php
 - AzPubblicaAnnuncio.php
 - AzResocontoAnnunci.php
 - AzModificaDati.php
- Pagine personali amministratore
 - AdminDashboard.php
 - AdminModUtenti.php
 - AdminModAziende.php
 - AdminModAnnunci.php
- Database (file: database.sql)

Silvia Bazzato:

- boh