Documentazione progetto LavoroTemporaneo

1 Introduzione 3

1.1 Informazioni sul progetto 3

1.2 Abstract 3

1.3 Scopo 3

2 Analisi 4

2.1 Analisi del dominio 4

2.2 Analisi e specifica dei requisiti 4

2.3 Use case 7

2.4 Pianificazione 7

2.5 Analisi dei mezzi 7

2.5.1 Software 7

2.5.2 Hardware 7

3 Progettazione 8

3.1 Design dell’architettura del sistema 8

3.2 Design dei dati e database 8

3.3 Design delle interfacce 8

3.4 Design procedurale 8

4 Implementazione 9

5 Test 9

5.1 Protocollo di test 9

5.2 Risultati test 10

5.3 Mancanze/limitazioni conosciute 10

6 Consuntivo 10

7 Conclusioni 10

7.1 Sviluppi futuri 10

7.2 Considerazioni personali 10

8 Bibliografia 10

8.1 Bibliografia per articoli di riviste: 10

8.2 Bibliografia per libri 10

8.3 Sitografia 11

9 Allegati 11

# Introduzione

## Informazioni sul progetto

Questo progetto viene realizzato dall’allievo Miro Joos, il docente Guido Montalbetti prende la parte del cliente.

Il progetto viene svolto al terzo anno della sezione di informatica alla scuola SAM Trevano.

La data di inizio del progetto è il 30 agosto 2021, mentre la consegna è prevista per il 23 dicembre 2021.

## Abstract

### Situazione iniziale

*Per questo progetto mi è stato chiesto di realizzare un applicativo web in grado di metter in contatto un potenziale Lavoratore con un Datore di Lavoro per un lavoro temporaneo (di breve durata).*

*Prima del compimento di questo progetto se uno studente voleva cercare un lavoro estivo poteva arrivare a perdere ore intere prima di trovarlo, grazie a questo applicativo nel lasso temporale di pochi minuti è possibile candidarsi per un lavoro part-time.*

### Attuazione

*Questo progetto richiede competenze necessarie per sviluppare un applicativo web e competenze sufficienti per sviluppare un database. È richiesto programmare delle pagine web che permettano di registrare, eliminare e modificare i dati all’interno del database. L’utente lavoratore dovrà poter candidarsi per delle offerte di lavoro, l’utente datore di lavoro dovrà poter pubblicare delle offerte di lavoro, infine l’utente amministratore dovrà poter gestire la fatturazione ed eseguire delle stampe.*

### Risultati

*Questo applicativo è realizzato da zero utilizzando i principali linguaggi WEB (HTML, CSS, PHP e Javascript). Per la parte di bancadati è stato utilizzato SQL.*

*Per lo sviluppo di questo progetto sono state necessarie competenze che ancora non possedevo, ho dunque dovuto fare un grande lavoro di documentazione.*

*L’intero applicativo risulta essere piuttosto semplice anche per l’utente medio senza basi dell’informatica.*

*Questo progetto permette di risparmiare diverso tempo sia per i giovani che di eventuali datori di lavoro, quali necessitano una mano per una data attività.*

*Alcuni giovani sono alla ricerca di un lavoretto per tirar su alcuni soldi, ora finalmente avranno un applicativo sul quale potersi affidare, questo sito web è un portale semplice e intuitivo che fornirà loro ciò di cui sono alla ricerca.*

## Scopo

Lo scopo didattico del progetto è quello di imparare a gestire nel modo più ottimale possibile un progetto IT. Il docente si immedesimandosi nel ruolo del cliente da possibilità all’allievo di affrontare per la prima volta una situazione simile a quelle con le quali ci si può confrontare nel mondo lavorativo.

Questa esperienza servirà all’allievo per prepararsi all’esame pratico di fine quarta.

Lo scopo operativo riguarda il creare un applicativo WEB dove si possano pubblicare, accettare e gestire proposte di lavoro. Le offerte di lavoro dovranno essere archiviate in modo da poterle visualizzare e / o stamparle in un secondo momento. Dovranno venir gestiti tre tipi differenti di utenti (amministratore, datore di lavoro e lavoratore), registrati in un database con tutti i rispettivi dati.

# Analisi

## Analisi del dominio

Spesso per poter trovare un lavoro estivo o part-time diversi giovani passano giornate intere a chiamare datori di lavoro o a navigare su diversi siti internet, il problema è che molto raramente si trovano delle offerte di lavoro a breve termine o per un part-time. Inoltre i contatti con i datori di lavoro risultano molte volte essere laboriosi, richiedendo fatica e tempo.

Il mio applicativo dovrà permettere a datori di lavoro che offrono lavori a breve termine di pubblicare annunci, mentre alle persone che sono in cerca di un lavoro part-time o estivo di visualizzare i vari annunci e contattare il potenziale datore di lavoro in pochi semplici click.

Sarà inoltre possibile sia per i datori di lavoro che per i giovani lavoratori visualizzare le fatturazioni, in modo da tener traccia dei propri guadagni e delle proprie spese.

Per accedere all’interfaccia del mio applicativo sarà necessario un dispositivo con accesso a internet e una conoscenza basilare per riuscire a registrarsi a un sito web munito di registrazione con conferma mail.

## Analisi e specifica dei requisiti

Questo capitolo è dedicato ai requisiti presenti nel QdC, con i rispettivi sotto requisiti.

Come primo requisito ho inserito la verifica dell’hosting dove il mio applicativo verrà caricato, senza la quale non sarebbe accessibile da nessun utente. Successivamente ho aggiunto i requisiti per la realizzazione del login, grazie al quale utenti, datori di lavoro e amministratori potranno interagire con il sito internet.

I successivi tre requisiti riguardano funzionalità utilizzabili dagli utenti.

Il settimo e l’ottavo requisito riguardano il corretto comportamento dei menu a dipendenza del tipo di utente loggato.

Infine gli ultimi requisiti sono specificati in modo da offrire più funzionalità agli utenti registrati, ma comunque fornire un utilizzo base a quelli utenti che stanno utilizzando il sito per la prima volta senza essersi ancora registrati.

|  |  |
| --- | --- |
| **ID: REQ-001** | |
| **Nome** | Verifica hosting |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | È necessario verificare che l’hosting funzioni correttamente. |

|  |  |
| --- | --- |
| **ID: REQ-002** | |
| **Nome** | Verifica funzionamento login |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | È necessario verificare la funzionalità di registrazione al sito |
| **Sotto requisiti** | |
| **001** | La registrazione dovrà prevedere un meccanismo di conferma tramite email |

|  |  |
| --- | --- |
| **ID: REQ-003** | |
| **Nome** | Verifica funzionamento pubblicazione offerta di lavoro |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | È necessario verificare se sia possibile pubblicare un annuncio di offerta di lavoro |

|  |  |
| --- | --- |
| **ID: REQ-004** | |
| **Nome** | Verifica funzionamento rimozione offerta di lavoro |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | È necessario verificare la funzionalità riguardante la rimozione di un offerta di lavoro |
| **Sotto requisiti** | |
| **001** | L’utente deve aver la possibilità di cancellare le proprie offerte di lavoro |

|  |  |
| --- | --- |
| **ID: REQ-005** | |
| **Nome** | Verifica funzionamento proporsi a un’offerta di lavoro |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | È necessario verificare la funzionalità riguardante il proporsi a un posto di lavoro |
| **Sotto requisiti** | |
| **001** | L’utente deve aver la possibilità di caricare dei documenti, es: Curriculum Vitae, Diplomi, ecc… |
| **002** | Completata la candidatura l’utente riceverà tramite eMail la conferma della loro iscrizione al posto di lavoro |

|  |  |
| --- | --- |
| **ID: REQ-006** | |
| **Nome** | Verifica funzionamento fatturazione |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | È necessario verificare la funzionalità riguardante la fatturazione |
| **Sotto requisiti** | |
| **001** | Costi gestione incarti dell’agenzia |
| **002** | L’utente potrà stampare le fatturazioni |

|  |  |
| --- | --- |
| **ID: REQ-007** | |
| **Nome** | Menu utenti |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | È necessaria una pagina dedicata agli utenti |
| **Sotto requisiti** | |
| **001** | Si necessita una funzionalità di login |
| **002** | Il datore di lavoro potrà pubblicare un’offerta di lavoro |
| **003** | Il datore potrà rimuovere una sua offerta di lavoro |
| **004** | Il lavoratore potrà accettare un’offerta di lavoro |

|  |  |
| --- | --- |
| **ID: REQ-008** | |
| **Nome** | Menu amministratori |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | È necessaria una pagina dedicata agli amministratori |
| **Sotto requisiti** | |
| **001** | Si necessita una funzionalità di login |
| **002** | L’amministratore potrà modificare le offerte di lavoro |
| **003** | L’amministratore potrà rimuovere le offerte di lavoro |
| **004** | L’amministratore potrà visualizzare le offerte di lavoro occupate |
| **005** | L’amministratore potrà verificare se sono state fatturate le ore di lavoro |
| **006** | Fatturare i lavori eseguiti dai collaboratori ai datori di lavoro |

|  |  |
| --- | --- |
| **ID: REQ-009** | |
| **Nome** | Verifica funzionamento sito utente non registrato |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | È necessario verificare il funzionamento del sito senza essersi registrati |
| **Sotto requisiti** | |
| **001** | L’utente deve aver la possibilità di ricercare e stampare i lavori offerti dal sito |

|  |  |
| --- | --- |
| **ID: REQ-010** | |
| **Nome** | Verifica funzionamento sito utente registrato |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** | È necessario verificare il funzionamento del sito dopo essersi registrati |
| **Sotto requisiti** | |
| **001** | L’utente può proporsi per un posto di lavoro |
| **002** | L’utente può offrire un posto di lavoro |

## Use case

I casi d’uso rappresentano l’interazione tra i vari attori e le funzionalità del prodotto.

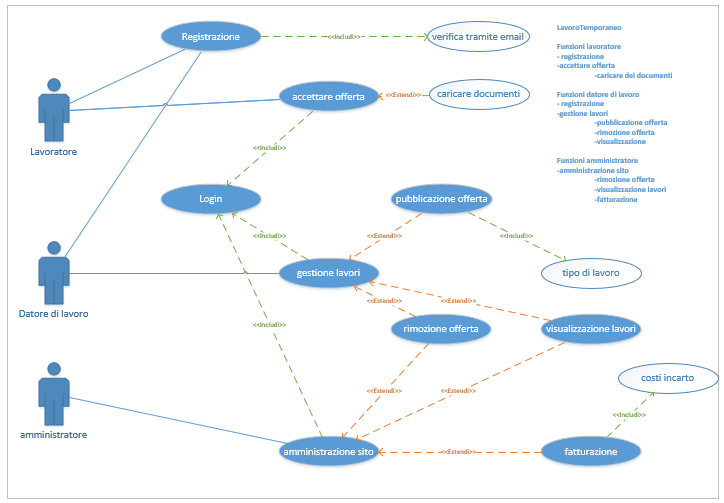


Figura 1: UseCase

All’interno dell’applicativo ho inseriti tre tipi diversi di account:

* **Lavoratore:** Il lavoratore è un account realizzato per gli utenti che fanno uso dell’applicativo web per cercare un lavoro. L’utilizzatore dell’applicativo quando si trova per la prima volta a navigare sul sito web ha la possibilità di registrarsi come lavoratore, la registrazione comprende una email di verifica. Una volta registrato come tale può accettare delle offerte di lavoro, con la possibilità di caricare dei documenti per la candidatura.
* **Datore di lavoro:** Il datore di lavoro è un account realizzato per gli utenti che fanno uso dell’applicativo web per cercare della manodopera. L’utilizzatore dell’applicativo quando si confronta per la prima volta con l’interfaccia del sito ha la possibilità di registrarsi come datore di lavoro, tale registrazione comprende una email di verifica. Una volta registrato come datore di lavoro l’utente potrà accedere al menu di gestione lavori, dal quale potrà: pubblicare delle offerte di lavoro, visualizzare le offerte pubblicate e rimuoverle.
* **Amministratore:** L’amministratore è un tipo di account realizzato per la gestione della fatturazione e la gestione dei lavori pubblicati. Questo tipo di account non comprende una registrazione da Form, dunque sarà possibile aggiungere amministratori unicamente interagendo direttamente con il DB.

## Pianificazione

Il tempo a disposizione per la realizzazione del progetto era dal 16 settembre 2021 al 23 dicembre 2021.

La settimana lavorativa si componeva di 6h.

Come si può vedere nella pagina successiva il modello per la pianificazione che ho scelto è il modello waterfall / a cascata.

Ho scelto questo modello poichè essendo questo il mio primo progetto ho pensato che un modello semplice e intuitivo come il waterfall fosse ideale.

Ho suddiviso le attività necessarie alla realizzazione del progetto in 6 macro categorie:

* **Analisi:** Il contenuto all’interno di questa categoria è composto da tutte quelle attività di comprensione della consegna e analisi del progetto.
* **Progettazione:** Questo gruppo contiene la parte di progettazione del progetto, essa comprende il design delle pagine Web e il design del Database.
* **Implementazione:** All’interno di questa categoria si trova il cuore del progetto, la parte di programmazione delle interfacce. Ho ritenuto necessario assegnare ad ogni interfaccia una stima del tempo necessario per la scrittura del codice, in modo da limitare il più possibile eventuali ritardi.
* **Test:** Questa categoria è composta dal test di tutte le funzionalità dell’applicativo Web. Come per la programmazione delle interfacce ho ritenuto fosse opportuno stimare il tempo necessario per eseguire ogni singolo test. I test sono stati effettuati gradualmente per facilitare l’individuazione di errori.
* **Integrazione:** La categoria di integrazione contiene l’unione di tutte le pagine Web, collegando ogni interfaccia con i rispettivi percorsi. Nell’integrazione ho compreso dei test, per verificare che tutti i percorsi siano corretti e le interfacce funzionino ancora correttamente.
* **Chiusura:** La chiusura comprende la parte finale del progetto, ovvero tutte quelle attività che necessitavano la fine del progetto per venire svolte. In questa macro insieme ho inserito la redazione dell’abstract che fornisce una breve rappresentazione dei contenuti, il controllo della documentazione e la rilegatura dei documenti prima della consegna

Per la durata di tutte le attività ho deciso di continuare parallelamente la documentazione. Ho pensato tale approccio fosse il migliore per documentare quasi immediatamente le attività fatte, in modo da non rischiare di tralasciare nulla.

|  |
| --- |
| Figura : Diagramma di Gantt |

## Analisi dei mezzi

Elencare e descrivere i mezzi disponibili per la realizzazione del progetto. Ricordarsi di sempre descrivere nel dettaglio le versioni e il modello di riferimento.

### Software

* **NetBeans**
* **WorkBench**
* **Project**
* **Visio**
* **Sqldbm**
* **Xampp**
* **Word**

### Hardware

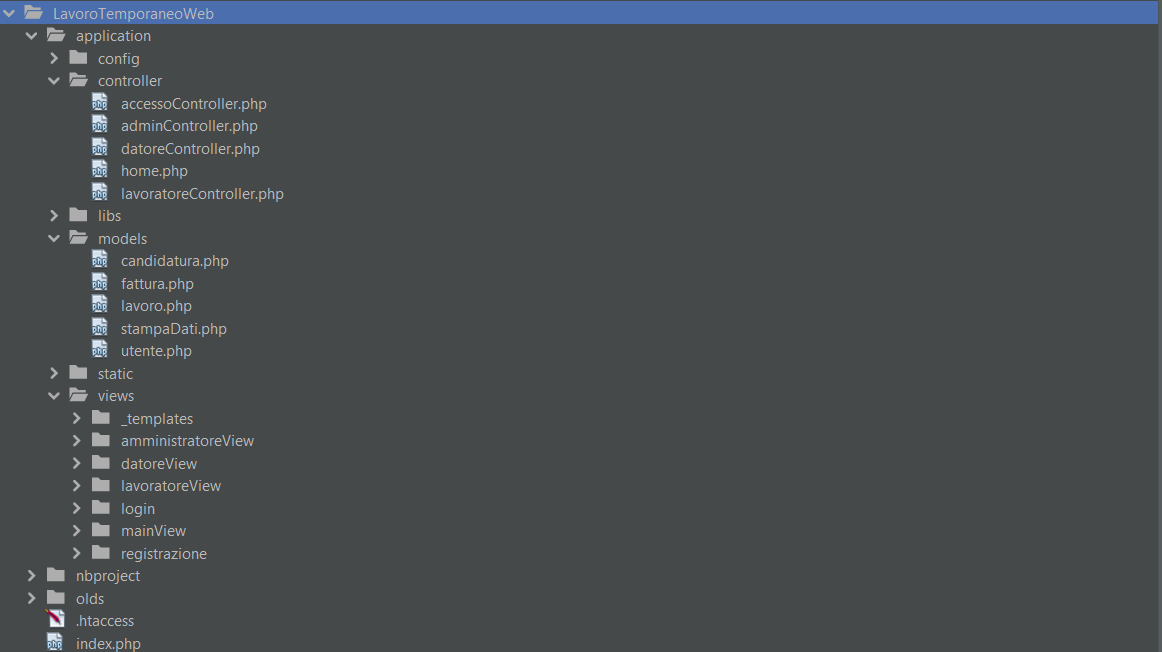
Il prodotto dovrà essere eseguito su un qualsiasi browser, dunque sarà necessario un qualsiasi hardware che abbia accesso a internet e la possibilità di navigare online tramite un browser.

Durante lo sviluppo è stato utilizzato un desktop e un laptop.

# Progettazione

## Design dell’architettura del sistema

L’applicativo Web è stato realizzato sulla base di un MVC.



* **Config:** All’interno di questa cartella sono presenti tutti i file di configurazione dell’applicativo Web.
* **Controller:** Tale cartella contiene tutti i controller del progetto. Questi file sono pensati per far da tramiti tra la comunicazione di Views e Models.
* **Libs:** Sono contenute tutte le librerie utilizzate da più file.
* **Models:** Questa cartella contiene tutte le classi che si occupano di gestire i dati.
* **Static:** All’interno di questa directory sono presenti tutti quei file di supporto che vengono puntati dalle interfacce. Ho scelto di collocare sotto un’unica cartella tali contenuti in modo da avere sempre lo stesso percorso per far riferimento agli stessi file.
* **Views:** Questa cartella contiene tutte le parti grafiche delle interfacce del progetto. Si tratta di file php contenenti riferimenti ai controller.

Di default viene caricato il controller home.php, il quale rimanda all’interfaccia mainView. Da essa sarà possibile accedere alla registrazione e al login.

Quando l’utente manda una richiesta di registrazione o di login il Form fa riferimento al file accessoController.php, il quale contiene una classe che gestisce tutte le operazioni necessarie per la registrazione di un account nel Database o per un Login.

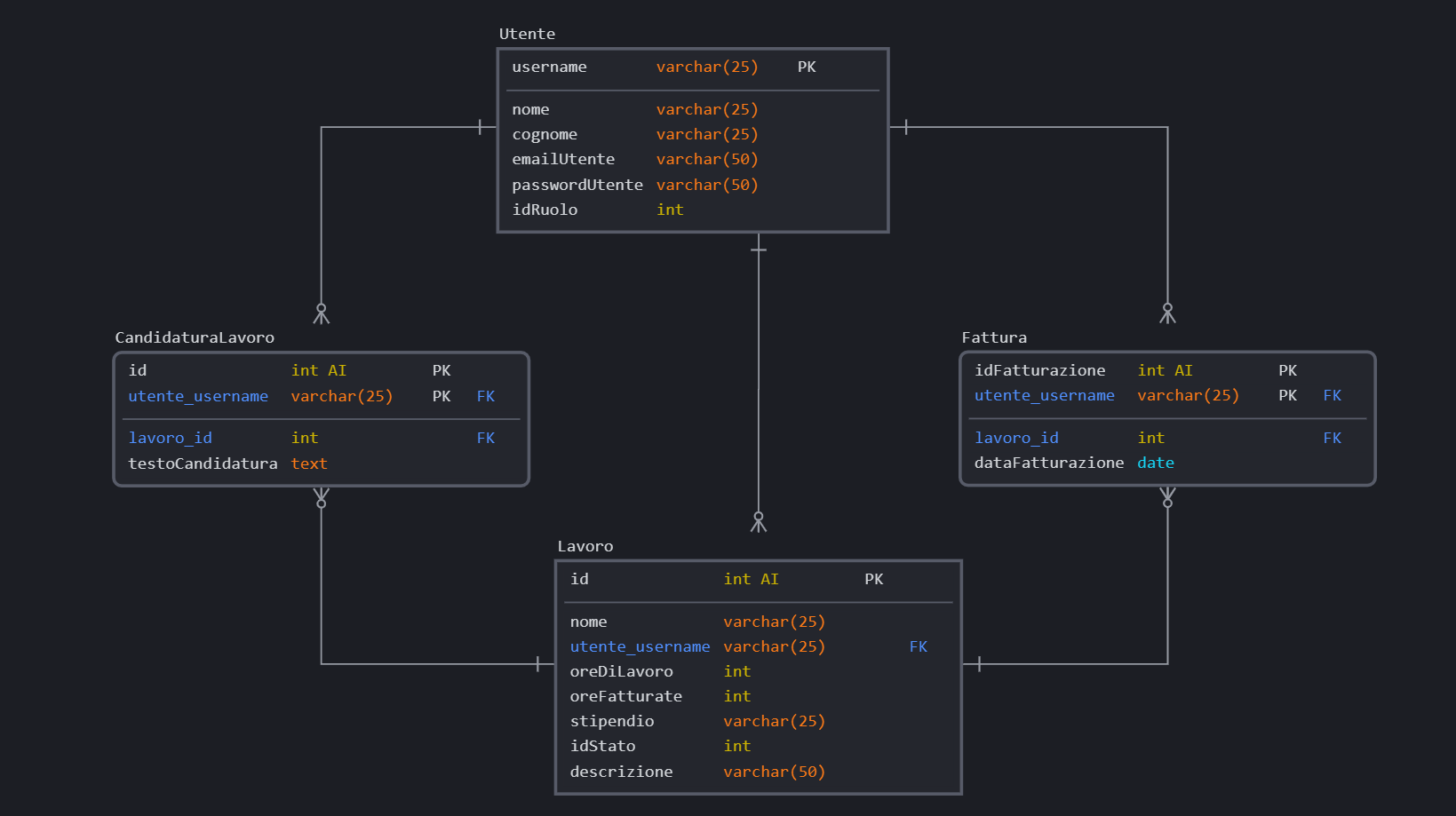
Una volta eseguito l’accesso, a seconda del tipo di account (amministratore, lavoratore o datore di lavoro), verrà caricata l’interfaccia rispettiva.

Ogni interfaccia possiede un controller specifico, in modo da suddividere la gestione delle operazioni effettuabili a seconda dell’accesso utilizzato.

Nella cartella “models” sono presenti delle classi correlate a ogni tabella presente nel Database. Ho deciso di usare questo approccio per poter suddividere le operazioni a seconda delle tabelle interessate dall’azione.

Nella directory “libs” ho salvato tutti gli stili e i file di supporto utilizzati dalle interfacce, così facendo ho potuto generalizzare un percorso valido per più file, velocizzando il lavoro e semplificando la struttura.

## Design dei dati e database

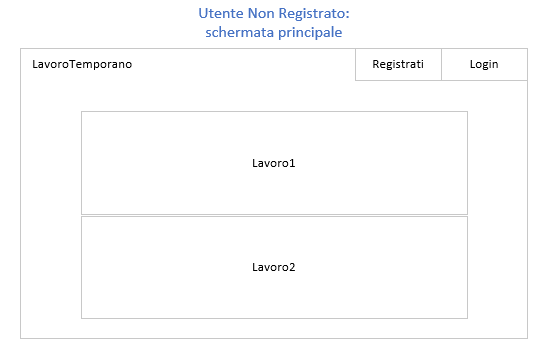


La struttura del database si basa su 4 tabelle:

* **Utente:** All’interno di questa tabella verranno registrati tutti gli utenti che faranno uso dell’applicativo web. Lo username viene usato come chiave primaria, non potranno esserci dunque due o più username uguali. L’attributo idRuolo viene utilizzato per definire la tipologia di utente: 2 equivale a “Lavoratore”; 1 equivale a “Datore di Lavoro”; 0 equivale a “Amministratore”.
* **Lavoro:** Questa tabella conterrà tutti i lavori generati dai “Datori di Lavoro”. Come chiave primaria è utilizzato un id, inoltre la tabella dispone di una ForeignKey per aver un riferimento allo username dell’utente che ha pubblicato il lavoro.
* **CandidaturaLavoro:** Tale tabella verrà utilizzata per registrare tutte le candidature mandate dagli utenti “Lavoratore” per proporsi a un’offerta di lavoro. Questa tabella utilizza come chiave primaria un id e presenta 2 ForeignKey: un riferimento all’utente che ha generato la candidatura e un riferimento al lavoro per il quale è stata proposta.
* **Fattura:** All’interno di questa tabella saranno registrate tutte le fatturazioni per il compenso degli utenti “Lavoratore” a seguito del completamento di un lavoro. Anche essa si distingue con una chiave primaria id e ha come ForeignKey un riferimento allo username dell’account che l’ha generata e uno al lavoro per la quale è stata redatta.

## Design delle interfacce

#### Interfaccia mainView, <<utente non registrato>>



La prima interfaccia riguarda la mainView dell’applicativo web. Tale interfaccia viene mostrata al caricamento del sito. In questa interfaccia vengono mostrati tutte le offerte di lavoro pubblicate dagli utenti “Datore di Lavoro”.

#### Interfaccia lavoro selezionato, utente <<non registrato>>



Questa interfaccia serve per non permettere ad utenti non registrati di formulare delle candidature, ma tuttavia permettere a questa fetta di utenti di visualizzare le informazioni dei lavori.

#### Interfaccia mainView, utente “Lavoratore”



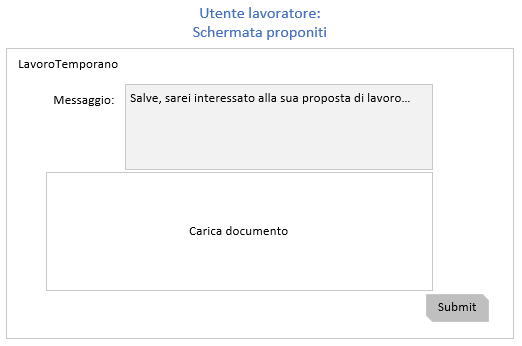
Questa interfaccia è apposita per gli utenti “Lavoratore”. Dalla barra di navigazione è possibile accedere ai lavori accettati.

#### Interfaccia del lavoro selezionato, utente “Lavoratore”



L’interfaccia riguarda il menu di informazioni dell’offerta di lavoro selezionata. Tramite questa interfaccia è possibile aprire l’interfaccia per mandare una candidatura per l’offerta di lavoro.

#### Interfaccia di proposta candidatura, utente “Lavoratore”



Questa interfaccia è stata creata per permettere agli utenti “Lavoratore” di candidarsi con un messaggio e caricando dei documenti per un’offerta di lavoro.

Sta poi al datore di lavoro decidere se accettare o meno la candidatura.

#### Interfaccia Lavori accettati, utente “Lavoratore”



Questa interfaccia serve agli utenti “Lavoratore” per poter visualizzare e gestire i lavori per i quali si sono candidati. Quando un lavoro è stato accettato dal rispettivo datore di lavoro non si può più eliminare la candidatura

#### Interfaccia mainView, utente “Datore di Lavoro”



Questa interfaccia è apposita per gli utenti “Datore di Lavoro”. Dalla barra di navigazione è possibile accedere ai lavori offerti e/o pubblicare un lavoro.

#### Interfaccia pubblica lavoro, utente “Datore di Lavoro”



Questa interfaccia è apposita per gli utenti “Datore di Lavoro”. Tramite la barra di navigazione superiore è possibile accedere all’interfaccia di pubblicazione di un’offerta di lavoro ed ai lavori offerti

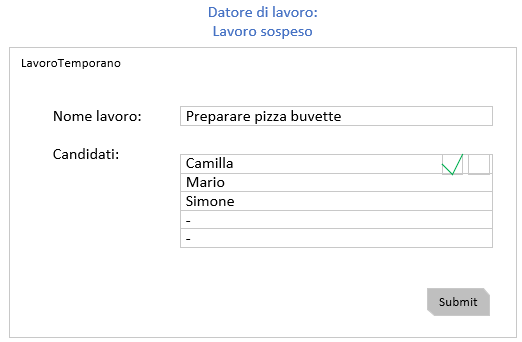
#### Interfaccia lavori offerti, utente “Datore di Lavoro”



Questa interfaccia stampa tutte le offerte di lavoro pubblicate con il proprio account. Direttamente da tale tabella sarà possibile rimuovere o modificare delle offerte di lavoro.

Inoltre è previsto un filtro per visualizzare solamente i lavori offerti / scaduti / sospesi.

#### Interfaccia Lavoro sospeso, utente “Datore di Lavoro”



Questa interfaccia permette agli utenti “Datore di Lavoro” di selezionare una delle candidature proposte.

#### Interfaccia mainView, utente “Amministratore”



Questa interfaccia è stata progettata appositamente per gli utenti amministratori, in modo da permettergli di amministrare il sito web.

.

#### Interfaccia schermata lavori, utente “Amministratore”



Questa interfaccia permette di visualizzare agli amministratori tutte offerte di lavoro pubblicate sul sito web. Essi hanno la possibilità di filtrare le varie pubblicazioni, vedere quelle archiviate e infine accedere alla pagina di fatturazione. Cliccando un’offerta occupata si può visualizzare le informazioni e creare una fatturazione

#### Interfaccia Lavoro occupato, utente “Amministratore”



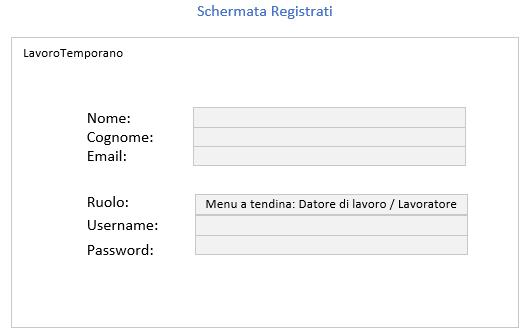
L’interfaccia soprastante servirà all’amministratore per poter gestire le offerte di lavoro pubblicate dall’utente “Datore di Lavoro”.

#### Interfaccia Fatturazioni, utente “Amministratore”



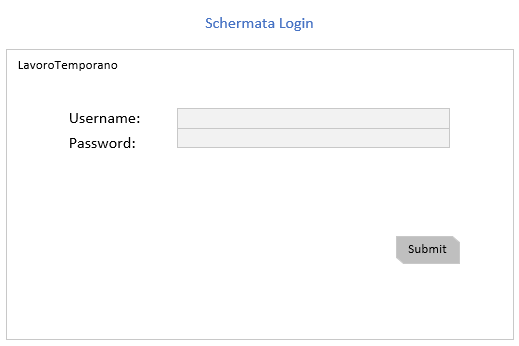
Questa interfaccia serve agli amministratori per gestire le fatture. L’interfaccia offre la possibilità di filtrare gli stipendi dei collaboratori, filtrare secondo una data passata, filtrare solo le fatture derivate dal costo dell’incarto e infine stampare le fatture.

#### Interfaccia schermata di registrazione



La seconda interfaccia rappresenta il menu di login del sito web. Tramite questa interfaccia sarà possibile registrarsi.

#### Interfaccia schermata di Login



Questa interfaccia è stata progettata per conferire agli utenti la possibilità di eseguire un login semplice e veloce.

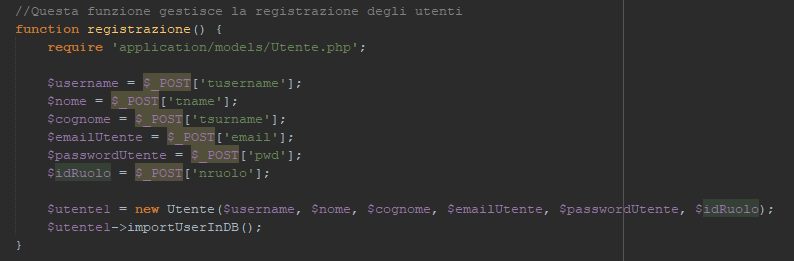
## Design procedurale

### Classe accessoController

Questa classe si occupa di gestire l’accesso degli utenti.

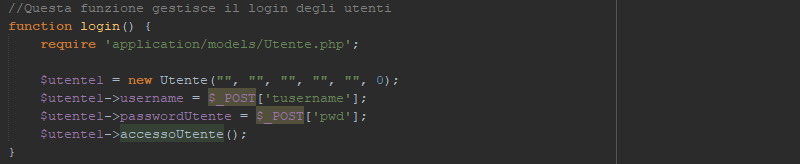
#### Function registrazione

Questa funzione istanzia un nuovo utente con i dati inseriti nel form dall’utente.



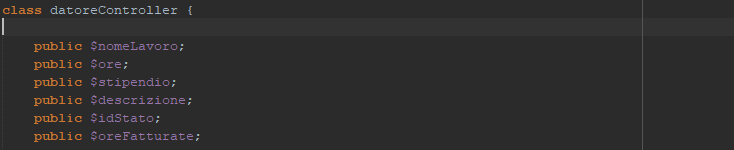
#### Function login

Questa funzione istanzia un nuovo utente e fa uso del metodo “accessoUtente” della classe Utente.



### Classe datoreController

Questa classe si occupa di intercettare le azioni eseguite nella View di un utente “Datore di Lavoro”.

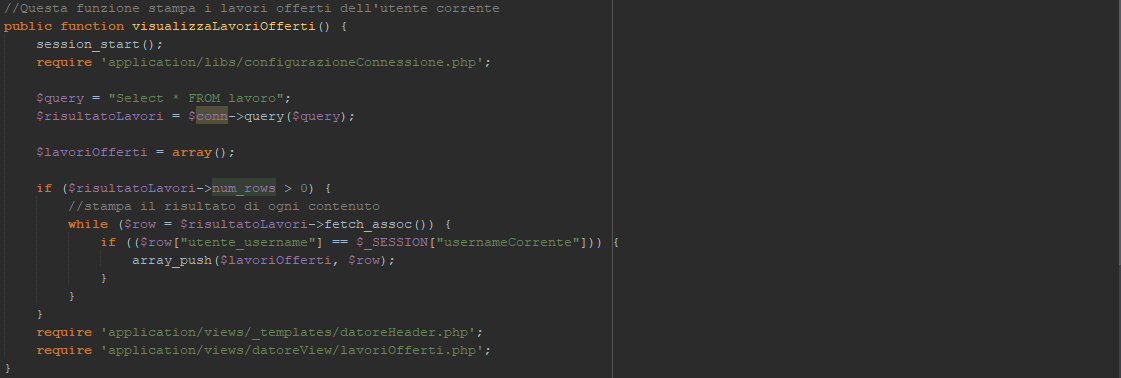


#### Function pubblicazioneLavoro

#### 

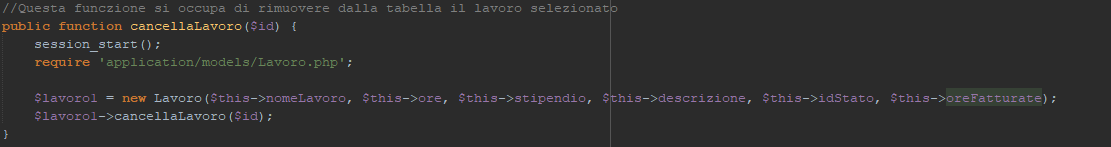
#### Function visualizzaLavoriOfferti

Questa funzione si occupa di caricare la View delle offerte di lavoro che sono state pubblicate dall’utente “datore di lavoro” corrente.



#### Function cancellaLavoro

Questa funzione si occupa di cancellare il lavoro selezionato nella tabella dei lavori offerti.



### Classe home

Questa classe contiene alcuni metodi per accedere direttamente ad alcune View.

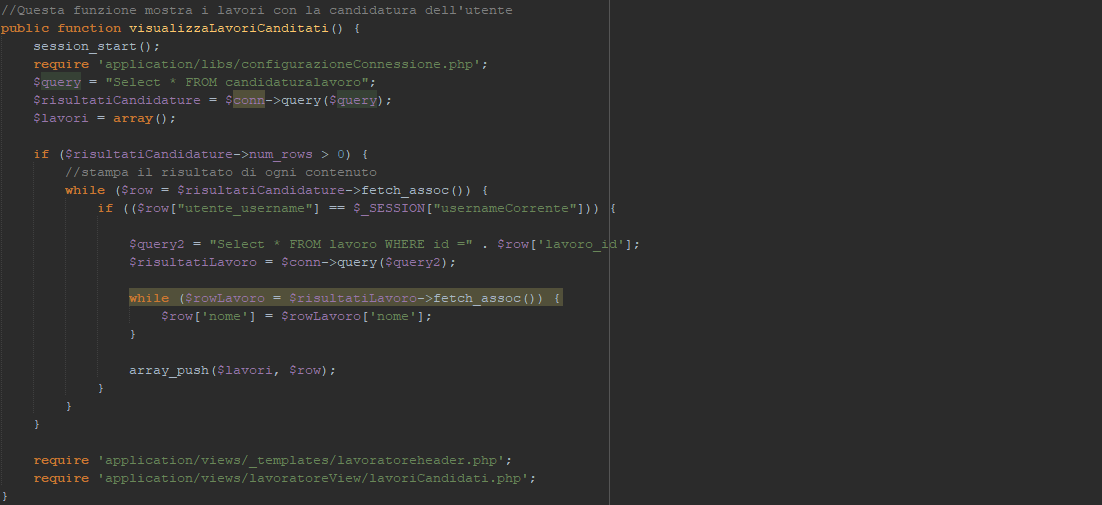


### Classe lavoratoreController

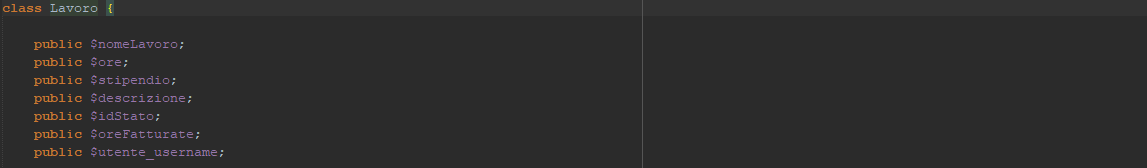
Questa classe gestisce le azioni generate dagli utenti “Lavoratore”.

#### Funzione visualizzaLavoriCandidati

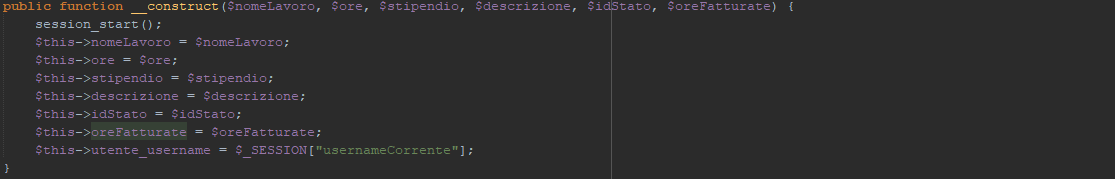
Questa funzione immette in una variabile tutti i lavori collegati allo username di un determinato lavoratore.



### Classe Lavoro

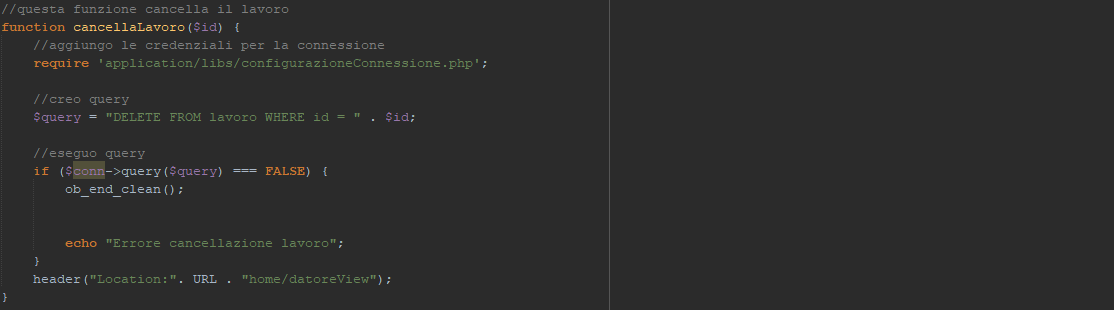
Questa classe rappresenta la tabella “Lavoro” del database. Gli attributi assegnati a questa classe sono gli stessi della tabella nel DB con l’aggiunta dell’informazione dello username dell’utente che provoca la generazione dell’istanza. 

#### Costruttore

Il costruttore assegna i valori degli attributi della classe. 

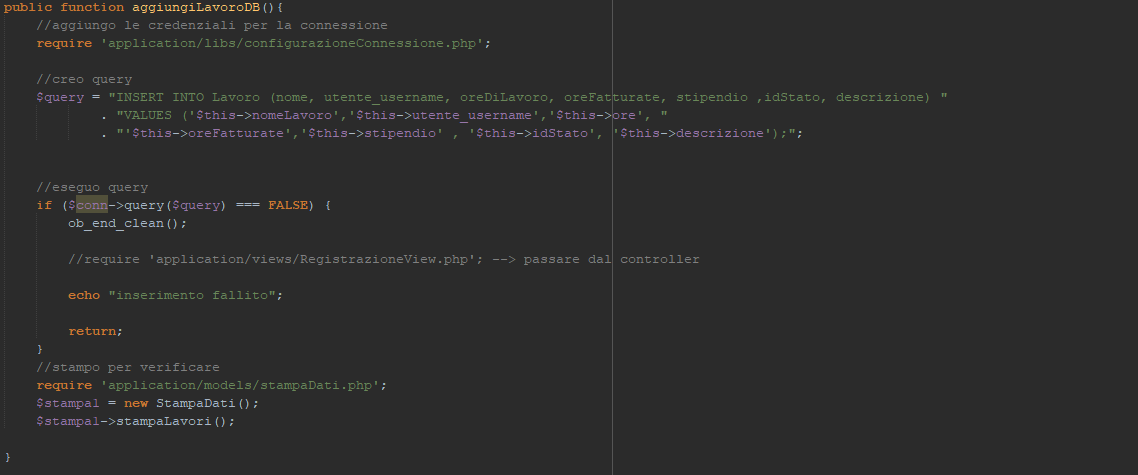
#### Funzione cancellaLavoro

Questa funzione rimuove il lavoro dalla tabella lavoro dell’id passato come parametro.



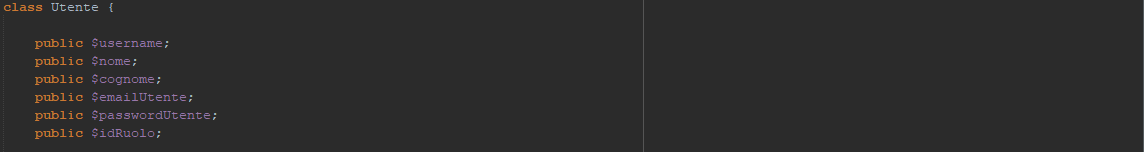
#### Funzione aggiungiLavoroDB

Questa funzione aggiunge la classe Lavoro al database tramite una query.



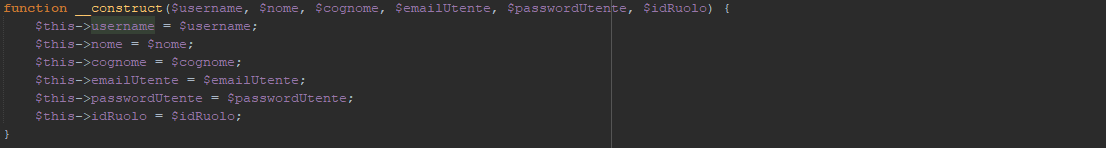
### Classe Utente

Questa classe rappresenta un utente, essa ha gli stessi attributi della tabella Utente nel DB.

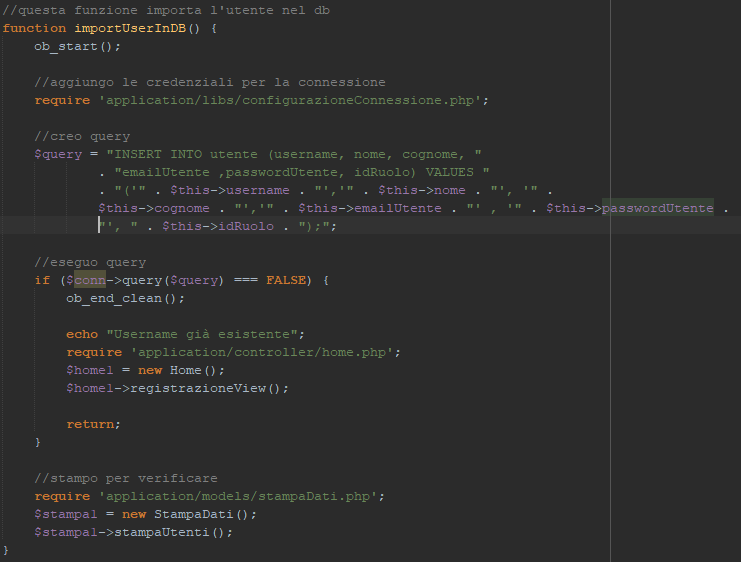


#### Costruttore

Il costruttore assegna i valori degli attributi della classe.

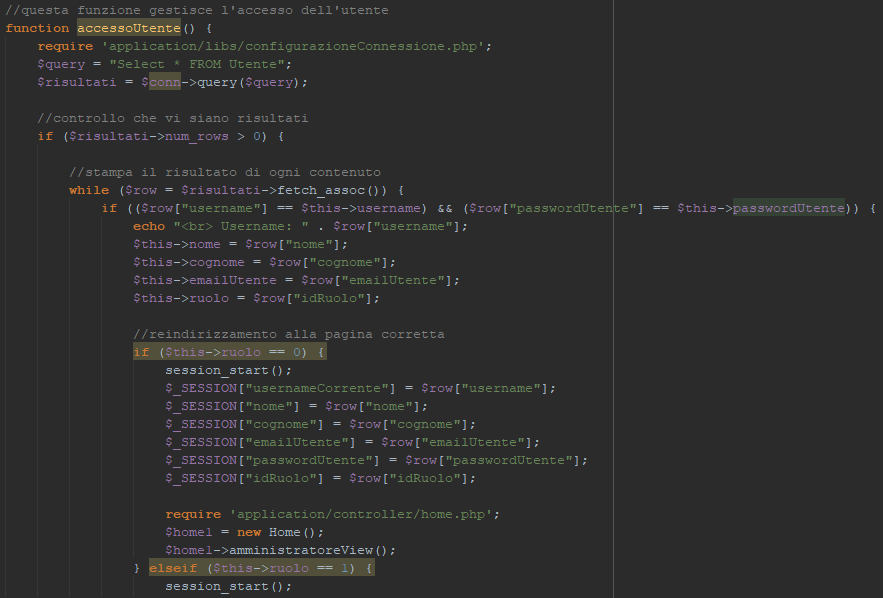
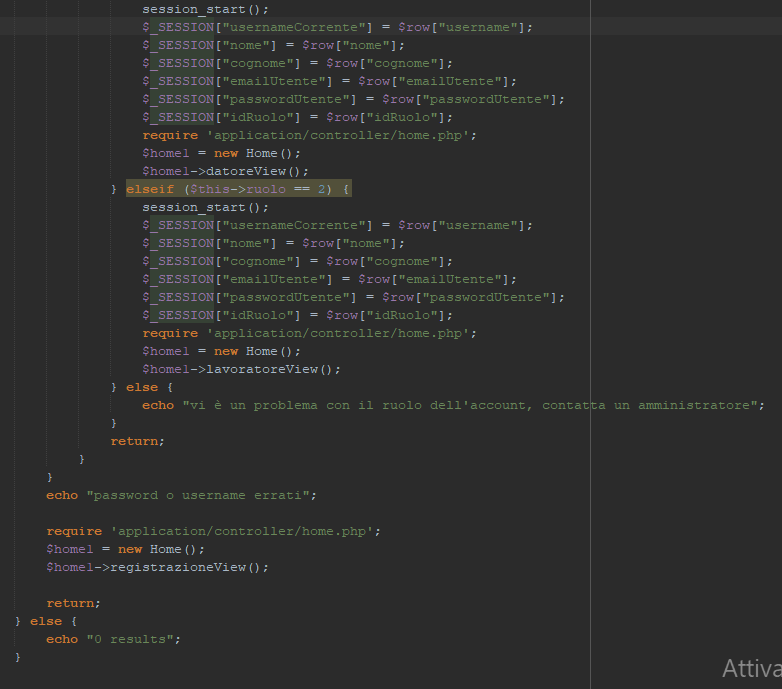


#### Funzione importUserInDB

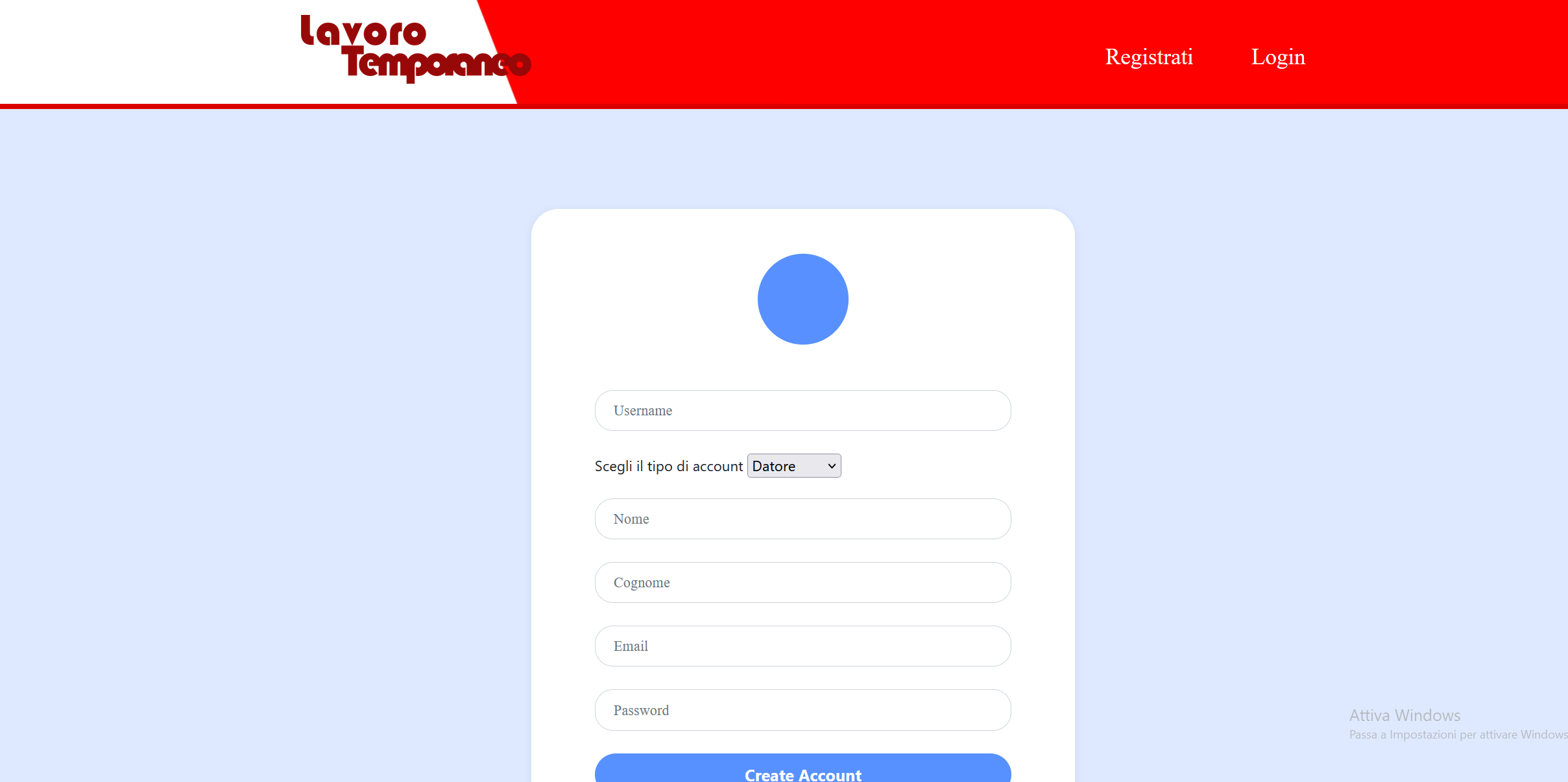
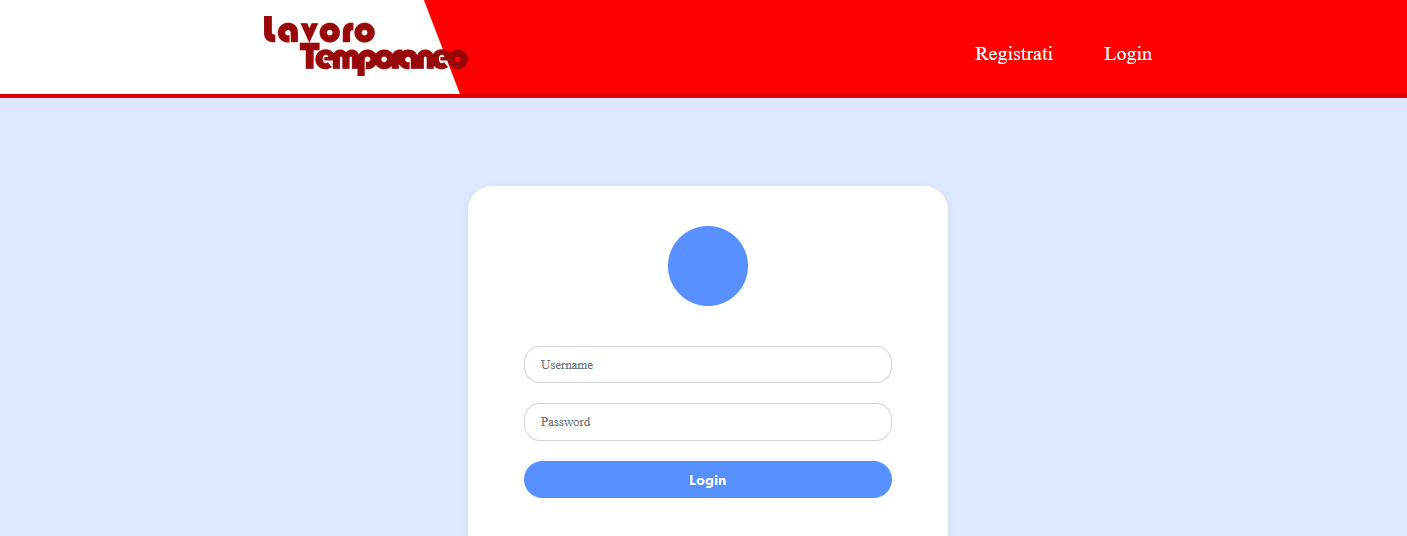
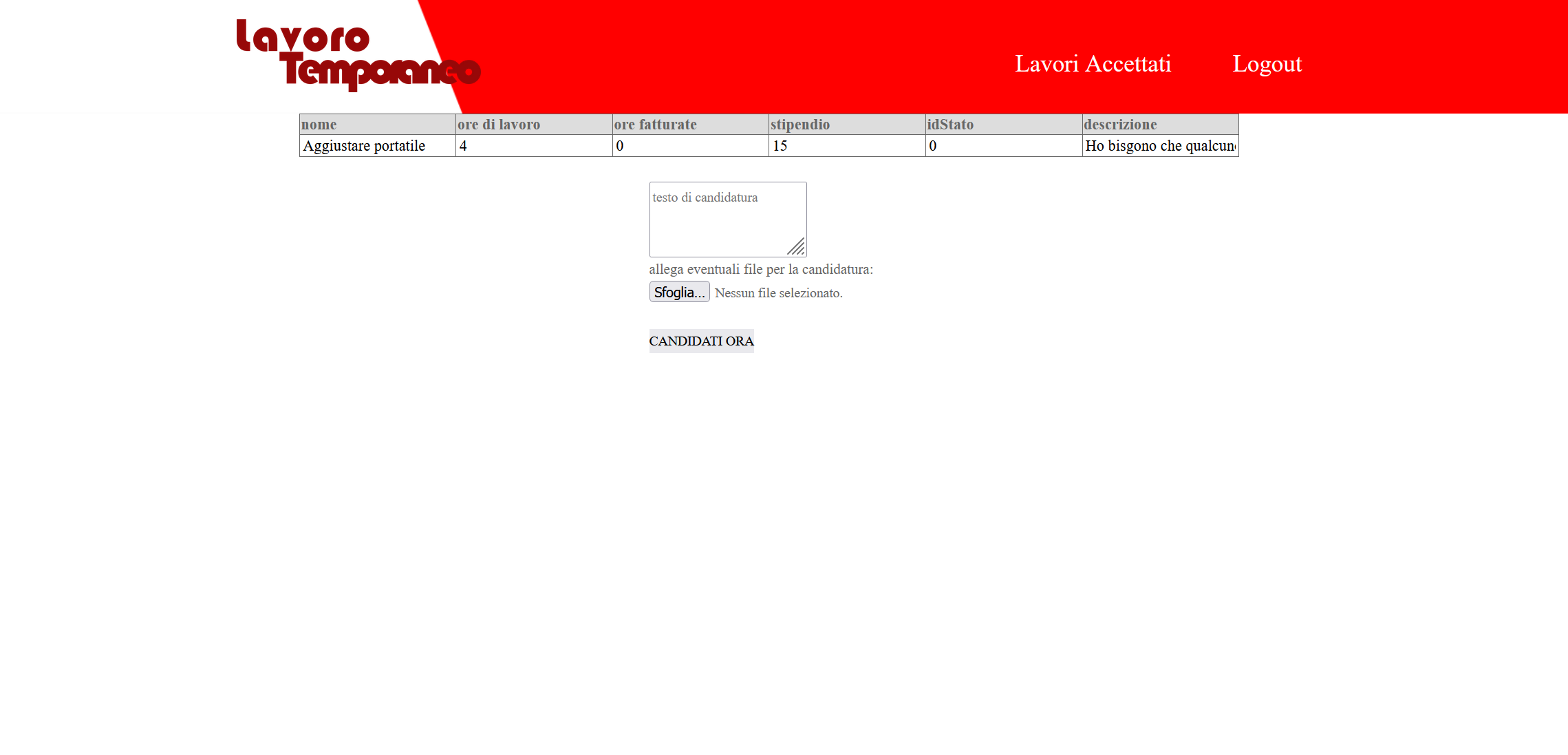
Questa funzione importa l’utente nel database. 

#### Funzione accessoUtente

Questa funzione gestisce l’accesso dell’utente, reindirizzandolo (una volta eseguito l’accesso) nelle corrette e rispettive interfacce. Inoltre questa funzione assegna dei valori alla costante globale $\_SESSION, in modo che tali dati riguardanti l’utente loggato siano sempre reperibili in qualsiasi posizione.

# Implementazione

* **MainView** 
* **Registrazione** 
* **Login** 
* **MainLavoratore** 
* **Lavoro selezionato**
* **Lavori accettati**Immagine che contiene testo

  Descrizione generata automaticamente
* **MainDatoreLavoro** 
* **LavoriOfferti** 
* **PubblicaLavoro** 
* **Main amministratore**

## Protocollo di test

Definire in modo accurato tutti i test che devono essere realizzati per garantire l’adempimento delle richieste formulate nei requisiti. I test fungono da garanzia di qualità del prodotto. Ogni test deve essere ripetibile alle stesse condizioni.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-001  REQ-01 | **Nome:** | Verifica del funzionamento dell’hosting. |
| **Descrizione:** | Sarà necessario verificare il funzionamento dell’hosting sul quale si caricherà il sito web. | | |
| **Prerequisiti:** |  | | |
| **Procedura:** | 1. Aprire con un browser l’url dell’hosting. 2. Attendere qualche secondo. 3. In caso di connessione non riuscita verificare che la porta usata per la connessione sia effettivamente. | | |
| **Risultati attesi:** | La pagina principale caricata sull’hosting | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-002  REQ-02 | **Nome:** | Verifica dei messaggi di errore del form di login. |
| **Descrizione:** | Sarà necessario verificare il funzionamento dei messaggi di errore del form login in caso di credenziali errate. | | |
| **Prerequisiti:** |  | | |
| **Procedura:** | 1. Aprire con un browser l’url dell’hosting. 2. Recarsi nella sezione di login del sito 3. Inserire nome utente e / o password errati. | | |
| **Risultati attesi:** | Messaggio di errore credenziali di login | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-003  REQ-02 | **Nome:** | Verifica accesso form di login |
| **Descrizione:** | Sarà necessario verificare il funzionamento del form di login con credenziali corrette. | | |
| **Prerequisiti:** | - | | |
| **Procedura:** | 1. Aprire con un browser l’url dell’hosting 2. Recarsi nella sezione di login 3. Inserire nome utente e password corrette | | |
| **Risultati attesi:** | Login riuscito | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-004  REQ-02 | **Nome:** | Verifica dei messaggi di errore del form di registrazione |
| **Descrizione:** | Sarà necessario verificare il funzionamento dei messaggi di errore del form di registrazione in caso di credenziali errate. | | |
| **Prerequisiti:** | - | | |
| **Procedura:** | 1. Aprire con un browser l’url dell’hosting. 2. Recarsi nella sezione di login. 3. Cliccare “registrazione”. 4. Registrarsi con un nome utente già esistente. | | |
| **Risultati attesi:** | Messaggio di errore username già in utilizzo | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-005  REQ-02 | **Nome:** | Verifica registrazione effettuata form di registrazione |
| **Descrizione:** | Sarà necessario verificare il funzionamento del form di registrazione. | | |
| **Prerequisiti:** | - | | |
| **Procedura:** | 1. Aprire in un browser l’url del sito web. 2. Recarsi nella sezione di login. 3. Cliccare “registrazione”. 4. Registrarsi con un nome utente non ancora già registrato. | | |
| **Risultati attesi:** | La pagina principale caricata sull’hosting. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-006  REQ-03 | **Nome:** | Verifica funzionamento operazione di pubblicazione offerta di lavoro. |
| **Descrizione:** | Sarà necessario verificare il funzionamento della pubblicazione di un’offerta di lavoro. | | |
| **Prerequisiti:** | - | | |
| **Procedura:** | 1. Aprire in un browser l’url del sito web. 2. Recarsi nella sezione di login. 3. Loggarsi con delle credenziali valide. 4. Pubblicare un’offerta di lavoro | | |
| **Risultati attesi:** | Viene pubblicata un’offerta di lavoro dall’utente utilizzato con i dati inseriti. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-007  REQ-04 | **Nome:** | Verifica funzionamento rimozione offerta di lavoro da parte di datore di lavoro. |
| **Descrizione:** | Sarà necessario verificare il funzionamento di rimozione di una delle offerte di lavoro pubblicate dall’utente stesso | | |
| **Prerequisiti:** | - | | |
| **Procedura:** | 1. Aprire in un browser l’url del sito web. 2. Recarsi nella sezione di login. 3. Loggarsi con delle credenziali valide. 4. Recarsi nella sezione “lavori”. 5. Entrare nella pagina dei lavori offerti. 6. Rimuovere uno dei lavori offerti. | | |
| **Risultati attesi:** | Il form del lavoro rimosso viene cancellato dalla home del sito internet. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-008  REQ-04 | **Nome:** | Verifica funzionamento rimozione offerta di lavoro da parte di amministratore |
| **Descrizione:** | Sarà necessario verificare il funzionamento di rimozione da parte di un amministratore di un’offerta di lavoro. | | |
| **Prerequisiti:** | - | | |
| **Procedura:** | 1. Aprire in un browser l’url del sito web. 2. Recarsi nella sezione di login. 3. Loggarsi con delle credenziali valide da amministratore. 4. Recarsi nella sezione “lavori”. 5. Rimuovere uno dei lavori offerti. | | |
| **Risultati attesi:** | Il form del lavoro rimosso viene cancellato dalla home del sito internet. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-009  REQ-05 | **Nome:** | Verifica funzionamento proporsi a offerta di lavoro |
| **Descrizione:** | Sarà necessario verificare il funzionamento di proporsi a un’offerta di lavoro. | | |
| **Prerequisiti:** | - | | |
| **Procedura:** | 1. Aprire in un browser l’url del sito web. 2. Recarsi nella sezione di login. 3. Loggarsi con delle credenziali valide. 4. Recarsi nella home page. 5. Selezionare un lavoro. 6. Proporsi. 7. Loggare l’account dell’utente che ha pubblicato il lavoro. 8. Verificare di trovare la candidatura. | | |
| **Risultati attesi:** | Nella sezione dei lavori del datore si troverà la candidatura. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-010  REQ-05 | **Nome:** | Verifica funzionamento verifica email |
| **Descrizione:** | Sarà necessario verificare il funzionamento di verifica email una volta eseguita una proposta al lavoro. | | |
| **Prerequisiti:** | - | | |
| **Procedura:** | 1. Aprire in un browser l’url del sito web. 2. Recarsi nella sezione di login. 3. Loggarsi con delle credenziali valide. 4. Recarsi nella home page. 5. Selezionare un lavoro. 6. Proporsi. 7. Verificare nella mailbox della email collegata all’account la conferma della candidatura. | | |
| **Risultati attesi:** | Nella mailbox sarà presente una mail di conferma. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-011  REQ-05 | **Nome:** | Verifica funzionamento caricamento documenti per la candidatura |
| **Descrizione:** | Sarà necessario verificare il funzionamento di verifica email una volta eseguita una proposta al lavoro. | | |
| **Prerequisiti:** | - | | |
| **Procedura:** | 1. Aprire in un browser l’url del sito web. 2. Recarsi nella sezione di login. 3. Loggarsi con delle credenziali valide. 4. Recarsi nella home page. 5. Selezionare un lavoro. 6. Caricare un documento. 7. Proporsi. | | |
| **Risultati attesi:** | Nella sezione dei lavori del datore si troverà la candidatura con il documento in allegato. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-012  REQ-06 | **Nome:** | Verifica funzionamento fatturazione |
| **Descrizione:** | Sarà necessario verificare il funzionamento della fatturazione. | | |
| **Prerequisiti:** | - | | |
| **Procedura:** | 1. Aprire in un browser l’url del sito web. 2. Recarsi nella sezione di login. 3. Loggarsi con delle credenziali amministratore. 4. Inviare la fatturazione al datore di lavoro | | |
| **Risultati attesi:** | Il datore di lavoro riceverà la fattura | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-013  REQ-06 | **Nome:** | Verifica funzionamento stampa fatturazioni |
| **Descrizione:** | Sarà necessario verificare il funzionamento della stampa delle fatturazioni. | | |
| **Prerequisiti:** | - | | |
| **Procedura:** | 1. Aprire in un browser l’url del sito web. 2. Recarsi nella sezione di login. 3. Loggarsi con delle credenziali datore di lavoro. 4. Recarsi nella sezione fatture. 5. Eseguire la stampa delle fatture. | | |
| **Risultati attesi:** | Stampa fatture. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-014  REQ-07 | **Nome:** | Verifica funzionamento utente non registrato |
| **Descrizione:** | Sarà necessario verificare il funzionamento del sito senza eseguire il login. | | |
| **Prerequisiti:** | - | | |
| **Procedura:** | 1. Aprire in un browser l’url del sito web. 2. Recarsi nella homepage 3. Selezionare un lavoro | | |
| **Risultati attesi:** | Verrà caricata la pagina di offerta del lavoro ma non ci potrà proporre | | |

## Risultati test

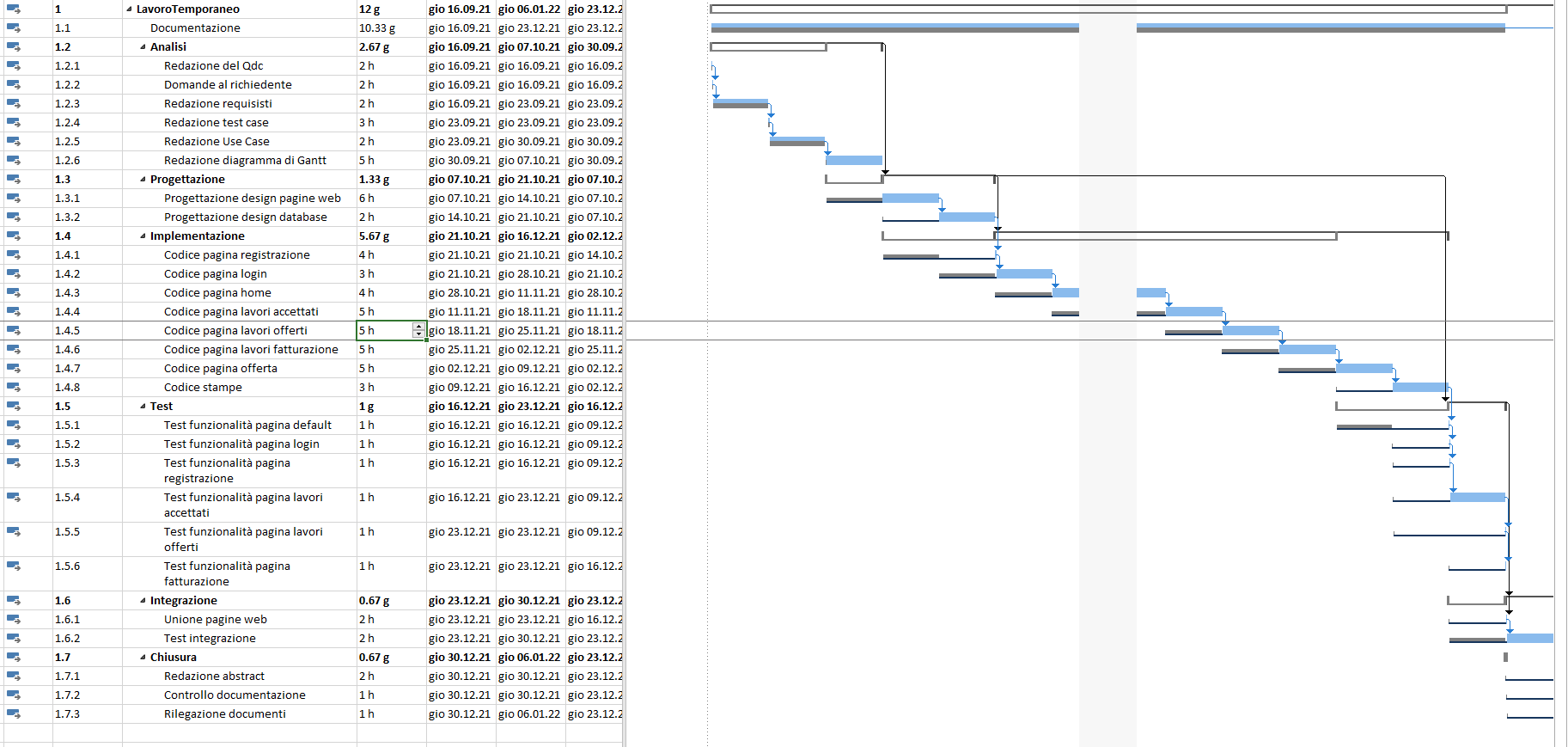
|  |  |  |  |
| --- | --- | --- | --- |
| Test case | Esito | Risultati | Data |
| TC-001 | PASSATO | L’hosting è funzionante. | 21.10.2021 |
| TC-002 | PASSATO | In caso di loggin errato viene notifiato all’utente. | 11.11.2021 |
| TC-003 | PASSATO | Il login funziona. | 11.11.2021 |
| TC-004 | PASSATO | Non è possibile registrarsi con uno username già esistente. | 11.11.2021 |
| TC-005 | PASSATO | La registrazione con nuovo username è funzionante. | 11.11.2021 |
| TC-006 | PASSATO | La pubblicazione dei lavori è operativa. | 16.12.2021 |
| TC-007 | PASSATO |  | 16.12.2021 |
| TC-008 | PASSATO |  | 23.12.2021 |
| TC-009 | PASSATO |  | 23.12.2021 |
| TC-011 | FALLITO | È possibile caricare un documento, tuttavia non viene salvato sul db. | 23.12.2021 |
| TC-012 | FALLITO | La fatturazione non è stata implementata. | 23.12.2021 |
| TC-013 | FALLITO | La stampa delle fatturazioni non è stata implementata. | 23.12.2021 |
| TC-014 | PASSATO | L’utente non registrato visualizza le offerte di lavoro ma non può proporsi | 23.12.2021 |

## Mancanze/limitazioni conosciute

Il progetto è incompleto, mancano diversi requisiti. Ho sottovalutato il tempo necessario al database, ho inoltre sottovalutato la mancanza di formazione base necessaria per questo progetto: 4 mesi fa avevo appena iniziato PHP e non ero affatto abile.

Ho investito troppe energie e tempo nella parte grafica del progetto, ritrovandomi ulteriormente più arretrato con i piani inizialmente stabiliti.

# Consuntivo

Come è possibile constatare quando è arrivato il momento della consegna del progetto mi trovavo con un ritardo pari a più di due settimane. Tale ritardo è stato causato principalmente dalla mancanza di margine per gli imprevisti nella progettazione, infatti per un contrattempo non ho avuto modo di lavorare al progetto per praticamente due settimane.

# Conclusioni

## Sviluppi futuri

Il progetto ha una base abbastanza solida, implementare una gestione automatica della fatturazione e caricarlo su un hosting online potrà indubbiamente farlo risplendere.

## Considerazioni personali

Da questo progetto ho imparato a non sottovalutare il tempo necessario alla pianificazione di qualcosa che non si è mai fatto. Bisogna sempre calcolare un margine di imprevisto, poiché ho vissuto sulla mia pelle che un imprevisto come il COVID può pesare notevolmente sullo sviluppo del progetto, se non si sono prese delle precauzioni.

Il progetto mi ha insegnato a lavorare in modo più ordinato e a mantenere uno storico di ciò che faccio, in modo da non perdere inutilmente del tempo.

# Bibliografia

## Sitografia

* <https://stackoverflow.com/>, 09.12.2021
* <https://www.php.net/> 14.10.2021
* <https://www.w3schools.com/> 25.11.2021

# Allegati

Elenco degli allegati:

* Diari di lavoro
* Design grafica
* Design database
* Codici sorgente/documentazione macchine virtuali
* Mandato e/o QdC
* Prodotto