Creazione e gestione sondaggi via web per progetto IUFFP con DSA e ipovedenti

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 8

2.4 Pianificazione 9

2.5 Analisi dei mezzi 9

2.6 Software 9

2.6.1 Hardware 9

3 Progettazione 10

3.1 Design dei dati e database 10

3.1.1 Utente 10

3.1.2 Sondaggio 10

3.1.3 Domanda 10

3.1.4 Report 11

3.1.5 Commento 11

3.1.6 Risposta 11

3.1.7 Richieste 11

3.2 Design delle interfacce 12

3.2.1 Pagina accesso sondaggio 12

3.2.2 Pagina accesso gestione 12

3.2.3 Pagina gestione utenti 13

3.2.4 Pagina sondaggio 14

4 Implementazione 15

4.1 Login e gestione utenti 15

4.1.1 Login 15

4.1.2 Creazione e visualizzazione utenti 18

4.1.3 Mail 19

4.1.4 Controller e model 20

4.2 Dashboard 21

4.2.1 Classi e metodi **Errore. Il segnalibro non è definito.**

4.3 Sondaggi 22

4.3.1 Visualizzazione dei sondaggi 22

4.3.2 Creazione dei sondaggi 22

4.3.3 Salvataggio dei sondaggi 25

4.3.4 Accesso al sondaggio 27

4.3.5 Commenti 29

4.4 Report 30

4.4.1 Controller e model 32

4.5 Ricerca 33

5 Test 35

5.1 Protocollo di test 35

5.2 Risultati test 39

5.3 Mancanze/limitazioni conosciute 39

6 Consuntivo 40

7 Conclusioni 41

7.1 Sviluppi futuri 41

7.2 Considerazioni personali 41

8 Bibliografia 41

8.1 Sitografia 41

9 Allegati 41

# Introduzione

## Informazioni sul progetto

Allievo coinvolto: Andrea Rauso

Classe: I4AC Scuola Arti e Mestieri Trevano

Docente responsabile: Fabrizio Valsangiacomo

Data inizio: 03/09/2019

Data Fine: 20/12/2019

## Abstract

*The Swiss Federal Institute for Vocational Education and Training (SFIVET) needs a renewed tool for creating surveys for students with certain learning problems such as dyscalculia, dyslexia, etc... This tool solves problems that were found in the previous web application.*

## Scopo

Lo scopo di questo progetto è di creare un gestionale per la creazione e gestione di sondaggi, verranno usate le varie nozioni apprese con i progetti dell’anno scolastico precedente in modo da prepararci al meglio per quello che sarà il progetto di esame, verranno impiegate anche le nozioni apprese nei moduli nel corso dell’anno quali, in questo caso, la programmazione lato server e la progettazione e la realizzazione di database.

# Analisi

## Analisi del dominio

Attualmente l’Istituto Universitario Federale per la Formazione Professionale (IUFFP) possiede già un gestionale per la gestione dei sondaggi ma il quale presenta degli errori di non poco conto, lo scopo di questo progetto è di migliorare il gestionale risolvendo le lacune presenti nell’attuale logica del sito.

Questo gestionale viene usato internamente alla scuola per fare sondaggi quali il gradimento su un determinato tema, ecc…

## Analisi e specifica dei requisiti

|  |  |
| --- | --- |
| **ID: REQ-001** | |
| **Nome** | Sito |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** |  |
| **Sotto requisiti** | |
| **001** | Il sito dovrà avere un accesso ai sondaggi tramite password |
| **002** | Il sito dovrà provvedere un login per la gestione dei sondaggi |

|  |  |
| --- | --- |
| **ID: REQ-002** | |
| **Nome** | Utenti |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** |  |
| **Sotto requisiti** | |
| **001** | Il sito prevede 2 livelli di accesso differenti |
| **002** | Gli utenti possono essere amministratori o docenti |
| **003** | L’amministratore ha la facoltà di creare, modificare e eliminare gli utenti, creare e eliminare sondaggi, aggiungere domande ai sondaggi, generazione, archiviazione e eliminazione dei report |
| **004** | Il docente ha la facoltà di creare e eliminare sondaggi, la generazione, archiviazione e eliminazione dei report. |
| **005** | Il docente può richiedere la creazione di un utente, solo l’amministrazione accetta o rifiuta le richieste |
| **006** | L’utente creato deve cambiare la password al primo accesso, se l’utente perde la password può richiederne una nuova |

|  |  |
| --- | --- |
| **ID: REQ-003** | |
| **Nome** | Gestione Utenti |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** |  |
| **Sotto requisiti** | |
| **001** | L’amministratore è l’unico a gestire gli utenti |
| **002** | L’amministratore crea e promuove gli utenti |

|  |  |
| --- | --- |
| **ID: REQ-004** | |
| **Nome** | Sondaggi |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** |  |
| **Sotto requisiti** | |
| **001** | I sondaggi vengono creati dai docenti e dagli amministratori |
| **002** | I sondaggi conterranno domande di più tipi |
| **003** | I sondaggi possono essere attivati e disattivati, alla riattivazione viene ricreata la password |

|  |  |
| --- | --- |
| **ID: REQ-005** | |
| **Nome** | Domande |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** |  |
| **Sotto requisiti** | |
| **001** | Le domande possono essere a risposta singola (Si/no, valutazione, ecc..) o multipla |
| **002** | Le domande possono avere dei commenti |

|  |  |
| --- | --- |
| **ID: REQ-006** | |
| **Nome** | Report |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** |  |
| **Sotto requisiti** | |
| **001** | Bisogna prevedere la possibilità di creare dei report statistici per i sondaggi |
| **002** | I report includeranno il numero di risposte date per ogni domanda posta |
| **003** | I report conterranno i commenti delle domande se abilitati |
| **004** | I report possono essere archiviati e cancellati |

|  |  |
| --- | --- |
| **ID: REQ-007** | |
| **Nome** | Ricerca |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** |  |
| **Sotto requisiti** | |
| **001** | All’interno delle tabelle deve essere possibile effettuare delle ricerche |

|  |  |
| --- | --- |
| **ID: REQ-008** | |
| **Nome** | Design del sito |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** |  |
| **Sotto requisiti** | |
| **001** | Il sito deve essere responsive su diversi dispositivi di visualizzazione |
| **002** | I dispositivi non supportati vengono notificati |
| **003** | In caso di problemi viene fatto visualizzare un testo alterativo |

|  |  |
| --- | --- |
| **ID: REQ-009** | |
| **Nome** | Design della GUI |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** |  |
| **Sotto requisiti** | |
| **001** | Il design del sito deve essere consistente tra tutte le pagine |
| **002** | Il sito deve rappresentare il corretto flusso di lavoro |

|  |  |
| --- | --- |
| **ID: REQ-010** | |
| **Nome** | Sicurezza accesso |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** |  |
| **Sotto requisiti** | |
| **001** | Devono essere implementate dei controlli all’accesso |
| **002** | Devono essere assegnate autorizzazioni chiare |

|  |  |
| --- | --- |
| **ID: REQ-011** | |
| **Nome** | Sicurezza |
| **Priorità** | 1 |
| **Versione** | 1.0 |
| **Note** |  |
| **Sotto requisiti** | |
| **001** | Il sito deve resistere ad attacchi ai dati |

## Use case

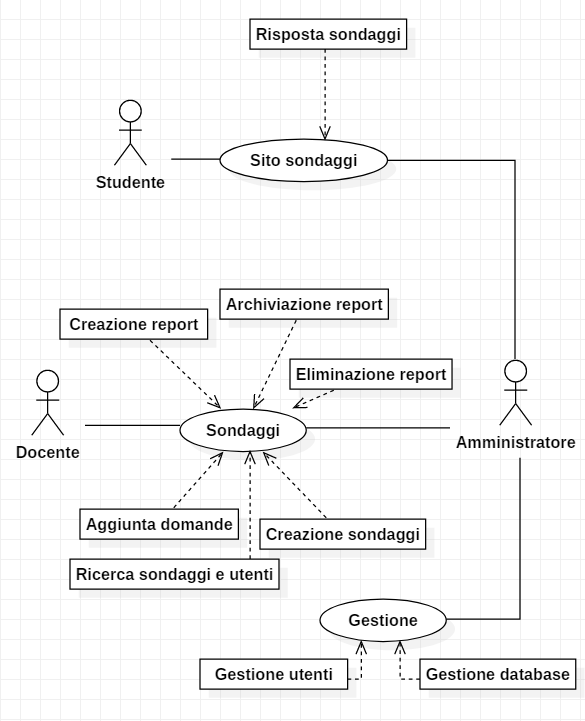


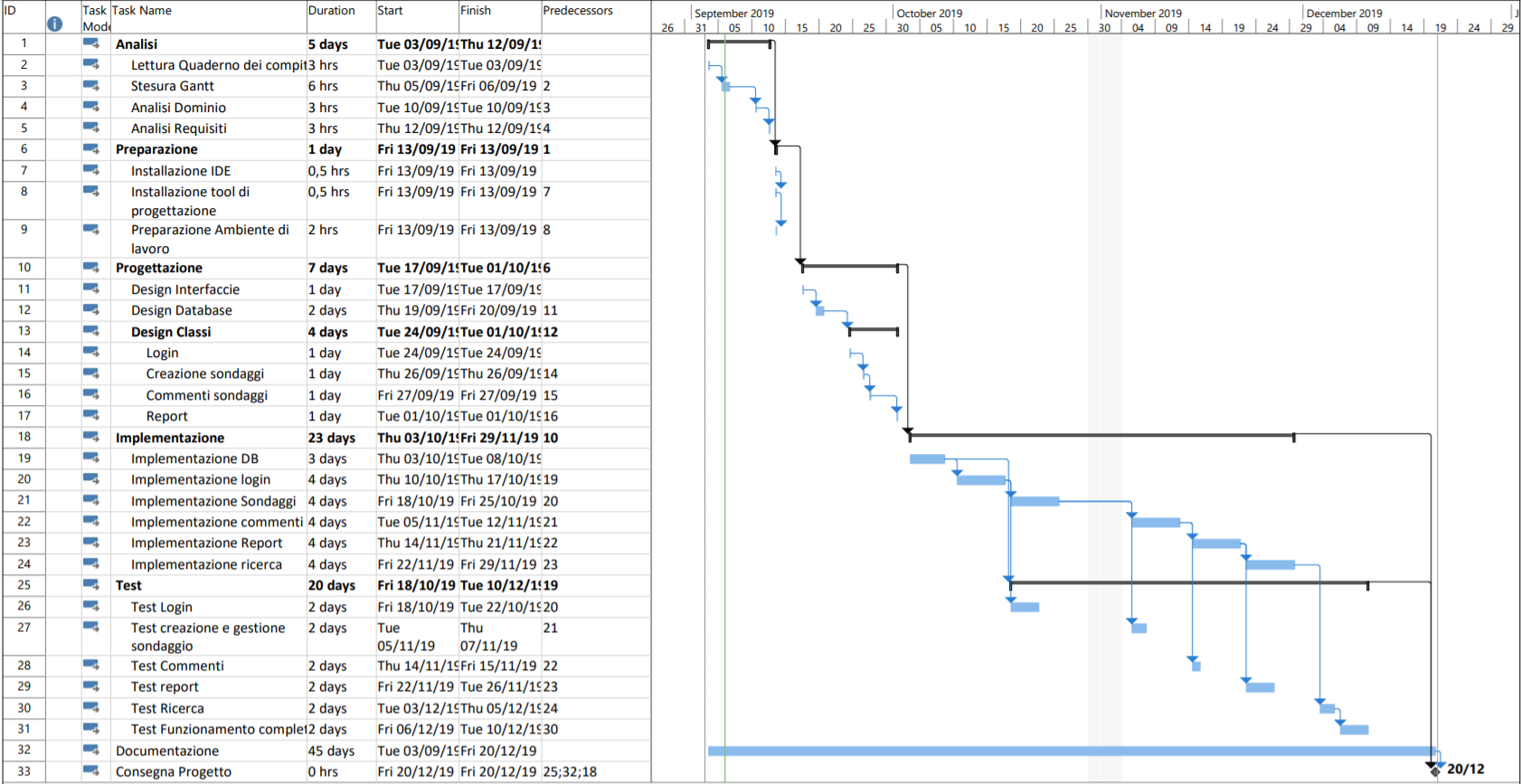
Figura - Diagramma Use case

Gli use case per questo progetto sono i seguenti: lo studente può solamente visionare il sito dei sondaggi e può solo rispondere ai sondaggi.

Il docente può accedere alla gestione dei sondaggi e può svolgere le funzioni di creazione di sondaggi, aggiunta di domande e la ricerca dei sondaggi.

L’amministratore svolge le stesse funzioni del docente con l’aggiunta della gestione degli utenti, quindi della loro creazione, modifica e eliminazione degli stessi e della gestione delle richieste della creazione di utenti.

## Pianificazione



Come pianificazione del progetto ho deciso di svolgere la parte di analisi della durata di 5 giorni di lavoro, in cui vengono inclusi la lettura del quaderno dei compiti, la stesura del Gantt e l’analisi di dominio e dei requisiti.

La seconda fase è quella di preparazione dove vado a preparare la macchina per la progettazione e l’implementazione, essendo le attività veloci questa fase dura solo una giornata.

Nella fase di progettazione ho messo le attività di design delle interfacce dove vado a creare i mockups delle varie pagine, il design del database in cui andrò a salvare tutte le informazioni del sito e la progettazione delle classi in cui progetto, in modo indicativo, la struttura delle classi del progetto.

La fase di implementazione è la fase più lunga in cui andrò a implementare le varie parti del progetto come il database, il login degli utenti, la creazione e la gestione dei sondaggi, ecc... La fase di test viene effettuata dopo che ogni attività viene completata.

## Analisi dei mezzi

## Software

* Microsoft Office 2016
* Microsoft Project 2016
* JetBrains PhpStorm 2019.1.2
* Xampp (Php 7.3.9)
* Google Chrome 77
* Template del sito: startbootstrap sb admin 2

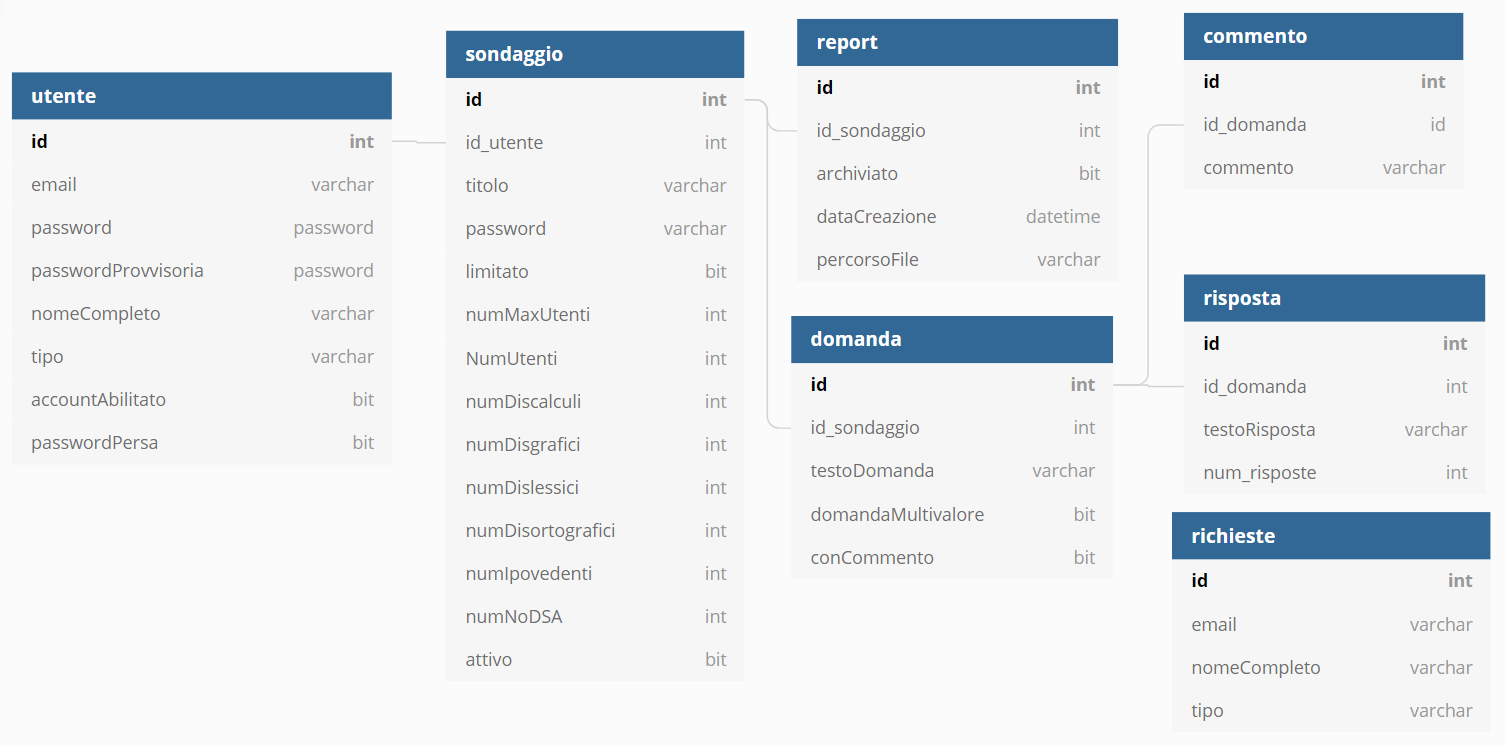
### Hardware

Portatile ASUS VivoBook Pro 15 N580GD-E4287T

* Processore: Intel i7-8750H
* RAM: 16 GB

# Progettazione

## Design dei dati e database



### Utente

La tabella utente contiene tutte le informazioni di base e di accesso degli utenti che hanno accesso al gestionale.

All’interno della tabella users vengono definiti l’id dell’utente, la sua email e la sua password per accedere al gestionale, la password provvisoria quando viene creato l’utente, il tipo di utente inteso come livello di accesso quale docente o amministratore, un flag per indicare l’abilitazione dell’account dopo che l’utente ha impostato per la prima volta la password e un flag per indicare la perdita della password e la spedizione di un link per cambiarla.

### Sondaggio

La tabella sondaggio contiene tutte le informazioni relative a un sondaggio. Viene definito un id per l’identificazione del sondaggio, l’id dell’utente che l’ha creato, il titolo del sondaggio, la password di tipo random per l’accesso al sondaggio da parte degli studenti. Viene indicato un flag se il sondaggio deve essere risposto da un ristretto numero di persone, se viene abilitato il flag si indicherà il numero massimo di persone e verrà aggiornato il valore delle persone che hanno risposto al sondaggio.

Questa tabella ha una relazione sull’attributo id\_utente con la tabella Utente con l’attributo id, se un utente viene eliminato vengono eliminati a cascata tutti i sondaggi

### Domanda

La tabella domanda contiene tutte le informazioni relative a una domanda del sondaggio.

All’interno della tabella domanda viene definito un id per l’identificazione della domanda, l’id del sondaggio di cui fa parte, un attributo contenente il testo della domanda.

La tabella ha 2 flag booleani, il primo, domandaMultivalore, che indicherà se alla domanda si possono selezionare più risposte, in caso contrario una singola risposta, il secondo flag, conCommento, indicherà invece se la domanda può avere una casella di testo per poter inserire dei commenti riguardanti la stessa.

Questa tabella ha una relazione sull’attributo id\_sondaggio con la tabella Sondaggio sull’attributo id, se un sondaggio viene eliminato vengono eliminate a cascata tutte le domande relative a quel sondaggio

### Report

La tabella report contiene le informazioni relative ai report creati dai docenti/amministratori dei sondaggi.

All’interno della tabella report viene definito un id per la sua identificazione, un id relativo al sondaggio a cui è collegato, un flag per indicare se il report è stato archiviato, un attributo indicante la data di creazione e un attributo contenente la posizione del file all’interno del server per facilitare il recupero dello stesso.

Questa tabella ha una relazione sull’attributo id\_sondaggio con la tabella Sondaggio sull’attributo id, se un sondaggio viene eliminato vengono eliminati tutti i report relativi a quel sondaggio.

I report pdf non vengono cancellati dal disco se vengono eliminati dal database.

### Commento

La tabella commento contiene tutte le informazioni relative ai commenti delle domande dati dagli utenti. All’interno di questa tabella viene definito un id per la sua identificazione, un id relativo alla domanda a cui è collegato e l’attributo contenente il commento dato dall’utente.

Questa tabella ha una relazione sull’attributo id\_domanda con la tabella Domanda sull’attributo id, se una domanda viene eliminata vengono cancellati tutti i commenti relativi a quella domanda.

### Risposta

La tabella risposta contiene le risposte delle domande, viene definito un id per l’identificazione, un id relativo alla domanda collegata alla risposta, un attributo contenente il testo della risposta e un attributo contenente il numero di risposte per quella risposta.

Questa tabella ha una relazione sull’attributo id\_domanda con la tabella Domanda sull’attributo id, se una domanda viene eliminata vengono cancellati tutte le risposte relative a quella domanda.

### Richieste

La tabella richieste conterrà le richieste di creazione degli utenti da parte dei docenti e dagli amministratori, viene definito un id per l’identificazione, il nome completo della persona, la mail della persona e il tipo di utente che si vuole creare (Amministratore, Docente)

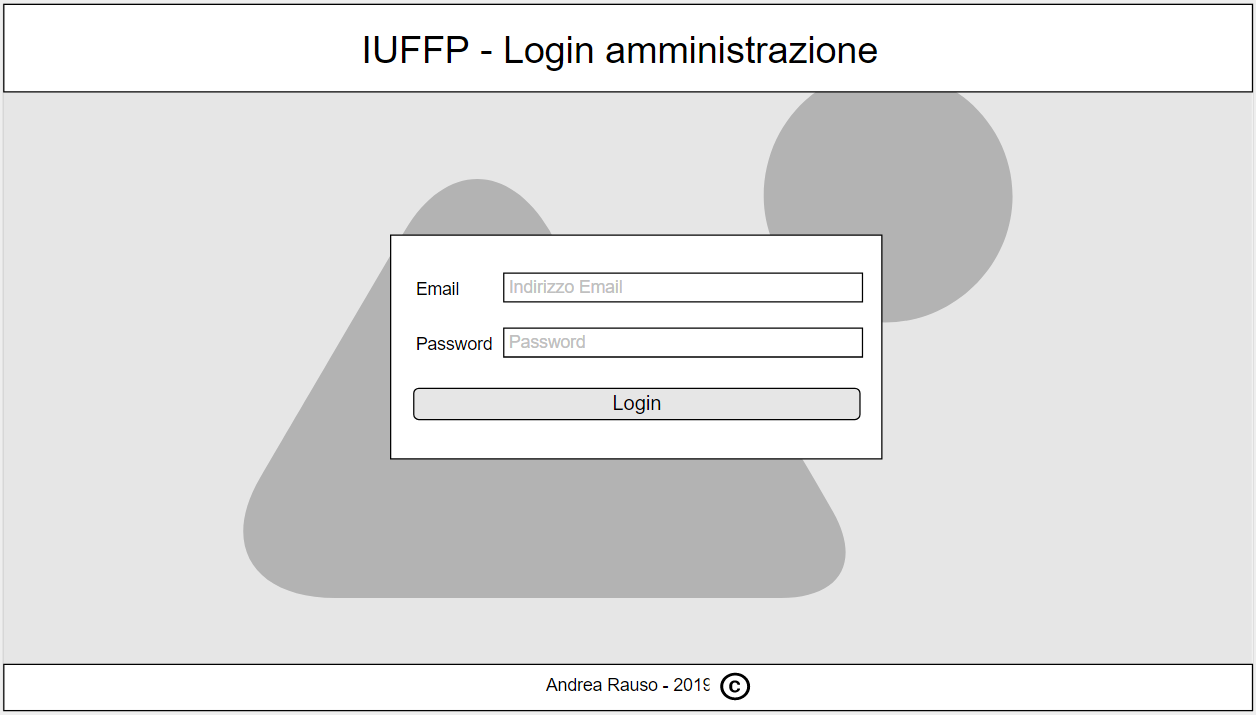
Questa tabella non ha relazioni poiché non hanno riferimenti con altri dati.

## Design delle interfacce

### Pagina accesso sondaggio

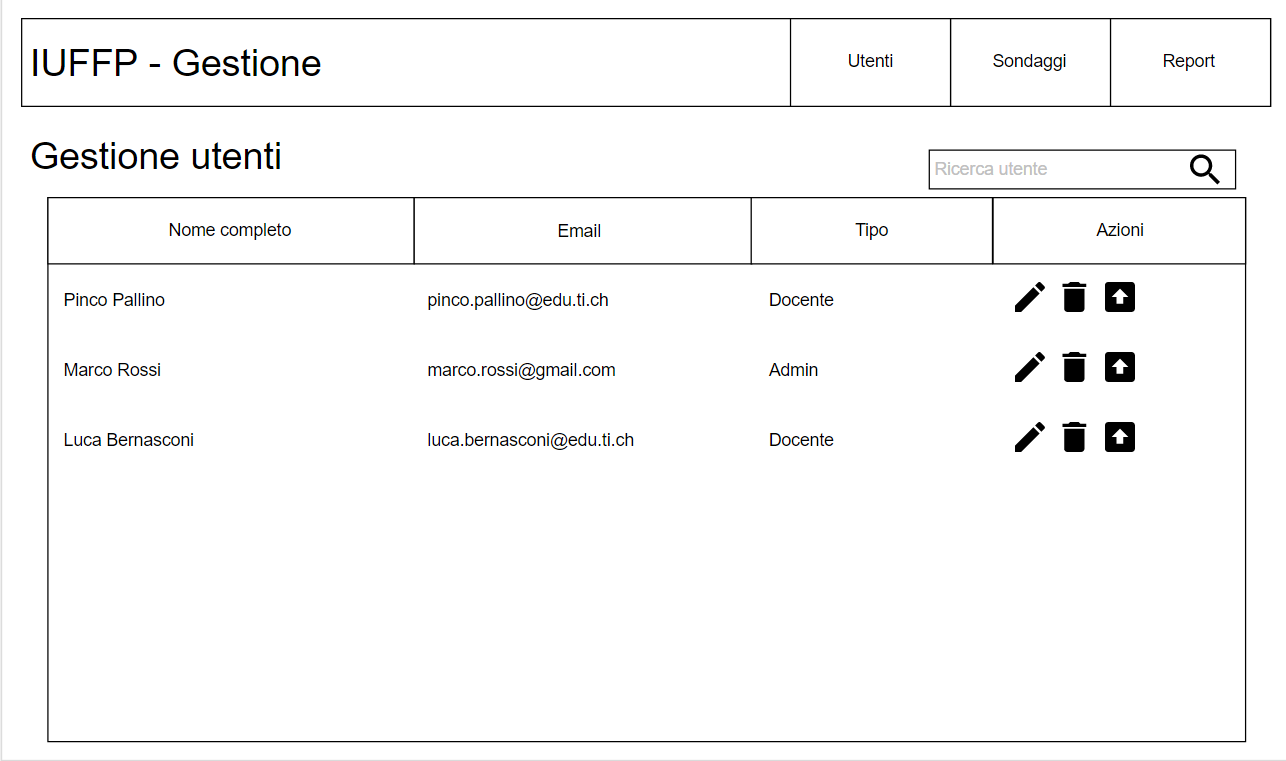
Per l’interfaccia della pagina principale ho pensato a una semplice pagine con un box centrale in cui gli utenti/studenti immettono la password per accedere al sondaggio, i docenti e gli amministratori per accedere alla parte di gestione del sito cliccheranno sul bottone di login dove verranno rimandati alla pagina di login.  
La pagina avrà come sfondo un immagine relativa alla scuola come una foto della sede ticinese dell’IUFFP.

### Pagina accesso gestione



La pagina di accesso per la gestione riprende le stesse caratteristiche della pagina principale, in questa pagina gli amministratori/docenti/utenti potranno inserire le proprie credenziali d’accesso (email e password) che dopo aver cliccato sul pulsante di login accederanno alle pagine di gestione dei sondaggi

### Pagina gestione utenti



La pagina di gestione degli utenti (accessibile soltanto agli amministratori del gestionale) permette di vedere in forma tabellare gli utenti che possono accedere. Nella tabella si possono vedere il nome completo dell’utente, l’indirizzo email con cui è stato registrato dall’amministratore, il tipo dell’utente e le azioni che si possono eseguire sull’utente.

Le azioni che si possono eseguire sono: la modifica delle informazioni dell’utente (all’interno della pagina di modifica è incluso un tasto per il ripristino della password dell’utente), la cancellazione dell’utente e la promozione dell’utente (da docente a amministratore, non è possibile il passaggio inverso).

### Pagina sondaggio



La pagina del sondaggio è la pagina che vedranno gli utenti (studenti) che andranno a svolgere il sondaggio, all’interno figureranno le domande create dai docenti amministratori. Le domande sono di tre tipi: domande a risposta multipla, domande si/no e domande di valutazione. Per ogni domanda è possibile aggiungere facoltativamente un campo per i commenti degli utenti che andranno poi a figurare nei report. All’inizio della pagina sono indicate dei possibili disturbi specifici dell’apprendimento (DSA) quali dislessia, discalculia, ecc.… che gli utenti dovranno cliccare se hanno uno o più disturbi.

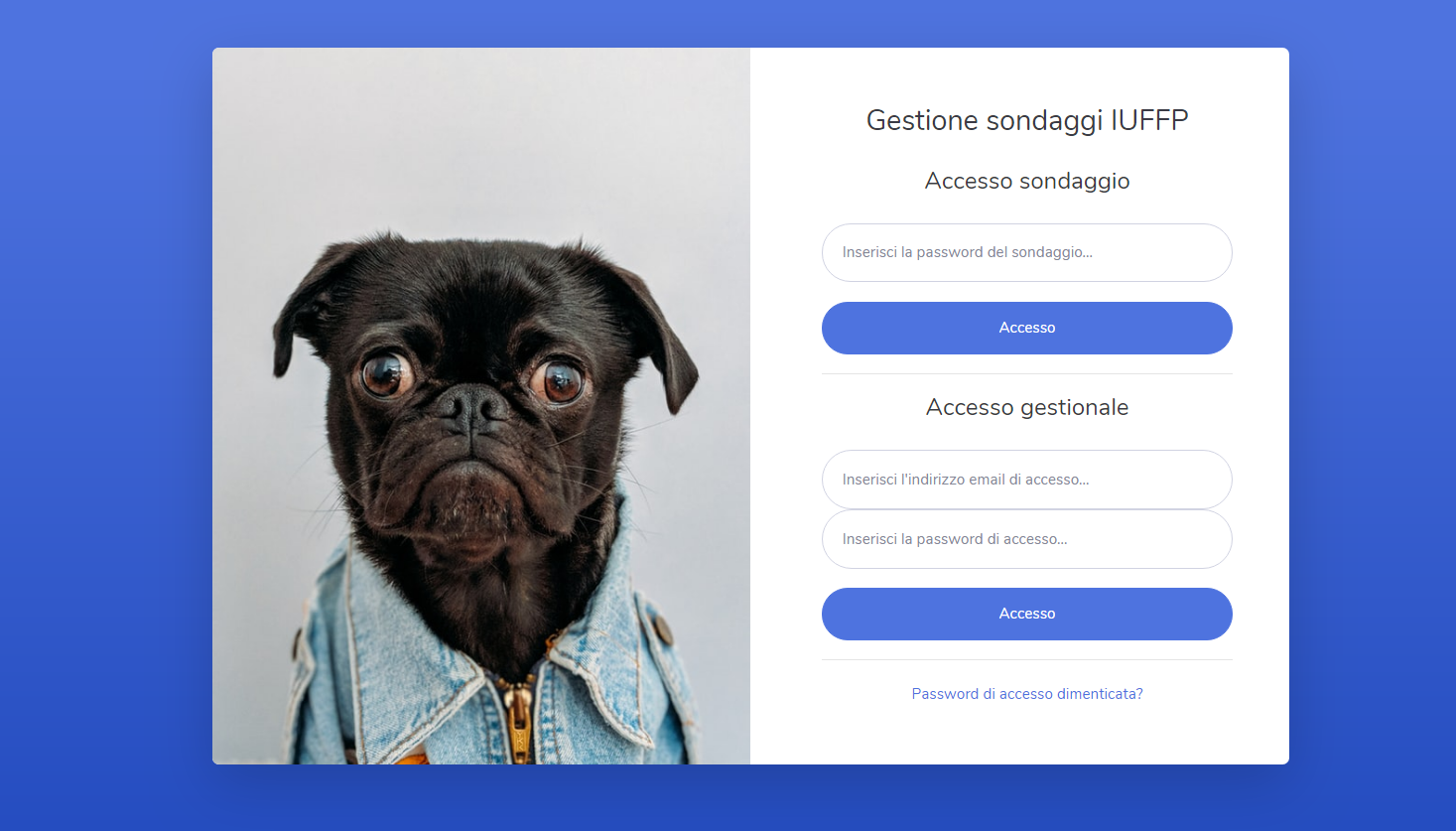
# Implementazione

## Login e gestione utenti

La prima parte dell’implementazione del mio progetto riguarda il sistema di login per l’accesso al gestionale

### Login

La pagina di login del gestionale è dove i docenti e gli amministratori possono accedere al gestionale, questa pagina è accessibile dalla pagina principale del sito sotto al form di accesso dei sondaggi degli utenti/studenti.



Per la ricerca dell’utente all’interno del database uso la libreria PDO (PHP Data Object) per accedere al database MySQL, di questa libreria sfrutto le prepared statement per rimuovere i problemi di SQL Injection che si possono verificare nel caso si costruisce la query da stringa.

In caso l’utente sbagli l’email e/o la password il controller che gestisce l’accesso ritornerà una stringa di errore dicendo che uno o più valori sono sbagliati.

Da questa pagina è possibile anche accedere i sondaggi inserendo la password corrispondente.

Metodo per il controllo del login per controllare i dati di accesso o nel caso di un utente appena creato di mandarlo al form di cambio della password:

**public function** login()  
{  
 *// session error* $\_SESSION[**"ProvvisoryPasswordError"**] = **""**;  
 $\_SESSION[**"LoginError"**] = **""**;  
 **if**($\_SERVER[**"REQUEST\_METHOD"**] === **"POST"**){  
 *//check if the user email and the password field in the login form is filled* **if**(**isset**($\_POST[**"userEmail"**]) && !**empty**($\_POST[**"userEmail"**]) && **isset**($\_POST[**"userPass"**]) && !**empty**($\_POST[**"userPass"**]))  
 {  
 $\_SESSION[**"userEmail"**] = $\_POST[**"userEmail"**];  
 $\_SESSION[**"userPass"**] = $\_POST[**"userPass"**];  
 *// get the user status (active or not)* $userStatus = Users::*getUserStatus*($\_SESSION[**"userEmail"**]);  
   
 *//if is active it will try to login* **if**($userStatus == **"1"**){  
 *//get the user information and check if exist* $result = Users::*getUser*($\_SESSION[**"userEmail"**],$\_SESSION[**"userPass"**]);  
 **if**($result){  
 *// authentcation and redirect to the dashboard* Auth::*authenticate*();  
 $\_SESSION[**"userType"**] = Users::*getUserType*($\_SESSION[**"userEmail"**]);  
 $\_SESSION[**"userId"**] = Users::*getUserId*($\_SESSION[**"userEmail"**]);  
 Application::*redirect*(**'dashboard/Dashboard'**);  
 }**else**{  
 *// return to the login page* $\_SESSION[**"LoginError"**] = **"L'username e/o la password non sono corretti"**;  
 Application::*redirect*(**"home/login"**);  
 }  
 }**else**{  
 *//check the user that need to chenge his password at the firs login* $provPassCheck = Users::*checkUserProvvisoryPassword*($\_SESSION[**"userEmail"**], $\_SESSION[**"userPass"**]);  
 *//if the information is correct* **if**($provPassCheck){  
 $\_SESSION[**"userType"**] = Users::*getUserType*($\_SESSION[**"userEmail"**]);  
 $\_SESSION[**"userId"**] = Users::*getUserId*($\_SESSION[**"userEmail"**]);  
 Application::*redirect*(**'login/passwordChangeForm'**);  
 }**else**{  
 $\_SESSION[**"LoginError"**] = **"L'username e/o la password non sono corretti"**;  
 Application::*redirect*(**"home/login"**);  
 }  
 }  
 }  
 }  
}

Come si può vedere dal codice controllo se i campi di inserimento della mail e della password sono riempiti,

controllo poi lo stato dell’utente per vedere se è attivo (ha già cambiato la password) o meno (è appena stato creato). Nel caso l’utente sia attivo controllo se l’utente esiste all’interno del database e controllo se le informazioni sono corrette e poi lo faccio accedere, nel caso di un utente appena creato si controlla se la password provvisoria mandata via mail sia corretta e pio lo manda al form di cambio della password.

Il codice seguente controlla la nuova password che vuole impostare l’utente guardando se le password contenute nel form combaciano, se sono corrette viene rimandato alla pagina di login per eseguire l’accesso con la nuova password, in caso contrario viene rimandato al form di cambio password:

**public function** saveProvvisoryPassword(){  
 $\_SESSION[**"ProvvisoryPasswordError"**] = **""**;  
 **if**($\_SERVER[**"REQUEST\_METHOD"**] == **"POST"**){  
 *//check if the password form is filled* **if**(**isset**($\_POST[**"provvisoryPassword"**]) && !**empty**($\_POST[**"provvisoryPassword"**])  
 && **isset**($\_POST[**"confirmProvvisoryPassword"**]) && !**empty**($\_POST[**"confirmProvvisoryPassword"**])) {  
 $provvisoryPassword = *htmlspecialchars*($\_POST[**"provvisoryPassword"**]);  
 $confirmProvvisoryPassword = *htmlspecialchars*($\_POST[**"confirmProvvisoryPassword"**]);  
  
 *//check if they are the same password* **if**(*strcmp*($provvisoryPassword,$confirmProvvisoryPassword) == 0){  
 Users::*savePassword*($provvisoryPassword);  
 Auth::*logout*();  
 }**else**{  
 $\_SESSION[**"ProvvisoryPasswordError"**] = **"Le 2 password non combaciano, riprova"**;  
 Application::*redirect*(**'login/passwordChangeForm'**);  
 }  
 }  
 }  
}

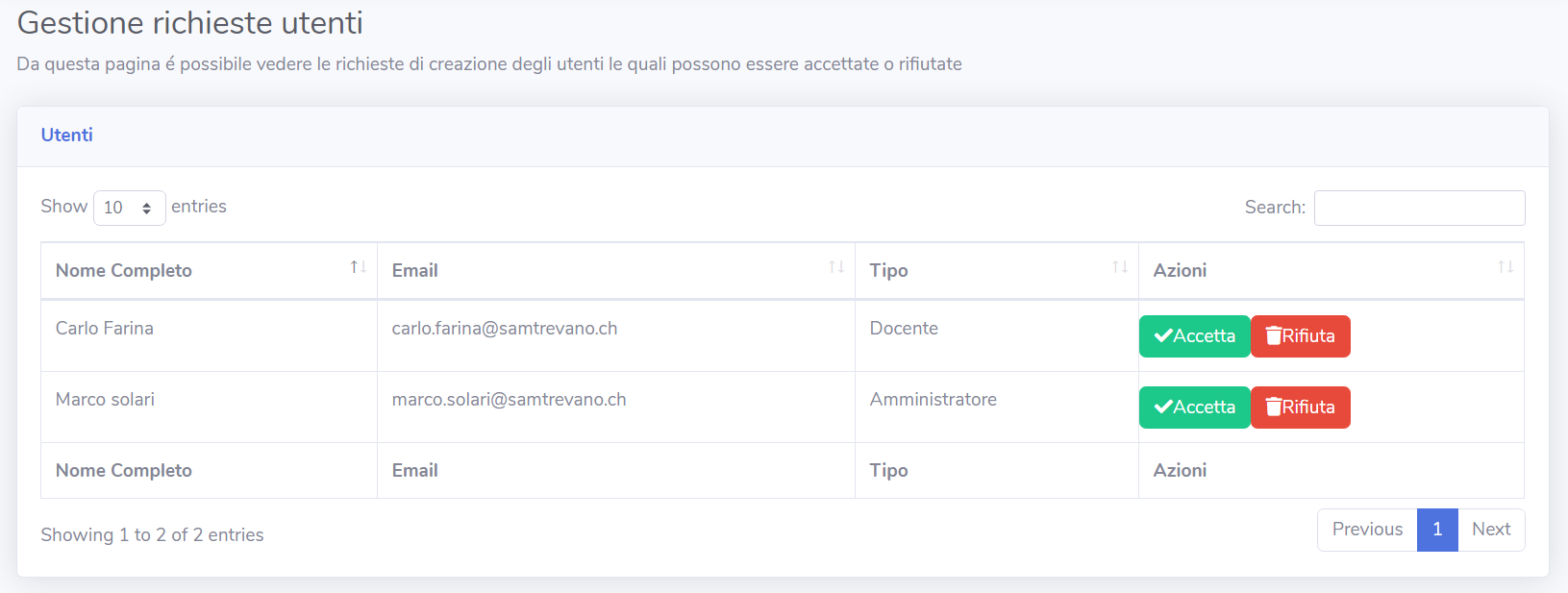
Nel caso invece l’utente dimentica la propria password può cliccare su link “password dimenticata” dove l’utente viene rimandato a un form dove dovrà inserire la sua email che verrà controllata se esiste all’interno della tabella degli utenti e poi manda la password con una nuova password provvisoria.

**public function** checkForgotPasssword(){  
 $\_SESSION[**"EmailError"**] = **""**;  
 *// check the method of request* **if**($\_SERVER[**"REQUEST\_METHOD"**] === **"POST"**){  
 *//check if the field is filled* **if**(**isset**($\_POST[**"userEmail"**]) && !**empty**($\_POST[**"userEmail"**])) {  
 *//check if the user exist* $userEmail = *htmlspecialchars*($\_POST[**"userEmail"**]);  
 $user = Users::*checkUserByEmail*($userEmail);  
 **if**($user == **true**){  
 *//create a new password for the user* $pass = Users::*createRandPassword*();  
 Users::*setProvvisoryPasswordByEmail*($pass,$userEmail);  
 $userName = Users::*getNameByEmail*($userEmail);  
 *//send the email to the user* MailSender::*lostPasswordSender*($userEmail,$pass,$userName);  
 }**else**{  
 $\_SESSION[**"EmailError"**] = **"Errore, questa mail non é collegata a un utente"**;  
 Application::*redirect*(**'home/login'**);  
 }  
 }  
 }  
}

### Creazione e visualizzazione utenti

Per la creazione degli utenti è stato creato un form accessibile dagli amministratori e dai docenti in cui si possono inserire le informazioni dell’utente da creare, il form necessita di poche informazioni quali il nome completo della persona, la mail con cui l'utente farà l’accesso al gestionale e il tipo di utente che bisogna creare.  
Il docente potrà solo creare utenti mentre l’amministratore potrà creare gli utenti e accettare le richieste di creazione degli stessi

.  
Appena il form viene completato le informazioni dell’utente finiscono prima in una tabella contenente tutte le richieste degli utenti le quali verranno visualizzate ed esaminate solo dagli amministratori, questi ultimi avranno la facoltà di accettare o rifiutare le richieste di creazione degli utenti.

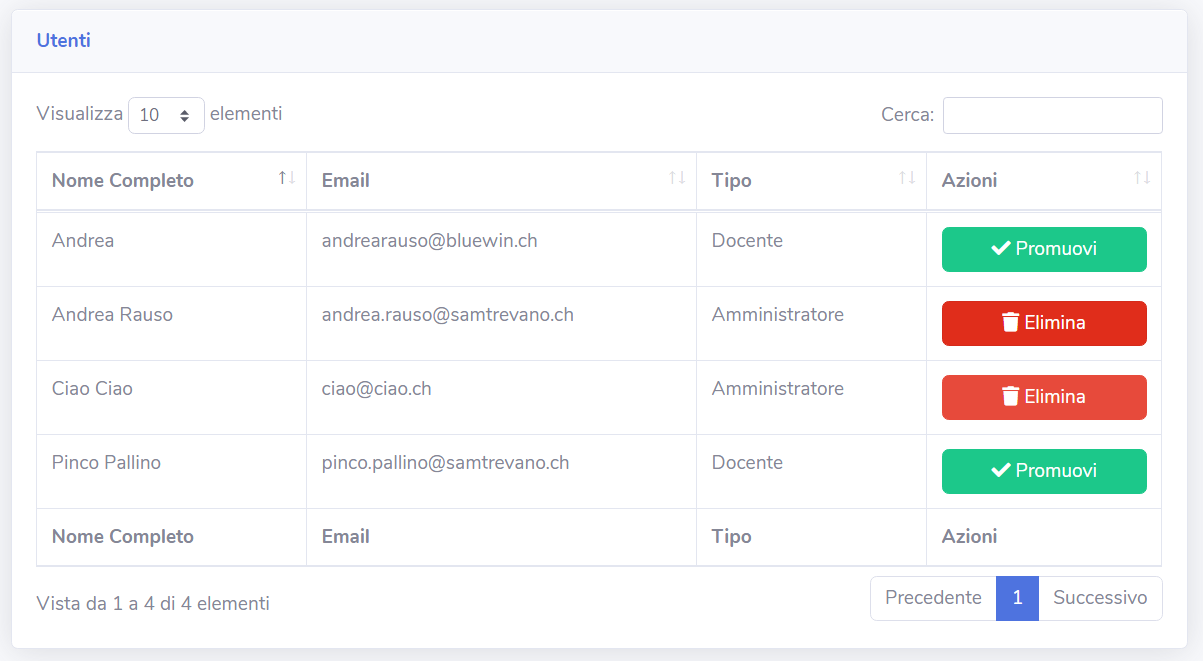


Quando le richieste vengono accettate viene inoltrata una richiesta all’controller Dashboard il quale poi chiamerà il metodo saveUser della classe Users che elaborerà la query per l’aggiunta dell’utente alla tabella corrispondente e se la query di aggiunta dell’utente va a buon fine eliminerà la richiesta dalla tabella delle richieste. Alla creazione dell’utente viene creata una password random che viene inviata in chiaro con la mail e ne viene fatto l’hash con l’algoritmo BCRYPT che viene salvato sul Database.

Nel caso invece venga rifiutata la richiesta viene solo eliminata la stessa dalla tabella delle richieste.

Gli utenti creati all’interno del gestionale sono visibili nella tabella alla pagina “Gestione utenti” da dove l’utente amministratore può promuovere gli utenti da docenti ad amministratori e può eliminarli, non può per ovvi motivi cancellare il proprio utente.

Ecco come si mostra la tabella utenti:



### Mail

L’invio di email viene fatto quando viene accettata la richiesta di creazione di un utente da parte di un amministratore o quando lo stesso utente non ricorda più la password per entrare nel gestionale. Per l’invio delle mail ho creato 2 metodi, uno per il primo caso (creazione) e il secondo per l’altro (password persa).

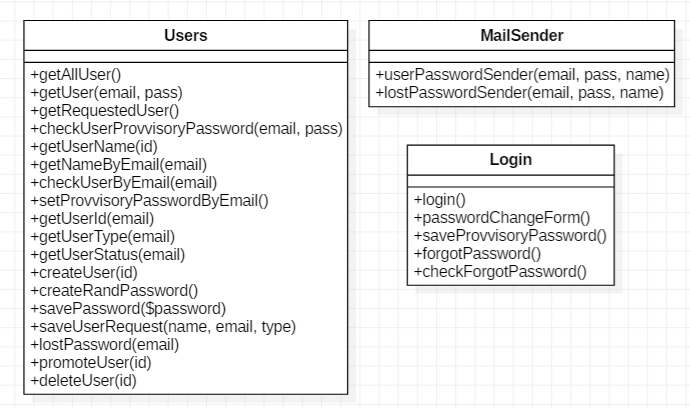
Esempio di mail per l’invio della password per la richiesta di una nuova password:

*// The receiver*$to = $email;  
*// The subject of the email*$subject = **"Invio Password"**;  
*/\* The header of the email including the sender and the program  
 that send the email\*/*$headers = **'From: fwebmaster@samtinfo.ch'** . **"\r\n"** .  
 **'X-Mailer: PHP/'** . *phpversion*() . **"\r\n"** .  
 **'Content-Type: text/html; charset=UTF-8'**;  
  
*//the text of the message*$message = **'Buongiorno signor/signora '**.$name. **', lei ha richiesto una nuova password per accedere al gestionale dei sondaggi'**. **"\r\n \r\n"** .  
 **'La sua mail per l\'accesso é: '**. $email . **"\r\n"** .  
 **'La sua password provvisoria è: '**. $pass . **"\r\n"** .  
 **'Appena avrà fatto il login le verrà chiesto di immettere una password che diventerà la sua nuova password di accesso.'**;  
  
*mail*($to, $subject, $message, $headers);

Per l’invio delle mail sfrutto il metodo mail contenuto nelle funzioni di base di PHP, la funzione richiede il destinatario della mail ($to), l’oggetto della mail ($subject), si ha bisogno di un header che contiene l’indirizzo del mittente che in questo caso è quello del server di scuola (fwebmaster@samtinfo.ch), X-Mailer conterrà il programma che è stato utilizzato per l’invio delle mail (es. Mozilla Thunderbird, Gmail, ecc...) che in questo contesto è PHP e serve anche il tipo di contenuto che si sta mandando e il charset per la formattazione corretta dei caratteri.

### Controller e model

Per l’implementazione del login e della gestione degli utenti ho creato le seguenti classi:



La classe Users gestisce tutte le informazioni relative agli utenti.

Descrizione della classe:

getAllUser: Metodo che ritorna tutti gli utenti contenuti nel database

getUser:Metodo che ritorna se l’utente esiste all’interno del database

getRequestedUser: Metodo che ritorna le richieste di creazione degli utenti

checkUserProvvisoryPassword: Metodo che controlla la password provvisoria dell‘utente

getUserName: Metodo che ritorna il nome dell’utente in base all’id

getNameByEmail: Metodo che ritorna il nome in base all‘email

checkUserByEmail: Metodo che controlla se esiste l’utente in base all‘email

setProvvisoryPasswordByEmail: metodo che imposta la password provvisoria all‘utente

getUserId: Metodo che ritorna l’id dell’utente

getUserType: Metodo che ritorna il tipo dell’utente (amministratore o docente)

getUserStatus: Metodo che ritorna se l’utente é stato attivato o meno

createUser: Metodo che permette la creazione della richiesta di creazione dell’utente

createRandPassword: Metodo che permette di creare una password random

savePassword: Metodo che permette di salvare la password dell’utente dopo che viene cambiata

saveUserRequest: Metodo che salva l’utente in base alla richiesta

lostPassword: Metodo che crea la password provvisoria per l’utente che l’ha persa e manda la mail

promoteUser: Metodo che permette la promozione di un utente da docente ad amministratore

DeleteUser: Metodo che permette l’eliminazione dell’utente

La classe MailSender si occupa principalemente dell’invio delle mail per le password  
Descrizione della classe:  
userPasswordSender: Questo metodo si occupa di mandare l’email con la password provvisoria quando viene creato l’utente

lostPasswordSender: Questo metodo si occupa di mandare l’email con la password provvisoria quando l’utente dimentica la propria password.

La classe Login è il controller che si occupa della gestione del login e dei form per il cambio di password

Descrizione dei metodi della classe:

login:

passwordChangeForm: Metodo che rimanda l’utente al form di cambio della password

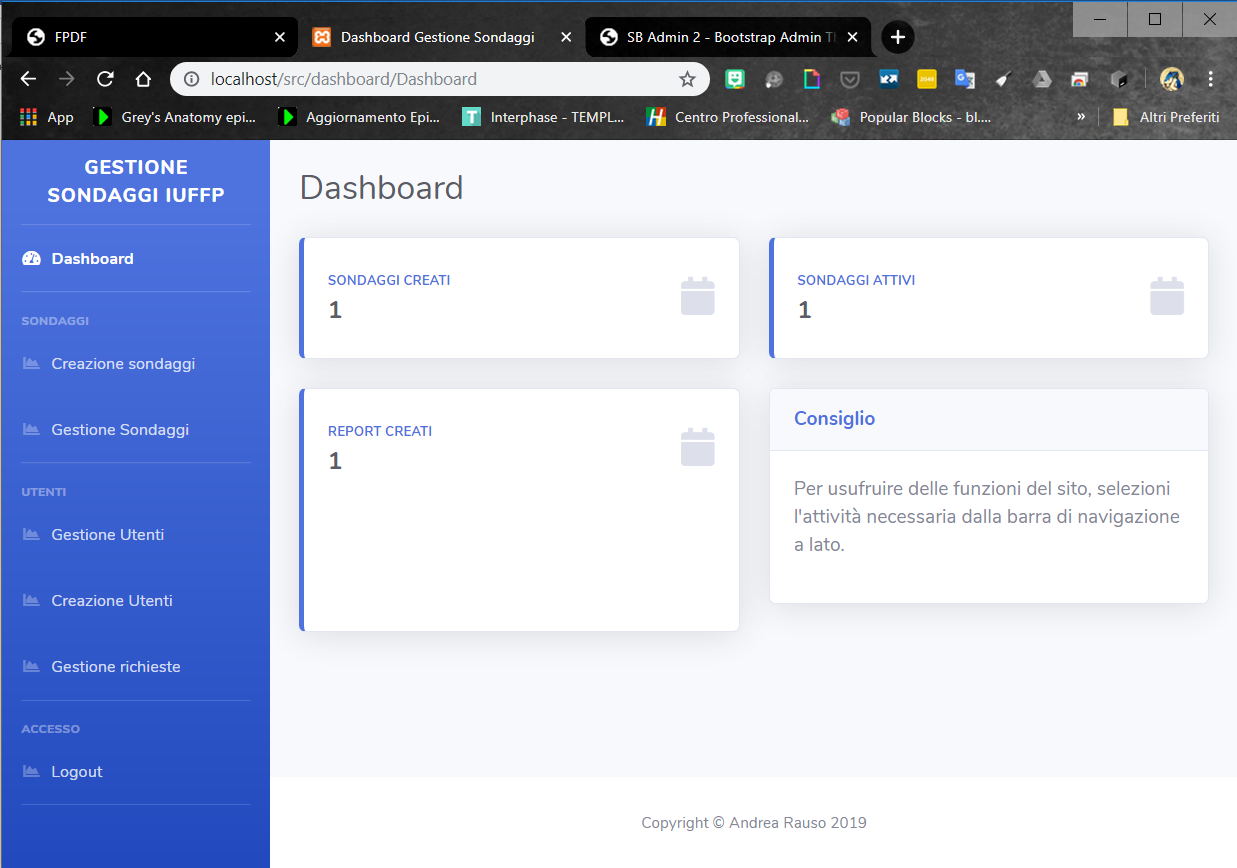
saveProvvisoryPassword: Metodo che richiama il model Users quando viene cambiata la password provvisoria

forgotPassword: Metodo che rimanda l’utente al form di inserimento della mail per mandare la password provvisoria

checkForgotPassword: Metodo che controlla la password provvisoria dell’utente che viene persa

## Dashboard

Appena l’amministratore fa l’accesso al gestionale viene mandato alla dashboard del sito dove da una barra laterale si può accedere alle pagine di creazione e gestione dei sondaggi e degli utenti.

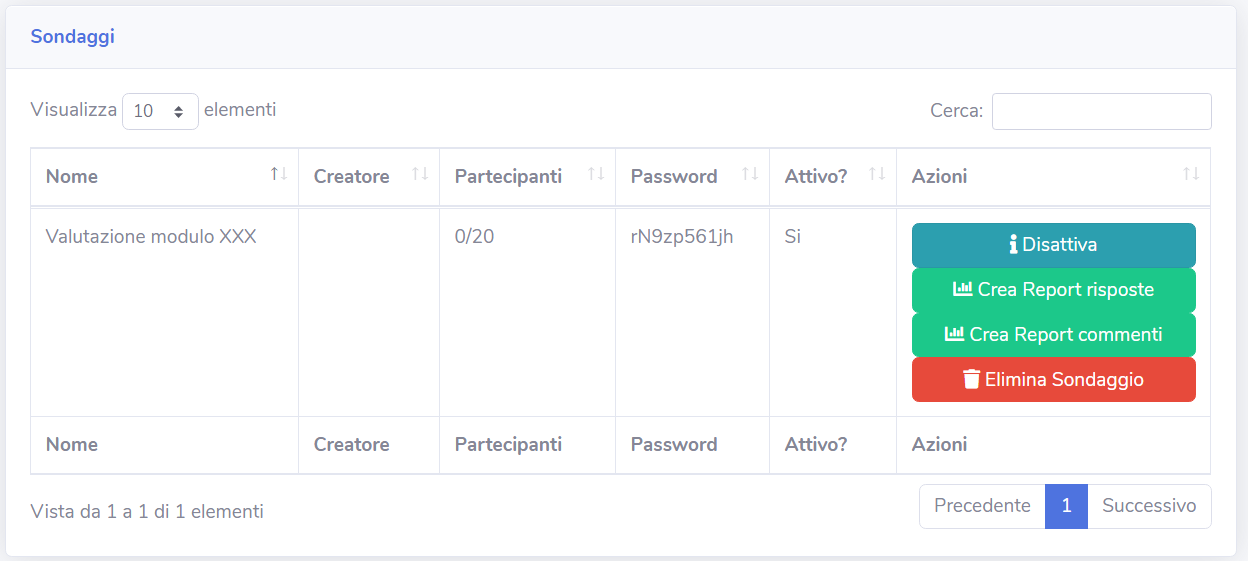
L’interfaccia è la seguente:

## Sondaggi

In questa sezione verrà descritta l’implementazione della parte dei sondaggi del sito andando a descrivere la creazione e l’aggiunta delle domande, il salvataggio dei sondaggi e l’accesso dei sondaggi da parte dell’utente.

### Visualizzazione dei sondaggi

Per visualizzare i sondaggi creati dagli utenti del gestionale viene visualizzata una tabella i cui dati vengono presi dal database del sito, i dati dei sondaggi che vengono mostrati sono il titolo del sondaggio, il creatore del sondaggio, il flag per indicare la limitazione degli utenti che possono partecipare al sondaggio, numero massimo (solo se il flag è impostato) di utenti che possono rispondere e numero di utenti che hanno già risposto. Alla tabella è aggiunto un campo contenente dei bottoni per creare un report, modificare il sondaggio e chiudere il sondaggio.

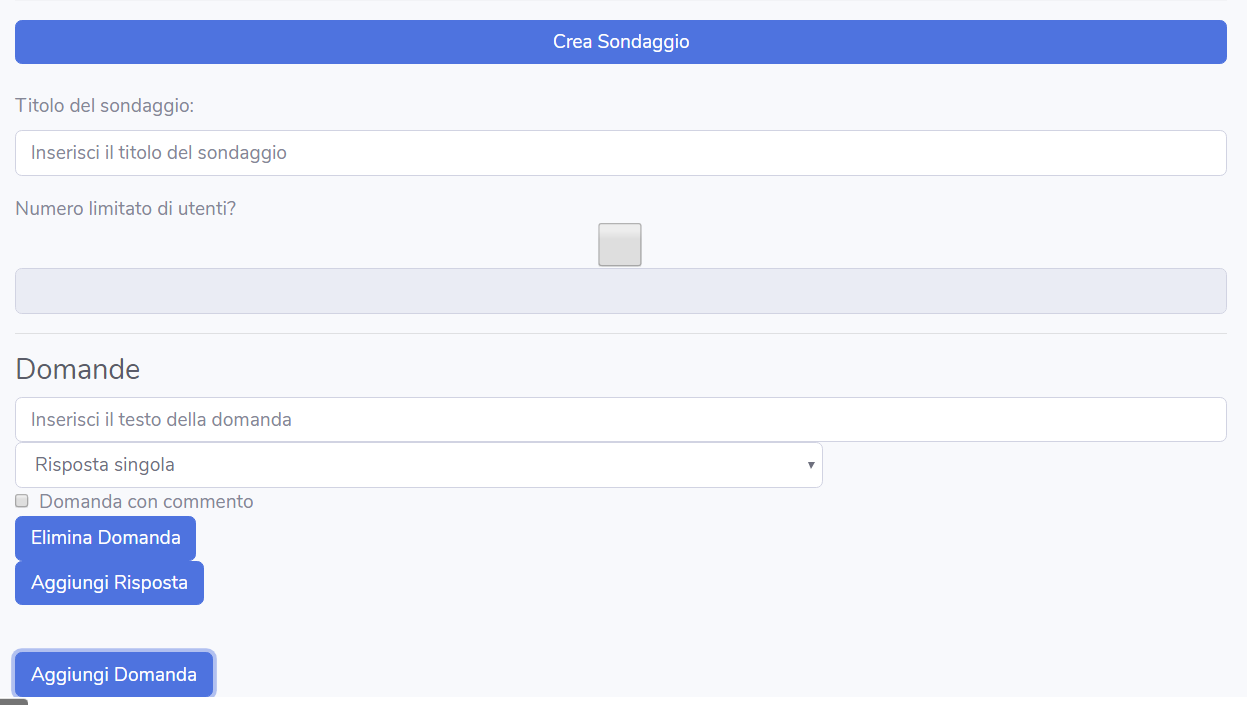


### Creazione dei sondaggi

Per la creazione dei sondaggi si viene rimandati in una pagina esterna in cui verrà fatto vedere il form per la creazione. All‘interno di questo form c’è il campo per immettere il titolo del sondaggio, un checkbox per indicare se al sondaggio possono rispondere un numero limitato di utenti e se questo box viene cliccato viene abilitato un box in cui inserire il numero di utenti massimo.

Per le domande che vengono create c’è un selettore per indicare quale tipo di domanda è (domanda a risposta singola, multipla, si/no, valutazione, ecc...), un input per inserire il testo della domanda, nel caso in cui si crea una domanda con risposte variabili (non si/no, valutazione) viene aggiunto un input per inserire il testo della risposta e un bottone per poter aggiungere le risposte. Alla fine del form della domanda un checkbox per indicare se la domanda avrà un box in cui l’utente potrà inserire i sondaggi.

Le domande possono essere aggiunte cliccando sul bottone “aggiungi domanda” che è localizzato in fondo al form.



Per la creazione delle domande all’interno del form di creazione mi baso su javascript dove in base a quale pulsante clicca esegue le azioni di aggiunta e l’eliminazione delle domande e delle risposte contenute all’interno del form, per la creazione degli elementi bi baso sulle funzionalita di DOM all’interno del documento html.

**var *questionNum*** = 0;  
**var *choiceNum*** = 0;  
**var *selectOption*** = [**"Risposta singola"**, **"Risposta Multipla"**, **"Risposta si/no"**, **"Valutazione"**];  
  
**function** *addQuestion*() {  
  
 *//create the question div* **var** br = ***document***.createElement(**"br"**);  
 **var** choiceDivName = **"Question"** + ***questionNum*** + **"Choices"**;  
 **var** surveyDiv = ***document***.getElementById(**"survey"**);  
 **var** questionDiv = ***document***.createElement(**"div"**);  
 *// create the input for the question title* **var** questionTitle = ***document***.createElement(**"input"**);  
 questionTitle.**type** = **"text"**;  
 questionTitle.**name** = **"Question"** + ***questionNum***;  
 questionTitle.**id** = **"Question"** + ***questionNum***;  
 questionTitle.**placeholder** = **"Inserisci il testo della domanda"**;  
 questionTitle.**classList**.add(**"form-control"**);  
 questionTitle.**required** = **true**;  
  
 questionDiv.**classList**.add(**"form-group"**);  
 *//append the input to the question div* questionDiv.appendChild(questionTitle);  
  
 *// create the question type selector* **var** selectType = ***document***.createElement(**"select"**);  
 selectType.**name** = **"Type"** + ***questionNum***;  
 selectType.**id** = **"Type"** + ***questionNum***;  
 **var** selectId = **"Type"** + ***questionNum***;  
 selectType.**onchange** = **function** () {  
 *changeChoice*(selectId, choiceDivName)  
 };  
 selectType.**classList**.add(**"form-control"**);  
 selectType.**classList**.add(**"col-md-8"**);  
 **for** (**var** i = 0; i < ***selectOption***.**length**; i++) {  
 **var** option = ***document***.createElement(**"option"**);  
 option.**value** = ***selectOption***[i];  
 option.**text** = ***selectOption***[i];  
  
 selectType.appendChild(option);  
 }  
 *//append the select to the question div* questionDiv.appendChild(selectType);  
  
 *// create the button for deleting the question* **var** button = ***document***.createElement(**"button"**);  
 button.**classList**.add(**"btn"**);  
 button.**classList**.add(**"btn-primary"**);  
 button.**id** = **"Question"** + ***questionNum***;  
 button.**value** = **"Elimina domanda"**;  
 button.**innerText** = **"Elimina Domanda"**;  
 **var** buttonId = **"Question"** + ***questionNum***;  
 button.**onclick** = **function** () {  
 *removeQuestion*(buttonId);  
 };  
   
 *// create the div for the comment checkbox* **var** commentDiv = ***document***.createElement(**"div"**);  
 commentDiv.**classList**.add(**"form-check"**);  
 *// create the button checkbox* **var** commentCheck = ***document***.createElement(**"input"**);  
 commentCheck.**classList**.add(**"form-check-input"**);  
 commentCheck.**type** =**"checkbox"**;  
 commentCheck.**id** = **"Question"** + ***questionNum*** + **"Comment"**;  
 commentCheck.**name** = **"Question"** + ***questionNum*** + **"Comment"**;  
 commentCheck.**value** = **"Domanda con commento"**;  
 *//create the label for the comment checkbox* **var** commentLabel = ***document***.createElement(**"label"**);  
 commentLabel.**htmlFor** = **"Question"** + ***questionNum*** + **"Comment"**;  
 commentLabel.**innerHTML** = **"Domanda con commento"**;  
 commentLabel.**classList**.add(**"form-check-label"**);  
  
 *//appending all elements* questionDiv.appendChild(commentDiv);  
 commentDiv.appendChild(commentCheck);  
 commentDiv.appendChild(commentLabel);  
 questionDiv.appendChild(br);  
 questionDiv.appendChild(br);  
 questionDiv.appendChild(button);  
  
 *//create div for the choice* **var** choiceDiv = ***document***.createElement(**"div"**);  
 choiceDiv.**name** = choiceDivName;  
 choiceDiv.**id** = choiceDivName;  
 questionDiv.appendChild(choiceDiv);  
  
 *// create the button for adding the choices* **var** choiceButton = ***document***.createElement(**"button"**);  
 choiceButton.**innerText** = **"Aggiungi Risposta"**;  
 choiceButton.**classList**.add(**"btn"**);  
 choiceButton.**classList**.add(**"btn-primary"**);  
 choiceButton.**onclick** = **function** () {  
 *addChoice*(***questionNum***, choiceDivName);  
 };  
  
 choiceDiv.appendChild(choiceButton);  
 questionDiv.appendChild(br);  
  
 questionDiv.**name** = **"Question"** + ***questionNum***;  
 questionDiv.**id** = **"Question"** + ***questionNum***;  
 surveyDiv.append(questionDiv);  
 ***questionNum***++;  
}

Come si vede dal codice vado a creare ogni elemento del form programmaticamente assegnandoli ogni attributo dopo aver creato l’oggetto

### Salvataggio dei sondaggi

Il salvataggio dei sondaggi viene fatto in base alla struttura del form fatta all’interno della pagina della creazione dei sondaggi. Per rendere semplice il collegamento di una risposta a una determinata domanda ho impostato i nomi all’interno del form nel modo seguente: question<numero>choice<numero>

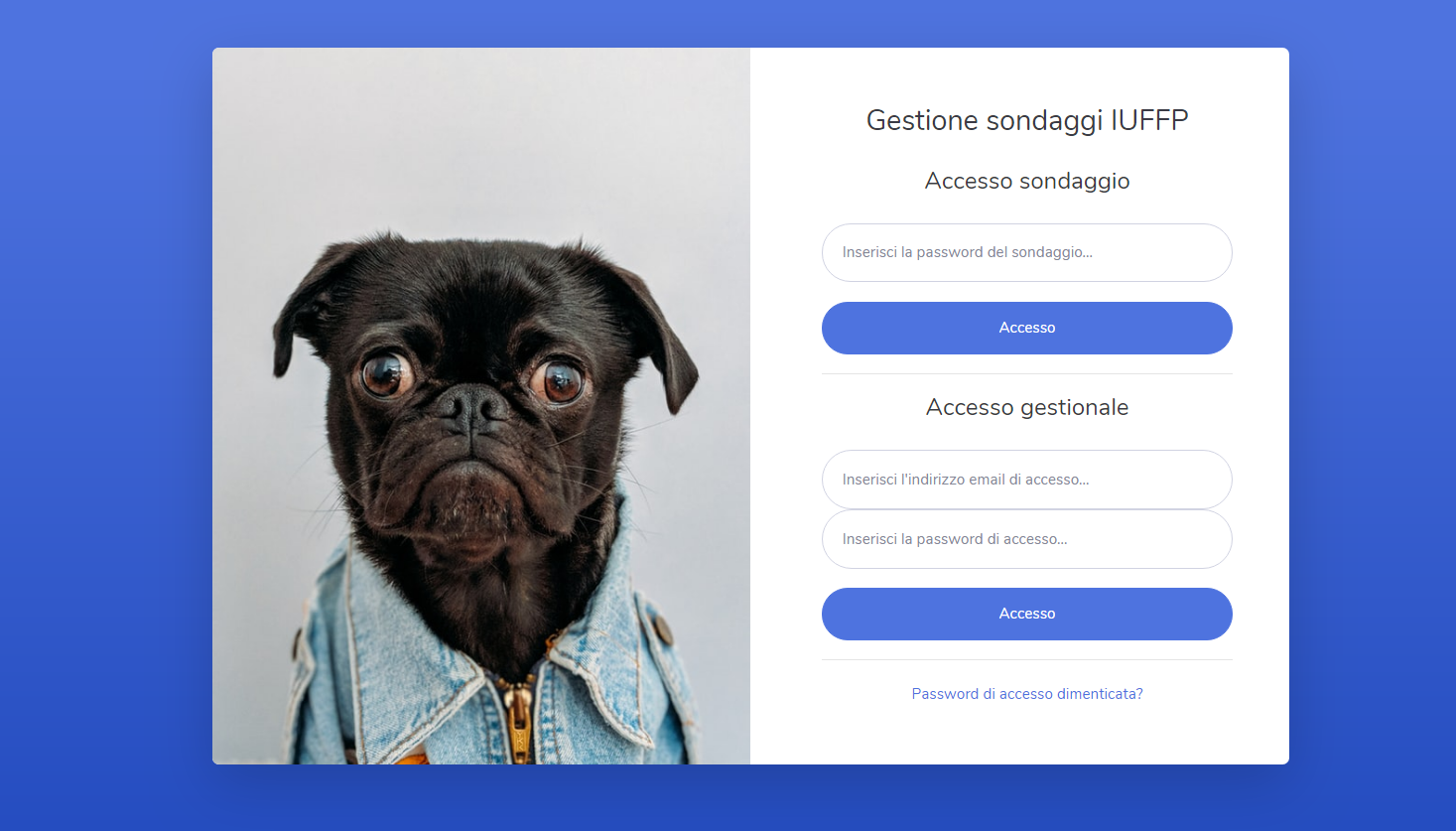
Quando vado ad analizzare tutte le variabili fornite dal post posso ciclare ogni singola domanda e ogni rispettiva risposta.

Il codice per fare tutto questo è il seguente:

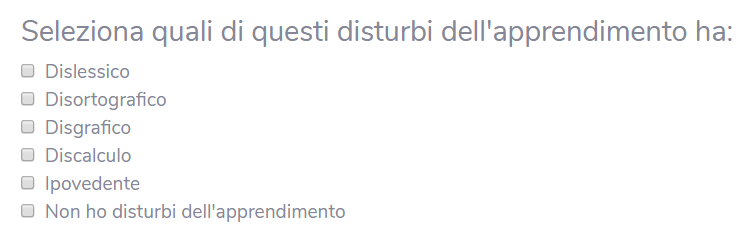
**public function** createSurvey()  
{  
 $surveyId = 0;  
 $surveyTitle = $\_POST[**"surveyTitle"**];  
   
 *//create the survey* **if**(**isset**($\_POST[**"MaxNumUser"**])){  
 $numUser = $\_POST[**"MaxNum"**];  
 $surveyId = Survey::*saveSurvey*($surveyTitle,1,$numUser);  
 }**else**{  
 $surveyId = Survey::*saveSurvey*($surveyTitle,0,0);  
  
 }  
 $questionId = 0;  
 *//for each question and choice created in the survey creator form* **foreach** ($\_POST **as** $key => $value){  
 *preg\_match\_all*(**"/[A-Za-z]+|\d+/"**, $key, $matches);  
 *print\_r*($matches);  
 **if**($matches[0][0] == **"Question"** && !**isset**($matches[0][2])){  
 *//creating all the key* $questionId = $matches[0][1];  
 $typeKey = **"Type"** . $questionId;  
 $questionTextKey = **"Question"** . $questionId;  
 $commentKey = **"Question"** . $questionId . **"Comment"**;  
 $choiceKey = **"Question"** . $questionId . **"Choice"**;  
 $questionType = $\_POST[$typeKey];  
 **echo** $\_POST[$typeKey];  
   
 *//depending of the type it will save the question and saves all choice created* **if**($questionType == **"Risposta singola"**){  
 **if**(**isset**($\_POST[$commentKey])){  
 $id = Survey::*saveQuestion*($surveyId,$\_POST[$questionTextKey],0,1);  
 **foreach** ($\_POST[$choiceKey] **as** $value){  
 Survey::*saveChoice*($id,$value);  
 }  
 }**else**{  
 Survey::*saveQuestion*($surveyId,$\_POST[$questionTextKey],0,0);  
 }  
  
 }**else if** ($questionType == **"Domanda si/no"**){  
 **if**(**isset**($\_POST[$commentKey])){  
 $id = Survey::*saveQuestion*($surveyId,$\_POST[$questionTextKey],0,1);  
 Survey::*saveChoice*($id,**"Si"**);  
 Survey::*saveChoice*($id,**"No"**);  
 }**else**{  
 Survey::*saveQuestion*($surveyId,$\_POST[$questionTextKey],0,0);  
 }  
  
 }**else if**($questionType == **"Valutazione"**){  
 **if**(**isset**($\_POST[$commentKey])){  
 $id = Survey::*saveQuestion*($surveyId,$\_POST[$questionTextKey],0,1);  
  
 Survey::*saveChoice*($id,**"Negativo"**);  
 Survey::*saveChoice*($id,**"Piuttosto negativo"**);  
 Survey::*saveChoice*($id,**"Indifferente"**);  
 Survey::*saveChoice*($id,**"Piuttosto positivo"**);  
 Survey::*saveChoice*($id,**"Positivo"**);  
 }**else**{  
 Survey::*saveQuestion*($surveyId,$\_POST[$questionTextKey],0,0);  
 }  
  
 }**else**{  
 **if**(**isset**($\_POST[$commentKey])){  
 $id = Survey::*saveQuestion*($surveyId,$\_POST[$questionTextKey],1,1);  
 **foreach** ($\_POST[$choiceKey] **as** $value){  
 Survey::*saveChoice*($id,$value);  
 }  
 }**else**{  
 Survey::*saveQuestion*($surveyId,$\_POST[$questionTextKey],1,0);  
 }  
 }  
 }  
 }  
}

### Accesso al sondaggio

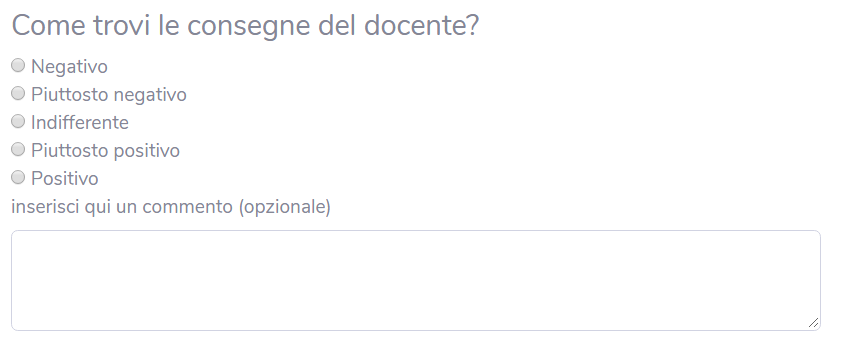
L’accesso ai sondaggi creati dagli amministratori e dai docenti viene effettuato attraverso la pagina principale del sito in cui è visualizzato un form per immettere la password del sondaggio che si vuole svolgere.

Le password per ogni singolo sondaggio sono create in modo random e viene effettuato un controllo nel caso che la password creata sia uguale a una dei sondaggi già creati in modo da crearne una nuova.

Appena verrà immesso la password e il nome dell’utente/studente il sito interrogherà il database per vedere se la password combacia con quelle contenute all’interno della tabella dei sondaggi e se la trova salva nella sessione l’id del sondaggio e poi rimanda alla pagina del sondaggio.

La pagina del sondaggio creerà il form del sondaggio in base alle domande create dagli amministratori, all’inizio del sondaggio viene incluso un form dove ci sono dei checkbox per indicare quali disturbi specifici dell’apprendimento ha la persona in questione, se l’utente non avesse nessun disturbo è possibile selezionare la casella “Non ho disturbi dell’apprendimento”.

Sotto al form del DSA vengono poi visualizzate le domande del sondaggio che sono state create dal docente/amministratore nell’ordine in cui sono state generate all’interno del form di creazione.



Alla fine del sondaggio si può cliccare sul pulsante “salva le risposte” che rimanderà alla pagina di login del sito.

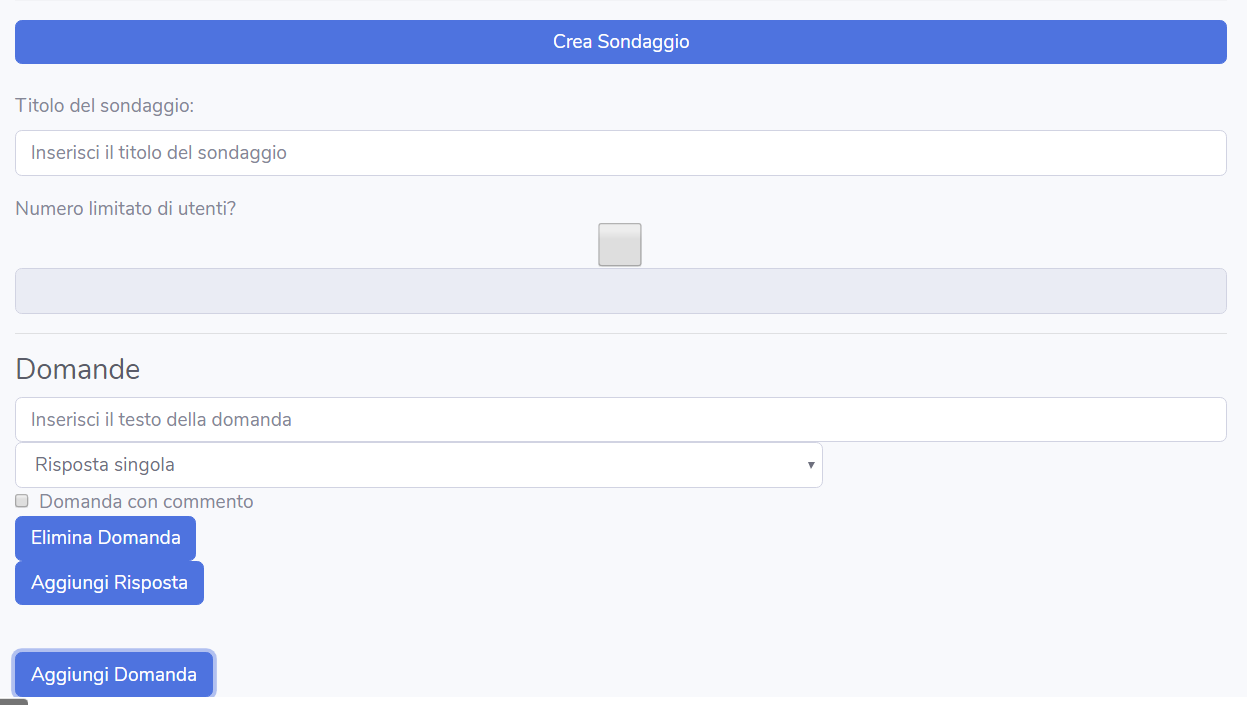
Per il salvataggio delle risposte seguo lo stesso principio che usato alla creazione del sondaggio dove in base al nome dell’input del form posso scomporre facilmente le variabili mandate in post e trovare le risposte date dall’utente.

Per il salvataggio delle risposte e dei commenti mi baso su 3 metodi contenuti nel classe Survey per il salvataggio delle informazioni per il DSA, il salvataggio delle risposte e il salvataggio dei commenti.

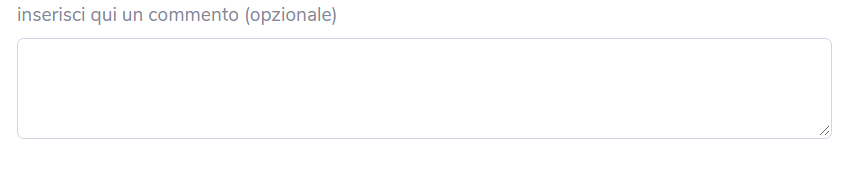
**public function** saveSurvey()  
{  
 *// get the survey id from session* $surveyId = $\_SESSION[**"surveyId"**];  
  
 *//update DSA value in the database* $DSAForm = $\_POST[**"DSAForm"**];  
 Survey::*updateDSA*($surveyId, $DSAForm);  
  
 *//for each POST it will update the value of choice based on the key value* **foreach** ($\_POST **as** $key => $value) {  
 *//the regex separates the string from the number  
 preg\_match\_all*(**"/[A-Za-z]+|\d+/"**, $key, $matches);  
 *//if it's a question* **if** ($matches[0] = **"question"**) {  
 *//if it's a checkbox form* **if** (*is\_array*($\_POST[$key])) {  
 **foreach** ($\_POST[$key] **as** $choice) {  
 Survey::*updateChoice*($choice);  
 }  
 *//if it's a radio button form* } **else** {  
 Survey::*updateChoice*($\_POST[$key]);  
 }  
 *// if it's a comment* } **else if** ($matches[0] == **"comment"**) {  
 $questionId = $matches[1];  
 $commentText = $\_POST[$key];  
 Survey::*saveComment*($questionId, $commentText);  
 }  
 }  
 *//update the survey value* Survey::*updateSurvey*($surveyId);  
 Application::*redirect*(**'home/home'**);  
}

### Commenti

L’aggiunta dei commenti all’interno delle domande dei sondaggi viene effettuato durante la creazione degli stessi, per poter aggiungere il box dei commenti all’interno della domanda è stato aggiunto un checkbox per indicare se l’utente, oltre alla risposta obbligatoria alla domanda, può inserire un coomento che invece è facoltativo.



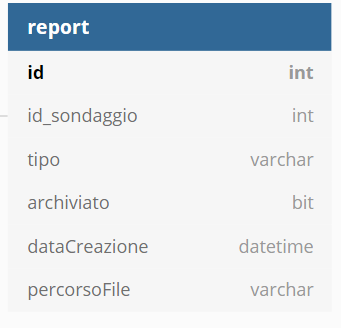
All’interno del form del sondaggio il box per inserire i commenti viene visualizzato subito dopo le risposte della domanda.



## Report

In questa parte viene descritta l’implementazione dei report i quali sono suddivisi in 2 tipi, il primo contenente le risposte alle domande mentre il secondo contiene i commenti alle domande.

Ho effettuato una modifica alla tabella del database in modo che tenesse conto del tipo di report che è stato creato da un docente/amministratore. Il campo inserito è chiamato “tipo” ed è di tipo varchar con dimensione 10, abbastanza per includere il tipo “Sondaggio” e “Commento”.

La tabella risulta la seguente:

Per la creazione dei report mi sono basato sulla libreria FPDF, il report si compone di tre parti: l’header, il corpo e il footer. L’header contiene il titolo del gestionale e il titolo del sondaggio, il corpo del report include nel caso dei sondaggio dei grafici a barre orizzontali dove per ogni barra viene scritta la risposta del sondaggio

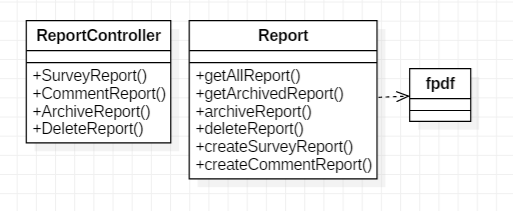
I commenti hanno un filename comune per entrambi basato su data di creazione, id del sondaggio e tipo di report, cosi da renderli univoci evitando la sovrascrittura di altri file.

All’interno del codice per creare i report sono state incluse delle parti della classe FPDF Sector che permette di creare i grafici a barre orizzontali.

Il metodo seguente permette di creare i report delle risposte dei sondaggi passando come attributo l’id del sondaggio, dopodiché va a prendere i vari dati dei DSA e delle domande per poter costruire i grafici delle risposte di quel sondaggio, il metodo per il report dei commenti è simile ma senza i grafici poiché deve solo stampare i commenti delle domande.

**public static function** createSurveyReport($id)  
{  
 $reportData = **array**();  
 $conn = Database::*get*();  
  
 *// get the survey* $surveyQuery = $conn->prepare(**"Select** *\** **from sondaggio where sondaggio.id = ?"**);  
 $surveyQuery->bindParam(1, $id, PDO::***PARAM\_STR***);  
  
 $surveyQuery->execute();  
 $survey = $surveyQuery->fetchAll();  
  
 *// fill the array with value of DSA* $DSAArray = **array**(**"Dislessico"** => $survey[0][**"numDislessici"**],  
 **"Disgrafico"** => $survey[0][**"numDisgrafici"**],  
 **"Disortografico"** => $survey[0][**"numDisortografici"**],  
 **"Discalculo"** => $survey[0][**"numDiscalculi"**],  
 **"Ipovedente"** => $survey[0][**"numIpovedenti"**],  
 **"Nessun DSA"** => $survey[0][**"numNoDSA"**]  
 );  
  
 *// get the question of the survey* $questionQuery = $conn->prepare(**"Select** *\** **from domanda where domanda.idSondaggio = ?"**);  
 $questionQuery->bindParam(1, $id, PDO::***PARAM\_STR***);  
  
 $questionQuery->execute();  
 $questions = $questionQuery->fetchAll();  
  
 **global** $title;  
 $title = $survey[0][**"title"**];  
  
 *// create the pdf* $diag = **new** PDF\_Sector(**"P"**, **"mm"**, **"A4"**);  
 $diag->AddPage();  
  
 *// create the horizontal bar chart for the DSA* $diag->SetFont(**'Arial'**, **'BIU'**, 12);  
 $diag->Cell(0, 5, **"Disturbi specifi dell'apprendimento"**, 0, 1);  
 $diag->Ln(8);  
 $valX = $diag->GetX();  
 $valY = $diag->GetY();  
 $diag->BarDiagram(190, 70, $DSAArray, **'%l : %v (%p)'**, **array**(255, 175, 100));  
 $diag->SetXY($valX, $valY + 80);  
  
 *// iterate the question and create for each the bar diagram* **foreach** ($questions **as** $question) {  
 $reportData = **array**();  
 $questionTitle = $question[**"testoDomanda"**];  
 *// get the chioce of the question* $choiceQuery = $conn->prepare(**"Select** *\** **from risposta where risposta.idDomanda = ?"**);  
 $choiceQuery->bindParam(1, $question[**"id"**], PDO::***PARAM\_STR***);  
 $choiceQuery->execute();  
 $choices = $choiceQuery->fetchAll();  
   
 **for** ($i = 0; $i < *count*($choices); $i++) {  
 $reportData[$choices[$i][**"testoRisposta"**]] = $choices[$i][**"numRisposte"**];  
 }  
 *// create the bar chart for the question* $diag->SetFont(**'Arial'**, **'BIU'**, 12);  
 $diag->Cell(0, 5, $questionTitle, 0, 1);  
 $diag->Ln(8);  
 $valX = $diag->GetX();  
 $valY = $diag->GetY();  
 $diag->BarDiagram(190, 70, $reportData, **'%l : %v (%p)'**, **array**(255, 175, 100));  
 $diag->SetXY($valX, $valY + 80);  
 }  
 *//save the file in the server* $filename = *date*(**"Y-m-d\_H:i:s"**) . **"\_"** . $id . **"\_"** . **"commentReport.pdf"**;  
 $diag->Output(**"F"**, ***FILE\_URL*** . **'/application/views/report/'** . $filename);  
  
 $location = ***FILE\_URL*** . **'/application/views/report/'** . $filename;  
 $date = *date*(**"Y-m-d\_H:i:s"**);  
   
 *//insert the report in the database* $conn = Database::*get*();  
 $reportQuery = $conn->prepare(**"insert into report values (null,?,?,'Sondaggio',0,?)"**);  
 $reportQuery->bindParam(1, $id, PDO::***PARAM\_INT***);  
 $reportQuery->bindParam(1, $date, PDO::***PARAM\_STR***);  
 $reportQuery->bindParam(1, $location, PDO::***PARAM\_STR***);  
 $reportQuery->execute();  
  
 Application::*redirect*(**'dashboard/SurveyManager'**);  
}

### Controller e model

Le classi in uso per l’implementazione dei report è la seguente:

ReportController è il controller che gestisce la creazione, l’archiviazione e l’eliminazione dei report.

Descrizione della classe:

**SurveyReport:** Metodo che richiama il model Report per la creazione del report delle risposte, quando finisce ritorna alla pagina di gestione dei report

**CommentReport:** Metodo che richiama il model Report per la creazione del report dei commenti, quando finisce ritorna alla pagina di gestione dei report

**ArchiveReport:** Metodo che richiama il model per poter archiviare un determinato report

**DeleteReport:** Metodo che richiama il model per poter eliminare un determinato report

La classe Report contiene i metodi che vengono chiamati dal controller ReportController per permettere la richiesta dei report, la creazione, l’archiviazione e la cancellazione dei report. Questa classe usa la classe della libreria FPDF per la creazione dei report sotto forma di PDF:

Descrizione della classe:

**getAllReport:** Metodo che ritorna tutti i report creati dagli utenti del gestionale che non sono stati archiviati

**getArchivedReport:** Metodo che ritorna i report archiviati

**archiveReport:** Metodo che in base all’id del report permette di archiviarlo

**deleteReport:** Metodo che permette di cancellare un report dal database

**createSurveyReport:** Metodo che permette di creare il report delle risposte dei sondaggi in base all’id del sondaggio

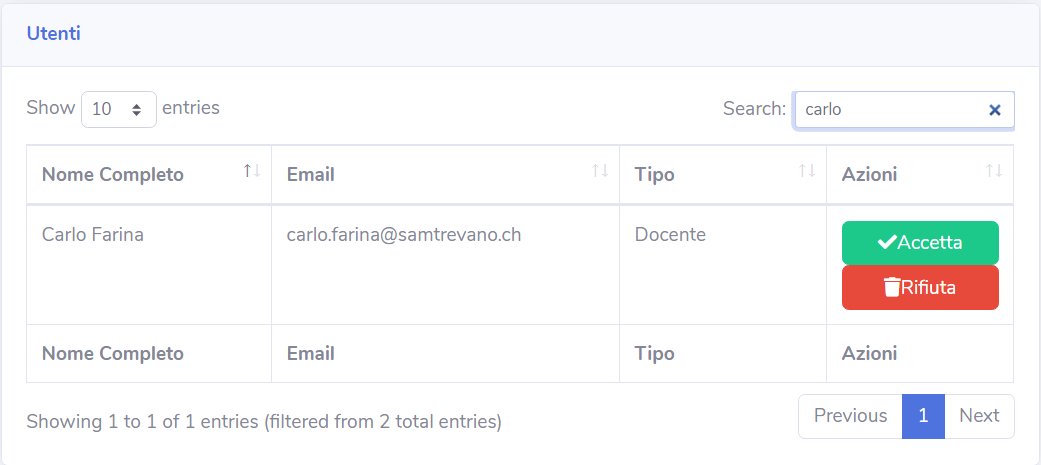
**createCommentReport:** Metodo che permette di creare il report dei commenti dei sondaggi in base all’id del sondaggio

## Ricerca

In questa parte viene descritta l’implementazione della ricerca all’interno della tabella dei sondaggi, degli utenti e dei report.

La ricerca all’interno delle tabelle è stata implementata assieme al template del sito il quale ha incorporato la libreria Datatables che permette di avere più funzionalità all’interno di una tabella come ricerca, ordinamento su una o più colonne, paginazione ecc...

Come si può vedere dalla tabella cercando carlo all’interno della tabella ne risulta 1 dai 2 filtrati.

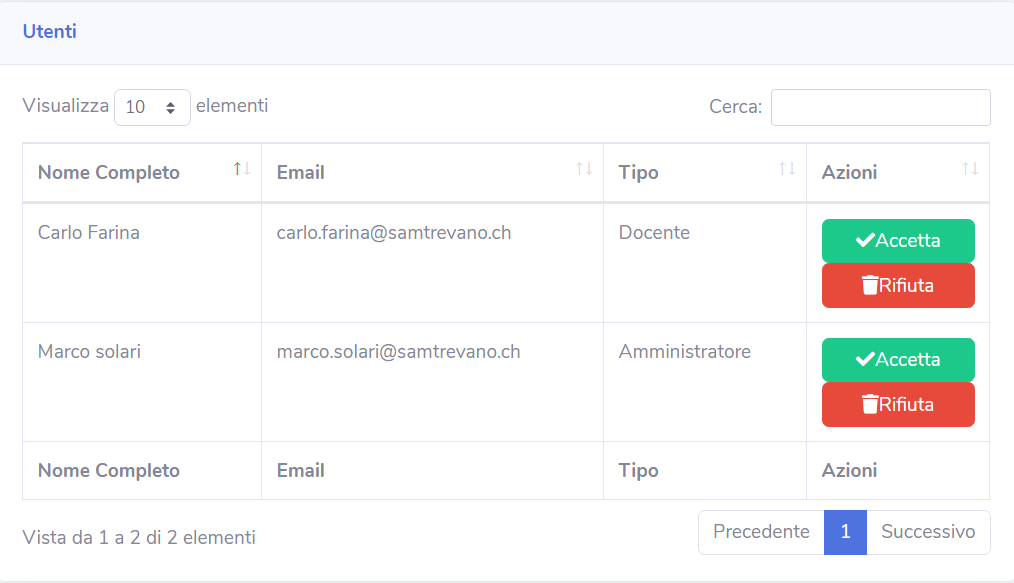


Essendo che la tabella viene impostata di base in lingua inglese ho dovuto cercare un modo di cambiare la lingua facendo vedere la tabella in italiano. Per impostare la lingua italiana alla tabella tramite jQuery ho impostato le informazioni della tabella alla sua inizializzazione.

**$**(***document***).**ready**(**function** () {  
 **$**(**'#userRequestData'**).**dataTable**({  
 **language**: {  
 **sEmptyTable**: **"Nessun dato presente nella tabella"**,  
 **sInfo**: **"Vista da \_START\_ a \_END\_ di \_TOTAL\_ elementi"**,  
 **sInfoEmpty**: **"Vista da 0 a 0 di 0 elementi"**,  
 **sInfoFiltered**: **"(filtrati da \_MAX\_ elementi totali)"**,  
 **sInfoPostFix**: **""**,  
 **sInfoThousands**: **"."**,  
 **sLengthMenu**: **"Visualizza \_MENU\_ elementi"**,  
 **sLoadingRecords**: **"Caricamento..."**,  
 **sProcessing**: **"Elaborazione..."**,  
 **sSearch**: **"Cerca:"**,  
 **sZeroRecords**: **"La ricerca non ha portato alcun risultato."**,  
 **oPaginate**: {  
 **sFirst**: **"Inizio"**,  
 **sPrevious**: **"Precedente"**,  
 **sNext**: **"Successivo"**,  
 **sLast**: **"Fine"** },  
 **oAria**: {  
 **sSortAscending**: **": attiva per ordinare la colonna in ordine crescente"**,  
 **sSortDescending**: **": attiva per ordinare la colonna in ordine decrescente"** }  
 }  
 });  
});

Quando il documento è pronto viene chiamata una funzione che andra a prendere la tabella in base all’id impostato. Da questo punto sotto language vengono impostate le varie stringhe di testo.

L’interfaccia della tabella con la lingua italiana impostata risulta la seguente:



# Test

## Protocollo di test

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-001  REQ-001 | **Nome:** | Accesso a sondaggio tramite password |
| **Descrizione:** | Test dell’accesso di un utente a un sondaggio | | |
| **Prerequisiti:** | Il sondaggio deve essere stato creato | | |
| **Procedura:** | 1. Prendere la password del sondaggio dato dall’amministratore/docent 2. Accedere alla pagina di login del sito 3. Inserire la password di accesso del sondaggio | | |
| **Risultati attesi:** | All’accesso si dovrebbe vedere il sondaggio collegato alla password, in caso contrario verrà ritornato un errore che non esiste il sondaggio. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-002  REQ-002 | **Nome:** | Cambiamento password al primo login |
| **Descrizione:** | Test del cambiamento della password dell’utente quando avviene il primo login al gestionale | | |
| **Prerequisiti:** | Deve essere creato un amministratore che possa creare e accettare gli utenti.  Il servizio di mail deve essere attivo | | |
| **Procedura:** | 1. Creare un utente di tipo Amministratore/docente inserendo le informazioni. 2. Accettare la richiesta di creazione dell’utente 3. Prendere i dati contenuti nella mail inviata all’indirizzo mail dell’utente 4. Accedere alla pagina di login 5. Inserire le informazioni nel form di login | | |
| **Risultati attesi:** | Al primo accesso deve essere visibile il form di cambio della password, in caso contrario verrà ritornato un errore che indicherà che l’utente ha sbagliato i dati d’accesso. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-003  REQ-002 | **Nome:** | Cambiamento della password quando viene persa |
| **Descrizione:** | Test della richiesta di una nuova password quando l’utente perde la propria | | |
| **Prerequisiti:** | Il servizio di mail deve essere attivo | | |
| **Procedura:** | 1. Accedere alla pagina di login del gestionale 2. Cliccare “Password dimenticata?” alla fine del form. 3. Inserire la propria mail di accesso 4. Prendere i dati d’accesso provvisori forniti nella mail inviata 5. Accedere alla pagina di login e inserire i dati d’accesso 6. Accedere con i nuovi dati | | |
| **Risultati attesi:** | All’accesso con i dati provvisori dovrà essere visibile il form per il cambio della password, in caso contrario verrà ritornato un errore che i dati di accesso sono errati.  L’utente poi potrà fare l’accesso con i nuovi dati di accesso. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-004  REQ-003 | **Nome:** | Test gestione degli utenti |
| **Descrizione:** | Si vuole testare la gestione degli utenti con la creazione, la gestione delle richieste e la promozione degli utenti. | | |
| **Prerequisiti:** | Deve essere creato un utente di tipo amministratore. | | |
| **Procedura:** | 1. Accedere al gestionale con un utente amministratore 2. Accedere alla pagina di creazione utenti 3. Creare un utente di tipo docente 4. Accedere alla pagina gestione richieste utenti 5. Accettare la richiesta di creazione dell’utente 6. Accedere con il nuovo utente e testare le pagine 7. Logout e accesso con utente amministratore 8. Accedere alla pagina gestione utenti 9. Promuovere l’utente appena creato 10. Accedere al gestionale con l’utente creato | | |
| **Risultati attesi:** | L’utente creato può consultare le pagine del gestionale come docente e dopo la promozione quelle dell’amministratore. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-005  REQ-004 | **Nome:** | Creazione di un sondaggio |
| **Descrizione:** | Si vuole testare la creazione e il salvataggio dei sondaggi | | |
| **Prerequisiti:** | Deve essere creato un utente di tipo amministratore o docente. | | |
| **Procedura:** | 1. Accedere al gestionale con un utente amministratore o docente 2. Accedere alla pagina di creazione sondaggio 3. Inserire le informazioni principali del sondaggio (titolo, utenti, ecc.) 4. Aggiungere delle domande di vario tipo 5. Aggiungere delle risposte alle domande 6. Cliccare su salva sondaggio 7. Accedere al sondaggio con la password creata al salvataggio del sondaggio | | |
| **Risultati attesi:** | Deve essere visibile la pagina del sondaggio con le domande immesse nel form di creazione. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-006  REQ-005 | **Nome:** | Test creazione delle domande |
| **Descrizione:** | Si vuole testare che quando l’utente sceglie che la domanda è di tipo sì/no o valutazione non ci sia modo di aggiungere delle risposte visto che sono già preimpostate | | |
| **Prerequisiti:** | Deve essere creato un utente di tipo amministratore o docente. | | |
| **Procedura:** | 1. Accedere al gestionale dei sondaggi 2. Andare alla pagina di creazione sondaggi 3. Creare delle domande selezionando come opzione si/no o valutazione 4. Cambiare le opzioni delle domande | | |
| **Risultati attesi:** | Il form non dovrà visualizzare il pulsante per l’aggiunta delle risposte quando l’utente sceglie le domande preimpostate e dovrà eliminare le risposte create mentre per le altre mostrerà il pulsante aggiungi domanda. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-007  REQ-005 | **Nome:** | Test creazione delle domande |
| **Descrizione:** | Si vuole testare che non sia possibile creare un sondaggio con delle domande vuote | | |
| **Prerequisiti:** | Deve essere creato un utente di tipo amministratore o docente. | | |
| **Procedura:** | 1. Accedere al gestionale 2. Andare alla pagina creazione 3. Creare delle domande senza aggiungere un testo 4. Creare delle risposte senza un testo 5. Cliccare su salva sondaggio | | |
| **Risultati attesi:** | Quando l’utente vuole salvare il sondaggio viene mostrato l’errore che l’utente deve riempire tutti i form. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-008  REQ-006 | **Nome:** | Creazione report risposte |
| **Descrizione:** | Si vuole testare la creazione del report delle risposte | | |
| **Prerequisiti:** | Deve essere creato un utente di tipo amministratore o docente.  Deve essere creato un sondaggio  Il sondaggio deve essere già stato fatto da almeno una persona | | |
| **Procedura:** | 1. Accedere al gestionale 2. Andare alla pagina Gestione Sondaggi 3. Cliccare sul pulsante “Crea report sondaggio” 4. Andare alla pagina Gestione report 5. Visualizzare il report creato cliccando su “Visualizza report” | | |
| **Risultati attesi:** | Il report deve contenere tutte le domande e le relative risposte e anche le risposte relative al form DSA. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-009  REQ-006 | **Nome:** | Creazione report risposte |
| **Descrizione:** | Si vuole testare la creazione del report dei commenti | | |
| **Prerequisiti:** | Deve essere creato un utente di tipo amministratore o docente.  Deve essere creato un sondaggio  Il sondaggio deve essere già stato fatto da almeno una persona | | |
| **Procedura:** | 1. Accedere al gestionale 2. Andare alla pagina Gestione Sondaggi 3. Cliccare sul pulsante “Crea report sondaggio” 4. Andare alla pagina Gestione report 5. Visualizzare il report creato cliccando su “Visualizza report” | | |
| **Risultati attesi:** | Il report dei commenti conterrà i commenti delle risposte che hanno il flag “conCommento” attivo. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-010  REQ-006 | **Nome:** | Archiviazione dei report |
| **Descrizione:** | Si vuole testare l’archiviazione dei report | | |
| **Prerequisiti:** | Deve essere creato un utente di tipo amministratore o docente.  Il sondaggio e un report relativo allo stesso deve essere stato creato. | | |
| **Procedura:** | 1. Accedere al gestionale 2. Andare alla pagina Gestione Report 3. Cliccare sul pulsante “Archivia report” di un qualsiasi report 4. Andare alla pagina Report Archiviati | | |
| **Risultati attesi:** | Il report non viene più mostrato nella tabella della pagina Gestione Report e solo nella pagina Report Archiviati | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-011  REQ-006 | **Nome:** | Eliminazione di un report |
| **Descrizione:** | Si vuole testare l’eliminazione dei report | | |
| **Prerequisiti:** | Deve essere creato un utente di tipo amministratore o docente.  Il sondaggio e un report relativo allo stesso deve essere stato creato. | | |
| **Procedura:** | 1. Accedere al gestionale 2. Andare alla pagina Gestione Report o Report Archiviato 3. Cliccare sul pulsante “Elimina report” di un qualsiasi sondaggio | | |
| **Risultati attesi:** | Il report non dovrebbe essere più visibile all’interno della tabella. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-012  REQ-007 | **Nome:** | Test della ricerca |
| **Descrizione:** | Si vuole testare la ricerca di testo all’interno delle tabelle di gestione del sito | | |
| **Prerequisiti:** | Deve essere creato un utente di tipo amministratore o docente. | | |
| **Procedura:** | 1. Accedere al gestionale con un qualsiasi utente 2. Andare in una pagina di gestione 3. All’interno della barra di ricerca inserire un testo specifico | | |
| **Risultati attesi:** | All’interno della tabella sono visibili soltanto gli elementi con contengono il testo della ricerca. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case:**  **Riferimento**: | TC-013  REQ-008 | **Nome:** | Design del sito |
| **Descrizione:** | Si vuole testare il design del sito nell’essere responsive | | |
| **Prerequisiti:** | Deve essere creato un utente di tipo amministratore o docente. | | |
| **Procedura:** | 1. Accedere al gestionale 2. Navigare tra le pagine del gestionale 3. Ridimensionare la finestra del browser in diversi formati 4. Se disponibile usare l’ispeziona elemento e mettere la visualizzazione da mobile e tablet e ridimensionare | | |
| **Risultati attesi:** | Le pagine sono visibili e navigabili su entrambe le piattaforme fisse e mobile | | |

## Risultati test

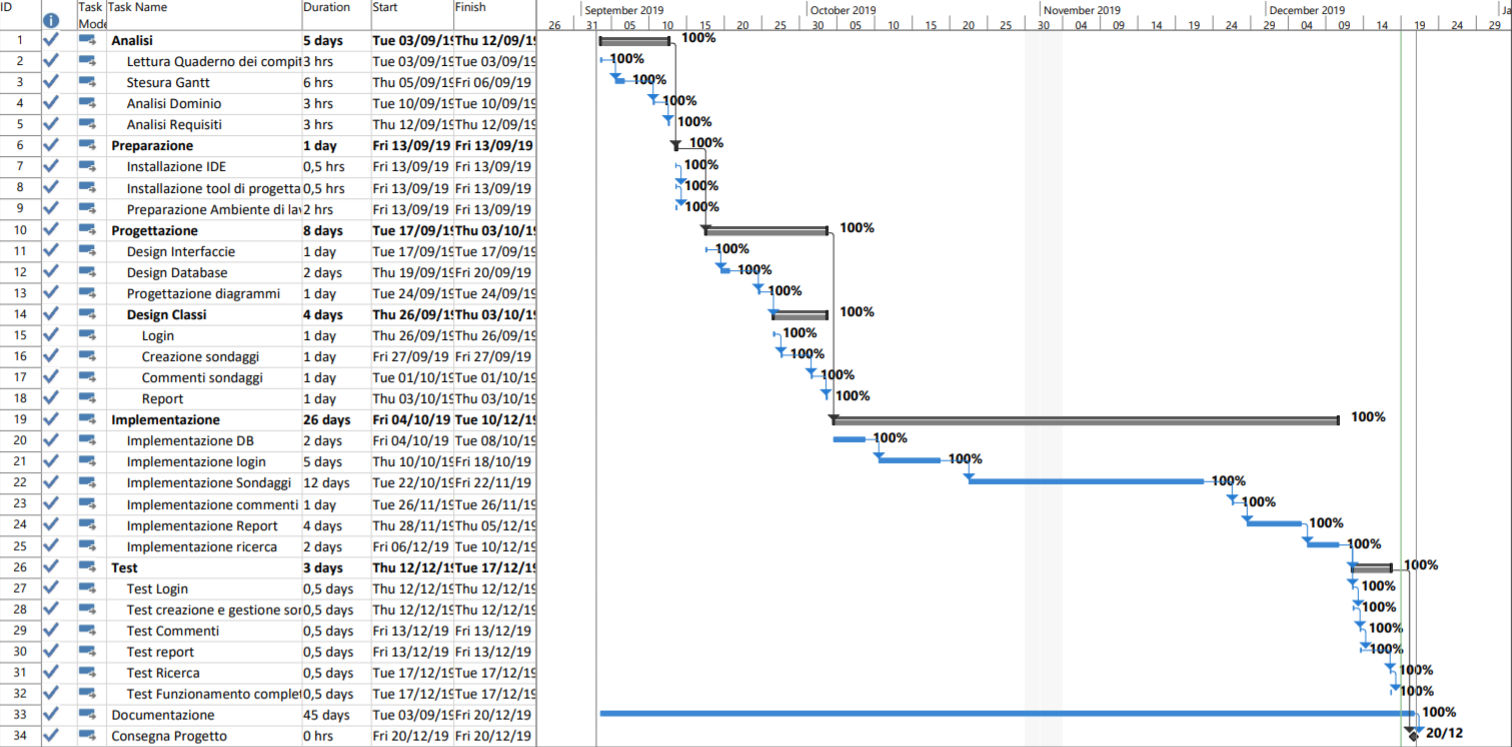
|  |  |  |
| --- | --- | --- |
| Test Case | Risultato | Osservazioni |
| TC-001 | POSITIVO | ... |
| TC-002 | POSITIVO |  |
| TC-003 | POSITIVO |  |
| TC-004 | POSITIVO |  |
| TC-005 | POSITIVO |  |
| TC-006 | POSITIVO |  |
| TC-007 | POSITIVO |  |
| TC-008 | NEGATIVO | Ci sono degli errori di fpdf che non mi permettono di creare i report delle risposte |
| TC-009 | NEGATIVO | Ci sono degli errori di fpdf che non mi permettono di creare i report |
| TC-010 | NEGATIVO | Non riuscendo a creare i report non sono riuscito a testare l’archiviazione degli stessi |
| TC-011 | NEGATIVO | Non riuscendo a creare i report non sono riuscito a testare l’eliminazione degli stessi |
| TC-012 | POSITIVO |  |
| TC-013 | POSITIVO |  |

## Mancanze/limitazioni conosciute

All’interno della creazione della tabella non esiste un check che controlla che tutte le domande abbiano almeno una risposta, solo per le domande a risposta singola e multipla poiché le altre le risposte sono già preimpostate.

Non è stato implementato un sistema per la gestione degli accessi alle pagine tra amministratore e docente.

# Consuntivo



Come si può vedere dal Gantt consuntivo per le parti di analisi e di progettazione sono rimasto nei tempi preventivati all’inizio del progetto, la parte di implementazione è durata molto più di quello che è stato preventivato dove la parte di implementazione dei sondaggi è quella che ne ha richiesto di più, principalmente perché non ho pensato bene alla strutturazione delle classi per poter implementare la creazione, gestione e visualizzazione. La parte di implementazione dei commenti è durata di meno poiché era inclusa in quella dei sondaggi. L’implementazione della ricerca è stata più corta poiché già contenuta all’interno del template che ho usta per il sito.

# Conclusioni

## Sviluppi futuri

Tra gli sviluppi futuri non vedo che cosa ci sia da migliorare oltre che le mancanze che ho descritto al capitolo 5.3.

## Considerazioni personali

Questo progetto mi ha permesso di mettere in pratica le nozioni che sono stati insegnati nei moduli di programmazione web, principalmente per l’uso di una architettura di codice che non ho mai usato nella pratica.

Mi ha fatto rendere conto di quanto è importante pianificare bene il progetto poiché durante l’implementazione mi sono reso conto che come ho pianificato non era abbastanza per completare quelle determinate parti.

# Bibliografia

## Sitografia

* http://stackoverflow.com/, Stack OverFlow, da 03.09 al 20.12
* <http://www.fpdf.org/>, FPDF , Novembre e Dicembre 2019
* <https://www.w3schools.com/> , Novembre e Dicembre 2019
* <https://laravel.com/>, Laravel, Ottobre 2019
* <https://www.php.net/>, PHP, da 03.09 al 20.12

# Allegati

Elenco degli allegati

* Diari di lavoro
* Quaderno dei compiti
* Prodotto
* Codice Sorgente