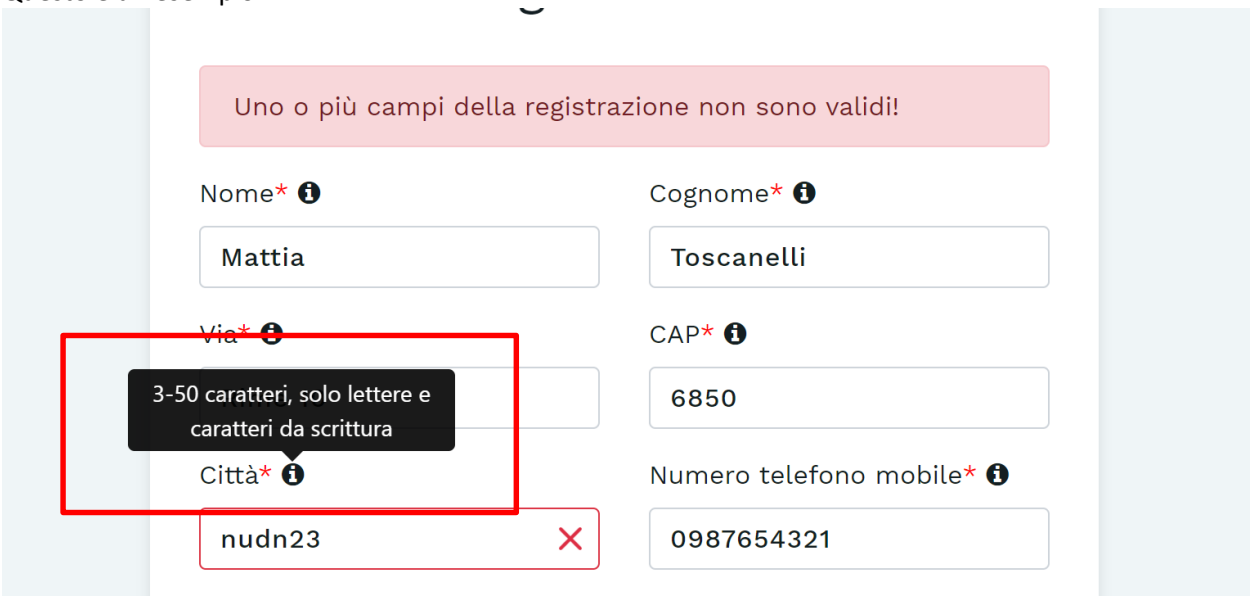
Diario di lavoro

Luogo	Scuola Arti e Mestieri Trevano
Data	10.02.2020

Lavori svolti

Durante la giornata di oggi mi sono dedicato alla validazione lato server della registrazione. Però, prima di cominciare, ho pensato che se un utente sbaglia ad inserire un campo ad esso non viene mostrato nessun messaggio di errore, ma solamente un contorno rosso sull'input errato. Perciò ho deciso si aggiungere un icona di info accanto ad ogni input per spiegare come deve essere compilato un determinato campo. Questo è un esempio:



Per mostrare il tootlip ho utilizzato il metodo di bootstrap ed è il seguente:

```
<i class="fa fa-info-circle" id="input_registration_5" data-toggle="tooltip" data-placement="top" title="3-50 caratteri, solo lettere e caratteri da scrittura"></i>
```

Una volta fatto ciò ho preso tutta la classe Validator dal mio vecchio progetto e la ho adattata a quello nuovo. In pratica ho rimosso i metodi inutili e ho aggiunto, come per la validazione lato client, il controllo del nap e della via.

```
/**
 * Metodo per verificare se il cap è valido.
 * @param $val Il cap da verificare.
 * @return false|int True se il cap è valido, False se non è valdio.
 */
function checkNap($val){
    $val = str_replace(" ", "", $val);
    $val = str_replace("-", "", $val);
    return preg_match('/^[0-9]{4,5}$/', $val);
}
```

```php
/**
 * Metodo per verificare una via.
 * @param $val La via da verificare.
 * @return bool True se la via è valida, False non è valida.
 */
function checkStreet($val){
    return (preg_match('/^[0-9a-zA-
ZàáâãäåąčćęèéêëėįìíîïłńòóôõöøùúûüųūÿýżźñçčšžÀÁÂÃÄÅĄĆČĖĘÈÉÊËÌÍÎÏĮŁŃÒÓÔÕÖØÙÚÛÜŲŪŸÝŻŹÑßÇ
ŒÆČŠŽð]{2}[0-9a-zA-
ZàáâãäåąčćęèéêëėįìíîïłńòóôõöøùúûüųūÿýżźñçčšžÀÁÂÃÄÅĄĆČĖĘÈÉÊËÌÍÎÏĮŁŃÒÓÔÕÖØÙÚÛÜŲŪŸÝŻŹÑßÇ
ŒÆČŠŽð ,.\'-]+$/', $val) && count($val)<=50);
}
```

Una volta aggiunti questi due metodi ho creato la classe RegistrazioneModel, la quale contiene un costruttore, il metodo per verificare tutti i campi e l'aggiunta di un utente al database. Il passaggio dei dati, nel form di registrazione avviene tramite un classico post con un submit. Il metodo nel controller che esegue il trasferimento dei dati è il seguente:

```php
/**
 * Metodo per inserire un nuovo utente nel database.
 */
public function insert(){
    require_once 'application/models/registrazioneModel.php';

    echo "<script>alert('bum');</script>";
    $name = $surname = $street = $nap = $city = $mNumber = $oNumber = $lNumber =
$email = $password1 = $password2 = "";
    if($_SERVER["REQUEST_METHOD"] == "POST"){
        $name = Util::test_input($_POST[POST_FIRST_NAME]);
        $surname = Util::test_input($_POST[POST_SURNAME]);
        $street = Util::test_input($_POST[POST_STREET]);
        $nap = Util::test_input($_POST[POST_NAP]);
        $city = Util::test_input($_POST[POST_CITY]);
        $mNumber = Util::test_input($_POST[POST_MOBILE_NUMBER]);
        $oNumber = Util::test_input($_POST[POST_OFFICE_NUMBER]);
        $lNumber = Util::test_input($_POST[POST_LANDLINE_NUMBER]);
        $email = Util::test_input($_POST[POST_EMAIL]);
        $password1 = Util::test_input($_POST[POST_PASSWORD]);
        $password2 = Util::test_input($_POST[POST_RE_PASSWORD]);
        $rm = new RegistrazioneModel($name, $surname, $street, $nap, $city, $mNumber,
$oNumber, $lNumber, $email, $password1, $password2);
        if($rm->insertUser()){
            //header("Location: ".URL."aspetta");
        }else{
            //header("Location: ".URL."errore");
        }
    }else{
        //header("Location: ".URL."errore");
    }
}
```

Che ha sua volta richiama questo metodo nella classe model:

```
function insertUser(){
    if($this->checkAll()) {
        $password = password_hash($this->password1, PASSWORD_DEFAULT);
        $hash = password_hash(rand(1,5000), PASSWORD_DEFAULT);
        require 'application/libs/sendMail.php';
        $body = "Benvenuto/a $this->name $this->surname,<br>
                 ti rigranzio per esserti iscritto su FastRent. Manca solo un piccolo
passaggio per incominciare ad
                 usare il sito, devi verificare la tua email!<br><br>
                 <a href='".URL."conferma/confirm/$hash/$this->email'> Per verificare
la tua mail clicca questo link!</a>";
        try{
            $s = new SendMail();
            $s->mailSend($this->email,"Verifica la tua email",$body);
            $insertUser = "INSERT INTO utente
(email,nome,cognome,via,citta,cap,tipo,password,hash_email,numero_mobile,
                           numero_fisso,numero_ufficio) VALUES ('$this-
>email','$this->name','$this->surname','$this->street',
                           '$this->city',$this->nap,0,'$password','$hash','$this-
>mNumber',";
            if($this->validator->isNullOrEmptyString($this->lNumber)){
                $insertUser .= "'',";
            }else{
                $insertUser .= "'$this->lNumber',";
            }
            if($this->validator->isNullOrEmptyString($this->oNumber)){
                $insertUser .= "'')";
            }else{
                $insertUser .= "'$this->oNumber')";
            }
            $this->util->fetchAndExecute($insertUser);
            header("Location: ".URL."emailInviata");
        } catch (Exception $e){
            header("Location: ".URL."errore");
            exit;
        }
        return true;
    }else{
        return false;
    }
}
```

In questo metodo l'utente viene aggiunto al database e viene inviata una e-mail per la conferma della registrazione. Per l'invio della e-mail, come nello scorso progetto utilizzo la classe PHPMailer **https://github.com/PHPMailer/PHPMailer**

Nel link di verifica è presente un token che mi permette di identificare univocamente il proprietario dell'account. A differenza del vecchio progetto, utilizzo il metodo di PHP Password_hash() che mi cripta la password con l'aggiunta di un salt. Per concludere la registrazione devo ancora aggiungere il controllo in ajax se la e-mail è già esistente e se la e-mail appartiene ad un sito di spam.

Problemi riscontrati e soluzioni adottate
Durante la giornata di oggi non ho avuto particolari problemi.

Punto della situazione rispetto alla pianificazione
Sono in anticipo di 4 ore rispetto alla mia pianificazione. Queste ore molto probabilmente serviranno per i successivi lavori.

Programma di massima per la prossima giornata di lavoro
Durante la prossima giornata completerò la parte di validazione lato server dei dati della registrazione.