

# Diario di lavoro

Luogo	Scuola Arti e Mestieri Trevano
Data	13.02.2020

## Lavori svolti

Durante la giornata di oggi ho concluso la validazione lato server della pagina di registrazione. In pratica ho aggiunto il controllo della e-mail (cioè verificare che la e-mail non sia già presente nel database e che non abbia come dominio un sito di email temporanee) con ajax. Il ajax è stato fatto in questo modo:

```
checkEmailDupDom(val) {
    var email = val;
    var response;
    return ($.ajax({
        url: (URL + "registrazione/checkEmail"),
        type: "POST",
        dataType: "json",
        data: {email: email},
        success: function (text) {
            response = text;
        }
    }));
}
```

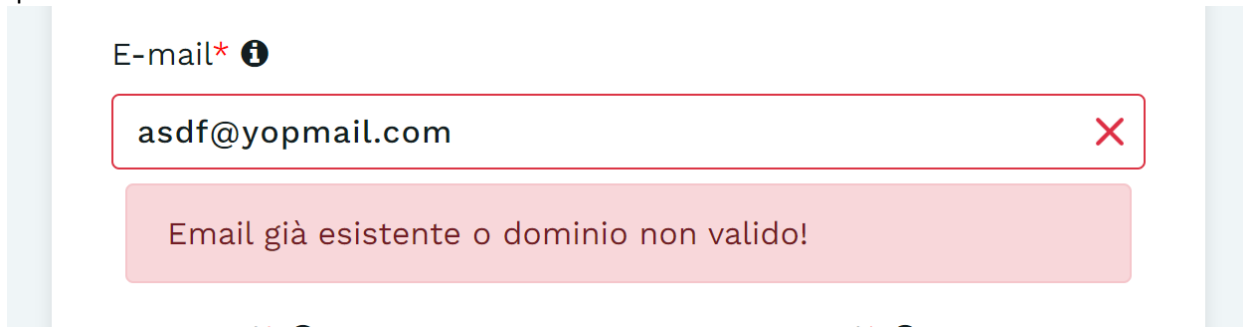
Questo metodo richiama il metodo checkEmail del controller registrazione:

```
public function checkEmail(){
    require_once 'application/models/registrazioneModel.php';
    if($_SERVER["REQUEST_METHOD"] == "POST") {
        $email = Util::test_input($_POST[EMAIL]);
        RegistrazioneModel::checkEmail($email);
    }else{
        header("Location: ".URL."errore");
    }
}
```

Che a sua volta richiama il metodo checkEmail del model RegistrazioneModel:

```
public static function checkEmail($email){
    require_once "application/libs/database.php";
    require_once "application/libs/validator.php";
    $connection = new Database();
    $validator = new Validator($connection);
    $ee = $validator->checkEmailExists($email);
    $ed = !$validator->checkEmailDomain($email);
    if($ee && $ed){
        echo 1;
    }else{
        echo 0;
    }
}
```

Questo model si preoccupa di eseguire i due controlli citati precedentemente e di scrivere un "1" se i controlli sono andati a buon fine oppure "0" se sono stati bocciati. In caso di bocciatura compare a schermo questa notifica:



Dopo aver fatto questo controllo in ajax ho fatto anche la verifica della email. In pratica nella email che invio è presente un link con all'interno un codice criptato con password\_hash e la email dell'utente. Quando l'utente clicca il link viene chiamato il metodo confirm del controller Conferma:

```
public function confirm($hash = null, $email = null){
    if(($email != null) && ($hash != null)) {
        require_once 'application/models/confermaModel.php';
        $conferma_model = new ConfermaModel();
        if($conferma_model->confirm(Util::test_input($hash),
Util::test_input($email))){
            header("Location: " . URL . "conferma");
        }else{
            header("Location: " . URL . "errore");
        }
    }else{
        header("Location: " . URL . "errore");
    }
}
```

Che si occupa di richiamare il metodo confirm del model ConfermaModel:

```
function confirm($hash, $email){
    $selectUsers = "SELECT * FROM utente WHERE email=%s AND hash_email=%s";
    $result = $this->connection->query($selectUsers, $email, $hash);
    if($result != null) {
        $updateUsers = "UPDATE utente SET hash_email='0', tipo=1 WHERE email=%s";
        $this->connection->query($updateUsers, $email);
        return true;
    }else{
        return false;
    }
}
```

Questo metodo controlla che la hash e la email del link inviato siano corrispondenti ai dati inseriti nel database. Se la conferma va a buon fine viene resettata la hash per sicurezza.

**Problemi riscontrati e soluzioni adottate**

Durante la giornata di oggi non ho avuto particolari problemi.

**Punto della situazione rispetto alla pianificazione**

Sono in linea con la pianificazione.

**Programma di massima per la prossima giornata di lavoro**

Durante la prossima giornata completerò la parte di recupero della password.