

Diario di lavoro

Luogo	Canobbio
Data	29.11.2019

Lavori svolti

Dirante la giornata di oggi ho svolto un colloquio con il docente formatore. I dettagli sono disponibili nella sezione Colloqui su GitHub.

Ho integrato PHPMailer nel progetto. Quando un utente si registra al sito riceve una mail con un link tramite il quale poter attivare l'account. Ho scaricato i file necessari di PHPMailer dalla repo su GitHub e li ho caricati nel progetto nella cartella Libs/PHPMailer. Ho aggiunto nel file di config la directory per eseguire l'autoload dei file.

```
$autoload_directories = array(
    "application/controllers/",
    "application/libs/",
    "application/libs/phpmailer/",
    "application/models/"
);
```

Ho creato la classe model MailModel per la gestione delle mail. Ha una proprietà mail privata che conterrà la mail da inviare.

```
public static function build()
{
    self::$mail = new PHPMailer(true);
    self::$mail->isSMTP();
    self::$mail->Host = 'smtp.gmail.com';
    self::$mail->Port = 587;
    self::$mail->SMTPAuth = true;
    self::$mail->Username = 'gestione parcheggi.samt@gmail.com';
    self::$mail->Password = 'Parcheggi_Admin_2019';
    self::$mail->setFrom('gestione parcheggi.samt@gmail.com', 'Gestione parcheggi');
}
```

Il metodo build() crea una nuova mail con le informazioni di base richieste per ogni mail.

```
public static function newUserMail($newUserMail, $nome, $cognome){
    self::build();
    self::$mail->addAddress($newUserMail, $nome." ".$cognome);
    self::$mail->Subject = "Registrazione effettuata";
    self::$mail->Body = "Il tuo account è stato creato con successo!<br><br>
        Per poter essere in grado di usare il tuo account clicca sul
        seguente link per attivarlo:<br>
        https://samtinfo.ch/gestione parcheggi2019/register/verify/?mail=".$newUserMail."&nome=
        ".$nome.
        "&cognome=".$cognome;
    self::$mail->send();
}
```

Il metodo newUserMail(\$newUserMail, \$nome, \$cognome) serve per formattare la mail da inviare quando un utente si registra al sito e deve attivare l'account. Riceve come parametri la mail, il nome e il cognome dell'utente. Ho lasciato il nome e il cognome per eseguire il testing della classe, ma durante le prossime ore aggiungerò al campo mail del database

l'attributo UNIQUE per poter passare al metodo solo la mail dell'utente.

Nel controller Registration ho aggiunto il metodo verify() che richiama il metodo per la verifica creato nel model RegistrationModel.

```
/**
 * Funzione che richiama il metodo per la verifica dell'utente.
 */
public function verify()
{
    if(isset($_GET['mail']) && isset($_GET['nome']) && !empty($_GET['cognome'])){

        $mail = Validator::testInput($_GET['mail']);
        $nome = Validator::validateCharAndSpace($_GET['nome']);
        $cognome = Validator::validateCharAndSpace($_GET['cognome']);

        RegisterModel::verify($mail, $nome, $cognome);
    }else{

        ViewLoader::load('home/index', array('activationNO'=>"Usa il link che ti è stato inviato per mail!"));
    }
}
```

Nel controller prendo i dati inviati nell'header tramite una richiesta get e li passo alla funzione verify() nel model.

```
public static function verify($mail, $nome, $cognome)
{
    self::$statement = Database::get()->prepare("select mail, nome, cognome
        from utente where mail=:mail and nome=:nome and cognome=:cognome;
    ");

    self::$statement->bindParam(':mail', $mail, PDO::PARAM_STR);
    self::$statement->bindParam(':nome', $nome, PDO::PARAM_STR);
    self::$statement->bindParam(':cognome', $cognome, PDO::PARAM_STR);

    self::$statement->execute();
    $result = self::$statement->fetch(PDO::FETCH_ASSOC);

    if($result > 0){
        self::$statement = Database::get()->prepare("update utente set attivo=true
        where mail=:mail and nome=:nome and cognome=:cognome and
attivo=false
    ");

        ViewLoader::load('home/index', array('activationOK'=>"Il tuo account è stato attivato con successo!"));
    }else{
        ViewLoader::load('home/index', array('activationNO'=>"L'URL è invalido o il tuo account è già attivo!"));
    }
}
```

In questa funzione preparo la query che prende i valori dal database per confrontarli con quelli passati dal browser. Se c'è una corrispondenza aggiorno lo stato della proprietà attivo a true e carico la view home con un messaggio di successo. Se invece non c'è una corrispondenza significa che l'url non è valido o che l'utente è già stato attivato.

Problemi riscontrati e soluzioni adottate

L'invio della mail funziona, ma il link da premere per l'attivazione dell'account non va. Devo ancora capire come mai.

Punto della situazione rispetto alla pianificazione

In ritardo.

Programma di massima per la prossima giornata di lavoro

Risolvere il problema con il link di attivazione, integrare l'invio della mail quando si effettua una prenotazione e svolgere i compiti discussi durante il colloquio.