WelTec
Te Whare Wānanga o te Awakairangi

CASUAL JOBS DATABASE V.1

Admin Manual

To:

Clement Sudhakar Swarnappa
Senior Academic Staff Member

School of Information Technology
DDI: +64 4 830 3130
Email: clement.sudhakar@weltec.ac.nz
URL: www.weltec.ac.nz
Post: Private Bag 39803, Lower Hutt 5045
Wellington, New Zealand

Project Team:

Supun Chandra Malimage – Project Manager

Kalaivani Ilango – Team Member

Ruban Ravi Sebastian – Team Member

SKR
Soar with pride

SKR TEAM, WELLINGTON INSTITUTE OF TECHNOLOGY
21, Kensington Avenue, Petone, Lower Hutt 5012

Table of Contents

Admin Features

- Setting Web Server
- IP and Port Configuration
- Install Apache
- Install PHP
- Install MariaDB
- Setting up application files
- Install SSL Certificate
- Transfer files using FTP

Database Configuration

This system developed using MariaDB back-end database, connection strings using PHP Data Object (PDO). Following code describes how this system connects with the database using PDO connection.

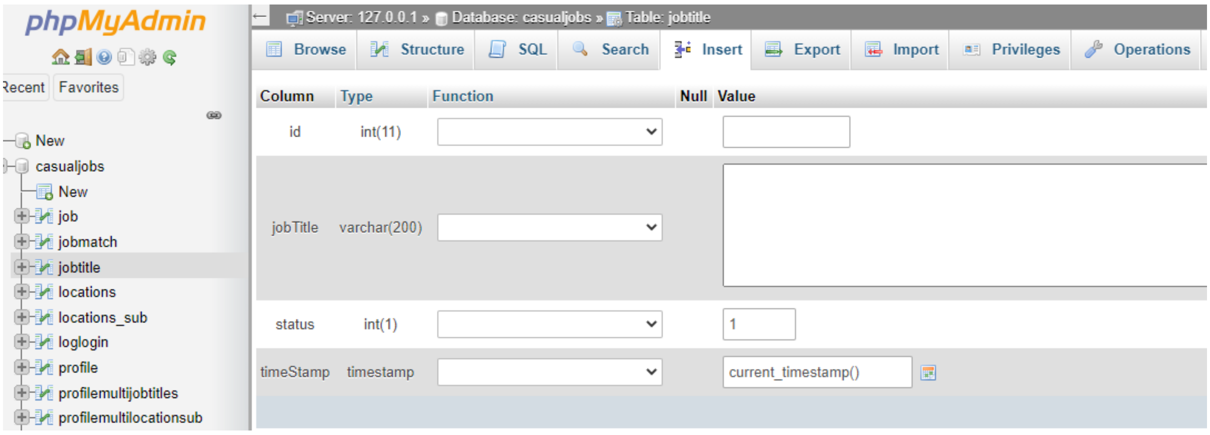
www/php/db.php

```
<?php
//declare(strict_types=1);
//error_reporting(-1); // maximum errors
error_reporting(E_ALL);
ini_set('display_errors', 'On');
ini_set('display_errors', '1'); // show on screen
ini_set('max_execution_time', 30000);


if (session_status() == PHP_SESSION_NONE) {

   session_set_cookie_params(28800);
   session_start();
   //ini_set('session.gc_maxlifetime', 288000);
}
//ini_set('max_execution_time', 12000);
date_default_timezone_set('NZ');


$DateNow = date("Y-m-d H:i:s");
$dbname = 'casualjobs';
$user = 'root';
$password = '********';

try{
$con = new PDO("mysql:host=localhost; dbname=".$dbname."; charset=utf8",
$user, $password);
$con->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);


}catch(PDOException $e){
echo $e->getMessage();
die();
}

?>
```

Job Details

Job Title

To input details of job title insert the required data into the job title database. Select the data type and length of the datatype. Then insert the job title based upon the requirements of the job.

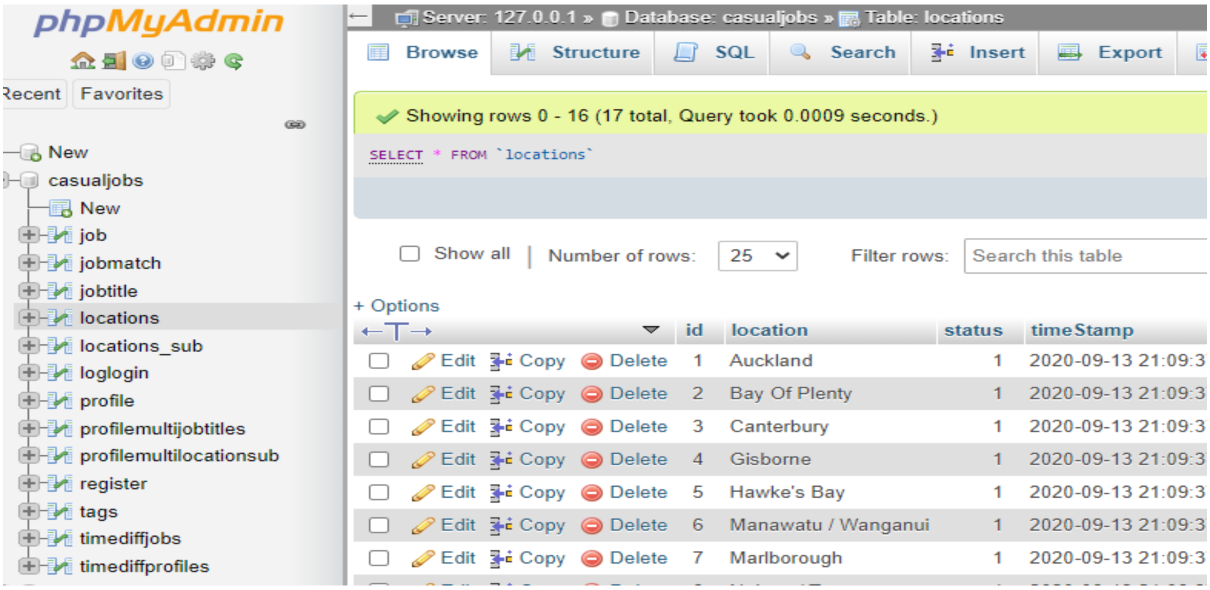


Server: 127.0.0.1 » Database: casualjobs » Table: jobtitle

Column	Type	Function	Null	Value
id	int(11)			
jobTitle	varchar(200)			
status	int(1)			1
timeStamp	timestamp			current_timestamp()

Location

To input the details of the Location, enter locations based upon the requirement of the job posted in the database. The locations can be edited, copied or deleted by using **Edit, Copy or Delete** options.



Server: 127.0.0.1 » Database: casualjobs » Table: locations

✓ Showing rows 0 - 16 (17 total, Query took 0.0009 seconds.)

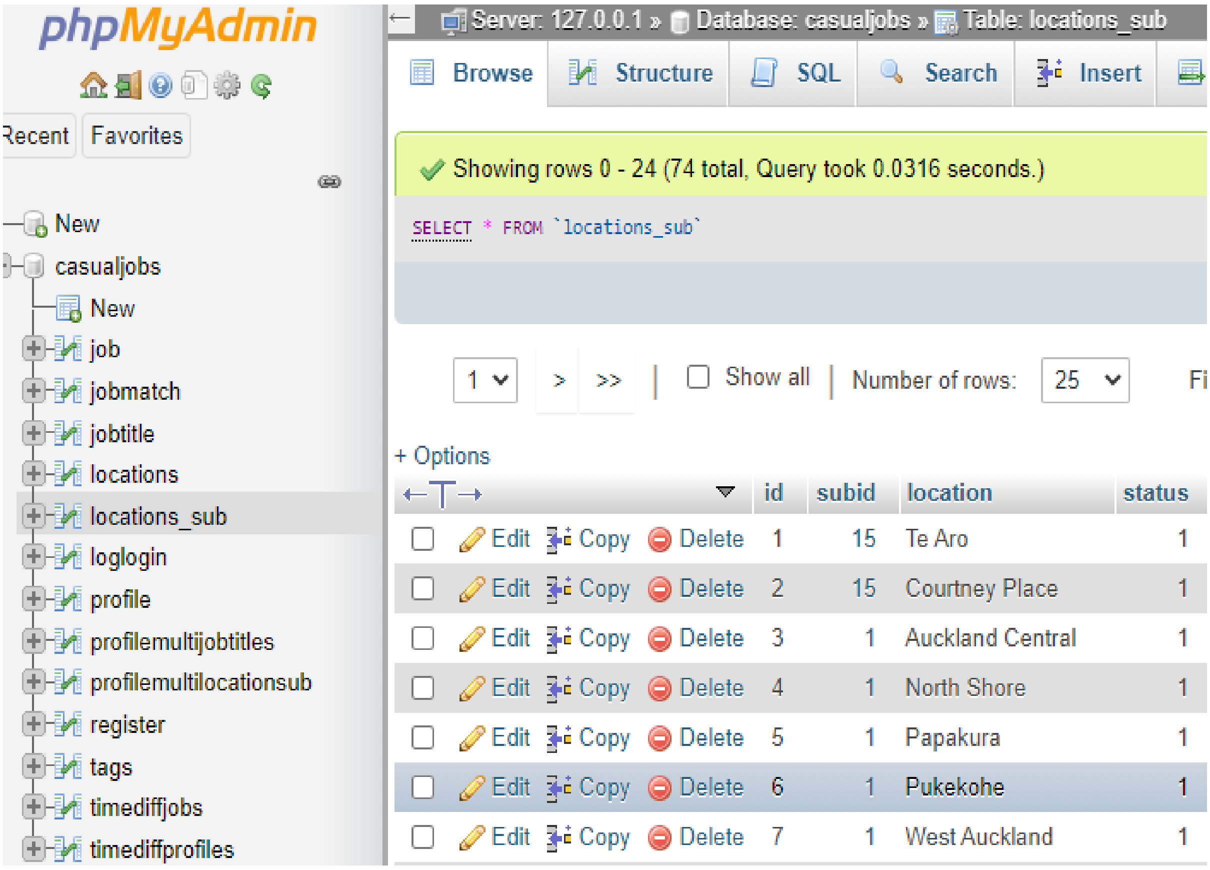
```
SELECT * FROM `locations`
```

□ Show all | Number of rows: 25 Filter rows: Search this table

				id	location	status	timeStamp
□	Edit	Copy	Delete	1	Auckland	1	2020-09-13 21:09:3
□	Edit	Copy	Delete	2	Bay Of Plenty	1	2020-09-13 21:09:3
□	Edit	Copy	Delete	3	Canterbury	1	2020-09-13 21:09:3
□	Edit	Copy	Delete	4	Gisborne	1	2020-09-13 21:09:3
□	Edit	Copy	Delete	5	Hawke's Bay	1	2020-09-13 21:09:3
□	Edit	Copy	Delete	6	Manawatu / Wanganui	1	2020-09-13 21:09:3
□	Edit	Copy	Delete	7	Marlborough	1	2020-09-13 21:09:3

Sub Location

To input the details of the Sub Location, enter sub locations based upon the requirement of the locations posted in the database. The sub locations can be edited, copied or deleted by using **Edit or Delete** options.



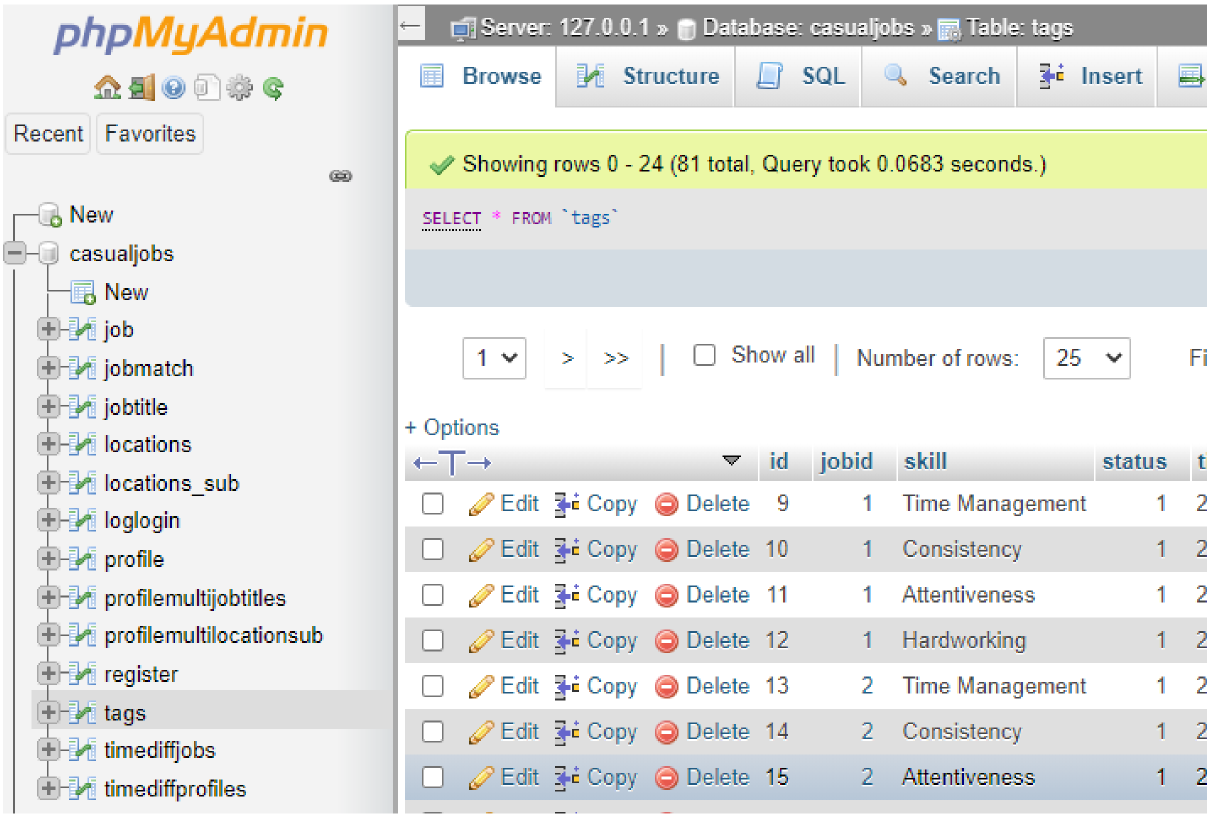
✓ Showing rows 0 - 24 (74 total, Query took 0.0316 seconds.)

```
SELECT * FROM `locations_sub`
```

				id	subid	location	status
□	Edit	Copy	Delete	1	15	Te Aro	1
□	Edit	Copy	Delete	2	15	Courtney Place	1
□	Edit	Copy	Delete	3	1	Auckland Central	1
□	Edit	Copy	Delete	4	1	North Shore	1
□	Edit	Copy	Delete	5	1	Papakura	1
□	Edit	Copy	Delete	6	1	Pukekohe	1
□	Edit	Copy	Delete	7	1	West Auckland	1

Skills

To input the details of the skills, enter skills depends upon the nature of the job title in the database. The skills can be edited, copied, or deleted by using **Edit or Delete** options.



✓ Showing rows 0 - 24 (81 total, Query took 0.0683 seconds.)

```
SELECT * FROM `tags`
```

		id	jobid	skill	status	t
□	Edit Copy Delete	9	1	Time Management	1	2
□	Edit Copy Delete	10	1	Consistency	1	2
□	Edit Copy Delete	11	1	Attentiveness	1	2
□	Edit Copy Delete	12	1	Hardworking	1	2
□	Edit Copy Delete	13	2	Time Management	1	2
□	Edit Copy Delete	14	2	Consistency	1	2
□	Edit Copy Delete	15	2	Attentiveness	1	2

Registration

Registration is a first page where users' will enter their details to start using the application and those details will be used to login into the application going forward. Registration details will be stored in the database at the backend, which would automatically detect the users whether they are an employer or job seeker when they login.

www/register > data.php

```php
<?php

include ('../php/db.php');

if(isset($_POST['email']) && isset($_POST['password'] ))
{

    $user_type = $_POST['userType'];
    $email = $_POST['email'];
    $password = md5($_POST['password']);
    $empName = $_POST['empName'];
    $address_line_1 = $_POST['address_line_1'];
    $address_line_2 = $_POST['address_line_2'];
    $suburb = $_POST['suburb'];
    $city = $_POST['city'];
    $postcode = $_POST['postcode'];
    $emp_tel = $_POST['emp_tel'];
    $imagePath = $_POST['actual_image_name'];
    if($imagePath == ''){
        $img = '0';
    }else{
        $img = '1';
    }
    $sql = "SELECT * FROM `register` where BINARY `email`= ?";
    $smt = $con->prepare($sql);
    $smt->execute(array($email));
    $row = $smt->fetch(PDO::FETCH_OBJ);
    $num_rows = $smt->rowCount();

    if($num_rows  != 0) {

        echo "exist";
        return false;
    }
    else {
    $sql = "INSERT INTO register (user_type, email, password, userName,
address1, address2, suburb, city, postcode, telephone, image, status)
VALUES (?,?,?,?,?,?,?,?,?,?,?,?)";
    $smt = $con->prepare($sql);
    $result = $smt->execute(array($user_type, $email, $password, $empName,
$address_line_1, $address_line_2, $suburb, $city, $postcode, $emp_tel,
$img, '1'));
    if($result == 1) {
        if($imagePath == ''){
            echo "ok";
            return false;
        }
        $last_id = $con->lastInsertId();
        $new_path = '';
        if($user_type == '1'){
            $new_path = "../employer/images/".$last_id.'.jpg';
        }else{
            $new_path = "../seeker/images/".$last_id.'.jpg';
        }
        $old_path = "../css/images/temp/".$imagePath;
        rename($old_path, $new_path);
        echo "ok";
        return false;
    }
    echo "error";
    return false;
    }
}
?>
```

Edit/Delete User Information

It is an important feature in all the application which consists users. Admin has the access to edit/delete users' profiles which falls under performance or seems inappropriate. Admin holds complete authority and rights to terminate any users without prior notice based on their performance.

www/editprofile > data.php

```php
<?php

include ('../php/db.php');

if(isset($_POST['email']) && isset($_POST['password'] ))
{

    $user_type = $_POST['userType'];
    $email = $_POST['email'];
    $password = $_POST['password'];
    $empName = $_POST['empName'];
    $address_line_1 = $_POST['address_line_1'];
    $address_line_2 = $_POST['address_line_2'];
    $suburb = $_POST['suburb'];
    $city = $_POST['city'];
    $postcode = $_POST['postcode'];
    $emp_tel = $_POST['emp_tel'];
    $imagePath = $_POST['actual_image_name'];
    $myid = $_SESSION['id'];
    $imagechanged = $_POST['imagechanged'];

    if($imagePath == ''){
        if($imagechanged == '1'){
            $img = '0';
        }else{
            $img = '1';
        }

    }else{
        $img = '1';
    }
    $sql = "SELECT * FROM register WHERE id=?";
    $smt = $con->prepare($sql);
    $smt->execute(array($myid));
    $rowCount = $smt->rowCount();
    if($rowCount == 0){
        echo "error";
        return false;
    }

    if($password == ''){
        $sql = "UPDATE register SET user_type=?, email=?, userName=?,
address1=?, address2=?, suburb=?, city=?, postcode=?, telephone=?, image=?,
status=? WHERE id=?";
        $smt = $con->prepare($sql);
        $result = $smt->execute(array($user_type, $email, $empName,
$address_line_1, $address_line_2, $suburb, $city, $postcode, $emp_tel,
$img, '1', $myid));
    }else{
        $sql = "UPDATE register SET user_type=?, email=?, password=?,
userName=?, address1=?, address2=?, suburb=?, city=?, postcode=?,
telephone=?, image=?, status=? WHERE id=?";
        $smt = $con->prepare($sql);
        $result = $smt->execute(array($user_type, $email, $password,
$empName, $address_line_1, $address_line_2, $suburb, $city, $postcode,
$emp_tel, $img, '1', $myid));
    }

        if($imagechanged == '1'){

        if($imagePath == ''){
            if($user_type == '1'){
                if(file_exists('../employer/images/'.$myid.'.jpg')){
                    unlink('../employer/images/'.$myid.'.jpg');
                }
            }else{
                if(file_exists('../seeker/images/'.$myid.'.jpg')){
                    unlink('../seeker/images/'.$myid.'.jpg');
                }
            }

            echo "ok";
            return false;
        }
        $last_id = $myid;
        $new_path = '';
        if($user_type == '1'){
            $new_path = "../employer/images/".$last_id.'.jpg';
        }else{
            $new_path = "../seeker/images/".$last_id.'.jpg';
        }
        $old_path = "../css/images/temp/".$imagePath;
        rename($old_path, $new_path);
        echo "ok";
        return false;
        }else{
        echo "ok";
        return false;
        }
}
?>
```

Login

After registration process is done, login page allows the user to access the application and update the jobs or profile. Login page is designed in a way that it will automatically detect whether it is an employer or job seeker profile and take them to the respective dashboard. Employer and job seeker can login into the application by using right login credentials which is registered during the registration process

www/login > check.php

```
<?php

include ('../php/db.php');
include ('logLogin.php');
```

```
if(isset($_POST['email']) && isset($_POST['password'] ))
{
    $email = $_POST['email'];
    $password = $_POST['password'];

    $sql = "SELECT * FROM `register` where BINARY `email`= ? and BINARY
`password`= ? AND status='1'";
    $smt = $con->prepare($sql);
    $smt->execute(array($email,md5($password)));
    $row = $smt->fetch(PDO::FETCH_OBJ);
    $num_rows = $smt->rowCount();

    if($num_rows  != 0) {
        $_SESSION['id'] = $row->id;
        $_SESSION['email'] = $row->email;
        $_SESSION['userName'] = $row->userName;
        $_SESSION['address1'] = $row->address1;
        $_SESSION['address2'] = $row->address2;
        $_SESSION['suburb'] = $row->suburb;
        $_SESSION['city'] = $row->city;
        $_SESSION['postcode'] = $row->postcode;
        $_SESSION['telephone'] = $row->telephone;
        $_SESSION['image'] = $row->image;
        $_SESSION['type'] = $row->user_type;
         $smt = $con->prepare("INSERT INTO loglogin
(username,ipAddress,browser,os,userAgent,logType) values( ?, ?, ?, ?, ?,
?);");
        $smt->execute(array($row->userName, $ipaddress, $user_browser,
$user_os, $user_agent, 'login'));

        echo "ok";
    }
    else {
        echo "error";
    }

}else{
    echo "error";
}
?>
```

Employer Dashboard

The employer dashboard database helps the employer to view their posted jobs, how many job seeker profiles matching with the jobs posted, shortlist the job seeker, decline the job seekers etc.,

www/employer > data.php

```
<?php
include ('../php/db.php');
require_once('timeDiff.php');
require_once('qua_other.php');
if(!isset($_SESSION['id']))
{
echo "Error";
return false;
```

```
}

if(isset($_POST['Reload'])){
    $_SESSION['updateMatch'] = '1';
    LoadData($con);
}
if(isset($_POST['LoadData'])){
    $_SESSION['updateMatch'] = '0';
    LoadData($con);
}


function runMatch($con){

$empID = $_SESSION['id'];

$smtjob = $con->prepare("SELECT * FROM job WHERE status = '1' AND
userid=?");
$smtjob->execute(array($empID));

$smtprofile = $con->prepare("SELECT * FROM profile WHERE status = '1'");
$smtprofilemultiJobTitles = $con->prepare("SELECT * FROM
profilemultijobtitles WHERE profileid=? AND jobTitleid = ?");
$smtprofileMultiLocationSub = $con->prepare("SELECT * FROM
profilemultilocationsub WHERE profileid=? AND locationSubid=?");

while($rowjob=$smtjob->fetch(PDO::FETCH_OBJ)){
    $smtprofile->execute();
    while($rowprofile=$smtprofile->fetch(PDO::FETCH_OBJ)){
        if($rowprofile->multiJobTitle == '1'){ //checking seeker multiple
jobTitle selections
            $smtprofilemultiJobTitles->execute(array($rowprofile->id,
$rowjob->jobTitleid)); //check seeker jobtitles with job table title
            $rowCountMultiJobTitle = $smtprofilemultiJobTitles->rowCount();
            if($rowCountMultiJobTitle != 0){ //jobTitle found in
profilemulti table. Now continue to location match

                $locationj = $rowjob->location;
                $locationp = $rowprofile->location;
                if($locationj == $locationp){

                    $locationSub = 0;
                    if($rowprofile->multiLocation == '0'){
                        $locationSub = 100;
                    }else {
                            //Location Sub matching
                            $smtprofileMultiLocationSub-
>execute(array($rowprofile->id, $rowjob->locationSub)); //check profile sub
locations match with job sub location
                            $rowCountMultiLocationSub =
$smtprofileMultiLocationSub->rowCount();
                            if($rowCountMultiLocationSub == 0){
                                    $locationSub = 0;
                                }else {
                                    $locationSub = 100;
                            }
                    }

                    //Job Title and location Matched
                    AddToMatchedTable($rowjob, $rowprofile, $locationSub,
$con);
```

```
                }

            }

        } else if($rowprofile->multiJobTitle == '0'){//any job title
selected by seeker

                $locationj = $rowjob->location;
                $locationp = $rowprofile->location;
                if($locationj == $locationp){

                    $locationSub = 0;
                    if($rowprofile->multiLocation == '0'){
                        $locationSub = 100;
                    }else{
                        //Location Sub matching
                        $smtprofileMultiLocationSub-
>execute(array($rowprofile->id, $rowjob->locationSub)); //check profile sub
locations match with job sub location
                        $rowCountMultiLocationSub =
$smtprofileMultiLocationSub->rowCount();
                        if($rowCountMultiLocationSub == 0){
                            $locationSub = 0;
                            }else {
                                $locationSub = 100;
                        }
                    }

                    //Job Title any and location Matched
                    AddToMatchedTable($rowjob, $rowprofile, $locationSub,
$con);

                    }
            }
    }//end profiles loop

  } //end jobs loop
} //end function

function LoadData($con){

$empID = $_SESSION['id'];
runMatch($con);

$allData = new stdClass();

$smt=$con->prepare("SELECT j.*, CASE WHEN j.jobType = 1 THEN 'Casual' WHEN
j.jobType = 2 THEN 'Part-time' WHEN j.jobType = 3 THEN 'One-Off' END as
jobTypeName, l.location loc, ls.location locsub, jt.jobTitle FROM job j
LEFT JOIN locations l ON j.location = l.id LEFT JOIN locations_sub ls ON
j.locationSub = ls.id LEFT JOIN jobtitle jt ON j.jobTitleid = jt.id WHERE
userid=? AND j.status='1'");
$smt->execute(array($empID));
$row = $smt->fetchAll(PDO::FETCH_OBJ);
$allData->jobs = $row;

$smt=$con->prepare("SELECT empid, jobid, COUNT(*) as countdata FROM
jobmatch WHERE empid=? AND status ='1' GROUP BY jobid");
$smt->execute(array($empID));
$row = $smt->fetchAll(PDO::FETCH_OBJ);
$allData->match = $row;
```

```
print json_encode($allData);
}
?>
```

Add New Job

Employer can add any number of jobs based on their requirements by adding necessary details through “Add New Jobs” feature. There is no limit set on adding jobs under this category.

www/addjob > data.php

```
<?php
include ('../php/db.php');
if(!isset($_SESSION['id'])){
    echo "login";
    return false;
}
if(isset($_POST['add']))
{

AddData($con);
}


function AddData($con){


    $userid = $_SESSION['id'];
    $dateAdd = $_POST['dateAdd'];
    $jobTitle = $_POST['jobTitle'];
    $jobType = $_POST['jobtype'];
    $Location = $_POST['Location'];
    $LocationSub = $_POST['LocationSub'];
    $payRate = $_POST['payrate'];
    $jobDes = $_POST['jobdes'];
    $startDate = $_POST['dateStart'];
    $endDate = $_POST['dateEnd'];
    $startType = $_POST['onGoing'];
    $endType = $_POST['endDateCheck'];
    $fromTime = $_POST['t1'];
    $toTime = $_POST['t2'];
    $totalHours = $_POST['totalHours'];
    $diffTimes = $_POST['diffTimes'];
    $qualification = $_POST['qualification'];
    $experience = $_POST['experience'];
    $skills = $_POST['skills'];
    $visaType = $_POST['visa'];
    $license = $_POST['license'];
    $vehicle = $_POST['vehicle'];
    $ethnicity = $_POST['ethnicity'];
    $age1 = $_POST['age1'];
    $age2 = $_POST['age2'];
    $gender = $_POST['gender'];

        $mon1 = $_POST['mon1'];
        $mon2 = $_POST['mon2'];
        $tue1 = $_POST['tue1'];
        $tue2 = $_POST['tue2'];
        $wed1 = $_POST['wed1'];
        $wed2 = $_POST['wed2'];
        $thu1 = $_POST['thu1'];
        $thu2 = $_POST['thu2'];
        $fri1 = $_POST['fri1'];
        $fri2 = $_POST['fri2'];
        $sat1 = $_POST['sat1'];
        $sat2 = $_POST['sat2'];
        $sun1 = $_POST['sun1'];
        $sun2 = $_POST['sun2'];

    $smt = $con->prepare("INSERT INTO job (userid, dateAdd, jobTitleid, 
jobType, location, locationSub, payRate, jobDes, startDate, startType, 
endDate, endType, fromTime, toTime, totalHours, timeDiff, qualification, 
experience, skills, visaType, license, vehicle, ethnicity, age1, age2, 
gender, status) VALUES(?,?,?,?,?,?,?,?,?,?,
?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?)");
    $result = $smt->execute(array($userid, $dateAdd, $jobTitle, $jobType, 
$Location, $LocationSub, $payRate, $jobDes, $startDate, $startType, 
$endDate, $endType, $fromTime, $toTime, $totalHours, $diffTimes, 
$qualification, $experience, $skills, $visaType, $license, $vehicle, 
$ethnicity, $age1, $age2, $gender, '1'));
    $jobID = $con->lastInsertId();

    if($diffTimes == '1'){
        $smtTimes = $con->prepare("INSERT INTO timediffjobs (jobid, mon1, 
mon2, tue1, tue2, wed1, wed2, thu1, thu2, fri1, fri2, sat1, sat2, sun1, 
sun2) VALUES(?,?,?,?,?,?,?,?,?,?,?,?,?,?,?)");
        $resultTimes = $smtTimes->execute(array($jobID, $mon1, $mon2, 
$tue1, $tue2, $wed1, $wed2, $thu1, $thu2, $fri1, $fri2, $sat1, $sat2, 
$sun1, $sun2));
    }

    echo "ok";
    return false;
}

?>
```

Edit/Delete Job

This feature helps employer to edit their posted jobs if there are any changes and they can also delete it if the position is filled or no more available.

www/editjob > data.php

```
<?php
include ('../php/db.php');
if(!isset($_SESSION['id'])){
    echo "login";
    return false;
}
if(isset($_POST['update']))
{
```

```
AddData($con);
}


function AddData($con){


    $userid = $_SESSION['id'];
    $jobid = $_POST['jobid'];
    $dateAdd = $_POST['dateAdd'];
    $jobTitle = $_POST['jobTitle'];
    $jobType = $_POST['jobtype'];
    $Location = $_POST['Location'];
    $LocationSub = $_POST['LocationSub'];
    $payRate = $_POST['payrate'];
    $jobDes = $_POST['jobdes'];
    $startDate = $_POST['dateStart'];
    $endDate = $_POST['dateEnd'];
    $startType = $_POST['onGoing'];
    $endType = $_POST['endDateCheck'];
    $fromTime = $_POST['t1'];
    $toTime = $_POST['t2'];
    $totalHours = $_POST['totalHours'];
    $diffTimes = $_POST['diffTimes'];
    $qualification = $_POST['qualification'];
    $experience = $_POST['experience'];
    $skills = $_POST['skills'];
    $visaType = $_POST['visa'];
    $license = $_POST['license'];
    $vehicle = $_POST['vehicle'];
    $ethnicity = $_POST['ethnicity'];
    $age1 = $_POST['age1'];
    $age2 = $_POST['age2'];
    $gender = $_POST['gender'];

        $mon1 = $_POST['mon1'];
        $mon2 = $_POST['mon2'];
        $tue1 = $_POST['tue1'];
        $tue2 = $_POST['tue2'];
        $wed1 = $_POST['wed1'];
        $wed2 = $_POST['wed2'];
        $thu1 = $_POST['thu1'];
        $thu2 = $_POST['thu2'];
        $fri1 = $_POST['fri1'];
        $fri2 = $_POST['fri2'];
        $sat1 = $_POST['sat1'];
        $sat2 = $_POST['sat2'];
        $sun1 = $_POST['sun1'];
        $sun2 = $_POST['sun2'];

    $smt = $con->prepare("SELECT * FROM job WHERE id=? AND userid=? AND
status = '1'");
    $smt->execute(array($jobid, $userid));
    $rowCount = $smt->rowCount();
    if($rowCount == 0){
        echo "You do not have permission to access this page";
        return false;
    }
    $rowjob = $smt->fetch(PDO::FETCH_OBJ);
    $smt = $con->prepare("UPDATE job SET jobTitleid=?, jobType=?, 
location=?, locationSub=?, payRate=?, jobDes=?, startDate=?, startType=?, 
endDate=?, endType=?, fromTime=?, toTime=?, totalHours=?, timeDiff=?, 
qualification=?, experience=?, skills=?, visaType=?, license=?, vehicle=?, 
ethnicity=?, age1=?, age2=?, gender=?, status=? WHERE id=? AND userid=?");
    $result = $smt->execute(array($jobTitle, $jobType, $Location, 
$LocationSub, $payRate, $jobDes, $startDate, $startType, $endDate, 
$endType, $fromTime, $toTime, $totalHours, $diffTimes, $qualification, 
$experience, $skills, $visaType, $license, $vehicle, $ethnicity, $age1, 
$age2, $gender, '1', $jobid, $userid));

    if($diffTimes == '1'){
        $smt = $con->prepare("SELECT * FROM timediffjobs WHERE jobid=? AND 
status='1'");
        $smt->execute(array($jobid));
        $rowCount = $smt->rowCount();
        if($rowCount == 0){
            $smtTimes = $con->prepare("INSERT INTO timediffjobs (jobid, 
mon1, mon2, tue1, tue2, wed1, wed2, thu1, thu2, fri1, fri2, sat1, sat2, 
sun1, sun2) VALUES(?,?,?,?,?,?,?,?,?,?,?,?,?,?,?)");
            $resultTimes = $smtTimes->execute(array($jobid, $mon1, $mon2, 
$tue1, $tue2, $wed1, $wed2, $thu1, $thu2, $fri1, $fri2, $sat1, $sat2, 
$sun1, $sun2));
        }else{
            $smtTimes = $con->prepare("UPDATE timediffjobs SET mon1=?, 
mon2=?, tue1=?, tue2=?, wed1=?, wed2=?, thu1=?, thu2=?, fri1=?, fri2=?, 
sat1=?, sat2=?, sun1=?, sun2=? WHERE jobid=? AND status='1'");
            $resultTimes = $smtTimes->execute(array($mon1, $mon2, $tue1, 
$tue2, $wed1, $wed2, $thu1, $thu2, $fri1, $fri2, $sat1, $sat2, $sun1, 
$sun2, $jobid));
        }

    }else{
        $smt = $con->prepare("SELECT * FROM timediffjobs WHERE jobid=? AND 
status='1'");
        $smt->execute(array($jobid));
        $rowCount = $smt->rowCount();
        if($rowCount != 0){
            $smt = $con->prepare("DELETE FROM timediffjobs WHERE jobid=? 
AND status='1'");
            $smt->execute(array($jobid));
        }
    }

    echo "ok";
    return false;
}

?>
```

Matched Profiles

Job Matching Algorithms would also help employers to get best candidate for their posted job without spending much time on viewing and filtering applications. Filtering profile, shortlisting the right candidate by reading several resumes can take days and weeks. Employer does not want to use more resources on filtering the shortlisting the profile, wherein this job matching algorithm undoubtedly help the employer to manage their resource efficiently and quickly www/matchedjobs > getJobs.php

```
<?php

include ('../php/db.php');
include ('../php/session.php');

if(!isset($_POST['jobID'])){
    return false;
}
$empType = $_SESSION['type'];
if(!$empType == '1'){
    return false;
}
$allData = new stdClass();
$empID = $_SESSION['id'];
$jobID = $_POST['jobID'];

    $sql = "SELECT j.dateAdd, jt.jobTitle,
    CASE WHEN j.jobType=0 THEN 'ANY'
    WHEN j.jobType=1 THEN 'Casual'
    WHEN j.jobType=2 THEN 'Part-time'
    WHEN j.jobType=3 THEN 'One-Off'
    END as jobTypeName, l.location, ls.location as locationsub, j.payRate,
j.jobDes, j.startDate, j.startType, j.endDate, j.endType, j.fromTime,
j.toTime, j.totalHours, j.timeDiff,
    CASE WHEN j.qualification=0 THEN 'Not Required'
    WHEN j.qualification=1 THEN 'Ordinary level'
    WHEN j.qualification=2 THEN 'College level'
    WHEN j.qualification=3 THEN 'High-School level'
    WHEN j.qualification=4 THEN 'University level'
    END as qualification,
    CASE WHEN j.experience=0 THEN 'Any'
    WHEN j.experience=1 THEN '1-3 Months'
    WHEN j.experience=2 THEN '3-6 Months'
    WHEN j.experience=3 THEN '6-12 Months'
    WHEN j.experience=4 THEN '12+ Months'
    END as experience, j.skills,
    CASE WHEN j.visaType=0 THEN 'Any'
    WHEN j.visaType=1 THEN 'Student Work Visa'
    WHEN j.visaType=2 THEN 'General Work Visa'
    WHEN j.visaType=3 THEN 'Working Holiday Visa'
    WHEN j.visaType=4 THEN 'Other Visa Type' END as visaType,
    CASE WHEN j.license=0 THEN 'Any'
    WHEN j.license=1 THEN 'Full'
    WHEN j.license=2 THEN 'Restricted'
    WHEN j.license=3 THEN 'International' END as license,
    CASE WHEN j.vehicle=0 THEN 'Not Required'
    WHEN j.vehicle=1 THEN 'Own Vehicle' END as vehicle,
    CASE WHEN j.ethnicity=0 THEN 'Any'
    WHEN j.ethnicity=1 THEN 'European'
    WHEN j.ethnicity=2 THEN 'Māori'
    WHEN j.ethnicity=3 THEN 'Pasifika'
    WHEN j.ethnicity=4 THEN 'Asian'
    WHEN j.ethnicity=5 THEN 'MELAA (Middle Eastern/Latin American/African)'
END AS ethnicity, j.age1, j.age2,
    CASE WHEN j.gender=0 THEN 'Any'
    WHEN j.gender=1 THEN 'Male'
    WHEN j.gender=2 THEN 'Female' END as gender, j.status, t.mon1, t.mon2,
t.tue1, t.tue2, t.wed1, t.wed2, t.thu1, t.thu2, t.fri1, t.fri2, t.sat1,
t.sat2, t.sun1, t.sun2
```

```
    FROM job j
    LEFT JOIN jobtitle jt ON j.jobTitleid = jt.id
    LEFT JOIN locations l ON j.location = l.id
    LEFT JOIN locations_sub ls ON ls.id = j.locationSub
    LEFT JOIN timediffjobs t ON t.jobid = j.id WHERE j.userid = ? AND j.id
= ?";
    $smt = $con->prepare($sql);
    $smt->execute(array($empID, $jobID));
    $row = $smt->fetch(PDO::FETCH_OBJ);
    $allData->job = $row;



    $sql = "SELECT jm.id, jm.profileid as profileid, r.userName, p.dateAdd,
IFNULL(x.jobnames, 'Any') as jobnames, IFNULL(y.location, CONCAT('All ',
l.location)) AS location, lsubj.location jlocation,
    CASE WHEN p.jobType=0 THEN 'Any'
    WHEN p.jobType=1 THEN 'Casual'
    WHEN p.jobType=2 THEN 'Part-time'
    WHEN p.jobType=3 THEN 'One-Off'
    END as pjobTypeName,
    CASE WHEN j.jobType=0 THEN 'Any'
    WHEN j.jobType=1 THEN 'Casual'
    WHEN j.jobType=2 THEN 'Part-time'
    WHEN j.jobType=3 THEN 'One-Off'
    END as jjobTypeName,
    jm.jobType, jm.locationSub, jm.payRate, jm.startDate, jm.endDate,
jm.t1, jm.t2, jm.noWeek, jm.mon, jm.tue, jm.wed, jm.thu, jm.fri, jm.sat,
jm.sun, jm.qualification, jm.experience, jm.skills, jm.visa, jm.license,
jm.vehicle, jm.ethnicity, jm.age, jm.gender, jm.totalMatch, jm.short,
    j.payRate as jpayRate, j.startDate as jstartDate, j.startType as
jstartType, j.endDate as jendDate, j.endType as jendType, j.fromTime as
jfromTime, j.toTime as jtoTime, j.timeDiff as jtimeDiff,
    CASE WHEN j.qualification=0 THEN 'Not Required'
    WHEN j.qualification=1 THEN 'Ordinary level'
    WHEN j.qualification=2 THEN 'College level'
    WHEN j.qualification=3 THEN 'High-School level'
    WHEN j.qualification=4 THEN 'University level'
    END as jqualification,
    CASE WHEN j.experience=0 THEN 'Any'
    WHEN j.experience=1 THEN '1-3 Months'
    WHEN j.experience=2 THEN '3-6 Months'
    WHEN j.experience=3 THEN '6-12 Months'
    WHEN j.experience=4 THEN '12+ Months'
    END as jexperience, j.skills jskills,
    CASE WHEN j.visaType=0 THEN 'Any'
    WHEN j.visaType=1 THEN 'Student Work Visa'
    WHEN j.visaType=2 THEN 'General Work Visa'
    WHEN j.visaType=3 THEN 'Working Holiday Visa'
    WHEN j.visaType=4 THEN 'Other Visa Type' END as jvisaType,
    CASE WHEN j.license=0 THEN 'Any'
    WHEN j.license=1 THEN 'Full'
    WHEN j.license=2 THEN 'Restricted'
    WHEN j.license=3 THEN 'International' END as jlicense,
    CASE WHEN j.vehicle=0 THEN 'Not Required'
    WHEN j.vehicle=1 THEN 'Own Vehicle' END as jvehicle,
    CASE WHEN j.ethnicity=0 THEN 'Any'
    WHEN j.ethnicity=1 THEN 'European'
    WHEN j.ethnicity=2 THEN 'Māori'
    WHEN j.ethnicity=3 THEN 'Pasifika'
    WHEN j.ethnicity=4 THEN 'Asian'
    WHEN j.ethnicity=5 THEN 'MELAA (Middle Eastern/Latin American/African)'
END AS jethnicity, j.age1 jage1, j.age2 jage2,
    CASE WHEN j.gender=0 THEN 'Any'
    WHEN j.gender=1 THEN 'Male'
    WHEN j.gender=2 THEN 'Female' END as jgender, j.status as jstatus,
jt.mon1 jmon1, jt.mon2 jmon2, jt.tue1 jtue1, jt.tue2 jtue2, jt.wed1 jwed1,
jt.wed2 jwed2, jt.thu1 jthu1, jt.thu2 jthu2, jt.fri1 jfri1, jt.fri2 jfri2,
jt.sat1 jsat1, jt.sat2 jsat2, jt.sun1 jsun1, jt.sun2 jsun2,

    p.payRate as ppayRate, p.startDate as pstartDate, p.startType as
pstartType, p.endDate as pendDate, p.endType as pendType, p.fromTime as
pfromTime, p.toTime as ptoTime, p.timeDiff as ptimeDiff,
    CASE WHEN p.qualification=0 THEN 'Prefer not to say'
    WHEN p.qualification=1 THEN 'Ordinary level'
    WHEN p.qualification=2 THEN 'College level'
    WHEN p.qualification=3 THEN 'High-School level'
    WHEN p.qualification=4 THEN 'University level'
    END as pqualification,
    CASE WHEN p.experience=0 THEN 'Prefer not to say'
    WHEN p.experience=1 THEN '1-3 Months'
    WHEN p.experience=2 THEN '3-6 Months'
    WHEN p.experience=3 THEN '6-12 Months'
    WHEN p.experience=4 THEN '12+ Months'
    END as pexperience, p.skills as pskills,
    CASE WHEN p.visaType=0 THEN 'Prefer not to say'
    WHEN p.visaType=1 THEN 'Student Work Visa'
    WHEN p.visaType=2 THEN 'General Work Visa'
    WHEN p.visaType=3 THEN 'Working Holiday Visa'
    WHEN p.visaType=4 THEN 'Other Visa Type' END as pvisaType,
    CASE WHEN p.license=0 THEN 'Prefer not to say'
    WHEN p.license=1 THEN 'Full'
    WHEN p.license=2 THEN 'Restricted'
    WHEN p.license=3 THEN 'International' END as plicense,
    CASE WHEN p.vehicle=0 THEN 'No Vehicle'
    WHEN p.vehicle=1 THEN 'Own Vehicle' END as pvehicle,
    CASE WHEN p.ethnicity=0 THEN 'Prefer not to say'
    WHEN p.ethnicity=1 THEN 'European'
    WHEN p.ethnicity=2 THEN 'Māori'
    WHEN p.ethnicity=3 THEN 'Pasifika'
    WHEN p.ethnicity=4 THEN 'Asian'
    WHEN p.ethnicity=5 THEN 'MELAA (Middle Eastern/Latin American/African)'
END AS pethnicity, p.age page,
    CASE WHEN p.gender=0 THEN 'Prefer not to say'
    WHEN p.gender=1 THEN 'Male'
    WHEN p.gender=2 THEN 'Female' END as pgender, p.status as pstatus,
pt.mon1 pmon1, pt.mon2 pmon2, pt.tue1 ptue1, pt.tue2 ptue2, pt.wed1 pwed1,
pt.wed2 pwed2, pt.thu1 pthu1, pt.thu2 pthu2, pt.fri1 pfri1, pt.fri2 pfri2,
pt.sat1 psat1, pt.sat2 psat2, pt.sun1 psun1, pt.sun2 psun2


    FROM jobmatch jm
    LEFT JOIN profile p ON p.id = jm.profileid
    LEFT JOIN register r ON p.userid = r.id
    LEFT JOIN (SELECT pj.profileid as profileid1,
GROUP_CONCAT(jtl.jobTitle) as jobnames FROM profilemultijobtitles pj LEFT
JOIN jobtitle jtl ON pj.jobTitleid = jtl.id GROUP BY pj.profileid) x ON
x.profileid1 = p.id
    LEFT JOIN (SELECT pl.profileid, GROUP_CONCAT(sl.location) as location
FROM
    profilemultilocationsub pl LEFT JOIN locations_sub sl ON
pl.locationSubid = sl.id GROUP BY pl.profileid) y ON y.profileid = p.id
```

```
    LEFT JOIN locations l ON p.location = l.id
    LEFT JOIN job j ON j.id=jm.jobid
    LEFT JOIN locations_sub lsubj ON j.locationSub = lsubj.id
    LEFT JOIN timediffjobs jt ON jt.jobid=jm.jobid
    LEFT JOIN timediffprofiles pt ON pt.profileid=p.id
    WHERE jm.empid=? and jm.jobid=? AND jm.status=1 ORDER BY jm.totalMatch
DESC";
    $smt = $con->prepare($sql);
    $smt->execute(array($empID, $jobID));
    $row = $smt->fetchAll(PDO::FETCH_OBJ);
    $allData->match = $row;

    print json_encode($allData);


?>
```

Also, it has shortlisted jobs alert and confirm file as below;

Shortlist

Based on the matching percentage, employer can shortlist the candidate profile by clicking the shortlist button.

www/matchedjobs > short.php

```
<?php

include ('../php/db.php');
include ('../php/session.php');

if(!isset($_POST['jobID'])){
    return false;
}
$empType = $_SESSION['type'];
if(!$empType == '1'){
    return false;
}

$jobid = (int)$_POST['jobID'];
$matchid = $_POST['matchid'];
$event = $_POST['short'];
if($event == 'add'){
    $smt = $con->prepare("SELECT * FROM jobmatch where id=? AND jobid=? AND
short=0");
    $smt->execute(array($matchid, $jobid));
    $rowCount = $smt->rowCount();
    if($rowCount == 0){
        echo "Error";
        return false;
    }

    $smt = $con->prepare("UPDATE jobmatch SET short=1 WHERE jobid=? AND
id=?");
```

```
    $smt->execute(array($jobid,$matchid));
    echo "ok";
    return false;
}else if($event == 'remove'){
    $smt = $con->prepare("SELECT * FROM jobmatch where id=? AND jobid=? AND
short=1 OR (short=2)");
    $smt->execute(array($matchid, $jobid));
    $rowCount = $smt->rowCount();
    if($rowCount == 0){
        echo "Error";
        return false;
    }

    $smt = $con->prepare("UPDATE jobmatch SET short=0 WHERE jobid=? AND
id=?");
    $smt->execute(array($jobid,$matchid));
    echo "ok";
    return false;
}else{
    echo "error";
    return false;
}


?>
```

Job Accept

Employer can accept the matched job by pressing the accept button.

www/matchedprofiles > short.php

```
<?php

include ('../php/db.php');
include ('../php/session.php');

if(!isset($_POST['profileID'])){
    return false;
}
$empType = $_SESSION['type'];
if(!$empType == '1'){
    return false;
}

$jobid = $_POST['profileID'];


$matchid = $_POST['matchid'];
$event = $_POST['short'];
if($event == 'add'){
    $smt = $con->prepare("SELECT * FROM jobmatch where id=? AND profileid=?
AND short=1");
    $smt->execute(array($matchid, $jobid));
    $rowCount = $smt->rowCount();
    if($rowCount == 0){
        echo "Error";
        return false;
    }

    $smt = $con->prepare("UPDATE jobmatch SET short=2 WHERE profileid=? AND
id=?");
    $smt->execute(array($jobid,$matchid));
    echo "ok";
    return false;
}else if($event == 'remove'){
    $smt = $con->prepare("SELECT * FROM jobmatch where id=? AND profileid=?
AND short=2");
    $smt->execute(array($matchid, $jobid));
    $rowCount = $smt->rowCount();
    if($rowCount == 0){
        echo "Error";
        return false;
    }

    $smt = $con->prepare("UPDATE jobmatch SET short=1 WHERE profileid=? AND
id=?");
    $smt->execute(array($jobid,$matchid));
    echo "ok";
    return false;
}else{
    echo "error";
    return false;
}


?>
```

Job Seeker Dashboard

The job seeker dashboard database helps the job seekers to view their added profiles, how many jobs matching with profiles, shortlisted by the employer, accept the shortlisted jobs, decline the shortlisted jobs etc.,

www/seeker > data.php

```
<?php
include ('../php/db.php');

if(!isset($_SESSION['id']))
{
echo "Error";
}

$empID = $_SESSION['id'];
$allData = new stdClass();

$sql ="SELECT p.*, IFNULL(x.jobnames, 'Any') AS jobnames,
IFNULL(y.location, CONCAT('All ', l.location)) AS location,
CASE WHEN p.jobType=0 THEN 'Any' WHEN p.jobType=1 THEN 'Casual' WHEN
p.jobType=2 THEN 'Part-time' WHEN p.jobType=3 THEN 'One-Off' END as
jobTypeName
FROM profile p
LEFT JOIN (SELECT pj.profileid, GROUP_CONCAT(jt.jobTitle) as jobnames FROM
profilemultijobtitles pj LEFT JOIN jobtitle jt ON pj.jobTitleid = jt.id
GROUP BY pj.profileid) x ON x.profileid = p.id
LEFT JOIN (SELECT pl.profileid, GROUP_CONCAT(sl.location) as location FROM
    profilemultilocationsub pl LEFT JOIN locations_sub sl ON
pl.locationSubid = sl.id GROUP BY pl.profileid) y ON y.profileid = p.id
LEFT JOIN locations l ON p.location = l.id
WHERE p.userid=? AND p.status='1'";

$smt=$con->prepare($sql);
$smt->execute(array($empID));
$rowProfile = $smt->fetchAll(PDO::FETCH_OBJ);

$allData->profiles = $rowProfile;


$smt=$con->prepare("SELECT empid, jobid, profileid,  COUNT(*) as countdata
FROM jobmatch WHERE seekerid=? AND status ='1' and short='1' GROUP BY
profileid");
$smt->execute(array($empID));
$row = $smt->fetchAll(PDO::FETCH_OBJ);
$allData->match = $row;


print json_encode($allData);
?>
```

Add New Profile

Job seeker can add any number of profiles based on their experience and interests by adding necessary details through “Add New profile” feature. There is no limit set on adding profiles under this category.

www/addprofile > data.php

```
<?php
include ('../php/db.php');

if(isset($_POST['add']))
{

AddData($con);
}


function AddData($con){


    $userid = $_SESSION['id'];
    $dateAdd = $_POST['dateAdd'];
    $jobTitle = $_POST['jobTitle'];
    $jobType = $_POST['jobtype'];
    $Location = $_POST['Location'];
    $LocationSub = $_POST['LocationSub'];
    $payRate = $_POST['payrate'];
    $startDate = $_POST['dateStart'];
    $endDate = $_POST['dateEnd'];
    $startType = $_POST['onGoing'];
    $endType = $_POST['endDateCheck'];
    $fromTime = $_POST['t1'];
    $toTime = $_POST['t2'];
    $totalHours = $_POST['totalHours'];
    $diffTimes = $_POST['diffTimes'];
    $qualification = $_POST['qualification'];
    $experience = $_POST['experience'];
    $skills = $_POST['skills'];
    $visaType = $_POST['visa'];
    $license = $_POST['license'];
    $vehicle = $_POST['vehicle'];
    $ethnicity = $_POST['ethnicity'];
    $age1 = $_POST['age1'];
    $gender = $_POST['gender'];

        $mon1 = $_POST['mon1'];
        $mon2 = $_POST['mon2'];
        $tue1 = $_POST['tue1'];
        $tue2 = $_POST['tue2'];
        $wed1 = $_POST['wed1'];
        $wed2 = $_POST['wed2'];
        $thu1 = $_POST['thu1'];
        $thu2 = $_POST['thu2'];
        $fri1 = $_POST['fri1'];
        $fri2 = $_POST['fri2'];
        $sat1 = $_POST['sat1'];
        $sat2 = $_POST['sat2'];
        $sun1 = $_POST['sun1'];
        $sun2 = $_POST['sun2'];

        $anyJobTitle = '1';
        $anySubLocation = '1';

    $jobTitleFilter = explode(',', $jobTitle);
    if(in_array(0, $jobTitleFilter)){
        $anyJobTitle = '0';
    }
    $LocationSubFilter = explode(',', $LocationSub);
    if(in_array(0, $LocationSubFilter)){
        $anySubLocation = '0';
    }


    $smt = $con->prepare("INSERT INTO profile (userid, multiJobTitle,
dateAdd, jobType, location, multiLocation, payRate, startDate, startType,
endDate, endType, fromTime, toTime, totalHours, timeDiff, qualification,
experience, skills, visaType, license, vehicle, ethnicity, age, gender,
status) VALUES(?,?,?,?,?,?,?,?, ?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?)");
    $result = $smt->execute(array($userid, $anyJobTitle, $dateAdd,
$jobType, $Location, $anySubLocation, $payRate, $startDate, $startType,
$endDate, $endType, $fromTime, $toTime, $totalHours, $diffTimes,
$qualification, $experience, $skills, $visaType, $license, $vehicle,
$ethnicity, $age1, $gender, '1'));

    $profileId = $con->lastInsertId();
    if($diffTimes == '1'){
        $smtTimes = $con->prepare("INSERT INTO timediffprofiles (profileid,
mon1, mon2, tue1, tue2, wed1, wed2, thu1, thu2, fri1, fri2, sat1, sat2,
sun1, sun2) VALUES(?,?,?,?,?,?,?,?,?,?,?,?,?,?,?)");
        $resultTimes = $smtTimes->execute(array($profileId, $mon1, $mon2,
$tue1, $tue2, $wed1, $wed2, $thu1, $thu2, $fri1, $fri2, $sat1, $sat2,
$sun1, $sun2));
    }

    if($anyJobTitle !='0'){
        foreach($jobTitleFilter as $jtd){
            $smt = $con->prepare("INSERT INTO profilemultijobtitles
(profileid, jobTitleid) VALUES(?,?)");
            $result = $smt->execute(array($profileId, $jtd));
        }

    }
    if($anySubLocation !='0'){
        foreach($LocationSubFilter as $jtd){
            $smt = $con->prepare("INSERT INTO profilemultilocationsub
(profileid, locationSubid) VALUES(?,?)");
            $result = $smt->execute(array($profileId, $jtd));
        }

    }
    echo "ok";
    return false;
}

?>
```

Edit/Delete Profile

This feature helps job seekers to edit their posted profiles if there are any changes and they can also delete it if the profile is no more needed.

www/editjobprofile > data.php

```
<?php
include ('../php/db.php');

if(isset($_POST['update']))
{

UpdateData($con);
}


function UpdateData($con){



    $userid = $_SESSION['id'];
    $jobid = $_POST['jobid'];
    $dateAdd = $_POST['dateAdd'];
    $jobTitle = $_POST['jobTitle'];
    $jobType = $_POST['jobtype'];
    $Location = $_POST['Location'];
    $LocationSub = $_POST['LocationSub'];
    $payRate = $_POST['payrate'];
    $startDate = $_POST['dateStart'];
    $endDate = $_POST['dateEnd'];
    $startType = $_POST['onGoing'];
    $endType = $_POST['endDateCheck'];
    $fromTime = $_POST['t1'];
    $toTime = $_POST['t2'];
    $totalHours = $_POST['totalHours'];
    $diffTimes = $_POST['diffTimes'];
    $qualification = $_POST['qualification'];
    $experience = $_POST['experience'];
    $skills = $_POST['skills'];
    $visaType = $_POST['visa'];
    $license = $_POST['license'];
    $vehicle = $_POST['vehicle'];
    $ethnicity = $_POST['ethnicity'];
    $age = $_POST['age1'];
    $gender = $_POST['gender'];

        $mon1 = $_POST['mon1'];
        $mon2 = $_POST['mon2'];
        $tue1 = $_POST['tue1'];
        $tue2 = $_POST['tue2'];
        $wed1 = $_POST['wed1'];
        $wed2 = $_POST['wed2'];
        $thu1 = $_POST['thu1'];
        $thu2 = $_POST['thu2'];
        $fri1 = $_POST['fri1'];
        $fri2 = $_POST['fri2'];
        $sat1 = $_POST['sat1'];
        $sat2 = $_POST['sat2'];
        $sun1 = $_POST['sun1'];
        $sun2 = $_POST['sun2'];

        $anyJobTitle = '1';
        $anySubLocation = '1';

    $jobTitleFilter = explode(',', $jobTitle);
    if(in_array(0, $jobTitleFilter)){
        $anyJobTitle = '0';
    }
    $LocationSubFilter = explode(',', $LocationSub);
    if(in_array(0, $LocationSubFilter)){
        $anySubLocation = '0';
    }


    $smt = $con->prepare("SELECT * FROM profile WHERE id=? AND userid=? AND
status = '1'");
    $smt->execute(array($jobid, $userid));
    $rowCount = $smt->rowCount();
    if($rowCount == 0){
        echo "You do not have permission to access this page";
        return false;
    }
    $rowjob = $smt->fetch(PDO::FETCH_OBJ);


    $smt = $con->prepare("UPDATE profile SET multiJobTitle=?, jobType=?,
location=?, multiLocation=?, payRate=?, startDate=?, startType=?,
endDate=?, endType=?, fromTime=?, toTime=?, totalHours=?, timeDiff=?,
qualification=?, experience=?, skills=?, visaType=?, license=?, vehicle=?,
ethnicity=?, age=?, gender=?, status=? WHERE id=? AND userid=?");
    $result = $smt->execute(array($anyJobTitle, $jobType, $Location,
$anySubLocation, $payRate, $startDate, $startType, $endDate, $endType,
$fromTime, $toTime, $totalHours, $diffTimes, $qualification, $experience,
$skills, $visaType, $license, $vehicle, $ethnicity, $age, $gender, '1',
$jobid, $userid));


    if($diffTimes == '1'){
        $smt = $con->prepare("SELECT * FROM timediffprofiles WHERE
profileid=? AND status='1'");
        $smt->execute(array($jobid));
        $rowCount = $smt->rowCount();
        if($rowCount == 0){
            $smtTimes = $con->prepare("INSERT INTO timediffprofiles
(profileid, mon1, mon2, tue1, tue2, wed1, wed2, thu1, thu2, fri1, fri2,
sat1, sat2, sun1, sun2) VALUES(?,?,?,?,?,?,?,?,?,?,?,?,?,?,?)");
            $resultTimes = $smtTimes->execute(array($jobid, $mon1, $mon2,
$tue1, $tue2, $wed1, $wed2, $thu1, $thu2, $fri1, $fri2, $sat1, $sat2,
$sun1, $sun2));
        }else{
            $smtTimes = $con->prepare("UPDATE timediffprofiles SET mon1=?,
mon2=?, tue1=?, tue2=?, wed1=?, wed2=?, thu1=?, thu2=?, fri1=?, fri2=?,
sat1=?, sat2=?, sun1=?, sun2=? WHERE profileid=? AND status='1'");
            $resultTimes = $smtTimes->execute(array($mon1, $mon2, $tue1,
$tue2, $wed1, $wed2, $thu1, $thu2, $fri1, $fri2, $sat1, $sat2, $sun1,
$sun2, $jobid));
        }

    }else{
        $smt = $con->prepare("SELECT * FROM timediffprofiles WHERE
profileid=? AND status='1'");
        $smt->execute(array($jobid));
        $rowCount = $smt->rowCount();
        if($rowCount != 0){
            $smt = $con->prepare("DELETE FROM timediffprofiles WHERE
profileid=? AND status='1'");
            $smt->execute(array($jobid));
        }
    }

    if($anyJobTitle !='0'){
        $smt=$con->prepare("DELETE FROM profilemultijobtitles WHERE
profileid=?");
        $smt->execute(array($jobid));

        foreach($jobTitleFilter as $jtd){
            $smt = $con->prepare("INSERT INTO profilemultijobtitles
(profileid, jobTitleid) VALUES(?,?)");
            $result = $smt->execute(array($jobid, $jtd));
        }
    }else{
        $smt=$con->prepare("DELETE FROM profilemultijobtitles WHERE
profileid=?");
```

```
        $smt->execute(array($jobid));
    }


    if($anySubLocation !='0'){
        $smt = $con->prepare("DELETE FROM profilemultilocationsub WHERE
profileid=?");
        $result = $smt->execute(array($jobid));

        foreach($LocationSubFilter as $jtd){
            $smt = $con->prepare("INSERT INTO profilemultilocationsub
(profileid, locationSubid) VALUES(?,?)");
            $result = $smt->execute(array($jobid, $jtd));
        }

    }else{
        $smt = $con->prepare("DELETE FROM profilemultilocationsub WHERE
profileid=?");
        $result = $smt->execute(array($jobid));
    }
    echo "ok";
    return false;
}

?>
```

Job Matching

Job Matching Algorithms would help job seekers get information about the applied jobs without waiting for long time. Filtering jobs and finding the ones that best fit their profile can take hours and applying to hundreds of jobs without comparing the requirements can only end in rejections and disappointment. Hence, this is very evident that the job matching algorithm supports the jobs seekers in many ways.

Below is the timetable availability matching algorithm

www/employer > timeDiff.php

```
<?php

function AddToMatchedTable($rowjob, $rowprofile, $locationSub, $con){
    $empid = $_SESSION['id'];
    $jobid = $rowjob->id;
    $profileid= $rowprofile->id;
    $seekerid = $rowprofile->userid;
    $jobTypej = $rowjob->jobType;
    $jobTypep = $rowprofile->jobType;
    if($jobTypep == '0'){
        $jobType = 100;
    }else if($jobTypej == $jobTypep){
        $jobType = 100;
    }else {
        $jobType = 0;
    }


    $payj = round($rowjob->payRate,2);
    $payp = round($rowprofile->payRate,2);
    if($payj >= $payp){
        $payRate = 100;
    }else{
        $payRate = 0;
    }

    $startDatej = new DateTime($rowjob->startDate);
    $startTypej = $rowjob->startType;
    $startDatep = new DateTime($rowprofile->startDate);
    $startTypep = $rowprofile->startType;

    $endDatej = new DateTime($rowjob->endDate);
    $endTypej = $rowjob->endType;
    $endDatep = new DateTime($rowprofile->endDate);
    $endTypep = $rowprofile->endType;

    $diff=date_diff($startDatej,$startDatep);
    $diff = (int)$diff->days;
    $startDate = 0;
    $endDate = 0;

    if($startTypej == '1' && $startTypep == '1'){
        $startDate = 100;
    }else if($startTypej == '0' && $startTypep == '1' ){
        $startDate = 100;
    }else if($startTypej == '0' && $startTypep == '0'){
        if($diff >= 0 && $diff <= 1){
            $startDate = 100;
        }else if($diff >= 2 && $diff <= 10){
            $startDate = 75;
        }else if($diff >= 11 && $diff <= 20){
            $startDate = 50;
        }else if($diff >= 21 && $diff <= 30){
            $startDate = 10;
        }else{
            $startDate = 0;
        }
    }else{
        $startDate = 100;
    }

    $diff=date_diff($endDatej,$endDatep);
    $diff = (int)$diff->days;
    if($endTypej == '1' && $endTypep == '1'){
        $endDate = 100;
    }else if($endTypej == '0' && $endTypep == '1' ){
        $endDate = 100;
    }else if($endTypej == '0' && $endTypep == '0'){
        if($diff >= 0 && $diff <= 1){
            $endDate = 100;
        }else if($diff >= 2 && $diff <= 10){
            $endDate = 75;
        }else if($diff >= 11 && $diff <= 20){
            $endDate = 50;
        }else if($diff >= 21 && $diff <= 30){
            $endDate = 10;
        }else{
            $endDate = 0;
        }
    }else{
        $endDate = 100;
    }

    $t1j = $rowjob->fromTime;
    $t2j = $rowjob->toTime;
    $tdiffj = $rowjob->timeDiff;
    $t1p = $rowprofile->fromTime;
    $t2p = $rowprofile->toTime;
    $tdiffp = $rowprofile->timeDiff;
    $week;
    $available;
    $total1 = $total2 = $total3 = $total4 = $total5 = 100;
    if($tdiffj == '1'){
        if($tdiffp == '1'){// both diff time table
            $week = timeDiffAllCalc($rowjob->id, $rowprofile->id, $con);
            $qualificationsOther = QualificationsOther($jobid, $profileid, 
$con);
            //echo "timeDiffAllCalc";
        }else{ //profile time fixed and jobs diff
            $week = timeDiffSeekerFixed($rowjob->id, $rowprofile->id, $t1p, 
$t2p, $con);
            $qualificationsOther = QualificationsOther($jobid, $profileid, 
$con);
            //echo "timeDiffSeekerFixed";
        }
    } else {
        if($tdiffp == '1'){ //jobs fixed time and profile diff timing
            $week = timeDiffEmpFixed($rowjob->id, $rowprofile->id, $t1j, 
$t2j, $con);
            $qualificationsOther = QualificationsOther($jobid, $profileid, 
$con);
            //echo "timeDiffEmpFixed";

        }else{//both fixed
            $week = timeDiffAllFixed($t1j, $t2j, $t1p, $t2p);
            $qualificationsOther = QualificationsOther($jobid, $profileid, 
$con);
            //echo "timeDiffAllFixed";
            //echo $available."<br>";
        }
    }

    $totalMatch = 0;
    $total1 = (int)$jobType + (int)$locationSub + (int)$payRate;
    $total1 = $total1 / 3;
    $total1 = number_format($total1);

    $total2 = (int)$startDate + (int)$endDate;
    $total2 = $total2 /2;
    $total2 = number_format($total2);


    if($week['diffdays'] == 0) {
        $total2days = $week['mon'] + $week['tue'] + $week['wed'] + 
$week['thu'] + $week['fri'] + $week['sat'] + $week['sun'];
        $total2days = $total2days / 7;
        $total3 = (int)$total2days;
    }else{
        $total2days = $week['t1'] + $week['t2'];
        $total2days = (int)$total2days / 2;
        $total4 = $total2days;
    }

    $total5 = $qualificationsOther['qualification'] +
$qualificationsOther['experience'] + $qualificationsOther['skills'] +
$qualificationsOther['visaType'] + $qualificationsOther['license'] +
$qualificationsOther['vehicle'] + $qualificationsOther['ethnicity'] +
$qualificationsOther['age'] + $qualificationsOther['gender'];
    $total5 = (int)$total5 / 9;

    $totalMatch = $total1 + $total2 + $total3 + $total4 + $total5;
    $tatalMatch = $totalMatch / 5;

    $smt=$con->prepare("SELECT * FROM jobmatch WHERE jobid= ? AND
profileid=?");
    $smt->execute(array($rowjob->id, $rowprofile->id));
    $rowCount = $smt->rowCount();
    if($rowCount == 0){
        $smt = $con->prepare("INSERT INTO jobmatch (empid, seekerid, jobid,
profileid, jobType, locationSub, payRate, startDate, EndDate, t1, t2,
noWeek, mon, tue, wed, thu, fri, sat, sun, qualification, experience,
skills, visa, license, vehicle, ethnicity, age, gender, totalMatch)
VALUES(?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?);");
        $smt->execute(array($empid, $seekerid, $rowjob->id, $rowprofile-
>id, $jobType, $locationSub, $payRate, $startDate, $endDate, $week['t1'],
$week['t2'], $week['diffdays'], $week['mon'], $week['tue'], $week['wed'],
$week['thu'], $week['fri'], $week['sat'], $week['sun'],
$qualificationsOther['qualification'], $qualificationsOther['experience'],
$qualificationsOther['skills'], $qualificationsOther['visaType'],
$qualificationsOther['license'], $qualificationsOther['vehicle'],
$qualificationsOther['ethnicity'], $qualificationsOther['age'],
$qualificationsOther['gender'], $tatalMatch));
    }else{
        if(isset($_SESSION['updateMatch'])){
            if($_SESSION['updateMatch'] == '1'){
                $sql = "UPDATE jobmatch SET jobType=?,locationSub=?,
payRate=?, startDate=?, EndDate=?, t1=?, t2=?, noWeek=?, mon=?, tue=?,
wed=?, thu=?, fri=?, sat=?, sun=?, qualification=?, experience=?, skills=?,
visa=?, license=?, vehicle=?, ethnicity=?, age=?, gender=?, totalMatch=?
WHERE jobid=? AND profileid=?";
                $smt = $con->prepare($sql);
                $smt->execute(array($jobType, $locationSub, $payRate,
$startDate, $endDate, $week['t1'], $week['t2'], $week['diffdays'],
$week['mon'], $week['tue'], $week['wed'], $week['thu'], $week['fri'],
$week['sat'], $week['sun'], $qualificationsOther['qualification'],
$qualificationsOther['experience'], $qualificationsOther['skills'],
$qualificationsOther['visaType'], $qualificationsOther['license'],
$qualificationsOther['vehicle'], $qualificationsOther['ethnicity'],
$qualificationsOther['age'], $qualificationsOther['gender'], $tatalMatch,
$rowjob->id, $rowprofile->id));

            }

        }
    }
```

```
} //end function


function timeDiffAllCalc($jobid, $profileid, $con){
    $smtj = $con->prepare("SELECT * FROM timediffjobs WHERE jobid=?");
    $smtj->execute(array($jobid));
    $rowj =$smtj->fetch(PDO::FETCH_OBJ);

    $smtp = $con->prepare("SELECT * FROM timediffprofiles WHERE
profileid=?");
    $smtp->execute(array($profileid));
    $rowp =$smtp->fetch(PDO::FETCH_OBJ);

    $mon = $tue = $wed = $thu = $fri = $sat = $sun = $t1 = $t2 =  $diffdays
= 0;

    $monj1 = strtotime($rowj->mon1);
    $monj2 = strtotime($rowj->mon2);
    $tuej1 = strtotime($rowj->tue1);
    $tuej2 = strtotime($rowj->tue2);
    $wedj1 = strtotime($rowj->wed1);
    $wedj2 = strtotime($rowj->wed2);
    $thuj1 = strtotime($rowj->thu1);
    $thuj2 = strtotime($rowj->thu2);
    $frij1 = strtotime($rowj->fri1);
    $frij2 = strtotime($rowj->fri2);
    $satj1 = strtotime($rowj->sat1);
    $satj2 = strtotime($rowj->sat2);
    $sunj1 = strtotime($rowj->sun1);
    $sunj2 = strtotime($rowj->sun2);

    $monp1 = strtotime($rowp->mon1);
    $monp2 = strtotime($rowp->mon2);
    $tuep1 = strtotime($rowp->tue1);
    $tuep2 = strtotime($rowp->tue2);
    $wedp1 = strtotime($rowp->wed1);
    $wedp2 = strtotime($rowp->wed2);
    $thup1 = strtotime($rowp->thu1);
    $thup2 = strtotime($rowp->thu2);
    $frip1 = strtotime($rowp->fri1);
    $frip2 = strtotime($rowp->fri2);
    $satp1 = strtotime($rowp->sat1);
    $satp2 = strtotime($rowp->sat2);
    $sunp1 = strtotime($rowp->sun1);
    $sunp2 = strtotime($rowp->sun2);

    if(abs($monj1-$monj2) == 0){ //full match if no time selected
        $mon = 100;
    }else{
        if(abs($monp1-$monp2) == 0){
            $mon =0;
        }else{
            if($monj1 >= $monp1 && $monj2 <= $monp2 ){ //within the job
timing
                $mon = 100;
            }else if($monj1 >= $monp1 && $monj2 >= $monp2){ //from match
and to not match
                $mon = 50;
            }else if($monj1 <= $monp1 && $monj2 <= $monp2 ){ // from not
match and to match
                $mon = 50;
            }else{
                $mon = 0;
            }
        }
    }

    if(abs($tuej1-$tuej2) == 0){
        $tue = 100;
    }else{
        if(abs($tuep1-$tuep2) == 0){
            $tue =0;
        }else{
            if($tuej1 >= $tuep1 && $tuej2 <= $tuep2 ){ //within the job
timing
                $tue = 100;
            }else if($tuej1 >= $tuep1 && $tuej2 >= $tuep2){ //from match
and to not match
                $tue = 50;
            }else if($tuej1 <= $tuep1 && $tuej2 <= $tuep2 ){ // from not
match and to match
                $tue = 50;
            }else{
                $tue = 0;
            }
        }
    }

    if(abs($wedj1-$wedj2) == 0){
        $wed = 100;
    }else{
        if(abs($wedp1-$wedp2) == 0){
            $wed =0;
        }else{
            if($wedj1 >= $wedp1 && $wedj2 <= $wedp2 ){ //within the job
timing
                $wed = 100;
            }else if($wedj1 >= $wedp1 && $wedj2 >= $wedp2){ //from match
and to not match
                $wed = 50;
            }else if($wedj1 <= $wedp1 && $wedj2 <= $wedp2 ){ // from not
match and to match
                $wed = 50;
            }else{
                $wed = 0;
            }
        }
    }

    if(abs($thuj1-$thuj2) == 0){
        $thu = 100;
    }else{
        if(abs($thup1-$thup2) == 0){
            $thu =0;
        }else{
            if($thuj1 >= $thup1 && $thuj2 <= $thup2 ){ //within the job
timing
                $thu = 100;
            }else if($thuj1 >= $thup1 && $thuj2 >= $thup2){ //from match
and to not match
                $thu = 50;
            }else if($thuj1 <= $thup1 && $thuj2 <= $thup2 ){ // from not
match and to match
                $thu = 50;
            }else{
                $thu = 0;
            }
        }
    }

    if(abs($frij1-$frij2) == 0){
        $fri = 100;
    }else{
        if(abs($frip1-$frip2) == 0){
            $fri =0;
        }else{
            if($frij1 >= $frip1 && $frij2 <= $frip2 ){ //within the job
timing
                $fri = 100;
            }else if($frij1 >= $frip1 && $frij2 >= $frip2){ //from match
and to not match
                $fri = 50;
            }else if($frij1 <= $frip1 && $frij2 <= $frip2 ){ // from not
match and to match
                $fri = 50;
            }else{
                $fri = 0;
            }
        }
    }
    if(abs($satj1-$satj2) == 0){
        $sat = 100;
    }else{
        if(abs($satp1-$satp2) == 0){
            $sat =0;
        }else{
            if($satj1 >= $satp1 && $satj2 <= $satp2 ){ //within the job
timing
                $sat = 100;
            }else if($satj1 >= $satp1 && $satj2 >= $satp2){ //from match
and to not match
                $sat = 50;
            }else if($satj1 <= $satp1 && $satj2 <= $satp2 ){ // from not
match and to match
                $sat = 50;
            }else{
                $sat = 0;
            }
        }
    }

    if(abs($sunj1-$sunj2) == 0){
        $sun = 100;
    }else{
        if(abs($sunp1-$sunp2) == 0){
            $sun =0;
        }else{
            if($sunj1 >= $sunp1 && $sunj2 <= $sunp2 ){ //within the job
timing
                $sun = 100;
            }else if($sunj1 >= $sunp1 && $sunj2 >= $sunp2){ //from match
and to not match
```

```
                $sun = 50;
            }else if($sunj1 <= $sunp1 && $sunj2 <= $sunp2 ){ // from not
match and to match
                $sun = 50;
            }else{
                $sun = 0;
            }
        }
    }

$final = compact("mon", "tue", "wed", "thu", "fri", "sat",
"sun","t1","t2","diffdays");
return $final;

}

function timeDiffSeekerFixed($jobid, $profileid, $t1p, $t2p, $con){
    $smtj = $con->prepare("SELECT * FROM timediffjobs WHERE jobid=? AND
status='1'");
    $smtj->execute(array($jobid));
    $rowj =$smtj->fetch(PDO::FETCH_OBJ);

    $mon = $tue = $wed = $thu = $fri = $sat = $sun = $t1 = $t2 = $diffdays
= 0;

    $monj1 = strtotime($rowj->mon1);
    $monj2 = strtotime($rowj->mon2);
    $tuej1 = strtotime($rowj->tue1);
    $tuej2 = strtotime($rowj->tue2);
    $wedj1 = strtotime($rowj->wed1);
    $wedj2 = strtotime($rowj->wed2);
    $thuj1 = strtotime($rowj->thu1);
    $thuj2 = strtotime($rowj->thu2);
    $frij1 = strtotime($rowj->fri1);
    $frij2 = strtotime($rowj->fri2);
    $satj1 = strtotime($rowj->sat1);
    $satj2 = strtotime($rowj->sat2);
    $sunj1 = strtotime($rowj->sun1);
    $sunj2 = strtotime($rowj->sun2);

    $t1p = strtotime($t1p);
    $t2p = strtotime($t2p);

    if(abs($monj1-$monj2) == 0){ //full match if no time selected
        $mon = 100;
    }else{
            if($monj1 >= $t1p && $monj2 <= $t2p ){ //within the job timing
                $mon = 100;
            }else if($monj1 >= $t1p && $monj2 >= $t2p){ //from match and to
not match
                $mon = 50;
            }else if($monj1 <= $t1p && $monj2 <= $t2p ){ // from not match
and to match
                $mon = 50;
            }else{
                $mon = 0;
            }
        }


    if(abs($tuej1-$tuej2) == 0){
        $tue = 100;
    }else{
            if($tuej1 >= $t1p && $tuej2 <= $t2p ){ //within the job timing
                $tue = 100;
            }else if($tuej1 >= $t1p && $tuej2 >= $t2p){ //from match and to
not match
                $tue = 50;
            }else if($tuej1 <= $t1p && $tuej2 <= $t2p ){ // from not match
and to match
                $tue = 50;
            }else{
                $tue = 0;
            }
        }

    if(abs($wedj1-$wedj2) == 0){
        $wed = 100;
    }else {
            if($wedj1 >= $t1p && $wedj2 <= $t2p ){ //within the job timing
                $wed = 100;
            }else if($wedj1 >= $t1p && $wedj2 >= $t2p){ //from match and to
not match
                $wed = 50;
            }else if($wedj1 <= $t1p && $wedj2 <= $t2p ){ // from not match
and to match
                $wed = 50;
            }else{
                $wed = 0;
            }
        }


    if(abs($thuj1-$thuj2) == 0){
        $thu = 100;
    }else{
            if($thuj1 >= $t1p && $thuj2 <= $t2p ){ //within the job timing
                $thu = 100;
            }else if($thuj1 >= $t1p && $thuj2 >= $t2p){ //from match and to
not match
                $thu = 50;
            }else if($thuj1 <= $t1p && $thuj2 <= $t2p ){ // from not match
and to match
                $thu = 50;
            }else{
                $thu = 0;
            }
        }


    if(abs($frij1-$frij2) == 0){
        $fri = 100;
    } else{
            if($frij1 >= $t1p && $frij2 <= $t2p ){ //within the job timing
                $fri = 100;
            }else if($frij1 >= $t1p && $frij2 >= $t2p){ //from match and to
not match
                $fri = 50;
            }else if($frij1 <= $t1p && $frij2 <= $t2p ){ // from not match
and to match
                $fri = 50;
            }else{
                $fri = 0;
            }
        }

    if(abs($satj1-$satj2) == 0){
        $sat = 100;
    } else{
            if($satj1 >= $t1p && $satj2 <= $t2p ){ //within the job timing
                $sat = 100;
            }else if($satj1 >= $t1p && $satj2 >= $t2p){ //from match and to
not match
                $sat = 50;
            }else if($satj1 <= $t1p && $satj2 <= $t2p ){ // from not match
and to match
                $sat = 50;
            }else{
                $sat = 0;
            }
        }


    if(abs($sunj1-$sunj2) == 0){
        $sun = 100;
    } else{
            if($sunj1 >= $t1p && $sunj2 <= $t2p ){ //within the job timing
                $sun = 100;
            }else if($sunj1 >= $t1p && $sunj2 >= $t2p){ //from match and to
not match
                $sun = 50;
            }else if($sunj1 <= $t1p && $sunj2 <= $t2p ){ // from not match
and to match
                $sun = 50;
            }else{
                $sun = 0;
            }
        }


$final = compact("mon", "tue", "wed", "thu", "fri", "sat",
"sun","t1","t2","diffdays");
return $final;

}


function timeDiffEmpFixed($jobid, $profileid, $tj1, $tj2, $con){


    $smtp = $con->prepare("SELECT * FROM timediffprofiles WHERE
profileid=?");
    $smtp->execute(array($profileid));
    $rowp =$smtp->fetch(PDO::FETCH_OBJ);

    $mon = $tue = $wed = $thu = $fri = $sat = $sun = $t1 = $t2 = $diffdays
= 0;

    $tj1 = strtotime($tj1);
    $tj2 = strtotime($tj2);


    $monp1 = strtotime($rowp->mon1);
    $monp2 = strtotime($rowp->mon2);
    $tuep1 = strtotime($rowp->tue1);
    $tuep2 = strtotime($rowp->tue2);
    $wedp1 = strtotime($rowp->wed1);
    $wedp2 = strtotime($rowp->wed2);
    $thup1 = strtotime($rowp->thu1);
    $thup2 = strtotime($rowp->thu2);
    $frip1 = strtotime($rowp->fri1);
    $frip2 = strtotime($rowp->fri2);
    $satp1 = strtotime($rowp->sat1);
    $satp2 = strtotime($rowp->sat2);
    $sunp1 = strtotime($rowp->sun1);
    $sunp2 = strtotime($rowp->sun2);

        if(abs($monp1-$monp2) == 0){
            $mon =0;
        }else{
            if($tj1 >= $monp1 && $tj2 <= $monp2 ){ //within the job timing
                $mon = 100;
            }else if($tj1 >= $monp1 && $tj2 >= $monp2){ //from match and to
not match
                $mon = 50;
            }else if($tj1 <= $monp1 && $tj2 <= $monp2 ){ // from not match
and to match
                $mon = 50;
            }else{
                $mon = 0;
            }
        }



        if(abs($tuep1-$tuep2) == 0){
            $tue =0;
        }else{
            if($tj1 >= $tuep1 && $tj2 <= $tuep2 ){ //within the job timing
                $tue = 100;
            }else if($tj1 >= $tuep1 && $tj2 >= $tuep2){ //from match and to
not match
                $tue = 50;
            }else if($tj1 <= $tuep1 && $tj2 <= $tuep2 ){ // from not match
and to match
                $tue = 50;
            }else{
                $tue = 0;
            }
        }



        if(abs($wedp1-$wedp2) == 0){
            $wed =0;
        }else{
            if($tj1 >= $wedp1 && $tj2 <= $wedp2 ){ //within the job timing
                $wed = 100;
            }else if($tj1 >= $wedp1 && $tj2 >= $wedp2){ //from match and to
not match
                $wed = 50;
            }else if($tj1 <= $wedp1 && $tj2 <= $wedp2 ){ // from not match
and to match
                $wed = 50;
            }else{
                $wed = 0;
            }
        }



        if(abs($thup1-$thup2) == 0){
            $thu =0;
        }else{
            if($tj1 >= $thup1 && $tj2 <= $thup2 ){ //within the job timing
                $thu = 100;
            }else if($tj1 >= $thup1 && $tj2 >= $thup2){ //from match and to
not match
                $thu = 50;
            }else if($tj1 <= $thup1 && $tj2 <= $thup2 ){ // from not match
and to match
                $thu = 50;
            }else{
                $thu = 0;
            }
        }



        if(abs($frip1-$frip2) == 0){
            $fri =0;
        }else{
            if($tj1 >= $frip1 && $tj2 <= $frip2 ){ //within the job timing
                $fri = 100;
            }else if($tj1 >= $frip1 && $tj2 >= $frip2){ //from match and to
not match
                $fri = 50;
            }else if($tj1 <= $frip1 && $tj2 <= $frip2 ){ // from not match
and to match
                $fri = 50;
            }else{
                $fri = 0;
            }
        }



        if(abs($satp1-$satp2) == 0){
            $sat =0;
        }else{
            if($tj1 >= $satp1 && $tj2 <= $satp2 ){ //within the job timing
                $sat = 100;
            }else if($tj1 >= $satp1 && $tj2 >= $satp2){ //from match and to
not match
                $sat = 50;
            }else if($tj1 <= $satp1 && $tj2 <= $satp2 ){ // from not match
and to match
                $sat = 50;
            }else{
                $sat = 0;
            }
        }


        if(abs($sunp1-$sunp2) == 0){
            $sun =0;
        }else{
            if($tj1 >= $sunp1 && $tj2 <= $sunp2 ){ //within the job timing
                $sun = 100;
            }else if($tj1 >= $sunp1 && $tj2 >= $sunp2){ //from match and to
not match
                $sun = 50;
            }else if($tj1 <= $sunp1 && $tj2 <= $sunp2 ){ // from not match
and to match
                $sun = 50;
            }else{
                $sun = 0;
            }
        }


$final = compact("mon", "tue", "wed", "thu", "fri", "sat",
"sun","t1","t2","diffdays");
return $final;

}


function timeDiffAllFixed($tj1, $tj2, $tp1, $tp2){

    $tj1 = strtotime($tj1);
    $tj2 = strtotime($tj2);
    $tp1 = strtotime($tp1);
    $tp2 = strtotime($tp2);

    $available = 0;
    $mon = $tue = $wed = $thu = $fri = $sat = $sun =  $diffdays = 0;
    $diffdays = 1;
            if($tj1 >= $tp1 && $tj2 <= $tp2 ){ //within the job timing
                $available = 100;
            }else if($tj1 >= $tp2 && $tj2 >= $tp2){ //from match and to not
match
                $available = 50;
            }else if($tj1 <= $tp1 && $tj2 <= $tp2 ){ // from not match and
to match
                $available = 50;
            }else{
                $available = 0;
            }
    $t1 = $available/2;
    $t2 = $available/2;
    $final = compact("mon", "tue", "wed", "thu", "fri", "sat",
"sun","t1","t2","diffdays");

return $final;

}



?>
```

Also, below file matches the other requirements of the questionaries wizard

www/employer > qua_other.php

```
<?php

function QualificationsOther($jobid, $profileid, $con){

    $smtj = $con->prepare("SELECT * FROM job WHERE id=?");
    $smtj->execute(array($jobid));
    $rowj =$smtj->fetch(PDO::FETCH_OBJ);

    $smtp = $con->prepare("SELECT * FROM profile WHERE id=?");
    $smtp->execute(array($profileid));
    $rowp =$smtp->fetch(PDO::FETCH_OBJ);

    $qualification = $experience = $skills = $visaType = $license =
$vehicle = $ethnicity = $age = $gender = 0;

    $quJob = $rowj->qualification;
    $exJob = $rowj->experience;
    $skillsJob = $rowj->skills;
    $visaJob = $rowj->visaType;
    $licenseJob = $rowj->license;
    $vehicleJob = $rowj->vehicle;
    $ethnicityJob = $rowj->ethnicity;
    $age1Job = (int)$rowj->age1;
    $age2Job = (int)$rowj->age2;
    $genderJob = $rowj->gender;


    $quProfile = $rowp->qualification;
    $exProfile = $rowp->experience;
    $skillsProfile = $rowp->skills;
    $visaProfile = $rowp->visaType;
    $licenseProfile = $rowp->license;
    $vehicleProfile = $rowp->vehicle;
    $ethnicityProfile = $rowp->ethnicity;
    $ageProfile = (int)$rowp->age;
    $genderProfile = $rowp->gender;

    if($quJob == '0'){
        $qualification = 100;
    }else{
        if($quJob == $quProfile){
            $qualification = 100;
        }else{
            $qualification = 0;
        }
    }


    if($exJob == '0'){
        $experience = 100;
    }else{
        if($exJob == $exProfile){
            $experience = 100;
        }else{
            $experience = 0;
        }
    }


    if($skillsJob == ''){
        $skills = 100;
    }else{
        $skillsJobArray = explode(',',$skillsJob);
        $skillsProfileArray = explode(',', $skillsProfile);
        $countSkillsJobs = count($skillsJobArray);
        $matchSkills = array_intersect($skillsJobArray,
$skillsProfileArray);
        $countMatch = count($matchSkills);

        $countSkillsJobs = 100/(int)$countSkillsJobs;
        $skills = (int)$countSkillsJobs * (int)$countMatch;

    }

    if($visaJob == '0'){
        $visaType = 100;
    }else{
        if($visaJob == $visaProfile){
            $visaType = 100;
        }else{
            $visaType = 0;
        }
    }

    if($licenseJob == '0'){
        $license = 100;
    }else{
        if($licenseJob == $licenseProfile){
            $license = 100;
        }else{
            $license =0;
        }
    }


    if($vehicleJob == '0'){
        $vehicle = 100;
    }else{
        if($vehicleJob == $vehicleProfile){
            $vehicle = 100;
        }else{
            $vehicle =0;
        }
    }


    if($ethnicityJob == '0'){
        $ethnicity = 100;
    }else{
        if($ethnicityJob == $ethnicityProfile){
            $ethnicity = 100;
        }else{
            $ethnicity =0;
        }
    }



    if($age1Job <= $ageProfile && $age2Job >= $ageProfile){
        $age = 100;
    }else{
        $age = 0;
    }

    if($genderJob == '0'){
        $gender = 100;
    }else{
        if($genderJob == $genderProfile){
            $gender = 100;
        }else{
            $gender = 0;
        }
    }


    $final =
compact("qualification","experience","skills","visaType","license","vehicle
","ethnicity","age","gender");
return $final;
}
?>
```

View Shortlisted Job

This feature helps the jobs seekers to find and view their shortlisted jobs which has been shortlisted by the employer

www/seeker > data.php

```
<?php
include ('../php/db.php');

if(!isset($_SESSION['id']))
{
echo "Error";
}

$empID = $_SESSION['id'];
$allData = new stdClass();

$sql ="SELECT p.*, IFNULL(x.jobnames, 'Any') AS jobnames,
IFNULL(y.location, CONCAT('All ', l.location)) AS location,
CASE WHEN p.jobType=0 THEN 'Any' WHEN p.jobType=1 THEN 'Casual' WHEN
p.jobType=2 THEN 'Part-time' WHEN p.jobType=3 THEN 'One-Off' END as
jobTypeName
FROM profile p
LEFT JOIN (SELECT pj.profileid, GROUP_CONCAT(jt.jobTitle) as jobnames FROM
profilemultijobtitles pj LEFT JOIN jobtitle jt ON pj.jobTitleid = jt.id
GROUP BY pj.profileid) x ON x.profileid = p.id
LEFT JOIN (SELECT pl.profileid, GROUP_CONCAT(sl.location) as location FROM
    profilemultilocationsub pl LEFT JOIN locations_sub sl ON
pl.locationSubid = sl.id GROUP BY pl.profileid) y ON y.profileid = p.id
```

```
LEFT JOIN locations l ON p.location = l.id
WHERE p.userid=? AND p.status='1'";

$smt=$con->prepare($sql);
$smt->execute(array($empID));
$rowProfile = $smt->fetchAll(PDO::FETCH_OBJ);

$allData->profiles = $rowProfile;


$smt=$con->prepare("SELECT empid, jobid, profileid,  COUNT(*) as countdata
FROM jobmatch WHERE seekerid=? AND status ='1' and short='1' GROUP BY
profileid");
$smt->execute(array($empID));
$row = $smt->fetchAll(PDO::FETCH_OBJ);
$allData->match = $row;


print json_encode($allData);
?>
```

Accept Shortlisted Job

Job Seeker can accept the shortlisted jobs which has been shortlisted by the employer by using this feature.

www/matchedjobs > short.php

```
<?php

include ('../php/db.php');
include ('../php/session.php');

if(!isset($_POST['jobID'])){
    return false;
}
$empType = $_SESSION['type'];
if(!$empType == '1'){
    return false;
}

$jobid = (int)$_POST['jobID'];
$matchid = $_POST['matchid'];
$event = $_POST['short'];
if($event == 'add'){
    $smt = $con->prepare("SELECT * FROM jobmatch where id=? AND jobid=? AND
short=0");
    $smt->execute(array($matchid, $jobid));
    $rowCount = $smt->rowCount();
    if($rowCount == 0){
        echo "Error";
        return false;
    }

    $smt = $con->prepare("UPDATE jobmatch SET short=1 WHERE jobid=? AND
id=?");
    $smt->execute(array($jobid,$matchid));
    echo "ok";
    return false;
}else if($event == 'remove'){
    $smt = $con->prepare("SELECT * FROM jobmatch where id=? AND jobid=? AND
short=1 OR (short=2)");
    $smt->execute(array($matchid, $jobid));
    $rowCount = $smt->rowCount();
    if($rowCount == 0){
        echo "Error";
        return false;
    }

    $smt = $con->prepare("UPDATE jobmatch SET short=0 WHERE jobid=? AND
id=?");
    $smt->execute(array($jobid,$matchid));
    echo "ok";
    return false;
}else{
    echo "error";
    return false;
}


?>
```

Employer Rating and Comments

For Job Seeker:

```
<?php

include ('../php/db.php');
include ('../php/session.php');

if(isset($_POST['comment'])){
        $rating = $_POST['rating'];
        $comment = $_POST['comment'];
        $seekerid = $_POST['seekerid'];
        $userid = $_SESSION['id'];


        $smt = $con->prepare("INSERT INTO ratingseeker (seekerid, userid,
comment, rating) VALUES(?,?,?,?)");
        $smt->execute(array($seekerid, $userid, $comment, $rating ));
        echo "ok";
        return false;

}
else {
        echo "error";
        return false;
}
?>
```

Job Seeker Rating and Comments

For Employer:

```
<?php

include ('../php/db.php');
include ('../php/session.php');

if(isset($_POST['comment'])){
        $rating = $_POST['rating'];
        $comment = $_POST['comment'];
        $empid = $_POST['empid'];
        $userid = $_SESSION['id'];


        $smt = $con->prepare("INSERT INTO ratingemp (empid, userid, 
comment, rating) VALUES(?,?,?,?)");
        $smt->execute(array($empid, $userid, $comment, $rating ));
        echo "ok";
        return false;

}
else{
        echo "error";
        return false;
}
?>
```

Logout

A log out button is present on all pages of this application. The log out button should be identified quickly by a user who wants to log out from the application, and it will redirect to the home page after successful logout.

www/php/logout.php

```
<?php

if (session_status() == PHP_SESSION_NONE) {
   ini_set('session.gc_maxlifetime', 288000);
   session_set_cookie_params(28800);
   session_start();
}
session_destroy();
//header('Location: ../home');
echo "<script type='text/javascript'> document.location = '../home'; 
</script>";
?>
```