



BRAIN STATION 23



Submitted by: CUET Industrial Attachment Team (BS-23)

Date of Submission: 19th January, 2017

TEAM MEMBERS

ID	NAME
M.A. Hasan Molla	1204006
Md Abdul Quaum	1204026
Ashfak Md Shibli	1204028
Shuvashish Sarker	1204108
A. S. M. Sadiqul Islam	1204113

INDEX

SL	Topic
1	Introduction
2	User Requirement Specification
3	System Requirement Specification
4	Our Idea & Concepts
5	Using VCS
6	Task Distribution
7	User Testing
8	Product Screenshots
9	Application Codes
10	Conclusion

Document Overview

Our Industrial Attachment was in Brain Station-23 a renowned software company in our country. They have various software projects around the world. It was in 2006, with little capital but a pocketful of belief that Raisul Kabir started Brain Station-23, a software company. Today, that company is valued at around Tk 20 crore and last year, generated revenues of around Tk 8 crore and won the HSBC Export Excellence Award 2013 in the small and medium enterprises category.

The company's application portfolio include: CITY Bank Touch Application, ROBI intranet portal, Incepta Mobile Sales Force Automation and AB Leave Management system. It also provides outsourcing service to its global clients such as British Telecom and Congral in the US.

In 2005, Latitude-23 received a big work order from a US client, and Kabir was hired to help execute it within three months. Even though it was pretty tight, it was well executed. In 2006, when work orders starting flowing in, they formed Brain Station-23.

The company is now focusing on banking and pharmaceutical sectors for expansion. It has plans to increase its workforce five-fold in the next four years. It now employs around 100. The work rate charged by the company is either hour based or project based, depending on the service offered, and the applicable business model. Rates can range between \$15 and \$20 an hour.

In terms of cost competitiveness in outsourcing services, Bangladesh ranks first, according to the study. Bangladesh Association of Software and Information Services set a target to earn \$1 billion from such exports and create 1 million professionals in five years.

Introduction

We were instructed to build a Web Application to get all the CVs from the Corporate account of Brain Station-23 in Bdjobs.com by automated login & Scrapping the Bdjobs websites data. Then complex search in those CVs using various keywords and categories.

User Requirement Specification

- Their problem was, they have to check many CV on bdjobs.com for various post of their company.
- So, their requirements were, we would develop a web application, in which CV are being downloaded.
- Their another requirement was, we would develop a search engine so that they can find out their desired candidate.

System Requirement Specification

Our system has two parts.

1. Backend (Webscrapping)
2. Frontend (UI)

The architecture of the system is followed in the figure 1 & 2.

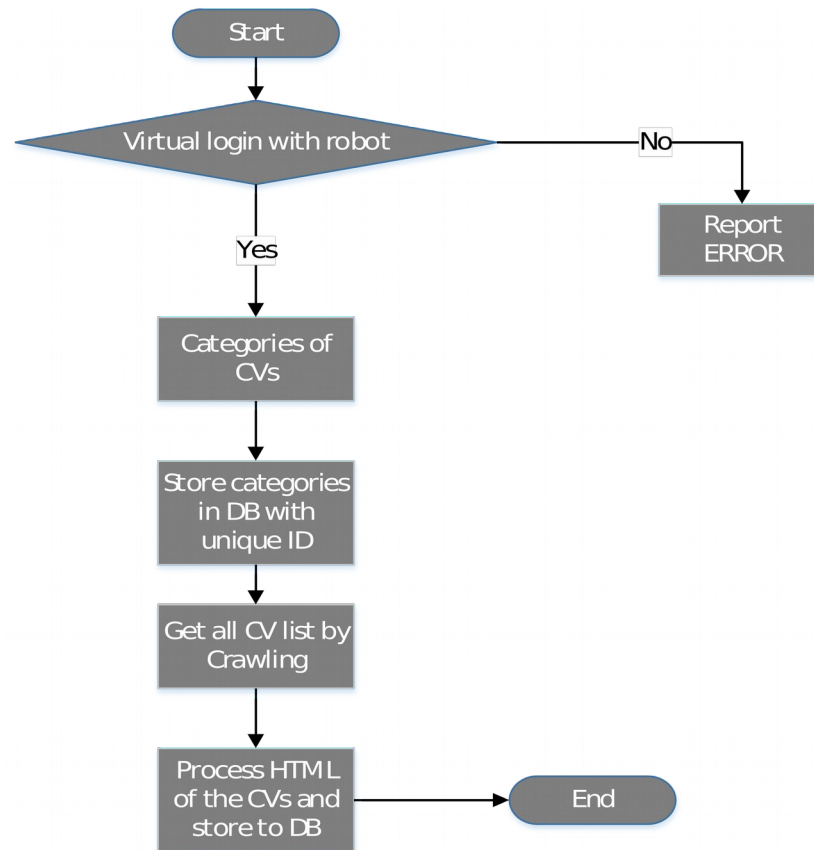


Figure 1: Backend System Architecture

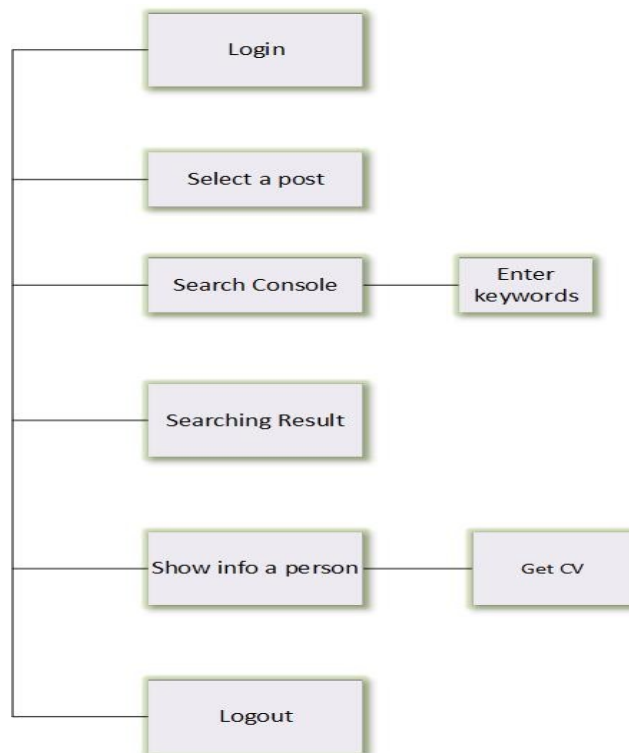


Figure 2: Frontend System Architecture

Required Application and System Environments

1. Editors & IDE: PHPstorm, Sublime text
2. Language: PHP, HTML, CSS, Java script
3. Frame work: Laravel, Symfony
4. Server: Apache
5. DataBase: MySQL

Our Idea & Concepts

Our Idea of building the Application:

- Programmatically login as brainstation user in bdjobs site.
- Scrap all job post
- Scrap all candidates data and store those in database
- Design a search engine to search their desired candidate(s)
- Scheduled command to keep database up-to-date

Learning Concepts:

Web scraping:

Web scraping means data scraping used for extracting data from websites. Web scraping software may access the World Wide Web(WWW) directly using the Hypertext Transfer Protocol or through a web browser. It is some kind of copying, in which specific data is gathered and copied from the web. It is a technique employed to extract large amounts of data from websites whereby the data is extracted and saved to a local file in your computer or to a database in table format. Typically, data are copied into a central local database or spreadsheet, for later retrieval or analysis. Web scraping involves with fetching web site and extracting from it. Fetching means the downloading of a page. Web crawling is a main component of web scraping. Once fetched, then extraction can take place. The content of a page may

be parsed, searched, reformatted, its data copied into a spreadsheet, and so on. Web scrapers typically take something out of a page, to make use of it for another purpose somewhere else. An example would be to find and copy names and phone numbers, or companies and their URLs, to a list.

Data displayed by most websites can only be viewed using a web browser. They do not offer the functionality to save a copy of this data for personal use. The only option then is to manually copy and paste the data - a very tedious job which can take many hours or sometimes days to complete. Web scraping is the technique of automating this process, so that instead of manually copying the data from websites, the web scraping will perform the same task within a fraction of the time.

A web scrapper will automatically load and extract data from multiple pages of websites based on requirement. It is either custom built for a specific website or is one which can be configured to work with any website. With the click of a button we can easily save the data available in the website to a file in computer.

cURL:

cURL is a command line tool for getting or sending files using URL syntax. Since cURL uses libcurl, it is a tool to transfer data from or to a server, using one of the supported protocols (DICT, FILE, FTP, FTPS, GOPHER, HTTP, HTTPS, IMAP, IMAPS, LDAP, LDAPS, POP3, POP3S, RTMP, RTSP, SCP, SFTP, SMB, SMBS, SMTP, SMTPS, TELNET and TFTP). The command is designed to work without user interaction.

curl offers a busload of useful tricks like proxy support, user authentication, FTP upload, HTTP post, SSL connections, cookies, file transfer resume, Metalink, and more.

Symfony:

Symfony is a set of reusable PHP components. Symfony Components are a set of decoupled and reusable PHP libraries. They are becoming the standard foundation on which the best PHP applications are built on. We can use any of these components in applications independently from the Symfony Framework. It is not an absolute necessity, but nonetheless it is very useful. It is a pledge of quality, upgradability and maintainability of applications at lower cost. Applications developed using it are interoperable with market standards.

Symfony DOM Crawler:

The crawler class provides methods to query and manipulate HTML and XML documents. An instance of the Crawler represents a set of DOMElement objects, which are basically nodes that can traverse easily. The DomCrawler will attempt to automatically fix HTML to match the official specification. For example, if we nest a <p> tag inside another <p> tag, it will be moved to be a sibling of the parent tag. This is expected and is part of the HTML5 spec. But if getting unexpected behavior, this could be a cause. And while the DomCrawler isn't meant to dump content, we can see the "fixed" version of your HTML by dumping it.

To find an image by its alt attribute, use the selectImage method on an existing crawler. This returns a crawler instance with just the selected image(s). Calling image () gives you a special image object. We can grab the information off of form as if it had just been submitted by using the PHP values. If it is using an external HTTP client, we can use the form to grab all of the information that need to create a POST request for the form.

Laravel:

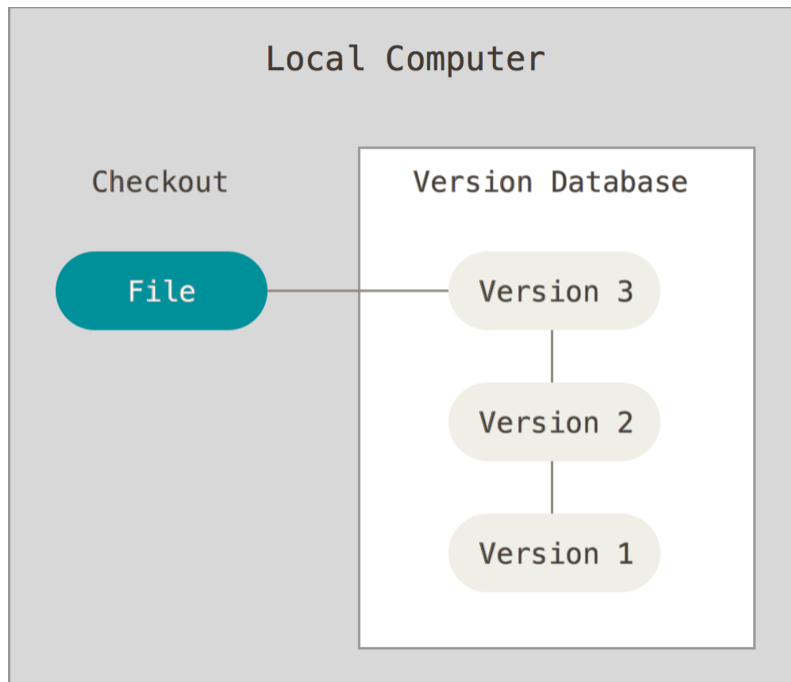
Laravel is a free, open-source PHP web framework, created by Taylor Otwell and intended for the development of web applications following the model–view–controller architectural pattern. Laravel is a web application framework with expressive, elegant syntax. We believe development must be an enjoyable, creative experience to be truly fulfilling. Laravel attempts to take the pain out of development by easing common tasks used in the majority of web projects, such as authentication, routing, sessions, and caching. Laravel is accessible, yet powerful, providing powerful tools needed for large, robust applications. A superb inversion of control container, expressive migration system, and tightly integrated unit testing support give the tools need to build any application with which are tasked.

Using VCS – GIT:

We have used VCS to control our apps version and backup. Many people's version-control method of choice is to copy files into another directory (perhaps a time-stamped directory, if they're clever). This approach is very common because it is so simple, but it is also incredibly error prone. It is easy to

forget which directory you're in and accidentally write to the wrong file or copy over files you don't mean to.

To deal with this issue, programmers long ago developed local VCSs that had a simple database that kept all the changes to files under revision control.



As we are working as a team git and version control helped us keep backup of our tasks.

Task Distribution

Task	Time	Assigned to
Web Crawler & Scrapping	7 days	Shibli, Sadiqul
Database Design	2 days	Hasan, Quaum
UI design	3 days	Shuvashish
Testing	1 day	All
Presentation & Overview	1 day	All

User Testing

After completing the task, we have tested our application if it works correctly as required. We have also given the HR of BS-23 for testing and requirements anything they need more. We have made changes as per their requirement instantly. And it was accepted and appreciated by them.

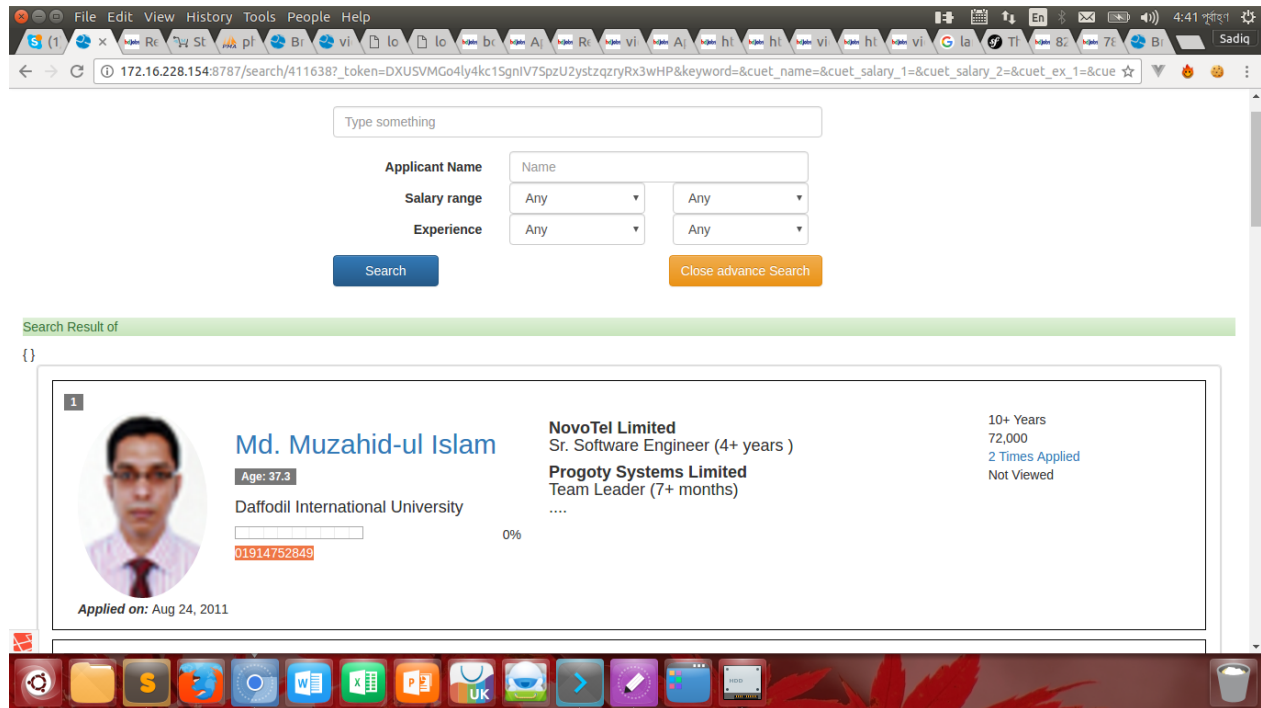


Figure 3: Testing our Web-app

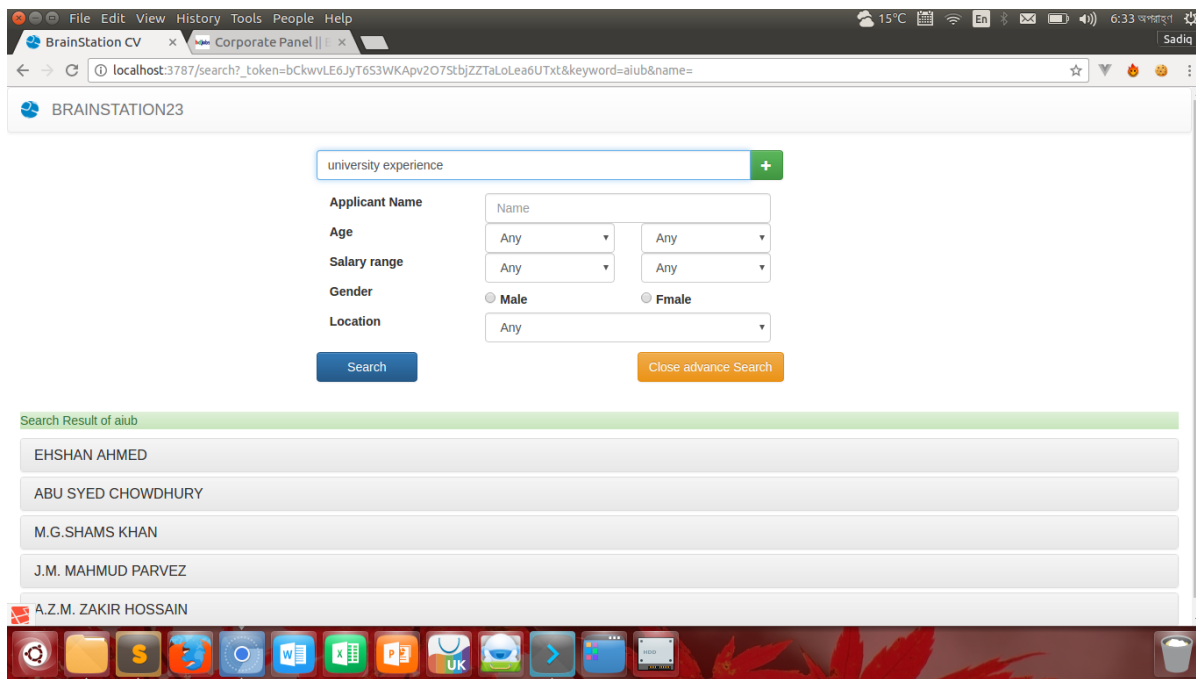


Figure 4: Testing by User

Product Screenshots:

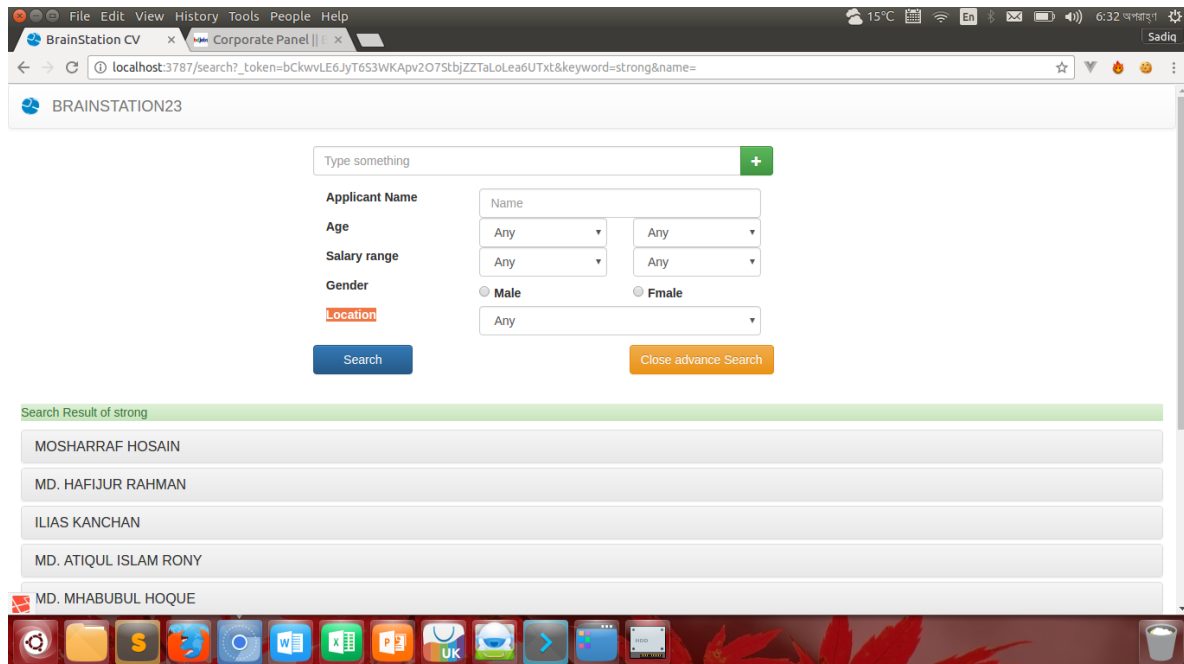


Figure 5: Searching in Our Application

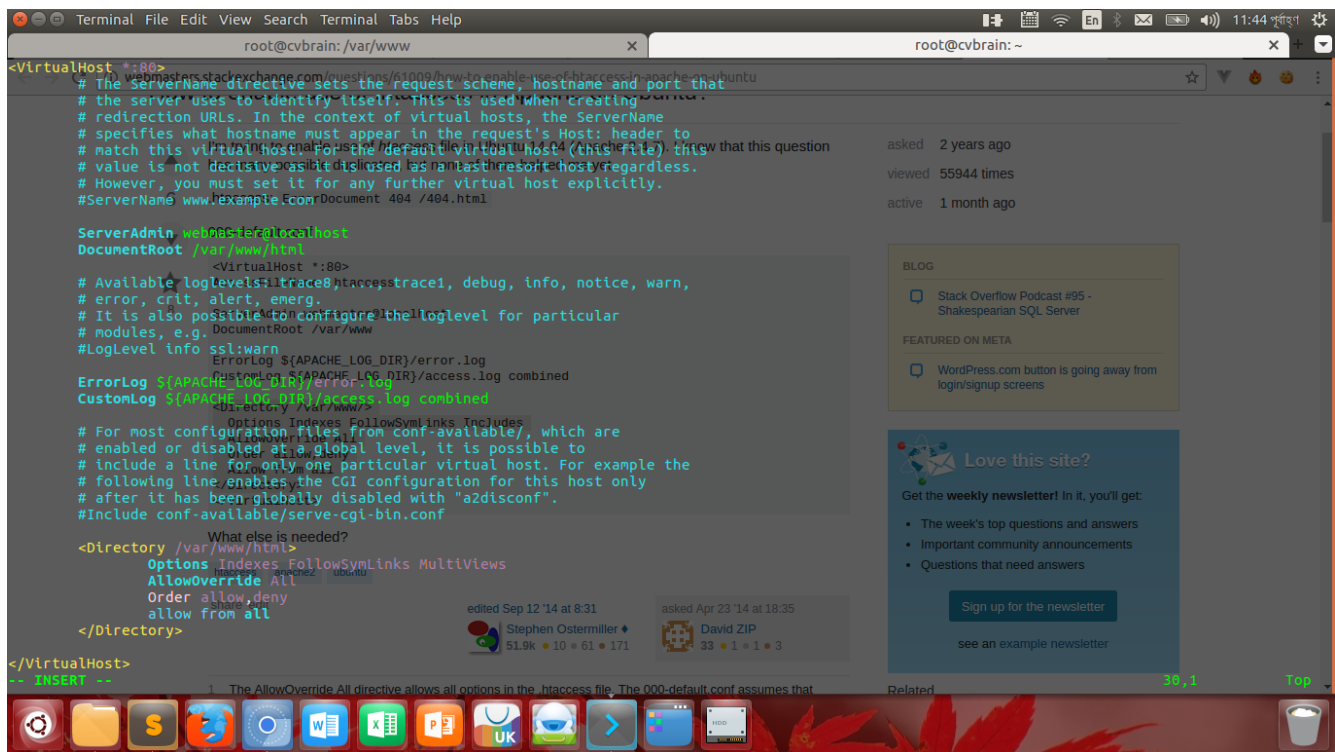


Figure 6: Debugging Process

Application Codes:

Controllers:

Controller.php

```
<?php
namespace App\Http\Controllers;
use Illuminate\Foundation\Bus\DispatchesJobs;
use Illuminate\Routing\Controller as BaseController;
use Illuminate\Foundation\Validation\ValidatesRequests;
use Illuminate\Foundation\Auth\Access\AuthorizesRequests;
class Controller extends BaseController
{
    use AuthorizesRequests, DispatchesJobs, ValidatesRequests;
}
```

HomeController.php

```
<?php
```

```

namespace App\Http\Controllers;
use Illuminate\Http\Request;
class HomeController extends Controller
{
    /**
     * Create a new controller instance.
     *
     * @return void
     */
    public function __construct()
    {
        $this->middleware('auth');
    }
    /**
     * Show the application dashboard.
     *
     * @return \Illuminate\Http\Response
     */
    public function index()
    {
        return view('home');
    }
}

```

SearchController.php

```
<?php
```

```

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use App\RawResume;

use App\ResumeLink;

use App\Http\Requests;

```

```

class SearchController extends Controller
{
    public function search(Request $request)
    {
        $searchKey = $request->input('keyword');

        $RawResumes = RawResume::where(function ($query) use ($searchKey, $request) {

            $query->where('name', 'like', '%'.str_replace(' ', '', $searchKey).'%')

            ->orWhere('address', 'like', "%{$searchKey}%")

            ->orWhere('mobile_email', 'like', "%{$searchKey}%")

            ->orWhere('personal', 'like', "%{$searchKey}%")

            ->orWhere('employment_history', 'like', "%{$searchKey}%");

            if (! empty($request->input('name')) ) {

                $query->where('name', 'like', "%{$request->input('name')}%");

            }

        }->paginate(10);

        //dd($RawResumes);

        //$RawResumes = RawResume::where('name', 'like', "%$searchKey%")->get();

        return view('search', compact('RawResumes', 'searchKey'));

    }
}

```

Authentication Controllers

LoginController.php

```

<?php
namespace App\Http\Controllers\Auth;
use App\Http\Controllers\Controller;

```

```

use Illuminate\Foundation\Auth\AuthenticatesUsers;

class LoginController extends Controller
{
    /**
     |-----
     | Login Controller
     |-----
     |
     | This controller handles authenticating users for the application and
     | redirecting them to your home screen. The controller uses a trait
     | to conveniently provide its functionality to your applications.
     |
     */
    use AuthenticatesUsers;

    /**
     * Where to redirect users after login.
     *
     * @var string
     */
    protected $redirectTo = '/';

    /**public function username()
    {
        return 'username';
    }*/
    /**
     * Create a new controller instance.
     *
     * @return void
     */
    public function __construct()
    {
        $this->middleware('guest', ['except' => 'logout']);
    }

```



```
}  
}
```

Job.php

```
<?php  
namespace App;  
use Illuminate\Database\Eloquent\Model;  
class Job extends Model  
{  
    protected $fillable = [  
        'name', 'longtext', 'bdjobs_id',  
    ];  
}
```

RawResume.php

```
<?php  
namespace App;  
use Illuminate\Database\Eloquent\Model;  
class RawResume extends Model  
{  
    protected $fillable = ['last_updated',  
        'name',  
        'mobile_no',  
        'mobile_email',  
        'office_phone',  
        'home_phone',
```

```
'address',  
  
'career_objective',  
  
'career_summary',  
  
'special_qualification',  
  
'employment_history',  
  
'academic_qualification',  
  
'training_summary',  
  
'career_and_application',  
  
'specialization',  
  
'extra_curricular_activities',  
  
'language_proficiency',  
  
'personal',  
  
'references',  
  
'link', 'html', 'api', 'bdjobs_id',  
  
'data_expected_salary',  
  
'data_years_of_experience'  
  
];  
  
}
```

Resume.php

```
<?php  
  
namespace App;  
  
use Illuminate\Database\Eloquent\Model;  
  
class Resume extends Model  
  
{
```

```
protected $fillable = [  
    'name', 'email', 'password',  
];  
}
```

ResumeLink.php

```
<?php  
  
namespace App;  
  
use Illuminate\Database\Eloquent\Model;  
  
class ResumeLink extends Model  
{  
    protected $fillable = [  
        'link', 'html', 'api', 'bdjobs_id',  
    ];  
}
```

User.php

```
<?php  
  
namespace App;  
  
use Illuminate\Notifications\Notifiable;  
use Illuminate\Foundation\Auth\User as Authenticatable;  
  
class User extends Authenticatable  
{  
    use Notifiable;  
  
    /**  
     * The attributes that are mass assignable.
```

```

*

* @var array

*/

protected $fillable = [

    'name', 'email', 'password',

];

/**

* The attributes that should be hidden for arrays.

*

* @var array

*/

protected $hidden = [

    'password', 'remember_token',

];

}

```

cURL:

cookie.txt

Netscape HTTP Cookie File

http://curl.haxx.se/docs/http-cookies.html

This file was generated by libcurl! Edit at your own risk.

```

corporate3.bdjobs.com      FALSE      /      TRUE 0      ASPSESSIONIDACUTSTCQ
      KCHHJDPAAKHHKOEAKPMC�KKK
corporate3.bdjobs.com      FALSE      /      FALSE 0
      ASPSESSIONIDACQTSTCQ IEOLHDPÄENPEPCBBMBIDEOBE
corporate3.bdjobs.com      FALSE      /      FALSE 0      Company      sCode
%2DEmpl=bra22440&ComNo=22547

```

synchronise.js

```
function synchronise()

{

    un0=getCookie("corporateUN0");

    un1=getCookie("corporateUN1");

    un2=getCookie("corporateUN2");

    un3=getCookie("corporateUN3");

    un4=getCookie("corporateUN4");

    //console.log(un0)

    if(un0!=null)

    {

        username0=un0.substring(0,un0.indexOf("#"));

        password0=getCookie("corporatePS0");

    }


    if(un1!=null)

    {

        username1=un1.substring(0,un1.indexOf("#"));

        password1=getCookie("corporatePS1");

    }


    if(un2!=null)

    {
```

```
username2=un2.substring(0,un2.indexOf("#"));

password2=getCookie("corporatePS2");

}

if(un3!=null)

{

    username3=un3.substring(0,un3.indexOf("#"));

    password3=getCookie("corporatePS3");

}

if(un4!=null)

{

    username4=un4.substring(0,un4.indexOf("#"));

    password4=getCookie("corporatePS4");

}

myusername=document.getElementById("NAME").value;

chkRem=document.getElementById("chkRem");

if(myusername==username0)

{
```

```
document.getElementById("PASS").value=password0;

chkRem.checked=true;

}

else if(myusername==username1)

{

    document.getElementById("PASS").value=password1;

    chkRem.checked=true;

}

else if(myusername==username2)

{

    document.getElementById("PASS").value=password2;

    chkRem.checked=true;

}

else if(myusername==username3)

{

    document.getElementById("PASS").value=password3;

    chkRem.checked=true;

}

else if(myusername==username4)

{

    document.getElementById("PASS").value=password4;

    chkRem.checked=true;

}
```

```

else

{

    //document.getElementById("PASS").value="";

    chkRem.checked=false;

}

}

function getCookie(c_name) {

    if (document.cookie.length > 0) {

        c_start = document.cookie.indexOf(c_name + "=");

        if (c_start != -1) {

            c_start = c_start + c_name.length + 1;

            c_end = document.cookie.indexOf(";", c_start);

            if (c_end == -1) {

                c_end = document.cookie.length;

            }

            return unescape(document.cookie.substring(c_start, c_end));

        }

    }

    return "";

}

```

takes2.php

```
<?php
```

```
class mycurl {
```



```
protected $_useragent = 'Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1';

protected $_url = "";

protected $_followlocation;

protected $_timeout;

protected $_maxRedirects;

protected $_cookieFileLocation = './cookie.txt';

protected $_post;

protected $_postFields;

protected $_referer = "http://corporate3.bdjobs.com/";

protected $_session;

protected $_webpage;

protected $_includeHeader;

protected $_noBody;

protected $_status;

protected $_binaryTransfer;

public $authentication = 0;

public $auth_name = "";

public $auth_pass = "";

public function useAuth($use){

    $this->authentication = 0;

    if($use == true) $this->authentication = 1;

}

public function setName($name){

    $this->auth_name = $name;
```

```

}

public function setPass($pass){

    $this->auth_pass = $pass;

}

    public function __construct($url,$followlocation = true,$timeOut = 300,$maxRedirects =
40,$binaryTransfer = false,$includeHeader = false,$noBody = false)

{

    $this->_url = $url;

    $this->_followlocation = $followlocation;

    $this->_timeout = $timeOut;

    $this->_maxRedirects = $maxRedirects;

    $this->_noBody = $noBody;

    $this->_includeHeader = $includeHeader;

    $this->_binaryTransfer = $binaryTransfer;

    $this->_cookieFileLocation = dirname(__FILE__).'./cookie.txt';

}

public function setReferer($referer){

    $this->_referer = $referer;

}

public function setCookiFileLocation($path)

{

    $this->_cookieFileLocation = $path;

}

public function setPost ($postFields)

```

```

{
    $this->_post = true;

    $this->_postFields = $postFields;
}

public function setUserAgent($userAgent)
{
    $this->_useragent = $userAgent;
}

public function createCurl($url = 'nul')
{
    if($url != 'nul'){
        $this->_url = $url;
    }

    $s = curl_init();

    curl_setopt($s,CURLOPT_URL,$this->_url);

    curl_setopt($s,CURLOPT_HTTPHEADER,array('Expect:'));

    curl_setopt($s,CURLOPT_TIMEOUT,$this->_timeout);

    curl_setopt($s,CURLOPT_MAXREDIRS,$this->_maxRedirects);

    curl_setopt($s,CURLOPT_RETURNTRANSFER,true);

    curl_setopt($s,CURLOPT_FOLLOWLOCATION,$this->_followlocation);

    curl_setopt($s,CURLOPT_COOKIEJAR,$this->_cookieFileLocation);

    curl_setopt($s,CURLOPT_COOKIEFILE,$this->_cookieFileLocation);

    if($this->authentication == 1){

        curl_setopt($s, CURLOPT_USERPWD, $this->auth_name.':'.$this->auth_pass);
    }
}

```

```

}

if($this->_post)
{
    curl_setopt($s,CURLOPT_POST,true);
    curl_setopt($s,CURLOPT_POSTFIELDS,$this->_postFields);
}

if($this->_includeHeader)
{
    curl_setopt($s,CURLOPT_HEADER,true);
}

if($this->_noBody)
{
    curl_setopt($s,CURLOPT_NOBODY,true);
}

/*

if($this->_binary)
{
    curl_setopt($s,CURLOPT_BINARYTRANSFER,true);
}

*/

curl_setopt($s,CURLOPT_USERAGENT,$this->_useragent);
curl_setopt($s,CURLOPT_REFERER,$this->_referer);

$this->_webpage = curl_exec($s);

    $this->_status = curl_getinfo($s,CURLINFO_HTTP_CODE);

```

```

        curl_close($s);
    }

    public function getHttpStatus()
    {
        return $this->_status;
    }

    public function __toString(){
        return $this->_webpage;
    }
}

$myCurl          =          new          mycurl('https://corporate3.bdjjobs.com/corporate_submit.asp?
NAME=brainsta&PASS=sajib@bs23&submit=Sign%20In&JobNo=&AppType');

//$myCurl->useAuth(true);

$myCurl->setName('brainsta');

$myCurl->setPass('sajib@bs23');

$myCurl->createCurl();

//echo $myCurl->__toString();

$myCurl          =          new          mycurl('https://corporate3.bdjjobs.com/corporate_login.asp?
1[.\\4a.a._0c=aic1\\2[=[o[sc=]p_t`p]=cn_o\\&`4a2_=^I]tcr`p`o^9^6]1b4^3a1^pa&b9c4[d`c_&be^ybg_&]
ocb]j`7b5b2\\dZeaaco[raCc4');

$myCurl->createCurl();

echo $myCurl->__toString();

?>

```

Conclusion:

At the end, we can say that, we had to work hard to fulfill their requirements. All though the task was not so easy, we could solve the problem because there was many user privilege and dynamic session system. But at last we were able to break the ice. The HR and CEO were very happy about the web application. And environment of the company and behavior of employee was very nice.