



PROJECT REPORT ON 'LINKEDIN FOR STUDENTS PLACEMENTS'

PRT 452 SOFTWARE ENGINEERING: PROCESS AND TOOLS

NAME	ROLE	Contact details
Dr. Simon Moss	Product Owner and Client	Simon.Moss@cdu.edu.au
Dr. Peter Shaw	Subject faculty	peter.shaw@cdu.edu.au
Soumith Reddy	Project Team Member	soumith1107@gmail.com
Rakesh Tata	Project Team Member	rakcoolcracker@gmail.com
Ajay Kumar	Project Team Member	vajaykumar111@gmail.com

Contents

Project Introduction.....	3
Requirements.....	3
Functional requirements.....	3
Non-functional requirements.....	4
Software requirements.....	4
Hardware requirements.....	4
UI Design.....	4
DataBase Design.....	8
ER diagram	9
UML Design.....	9
Class Diagram.....	9
Activity Diagram.....	9
Sequence diagram.....	11
Use Case Diagram.....	12
Agile process.....	14
Pair Programming.....	14
User story Cards.....	14
Test cases.....	15
CODE.....	18
References.....	38

Project Introduction

LinkedIn is a web application which can be accessed by Students and Recruiter's. This web application displays all available job opportunities to every student who registers and sign in for job opportunities. Students must register in this web application and can search for any job posted by recruiter. The student who registered as student can only apply for the job. The user who registered as a recruiter can post for a job opportunity. Recruiter can post as many as jobs in the web application and can also see through the available jobs in the web application. This application has search bar where user can filter all the suitable list of job opportunities with the required location or company name or job role or job type which helps user to find required job easy.

Requirements

Functional requirements

The web application shall

- Initiate the signup process.
- Verify the email address used registration for genuine users.
- Allow the recruiter to post the placements.
- Allow the recruiter to post all the requirements needed for the job.
- Allow the students to search for the placements.
- Not allow the non registered users to post a job.
- Allow the registered students to apply for the job.
- Allow the users both students and recruiter to sign out.

Non-functional requirements

- The web application should be easy to use and the search process should be accurate.
- The web application should be available all time.
- Details of the users should be secured in the web application.
- The web applications ought to have the capacity to keep up the immense dataset.
- Response times for search jobs and post jobs in web application should be less.

- The web application should be friendly to all browsers and should function easily on all platforms

Software requirements:

- SQL SERVER Express 2008 R2.
- Visual Studio Community 2015.
- Supported Operating System: Windows 10, Windows 8.1, Windows 8, Windows 7 SP 1, Windows Server 2012 R2, Windows Server 2012, Windows Server 2008 R2 SP1.

Hardware requirements:

- 1.6 GHz or faster processor
- 1 GB of RAM (1.5 GB if running on a virtual machine)
- 10 GB of available hard disk space.

UI Design:

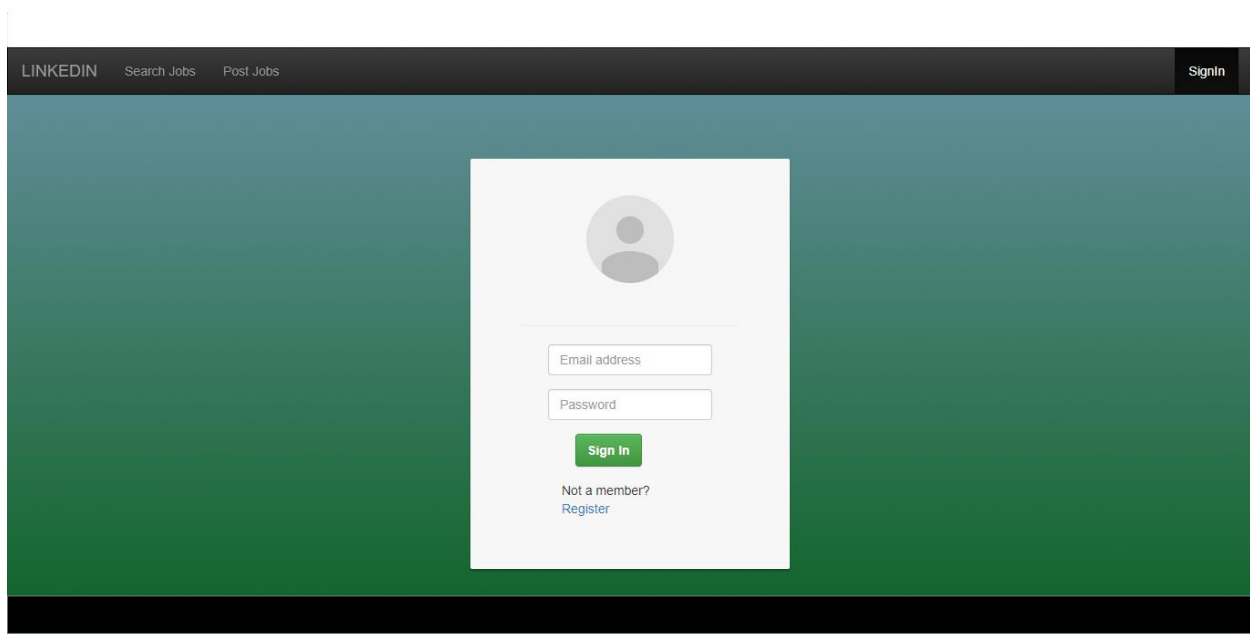


Fig 1: HOME PAGE.

LINKEDIN
Search Jobs
Post Jobs
Signin

Sign Up

Are you a? ☒ Student ☐ Recruiter

First Name Last Name

Email Contact Number

Gender ☒ Male ☐ Female

Matriculation(%) Intermediate(%)

Graduation(%) Post Graduation(%)

Password Confirm Password

Sign Up

Fig 2: Registration/login Form is displayed

LINKEDIN
Search Jobs
SignOut

Search Jobs

Show 10 entries

Search:

Company Name	Website	Company Email	Job Type	Course	Location	
deloitte	www.deloitte.com	dbz@deloitte.com	Parttime	Goa	MSc	Apply
qwerty	www.q.com	q@q.com	Fulltime	1.18.1	MBA	Apply
TCS	www.tcs.com	xyz@tcs.com	Fulltime	hyderabad	MBA	Apply
TCS	www.tcs.com	xyz@tcs.com	Fulltime	hyderabad	MBA	Apply
wipro	www.wipro.com	qw@wipro.com	Fulltime	pune	MBA	Apply

Showing 1 to 5 of 5 entries

Previous 1 Next

Fig 3: Student Jobs

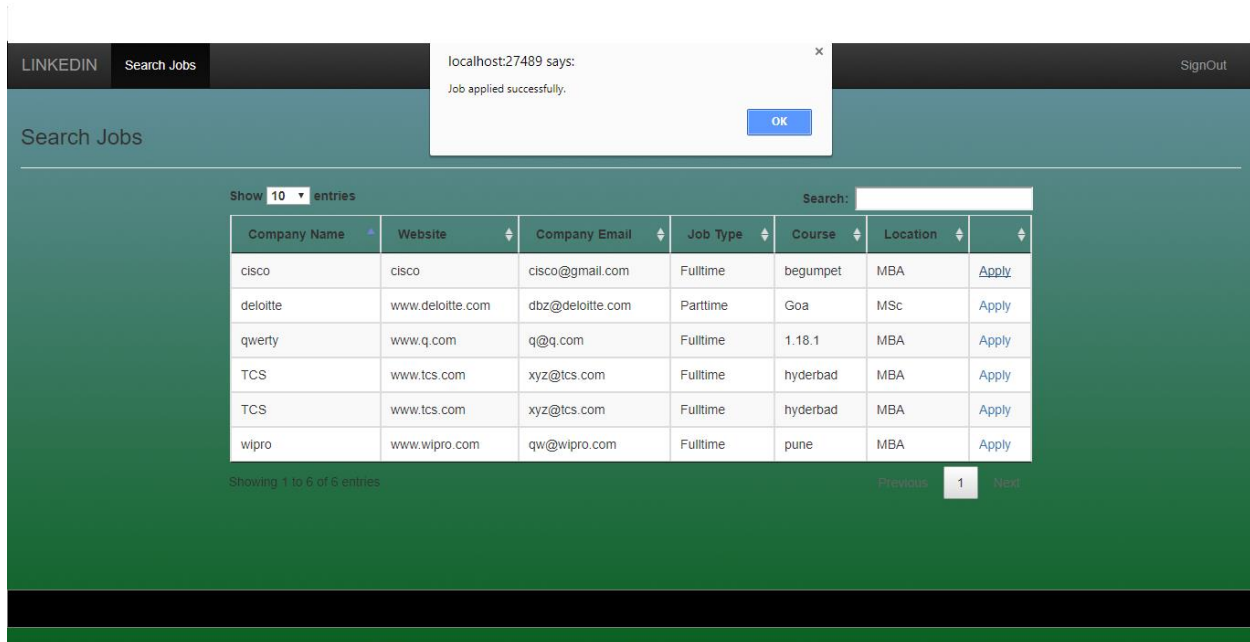


Fig 4: Student successfully applied for Job.

The screenshot shows a web application interface for recruiter registration. At the top, there is a navigation bar with 'LINKEDIN', 'Search Jobs', 'Post Jobs', and 'SignIn'. The 'Sign Up' section is displayed. It includes a form with the following fields: 'Are you a?' with radio buttons for 'Student' and 'Recruiter' (selected), 'First Name', 'Last Name', 'Email', 'Contact Number', 'Gender' with radio buttons for 'Male' and 'Female' (selected), 'Organisation', 'Address', 'Password', and 'Confirm Password'. A green 'Sign Up' button is located at the bottom of the form.

Fig 5: Recruiter Registration/login Form is displayed

LINKEDIN
Search Jobs
Post Jobs
SignOut

Post Job Requirement

Company Information

Company Name

Company Website

Company Email

Job Information

Job Type

☒ Full Time
☐ Part Time

Location

Required Course

Description

Post Job

Fig 6: Recruiter's Job Post.

LINKEDIN
Search Jobs
Post Jobs
SignOut

Search Jobs

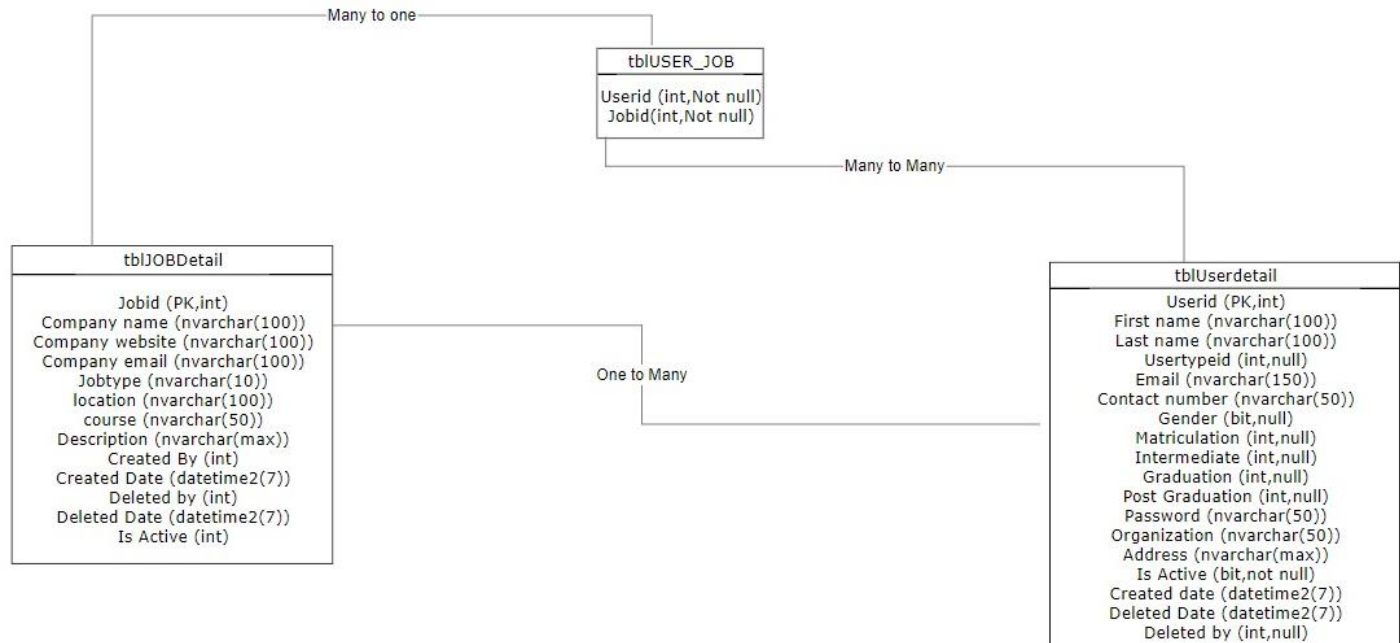
Show 10 entries
Search:

Company Name	Website	Company Email	Job Type	Course	Location	
cisco	cisco	cisco@gmail.com	Fulltime	begumpet	MBA	
deloitte	www.deloitte.com	dbz@deloitte.com	Parttime	Goa	MSc	
qwerty	www.q.com	q@q.com	Fulltime	1.18.1	MBA	
TCS	www.tcs.com	xyz@tcs.com	Fulltime	hyderabad	MBA	
TCS	www.tcs.com	xyz@tcs.com	Fulltime	hyderabad	MBA	
wipro	www.wipro.com	qw@wipro.com	Fulltime	pune	MBA	

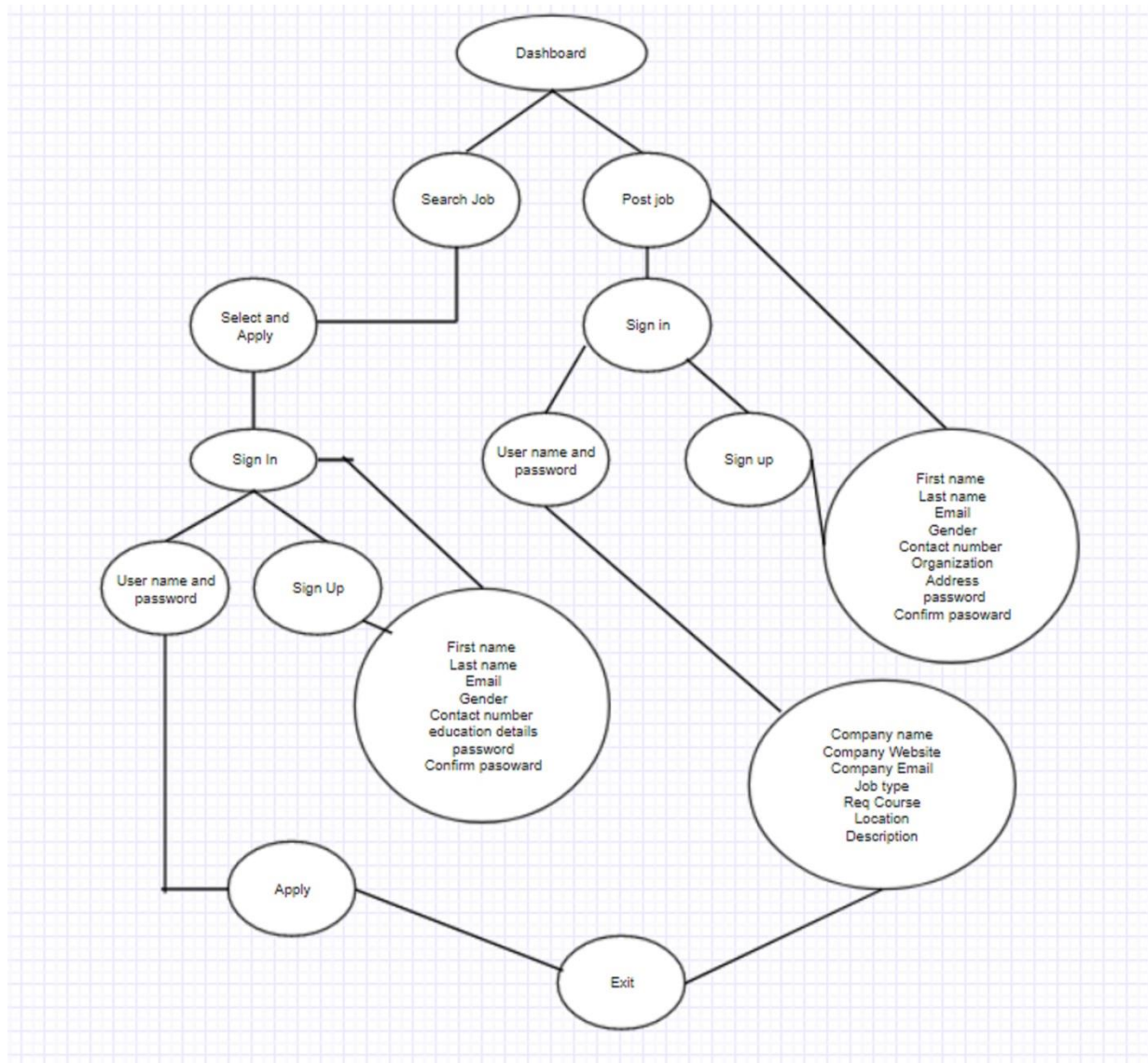
Showing 1 to 6 of 6 entries
Previous
1
Next

Fig 7: Recruiter Jobs.

Database Design:



E R Diagram



UML Design

Class Diagram

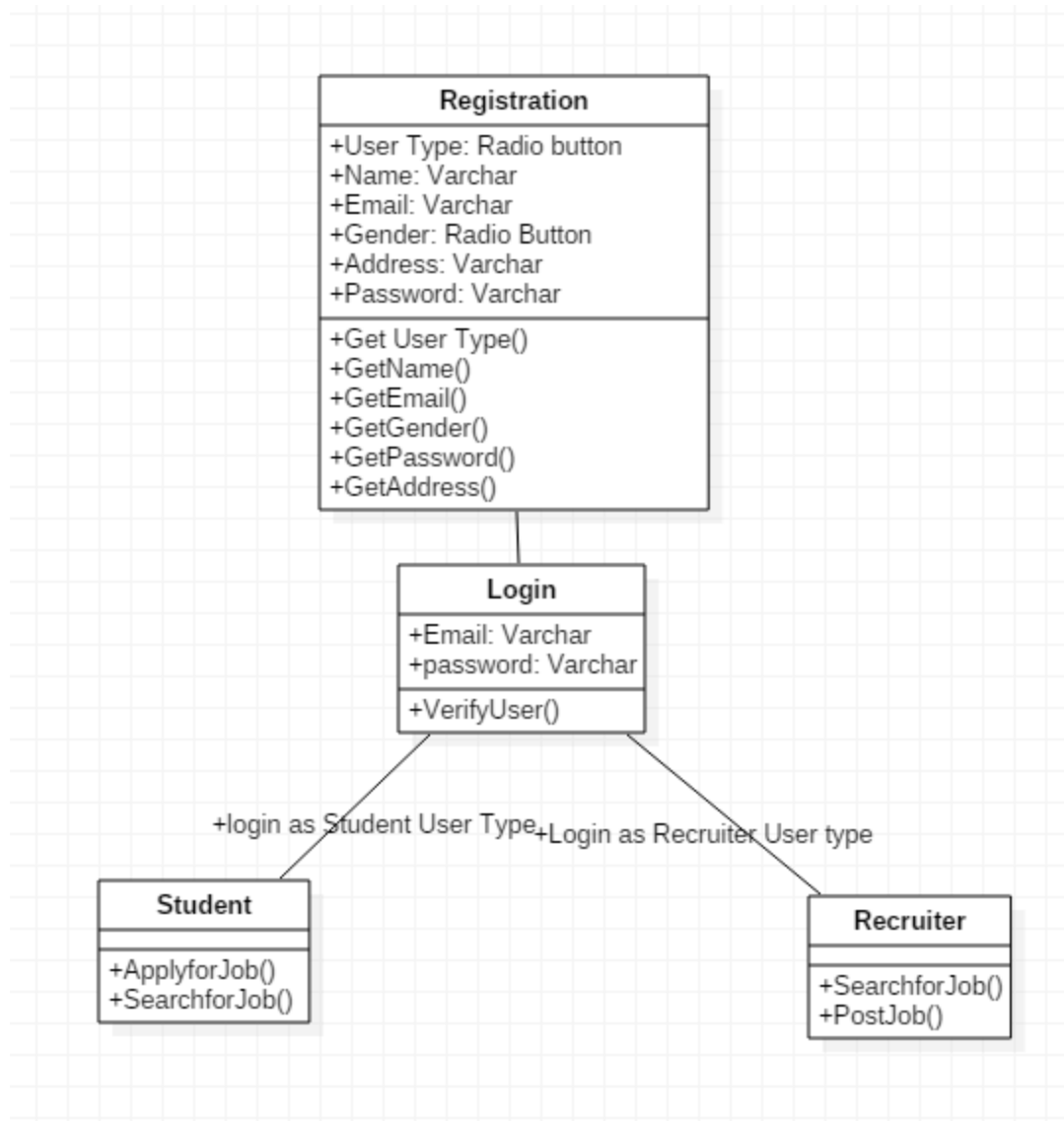


Figure-1 Class Diagram

Activity Diagram

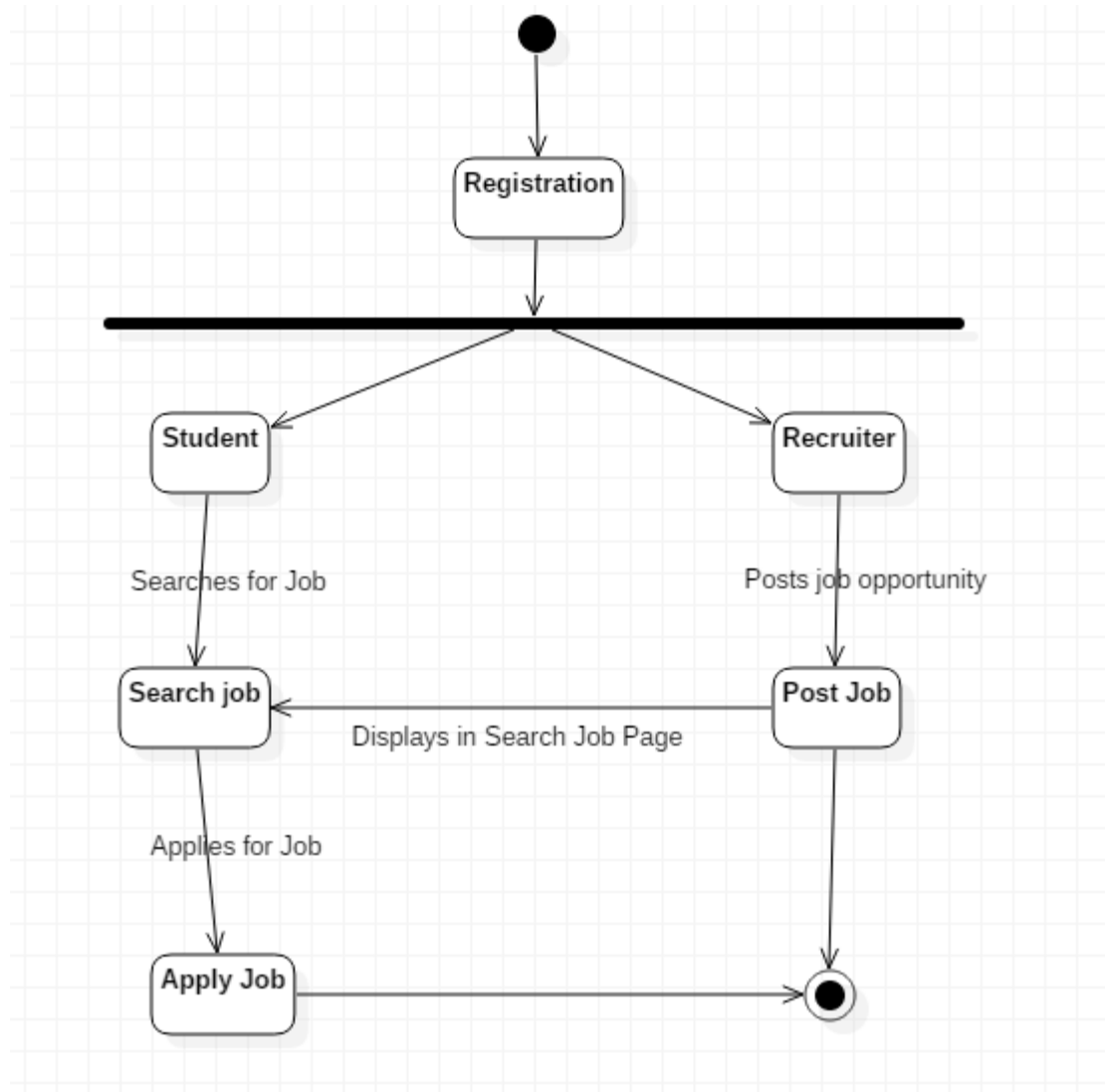


Figure-2 Activity Diagram

Sequence diagram

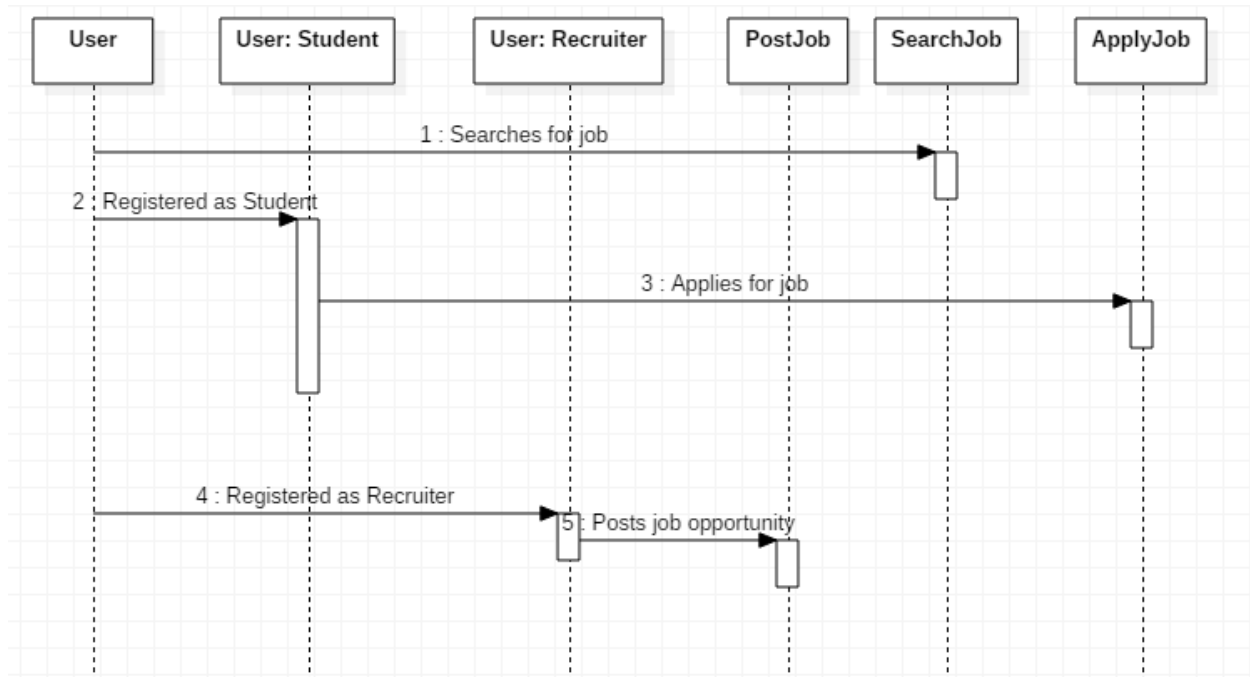


Figure-3 Sequence Diagram

Use Case Diagram

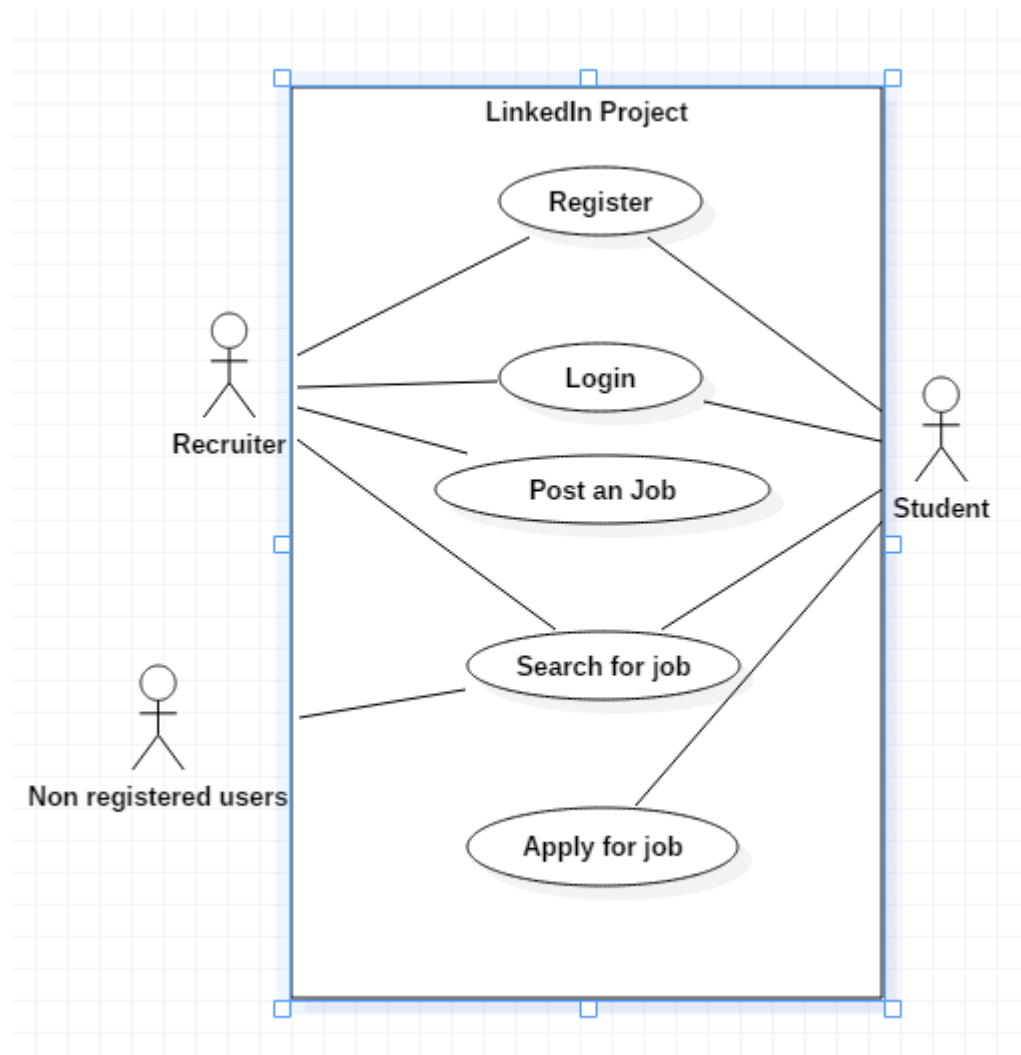


Figure-4 Use Case Diagram

Agile process

Agile software advancement alludes to a gathering of programming improvement approaches in light of iterative advancement, where prerequisites and arrangements advance through the joint effort between self-sorting out cross-practical groups. Agile procedures, for the most part, advance a trained project management process that supports visit review and adjustment, an initiative rationality that energizes cooperation, self-association and responsibility, an arrangement of designing accepted procedures proposed to take into account fast conveyance of astounding programming, and a business approach that adjusts improvement to client needs and friends objectives. Agile development alludes to any improvement procedure that is lined up with the ideas of the Agile Manifesto.

For our PROJECT, we had Dr. Simon Moss Associate Professor in Psychology as our customer. We had the initial meeting was devoted towards necessities and after that, we created client stories from that point.

Iteration 1: Working prototype with working functionalities as providing different registration forms for Student and recruiter..

Iteration 2: Login Credentials verification process and Search bar to filter list of job opportunities

Iteration 3: Sends Student to Apply job and Recruiter to post job opportunity.



User story Cards

As a **client**, I want a database to store all posted job details, Student details, Recruiter Details. Every student will be asked for all his percentages in his educational career which will be stored in the database and be visible to the recruiters who wanted to select the student who is at his

best performance. In this web application recruiter will be asked to specify job field in order to select student on his specific interest.

As a **user**, I need a web application which searches for the relevant job, post a Job, Applies for Job. Every student must register in the web application to see the jobs posted. Students need to post their academic career while registering through the process. Recruiter will select the student on his educational background.

Pair Programming:

Pair Programming is an agile technique where two programmers work at a single location. Although pair programming increases the man hours that are needed to deliver code, but it reduces the defects in the code. Two programmers perceive the problem in different ways, and have different experiences and offer different solutions to the problem which helps in increasing the quality of the design.

Module	Description	Pair Programmers
UI development	Complete Design of the user interface.	Rakesh, Ajay kumar
Database Design	Design of the Database	Rakesh, Soumith.
Website Development	Complete design of the website according to requirements.	Soumith, Ajay Kumar.

Testing

Testing:

Negative Testing:

Tested Scenarios:

- To verify non- registered user can apply / post an job.(Non registered user cannot apply/post Job)
- To verify Registered User(Student) can Post an Job (Student cannot Post an Job).
- To verify Registered User(Recruiter) can apply for Job (Recruiter cannot apply for an job).

Positive Testing:

Tested Scenarios:

- To verify non- registered user can Search for job.
- To verify Registered User(Recruiter) can Post an Job.
- To verify Registered User(Student) can apply for Job.

Test cases

Test Case ID	Test Scenario	Test Steps	Example Data	Expected Results	Actual Results	Pass/Fail
TC01	Verify email	<ol style="list-style-type: none"> 1. Go to website. 2. Sign up. 3. Enter Email address. 4. Click Submit. 5. Verification email sent. 	abc@gmail.com	Email is verified.	Email is verified.	Pass
TC02	Verify email	<ol style="list-style-type: none"> 1. Go to website. 2. Sign up. 3. Enter Email address. 4. Click Submit. 5. Verification email sent. 	abc@gmail.com	Email is verified.	Email is not verified.	Fail
TC03	Verify login	<ol style="list-style-type: none"> 1. Go to website 2. Click on Login. 3. Enter email and password 4. Click submit. 	abc@gmail.com *****	Login successful	Login successful	Pass
TC04	Verify Login	<ol style="list-style-type: none"> 1. Go to website 2. Click on Login. 3. Enter email and password 4. Click submit. 	abc@gmail.com *****	Login successful	Login unsuccessful	Fail
TC05	Add a Placement	<ol style="list-style-type: none"> 1. Go to website. 2. Recruiter Enter Email. 3. Enter Password. 4. Click Submit. 5. Enter all 	abc@gmail.com ***** Company name: Cisco Company Email:	Job successfully posted	Job successfully posted	Pass

		details. 6. Click on Post Job.	abc@ Companyname.com Job Type: Fulltime Required Course: CCNA			
TC06	Add a Placement	1. Go to website. 2. Recruiter Enter Email. 3. Enter Password. 4. Click Submit. 5. Enter all details. 6. Click on Post Job.	abc@gmail.com ***** Company name: Cisco Company Email: abc@ Companyname.com Job Type: Fulltime Required Course: CCNA	Job successfully posted	Job successfully posted	Fail
TC07	Search For a Job	1. Go to website. 2. Enter Email and password. 3. Click Sign in 4. Enter Keywords. 5. Click enter.	abc@gmail.com Cisco	Jobs retrieved successfully	Jobs retrieved successfully.	Pass
TC08	Search For a Job	1. Go to website. 2. Enter Email and password. 3. Click Sign in 4. Enter Keywords. 5. Click enter.	abc@gmail.com Cisco	Jobs retrieved successfully	Jobs not retrieved.	Fail

CODE

SignIn.aspx

```
<%@ Page Title="" Language="C#" MasterPageFile="~/MainSite.Master" AutoEventWireup="true"
CodeBehind="SignIn.aspx.cs" Inherits="LinkedInPortal.Register.SignIn" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">
    <script>
        $(document).ready(function () {
            $('#liSignIn').addClass('active');
            $('#liSearchJob').removeClass('active');
            $('#liPostJob').removeClass('active');
        });
    </script>
</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">
    <div class="container">
        <div class="card card-container">
            <!--  -->
            
            <p id="profile-name" class="profile-name-card"></p>
        <div class="container-fluid">
            <!--<h3>Sign In</h3>-->
            <hr />
            <div class="row">
                <div class="col-md-2">
                </div>
                <div class="col-md-8">
                    <div class="form-horizontal">
                        <div class="form-group">
```



```
</asp:Content>
```

SignIn.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace LinkedInPortal.Register
{
    public partial class SignIn : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {

        }

        protected void btnSignIn_Click(object sender, EventArgs e)
        {
            DataTable dtdData = LinkedInPortalBAL.Register.LoginDetail(txtUsername.Text, txtPassword.Text);
            if (dtdData.Rows.Count > 0)
            {
                Session["UserID"] = dtdData.Rows[0]["UserId"].ToString();
                Session["Username"] = dtdData.Rows[0]["Firstname"].ToString() + dtdData.Rows[0]["Lastname"].ToString();
                Session["UserTypeId"] = dtdData.Rows[0]["UserTypeId"].ToString();
                Response.Redirect("../Home/Dashboard.aspx");
            }
        }
    }
}
```

SignUp.aspx

```
<%@ Page Title="" Language="C#" MasterPageFile="~/MainSite.Master" AutoEventWireup="true"
CodeBehind="SignUp.aspx.cs" Inherits="LinkedInPortal.Register.SignUp" %>
```

```

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">
    <script>
        $(document).ready(function () {
            $('#liSignIn').addClass('active');
            $('#liSearchJob').removeClass('active');
            $('#liPostJob').removeClass('active');

            $('#ContentPlaceHolder1_rdbtnRegisterOption_1').change(function () {
                $('#divStudent').hide();
                $('#divRecruiter').show();
            });
            $('#ContentPlaceHolder1_rdbtnRegisterOption_0').change(function () {
                $('#divStudent').show();
                $('#divRecruiter').hide();
            });

        });
    </script>

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">
    <div class="container-fluid">
        <h3>Sign Up</h3>
        <hr />
        <div class="row">
            <div class="col-md-2">
            </div>
            <div class="col-md-8">
                <div class="form-horizontal">
                    <div class="form-group">
                        <div class="col-md-2 control-label">
                            Are you a?
                        </div>
                        <div class="col-md-7">
                            <asp:RadioButtonList ID="rdbtnRegisterOption" runat="server" RepeatDirection="Horizontal">
                                <asp:ListItem Text="Student" Value="1" Selected="True"></asp:ListItem>
                                <asp:ListItem Text="Recruiter" Value="2"></asp:ListItem>
                            </asp:RadioButtonList>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>

```

[illegible]

```

        </asp:RadioButtonList>
    </div>
</div>
<div id="divStudent">
    <div class="form-group">
        <div class="col-md-2 control-label">
            Matriculation(%)
        </div>
        <div class="col-md-4">
            <asp:TextBox ID="txtMatriculation" runat="server" CssClass="form-control"></asp:TextBox>
        </div>
        <div class="col-md-2 control-label">
            Intermediate(%)
        </div>
        <div class="col-md-4">
            <asp:TextBox ID="txtIntermediate" runat="server" CssClass="form-control"></asp:TextBox>
        </div>
    </div>
    <div class="form-group">
        <div class="col-md-2 control-label">
            Graduation(%)
        </div>
        <div class="col-md-4">
            <asp:TextBox ID="txtGraduation" runat="server" CssClass="form-control"></asp:TextBox>
        </div>
        <div class="col-md-2 control-label">
            Post Graduation(%)
        </div>
        <div class="col-md-4">
            <asp:TextBox ID="txtPostGraduation" runat="server" CssClass="form-control"></asp:TextBox>
        </div>
    </div>
</div>
<div id="divRecruiter" style="display:none;">
    <div class="form-group">
        <div class="col-md-2 control-label">
            Organisation
        </div>
        <div class="col-md-4">

```



```
</asp:Content>
```

SignUp.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace LinkedInPortal.Register
{
    public partial class SignIn : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {

        }

        protected void btnSignIn_Click(object sender, EventArgs e)
        {
            DataTable dtdData = LinkedInPortalBAL.Register.LoginDetail(txtUsername.Text, txtPassword.Text);
            if (dtdData.Rows.Count > 0)
            {
                Session["UserID"] = dtdData.Rows[0]["UserId"].ToString();
                Session["Username"] = dtdData.Rows[0]["Firstname"].ToString() + dtdData.Rows[0]["Lastname"].ToString();
                Session["UserTypeId"] = dtdData.Rows[0]["UserTypeId"].ToString();
                Response.Redirect("../Home/Dashboard.aspx");
            }
        }
    }
}
```

Post.aspx

```
<%@ Page Title="" Language="C#" MasterPageFile="~/MainSite.Master" AutoEventWireup="true"
CodeBehind="Post.aspx.cs" Inherits="LinkedInPortal.Job.Post" %>
```

[illegible]

```

    </div>
</div>
<div class="form-group">
    <div class="col-md-2 control-label">
        Company Email
    </div>
    <div class="col-md-4">
        <asp:TextBox ID="txtEmail" runat="server" CssClass="form-control"></asp:TextBox>
    </div>
    <div class="col-md-2 control-label">
    </div>
    <div class="col-md-4">
    </div>
</div>
<div class="form-group">
    <div class="col-md-12">
        <h4>Job Information</h4>
        <hr />
    </div>
</div>
<div class="form-group">
    <div class="col-md-2 control-label">
        Job Type
    </div>
    <div class="col-md-4">
        <asp:RadioButtonList ID="rdbtnJobType" runat="server" RepeatDirection="Horizontal">
            <asp:ListItem Text="Full Time" Value="Fulltime" Selected="True"></asp:ListItem>
            <asp:ListItem Text="Part Time" Value="Parttime"></asp:ListItem>
        </asp:RadioButtonList>
    </div>
    <div class="col-md-2 control-label">
        Location
    </div>
    <div class="col-md-4">
        <asp:TextBox ID="txtLocation" runat="server" CssClass="form-control"></asp:TextBox>
    </div>
</div>

```



```

using System.Collections.Generic;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace LinkedInPortal.Job
{
    public partial class Post : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            if (Convert.ToString(Session["UserID"]) == "")
            {
                Response.Redirect("../Register/SignIn.aspx");
            }
        }

        protected void btnPost_Click(object sender, EventArgs e)
        {
            string strJobType = "";
            for (int i = 0; i < rdbtnJobType.Items.Count; i++)
            {
                if (rdbtnJobType.Items[i].Selected)
                {
                    strJobType = rdbtnJobType.Items[i].Value;
                }
            }

            DataTable dtData = LinkedInPortalBAL.Job.PostJob(txtCompanyName.Text, txtWebsite.Text, txtEmail.Text,
            strJobType, txtLocation.Text, ddlCourse.SelectedItem.Value, txtDescription.Text, Convert.ToInt16(Session["UserID"]));
            if (dtData.Rows.Count > 0)
            {
                Response.Redirect("../Job/Search.aspx");
            }
        }
    }
}

```

Search.aspx

```
<%@ Page Title="" Language="C#" MasterPageFile="~/MainSite.Master" AutoEventWireup="true"
CodeBehind="Search.aspx.cs" Inherits="LinkedInPortal.Job.Search" %>
```

```
<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">
```

```
<script type="text/javascript">
```

```
$(document).ready(function () {
    var hdnUserTypeID = $('#ContentPlaceHolder1_hdnUserTypeID').val();
    $.ajax({
        type: "GET",
        dataType: "json",
        url: "../Service/JobWebService.asmx/GetJobList",
        success: function (data) {
            var datatableVariable = $('#tblJobList').DataTable({
                data: data,
                columns: [
                    { 'data': 'CompanyName' },
                    { 'data': 'CompanyWebsite' },
                    { 'data': 'CompanyEmail' },
                    { 'data': 'JobType' },
                    { 'data': 'Location' },
                    { 'data': 'Course' },
                    {
                        'data': 'JobID',
                        "render": function (data, type, row, meta) {
                            if (type === 'display' && hdnUserTypeID != 2) {
                                data = '<a href="#" onclick="return checkForApply(' + data + ')">Apply</a>';
                            } else { data = "" }
                            return data;
                        }
                    }
                ]
            });
        }
    });
});
```

```

function checkForApply(data1) {
    var hdnUserTypeID = $('#ContentPlaceHolder1_hdnUserTypeID').val();
    if (hdnUserTypeID != "") {
        $.ajax({
            type: "GET",
            dataType: "json",
            data: { 'UserID': hdnUserTypeID, 'JobID': data1 },
            url: "../Service/JobWebService.aspx/ApplyJob",
            success: function (data) {
                alert('Job applied successfully.');
```

```

            },
            error: function (result, error) {
                alert('Something went wrong. Please try again later.');
```

```

                return false;
            }
        });
    } else {
        alert('Please login to apply for the job.');
```

```

    }
}
</script>
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">
    <div class="container-fluid">
        <h3>Search Jobs</h3>
        <hr />
        <div class="row">
            <div class="col-md-2">
                <asp:HiddenField ID="hdnUserTypeID" runat="server" />
            </div>
            <div class="col-md-8">
                <table id="tblJobList" class="table table-striped table-bordered">
                    <thead>
                        <tr>
                            <th>Company Name</th>
                            <th>Website</th>
                            <th>Company Email</th>

```

```

        <th>Job Type</th>
        <th>Course</th>
        <th>Location</th>
        <th></th>
    </tr>
</thead>
</table>
</div>
<div class="col-md-2">
</div>
</div>
</div>
</asp:Content>

```

Search.aspx.cs

```

using System;
using System.Collections.Generic;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace LinkedInPortal.Job
{
    public partial class Search : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            if (Session["UserId"] != null)
                hdnUserId.Value = Session["UserId"].ToString();
            else
                hdnUserId.Value = "";
        }
    }
}

```


User Registration table query.

```
CREATE TABLE [dbo].[tblUser](
    [UserId] [int] IDENTITY(1,1) NOT NULL,
    [Firstname] [nvarchar](100) NOT NULL,
    [Lastname] [nvarchar](100) NULL,
    [UserTypeId] [int] NULL,
    [Email] [nvarchar](150) NULL,
    [ContactNumber] [nvarchar](50) NULL,
    [Gender] [bit] NULL,
    [Matriculation] [int] NULL,
    [Intermediate] [int] NULL,
    [Graduation] [int] NULL,
    [PostGraduation] [int] NULL,
    [Password] [nvarchar](50) NULL,
    [Organisation] [nvarchar](50) NULL,
    [Address] [nvarchar](max) NULL,
    [IsActive] [bit] NOT NULL,
    [CreatedDate] [datetime2](7) NULL,
    [DeletedDate] [datetime2](7) NULL,
    [DeletedBy] [int] NULL,
    CONSTRAINT [PK_tblUser] PRIMARY KEY)
```

Post Job table Query:

```
CREATE TABLE [dbo].[tblJobDetail](
    [JobID] [int] IDENTITY(1,1) NOT NULL,
    [CompanyName] [nvarchar](100) NULL,
    [CompanyWebsite] [nvarchar](100) NULL,
    [CompanyEmail] [nvarchar](100) NULL,
    [JobType] [nvarchar](10) NULL,
```

```

[Location] [nvarchar](100) NULL,
[Course] [nvarchar](50) NULL,
[Description] [nvarchar](max) NULL,
[CreatedBy] [int] NULL,
[CreateDate] [datetime2](7) NULL,
[DeletedBy] [int] NULL,
[DeletedDate] [datetime2](7) NULL,
[IsActive] [bit] NULL,
CONSTRAINT [PK_tblJobDetail] PRIMARY KEY)

```

Mainsite.master

```

<%@ Master Language="C#" AutoEventWireup="true" CodeBehind="MainSite.master.cs"
Inherits="LinkedInPortal.MainSite" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css" />
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap-theme.min.css" />
    <link href="Content/Site.css" rel="stylesheet" />
    <script src="https://code.jquery.com/jquery-1.12.4.js"></script>
    <script src="https://code.jquery.com/ui/1.12.1/jquery-ui.min.js"></script>

    <asp:ContentPlaceHolder ID="head" runat="server">
        <script>
            $(document).ready(function () {

                });
        </script>
    </asp:ContentPlaceHolder>
</head>
<body>

```

```

<form id="form1" runat="server">
  <nav class="navbar navbar-inverse">
    <div class="container-fluid">
      <!-- Brand and toggle get grouped for better mobile display -->
      <div class="navbar-header">
        <button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#bs-example-
navbar-collapse-1" aria-expanded="false">
          <span class="sr-only">Toggle navigation</span>
          <span class="icon-bar"></span>
          <span class="icon-bar"></span>
          <span class="icon-bar"></span>
        </button>
        <a class="navbar-brand" href=" ../Home/Dashboard.aspx">LINKEDIN</a>
      </div>

      <!-- Collect the nav links, forms, and other content for toggling -->
      <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
        <ul class="nav navbar-nav">
          <li id="liSearchJob" class="active"><a href=" ../Job/Search.aspx">Search Jobs <span class="sr-
only">(current)</span></a></li>
          <li id="liPostJob" runat="server"><a href=" ../Job/Post.aspx">Post Jobs</a></li>
        </ul>
        <ul class="nav navbar-nav navbar-right">
          <li id="liSignIn">
            <asp:LinkButton ID="lnkSignIn" runat="server" OnClick="lnkSignIn_Click">SignIn</asp:LinkButton></li>
            <%-- <li class="dropdown">
              <a href="#" class="dropdown-toggle" data-toggle="dropdown" role="button" aria-haspopup="true" aria-
expanded="false">Dropdown <span class="caret"></span></a>
              <ul class="dropdown-menu">
                <li><a href="#">Action</a></li>
                <li><a href="#">Another action</a></li>
                <li><a href="#">Something else here</a></li>
                <li role="separator" class="divider"></li>
                <li><a href="#">Separated link</a></li>
              </ul>
            </li>--%>
          </ul>

```

```

        </div>
        <!-- /.navbar-collapse -->
    </div>
    <!-- /.container-fluid -->
</nav>
<div>
    <asp:ContentPlaceHolder ID="ContentPlaceHolder1" runat="server">
        </asp:ContentPlaceHolder>
    </div>
</form>
<link rel="stylesheet" type="text/css" href="https://cdn.datatables.net/u/dt/dt-1.10.12/datatables.min.css" />
<script type="text/javascript" src="https://cdn.datatables.net/u/dt/dt-1.10.12/datatables.min.js"></script>
<div style="border: 1px solid gray; width: 100%; bottom: 0; margin-bottom: 20px; padding: 20px; position: absolute;
left: 0; color: gainsboro; background-color:black">

    </div>
</body>

</html>

```

Dashboard.aspx

```

<%@ Page Title="" Language="C#" MasterPageFile="~/MainSite.Master" AutoEventWireup="true"
CodeBehind="Dashboard.aspx.cs" Inherits="LinkedInPortal.Dashboard.Dashboard" %>
<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

</asp:Content>

```

Dashboard.aspx.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

```

```

namespace LinkedInPortal.Dashboard
{
    public partial class Dashboard : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {

        }
    }
}

```

References:

1. Anon, (2017). [online] Available at: <https://www.linkedin.com/pulse/how-does-waterfall-agile-work-together-hybrid-approach-debarshi-roy/> [Accessed 9 Oct. 2017].
2. Sharma, L. (2017). *Positive Vs Negative Testing*. [online] Toolsqa.com. Available at: <http://toolsqa.com/software-testing/positive-vs-negative-testing/> [Accessed 9 Oct. 2017].
3. BV, D. (2017). *Introduction to database design*. [online] Datanamic.com. Available at: <http://www.datanamic.com/support/lt-dez005-introduction-db-modeling.html> [Accessed 9 Oct. 2017].