GROUP ASSIGNMENT (Sprint 1 Documentation)

LEVEL 5

COMP50001-2 – Commerical Computing 2

IF2161SE

Hand Out Date: 29th October 2021

Hand In Date: 16th November 2021

BY :

MOHAMED AZHAR MUSHARRAF [CB008866]

MOHANRAJ KAVINKUMAR NAIDU [CB008222]

MOHAMED ZAKEER MOHAMED FAIQUE INAAS [CB008515]

THARINDU DHARSHAKA KATHRIARACHCHI [CB007094]

Table of Contents

1.0 Introduction 4
2.0 Workload Matrix 5
3.0 Project Charter 9
3.1 Background 9
3.2 Goals 9
3.3 Scope 9
3.4 Key Stakeholders 9
3.5 Project Milestones 10
3.6 Project budget 10
3.6.1 Available hosting and Domain plans 10
4.0 The Work Break Down Structure 11
5.0 Sprint goal 12
6.0 Functions carried out 13
7.0 Time Estimation 14
7.1 AOA diagram 15
7.2 AON Diagram 16
7.3 Gantt Chart 17
8.0 Use Case Diagram and Specification 18
9.0 Activity Diagrams 28
9.1 Login Function 28
9.2 Register Function 29
9.3 Edit profile function 30
9.4 Profile verification 31
10.0 Implementation 32
10.1 Login function 32
10.2 Register Function (Applicant) 37
10.3 Edit Profile function (Applicant) 41
10.4 Profile verification function 45
11.0 Interfaces 46
11.1 Admin login page (Wireframe) 46
11.2 Admin login page (Actual View) 47
11.3 Applicant Profile Verification page for Admin (Wireframe) 48

11.4 Applicant Profile Verification page for Admin (Actual View) 49
11.5 Employer Profile verification page for Admin (Wireframe) 50
11.6 Employer Profile verification for Admin (Actual View) 51
11.7 Login page (Wireframe) 52
11.8 Login Page (Actual View) 53
11.9 Register for Applicants (Wireframe) 54
11.10 Register for Applicants (Actual View) 55
11.11 Register for Employer (Wireframe) 56
11.12 Register For Employer (Actual View) 57
11.13 My Profile page for Employer (Wireframe) 58
11.14 My profile page for Employer (Actual View) 59
11.15 My Profile page for Applicant (Wireframe) 60
11.16 My Profile page for Applicant (Actual view) 61
12.0 Test Cases 62
12.1 Log in [Admin] 62
12.2 Log in [Employer] 64
12.3 Log in [Applicant] 68
12.4 Register [Employer] 71
12.5 Register [Applicant] 74

1.0 Introduction

This document consists of everything related to **Futureseekers.lk** web app's first sprint, which has use case diagrams, activity diagrams, wireframes and some snips of relevant codes and web pages and etc.

The project files are uploaded in GitHub and can be accessed in this link below.

https://github.com/FaiqueMZM/futureseekerslk

This project is done using PHP, HTML, CSS etc. CodeIgniter 4 is used as a php framework during this project and the team has used MVC model architecture to build this website.

2.0 Workload Matrix

Tasks	Description	Assignee	Due date	Status	Effort (Days spent)
Use case specification	Use cases are used to explain the behavior of the system. Use cases Use cases will show actors of the system, their relationships.	Kavin	2021.09.01	Done	1 day
Drawing activity diagrams	An activity diagram helps explain a certain use case. It will explain the flow of activities through a system.	Kavin Musharraf	2021.09.04	Done	2 days

Designing wireframes	Wireframes are two-dimensional illustration of a page's interface. It helps when designing the web page.	Kavin Musharraf	2021.09.04	Done	3 days
Developing landing page and styling	Landing page is the first webpage of the website. This page will give the user choice of login or register.	Inaas Kavin	2021.09.06	Done	2 days
Developing login and registration functions with proper validations and styling	This function will allow user to register as an applicant, an employer and the employer can register his/her company	Inaas Kavin Musharraf	2021.09.13	Done	3 days
Developing basic admin portal	This is the main page of admin portal where admin will do	Inaas Kavin Musharraf	2021.09.13	Done	2 days

	website related taks				
Profile verification	This function will allow the admin to accept or reject profiles that are registered.	Inaas Kavin	2021.09.12	Done	2 days
Edit profile	This function will allow users to edit their profiles	Inaas Kavin Musharraf	2021.09.13	Done	2 days
Testing landing page	Checking if the buttons are working, if button clicks are leading user to correct pages, if the web site is working on mobile devices	Tharindhu	2021.09.14	Done	3 days.
Testing Registration and login	Checking if the data which are gotten from users are used and handled properly.	Tharindhu	2021.09.16	Done	2 days

Testing edit profile function	Checking if the data in databases changes when user edits their details	Tharindhu	2021.09.16	Done	3 days
Testing admin portal	Checking if the admin can enter the admin portal and do admin related tasks	Tharindhu	2021.09.16	Done	3 days
Testing profile verification function	Checking if the admin can accept or reject users from the website	Tharindhu	2021.09.16	Done	3 days

3.0 Project Charter

3.1 Background

This is a web platform that provides services for companies and job seekers. Companies can post a job opportunity while job seekers can apply for those job opportunities. Both companies and job seekers must create a user profile. There will be an administrator dashboard for administrative functions.

3.2 Goals

This platform aims to create a modern, efficient and effective hiring experience for both companies and job seekers.

3.3 Scope

A web platform that helps companies and job seekers to get connected easily.

3.4 Key Stakeholders

Role	Name
Product Owner	Tharanga Peiris

Business Analyst	Mohanraj Kavinkumar Naidu
Scrum Master	Mohamed Azhar Musharraf
Developer	Mohamed Zakkeer Mohamed Faique Inaas
Quality Assurance engineer	Tharindu Darshaka Kathriarachchi

Table 1 Key Stakeholders

3.5 Project Milestones

Project Milestones	Milestone Delivery
Project Concept	End of the second week
Sprint 01	End of the fifth week
Sprint 02	End of the eighth week
Sprint 03	End of the eleventh week
Final Documentation/ Live Release	End of the twelfth week

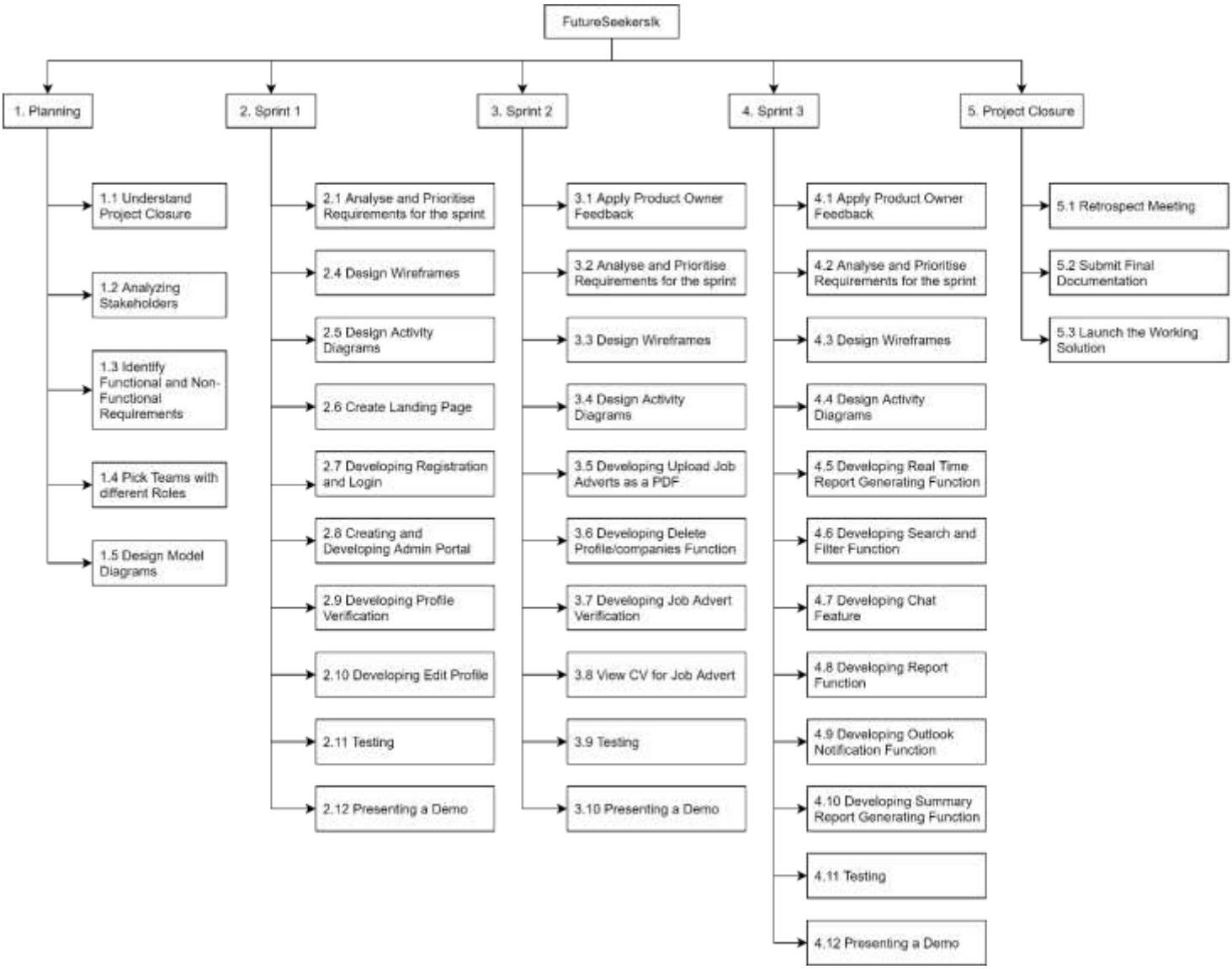
Table 2 Project Milestones

3.6 Project budget

3.6.1 Available hosting and Domain plans

Item	Cost
AWS hosting	$150[per month]
Bluehost dedicated hosting	$119[per month]
Azure Virtual Machines	$22[per month]
.com domain from the register.lk	LKR2,000[per year]
.lk Domain from the register.lk	LKR3,000[per year]

4.0 The Work Break Down Structure



5.0 Sprint goal

The development of **Futureseekers.lk** web app has three sprints, and this is the end of first sprint. This Sprint covered the most important part of this project which is putting the foundation for development process of second sprint. The goal of this sprint is to have a working registration/login system, A database with tables which will store all the user details and company details and properly styled pages.

6.0 Functions carried out

- Registration and login

  This website has 3 types of users, which are Job seeker, Employer and Admin. The admin details are entered into the database manually. There isn't a page for admin registration. As for job seekers and employers they can register for this website by going to registration page and filling a form and submitting. Employers and Applicants have separate registration forms. After registering to the website all the user details will be saved in the database. The registered users has be verified by the admin to be able to edit their details again. If a user is already registered, they can use the username and password to login by entering the correct credentials.

- Developing Admin portal

  This website has admins to keep the website a safe place. The admin portal will have different features such as verifying profile, verifying job advertisements. In this sprint we have just created the interface without the functions.

- Profile verification

  This function is only available to admins. The admin can accept, reject or delete. The users won't be able to change anything in their profile until the admin verifies their profile.

- Edit profile

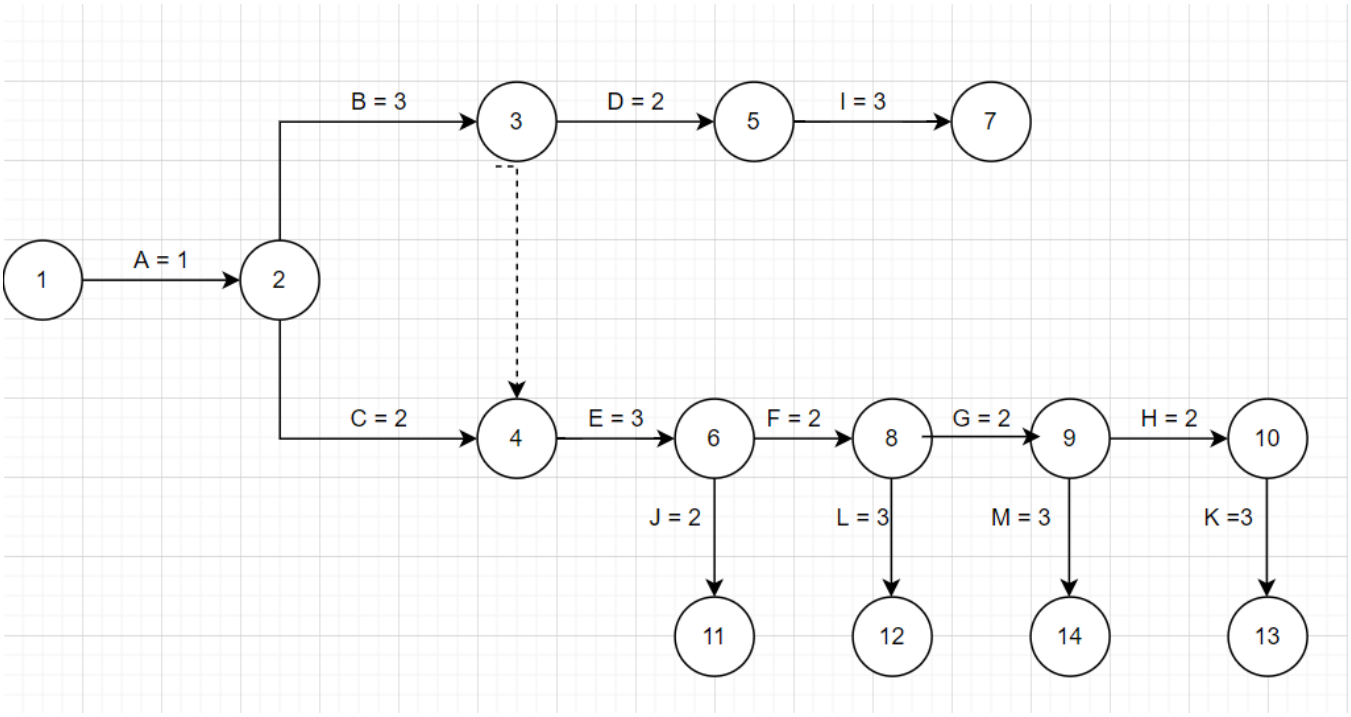
  After verification every user has the ability to change their details such as contact number, username etc.

7.0 Time Estimation

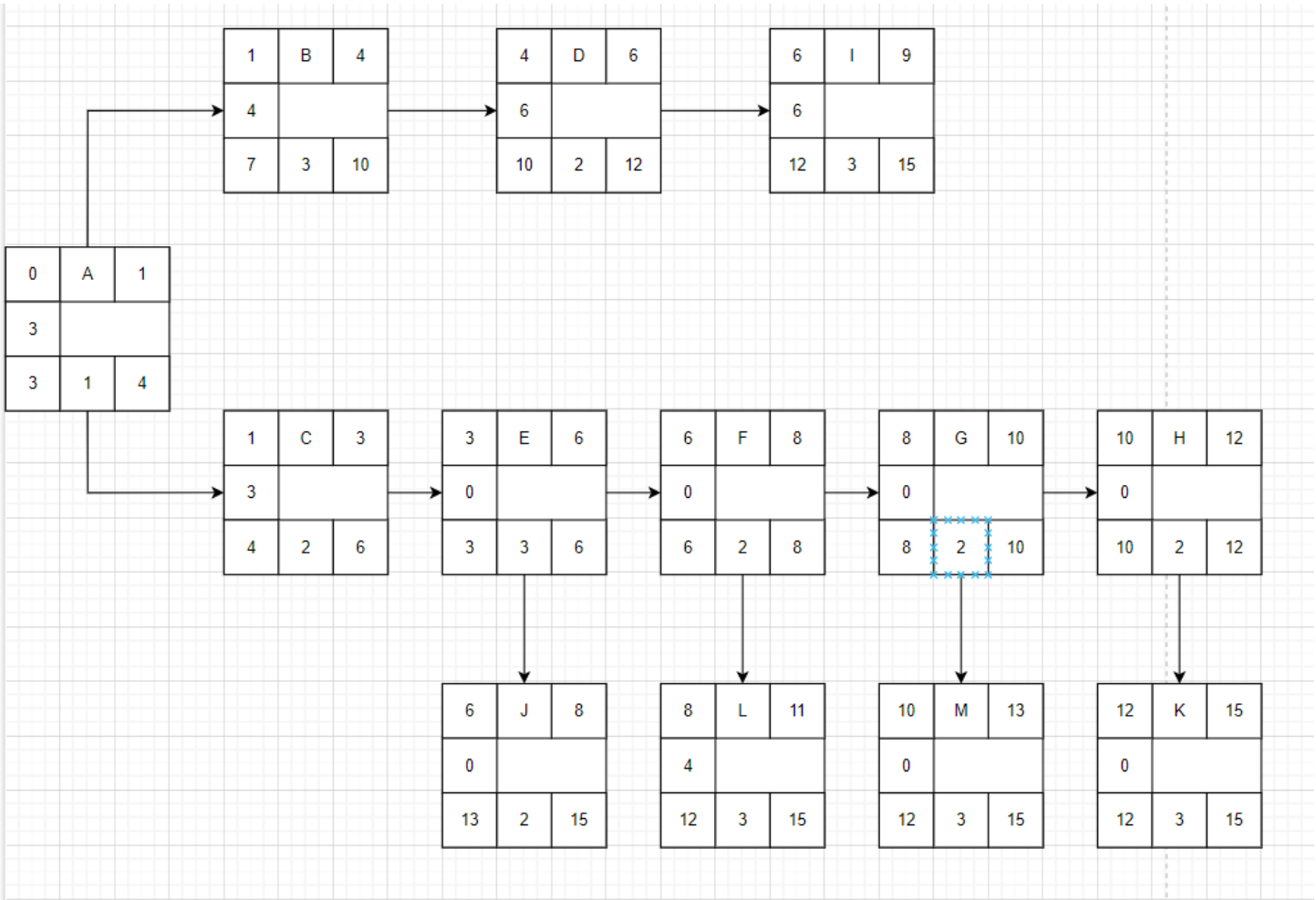
Activity Id	Activity	Predecessors	Duration
A	Use case specification	_	1 day
B	Designing wireframes	A	3 days
C	Drawing activity diagrams	A	2 days
D	Developing landing page and styling	B	2 days
E	Developing login and registration functions with proper validations and styling	C, B	3 days
F	Developing basic admin portal	E	2 days
G	Profile verification	F	2 days
H	Edit profile	G	2 days
I	Testing landing page	D	3 days.
J	Testing Registration and login	E	2 days
K	Testing edit profile function	H	3 days
L	Testing admin portal	F	3 days

M	Testing profile verification function	G	3 days

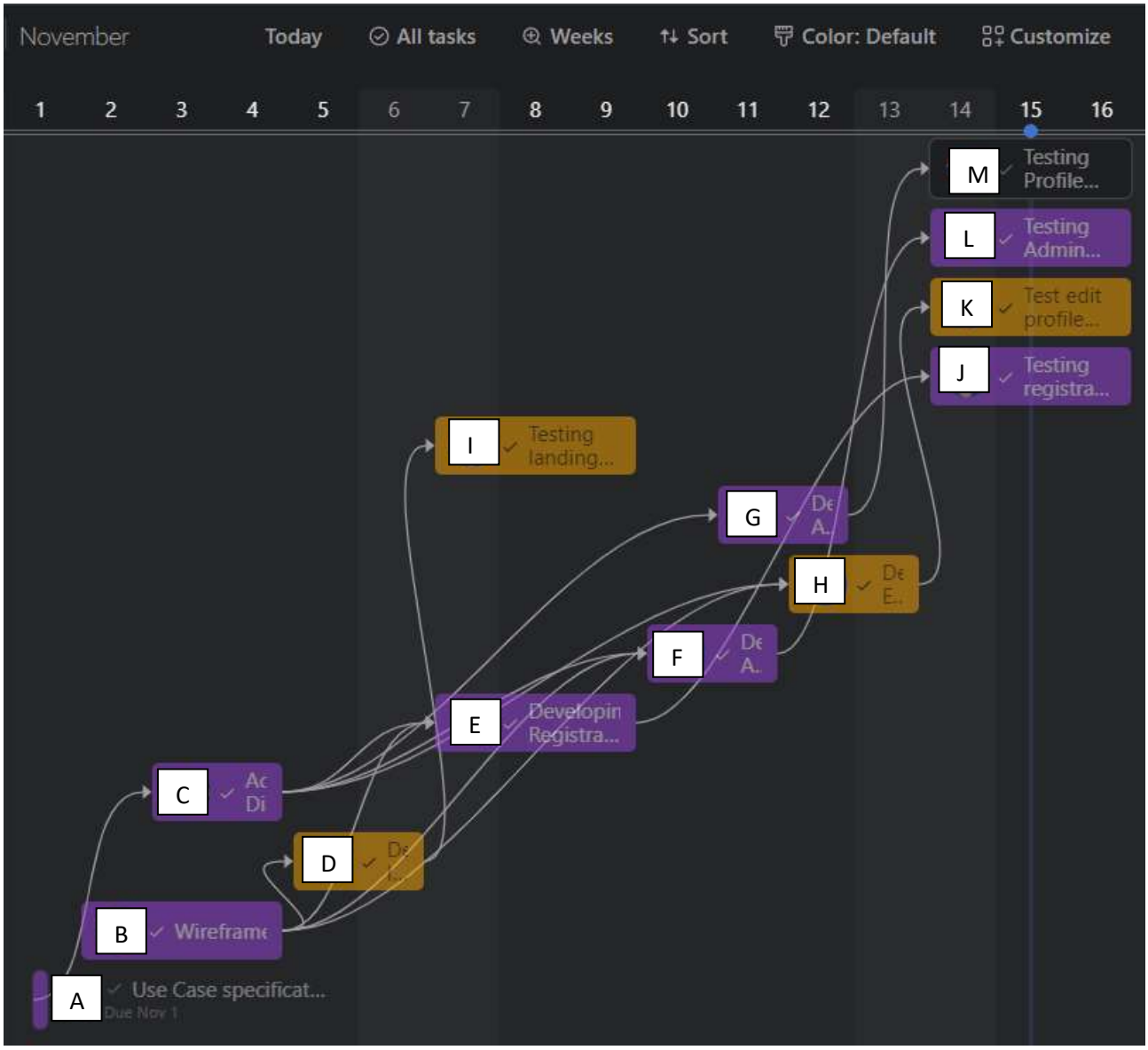
7.1 AOA diagram



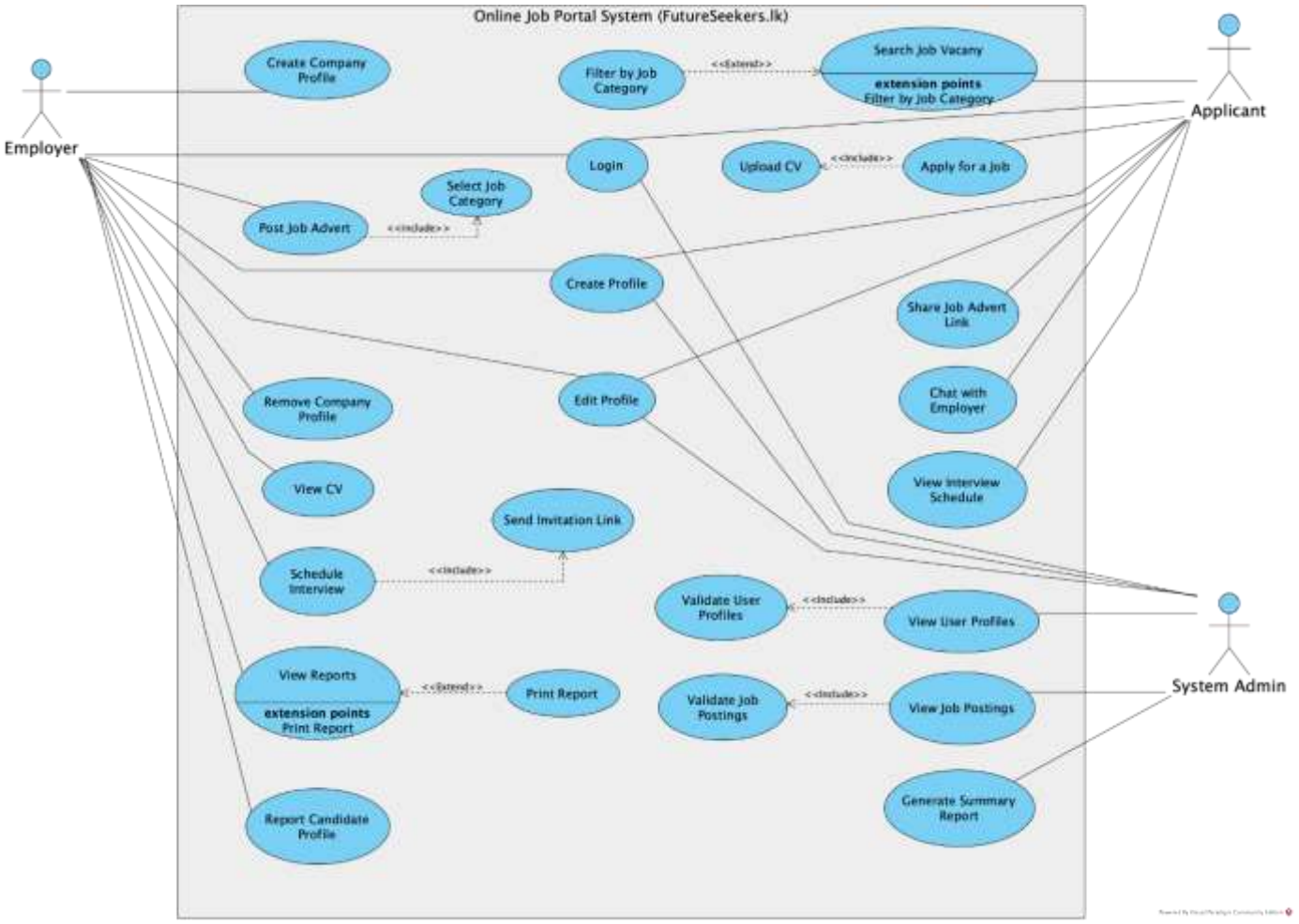
7.2 AON Diagram



7.3 Gantt Chart



8.0 Use Case Diagram and Specification



Use Case Name	**Create Profile**
Description	Allows the Employer of a company to register himself into FutureSeekers.lk
Primary Actor	Employer
Pre-Conditions	None
Basic Flow	1. Use Case begins when the Employer clicks on the **'Register Now'** link from the Login Page of FutureSeekers.lk. (There would be 2 buttons in the login page, one for Applicant and one for Employer. 2. The user will be directed to the Employer registration page. 3. The Employer must enter his details such as Name, Contact Number, job position, email and the name of the Company. 4. In the Registration page there should be a message for the user: "After Registration, you're required to pay a fee of Rs 1,000 and send the receipt via email to futureseekers@gamil.com. Our Team will verify your profile within 24 hours and would send you a mail." 5. After the user clicks on **'Register Now'** button, the details are validated and are updated to the database and a success message must be shown on the page.
Extensions	After **'Register Now'** button is clicked, if the validation fails: 1. System would display a validation error message, for example 'Fields cannot be empty'. 2. Use case ends
Post Conditions	The User can login to the system and view the profile details but cannot edit the profile details unless the profile verification is completed by the Admin.
Optional Information	The Employer can only make a company profile **once the registration fee receipt is validated by FutureSeekerslk.**

Use Case Name	**Login**
Description	Allows the Employer to login to his account in order to perform his operations such as posting a job advert.
Primary Actor	Employer
Pre-Conditions	Employer must pay the registration fee and send the receipt to FutureSeekers.lk team, and the profile must be validated successfully for the profile to be registered in the system.
Basic Flow	1. Use Case begins when the Employer enters the valid credentials and clicks on the **'Login'** Button on the FutureSeekers.lk login page. 2. The username and the password are validated and queried with the database to find a match. 3. After successful validations, the User will receive a dialog box, 'Welcome Back!' and will be directed to the homepage (All Jobs Page) which would complete the use case.
Extensions	After **'Login'** button is clicked, if the user doesn't exist: System would display a validation error message, for example 'User Doesn't Exist!'. Use case ends.
Post Conditions	The Employer can only perform their functions once the profile is **verified by the FutureSeekers.lk team**. If the account verification is still on pending status, then the Employer temporarily gets limited privileges where he/she will **only be able to view the profile information.**
Optional Information	If the Employer is logging in for the first time, he/she will be required to create a **company profile** before they can post any job advertisements.

Use Case Name	**Create Company Profile**
Description	Allows the Employer to create his/her company profile.
Primary Actor	Employer
Pre-Conditions	Employer's profile must be successfully validated by the FutureSeekers.lk team and user must be logged in.
Basic Flow	1. Use Case begins when the user logs in successfully. 2. If the user logs in for the first time, then the user will be set to the 'My Profile' Page and prompted to fill in the details of his/her company such as Contact No, Email, Company Logo. 3. Once **'Save Changes'** button is clicked the details are sent to the database and the use case would with a Success Message.
Extensions	After **'Update Profile** button is clicked if the validation fails: 3. System would display a validation error message, for example 'Fields cannot be empty'. 4. Use case ends
Post Conditions	The Employer can view the Company Profile and will now be able to post job advertisements.
Optional Information	None

Use Case Name	**Edit Profile**
Description	Allows the Employer of a company to edit the company profile details or the employer details (In the case where the old employer leaves and a new one is required).
Primary Actor	Employer
Pre-Conditions	The Employer must be a verified user and logged in.
Basic Flow	1. Use case begins when the user clicks the **'My Profile'** button on the navigation bar. 2. The Company and the Employer details will be fetched from the database and displayed to the user. 3. The Employer will be able to update the details and once he/she has completed, would click on the **'Save Changes** Button. 4. The new details will be updated on the database and the use case would end with a Success Message.
Extensions	After **'Update Profile'** button is clicked, if the validation fails: 5. System would display a validation error message, for example 'Username already exists, Try a different one' 6. Use case ends
Post Conditions	The Employer would be able to view the updated profile details.
Optional Information	None

Use Case Name	**Create Profile**
Description	Allows the Job Seeker to register themselves into FutureSeekers.lk
Primary Actor	Job Applicant
Pre-Conditions	None
Basic Flow	6. Use Case begins when the Applicant clicks on the **'Register Now'** link from the Login Page of FutureSeekers.lk. (There would be 2 buttons in the login page, one for Applicant and one for Employer. 7. The user will be directed to the Applicant registration page. 8. The Applicant must fill his/her details such as Name, Address, email etc. 9. After the user clicks on **'Register Now'** button, the details are validated and are updated to the database and the use case would end with a Success Message.
Extensions	After **'Register Now'** button is clicked, if the validation fails: 1. System would display a validation error message, for example 'Fields cannot be empty'. 2. Use case ends
Post Conditions	The Applicant can login to the system and view the profile details but cannot edit the profile details unless the profile verification is completed by the Admin.
Optional Information	None

Use Case Name	**Login**
Description	Allows the Job Seeker to login to his account in order to perform his operations such as applying for a job.
Primary Actor	Job Applicant
Pre-Conditions	The Applicant's profile must be verified successfully for the profile to be registered in the system.
Basic Flow	1. Use Case begins when the Applicant enters the valid credentials and clicks on the **'Login'** Button on the FutureSeekers.lk Login page. 2. The username and the password are validated and queried with the database to find a match. 3. After successful validations, the User will receive a dialog box, 'Welcome Back!' and will be directed to the Jobs Page which would complete the use case.
Extensions	After **'Login'** button is clicked, if the user doesn't exist: 1. System would display a validation error message, for example 'User Doesn't Exist!'. 2. Use case ends.
Post Conditions	The Applicant can only perform their functions once the profile is **verified by the FutureSeekers.lk team**. If the account verification is still on pending status, then the Applicant temporarily gets limited privileges where he/she will **only be able to view their profile information.**
Optional Information	None

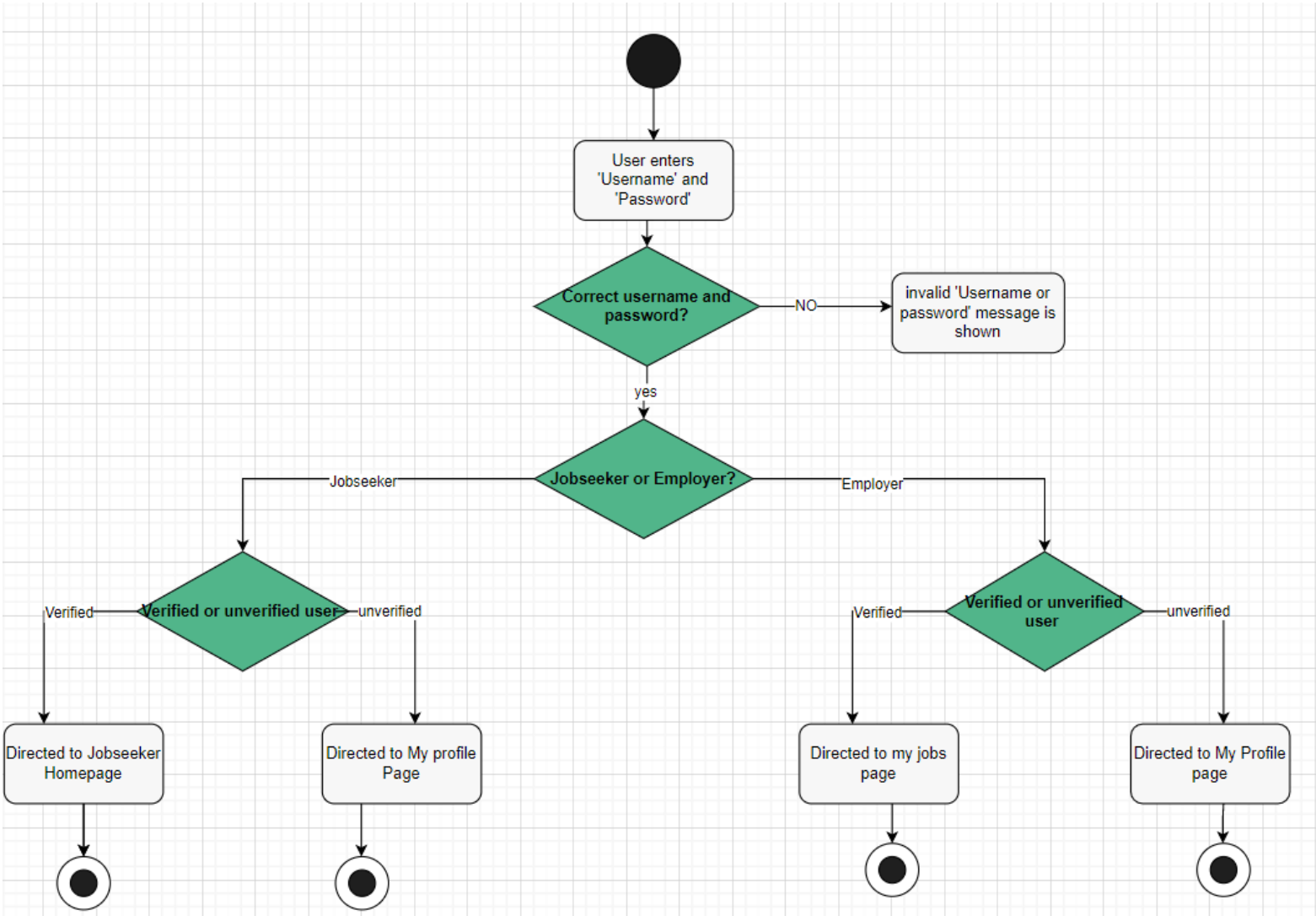
Use Case Name	**Edit Profile**
Description	Allows the Job Applicant to edit his/her details.
Primary Actor	Job Applicant
Pre-Conditions	The Applicant must be a verified user and logged in.
Basic Flow	1. Use case begins when the applicant clicks the **'My Profile'** button on the navigation bar. 2. The Applicant details will be fetched from the database and displayed to the user. 3. The Applicant will be able to update the details and once he/she has completed, would click on the **'Save Changes** Button. 4. The new details will be updated on the database and the use case would end by displaying a Success Message.
Extensions	After **'Update Profile'** button is clicked, if the validation fails: 7. System would display a validation error message, for example 'Username already exists, Try a different one' 8. Use case ends
Post Conditions	The Applicant would be able to view the updated profile details.
Optional Information	None

Use Case Name	**View User Profiles**
Description	Allows the Admin to view the user profiles (Employer and Applicant) and verify their profiles.
Primary Actor	System Admin
Pre-Conditions	There must be a **separate portal for the admin** to login and perform administrative tasks such as accepting or rejecting user profiles. The super admin's credentials will be permanently available and is added from the backend (database). Admin will not be able to delete any profile requests, however he/she can reject the requests. (The Delete Function only applies to the user profiles that are already accepted in the system.)
Basic Flow	1. Use case begins when the admin accesses the admin portal through the URL (by typing '/admin' after the root path). 2. The admin portal landing page must be displayed with username and password fields, and a **'Login Button'** 3. The Super admin will already be registered from the backend, so he/she would have to enter their credentials to gain access. 4. The username and password will be validated with the database and the admin will be directed to the **Administrative Summary Page.** 5. The admin will select the **'Manage Profiles'** drop-down button where there would be 2 options (Manage Applicant and Manage Employer). Clicking on one of these options would show the admin a table which lists all the new profile requests, verified users, rejected users, and deleted users. 6. The admin will have the privilege to perform the following: • Accept/Reject/Delete User profiles (Applicant and Employer). 7. The admin would have to click on a row within the table and then click on Accept/Reject/Delete Button. 8. Accepting a user profile means that the user's profile is verified successfully and vice versa. This status change would immediately get updated to the database.

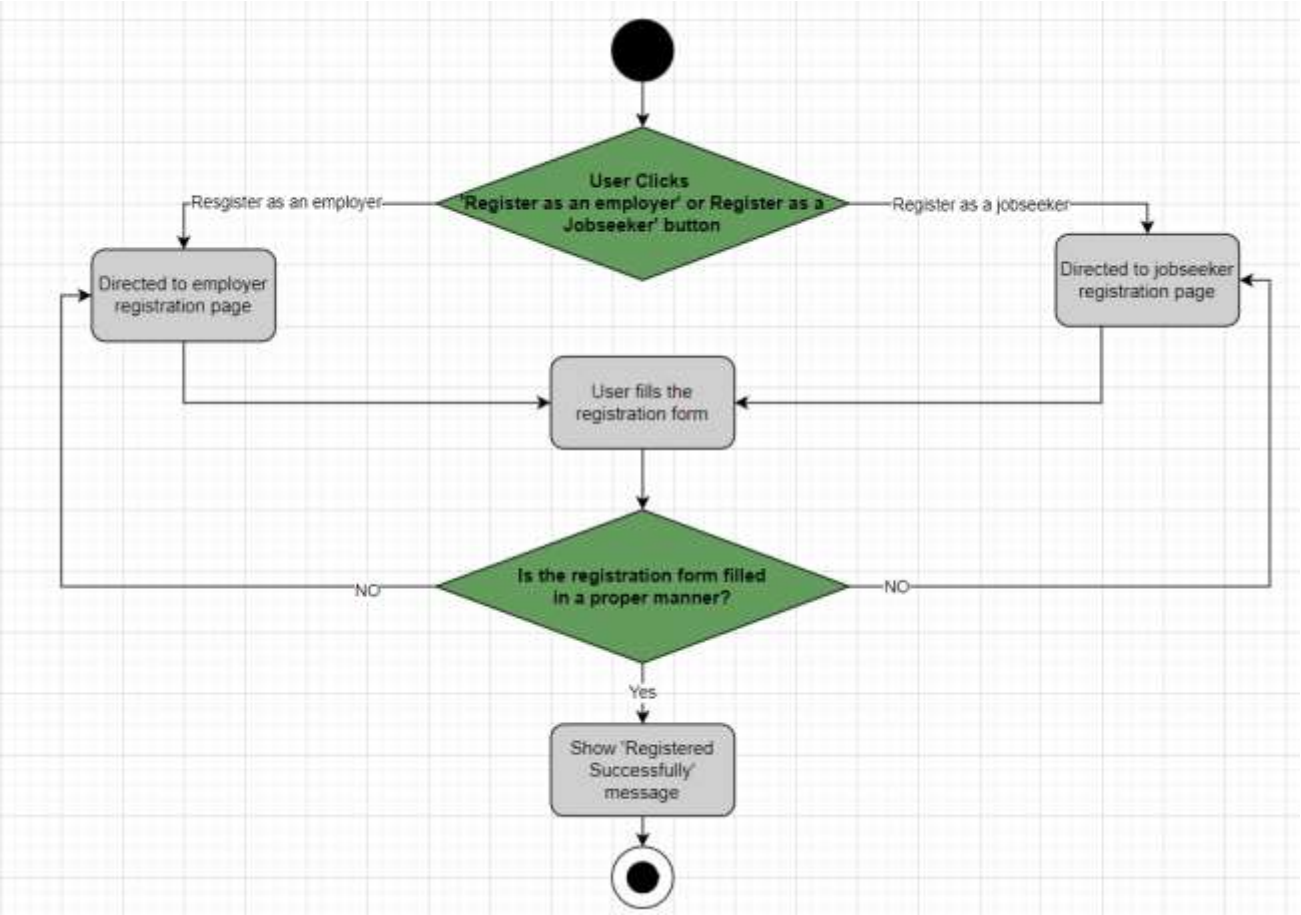
	9. Once a profile status is updated by the admin, they would then send an email to the relevant user acknowledging the registration which would complete the use case.
Extensions	After **'Login'** button is clicked, if the admin credentials are invalid: 9. System would display a validation error message, for example 'Invalid Credentials, you don't have administrative rights' 10. Use case ends
Post Conditions	The admin would be able to perform all the administrative tasks.
Optional Information	None

9.0 Activity Diagrams

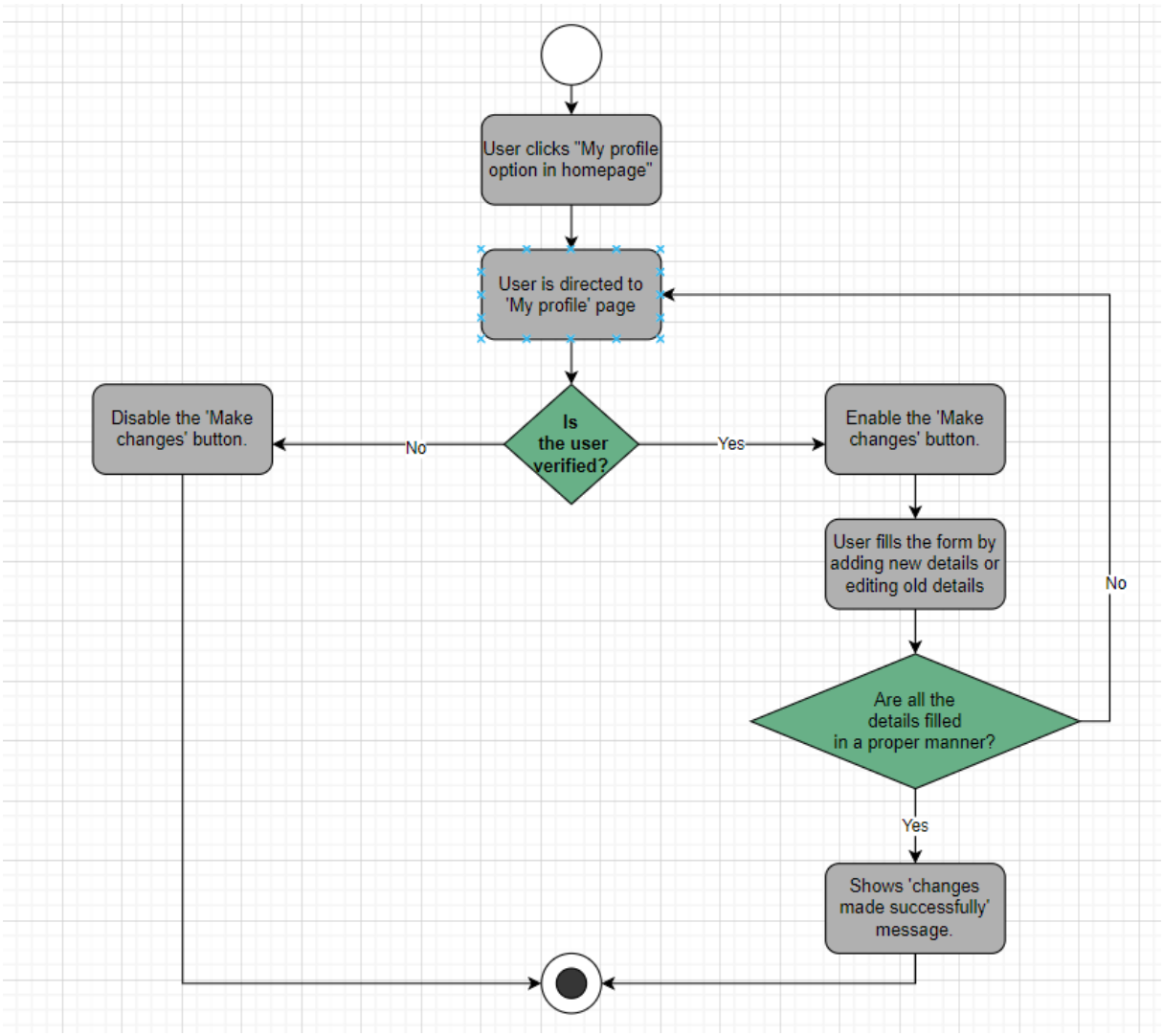
9.1 Login Function



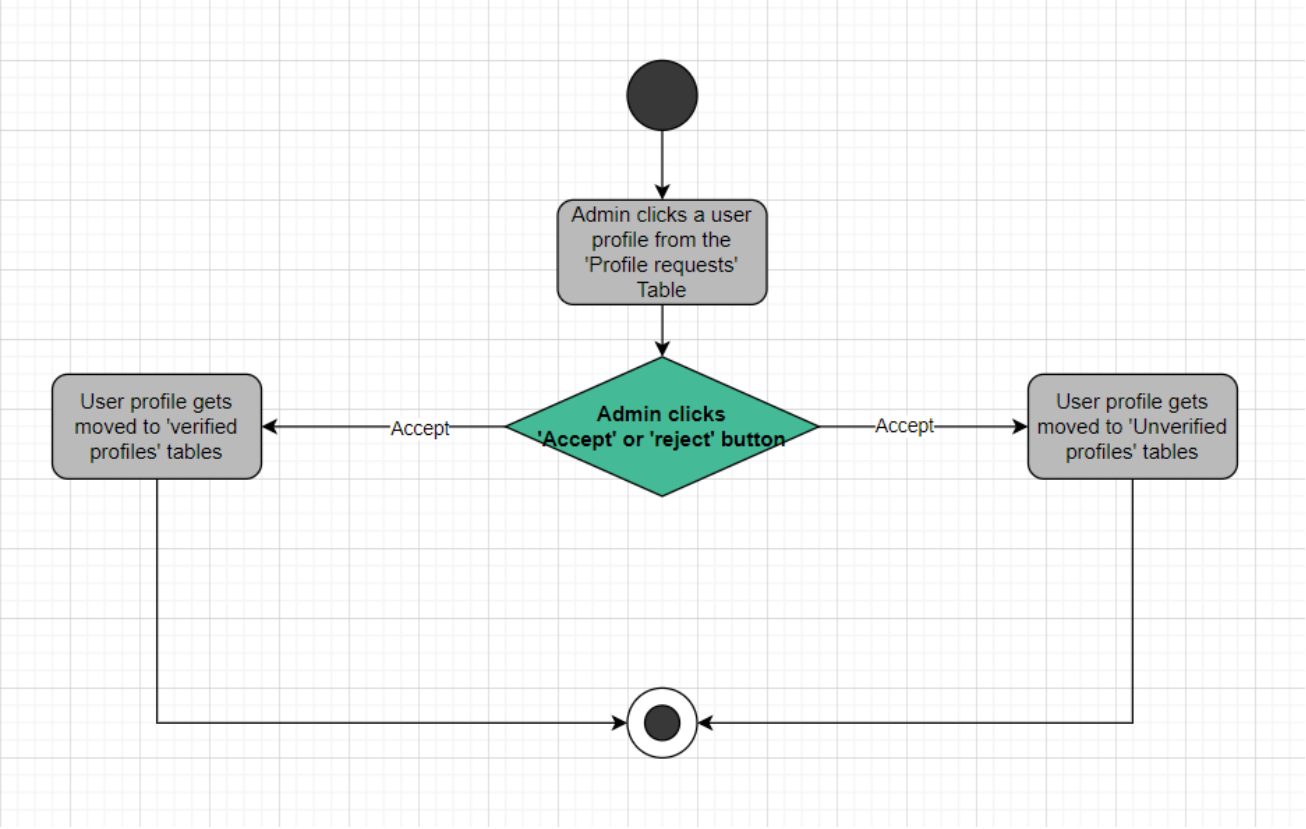
9.2 Register Function



9.3 Edit profile function



9.4 Profile verification



10.0 Implementation

10.1 Login function

```
public function login()
{
        if(session()->get('user_id')!= null){
                if(session()->get('user_type') == "employer") {
                        return redirect()->to('EmployerHome/index');
                }
                else if(session()->get('user_type') == "applicant") {
                        return redirect()->to('ApplicantHome/index');
                }

        }
        $validation = $this->validate([
                'username' => [
                        'rules' => 'required|is_not_unique[user_account.username]',
                        'errors' => [
                                'required' => 'Username is required',
                                'is_not_unique' => 'This username does not exist'
                        ]
                ],
                'password' => [
                        'rules' => 'required',
                        'errors' => [
                                'required' => 'Password is Required'
                        ]
                ]
        ]);
```

```
        if (!$validation) {
                return view('/Home/index', ['validation' => $this->validator]);
        } else {
                $username = $this->request->getPost('username');
                $password = $this->request->getPost('password');

                $userAccountModel = new \App\Models\userAccountModel();
                $user_info = $userAccountModel->where('username', $username)->first();

                if ($user_info != null && $user_info['password'] == $password) {
                        // $session = session();
                        // $session->regenerate();
                        session()->set('user_id', $user_info['id']);
                        session()->set('user_type', $user_info['type']);


                        $user_id = session()->get('user_id');
                        $userAccountModel = new \App\Models\userAccountModel();
                        $user_type = $userAccountModel->where('id', $user_id)->first();
                        if ($user_type['type'] == "employer") {
                                return redirect()->to("//EmployerHome/index")->with('info', 'Login Successful');
                        } elseif ($user_type['type'] == "applicant") {
                                return redirect()->to("//ApplicantHome/index")->with('info', 'Login Successful');
                        } else {
                                return view('Home/index');
                                session()->setFlashdata('fail', 'Incorrect Password!');
                        }
                } else {
                        session()->setFlashdata('fail', 'Incorrect Password!');
                        return redirect()->to('/Home')->withInput();
                        //return redirect()->to("/Home/loginFailed")->with('info', 'Given Name or Password is not correct, try again');
                }
        }
}
```

```
  if (session()->get('user_id') == null || session()->get('user_type') == "applicant") {
    return redirect()->to('Home/index')->with('fail', 'You must be logged in..');;
  }
  $session = session();
  $session->regenerate();
  $user_id = session()->get('user_id');
  $UserAccount = new \App\Models\userAccountModel();
  $EmployerM = new \App\Models\employerModel();
  $CompanyM = new \App\Models\companyModel();
  $user_info = $UserAccount->where('id', $user_id)->first();

  $query_employer = $EmployerM->query("Select * from employer where user_account_id = $user_id");
  foreach ($query_employer->getResult() as $row) {
    $companyid = $row->company_id;
  }
  // echo $companyid;

  $query_company = $CompanyM->query("Select * from company where id = $companyid");
  foreach ($query_company->getResult() as $row2) {
    $companyemail = $row2->email;
  }
  // echo $companyname;
  if ($user_info['status'] == 0 || $companyemail == null) {
    return view('MyProfileEmployer/index');
  }

  return view('EmployerHome/index');
}
```

```php
public function index()
{
  if(session()->get('user_id')== null || session()->get('user_type') == "employer"){
    return redirect()->to('Home/index')->with('fail', 'You must be logged in..');;
  }
  $session = session();
  $session->regenerate();
  $user_id = session()->get('user_id');
  $UserAccount = new \App\Models\userAccountModel();
  $user_info = $UserAccount->where('id', $user_id)->first();
  if ($user_info['status'] == 0) {
    return view('MyProfileApplicant/index');
  }

  return view('ApplicantHome/index');
}
public function logout()
      {
              $session = session();
              $session->destroy();
              return redirect()->to('Home/index')->with('fail', 'You are Logged out');
      }
}
```

- The website has two types of logins one for Applicants, Employers and the other login function is for Admins to log in to the admin portal.
- Every user has to enter correct credentials to log in and the user has to be registered first before attempting to login.
- If the user enters invalid data, the system will show errors.
- When an unverified Applicant login they are directed to ‘My profile’ page, if they are verified, they will be directed to Homepage.
- When an unverified Employer login they are directed to ‘My profile’ page, if they are verified and not yet configured their company profile, they will be directed to My profile page. If they are verified and have configured their company profile they will be directed to ‘My Jobs’ page.

10.2 Register Function (Applicant)

```
public function createprofile() {
  if(session()->get('user_id')!= null){
                    if(session()->get('user_type') == "employer") {
                            return redirect()->to('EmployerHome/index');
                    }
                    else if(session()->get('user_type') == "applicant") {
                            return redirect()->to('ApplicantHome/index');
                    }

            }

  $validation = $this->validate([
    'name' => [
      'rules' => 'required',
      'errors' => [
          'required' => 'Name is Required'
      ]
    ] ,
    'address' => [
      'rules' => 'required',
      'errors' => [
          'required' => 'Address is Required'
      ]
    ] ,
    'email' => [
      'rules' => 'required|valid_email|is_unique[job_seeker.email]',
      'errors' => [
          'required' => 'Email is Required',
          'valid_email' => 'Must be a valid Email Address',
          'is_unique' => 'Email already exists',
      ]
    ] ,
```

```
  'contactNo' => [
    'rules' => 'required',
    'errors' => [
        'required' => 'Contact No is required'
    ]
  ] ,
  'dob' => [
    'rules' => 'required',
    'errors' => [
        'required' => 'Date of Birth is Required'
    ]
  ] ,
  'currentJobTitle' => [
    'rules' => 'required',
    'errors' => [
        'required' => 'Job Position is Required'
    ]
  ] ,
  'username' => [
    'rules' => 'required|is_unique[user_account.username]',
    'errors' => [
        'required' => 'Username is required',
        'is_unique' => 'This username already exist'
    ]
  ],
  'password' => [
    'rules' => 'required',
    'errors' => [
        'required' => 'Password is Required'
    ]
  ]

]);
```

```php
if (!$validation) {
  return view('/RegisterApplicant/index', ['validation' => $this->validator]);
} else {
  //getting values from form
  $name = $this->request->getPost('name');                    //jobseeker
  $address = $this->request->getPost('address');              //jobseeker
  $email = $this->request->getPost('email');                  //jobseeker
  $contactNo = $this->request->getPost('contactNo');          //jobseeker
  $dob = $this->request->getPost('dob');                      //jobseeker
  $currentJobTitle = $this->request->getPost('currentJobTitle');  //jobseeker
  $username = $this->request->getPost('username');            //useraccount
  $password = $this->request->getPost('password');            //useraccount

  //useraccount table info
  $valuesUser = [
                'username' => $username,
                'password' => $password,
                'type' => "applicant"
    ];

  $userAccountModel = new \App\Models\userAccountModel();

  //inserting info to db table
  $queryUser = $userAccountModel->insert($valuesUser);
  if(!$queryUser){
    // return redirect()->back()->with('fail', 'Please try again later..');
  } else {
    // return redirect()->to('Register')->with('success', 'User Registration Successful');
  }
```

```
  //jobseeker table info
  $valuesSeeker = [
                    'name' => $name,
                    'address' => $address,
                    'email' => $email,
                    'contactNo' => $contactNo,
                    'dob' => $dob,
                    'currentJobTitle' => $currentJobTitle,
                    'user_account_id' => $queryUser
                  ];

  $jobSeekerModel = new \App\Models\jobSeekerModel();

  //inserting info to db table
  $querySeeker = $jobSeekerModel->insert($valuesSeeker);
  if(!$querySeeker){
    return redirect()->back()->with('fail', 'Please try again later..');
  }
  else{
    return redirect()->to('RegisterApplicant')->with('success', 'User Registration Successful');
  }
}
```

- There are two types of registration, one for Applicants and the other is for Employers. (Admins don't have a registration, they are added to web site by main admin and main admin's details are added to the database manually).
- The User should enter valid details before clicking the 'Register' or they will get error messages telling to enter valid inputs.
- If the user enters everything in a proper manner the website will show a 'Register successfully' message.
- After that the user can use their credentials to log in to the system.

10.3 Edit Profile function (Applicant)

```
public function editProfile()
{
  if(session()->get('user_id')== null || session()->get('user_type') == "employer"){
    return redirect()->to('Home/index')->with('fail', 'You must be logged in..');;
  }
  $validation = $this->validate([
    'name' => [
      'rules' => 'required',
      'errors' => [
        'required' => 'Name is Required'
      ]
    ],
    'address' => [
      'rules' => 'required',
      'errors' => [
        'required' => 'Address is Required'
      ]
    ],
    'email' => [
      'rules' => 'required|valid_email',
      'errors' => [
        'required' => 'Email is Required',
        'valid_email' => 'Must be a valid Email Address'
      ]
    ],
    'contactNo' => [
      'rules' => 'required',
      'errors' => [
        'required' => 'Contact No is required'
      ]
    ],
```

```
  'dob' => [
    'rules' => 'required',
    'errors' => [
      'required' => 'Date of Birth is Required'
    ]
  ],
  'currentJobTitle' => [
    'rules' => 'required',
    'errors' => [
      'required' => 'Job Position is Required'
    ]
  ],
  'username' => [
    'rules' => 'required',
    'errors' => [
      'required' => 'Username is required'
    ]
  ],
  'password' => [
    'rules' => 'required',
    'errors' => [
      'required' => 'Password is Required'
    ]
  ]
]);
```

```php
if (!$validation) {
  //echo "Validation Error";
  return view('/MyProfileApplicant/index', ['validation' => $this->validator]);
} else {
  $name = $this->request->getPost('name');
  $address = $this->request->getPost('address');
  $email = $this->request->getPost('email');
  $dob = $this->request->getPost('dob');
  $contactNo = $this->request->getPost('contactNo');
  $currentJobTitle = $this->request->getPost('currentJobTitle');
  $username = $this->request->getPost('username');
  $password = $this->request->getPost('password');

  session();
  session()->regenerate();
  $user_id = session()->get('user_id');

  // echo $user_id;
  // echo $name;
  // echo $address;
  // echo $email;
  // echo $dob;
  // echo $contactNo;
  // echo $currentJobTitle;

  $UserAccountM = new \App\Models\userAccountModel();
  $queryuser = $UserAccountM->query("Update user_account
                                     Set username = '$username',
                                         password = '$password'
                                     Where id = $user_id");
```

```
      if (!$queryuser) {
        //echo "fail";
      }

      $JobSeekerM = new \App\Models\jobSeekerModel();
      $queryJobSeeker = $JobSeekerM->query("Update job_seeker
                                    Set name = '$name',
                                        address = '$address',
                                        email = '$email',
                                        dob = '$dob',
                                        contactNo = $contactNo,
                                        currentJobTitle = '$currentJobTitle'
                                    Where user_account_id = $user_id");
      if (!$queryJobSeeker) {
        return redirect()->back()->with('fail', 'Please try again later..');
      } else {
        return redirect()->to('MyProfileApplicant')->with('success', 'Changes made successfully');
      }
    }
  }
}
```

- Every verified user has the ability to edit their profile details by going to the ‘My profile page’ and typing valid inputs and clicking the ‘save changes’ button.
- Unverified users also can go to the ‘My Profile’ page to view their details but they won’t be able to edit their data and the ‘Save changes’ button will be disabled.

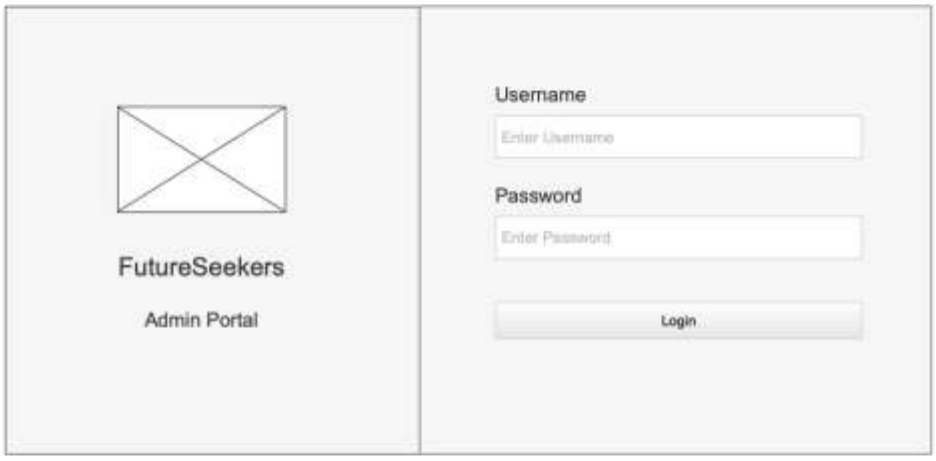
10.4 Profile verification function

```
  public function verify()
  {
    if(session()->get('admin_id')== null){
      return redirect()->to('Admin/index')->with('fail', 'You must be logged in..');;
    }
    //To Accept User--Changing Status to 1
    if (isset($_POST['auser'])) {
      $useraccountid = $this->request->getPost('account_id');
      echo $useraccountid;
      $UserAccount = new \App\Models\userAccountModel();
      $query = $UserAccount->query("update user_account set status = 1 where id = $useraccountid");
      return redirect()->to("//AdminApplicantProfiles/index")->with('info', 'Changes Made Succesfully');
    }
    //To Reject User--Changing Status to 2
    if (isset($_POST['ruser'])) {
      $useraccountid = $this->request->getPost('account_id');
      echo $useraccountid;
      $UserAccount = new \App\Models\userAccountModel();
      $query = $UserAccount->query("update user_account set status = 2 where id = $useraccountid");
      return redirect()->to("//AdminApplicantProfiles/index")->with('info', 'Changes Made Succesfully');
    }
    //To Delete User--Changing Status to 2
    if (isset($_POST['duser'])) {
      $useraccountid = $this->request->getPost('account_id1');
      echo $useraccountid;
      $UserAccount = new \App\Models\userAccountModel();
      $query = $UserAccount->query("update user_account set status = 3 where id = $useraccountid");
      return redirect()->to("//AdminApplicantProfiles/index")->with('info', 'Changes Made Succesfully');
    }
  }
}
```

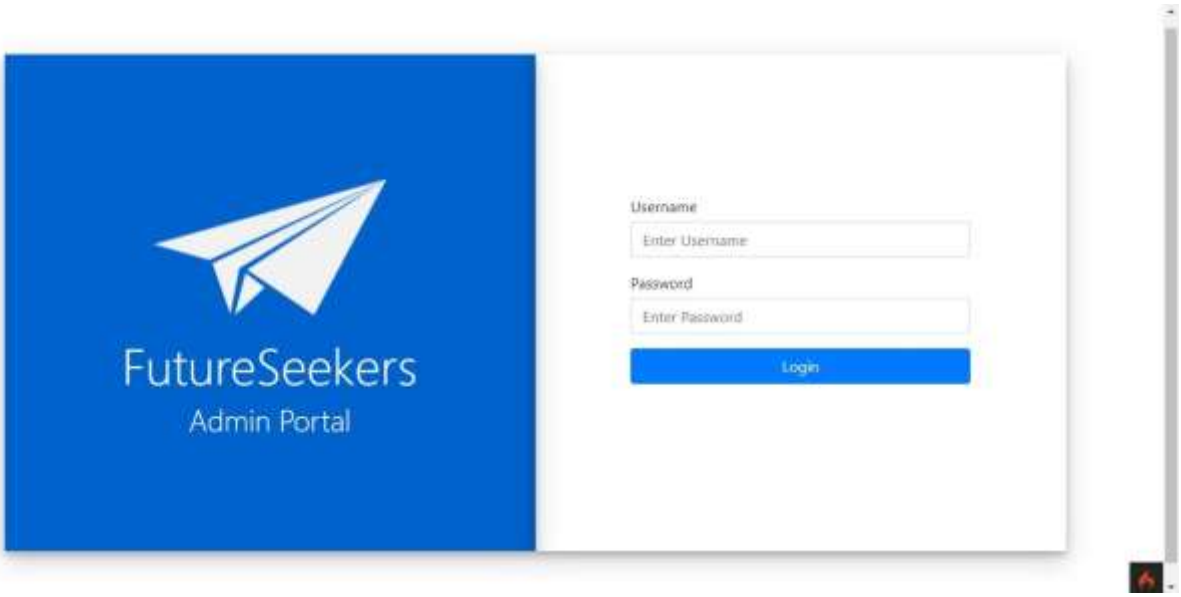
- This feature is only available to the admin
- The admin will be able to see a list of verified, unverified users list and he/she will be able to pick users one by one will be able to accept or reject.

11.0 Interfaces

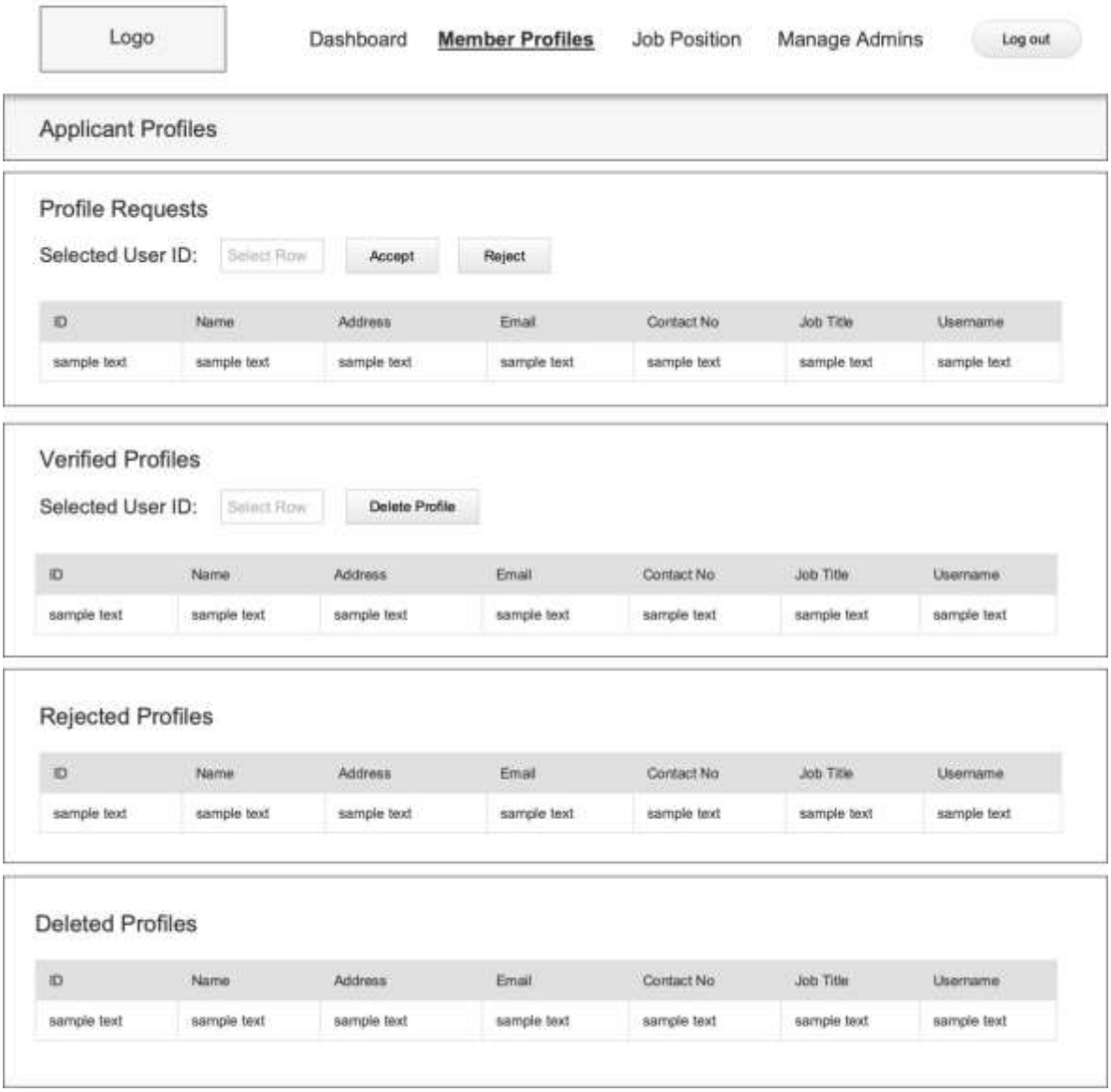
11.1 Admin login page (Wireframe)



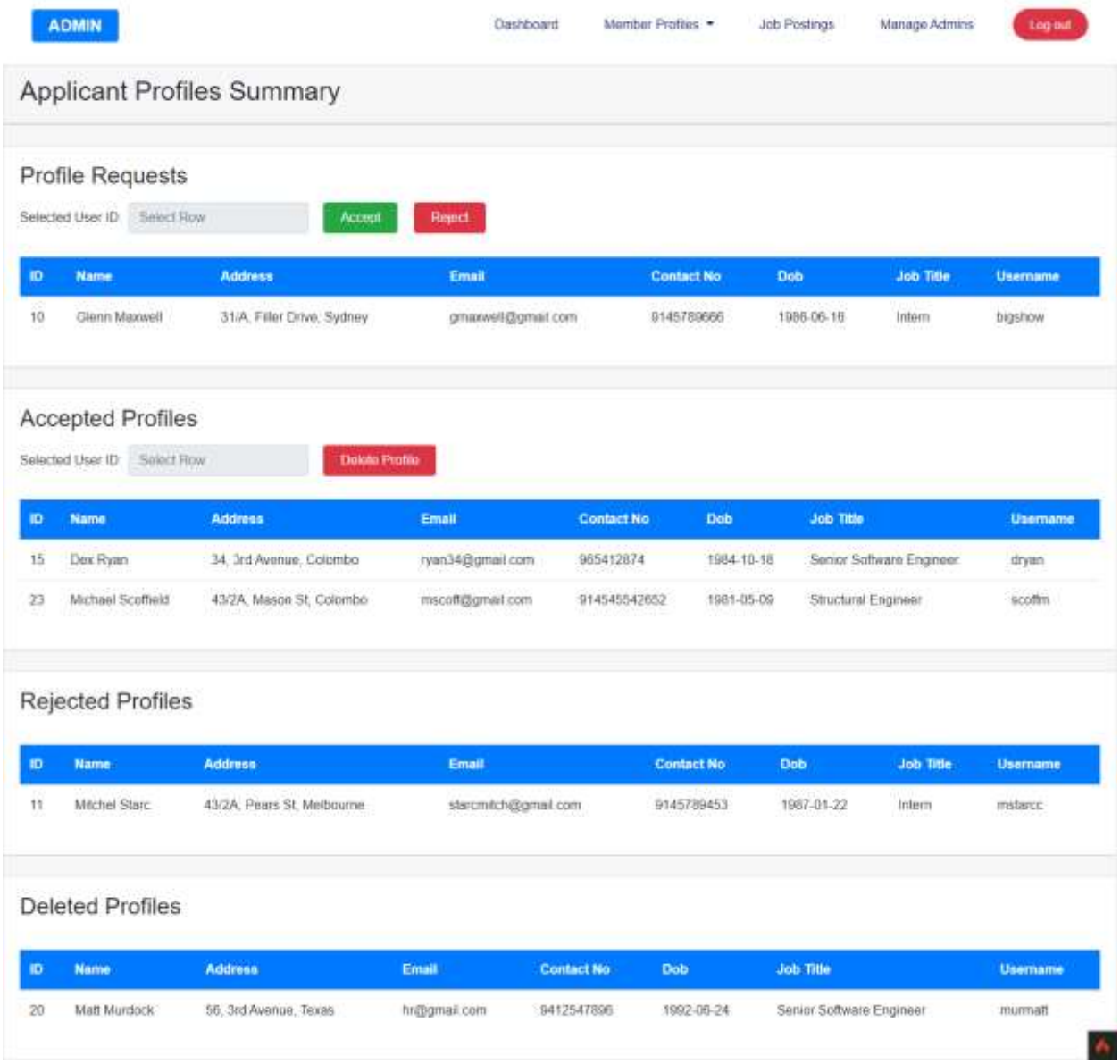
11.2 Admin login page (Actual View)



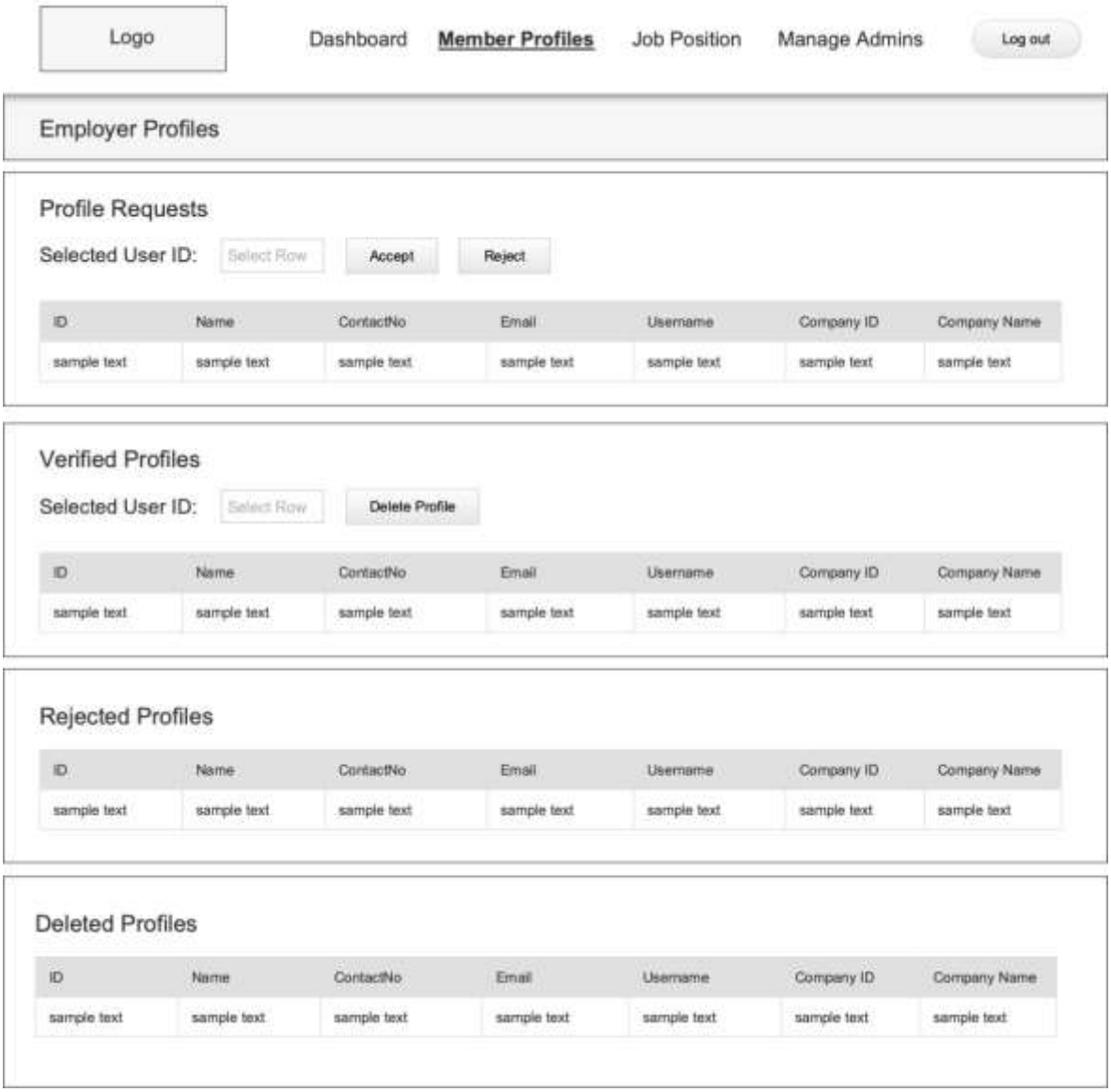
11.3 Applicant Profile Verification page for Admin (Wireframe)



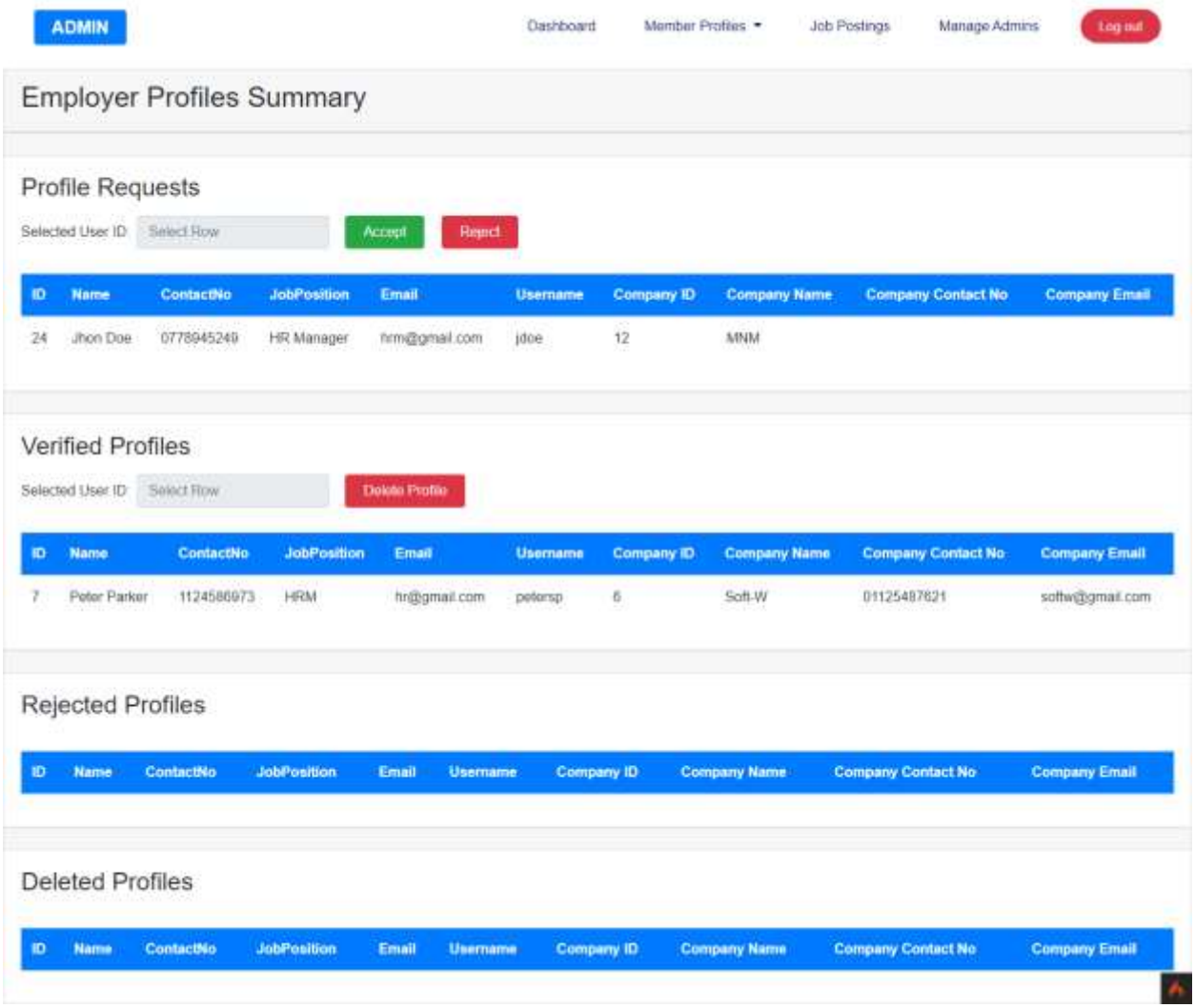
11.4 Applicant Profile Verification page for Admin (Actual View)



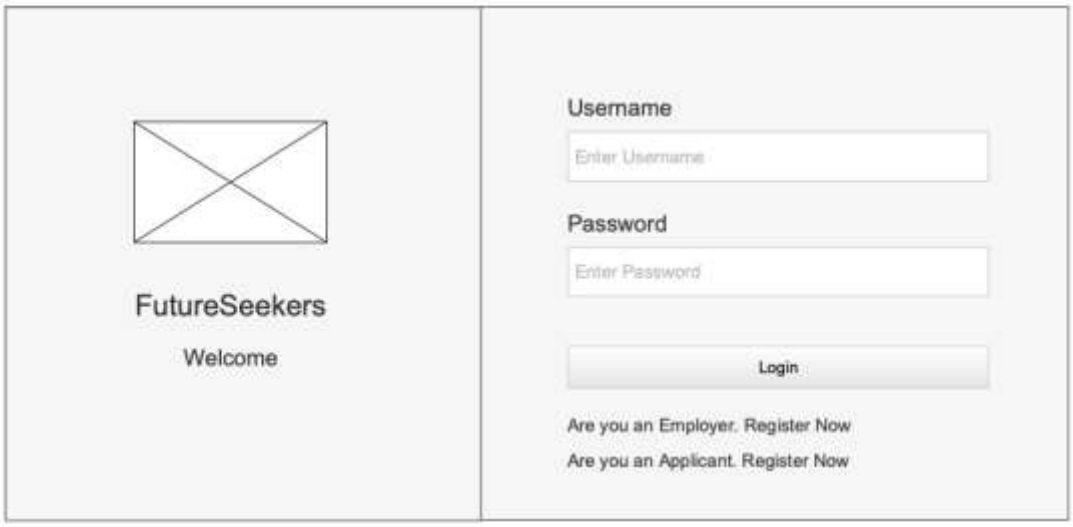
11.5 Employer Profile verification page for Admin (Wireframe)



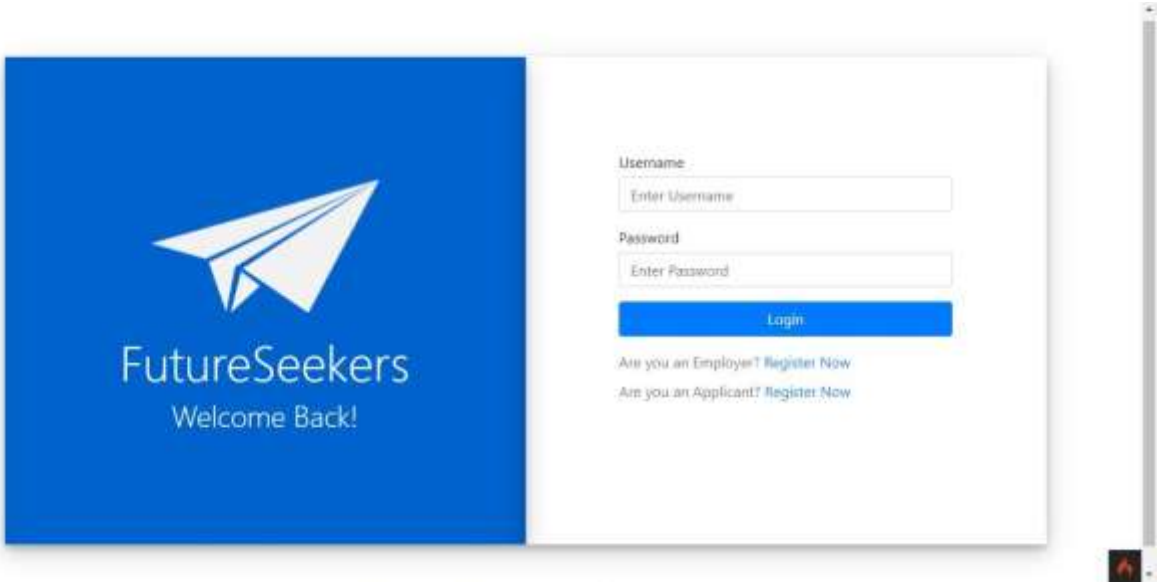
11.6 Employer Profile verification for Admin (Actual View)



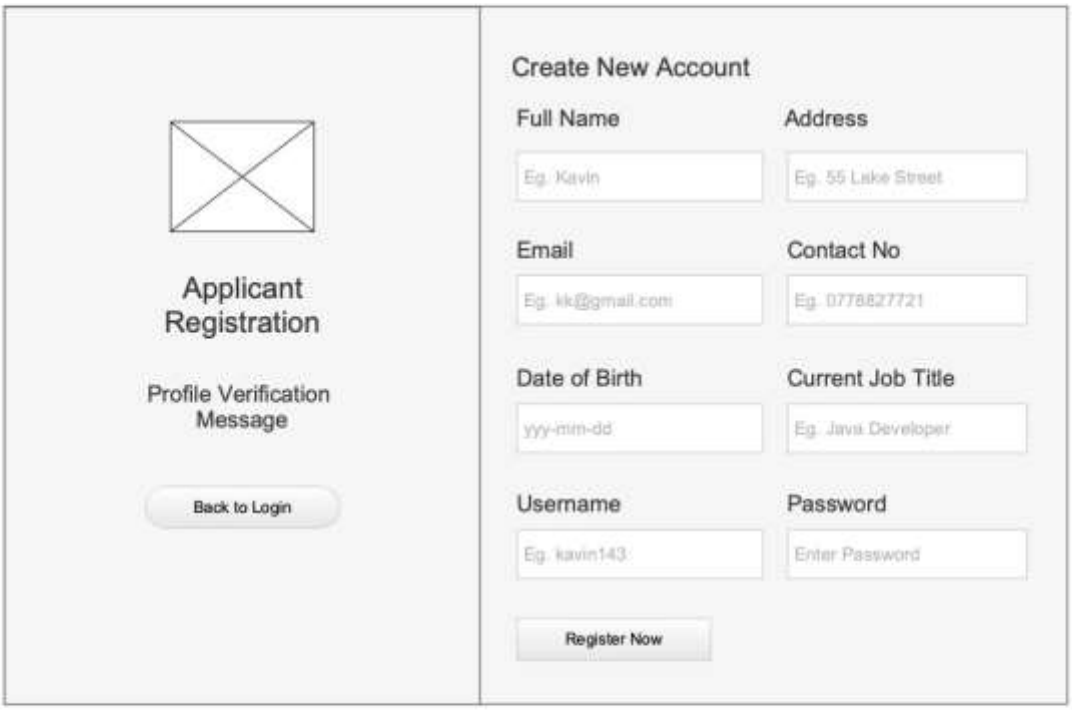
11.7 Login page (Wireframe)



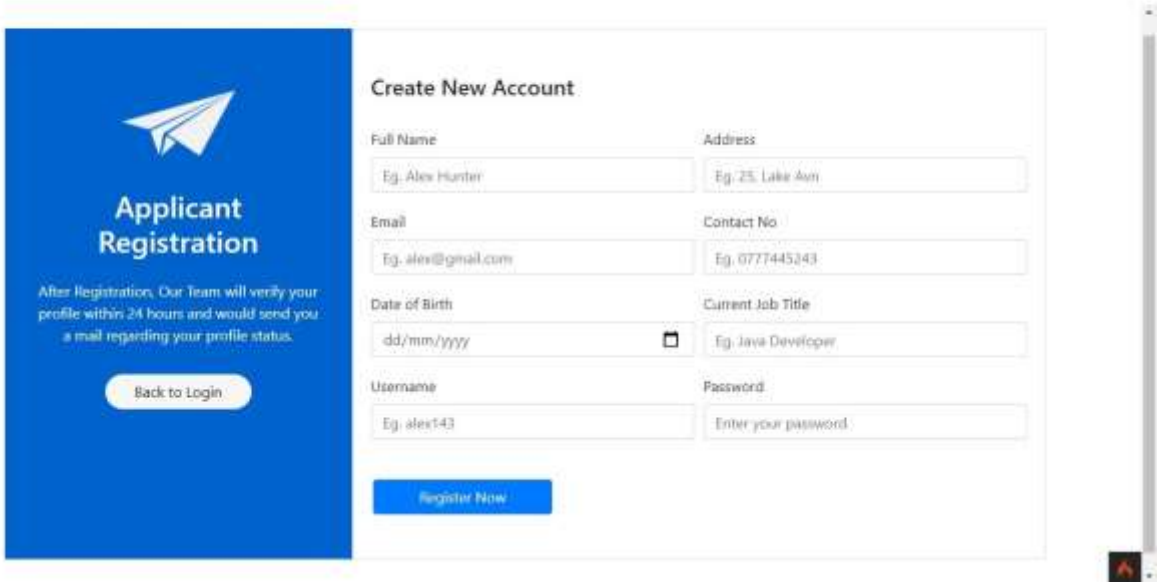
11.8 Login Page (Actual View)



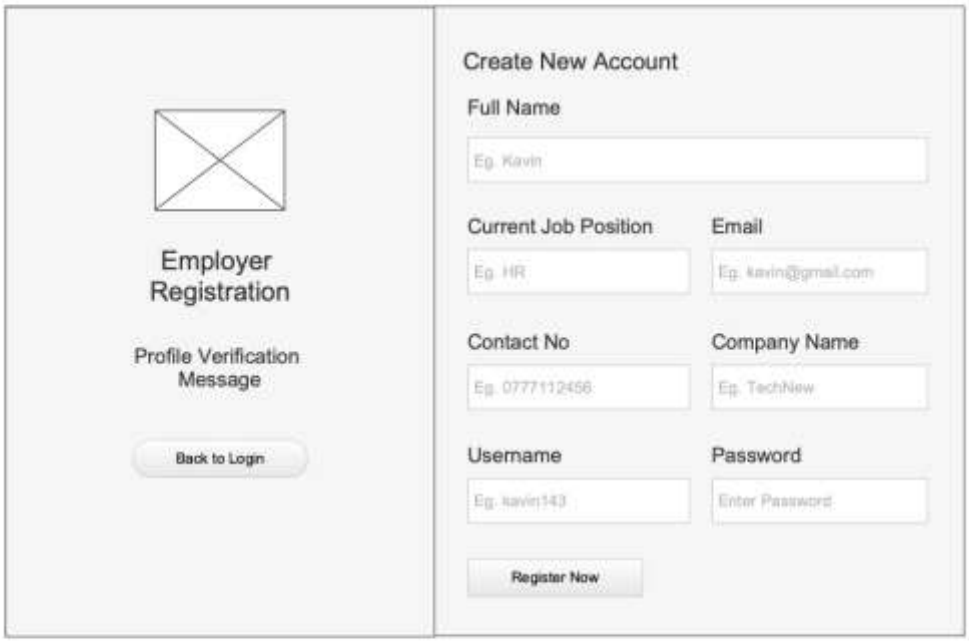
11.9 Register for Applicants (Wireframe)



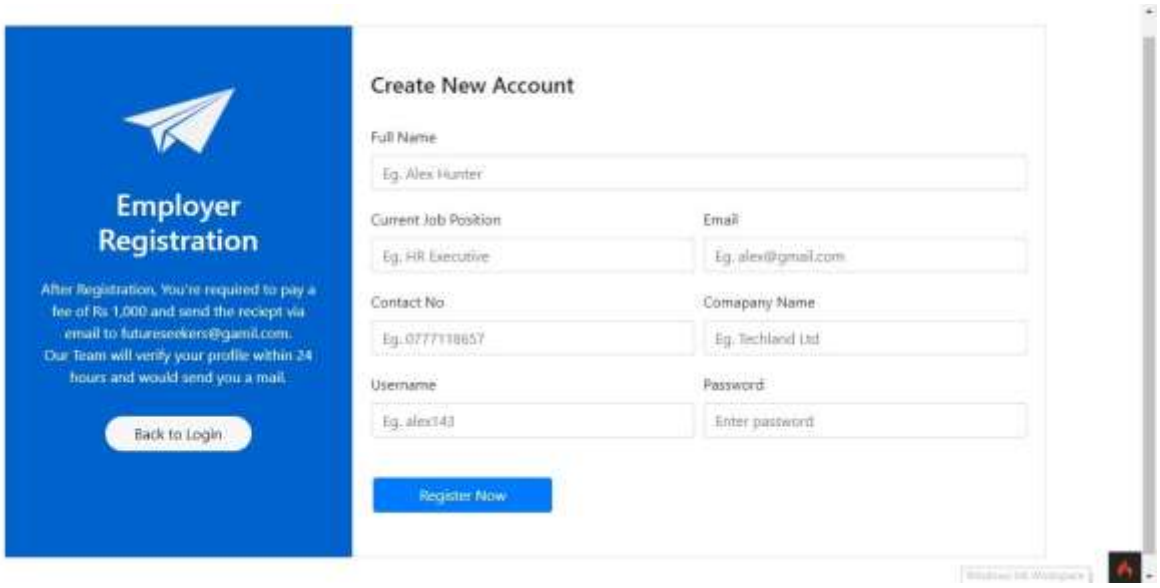
11.10 Register for Applicants (Actual View)



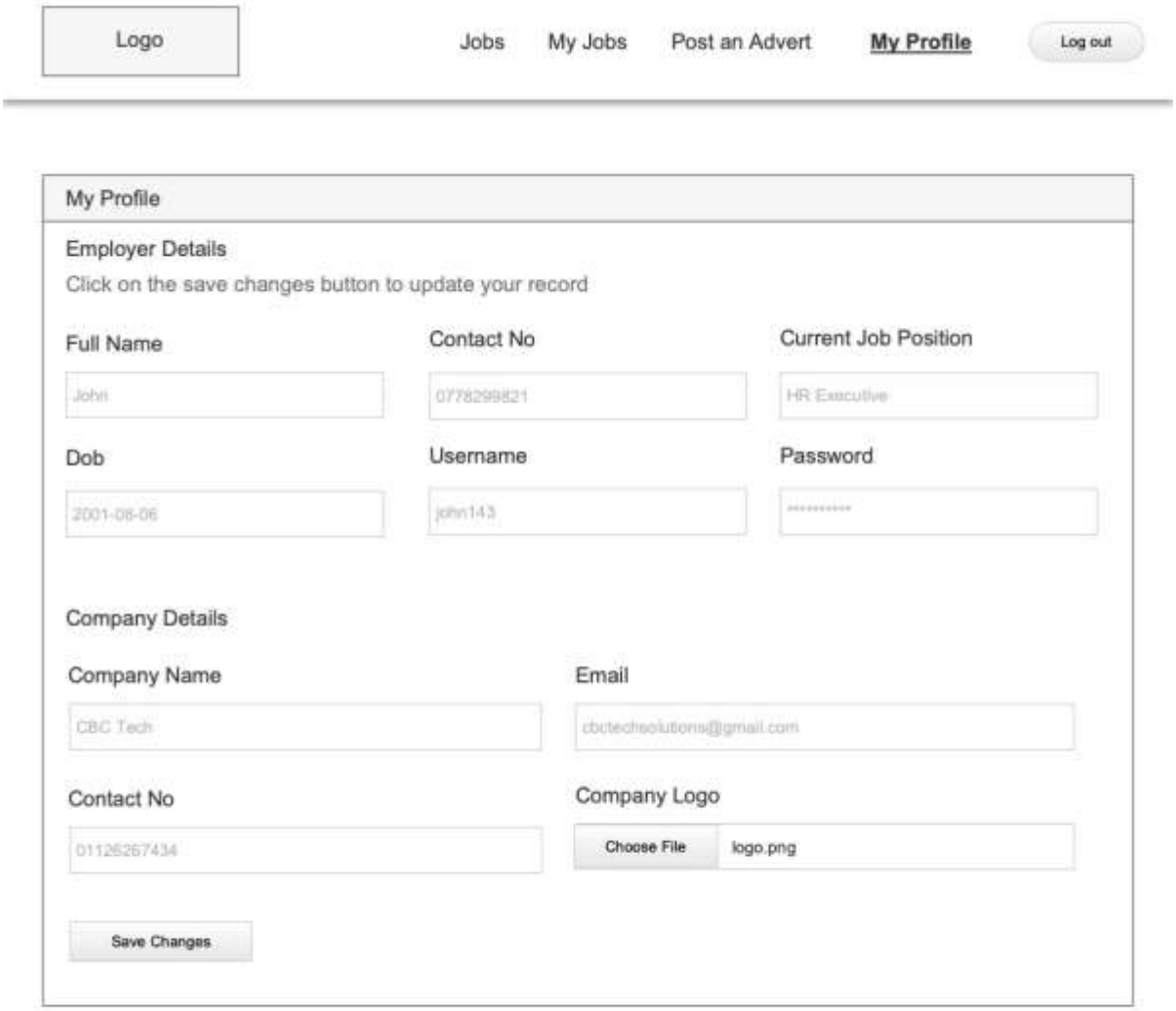
11.11 Register for Employer (Wireframe)



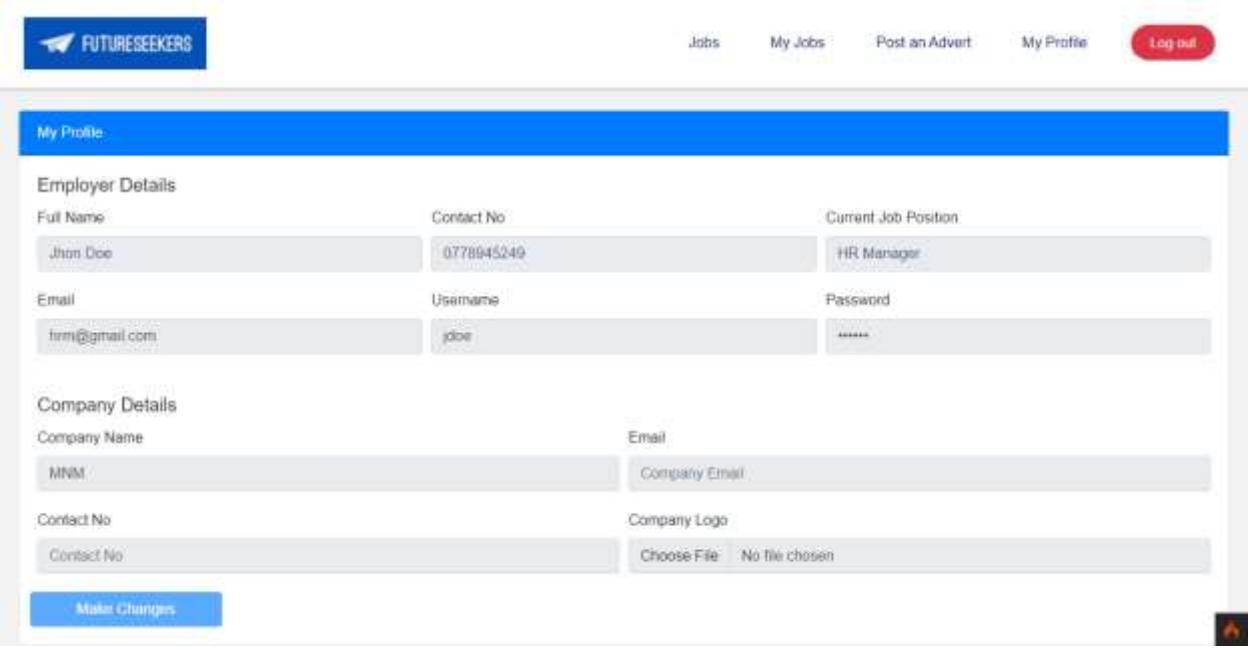
11.12 Register For Employer (Actual View)



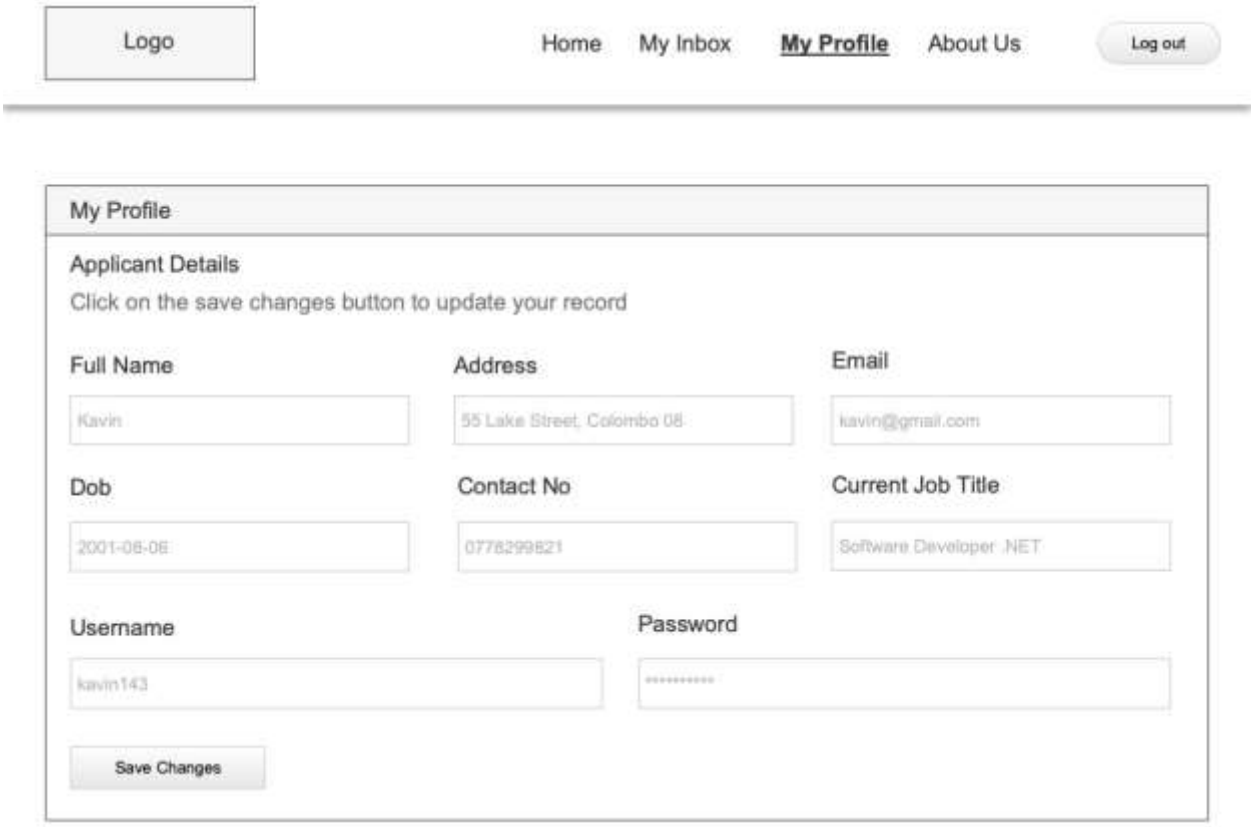
11.13 My Profile page for Employer (Wireframe)



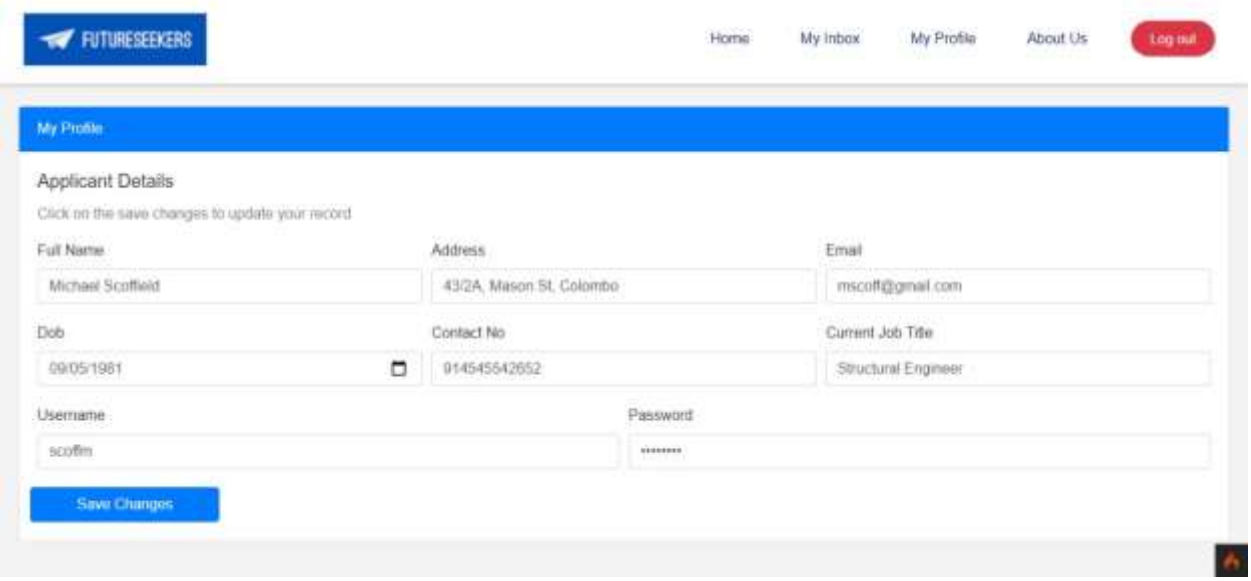
11.14 My profile page for Employer (Actual View)



11.15 My Profile page for Applicant (Wireframe)



11.16 My Profile page for Applicant (Actual view)



12.0 Test Cases

12.1 Log in [Admin]

Following test cases are conducted to test the log in function and log in page UI element for the “Admin” accounts.

Test Scenario ID	Login 01	**Test Case ID**	Admin Login 01
Test Case Description	Admin Login – With valid user details	**Test Priority**	High
Pre-Requisite	A valid Admin account	**Post-Requisite**	User stay logged in until hitting the log out button

Test Execution Steps:

S.No	Action	Inputs	Expected Output	Actual Output	Test Browser	Test Result	Test Comments
1	Direct to the “Admin” login page	http://localhost/futureseekerslk/public/Admin	Load “Admin” Login page	Load “Admin” Login page	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost
2	Enter valid username & Password and hit login button	Username: Admin_1 Password: adminfs123	Successfully log in to the account and direct to Admin Dashboard	Logged in to the account and directed to Admin Dashboard	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost Implementations of the Admin Dashboard comes in future sprints

Test Scenario ID	Login 01	**Test Case ID**	Admin Login 02
Test Case Description	Admin Login – With invalid user details	**Test Priority**	High
Pre-Requisite	NA	**Post-Requisite**	NA

Test Execution Steps:

S.No	Action	Inputs	Expected Output	Actual Output	Test Browser	Test Result	Test Comments
1	Direct to the "Admin" login page	http://localhost/futureseekerslk/public/Admin	Load "Admin" Login page	Load "Admin" Login page	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost
2	Enter invalid Username & Any Password and enter login button	Username: Admin Password: adminfs123111	Displaying "This username does not exist" under the username field	Displayed "This username does not exist" under the username field	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost
3	Enter valid username & incorrect Password and enter login button	Username: Admin_1 Password: adminfs123111	Displaying "Incorrect Password!" error massage	Displaying "Incorrect Password!" error massage	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost

12.2 Log in [Employer]

Following test cases are conducted to test the log in function and log in page UI element for the “Employer” accounts

Test Scenario ID	Login 02	**Test Case ID**	Employer Login 01
Test Case Description	Employer login with valid user details of an approved and company details filled account	**Test Priority**	Medium
Pre-Requisite	Approved and company details filled Employer account	**Post-Requisite**	User stay logged in until hitting the log out button

Test Execution Steps:

S.No	Action	Inputs	Expected Output	Actual Output	Test Browser	Test Result	Test Comments
1	Direct to the “Public” login page	http://localhost/futureseekerslk/public/	Load “Public” Login page	Load “Public” Login page	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost
2	Enter valid username & Password and hit login button	Username: joeblack Password: joeblack	Successfully log in to Employer account and direct to the Jobs Page	Logged in to Employer account and direct to the Jobs Page	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost. Implementations of the Jobs Page comes in future sprints

Test Scenario ID	Login 02	**Test Case ID**	Employer Login 02
Test Case Description	Employer login with valid user details of non-activated account	**Test Priority**	Medium
Pre-Requisite	A valid and non-activated Employer account	**Post-Requisite**	User stay logged in until hitting the log out button

Test Execution Steps:

S.No	Action	Inputs	Expected Output	Actual Output	Test Browser	Test Result	Test Comments
1	Direct to the "Public" login page	http://localhost/futureseekerslk/public/	Load "Public" Login page	Load "Public" Login page	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost
2	Enter valid username & Password and hit login button	Username: tharindukathriarachch Password: tharindukathriarachchi	Successfully log in to Employer account and direct to the My Profile page.	Logged in to Employer account and directed to the My Profile page.	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost
3	Entering company details	Typing any text in company details fields	Company details fields should be locked	User can enter data to text fields	Google Chrome Version 95.0.4638.69	Fail	Tested in localhost
4	Hitting "Make Changes" Button	NA	"Make Changes" button should be unclickable	"Make Changes" button is unclickable	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost

Test Scenario ID	Login 02	**Test Case ID**	Employer Login 03
Test Case Description	Employer login with valid user details of an activated account with empty company details	**Test Priority**	Medium
Pre-Requisite	A valid and activated Employer account with empty company details	**Post-Requisite**	User stay logged in until hitting the log out button

Test Execution Steps:

S.No	Action	Inputs	Expected Output	Actual Output	Test Browser	Test Result	Test Comments
1	Direct to the "Public" login page	http://localhost/futureseekerslk/public/	Load "Public" Login page	Load "Public" Login page	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost
2	Enter valid username & Password and hit login button	Username: rashmikaishan Password: rashmikaishan	Successfully log in to Employer account and direct to the Jobs Page.	Logged in to Employer account and directed to the My Profile Page.	Google Chrome Version 95.0.4638.69	Fail	Tested in localhost
3	Entering company details	Typing any text in company details fields	User should be Able to enter Company details	User can enter data to text fields	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost
4	Hitting "Make Changes" Button	NA	"Make Changes" button should be clickable	"Make Changes" button is clickable	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost

Test Scenario ID	Login 02	**Test Case ID**	Employer Login 04
Test Case Description	Employer Login – With invalid user details	**Test Priority**	High
Pre-Requisite	NA	**Post-Requisite**	NA

Test Execution Steps:

S.No	**Action**	**Inputs**	**Expected Output**	**Actual Output**	**Test Browser**	**Test Result**	**Test Comments**
1	Direct to the "Public" login page	http://localhost/futureseekerslk/public/	Load "Public" Login page	Load "Public" Login page	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost
2	Enter invalid Username & Any Password and enter login button	Username: asss Password: haha	Displaying "This username does not exist" under the username field	Displayed "This username does not exist" under the username field	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost
3	Enter valid username & incorrect Password and enter login button	Username: joeblack Password: adminfs123111	Displaying "Incorrect Password!" error massage	Displaying "Incorrect Password!" error massage	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost

12.3 Log in [Applicant]

Following test cases are conducted to test the log in function and log in page UI element for the “Applicant” accounts

Test Scenario ID	Login 03	**Test Case ID**	Applicant Login 01
Test Case Description	Applicant login with valid user details of an approved account	**Test Priority**	Medium
Pre-Requisite	Approved Applicant account	**Post-Requisite**	User stay logged in until hitting the log out button

Test Execution Steps:

S.No	Action	Inputs	Expected Output	Actual Output	Test Browser	Test Result	Test Comments
1	Direct to the “Public” login page	http://localhost/futureseekerslk/public/	Load “Public” Login page	Load “Public” Login page	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost
2	Enter valid username & Password and hit login button	Username: sherlockholmes Password: sherlockholmes	Successfully log in to Applicant account and direct Applicant landing page	Logged in to Applicant account and direct Applicant landing page	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost. Implementations of the Applicant landing page comes in future sprints

Test Scenario ID	Login 03	**Test Case ID**	Applicant Login 02
Test Case Description	Applicant login with valid user details of non-activated account	**Test Priority**	Medium
Pre-Requisite	A valid and non-activated Applicant account	**Post-Requisite**	User stay logged in until hitting the log out button

Test Execution Steps:

S.No	Action	Inputs	Expected Output	Actual Output	Test Browser	Test Result	Test Comments
1	Direct to the "Public" login page	http://localhost/futureseekerslk/public/	Load "Public" Login page	Load "Public" Login page	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost
2	Enter valid username & Password and hit login button	Username: dunilshayapa Password: dunilshayapa	Successfully log in to Applicant account and direct to the My Profile page.	Logged in to Applicant account and directed to the My Profile page.	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost
3	Entering Application Details	Typing any text in Application Details fields	Application Details fields should be locked	User can enter data to text fields	Google Chrome Version 95.0.4638.69	Fail	Tested in localhost
4	Hitting "Save Changes" Button	NA	"Save Changes" button should be unclickable	"Save Changes" button is unclickable	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost

Test Scenario ID	Login 03	**Test Case ID**	Applicant Login 03
Test Case Description	Applicant Login – With invalid user details	**Test Priority**	High
Pre-Requisite	NA	**Post-Requisite**	NA

Test Execution Steps:

S.No	Action	Inputs	Expected Output	Actual Output	Test Browser	Test Result	Test Comments
1	Direct to the "Public" login page	http://localhost/futureseekerslk/public/	Load "Public" Login page	Load "Public" Login page	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost
2	Enter invalid Username & Any Password and enter login button	Username: bass Password: buus	Displaying "This username does not exist" under the username field	Displayed "This username does not exist" under the username field	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost
3	Enter valid username & incorrect Password and enter login button	Username: dunilshayapa Password: sss	Displaying "Incorrect Password!" error massage	Displaying "Incorrect Password!" error massage	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost

12.4 Register [Employer]

Following test cases are conducted to test the Register function and Register page UI element for the "Employer" accounts

Test Scenario ID	Register 01	**Test Case ID**	Register Employee 01
Test Case Description	Successful path. (Registering providing all valid data)	**Test Priority**	High
Pre-Requisite	Systems registering function should be up and running	**Post-Requisite**	A new User account will be created

Test Execution Steps:

S.No	Action	Inputs	Expected Output	Actual Output	Test Browser	Test Result	Test Comments
1	Direct to the "Public" login page	http://localhost/futureseekerslk/public/	Load "Public" Login page	Load "Public" Login page	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost
2	Direct to "Employer Registration" page	Selecting "Are you an Employer? Register Now" link	Load "Employer Registration page"	Load "Employer Registration page"	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost
3	Enter valid data and hit "Register Now" button	Full Name; Nuwan Kularathna Current Job Position; Director – HR Email; Contact No; 0778445112 nk@gmail.com Comapany Name; CLS Username: nuwankularathna Password: nuwankularathna	Successfully Creating a new "Employer" account and displaying "User Registration Successful" massage then, direct to the "Log in page" page.	Successfully Created a new "Employer" account and displayed "User Registration Successful" massage then, stays in the "Employer Registration" page.	Google Chrome Version 95.0.4638.69	Fail	Tested in localhost [Amount creation function works properly. Just the directing to the "log in" page doesn't work]

Test Scenario ID	Register 01	**Test Case ID**	Register Employee 02
Test Case Description	Unsuccessful path. (Registering With invalid data valid data)	**Test Priority**	High
Pre-Requisite	Systems registering function should be up and running	**Post-Requisite**	NA

Test Execution Steps:

S.No	Action	Inputs	Expected Output	Actual Output	Test Browser	Test Result	Test Comments
1	Direct to the "Public" login page	http://localhost/futureseekerslk/public/	Load "Public" Login page	Load "Public" Login page	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost
2	Direct to "Employer Registration" page	Selecting "Are you an Employer? Register Now" link	Load "Employer Registration page"	Load "Employer Registration page"	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost
3	Enter invalid email format and all other fields with valid data.	Full Name; Nuwan Kularathna Current Job Position; Director – HR Email; Contact No; 0778445112 abc Comapany Name; CLS Username: nuwankularathna Password: nuwankularathna	Displaying the pop up error message saying that the email format needs @ sign	Displaying the error message	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost

4	Enter already existing username with valid data for other fields	Full Name; Nuwan Kularathna Current Job Position; Director – HR Email; Contact No; 0778445112 nk@gmail.com Comapany Name; CLS Username: nuwankularathna Password: nuwankularathna	Displaying “This username already exist” under the username field	Displayed“This username already exist” under the username field	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost
5	Leave all fields free and hit “Register Now” button	Non	Display an error message in format of “ ‘Field name’ is required” under all fields	Displayed all error messages	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost

12.5 Register [Applicant]

Following test cases are conducted to test the Register function and Register page UI element for the "Employer" accounts

Test Scenario ID	Register 02	**Test Case ID**	Register Applicant 01
Test Case Description	Successful path. (Registering providing all valid data)	**Test Priority**	High
Pre-Requisite	Systems registering function should be up and running	**Post-Requisite**	A new User account will be created

Test Execution Steps:

S.No	**Action**	**Inputs**	**Expected Output**	**Actual Output**	**Test Browser**	**Test Result**	**Test Comments**
1	Direct to the "Public" login page	http://localhost/futureseekerslk/public/	Load "Public" Login page	Load "Public" Login page	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost
2	Direct to "Applicant Registration" page	Selecting "Are you an Applicant? Register Now" link	Load "Applicant Registration page"	Load "Applicant Registration page"	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost
3	Enter valid data and hit "Register Now" button	Full Name; Namal Rajapaksha Address; 557/A8 Hamnbanthota Email; na@gmail.com Contact No; 0778995336 Date of Birth; 10-08-1996 Current Job Title; BA Username; namalrajapaksha Password; namalrajapaksha	Successfully Creating a new "Applicant" account and displaying "User Registration Successful" massage then, direct to the "Log in page" page.	Successfully Created a new "Applicant" account and displayed "User Registration Successful" massage then, stays in the "Applicant Registration" page.	Google Chrome Version 95.0.4638.69	Fail	Tested in localhost [Amount creation function works properly. Just the directing to the "log in" page doesn't work]

Test Scenario ID	Register 02	**Test Case ID**	Register Applicant 02
Test Case Description	Unsuccessful path. (Registering With invalid data valid data)	**Test Priority**	High
Pre-Requisite	Systems registering function should be up and running	**Post-Requisite**	NA

Test Execution Steps:

S.No	Action	Inputs	Expected Output	Actual Output	Test Browser	Test Result	Test Comments
1	Direct to the "Public" login page	http://localhost/futureseekerslk/public/	Load "Public" Login page	Load "Public" Login page	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost
2	Direct to "Applicant Registration" page	Selecting "Are you an Applicant? Register Now" link	Load "Applicant Registration page"	Load "Applicant Registration page"	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost
3	Enter invalid email format and all other fields with valid data.	Full Name; Namal Rajapaksha Address; 557/A8 Hamnbanthota Email; na Contact No; 0778995336 Date of Birth; 10-08-1996 Current Job Title; BA Username; namalrajapaksha Password; namalrajapaksha	Displaying the pop up error message saying that the email format needs @ sign	Displaying the pop up error message saying that the email format needs @ sign	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost

4	Enter already existing username with valid data for other fields	Full Name; Namal Rajapaksha Address; 557/A8 Hamnbanthota Email; na@gmail.com Contact No; 0778995336 Date of Birth; 10-08-1996 Current Job Title; BA Username; namalrajapaksha Password; namalrajapaksha	Displaying "This username already exist" under the username field	Displayed"This username already exist" under the username field	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost
5	Leave all fields free and hit "Register Now" button	Non	Display an error message in format of " 'Field name' is required" under all fields	Displayed all error messages	Google Chrome Version 95.0.4638.69	Pass	Tested in localhost