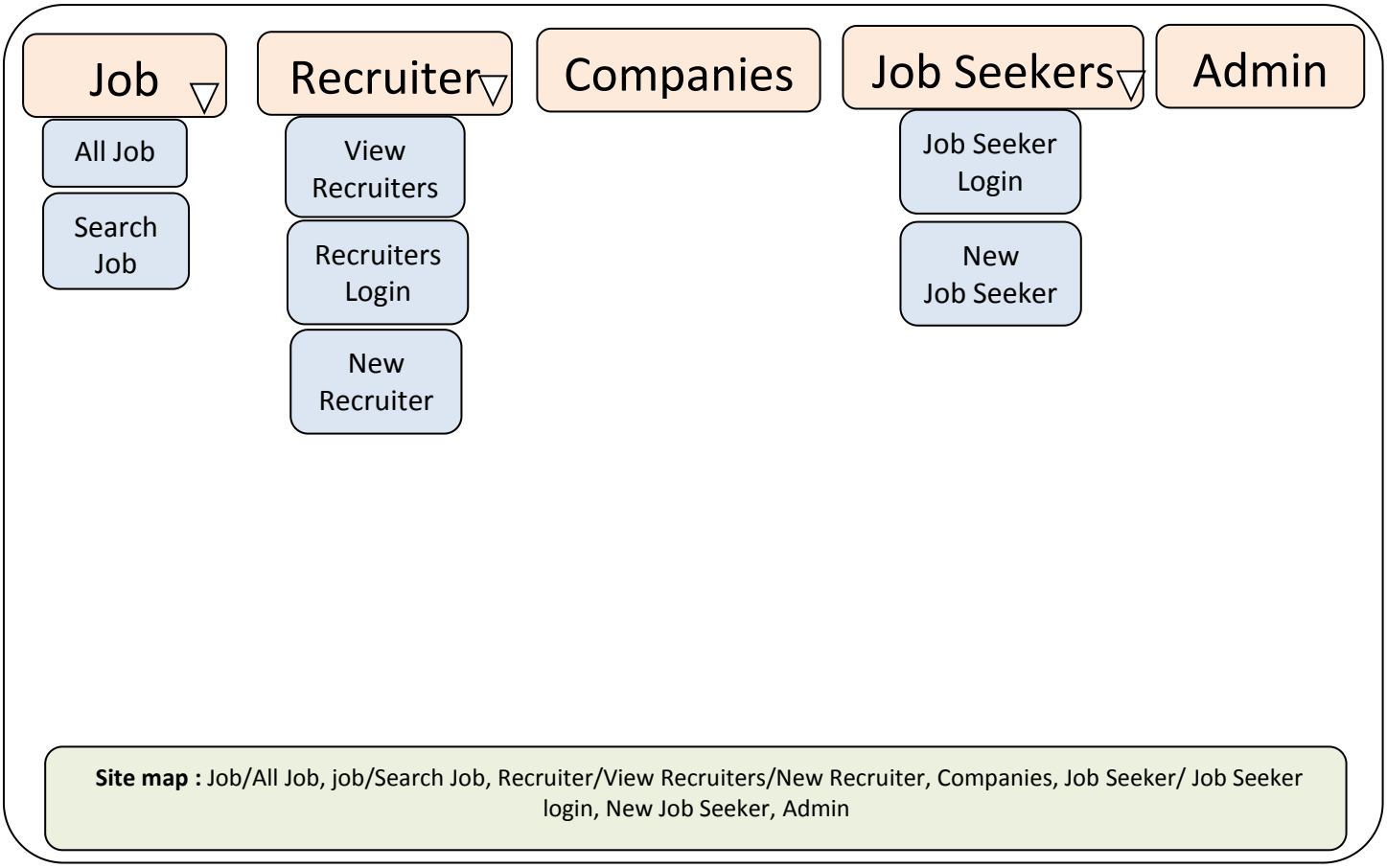
ASP.NET Project

Project Name: Online Job Portal
Front-end Design: HTML, CSS, JavaScript & ASP.NET Control
Back-end Coding: ASP.NET with C#.NET Code.
Back-end Database: SQL Database (MDF)

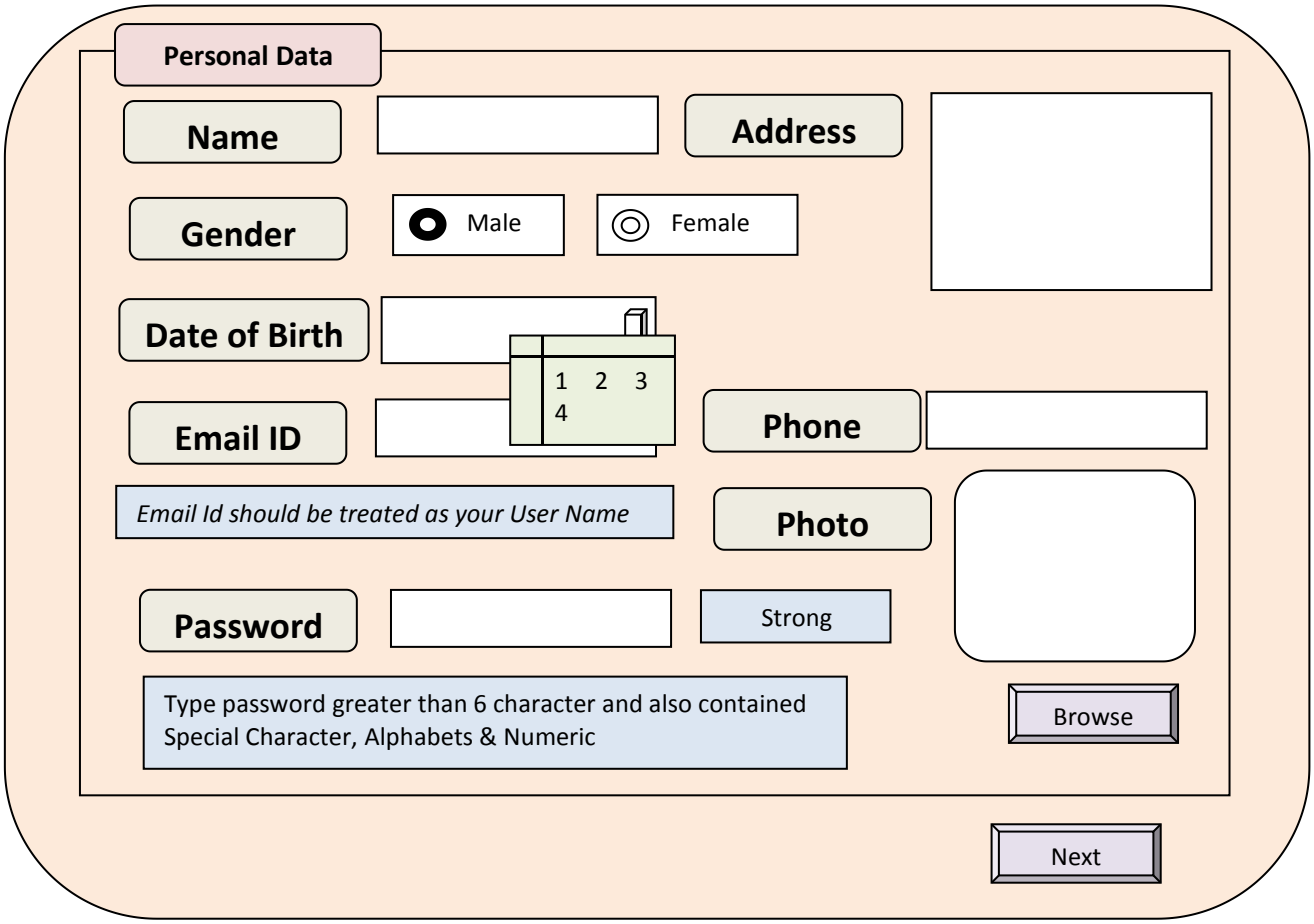
Page 1: Home Page

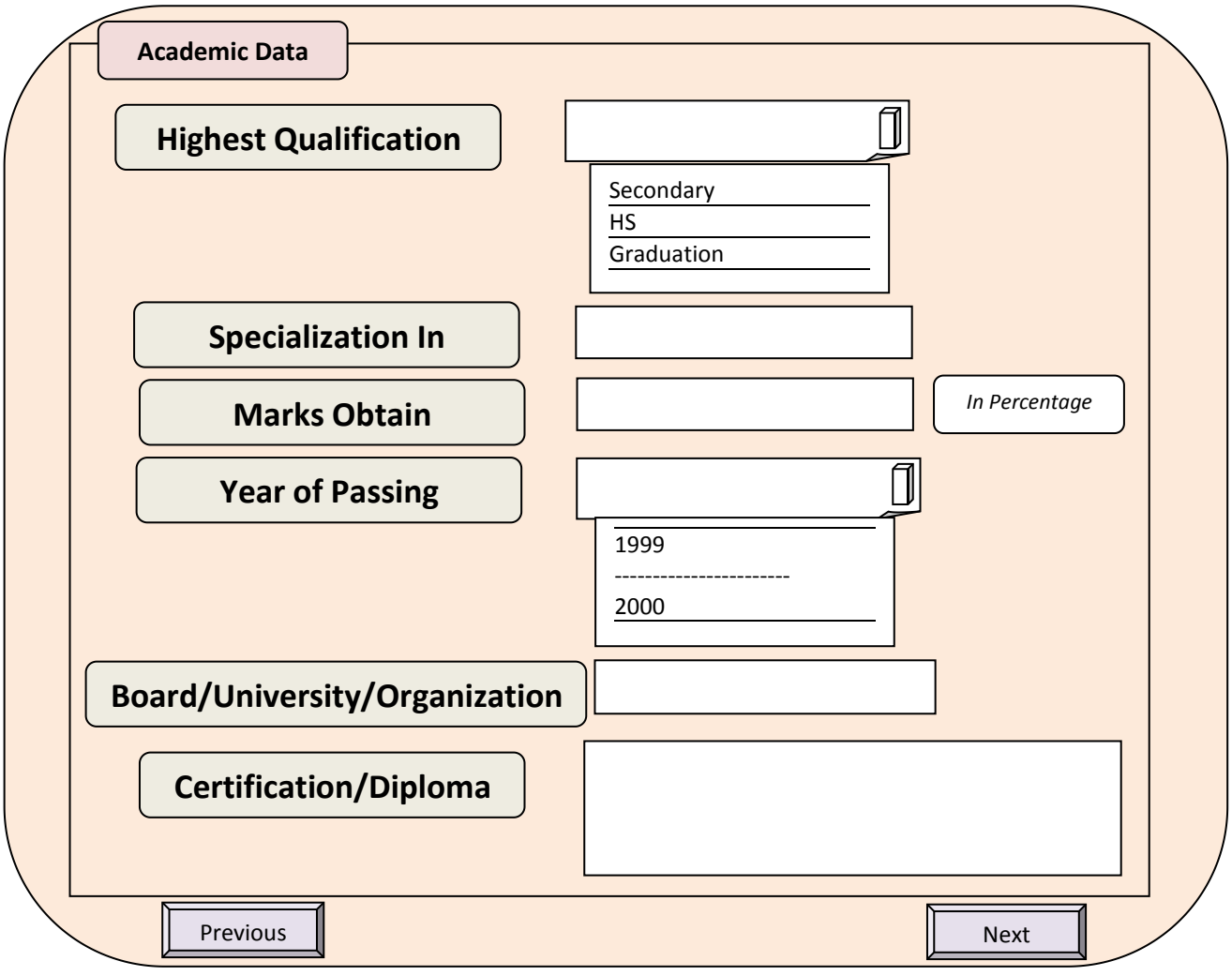


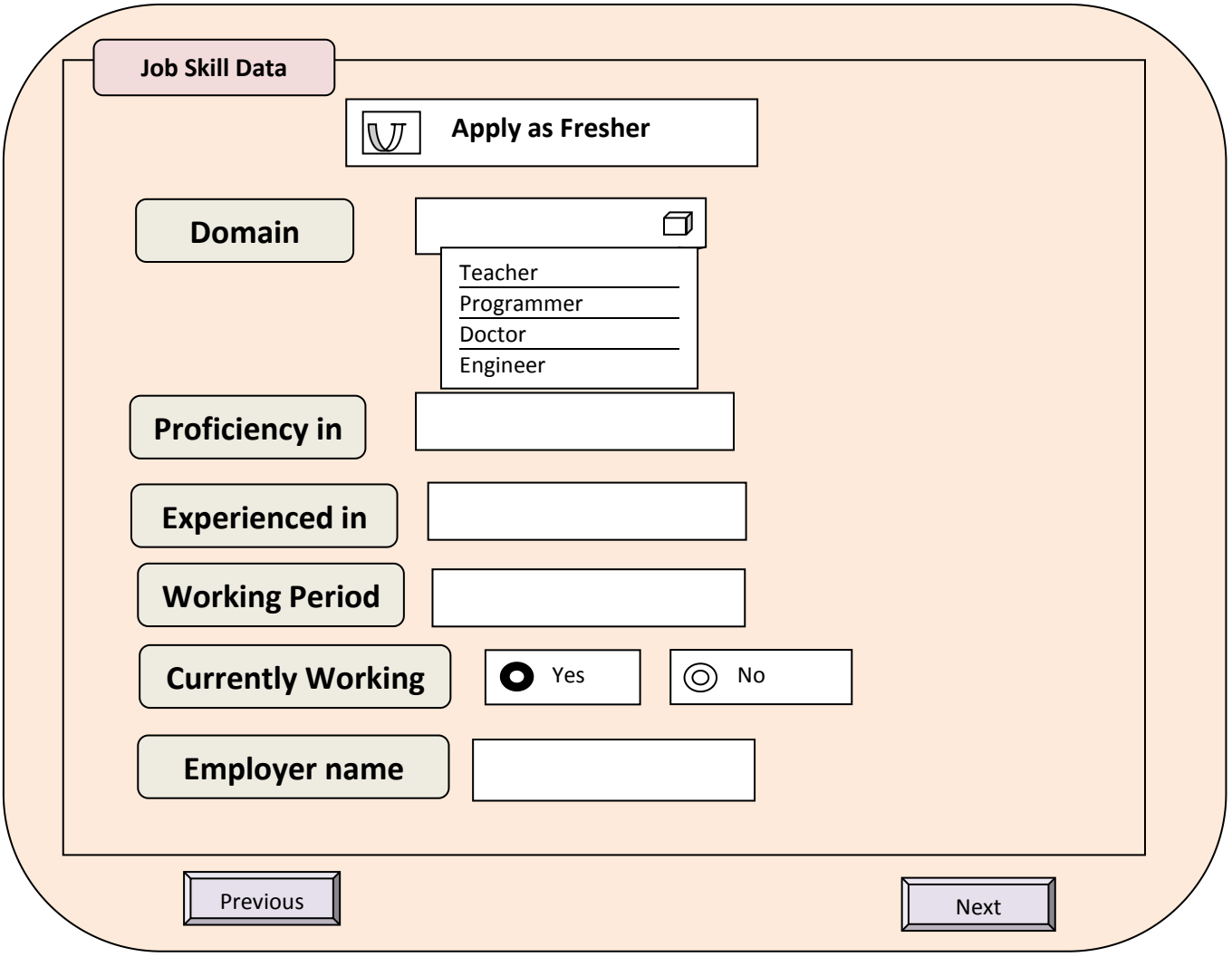
Algorithm of the Home page: A viewer should be at first display the Home Page and he/she should be doing following works.

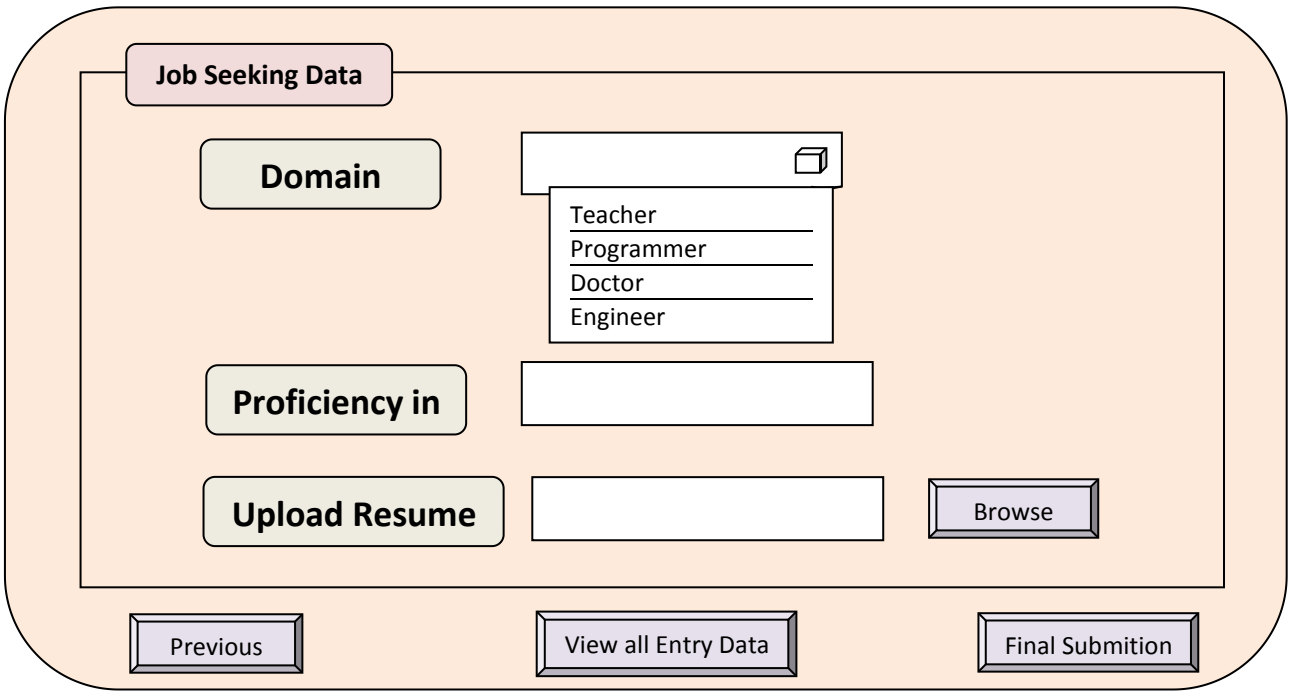
1) **Job a) All Job:** View list of all posted Jobs.
   b) **Search Job:** Search or Filter Job list by customize criteria like skill, qualification, salary, domain
2) **Recruiter, a) View Recruiter:** View all recruiters who already registered to this system.
   B) **Recruiter Login:** A registered recruiter should be login to the recruiter panel.
   C) **New recruiter;** A person should be register to this system as a recruiter.
3) **Companies:** Display all companies placement status.
4) **Job Seeker, a) job Seeker Login:** A registered job seeker should be login to Job Seeker Panel.
   B) New Job Seeker: A person should be register to this system as a Job Seeker.
5) **Admin:** This the admin panel for the Website.

Page 2: New Job Seeker Page









Algorithm about New Job Seeker Page:

A user should be register him/her as a Job Seeker, then he/she have to fill up the above form, then he/she should be login to the system by entering email id as username and type password. After that he/she will be access to the Job Seeker Panel.

All data should be insert into the JobSeeker Table within JobPortal Database.

Database:

Database Name: JobPortal.mdf

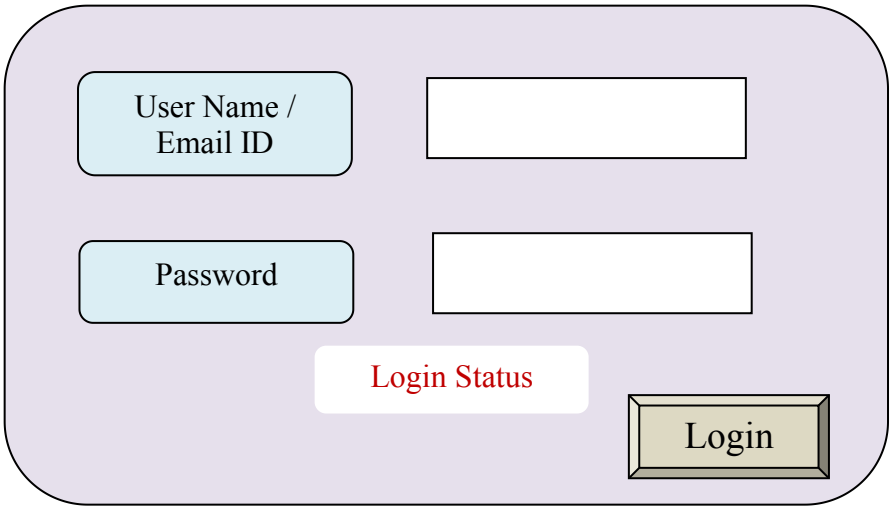
Table Name:
Table 1: JobSeeker

Table Structure

Field Name	Data-type	Length	Constraints
Jobseeker_id	int	4	Primary Key, Not Null and Auto Increment
Sname	Varchar	15	Not Null
Address	Varchar	50	Not Null
Gender	Varchar	8	Not Null
dob	Varchar	15	Not Null
Phone	Varchar	10	Not Null
Email	Varchar	20	Not Null
Passwords	Varchar	10	Not Null
Photo	Varchar	50	Not Null
Highquali	Varchar	15	Not Null
Specialization	Varchar	15	Not Null
marks	Int	4	Not Null
passyear	Varchar	15	Not Null
Univ	Varchar	15	Not Null
Certificate	Varchar	15	Null
Job_status	Varchar	15	Not Null
Domain	Varchar	15	Not Null
Proficiency	Varchar	15	Not Null
experience	Varchar	15	Not Null
Work_period	Varchar	15	Not Null
Currently_work	Varchar	15	Not Null
Company_name	Varchar	15	Not Null
Apply_domain	Varchar	15	Not Null
Apply_proficiency	Varchar	15	Not Null
Resume	Varchar	15	Not Null

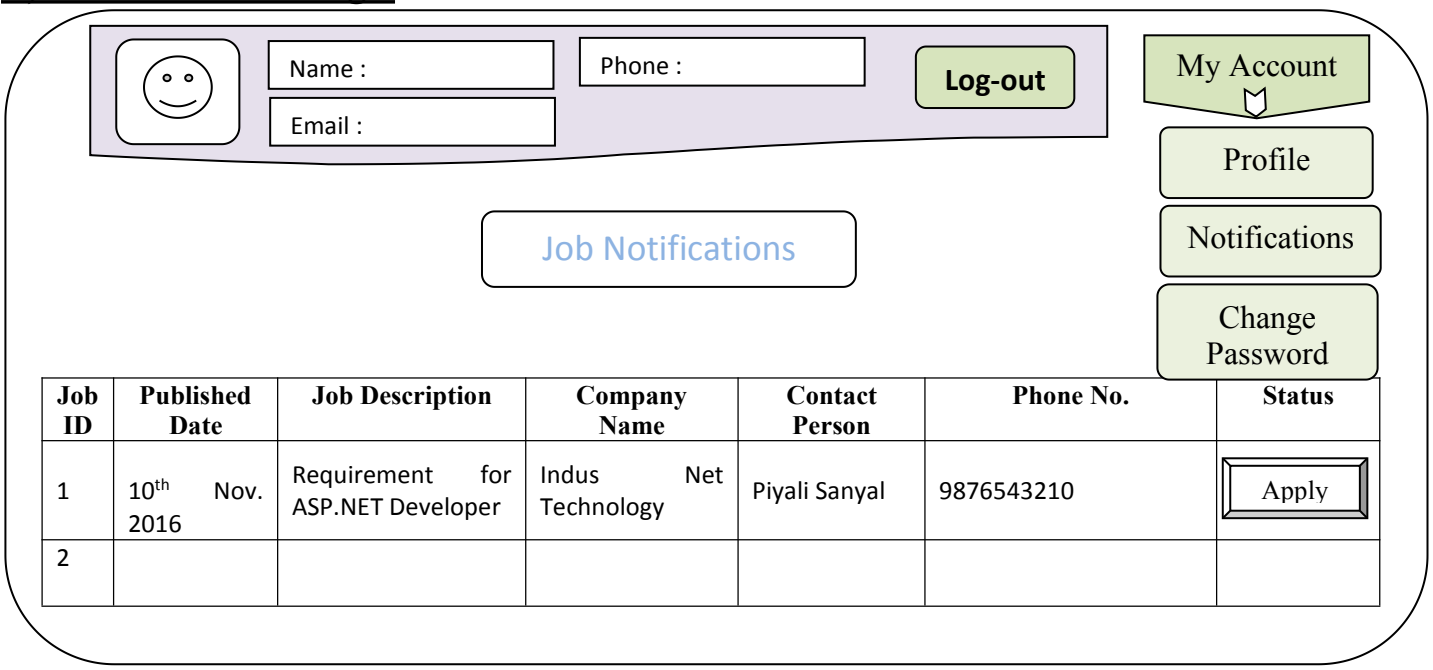
Algorithm behind JobSeeker Table:
1) Jobseeker_id should be in primary key, not null and auto-incremented
2) Photo field should carry the uploaded image full name, but all uploaded image files should be store in your server directory.
3) Job_status should be carry the data Fresher or Experience
4) Currently_work should be carry the data Yes or No
5) Resume field should carry the uploaded resume file name, but all uploaded resume files should be store in your server directory.

Page 3: Job Seeker Login Page:



Algorithm of Job Seeker Login Page:
A registered Job Seeker should be login into the system by entering Email ID and Password.
These information should be fetch from JobSeeker Table by back-end coding.

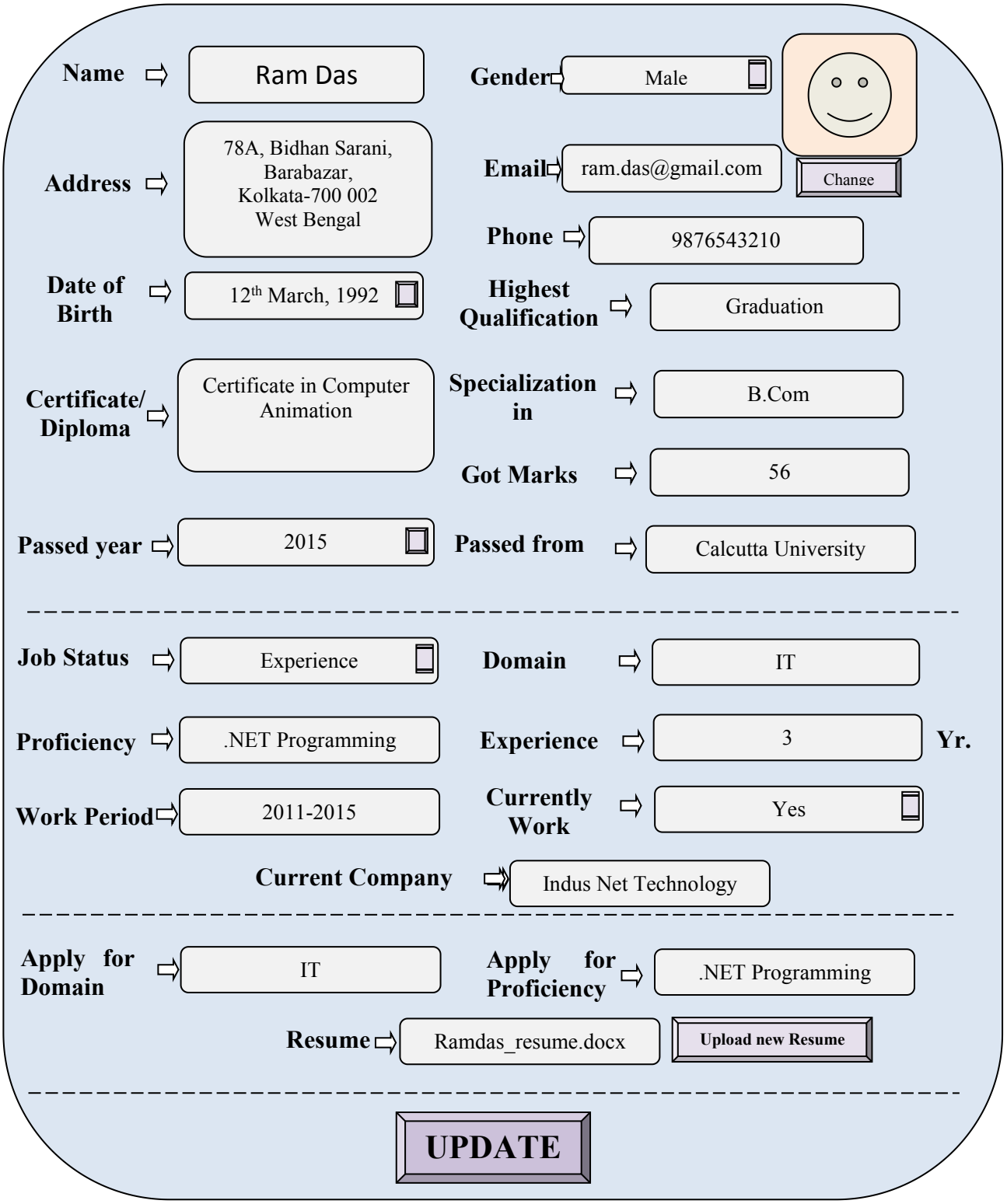
Page 4: Job Seeker Panel Page:
A) JobNotification Page:



Job ID	Published Date	Job Description	Company Name	Contact Person	Phone No.	Status
1	10th Nov. 2016	Requirement for ASP.NET Developer	Indus Net Technology	Piyali Sanyal	9876543210	Apply
2						

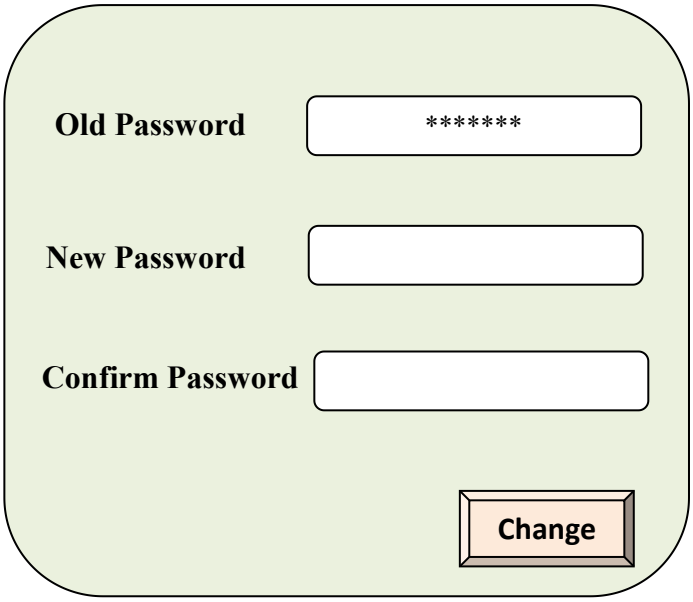
Algorithm of Job Seeker Panel Page and Job Notification Page:
1) A registered Job Seeker should be login into the Job Seeker Panel where he/she should be follow a Job Notification Page and from here he/she should apply for Job.
2) Also there will be a menu My Account to navigate through Profile Page and Job Notification.
3) The Job Notification table should be displayed the matching the field Apply_Proficiency of JobSeeker Table and Job_Description field of JobPosted Table.
4) A Job Seeker when click on Apply Button, then the JobSeeker_ID and JobPosted_ID will be insert into JobApplied Table.

B) Profile Page:



Algorithm of Job Seeker Profile Page:
A registered Job Seeker should be modify or update his/her profile information including upload new resume.

C) Change Password Page:



Algorithm of Job Seeker Password Change Page:
1) A registered Job Seeker should be change his/her old password by replacing new password.
2) The Job Seeker table record should be updates it's password entry.

D) Log-out Job Seeker Panel Page:

This is have no design, it has only back-end code to logout from JobSeeker Panel.

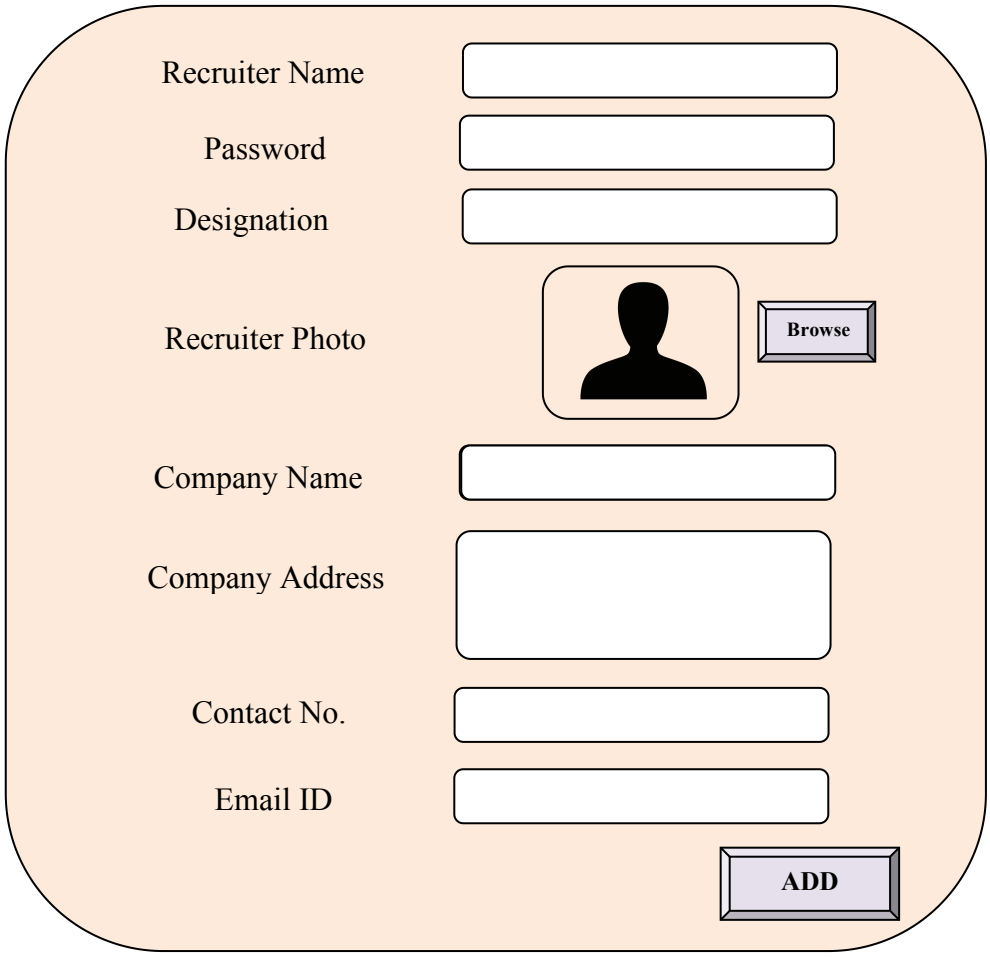
Table Name:
Table 1: JobRecruiter

Table Structure

Field Name	Data Type	Length	Constraints
Recruiter_id	Int	4	Primary Key, Not Null and Auto Increment
Recruiter_name	Varchar	15	Not Null
passwords	Varchar	15	Not Null
Recruting_company	Varchar	15	Not Null
Company_location	Varchar	15	Not Null
Recruiter_designation	Varchar	15	Not Null
Recruiter_email	Varchar	15	Not Null
Recruiter_phone	Varchar	15	Not Null
Recruiter_photo	Varchar	15	Not Null

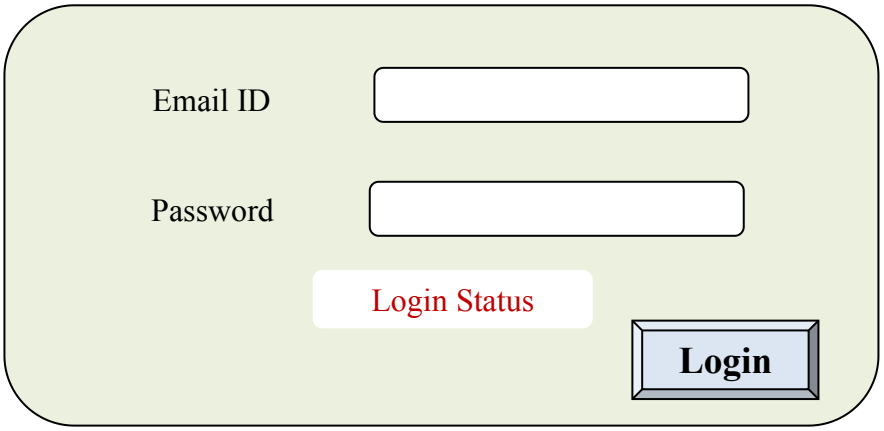
Algorithm behind JobRecruiter Table:
1) In the jobRecruiter table store all information about Recruiter details.
2) Recruiter_id should be in primary key, not null and auto-incremented
3) Recruiter_photo field should carry the uploaded image full name, but all uploaded image files should be store in your server directory.

Page 5: New Recruiter Page:



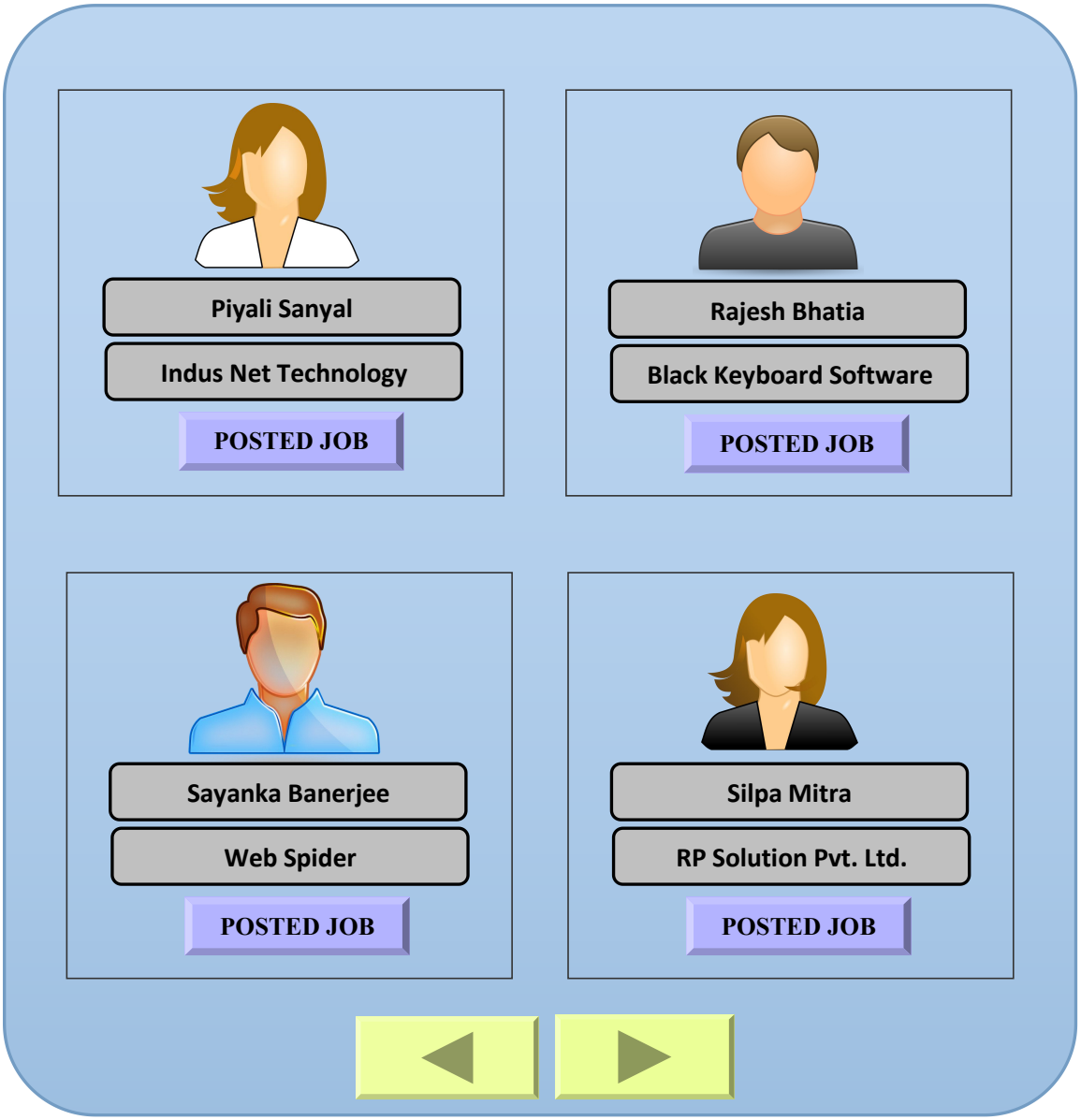
Algorithm behind New Recruiter Page:
1) A new recruiter should be enter his/her relevant information through this form.
2) These all information should be store into JobRecruiter Table.

Page 6: Login Recruiter Page:



Algorithm behind Login Recruiter Page:
1) A registered recruiter should be login into Recruiter Panel by entering his/her Email ID and Password.
2) These all information should be search from JobRecruiter Table.

Page 7: View Recruiters Page:



Algorithm behind View Recruiter Page:

1) Any person should be view details about all recruiters who are already registered in the system.
2) Click on "Posted Job" Button to see Posted Job List according to individual recruiters.
3) All records should be displayed in DataList Control and one page maximum display four recruiter details in individual box, remains details should be displayed in Pagging Concept by next &previous navigation.

Table Name:
Table 1: JobPosted

Table Structure

Field Name	Data Type	Length	Constraints
Jobposted_id	Int	4	Primary Key, Not Null and Auto Increment
Job_description	Varchar	15	Not Null
Job_Posted_Date	Varchar	15	Not Null
Job_Status	Varchar	15	Not Null
No_Vacancies	int	4	Not Null
Recruiter_id	int	4	Foreign key of JobRecruiter Table

Algorithm behind JobPosted Table:
1) This table store the Job details posted by recruiters
2) The Job_Status carry the information of Applicant Status that is Fresher/Experienced.
3) The Recruiter_id field should be a foreign key against the another Recruiter_id field of JobRecruiter Table.

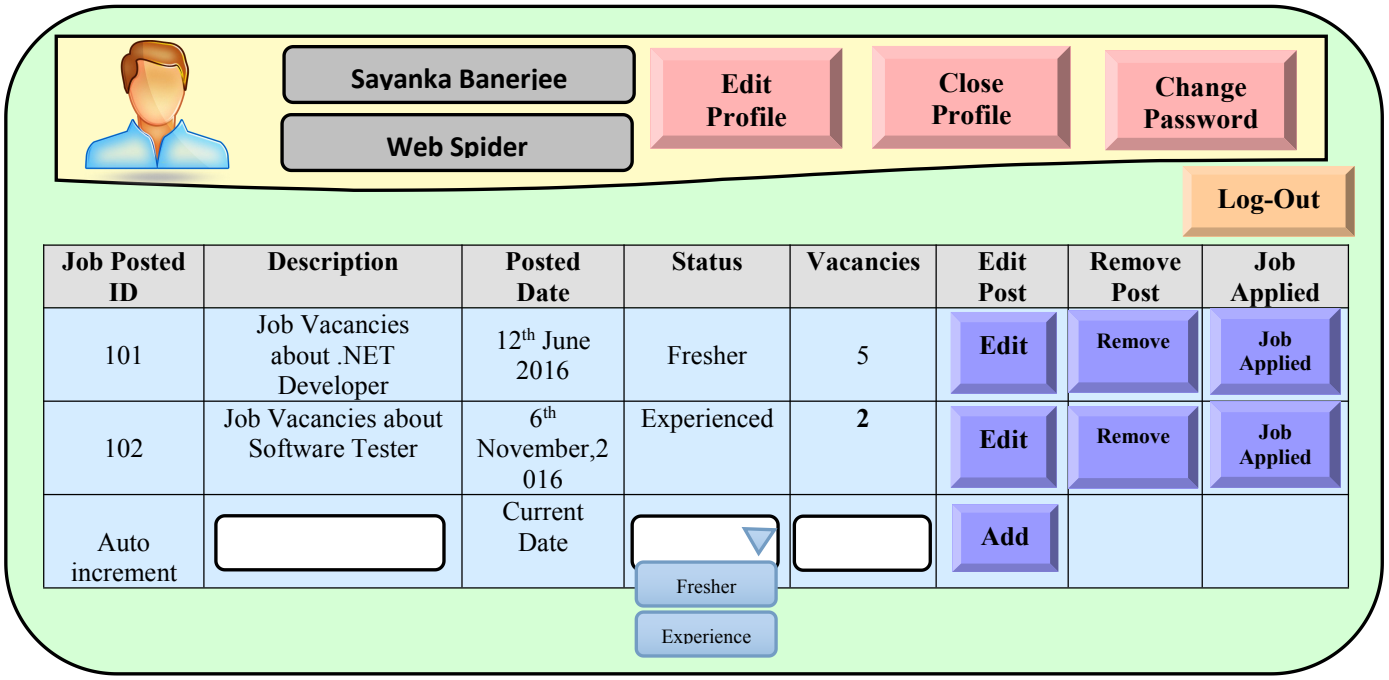
Table Name:
Table 1: JobApplied

Table Structure

Field Name	Data Type	Length	Constraints
Apply_id	Int	4	Primary Key, Not Null and Auto Increment
Jobposted_id	Int	4	Foreign key of JobPosted Table
Jobseeker_id	int	4	Foreign key of JobSeeker Table
Jobapply_date	Varchar	15	Not Null

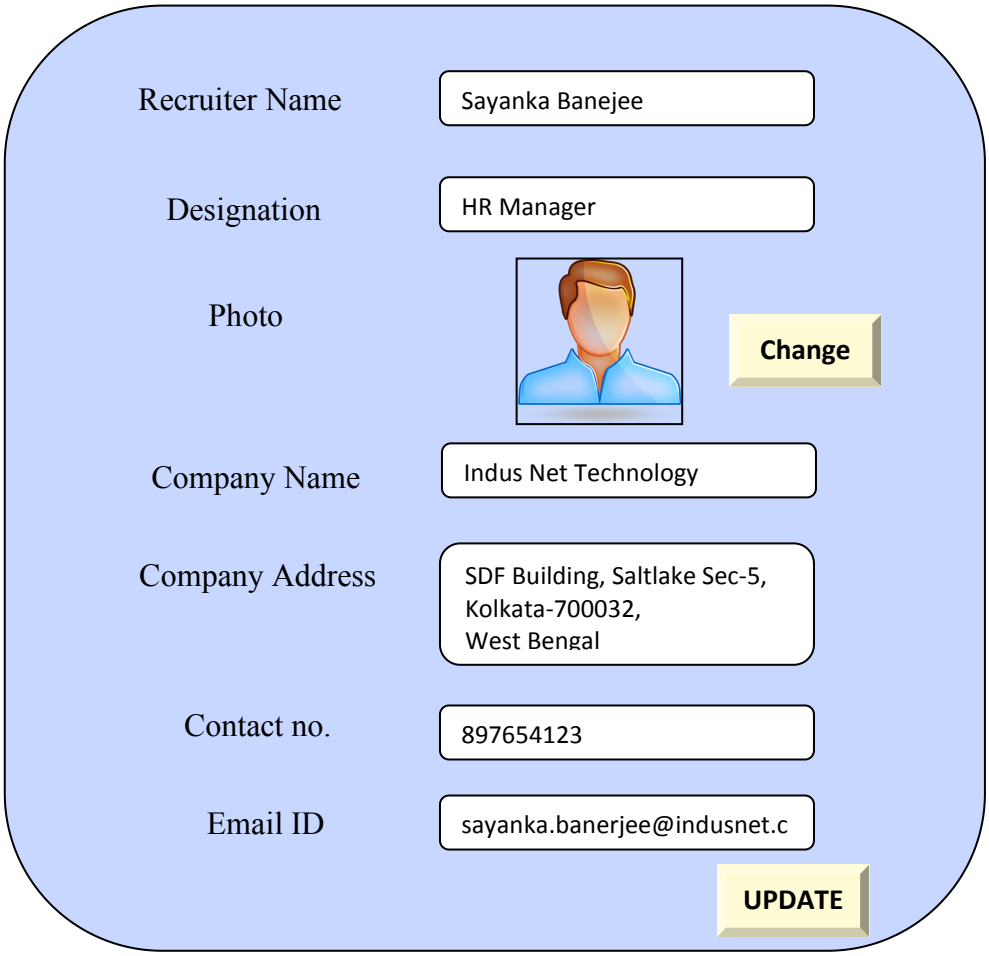
Algorithm behind JobApplied Table:
1) This table store the Applied Job details by Job Seeker
2) The Jobposted_id field should be a foreign key against the another Job_id field of JobPosted Table.
3) The Jobseeker_id field should be a foreign key against the another Job_id field of JobSeeker Table.

Page 8: Job Recruiters Dashboard Page:



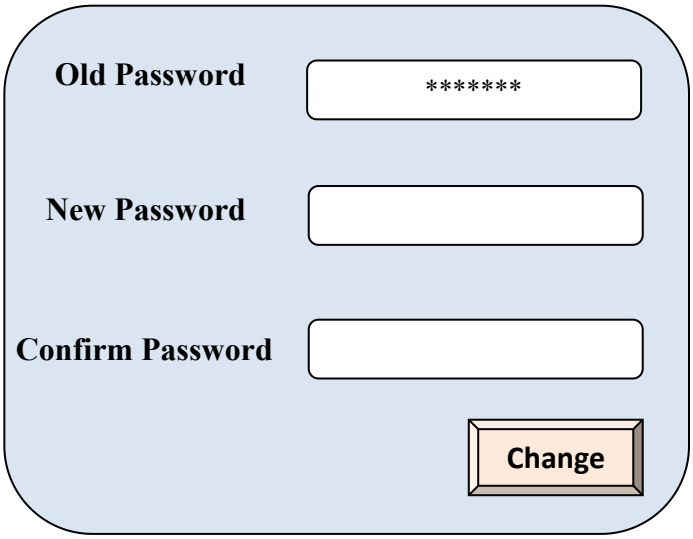
Algorithm behind Job Recruiter Dashboard Page:
1) This page should be displayed after successfull login by Job Recruiter
2) From this page a Recruiter can view the list of posted job, edit existing post, remove post and also review the details of the Job Seeker applied for a job.
3) The Job Recruiter should be edit and close his/her profile, also change his/her profile password.

Page 8-A: Job Recruiters Edit Page:



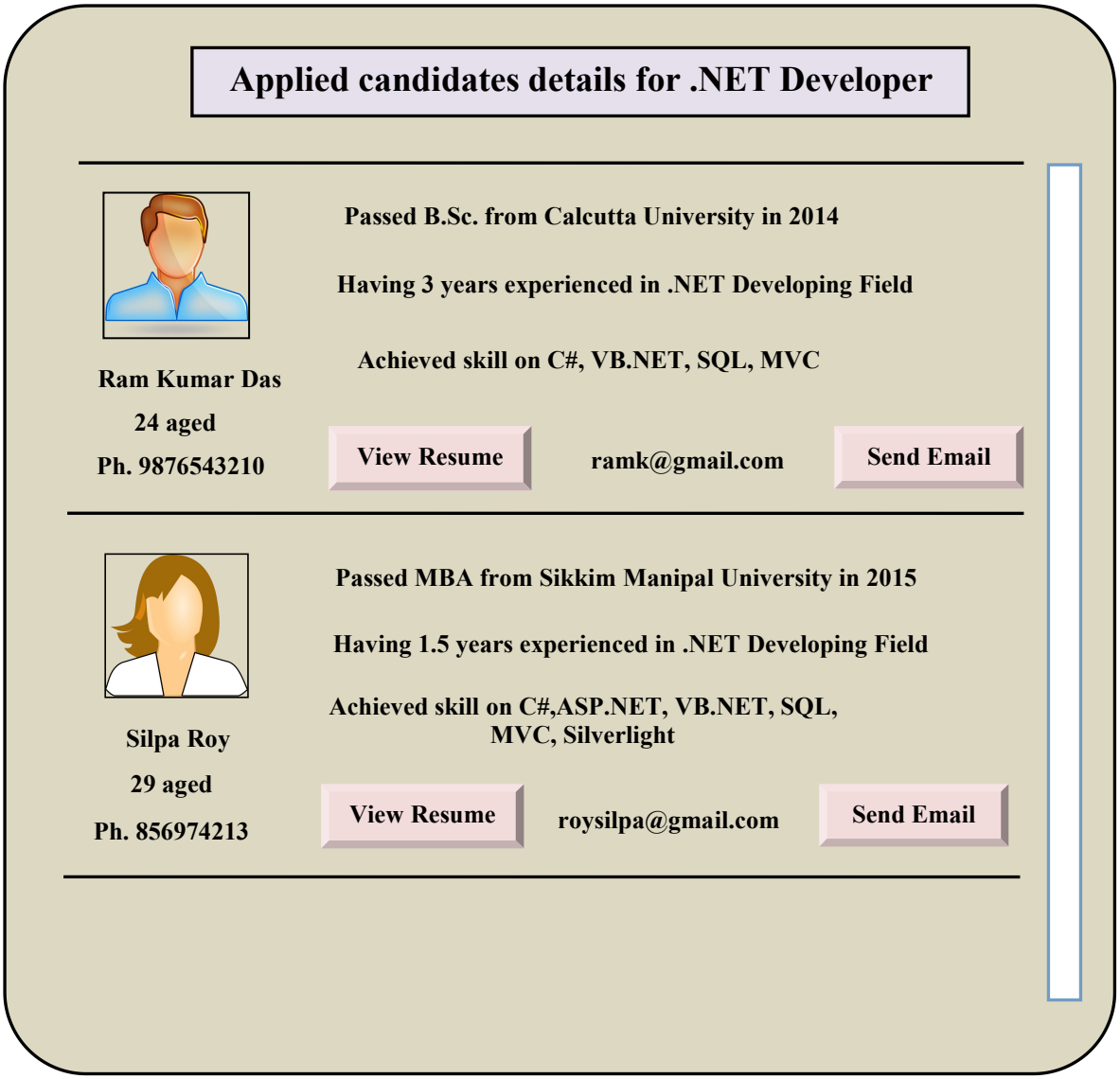
Algorithm behind Job Recruiter Edit Page:
A Recruiter should be edit his/her profile information including photo but except password.

Page 8-B: Change Password Page:



Algorithm of Job Recruiter Password Change Page:
3) A registered Job Recruiter should be change his/her old password by replacing new password.
4) The JobRecruiter table record should be updates it's password entry.

Page 8-C: Job Applied Page:



Algorithm of Job Applied Page:
1) A registered Job Recruiter should be view all details of Candidate which those applied for the specific job.
2) These record should be fetch from JobApplied table depends on Posted_Job ID and Job_Seeker ID.
3) The Job Recruiter should be send email to a candidate for invitation about interview.

****

End of Project