# Online Job Portal

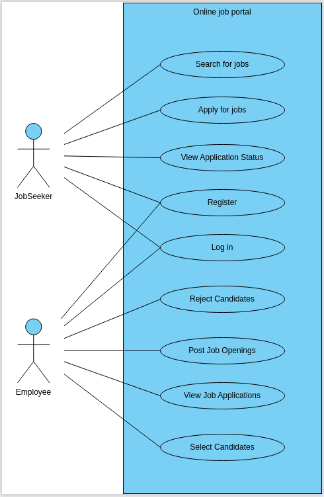
## 1.Introduction

Our project aims to develop a comprehensive Job Portal platform that caters to both Employers and Job Seekers, providing a seamless and effective means for job posting, application, and management. This portal will serve as a bridge between companies searching for talent and individuals seeking employment opportunities.

2. Project Objectives:

The main objectives of the Online Job portal are to:

* The objective of our Job Portal project is to streamline the job-seeking process for candidates by providing a user-friendly platform for efficient job discovery and simplified application procedures.
* Simultaneously, the portal aims to empower employers with tools for easy job posting and effective management of applications.
* Real-time application tracking, secure user authentication, and mobile responsiveness are prioritized to enhance the overall user experience.



## 3. Functional Requirements

### #1 Registration & Login:

a. Registration:

* Users should be able to create a new account by providing necessary details such as name, email, password, etc.
* Implement email verification for user registration.

b. Login:

* Users (Employers and Job Seekers) should be able to log in with their registered email and password.
* Include an option for password recovery/reset.

c. Landing:

* Upon successful login, users should be directed to a personalized landing page based on their role (Employer/Job Seeker).
* -Provide navigation options for the main features of the platform.

d. Jobs:

* Users should be able to browse and search for available jobs.
* Display a list of job posts with basic details (title, company, location, etc.).

e. Job Details:

* Clicking on a job post should open a detailed view with information such as job description, requirements, application instructions, etc.

### #2 . For Employer:

a. Create Job:

* Employers should have the ability to create a new job post.
* Include fields for job title, company name, location, job description, requirements, application deadline, etc.

b. Job Listing:

* Employers should be able to view a list of all the job posts they have created.
* Include options to edit, delete, or deactivate a job post.

c. See All Job Applications:

* Employers should have a centralized view of all job applications received.
* Include sorting and filtering options based on application status, date, etc.

#3 For Job Seeker:

a. My Jobs:

* Job seekers should have a dashboard showing the jobs they have applied to.
* Include status updates for each application (e.g., pending, reviewed, rejected).

b. Apply:

* Job seekers should be able to apply for a job by submitting their resume and a personalized cover letter.
* Include a confirmation message upon successful application.

c. See All Job Posts:

* Job seekers should have access to a comprehensive list of all available job posts.
* Include sorting and filtering options based on job title, company, location, etc.

d. Status of Application:

* Job seekers should be able to track the status of their job applications.
* Include notifications or updates on any changes in application status.

### 4. Technology Stack:

* Back End: ASP.Net Core Web Api
* Front End: Angular
* Database : SQL Server

### Requirements Phase

* You need to interact with each other about the functionality in detail, the error scenarios.
* Provide a description not exceeding 10 lines about the functionality present/being built, call out error scenarios.

### Design Phase

* Create an ERD
* Determine what values need to be stored for each entity and determine suitable

1. primary keys
2. foreign keys
3. indexes

* Decide what entities we need to consider in a data model
* List down the assumptions does your model make?
* List down the information in the text was irrelevant to your design?
* List down any further restrictions might you want to apply to the data?
* Identify Business Functions
* List down all the queries that you need for queries/stored procedures etc., as per your need.

### Coding

* You need to make the 5 pages mentioned in Functional requirements section.
* You need to build all services required to handle the functionality and error scenarios.
* Write search, add, edit, delete Stored Procedures for each of the business function.

### Testing

* Set-up Test Data (manual or script or bulk upload thru excel)
* Perform full end to end testing with different scenarios.
* We will do Acceptance Testing based on this document.
* Discuss among yourselves on the requirements and prepare manual Test cases.

## Solution submission

1. Project Code Folder (entire .Net code + SQL Code)
2. Functional Requirements Document
3. ER Diagram