

Job Portal System - Documentation

User Types:

1. Job Seeker
2. Employer
3. Admin

Core Features:

*For Job Seeker:

- Register/Login + related jobs
- Profile update
- Upload CV +Categories
- Browse & filter jobs
- Apply to jobs
- Save favorite jobs
- Track application status +search job applications

*For Employer:

- ~~Company account~~
- Post/Edit/Delete jobs +change password
- Manage applications
- Accept/Reject candidates
- Send messages to applicants +send email

For Admin:

- Manage users
- Control job postings
- Handle categories
- Delete/Suspend accounts

System Flow:

1. User registers as Job Seeker or Employer.
2. Job Seeker views and applies to jobs.
3. Employer posts and manages jobs, reviews applicants.
4. Admin oversees the entire platform, moderates users and content.

MVC Architecture:

Model:

- User, Job, Application, Category, Company

Controller:

- AuthController
- JobController
- ApplicationController
- ProfileController
- AdminController

View:

- Login/Register Page
- User Dashboard
- Job Listings/Details
- Application Submission
- Admin & Employer Dashboards

UML Class Diagram

Main Entites

1. User

UserID (PK)

FullName

Email

Password

Role (Seeker / Employer / Admin)

CreatedAt

2. Job

JobID (PK)

Title

Description

Location

Salary

CategoryID (FK)

CompanyID (FK)

PostedAt (time of this post)

3. Application

ApplicationID (PK)

JobID (FK)

UserID (FK - Seeker)

CVFilePath

Status (Pending / Accepted / Rejected)

AppliedAt (thime of seeker that apply on the job)

4. Category

CategoryID (PK)

Name

5. Company

CompanyID (PK)

Name

Description

Logo

6. Message

MessageID (PK)

SenderID (FK - User)

ReceiverID (FK - User)

Content (of messg)

SentAt (time that messg send)

(Relationships)

User ↔ Application:

One-to-Many باحث عن عمل يقدم على أكثر من وظيفة → علاقة

Job ↔ Application:

One-to-Many كل وظيفة يمكن أن تُقدم عليها طلبات متعددة → علاقة

Category ↔ Job:

One-to-Many كل وظيفة تتبع تصنيف معين → علاقة

Company ↔ Job:

One-to-Many كل شركة تنشر وظائف متعددة → علاقة

User ↔ Message:

Self One-to-Many المستخدم يرسل ويستقبل رسائل → علاقة

