# Employment Portal Design Document

## Database Purpose:

The purpose of the employment portal database is to maintain the records of candidates and employers in order to provide essential information to fulfil their requirements.

## Business Problems Addressed:

- Allows employment portal staff to generate descriptive reports.
- To permit recruiters to post their job vacancies and candidates to apply for the same.
- Enable recruiters to view and shortlist candidates to schedule interviews based on their profiles.
- Authorize the portal staff to fetch and analyze reports of user account plans (Free, Standard, Premium).
- To generate a report for number of recruiters posting job vacancies and number of candidates applying for the same.

## Business Rules:

- Each user should either be a candidate or a recruiter.
- Candidate can have zero or more qualification.
- Candidate can have zero or more skillset.
- Each skillset can have zero or more certifications.
- Each company has at least one recruiter.
- Each recruiter posts zero or many job vacancies.
- Each candidate can apply to zero or multiple job vacancies.
- Each user will have any one account plan type at a time.
- Each user will make payments according to the account plan selected.
- Each portal staff manages one or many users (Recruiter or Candidate).
- Each candidate must add one and only one resume.

**Design Decisions:**

| Entity Name | Why Entity Included | How Entity is Related to Other Entities |
|---|---|---|
| Users | One of the key entities of the database is the user entity. It is a superclass with two user type subclasses – Candidate and CompanyRecruiter. This entity keeps track of all the user details including login and personal details. Each user can add only a single address. The last login date will help keep an update of how long the user has been inactive. | The user entity as a superclass is related to Candidate and CompanyRecruiter subtypes. These subtypes inherit the common user attributes along with their own. The user entity is also related to the user payment. Users can make multiple payments and have access to various features of the portal based on the account plan type chosen. It is also related to the account plan entity to keep track of the plan selected by the user. Each user will be managed by a portal staff. |
| UserPayment | UserPayment entity is used to keep a record of all the payments made by each user with attributes like payment amount and date. The payment validity date is a derived attribute calculated using the payment date and the account plan duration. | The UserPayment entity is related to the User entity and also the AccountPlan entity. At the end of the validity period or with an upgrade in plan type a new payment has to be made by the user. |
| AccountPlan | This includes a list of plan types that user can choose from along with its price attribute. Each selected plan will grant a specific access to various features offered by the portal. A free user will have only limited access with no payment required. | The AccountPlan entity is related to the user payment entity to keep record of the account types for which the user has made respective payments. It is also related to the User entity to keep an update of the current plan of the user. |
| Candidate | Candidate entity is one of the subclasses of the superclass entity Users. One of the primary purposes of this entity is to save additional candidate information. | As one of the key entities, the PK candidate_user_id relates it to different entities: Resume, Qualification, SkillSets, JobVacancies and Certification. |
| Candidate_has_Qualifications | Candidate_has_Qualifications is an associative entity that maintains the information of multiple qualifications of a candidate. | |

| | | |
|---|---|---|
| **Qualifications** | Qualifications entity is used to keep records of the various qualifications that the candidates can possess. | The Qualifications entity is related to Candidate entity via associative entity Candidate_has_Qualifications as there is a many to many relationship. A candidate can have multiple qualifications and similarly there could be many candidates for a qualification. |
| **University** | This entity is included to keep track of all the universities which could belong to multiple candidates and to avoid repeating university names. | It is related to the associative entity derived from Candidate and Qualifications. |
| **Company** | The Company entity contains the list of companies enrolled with the portal. This entity maintains only the headquarter location of the company. | This entity is related to the CompanyRecruiter entity where each company can have multiple recruiters. |
| **CompanyRecruiter** | CompanyRecruiter entity is one of the subclasses of the superclass entity Users. This entity maintains the details of the recruiters of their respective companies. | This key entity is related to JobVacancies, Users and Company. The recruiter belongs to a certain company for which he can create or update a job vacancy. |
| **SkillSet** | Skillset is an entity which consists of skills that a candidate can possess. This will help the recruiter to shortlist candidates based on their skills. | SkillSet is related to candidate via its associative entity due to many-to-many relationship. |
| **Candidate_has_SkillSet** | This is an associative entity which consists of relation between candidates and skills. It contains details of skills possessed by a particular candidate. | |
| **Certifications** | Certifications entity has the list of certifications related to the skills. There can be multiple certifications for the same Skillset. | It is connected to the associative entity Candidate_Skill_Certification which contains the information about certifications for a skill of a particular candidate. |
| **Candidate_Skill_Certification** | This entity is an associative entity which maintains certificate details and the skillset which the certificate is related to. Certification number is unique to each certificate assigned to individual candidate. | |

| | | |
|---|---|---|
| **JobVacancies** | This entity consists of the jobs posted by the company recruiter. Job_is_active attribute maintains the status of the job posted. | The JobVacancies entity is related to the Candidate entity through an associative entity due to the many-to-many relationship. It is also connected to the CompanyRecruiter entity. |
| **Candidate_applies_JobVacancies** | This entity is an associative entity between Candidate and JobVacancies due to their many-to-many relationship. | |
| **InterviewSchedule** | This entity keeps a record of the interview scheduled for a job applied by the candidate. | This entity is related to the Candidate_applies_JobVacancies and it will populate a tuple only if a candidate is selected for the Interview. |
| **PortalStaff** | Portal Staff entity consists details of the Job portal staff. The staff will have varied access to the database to perform different administrative tasks. | PortalStaff is related to the Users entity. |
| **Resume** | This entity stores the path to the candidate's resume stored on the portal server. Each candidate has to upload a resume. | This entity is related to the Candidate entity. |