

MASTER MATHÉMATIQUES ET INFORMATIQUE

APPLIQUEES AUX SCIENCES HUMAINES ET SOCIALES

PARCOURS DOUBLE COMPÉTENCE : INFORMATIQUE ET SCIENCES SOCIALES

PROJET – BOITE D’INTERIME

Prof : Pierre BLARRE

2021-2022

Anis KADRI & Martina DODEVSKA

**Partie 1 - Conception (MCD)**

**Les propositions**

**Après une lecture de l’énoncé on peut en tirer les informations suivantes :**

1. **Les salariés**

* Ont des rôles (Directeur, manager, secrétaire, accompagnateur, etc.)
* Publient des offres d’emploi proposés par des entreprises

On définit un type d’entité salarié

**Cardinalités**

* 0 ... N Offre d’emploi

1. **Les entreprises**

* Envoie des offres d’emplois qui lui seront rattachées

On définit un type entreprise

* 0.. N Offre d’emploi

1. **Offre** **d’emploi**

* Intitulé
* Type : stage ou emploi ou ….
* Salaire brut mensuel par taux horaire
* Type de contrat
* Date de début et date de fin (pas un cdi)
* Statut : ouvert ou fermé

Note : rattachés à une entreprise a des métiers ou à des compétences

On définit un type emploi/stage

**Cardinalités**

* 0.. N candidats

1. **Les** **candidats**

* Une **description**
* Des **diplômes** et niveaux d’études
* Des **expériences** professionnelles : (CDI, CDD,…, + date de début et date de fin)
* Des **métiers** (: Développeur Web, Analyste Data, Charpentier, Soudeur, Professeur, etc.)
* Des **compétences** (Logiciels bureautiques, Métallurgie, Java, MySQL, Formation, Dessin Technique, etc.)
* **État** : disponible ou indisponible

**Cardinalités**

* Postule pour 0.. N offres

1. **Personne**

* Statut : salarié, candidat
* Nom
* Prénom
* Date de naissance
* Adresse
* Numéro de téléphone
* Adresse
* Adresse électronique

On définit un type d’entité Personne

**Cardinalités**

* 1.. N candidats / Employés

**Les dépendances fonctionnelles**

**Table : Candidat**

|  |
| --- |
| CANDIDAT |
| Id\_Candidat Nom  Prénom  Adresse  Date de naissance  Num de téléphone  Adresse mail  Diplôme  Niveau d’étude  Expérience  Métiers  Compétences  Statut |

**Proposition :**

**1FN :** Tous les attributs de la table ont des valeurs atomiques.

**2FN** : Tous les attributs dépendent totalement de la **clé «Id\_Candidat »**.

**3FN** : afin de s’assurer que les attribue de la table ne sont pas composé, atomique Pour être en 3NF il faut **décomposer** cette table :

|  |
| --- |
| CANDIDAT |
| Id\_Candidat Nom  Prénom  Adresse  Date de naissance  Num de téléphone  Adresse mail  Diplôme  Niveau d’étude  Statut |

|  |
| --- |
| METIERS |
| Développeur Web, Analyste Data  Charpentier  Codeur  Professeur id\_candidat |

|  |
| --- |
| EXPERIENCES |
| type  date début  date fin  id\_candidat |

|  |
| --- |
| COMPETENCES |
| Logiciels bureautiques Métallurgie MySQL  Formation id\_candidat |

**TABLE : OFFRE EMPLOI**

|  |
| --- |
| OFFRE EMPLOI |
| Id\_Offre  Intitulé  Salaire brut  Date de debut  Date de fin  Type de contrat  Taux horaire  Nombre heures  Id\_Candidat  Id\_Salarié  Id\_entreprise |

**1FN :** Tous les attributs de la table ont des valeurs atomiques.

**2FN :**

Id\_Offre → Intitulé  
Id\_Offre → Id\_entreprise  
Id\_Offre → Id\_Salarié

**3FN :** La troisième forme normale existe déjà vu qu’on a des attributs de transitions qui sont places dans des autres tables (id\_candidat, id\_salarie, id\_entreprise)

**TABLE : SALARIE**

|  |
| --- |
| SALARIE |
| Id\_SALARIE  Role  Salaire  ~~Commision~~ |

**2NF** : On a des attributs qui dépendent totalement de **la clé.**   
**3NF :**

Id\_SALARIE→Role

Id\_SALARIE→ Salaire

**Justification des choix**

La composition de notre schéma est inspirée de l’énoncé du sujet. Le but c’était de représenter le projet de manière simple et compréhensible. On a choisi de crée des tables pour les classes qui nous semblent indispensable pour la réalisation de l’étude qui répondre le mieux à l’énoncé de l’entreprise. Les cibles du projet de l’entreprise concerné, sont les candidats qui postulent pour une offre d’emploi.

En partageant les mêmes attributs, les « Salariés » avec les « Candidats », pour factoriser les informations, on a créé une table associative nommée « Personne ». De plus, le choix de crée une table « Compétences » et basée sur le fait que c’est un des points indispensables qu’un candidat doit posséder pour réussir sa candidature.

De l’autre côté, cela permettra à l’entreprise de faire une sélection des candidats de manière plus subtile. Enfin les relations entre nos tables permettent de minimiser les données redondantes que les tables peuvent contenir.

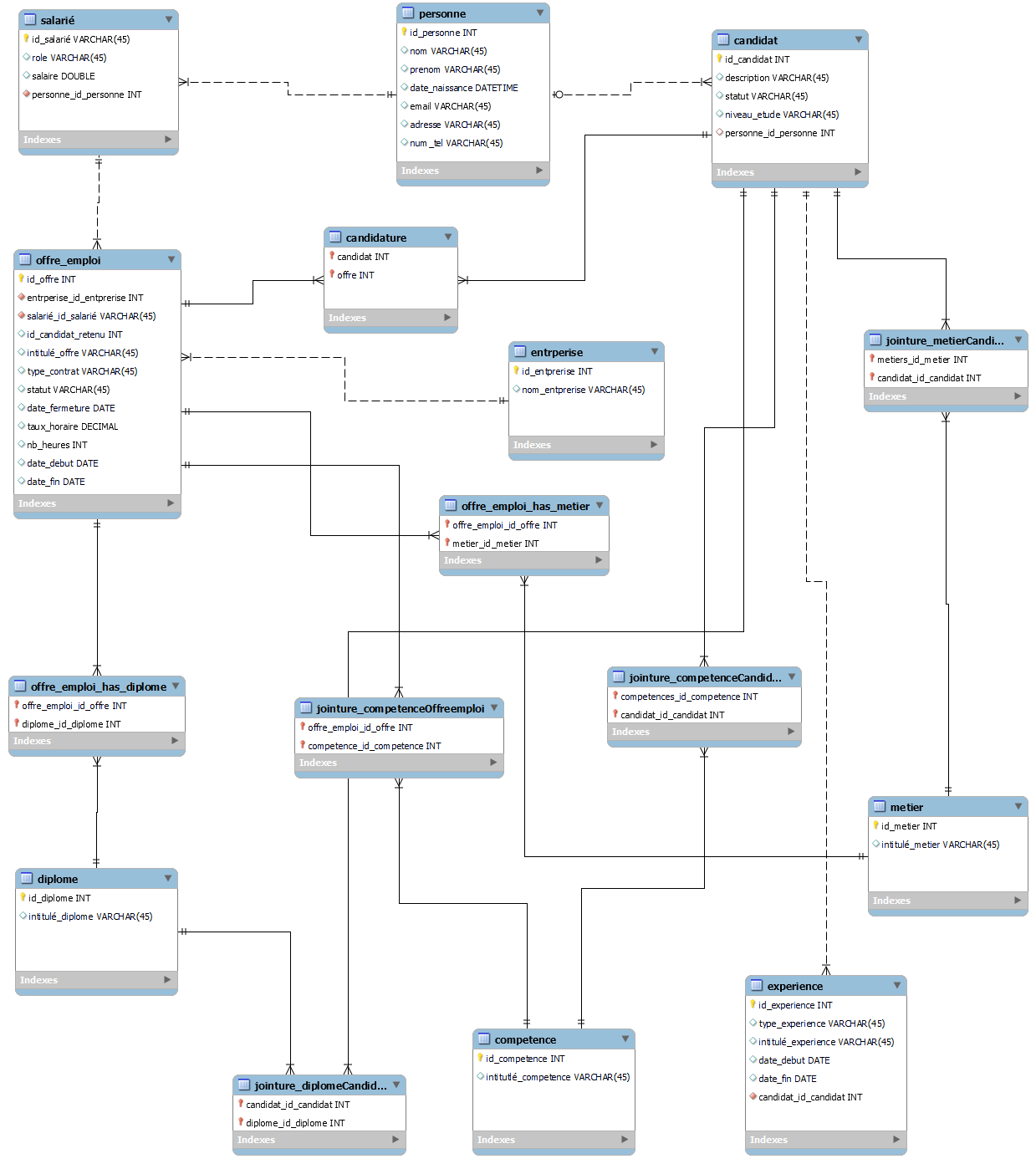
1. **Une impression d’écran du schéma**

Une image contenant texte, portable, écran, capture d’écran

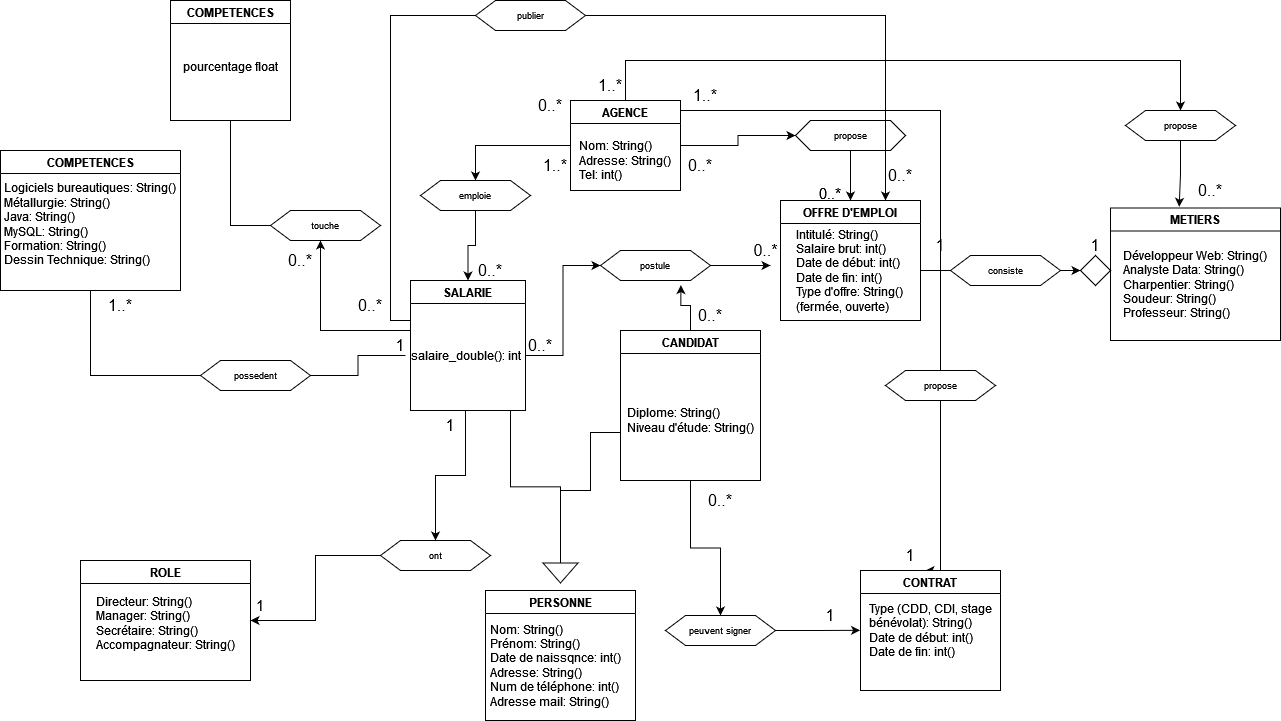
Description générée automatiquement

**Partie 2 - Production (MLD) - 10pts)**

1. **Une impression d’écran du votre schéma de table**



1. **Impression d’écran du diagramme de classe**

****

**Partie 3 - Requêtes SQL - 10pts**

**--1-Une liste des candidats et des offres auxquelles ils ont postulées.**

use mydb;

select p.nom, p.prenom, o.intitulé\_offre

from personne p , candidat c, offre\_emploi o, candidature d

-- jointure 01

where p.id\_personne = c.personne\_id\_personne

-- jointure 02

and c.id\_candidat = d.candidat

-- jointure 03

and o.id\_offre = d.offre

order by p.nom

**-- 2- Proposez une ou plusieurs requêtes pour afficher toutes les informations d’un candidat (Informations personnelles, diplômes, métiers, compétences, offres d’emploi, etc.)**

-- infos personnelles

(select \* from personne where id\_personne=1)

-- diplome

select intitulé\_diplome

from jointure\_diplomecandidat j, candidat c, diplome d

where j.candidat\_id\_candidat = c.id\_candidat and j.diplome\_id\_diplome = id\_diplome

and c.id\_candidat=1

-- metier

select intitulé\_metier

from jointure\_metiercandidat j, candidat c, metier m

where j.candidat\_id\_candidat = c.id\_candidat and j.metiers\_id\_metier = id\_metier

and c.id\_candidat=1

-- competence

select intitutlé\_competence

from jointure\_competencecandidat j, candidat c, competence p

where j.candidat\_id\_candidat = c.id\_candidat and j.competences\_id\_competence = id\_competence

and c.id\_candidat=1

--2- offre d'emploi

select intitulé\_offre

from offre\_emploi o, candidat c, candidature d

where d.candidat = c.id\_candidat and o.id\_offre = d.offre

and c.id\_candidat=1

**--3- Lister les candidats qui ont postulé à toutes les offres d’emploi**

SELECT d.candidat, count(d.offre) as total

FROM candidature d

GROUP BY candidat

having total = (select count(id\_offre) as total from offre\_emploi)

**--4-Calculer la somme des commissions perçues par l’agence (toutes périodes confondues)**

select sum(taux\_horaire\*nb\_heures\*0.1) as commission

from offre\_emploi

where statut ='FERMEE'

**--5- Même chose que la 4, mais cette fois çi groupées par mois**

select sum(taux\_horaire\*nb\_heures\*0.1) as commission\_agence ,MONTH(date\_fermeture) as mois

from offre\_emploi

where statut ='FERMEE'

group by MONTH(date\_fermeture)

**--6 Même chose que 4 et 5, mais cette fois çi calculer les commissions par salarié et non par l’agence**

-- pareil que le 4

select sum(taux\_horaire\*nb\_heures\*0.01), salarié\_id\_salarié

from offre\_emploi

where statut ='FERMEE'

group by salarié\_id\_salarié

-- pareil que le 5

select sum(taux\_horaire\*nb\_heures\*0.01), salarié\_id\_salarié,MONTH(date\_fermeture)

from offre\_emploi

where statut ='FERMEE'

group by MONTH(date\_fermeture)

order by salarié\_id\_salarié desc

**--7-Lister les compétences qui ne sont pas rattachées à des candidats**

use mydb;

select intitutlé\_competence

from competence

where id\_competence not in

(select competences\_id\_competence from jointure\_competencecandidat)

**--8- Lister les candidats potentiels (qui n’ont pas encore candidaté)**

**--pour une offre d’emploi donnée (c.à.d les candidats qui sont liés**

**--aux métiers ou compétences requises par l’offre d’emploi)**

CREATE VIEW Vue\_1

AS

SELECT id\_candidat

FROM candidat c

WHERE id\_candidat IN (

SELECT candidat\_id\_candidat

FROM jointure\_metiercandidat

WHERE metiers\_id\_metier IN (SELECT metier\_id\_metier FROM offre\_emploi\_has\_metier WHERE offre\_emploi\_id\_offre = 4))

OR id\_candidat IN (

SELECT candidat\_id\_candidat

FROM jointure\_competencecandidat

WHERE competences\_id\_competence IN (SELECT competence\_id\_competence FROM jointure\_competenceoffreemploi jc WHERE offre\_emploi\_id\_offre = 4))

AND c.id\_candidat NOT IN (SELECT candidat FROM Candidature d WHERE d.offre = 4);

**--9-Ajouter les candidatures pour les candidats listé en requête 8 (avec une ou plusieurs requêtes**

INSERT INTO candidature (candidat, offre)

SELECT candidat, 4

FROM Vue\_1;

**--10-Fermer une offre d’emploi et l’attribuer à un candidat**

UPDATE offre\_emploi

SET id\_candidat\_retenu = (SELECT candidat FROM candidature WHERE offre = 4 ORDER BY RAND() limit 1),

Statut= "FERMEE"

WHERE ID\_OFFRE = 4;