~~2 restaurants et~~ un parking de 1000 places sont accessibles à quelques dizaines de mètres. ~~L’hiver, la salle N°2 (la plus grande) est convertie en centre d’hébergement pouvant accueillir au maximum 100 personnes~~. Le parking est divisé en 2 parties séparées par les 2 restaurants, la partie la plus proche des salles est clôturée et sécurisée. Sa capacité est de 250 places et seules les personnes possédant un billet d’entrée à un évènement peuvent y accéder. A l’entrée, le conducteur scanne son billet d’entrée et paye 3€ (tarif unique). La seconde partie du parking n’est pas surveillée et son utilisation est gratuite. La surveillance est assurée 24h/24 par 1 gardien la journée et 2 gardiens la nuit. Durant son service, chaque gardien doit effectuer plusieurs rondes et passer par 4 points de contrôle (parking, salle 1, salle 2, terrain extérieur) où il scannera son badge pour prouver son passage. En cas d’incident, un rapport est rédigé sur une main courante, le rapport indiquera si les secours ont dû être appelés. ~~Le poste de sécurité est situé dans la salle N°2.~~

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nom | Signification | Type | Longueur | Contraintes |
| Parking\_id | Numéro du parking | N | 11 | Identifiant, auto incrémenté |
| Parking\_price | Prix de la place de parking | N | 6,4 | >= 0, facultatif |
| Parking\_places | Nombre de places | N | 4 | Strictement positif, obligatoire |
| Passage\_registration | Plaque d’immatriculation | AN | 11 | Obligatoire, format plaque immatriculation, identifiant |
| Passage\_in | Heure d’entrée dans le parking | DT |  | Format YYYY-MM-DD HH:ii:ss, identifiant |
| Passage\_out | Heure de sortie du parking | DT |  | Format YYYY-MM-DD HH:ii:ss ; strictement supérieur passage\_in |
| Checkpoint\_name | Nom du point de passage | AN | 50 | Identifiant, pas de caractères spéciaux, accents autorisés |
| Checkpoint\_passage | Date de passage du gardien | DT |  | Format YYYY-MM-DD HH:ii:ss |
| Incident\_number |  | N | 11 | Identifiant, auto incrémenté |
| Incident\_label |  | AN | 100 | Obligatoire |
| Incident\_date |  | DT |  | Format YYYY-MM-DD HH:ii:ss |
| Incident\_rescue |  | B | 1 | Obligatoire |
| Incident\_rescue\_hour |  | DT |  | Format YYYY-MM-DD HH:ii:ss |
| Incident\_details |  | AN | 65K | Obligatoire |
| Incident\_resolution |  | AN | 65K | Facultatif |
| Type\_name |  | AN | 100 | Identifiant |

## Règles de gestion

Un gardien peut effectuer 0 ou plusieurs rondes  
Une ronde est effectuée par 1 seul gardien

Une ronde concerne 1 ou plusieurs checkpoints  
Un checkpoint est associé à 0 ou plusieurs rondes

Un passage de voiture se fait dans 1 seul parking  
Un parking peut voir 0 ou plusieurs passages de voitures

Un gardien peut renseigner 0 ou plusieurs incidents  
Un incident est renseigné par un seul gardien

Un incident peut être lié à 1 ou plusieurs types d’incident  
Un type d’incident peut être lié à 0 ou plusieurs incidents

Un incident se déroule dans 1 seul lieu (checkpoint)  
Un lieu (checkpoint) peut accueillir 0 ou plusieurs incidents

## Dépendances fonctionnelles simples

Parking\_id 🡪 parking\_price, parking\_places

Checkpoint\_name 🡪

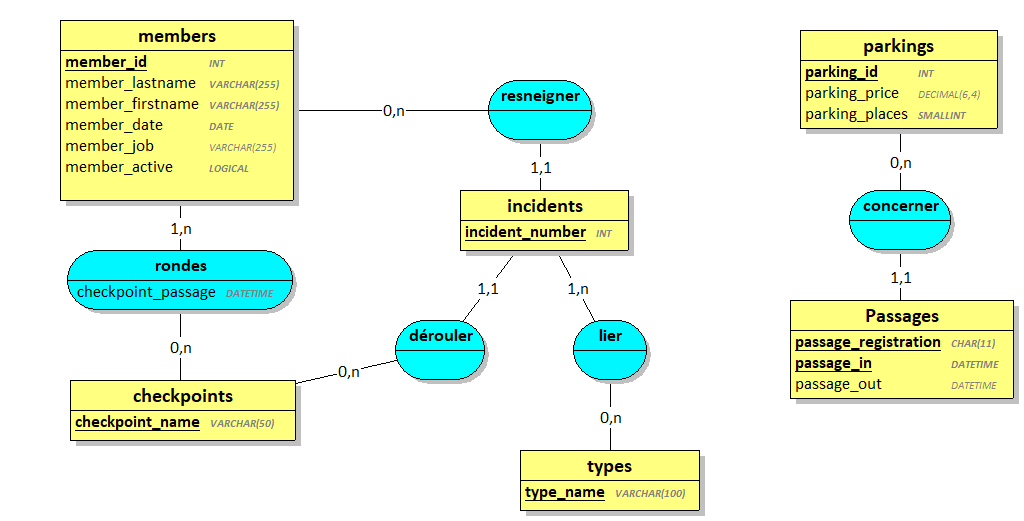
Incident\_number 🡪 incident\_label, incident\_date, incident\_rescue, incident\_rescue\_hour, incident\_details, incident\_resolution, member\_id

Incident\_type\_name 🡪

## Dépendances fonctionnelles composées

Passage\_registration, Passage\_in 🡪 passage\_out

## Modèle conceptuel des données



## Modèles Logique des données

Parkings ( **parking\_id**, parking\_price, **parking\_places** )

Passages ( **passage\_registration**, **passage\_in**, passage\_out, **#parking\_id** )

Members ( **member\_id**, **member\_lastnid, ame, member\_firstname, member\_date,** member\_job**, member\_active** )

Checkpoints ( **checkpoint\_name** )

Rondes ( **#member\_id**, **#checkpoint\_name**, **checkpoint\_passage**)

Types ( **type\_name** )

Incidents ( **incident\_number**, **incident\_label, incident\_date, incident\_rescue\_, incident\_rescure\_hour, incident\_details**, incident\_resolution, **#member\_id, #checkpoint\_name** )

Incidents\_types ( **#incident\_number, #type\_name** )