|  |
| --- |
| Henallux -IESN |
| Programmation Orienté Objet Avancé |
| Analyse |
| **20/08/2018** |
|  |
|  |

**FRICOT Damien**

Table des matières

[Domaine d’application 2](#_Toc522471824)

[Base de données 3](#_Toc522471825)

[Documentation 3](#_Toc522471826)

[Schéma conceptuel 9](#_Toc522471827)

[Fonctionnalités 10](#_Toc522471828)

[L’insertion 11](#_Toc522471829)

[Modification 11](#_Toc522471830)

[Suppression 12](#_Toc522471831)

[Listing de tables 12](#_Toc522471832)

[Recherches 13](#_Toc522471833)

[Tâche métier 15](#_Toc522471834)

Domaine d’application

Le domaine d’application concerne une société namuroise de location de vélos urbains « Libiavélo ». A travers plusieurs villes de l’entité, on retrouve plusieurs stations, dans chacune des vélos sont mis à la disposition du public. Le principe reste simple : un nouvel usager se présente à une station, s’identifie grâce à son abonnement, retire un vélo de la station et une fois son usage terminé dépose le vélo dans n’importe laquelle des stations.

A intervalle régulier des employés de la société (les préposés) veillent à ce que dans chaque station on dispose d’un nombre de vélo correct. Ni trop de vélo pour permettre la restitution des derniers en fin d’utilisation, ni trop peu pour permettre la location. En cas de besoin les vélos sont dispatchés d’une station à l’autre pour équilibrer le nombre à chaque station à l’aide d’ordres de transport remplit par le préposé de la station.

Quand un préposé remarque une défaillance à un vélo, il crée un ordre de réparation remplit avec ses observations. C’est le chef d’atelier qui va compléter et confirmer l’ordre de réparation. Par la même occasion, un ordre de transport pour acheminer le vélo à l’atelier est créé.

Le programme java se concentrera sur le développement d’un menu organisant l’ajout, la suppression et la modification d’un employé. D’une recherche concernant les livreurs de vélo entre stations. Ainsi que la tâche métier d’ordre de réparation décrite ci-dessus.

Base de données

Documentation

|  |  |  |  |
| --- | --- | --- | --- |
| STATION | TYPES | CONDITIONS | INFORMATIONS |
| id | **Integer** | **Primary key** | **Numéro identifiant de chaque station** |
| label | **Varchar** | **Not null** | **Libellé de chaque station** |
| nbBikeMinWarn | **Integer** | **Not null** | **Nombre minimale de vélo requis pour la limite de warning par station** |
| nbBikeMinContr | **Integer** | **Not null** | **Nombre minimale de vélo requis pour la limite de contrôle par station** |
| nbBikeMaxWarn | **Integer** | **Not null** | **Nombre maximal de vélo requis pour la limite de warning par station** |
| nbBikeMaxContr | **Integer** | **Not null** | **Nombre maximale de vélo requis pour la limite contrôle par station** |
| dateCreation | **Date** | **Not null** | **Date de création de chaque station** |
| isCover | **Boolean** | **Not null** | **Savoir si une station est couverte d’un toit ou non** |
| coordGPS | **Varchar** | **[0-1]** | **Coordonnée géographique d’une station** |
| street | **Varchar** | **Not null** | **Nom de la rue ou se situe chaque station** |
| Id\_Zone | **Integer** | **Foreign Key (Zone), Not null** | **Numéro identifiant de chaque zone où se situent les stations** |
| Id\_locality | **Integer** | **Foreign Key (Locality), Not null** | **Numéro identifiant de chaque localité** |
| ID : id | | | |
| REF : LOCALITY.id | | | |
| CHECK : (nbBikeMinWarn > nbBikeMinContr), (nbBikeMaxWarn < nbBikeMaxContr) | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| LOCALITY | TYPES | CONDITIONS | INFORMATIONS |
| id | **Integer** | **Primary key** | **Numéro identifiant de chaque localité** |
| label | **Varchar** | **Unique, Not null** | **Libellé de chaque localité** |
| postalCode | **Integer** | **Not null** | **Code Postal de chaque localité** |
| ID : id | | | |
| UNIQUE : (name,postalCode) | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| BIKE | TYPES | CONDITIONS | INFORMATIONS |
| id | **Integer** | **Primary Key** | **Numéro identifiant de chaque vélo** |
| isDamaged | **Boolean** | **Not null** | **Savoir si un vélo est considéré comme endommager ou non** |
| datePurchase | **Date** | **Not null** | **Date d’achat de chaque vélo** |
| id\_Station | **Integer** | **Foreign Key (Station), [0-1]** | **Numéro identifiant de la station ou se situe un vélo** |
| id\_Entreprise | **Integer** | **Foreign Key (Entreprise), Not null** | **Numéro identifiant de l’entreprise qui a fourni un vélo** |
| ID : id | | | |
| REF : STATION.id  ENTREPRISE.id | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| ENTREPRISE | TYPES | CONDITIONS | INFORMATIONS |
| id | **Integer** | **Primary Key** | **Numéro identifiant de chaque entreprise** |
| label | **Varchar** | **Unique** | **Nom de l’entreprise** |
| location | **Varchar** | **Not null** | **Location de l’entreprise** |
| ID : id | | | |
| UNIQUE : name | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| REPARATION\_RECORD | TYPES | CONDITIONS | INFORMATIONS |
| id | **Integer** | **Primary Key** | **Numéro identifiant de chaque fiche de réparation** |
| isValidate | **Boolean** | **Not null** | **Indique si un ordre de réparation est validé ou non** |
| dateBegin | **Date** | **Not null** | **Date de début de la réparation d’un vélo** |
| dateEnd | **Date** | **[0-1]** | **Date de fin de la réparation d’un vélo** |
| note | **Varchar** | **[0-1]** | **Remarque optionnelle en fin de réparation** |
| workOrder | **VarChar** | **Not null** | **Ordre de travail pour chaque réparation établit au préalable** |
| isDownGraded | **Boolean** | **Not null** | **Indique si un vélo est considéré comme déclassé** |
| id\_Bike | **Integer** | **Foreign Key (Bike), Not null** | **Numéro identifiant de chaque vélo** |
| Id\_Workshop | **Integer** | **Foreign Key (Workshop), [0-1]** | **Numéro identifiant de chaque atelier** |
| ID : id | | | |
| REF : BIKE.id  WORKSHOP.id | | | |
| CHECK : (dateBegin < dateEnd) | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| HISTORICAL | TYPES | CONDITIONS | INFORMATIONS |
| dateArrival | **Date** | **Not null** | **Date d’arrivée d’un vélo dans une station** |
| dateLeaving | **Date** | **Not null** | **Date de départ d’une station pour un vélo** |
| id\_Station | **Integer** | **Foreign Key (Station), Not null** | **Numéro identifiant de chaque station** |
| id\_Bike | **Integer** | **Foreign Key (Bike), Not null** | **Numéro identifiant de chaque vélo** |
| ID : BIKE.id, STATION.id, dateArrival | | | |
| REF : BIKE.id  STATION.id | | | |
| CHECK : (dateArrival < dateLeaving) | | | |

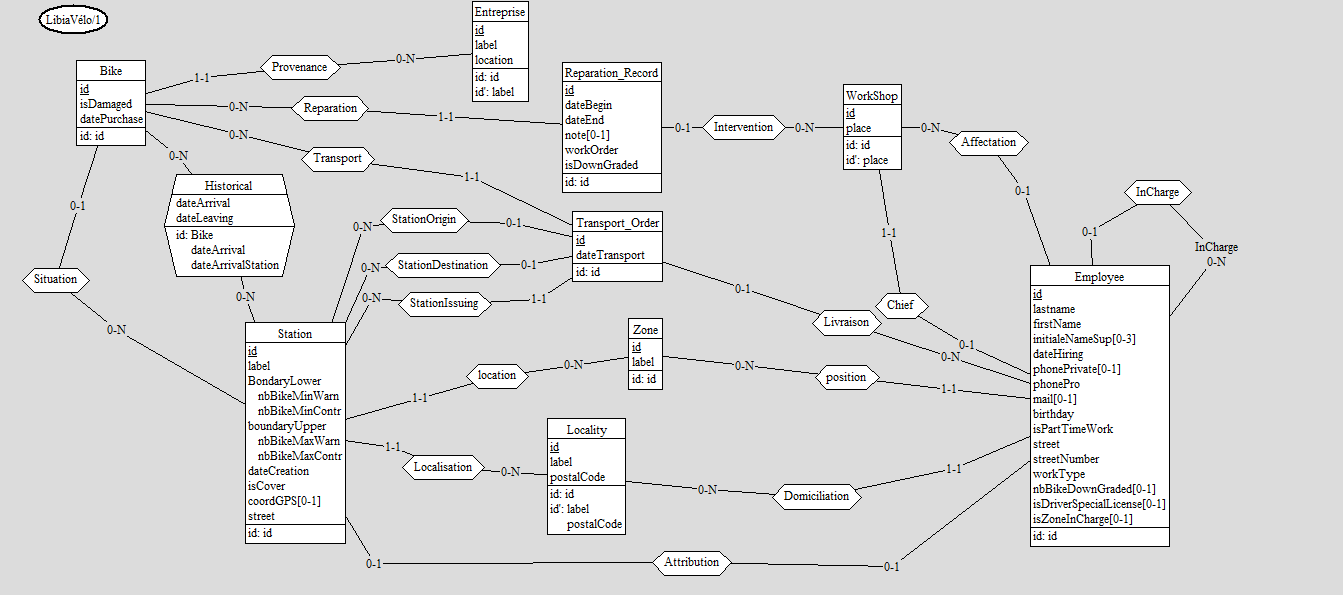
|  |  |  |  |
| --- | --- | --- | --- |
| EMPLOYEE | TYPES | CONDITIONS | INFORMATIONS |
| id | **Integer** | **Primary Key** | **Numéro identifiant de chaque employé** |
| lastName | **Varchar** | **Not null** | **Nom de chaque employé** |
| firstName | **Varchar** | **Not null** | **Prénom de chaque employé** |
| initialNameSupp | **Char** | **[0-3]** | **Initial des prénoms supplémentaires des employés** |
| dateHiring | **Date** | **Not null** | **Date d’embauche de chaque employé** |
| phonePrivate | **Integer** | **[0-1]** | **Numéro de téléphone privé des employés** |
| phonePro | **Integer** | **Not null** | **Numéro de téléphone professionnel de chaque employé** |
| mail | **Varchar** | **[0-1]** | **Adresse email privé des employés** |
| birthday | **Date** | **Not null** | **Date de naissance de chaque employé** |
| isPartTimeWork | **Boolean** | **Not null** | **Indique si un employé travail à temps partiel** |
| street | **Varchar** | **Not null** | **Libellé de la rue du domicile de chaque employé** |
| streetNumber | **Integer** | **Not null** | **Numéro de la rue du domicile de chaque employé** |
| workType | **Varchar** | **Not null** | **Type d’emploi de chaque employé (Livreur, Préposé, Administratif, Réparateur)** |
| isDriverSpecialLicense | **Boolean** | **[0-1]** | **Indique si les employés de type « livreur » dispose d’un permis plateau** |
| isLeader | **Boolean** | **[0-1]** | **Indique si les employés sont responsables (d’une zone pour les préposé ou d’un atelier pour les réparateur)** |
| id\_Workshop | **Integer** | **Foreign Key (Workshop), [0-1]** | **Numéro identifiant de l’atelier attribué** |
| Id\_Station | **Integer** | **Foreign Key (Station), [0-1]** | **Numéro identifiant de la station attribuée** |
| Id\_Locality | **Integer** | **Foreign Key (Locality), Not null** | **Numéro identifiant de chaque localité** |
| Id\_Zone | **Integer** | **Foreign Key (Zone), [0-1]** | **Numéro identifiant de chaque zone où se situent les stations** |
| inCharge\_Employee | **Integer** | **Foreign Key (Employee), [0-1]** | **Référence à l’identifiant du responsable de l’employé** |
| ID : id | | | |
| REF : WORKSHOP.id  EMPLOYEE.id  LOCALITY.id | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| TRANSPORT\_ORDER | TYPES | CONDITIONS | INFORMATIONS |
| id | **Integer** | **Primary Key** | **Numéro identifiant de chaque ordre de transport** |
| dateTransport | **Date** | **Not Null** | **Date prévue pour le transport** |
| id\_Bike | **Integer** | **Foreign Key (Bike), Not null** | **Numéro identifiant de chaque vélo** |
| id\_Employee | **Integer** | **Foreign Key (Employee), Not null** | **Numéro identifiant de chaque transporteur** |
| id\_Station\_issuance | **Integer** | **Foreign Key (Station), Not null** | **Numéro identifiant de la station émettrice de l’ordre de transport** |
| id\_Station\_origin | **Integer** | **Foreign Key (Station), [0-1]** | **Numéro identifiant de la station d’origine du vélo** |
| id\_Station\_destination | **Integer** | **Foreign Key (Station), [0-1]** | **Numéro identifiant de la station de destination du vélo** |
| Id\_WorkShop | **Integer** | **Foreign Key (WorkShop), [0-1]** | **Numéro identifiant de l’atelier ou sera envoyé le vélo le cas échéant** |
| ID : id | | | |
| REF : BIKE.id  STATION.id  EMPLOYEE.id  WORKSHOP.id | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| WORKSHOP | TYPES | CONDITIONS | INFORMATIONS |
| Id | **Integer** | **Primary Key** | **Numéro identifiant de chaque atelier** |
| place | **Varchar** | **Not null** | **Lieu ou est implanté l’atelier** |
| ID : id | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| ZONE | TYPES | CONDITIONS | INFORMATIONS |
| Id | **Integer** | **Primary Key** | **Numéro identifiant de chaque zone où se situent les stations** |
| label | **Varchar** | **Not null** | **Libellé de chaque zone** |
| ID : id | | | |

Schéma conceptuel

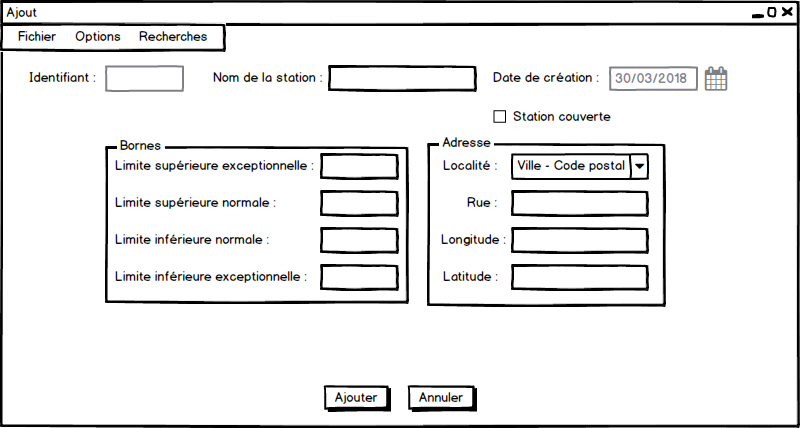


Fonctionnalités

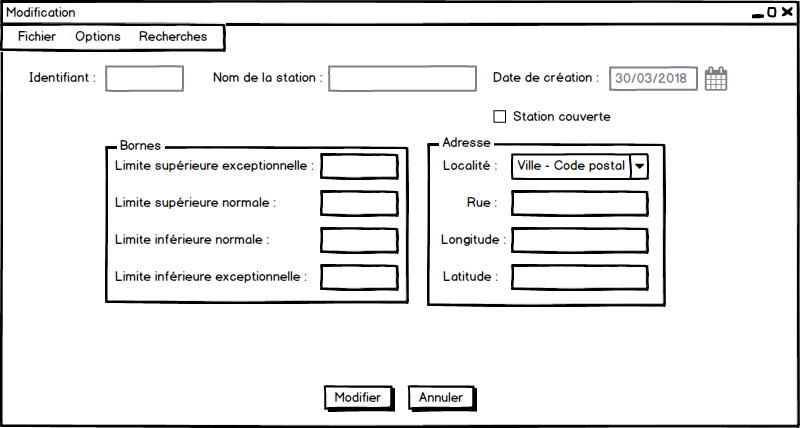
La table utilisée sera la table **« EMPLOYEE »**, dans laquelle on ajoute, modifie ou supprime un employé avec les caractéristiques suivante ;

|  |  |
| --- | --- |
| COLONNE | INFORMATIONS |
| id | Numéro identifiant de chaque employé |
| lastName | Nom de chaque employé |
| firstName | Prénom de chaque employé |
| initialNameSupp | Initial des prénoms supplémentaires des employés |
| dateHiring | Date d’embauche de chaque employé |
| phonePrivate | Numéro de téléphone privé des employés |
| phoneProf | Numéro de téléphone professionnel de chaque employé |
| mail | Adresse email privé des employés |
| birthday | Date de naissance de chaque employé |
| isPartTimeWork | Indique si un employé travail à temps partiel |
| street | Libellé de la rue du domicile de chaque employé |
| streetNumber | Numéro de la rue du domicile de chaque employé |
| workType | Type d’emploi de chaque employé (Livreur, Préposé, Chef Atelier, Réparateur, …) |
| driverSpecialLicense | Indique si les employés de type « livreur » dispose d’un permis plateau |
| isLeader | Indique si les employés sont responsables ou non |
| id\_Workshop | Numéro identifiant de l’atelier attribué |
| Id\_Station | Numéro identifiant de la station attribuée |
| Id\_Locality | Numéro identifiant de chaque localité |
| inCharge\_Employee | Référence à l’identifiant du responsable de l’employé |

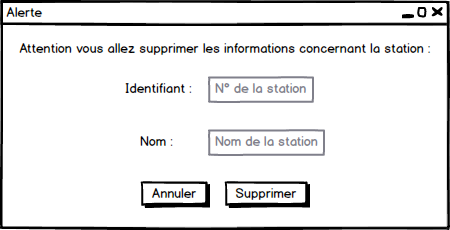
L’insertion



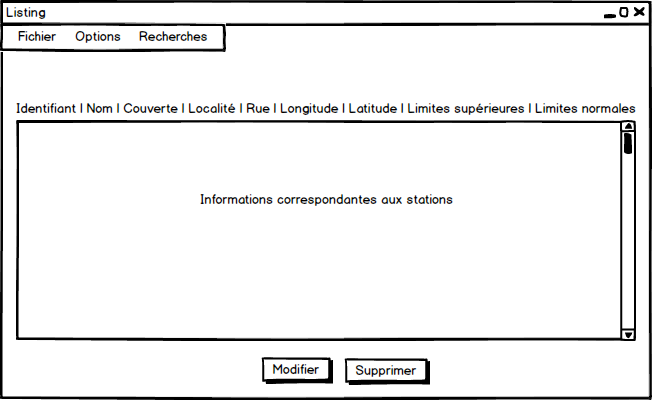
Modification



Suppression

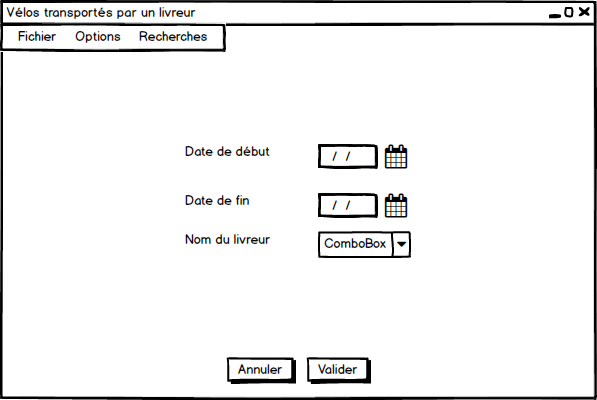


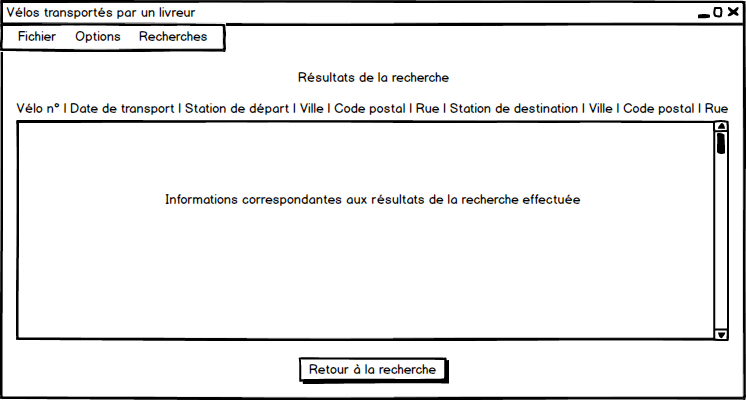
Listing de tables



Recherches

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Recherche numéro : 2 (Livraison) | | | | |
| Titre de la recherche : Vélo transporté par le livreur | | | | |
| Objectif de la recherche : Afficher les informations de station et des ordres de transport d’un livreur donné entre deux dates | | | | |
| Jointure entre les tables : | | | |  |
| Table 1 :  Employé | **Table 2 :**  Vélo | **Table 3 :**  Station | **Table 4 :**  OrdreTransport |  |
| Entrée : |  |  |  |  |
|  | **Critère** |  | **Composant** |  |
|  | dateDebut |  | Jspinner |  |
|  | dateFin |  | JSpinner |  |
|  | livreur |  | JComboBox : | nom + prenom |
|  |  |  |  |  |
| Sortie : |  |  |  |  |
|  | **Nom de la colonne** | Provenant de la table | **Nom de la table** |  |
|  | dateTrasnport |  | OrdreTransport |  |
|  | idBike |  | Velo |  |
| (StationEmettrice) | LabelStation |  | Station |  |
|  | labelStreet |  | Station |  |
|  | coordGPS |  | Station |  |
|  | labelLocality |  | Localite |  |
|  | postalCode |  | Localite |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |





Tâche métier

La tâche métier sera réalisée par le chef d’atelier. Elle consistera à compléter (si nécessaire) et confirmer un ordre de réparation préalablement émis par un préposé.

Sur base des observations décrite dans le document par le préposé, le chef d’atelier peut le compléter à l’aide d’indications provenant d’un listing de réparations types et de leurs caractéristiques. Par exemple si un pneu est crevé, il faut vérifier le pneu ainsi que la chambre a air et le cas échéant les remplacer.

Si l’ordre de réparation est confirmé, alors un ordre de transport pour acheminer le vélo à l’atelier est remplit par le chef d’atelier. Dans ce document il renseignera le numéro du vélo, la station émettrice de l’ordre de réparation et la destination désignée parmi la liste des ateliers disponibles. Pour terminer par être transmis au responsable logistique. Dans le cas où l’ordre de réparation est infirmé par le chef d’atelier, alors le document est abandonné et le processus s’arrête.

