

Documentation iteration2

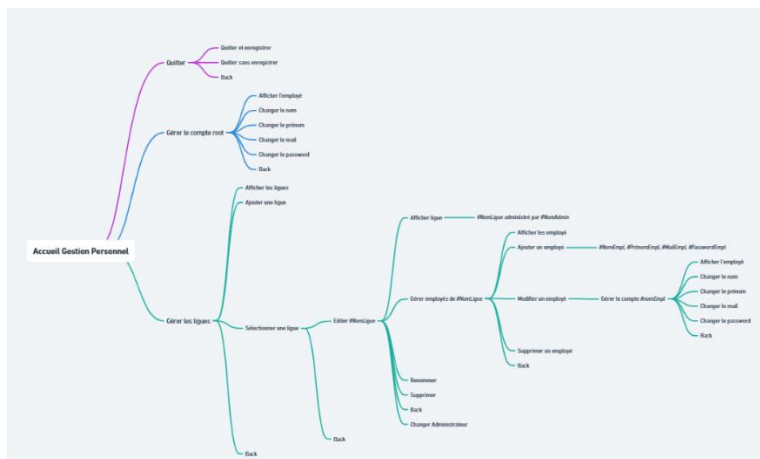
Création des tables et contraintes en SQL dans un fichier.SQL en rapport avec le MDC.

```
@@ -0,0 +1,7 @@
+ alter table UTILISATEUR
+ add constraint fk_idType
+ foreign key(idType) references TYPE_UTILISATEUR(idType);
+
+ alter table UTILISATEUR
+ add constraint fk_numLigue
+ foreign key(numLigue) references LIGUE(numLigue);
+
```

```
+ alter table UTILISATEUR
+ drop foreign key fk_idType;
+ alter table UTILISATEUR
+ drop foreign key fk_numLigue;
+
+
+ DROP TABLE IF EXISTS UTILISATEUR;
+ DROP TABLE IF EXISTS TYPE_UTILISATEUR;
+ DROP TABLE IF EXISTS LIGUE;
+
+ CREATE TABLE UTILISATEUR (
+   idUtilisateur INT PRIMARY KEY NOT NULL,
+   nomUtil VARCHAR(50),
+   prenomUtil VARCHAR(50),
+   mailUtil VARCHAR(250),
+   passwordUtil VARCHAR(50),
+   date_arrivee date,
+   date_depart date,
+   idType INT,
+   numLigue INT null
+ );
+
+ CREATE TABLE TYPE_UTILISATEUR (
+   idType INT PRIMARY KEY NOT NULL,
+   libelleType VARCHAR(50)
+ );
+
+ CREATE TABLE LIGUE (
+   numLigue INT PRIMARY KEY NOT NULL,
+   nomLigue VARCHAR(50)
+ );
```

Dans ligueConsole un menu pour sélectionner un employé avant de le modifier ou le supprimer grâce aux options dans le code (ligne 152).

L'arbre heuristique a été mis à jour, on peut désormais changer d'admin d'une ligue à partir de Edite nomligue.



Voici l'option permettant de changer l'admin tjrs dans ligueConsole.

```

160 + // option changer d'admin
160 161 private List<Employe> changerAdministrateur(final Ligue ligue) {
161 162     return new List<>("Changer d'administrateur", "c", () -> new ArrayList<>(ligue.getEmployes()),
162 -         employe -> ligue.setAdministrateur(employe));
163 - }
163 + (index, element) -> {ligue.setAdministrateur(element); system.out.println(element + " est le nouvel
164 + administrateur");});
164 +
164 165 private List<Employe> modifierEmploye(final Ligue ligue) {
165 166     return new List<>("Modifier un employé", "e",
166 167

```

Ajout du menu pour changer d'admin.

```

64 64 Menu menu = new Menu("Editer " + ligue.getNom());
65 65 menu.add(afficher(ligue));
66 66 menu.add(gererEmployes(ligue)); // menu selectionner un employer ajouter
67 - menu.add(changerAdministrateur(ligue));
67 + menu.add(changerAdministrateur(ligue)); // option changer d'admin
68 68 menu.add(changerNom(ligue));
69 69 menu.add(supprimer(ligue));
70 70 menu.addBack("q");

```

Saisir les dates d'ajout d'un employé dans ligueConsole. Ce code permet d'ajouter une option dans un menu pour permettre l'ajout d'un employé à une ligue donnée.

```
84 84                                     (element) -> editerLigue(element));
85 85                                     }
86 86
87 + // demande en plus les dates lorsqu'on ajoute un employé avec gestion des exceptions
87 88     private Option ajouterEmploye(final Ligue ligue) {
88 89         return new Option("ajouter un employé", "a",
89 90             () -> {
```

Affichage des dates de départ et d'arrivée d'un employé dans employe.java (vers ligne210).