

Documentation

PROJET : PERSONNEL

AUTEUR : FOY ORIANE

Sommaire

Documentation	1
Sommaire.....	1
1. Introduction	1
1.1. Contexte.	2
1.2. Objectifs.	2
1.3. Technologies utilisées.	2
2. Cahier des charges	3
2.1. Besoins fonctionnels.....	3
2.2. Contraintes techniques	3
3. Conception du site.....	3
3.1. MCD.	4
3.2. Arbre heuristique	4
3.3. Arborescence du site.	4
4. Documentation technique	6
4.1 Classes.....	6
1. EmployeConsole:.....	6
2. LigueConsole.....	6
3. PersonnelConsole	7
4. Employe	7
5.Ligue.....	7
6.GestionPersonnel	7
4.2 Exception.....	7
1. DateException	7
2. DroitsInsuffisants.....	7
3. ImpossibleDeSupprimerRoot	7
4. SauvegardeImpossible	7

4.3 Tests Unitaires.....	7
1. LocalDateTests	7
2. TestEmployeSetters	7
3. testLigue.....	7
4.4 Interface.	7
1. Passerelle.....	8
5. Documentation utilisateur.....	8
5.1 Ajouter une ligue.....	8
5.2 Ajouter un employé.....	8
5.3 Modifier une ligue.....	9
5.4 Modifier un employé.....	10
6.Conclusion	11
6.1 Bilan.	11

1. Introduction

1.1. Contexte.

Dans le cadre de ma formation en BTS Services Informatiques aux Organisations, option Solutions Logicielles et Applications Métiers (SLAM), j'ai réalisé un projet de développement d'une application de gestion des employés des ligues pour la M2L, une organisation fictive.

La M2L est un organisme chargé de la gestion et du suivi de plusieurs ligues sportives. Chaque ligue dispose de ses propres employés avec différents niveaux d'accès à la gestion de leur ligue, et la M2L souhaite centraliser la gestion de ces informations.

Pour ce projet, [un code de base de l'application](#) a été fourni, incluant certaines fonctionnalités et la structure initiale. Mon rôle a été de compléter et modifier ce code, en ajoutant des fonctionnalités, en adaptant l'architecture pour la rendre multi-utilisateurs, et en implémentant la gestion des droits et habilitations selon les différents profils.

Le projet s'inscrit dans une démarche professionnelle, car il permet de mettre en pratique mes compétences en Java, en architecture 3-tiers et en gestion de bases de données. L'application constitue également un exemple concret de projet multi-utilisateurs grâce à l'utilisation d'une base de données.

1.2. Objectifs.

L'objectif de ce projet est de développer une application Java afin de mettre en pratique mes compétences acquises en programmation orientée objet et en conception logicielle.

Ce projet me permet de :

- Appliquer et approfondir mes connaissances en Java.
- Développer une application multi-utilisateur utilisant une base de données.
- Comprendre et mettre en œuvre la gestion des droits et habilitations des utilisateurs.
- Acquérir de l'expérience dans la conception et la réalisation d'un projet complet, de la planification à la mise en œuvre.

En somme, ce projet constitue une opportunité de renforcer mes compétences techniques, de produire un projet concret et structuré, et d'illustrer mon savoir-faire en développement logiciel.

1.3. Technologies utilisées.

Langages : Java

Base de données : MySQL

IDE: Eclipse.

Versionning : GitHub

2. Cahier des charges

2.1. Besoins fonctionnels.

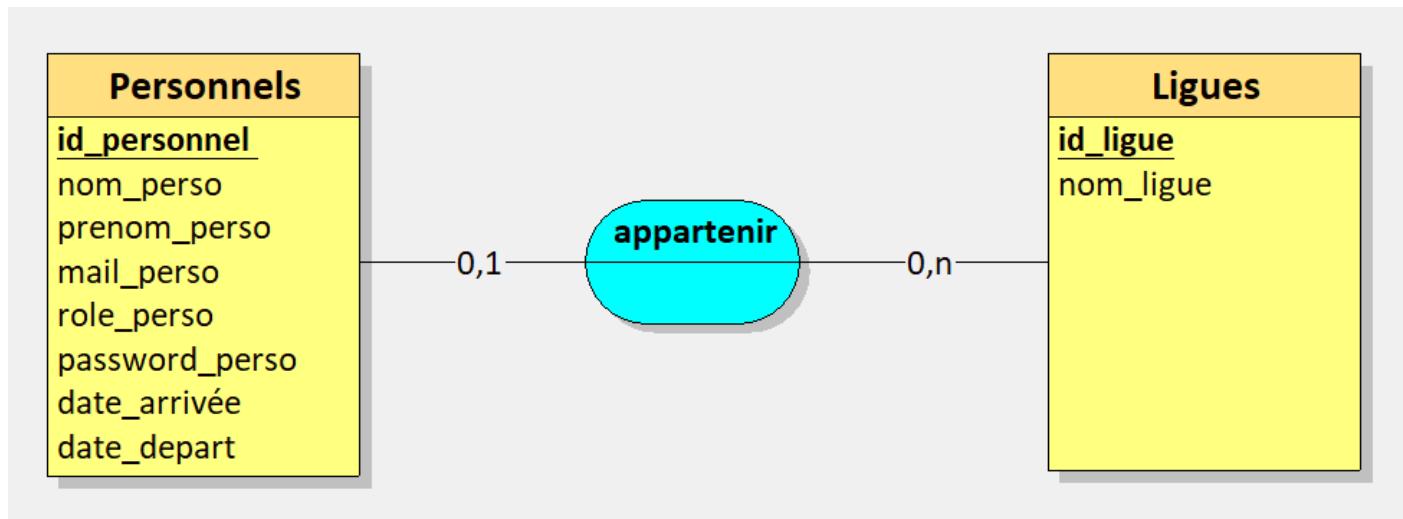
- Un simple employé de ligue peut ouvrir l'application et s'en servir comme un annuaire, mais il ne dispose d'aucun droit d'écriture.
- Un employé par ligue est administrateur et dispose de droits d'écriture peut gérer la liste des employés de sa propre ligue avec une application bureau.
- Le super-administrateur a accès en écriture à tous les employés des ligues. Il peut aussi gérer les comptes des administrateurs des ligues avec une application accessible en ligne de commande.
- L'application doit être rendue multi-utilisateurs grâce à l'utilisation d'une base de données.
- Les trois niveaux d'habilitation ci-dessus doivent être mis en place.
- Ajouter/modifier/supprimer une ligue
- Ajouter/modifier/supprimer un employé

2.2. Contraintes techniques

- L'application doit conserver une séparation claire entre la couche présentation (interface utilisateur), la logique métier et la couche d'accès aux données.
- L'application doit être capable de gérer plusieurs utilisateurs avec des niveaux d'habilitation différents : simple employé, administrateur de ligue, super-administrateur.
- Les données doivent être stockées dans une base de données relationnelle pour assurer la persistance et la sécurité des informations.
- L'application doit être développée en Java.

3. Conception du site

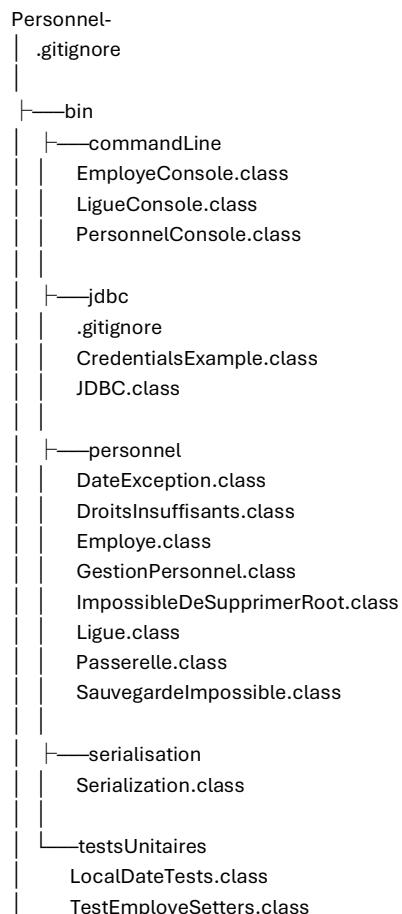
3.1. MCD.



3.2. Arbre heuristique

<https://www.gloomaps.com/er3vhrobnb>

3.3. Arborescence du site.



```
testLigue.class

Personnel
|.classpath
|.gitignore
|.project
pom.xml

|.settings
| org.eclipse.jdt.core.prefs
| org.eclipse.ltk.core.refactoring.prefs
| org.eclipse.m2e.core.prefs

Documents
| Arbre heuristique
| arboAP.jpg

MCD
| MCD.png
| Nouveau MCD.jpg
| ScriptPersonnel.sql
| Thumbs.db

src
| commandLine
| EmployeConsole.java
| LigueConsole.java
| PersonnelConsole.java

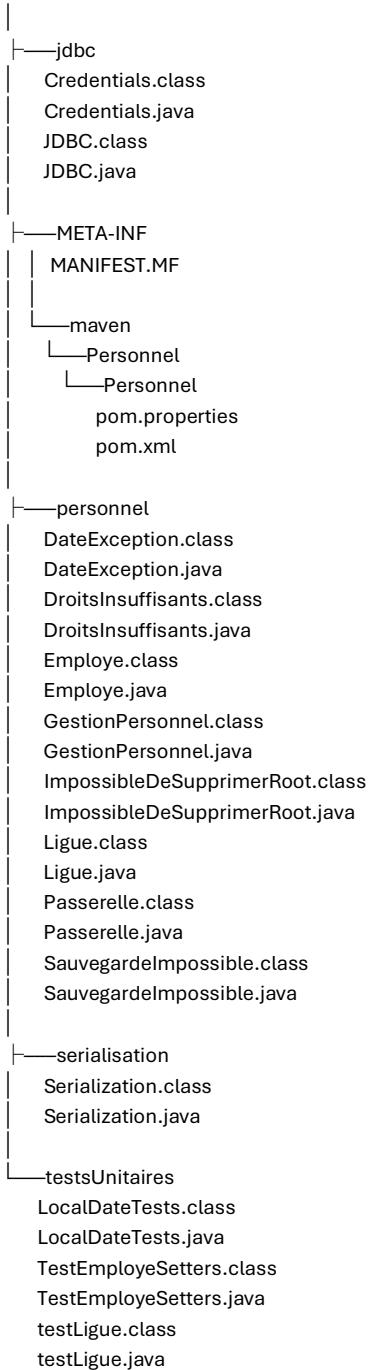
| jdbc
| .gitignore
| Credentials.java
| CredentialsExample.java
| JDBC.java

| personnel
| DateException.java
| DroitsInsuffisants.java
| Employe.java
| GestionPersonnel.java
| ImpossibleDeSupprimerRoot.java
| Ligue.java
| Passerelle.java
| SauvegardeImpossible.java

| serialisation
| Serialization.java

testsUnitaires
| LocalDateTests.java
| TestEmployeSetters.java
| testLigue.java

target
| classes
| commandLine
| EmployeConsole.class
| EmployeConsole.java
| LigueConsole.class
| LigueConsole.java
| PersonnelConsole.class
| PersonnelConsole.java
```



4. Documentation technique

4.1 Classes.

1. EmployeConsole:

Les menus et options dans la console liés aux employés

2. LigueConsole

Les menus et options dans la console liés aux Ligues.

3. PersonnelConsole

Les menus et options dans la console liés à la gestion du personnel.

4. Employe

Employé d'une ligue hébergée par la M2L.

5. Ligue

Représente une ligue.

6. GestionPersonnel

Gestion du personnel.

4.2 Exception.

1. DateException

Exception levée si l'utilisateur tente d'enregistrer une date incohérente ou d'un mauvais format.

2. DroitsInsuffisants

Levée si un administrateur tente d'effectuer une opération sur une autre ligue que la sienne.

3. ImpossibleDeSupprimerRoot

Levée si l'on tente de supprimer le super-utilisateur.

4. SauvegardeImpossible

Exception levée lorsqu'il est impossible de sauvegarder le gestionnaire.

4.3 Tests Unitaires.

1. LocalDateTests

Test unitaires pour vérifier les exceptions de dates.

2. TestEmployeSetters

Test unitaires pour vérifier les méthodes et exceptions liés à la manipulation de la classe employé.

3. testLigue

Test unitaires pour vérifier les méthodes et exceptions liés à la manipulation de la classe ligue.

4.4 Interface.

1. Passerelle

Contrat entre la couche métier et la couche de données.

5. Documentation utilisateur

5.1 Ajouter une ligue.

- Se connecter
- Sélectionner l'option “Gérer les ligues” (l)
- Sélectionner l'option “Ajouter une ligue” (a)
- Donner un nom à la ligue et appuyer sur entrer

?

5.2 Ajouter un employé.

- Se connecter
- Sélectionner l'option “Gérer les ligues” (l)
- Sélectionner l'option “Sélectionner une ligue” (e)
- Sélectionner l'option “Editer <ligue du futur employé>”
- Sélectionner l'option ”Gérer les employés de <ligue du futur employé>”(e)
- Sélectionner l'option “ajouter un employé” (a)
- Remplir toutes les informations relatives à l'employé demandées

```
PersonnelConsole (1) [Java Application] C:\Users\orian\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_21.0.8.v20250724-1412\jre\bin\javaw.exe (1 déc. 2025, 17:46:19 elapsed)
password : toor
Gestion du personnel des ligues
c : Gérer le compte root
l : Gérer les ligues
q : Quitter

Select an option : l

-----
Gérer les ligues
l : Afficher les ligues
a : Ajouter une ligue
e : Sélectionner une ligue
q : Back

Select an option : a
nom : Esclade

-----
Gérer les ligues
l : Afficher les ligues
a : Ajouter une ligue
e : Sélectionner une ligue
q : Back

Select an option : q

-----
Gestion du personnel des ligues
c : Gérer le compte root
l : Gérer les ligues
q : Quitter
```

```
Problems Javadoc Declaration Console X Install Java 25 Support Eclipse IDE for Java Developers 2025-12 RC1
PersonnelConsole (1) [Java Application] C:\Users\orian\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_21.0.8.v20250724-1412\jre\bin\javaw.exe (1 déc. 2025, 17:46:19 elapsed)
Select an option : l

-----
Gérer les ligues
l : Afficher les ligues
a : Ajouter une ligue
e : Sélectionner une ligue
q : Back

Select an option : e

-----
Sélectionner une ligue
1 : Editer Basket
2 : Editer Escalade
3 : Editer Esclade
4 : Editer Foot

Select an option : 1

-----
Editer Basket
l : Afficher la ligue
e : Gérer les employés de Basket
r : Renommer
d : Supprimer
q : Back

Select an option : e

-----
Gérer les employés de Basket
m : Sélectionner un employé
```

```
Select an option : e

-----
Gérer les employés de Basket
m : Sélectionner un employé
v : Afficher les employés
a : ajouter un employé
q : Back

Select an option : a
Date d'arrivée (AAAA-MM-JJ) :
```

5.3 Modifier une ligue.

- Se connecter
- Sélectionner l'option “Gérer les ligues” (l)

- Sélectionner l'option “Sélectionner une ligue” (e)
- Sélectionner l'option “Editer <ligue à modifier>”
- Sélectionner l'option souhaitée : r : Renommer, d : Supprimer

```

PersonnelConsole (1) [Java Application] C:\Users\orian\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_21.0.8.v20250724-1412\jre\bin\javaw.exe (1 déc. 2025, 18:04:11 elapsed)
password : toor
Gestion du personnel des ligues
c : Gérer le compte root
l : Gérer les ligues
q : Quitter

Select an option : l

-----
Gérer les ligues
l : Afficher les ligues
a : Ajouter une ligue
e : Sélectionner une ligue
q : Back

Select an option : e

-----
Sélectionner une ligue
1 : Editer Basket
2 : Editer Escalade
3 : Editer Esclade
4 : Editer Foot

Select an option : 1

-----
Editer Basket
l : Afficher la ligue
e : Gérer les employés de Basket
r : Renommer
d : Supprimer
q : Back

```

5.4 Modifier un employé.

- Se connecter
- Sélectionner l'option “Gérer les ligues” (l)
- Sélectionner l'option “Sélectionner une ligue” (e)
- Sélectionner l'option “Editer <ligue de l'employé>”
- Sélectionner l'option ”Gérer les employés de <ligue de l'employé>”(e)
- Sélectionner l'option : “ Sélectionner un employé” (m)
- Sélectionner l'option “1 : Gérer le compte <nom employé>
- Choisir la modification à apporter à l'employé

```
Problems @ Javadoc Declaration Console X i Install Java 25 Support i Eclipse IDE for Java Developers 2025-12 RC1

PersonnelConsole (1) [Java Application] C:\Users\orian\p2\pool\plugins\org.eclipse.jdt.openjdk.hotspot.jre.full.win32.x86_64.21.0.8.v20250724-1412\jre\bin\javaw.exe (1 déc. 2025, 17:58:52 elapsed)

password : toor
Gestion du personnel des ligues
c : Gérer le compte root
l : Gérer les ligues
q : Quitter

Select an option : l

-----
Gérer les ligues
l : Afficher les ligues
a : Ajouter une ligue
e : Sélectionner une ligue
q : Back

Select an option : e

-----
Sélectionner une ligue
1 : Editer Basket
2 : Editer Escalade
3 : Editer Esclade
4 : Editer Foot

Select an option : 1

-----
Editer Basket
l : Afficher la ligue
e : Gérer les employés de Basket
r : Renommer
d : Supprimer
q : Back

Select an option : m

-----
Sélectionner un employé
i : Gérer le compte ori

Select an option : i

-----
Gérer le compte ori
l : Afficher l'employé
n : Changer le nom
p : Changer le prénom
e : Changer le mail
x : Changer le password
u : Changer la date du début de contrat
r : Ajouter la date de fin de contrat
k : Mettre admin
z : Supprimer l'employé
q : Back

Select an option :
```

6. Conclusion

6.1 Bilan.

La création de ce portfolio m'a permis de m'exercer aux langages HTML, CSS et JavaScript, ainsi qu'à

La réalisation de cette application Java m'a permis de mettre en pratique mes compétences en programmation orientée objet et de mieux comprendre l'architecture 3-tiers dans un contexte concret.

Travailler à partir d'un code de base fourni et le compléter m'a appris à analyser un projet existant, identifier les points à améliorer et implémenter de nouvelles fonctionnalités tout en respectant l'architecture existante.

Enfin, ce projet constitue un exemple concret de développement logiciel professionnel, me permettant de consolider mes compétences en Java, en gestion de bases de données et en conception d'applications complètes, tout en produisant un projet fonctionnel et structuré.