

# MarieTeam Client Lourd

## **Présentation**

Ce projet Java, est une application cliente destinée à la consultation et la génération de brochures pour une compagnie maritime fictive, MarieTeam.

Elle permet notamment de charger les données de la base de données, de manipuler des objets métiers (bateaux, équipements...), et de générer dynamiquement un PDF de présentation des bateaux voyageurs.

Le projet s'articule autour d'un ensemble de classes Java, d'une base de données relationnelle (MySQL) et d'un outil de génération de documents PDF.

## **Technologies utilisées**

- Java 17 ou supérieur  
Langage principal de développement, orienté objet.
- Maven  
Utilisé pour la gestion des dépendances et la compilation du projet.
- MySQL  
Base de données utilisée pour stocker les informations sur les bateaux, traversées, utilisateurs, etc.
- JDBC  
Pour les connexions et requêtes SQL à la base de données.
- iText (v5.5.13.3)  
Bibliothèque de génération de documents PDF.



## **Structure du projet**

```
PjtMarieTeamCL/  
├─ src/  
│   └─ main/java/fr/dly_vrn/  
│       ├── Bateau.java  
│       ├── BateauVoyageur.java  
│       ├── Brochure_PDF.java  
│       ├── Equipement.java  
│       ├── Passerelle.java  
│       └─ JeuEnregistrement.java  
├─ src/main/resources/  
│   └─ db.properties  
├─ marieteam_bdd.sql  
├─ pom.xml  
└─ BateauVoyageur.pdf (exemple généré)
```

## **Fonctionnalités**

### *Chargement des données*

Le chargement des bateaux voyageurs et de leurs équipements est centralisé dans la classe Passerelle. Les données sont extraites via des requêtes SQL exécutées à travers la classe JeuEnregistrement, qui encapsule la logique JDBC.

### *Génération PDF*

La classe Brochure\_PDF utilise la bibliothèque iText pour générer automatiquement une brochure au format PDF, contenant la description et l'image de chaque bateau.

### *Interface utilisateur avec édition des bateaux*



L'application inclut une interface graphique (probablement basée sur Java Swing) qui permet de visualiser et de modifier les informations des bateaux. Par exemple :

- Changement du nom d'un bateau
- Mise à jour des dimensions (longueur, largeur)
- Modification de la vitesse

Ces modifications sont ensuite enregistrées et peuvent être exploitées dans la génération du PDF.

### *Interaction avec la base de données*

Les informations sont stockées dans une base MySQL. La configuration est définie dans le fichier db.properties :

```
db.url=jdbc:mysql://localhost:3306/marieteam_bdd
db.user=root
```

## **Modèle objet**

### *Bateau*

Classe de base représentant un bateau : nom, longueur, largeur, etc.

### *BateauVoyageur*

Hérite de Bateau. Contient des informations supplémentaires : vitesse, image, liste d'équipements.

### *Equipement*

Objet simple contenant un identifiant et une description.

### *Passerelle*

Classe utilitaire pour charger les données depuis la base. Contient notamment :



- chargerLesBateauxVoyageurs()
- chargerLesEquipements(idBateau)

### ***Brochure\_PDF***

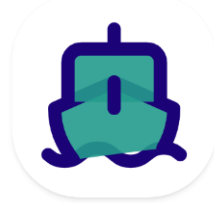
Permet de générer le document PDF à partir d'une collection de BateauVoyageur.

## **Base de données**

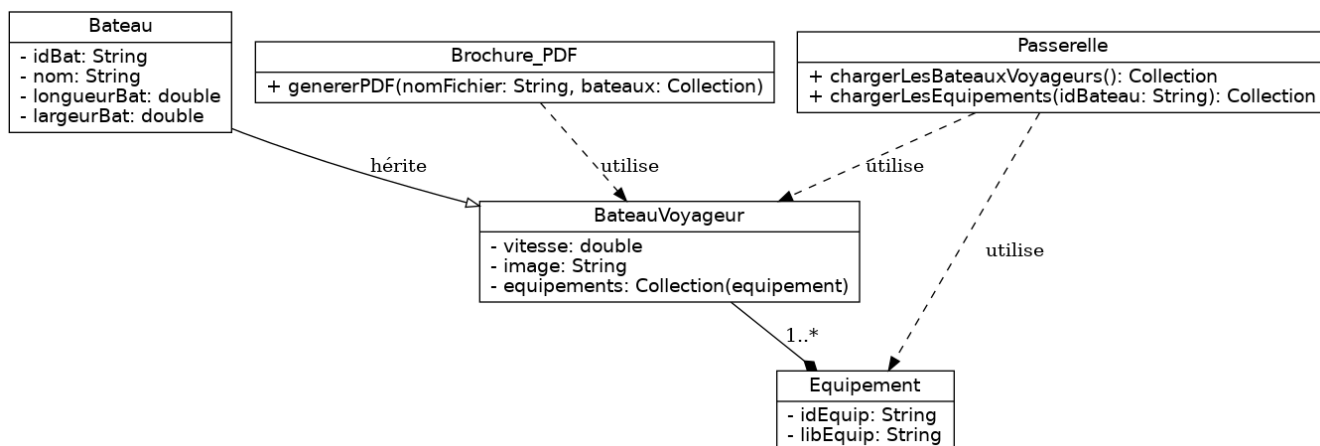
La structure de la base est définie dans le fichier marieteam\_bdd.sql. Elle comprend entre autres les tables suivantes :

Table	Description
<b>bateau</b>	Informations sur les bateaux
<b>equipement</b>	Liste des équipements possibles
<b>bateau_equipement</b>	Table de liaison entre bateaux et équipements
<b>liaison, traversee</b>	Gèrent les trajets et les horaires
<b>type, tarifer</b>	Tarifs en fonction du type et de la période
<b>reservation</b>	Réservations effectuées
<b>utilisateur</b>	Clients et administrateurs

## **Diagramme UML**

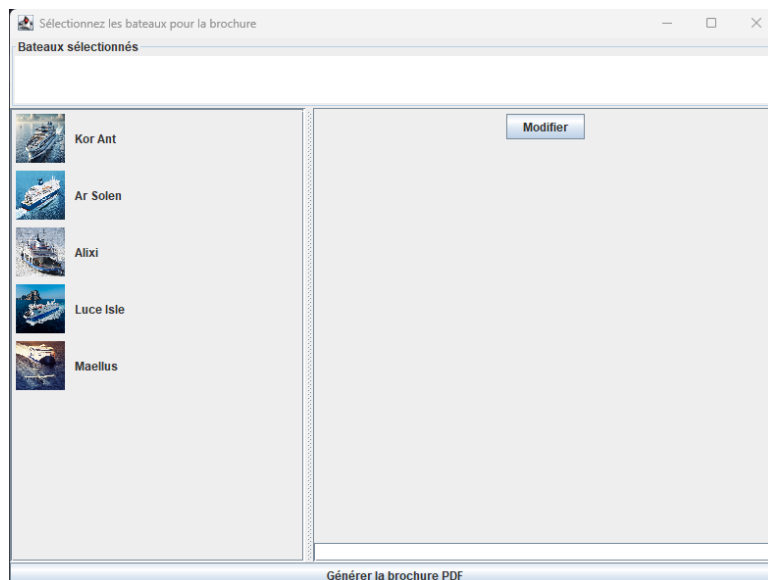


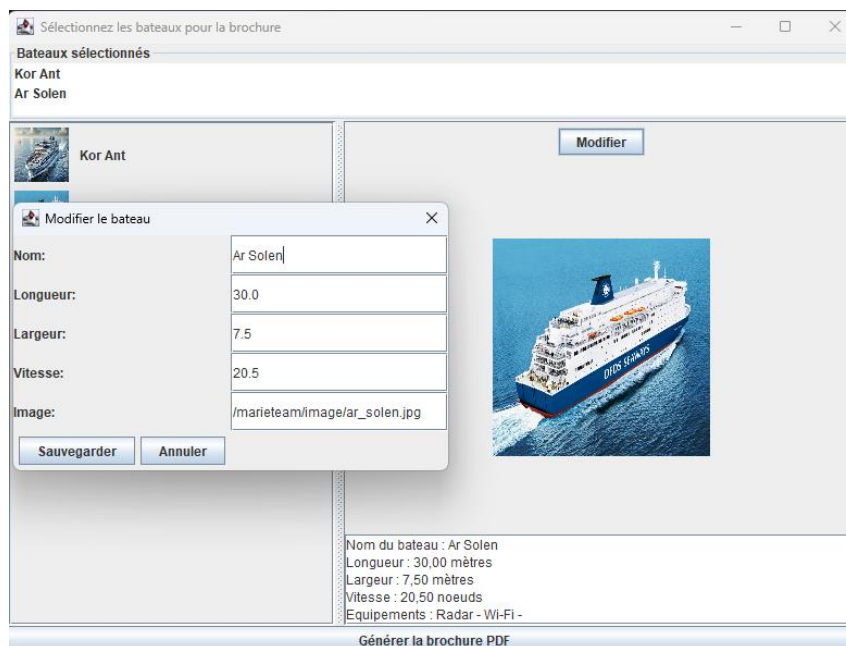
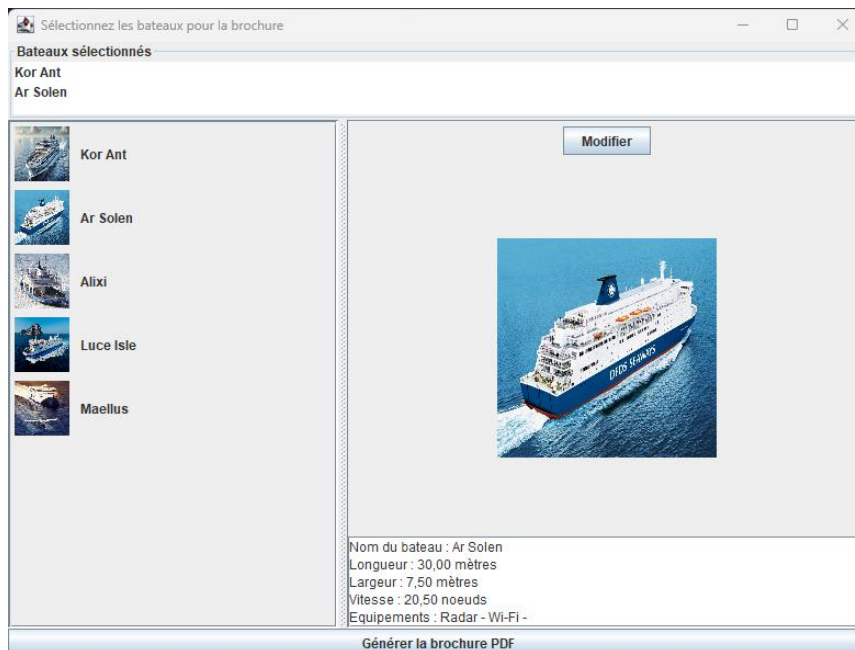
Un diagramme UML a été réalisé pour illustrer les relations entre les principales classes Java du projet :



Ce projet est un bon exemple de séparation des responsabilités en Java, avec un accès structuré aux données, une modélisation orientée objet claire et une sortie imprimable via PDF. Il peut servir de base pour une extension vers une interface graphique ou une application web plus complète.

## **Illustration du projet :**



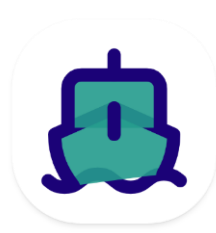


## COMPETENCES

### 1.4 Travailler en mode projet

Ce projet a été réalisé en collaboration avec mon collègue **Noah Verin**

On a travaillé via un repository GitHub



Repository page for **PjtMarieTeamCL** by **Romaindly**. The repository is private and has 1 branch (master) and 0 tags. It shows 2 commits from yesterday.

File	Commit Message	Commit Date
.idea	chore: initial commit	yesterday
src/main	Initial commit	yesterday
.gitignore	Initial commit	yesterday
BateauVoyageur.pdf	Initial commit	yesterday
BateauVoyageur_Selection.pdf	Initial commit	yesterday
marieteam_bdd.sql	Initial commit	yesterday
pom.xml	Initial commit	yesterday

There is also a **README** file.

## 1.5 Mettre à disposition un service informatique

Le projet permet aux utilisateurs de :

**Consulter les bateaux** depuis une base de données,

**Modifier leurs informations** (nom, dimensions, équipements...),

**Générer une brochure PDF** automatiquement.

Tout fonctionne dans une application avec interface graphique, c'est donc un **service prêt à l'emploi** pour un utilisateur.