Adrian CHIAPELLO, MARTIN LECHEMIA, MATHIEU TRUCHOT, TOMMY DESHAIRS



Projet MASTERMIND

Version 1.0

February 20, 2018

# Projet MASTERMIND

## DOCUMENT DE COMMUNICATION DU PROJET

Documentation technique pour le projet Mastermind réalisé en Java

### TABLE DU PROJET

| Document | Destinataire | Tâche | Dernière MàJ |
| --- | --- | --- | --- |
| Développement | Martin & Tommy | Logiciel | 20/02/2018 |
| Architecture | Tommy & Martin | Base de Données | 20/02/2018 |
| Graphiques | Mathieu & Adrian | Gantt | Diagrammes | 20/02/2018 |
| Documentation | Mathieu & Adrian | Finalisation | 20/02/2018 |
| Application Testing | Mathieu & Adrian | Tests | Corrections | 20/02/2018 |
|  |  |  |  |
|  |  |  |  |

## EQUIPE

Le projet est composé de 4 acteurs devant réaliser un Mastermind en language Java

### Objectifs de l’équipe

* Application stable
* Jeu fonctionnel
* Documentation Technique
* Test de l’application
* Développement Grand Public

### Répartition de l’équipe

**Mastermind**

| Nom de l’équipe | Objectifs | Leader | Role |
| --- | --- | --- | --- |
| Martin & Tommy | Application Stable | Martin | Développement |
| Martin & Tommy | Jeu Fonctionnel | Martin | Développement |
| Adrian & Mathieu | Documention Technique | Mathieu | Documentation |
| Adrian & Mathieu | Test de l’application | Adrian | Test / Debug / Report |
| Equipe | Développement Grand Public | Martin | Développement |

### HISTorique & RÈGLES DU JEU

1. Historique

Le mastermind est un jeu de société pour deux joueurs dont le but est de trouver une combinaison de couleurs. C’est un jeu de réfléxion et de déduction inventé par Mordecai Meirowitz

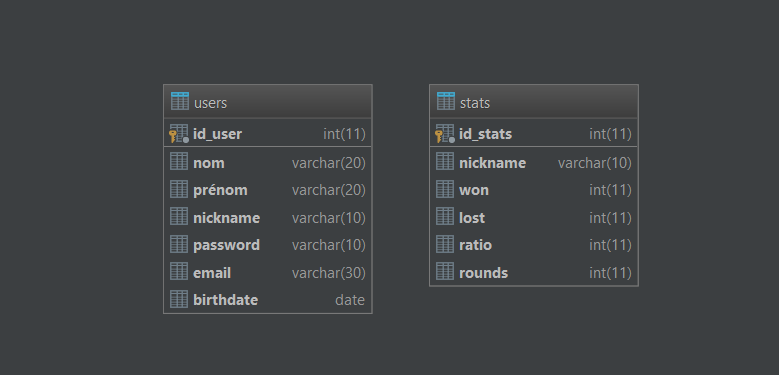
1. Règles du jeu
   1. Lorsque l’utilisateur doit deviner

La combinaison est générée par l’Intelligence Artificielle :

* La combinaison comporte 5 chiffres
* La combinaison peut comporter des doublons
* La combinaison est composée de chiffres entre 1 et 9
* 10 tentatives pour trouver la combinaison
  1. Lorsque l’IA (Intelligence Artificielle) doit deviner
* La combinaison doit comporter 5 chiffres
* Possibilité de proposer des doublons
* Combinaison composée de chiffres entre 1 et 9
* Au bout de 10 tentative l’utilisateur gagne contre l’IA

### DÉVELOPPEMENT

1. Structure de la base de donnée



La Base de donnée est composée de deux tables :

* Users
* Stats
  1. La table user

Id\_user : Détermine l’id de l’utilisateur

Nom : Nom de famille de l’utilisateur

Prénom : Prénom de l’utilisateur

Nickname : Pseudo de l’utilisateur

Password : Mot de passse de l’utilisateur

Email : Email de l’utilisateur

Birthdate : Anniversaire de l’utilisateur

* 1. La table stats

Id\_stats : Détermine l’ID de la statistique du Joueur

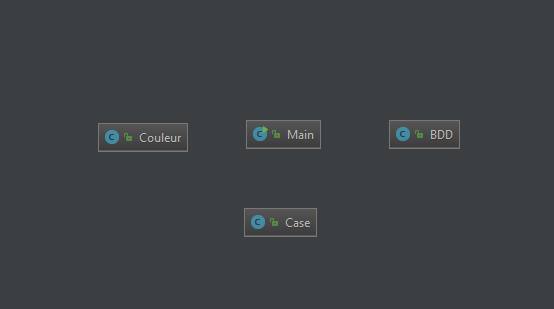
Nickname : Pseudo des utilisateurs ayant joué

Won : Nombre de victoire(s)

Lost : Nombre de défaite(s)

Ratio : Moyenne des statistiques : Victoires / Défaites

Rounds : Nombre de tours moyen avant d’avoir trouvé la combinaison

1. Architecture

Le projet est composé d’un Main, d’une classe Couleur, d’une classe Case et d’une Classe BDD

1. Classe Main
2. Connexion

La connexion identifie l’utilisateur au démarrage de l’application, s’il n’est pas enregistré il peut le faire directement dans l’interface. Après s’être connecté il pourra modifier son compte s’il le souhaite.

1. Choix du Jeu

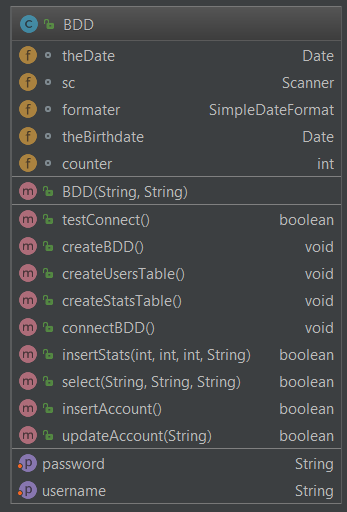
L’utilisateur a le choix entre deux modes de jeu soit jouer contre l’IA, dans ce cas là l’IA lui fournit une combinaison à trouver. Le deuxième choix lui permet de faire deviner la combinaison à l’ordinateur.

1. Aide

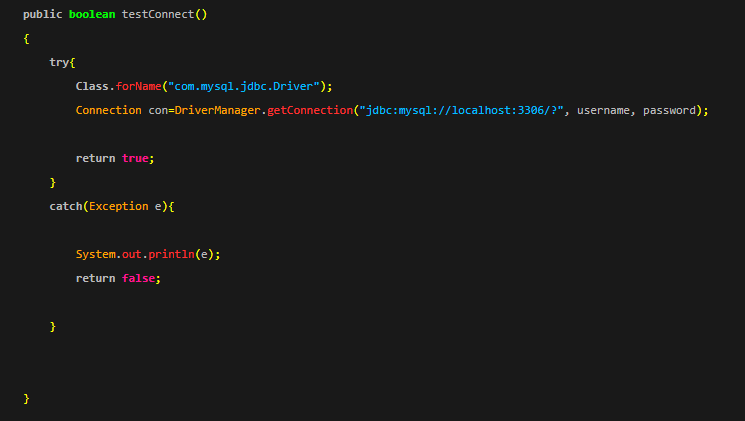
L’aide permet d’afficher les règles du jeu à l’utilisateur afin qu’il comprenne le fonctionnement du jeu.

1. Déroulement de la partie
2. Classe BDD

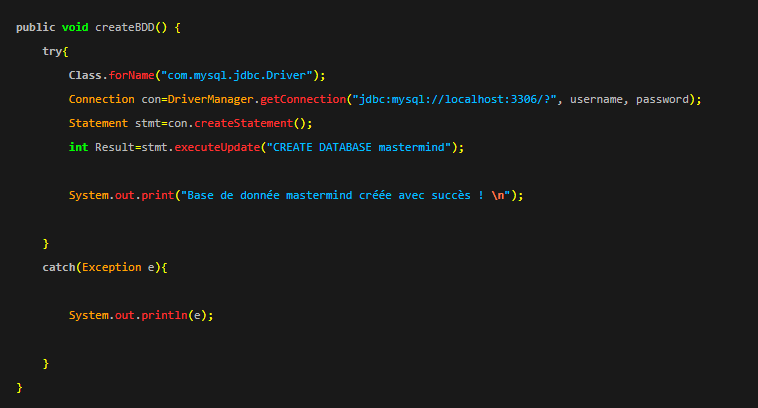
La classe Base de donnée contient un script qui crée la base de données et les tables lors de la première connexion. Il contient également toutes les fonctions de requêtes SQL



On réalise d’abord un test de connexion à la BDD avec la fonction testConnect :



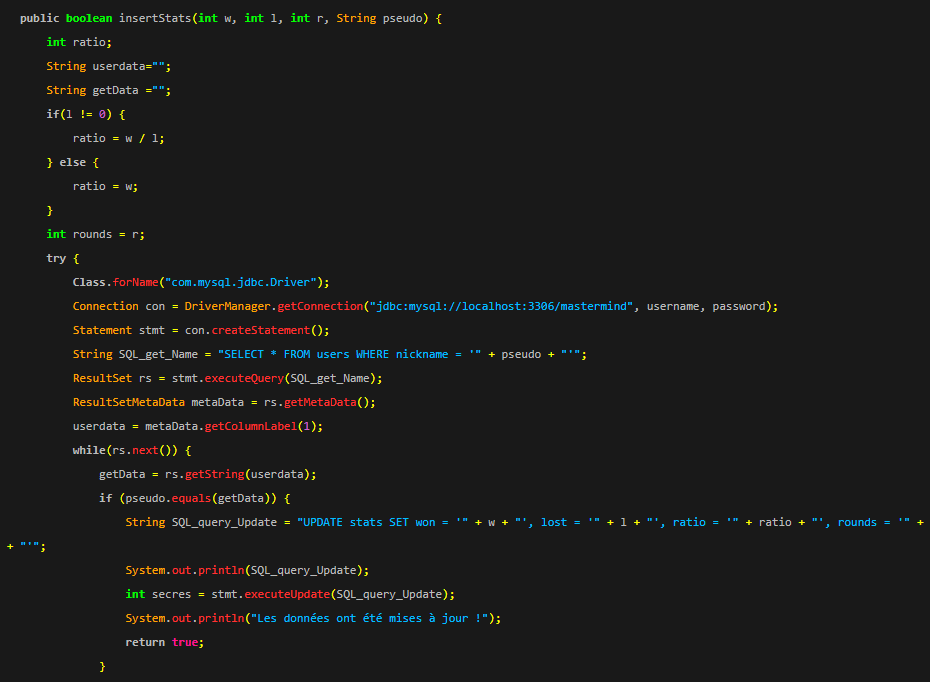
La fonction createBDD() crée la base de donnée, dans le cas où elle existe déjà on retourne une erreur avec Base de donnée déjà créée.



Fonction de création de table Users et Stats :



La function insertStats() permet d’insérer les statistiques dans la Base de donnée en fonction des statiques du joueur (Win/Loose/Ratio + Pseudo) :



La fonction select permet de retourner le résultat d’une requête en BDD.

On y place en paramètre, le nom du champs, le nom de la table et la donnée entrée par l’utilisateur



La fonction insertAccount démarre lorsque l’utilisateur spécifie qu’il n’a pas de compte.



Elle lui propose alors des champs à renseigner pour compléter ses informations et créer son compte

1. Classe Couleur

La Classe couleur permet de gérer les couleurs et de retourner si elle est présente ou non. Elle retourne également les positions possibles.

1. Classe Case

La Classe Case permet de retourner les cases où les couleurs ont été retournées.

Booléen -> Trouvé

Booléen Couleur -> True / False