

# Projet Java - Master Mind

Cahier des charges



# TABLE DES MATIERES

Contexte	3
Objectifs	3
Fonctionnement	3
Ressources	3
Livrables	4
Délais	4

## → Contexte

Avant de le proposer sur smartphones (un graphiste est en train de réaliser la maquette de l'UX), HASBRO vous demande de réaliser en Java son jeu de MasterMind sans UI graphique.

## → Objectifs

Plusieurs objectifs devront être atteints pour réaliser le projet

- Appliquer les règles d'algorithmie en Java
- Penser et concevoir « objet »
- Créer les entités nécessaires (classes, constructeurs, méthodes...)
- Manipuler les tableaux
- Apprendre à déboguer efficacement
- Conceptualiser et manipuler une base de données
- Réaliser une application console de mastermind.
- Réaliser l'authentification de l'application

## → Fonctionnement

L'application devra demander à l'utilisateur de s'authentifier ou de s'inscrire la première fois. Puis l'utilisateur atteindra un menu :

- Il pourra choisir le mode de jeu dans un menu
- afficher son profil
- le modifier
- afficher des statistiques et les exporter en .csv

Enfin il pourra quitter l'application.

## → Ressources

Pour ce projet, nous pouvons compter sur une équipe de 4 personnes : Cyril Martorell, Romain Iacopino, Andy Lopy, Jérémie Lopez.

Le projet sera développé en Java, la base de données sera modélisée en sql.

Nous utilisons Git pour collaborer notre code.

## → Livrables

Nous devons livrer un zip du dossier racine de la source de votre application, un exécutable ainsi qu'un rapport. La source devra comprendre une classe Main (programme principal) et un fichier séparé pour chacune des classes utilisées.

Nous pouvons répartir nos classes en différents packages si nous le souhaitons. Le rapport contient une introduction, une présentation des acteurs du projet, un cahier des charges, les spécifications de l'application, modélisation de la base de données, diagramme de classes métier, diagramme de cas d'utilisation, diagramme de séquence, algorithmes mis en places (pseudo-code et explications de ces derniers), règles du jeu, mode d'emploi et de déploiement du jeu, patterns de conception mis en œuvres, planification, annexes ainsi qu'un bilan final du projet.

Notre code doit être 100% personnel et annoté afin de générer une Javadoc. Aucune API externe tolérés. Une fois dézippée, l'application devra être immédiatement compilable.

## → Délais

Nous avons 18h en tout et pour tout, pour réaliser le projet.