

Projet : un jeu de morpion avancé

UCBL - Département Informatique de Lyon 1 – BDW - 2025

- Ce projet est évalué par une note de soutenance. Un suivi d'avancement sera réalisé à chaque séance.
- Cette UE est avant tout une introduction aux **bases de données**. La non-utilisation d'une BD relationnelle dans le projet peut se voir attribuer la note 0.
- L'utilisation du serveur **bdw-server** est obligatoire.
- Le rendu du projet et la soutenance ont lieu lors de la dernière séance de TP. Tout rendu ne respectant pas les consignes peut se voir attribuer la note 0.
- Le projet est réalisé en binôme (pas de trinôme). Un binôme signifie deux étudiant·e·s **du même groupe de TD**.
- Attention à la gestion de votre temps : concentrez-vous en priorité sur l'implémentation des fonctionnalités demandées. Cela ne sert à rien de gagner un point bonus pour quelques améliorations mineures mais de perdre plusieurs points pour des fonctionnalités non développées.

L'objectif du projet est de développer un jeu de morpions¹ "avancé" (i.e., les morpions se tapent dessus).

Dans le jeu de morpion classique, il faut être le ou la première à aligner sur une grille une rangée de morpions (soit horizontalement, soit verticalement, soit en diagonale). Pour simplifier, on considère que les deux joueuses jouent sur la même machine, chacune leur tour (pas de jeu en réseau), et en effectuant une seule action par tour parmi : placer un morpion sur la grille, attaquer un morpion, lancer un sort. La grille est limitée aux dimensions 3x3 cases et 4x4 cases. Dans cette version avancée, les règles sont étendues pour que les morpions soient "actifs" pendant la partie, mais elles restent volontairement basiques pour ne pas surcharger la programmation de l'application. Une équipe gagne quand elle aligne une rangée de morpions (3 morpions sur une grille de 3x3 ou 4 morpions sur une grille de 4x4), ou quand tous les morpions adverses sont morts, ou quand l'adversaire ne peut plus faire d'action. La partie s'arrête également quand le nombre maximal de tours est atteint (aucun gagnant). Les morpions ont 4 caractéristiques : points de vie, points d'attaque, points de mana, points de réussite. La somme des points de ces 4 caractéristiques vaut 15 (valeur par défaut d'un paramètre de configuration) et un morpion possède au minimum 1 point dans chaque caractéristique. Quand les points de vie d'un morpion sont à zéro, il est mort (et la case de la grille redévient libre). Quand un morpion attaque ou lance un sort, on évalue d'abord si l'action réussit : la probabilité de réussite est égale à $10 \times \text{points de réussite}$ (i.e., une valeur tirée aléatoirement entre 0 et 100 doit être inférieure à cette probabilité pour réussir l'action²). Pour chaque action réussie, le morpion gagne 0.5 point de réussite. Pour les attaques : un morpion peut attaquer un morpion situé sur une case adjacente horizontale ou adjacente verticale à la sienne. Quand une attaque est réussie, l'attaquant inflige un nombre de dégâts égal à ses points d'attaque (i.e., on déduit les points d'attaque de l'attaquant aux points de vie de l'attaqué). Pour les sorts, trois sont disponibles : la traditionnelle *boule de feu*, qui inflige 3 points de dégâts à un morpion, le réconfortant sort de *soin*, qui ajoute 2 points de vie à un morpion, et le redouté sort *armageddon*, qui détruit complètement une case (ce qui la rend inutilisable par la suite et tue un éventuel morpion placé dessus). Quand un mage essaie de lancer un sort, il perd un nombre de points de mana égal au coût du sort, à savoir 2 pour la boule de feu, 1 pour le sort de soin et 5 pour armageddon. Enfin, les morpions sont loyaux (on n'attaque pas un morpion de la même équipe !) et non kamikazes (pas d'armageddon sur sa propre case!).

¹Règles du jeu de morpion, [http://fr.wikipedia.org/wiki/Morpion_\(jeu\)](http://fr.wikipedia.org/wiki/Morpion_(jeu)) et <http://fr.wikihow.com/jouer-au-morpion>

²Oui, certains morpions très habiles peuvent avoir plus de 100% de réussite...

1 Des spécifications au script SQL

Les spécifications sont les suivantes. Dans ce jeu, un morpion possède un nom, une image, des points de vie, d'attaque, de mana, de réussite. Une équipe possède un nom, une couleur unique (pour distinguer les équipes), une date de création et entre 6 et 8 morpions. Notez qu'en réalité, il s'agit de *template* ou modèle de morpion, donc un morpion peut faire partie de plusieurs équipes. Une partie oppose deux équipes, et on stocke ses dates de début et de fin, et l'équipe gagnante. Une partie utilise une configuration datée, qui permet de stocker les paramètres : taille de la grille et nombre maximal de tours. Enfin, on conserve un journal qui liste toutes les actions réalisées. Chaque ligne du journal s'identifie par un numéro unique au niveau de la partie. Chaque ligne est datée et possède un texte décrivant l'action réalisée.

Dans un premier temps, nous allons concevoir la base de données à partir des spécifications. Votre modélisation doit respecter au mieux ces spécifications. Les délivrables produits (diagramme, schéma relationnel, scripts SQL) ne sont pas à rendre mais peuvent être demandés pendant une soutenance.

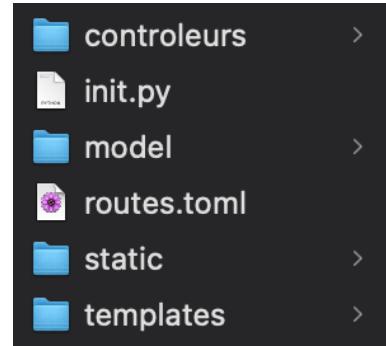
1. Produisez un diagramme entité / association pour ces spécifications (par exemple avec [MoCoDo](#), [Looping](#), [AnalyseSI](#) ou [JMerise](#) - attention, chaque outil a ses contraintes et limitations).
2. Produisez le schéma relationnel dérivé de votre diagramme E/A. Si vous générez ce schéma avec un outil de modélisation, il est recommandé de le vérifier et éventuellement de le corriger / compléter.
3. Produisez le script SQL permettant la création de la base de données. Si vous générez ce script avec un outil de modélisation, il sera nécessaire de le vérifier et de le corriger / compléter (ces modifications devraient être stockées dans un autre fichier que celui généré par l'outil de modélisation). Enfin, insérez quelques instances fictives dans les tables.

2 Design du site et pages statiques

Votre site est codé selon l'architecture MVC, et vos fichiers doivent impérativement respecter l'arborescence suivante pour fonctionner avec `bdw-server` : des répertoires `controleurs`, `model`, `static` et `templates` (voir TP4 et TP5).

Vous êtes libres de concevoir vos pages comme bon vous semble. Mais chacune de vos pages doit avoir les zones suivantes :

- Un entête (`<header>`, avec un logo et un nom de site) ;
- Un menu (`<nav>`), dont les libellés seront explicites ;
- Le contenu de la page (`<main>`), qui correspond aux fonctionnalités développées dans les sections suivantes.
- Un pied de page (`<footer>`), avec un lien vers la page de BDW et des remerciements (e.g., pour les images).



Il ne faut **pas** implémenter un système d'authentification³ ! La mise en page et mise en forme se feront évidemment avec des styles CSS. L'esthétique est prise en compte lors de la notation, aussi, soignez votre site. Le projet sera évalué avec le navigateur **Mozilla Firefox**, donc vérifiez le rendu de votre site avec ce navigateur !

3 Fonctionnalité 1 : accueil et statistiques

Concevez la page d'accueil de votre site. Personnalisez-la comme vous le souhaitez (e.g., objectifs du site, tutoriel).

Cette page d'accueil affiche également des statistiques, notamment :

- Nombre d'instances pour 3 tables de votre choix ;
- Top-3 des équipes avec le plus de victoires ;
- Partie la plus rapide et celle la plus longue ;
- Nombre moyen de lignes de journalisation, pour chaque couple (mois, année).

³Un système d'authentification se base sur des protocoles standardisés et sécurisés comme SSL, Kerberos ou CAS.

4 Fonctionnalité 2 : gestion des équipes

Cette seconde fonctionnalité consiste à créer et supprimer une équipe :

- Ajouter une dizaine de morpions directement dans votre BD (e.g., avec des requêtes SQL). Vous devez obligatoirement utiliser les images fournies pour les morpions ;
- Développer une page qui permet de créer une nouvelle équipe, en sélectionnant des morpions parmi ceux présents dans la BD ;
- Développer une page qui liste les équipes disponibles, et qui permet de supprimer une équipe.

5 Fonctionnalité 3 : partie normale

Cette fonctionnalité permet à 2 personnes de jouer une partie normale (sans les règles avancées) :

- Développer une page pour choisir 2 équipes, une taille de grille et un nombre maximal de tours. Ces 2 derniers paramètres constituent la configuration de la partie ;
- Développer une page pour gérer un tour de jeu. Vous êtes libres concernant le style graphique de la grille. La joueuse dont c'est le tour choisit un morpion de son équipe et une case libre où le placer ;
- Vérifier les conditions d'arrêt (victoire, grille pleine, nombre de tours atteint).

Les informations sur la partie seront progressivement stockées en base de données, en particulier dans le journal. Réfléchissez aux structures de données (en prenant en compte la fonctionnalité 4). Pensez à gérer les erreurs (messages pertinents).

6 Fonctionnalité 4 : partie avancée

L'objectif est de créer une nouvelle page pour jouer avec les règles avancées. Il est fortement conseillé de conserver la fonctionnalité 3 et de créer de nouveaux fichiers. La page qui gère le tour permet en plus de gérer les combats et les sorts.

Lors de la soutenance, il est fort probable que les enseignant·e·s testent votre application en saisissant des valeurs absurdes susceptibles de déclencher des erreurs dans votre application. Alors essayez de penser au pire ! Libres à vous d'ajouter de nouvelles fonctionnalités (e.g., partie contre la machine, nouveaux sorts).



Exemple de grille (ici victoire de la joueuse verte)

7 Préparation des livrables

Deux livrables sont à rendre **le jour de soutenance de votre projet, avant 23h59, sur Tomuss** :

- Une archive de votre site web, en **zip** ou **rar** (colonne *archive_projet*), qui contient a-minima :
 - le répertoire de votre site (code complet, commenté et indenté, respectant l'arborescence de la section 2 et incluant votre fichier de configuration **config.toml**) ;
 - les fichiers de conception de la BD, i.e., le diagramme entité/association (format **png** ou **pdf**), le schéma relationnel sous forme textuelle (format **txt**, **pdf**, **html**, ou **markdown**) et le script SQL **exécutable** de création de votre base de données avec des instances en nombre suffisant (format **txt** ou **sql**) .
- Une affiche en **pdf**, de 1 page maximum (colonne *affiche_projet*). Ce document n'est pas évalué directement, mais offre à vos enseignant·e·s un aperçu de votre site (sans l'installer), notamment pour harmoniser les notes après la soutenance. Sur cette affiche (au style graphique libre), vous mettrez :
 - les noms et prénoms du binôme ;
 - un résumé des fonctionnalités implémentées (e.g., sous forme de liste ou tableau) ;
 - le diagramme entité/association ;
 - des captures d'écran annotées de votre site.

8 Soutenance

La soutenance dure environ 20 minutes par binôme et se décompose en deux parties : une démonstration du site (5 minutes) et une séance de questions (15 minutes). Les conditions suivantes s'appliquent :

- Une bonne démonstration doit être préparée : réfléchissez à un scénario (e.g., ce que vous allez montrer, dans quel ordre, quelles valeurs seront saisies dans les formulaires) et à la manière de le présenter (e.g., transitions, répartition du temps de parole, interaction ou pas avec l'enseignant-e). Concentrez-vous sur les points importants (e.g., si vous avez implémenté les dernières fonctionnalités, inutile de montrer les liens dans le footer...). Enfin, respectez le temps (vous serez interrompus au bout des 5 minutes, et pénalisés) ;
- Les deux membres du binôme doivent parler lors de la démonstration. Entraînez-vous ;
- Vous présentez soit sur votre machine, soit sur une machine de la fac. Prévoyez d'arriver quelques minutes en avance devant la salle, et quand on vous fait entrer, préparez-vous pour la présentation et les questions (i.e., lancement du site dans le navigateur, ouverture des fichiers source) ;
- Vous devez répondre à différentes questions pendant les 15 minutes. Donc donnez des **réponses claires et concises**. Si vous ne savez pas répondre à une question, dites-le pour ne pas perdre de temps ;
- Si vous arrivez en retard à votre soutenance, vous passez en fin de séance... ou pas du tout selon les disponibilités de votre jury.