

Programmation événementielle :  
Visual Basic .NET

## Le jeu du *Démineur*

### Principe du jeu

” Le **Démineur** (*Minesweeper*) est un jeu vidéo de réflexion dont le but est de localiser des mines cachées dans une grille représentant un champ de mines virtuel, avec pour seule indication le nombre de mines dans les zones adjacentes. ”

(source Wikipedia<sup>1</sup>)

Dans notre variante, le joueur ne disposera que d’un **temps limité** pour déterminer la position des mines sur le terrain. Ce projet est à réaliser en **binôme**. Toute partie de code (ou de rapport) partagée entre binômes sera considérée comme du plagiat et lourdement sanctionnée comme tel.

## 1 Partie ressource : le jeu (20 pts)

### Base de développement (10pts)

L’application devra débiter en affichant au joueur un *formulaire d’accueil*. Ce formulaire présentera, au minimum :

- une *ComboBox* permettant de saisir le nom du joueur. Une saisie dans une de ces *ComboBox* affichera la liste des joueurs déjà connus de l’application ;
- un *Button* permettant de lancer une nouvelle partie en appelant le formulaire de jeu. Le lancement de la partie n’est possible que si le joueur a entré un **nom d’au moins 3 caractères** ;
- un *Button* permettant de quitter l’application, à confirmer par une *MsgBox* ;
- un *Button* lançant l’affichage du tableau des scores via un formulaire dédié.

Une fois la partie débutée, le jeu présentera au joueur un grilles de **64 cases**, représentées par des *Buttons*, disposées en un carré de 8 sur 8. Le joueur disposera alors de **60 secondes** pour **localiser un maximum** des **10 mines** disposée idéalement aléatoirement. A tout moment, un premier *Label* rappellera le nom du joueur tandis qu’un second indiquera le temps restant pour résoudre le jeu. Un *Button* permettra d’abandonner la partie en cours après confirmation par une *MsgBox*.

---

1. [https://fr.wikipedia.org/wiki/Démineur\\_\(genre\\_de\\_jeu\\_vidéo\)](https://fr.wikipedia.org/wiki/Démineur_(genre_de_jeu_vidéo))

Le joueur pourra alors cliquer sur un premier *Button*, ce qui aura pour effet de révéler le contenu de la première case :

- soit il révèle **une case contenant une mine**, ce qui aura pour effet de lui faire perdre instantanément la partie ;
- soit il révèle **une case adjacente à au moins une mine**, ce qui fera apparaître un chiffre indiquant le nombre de mines adjacentes à cette case ;
- soit enfin il révèle **une case vide n'avoisinant pas de mine** : dans ce cas les case adjacentes sont également révélées et le processus se répète jusqu'à ce que la zone vide soit entièrement délimitée par des chiffres.

Le joueur continue ensuite de révéler des cases sur le même principe, jusqu'à ce que toutes les cases ne contenant pas de mine soient révélées, que le joueur révèle une mine ou que le temps soit écoulé. Le jeu notifie alors le joueur de l'issue de la partie, à savoir :

- le nombre de cases qu'il a réussi à révéler ;
- le temps qu'il a mis à révéler ces cases,

un joueur pouvant par exemple révéler 40 cases en 55 secondes, et ne rien jouer pendant le temps restant. Le jeu revient ensuite au formulaire principal après avoir mis à jour les statistiques de ce joueur.

D'un point de vue technique, le joueur interagira avec l'application via un ensemble de formulaires *WinForm*, tandis que le contrôle du jeu sera assuré par un *Module* contenant les données du jeu (la position des mines) et présentant un ensemble de fonctions permettant le déroulement du jeu (notamment pour jouer un coup, ou encore savoir si la case révélée contient ou avoisine une mine).

Au delà de l'aboutissement du développement, votre application sera également évaluée sur sa robustesse et son ergonomie.

## Prolongement du développement (10pts)

Une fois la base de l'application fonctionnelle, vous complétez votre développement en y intégrant d'autres fonctionnalités selon vos choix et vos envies. Par exemple :

- le *formulaire principal* pourra proposer un *Button* d'options, faisant apparaître un nouveau formulaire permettant de fixer les différents paramètres du jeu. Ce formulaire pourra proposer à l'utilisateur, via un ensemble de *Label*, *TextBox*, *RadioButton*, *CheckBox*, *ComboBox*, *ScrollBar*, etc., de modifier certains aspects du jeu tels que :
  - la taille de la grille de jeu ou le nombre de mines à localiser ;
  - la désactivation ou le réglage du temps alloué au joueur pour la résolution du jeu ;
  - le choix entre plusieurs thèmes visuels ;
  - le chemin d'accès au fichier pour la sauvegarde ;
- activer un *Button* permettant de faire une pause sur le formulaire de jeu. Lorsque le jeu sera en pause, le temps ne se décomptera plus, mais le joueur ne pourra plus cliquer sur les cases pour les révéler. Ce même bouton permettra la reprise du jeu ;
- etc.
- le jeu pourra faire appel aux outils développés dans le cadre du module d'*initiation au développement*, pour par exemple générer une grille à résoudre, ou encore proposer au joueur un prochain coup à jouer via un *Button* sur le formulaire de jeu.

Le choix et l'implémentation de ces fonctionnalités sont laissées libres aux développeurs, chaque apport sera évalué et permettra l'attribution de points. D'autres options peuvent également être envisageables, et conduiront également à l'attribution de points supplémentaires selon leur pertinence. Toutes les fonctionnalités implémentées de cette façon devront impérativement être documentées dans le rapport final.

## 2 Partie Saé : enregistrement des joueurs (10pts)

L'application que vous développerez devra mémoriser les noms des joueurs ayant joué au moins une partie, ainsi que leurs données de jeu. Pour cela, un *Module* devra garder en mémoire, pour chaque joueur :

- son nom ;
- le nombre de cases découvertes et le temps écoulé jusqu'à la dernière case révélée ou avant de découvrir une mine ;
- le nombre de parties jouées ;
- le cumul du temps de jeu pour ce joueur.

Un *Button* sur le formulaire d'accueil permettra d'afficher un formulaire présentant les données des joueurs ayant utilisé l'application :

- les données des différents joueurs seront présentées par *ListBox* synchronisées présentant leur nom ainsi que leur meilleur nombre de cases révélées et le temps associé ;
- l'affichage des joueurs dans ces *ListBox* sera trié selon le nombre de cases découvertes par ces joueurs (les ex-æquo seront départagés selon le temps mis à terminer la partie, ou écoulé jusqu'à la dernière case révélée ou avant de dévoiler une mine). Un *Button* permettra de basculer entre un ordre d'affichage croissant ou décroissant ;
- une *ComboBox* permettra de rechercher un joueur par son nom, puis d'afficher ses données détaillées (nom, meilleur nombre de cases révélées et temps associé, temps cumulé, nombre de parties) par *MsgBox* au clic sur un *Button*. La sélection d'un joueur par une des listes devra afficher son nom dans cette *ComboBox*.

De plus, toutes ces informations devront être **stockées dans un fichier**, puis automatiquement rechargées lors d'un prochain démarrage de l'application.

Cette partie de votre développement sera évaluée sur la base du strict respect du cahier des charges, et devra respecter les principes abordés en cours et séances TP tels que l'encapsulation des données ou encore le découplage entre modèle de données et interface utilisateur.

### Dossier à fournir :

Le rendu de votre projet se fera sous forme d'un dossier, à déposer sur **Moodle** au plus tard le **vendredi 20 mai**, et devra comporter les sections suivantes :

- une introduction présentant l'application et son utilisation illustrée par des captures d'écran de vos formulaires ;
- un schéma d'ordonnancement des formulaires, présentant les liens entre les formulaires et les événements permettant le passage de l'un à l'autre ;
- une documentation des fonctionnalités abouties et prolongements implémentés ; en particulier vous détaillerez la procédure de mise en place du jeu ainsi que les interactions avec les différents modules ;
- une conclusion présentant les pistes d'amélioration de votre projet.

Ce dossier sera à déposer sur **Moodle**, dans une archive *zip* contenant également votre **projet Visual Studio**.

