

Projet Java : Démineur

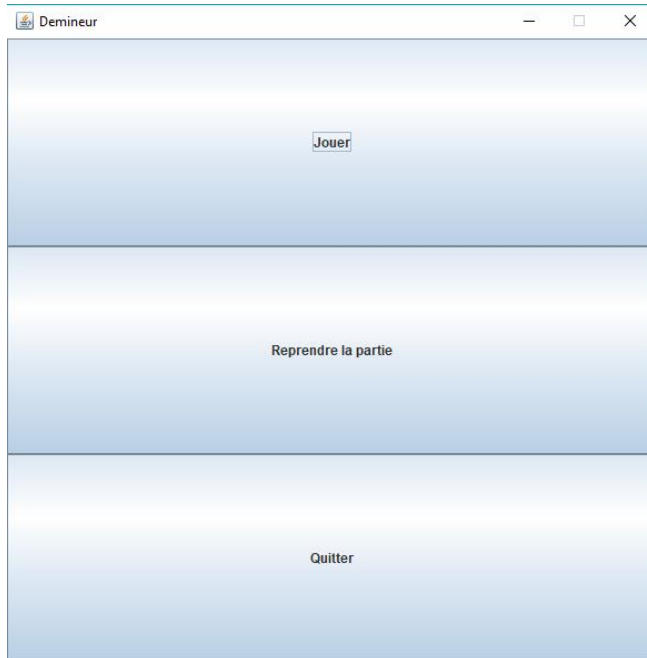
Rapport

Introduction :

Le démineur est un jeu de réflexion dont le but est de localiser des mines cachées dans un champ virtuel avec pour seule indication le nombre de mines dans les zones adjacentes. Le but de ce projet était de réaliser ce jeu en langage Java.

Les différentes fonctionnalités de notre programme :

Lorsque le programme est lancé, l'utilisateur arrive sur un menu avec 3 choix possibles :

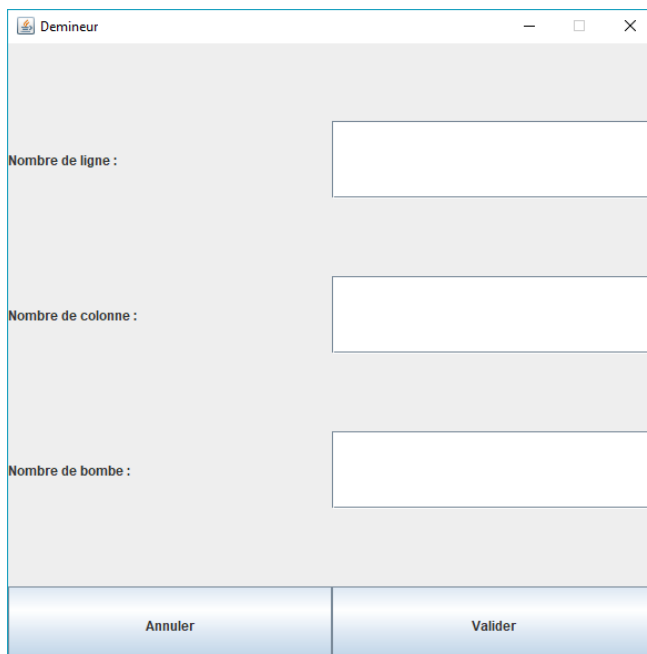


L'utilisateur peut choisir de jouer en cliquant sur le bouton « Jouer ».

Il peut également choisir de reprendre une partie, s'il en a sauvegardé une par le passé en cliquant sur le bouton « Reprendre la partie ».

Enfin il peut choisir de quitter le programme en cliquant sur le bouton « Quitter ».

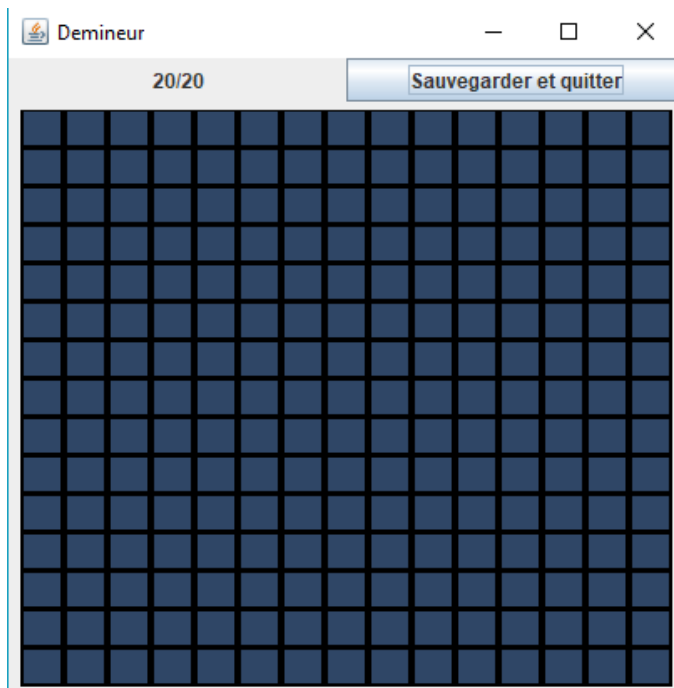
S'il choisit de jouer, il arrivera sur le menu ci-dessous :



Ce menu va lui permettre de choisir la configuration (nombre de ligne, de colonne et de bombe) avec laquelle il souhaite jouer. Il pourra valider ses choix grâce au bouton « Valider ».

Il aura également la possibilité de revenir au menu principal grâce au bouton « Annuler ».

En cliquant sur valider, il arrivera sur la fenêtre ci-dessous :

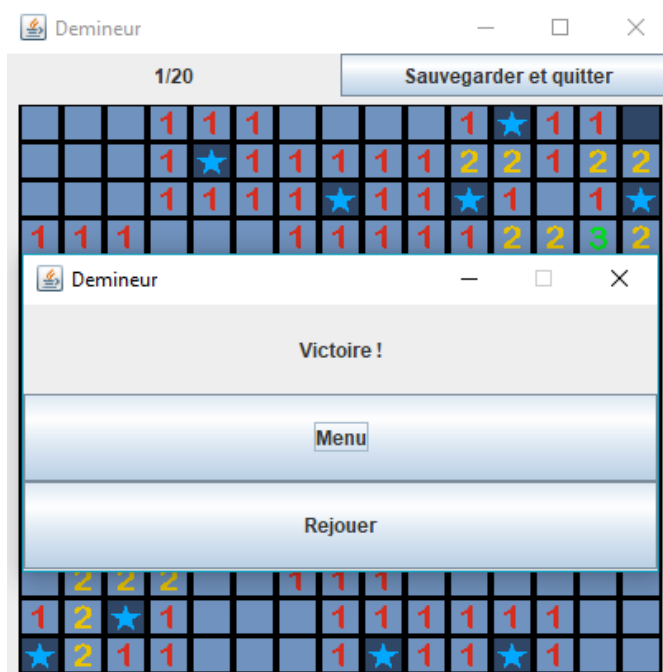


Cette fenêtre est la fenêtre de jeu. Elle aura la taille et le nombre de bombe que l'utilisateur a choisis (ici nous avons une grille 15x15 avec 20 bombes).

Sur cette fenêtre il va pouvoir jouer au jeu du démineur en cliquant sur les cases à l'aide de sa souris.

Il pourra également choisir de sauvegarder sa partie et de revenir au menu principal grâce au bouton « Sauvegarder et quitter ». Il pourra également effectuer cette action en cliquant sur la croix en haut à droite de la fenêtre.

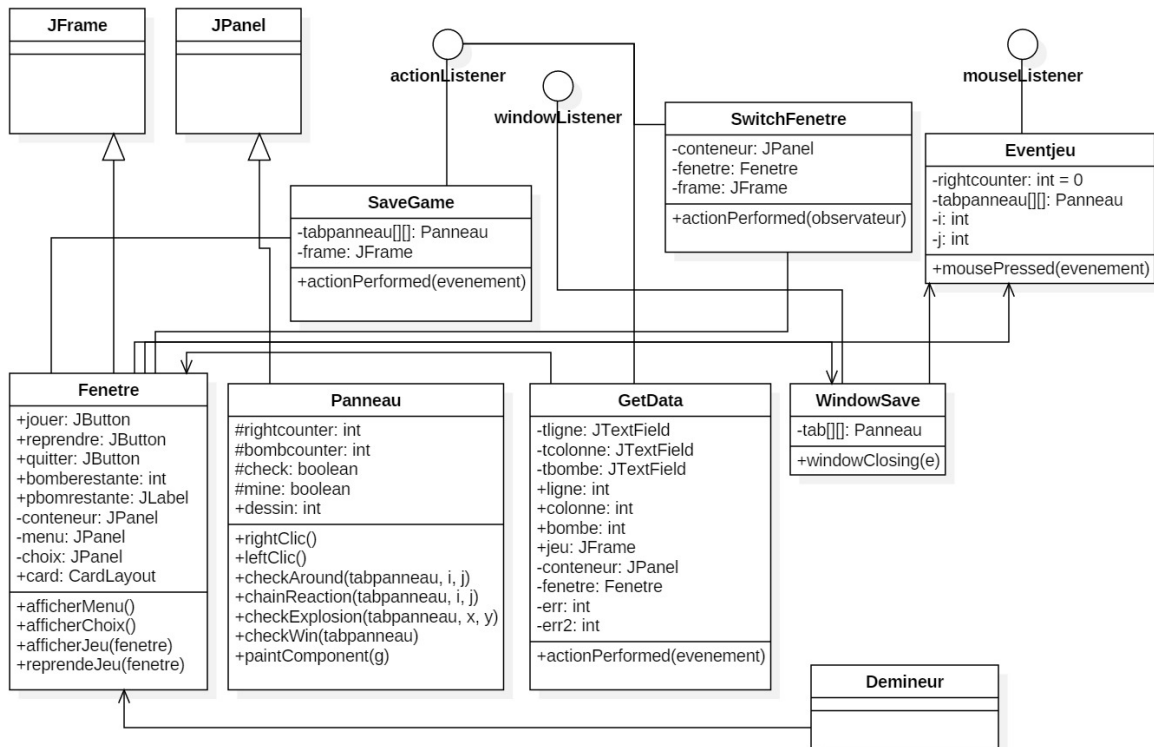
Une fois la partie terminée (que ce soit une victoire ou une défaite) l'utilisateur aura 2 choix possibles :



Soit l'utilisateur décide de retourner au menu principal, Il cliquera alors sur le bouton « menu ».

Soit il décide de rejouer avec la même configuration. Dans ce cas il lui suffira de cliquer sur le bouton « Rejouer ».

Présentation de la structure du programme :



Explication du mécanisme de sauvegarde :

Pour sauvegarder une partie nous procédons de la façon suivante :

- Ouverture du fichier dans lequel les données vont être écrites
- Ecriture des variables contenant le nombre de ligne, colonne, bombe et de bombes restantes dans ce fichier
- Début d'une boucle dans laquelle nous parcourons le tableau de panneau qui est utilisé pour représenter les cases du jeu
- Ecriture des caractéristiques actuelles de chaque panneau (nombre de clic droit, nombre de bombe alentour, miné ou non, découvert ou non et enfin quel image est située dessus)
- Fermeture du fichier

Exposition de l'algorithme qui permet de révéler plusieurs cases :

Voici le code qui nous permet de révéler plusieurs cases lors d'un clic :

```
public void chainReaction(Panneau[][] tabpanneau, int i, int j){
    if(tabpanneau[i][j].mine==false){
        tabpanneau[i][j].checkAround(tabpanneau, i, j);
        if(tabpanneau[i][j].bombcounter==0){
            for(int i2 = i - 1; i2< i + 2; i2++){
                for(int j2 = j - 1; j2< j + 2; j2++){
                    if(i2>=0 && i2<=GetData.ligne -1 && j2>=0 && j2<=GetData.colonne -1){
                        if(tabpanneau[i2][j2].check==false){
                            tabpanneau[i2][j2].chainReaction(tabpanneau, i2, j2);
                        }
                    }
                }
            }
        }
    }
}
```

Concrètement, lorsque l'utilisateur clique sur une case, la fonction chainReaction vérifie si la case contient une bombe. Si ce n'est pas le cas, la fonction checkAround est appelée. Elle permet de vérifier si les 8 cases autour de celle sur laquelle l'utilisateur a cliqué comportent une mine. Si ce n'est toujours pas le cas la fonction chainReaction rappelle checkAround. Et cela jusqu'à ce qu'une case contienne une bombe.

Conclusion personnelle :

Adrien : J'ai trouvé ce projet très enrichissant et très intéressant. J'ai pris beaucoup de plaisir à le réaliser et j'ai appris énormément de chose : utiliser un tableau de panneau, utiliser la méthode GetText(), faire un makefile en java. Je suis très satisfait du travail que nous avons réalisé, le jeu est parfaitement fonctionnel et assez agréable esthétiquement parlant.

Grâce à ce projet je me suis également intéressé au jeu du démineur. En effet, je ne me suis jamais réellement penché sur ce jeu qui est pourtant très addictif et bien plus stratégique que ce que je ne pensais auparavant. Je finirai par dire que je suis très heureux d'avoir réalisé ce projet !

Alexandre : J'ai beaucoup aimé faire ce projet, qui était beaucoup plus intéressant que le précédent. Le Java permet plus de liberté que le C, mais reste, à mon avis, plus compliqué. Le jeu du démineur est plus complexe et amusant que le taquin, j'étais donc plus motivé pour finir le travail. De plus, ce projet a grandement augmenté mes compétences en Java, et m'a fait découvrir le potentiel qu'a ce langage. Je suis plutôt satisfait de notre travail global, mis à part l'apparence de base du java dans le menu, que l'on a laissé par manque de temps, et les quelques bug que nous n'avons pas réussi à fixer.