



Université de la Manouba
Ecole Nationale Des Sciences De L'Informatique

SUJET :
Rapport du jeu Démineur

par :
Maalej Yessin
Ben Ahmar Houcem
Frikha Bilel
Baklouti Mohammed

Ralisé au sein de :
L'Ecole Nationale Des Sciences De L'Informatique

Sous l'encadrement de :
Mr. Majdi Jribi

Table des matières

Introduction gènèrale	1
1 Présentation	2
1.1 Historique	2
1.2 Réglement	2
2 Spécification des besoins	4
2.1 Besoins fonctionnels	4
2.2 Besoins non fonctionnels	4
3 Conception	6
3.1 Diagramme des fonctions	6
3.2 Choix Niveau	7
3.3 Generer (entier nbre)	7
3.4 Remplir Bombe(cellule** grille,entier nbre)	7
3.5 Remplir (cellule** grille)	7
3.6 Jeu (cellule** grille)	7
4 Réalisation	9
4.1 Le mode console	9
4.1.1 choix du niveau du jeu	9
4.1.2 Initialisation de la grille	9
4.1.3 Remplissage de la grille	10
4.1.4 Generation de la grille	11
4.2 Le mode graphique	11
Conclusion	22

Introduction générale

Les Jeux de réflexion s'adressent à presque tous les publics, et on en trouve pour tous les âges et tous les goûts. Le niveau de difficulté de ces jeux est ajustable ce qui fait que débutants comme joueurs confirmés y trouvent un plaisir réel et passionné. Les meilleurs dans lesquels le joueur doit faire marcher à fond tous ces neurones pour arriver à solutionner les problèmes que vous aller rencontrer. D'autres aptitudes complémentaires peuvent et sont souvent mises à profit comme le sens de l'observation, la logique, le sens de l'orientation, et bien d'autres encore. Dans ce cadre là, nous avons jugé utile de réaliser une jeu de réflexion tres populaire qui est " Le démineur " Pour mettre en relief les grands axes de notre projet, ce rapport a été divisé en quatre chapitre : Dans le premier chapitre, un historique du jeu ainsi que ses règles seront présentés. Le deuxième partie sera consacré à la spécification des besoins. Le troisièmesera dédié pour l'étude conceptuelle de notre projet. Dans le dernier, les différents étapes de réalisation seront expliquées.

Chapitre 1

Présentation

Introduction

Dans cette partie, nous donnerons dans une première étape une présentation du jeu de point de vue historique et dans une seconde étape les règles du jeu sont mentionnés.

1.1 Historique

Démineur a ses origines dans les premiers jeux de mainframe des années 1960 et 1970. Le premier ancêtre de démineur était Cube Jerimac Ratliff. Le style de jeu de base est devenu un segment populaire du genre jeu de puzzle au cours des années 1980, avec des titres tels que Miné-Out (Quicksilver, 1983), Yomp (Virgin Interactive, 1983), et le Cube. Cube a été remplacé par logique implacable (ou RLogic pour faire court), par Conway, Hong, et Smith, disponible pour MS-DOS, dès 1985, RLogic eu une plus grande similitude avec Démineur que de Cube dans le concept.

1.2 Règlement

Le démineur est un jeu de réflexion dont le principe est de localiser des mines cachées dans une grille à deux dimensions en se basant sur des chiffres indiquant le nombre des mines dans les cases adjacentes. À chaque étape du jeu, le joueur voit un ensemble de cases rassemblées en un rectangle. Chacune d'entre elles peut, ou bien cacher une mine, ou bien être libre. Si elle est libre et que chacune des cases adjacentes l'est également, la case reste vide. Si en revanche au moins l'une des huit cases avoisinantes cache

une mine, un chiffre indique dans la case meme le nombre de cases voisines cachant des mines. Un joueur peut signaler toutes les mines présumées par un drapeau. En comparant différentes informations, on peut ainsi progresser dans le déminage du terrain. Le joueur perd la partie s'il selectionne une case contenant une mine. Il gagne la partie s'il découvre toutes les cases ne contenant pas de mines.

Conclusion

Le principe du jeu " démineur " s'avère intéressant et nous inspire des enrichissements que nous préciserons dans le seconde chapitre.

Chapitre 2

Spécification des besoins

Introduction

Les outils informatiques nous offrent une grande flexibilité pour la réalisation des besoins fonctionnels aussi bien que les Besoins non fonctionnels que nous détaillerons dans ce chapitre.

2.1 Besoins fonctionnels

- Le système doit permettre la génération d'une grille initiale avec un nombre de case préalablement choisie par l'utilisateur.
- Le système doit permettre au joueur la possibilité de choisir le niveau de jeu qui le convient (Facile, Moyen, Difficile)
- Le système doit permettre à l'utilisateur la sélection d'une case de la grille.
- Le système doit permettre l'affichage du contenu d'une case sélectionné et de ces voisins.
- Le système doit afficher au joueur son résultat (échec ou gagnant) ainsi que son score.

2.2 Besoins non fonctionnels

- **La rapidité :** Il est exigeant d'avoir une application en temps réel pour assurer son bon rendement.
- **La convivialité :** Procurer des interfaces conviviales et simples à utiliser.

- **La maintenance** : Le code de l'application doit être lisible et compréhensible pour pouvoir la maintenir facilement et rapidement de façon à faciliter sa mise à jour et l'amélioration de certains détails ainsi que l'ajout d'autres fonctions.
- **L'ergonomie** : L'interface doit être agréable et facile à manipuler.

Conclusion

L'analyse des besoins est une étape primordiale pour délimiter les principaux objectifs du projet, qui nous facilitera la réalisation des différentes conceptions que nous aborderons dans le troisième chapitre.

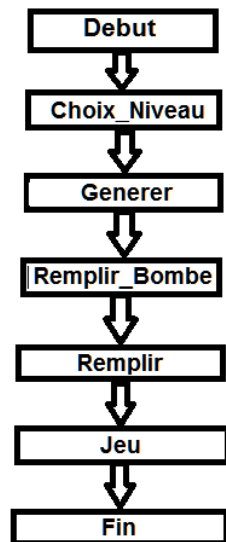
Chapitre 3

Conception

Introduction

Avant de commencer la phase de la programmation le developpeur doit passer par la phase de la conception pour trouver les idées qui lui permettent de satisfaire les besoins déjà spécifiés.

3.1 Diagramme des fonctions



3.2 Choix Niveau

Cette fonction ne possède aucun argument et elle retourne un variable "nbre" de type entier qui indique le niveau du jeu choisie par l'utilisateur en saisissant la valeur qui sera affecté a cette variable.

S'il saisie 1 alors il a choisi le niveau facile .

S'il saisie 2 alors il a choisi le niveau moyen .

S'il saisie 3 alors il a choisi le niveau difficile.

3.3 Generer (entier nbre)

Puisque pour chaque niveau la taille de la grille change, nous avons simulée la gille a une matrice carré dynamique de taille variable selon la valeur de "nbre" donnée en argument. La case de la grille est une structure de donnée appelée "cellule", elle contient une variable "val" de type entier qui renseigne sur le contenu de la cellule est une variable booleene "select" qui indique si la case est déjà saisie par le joueur ou non. Plus le niveau du difficulté du jeu augmente plus le nombre de cellule augmente.

3.4 Remplir Bombe(cellule** grille,entier nbre)

Selon la valeur affectée a la variable "nbre" le nombre de bombe contenu dans la grille sera : Si (nbre=1) alors le nombre de bombe est 16. Si (nbre=2) alors le nombre de bombe est 40. Si (nbre=3) alors le nombre de bombe est 99. Le remplissage de la grille par les bombes est effectué d'une manière aléatoire.

3.5 Remplir (cellule** grille)

Une boucle se déclenche pour parcourir la grille case par case.Si la case est une bombe alors la variable "val" de toutes les cases voisines á la case parcourue et qui ne contiennent pas une bombe seront incrémentées de 1.

3.6 Jeu (cellule** grille)

Pour accéder au contenu de la case il suffit que le joueur tape les coordonnées de la case choisie en donnant le numéro de la ligne et de la colonne.

Dans ce cas variable "select" de la case reçoit "true" pour empêcher une resaisie de la cellule.

Chapitre 4

Réalisation

Introduction

Dans ce chapitre nous allons décrire comment nous avons réalisé le jeu . Nous allons dans une première étape donner le déroulé suivi pour réaliser le jeu en mode console puis dans une seconde étape nous allons donner le déroulé suivi pour réaliser le jeu en mode graphique.

4.1 Le mode console

4.1.1 choix du niveau du jeu

Nous avons choisi de réaliser le jeu avec 3 niveaux :

- **Facile** : La grille contient 81 cases (9 lignes et 9 colonnes) dans lesquelles existent 10 bombes.
- **Moyen** : La grille contient 256 cases (16 lignes et 16 colonnes) dans lesquelles existent 40 bombes
- **Difficile** : La grille contient 900 cases (30 lignes et 30 colonnes) dans lesquelles existent 99 bombes. Nous allons afficher au joueur ces 3 niveaux du jeu numérotés de 1 jusqu'à 3 et en saisissant le numéro du niveau voulu son choix sera effectué.

4.1.2 Initialisation de la grille

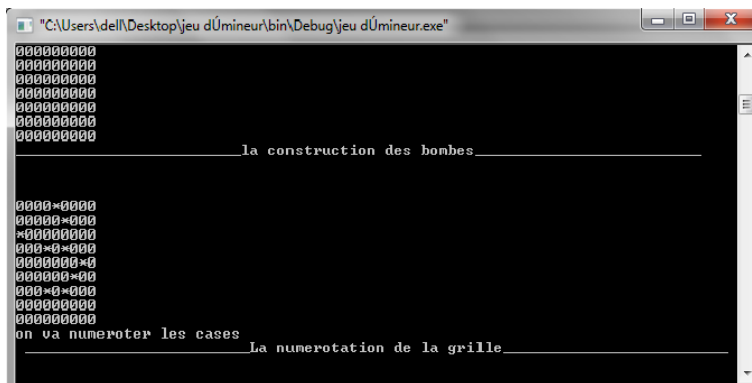
La grille est une matrice allouée dynamiquement à deux dimensions, de tailles différentes selon le niveau du jeu choisi par le joueur. Les cases sont définies sous la forme d'une structure de données appelée "cellule" composée

d'une variable "valeur" de type entier et une variable "select" de type boolean qui a pour rôle de vérifier si la case est choisie ou non par le joueur. Initialement, la variable "valeur" de chaque case est initialisée à zéro et la variable "select" est initialisée à faux .



4.1.3 Remplissage de la grille

La remplissage des bombes se fait d'une manière aléatoire. La numérotation des cases restantes se fait selon la présence de bombes dans les cases voisines.



```

"C:\Users\dell\Desktop\jeu d'Umineur\bin\Debug\jeu d'Umineur.exe"
00000000*0
00000000*00
000*0*000
0000000000
0000000000
on va numeroter les cases
La numerotation de la grille

resultat
0001*2100
11012*100
*11132200
111*2*211
0011223*1
001122*21
001*2*210
001121100
000000000
  
```

4.1.4 Generation de la grille

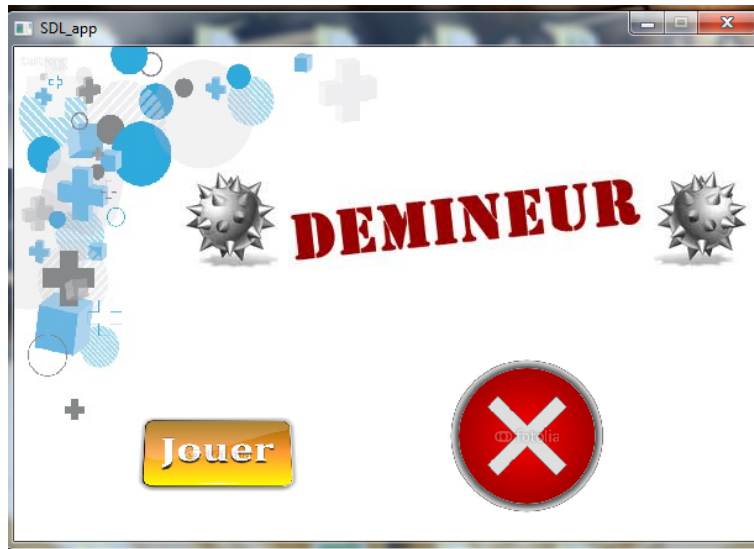
Enfin, les cases de la grille finale ne sont pas visibles pour le joueur puisque a l'affichage chaque case est representée par double barre.

```

"C:\Users\dell\Desktop\jeu d'Umineur\bin\Debug\jeu d'Umineur.exe"
11012*100
*11132200
111*2*211
0011223*1
001122*21
001*2*210
001121100
000000000
  
```

4.2 Le mode graphique

En fait on a amélioré le jeu pourqu'elle soit en mode graphique. Cette fenetre vous donne le choix de jouer ou quitter.



Cette fenêtre correspond à la page d'accueil du jeu. Le joueur a le choix de jouer ou bien de quitter.






: Ce bouton permet au joueur d'accéder au jeu.

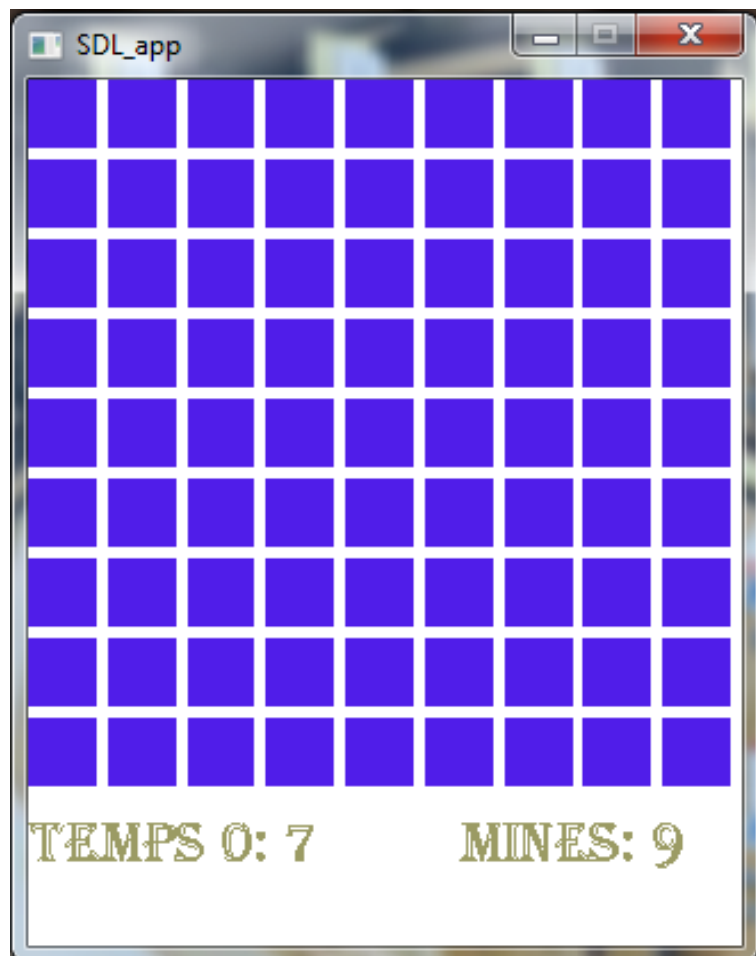


: Ce bouton permet au joueur de quitter le jeu.

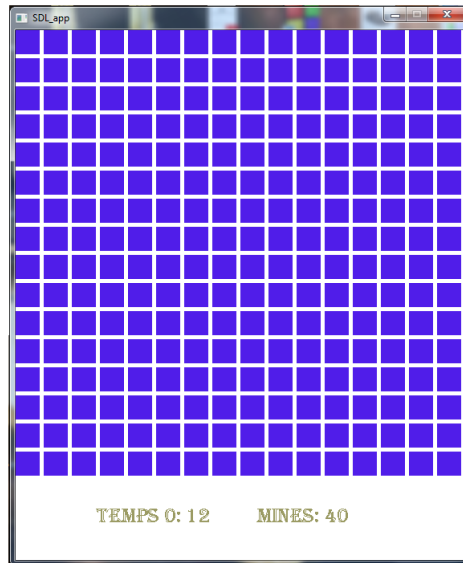


Cette fenêtre permet au joueur de choisir le niveau du jeu.

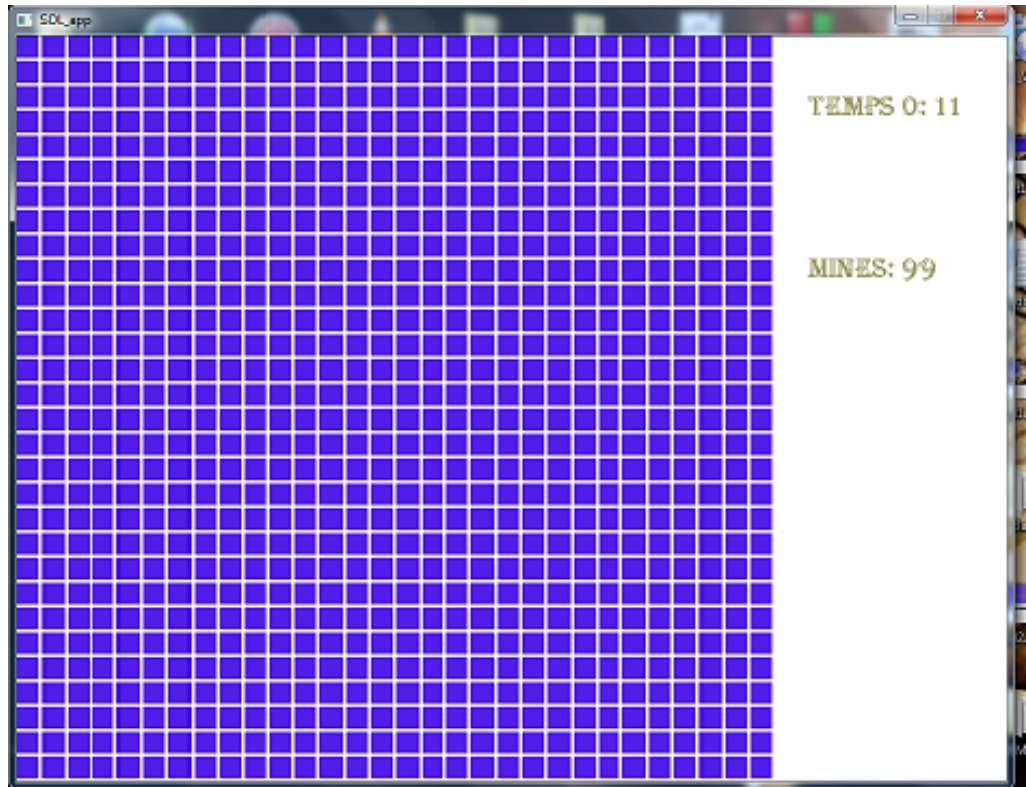
- | | |
|---|--|
|  | :Ce bouton permet au joueur de jouer le niveau facile du jeu. |
|  | :Ce bouton permet au joueur de jouer le niveau moyen du jeu. |
|  | :Ce bouton permet au joueur de jouer le niveau difficile du jeu. |



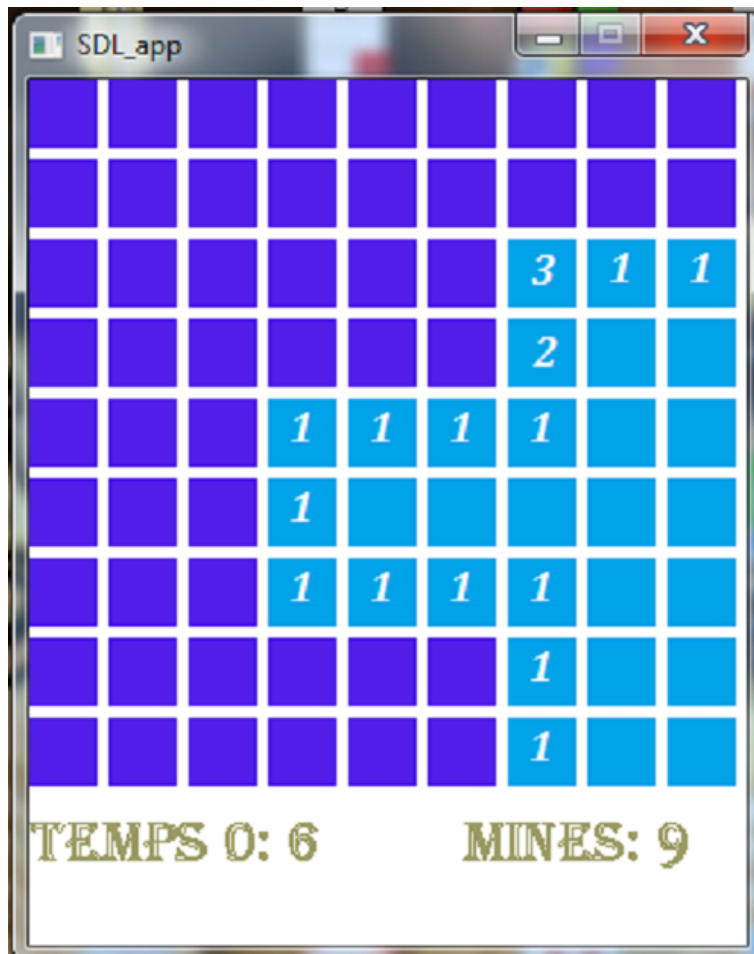
Losque le joueur choisie le niveau facile cette grille apparait.



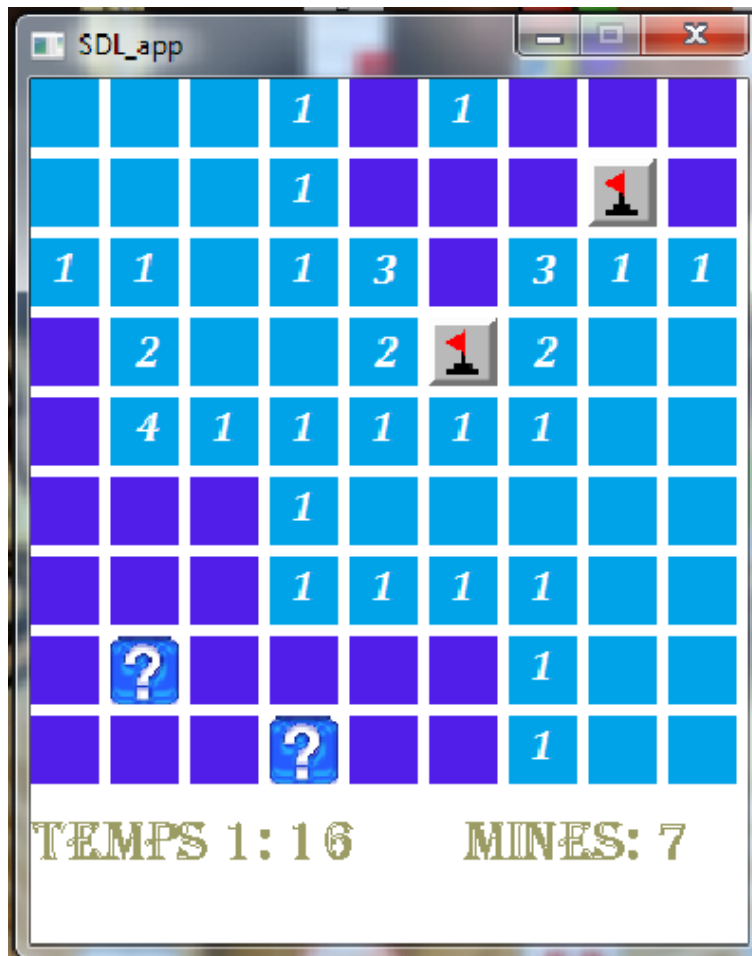
Losque le joueur choisie le niveau moyen cette grille apparait.



Losque le joueur choisie le niveau diffacile cette grille apparait.

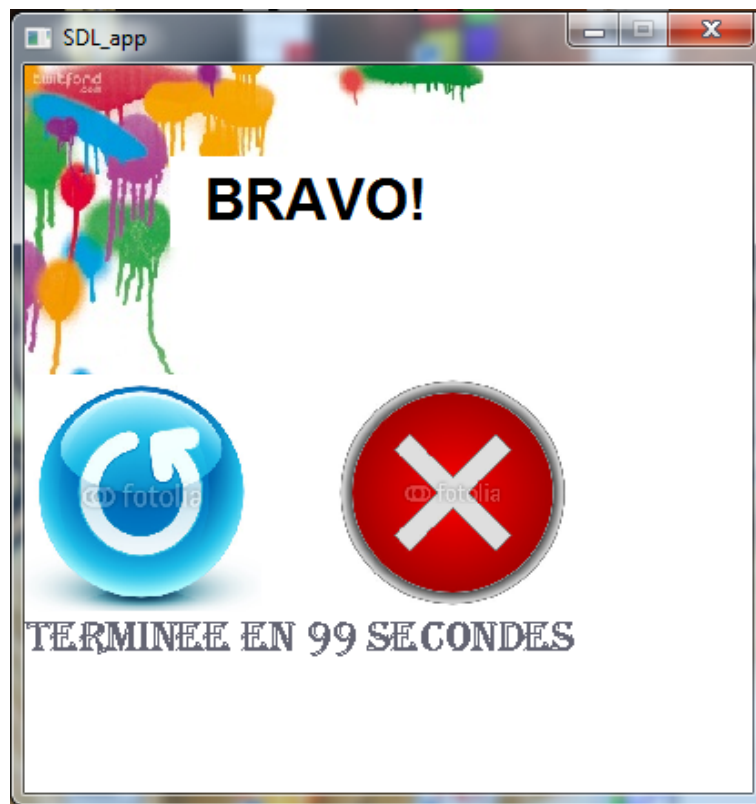


Nous allons prendre par exemple le niveau facile.
 Le joueur doit choisir une case en cliquant sur elle par le curseur gauche de la souris il sera mieux de choisir une case de frontière, l'ouverture de cette case engendre l'ouverture de tous les cases voisines vides.












Le joueur peut marquer les cases qu'il devine qu'elle contiennent une bombes en cliquant sur le bouton droit.

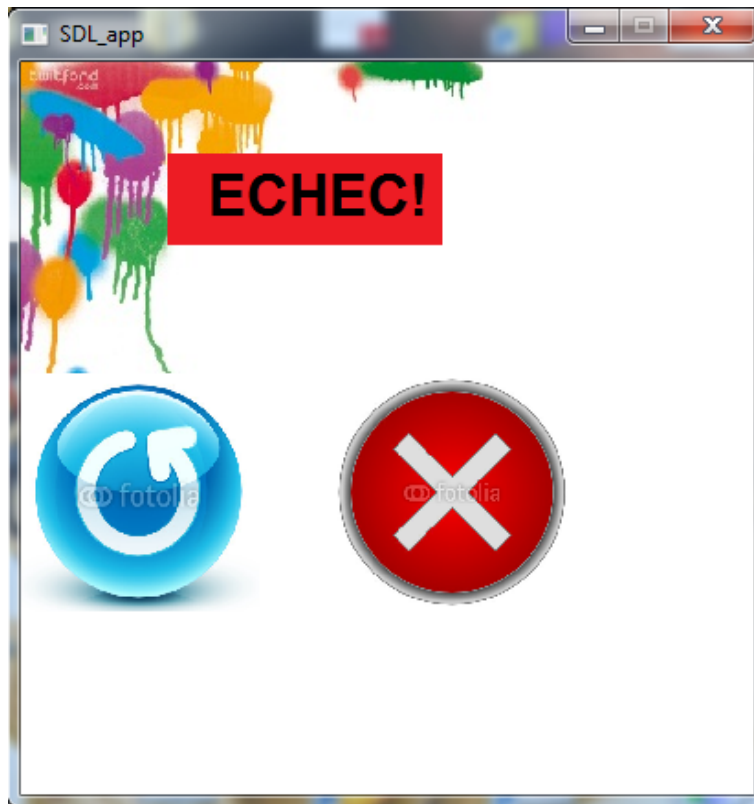
Le joueur peut marquer aussi les cases qui lui semble inconnues en cliquant 2 fois consécutives sur le bouton droit.



Cette fenêtre apparaît lors du succès qui correspond à la découverte de tout les bombes.

1	1	1		1	1	2		1
	1	2	2	3	2		2	1
1	1	1		2		2	1	
		1	1	2	1	1		
1	1	2	1	1				
1		2		2	1	1		
1	1	2	1	2		1		
				1	1	1		
TEMPS 0: 9				MINES: 9				

Si le joueur choisit une case contenant une bombe alors le contenu de toutes les cases de la grille seront affich .



Cette fenêtre apparait lors de l'échec.



: Ce bouton permet au joueur de rejouer le jeu.



: Ce bouton permet au joueur de quitter le jeu.

Conclusion

Au-delà de son public traditionnel, le jeu vidéo touche aujourd'hui les hommes comme les femmes, les enfants comme les adultes. Il devient une véritable passerelle entre les générations.

Ce projet a pour nous été l'occasion de passer de l'autre côté du rideau du monde des jeux vidéos. Cette expérience s'est avérée enrichissante sur tous les plans. Aussi bien au niveau Technique où nous avons pu parfaire nos connaissances en algorithmique , améliorer notre pratique du C++ et découvrir la richesse de OpenGL qu'au niveau humain où nous avons acquis un esprit collectif de travail .