

Pham Alexandre

Barbesier Axel

Documentation du Projet Skulls'n Bones

Sommaire :

- 1.A Propos
2. Répartition des rôles
- 3.Technologies utilisées
- 4.Fonctionnalité
- 5.Structure du Code

1. A Propos

Skulls 'n Bones est un jeu de Type Slay them All (le but est de tuer/détruire le plus grand nombre d'ennemis) par vagues (un certain nombre de zombies apparaissent par vague, quand ils meurent tous, une nouvelle vague apparaît) dans lequel les statistiques des parties sont envoyées et enregistrées sur une base de données et sont disponibles sur un site permettant

aux joueurs de comparer leurs performances entre eux et d'établir un classement des meilleurs joueurs

2. Répartition des rôles

Axel Barbesier s'est occupé du site et Alexandre Pham s'est occupé du jeu, nous avons aussi fait certaines tâches en commun tel qu'affiché dans le tableau :

GROUPE : Projet Dev		Grille	Tableau	Graphiques	Planning	...
Titre	Affection	Date de début	Date d'échéance	Compartiment	Progression	Priorité
Gestion du Joueur et des ennemis	BA		30/4/2024	Terminé	Terminée	! Important
Gestion des inputs	BA	4/1/2024	20/1/2024	Terminé	Terminée	! Important
Implémenter l'A ennemie	BARBESIER Axel	4/1/2024	28/1/2024	Terminé	Terminée	! Important
Ajout de projectiles	PHAM Alexandre	28/1/2024	28/1/2024	Terminé	Terminée	Moyen
Etablir Le CRUD	BA	4/1/2024	30/4/2024	Terminé	Terminée	! Important
Ajouter gameplay	BA	24/2/2024	30/5/2024	Terminé	Terminée	↓ Minimum
Configurer hébergement BDD/jeu	PHAM Alexandre	4/1/2024	22/2/2024	Terminé	Terminée	! Important
Page HTML/CSS	BARBESIER Axel			Terminé	Terminée	! Important
Afficher les infos de la BDD sur le site	BARBESIER Axel	4/1/2024	21/3/2024	Terminé	Terminée	! Important
Gestion d'entrée pseudo	PHAM Alexandre	4/1/2024	8/3/2024	Terminé	Terminée	! Important
Base du jeu	PHAM Alexandre			Terminé	Terminée	! Important
Liaison jeu -> BDD	PHAM Alexandre			Terminé	Terminée	! Important
Config InfinityFree	BA			Terminé	Terminée	! Important
Créer mode de jeu (vagues d'ennemis)	BA	7/3/2024	31/3/2024	Terminé	Terminée	↓ Minimum

3. Technologies Utilisées

Pour mener à bien le projet à bien nous avons utilisé **Window Form Application** comme moteur de jeu et **C#** comme langage et **Free BDD** comme hébergeur de BDD, **InfinityFree** pour héberger notre site et PHP pour faire la communication BDD/Site.

4.Fonctionnalités

Jeu

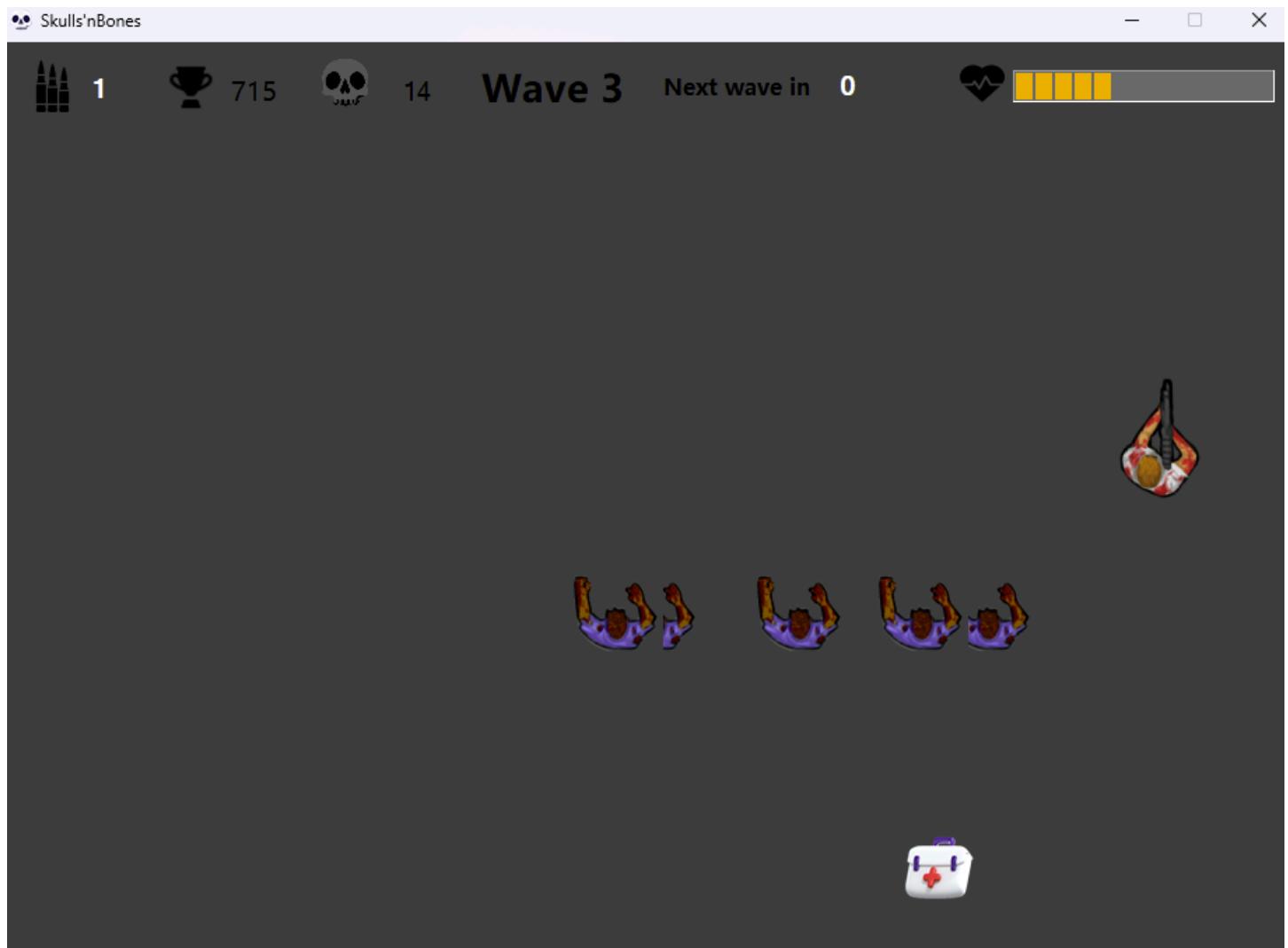
Système de vague

Le jeu dispose d'un système de "vague" : à chaque vague un nombre limité de zombie apparaissent, quand les zombies sont tous éliminés, un nouvelle vague se lance avec une plus importante de zombies



Système de Soin

le joueur peut se soigner en récupérant des trousse de soins lâchées par les zombies quand ils sont tués (de manière aléatoire), ce trousse permet de regagner l'intégralité de sa santé



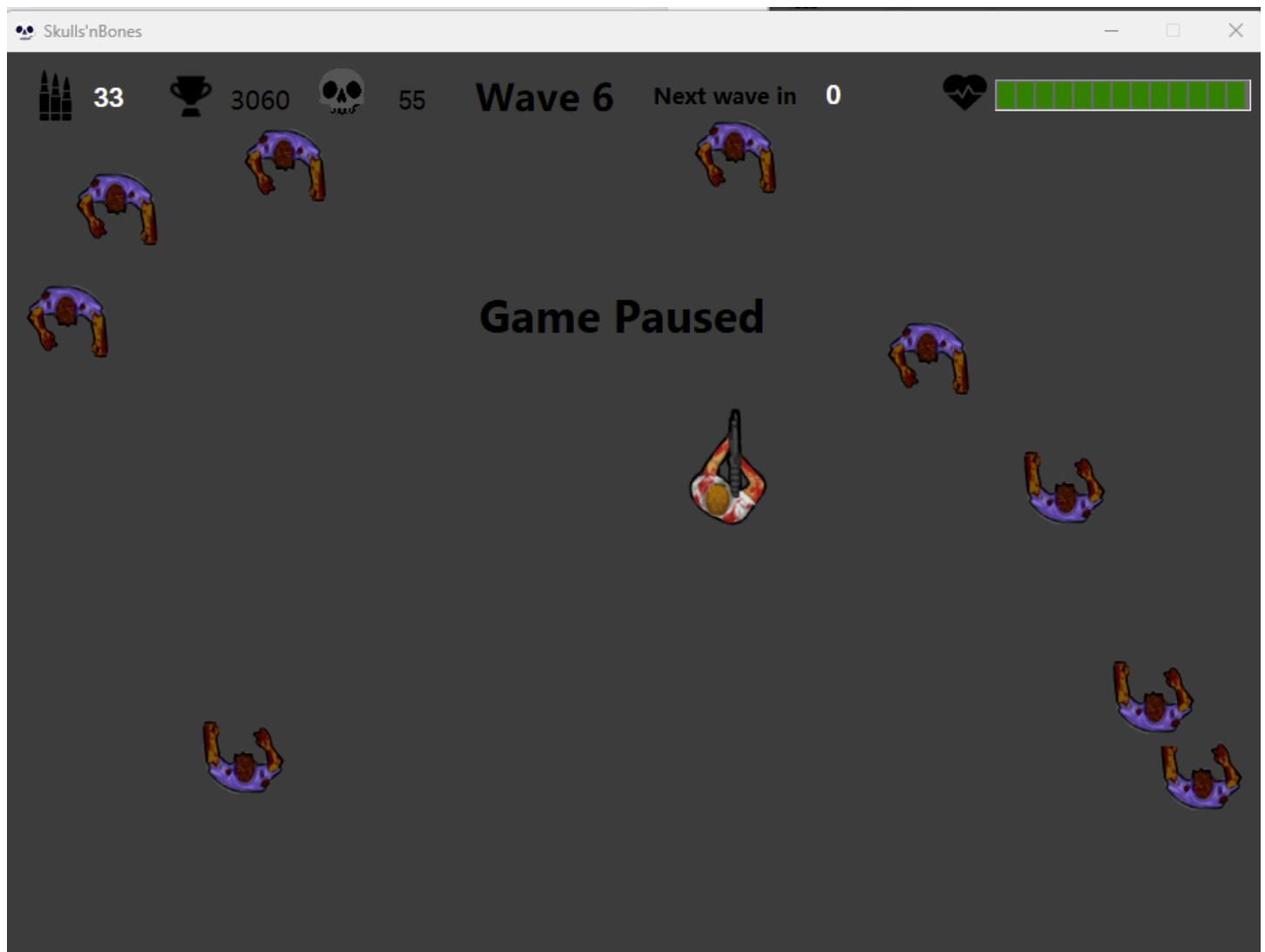
Obtention de munition

Les zombie peuvent lâcher des munitions à leurs morts et des munitions apparaissent aléatoirement en jeu lors que le joueur n'a plus de balles



Système de Pause

Le joueur peut mettre en pause la partie (sauf entre les vagues) en appuyant sur P et la reprendre en rappuyant sur la touche



Système de Suivi des Zombies ("AI")

Les Zombies suivent le joueur et lui font des dégâts quand il le touchent ,ils adaptent leurs trajectoire selon celle du joueur

33



3060



55

Wave 6

Next wave in 0



Game Paused



Site

Statistique de chaque partie du jeu disponibles

Les parties du jeu sont enregistrées et sont disponibles sur notre site, il est possible de trier les données par ordre croissant et décroissant selon les catégories

The screenshot shows the 'Skulls'n Bones official Site' Hall of Fame page. At the top, there's a navigation bar with a skull icon, 'About', and 'Hall of Fame'. Below the header, a banner reads 'Welcome to the official site of Skulls'n Bones, here you can see your score and the scores of your friends'. A warning message states '⚠ Only the scores of the wave mode are registered'. A note below says 'use the middle of the mouse to scroll down the score table'. The main content is a table with columns: Username, Score, Kills, and Game_Id. The table lists 20 entries from highest to lowest score.

Username	Score	Kills	Game_Id
barboz	3985	72	1
barboz	1930	35	2
barboz	2105	38	3
barboz	5070	91	4
barboz	7470	134	5
barboz	2710	48	6
barboz	250	5	7
Skyroez552	1150	12	8
Skyroez550	1150	12	9
Skyroez552	1250	23	10
skydefender235	1160	21	11
ynov	1110	20	12
test505	2470	44	13
barbade414	565	11	14
cata545	0	0	15
eee	920	16	16
bingo250	945	18	17
red	4310	78	18
red	2480	45	19
red	2520	45	20

Classement des meilleurs joueurs

Sur les parties enregistrée, le site cherche et charge les dix meilleurs joueur (par le score) du jeux et les affiche dans la rubrique "Hall of Fame", il est aussi possible de trier les données de manière croissante et décroissante selon les catégories



Skulls'n Bones Best Players

In the tumult of zombie horrors, unsurpassed heroes emerge, masters of survival and extermination. Their names resonate like legends, their exploits defy the impossible. With unparalleled skill, they navigate the hordes of the undead, sowing chaos and destruction in their wake.

This leaderboard is for the wave mode only

Place	Username	Score	Kills	Avg Kills
1	Razar <small>FIRST</small>	28175	329	168.0000
2	LEXA <small>SECOND</small>	25305	453	280.5000
3	defaced <small>THIRD</small>	22385	276	28.4444
4	rtyrtuy <small>TOP 10</small>	13050	235	235.0000
5	frgrtd <small>TOP 10</small>	12910	233	233.0000
6	red <small>TOP 10</small>	11190	198	58.6400
7	ILEXAI <small>TOP 10</small>	10775	201	201.0000
8	zerezet <small>TOP 10</small>	10385	190	190.0000
9	TimyTheTester <small>TOP 10</small>	10015	181	108.3333
10	installedonmmpc <small>TOP 10</small>	9300	169	169.0000

Jeu téléchargeable depuis le site

Le jeu peut être téléchargé et installé depuis notre site

Play The Game

You can install the game from here, no need to install a game with a huge size, our game is light and ready to play

download

estimated size 6.5mb

5. Structure du code

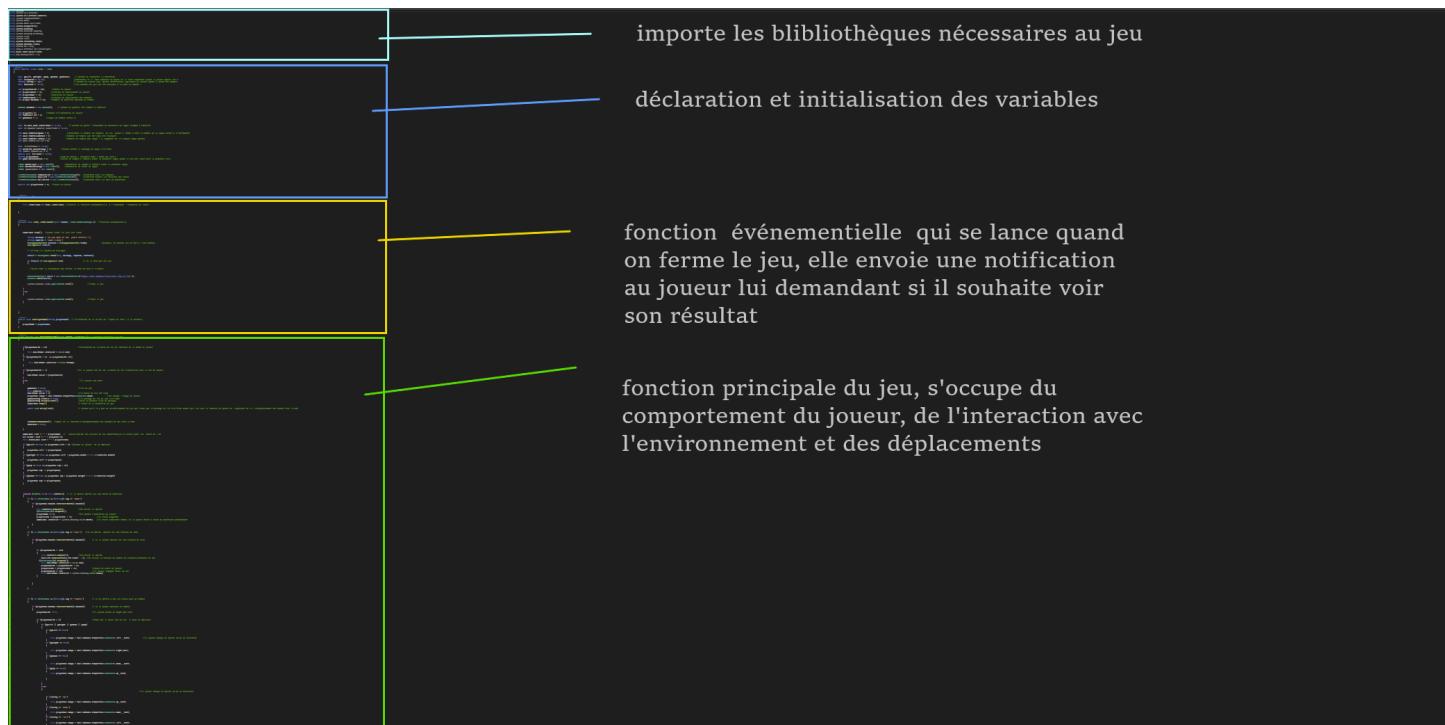
le code du jeu est organisé autour des fichiers principaux suivants :

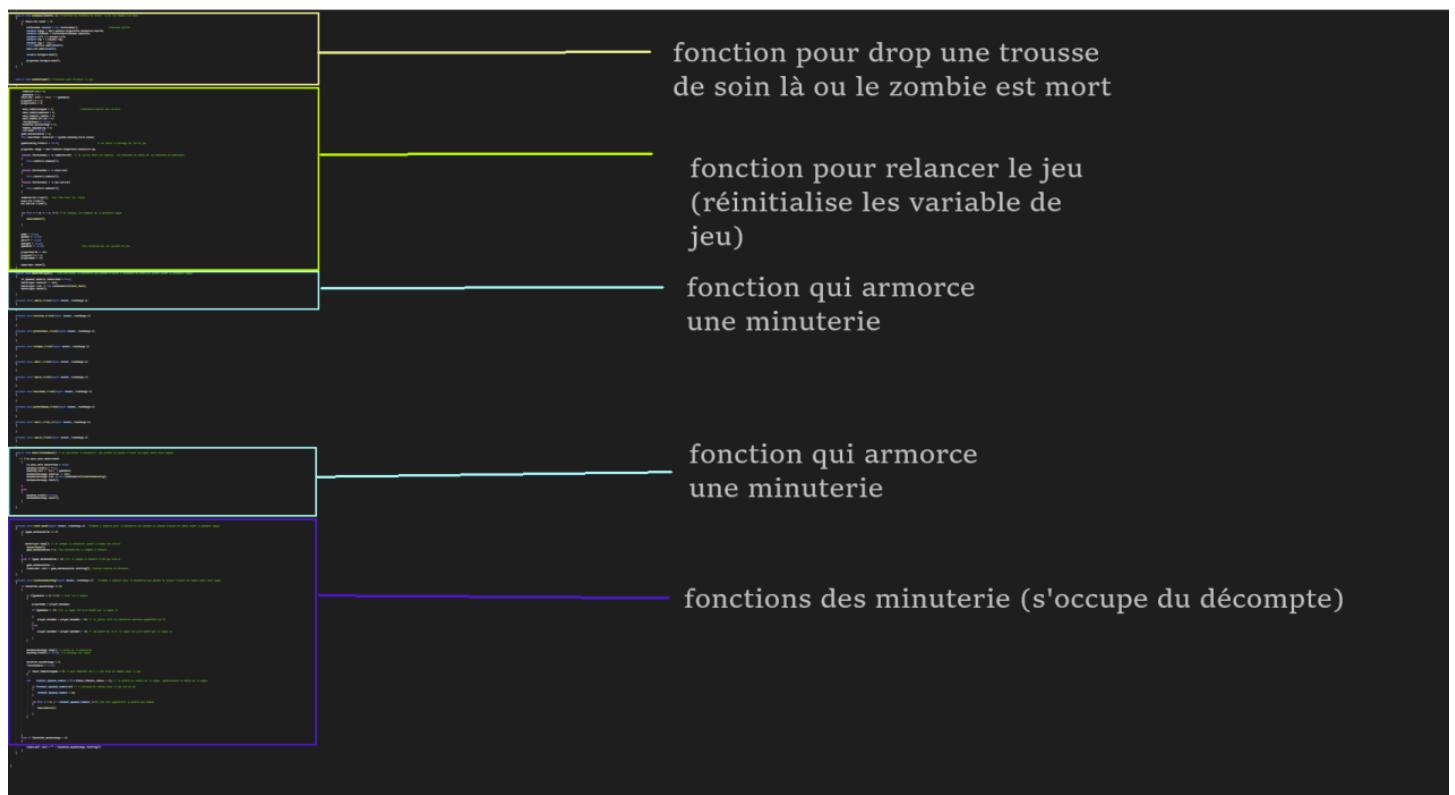
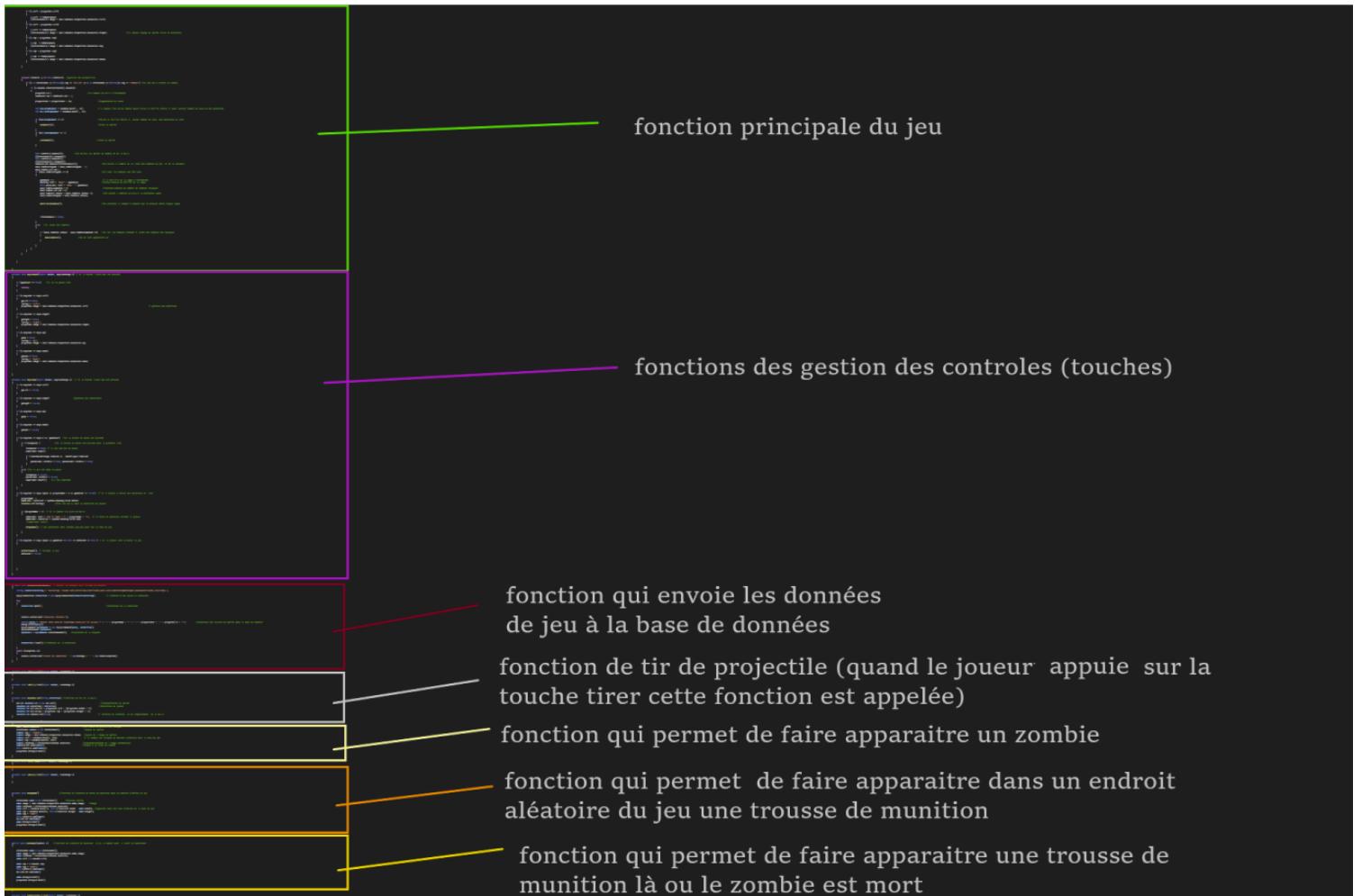
- Dossier principal du jeu

- Form1.cs
- Form2.cs
- Form1.Designer.cs
- Form2.Designer.cs
- Program.cs

Form1.cs

Ce fichier s'occupe du fonctionnement du jeu





Form2.cs

fichier qui s'occupe du comportement de la fenêtre demandant d'entrer un pseudo au joueur

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Diagnostics;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
```

```
namespace Skellibase
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            if (PseudoInput.Text.Length > 0 && PseudoInput.Text.Contains(" "))
            {
                MessageBox.Show("Le pseudo ne doit pas contenir d'espace !");
                PseudoInput.Text = "";
            }
        }

        private void Confirm_Click(object sender, EventArgs e)
        {
            if (PseudoInput.Text.Length > 0 && PseudoInput.Text.Contains(" "))
            {
                MessageBox.Show("Le pseudo ne doit pas contenir d'espace !");
                PseudoInput.Text = "";
            }
            else
            {
                if (JeuGame != null)
                {
                    JeuGame.Close();
                }
                else
                {
                    JeuGame = new Form();
                    JeuGame.Text = "Jeu";
                    JeuGame.Size = new Size(800, 600);
                    JeuGame.StartPosition = FormStartPosition.CenterScreen;
                    JeuGame.TopMost = true;
                    JeuGame.FormBorderStyle = FormBorderStyle.FixedSingle;
                    JeuGame.MaximizeBox = false;
                    JeuGame.MinimizeBox = false;
                    JeuGame.CenterToScreen();
                    JeuGame.Show();
                    JeuGame.SetPlayerName(PseudoInput.Text);
                    JeuGame.InitializeComponent();
                    JeuGame.InitFormPlayer();
                }
            }
        }

        private void DestroyForm()
        {
            if (JeuGame != null)
            {
                JeuGame.Close();
            }
        }

        private void Info_Click(object sender, EventArgs e)
        {
        }

        private void Errormessage_Click(object sender, EventArgs e)
        {
        }
    }
}
```

importe les bibliothèque

initialise les controles

fonction qui gère les entrées utilisateur