

CFPT

# Guide d'installation

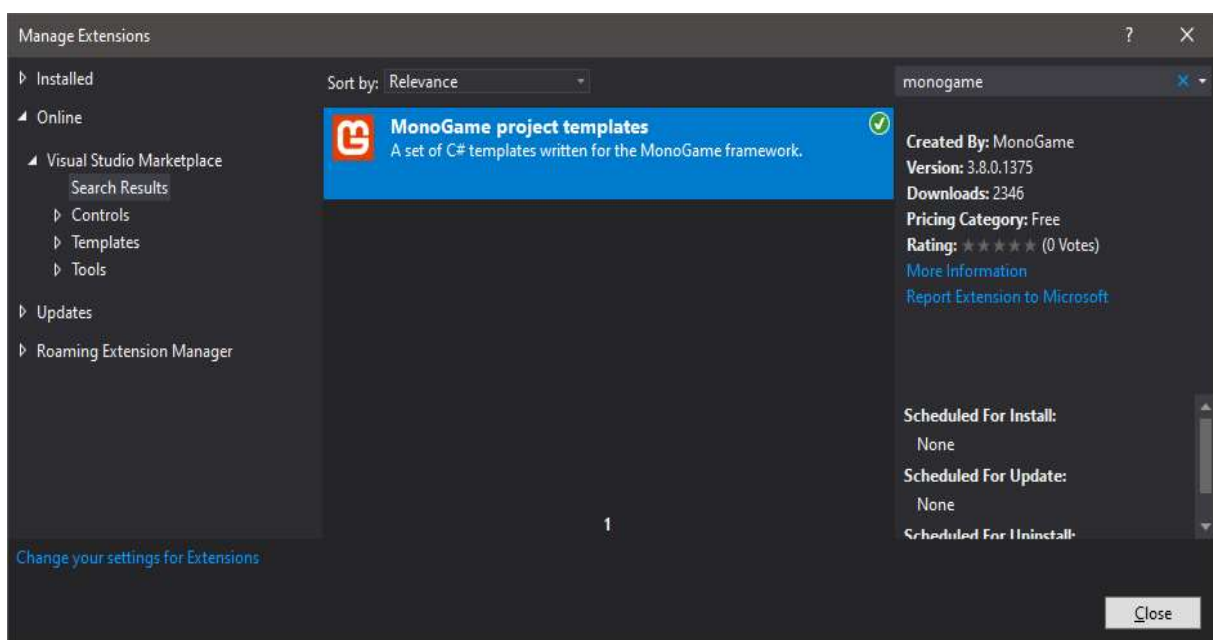
Groupe Bugs Destroyer  
21/10/2021

## Table des matières

Installation de MonoGame.....	1
Installation du jeu .....	2
Installation de la police .....	2
Les diagrammes de classe .....	3

### 1. Installation de MonoGame

Vous devrez installer l'extension Visual Studio 2019, qui peut être installée à partir de " *Extensions -> Gérer les extensions* " dans la barre de menus de Visual Studio.



Une fois qu'il est ouvert, recherchez simplement **MonoGame** dans la fenêtre de recherche en haut à droite, comme indiqué ci-dessus, et installez les "modèles de projet MonoGame". Vous avez maintenant les modèles MonoGame installés, prêts à créer de nouveaux projets.

## 1.2 Installer l'éditeur MGCB

MGCB Editor est un outil d'édition de fichiers .mgcb, qui sont utilisés pour créer du contenu.

Pour enregistrer l'outil MGCB Editor avec Windows et Visual Studio 2019, exécutez ce qui suit à partir de l'invite de commande.

```
dotnet tool install --global dotnet-mgcb-editor
mgcb-editor --register
```

- Générateur de contenu MonoGame (MGCB) : `dotnet tool install -g dotnet-mgcb`
- Éditeur MGCB (anciennement outil Pipeline) : `dotnet tool install -g dotnet-mgcb-editor`
- Compilateur d'effets MonoGame (MGFXC ; auparavant 2MGFX) : `dotnet tool install -g dotnet-mgfxc`

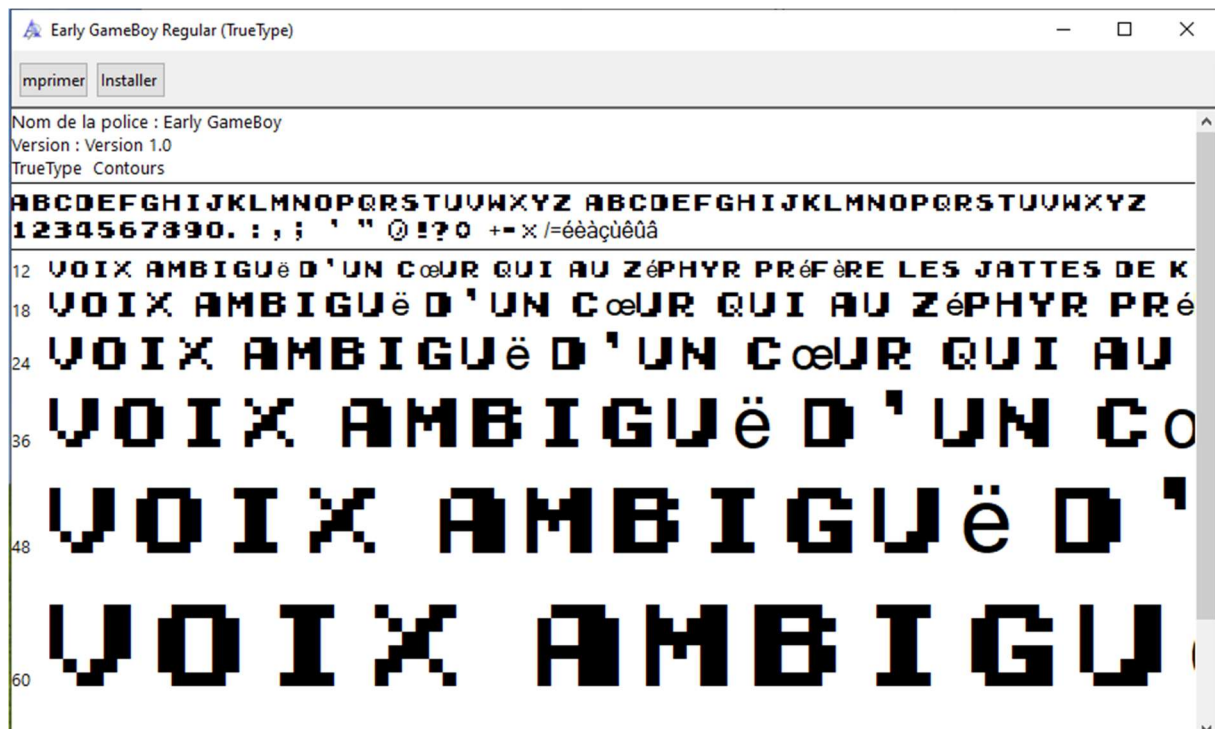
Après l'installation, exécutez `mgcb-editor --register` pour enregistrer MGCB Editor comme application par défaut pour les fichiers mgcb.

## 2. Installation du jeu

Le lien du GitHub : <https://github.com/AlecInfo/BugsDestroyer.git>

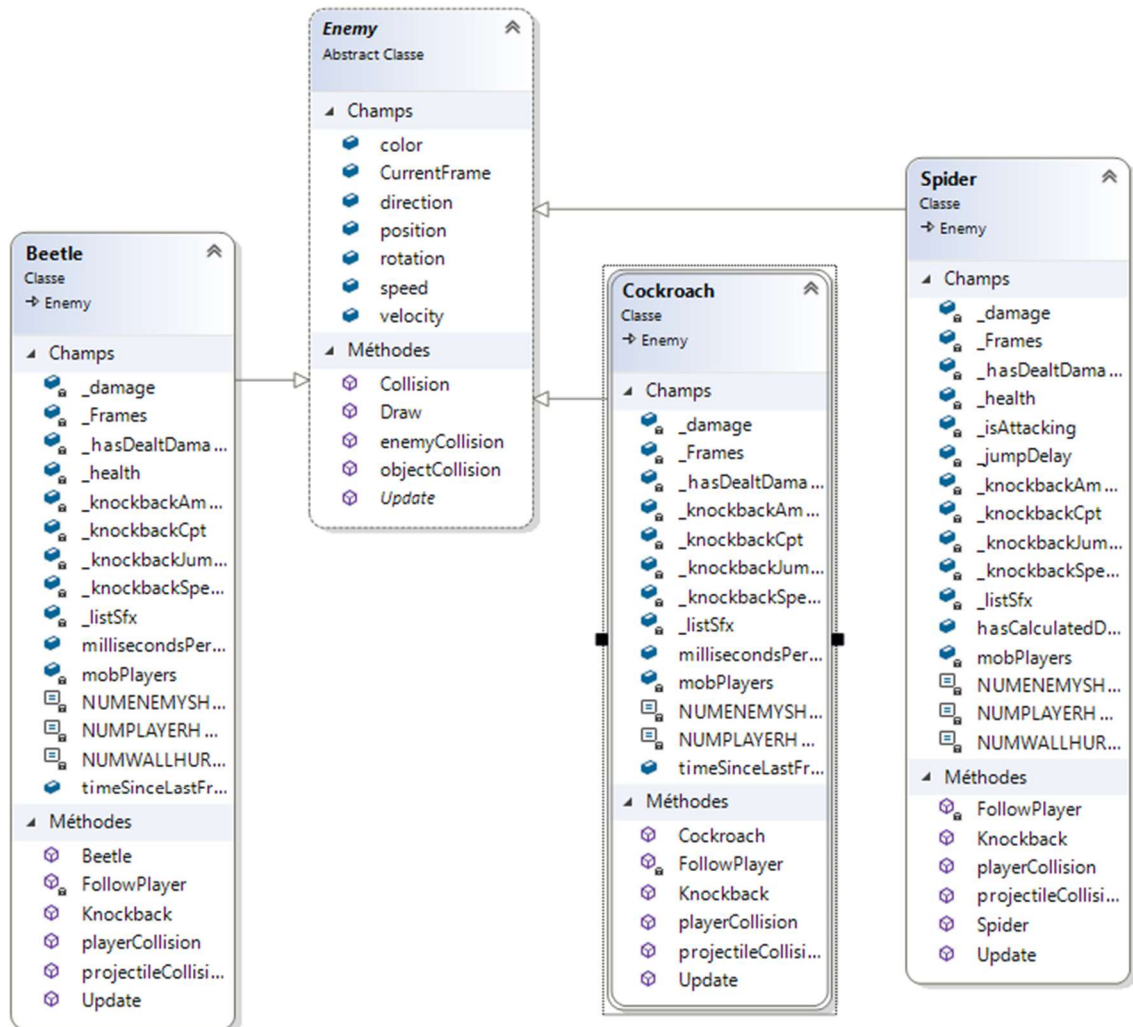
## 3. Installation de la police

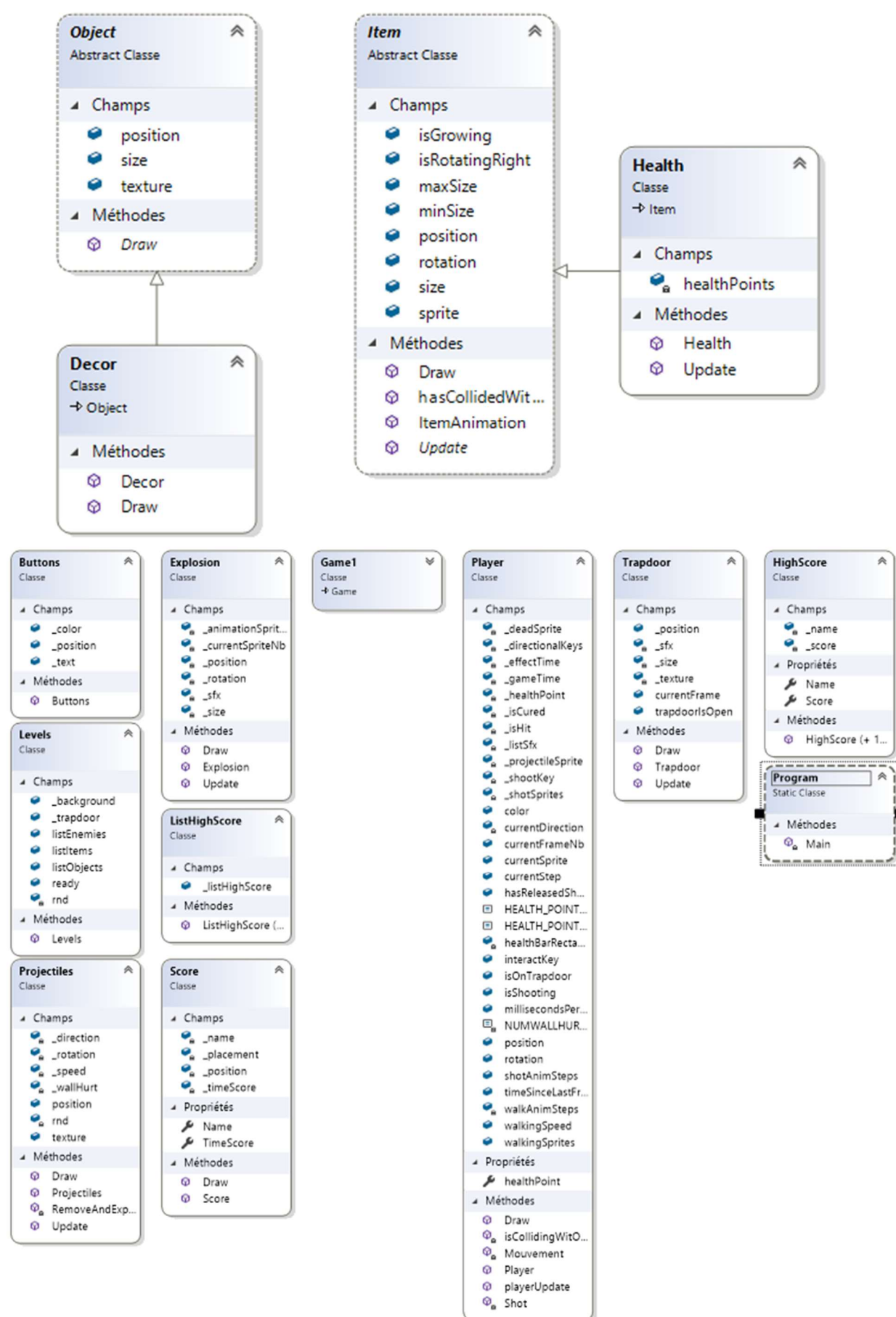
Dans le fichier font vous trouvez « Early GameBoy.ttf » ouvrez le.



Cliquer sur « Installer ».

## 4. Les diagrammes de classe





## 5. Explication du code

La classe Game1 représente l'appelle de toutes les autres classes, c'est la classe principale du jeu.

Dans la méthode Load, c'est tous ce qu'on doit charger exemple : les images ect...

Il y a plusieurs régions :

Music / SoudEffect : représente les musiques etc...

Decor : représente le décor du jeu.

Players Walking : représente les déplacements du joueur.

Players Shooting : représente le tir.

Players Dead : représente la mort.

Projectiles : représente les projectiles du jeu.

Enemys : pour les ennemies.

Items : représente les items du jeu.

```
0 references | 1 file | 11 y 8 3 heures | 3 auteurs, 33 modifications
protected override void LoadContent()
{
    _spriteBatch = new SpriteBatch(GraphicsDevice);

    Music / SoundEffect

    Decor

    Players Walking

    Players Shooting

    Players Dead

    Players Health Bar

    Projectiles

    Enemys

    Items

    LevelLoad(_listSfx);

    menuLoad();

    menuPauseLoad();

    gameOverLoad();

    winLoad();
}
```

Dans la méthode Update, c'est tous les codes pour réactualise chaque frame.

Timer : pour afficher le temps du jeu.

Levels : pour les niveaux du jeu.

Players : pour les déplacements.

Enemys : pour les ennemies.

Items : pour les items du jeu.

```
protected override void Update(GameTime gameTime)
{
    if (GamePad.GetState(PlayerIndex.One).Buttons.Back == ButtonState.Pressed || Keyboard
        Exit();

    // si le joueur est dans le menu de bienvenu
    if (isOnMenu)
    {
        menuUpdate(gameTime);
    } // sinon si il est pas dans le menu
    else if (!isOnMenu)
    {
        // si le joueur est pas dans le menu pause
        if (!isPause)
        {
            Timer
            Levels
            Players
            Enemys
            Items
            Projectiles

            gameOverUpdate(gameTime);
        }

        winUpdate(gameTime);

        // acctualisation du menu pause

        if (!isDead) // if all player aren't dead
        {
            // acctualisation du menu pause
            menuPauseUpdate(gameTime);
        }
    }

    base.Update(gameTime);
}
```



Projectiles : Pour que les joueurs puissent tirer.

Dans la méthode Draw, c'est l'affichage du jeu.

Decor : pour afficher le décor.

Player : pour afficher les joueurs.

Enemies : pour afficher les ennemis.

Items : pour afficher les items du jeu.

Ombre : pour afficher les ombres du jeu.

Projectiles : pour afficher les tirs.

Timer : pour afficher le temps du jeu.

0 références | alec.ptt, Il y a 5 heures | 4 auteurs, 53 modifications

```
protected override void Draw(GameTime gameTime)
{
    GraphicsDevice.Clear(Color.Black);
    _spriteBatch.Begin(samplerState: SamplerState.PointClamp);

    // si le joueur est dans le menu de bienvenu
    if (isOnMenu)
    {
        // affichage du menu
        menuDraw(gameTime);
    } // sinon si il n'est pas dans le menu
    else if (!isOnMenu)
    {
        Decor
        Player
        Enemies
        Items
        Ombre
        Projectiles
        timer

        _spriteBatch.Draw(_menuPauseImages[0], new Vector2(0, -100));

        // affichage du menu pause
        menuPauseDraw(gameTime);

        // affichage du gameOver
        gameOverDraw(gameTime);

        // affichage de la page de win
        winDraw();
    }

    _spriteBatch.End();
    base.Draw(gameTime);
}
```