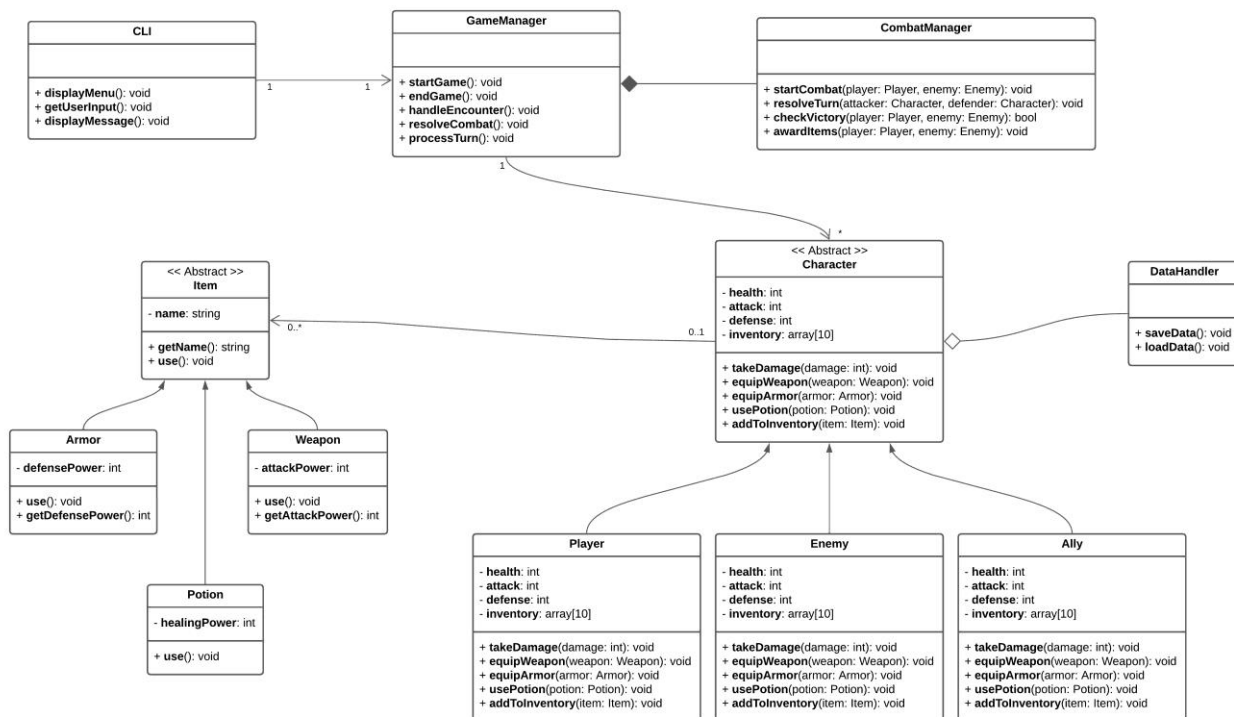


Un petit jeu de rôle :

Pour commencer, tout n'est pas à 100% fonctionnel et quelques erreurs se sont glissées dans le code comme le chargement qui ne fonctionne pas comme prévu et n'a pas été bien pensé. Celui-ci devra être réétudié pour une prochaine évolution.

Diagramme UML :

Celui-ci n'est pas entièrement à jour mais représente bien la conception globale du programme, certaines méthodes ont été ajoutées que je n'ai pas eu le temps de rajouter.



Algorithme :

Le jeu commence par un menu permettant de choisir si on veut lancer une nouvelle partie, charger une partie ou quitter le jeu.

Si on choisit de commencer une nouvelle partie, nous devons choisir une classe de personnage (3 choix disponibles).

Une fois la partie lancée, le joueur peut soit continuer dans l'aventure (donc rencontré un autre personnage aléatoire), soit consulter ses statistiques et son inventaire, soit équiper un objet (armure ou arme), soit arrêter le jeu auquel cas la partie sera sauvegardée automatiquement.

Je vous laisserais découvrir le boss qui apparaît rarement (avec un petit nom mignon), il vous faudra pour cela être très persévérant.

Pensez à activer votre son par ailleurs lorsque vous jouez au jeu :)

Jeux de tests :

Choix de personnage :

Quand on choisit un personnage :

```
-----  
| Choose your character |  
-----  
| 1. Berserker          |  
| 2. Paladin            |  
| 3. Warrior            |  
-----  
Enter your choice: 1|
```

```
Name: Berserker  
Health: 90  
Attack: 16  
Defense: 9  
Equipped Weapon: None  
Equipped Armor: None  
Inventory:  
0: Empty
```

On obtient bien le personnage de la classe souhaitée.

Statistiques et inventaire :

Quand on choisit de visionner les statistiques et l'inventaire du personnage, celui-ci fonctionne bien (sauf mes noms des équipements qui ne s'affichent pas comme il faut mais ils sont quand même bien stockés).

```
Choose your action:  
1. Move forward  
2. View inventory and stats  
3. Equip item  
4. Quit game  
2  
  
Name: Berserker  
Health: 90  
Attack: 16  
Defense: 9  
Equipped Weapon: None  
Equipped Armor: None  
Inventory:  
0: Empty  
1: Empty  
2: Empty  
3: Empty  
4: Empty  
5: Empty  
6: Empty  
7: Empty  
8: Empty  
9: Empty
```

Avancement dans l'aventure :

On peut rencontrer soit un objet que l'on peut ramasser, soit un ennemi qu'il va falloir affronter, soit un allié qui nous proposer le choix d'un objet parmi son inventaire.

```
Choose your action:
1. Move forward
2. View inventory and stats
3. Equip item
4. Quit game
1

You found an item on the ground: Health Potion! Do you want to pick it up? (y/n)
|
```

```
Choose your action:
1. Move forward
2. View inventory and stats
3. Equip item
4. Quit game
1
A wild Goblin has appeared!
```

```
Player attacks enemy!
Enemy health: 39
```

```
Enemy attacks player!
Player health: 89
```

```
Player attacks enemy!
Enemy health: 28
```

```
Enemy attacks player!
Player health: 88
```

```
Player attacks enemy!
Enemy health: 17
```

```
Enemy attacks player!
Player health: 87
```

```
Player attacks enemy!
Enemy health: 6
```

```
Enemy attacks player!
Player health: 86
```

```
Choose your action:
```

1. Move forward
 2. View inventory and stats
 3. Equip item
 4. Quit game
- 1

You encountered an ally!

```
Choose an item (enter the index):
```

```
0: Name: Shield, Type: Armor, Power: 3
```

```
1: Name: Mana Potion, Type: Potion, Power: 59
```

Ajout d'un objet dans l'inventaire :

On peut récupérer des objets au fil de l'aventure et les récupérer dans notre inventaire.

```
Choose an item (enter the index):
0: Name: Health Potion, Type: Potion, Power: 50
1: Name: Sword, Type: Weapon, Power: 7
1

Item added to inventory:

Choose your action:
1. Move forward
2. View inventory and stats
3. Equip item
4. Quit game
2

Name: Berserker
Health: 85
Attack: 16
Defense: 9
Equipped Weapon: None
Equipped Armor: None
Inventory:
0: Health Potion
1: Sword
2: Empty
3: Empty
4: Empty
5: Empty
6: Empty
7: Empty
8: Empty
9: Empty
```

```
You found an item on the ground: Health Potion! Do you want to pick it up? (y/n)
y

Item added to your inventory.

Choose your action:
1. Move forward
2. View inventory and stats
3. Equip item
4. Quit game
2

Name: Berserker
Health: 90
Attack: 16
Defense: 9
Equipped Weapon: None
Equipped Armor: None
Inventory:
0: Health Potion
1: Empty
2: Empty
3: Empty
4: Empty
5: Empty
6: Empty
7: Empty
8: Empty
9: Empty
```

Équiper un objet :

Quand on a des objets (armure ou arme) dans son inventaire on peut les équiper pour augmenter la puissance ou la résistance du joueur. L'affichage du nom ne fonctionne pas dans l'inventaire.

```
Equipment:
0: Health Potion
1: Sword
2: Health Potion
3: Mana Potion
4: Empty
5: Empty
6: Empty
7: Empty
8: Empty
9: Empty

Enter the index of the item to equip or -1 to go back: 1
Weapon equipped: Sword

Choose your action:
1. Move forward
2. View inventory and stats
3. Equip item
4. Quit game
2

Name: Berserker
Health: 80
Attack: 23
Defense: 9
Equipped Weapon: ???
Equipped Armor: None
Inventory:
0: Health Potion
1: Empty
2: Health Potion
3: Mana Potion
4: Empty
5: Empty
6: Empty
7: Empty
8: Empty
9: Empty

Equipment:
0: Shield
1: Empty
2: Empty
3: Empty
4: Empty
5: Empty
6: Empty
7: Empty
8: Empty
9: Empty

Enter the index of the item to equip or -1 to go back: 0
Armor equipped: Shield

Choose your action:
1. Move forward
2. View inventory and stats
3. Equip item
4. Quit game
2

Name: Warrior
Health: 100
Attack: 14
Defense: 28
Equipped Weapon: None
Equipped Armor: ???
Inventory:
0: Empty
1: Empty
2: Empty
3: Empty
4: Empty
5: Empty
6: Empty
7: Empty
8: Empty
9: Empty
```

Cela vérifie également qu'une arme augmente l'attaque du personnage et qu'une armure augmente la défense du personnage.

Lorsque le joueur souhaite équiper un objet alors qu'il en a déjà un du même type, le premier sera alors détruit et remplacé par le nouveau.

```
Equipment:
0: Health Potion
1: Empty
2: Shield
3: Empty
4: Empty
5: Empty
6: Empty
7: Empty
8: Empty
9: Empty

Enter the index of the item to equip or -1 to go back: 2
Armor unequipped
Armor equipped: Shield

Choose your action:
1. Move forward
2. View inventory and stats
3. Equip item
4. Quit game
2

Name: Warrior
Health: 100
Attack: 14
Defense: 22
Equipped Weapon: None
Equipped Armor: ???
Inventory:
0: Health Potion
1: Empty
2: Empty
3: Empty
4: Empty
5: Empty
6: Empty
7: Empty
8: Empty
9: Empty
```

Utiliser une potion :

Si les points de vie du joueur tombent à 0 et qu'il possède au moins une potion dans son inventaire, celle-ci sera automatiquement utilisée pour le sauver.

```
Choose your action:
1. Move forward
2. View inventory and stats
3. Equip item
4. Quit game
1
A wild Orc has appeared!

Enemy attacks player!
Player health: 0
Player uses Healing Potion!
```

Mort du personnage :

Lorsque les points de vie du personnage tombent à 0 et qu'il n'a plus de potion en sa possession, alors la partie est terminée et le jeu s'arrête.

```
Enemy attacks player!
Player health: 0
Player has been defeated!
root@LAPTOP-OL4QNR5Q:/mnt/c/Users/bruf1/Cours HEXAGONE/C & C++/tps/tp3/tlsf_game#
```

Sauvegarde de la partie :

Quand on quitte le jeu (sans mourir), la partie est automatiquement sauvegardée dans un fichier "save.txt".

```
≡ save.txt
1  Warrior
2  100
3  14
4  10
5  None
6  None
7  10
8  Weapon
9  Sword
10 5
11
```

Chargement d'une partie :

Le chargement d'une partie n'est pas entièrement fonctionnel mais une partie du code est présente.

```
-----  
|   The legend of   |  
| Segmentation Fault |  
-----  
| 1. New Game      |  
| 2. Load Save     |  
| 3. Exit          |  
-----  
Enter your choice: 2  
Loading save...  
Item type: Weapon  
Item name: Sword  
-----  
|   The legend of   |  
| Segmentation Fault |  
-----  
| 1. New Game      |  
| 2. Load Save     |  
| 3. Exit          |  
-----  
Enter your choice: █
```

Quitter le jeu :

```
Choose your action:  
1. Move forward  
2. View inventory and stats  
3. Equip item  
4. Quit game  
4  
Exiting game...  
root@LAPTOP-OL4QNR5Q:/mnt/c/Users/brufl/Cours HEXAGONE/C & C++/tps/tp3/tlsf_game# █
```

Partie graphique :

J'ai commencé à développer une partie graphique mais je l'ai vite abandonné pour le moment par manque de temps.

Partie sonore :

La partie sonore a été gérée grâce à la librairie SDL, c'est pourquoi vous devrez l'installer. Les instructions sont données dans le README du projet.

Principes algorithmiques :

La plupart des principes algorithmiques sont plus ou moins bien expliqués par les commentaires contenus dans le code.

Arborescence du projet :

tlsf_game/

```
|
|└─ bin/
|  └─ main
|
|└─ build/
|   └─ BusinessLogicLayer/
|      └─ GameManager.o
|      └─ FightManager.o
|
|   └─ DataLayer/
|      └─ Character.o
|      └─ Player.o
|      └─ Enemy.o
|      └─ Ally.o
|      └─ Item.o
|      └─ Potion.o
|      └─ Weapon.o
|      └─ Armor.o
|      └─ ItemType.o
|      └─ DataHandler.o
|
|   └─ PresentationLayer/
|      └─ CLI.o
|      └─ MusicPlayer.o
|      └─ GUI.o (interface graphique sera développée plus tard)
|
|└─ include/
|   └─ BusinessLogicLayer/
|      └─ GameManager.h
|      └─ FightManager.h
|
|   └─
```



```
| ├── DataLayer/  
| | ├── Character.h  
| | ├── Player.h  
| | ├── Enemy.h  
| | ├── Ally.h  
| | ├── Item.h  
| | ├── Potion.h  
| | ├── Weapon.h  
| | ├── Armor.h  
| | ├── ItemType.h  
| | └── DataHandler.h  
| |  
| └── PresentationLayer/  
|   ├── CLI.h  
|   ├── MusicPlayer.h  
|   └── GUI.h (interface graphique sera développée plus tard)  
|  
└── src/  
    ├── BusinessLogicLayer/  
    | ├── GameManager.cpp  
    | └── FightManager.cpp  
    |  
    ├── DataLayer/  
    | ├── Character.cpp  
    | ├── Player.cpp  
    | ├── Enemy.cpp  
    | ├── Ally.cpp  
    | ├── Item.cpp  
    | ├── Potion.cpp  
    | ├── Weapon.cpp  
    | └── Armor.cpp
```

```
|  └─ ItemType.cpp
|  └─ DataHandler.cpp
|
└─ PresentationLayer/
|  └─ CLI.cpp
|  └─ MusicPlayer.cpp
|  └─ GUI.cpp (interface graphique sera développée plus tard)
|
└─ main.cpp
```

Lien GitHub du projet :

https://github.com/BrunelFlorian/tlsf_game

Installation SDL2 (obligatoire) :

Install SDL2 (IMPORTANT)

```
sudo apt-get install libSDL2-dev libSDL2-mixer-dev
```