



Report of

# *SouL Knight*

By 周博扬

Sacabambaspis\_aaaaaa

上海科技大学

2024533141



| aaaaaa





# Information

Group\_name: Sacabambaspis\_aaaaa

Game\_name: Soul\_Knight

Author: 周博扬 2024533141

Repository: [aaaaa114514-1/Socabambaspis\\_aaaaa](https://github.com/aaaaa114514-1/Socabambaspis_aaaaa)



| aaaaaa





Scenes

Characters

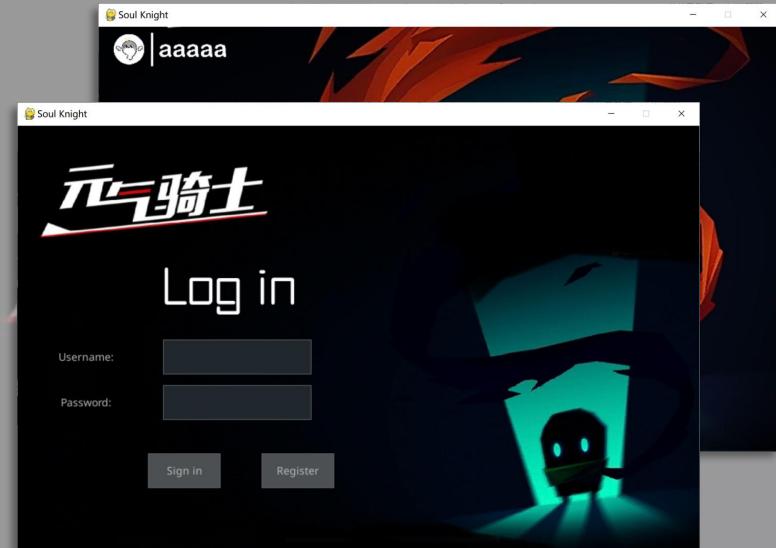
System & Value

LLM Agent

Code

Music & Art  
Creativity

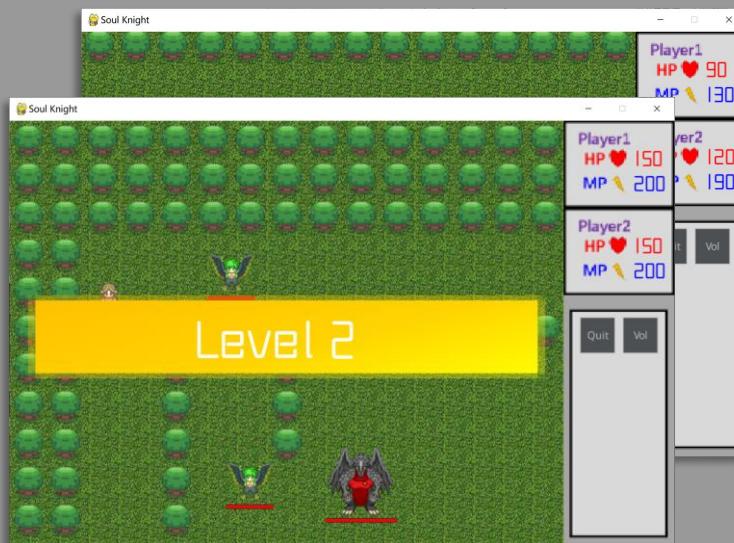
# Scenes



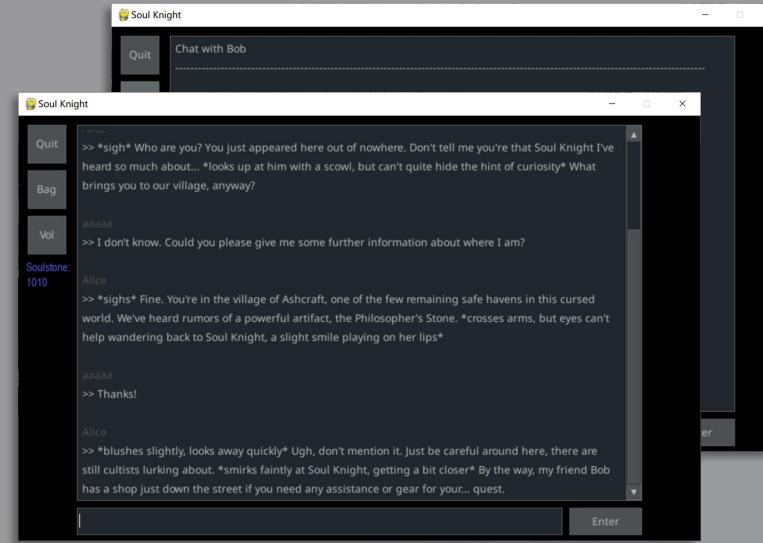
Opening & Login



Menu



Fight



Chat with npc



Plot Appreciation

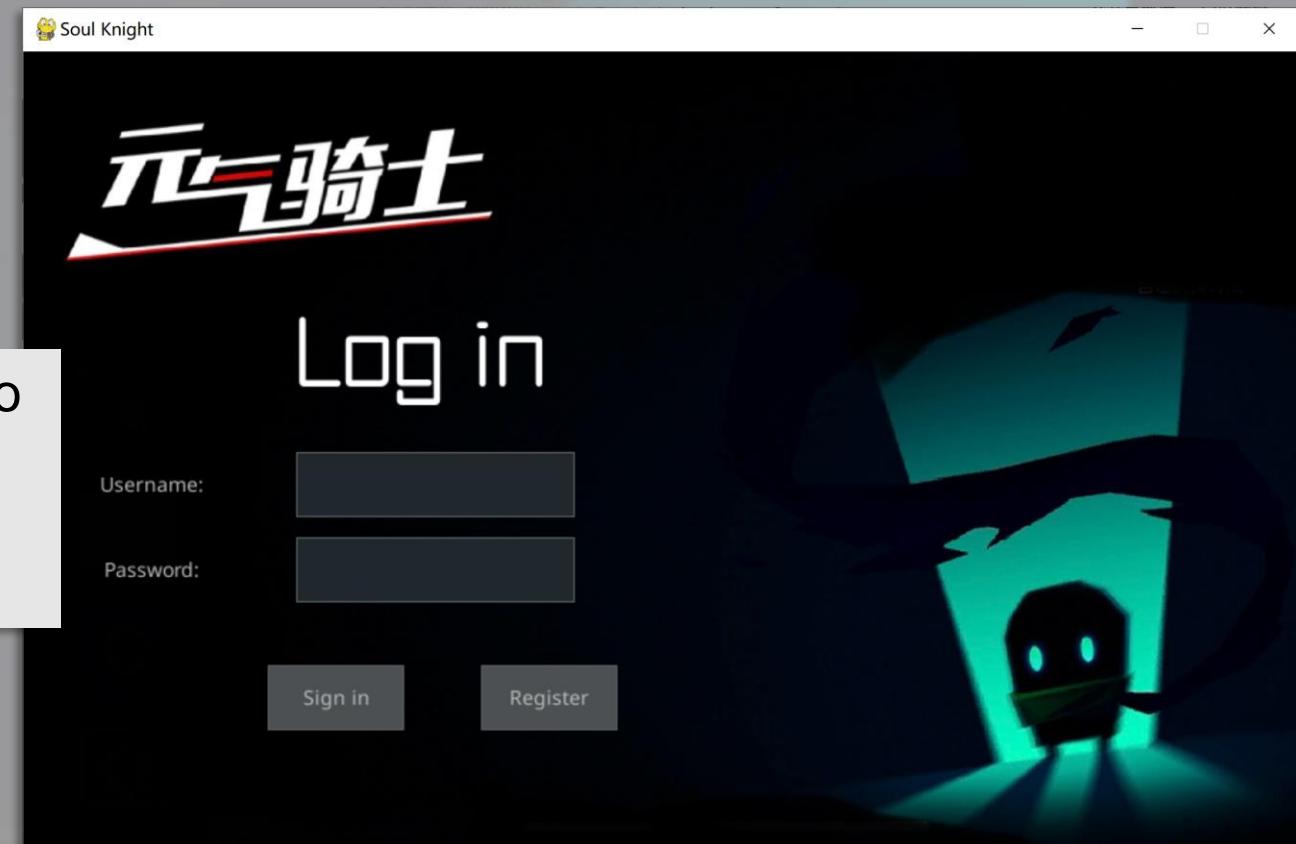
aaaaa

The opening screen of the game 'Soul Knight'. It features a knight character with a red cape and a glowing blue sword. The text '元气骑士' is displayed in large, stylized white characters on the left side of the screen.

The login page includes two input boxes, two buttons and several labels. It will check the username and the password with the database and make the correct response.

## Opening & Login

The opening page has manually implemented a fade-in effect and you can press **any keys** or **click** the screen to skip the opening animation.



# Opening & Login

Log in

Username: aaaaa

Password:

Sign in Register

The username already exists

Log in

Username:

Password: 111

Sign in Register

The username cannot be empty

The login status will be prompted below.

Log in

Username: aaa11

Password:

Sign in Register

Register successfully!

When registering, it is allowed not to set a password, but the username cannot be empty.

The username will be compared with the database and cannot be duplicated. There will be different prompts for duplicate username registration, empty username, incorrect username and password, successful registration, and successful login.

When registering with an incorrect password or a duplicate username, the password input box will automatically clear.

Log in

Username: aaaaa

Password:

Sign in Register

Welcome, aaaaa !

Log in

Username: aaa11

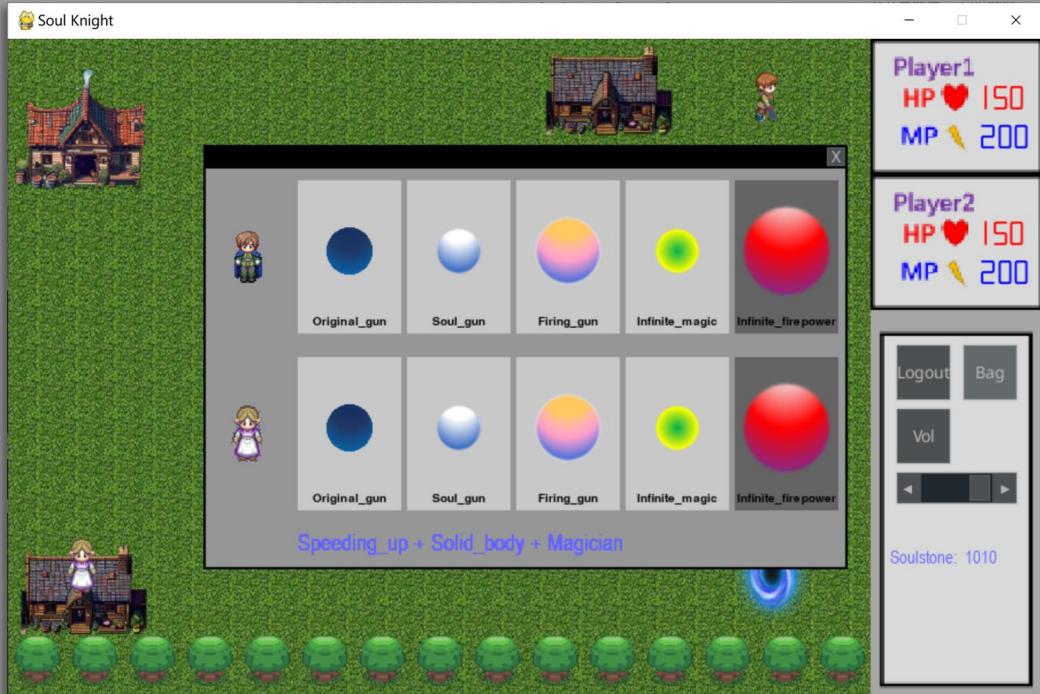
Password:

Sign in Register

Incorrect username or password

# Menu

The menu page is in the form of a map, and players can trigger various events by walking around and pressing **Space**, including conversations with npcs and acquiring or exchanging resources, or heading to battles.



You can **click** on "Logout" on the right to log out and return to the login page.

You can **click** on "Vol" on the right and **drag** the control slider to adjust the volume. Hide the control slider when **clicking** "Vol" again.

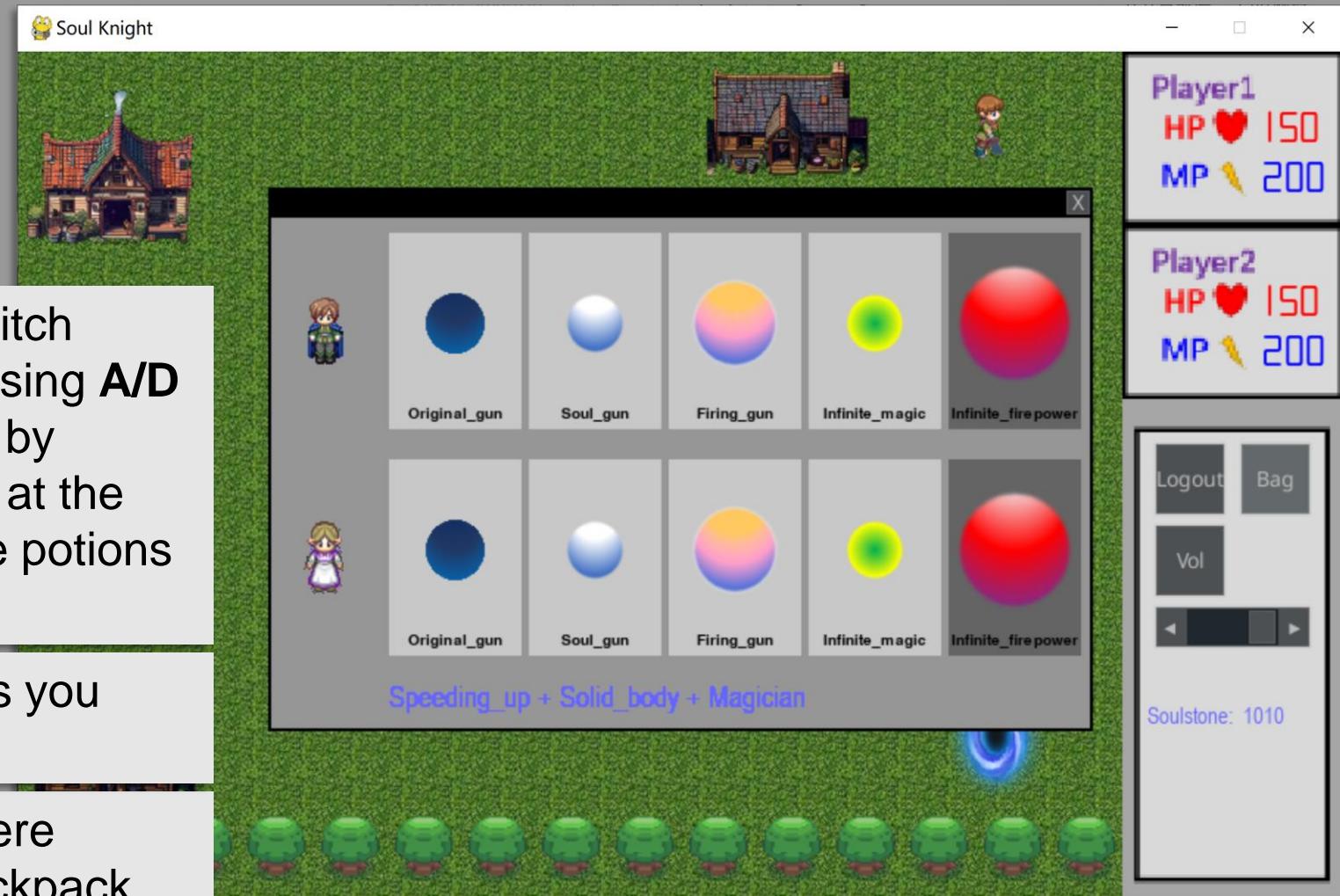
You can **click** on the "Bag" on the right to open the backpack, view and switch weapons, and use them infinitely in the menu page.

# Menu

On the backpack interface, you can switch between two players' weapons by pressing **A/D** or **←/→**. You can also switch weapons by **clicking** with the mouse. The **blue** text at the bottom of the child window displays the potions you have.

You can only choose from the weapons you have already obtained.

Press **B**, **enter** or **Esc**, or **click** anywhere outside the child window to exit the backpack interface



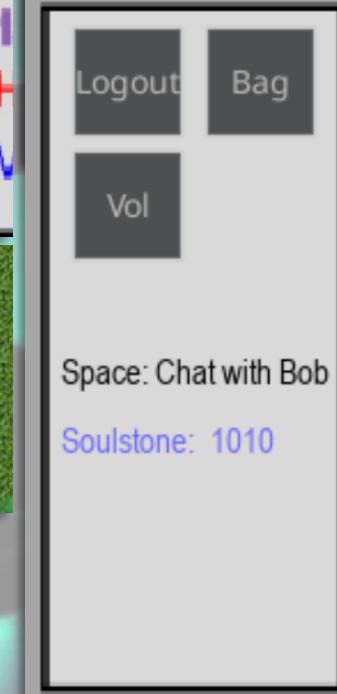
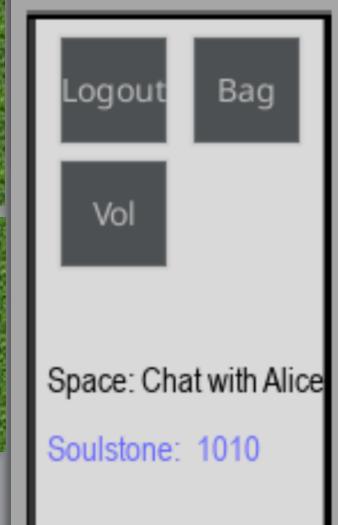
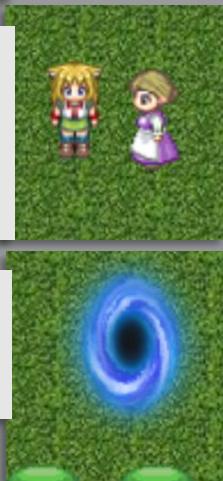
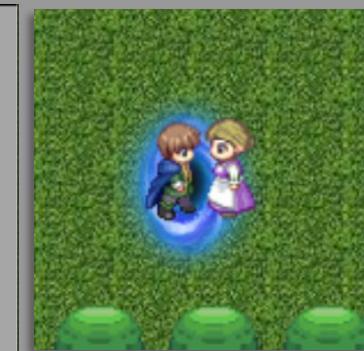
These functions also have applications in other scenes.

# Menu

The number of Soulstone and a prompt message are displayed in the bottom right corner of the screen.

When the player(s) approach the portal, a prompt will appear to start single or multiplayer gameplay.

When the player approaches the NPC, they will be prompted to chat with the NPC.





**Press the 'Quit' button on the right to exit the current battle and return to the menu interface. (Also used in other scenes)**

## Fight

In the fight interface, you can operate the character(s) to move or fire bullets.

The sidebar displays the HP and MP of two players respectively. When one's HP gets down to 0 or below, then he is defeated. (Players won't be defeated in menu) Shooting consumes magic power (MP). When one's MP cannot afford a single shot, then he cannot shoot anymore.

Players cannot pass through obstacles, but enemies can. Some bullets are set to be able to pass through obstacles, while others cannot. Once the bullet is out of the screen (excluding the right sidebar), it no longer exists.

The above mechanisms are also used in the menu.



The enemy will move in a loop (or stay still) according to the trajectory set in the level file and fire bullets at your character's current position at fixed time intervals.

Enemy units will not cause harm to each other. So does players.

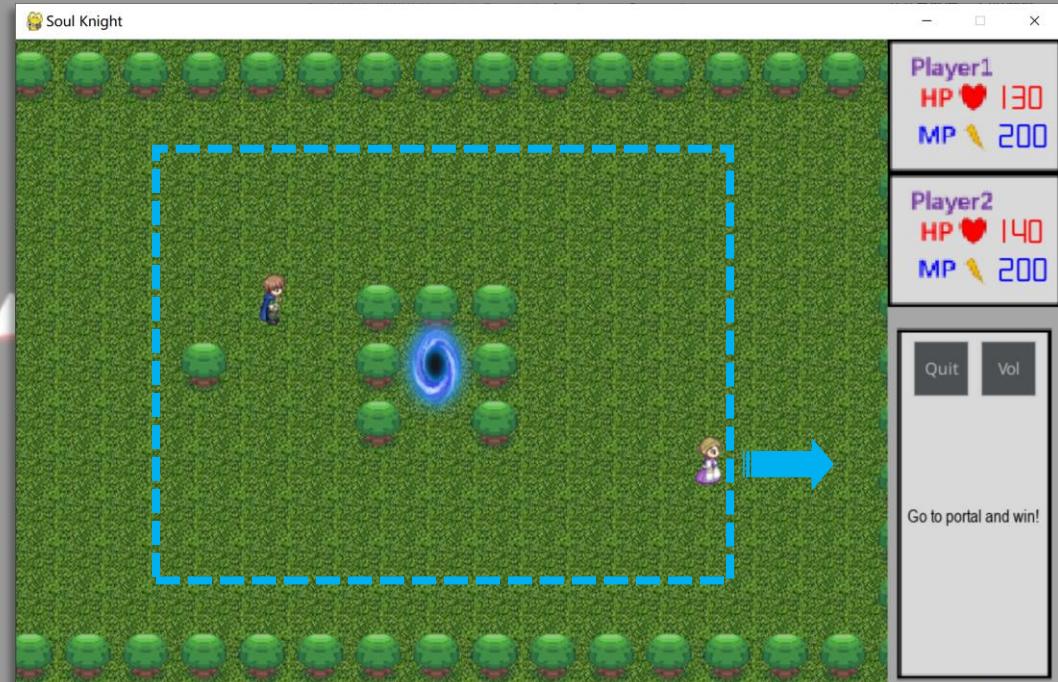
## Fight

Your goal is to defeat the enemies and then approach the portal.

When you defeat all the enemies, the portal will appear somewhere in the map.

All surviving players need to follow the instructions to enter the portal simultaneously.





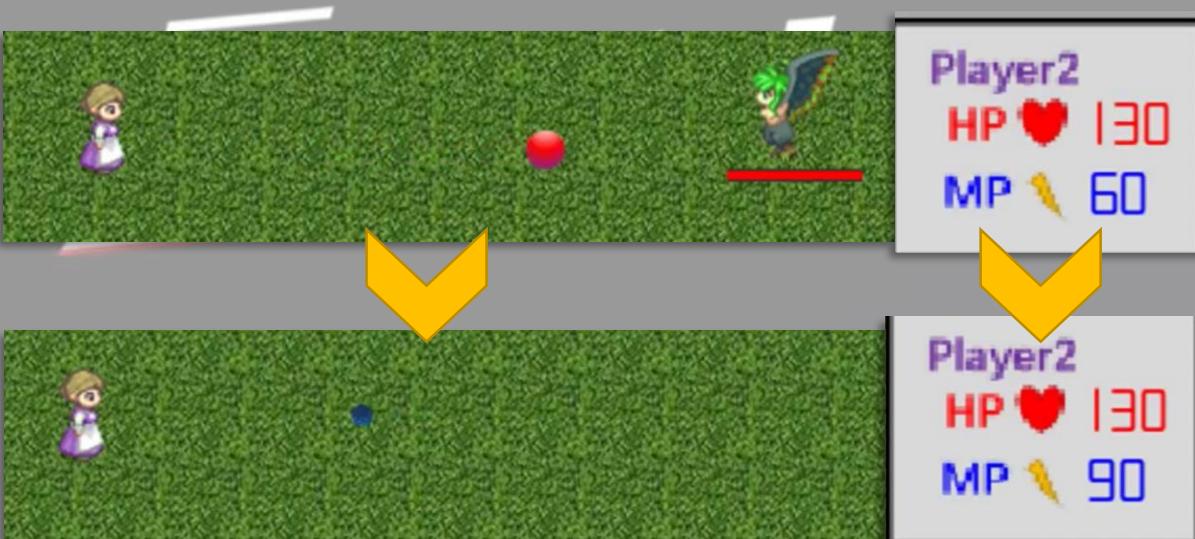
# Fight

The fight interface supports camera following.

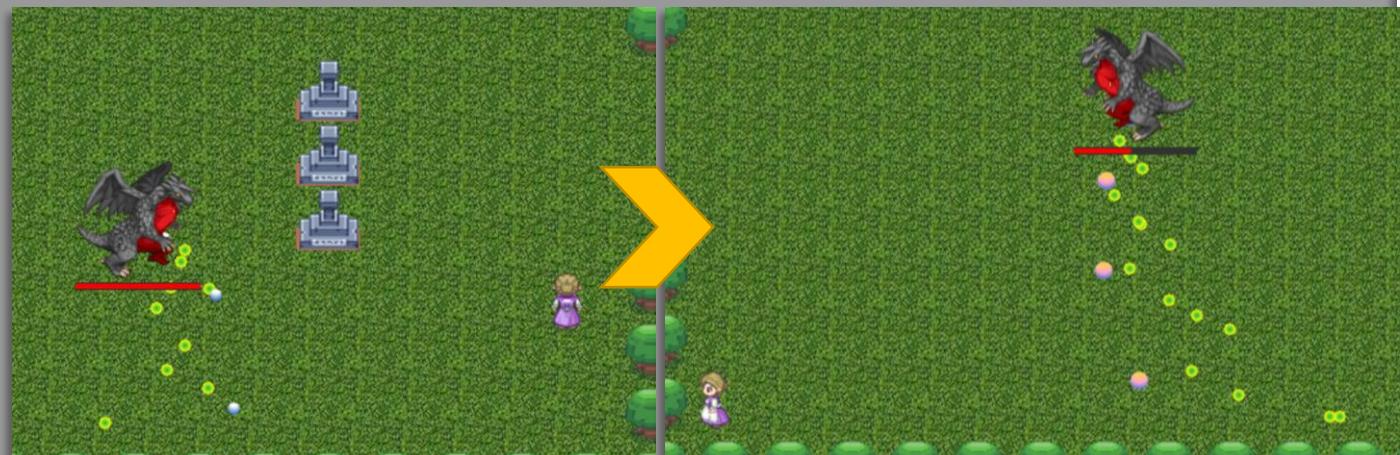
There is an invisible rectangular judgment box in the middle of the map, and neither player can leave the judgment box unless the map has reached the boundary. If two players are on opposite sides or corners of the judgment box, they cannot push the camera; Until one player moves inside the decision box and leaves the boundary, another player can move towards the boundary to push the camera to follow.

In single player mode, camera following will follow the traditional mode.

# Fight



Player who kills enemies can earn 30 MP each.



In menu page or in a battle, you can press **Esc** to minimize the game window, pause the music and open CET-6 review material ! (Have a try !)

大学英语六级考试 2022 年 9 月真题(第一套)

**Part I**

**Writing**

**(30 minutes)**

**Directions:** For this part, you are allowed 30 minutes to write an essay that begins with the sentence "It is now widely accepted that mutual trust and openness is the key to promoting cooperation." You can make comments, cite examples or use your personal experiences to develop your essay. You should write at least 150 words but no more than 200 words.

**Part II**

**Listening Comprehension**

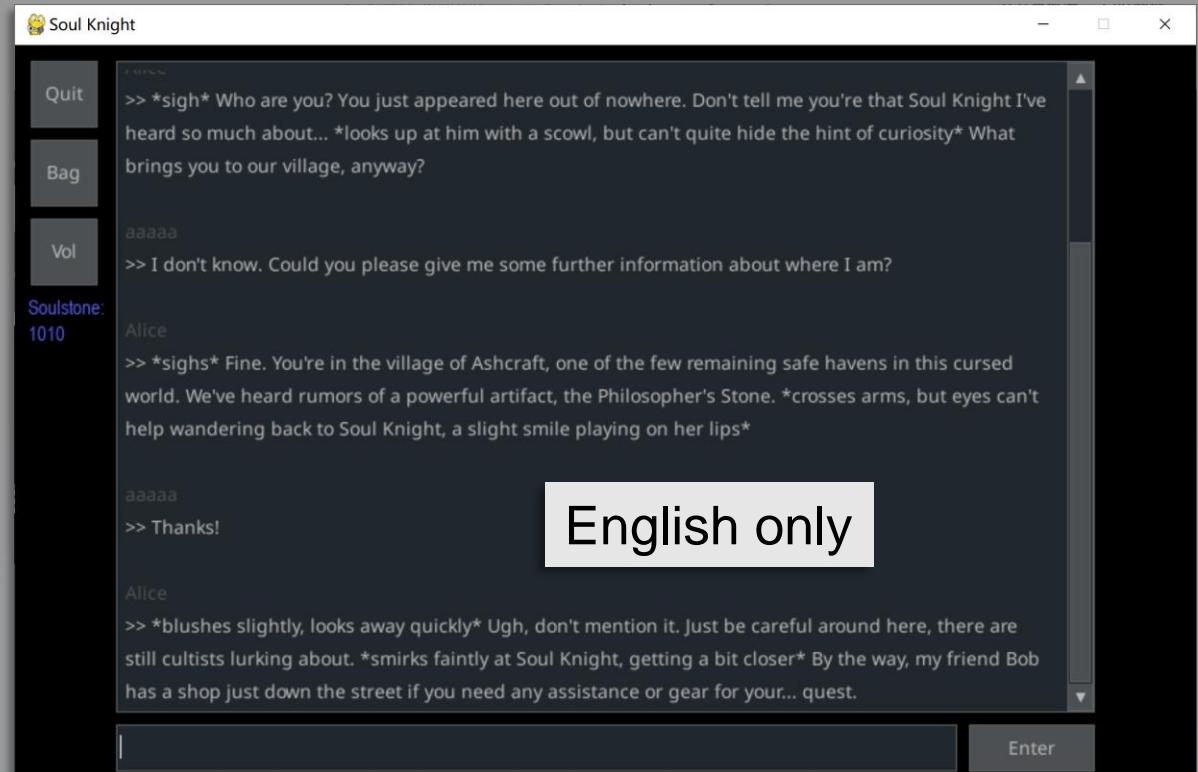
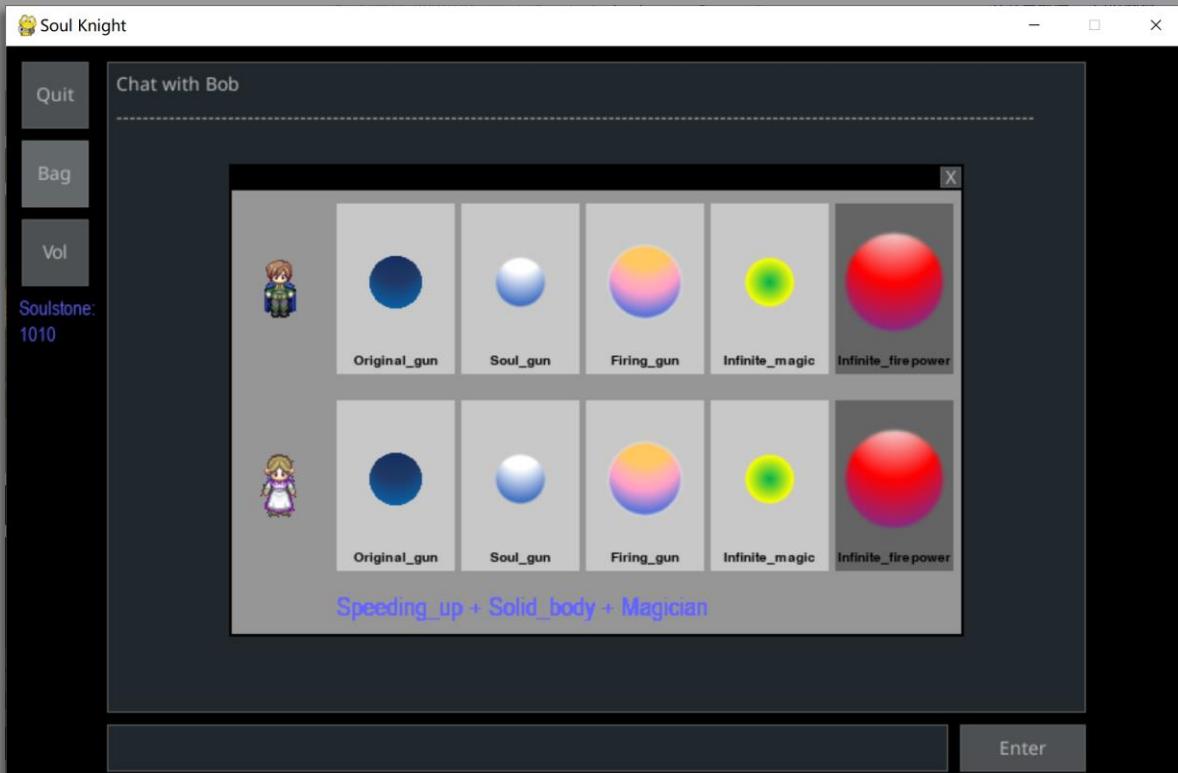
**(30 minutes)**

Section A

**Directions:** In this section, you will hear two long conversations. At the end of each conversation, you will hear

# Chat with npc

In the real-time conversation interface, you can use the “Bag”, “Quit” and “Vol” functions mentioned earlier, too.



English only

You can communicate with AI supported NPCs: Alice and Bob. The dialogue interface can be flipped up and down by **dragging** the scroll bar.

You can key in the text box below and press **Enter** (on keyboard or keypad) or **press the button Enter** to send a message to him/her. When the message is empty, the button cannot be pressed.

# Chat with npc

Alice is a Tsundere catgirl, and she knows something about the world. You can ask her to tell you some background information of the world. Also, you may **get 1~3 Soulstones from her** as a gift. ~~Living off a woman couldn't be your main revenue sources.~~



Bob is an elf boy shopkeeper. He owns a weapon shop that sells weapons and potions. You can **trade with him by chatting** with him.

**Both characters are romanceable.**

There is a parameter called likeability ranging from 0~100.

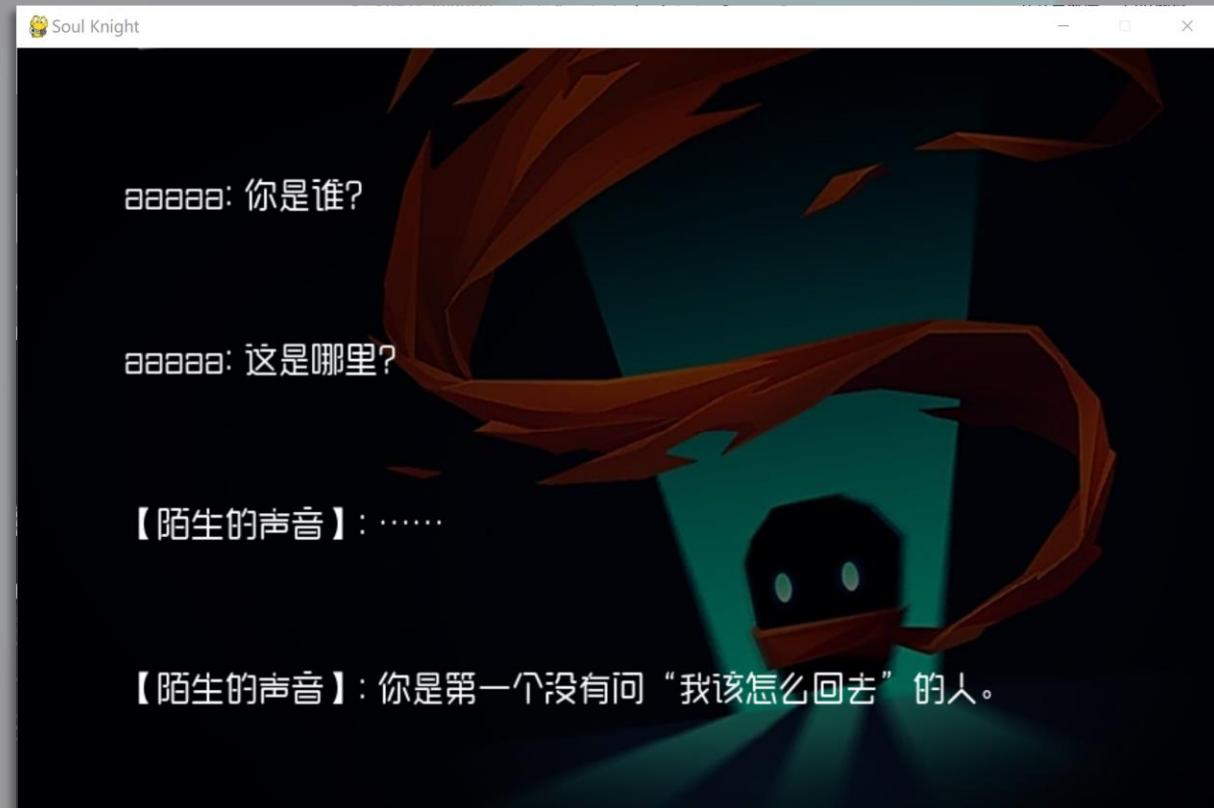
You can increase your likeability , as well as your relationship by chatting.



You can **scroll up** the mouse wheel (not in auto mode) to check the history dialogue. You can then **scroll up/down** to turn pages. When you turn to the last page and you can **scroll down** to quit the history mode. You can also press **Esc** to quit the history mode.

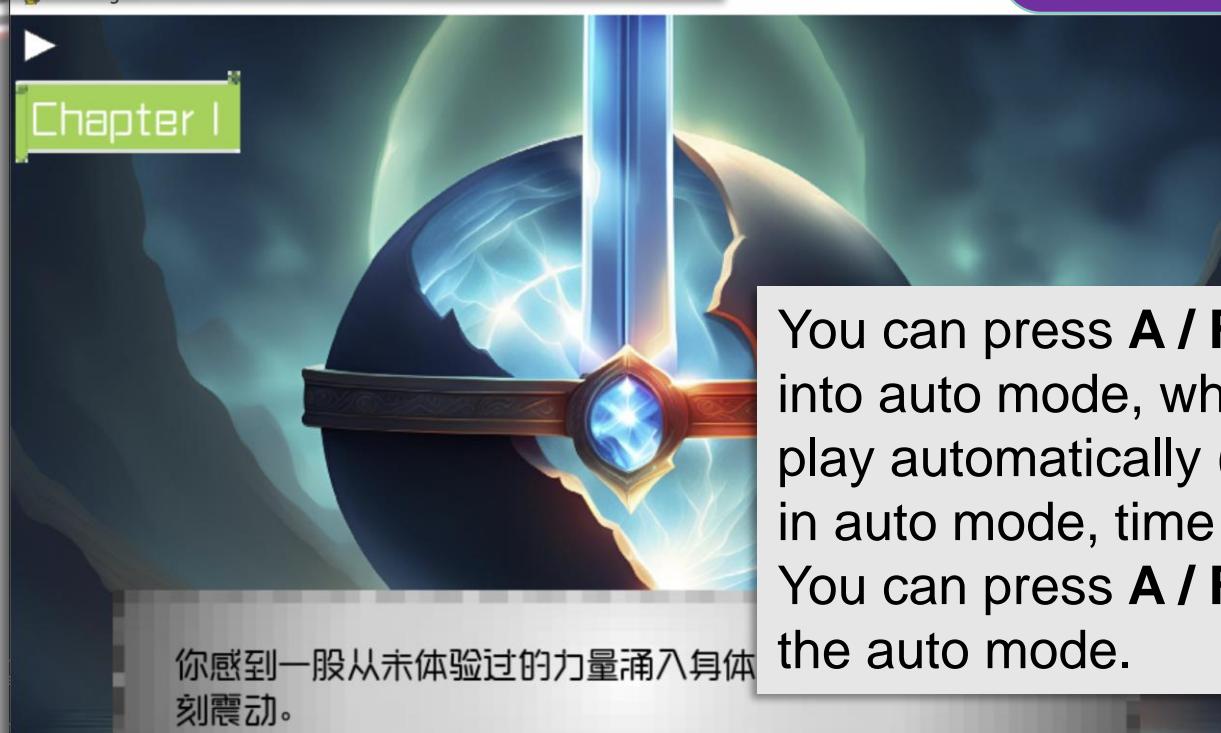
# Plot Appreciation

The inspiration for the interface design comes from Galgame.



# Plot Appreciation

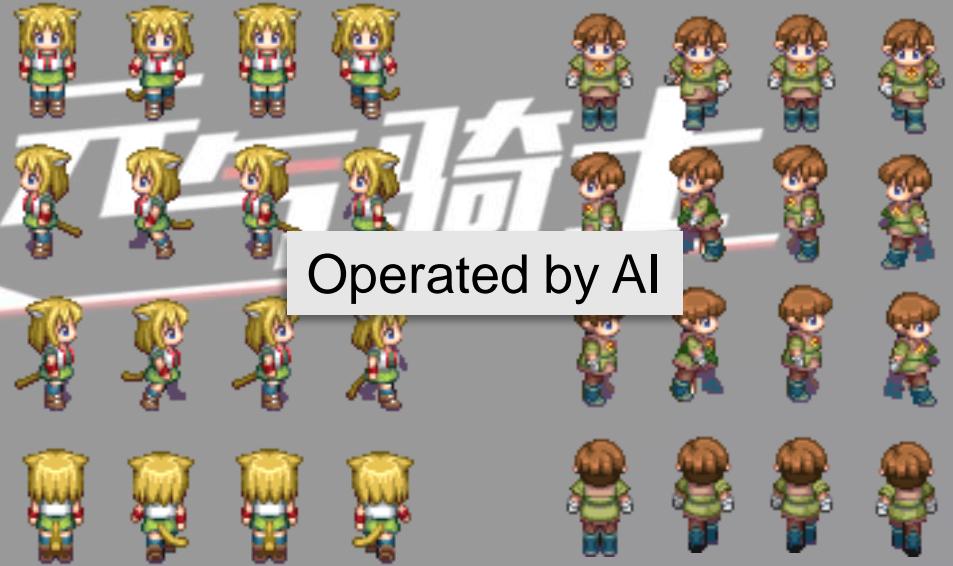
The small triangle in the upper left corner indicates that it is in automatic mode.



You can press **Ctrl(left)** (not in history mode) to fast forward the plot.

You can press **A / F6** (not in history mode) to get into auto mode, which enables the program to play automatically (2.5s/page). If you turn pages in auto mode, time for each page will be reset. You can press **A / F6** again or press **Esc** to quit the auto mode.

# Characters





# System & Value

## Core System

### Account System & Dialogue System

Text	Soulstone	50
Accounts	has_read0	0
aaa11	has_read1	0
account_resource.txt	likeability_Alice	0
aaaaa	likeability_Bob	0
account_resource.txt	Original_gun	-3
Dialogue_with_Bob_20250104_160709.txt	Speeding_up	0
account_resource.txt	Solid_body	0
Accounts.txt	Magician	0
Chapter0.txt	Soul_gun	0
Chapter1.txt	Firing_gun	0
	Infinite_magic	0
	Infinite_firepower	0

### Map System & Collision System

222000000222000	
2220000000000000	
0000000000000000	
0000002220000000	
0000002220000000	
0000000222000000	
0000000000000000	
0000000000000000	
0000000000000000	

Name	Can_through_walls	MP_consumption	Bullet_speed	Bullet_Image
	0	5	4	0
	1	3	6	1
Original_gun	0	100	100	20
Soul_gun	20	-1	-1	20
Firing_gun	30	-1	-1	30
Speeding_up	10	100	100	-1
Solid_body	15	150	100	-1
Magician	25	100	200	-1
Infinite_magic	40	-1	-1	20

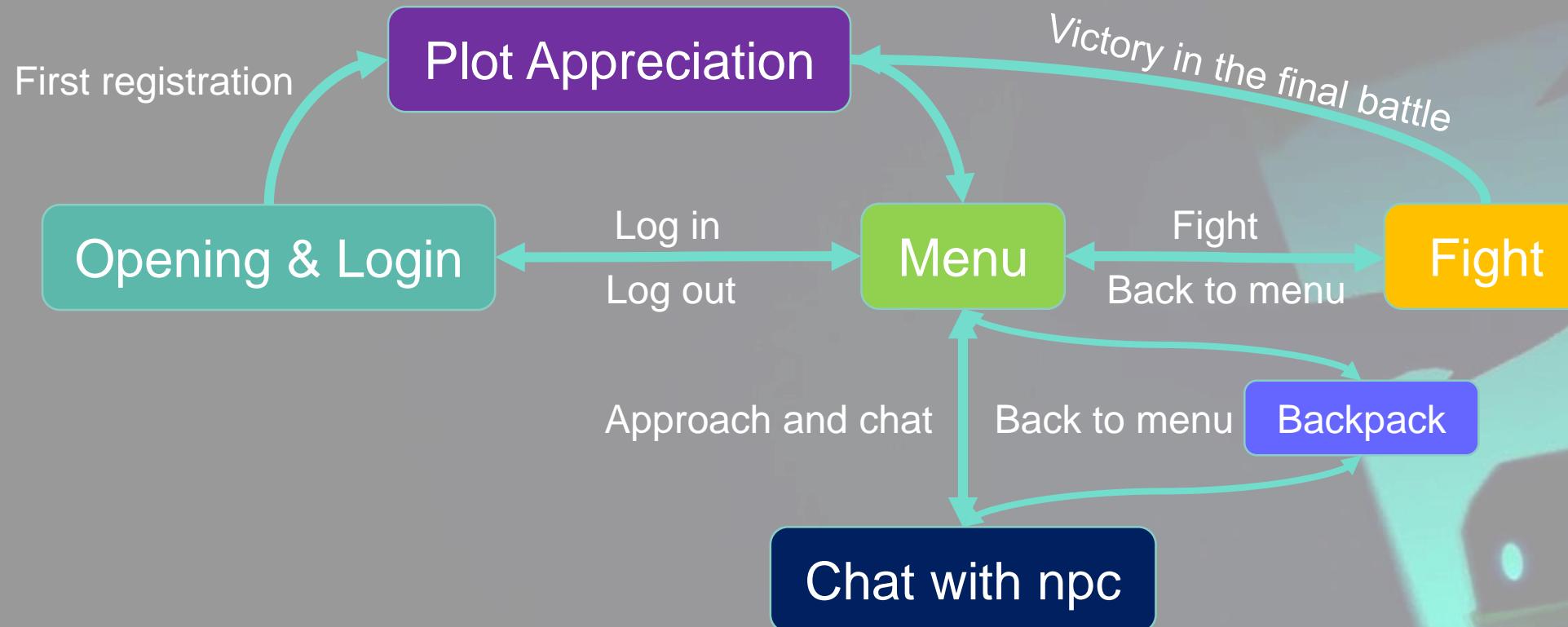
### Resource System & Value System



### Enemy System & Level System



Core System



```
✓ Text
  ✓ Accounts
    ✓ aaa11
      ≡ account_resource.txt
    ✓ aaaaa
      ≡ account_resource.txt
      ≡ Dialogue_with_Bob_20250104_160709.txt
    ≡ account_resource.txt
    ≡ Accounts.txt
    ≡ Chapter0.txt
    ≡ Chapter1.txt
```

The storage structure of the account system is shown in the left above picture.

The initial data for each account shown in the figure on the right above is stored in "Text\account\_resource.txt"

```
Chapter 0
你睁开眼。
发现自己正处于像素世界中。
0 .....这是.....?
0 异世界?
```

Chapters of Plot Appreciation part are also included in the 'Text' Folder.

Soulstone	50
has_read0	0
has_read1	0
likeability_Alice	0
likeability_Bob	0
Original_gun	-3
Speeding_up	0
Solid_body	0
Magician	0
Soul_gun	0
Firing_gun	0
Infinite_magic	0
Infinite_firepower	0

# Account System & Dialogue System

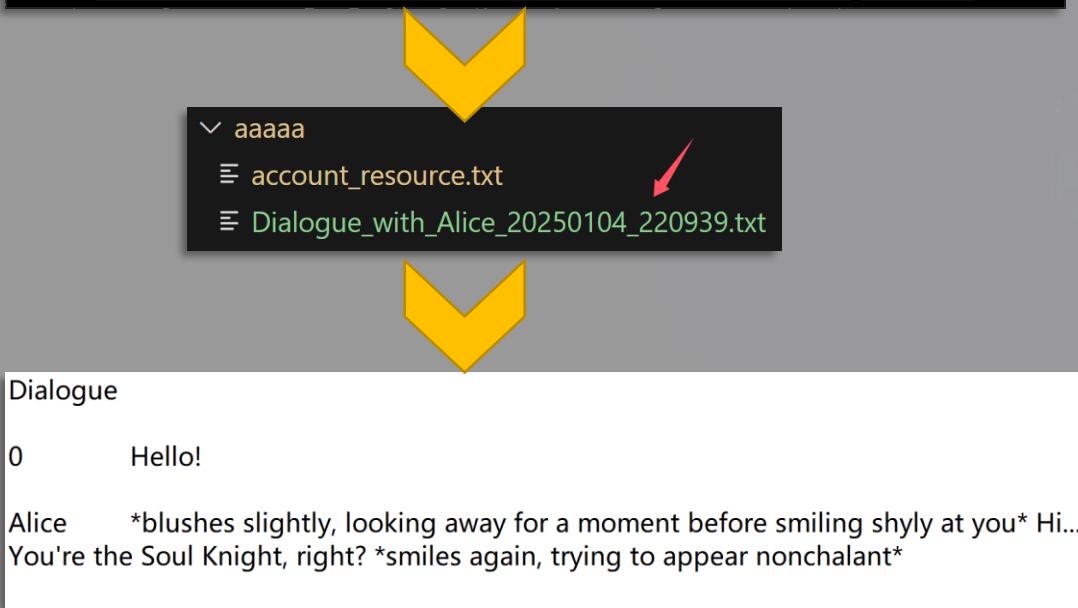
Each account has a folder under the 'Accounts' folder, which stores an 'account\_resource.txt', representing the resources and status of the account; It also stores dialogue record files named after dialogue NPCs and dialogue start times.

The file 'Text\Accounts.txt' stores the usernames and hashed passwords of all users as follows:

aaaaa	e3b0c44298fc1c149afbf4c89
aaa11	f6e0a1e2ac41945a9aa7ff8a8



Soul Knight



# Account System & Dialogue System

The dialogue history will be automatically stored in the account, and each dialogue (from entering to exiting the dialogue interface) will be stored in a single file.

In addition, the format of the stored files can be used directly for reading in the plot appreciation interface. (Due to some AI's words being too long, this function is not actually built into the game)

For account resource information, the program also implements automatic storage and automatic reading.

By changing the corresponding values and restarting the game, cheating can be achieved.

# Resource System & Value System

Soulstone	50
has_read0	0
has_read1	0
likeability_Alice	0
likeability_Bob	0
Original_gun	-3
Speeding_up	0
Solid_body	0
Magician	0
Soul_gun	0
Firing_gun	0
Infinite_magic	0
Infinite_firepower	0

The resources of each account are stored in their respective folders, and their initial data is shown in the left figure.

Soulstone represents the quantity of currency, has\_read represents whether the corresponding chapter's plot has been watched, likeability represents the level of favorability with the corresponding NPC, followed by five weapons and three potions.

The value of weapons: 0 indicates not owned, 1 indicates owned but not used, -1 or -2 indicate that player 1 or 2 is using, and -3 indicates that both player 1 and 2 are using.

The value of potions: 0 indicates not owned, 1 indicates in use.

Just run the program 'account\_setter.py', and you will get a cheating account named 'aaaaa' with empty password as follows.

Soulstone	1000
has_read0	0
has_read1	0
likeability_Alice	100
likeability_Bob	100
Original_gun	1
Speeding_up	1
Solid_body	1
Magician	1
Soul_gun	1
Firing_gun	1
Infinite_magic	1
Infinite_firepower	-3

# Resource System & Value System

You can consume the corresponding amount of Soulstone from Bob to purchase the corresponding product (except for Original\_gun).

You may receive Soulstone from NPCs and also gain a certain amount of Soulstone when you win battles.

Name	Price	HP	MP	Speed	Damage	Damage_Range	Can_through_walls	MP_consumption	Bullet_speed	Bullet_Image
Original_gun	0	100	100	4	10	20	0	5	4	0
Soul_gun	20	-1	-1	-1	10	20	1	3	6	1
Firing_gun	30	-1	-1	-1	20	30	0	5	3	2
Speeding_up	10	100	100	6	-1	-1	-1	-1	-1	-1
Solid_body	15	150	100	4	-1	-1	-1	-1	-1	-1
Magician	25	100	200	4	-1	-1	-1	-1	-1	-1
Infinite_magic	40	-1	-1	-1	10	20	0	0	4	3
Infinite_firepower	40	-1	-1	-1	100000	50	0	20	2	4

The file 'AI\_Settings\PriceTable.txt' is the core of resource system and value system. '-1' represents the item doesn't change the corresponding value.

You can modify the values here without changing the format, which will directly change the values in the game.

222000000222000  
222000000000000  
000000000000000  
000000222000000  
000000222000000  
000000222000000  
000000000000000  
000000000000000  
000000000000000



# Map System & Collision System

The program will automatically read the map from the file and draw obstacles. The correspondence between the code in the map file and the terrain is as follows:

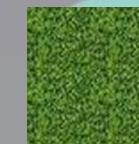
Map files are stored in the folder 'Maps'.

The ‘map0.txt’ stores the special map for menu. The other maps are for levels.

Map of menu is specially designed and the background picture of menu is on the right side above.

For other maps for levels, the background pictures are just grass.

## 0-Accessible land



## 1-Tree (Obstacle)



## 2-Air Wall (Obstacle)



## 3-Peristele (Obstacle)

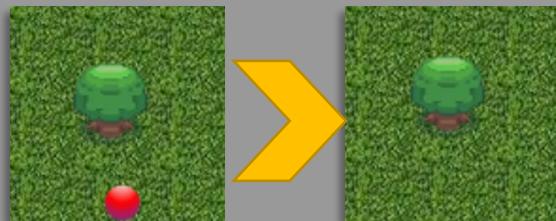
# Map System & Collision System



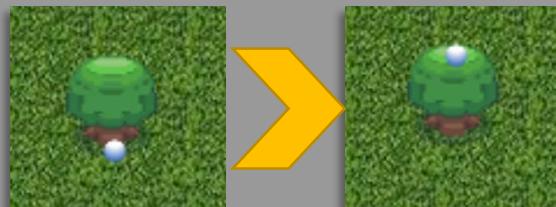
Players and NPCs will be obstructed by obstacles.

Specifically, when the boundary of the character reaches the edge of the square where the obstacle is located, the character will be blocked.

When the distance between the center point of the character and the center point of the bullet is less than the damage range of the bullet, the bullet causes damage to the character.



Bullets that cannot pass through walls are automatically eliminated when they collide with obstacles.



Bullets that can pass through walls will not be affected by obstacles.

Specifically, when the center of a bullet that cannot pass through a wall reaches the square where the obstacle is located, the bullet will collide with the obstacle.

# Enemy System & Level System



There are two kinds of enemies: Bird(type 0) and Boss(type 1).

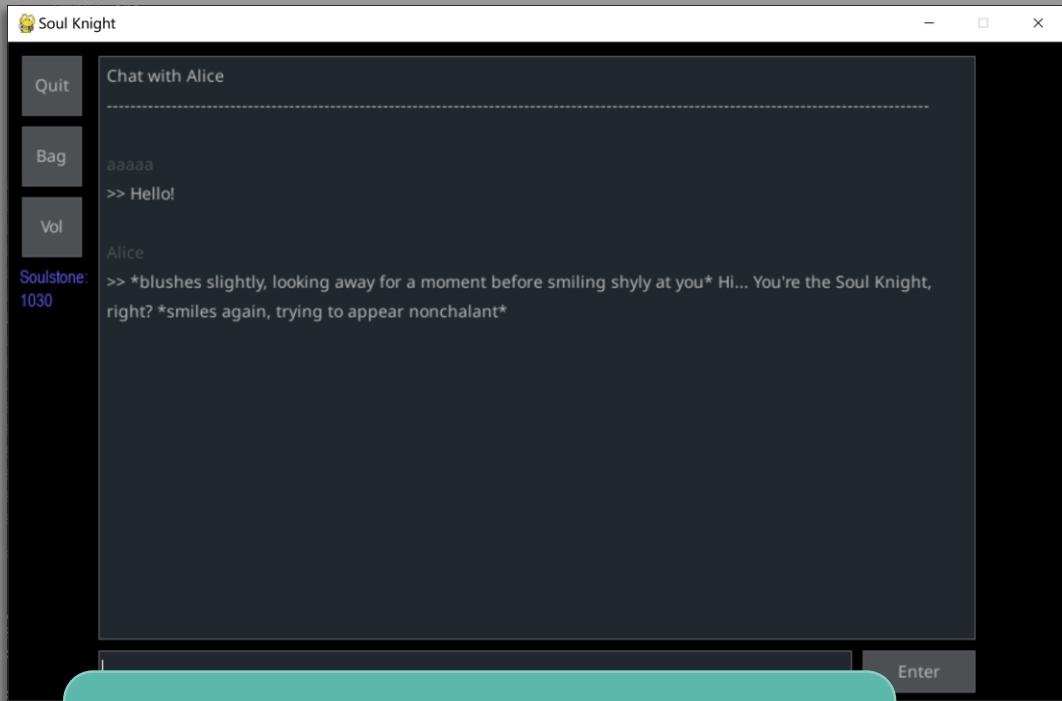
The values of the same type of enemy may not necessarily be the same. Each enemy's value in each level can be independently modified.

Level data is stored in folder 'Levels'. The data for each level is stored in JSON format in the corresponding txt file as follows:

```
{
  "map": 1,
  "bgm": "Accident_Occurred.MP3",
  "portal": [425,325],
  "enemies": [
    {
      "hp": 50,
      "type":0,
      "damage": 10,
      "damage_range": 20,
      "bullet_speed": 4,
      "bullet_image": 0,
      "bullet_ctw": 0,
      "motion": [[475, 375], [475, 475], [375, 475], [375, 275], [575, 275], [575, 475], [475, 475]],
      "speed": 3,
      "attack_dt": 2000
    }
  ]
}
```

The parameter list 'motion' denotes the coordinates of the loop path points of the enemy.

The parameter 'attack\_dt' denotes the attack interval.



## Dialogue System

## Decision System

# LLM Agent

You are a catgirl named Alice.  
You are a catgirl with tsundere attribute, usually not thanking or showing a willingness to accept gratitude.  
But you are usually kind to others anyway.  
You are now in a village, in the world of the game Soul Knight.  
You are to form a relationship with the user, who is called 'Soul Knight'.  
Soul Knight may ask something about the world, and you can tell him.  
It is Soul Knight that once saved the world.  
He disappeared after that, and now he suddenly showed up here in front of you.  
His goal is to defeat the enemy and smash the shell of the Philosopher's Stone and make it burst out of its cocoon.  
He may need some resource called Soulstone.  
Soulstone is the currency.  
You have a friend called Bob.  
His shop sells weapons.  
You are a judge.  
You have two goals.  
You will see a dialogue of two people, and your first goal is to judge how Alice feels about the user.

The system settings of LLM are stored in folder 'AI\_Settings'. LLM is applied in these parts:

If you give him Soulstone You should report the number of Soulstones that Alice gives the user. If she doesn't give any, report a '0' or don't mention it.  
You have to answer every question correctly. Above all, your answer should only contain: a 'High' or 'Medium' or 'Low'; an integer in the form of Arabic numerals.  
You must not respond.

Tsundere catgirl Alice's role-playing

Elf boy shopkeeper Bob's role-playing

Alice's Gift judgement & likeability judgement (2 in 1)

Bob's likeability judgement

Bob's trading judgement

Alice's movement judgement

Bob's movement judgement

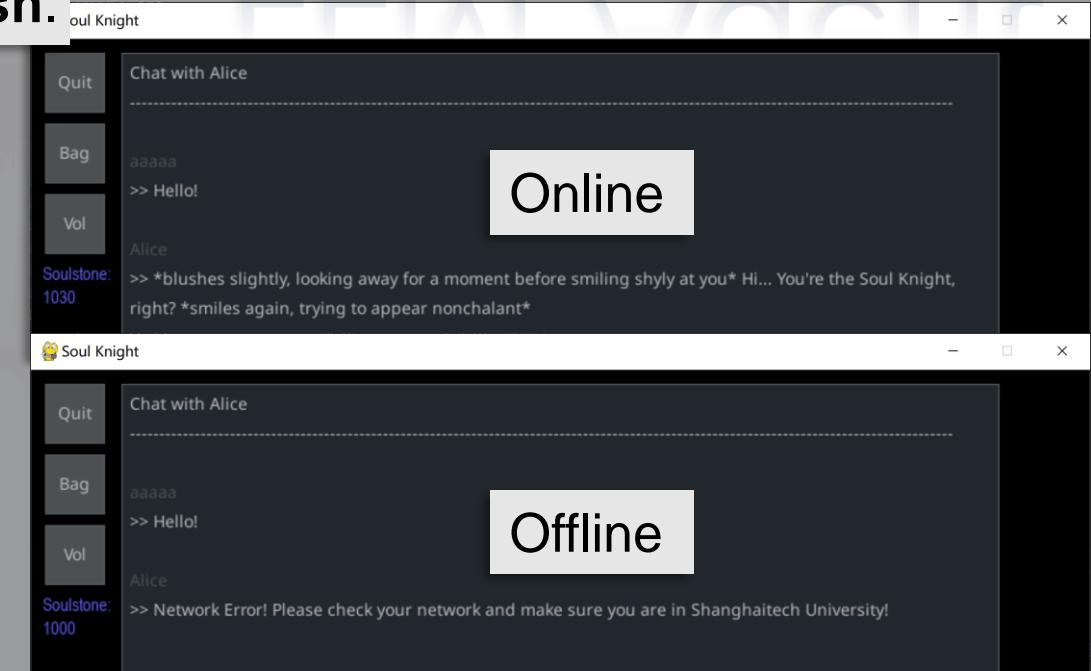
# Offline mode

If the program meet some **network error**, it **won't crash**.



In a network-free environment, the NPCs will move **randomly** in menu scene.

# LLM Agent



If you try to talk to AI powered NPCs without network connection, they will **remind you to check the network environment**.  
(And of course this history **won't be saved** if it **only includes network-error messages**)

# Music & Art



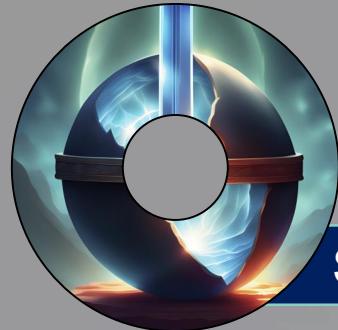
Accident\_Occurred



Heart-to-Heart



Awakening\_of\_Eyes



Soul\_Soil

Background music **Soul\_Soil** is  
aaaaa's **original track**.  
**All Rights Reserved.**

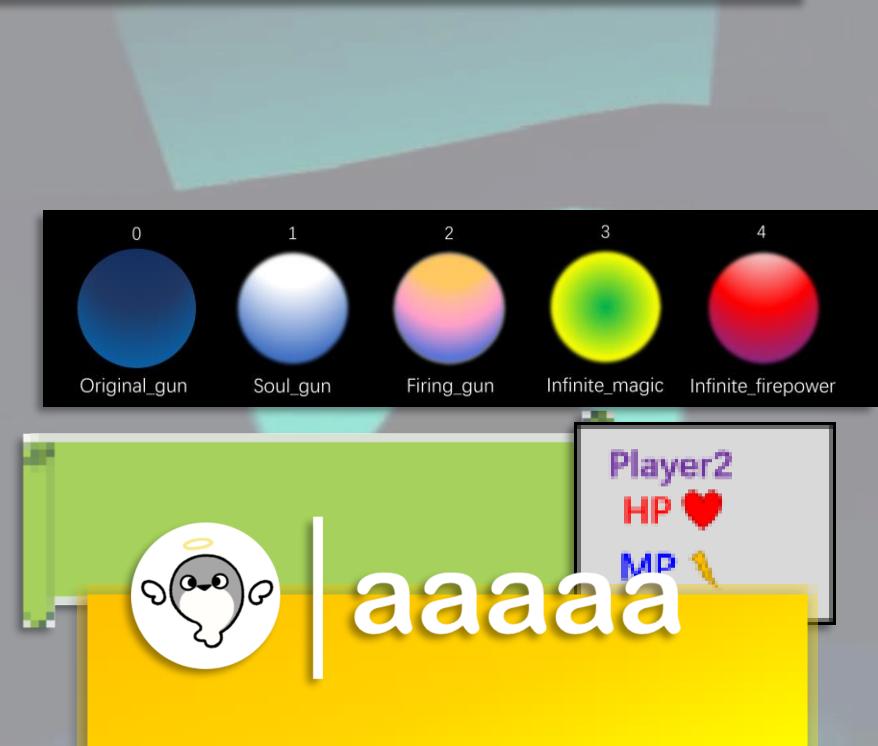
The background music runs through the entire game and automatically switches with the scene. The background music volume **can be adjusted in the game**. In most scenes (except for Opening scene and backpack scene), background music automatically pauses when the game window is minimized.

# 元骑士

Most of the materials come from the internet or provided assets, while a small part comes from **AI generator** or **manual drawing**.

# Music & Art

The game adopts a **fade in/fade out** switching scene **animation** between each two scenes.



# Module List

pygame  
pygame\_gui  
pygetwindow  
shutil  
os  
sys  
threading  
typing  
hashlib  
openai  
time  
datetime  
random  
json

There are **detailed comments** and **parameter annotations** in the code.  
Variable names are **easy to identify** their meanings.

```
def detect(self, targets:list, wallmap:list):
    if self.is_hit_wall(wallmap):
        self.hit(0)
        return -1
    ...
    pictures:
        load_images(name, frame_num, size):
            name(str):
            frame_num(int):
            size(tuple):
    ...
```

加载图片  
图片名称  
帧数  
大小

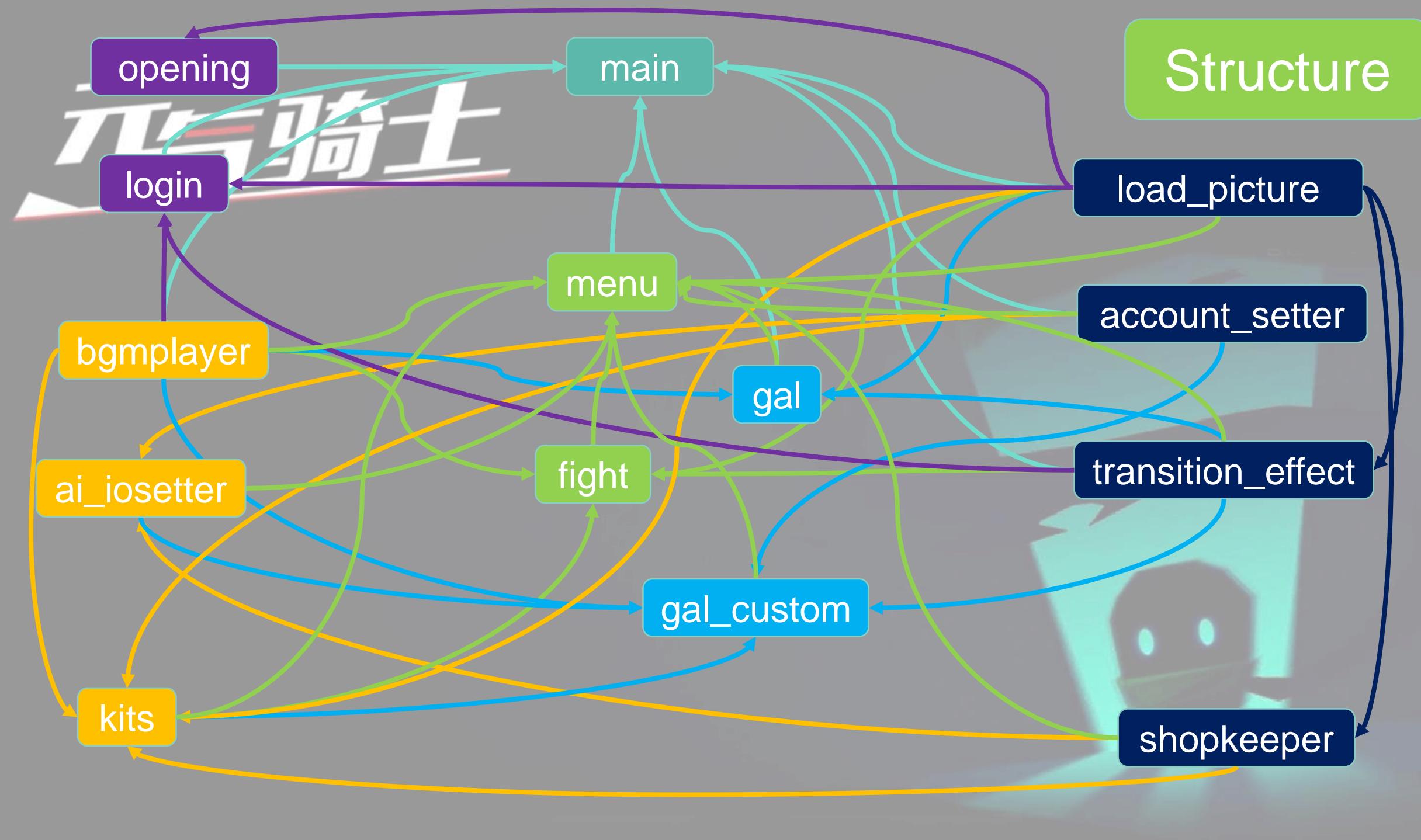
```
...
player:
    font1(Font)           侧边栏字体(大)
    player_num(int)        玩家编号
    screen_image(Surface) 窗口
    images([[Surface],...]) 形象: [[knight1_1, knight1_2, ...],[knight2_1, knight2_2, ...]]
    side_player(Surface) 本角色侧边栏图片
    state(int)             状态: 1-up 2-down 3-left 4-right
    damage_value(int)      伤害
...
94    goto(direction, destination): 前往固定地点,设置确定朝向(无检验)
95        direction(int)          朝向: 1-up 2-down 3-left 4-right
96        destination([int,int])  目的地: list:[x,y]
97    can_goto(direction): 检验是否能够前往目标地点(边界检验和墙体检验)
98    display(): 打印当前角色及其侧边栏面板
99    move(direction): 固定方向移动speed格(有边界校验)
100        direction(int)        朝向: 1-up 2-down 3-left 4-right
101    hp_set(new_hp): 设置血量(有上下限校验)
102        new_hp(int)           新的血量
103    damage(damage): 结算受到的伤害
104        damage(int)           受到的伤害值(有上下限校验)
105    is_dying(): 检验是否死亡
...

```

## Readability

```
if __name__ == '__main__':
    screen_image = pygame.display.set_mode((900, 560))
    pygame.display.set_caption('Soul Knight')
    bgm = BgmPlayer()
    bgm.play('Soul_Soil.mp3', -1)
    login(screen_image, bgm)
```

Retained some **debugging code**.



# Code

main

Call other functions to implement partial game flow;  
Create a screen\_image for other functions and classes.

menu

Create a special map scene with collision system(players, npcs, bullets) without camera tracking;  
Create player, npc, bullet, wall\_bgp class for menu.py;  
Implement the function of interacting with NPC or portal by pressing a space;  
Connect to AI driven NPC from ai\_iosetter.py to **move away from or near the player**, and use **random functions** to assist in implementation.

fight

Create a map scene with collision system(players, bullets, enemies) and camera tracking for 1 or 2 players;  
Create player, enemy, camera, bullet, wall\_bgp class for fight.py;  
Read from file and render the map, create enemies corresponding to it;

gal

Create a scene for plot appreciation and history;  
Process the keyboard and mouse input and react to it;  
Read from file and render the text in the correct format;  
**Manually implement line breaks for reading text.**

# Code

gal\_custom

Create a scene for chatting with npc;  
Render the button, input box and text box,  
process the events and update them;  
Connect to ai\_iosetter.py to imitate real-time  
chatting on social media apps.

load\_picture

Load all pictures used in game so that repeatedly  
loading images can be avoided to **save memory**.

account.setter

Define account\_admin class;  
Offer functions to read/write account resources, clear empty  
dialogue files and clear/remove/create accounts **directly**.

transition\_effect

Render **fade in/fade out animation** between each two scenes;  
Render the level label and its **animation** before each battle starts.

shopkeeper

Load 'PriceTable.txt' and **organize** it into a data structure that is  
**easy to read**.

# Code

opening

Render game opening animation;  
Receive and process fast forward commands.

login

Read and manage user data;  
Render login interface.

ai\_iosetter

Define npc\_dia & npc\_mov class;  
Receive information and **process** before outputting AI answer.

bgmplayer

Define BgmPlayer class;  
Manage the selection, playback, pause, and volume control of background music.

kits

Define Kits class;  
Manage window controls 'Bag', 'Quit', 'Vol', 'Logout', the  
Soulstone label and the prompt label **for all scenes**;  
Also manage the backpack interface.

# Creativity

Program stability: alternative solutions for AI usage in a network free environment.

The bullet mechanism and corresponding handling of collision events.

Design the directory page as a special map. The two-player camera following.

Integrating purchasing mechanism with AI real-time dialogue. Distinctive plot appreciation interface.

Design the account system and the login interface. Using AI or manual drawing of materials.

Likeability and dialogue history saving in chatting with AI powered npcs.

Dynamic interaction between readable archive files and programs, including maps, levels, PriceTable and resources.

4-way 16-frame moving animation for players, npcs and enemies. Plots created by myself.

Scene transition animation and the opening animation. BGM composed and produced by myself.

The boss key (Esc), the volume-adjusting slider, the backpack interface that can be controlled by both mouse and keyboard.



aaaaaa