

Dungeons and Demons

Играта е направена со помош на Pygame и е инспирирана од игри како Diablo, Doom, Path of Exile.

Играта се игра така што се управува со карактер кој е нападнат од други карактери, и треба да собори 10 од нив за да помине кон следното ниво. Како се продолжува на секое следно ниво, негативците стануваат побрзи, поголеми и почесто и побрзо се појавуваат и напаѓаат. Играта има вкупно 10 нивоа, секое ниво со различни негативци.

Како е изработена играта?

1. **Иницијализација на Pygame** – Поставување на прозорец, фрејмрејт и основни параметри.
2. **Вчитување ресурси** – Графика (позадина, играч, непријатели, напади) и аудио (музика, звуци).
3. **Главна јамка** – Управува со сите акции:
 - Движење на играчот
 - Напади
 - Појава и движење на демони
 - Судири (удар на играчот или уништување на непријател)
 - Напредување низ нивоа или завршување на играта
4. **Функции за динамична тежина** – Со секое ниво, демоните стануваат побрзи, поголеми и потешки за убивање.
5. **Прикажување на различни екрани** – Интро, транзиција меѓу нивоа, победа или пораз.

Функционалности на играта

- **Движење на играчот** (стрелки на тастатура)
- **Напад со магија** (SPACE копче)
- **Појава на демони** – Секое ниво има 10 демони
- **Судири** – Ако демонот го допре играчот, тој губи здравје

- **Зголемување на тежината** – Со секое ниво демони стануваат побрзи и појаки
- **Победа и пораз** – Победуваш ако преживееш 10 нивоа, губиш ако ти се намали здравјето на 0
- **Фон музика и звучни ефекти**

1. Иницијализација на Pygame и дефинирање на константи

Во почетокот, се иницијализира Pygame и се поставуваат основни параметри за играта:

- **Резолуција:** 1280x720
- **FPS:** 60 (за мазно движење)
- **Број на демони по ниво:** 10
- **Почетни вредности за големина и брзина на непријателите, како и брзина на играчот.**

2. Вчитување и поставување на графики и звуци

Сите слики за позадината, играчот, нападите, непријателите, текстовите за нивоа и звукот се вчитуваат тука.

```
background = pygame.image.load(r"Background.png")
```

```
background = pygame.transform.scale(background, (WIDTH, HEIGHT))
```

```
player_img = pygame.image.load(r"Necromancer.png")
```

```
player_img = pygame.transform.scale(player_img, (80, 80))
```

```
attack_img = pygame.image.load(r"Attack.png")
```

```
attack_img = pygame.transform.scale(attack_img, (40, 40))
```

```
music = r"Background Music.mp3"
```

Звучна поддршка

Фон музиката се вчитува и пушта во бесконечна репродукција со намален волумен.

```
pygame.mixer.music.load(music)
pygame.mixer.music.set_volume(0.2)
pygame.mixer.music.play(-1)
```

3. Функции за прикажување на различни екрани

Прикажување на интро, победа или пораз

Оваа функција се користи за прикажување на црн екран со дадена слика (интро, победа или пораз), а потоа пауза од 3 секунди.

```
def show_screen(image):
    screen.fill((0, 0, 0))
    image_rect = image.get_rect(center=(WIDTH // 2, HEIGHT // 2))
    screen.blit(image, image_rect.topleft)
    pygame.display.flip()
    time.sleep(3)
```

4. Математички функции за динамичка тежина на играта

Во зависност од нивото, се зголемува брзината, големината и брзината на создавање на непријателите, како и брзината на играчот.

5. Прикажување на транзиција помеѓу нивоа

Кога играчот поминува во следното ниво, се прикажува порака „Get Ready for Level X“, а потоа одбројува од 3 до 1.

Приказ на позадина и HUD

Во секоја итерација, се прикажува позадината, играчот, тековното ниво и бројот на убиени демони.

6. Движење на играчот

```
keys = pygame.key.get_pressed()
player_speed = get_player_speed(level)
if keys[pygame.K_LEFT] and player.x > 0:
    player.x -= player_speed
if keys[pygame.K_RIGHT] and player.x < WIDTH - player.width:
    player.x += player_speed
if keys[pygame.K_UP] and player.y > 0:
    player.y -= player_speed
if keys[pygame.K_DOWN] and player.y < HEIGHT - player.height:
    player.y += player_speed
```

7. Генерирање на непријатели и нивно движење

Секој непријател постепено се движи кон играчот.

```
for enemy in enemies[:]:
    speed = get_enemy_speed(level)
    if enemy.x < player.x:
        enemy.x += speed
    elif enemy.x > player.x:
        enemy.x -= speed
    if enemy.y < player.y:
        enemy.y += speed
    elif enemy.y > player.y:
        enemy.y -= speed
```

8. *Напад на непријателите*

Ако нападот (проектилот) погоди непријател, тој исчезнува.

```
if attack:
    attack.y -= attack_speed
    if attack.y < 0:
        attack = None

for enemy in enemies[:]:
    if attack and attack.colliderect(enemy):
        enemies.remove(enemy)
        killed_enemies += 1
        attack = None
```

9. *Завршување на играта*

Откако играчот ќе излезе од играта, *pygame.quit()* ја затвора програмата.

Референци

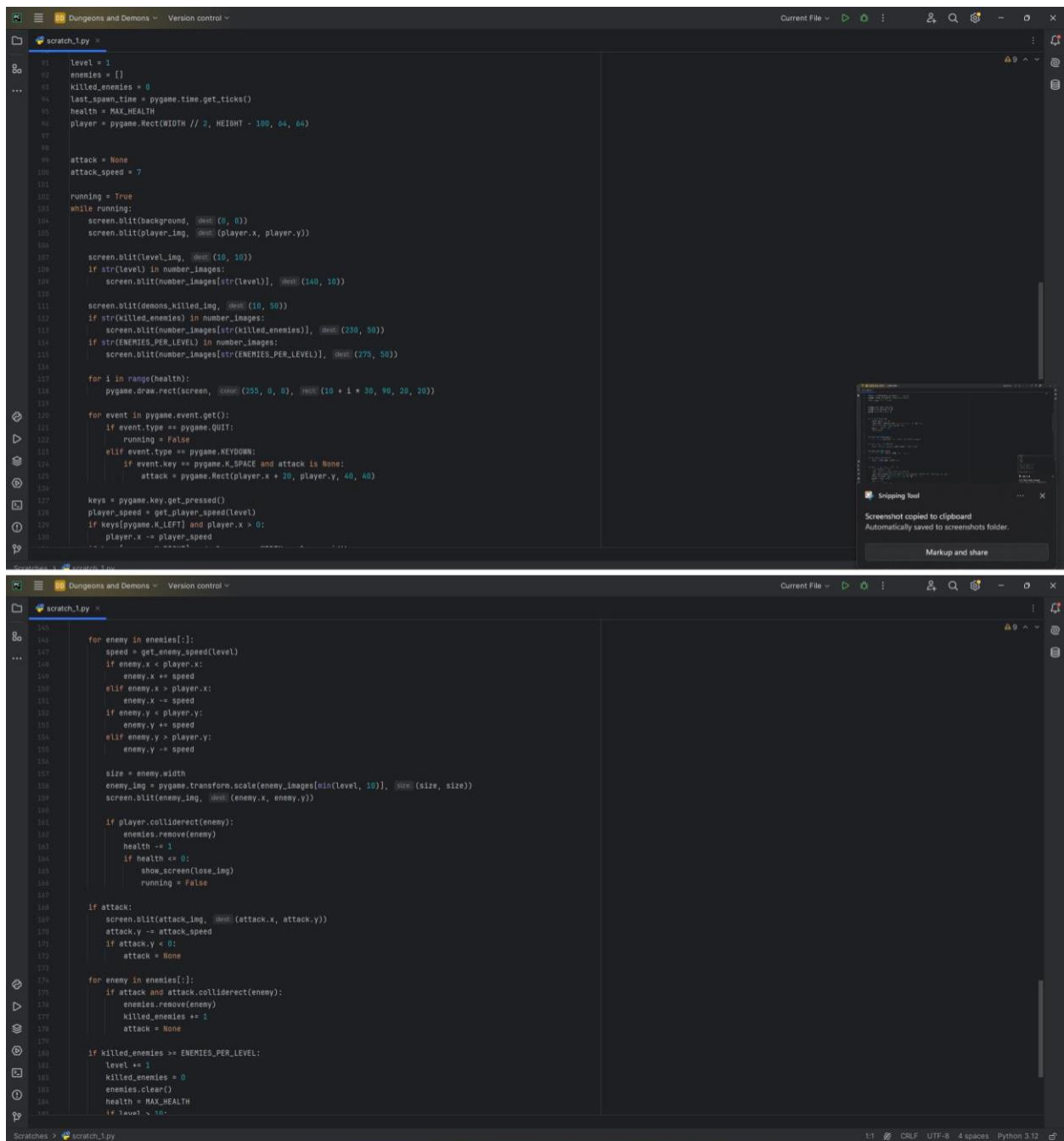
1. <https://www.reddit.com/search/?q=rpg+pygame&cld=fb03b8a2-aa3f-44a7-83f6-bbee35d78446&ild=79eeced5-0cce-407a-aa3f-47bafa1b6d75>
2. https://craftpix.net/freebies/?srltid=AfmBOooAAST_F7hiaymwVBYBzlHoyYIGU3E8i6_LGz0P5ZXAieAECO9j
3. <https://github.com/>
4. <https://www.fontbolt.com/font/diablo-font/>

```
Dungeons and Demons - Version control - Current File - Python 3.12

scratch.py
1 import pygame
2 import random
3 import time
4
5 pygame.init()
6
7 WIDTH, HEIGHT = 1280, 720
8 FPS = 60
9 INITIAL_SPAWN_INTERVAL = 15000
10 ENEMIES_PER_LEVEL = 10
11 INITIAL_ENEMY_SIZE = 44
12 INITIAL_ENEMY_SPEED = 1
13 PLAYER_SPEED = 4
14 MAX_HEALTH = 5
15
16
17 background = pygame.image.load(r"C:\Users\B MAN\Desktop\Dungeons and Demons\Background.png")
18 background = pygame.transform.scale(background, (WIDTH, HEIGHT))
19 player_img = pygame.image.load(r"C:\Users\B MAN\Desktop\Dungeons and Demons\Recromancer.png")
20 player_img = pygame.transform.scale(player_img, (80, 80))
21 attack_img = pygame.image.load(r"C:\Users\B MAN\Desktop\Dungeons and Demons\Attack.png")
22 attack_img = pygame.transform.scale(attack_img, (40, 40))
23 music = r"C:\Users\B MAN\Desktop\Dungeons and Demons\Background Music.mp3"
24 intro_img = pygame.image.load(r"C:\Users\B MAN\Desktop\Dungeons and Demons\Intro.png")
25 intro_img = pygame.transform.scale(intro_img, (1200, 100))
26 lose_img = pygame.image.load(r"C:\Users\B MAN\Desktop\Dungeons and Demons\Loss.png")
27 win_img = pygame.image.load(r"C:\Users\B MAN\Desktop\Dungeons and Demons\Win.png")
28
29 enemy_images = [pygame.image.load(r"C:\Users\B MAN\Desktop\Dungeons and Demons\Level\{i}.png") for i in range(1, 11)]
30
31
32 level_img = pygame.image.load(r"C:\Users\B MAN\Desktop\Dungeons and Demons\Level.png")
33 level_img = pygame.transform.scale(level_img, (120, 32))
34 demons_killed_img = pygame.image.load(r"C:\Users\B MAN\Desktop\Dungeons and Demons\Demons Killed.png")
35 demons_killed_img = pygame.transform.scale(demons_killed_img, (200, 32))
36
37
38 number_images = [str(i) for i in range(1, 11)]
39 number_images = [pygame.transform.scale(pygame.image.load(r"C:\Users\B MAN\Desktop\Dungeons and Demons\{i}.png"), (32, 32)) for i in range(11)]
40
41
Scratches - scratch.py 1:1 CRLF UTF-8 4 spaces Python 3.12
```

```
Dungeons and Demons - Version control - Current File - Python 3.12

scratch.py
42
43 screen = pygame.display.set_mode(WIDTH, HEIGHT)
44 pygame.display.set_caption("Dungeons and Demons")
45 clock = pygame.time.Clock()
46
47
48 pygame.mixer.music.load(music)
49 pygame.mixer.music.set_volume(0.2)
50 pygame.mixer.music.play(-1)
51
52
53 def show_screen(image):
54     screen.fill((0, 0, 0))
55     image_rect = image.get_rect(center=(WIDTH // 2, HEIGHT // 2))
56     screen.blit(image, image_rect.topleft)
57     pygame.display.flip()
58     time.sleep(3)
59
60
61 def get_enemy_speed(level):
62     return min(PLAYER_SPEED - 1, INITIAL_ENEMY_SPEED + level * 0.5)
63
64
65 def get_spawn_interval(level):
66     return max(2000, INITIAL_SPAWN_INTERVAL - level * 1000)
67
68
69 def get_enemy_size(level):
70     return min(120, INITIAL_ENEMY_SIZE + level * 5)
71
72
73 def get_player_speed(level):
74     return PLAYER_SPEED + level * 0.2
75
76
77 def display_level_transition(level):
78     screen.fill((0, 0, 0))
79     font = pygame.font.Font(None, 48)
80     text = font.render(f"Get Ready for Level {level}", True, (255, 255, 255))
81     screen.blit(text, (WIDTH // 2 - 150, HEIGHT // 2 - 20))
82     pygame.display.flip()
83     time.sleep(3)
84     for i in range(1, 0, -1):
85         screen.fill((0, 0, 0))
86
Snipping tool
Screenshot copied to clipboard
Automatically saved to screenshots folder.
Markup and share
```



Изработиле:

Петар Стојановски 223204

Дамир Бачевац 223026