

Рівненський державний гуманітарний університет

Кафедра ІКТ та МБІ

Технічна документація для проекту «Zmiika_2.0»

Виконали:

студентка 2 курсу

факультету математики та інформатики

групи ЦТ-21

Алконова Діана,

студент 1 курсу

факультету математики та інформатики

групи ЦТ-11

Бондарук Валентин

Рівне-2024

Зміст

1. Вступ.....	2
2. Необхідні технології	2
3. Структура проекту	2
5. Вимоги до системи.....	6
6. Автори	6

1. Вступ

Проект "Змійка" — це реалізація класичної гри, де гравець керує змійкою, яка рухається по екрану, збираючи їжу і уникаючи зіткнень зі стінами або власним тілом. Гра розроблена на Python з використанням бібліотеки Pygame.

2. Необхідні технології

- **Мова програмування:** Python
- **Бібліотека для створення графічного інтерфейсу:** Pygame
- **Інструменти для управління проектом:**
 - Git для контролю версій
 - GitHub для зберігання репозиторію та спільної роботи учасників команди
- **Інструменти для управління задачами:**
 - Trello для створення дошки для спринтів, User Stories, завдань і підзадач
- **Середовище розробки:**
 - Visual Studio Code

3. Структура проекту

```
import random, pygame, sys
from pygame.locals import *

FPS = 15
WINDOWWIDTH = 640
WINDOWHEIGHT = 480
CELLSIZE = 20

assert WINDOWWIDTH % CELLSIZE == 0, "Window width must be a multiple of cell size."
assert WINDOWHEIGHT % CELLSIZE == 0, "Window height must be a multiple of cell size."

CELLWIDTH = int(WINDOWWIDTH / CELLSIZE)
CELLHEIGHT = int(WINDOWHEIGHT / CELLSIZE)

WHITE = (255, 255, 255)
BLACK = (0, 0, 0)
RED = (255, 0, 0)
```

```

GREEN = (0, 255, 0)
DARKGREEN = (0, 155, 0)
DARKGRAY = (40, 40, 40)
BGCOLOR = BLACK

UP = 'up'
DOWN = 'down'
LEFT = 'left'
RIGHT = 'right'

HEAD = 0

def main():
    global FPSCLOCK, DISPLAYSURF, BASICFONT
    pygame.init()
    FPSCLOCK = pygame.time.Clock()
    DISPLAYSURF = pygame.display.set_mode((WINDOWWIDTH, WINDOWHEIGHT))
    BASICFONT = pygame.font.Font('freesansbold.ttf', 18)
    pygame.display.set_caption('Zmiika 2.0')

    showStartScreen()
    while True:
        runGame()
        showGameOverScreen()

def runGame():
    startx = random.randint(5, CELLWIDTH - 6)
    starty = random.randint(5, CELLHEIGHT - 6)
    wormCoords = [{ 'x': startx, 'y': starty },
                  { 'x': startx - 1, 'y': starty },
                  { 'x': startx - 2, 'y': starty }]
    direction = RIGHT
    apple = getRandomLocation()

    while True:
        for event in pygame.event.get():
            if event.type == QUIT:
                terminate()
            elif event.type == KEYDOWN:
                if (event.key == K_LEFT or event.key == K_a) and direction !=
RIGHT:
                    direction = LEFT
                elif (event.key == K_RIGHT or event.key == K_d) and direction !=
LEFT:
                    direction = RIGHT
                elif (event.key == K_UP or event.key == K_w) and direction != DOWN:
                    direction = UP
                elif (event.key == K_DOWN or event.key == K_s) and direction != UP:
                    direction = DOWN
                elif event.key == K_ESCAPE:
                    terminate()

            if wormCoords[HEAD]['x'] == -1 or wormCoords[HEAD]['x'] == CELLWIDTH or \

```

```

        wormCoords[HEAD]['y'] == -1 or wormCoords[HEAD]['y'] == CELLHEIGHT:
            return
    for wormBody in wormCoords[1:]:
        if wormBody['x'] == wormCoords[HEAD]['x'] and wormBody['y'] ==
wormCoords[HEAD]['y']:
            return

    if wormCoords[HEAD]['x'] == apple['x'] and wormCoords[HEAD]['y'] ==
apple['y']:
        apple = getRandomLocation()
    else:
        del wormCoords[-1]

    if direction == UP:
        newHead = {'x': wormCoords[HEAD]['x'], 'y': wormCoords[HEAD]['y'] - 1}
    elif direction == DOWN:
        newHead = {'x': wormCoords[HEAD]['x'], 'y': wormCoords[HEAD]['y'] + 1}
    elif direction == LEFT:
        newHead = {'x': wormCoords[HEAD]['x'] - 1, 'y': wormCoords[HEAD]['y']}
    elif direction == RIGHT:
        newHead = {'x': wormCoords[HEAD]['x'] + 1, 'y': wormCoords[HEAD]['y']}
    wormCoords.insert(0, newHead)
    DISPLAYSURF.fill(BG_COLOR)
    drawGrid()
    drawWorm(wormCoords)
    drawApple(apple)
    drawScore(len(wormCoords) - 3)
    pygame.display.update()
    FPSCLOCK.tick(FPS)

def drawPressKeyMsg():
    pressKeySurf = BASICFONT.render('Press a key to play.', True, DARKGRAY)
    pressKeyRect = pressKeySurf.get_rect()
    pressKeyRect.topleft = (WINDOWWIDTH - 200, WINDOWHEIGHT - 30)
    DISPLAYSURF.blit(pressKeySurf, pressKeyRect)

def checkForKeyPress():
    if len(pygame.event.get(QUIT)) > 0:
        terminate()
    keyUpEvents = pygame.event.get(KEYUP)
    if len(keyUpEvents) == 0:
        return None
    if keyUpEvents[0].key == K_ESCAPE:
        terminate()
    return keyUpEvents[0].key

def showStartScreen():
    titleFont = pygame.font.Font('freesansbold.ttf', 100)
    titleSurf1 = titleFont.render('Zmiika 2.0', True, WHITE, DARKGREEN)
    titleSurf2 = titleFont.render('Zmiika 2.0', True, GREEN)

    degrees1 = 0
    degrees2 = 0

```

```

while True:
    DISPLAYSURF.fill(BG_COLOR)
    rotatedSurf1 = pygame.transform.rotate(titleSurf1, degrees1)
    rotatedRect1 = rotatedSurf1.get_rect()
    rotatedRect1.center = (WINDOWWIDTH / 2, WINDOWHEIGHT / 2)
    DISPLAYSURF.blit(rotatedSurf1, rotatedRect1)

    rotatedSurf2 = pygame.transform.rotate(titleSurf2, degrees2)
    rotatedRect2 = rotatedSurf2.get_rect()
    rotatedRect2.center = (WINDOWWIDTH / 2, WINDOWHEIGHT / 2)
    DISPLAYSURF.blit(rotatedSurf2, rotatedRect2)

    drawPressKeyMsg()

    if checkForKeyPress():
        pygame.event.get()
        return
    pygame.display.update()
    FPSCLOCK.tick(FPS)
    degrees1 += 3
    degrees2 += 7

def terminate():
    pygame.quit()
    sys.exit()

def getRandomLocation():
    return {'x': random.randint(0, CELLWIDTH - 1), 'y': random.randint(0,
CELLHEIGHT - 1)}

def showGameOverScreen():
    gameOverFont = pygame.font.Font('freesansbold.ttf', 150)
    gameSurf = gameOverFont.render('Game', True, WHITE)
    overSurf = gameOverFont.render('Over', True, WHITE)
    gameRect = gameSurf.get_rect()
    overRect = overSurf.get_rect()
    gameRect.midtop = (WINDOWWIDTH / 2, 10)
    overRect.midtop = (WINDOWWIDTH / 2, gameRect.height + 10 + 25)

    DISPLAYSURF.blit(gameSurf, gameRect)
    DISPLAYSURF.blit(overSurf, overRect)
    drawPressKeyMsg()
    pygame.display.update()
    pygame.time.wait(500)
    checkForKeyPress()

    while True:
        if checkForKeyPress():
            pygame.event.get()
            return

def drawScore(score):
    scoreSurf = BASICFONT.render('Score: %s' % (score), True, WHITE)

```

```

scoreRect = scoreSurf.get_rect()
scoreRect.topleft = (WINDOWWIDTH - 120, 10)
DISPLAYSURF.blit(scoreSurf, scoreRect)

def drawWorm(wormCoords):
    for coord in wormCoords:
        x = coord['x'] * CELLSIZE
        y = coord['y'] * CELLSIZE
        wormSegmentRect = pygame.Rect(x, y, CELLSIZE, CELLSIZE)
        pygame.draw.rect(DISPLAYSURF, DARKGREEN, wormSegmentRect)
        wormInnerSegmentRect = pygame.Rect(x + 4, y + 4, CELLSIZE - 8, CELLSIZE -
8)
        pygame.draw.rect(DISPLAYSURF, GREEN, wormInnerSegmentRect)

def drawApple(coord):
    x = coord['x'] * CELLSIZE
    y = coord['y'] * CELLSIZE
    appleRect = pygame.Rect(x, y, CELLSIZE, CELLSIZE)
    pygame.draw.rect(DISPLAYSURF, RED, appleRect)

def drawGrid():
    for x in range(0, WINDOWWIDTH, CELLSIZE):
        pygame.draw.line(DISPLAYSURF, DARKGRAY, (x, 0), (x, WINDOWHEIGHT))
    for y in range(0, WINDOWHEIGHT, CELLSIZE):
        pygame.draw.line(DISPLAYSURF, DARKGRAY, (0, y), (WINDOWWIDTH, y))

if __name__ == '__main__':
    main()

```

5. Вимоги до системи

- **Операційна система:** Windows 10, 11
- **Python:** версія 3.6 або новіша
- **Pygame:** версія 1.9.6 або новіша

6. Автори

1. Алконова Діана:

- Роль: Дизайнер, Програміст
- E-mail(+github): dianadianadog123@gmail.com(dialkonova)

2. Бондарук Валентин:

- Роль: Програміст
- E-mail(+github): bondaruk.valentyn.poct23@rshu.edu.ua (valentynbondaruk)

