

นางสาวสิริกาญจน์ ใหม่กุญจน์ ๖๖๑๓๑๐๐๖๔

"River Rush!!"

คำอธิบายการทำงาน = River Rush เป็นโค้ดภาษา C++ ที่สามารถเรียกใช้ API จากภาษา Python ได้ ภาษา Python จะเป็นภาษาที่เขียนมาเพื่อให้ผู้ใช้งานสามารถเขียนโปรแกรมได้โดยง่าย แต่ภาษา C++ นั้นจะต้องเขียนมาในรูปแบบของข้อความ แต่ภาษา Python นั้นจะมีรูปแบบที่เขียนง่ายและเข้าใจง่ายกว่าภาษา C++ มาก ภาษา Python นี้สามารถทำงานได้รวดเร็วและมีประสิทธิภาพมากกว่าภาษา C++ อย่างมาก

โค้ดภาษา C++ นี้จะถูกเขียนมาด้วยภาษา Python + RSide สำหรับสร้างล่วงติดต่อผู้ใช้ (UI) และภาษา Python นี้จะถูกใช้ในการสร้างเครื่องมือ (Tool) อย่างเช่น Maya หรือ Blender ที่มี GUI ทางด้าน Sublime

River Rush นี้รับการแสดงผลด้วยภาษา Java ซึ่งแสดงผลลัพธ์ของเกมในหน้าจอ Chrome Dino ที่ทุกคนคุ้นเคยแล้ว คุณสามารถเล่นเกมนี้ได้โดยตรงบนเว็บเบราว์เซอร์ เช่น Google Chrome, Mozilla Firefox, Internet Explorer ฯลฯ

ส่วนประกอบหลัก และโครงสร้างของเกม

ส่วนประกอบหลัก - รูปภาพและเสียง: ภาพและเสียงที่จำเป็นสำหรับการสร้างเกม เช่น ภาพน้ำ ภูเขา ต้นไม้ ฯลฯ

1. หน้าจอเมนูหลัก (Menu)

2. หน้าจอเกม (Game)

3. หน้าจอจบเกม (Game Over)

- การควบคุมผู้เล่น: ใช้เมาส์หรือจอยมาควบคุมการเคลื่อนไหว เช่น ไปซ้าย Left และ Right ไปขวา

การทำงาน

- การสร้างตัวละครร้าย: เก็บจับผู้เล่นที่เดินทางมา (หัวใจ 1 ตัว) ก่อนที่จะถูกฆ่า นำกลับบ้าน ยกเว้นเมือง และเมืองน้ำ แต่เมืองน้ำจะหายไปเมื่อผู้เล่นเดินทางผ่านไป

- ระบบเวลา: ใช้ QTimer ล่าช้า 2 ตัว ในการอัปเดตเวลา

1. Game Timer - ทำงานทุก 30 มิลลิวินาที เมื่อผู้เล่นเดินทางผ่านเมืองน้ำ หรือเมืองทุ่มน้ำ ระบบจะเริ่มต้นอัปเดตเวลาอีกครั้ง

2. Spaw Timer : ทำงานทุก 1 วินาที เมื่อผู้เล่นเดินทางผ่านเมืองน้ำ ระบบจะเริ่มต้นอัปเดตเวลาอีกครั้ง

- ระบบตรวจสอบการชน: ตรวจสอบว่าตัวละครของผู้เล่นชนกับตัวละครร้ายหรือไม่ ถ้าชนกัน ผู้เล่นจะถูกฆ่า

សេចក្តីថ្ងៃនេះខ្សែពាណិជ្ជកម្មទាំងទៀត

1. ផែនក្នុងលេខរបាយកំណើនរបស់ខ្លួន នូវកំណើនទៅលើលទ្ធផល។

2. ផែនក្នុងលេខរបាយកំណើនរបស់ខ្លួន នូវកំណើនទៅលើលទ្ធផល +1

គេហទ័រ

- ការដោឡូលេខរបាយកំណើនទៅលើលទ្ធផល: គោលទៅរួមសែរកិច្ចរាយគោលទៅនឹងក្នុងការ

5 ក្រុងការ

- ការដោឡូលេខរបាយកំណើនទៅលើលទ្ធផល: គោលទៅរួមសែរកិច្ចរាយគោលទៅនឹងក្នុងការ

ផែនក្នុងលេខរបាយកំណើនទៅលើលទ្ធផល

- ការដោឡូលេខរបាយកំណើនទៅលើលទ្ធផល: ផែនក្នុងការដោឡូលេខរបាយកំណើនទៅលើលទ្ធផល

(ដែលត្រូវការការពិនិត្យ) និង "ការដោឡូលេខរបាយកំណើនទៅលើលទ្ធផល"

ក្រុងការរួមរាល់ការរបៀប project_v1.py នៅក្នុងឯកសារនេះ និងការរាយការណ៍ការងារប្រចាំថ្ងៃ

សំគាល់ការងារក្នុងការរបៀប

```

from PySide6 import QtWidgets, QtGui, QtCore
from shiboken6 import wrapInstance
import maya.OpenMayaUI as omui
import random
import os
    
```

from PySide6 import QtWidgets, Qt+Widgets, QtCore → ធ្វើការនូវ "QtWidgets" ដើម្បីក្នុង "QtCore" ដើម្បីក្នុងការ

from shiboken6 import wrapInstance → តុកការខ្សែតុកការនូវ "QtCore" ដើម្បីក្នុងការ

import maya.OpenMayaUI as omui → សម្រាប់ការប្រើប្រាស់ក្នុងការរបៀបក្នុង Maya

import random → ឱ្យឱ្យ "តុកការ" អនុញ្ញាតការងារនៃក្នុង Maya និងក្នុង "wrapInstance"

import os → សម្រាប់ការប្រើប្រាស់ក្នុងការរបៀបក្នុងការ

↓ ធ្វើការស្នើសុំ "ការកំណើនទៅលើលទ្ធផល" images/ និង icons/ ក្នុង

(ផ្ទាល់ខ្លួនទៅក្នុង Maya)

def maya_main_window():

main_window_ptr = omui.MQtUtil.mainWindow()

return wrapInstance(int(main_window_ptr), QtWidgets.QWidget)

(ស៊ែរការងារក្នុងការរបៀប)

class project_v1(QtWidgets.QDialog):

(ជូនការងារក្នុងការរបៀប)

def __init__(self, parent=maya_main_window()):

(ការកំណើនទៅលើលទ្ធផល (Game Constants))

self.boat_speed = 20 # ការរៀនការកំណើនទៅលើលទ្ធផល

self.boat_x_start = 300 # តារាងខាងក្រោម X (នាមីម៉ែត្រ) ទីនើនការកំណើនទៅលើលទ្ធផល

self.boat_y_start = 600 # ຕາມເນື່ອງ Y (ໃຈຂອດຕັ້ງ) ເວັບຕົກນອງເວົວ (ກໍາສັ່ງລົງທຶນ)
 self.boat_x = self.boat_x_start # ຕົກປູ່ເກີ່ມຕົກແຍ້ງ X ຢູ່ຄູ່ຫຸນ
 self.boat_y = self.boat_y_start # ຕົກປູ່ເກີ່ມຕົກແຍ້ງ Y ຢູ່ຈຸດັນ

(ຕົກປູ່ເກີ່ມຕົກແຍ້ງ Game Variables)

self.obstacles = [] # List ລາຍການວ່າ ສ້າງວົນເກີ່ມ "Widget ສ້າງກິດຂອງ" ທີ່ຈະ
 self.obstacle_speed = 3 # ກວາມເຮັດວຽກຕົກນອງສັ່ງກິດຂອງ
 self.high_score = 0 # ດະກຳຂອງສູນສຸດ (ເຮັດທຶນ)

(ຕັ້ງດໍາ Path ລົງໂປລັດວ່າ "images" ບັນຍາກິດ)

self.IMAGE_DIR = "C:/Users/nadia/Documents/maya/2026/.../images"

(ຕັ້ງດໍາ Timers ຊົດເຕີກເກມ - ສ້າງວົນໄດ້ລົ່ວມທີ່, ເພີ້ມ, ອຸ່ນການໃຫຍ່)

self.game_timer = QCore.QTimer(self)

(ບໍ່ໄດ້ເຮັດວຽກ (slot) ຊົງກໍາ self.update_game)

self.game_timer.timeout.connect(self.update_game)

▷ (ສ້າງສົ່ງກິດຊ່າງ)

self.spawn_timer = QCore.QTimer(self)

self.spawn_timer.timeout.connect(self.spawn_obstacle)

(ເລີກສົ່ງ self.setup_UI ເຊັ່ນສ້າງນ້ຳຕາມໄປຮັງການ)

self.setup_UI()

(ຜົດຖານ setup_UI (ຕັ້ງດໍາ UI ນັກ)

def setup_UI(self): # ຕັ້ງດໍາ Icon

icon_path = os.path.join(self.IMAGE_DIR, ".../icons", "icons1.png")

ດ້ວຍຫຼາຍກາ

self.setWindowTitle("RIVER RUSH GAME!!!!")

self.setFixedSize(700, 800)

self.setFocusPolicy(Qt.StrongFocus)

ຕັ້ງສະນົມຫຼາ

self.stackedLayout = QtWidgets.QStackedLayout(self)

self.setLayout(self.stackedLayout)

ນັກາມ

```
self.menu_widget = self._create_menu_widget()
self.stacked_layout.addWidget(self.menu_widget)
```

ພັກ

```
self.game_widget = self._create_game_widget()
self.stacked_layout.addWidget(self.game_widget)
```

ວິທີລົງການ

```
self.game_over_widget = self._create_over_widget()
self.stacked_layout.addWidget(self.game_over_widget)
```

ຕັ້ງຕຳອຸນດາເຮືອດັນ ນັກາມ ນັກາມ

```
self.stacked_layout.setCurrentWidget(self.menu_widget)
```

ຜັກນີ້ແຈ້ງ Widget ທີ່ຈະນັກໃຈ “ນັກາມ”

```
def _create_menu_widget(self):
```

```
    widget = QtWidgets.QWidget()
```

```
<    widget.setStyleSheet("background-color: #90EE90;")
```

```
Layout = QtWidgets.QVBoxLayout(widget)
```

```
image_label = QtWidgets.QLabel()
```

```
image_path = os.path.join(self.IMAGE_DIR, "BG1.PNG")
```

```
pixmap = QtGui.QPixmap(image_path)
```

```
Layout.addWidget(image_label)
```

ສັງເກດ

```
start_button = QtWidgets.QPushButton("ເຮືອກສຳເນົາ")
```

```
quit_button = QtWidgets.QPushButton("ອອກ")
```

```
start_button.clicked.connect(self.start_game)
```

```
quit_button.clicked.connect(self.close)
```

```
start_button.setStyleSheet(button_style_green)
```

```
quit_button.setStyleSheet(button_style_red)
```

Layout.addLayout(button_layout)
Layout.addStretch()

return widget

// អង្គភាពនៅលើក្រុងការបង្កើត

def _create_game_widget(self):
 widget=QtWidgets.QWidget()

* ការបង្កើតខ្លួន និងចំណាំ

info_layout=QtWidgets.QHBoxLayout()

self.heart_label=QtWidgets.QLabel("o": 3)

self.high_score_label=QtWidgets.QLabel(f"នូវតម្លៃសរុបគឺ: {self.high_score}")

self.score_label=QtWidgets.QLabel("នូវតម្លៃ: 0")

info_layout.addWidget(self.heart_label)

info_layout.addStretch()

< info_layout.addWidget(self.high_score_label)

info_layout.addSpacing(15)

info_layout.addWidget(self.score_label)

Layout.setLayout(info_layout)

ការរំពឹងដំឡើង

self.boat=QtWidgets.QLabel(self.game_area)

self.boat.setFocusPolicy(QtCore.Qt.NoFocus)

ship_image_path=os.path.join(self.IMAGE_DIR,"ship01.PNG")

self.boat.setPixmap(QtGui.QPixmap(ship_image_path).scaled(100, 100, QtCore.Qt.KeepAspectRatio))

រាយការណ៍ទីនឹងនៃការបង្កើត (នូវ self.game_area)

self.boat.setGeometry(self.boat_x, self.boat_y, 100, 100)

self.boat.setStyleSheet("outline: none; border: none;")

return widget

ផ្លាស់បន្ទាន់ create_game_over_widget "អង្គារលើកវាង"

```
def _create_game_over_widget(self):
    widget = QtWidgets.QWidget()
```

```
game_over_label = QtWidgets.QLabel("||ឆ្វុ||តិច")
```

```
Layout.addWidget(game_over_label)
```

* ស្ម័គ្រមនេះនឹង

```
self.final_score_label = QtWidgets.QLabel("ពាក្យខ្លួនអាចដល់០")
```

```
layout.addWidget(self.final_score_label)
```

* ស្ម័គ្រមនេះនឹង

```
restart_button = QtWidgets.QPushButton("រាយចកចំ")
```

```
back_to_menu_button = QtWidgets.QPushButton("រាយមុខ្ឌាន់ការ")
```

```
restart_button.clicked.connect(self.start_game)
```

```
back_to_menu_button.clicked.connect(self.back_to_menu)
```



```
Layout.addSpacing(40)
```

```
Layout.addWidget(restart_button)
```

```
Layout.addWidget(back_to_menu_button)
```

```
Layout.addStretch()
```

```
return widget
```

(ជំនួយកែបាយការណ៍)

```
def start_game(self):
```

```
self.reset_game_state()
```

```
self.stackLayout.setCurrentWidget(self.game_widget)
```

```
self.setFocus()
```

```
self.game_timer.start(30)
```

```
self.spawn_timer.start(1000)
```

ចិត្តសំណង់ទីនៅ
def reset_game_state(self):

self.game_timer.stop()

self.spawn_timer.stop()

លាងសង្កែបខ្លាំង

for obs in self.obstacles:

obs.deleteLater()

self.obstacles.clear()

ការចេញចាយហុងនៃវេរណី

self.boat_x = self.boat_x_start

self.boat_y = self.boat_y_start

self.boat.move(self.boat_x, self.boat_y)

ការផ្តល់ព័ត៌មាន

self.score = 0

self.lives = 3

self.obstacle_speed = 10

របៀបបង្ហាញគម្រោង Label

self.score_label.setText(f"សcore: {self.score}")

self.hear_label.setText(f"លើក: {self.lives}")

(នូវភាព "ជួយកើតឡើងដែលត្រូវការពេលវេលា" នូវការបង្ហាញ gam_timer (ក្នុង 30ms))

def update_game(self):

គុណសង្គមការងារនៃការបង្ហាញ

for obs in self.obstacles[:]:

obs.move(obs.x(), obs.y() + self.obstacle_speed)

ការចេញចាយហុងនៃវេរណី 1: ឈរឲ្យដែលមិនអាច

if obs.y() > self.game_area.height():

self.obstacles.remove(obs)

obs.deleteLater()

self.score += 1

បាតកនីងការបង្ហាញ

self.score_label.setText(f"សcore: {self.score}")

ពន្លាលសក៍ " កើតឡើងខ្លួយ : 101 រាង
 if self.score > 0 and self.score % 5 == 0:
 self.obstacle_speed += 2 # ផ្តល់ការបានចូល
 print(f"កើតឡើងបាន! ការបានចូលនេះ: {self.obstacle_speed}")

ពន្លាលសក៍ " កើតឡើង 2: ឆ្នាំ " ដែលគាំទ្រូវ នឹងបានចូលទៅក្នុងការបានចូល
 if self.boat.geometry().intersects(obs.geometry()):
 self.obstacles.remove(obs)
 obs.deleteLater()

self.lives -= 1 # គ្មាន 1 គឺជាបានបានអនុវត្តន៍
 self.heart_label.setText(f=♡: {self.lives})

* ពន្លាលសក៍ " កើតឡើងប្រហែល : 1 រូបឯណា "

if self.lives <= 0:
 self.game_over() # នឹងបង្ហាញការបានចូលទៅក្នុងការបានចូល
 return # ដោយបានចូលទៅក្នុងការបានចូល

* ឧទាឡ់: "ជូនកំរាមសរាយគីឡូ" គឺរួមរាយនៃការបានចូលទៅក្នុងការបានចូល
 def spawn_obstacle(self):
 obstacle = QtWidgets.QLabel(self.game_area)
 obstacle.setFocusPolicy(QtCore.Qt.NoFocus)

* 2. សំខាន់លាក់ x (ឱ្យនូវការបានចូល)

x = random.randint(0, self.game_area.width() - 80)

* 3. សំខាន់លាក់ (ការបានចូលទៅក្នុងការបានចូល)

if random.choice([True, False]):
 image_path = os.path.join(self.IMAGE_DIR, "Logo2.PNG")
 else:
 image_path = os.path.join(self.IMAGE_DIR, "rock01.PNG")

* ការបានចូលទៅក្នុងការបានចូល

obstacle.setPixmap(QtGui.QPixmap(image_path).scaled(80, 80, QtCore.Qt.KeepAspectRatio))
 obstacle.setGeometry(x, -80, 80, 70)
 obstacle.setStyleSheet("outline: none; border: none;")

```
obstacle.show()  
self.obstacles.append(obstacle)
```

ຜົນກໍາຊົງລົມປາກອະ

```
def game_over(self):
```

ପ୍ରକାଶନ

self.game_timer.stop()

self.spawn_timer.stop()

ପ୍ରୟୋଗରେ ମାନ୍ୟମାତ୍ରାଙ୍କିତ

if self.score > self.high_score:

self.high-score = self.score

```
self.high_score_label.setText(f"High Score: {self.high_score}")
```

ເສດຖະກ “ຕະຫຼາມຈຸບັນ” ປຸ່ອຫຼັງລົບເມນ

```
self.final_score_label.setText(f"Final Score: {self.score}")
```

ពេលវេលាភាសាអង់គ្លេស

```
self.stacked_layout.setCurrentWidget(self.game_over_widget)
```

ກສບພໍາໜ້າມຸນ່ວນັກ

def back-to-menu(self):

```
self.stackLayout.setCurrentWidget(self.menuWidget)
```

self.setFocus()

✽ ឧបនគ់ សម្រេចអណីករង់ វិចិករៀល នូវតី និងវិភាគរៀល ជូនដែលបានបង្ហាញ

det keyPressEvent (self, event).

if not self.game_timer.isActive()

return

ពេទ្យលេខាប្រចាំឆ្នាំ

if event.Key() == a + core.AltKey - Left:

`self.boat_x = max(0, self.boat_x - self.boat_speed)`

```
elif event.key() == QTCORE.Qt.Key_Right:
```

`max_x = self.game_area.width() - self.boat.width()`

`self.boat_x = min(max_x, self.boat_x + self.boat_speed)`

self.boat.move(self.boat_x, self.boat_y)

project-util.py

ចូលក្នុង project-ur នៃ 'importlib'

import importlib

import project-ui

* សំឡុងអតិថិជនកុំពេក 'project-ui' ទីនេះ

importlib.reload(project-ui)

ផ្លូវជាមុនក្នុង

* រាយការណ៍ទីនេះ 'run' ដែលធ្វើការ VI

def run():

try :

global river_rush_ui_instance

river_rush_ui_instance.close() ផ្តល់ទូទៅ

river_rush_ui_instance.deleteLater() ផ្តល់ទូទៅ

except:

pass

* ត្រូវ VI តាមឯកសារ (ការកែតាំងនូវរឿងគុណ)

river_rush_ui_instance = project_ur.project_ui()

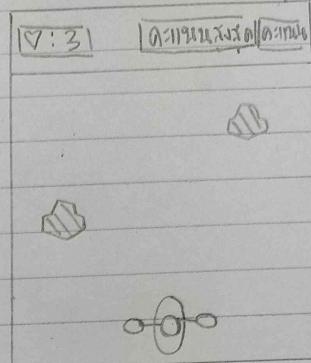
* នៅលើ VI តាមឯកសារ

river_rush_ui_instance.show()

ផ្តល់ទូទៅ VI ដែលនឹងបង្កើតឡើង

return river_rush_ui_instance

ออกแบบ UI Sketch หรือ Wireframe



หน้า 1

หน้ากาก



หน้า 2

หน้ากาก



ค่าไฟฟ้าคงที่ : 26

เงินเดือน

ตั้งแต่เดือนล่าสุด

หน้า 3

หน้าจอมาก

ขั้นตอนการทํางาน

1. ริเริ่มต้น รังสรรค์โครงสร้าง → 2. (แสดงผล) "หน้ากาก" → 4. (แสดงผล) "หน้ากาก"

3. (ผู้ใช้เลือก)

• ถ้ากด "ยก" → (จบ)

• ถ้ากด "เริ่มเกม" → ไปข้อ 4

5. (ผู้ใช้เลือก)

• ถ้า "ชน" → ชี้วัดตกลง → ชี้วัด หลอก?

• (ค้างไว้) → ไปข้อ 6

• (ก้าวไว้) → วนกรอบนาฬีนั่นต่อ

• ถ้า "กลับไป" → ค่าใหม่ๆ →

วนกรอบนาฬีนั่นต่อ

7. (ผู้ใช้เลือก)

• ถ้ากด "ก้าวเดินสู่ชีวิต" → วนกรอบไปข้อ 2 ← 6. (แสดงผล) "หน้ากาก"

• ถ้ากด "เปลี่ยนชื่อฟาร์ม" → วนกรอบไปข้อ 4

สรุปผลการนั่งเอนฯ

- การสร้างการนั่งเอนฯ: เรียนรู้การใช้ QStackedLayout และมีการรับเมื่อปุ่มกดมีชื่อคลิกหน้าจอ
- การควบคุมเกม: ใช้ฟังก์ชัน QTimer เพื่อรับ Game Loop และฟังก์ชัน keyPressEvent เพื่อรับ Input ตามผู้เล่น
- การจัดการข้อมูลเจ้าตัว (00P): ใช้ตัวเรียนรู้วิธีการสร้างและทำลาย Widget (ส่วนก้อนของ) และมีติดตามการเปลี่ยนแปลง
- ภาษาเขียนมัตต์ Maya: ใช้ไฟล์ภาษา Python ที่บันทึก UI ของ PySide2 เช่นกันนั่นคือตัวของคุณ Maya Cshiboken, OpenMayaUI