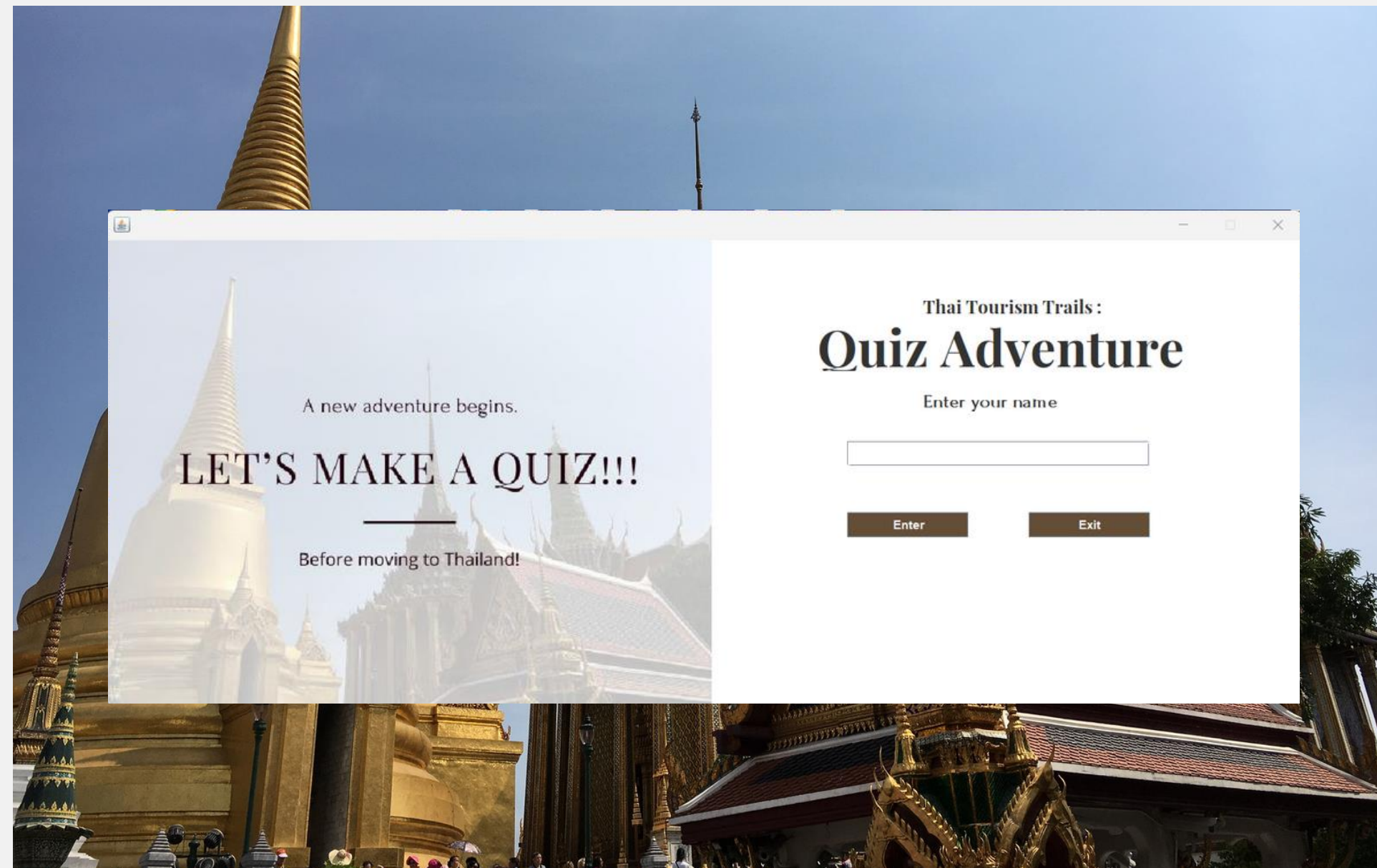
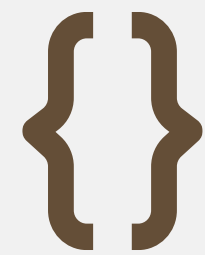


oooo

THAI TOURISM TRAILS

QUIZ ADVENTURE



นาย ณัฐพล พิล้าไชย รหัสนิสิต 6630300262
วิชา PROGRAMMING FUNDAMENTALS II รหัสวิชา 03603112-65
อาจารย์ผู้สอน กุลวดี สมบูรณ์วิวัฒน์ ปีการศึกษา 2566

oooo

TABLE OF CONTENTS



1

INTRODUCTION
& REASON

2

MAIN FEATURES

3

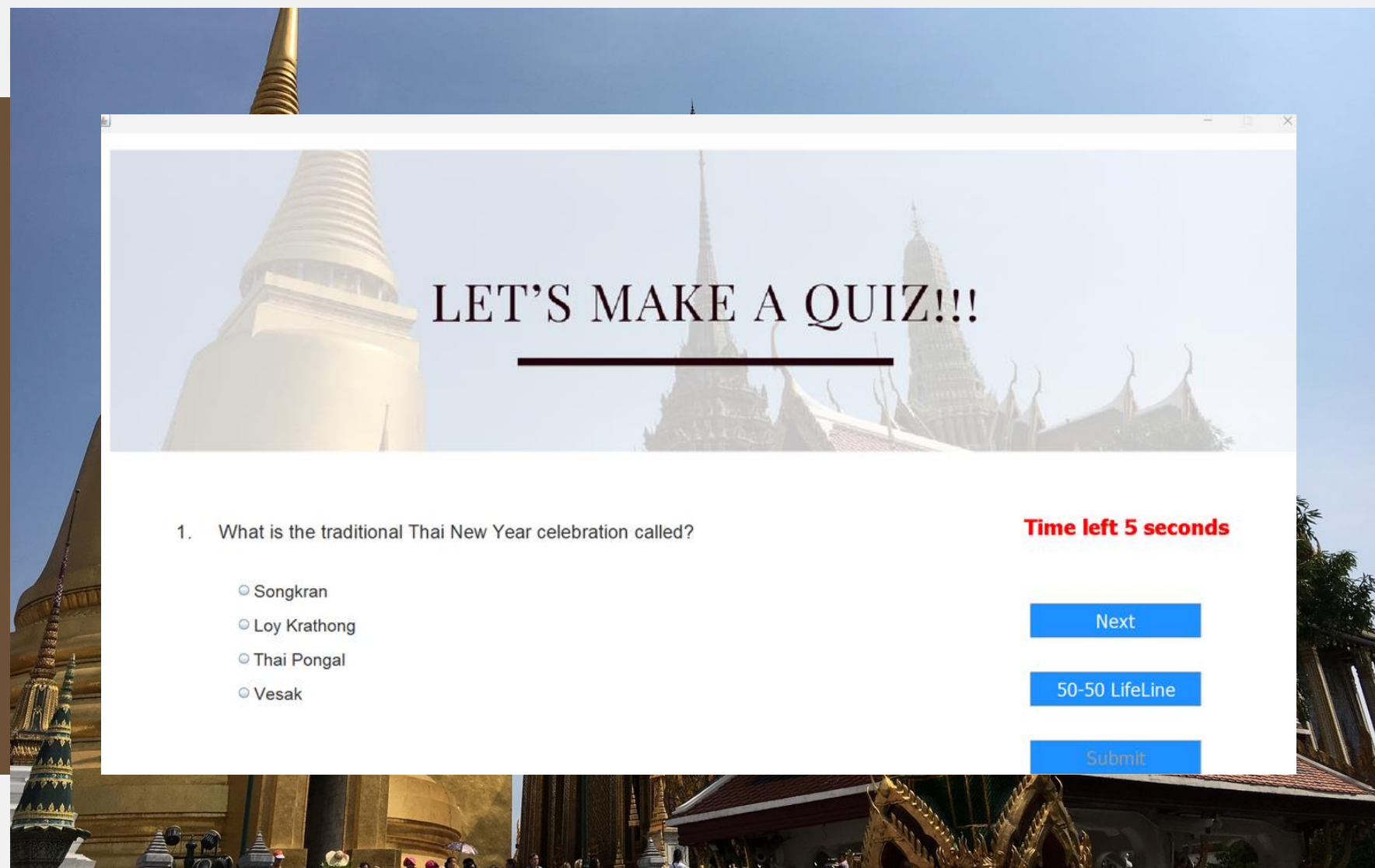
ALGORITHM

4

INSTALLATION

oooo

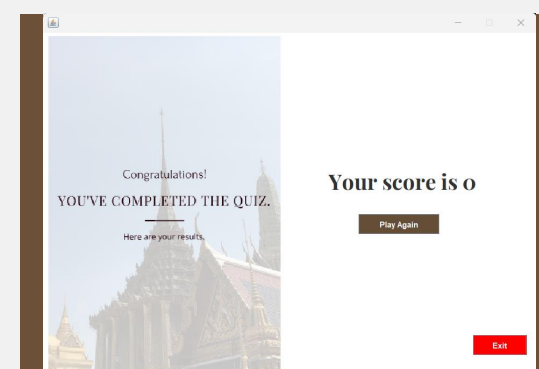
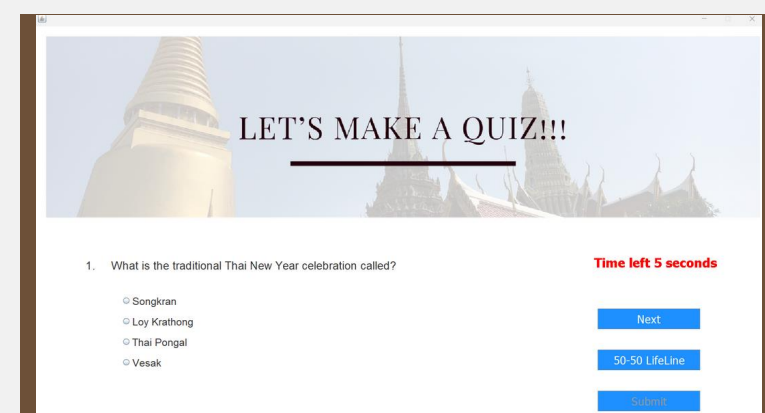
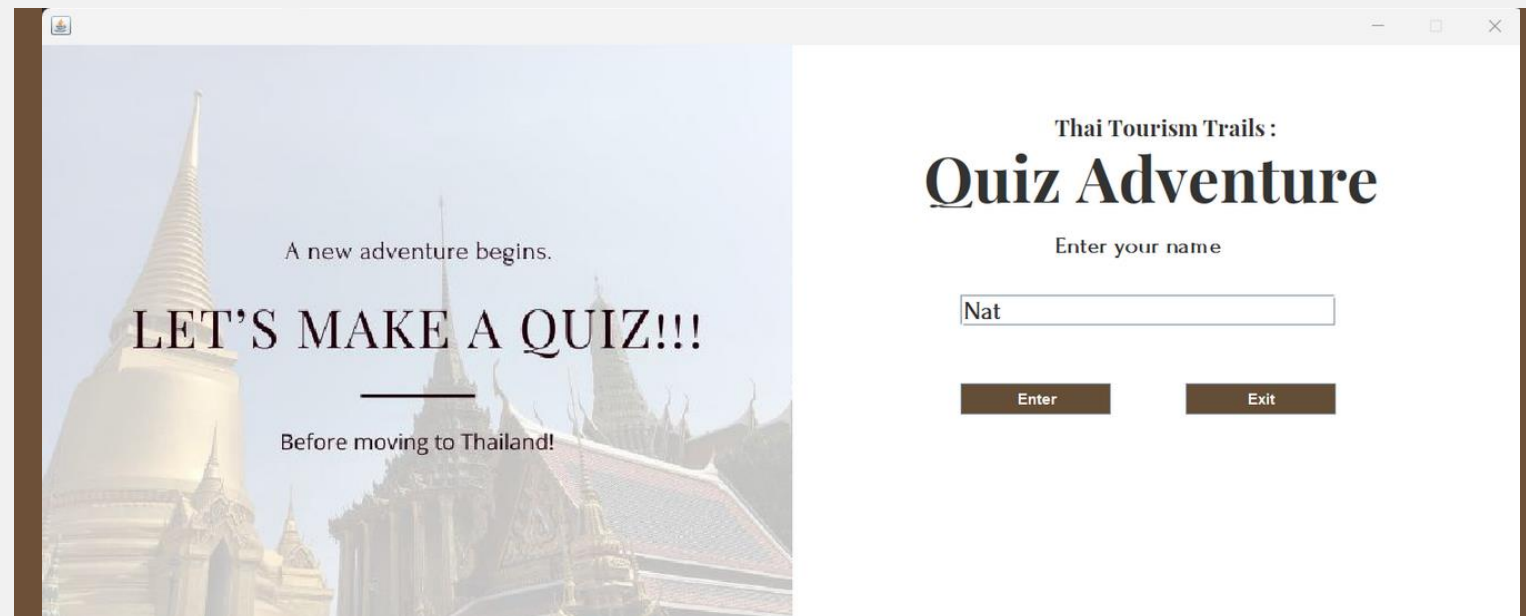
WHAT IS THIS PROJECT?



โปรเจกต์นี้เป็นแอปพลิเคชัน Quiz ชื่อว่า "ThaiTourism Trails: Quiz Adventure" ที่ถูกพัฒนาขึ้นด้วยภาษา Java เพื่อสร้างแอปพลิเคชันตอบคำถามโดยในแต่ละรอบของเกมผู้ใช้จะต้องตอบคำถามที่เกี่ยวข้องกับสถานที่ท่องเที่ยวหรือประเพณีและวัฒนธรรมของประเทศไทยที่กำหนดไว้ โดยมีเวลาตอบคำถามในแต่ละข้อที่จำกัด ให้เพื่อเพิ่มความท้าทายและสร้างสมาธิให้กับผู้เล่น

๐๐๐๐

HOW TO PLAY THIS APP



1. เริ่มต้นด้วยการเปิดแอปและพบกับหน้าจอ Login ที่ให้ผู้ใช้กรอกชื่อผู้ใช้
2. หลังจากกรอกชื่อผู้ใช้แล้ว ผู้ใช้สามารถเลือกคลิกที่ปุ่ม "Enter" เพื่อเข้าสู่ระบบหรือปุ่ม "Exit" เพื่อออกจากแอป
3. เมื่อเข้าสู่ระบบ ผู้ใช้จะเห็นหน้าจอ Rules ซึ่งแสดงกฎการเล่นและข้อแนะนำในการเล่นเกม
4. หลังจากอ่านกฎและข้อแนะนำเสร็จสิ้น ผู้ใช้สามารถคลิกที่ปุ่ม "Start" เพื่อเริ่มเล่นเกม
5. ในหน้าเล่นเกม (Quiz) ผู้เล่นจะพบกับคำถามที่ต้องตอบโดยเลือกคำตอบที่ถูกต้อง
6. ผู้เล่นสามารถใช้ Lifeline เพื่อช่วยลดตัวเลือกคำตอบลงเหลือเพียงสองตัวเลือก
7. เมื่อตอบคำถามทั้งหมดเสร็จสิ้น ผู้เล่นสามารถคลิกที่ปุ่ม "Submit" เพื่อดูคะแนนที่ได้รับ
8. หลังจากดูคะแนน ผู้เล่นสามารถเลือกที่จะเล่นเกมใหม่หรือออกจากแอปได้โดยคลิกที่ปุ่ม "Play Again" หรือ "Exit" ตามต้องการ

o o o o

WHY THIS PROJECT



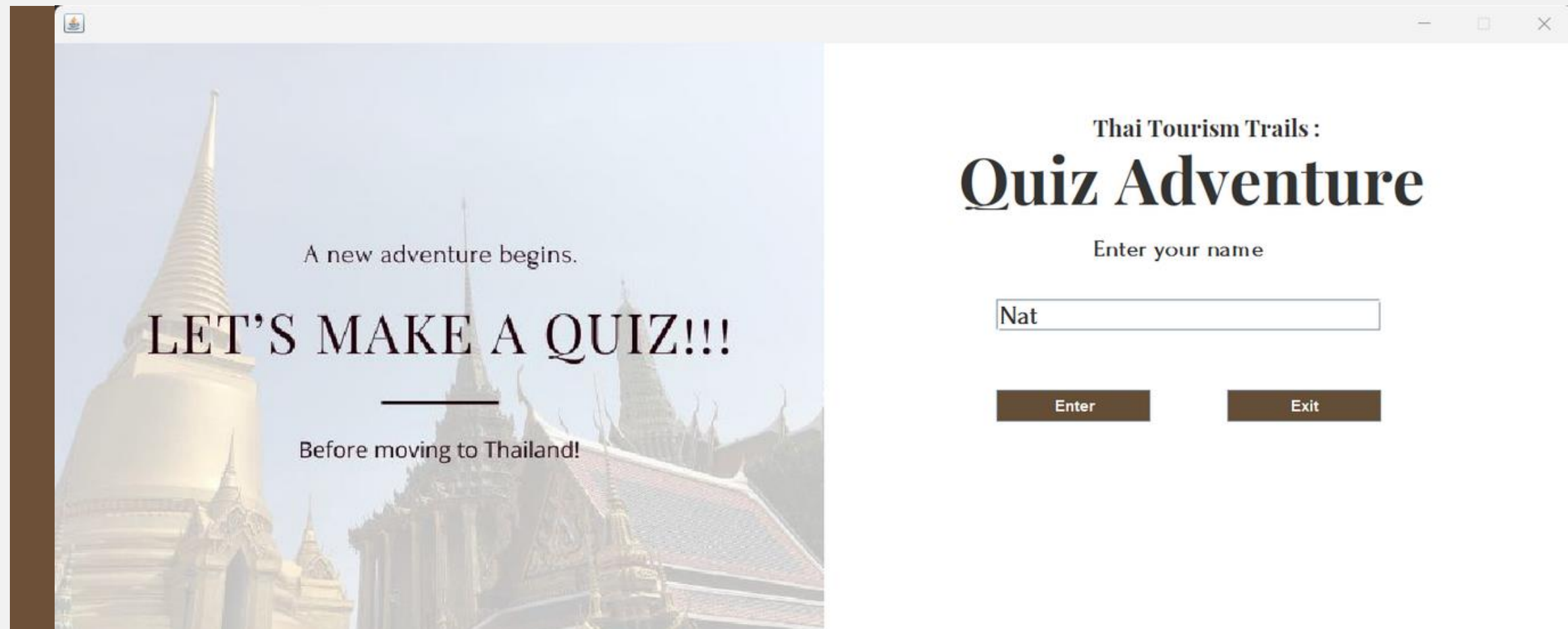
โปรเจกต์นี้ยังเป็นโอกาสที่ดีในการฝึกทักษะการเขียนโปรแกรมภาษา Java และการออกแบบ User Interface (UI) ที่น่าสนใจและใช้งานง่าย ซึ่งสามารถนำไปประยุกต์ใช้ในโปรเจกต์อื่นๆ ได้ในอนาคต

โปรเจกต์นี้จึงเป็นทางเลือกที่ดีสำหรับผมที่ต้องการเข้าใจและพัฒนาทักษะในด้านการพัฒนาโปรแกรมและการสร้างแอปพลิเคชัน จากที่เรียนมาในวิชา Programming Fundamentals II

oooo

MAIN FEATURES: Login and Registration

Login Class



หากผู้ใช้กด Enter หลังจากป้อนชื่อของพวกเขาแล้ว โปรแกรมจะย้ายไปยังหน้าต่างถัดไป
คือหน้าต่าง Display Game Rules

หน้าต่างนี้เป็นหน้าต่างแรกของแอปฯจะแสดง
ช่องให้ User ป้อนชื่อของพวกเขาและกดปุ่ม
“Enter” เพื่อเข้าสู่ระบบหรือ “Exit” เพื่อออกจาก
แอปพลิเคชัน

หน้าต่าง Login ใช้ Java Swing ในการสร้าง
อินเทอร์เฟซกราฟิก โดยใช้คลาส JFrame เป็น
หน้าต่างหลักของแอปพลิเคชัน เพื่อแสดงหน้าต่างที่ผู้ใช้
จะใช้ป้อนข้อมูลการเข้าสู่ระบบ นอกจากนี้ยังใช้
JButton เพื่อสร้างปุ่มที่ User สามารถคลิกเพื่อเข้าสู่
ระบบหรือออกจากแอปพลิเคชันได้ ซึ่งจะมีปุ่ม "Enter"
เพื่อเข้าสู่ระบบและปุ่ม "Exit" เพื่อออกจากแอปพลิเคชัน
ทั้งสองปุ่มนี้ถูกสร้างโดยใช้คลาส JButton ภายใน
หน้าต่าง JFrame ของคลาส Login โดยวางไว้ด้านล่าง
ของช่องใส่ข้อมูลชื่อผู้ใช้ โดยทั้งสองปุ่มมีการเชื่อมโยง
กับเมทอด actionPerformed เพื่อให้ทำงานเมื่อผู้ใช้
คลิกที่ปุ่มดังกล่าว.



MAIN FEATURES: Display Game Rules



มีหน้าต่างที่นำเสนอกฎและกติกาของเกมในรูปแบบของข้อความ โดยมีเนื้อหาประกอบด้วยกฎเกณฑ์การเล่นเกม การใช้ชีวิตช่วยเหลือ (Lifeline) การสิ้นสุดเกม การเล่นอีกครั้ง และการออกจากเกม

นอกจากนี้ยังมีปุ่ม **"Start"** เพื่อเริ่มเกมและปุ่ม **"Back"** เพื่อย้อนกลับไปยังหน้า **Login** เมื่อต้องการกลับมา.

สำหรับการออกแบบ UI ใช้คลาส JFrame เช่นเดียวกับคลาส Login โดยมีพื้นหลังสีขาวและใช้การจัดวางแบบ null layout เพื่อความยืดหยุ่นในการจัดวางองค์ประกอบ มีปุ่ม "Start" และ "Back" เป็นปุ่มสำคัญที่ช่วยให้ผู้เล่นสามารถเริ่มเกมหรือย้อนกลับไปหน้าจอ Login ได้ตามต้องการ.

เมื่อกดปุ่ม **"Start"** จะไปยังคลาส Quiz เพื่อเริ่มการทำคำถามของเกม ในขณะที่การกดปุ่ม **"Back"** จะนำผู้ใช้กลับไปยังหน้า Login เพื่อให้สามารถล็อกอินใหม่หรือออกจากระบบได้ ทำให้มีประสิทธิภาพในการนำทางและใช้งานระบบได้อย่างสะดวกสบายโดยไม่ซับซ้อน.



o o o o

MAIN FEATURES: Questioning Mechanism

Quiz Class

คลาส Quiz เป็นคลาสที่ให้ผู้เล่นเข้าสู่กระบวนการเล่นเกม Quiz ของ Thai Tourism Trails: Quiz Adventure โดยมีหน้าต่างที่แสดงคำถามและตัวเลือกต่างๆ พร้อมทั้งปุ่มต่างๆ เพื่อจัดการกับการเล่นเกม โดยมีฟีเจอร์หลักๆ ดังนี้:

- 1.แสดงคำถามและตัวเลือก:** แสดงคำถามและตัวเลือกที่เกี่ยวข้องกับท่องเที่ยวในประเทศไทย โดยข้อมูลถูกเก็บไว้ในอาร์เรย์เพื่อความสะดวกในการจัดการและแสดงผล.
- 2.จับเวลา:** มีการจับเวลาที่แสดงผลบนหน้าต่างเพื่อให้ผู้เล่นรู้เวลาที่เหลือในการตอบคำถาม โดยหากเวลาหมดลงก่อนที่ผู้เล่นจะตอบได้ ระบบจะดำเนินการปิดหน้าต่างเกม.
- 3.ตัวเลือกการเล่น:** มีปุ่ม "Next" เพื่อย้ายไปยังคำถามถัดไป, ปุ่ม "50-50 LifeLine" เพื่อใช้ชีวิตช่วยเหลือที่จะลดตัวเลือกลงเหลือ 2 ตัวเลือก, และปุ่ม "Submit" เพื่อส่งคำตอบและดูผลคะแนน.
- 4.การตรวจสอบคำตอบ:** เมื่อผู้เล่นส่งคำตอบด้วยปุ่ม "Submit" ระบบจะตรวจสอบคำตอบและคำถามทั้งหมดที่ผู้เล่นตอบ และคำนวณคะแนนที่ได้ตามความถูกต้องของคำตอบ.

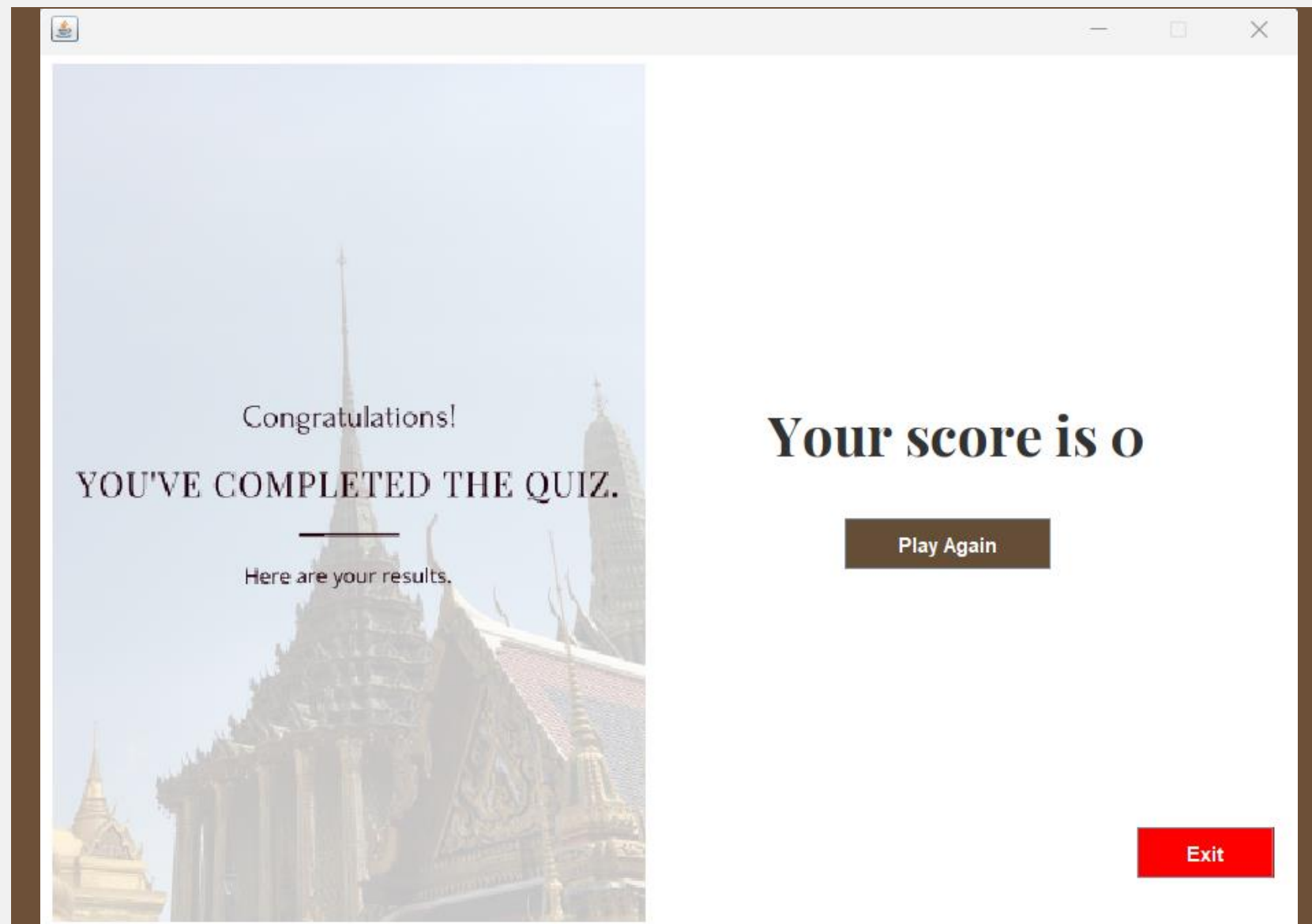
สำหรับการออกแบบ UI ใช้คลาส JFrame เช่นเดียวกับคลาสอื่นๆ และใช้การจัดวางแบบ null layout เพื่อความยืดหยุ่นในการจัดวางองค์ประกอบ แสดงผลข้อความและปุ่มต่างๆ ที่เกี่ยวข้อง สร้างประสบการณ์การใช้งานที่สะดวกและเข้าใจง่ายสำหรับผู้เล่น.



o o o o

MAIN FEATURES: Score Display

Score Class



คลาส **Score** เป็นคลาสที่แสดงคะแนนที่ผู้เล่นได้รับหลังจากเล่นเกม **Quiz** ของ **Thai Tourism Trails: Quiz Adventure** โดยมีฟีเจอร์หลักๆ ดังนี้:

- 1. แสดงคะแนน:** แสดงคะแนนที่ผู้เล่นได้รับหลังจากทำการตอบคำถามทั้งหมดในเกม โดยแสดงผลเป็นข้อความว่า "Your score is [คะแนน]" เพื่อแสดงคะแนนที่ผู้เล่นได้รับ.
- 2. ปุ่ม Play Again และ Exit:** มีปุ่ม "Play Again" เพื่อเริ่มเล่นเกมใหม่หลังจากเล่นเสร็จสิ้นแล้ว และปุ่ม "Exit" เพื่อออกจากแอปพลิเคชัน.

สำหรับการออกแบบ UI ใช้คลาส JFrame เพื่อสร้างหน้าต่าง UI และมีปุ่มและข้อความที่แสดงผลคะแนนใน UI เพื่อแสดงผลแก่ผู้เล่น การจัดวางองค์ประกอบในหน้าต่าง UI นั้นจะให้ความสะดวกและความเข้าใจง่ายสำหรับผู้เล่นในการใช้งานคลาส Score โดยใช้การจัดวางแบบ null layout เช่นเดียวกับคลาสอื่น ๆ ในโปรเจกต์.



oooo

CLASS DIAGRAM

Class Diagram มีความหมายและการทำงานต่างๆ ดังนี้:

1.คลาส Login:

- คลาสนี้ใช้สำหรับการเข้าสู่ระบบของ User
- มีฟังก์ชัน login() เพื่อทำการเข้าสู่ระบบ
- มีฟังก์ชัน exit() เพื่อออกจากแอปพลิเคชัน

2.คลาส Rules:

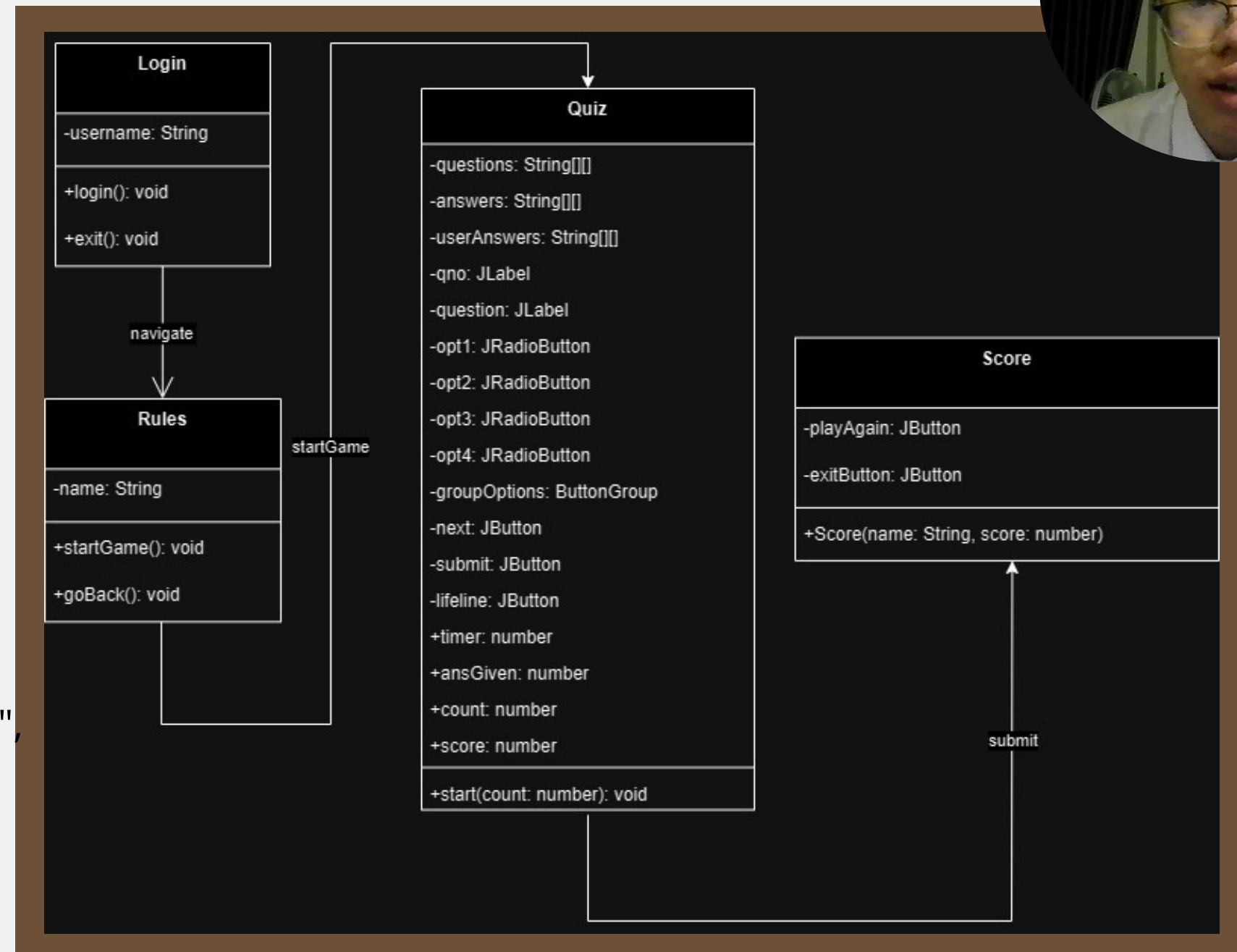
- คลาสนี้แสดงกฎการเล่นของเกม
- มีฟังก์ชัน startGame() เพื่อเริ่มเกม
- มีฟังก์ชัน goBack() เพื่อย้อนกลับไปหน้าก่อนหน้า

3.คลาส Quiz:

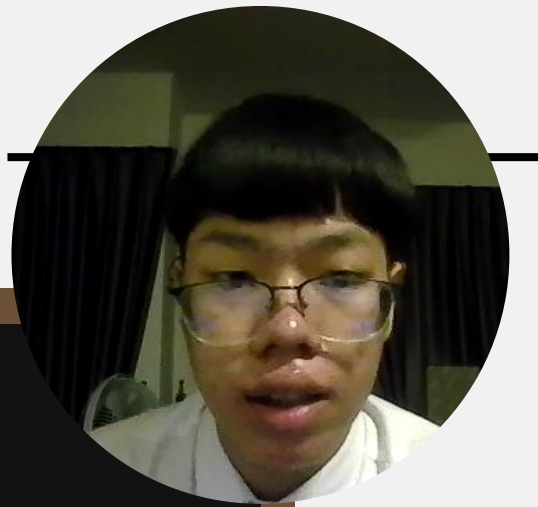
- คลาสนี้ใช้สำหรับการทำคำถามในเกม
- มีองค์ประกอบต่างๆ เช่น คำถาม, ตัวเลือก, ปุ่ม "ถัดไป", ปุ่ม "ส่งคำตอบ", ฯลฯ
- มีฟังก์ชัน start(count: number) เพื่อเริ่มการทำคำถาม

4.คลาส Score:

- คลาสนี้แสดงคะแนนที่ผู้เล่นได้รับหลังจากเล่นเกม
- มีปุ่ม "เล่นอีกครั้ง" เพื่อเริ่มเกมใหม่
- มีปุ่ม "ออกจากแอป" เพื่อออกจากแอปพลิเคชัน



การทำงานของแต่ละคลาสมีการเชื่อมโยงกันไปยังคลาสอื่นๆ ในลำดับตามที่ใช้กดปุ่ม หรือทำการใช้งานต่างๆ เพื่อให้แอปพลิเคชันทำงานตามที่ต้องการได้ในแต่ละขั้นตอนของการ เล่นเกมหรือใช้งานโปรแกรมคอมพิวเตอร์ต่างๆ ในแอปพลิเคชันนี้



o o o o

ALGORITHM

```
package quiz.appication;

import javax.swing.*;
import java.awt.*;
import java.awt.event.*;

public class Login extends JFrame implements ActionListener{

    JButton enter,exit;
    JTextField tfname;

    Login() {
        getContentPane().setBackground(Color.WHITE);
        setLayout(null);

        //image 1
        ImageIcon i1 = new
        ImageIcon(ClassLoader.getResource("icons/login.png"));
        JLabel image = new JLabel(i1);
        image.setBounds(0,0,600,480);
        add(image);

        //SubHeading
        JLabel Subheading = new JLabel("Thai Tourism Trails :");
        Subheading.setBounds(810,40,400,50);
        Subheading.setFont(new Font("Playfair Display",Font.BOLD,18));
        Subheading.setBackground(Color.decode("#211A14"));
        add(Subheading);

        //Label 1
        JLabel heading = new JLabel("Quiz Adventure");

        heading.setBounds(705,80,400,50);
        heading.setFont(new Font("Playfair Display",Font.BOLD,50));
        heading.setBackground(Color.decode("#211A14"));
        add(heading);

        //Label 2
        JLabel name = new JLabel("Enter your name");
        name.setBounds(810,150,300,20);
        name.setFont(new Font("Forum",Font.BOLD,18));
        name.setBackground(Color.decode("#211A14"));
        add(name);

        //Text Field
        tfname = new JTextField();
        tfname.setBounds(735,200,300,25);
        tfname.setFont(new Font("Forum",Font.BOLD,20));
        add(tfname);
```

```
//Button 1
enter = new JButton("Enter");
enter.setBounds(735, 270, 120, 25);
enter.setBackground(Color.decode("#644E37"));
enter.setForeground(Color.WHITE);
enter.addActionListener(this);
add(enter);

// Key Bindings(Enter)
InputMap inputMap =
getRootPane().getInputMap(JComponent.WHEN_IN_FOCUSED_WINDOW);
ActionMap actionMap = getRootPane().getActionMap();

inputMap.put(KeyStroke.getKeyStroke(KeyEvent.VK_ENTER, 0),
"rulesPressed");
actionMap.put("rulesPressed", new AbstractAction() {
    @Override
    public void actionPerformed(ActionEvent e) {
        enter.doClick();
    }
});

//Button 2
exit = new JButton("Exit");
exit.setBounds(915, 270, 120, 25);
exit.setBackground(Color.decode("#644E37"));
exit.setForeground(Color.WHITE);
exit.addActionListener(this);
add(exit);

//set size
setSize(1200,500);
setLocation(200,150);
setResizable(false); // Lock Resolution
setVisible(true);
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}

@Override
public void actionPerformed(ActionEvent ae) {
    if (ae.getSource() == enter) {
        String name = tfname.getText();
        setVisible(false);
        new Rules(name);
    } else if (ae.getSource() == exit) {
        //Exit app
        System.exit(0);
    }
}

public static void main(String[] args) {
    new Login();
}
}
```

Login Class

- login():
 - Algorithm: ในการเรียกใช้เมธอดนี้ ระบบจะตรวจสอบข้อมูลที่ผู้ใช้ป้อนเข้ามา เช่น ชื่อผู้ใช้และรหัสผ่าน หากข้อมูลถูกต้อง ระบบจะอนุญาตให้ผู้ใช้เข้าสู่ระบบ หากไม่ถูกต้อง ระบบจะแจ้งเตือนผู้ใช้และขอให้ป้อนข้อมูลใหม่
- exit():
 - Algorithm: เมธอดนี้ใช้ในการปิดแอปพลิเคชันหรือหน้าต่างของระบบ



o o o o

ALGORITHM

Rules Class

- startGame():
 - Algorithm: เมื่อบริษัทใช้เริ่มเกม โดยระบบจะแสดงกฎการเล่นและข้อแนะนำต่าง ๆ เพื่อให้ผู้เล่นเข้าใจเกม และจะเริ่มเกมหลังจากที่ผู้เล่นกดปุ่ม "Start"
- goBack():
 - Algorithm: เมื่อบริษัทใช้ในการกลับไปยังหน้าจอเมื่อผู้เล่นต้องการกลับมาดูกฎและข้อแนะนำเพิ่มเติม

```
Rules.java

package quiz.application;

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class Rules extends JFrame implements ActionListener {

    String name;
    JButton start, back;

    public Rules(String name) {
        this.name = name;
        getContentPane().setBackground(Color.WHITE);
        setLayout(null);

        //subHeading
        JLabel subHeading = new JLabel("Welcome,"+name+" Here is a rule:");
        subHeading.setBounds(40, 10, 700, 30);
        subHeading.setFont(new Font("Playfair Display", Font.BOLD, 18));
        subHeading.setBackground(Color.decode("#211A14"));
        add(subHeading);
        //heading
        JLabel heading = new JLabel("Rules of Thai Tourism Trails: Quiz Adventure");
        heading.setBounds(40, 40, 700, 30);
        heading.setFont(new Font("Playfair Display", Font.BOLD, 28));
        heading.setBackground(Color.decode("#211A14"));
        add(heading);

        JLabel rules = new JLabel();
        rules.setFont(new Font("Tahoma", Font.PLAIN, 16));
        rules.setBackground(Color.decode("#211A14"));
        rules.setText(
            "<html>" +
            "<b>Answering Questions:</b><br>" +
            "1. Players must answer each question continuously without skipping any.<br>" +
            "2. Each question must be answered with only one choice.<br>" +
            "<br>" +
            "<b>Using Lifeline:</b><br>" +
            "1. The \"50-50\" Lifeline can be used only once.<br>" +
            "2. When using the Lifeline, players will receive assistance in reducing the answer choices to only two.<br>" +
            "3. The purpose of using the Lifeline is to assist players in answering questions more easily, rather than providing the correct answer.<br><br>" +
            "<b>Ending the Game:</b><br>" +
            "1. When players have answered all the questions, or pressed the \"Submit\" button, the game will end.<br>" +
            "2. Players will receive points for correct answers, and the total score will be counted and displayed on the summary page.<br><br>" +
            "<b>Playing Again:</b><br>" +
            "1. After the game ends, players can play again by pressing the \"Play Again\" button.<br><br>" +
            "<b>Exiting the Game:</b><br>" +
            "1. Players can exit the game and close the program by pressing the \"Exit\" button.<br><br>" +
            "<b>Recommendations:</b><br>" +
            "1. While playing, it is recommended that players use their time wisely to think and answer the questions correctly.<br>" +
            "2. It is recommended that players follow the rules and use the Lifeline correctly and effectively.<br><br>" +
            "<b>Note:</b> The administrators reserve the right to change or modify the rules without prior notice.<br><br>" +
            "<b>Good Luck :)</b>" +
            "<html>"
        );

    }

}
```

```
JScrollPane scrollPane = new JScrollPane(rules);
scrollPane.setBounds(20, 90, 700, 350);
add(scrollPane);

// Center JLabel rules
rules.setVerticalAlignment(SwingConstants.CENTER);
rules.setHorizontalAlignment(SwingConstants.CENTER);

// Back button
back = new JButton("Back");
back.setBounds(250, 500, 100, 30);
back.setBackground(Color.decode("#644E37"));
back.setForeground(Color.WHITE);
back.addActionListener(this);
add(back);

// Start button
start = new JButton("Start");
start.setBounds(400, 500, 100, 30);
start.setBackground(Color.decode("#644E37"));
start.setForeground(Color.WHITE);
start.addActionListener(this);
add(start);

setSize(800, 650);
setLocation(300, 150);
setResizable(false); // Lock Resolution
setVisible(true);
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}

public void actionPerformed(ActionEvent ae) {
    if (ae.getSource() == start) {
        setVisible(false);
        new Quiz(name);
    } else if (ae.getSource() == back) {
        setVisible(false);
        new Login();
    }
}

public static void main(String[] args) {
    new Rules("User");
}

}
```



o o o o

ALGORITHM

Quiz Class

- start(count: number):
 - Algorithm: เมรอดนี้ใช้เริ่มต้นการเล่นเกม โดยระบบจะโหลดคำถามและตัวเลือกที่เกี่ยวข้องจากแหล่งข้อมูล และแสดงบนหน้าจอเพื่อให้ผู้เล่นตอบคำถาม และจะเลือกข้อถัดไปหลังจากที่ผู้เล่นตอบคำถามแต่ละข้อ
- submit():
 - Algorithm: เมรอดนี้ใช้สำหรับการส่งคำตอบของผู้เล่นไปตรวจสอบ โดยจะนับคะแนนและแสดงผลลัพท์หลังจากผู้เล่นตอบคำถามทั้งหมด
- lifeline():
 - Algorithm: เมรอดนี้ใช้ในการใช้ Lifeline เพื่อช่วยในการลดตัวเลือกคำตอบที่ไม่ถูกต้อง



```
package quiz.appication;

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class Quiz extends JFrame implements ActionListener{

    String questions[] = new String[10][5];
    String answers[] = new String[10][5];
    String usernames[] = new String[10][5];

    JButton opt1,opt2,opt3,opt4;
    JButton groupOptions;
    JButton next,submit,lifeline;

    public static int timer = 15;
    public static int now_give = 0;
    public static int count = 0;
    public static int score = 0;

    String name;
    Quiz(String name) {
        this.name = name;
        setBounds(50,50,650,650);
        getContentPane().setBackground(Color.WHITE);
        setLayout(null);

        //image 1
        ImageIcon img1 = new
        ImageIcon(ClassLoader.getSystemResource("icons/quiz.png"));
        JLabel image = new JLabel(img1);
        image.setBounds(50,50,150,150);
        add(image);

        qno = new JLabel("");
        qno.setBounds(180, 550, 50, 50);
        qno.setFont(new Font("Tahoma",Font.PLAIN,22));
        add(qno);

        question = new JLabel("");
        question.setBounds(180, 580, 500, 600);
        question.setFont(new Font("Tahoma",Font.PLAIN,22));
        add(question);

        questions[0][0] = "What is the traditional Thai New Year celebration called?";
        questions[0][1] = "Songkran";
        questions[0][2] = "Loy Krathong";
        questions[0][3] = "Thai Pongal";
        questions[0][4] = "Vesak";

        questions[1][0] = "Which Thai city is famous for its floating markets?";
        questions[1][1] = "Bangkok";
        questions[1][2] = "Pattaya";
        questions[1][3] = "Chiang Mai";
        questions[1][4] = "Damnoen Saduak Market";

        questions[2][0] = "What is the name of the famous Thai dish consisting of stir-fried rice noodles?";
        questions[2][1] = "Pad Thai";
        questions[2][2] = "Tom Yum Curry";
        questions[2][3] = "Som Tum";
        questions[2][4] = "Tom Yum";

        questions[3][0] = "Which Thai festival is celebrated during the full moon of the twelfth lunar month?";
        questions[3][1] = "Songkran";
        questions[3][2] = "Loy Krathong";
        questions[3][3] = "Thai Pongal";
        questions[3][4] = "Vesak";

        questions[4][0] = "What is the capital city of Thailand?";
        questions[4][1] = "Bangkok";
        questions[4][2] = "Phuket";
        questions[4][3] = "Chiang Mai";
        questions[4][4] = "Pattaya";

        questions[5][0] = "Which Thai island is known for its vibrant nightlife and full moon parties?";
        questions[5][1] = "Koh Samui";
        questions[5][2] = "Phuket";
        questions[5][3] = "Koh Phi Phi";
        questions[5][4] = "Koh Phangan";

        questions[6][0] = "What is the name of the historical park in Thailand featuring ruins of ancient temples?";
        questions[6][1] = "Ayutthaya Historical Park";
        questions[6][2] = "Sukhothai Historical Park";
        questions[6][3] = "Thao Son National Park";
        questions[6][4] = "Erawan National Park";
```

```
questions[7][0] = "Which river is considered the lifeline of Bangkok?";
questions[7][1] = "Chao Phraya River";
questions[7][2] = "Mekong River";
questions[7][3] = "Ping River";
questions[7][4] = "Nam River";

questions[8][0] = "What is the traditional Thai greeting?";
questions[8][1] = "Sawasdee";
questions[8][2] = "Khom";
questions[8][3] = "Namaste";
questions[8][4] = "Konnichiwa";

questions[9][0] = "Which Thai festival is known for its release of decorated buxtons, into rivers and lakes?";
questions[9][1] = "Songkran";
questions[9][2] = "Loy Krathong";
questions[9][3] = "Thai Pongal";
questions[9][4] = "Vesak";

answers[0][0] = "Songkran";
answers[0][1] = "Damnoen Saduak Market";
answers[0][2] = "Pad Thai";
answers[0][3] = "Loy Krathong";
answers[0][4] = "Bangkok";
answers[1][0] = "Koh Phangan";
answers[1][1] = "Ayutthaya Historical Park";
answers[1][2] = "Chao Phraya River";
answers[1][3] = "Sawasdee";
answers[1][4] = "Loy Krathong";

//Option 1
opt1 = new JButton(opt1);
opt1.setBounds(170, 520, 700, 50);
opt1.setBackground(Color.WHITE);
opt1.setFont(new Font("Dialog",Font.PLAIN,30));
add(opt1);

//Option 2
opt2 = new JButton(opt2);
opt2.setBounds(170, 540, 700, 50);
opt2.setBackground(Color.WHITE);
opt2.setFont(new Font("Dialog",Font.PLAIN,30));
add(opt2);

//Option 3
opt3 = new JButton(opt3);
opt3.setBounds(170, 560, 700, 50);
opt3.setBackground(Color.WHITE);
opt3.setFont(new Font("Dialog",Font.PLAIN,30));
add(opt3);

//Option 4
opt4 = new JButton(opt4);
opt4.setBounds(170, 580, 700, 50);
opt4.setBackground(Color.WHITE);
opt4.setFont(new Font("Dialog",Font.PLAIN,30));
add(opt4);

//Group Option
groupOptions = new ButtonGroup();
groupOptions.add(opt1);
groupOptions.add(opt2);
groupOptions.add(opt3);
groupOptions.add(opt4);

next = new JButton("Next");
next.setBounds(180, 580, 200, 40);
next.setFont(new Font("Tahoma",Font.PLAIN,22));
next.setBackground(new Color(30,144,255));
next.setForeground(Color.WHITE);
next.addActionListener(this);
add(next);

lifeline = new JButton("50-50 Lifeline");
lifeline.setBounds(180,610,200,40);
lifeline.setFont(new Font("Tahoma",Font.PLAIN,22));
lifeline.setBackground(new Color(30,144,255));
lifeline.setForeground(Color.WHITE);
lifeline.addActionListener(this);
add(lifeline);

submit = new JButton("Submit");
submit.setBounds(180,710,200,40);
submit.setFont(new Font("Tahoma",Font.PLAIN,22));
submit.setBackground(new Color(30,144,255));
submit.setForeground(Color.WHITE);
submit.addActionListener(this);
add(submit);

Start(count);

setResizable(false); // Lock Resolution
setVisible(true);
}
```

```
public void actionPerformed(ActionEvent ae){
    if(ae.getSource() == next){
        //repeat
        repeat();
        //reset options
        opt1.setEnabled(true);
        opt2.setEnabled(true);
        opt3.setEnabled(true);
        opt4.setEnabled(true);
        //last ans_give
        ans_give = 1;
        if(groupOptions.getSelection() == null){
            //no select
            usernames[count][0] = "";
        }else{
            //select option --> get user answer
            usernames[count][0] =
            groupOptions.getSelection().getActionCommand();
        }

        if(count == 9){
            //If count is 9 but question is 9(already pass number now is number 10)
            //close next button
            next.setEnabled(false);
            //open submit button
            submit.setEnabled(true);
        }

        count++;
        Start(count);
    }else if(ae.getSource() == lifeline){
        if(count == 2 || count == 4 || count == 6 || count == 8 || count == 10){
            opt2.setEnabled(false);
            opt3.setEnabled(false);
        }else{
            opt1.setEnabled(false);
            opt4.setEnabled(false);
        }

        //use only 1 time
        lifeline.setEnabled(false);
    }else if(ae.getSource() == submit){
        ans_give = 1;
        if(groupOptions.getSelection() == null){
            //no select
            usernames[count][0] = "";
        }else{
            //select option --> get user answer
            usernames[count][0] =
            groupOptions.getSelection().getActionCommand();
        }

        //check all answer
        for(int i = 0;i < usernames.length;i++){
            if(usernames[i][0].equals(answers[i][1])){
                //correct
                score += 1;
            }else{
                //wrong
                score += 0;
            }
        }

        setVisible(false);
        //score
        new Score(name, score);
        //reset count
        count = 0;
        System.out.println("reset count");
        System.out.println("now count = "+count);
    }

    public void paint(Graphics g){
        super.paint(g);

        String time = "Time left "+ timer + " seconds";// 15
        g.setColor(Color.RED);
        g.setFont(new Font("Tahoma",Font.BOLD,20));
        if(timer > 0){
            g.drawString(time, 100, 500);
        }else{
            g.drawString("Times up!!!", 100, 500);
        }

        timer--;

        try {
            Thread.sleep(1000);
            repaint();
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

```
if(ans_give == 1){
    //when already pass next or submit button
    //reset options and start new timer
    ans_give = 0;
    timer = 15;
} else if(timer < 0){
    //time up
    //set new timer
    timer = 15;
    opt1.setEnabled(true);
    opt2.setEnabled(true);
    opt3.setEnabled(true);
    opt4.setEnabled(true);

    if(count == 9){
        //Pass Question number 9 (now display question number 10)
        next.setEnabled(false);
        submit.setEnabled(true);
    }

    if(count == 0){ //submit button
        if(groupOptions.getSelection() == null){
            usernames[count][0] = "";
        }else{
            usernames[count][0] =
            groupOptions.getSelection().getActionCommand();
        }

        //check answer
        for(int i = 0;i < usernames.length;i++){
            if(usernames[i][0].equals(answers[i][1])){
                score += 1;
            } else{
                score += 0;
            }
        }

        setVisible(false);
        //score
        new Score(name, score);
    }

    else{
        //next button
        if(groupOptions.getSelection() == null){
            //no select answer
            usernames[count][0] = "";
        }else{
            usernames[count][0] =
            groupOptions.getSelection().getActionCommand();
        }

        count++;

        //start new number question
        Start(count);
    }

    public void Start(int count){
        System.err.println("Now Count = "+count);

        qno.setText(""+ (count + 1) + ".");
        question.setText(questions[count][0]);
        opt1.setText(questions[count][1]);
        opt2.setText(questions[count][2]);
        opt3.setText(questions[count][3]);
        opt4.setText(questions[count][4]);

        opt1.setActionCommand(questions[count][1]);
        opt2.setActionCommand(questions[count][2]);
        opt3.setActionCommand(questions[count][3]);
        opt4.setActionCommand(questions[count][4]);

        groupOptions.clearSelection();
    }

    public static void main(String[] args){
        new Quiz("User");
    }
}
```


o o o o

ALGORITHM

Score Class

Score(name: String, score: number):

- Algorithm: เมธอดนี้ใช้ในการแสดงคะแนนที่ผู้เล่นได้รับ และเมื่อผู้เล่นเลือก "Play Again" จะเริ่มเกมใหม่ หรือเมื่อเลือก "Exit" จะออกจากแอปพลิเคชัน

```
Score.java

package quiz.application;

import java.awt.BorderLayout;
import java.awt.Color;
import java.awt.Font;
import java.awt.Image;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;

public class Score extends JFrame implements ActionListener{
    JButton playAgain,exitButton;

    public Score(String name,int score) {
        setBounds(400,150,750,550);
        getContentPane().setBackground(Color.WHITE);
        setLayout(null);

        //image 1
        ImageIcon i1 = new
        ImageIcon(ClassLoader.getSystemResource("icons/score.png"));
        JLabel image = new JLabel(i1);
        image.setBounds(0,5,360,500);
        add(image);

        //Score
        JLabel lblscore = new JLabel("Your score is "+score);
        lblscore.setBounds(425, 200,700,40);
        lblscore.setFont(new Font("Playfair Display",Font.BOLD,32));
        add(lblscore);

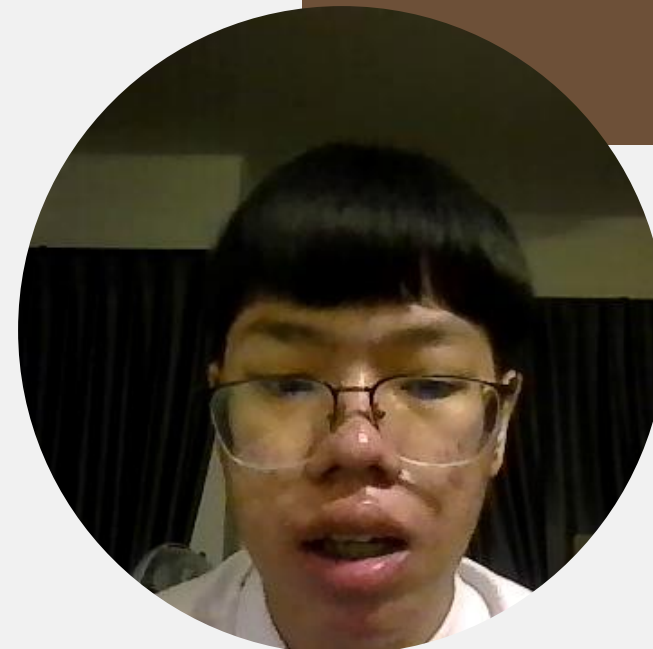
        //Play again button
        playAgain = new JButton("Play Again");
        playAgain.setBounds(470,270,120,30);
        playAgain.setBackground(Color.decode("#644E37"));
        playAgain.setForeground(Color.WHITE);
        playAgain.addActionListener(this);
        add(playAgain);

        //Exit button
        exitButton = new JButton("Exit");
        exitButton.setBounds(640,450,80,30);
        exitButton.setBackground(Color.red);
        exitButton.setForeground(Color.WHITE);
        exitButton.addActionListener(this);
        add(exitButton);

        setVisible(true);
        setResizable(false); // Lock Resolution
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }

    public void actionPerformed(ActionEvent ae){
        if(ae.getSource() == playAgain){
            setVisible(false);
            new Login();
        }
        else if(ae.getSource() == exitButton){
            //Exit app
            System.exit(0);
        }
    }

    public static void main(String[] args) {
        new Score("User",0);
    }
}
```

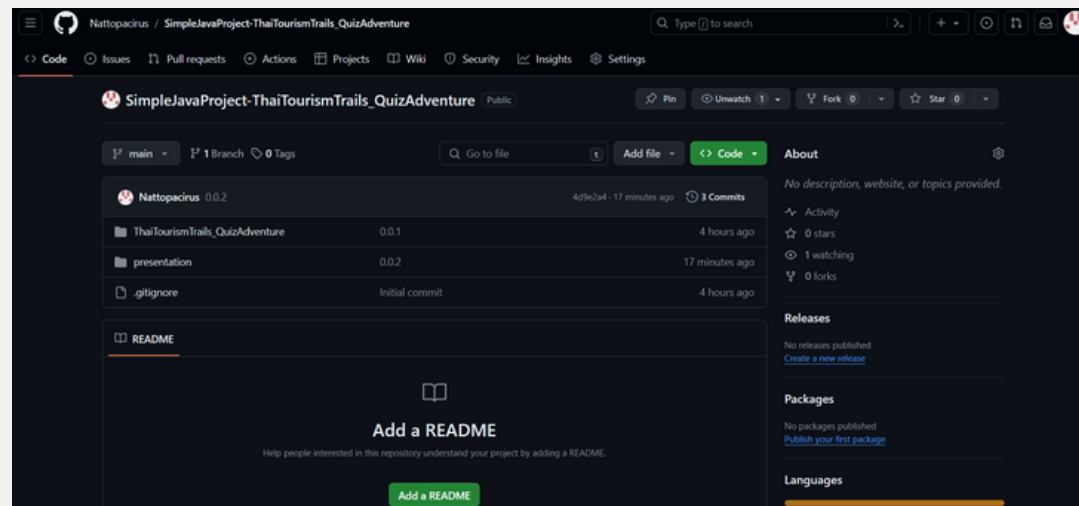


o o o o

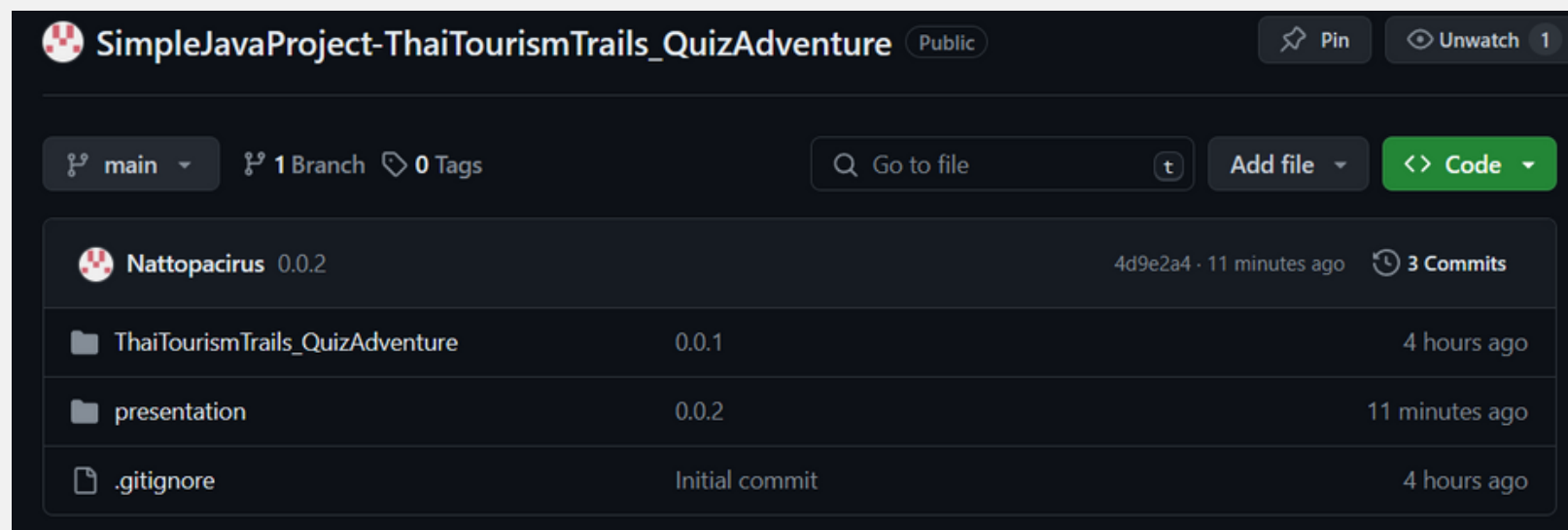
INSTALLATION

1.เข้า website ->

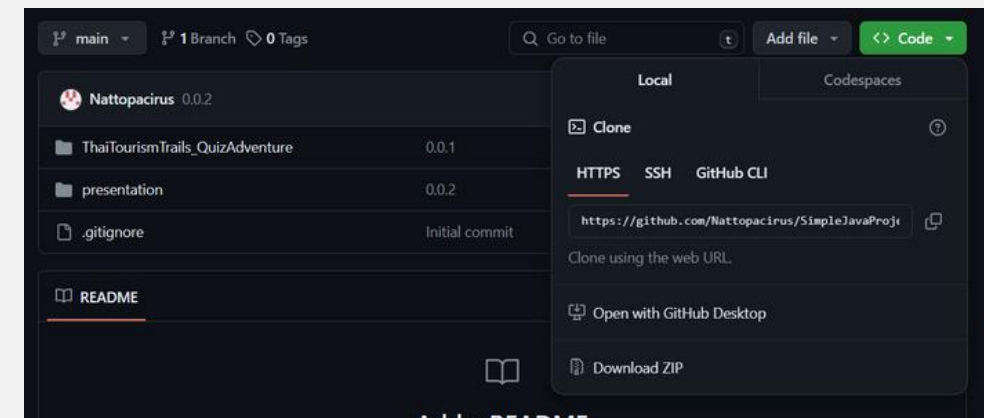
https://github.com/Nattopacirus/SimpleJavaProject-ThaiTourismTrails_QuizAdventure



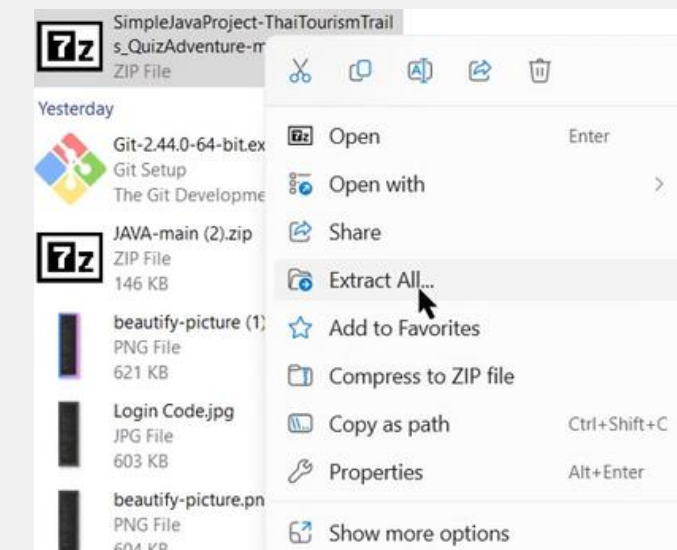
2. Click ที่ปุ่ม code



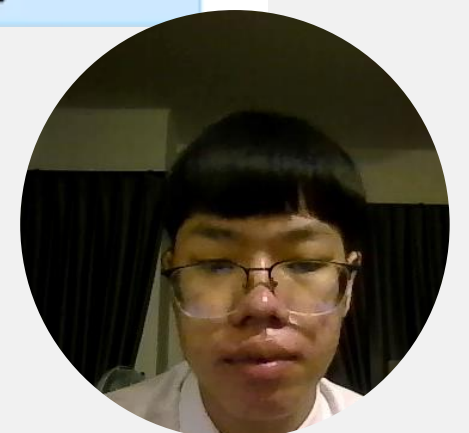
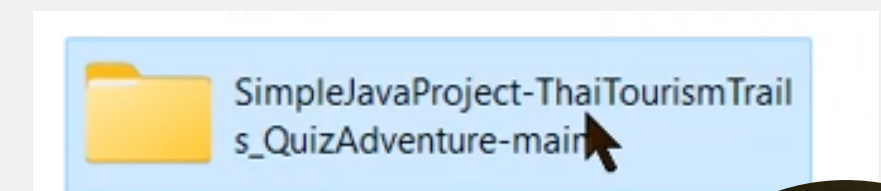
3. Click ที่ปุ่ม Download ZIP เพื่อ download file (source code + jar) ของโปรแกรม



4. เมื่อได้ไฟล์ zip จากการDownloadมาแล้วก็ทำการ extract file ออกมา



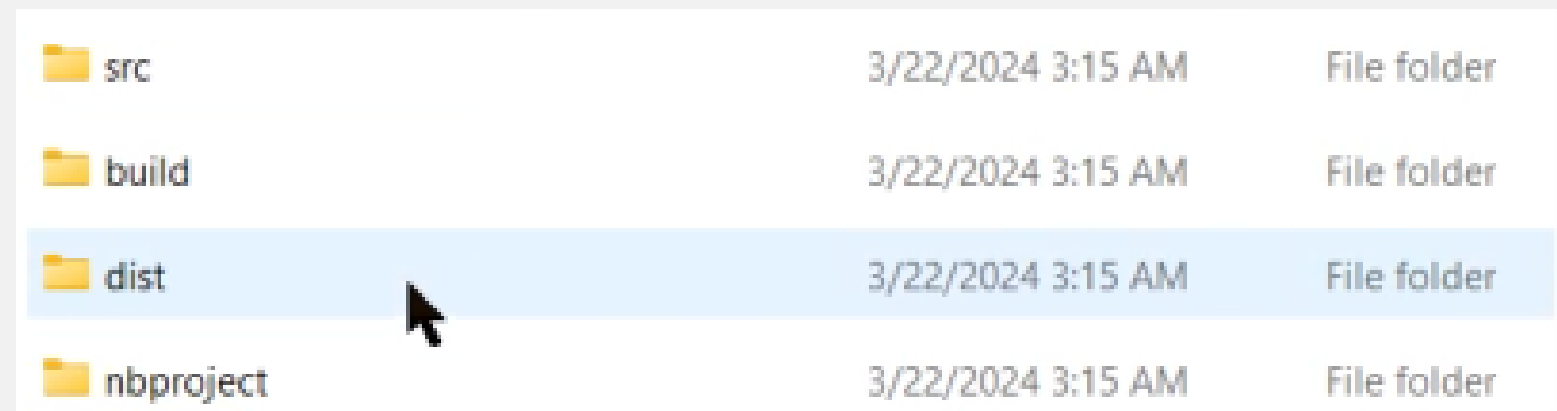
5. และกด folder ที่ทำการ extract file มา



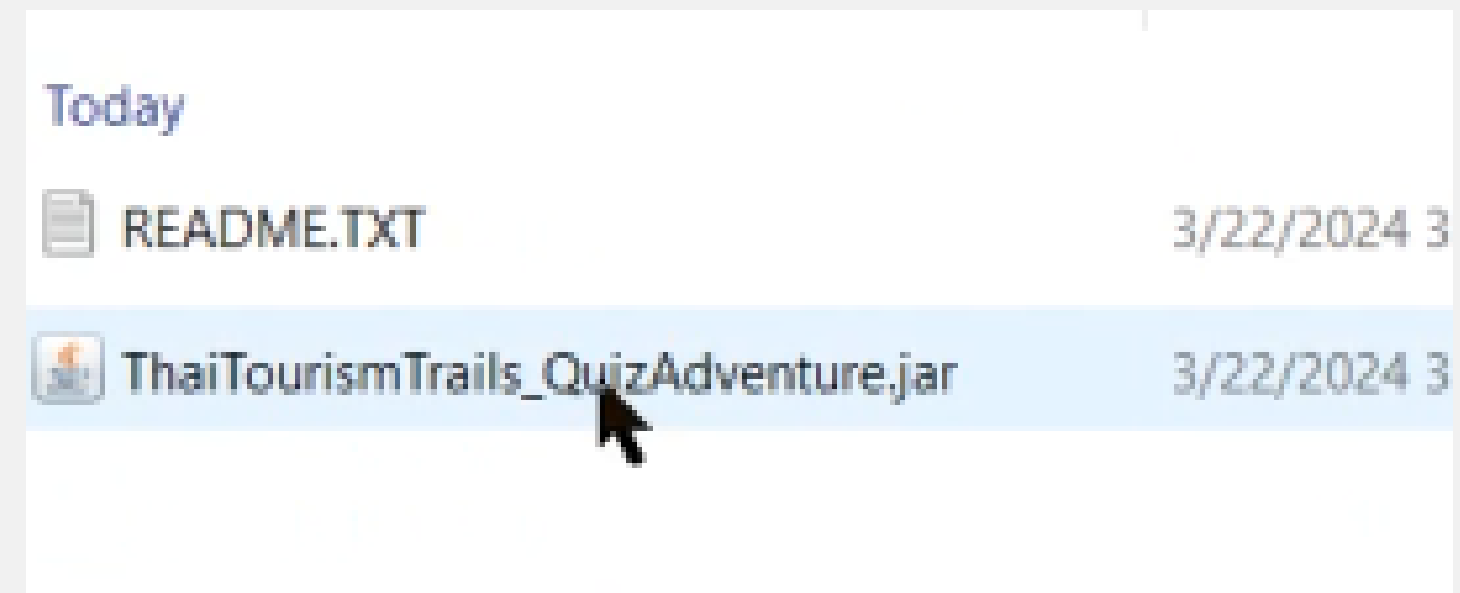
oooo

INSTALLATION

6. กดเข้าไปที่ ...\\SimpleJavaProject-ThaiTourismTrails_QuizAdventure-main\\ThaiTourismTrails_QuizAdventure\\dist



7. แล้วก็เจอกับโปรแกรม ThaiTourismTrails_QuizAdventure.jar สามารถกดใช้ได้เลย

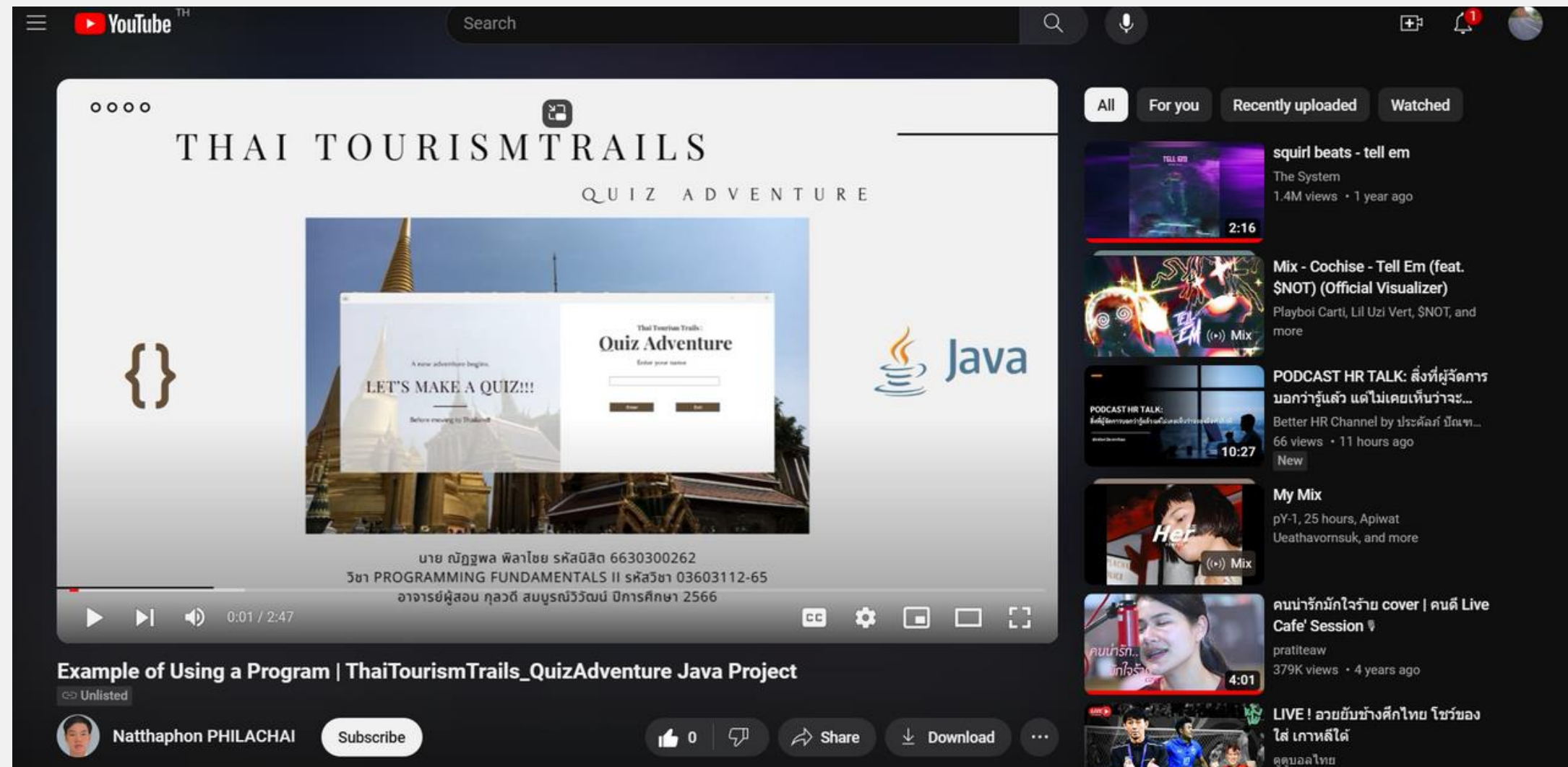


(ก่อนที่จะใช้งานโปรแกรมต้องลง java ก่อน -> <https://www.java.com/en/>)

*หมายเหตุ: ผู้พัฒนาใช้ Java เวอร์ชัน: java 21.0.2 2024-01-16 LTS

○○○○

EXAMPLE OF USING A PROGRAM



Youtube : <https://youtu.be/uKEi0TSQzCg>



THANK
YOU

