



โครงงาน

เรื่อง เกม Plane Game

เสนอ

ผศ.ดร.กฤตคม ศรีจิรานนท์

โดย

นางสาวรุจิราพร บุญอุ้ม 6309700075 ลำดับที่ 2

รายงานนี้เป็นส่วนหนึ่งของวิชา คพ.111

หลักสูตรวิทยาการคอมพิวเตอร์ (เทคโนโลยีการเรียนรู้)

คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยธรรมศาสตร์

ภาคเรียนที่ 2 ปีการศึกษา 2567

## คำนำ

รายงานฉบับนี้เป็นส่วนหนึ่งของรายวิชา คพ.111 แนวคิดเชิงวัตถุซึ่งรายวิชานี้คณะผู้จัดทำได้มีโอกาส  
แสดงความสามารถของแต่ละคนในการออกแบบการเขียนโปรแกรมที่ได้ศึกษามาโดยนำองค์ความรู้ต่าง ๆ มา  
ประยุกต์ใช้ให้เกิดเป็นผลงานชิ้นนี้ ทำให้สามารถเรียนรู้และฝึกฝนประสบการณ์ได้เป็นอย่างดี รายงานฉบับนี้  
ผู้จัดทำได้รวบรวมสิ่งที่ช่วยสรรสร้างให้เกิดผลงานชิ้นนี้และความรู้ที่ได้นำมาใช้ในระหว่างการสร้างผลงาน  
ผู้จัดทำหวังเป็นอย่างยิ่งว่ารายงานฉบับนี้จะเป็นประโยชน์สูงสุดแก่ผู้ที่ได้อ่าน อย่างถึงที่สุด

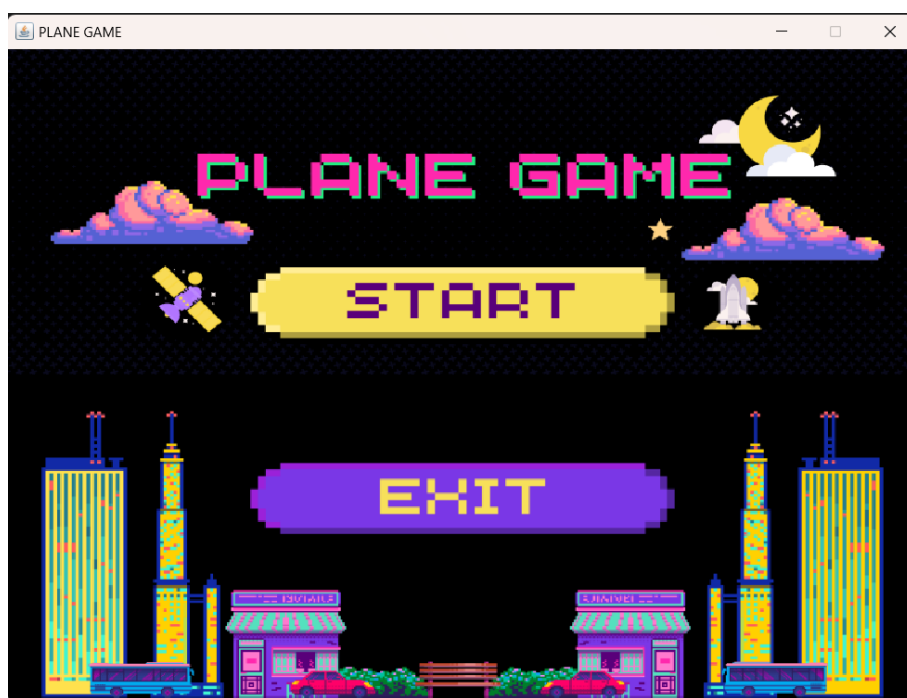
ผู้จัดทำ

## สารบัญ

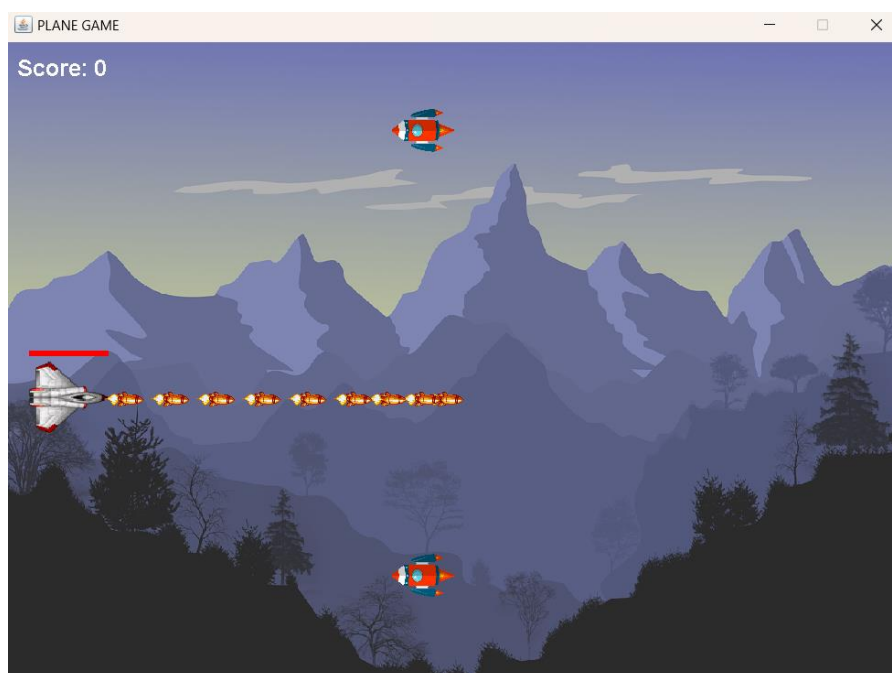
| เรื่อง              | หน้า |
|---------------------|------|
| คำนำ.....           | ก    |
| สารบัญ.....         | ๗    |
| วิธีการเล่นเกม..... | 1    |
| Class Diagram.....  | 5    |
| คลาส Player.....    | 6    |
| คลาส Bullet.....    | 6    |
| คลาส Rocket.....    | 7    |
| คลาส Monter.....    | 7    |
| คลาส UFO.....       | 8    |
| คลาส Menu.....      | 8    |
| คลาส Sound.....     | 9    |
| คลาส PanelGame..... | 9    |
| คลาส Main.....      | 10   |
| ที่มาของโค้ด.....   | 10   |

## วิธีการเล่นเกม

- เมื่อกดเข้าเกมจะเจอหน้าเมนูเริ่มเกม โดยในหน้านี้จะมีอยู่ 2 ปุ่ม คือ 1.Start เมื่อกดปุ่มนี้แล้วจะเข้าสู่หน้าเกมและเริ่มเล่นเกม 2.Exit เมื่อกดปุ่มนี้แล้วจะทำการกดปิดเกม



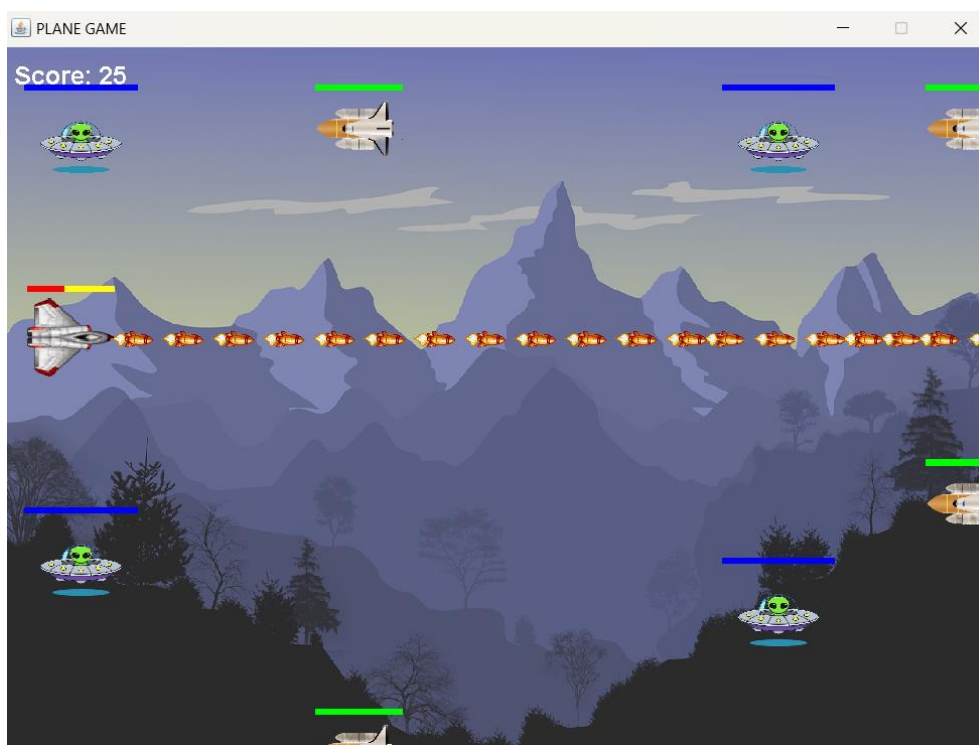
- เมื่อเข้าสู่หน้าเล่นเกม สามารถกดปุ่มขึ้น - ลง เพื่อขยับเครื่องบินและยิง Rocket



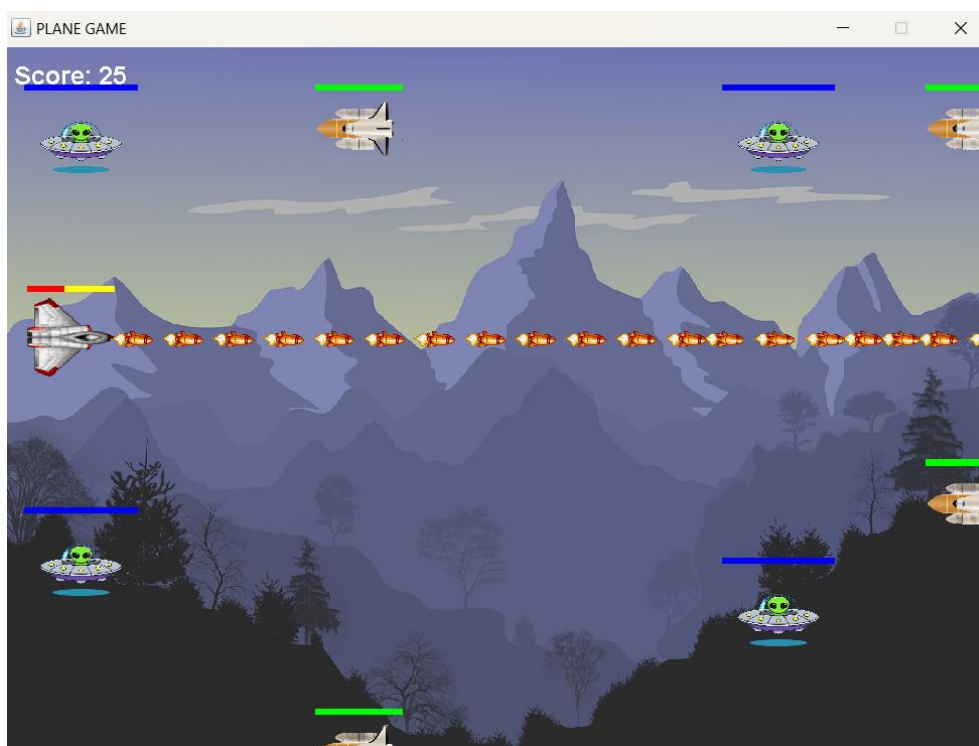
- เมื่อยิง Rocket จนได้มากกว่าหรือเท่ากับ 8 คะแนน แล้วจะมี Monter ออกมา



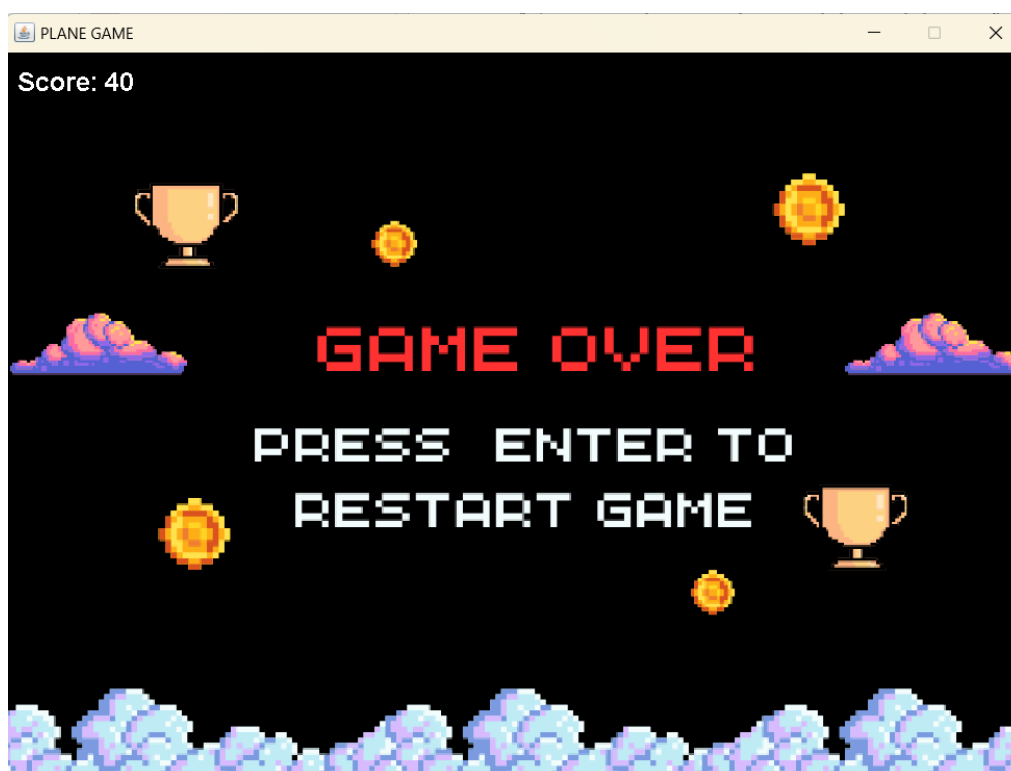
- เมื่อยิง Monter จนได้มากกว่าหรือเท่ากับ 15 คะแนน แล้วจะมี UFO ออกมา



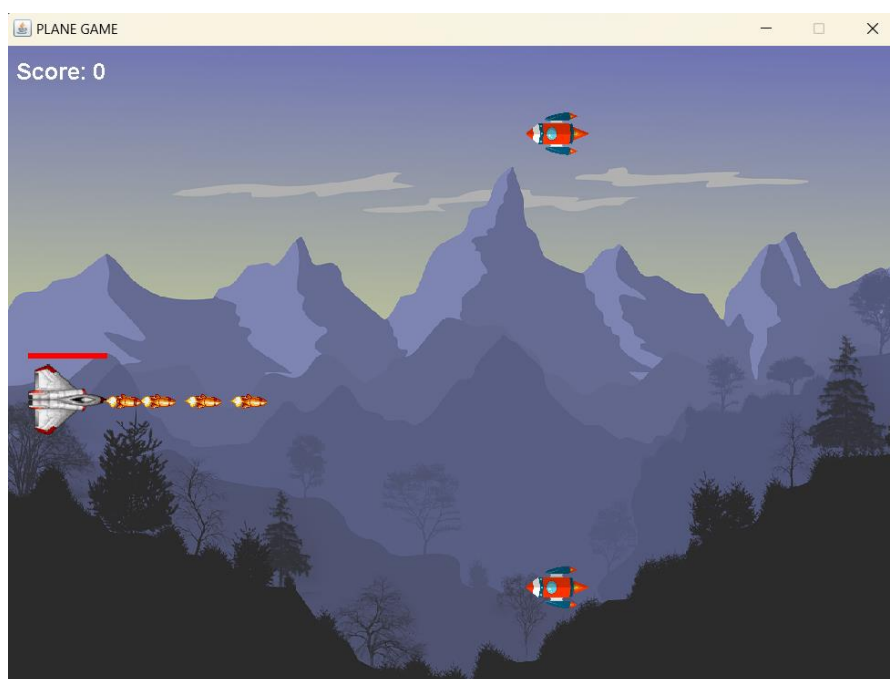
- ถ้า Rocket Monter หรือ UFO ชนผู้เล่น เลือดของผู้เล่นจะลดลง



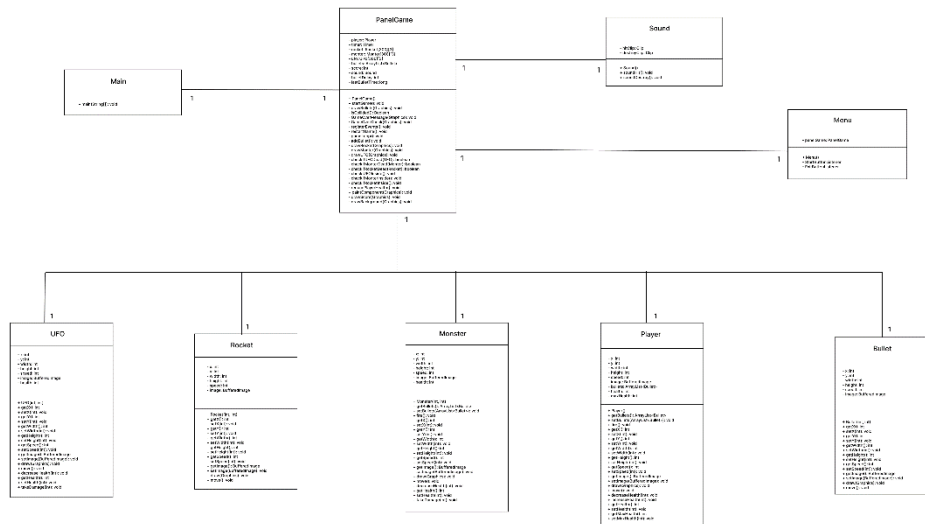
- ถ้าเลือดของผู้เล่นหมดจะเข้าสู่หน้า Game Over โดยในหน้านี้จะแสดงคะแนนที่ผู้เล่นทำได้



- ถ้าต้องการเริ่มเกมใหม่ สามารถกดปุ่ม Enter เพื่อเริ่มเกมใหม่ได้



### Class Diagram





## คลาส Player

ใช้สำหรับจัดการข้อมูลและพฤติกรรมของผู้เล่นในเกม ภายในคลาสมีตัวแปรเกี่ยวกับตำแหน่ง (x, y) ของผู้เล่น ขนาดของตัวละคร (width, height) ความเร็ว (speed) รูปภาพของตัวละคร (image) รวมถึงค่าเลือด (health) และค่าเลือดสูงสุด (maxHealth) โดยคอนสตรักเตอร์จะกำหนดค่าเริ่มต้นของตัวแปรทั้งหมด รวมถึงโหลดรูปภาพของตัวละครจากไฟล์บนดิสก์ด้วย

- เมทอด fire() ใช้สำหรับยิงกระสุน โดยคำนึงถึงตำแหน่งของจุดหัวใจของตัวละคร และสร้างกระสุนใหม่เพิ่มลงใน ArrayList ของกระสุน
- เมทอด draw() ใช้สำหรับวาดตัวละครของผู้เล่นบนกราฟิกส์ รวมถึงวาดแถบสีเหลืองแสดงค่าเลือด และแถบสีแดงแสดงสถานะเลือด
- เมทอด move() ใช้สำหรับเคลื่อนที่ตัวละครของผู้เล่น โดยเช็คว่ายังอยู่ในขอบเขตของหน้าจอหรือไม่ เพื่อป้องกันการเคลื่อนที่ออกนอกขอบเขต
- เมทอด decreaseHealth() และ increaseHealth() ใช้สำหรับลดและเพิ่มค่าเลือดของผู้เล่น โดยให้ค่าเลือดไม่เกินค่าเลือดสูงสุดที่กำหนดไว้
- นอกจากนี้ยังมีเมทอด getter และ setter สำหรับตัวแปรต่าง ๆ เพื่อให้สามารถเข้าถึงและแก้ไขค่าได้จากภายนอกคลาส

## คลาส Bullet

ใช้สำหรับจัดการกระสุนในเกมของคุณ มีดังนี้:

- มีตัวแปรเกี่ยวกับตำแหน่ง (x, y) ของกระสุน และขนาดของกระสุน (width, height) เพื่อกำหนดตำแหน่งและขนาดของกระสุนในการวาดบนหน้าจอ
- มีความเร็ว (speed) เพื่อกำหนดความเร็วในการเคลื่อนที่ของกระสุน
- มีรูปภาพของกระสุน (image) เพื่อแสดงกระสุนบนหน้าจอ
- มีคอนสตรักเตอร์ที่รับพารามิเตอร์เริ่มต้นของตำแหน่ง (x, y) และโหลดรูปภาพของกระสุนจากไฟล์
- นอกจากนี้ยังมีเมทอด getter และ setter เพื่อเข้าถึงและแก้ไขค่าของตัวแปรในคลาส และเมทอด draw() เพื่อวาดกระสุนบนกราฟิกส์ และเมทอด move() เพื่อเคลื่อนที่กระสุนในแนวแกน x ตามความเร็วที่กำหนด

## คลาส Rocket

ใช้สำหรับจัดการกับ **Rocket** ที่จะปรากฏในเกมของคุณ โดยมีดังนี้:

- มีตัวแปรเกี่ยวกับตำแหน่ง (x, y) ของ **Rocket** และขนาดของ **Rocket** (width, height) เพื่อกำหนดตำแหน่งและขนาดของ **Rocket** ในการวาดบนหน้าจอ
- มีความเร็ว (speed) เพื่อกำหนดความเร็วในการเคลื่อนที่ของ **Rocket**
- มีรูปภาพของ **Rocket** (image) เพื่อแสดง **Rocket** บนหน้าจอ
- มีคอนสตรักเตอร์ที่รับพารามิเตอร์เริ่มต้นของตำแหน่ง (x, y) และโหลดรูปภาพของ **Rocket** จากไฟล์บนดิสก์
- นอกจากนี้ยังมีเมทอด getter และ setter เพื่อเข้าถึงและแก้ไขค่าของตัวแปรในคลาส และเมทอด draw() เพื่อวาด **Rocket** บนกราฟิกส์ และเมทอด move() เพื่อเคลื่อนที่ **Rocket** ในแนวแกน x ตามความเร็วที่กำหนด

## คลาส Monter

ใช้สำหรับจัดการกับ **Monter** ในเกมของคุณ โดยมีดังนี้:

- มีตัวแปรเกี่ยวกับตำแหน่ง (x, y) ของ **Monter** และขนาดของตัวละคร (width, height) เพื่อกำหนดตำแหน่งและขนาดของ **Monter** ในการวาดบนหน้าจอ
- มีความเร็ว (speed) เพื่อกำหนดความเร็วในการเคลื่อนที่ของ **Monter**
- มีรูปภาพของ **Monter** (image) เพื่อแสดง **Monter** บนหน้าจอ
- มีค่าเลือด (health) เพื่อติดตามและจัดการกับสุขภาพของ **Monter** นอกจากนี้ยังมีเมทอด getter และ setter เพื่อเข้าถึงและแก้ไขค่าของตัวแปรในคลาส และเมทอด draw() เพื่อวาด **Monter** บนกราฟิกส์ และเมทอด move() เพื่อเคลื่อนที่ **Monter** ในแนวแกน x ตามความเร็วที่กำหนด และเมทอด takeDamage() เพื่อลดค่าเลือดของ **Monter** เมื่อได้รับการโจมตี

## คลาส UFO

ใช้สำหรับจัดการกับ UFO ที่จะปรากฏในเกมของคุณ โดยมีดังนี้:

- มีตัวแปรเกี่ยวกับตำแหน่ง (x, y) ของ UFO และขนาดของยานอวกาศต่างดาว (width, height) เพื่อกำหนดตำแหน่งและขนาดของ UFO ในการวาดบนหน้าจอ
- มีความเร็ว (speed) เพื่อกำหนดความเร็วในการเคลื่อนที่ของ UFO
- มีรูปภาพของ UFO (image) เพื่อแสดง UFO บนหน้าจอ
- มีค่าเลือด (health) เพื่อติดตามและจัดการกับสุขภาพของ UFO
- นอกจากนี้ยังมีเมทอด getter และ setter เพื่อเข้าถึงและแก้ไขค่าของตัวแปรในคลาส และเมทอด draw() เพื่อวาด UFO บนกราฟิกส์ และเมทอด move() เพื่อเคลื่อนที่ UFO ในแนวแกน x ตามความเร็วที่กำหนด และเมทอด takeDamage() เพื่อลดค่าเลือดของ UFO เมื่อถูกโจมตี

## คลาส Menu

ใช้สำหรับแสดงเมนูเริ่มต้นของเกม PLANE GAME โดยมีดังนี้:

- กำหนดหัวข้อหน้าต่าง (setTitle) เป็น "PLANE GAME" และกำหนดขนาดของหน้าต่าง (setSize) เป็น 800x600 พิกเซล
- กำหนดปุ่ม Start และ Exit โดยใช้ JButton และกำหนดรูปภาพให้กับปุ่มด้วย ImageIcon จากไฟล์รูปภาพที่อยู่ในที่เก็บไฟล์ของเกม
- เมื่อคลิกปุ่ม Start จะทำการลบเนื้อหาทั้งหมดในหน้าต่าง (removeAll) และเพิ่ม PanelGame ลงในหน้าต่าง (add(panelGame)) เพื่อเริ่มเกม
- เมื่อคลิกปุ่ม Exit จะส่งคำสั่งให้โปรแกรมจบการทำงาน (System.exit(0))
- โดย PanelGame นั้นจะเป็น JPanel ที่ใช้สำหรับการแสดงเกม PLANE GAME และจะถูกเรียกใช้เมื่อกดปุ่ม Start ในเมนูของเกม

## คลาส Sound

ใช้สำหรับเล่นเสียงในเกม PLANE GAME โดยมีดังนี้:

- ใช้งานไลบรารี javax.sound.sampled เพื่อจัดการกับเสียง
- มีตัวแปร hitClip และ destroyClip เพื่อเก็บเสียงที่ต้องการเล่น
- ในเมทอด constructor จะโหลดไฟล์เสียง hit.wav และ destroy.wav จากที่อยู่ที่กำหนด และเก็บ Clip เสียงไว้ใน hitClip และ destroyClip
- มีเมทอด soundHit() เพื่อเล่นเสียง hit.wav และ soundDestroy() เพื่อเล่นเสียง destroy.wav

## คลาส PanelGame

ใช้สำหรับแสดงเกมของ PLANE GAME บนหน้าต่างของ Java Swing โดยมีดังนี้

กำหนดตัวแปรและอ็อบเจกต์:

- มีตัวแปร player เพื่อเก็บข้อมูลของผู้เล่น
- มีตัวแปร rocket, monter, ufo เก็บข้อมูลของ rocket, monter, ufo
- มี ArrayList เก็บลูกศรที่ผู้เล่นยิงออกไป
- มีตัวแปร score เก็บคะแนนของผู้เล่น

สร้างเงื่อนไขการเคลื่อนไหว:

- ใช้ Timer เพื่อให้เกมมีการเคลื่อนไหวโดยตลอด
- ใช้ Iterator เพื่อเช็คการชนกับ rocket, monter, ufo และยิงลูกศรโดยใช้ KeyListener

การเช็คและการปรับปรุงสถานะ:

- เมื่อยิงลูกศรตกเข้าสู่ rocket, monter, ufo จะลบลูกศรและลดพลังชีวิตของ rocket, monter, ufo
- เมื่อผู้เล่นชนกับ rocket, monter, ufo จะลดพลังชีวิตของผู้เล่น
- เมื่อผู้เล่นพลังชีวิตหมด จะแสดงข้อความ Game Over

การวาดกราฟิก:

- ใช้ไฟล์รูปภาพสำหรับพื้นหลัง, rocket, monter, ufo
- ใช้ drawImage เพื่อวาดรูปภาพต่างๆ และ fillRect เพื่อวาดแถบแสดงพลังชีวิต

### คลาส Main

เป็นคลาสหลักที่ใช้สำหรับเริ่มเกม PLANE GAME โดยเรียกใช้เมทอด `startGame()` จากคลาส `PanelGame` เพื่อเริ่มเกมขึ้นมาแสดงบนหน้าต่างของ Java Swing ในโปรแกรม Java โดยที่จะเรียกเมทอด `startGame()` นี้ผ่านคลาส `PanelGame`

### ระบุแหล่งที่มาของโค้ด

- `private Boolean isCollided()` ใน `PanelGame` นำมาจาก ChatGPT
- `private boolean checkIfRocketDead(Rocket rockets)` ใน `PanelGame` นำมาจาก ChatGPT
- `public static void startGame()` ใน `PanelGame` นำมาจาก ChatGPT
- `private void checkIfRocketInside()` ใน `PanelGame` นำมาจาก ChatGPT
- `private class StartButtonListener implements ActionListener` ใน `Menu` นำมาจาก ChatGPT
- `public Sound()` ใน `Sound` นำมาจาก ChatGPT