A circular green and gold emblem

AI-generated content may be incorrect.

รายงาน

เรื่อง **Inferno & Tide**

**จัดทำโดย**

**นายพงศกร เกศนาคินทร์ 6730300345**

**นายรพีพงศ์ ธนปกรณ์เกียรติ 6730300507**

**เสนอ**

**อาจารย์กุลวดี สมบูรณ์วิวัฒน์**

**รายงานเล่มนี้เป็นส่วนหนึ่งของวิชาหลักการโปรแกรมเบื้องต้น II**

**มหาวิทยาลัยเกษตรศาสตร์ วิทยาเขตศรีราชา**

**ภาคเรียนที่ 2 ปีการศึกษา 2567**

1. **Problem Statement**

ปัญหาและแรงบันดาลใจ (Motivation/Pain Point)

ช่วงกระแสเกมมีความหลากหลายมากขึ้น ผู้เล่นต่างมองหาประสบการณ์ที่ไม่ซ้ำแบบใครและให้ความรู้สึกมีส่วนร่วมกับเกมมากขึ้น โครงงาน “**Inferno & Tide**” ถูกพัฒนาขึ้นเพื่อตอบโจทย์นี้ โดยมีแรงบันดาลใจจากความต้องการที่จะผสมผสานความสนุกของเกม Tower defense ( TD ) เป็น ประเภทย่อย ของ เกมวางแผน ที่มีเป้าหมายในการป้องกันดินแดนหรือทรัพย์สินของผู้เล่นโดยขัดขวางผู้โจมตีของศัตรูหรือหยุดศัตรูไม่ให้เข้าถึงทางออก

* Pain Point: o
* ผู้เล่นมักประสบปัญหาในการเลือกใช้ในการวางตำแหน่งต่างที่ในพื้นที่เฉพาะนั้น
* ผู้เล่นต้องการประสบการณ์ที่ผสมผสานระหว่างยุทธ์แบบผลัดตาการเลือกกลยุทธ์และการวางตำแหน่งขององค์ประกอบการป้องกันถือเป็นกลยุทธ์ที่สำคัญของประเภทนี้
* โดยทั่วไปการสร้างโครงสร้างต่างๆ ที่หลากหลายซึ่งทำหน้าที่บล็อก ขัดขวาง โจมตี หรือทำลายศัตรูโดยอัตโนมัติ ต้องคิดดด้วยมันเหมาะไหม

การแก้ปัญหาเหล่านี้จึงเป็นแรงบันดาลใจให้ทีมพัฒนานำเอาแนวคิดใหม่ ๆ มาสร้างเกมที่มีความสมดุลระหว่างความบันเทิงและความท้าทายด้าน เทคนิค

1. **Features**

โครงงาน **Inferno & Tide** มีคุณลักษณะหลักที่ช่วยเพิ่มความสนุกและความท้าทายให้กับผู้เล่นดังนี้:

* **ระบบศัตรู:**

ระบบนี้ทำให้ มอนสเตอร์เกิดขึ้นมาและเคลื่อนที่ไปตามเส้นทาง โดยศัตรูแต่ละตัวจะมี พลังชีวิต (HP), ความเร็ว (Speed), และค่ารางวัล (Reward) เมื่อถูกกำจัด

* **ระบบเดินตามเส้นทางเดิน:**

ระบบนี้ทำให้ ศัตรูเดินจากจุดเริ่มต้นไปยังฐานของผู้เล่น โดยไม่ออกนอกเส้นทาง

* **ระบบฐาน:**

ฐานเป็น เป้าหมายของศัตรู ถ้าศัตรูเดินมาถึงฐานได้ HP ของฐานจะลดลง และหาก HP หมด เกมจะจบลง

* **ระบบหอคอย:**

หอคอยเป็นอาวุธหลักของผู้เล่น ใช้ป้องกันฐานโดย โจมตีศัตรูที่เข้ามาในระยะ

* **ระบบยิงกระสุน:**

ระบบนี้ทำให้ หอคอยสามารถยิงกระสุนใส่ศัตรู โดยกระสุนอาจมี ความเร็ว และ ดาเมจ

* **ระบบ WAVE:**

ระบบนี้ควบคุมการปล่อยศัตรูเป็นรอบๆ **(Wave-Based Gameplay)** โดยศัตรูจะมาเป็น **ชุด** และค่อยๆ ยากขึ้น

**คุณลักษณะเพิ่มเติม** **(Optional Features)**

* **อัพเดตสถานะของศัตรู:**

ถ้ากาหว้าเลือกศัตรูหมดเมื่อไหร่ก็จะหายไปทันทีหรือตายนั้นเอง

* **ระบบเงิน:**

เมื่อกำจัดได้ก็จะได้เงินมาวางฐานเพิ่มเติม

1. **Program Design**

**User Interface (UI)**

* Main Menu:

เมื่อเข้าสู่จะพบหน้าตา UI ก็จะมีตัวเลือกให้เริ่มเกม และ ปุ่มออก

* Sound:

ในแต่ละหน้าจะมีเสียงเพลงประกอบในแต่ละฉาก

* Game play Screen:

หน้าจอการเล่นจะแสดง โชว์เลือดเรา ฐานที่ตั้ง ศัตรู และเส้นทางเดิน การแสดงผลแบบนี้ช่วยให้ผู้เล่นสามารถวางป้อมฐานเพื่อจัดการศัตรูมาแต่ละเวฟ และคิดแผนว่าจะวางฐานที่กำหนดส่วนนั้นยังไงให้อยู่รอดนานที่สุด

* Select Bar:

เป็นแถบให้ฐานมาวางเพิ่มเติมในสนามเกมในแผนที่

* Pause Menu:

ช่วยให้ผู้เล่นหยุดพักชั่วคราวเพื่อจัดการธุระส่วนตัวและไม่ว่างช่วงหนึ่ง ให้โปรแกรมพักค้างหน้านั้นหยุดไว้พอเสร็จธุระเสร็จ

* 1. Class Diagram

การออกแบบโครงสร้างโปรแกรมใช้แนวคิดแบบ Object Oriented Programming (OOP) โดยแบ่งคลาสหลักๆ ออกเป็นส่วน ๆ ตามฟังก์ชันการทำงาน ดังนี้

* AppClass:

เป็นแอพหลักในการแสดงผลหน้าจอเริ่มต้นของเกม และเป็นตัวเริ่มต้น

* Army:

เป็นคราสในการจัดการหน่วยทหารในเกม การเพิ่มหน่วยทหารลง, การโจมตีศัตรูในระยะยิง, การหยุดการยิง, เมธอดการยิงศัตรู, และเมธอดการตรวจสอบว่าศัตรูอยู่ในระยะยิง

* EnemyClass:

เป็นคลาสที่ใช้ในการจัดการศัตรูในเกม

* EnemyPathClass:

เป็นคลาสที่ใช้ในการจัดการเส้นทางของศัตรูในเกม ให้โชว์มาหน้าจอ และโชว์สีเส้นทาง

* Game controllerClass:

เป็นคราสที่ใช้ในการควบคุมการทำงานของเกม ตั้งแต่เริ่มต้นของเกมไม่ว่าจะหอคอยผ่านไปหลาย wave แล้วไปก็ตามศัตรูก็จะเกิดถี่เรื่อยๆ ควบคุมศัตรูให้เดินตามเส้นไป ควบคุมกระสุนในการยิง มีการใช้จ่ายเงิน และได้จากการฆ่าศัตรู ควบคุมพลังชีวิตว่าปัจจุบันมีเท่าไหร่ ก็อัพเดตค่าเวลานั้นและเมื่อไหร่หมดเกมก็จบจง

* GameLevelClass:

เป็นคราสที่ใช้ในการจัดระดับของเกม เพื่อให้เกมมีความท้าทายระดับยากขึ้นเมื่อผ่านไปหลาย wave

* GamePlayClass:

เป็นคราสที่ใช้ในการจัดการเล่นเกม โดยจัดการค่า hp, money, wave , เพลงพื้นหลัง ตัวหอคอยที่เลือกในเกม ยังจะจัดการการยิงของหอคอย ทั้งเส้นทางและการคลิ๊กเม้าส์เพื่อมาวางหอคอย

* HpClass:

คลาสที่ใช้ในการจัดการพลังชีวิตของฐานในเกม และอัพเดทพลังชีวิตพื้นฐานที่เปลี่ยนแปลงไป และเมื่อ พลังชีวิตหมดเกมก็จะจบ

* MoneyClass:

คราสที่ใช้ในการจัดการระบบเงินในเกมเป็นการแสดงจำนวนเงินบนหน้าจอ และยังเก็บค่าปัจจุบัเล่น และเมื่อฆ่าศัตรูได้ก็จะได้เงิน เมื่อ wave ก็จะได้รับรางวัล

* ProjectileClass:

คลาสที่ใช้ในการจัดการการยิงกระสุนในเกมใช้ในการสร้างกระสุนและกำหนดเส้นทางการเคลื่อนที่ของกระสุน เพื่อตรวจจับการชนระหว่างกระสุนและศัตรู ลบกระสุนและศัตรูออกจาก Pane เมื่อมีการชนกัน

* WaveClass:

เป็นคลาสที่ใช้ในการจัดการการเกิดของศัตรูในเกม การสร้างศัตรูใน wave โดยมีการคำนวณพลังชีวิตและความเร็วของศัตรู การเคลื่อนที่ของศัตรูตามเส้นทางที่กำหนด

* Ui:

เป็นหน้าอินเทอร์เฟสถูกออกเพื่อดูเข้าใจง่าย และ องค์ประกอบที่เข้าใจง่าย

การออกแบบคลาสเหล่านี้จะช่วยให้โปรแกรมมีความยืดหยุ่น สามารถขยายหรือแก้ไขเพิ่มเติมได้ง่ายต่ออนาคตข้างหน้า พร้อมทั้งรักษาความเป็นโมดูลาร์ (Modular) ทำให้การพัฒนาและบำรุงรักษาโปรแกรมเป็นไปอย่างมีประสิทธิภาพ

1. **Program Installation**

**4.1ขั้นตอนการติดตั้งโปรแกรม**

เพื่อให้ผู้ใช้สามารถติดตั้งและเริ่มใช้งานเกม **“Inferno & Tide” ได้อย่างง่าย รายละเอียดขั้นตอนการติดตั้งดังนี้**

1. **Clone Repository จาก GitHub:**

เปิดเทอร์มินัลหรือ command prompt แล้วใช้คำสั่งต่อไปนี้เพื่อดาวโหลดโค้ดโปรเจกต์

**https://github.com/korn845/256702-F2-Team13-345-507?tab=readme-ov-file#**

**2.ติดตั้ง JAVA และ Maven:**

**ตรวจดูให้แน่ใจว่าคุณติดตั้ง Java Development Kit (JDK) เวอร์ชันที่รองรับและ Maven เพื่อจัดการน dependency และการ build โดยการ**

* **ดาวน์โหลด JDK**
* **ติดตั้ง Maven**

**3. ติดตั้ง Dependencies และ Build โดยการ:**

**ไปที่โฟลเดอร์โปรเจกต์ที่ clone มาจาก GitHub จากนั้นรันคำสั่ง:**

**mvn install**

คำสั่งนี้จะดาวน์โหลด library และ dependency ที่จำเป็นสำหรับการทำงานของโปรแกรม

**4. รันโปรแกรม:**

**หลังจากที่ build สำเร็จแล้ว ให้ใช้คำสั่ง:**

**Mvn javafx:run**

เพื่อเริ่มต้นเกม หากทุกอย่างถูกต้อง หน้าจอเกมจะเปิดขึ้นและคุณสามารถเริ่มเล่นได้ทันที

**4.2. การอ้างอิงและเอกสารเพิ่มเติม**

* **เอกสารและบทเรียน:**

มีการอ้างอิงจากบทเรียนและตัวอย่างโค้ดจาก FXGL Wiki และ JavaFX

* JavaFX Documentation: <https://openjfx.io/>
* FXGL Tutorial: <https://github.com/AlmasB/FXGL/wiki/FXGL-11>
* ต้นแบบ

มอนจะขยับยังไง <https://stackoverflow.com/questions/30044870/tower-defense-game-moving-the-enemies-on-the-screen>

* วิธีการจับการเคลื่อนไหวศัตรู: <https://stackoverflow.com/questions/23553604/how-to-track-an-object-during-animation>
* **GitHub Repository:**

สำหรับการตรวจสอบโค้ดและติดตามการพัฒนา สามารถเข้าไปดูรายละเอียดและการอัปเดตเพิ่มเติมได้ที่

https://github.com/korn845/256702-F2-Team13-345-507.git