



แมพกระโดด เก็บของถึงจุดจบ
Parkour Monster Map

กฤษฎา ทวีธรรม	รหัส 65201280001
ธารทอง ชินอุดมพงศ์	รหัส 65201280011

โครงการนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ประกาศนียบัตรวิชาชีพ
แผนกวิชาช่างเทคนิคคอมพิวเตอร์ วิทยาลัยเทคนิคจันทบุรี
ปีการศึกษา 2567

แมพกระโดด เก็บของถึงจุดจบ
Parkour Monster Map

กฤษฎะ ทวีธรรม	รหัส 65201280001
ธารทอง ชินอุดมพงศ์	รหัส 65201280011

โครงการนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ประกาศนียบัตรวิชาชีพ
แผนกวิชาช่างเทคนิคคอมพิวเตอร์ วิทยาลัยเทคนิคจันทบุรี
ปีการศึกษา 2567



ใบรับรองโครงการ

แผนกวิชาช่างเทคนิคคอมพิวเตอร์ วิทยาลัยเทคนิคจันทบุรี

ชื่อโครงการ	แมพกระโดด เก็บของถึงจุดจบ (Parkour Monster Map)		
ชื่อนักศึกษา	นายภฤชชญา	ทวีธรรม	รหัส 65201280001
	นางสาวธารทอง	ชินอุดมพงศ์	รหัส 65201280011
ครูที่ปรึกษา	นายศัตราวุธ	ศรีชาติ	
ครูที่ปรึกษาร่วม	นายอาทิตย์	ชิดชอบ	

ได้รับการอนุมัติให้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรประกาศนียบัตรวิชาชีพ
สาขางานเทคนิคคอมพิวเตอร์ แผนกวิชาช่างเทคนิคคอมพิวเตอร์

คณะกรรมการสอบโครงการ

.....ประธานกรรมการ	กรรมการ
(นายอาทิตย์ ชิดชอบ)	(นายฉัตรพลฤกษ์ สีสิม)
.....กรรมการ	กรรมการ
(นายปิยพงษ์ เกตุวงษา)	(นางสาวประภาพร บันทา)

รับรองโครงการ

.....
(นายศัตราวุธ ศรีชาติ)

หัวหน้าแผนกวิชาช่างเทคนิคคอมพิวเตอร์

..... / /2567

ชื่อโครงการ	แมพกระโดด เก็บของถึงจุดจบ (Parkour Monster Map)	
นักศึกษา	นายกฤษชญา ทวีธรรม	รหัส 65201280001
	นางสาวธารทอง ชินอุดมพงศ์	รหัส 65201280011
ที่ปรึกษาโครงการ	นายศัตรารุช ศรีชาติ	
ที่ปรึกษาร่วม	นายอาทิตย์ ชิตชอบ	
สาขาวิชา	ช่างเทคนิคคอมพิวเตอร์	
ปีการศึกษา	2567	

บทคัดย่อ

ในปัจจุบันเกมแพลตฟอร์มรูปแบบการเล่นแนวปาร์กัวร์ ได้รับความนิยมอย่างมากในวงการเกม เนื่องจากเป็นเกมที่ทำนายผู้เล่นในเรื่องของการควบคุมตัวละครให้เคลื่อนไหวผ่านสิ่งกีดขวางอย่างมีประสิทธิภาพ อย่างไรก็ตาม การพัฒนาเกมในรูปแบบนี้ยังคงมีความซับซ้อน โดยเฉพาะอย่างยิ่งในการสร้างสรรค์ฉากเกม และกลไกการเคลื่อนไหวที่สมจริง และสนุกสนาน อีกทั้งผู้เล่นมักต้องการการปรับแต่งเฟรมเรต และคุณภาพของภาพให้เหมาะสมกับเครื่องที่ใช้งาน ทางคณะผู้จัดทำจึงได้สร้างแผนที่ปาร์กัวร์ 2D โดยการใช้ Unity Hub ในการพัฒนา เพื่อให้ผู้เล่นได้สัมผัสประสบการณ์การเล่นเกมที่ท้าทายและสนุกสนาน อีกทั้งยังมีตัวเลือกในการปรับแต่งค่าต่าง ๆ

ดังนั้น คณะผู้จัดทำจึงได้พัฒนาแผนที่ปาร์กัวร์ 2D ขึ้นมาเพื่อให้ผู้เล่นสามารถเล่นได้อย่างท้าทาย พร้อมทั้งมีการปรับแต่งระบบเกมให้สอดคล้องกับความต้องการของผู้เล่น โดยเฉพาะอย่างยิ่งในด้านความยืดหยุ่นในการปรับเปลี่ยนคุณภาพของกราฟิกและความเร็วของเกมให้เหมาะสมกับการใช้งาน

ผลการประเมินแผนที่ปาร์กัวร์ 2D พบว่าจากการประเมินแบบสอบถามการใช้งาน ได้ค่าเฉลี่ย ($\bar{X} = 4.21$) , (S.D. = 0.86) อยู่ในระดับมาก โดยหัวข้อที่ได้รับคะแนนมากที่สุดคือ ความสวยงาม ความทันสมัย น่าสนใจ มีค่าเฉลี่ย ($\bar{X} = 4.28$) , (S.D. = 0.93) และปุ่มต่างๆในเกมสามารถทำงานได้อย่างมีประสิทธิภาพ มีค่าเฉลี่ย ($\bar{X} = 4.29$) , (S.D. = 0.82) นอกจากนี้ การทดสอบการทำงานของระบบเกมโดยการเล่นและบันทึกผลการตอบสนองของเกม จำนวน 10 ครั้ง สรุปได้ว่า แผนที่ปาร์กัวร์ 2D ที่พัฒนาขึ้นนี้มีประสิทธิภาพในการใช้งานอยู่ในระดับสูง และเหมาะสมสำหรับการนำไปใช้เพื่อความบันเทิง

คำสำคัญ: C# Programming, 2D/3D Game Development

Project Title	Parkour Monster Map		
Team Members	Mr. Kitchaya	Taweetum	ID 65201280001
	Miss. Trantong	Chinudompong	ID 65201280011
Advisor	Mr. Satrawut	Srichat	
Co-advisor	Mr. Athit	Chidchob	
Department	Computer Technician		
Academic Year	2024		

Abstract

Currently, platformer games with parkour-style gameplay have gained immense popularity in the gaming industry. These games challenge players to control characters as they navigate obstacles efficiently. However, developing such games remains complex, particularly in creating game scenes and realistic movement mechanics that are both engaging and enjoyable. Moreover, players often want to adjust the frame rate and image quality to suit their hardware. Therefore, the team has developed a 2D Parkour Map using Unity Hub to provide players with a challenging and entertaining gaming experience. Additionally, options such as adjusting the resolution and game speed are available to optimize the experience for different hardware specifications, enhancing the overall gameplay.

Thus, the development team has designed the 2D parkour map to offer players a challenging gameplay experience, with flexible system customization to meet the needs of players. Specifically, the game allows for easy modification of graphical quality and game speed to suit different hardware capabilities.

The evaluation results of the 2D Parkour map, based on a user survey, revealed an average score ($\bar{x} = 4.21$, S.D. = 0.86), indicating a high level of satisfaction. The highest rated aspect was the aesthetics, modernity, and attractiveness have Average ($\bar{x} = 4.28$, S.D. = 0.93), followed by the efficiency of the game's buttons have Average ($\bar{x} = 4.29$, S.D. = 0.82). Additionally, a system functionality test, involving gameplay and response logging over 10 trials, concluded that the developed 2D Parkour map demonstrates a high level of performance and is well-suited for entertainment.

Keywords: C# Programming, 2D/3D Game Development

กิตติกรรมประกาศ

คณะผู้จัดทำโครงการ Parkour Monster Map ขอขอบพระคุณอาจารย์ที่ปรึกษาโครงการทั้งสองท่าน ได้แก่ อาจารย์ศัตราวุธ ศรีชาติ และอาจารย์อาทิตย์ ชิดชอบ อาจารย์ปิยพงษ์ เกตุภูงา อาจารย์ประภาพร บันเทา อาจารย์ฉัตรพฤษ์ สีสิม ซึ่งอาจารย์ทั้งห้าท่านที่กล่าวถึงได้ให้การสนับสนุนในการทำโครงการ และคำปรึกษาที่มีประโยชน์ ตั้งแต่ต้นจนโครงการประสบผลสำเร็จ รวมไปถึงการให้คำแนะนำในการปรับปรุงแก้ไขรูปแบบโครงการซึ่งทำให้โครงการมีการพัฒนา และสมบูรณ์มากยิ่งขึ้น

นอกจากนี้ ขอขอบคุณเพื่อนสมาชิกในกลุ่มโครงการ และเพื่อนกลุ่มอื่น ๆ ที่ร่วมแรงร่วมใจ และได้ช่วยเหลือเกื้อกูลกันในการทำงานหลายอย่างจนสำเร็จลุล่วง โดยการดำเนินงานแม้จะมีปัญหา แต่สมาชิกในกลุ่มก็ได้ช่วยกันแก้ไขปัญหาจนกระทั่งสำเร็จลุล่วงตามเป้าหมายที่วางไว้เป็นอย่างดี

กฤษชญา ทวีธรรม
ธารทอง ชินอุดมพงศ์

สารบัญ

	หน้า
บทคัดย่อ	ก
กิตติกรรมประกาศ	ข
สารบัญ	ค
สารบัญภาพ	จ
สารบัญตาราง	ฉ
บทที่ 1 บทนำ	1
1.1 ความเป็นมาของโครงการ	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ขอบเขตของโครงการ	1
1.4 ประโยชน์ที่คาดว่าจะได้รับ	2
1.5 การดำเนินงานโครงการ	2
1.6 นิยามศัพท์	2
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง	3
2.1 ความรู้เบื้องต้นเกี่ยวกับ Unity hub	3
2.2 ความรู้เบื้องต้นเกี่ยวกับ ภาษา C#	4
2.3 ความรู้เบื้องต้นเกี่ยวกับ Visual Studio Code	4
2.4 ความรู้เบื้องต้นเกี่ยวกับ การใช้โปรแกรม Unity hub	5
2.5 งานวิจัยที่เกี่ยวข้อง	5
บทที่ 3 วิธีการดำเนินงาน	6
3.1 ศึกษาความรู้เกี่ยวกับหัวข้อหรือเนื้อหาที่จัดทำโครงการ	6
3.2 ออกแบบโครงสร้าง	7
3.3 การเขียนโปรแกรม C#	9
3.4 สถิติที่ใช้ในการวิจัยโครงการ	18
3.5 สรุปผลวิธีการดำเนินงาน	19

สารบัญ (ต่อ)

	หน้า
บทที่ 4 ผลการดำเนินงาน	20
4.1 ผลการสร้าง Parkour Monster Map.....	20
4.2 ผลการวิเคราะห์แบบประเมิน Parkour Monster Map	23
4.3 ผลการทดลองการใช้งานและบันทึกผล Parkour Monster Map.....	24
4.4 สรุปผลการดำเนินงาน	25
บทที่ 5 สรุปผลการดำเนินงาน	26
5.1 ผลการดำเนินโครงการ.....	26
5.2 อุปสรรคและแนวทางในการแก้ปัญหา	26
5.3 ข้อเสนอแนะ.....	26
บรรณานุกรม.....	27
ภาคผนวก ก การดำเนินงานจัดทำโครงงาน Parkour Monster Map.....	28
ภาคผนวก ข คู่มือการใช้งาน Parkour Monster Map.....	33
ภาคผนวก ค แบบประเมินความพึงพอใจ.....	44
ประวัติผู้จัดทำโครงงาน	50

สารบัญรูป

	หน้า
รูปที่ 2.1 Unity Hub	3
รูปที่ 2.2 C#.....	4
รูปที่ 2.3 Visual Studio.....	5
รูปที่ 2.4 Y8	5
รูปที่ 3.1 Block Diagram Parkour Monster Map.....	7
รูปที่ 3.2 หน้าต่างภายใน Parkour Monster Map.....	8
รูปที่ 3.3 หน้าต่างของ Parkour Monster Map.....	9
รูปที่ 3.4 เข้าเกม	10
รูปที่ 4.1 หน้าต่างหลักของ Parkour Monster Map	20
รูปที่ 4.2 โครงสร้างหน้าต่างเกี่ยวกับของ Parkour Monster Map	21
รูปที่ 4.3 หน้าการเล่นเกม Parkour Monster Map.....	22
รูปที่ 4.4 ตอนจบ Parkour Monster Map	22

สารบัญตาราง

	หน้า
ตารางที่ 3.1 จากรูปที่ 3.3 ประกอบไปด้วยหลายอุปกรณ์	9
ตารางที่ 4.1 ข้อมูลทั่วไปของผู้ตอบแบบสอบถาม	22
ตารางที่ 4.2 ความพึงพอใจ และประสิทธิภาพของ Parkour Monster Map	22
ตารางที่ 4.3 ผลการทดลองของการเล่นเกม Parkour Monster Map	23
ตารางที่ 4.4 ผลการทดลองของการเล่นเกม Parkour Monster Map ที่ 4.3	24

บทที่ 1

บทนำ

1.1 ความเป็นมาของโครงการ

ในปัจจุบันเห็นว่าเกมแนว Parkour ยังคงเป็นที่นิยมอย่างต่อเนื่อง โดยมีการพัฒนา และปรับปรุงกราฟิก ระบบการเล่น และเนื้อเรื่องให้มีความน่าสนใจมากขึ้น ผู้เล่นจะได้สัมผัสประสบการณ์การวิ่งกระโดดปีนป่ายในสภาพแวดล้อมที่หลากหลาย ตั้งแต่เมืองใหญ่ที่เต็มไปด้วยตึกสูง ไปจนถึงป่าเขาที่ท้าทาย

เกมจะมีการออกแบบให้สามารถเล่นได้แค่ผู้เล่นคนเดียว (Single Player) เกมมีด่าน (Level) และความท้าทายต่าง ๆ ตามความสามารถของผู้เล่น โดยแบ่งออกเป็น 3 ด่าน โดยเพิ่มความท้าทายด้วยการจับเวลา ในส่วนของระบบควบคุม (Control) เกมด้วยคีย์บอร์ด และมีการออกแบบกราฟิกที่สวยงาม และความสนุกสนานในการเล่น

ดังนั้นทางคณะผู้จัดทำ จึงได้สร้าง Parkour Monster Map ขึ้นเพื่อสนับสนุนในการฝึกสมาธิ เพื่อสามารถจดจ่อต่อสิ่งนั้นได้โดยไม่สนใจสิ่งอื่น จุดประกายความคิดสร้างสรรค์ สามารถนำไปประยุกต์แก้ไขใช้สอยกับงานอื่น ๆ ได้

1.2 วัตถุประสงค์ของโครงการ

1.2.1 เพื่อสร้างและศึกษา Parkour Monster Map

1.2.2 เพื่อประเมินความพึงพอใจ และประสิทธิภาพของ Parkour Monster Map

1.3 ขอบเขตของโครงการ

1.3.1 เกมมีด่าน (Level) และความท้าทายต่าง ๆ ตามความสามารถ โดยมีการแบ่งด่านออกเป็น 3 ด่านหลัก ซึ่งผู้เล่นต้องทำภารกิจในแต่ละด่านให้สำเร็จเพื่อไปยังด่านต่อไป

1.3.2 เกมจะมีการจับเวลา (Timer) เพื่อให้สามารถแสดงผลการจับเวลาของผู้เล่นได้

1.3.3 สามารถใช้คีย์บอร์ดเพื่อควบคุม (Control) เพื่อควบคุมตัวละครในเกม

1.3.4 มีการออกแบบกราฟิก และออกแบบเกมให้มีความสนุกสนานในการเล่น

1.4 ประโยชน์ที่คาดว่าจะได้รับ

- 1.4.1 ได้ความรู้เกี่ยวกับการใช้ Unity hub
- 1.4.2 ได้ความรู้จากการเขียนภาษา C#
- 1.4.3 ได้ความรู้และทดลองทำเกี่ยวกับ C#
- 1.4.4 ได้นำเทคโนโลยีสมัยใหม่มาใช้ให้เกิดประโยชน์

1.5 การดำเนินงานโครงการ

- 1.5.1 สอบถามและรวบรวมข้อมูลในการจัดทำโครงการ
- 1.5.2 ศึกษาข้อมูลที่เกี่ยวข้องกับโครงการ
- 1.5.3 เสนอโครงการเพื่อขออนุมัติ
- 1.5.4 ทดสอบเพื่อหาประสิทธิภาพการทำงาน
- 1.5.5 วิเคราะห์และสรุปผลการทำงาน

1.6 นิยามศัพท์

1.6.1 C# คือ ภาษาที่ใช้กับคอมพิวเตอร์อย่างหนึ่งเพื่อให้ผู้ใช้สามารถนำไปสร้างเกม เว็บไซต์ หรือสิ่งอื่น ๆ ได้ บนคอมพิวเตอร์ เป็นหนึ่งในหลาย ๆ ภาษาของคอมพิวเตอร์

1.6.2 Programming คือ การเขียนโปรแกรมคอมพิวเตอร์ หรือเรียกให้สั้นลงว่า การเขียนโปรแกรม หรือ การเขียนโค้ด เป็นขั้นตอนการเขียน ทดสอบ และดูแลซอร์สโค้ดของโปรแกรมคอมพิวเตอร์

1.6.3 Parkour คือ พยายามที่จะไปจากจุดหนึ่งไปยังอีกจุดหนึ่งด้วยวิธีที่เร็วที่สุด

1.6.4 Map คือ แผนที่ในพื้นที่ใดแผนที่หนึ่งที่ถูกระบุไว้บนแผนที่

บทที่ 2

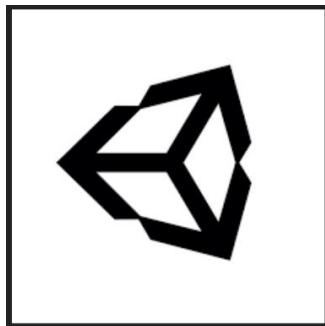
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงหลักการ และทฤษฎีพื้นฐานของ Parkour Monster Map โดยภายในเนื้อหาจะประกอบไปด้วยความรู้เบื้องต้นเกี่ยวกับ Parkour Monster Map , C# การเขียนโปรแกรม และงานวิจัยที่เกี่ยวข้อง ดังรายละเอียดต่อไปนี้

- 2.1 ความรู้เบื้องต้นเกี่ยวกับ Unity hub
- 2.2 ความรู้เบื้องต้นเกี่ยวกับ ภาษา C#
- 2.3 ความรู้เบื้องต้นเกี่ยวกับ Visual Studio
- 2.4 ความรู้เบื้องต้นเกี่ยวกับ การใช้โปรแกรม Unity hub
- 2.5 งานวิจัยที่เกี่ยวข้อง

2.1 ความรู้เบื้องต้นเกี่ยวกับ Unity hub

Unity Hub เป็นโปรแกรมที่ช่วยจัดการโปรเจกต์ และติดตั้ง Unity Editor หลายเวอร์ชันได้อย่างสะดวก ผู้ใช้สามารถเปิดโปรเจกต์เก่า สร้างโปรเจกต์ใหม่ และจัดการกับเวอร์ชันต่าง ๆ ของ Unity ได้ง่ายขึ้น นอกจากนี้ ยังสามารถล็อกอินเพื่อเชื่อมต่อกับบริการของ Unity เช่น Asset Store และ Cloud Unity Hub ยังช่วยตรวจสอบ และจัดการลิขสิทธิ์การใช้งาน Unity ได้โดยตรง เพื่อให้การพัฒนาเกมหรือแอปพลิเคชันราบรื่นมากขึ้น เป็นต้น [1]



รูปที่ 2.1 Unity Hub

2.2 ความรู้เบื้องต้นเกี่ยวกับ ภาษา C#

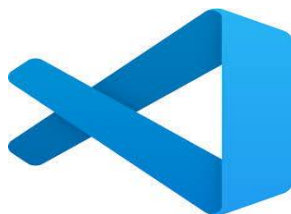
ภาษา C# (C-Sharp) เป็นภาษาที่พัฒนาโดย Microsoft เน้นการเขียนโปรแกรมแบบเชิงวัตถุ (Object-Oriented Programming - OOP) ซึ่งช่วยให้โปรแกรมเมอร์สามารถจัดโครงสร้างโค้ดให้เป็นระบบระเบียบ และง่ายต่อการจัดการ C# ทำงานร่วมกับ .NET Framework ซึ่งเป็นแพลตฟอร์มที่มีไลบรารี และเครื่องมือที่ช่วยเร่งการพัฒนาได้อย่างมีประสิทธิภาพ ทำให้ลดข้อผิดพลาดในการใช้ประเภทข้อมูลที่ไม่ถูกต้อง นอกจากนี้ยังเป็นภาษาหลักที่ใช้ใน Unity ซึ่งเป็นเอนจินเกมยอดนิยม จึงเหมาะกับการพัฒนาเกม การใช้ C# มาพร้อมกับเครื่องมืออย่าง Visual Studio ที่ช่วยให้การเขียนและแก้ไขโค้ดสะดวกมากขึ้น [2]



รูปที่ 2.2 C#

2.3 ความรู้เบื้องต้นเกี่ยวกับ Visual Studio

Visual Studio (Kachapot, 2019) Visual Studio คือ โปรแกรมตัวหนึ่งที่เป็นเครื่องมือที่ช่วยพัฒนาซอฟต์แวร์และระบบต่าง ๆ ซึ่งสามารถติดต่อสื่อสารพูดคุยกับคอมพิวเตอร์ได้ในระดับหนึ่งแล้ว แต่ยังไม่สามารถพัฒนาเป็นระบบเองได้ เหมาะสมสำหรับภาษา VB และ VB.NET เนื่องจากไมโครซอฟต์ได้พัฒนาโปรแกรมและภาษาขึ้นมาควบคู่กันเพื่อให้ใช้งานได้ซึ่งกัน และกัน ซึ่งจะมีพวกนักโปรแกรมเมอ์นำเครื่องมือมาใช้ในการพัฒนาต่อยอดให้เกิดเป็นระบบต่าง ๆ หรือเป็นเว็บไซต์ และแอปพลิเคชันต่าง ๆ Visual Studio ทำอะไรได้บ้าง เป็นเครื่องมือที่ใช้สร้างโปรแกรม ต่าง ๆ เช่น โปรแกรมที่รันบน ระบบปฏิบัติการ Windows เช่น โปรแกรมคำนวณเลข - โปรแกรมฐานข้อมูล เช่น Microsoft access, Microsoft SQL server - คอมโพเนนท์ทางด้าน Active X - โปรแกรมที่รันบนอินเทอร์เน็ต เป็นต้น



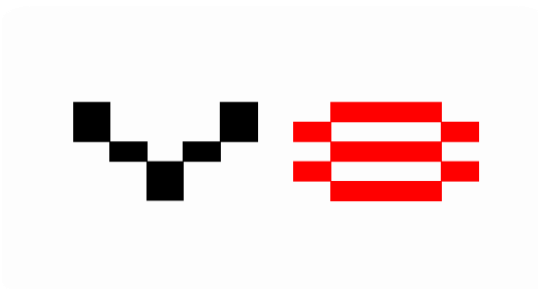
รูปที่ 2.3 Visual Studio

2.4 ความรู้เบื้องต้นเกี่ยวกับ การใช้โปรแกรม Unity hub

Unity เป็นเครื่องมือพัฒนาเกมที่มีอินเตอร์เฟซหลักประกอบด้วย Scene View สำหรับออกแบบฉาก, Game View เพื่อแสดงผลเกม, Hierarchy ที่แสดงรายการวัตถุทั้งหมดในฉาก และเลือก "3D Object" จากนั้นควบคุมวัตถุด้วยเครื่องมือในการย้าย, หมุน, และปรับขนาด การเขียนสคริปต์ C# สามารถทำได้โดยสร้างสคริปต์ใหม่ใน Project View และเขียนโค้ดเพื่อควบคุมพฤติกรรมของวัตถุ นอกจากนี้คุณยังสามารถนำเข้า Asset เช่น ภาพ และเสียงลงใน Project View เพื่อใช้ในโปรเจกต์ เมื่อพร้อมทดสอบเกมให้คลิกปุ่ม "Play" ใน Game View และเมื่อเสร็จสิ้นโปรเจกต์สามารถสร้าง Build ได้ที่ "File" > "Build Settings" โดยเลือกแพลตฟอร์มที่ต้องการ การทดลองทำโปรเจกต์เล็ก ๆ จะช่วยให้เข้าใจการใช้งาน Unity ได้เร็วขึ้น [3]

2.5 งานวิจัยที่เกี่ยวข้อง

Y8 เป็นเว็บไซต์ที่ให้บริการเกมออนไลน์หลากหลายประเภท รวมถึงเกมที่เล่นได้ทันทีในเบราว์เซอร์ เช่น เกมยิงปืน, เกมผจญภัย, เกมกีฬา และเกมปริศนา นอกจากนี้ยังมีฟีเจอร์อื่น ๆ เป็นแหล่งบันเทิงออนไลน์ที่สนุก และเข้าถึงได้ง่าย เหมาะสำหรับผู้ที่ชอบเล่นเกม และต้องการความสนุกสนานในเวลาว่าง [4]



รูปที่ 2.4 Y8

บทที่ 3

วิธีการดำเนินงาน

จากการศึกษา และออกแบบ Parkour Monster Map ได้รวบรวมเนื้อหาข้อมูลต่าง ๆ ที่เกี่ยวข้อง จึงสามารถดำเนินการออกแบบเกม Parkour Monster Map และแผนภาพบริบทของข้อมูลซึ่งมีขั้นตอนการดำเนินงานดังต่อไปนี้

3.1 ศึกษาความรู้เกี่ยวกับหัวข้อหรือเนื้อหาที่จัดทำโครงการ

การดำเนินงานและการสร้าง Parkour Monster Map ได้ศึกษาหาข้อมูลดังนี้

3.1.1 ศึกษาการใช้งาน Unity Hub

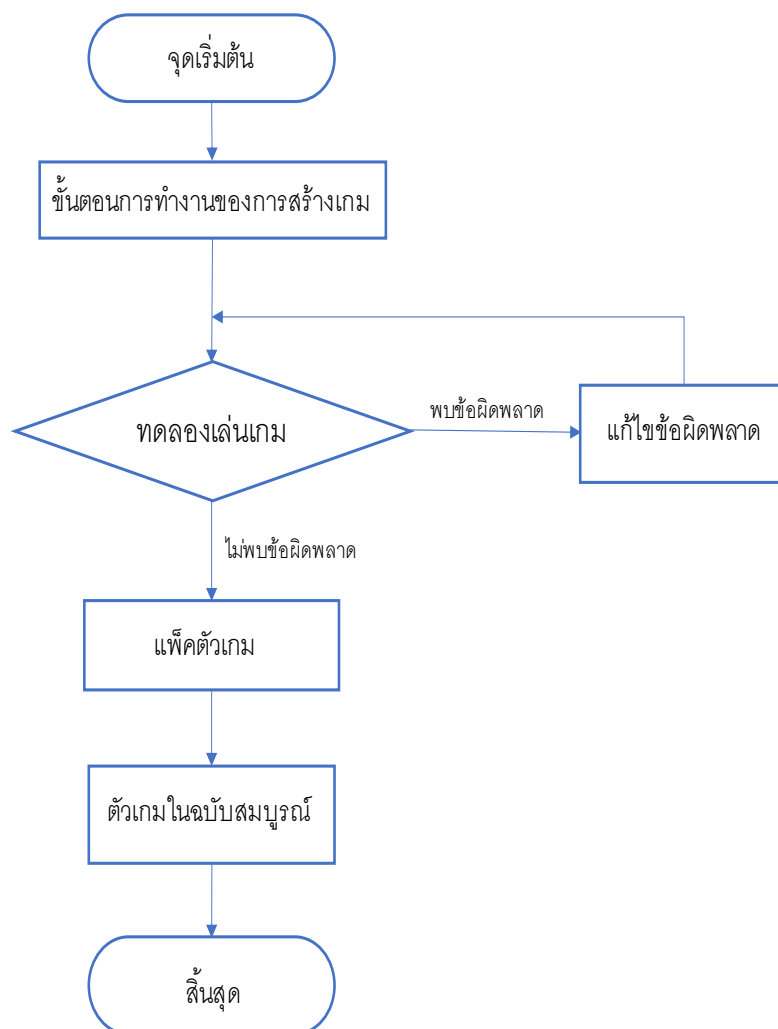
3.1.2 ศึกษาการเขียนภาษา C#

3.1.3 ศึกษาวิธีการงานใช้โปรแกรม Visual Studio Code

3.1.4 ศึกษาเกี่ยวกับ Game Design

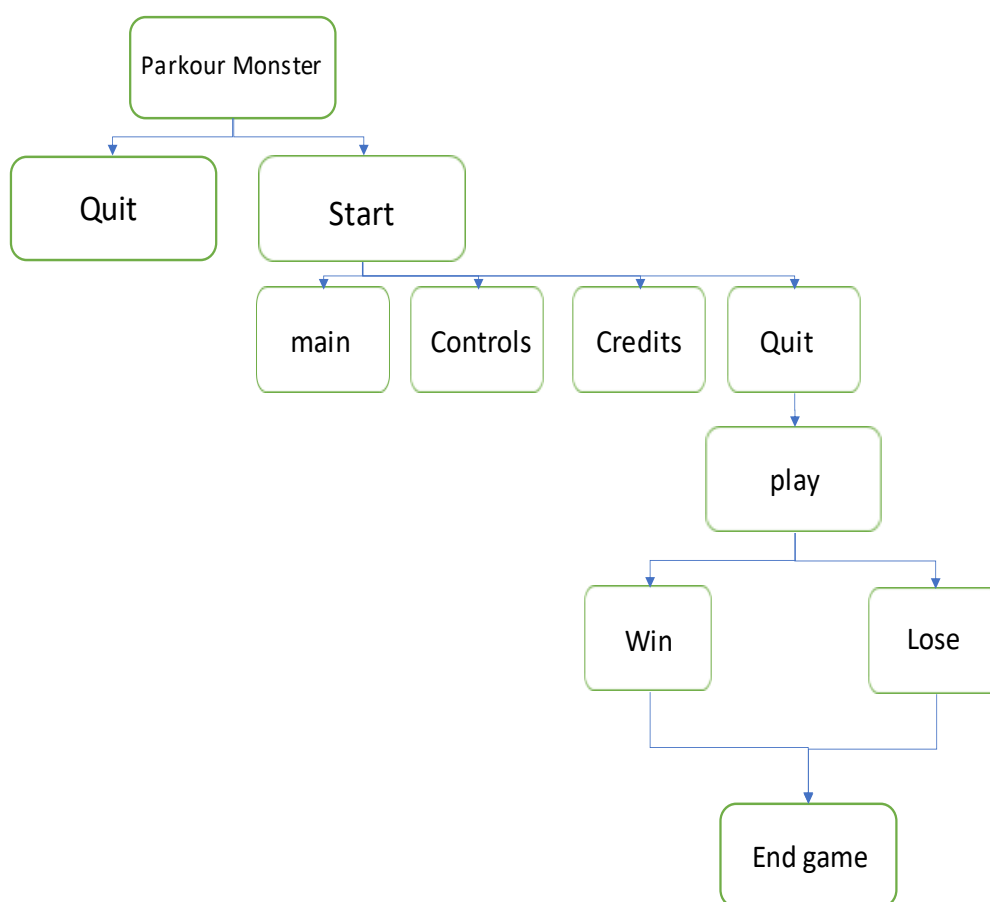
3.2 ออกแบบโครงสร้าง

3.2.1 ส่วนประกอบของ Parkour Monster Map ดังรูปที่ 3.1



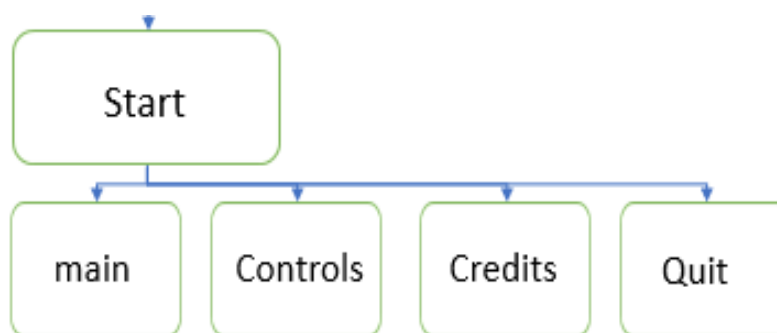
รูปที่ 3.1 Block Diagram Parkour Monster Map

3.2.2 โครงสร้างของ Parkour Monster Map



รูปที่ 3.2 หน้าต่างภายใน Parkour Monster Map

3.2.3 หน้าที่ใช้งานของแต่ละหน้าต่าง



รูปที่ 3.3 หน้าต่างของ Parkour Monster Map

ตารางที่ 3.1 จากรูปที่ 3.3 ประกอบไปด้วยหลายอุปกรณ์ ดังนี้

หมายเลข	ชื่อองค์ประกอบ	การใช้งาน
1	Start	กดเพื่อเริ่มเกม
2	main	บทกรินนำ
3	Controls	หน้าสอนการควบคุม
4	Credits	อ้างอิงถึงคนสร้างและพัฒนา
5	Quit	ออก

3.3 การเขียนโปรแกรม C#

การออกแบบเว็บไซต์ของ Parkour Monster Map และการกำหนดลักษณะของการทำงานที่เขียนโปรแกรมลงใน Visual Studio Code และสั่งให้มีหน้าต่าง นำทางไปยังหน้าต่าง ๆ ของ Parkour Monster Map เพื่อให้เข้าใจถึงกระบวนการทำงานทั้งหมดของ Parkour Monster Map สามารถอธิบายได้ดังนี้



รูปที่ 3.4 เข้าเกม

โปรแกรมที่ 3.1 โค้ด เข้าเกม

```

1.using UnityEngine;
2.using UnityEngine.SceneManagement;
3.public class MainMenu : MonoBehaviour
4.{
5.    public void StartGame()
6.    {
7.        SceneManager.LoadScene("PakourMap");
8.    }
9.}

```

จากโปรแกรมที่ 3.1 แสดงส่วน หน้าเข้าเกม สามารถอธิบายในรายละเอียดได้ดังนี้

บรรทัดที่ 1 บรรทัดนี้ใช้เพื่อนำ namespace Unity Engine เข้ามาใช้งาน ซึ่งจะเป็นการนำ namespace ที่มีคลาสและฟังก์ชันต่าง ๆ ที่จำเป็นสำหรับการพัฒนาเกมใน Unity

บรรทัดที่ 2 ใช้เพื่อนำ namespace UnityEngine.SceneManagement เข้ามาใช้งาน ซึ่งมีคลาสและฟังก์ชันที่เกี่ยวข้องกับการจัดการฉากในเกม

บรรทัดที่ 3 ประกาศคลาสสาธารณะชื่อ MainMenu คลาสนี้จะถูกแนบไปกับ GameObject ใน Unity เพื่อควบคุมพฤติกรรมของ Game Object นั้น MonoBehaviour: สืบทอดคลาส MonoBehaviour ซึ่งเป็นคลาสพื้นฐานสำหรับการสร้างสคริปต์ใน Unity

บรรทัดที่ 5 ประกาศเมธอดสาธารณะชื่อ StartGame() ซึ่งหมายความว่าเมธอดนี้เราสามารถ

เรียกใช้ได้จากสคริปต์อื่น ๆ

บรรทัดที่ 7 เป็นเมธอดที่ใช้สำหรับโหลดฉากที่มีชื่อว่า PakourMap เข้ามาแทนที่ฉากปัจจุบัน

โปรแกรมที่ 3.2 โค้ด ออกเกม

```

1.using System.Collections;
2.using System.Collections.Generic;
3.using UnityEngine;
4.public class Quit : MonoBehaviour
5.{
6.    public void QuitGame()
7.    {
8.        #if UNITY_EDITOR
9.            UnityEditor.EditorApplication.isPlaying = false;
10.        #else
11.            Application.Quit();
12.        #endif
13.    }
14.}

```

จากโปรแกรมที่ 3.2 แสดงโค้ด ออกเกม สามารถอธิบายในรายละเอียดได้ดังนี้

บรรทัดที่ 1 - 2 ใช้เพื่อนำ namespace ที่เกี่ยวข้องกับการจัดการข้อมูลแบบ collection เข้ามาใช้งาน ซึ่งโดยทั่วไปจะไม่ค่อยได้ใช้ในสคริปต์นี้

บรรทัดที่ 3 บรรทัดนี้ใช้เพื่อนำ namespace ที่อยู่ใน UnityEngine นำเข้ามาใช้งาน ซึ่งเป็น namespace ที่มีคลาสและฟังก์ชันต่าง ๆ ที่จำเป็นสำหรับการพัฒนาเกมใน Unity

บรรทัดที่ 4 ประกาศคลาสสาธารณะชื่อ Quit คลาสนี้จะถูกแนบไปกับ GameObject ใน คิว Unity เพื่อควบคุมพฤติกรรมของ GameObject นั้น MonoBehaviour: สืบทอดคลาส MonoBehaviour ซึ่งเป็นคลาสพื้นฐานสำหรับการสร้างสคริปต์ใน Unity

บรรทัดที่ 6 ประกาศเมธอดสาธารณะชื่อ QuitGame() ซึ่งจะหมายความว่าเมธอดนี้สามารถเรียกใช้ได้จากสคริปต์อื่น ๆ

บรรทัดที่ 9 - 11 เป็น preprocessor directive ที่ใช้ตรวจสอบว่าโค้ดกำลังรันอยู่ใน Unity Editor หรือไม่ หากอยู่ใน Editor จะตั้งค่า isPlaying เป็น false เพื่อหยุดการรันเกม หากไม่ได้อยู่ใน Editor จะเรียกใช้เมธอด Quit() เพื่อปิดเกม

โปรแกรมที่ 3.3 โค้ด Animation Player

```

1.using System.Collections;
2.using System.Collections.Generic;
3.using UnityEngine;
4.using Platformer.Gameplay;
5.using static Platformer.Core.Simulation;
6.using Platformer.Model;
7.using Platformer.Core;
8.namespace Platformer.Mechanics
9.{
10.    public class PlayerController : KinematicObject
11.    {
12.        public AudioClip jumpAudio;
13.        public AudioClip respawnAudio;
14.        public float maxSpeed = 5;
15.        public float jumpTakeOffSpeed = 7;
16.        public JumpState jumpState = JumpState.Grounded;
17.        private bool stopJump;
18.        /*internal new*/ public Collider2D collider2d;
19.        /*internal new*/ public AudioSource audioSource;
20.        public Health health;
21.        public bool controlEnabled = true;
22.        bool jump;
23.        Vector2 move;
24.        SpriteRenderer spriteRenderer;
25.        internal Animator animator;

```

โปรแกรมที่ 3.3 (ต่อ) โค้ด Animation Player

```

26.         readonly PlatformerModel model =
Simulation.GetModel<PlatformerModel>();
27.         public Bounds Bounds => collider2d.bounds;
28.         void Awake() {
29.             health = GetComponent<Health>();
30.             audioSource = GetComponent<AudioSource>();
31.             collider2d = GetComponent<Collider2D>();
32.             spriteRenderer = GetComponent<SpriteRenderer>();
33.             animator = GetComponent<Animator>();
34.         }
35.         protected override void Update() {
36.             if (controlEnabled) { // รับการป้อนข้อมูลแนวนอน (ซ้าย-ขวา)
37.                 move.x = Input.GetAxis("Horizontal"); // ตรวจสอบว่าผู้เล่น
กระโดด
38.                 if (jumpState == JumpState.Grounded &&
Input.GetButtonDown("Jump"))
39.                     jumpState = JumpState.PrepareToJump;
40.                 else if (Input.GetButtonUp("Jump")) {
41.                     stopJump = true;
42.                     Schedule<PlayerStopJump>().player = this;
43.                 }
44.             }
45.             else {
46.                 move.x = 0;
47.             }
48.             UpdateJumpState();
49.             base.Update();

```

โปรแกรมที่ 3.3 (ต่อ) โค้ด Animation Player

```

50.     }
51.     void UpdateJumpState() {
52.         jump = false;
53.         switch (jumpState) {
54.             case JumpState.PrepareToJump:
55.                 jumpState = JumpState.Jumping;
56.                 jump = true;
57.                 stopJump = false;
58.                 break;
59.             case JumpState.Jumping:
60.                 stopJump = false;
61.                 break;
62.             case JumpState.Jumping:
63.                 if (!IsGrounded) {
64.                     Schedule<PlayerJumped>().player = this;
65.                     jumpState = JumpState.InFlight;
66.                 }
67.                 break;
68.             case JumpState.InFlight:
69.                 if (IsGrounded) {
70.                     Schedule<PlayerLanded>().player = this;
71.                     jumpState = JumpState.Landed;
72.                 }
73.                 break;
74.             case JumpState.Landed:
75.                 jumpState = JumpState.Grounded;

```

โปรแกรมที่ 3.3 (ต่อ) โค้ด Animation Player

```

76.                break;
77.            }
78.        }
79.        protected override void ComputeVelocity() {
80.            if (jump && IsGrounded) {
81.                velocity.y = jumpTakeOffSpeed * model.jumpModifier;
82.                jump = false;
83.            }
84.            else if (stopJump) {
85.                stopJump = false;
86.                if (velocity.y > 0) {
87.                    velocity.y = velocity.y * model.jumpDeceleration;
88.                }
89.            } // ขยับไปทางซ้ายหรือขวา
90.            if (move.x > 0.01f)
91.                spriteRenderer.flipX = false; // หันหน้าไปทางขวา
92.            else if (move.x < -0.01f)
93.                spriteRenderer.flipX = true; // หันหน้าไปทางซ้าย
94.            animator.SetBool("grounded", IsGrounded);
95.            animator.SetFloat("velocityX", Mathf.Abs(velocity.x) / maxSpeed); //
กำหนดความเร็ว
96.            targetVelocity = move * maxSpeed;
97.        }

```

โปรแกรมที่ 3.3 (ต่อ) โค้ด Animation Player

```

98.    public enum JumpState {
99.        Grounded,
100.        PrepareToJump,
101.        Jumping,
102.        InFlight,
103.        Landed
104.    }
105. }
106.}

```

จากโปรแกรมที่ 3.3 แสดงโค้ด Animation Player สามารถอธิบายในรายละเอียดได้ดังนี้
บรรทัดที่ 1 - 7 นำเข้าคลาสพื้นฐานของ Unity และการจัดการข้อมูล และ นำเข้าคลาสที่เกี่ยวข้องกับระบบเกมแพลตฟอร์มของคุณ

บรรทัดที่ 8 – 11 Player Controller เป็นคลาสที่จัดการการควบคุมตัวละคร และก็มีอีกคือ KinematicObject เป็นคลาสฐานที่อาจมีฟังก์ชันการทำงานพื้นฐานสำหรับการควบคุมการเคลื่อนไหว

บรรทัดที่ 12 – 13 คลิปเสียงที่ใช้ในเกม

บรรทัดที่ 14 – 15 ความเร็วสูงสุดของตัวละคร และความเร็วเริ่มต้นเมื่อกระโดด

บรรทัดที่ 16 สถานะของการกระโดด (Grounded , PrepareToJump , Jumping , InFlight , Landed)

บรรทัดที่ 17 ตัวแปรที่ใช้ในการหยุดการกระโดด

บรรทัดที่ 18 Collider สำหรับตรวจจับการชน

บรรทัดที่ 19 ตัวจัดการเสียง

บรรทัดที่ 20 ตัวแปรที่จัดการสุขภาพของตัวละคร

บรรทัดที่ 21 การเปิด / ปิดการควบคุมของตัวละคร

บรรทัดที่ 28 - 34 ฟังก์ชันนี้ถูกเรียกใช้เมื่อตัวละครถูกสร้างขึ้น และใช้เพื่อรับการอ้างอิงไปยังคอมโพเนนต์อื่น ๆ ของ Game Object

บรรทัดที่ 35 - 50 ฟังก์ชันนี้อัปเดตการควบคุมของตัวละครในทุก ๆ เฟรม มีตรวจสอบการป้อนข้อมูลแนวนอน (move.x) และสถานะการกระโดด (jumpState) เรียกใช้ Update Jump State() เพื่ออัปเดตสถานะการกระโดด

บรรทัดที่ 51 - 78 ฟังก์ชันนี้อัปเดตสถานะการกระโดดตามสถานการณ์ปัจจุบัน แล้วใช้ switch เพื่อตรวจสอบและเปลี่ยนแปลงสถานะการกระโดด

บรรทัดที่ 79 - 97 ฟังก์ชันนี้คำนวณความเร็วของตัวละคร แล้วก็ตั้งค่าความเร็วในการกระโดดและปรับความเร็วตามการเคลื่อนที่แนวนอน อัปเดตการแสดงผลของตัวละคร เช่น การหมุนภาพและแอนิเมชัน เป็นต้น

บรรทัดที่ 98 - 106 Jump State เป็น enumeration ที่กำหนดสถานะต่าง ๆ ของการกระโดด เช่น Grounded, PrepareToJump, Jumping, InFlight, และ Landed เป็นต้น

3.4 เพื่อประเมินความพึงพอใจและประสิทธิภาพของ Parkour Monster Map

3.4.1 เครื่องมือที่ใช้การประเมินความพึงพอใจของ Parkour Monster Map จะใช้แบบสอบถาม ซึ่งแบบถามแบ่งเป็น 2 ส่วน ดังนี้

ส่วนที่ 1 แบบสอบถามในเรื่องปัจจัยส่วนบุคคลของผู้ตอบแบบสอบถามซึ่งประกอบไปด้วยคำถามเกี่ยวกับ

1. เพศ
2. ช่วงอายุ

ส่วนที่ 2 แบบสอบถามเกี่ยวกับความพึงพอใจของ Parkour Monster Map แบ่งประเด็นย่อยออกเป็น 2 ประเด็น ดังนี้

3.4.1 ค่าเฉลี่ยเลขคณิต หมายถึง การหาผลรวมของข้อมูลทั้งหมดด้วยจำนวนข้อมูลทั้งหมด
 ดังสมการ

ประเด็นที่ 1 ด้านความสวยงาม ความทันสมัย น่าสนใจ

ประเด็นที่ 2 ด้านปุ่มต่างๆในเกมสามารถทำงานได้อย่างมีประสิทธิภาพ

โดยใช้มาตรวัดระดับความพึงพอใจ 5 ระดับ คือ ดีมาก ดี ปานกลาง พอใช้ และควรปรับปรุง โดยทุกข้อเป็นข้อความที่มีลักษณะเป็นบวกตามลำดับ โดย

- ระดับ 5 หมายถึง ดีมาก
- ระดับ 4 หมายถึง ดี
- ระดับ 3 หมายถึง ปานกลาง

ระดับ 2 หมายถึง พอใช้

ระดับ 1 หมายถึง ควรปรับปรุง

3.5 สรุปผลวิธีการดำเนินงาน

3.5.1 การเก็บรวบรวมข้อมูลสร้างแบบสอบถามโดยแบบสอบถามออนไลน์สำเร็จรูปโดยใช้ Google Form จากนั้นติดต่อประสานงานกับกลุ่มตัวอย่างจำนวน 10 คน เพื่อตอบแบบสอบถาม และแจ้งวัตถุประสงค์ของการทำ โครงการ โดยใช้เวลาในการเก็บรวบรวมข้อมูล เป็นเวลา 5 วัน ตั้งแต่วันที่ 30 สิงหาคม 2567 ถึงวันที่ 3 กันยายน 2567 ซึ่งสามารถรวบรวมได้ทั้งหมด 18 คน

3.5.2 การวิเคราะห์ข้อมูล

ส่วนที่ 1 ค่าเฉลี่ยเลขคณิต หมายถึง การหาผลรวมของข้อมูลทั้งหมดด้วยจำนวนข้อมูลทั้งหมด ดังสมการ

$$\bar{x} = \frac{\sum x_i}{n}$$

เมื่อ $\bar{x}$ คือ ค่าเฉลี่ยเลขคณิต
 $\sum x_i$ คือ ผลบวกของข้อมูลทุกค่า
 n คือ จำนวนข้อมูลทั้งหมด

ส่วนที่ 2 ค่าเบี่ยงเบนมาตรฐาน ส่วนเบี่ยงเบนมาตรฐานเป็นค่าวัดการกระจายที่สำคัญทางสถิติเพราะเป็นค่าที่ใช้บอกถึงการกระจายของข้อมูลได้ดีกว่าค่าพิสัย และค่าส่วนเบี่ยงเบนเฉลี่ย ดังสมการ

$$S.D. = \sqrt{\frac{(x_i - \bar{x})^2}{n - 1}}$$

S.D. คือ ส่วนเบี่ยงเบนมาตรฐาน
 x_i คือ ข้อมูล (ตัวที่ 1,2,3...,n)
 $\bar{x}$ คือ ค่าเฉลี่ยเลขคณิต

n คือ จำนวนของข้อมูลทั้งหมด

ส่วนที่ 3 การวิเคราะห์ความพึงพอใจของ Parkour Monster Map ใช้ในการแจกแจงความถี่ และหาค่าร้อยละแล้วนำเสนอในรูปแบบตาราง วิเคราะห์โดยหาค่าเฉลี่ย และส่วนเบี่ยงเบนมาตรฐาน โดยกำหนดค่าเฉลี่ยระดับคะแนนความพึงพอใจเป็น 5 ระดับ โดยคำนวณแล้วนำมาจัดช่วงคะแนนดังนี้

$$\begin{aligned}\text{ช่วงระดับคะแนน} &= \frac{\text{คะแนนสูงสุด} - \text{คะแนนต่ำสุด}}{\text{จำนวนระดับ}} \\ &= \frac{5-1}{5} = 0.8\end{aligned}$$

ระดับความคิดเห็นตามขนาดของช่องระดับคะแนนเป็น ดังนี้

ค่าเฉลี่ยอยู่ระหว่าง 4.21 – 5.00	มีความคิดเห็นอยู่ในระดับมากที่สุด
ค่าเฉลี่ยอยู่ระหว่าง 3.41 – 4.20	มีความคิดเห็นอยู่ในระดับมาก
ค่าเฉลี่ยอยู่ระหว่าง 2.61 – 3.40	มีความคิดเห็นอยู่ในระดับปานกลาง
ค่าเฉลี่ยอยู่ระหว่าง 1.81 – 2.60	มีความคิดเห็นอยู่ในระดับพอใช้
ค่าเฉลี่ยอยู่ระหว่าง 1.10 – 1.80	มีความคิดเห็นอยู่ในระดับควรปรับปรุง

บทที่ 4

ผลการดำเนินงาน

จากวิธีการดำเนินงาน การสร้าง Parkour Monster Map เมื่อดำเนินการเขียน และออกแบบ หน้าตาภายในเกมแล้ว สามารถสรุปผลการดำเนินงานได้ดังต่อไปนี้

4.1 ผลการสร้าง Parkour Monster Map

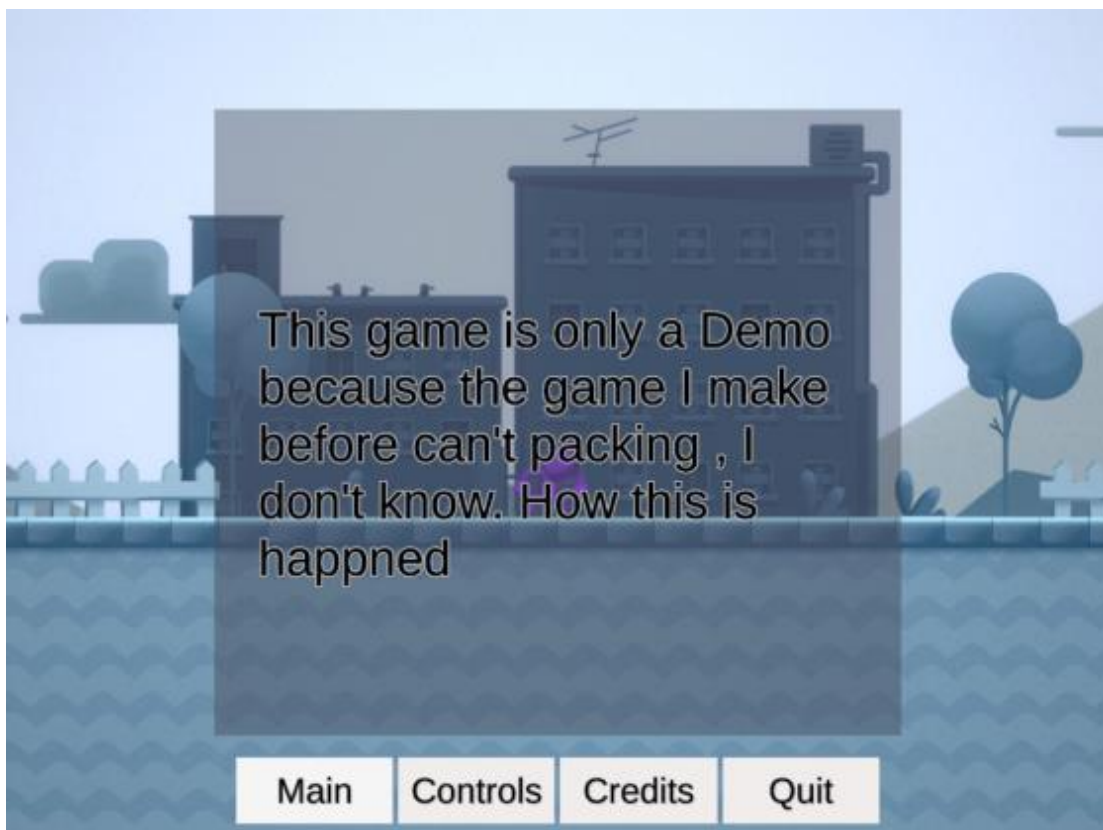
การสร้าง Parkour Monster Map เมื่อดำเนินการเสร็จแล้วได้ส่วนประกอบดังนี้

4.1.1. หน้าหลัก และปุ่มนำทางต่าง ๆ ดังรูปที่ 4.1



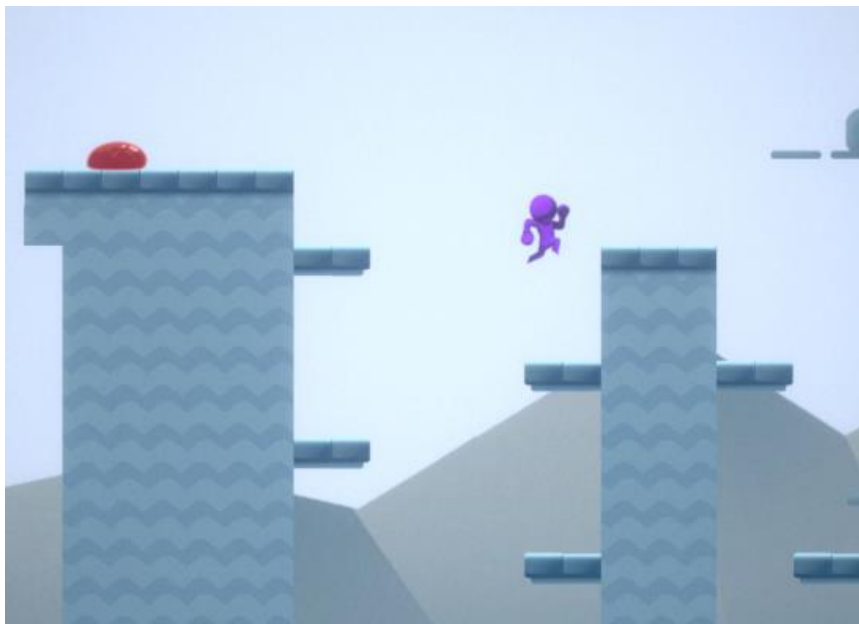
รูปที่ 4.1 หน้าต่างหน้าหลักของ Parkour Monster Map

4.1.2. หน้าเกี่ยวกับของ Parkour Monster Map



รูปที่ 4.2 โครงสร้างหน้าต่างเกี่ยวกับของ Parkour Monster Map

4.1.3. หน้าการเล่นเกม



รูปที่ 4.3 หน้าการเล่นเกม Parkour Monster Map

4.1.4. หน้าตอนจบของ Parkour Monster Map



รูปที่ 4.4 ตอนจบ Parkour Monster Map

4.2 ผลการวิเคราะห์แบบประเมิน Parkour Monster Map

4.2.1 ข้อมูลทั่วไปของผู้ตอบแบบสอบถามประเมิน Parkour Monster Map

ตารางที่ 4.1 ข้อมูลทั่วไปของผู้ตอบแบบสอบถาม

ข้อมูลทั่วไป	จำนวน(คน)
1.เพศ ชาย	9
หญิง	8
2. ช่วงอายุ ต่ำกว่า 15	16
15-30	1
มากกว่า 20	1

จากตารางที่ 4.1 จำนวนผู้ตอบแบบสอบถาม Parkour Monster Map จำแนกด้วย เพศคือ เพศชาย จำนวน 9 คน เพศหญิง จำนวน 8 คน ช่วงอายุต่ำกว่า 15 ปี จำนวน 16 คน 15-30 ปี จำนวน 1 คน มากกว่า 20 ปี จำนวน 1 คน

4.2.2 ผลการประเมินแบบสอบถาม Parkour Monster Map

ตารางที่ 4.2 ความพึงพอใจ และประสิทธิภาพของ Parkour Monster Map

รายการประเมินผล	จำนวน	$\bar{x}$	S.D.	ระดับความพึงพอใจ
ด้านการออกแบบ				
1. ความสวยงาม ความทันสมัย น่าสนใจ	18	4.28	0.93	ดีมาก
2. เกมความง่าย/ยากในการเล่น	17	4.24	0.73	ดีมาก
3. เกมมีความราบรื่นในการเล่น (ไม่กระตุก/ค้าง)	17	4.06	1.00	ดี
4. การจัดวางมีความเรียบร้อย	17	4.18	0.79	ดี
5. ปุ่มต่างๆในเกมสามารถทำงานได้อย่างมีประสิทธิภาพ	17	4.29	0.82	ดีมาก
รวม	86	4.21	0.86	ดีมาก

จากตารางที่ 4.2 พบว่าประชากรผู้ที่ตอบแบบสอบถามเกี่ยวกับโครงการ เมื่อพิจารณาผลการประเมินเฉลี่ยอยู่ในระดับมากที่สุด ($\bar{X} = 4.21$) , (S.D. = 0.86) จากผลการประเมินพบว่าผู้ตอบแบบสอบถามให้คะแนนมากที่สุด เป็นจำนวน 2 ข้อ คือ หัวข้อของความสวยงาม ความทันสมัย น่าสนใจ ($\bar{X} = 4.28$) , (S.D. = 0.93) และหัวข้อของปุ่มต่าง ๆ ในเกมสามารถทำงานได้อย่างมีประสิทธิภาพ ($\bar{X} = 4.29$) , (S.D. = 0.82) แผนที่ยาการ์ 2D ที่พัฒนาขึ้นนี้จะมีประสิทธิภาพในการใช้งานอยู่ในระดับสูง และเหมาะสมสำหรับการนำไปใช้เพื่อความบันเทิงอย่างกว้างขวาง

ตอนที่ 3 ข้อเสนอแนะ

1. ทำให้ล็อกอินกับแพลตฟอร์มอื่นได้อีก

4.3 ผลการทดลองการใช้งานและบันทึกผล Parkour Monster Map

4.3.1 ผลการทดลองการใช้งาน และบันทึกผลการเล่นเกมเป็นจำนวน 10 ครั้งซึ่งได้แบ่งตารางบันทึกผลเป็นตารางละ 2 ครั้ง ได้บันทึกผลการเป็นจำนวน ครั้ง ดังตารางที่ 4.3

ตารางที่ 4.3 ผลการทดลองของการเล่นเกม Parkour Monster Map

รอบการ ทดสอบ	การทำงานของเกมแต่ละด่าน (ครั้ง)				
	1-2 ครั้ง	3-4 ครั้ง	5-6 ครั้ง	7-8 ครั้ง	9-10 ครั้ง
1	ใช้งานได้	ใช้งานได้	ใช้งานได้	ใช้งานได้	ใช้งานได้
2	ใช้งานได้	ใช้งานได้	ใช้งานได้	ใช้งานได้	ใช้งานได้
3	ใช้งานได้	ใช้งานได้	ใช้งานได้	ใช้งานได้	ใช้งานได้
ร้อยละ	ร้อยละ 100	ร้อยละ 100	ร้อยละ 100	ร้อยละ 100	ร้อยละ 100

จากตารางที่ 4.3 แสดงให้เห็นการทดสอบการทำงานของเกมแต่ละด่านในจำนวน 10 ครั้ง โดยแต่ละด่านได้ทำการบันทึกผลจำนวนการทำงานของเกม ในการทดลองเล่นเกมที่ 1-2 ครั้งการควบคุมต่าง ๆ ในด้านทั้งหมดสามารถทำงานได้ ร้อยละ 100 ของการทำงาน การทดลองที่ 3-4 ครั้ง ทั้งหมดผ่าน ร้อยละ 100 การทดลองที่ 5-6 ครั้ง เข้าเล่นเกมได้ ร้อยละ 100 การทดลองที่ 7-8 ครั้ง ระบบต่าง ๆ หลังเข้าเล่นเกมเป็น ร้อยละ 100 และในการทดลองที่ 9-10 ครั้ง การทำงานของเกมแต่ละด่านผ่าน ร้อยละ 100

4.3.2 ผลการทดลองการใช้งาน และบันทึกผลการทดสอบการทำงานของเกมแต่ละด้าน ในจำนวน 10 ครั้งซึ่งได้แบ่งตารางบันทึกผลเป็นตารางละ 3 ครั้ง ได้บันทึกผลการเป็นจำนวนครั้ง ดังตารางที่ 4.4

ตารางที่ 4.4 ผลการทดลองของการเล่นเกม Parkour Monster Map ที่ 4.3

จำนวนครั้ง	ทำงาน	ไม่ทำงาน
1-2	√	
3-4	√	
5-6	√	
7-8	√	
9-10	√	
ร้อยละ	100%	

จากตารางที่ 4.4 แสดงให้เห็นว่าสามารถเข้าเล่นเกมได้ และสามารถทำงานได้อย่างปกติในจำนวน 10 ครั้งโดยแต่ละ ครั้งได้ทำการบันทึกผลจำนวนการทดสอบการทำงานของเกมแต่ละด้านเป็นครั้งในการทดลองการทำงานที่ 1-10 ครั้ง การเข้าเล่นเกมสามารถทำงานได้ ร้อยละ 100

4.4 สรุปผลการดำเนินงาน

ผู้จัดทำได้ทำการทดสอบ Parkour Monster Map โดยเขียนโปรแกรมเริ่มเกม เพื่อใช้ในการทดสอบการเริ่มเกมในแต่ละครั้งที่มีการ User เข้าเล่นเกมทำการบันทึกผลการเข้าเล่นจำนวน 10 ครั้ง

การประเมินแบบสอบถาม Parkour Monster Map ได้ค่าเฉลี่ย ($\bar{X}$ = 4.21) , (S.D. = 0.86) อยู่ในระดับมากที่สุด ซึ่งค่าเฉลี่ยที่มากที่สุด คือ หัวข้อของความสวยงาม ความทันสมัย น่าสนใจ ได้ค่าเฉลี่ย ($\bar{X}$ = 4.28) , (S.D. = 0.93) และหัวข้อของปุ่มต่าง ๆ ในเกมสามารถทำงานได้อย่างมีประสิทธิภาพ ($\bar{X}$ = 4.29) , (S.D.= 0.82) แผนที่ป้ากัวร์ 2D ที่พัฒนาขึ้นนี้มีประสิทธิภาพในการใช้งานอยู่ในระดับสูง และเหมาะสมสำหรับการนำไปใช้เพื่อความบันเทิงอย่างกว้างขวาง

ผลการทดลองการใช้งานและบันทึกผลการเข้าเล่นเกมเป็นจำนวน 10 ครั้งและ การทำงานของโปรแกรมจำนวน 10 ครั้งได้ผลการเข้าเกมผ่านทุกครั้งสามารถทำงานได้เป็น ร้อยละ 100 ของการทำงาน ผลการทดสอบเข้าเกม สามารถทำงานได้อย่างปกติ

บทที่ 5

สรุปผลการดำเนินงาน

5.1 ผลการดำเนินโครงการ

โครงการนี้นำเสนอการสร้าง Parkour Monster Map มีจุดมุ่งหมายเพื่อเพิ่มความสามารถทักษะในการอ่าน และเขียน โดยมีภาคการทำงาน หลัก ๆ ของเว็บไซต์ คือ ในภาคการทำงานนี้ มีส่วนประกอบของเว็บไซต์อ่าน และเขียน สร้างขึ้นจาก โปรแกรม Visual Studio Code เป็นโปรแกรมที่ใช้ในการเขียนเว็บไซต์ ทั้งหมดใน Parkour Monster Map

วิธีทดสอบ Parkour Monster Map การทดลอง Parkour Monster Map ได้ทำการเขียนเว็บไซต์ผ่านโปรแกรม Visual Studio Code เพื่อใช้ในการทดสอบการทำงานของล็อกอิน โดยทำการบันทึกผลการทำงานของล็อกอิน จำนวน 10 ครั้ง

ซึ่งจากผลการประเมิน Parkour Monster Map พบว่าการประเมินแบบสอบถามของ Parkour Monster Map ได้ค่าเฉลี่ย ($\bar{X}$ = 4.21) , (S.D. = 0.86) อยู่ในระดับมากที่สุด ซึ่งมีค่าเฉลี่ยที่มากที่สุด 2 หัวข้อ คือ หัวข้อความสวยงาม ความทันสมัย น่าสนใจ ($\bar{X}$ = 4.28) , (S.D. = 0.93) และ หัวข้อของปุ่มต่าง ๆ ในเกมสามารถทำงานได้อย่างมีประสิทธิภาพ ($\bar{X}$ = 4.29) , (S.D.= 0.82) แผนที่ ปากัวร์ 2D ที่พัฒนาขึ้นนี้มีประสิทธิภาพในการใช้งานอยู่ในระดับสูง และเหมาะสมสำหรับการนำไปใช้เพื่อความบันเทิงอย่างกว้างขวาง

5.2 อุปสรรคและแนวทางในการแก้ปัญหา

5.2.1 ยังไม่สามารถทำหน้าเข้าเกมได้ แก้ไขโดยการสร้างเพิ่มเข้าไป

5.2.2 หน้าปรับแต่งภายในเกมยังไม่มี แก้ไขโดยการสร้างเพิ่มเข้าไป

5.2.3 ยังไม่สามารถทำฉากจบที่ดีพอได้ แก้ไขโดยการปรับแต่ง และสร้างฉากจบที่ต้องการขึ้นมา

5.3 ข้อเสนอแนะ

5.3.1 เพิ่มแพลตฟอร์มลงเกมให้มากกว่านี้

5.3.2 เพิ่มมอนสเตอร์ให้มีความหลากหลาย

5.3.3 เพิ่มการเก็บคะแนนเมื่อเก็บคริสตัน

บรรณานุกรม

เว็บไซต์

- [1] unity (13 กันยายน 2567) การสร้างเกมด้วย unity [ระบบออนไลน์]
แหล่งที่มา https://unity3d-thailand.blogspot.com/p/blog-page_63.html
- [2] มาคัส (13 กันยายน 2567) ภาษา C# [ระบบออนไลน์]
แหล่งที่มา <https://marcuscode.com/lang/csharp>
- [3] มาคัส (13 กันยายน 2567) ติดตั้งและใช้งาน Visual Studio Code [ระบบออนไลน์]
แหล่งที่มา <https://marcuscode.com/2020/07/installing-visual-studio-code-on-windows>
- [4] Y8 (13 กันยายน 2567) เกม [ระบบออนไลน์]
แหล่งที่มา <https://th.y8.com/>

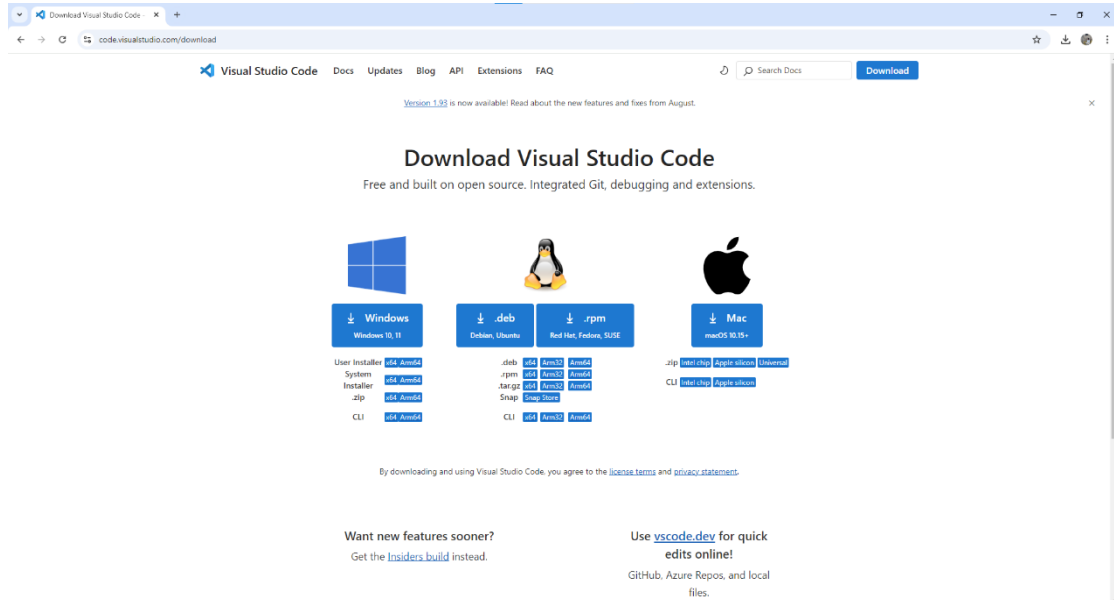
อ้างอิง

- 1. Unity3D (13 กันยายน 2567) การสร้างเกมจากโปรแกรม Unity3D [ระบบออนไลน์]
แหล่งที่มา <https://www.youtube.com/watch?v=pdIvBk0xbUU>
- 2. KongRuksiam (13 กันยายน 2567) เขียนโปรแกรมภาษา C# เริ่มต้น [ระบบออนไลน์]
แหล่งที่มา https://www.youtube.com/watch?v=4_c5EBr0whM
- 3. Brackeys (13 กันยายน 2567) How to Program in C# [ระบบออนไลน์]
แหล่งที่มา <https://www.youtube.com/watch?v=jGD0vn-QIkg>
- 4. Patiphan (13 กันยายน 2567) สอนใช้งาน VScode ให้เป็นเซียน [ระบบออนไลน์]
แหล่งที่มา <https://www.youtube.com/watch?v=RhS0tdQsu70>

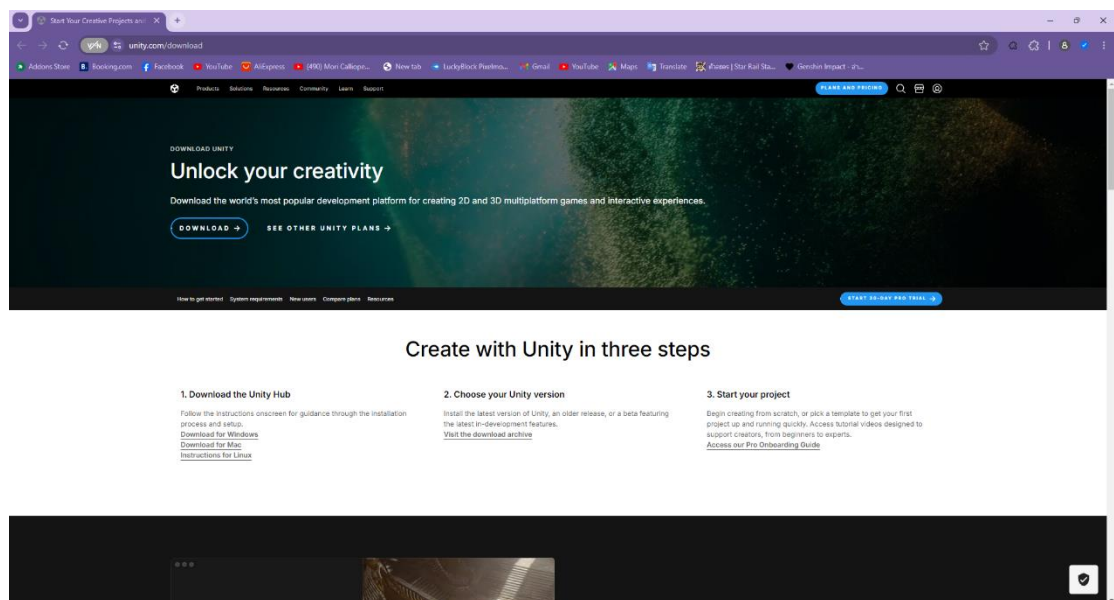
ภาคผนวก ก

การดำเนินงานจัดทำโครงการ Parkour Monster Map

ภาคผนวก ก การดำเนินงานจัดทำโครงงาน Parkour Monster Map



รูปที่ ก.1 ติดตั้ง Visual Studio Code



รูปที่ ก.2 ติดตั้ง Unity Hub

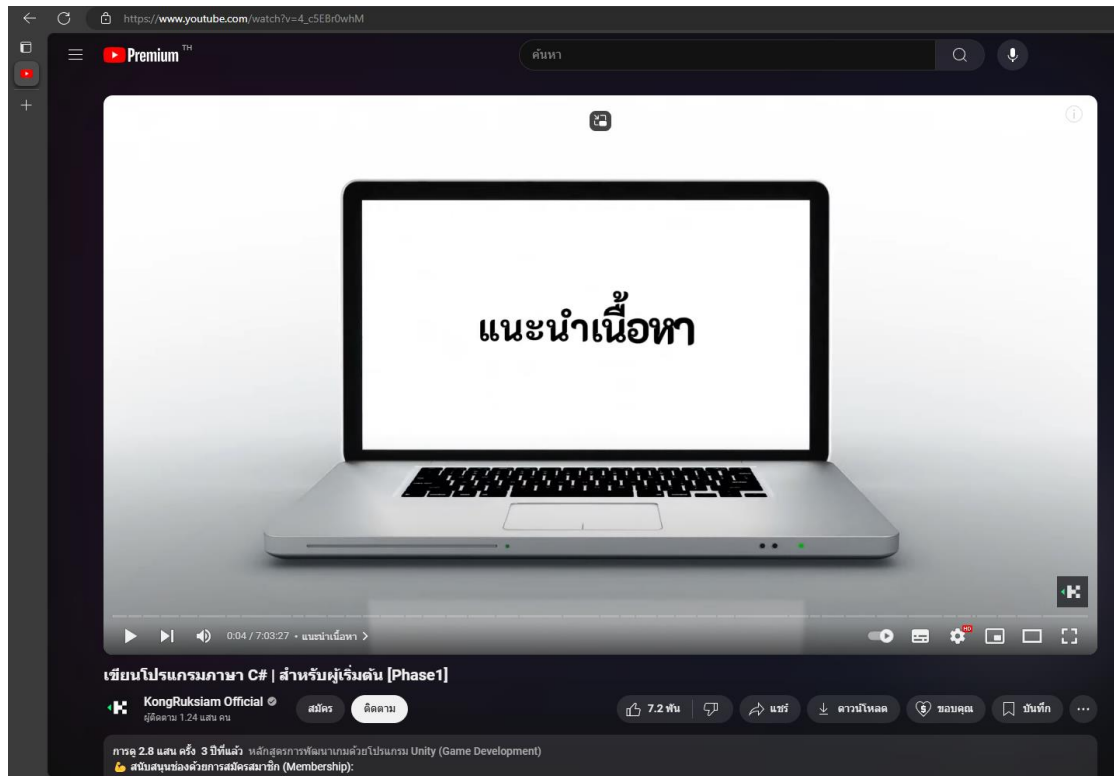
ทำการติดตั้งโปรแกรม Visual Studio Code เพื่อใช้ในการเขียน Parkour Monster Map



รูปที่ ก.3 อธิบายเนื้อหา



รูปที่ ก.4 การเขียนเนื้อหาภายในเกม



รูปที่ ก.7 ศึกษาวิธีการเขียนภาษา C#

ศึกษารูปเล่มเพื่อทำรายงานของ Parkour Monster Map เพื่อประกอบการนำเสนอบนคอมพิวเตอร์เพื่อนำไปสู่ตัวเกมจริง

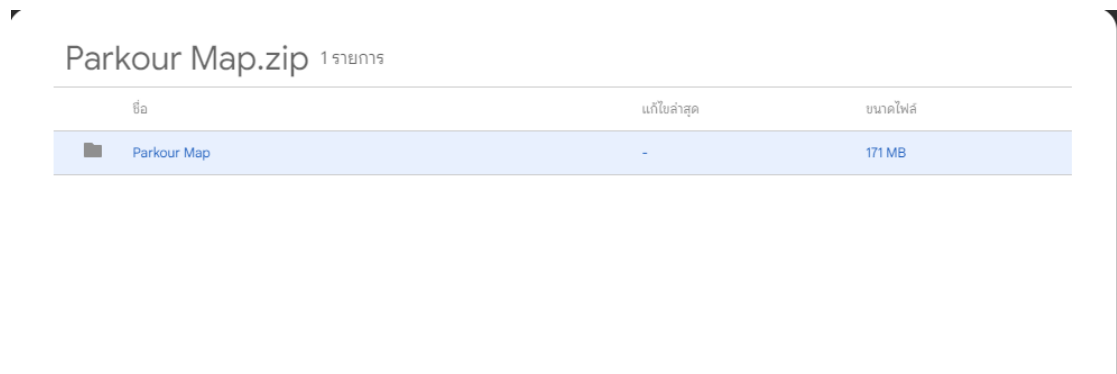
ภาคผนวก ข

คู่มือการใช้งาน Parkour Monster Map

ภาคผนวก ข คู่มือการใช้งาน Parkour Monster Map

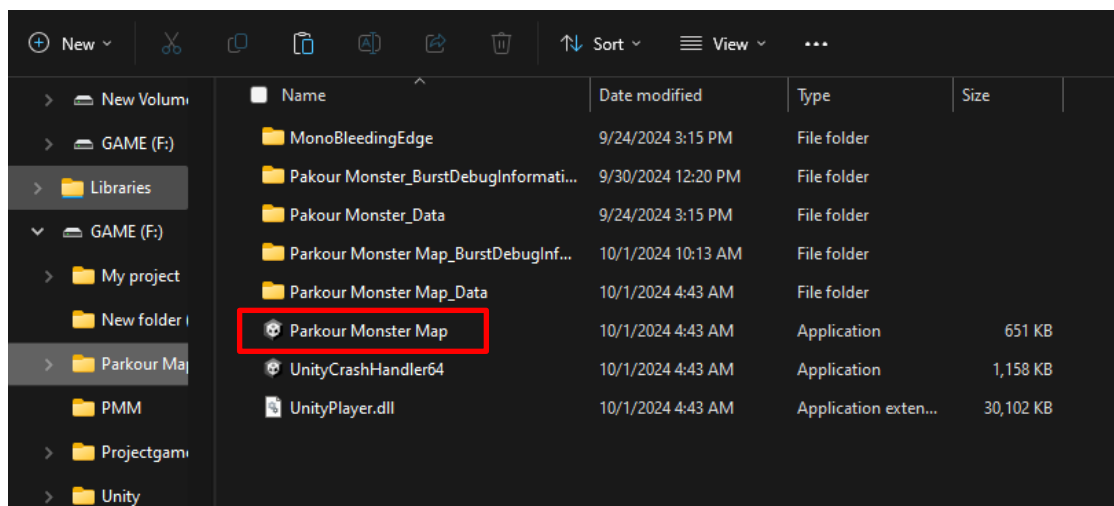
1.ติดตั้งผ่านแพลตฟอร์ม แล้วกดที่ Parkour Monster Map

https://drive.google.com/file/d/1cUllow7N6TpBy3FrCVqyU_BnhKX88Fm/view?usp=drive_link



รูปที่ ข.1 ทำการ Download

2.กดที่ Parkour Monster Map



รูปที่ ข.2 กดที่ Parkour Monster Map

3. เราจะเจอหน้าแรกของเกม



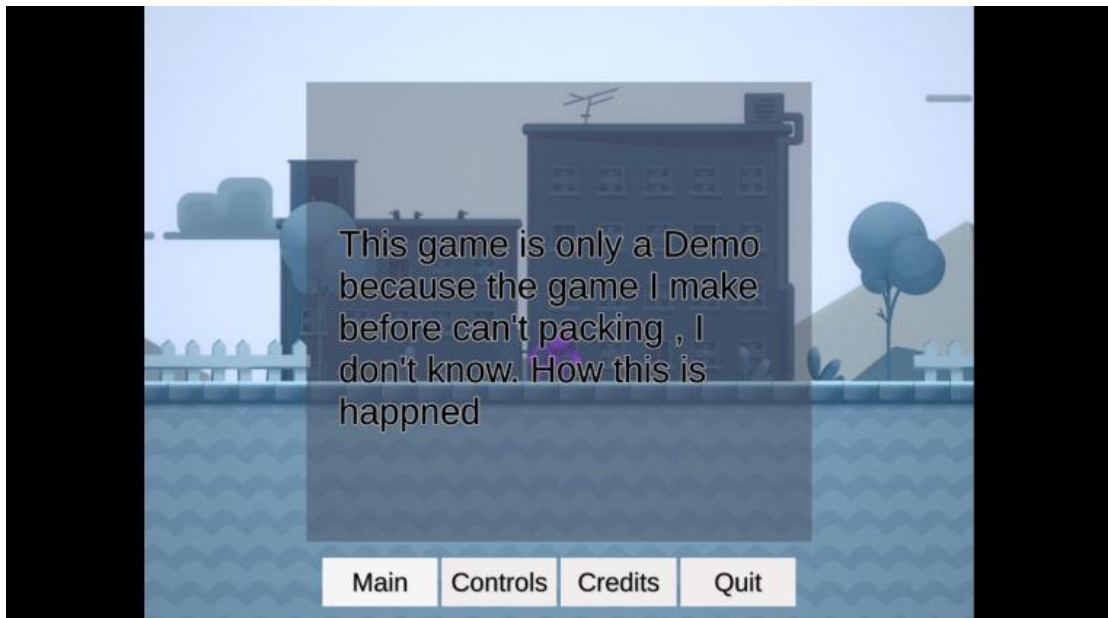
รูปที่ ข.3 หน้าแรกของเกม

4. กด Start เพื่อเริ่มเกม



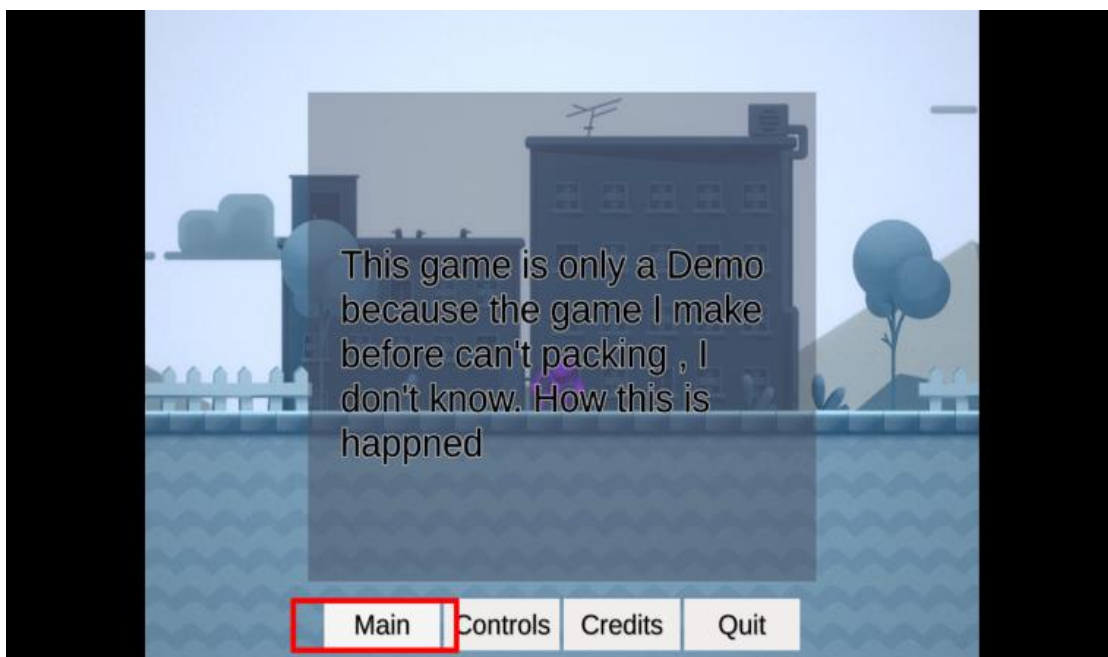
รูปที่ ข.4 กด Start เพื่อเริ่มเกม

5. หน้าเมนูก่อนเล่นเกม



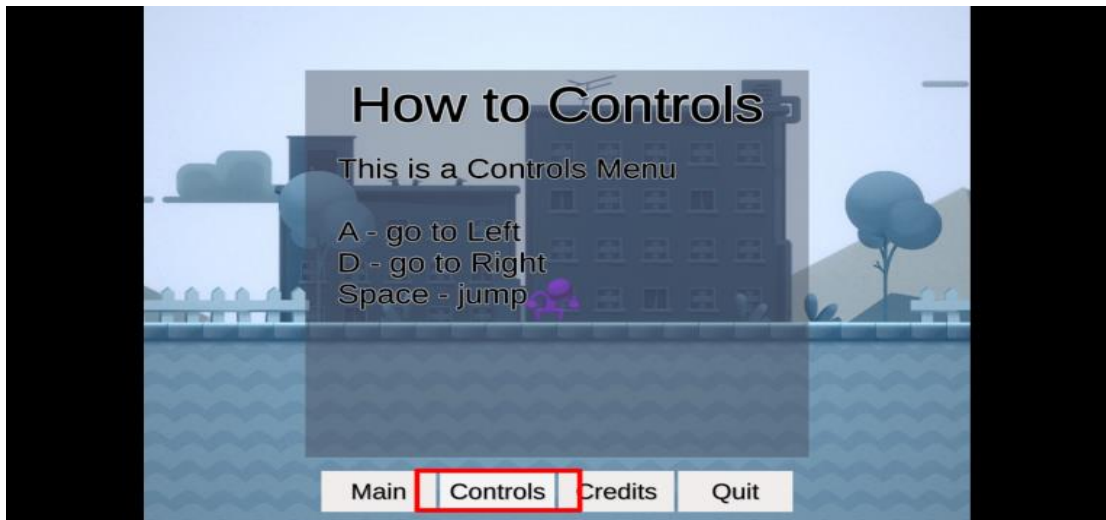
รูปที่ ข.5 กด Start เพื่อเริ่มเกม

6. หน้าของ Main จะเป็นข้อความจากผู้สร้างเกม



รูปที่ ข.6 หน้าของ Main

7. หน้าของ Controls จะเป็นหน้าสอนการควบคุมตัวละคร



รูปที่ ข.7 หน้าของ Controls

8. หน้าของ Credits จะเป็นหน้าของผู้สร้างเกมทั้ง 2 คน



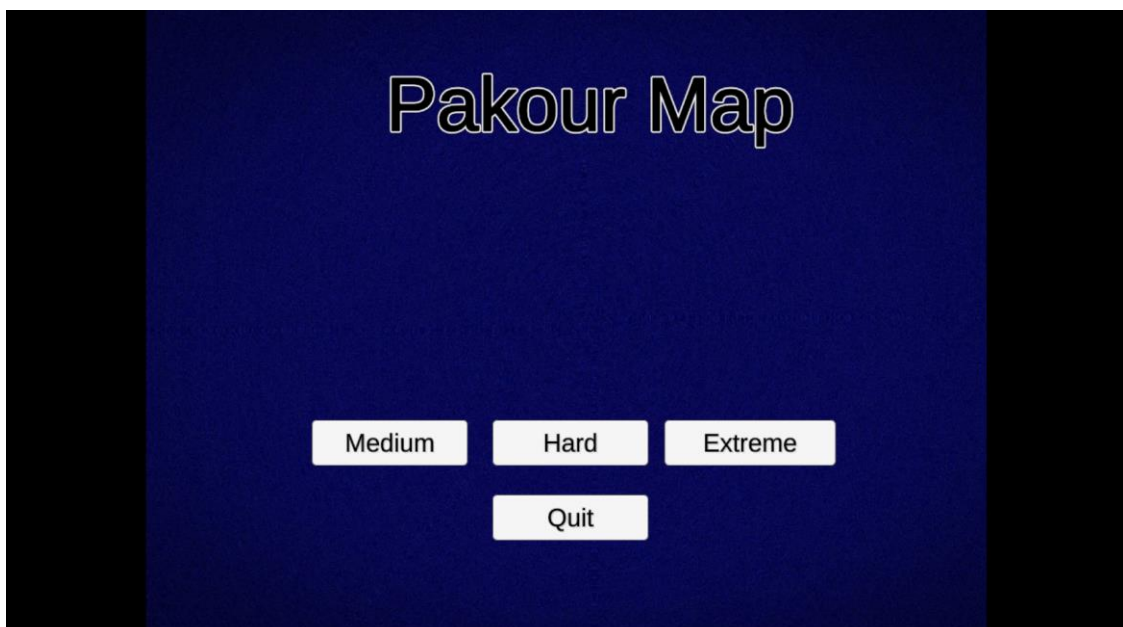
รูปที่ ข.8 หน้าของ Credits

9. กด Quit เพื่อออกไปหน้าเลือกด่าน



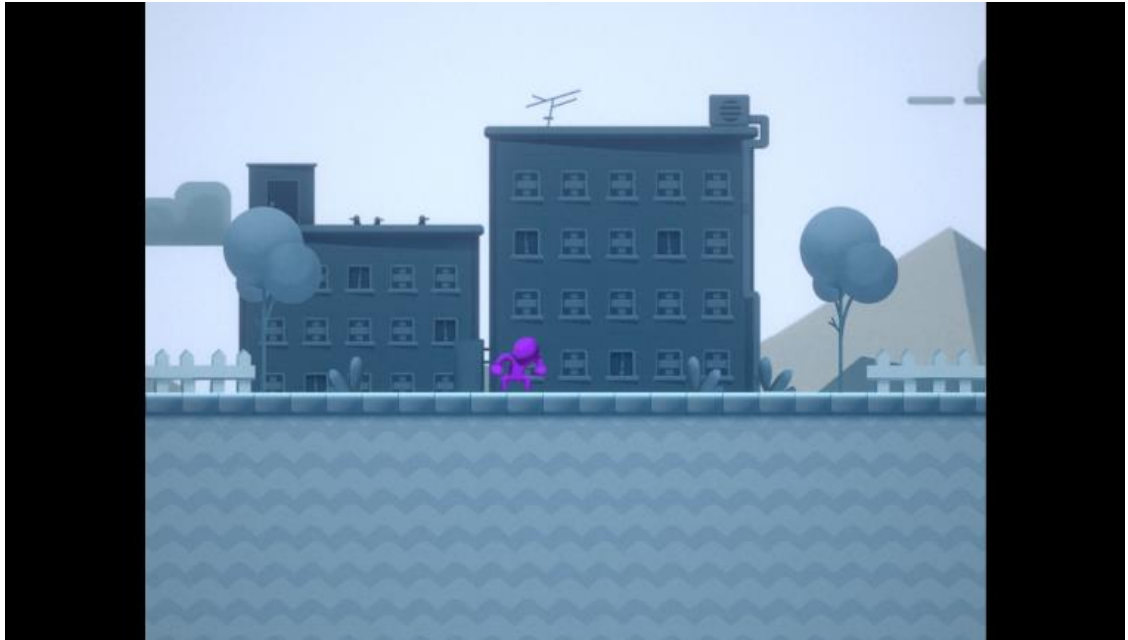
รูปที่ ข.9 Quit เพื่อไปหน้าเลือกด่าน

10. ออกมาจะเห็น 3 ด่าน คือ Medium Hard และExtreme



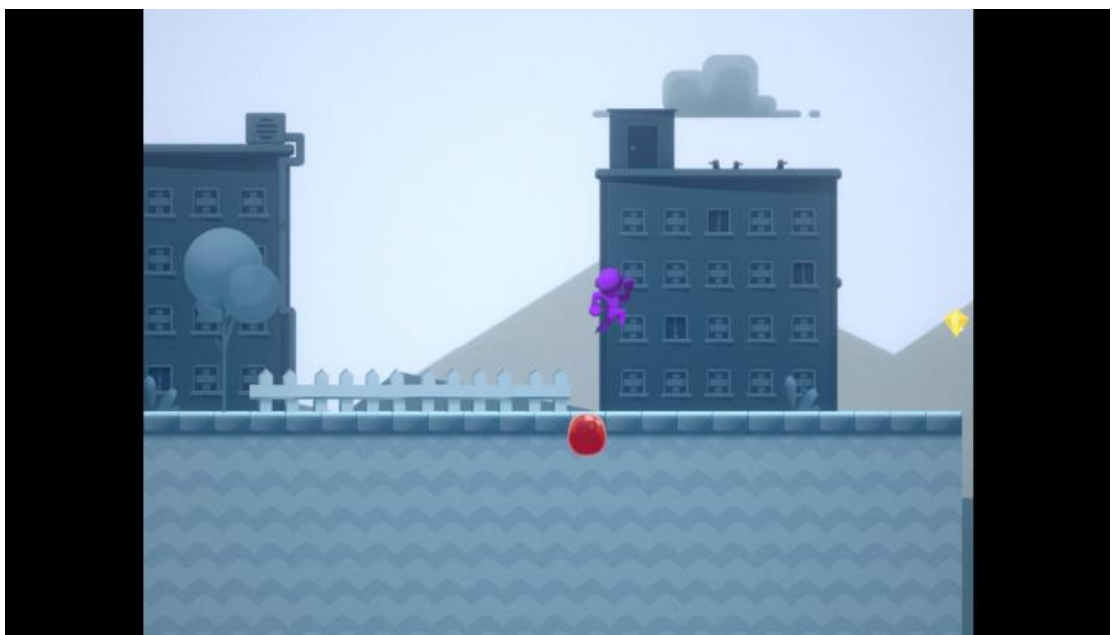
รูปที่ ข.10 หน้าเลือกด่าน

11. เล่นเกมได้เลย



รูปที่ ข.11 เริ่มเกม

12. เมื่อเจอ Monster ให้กระโดดเหยียบ



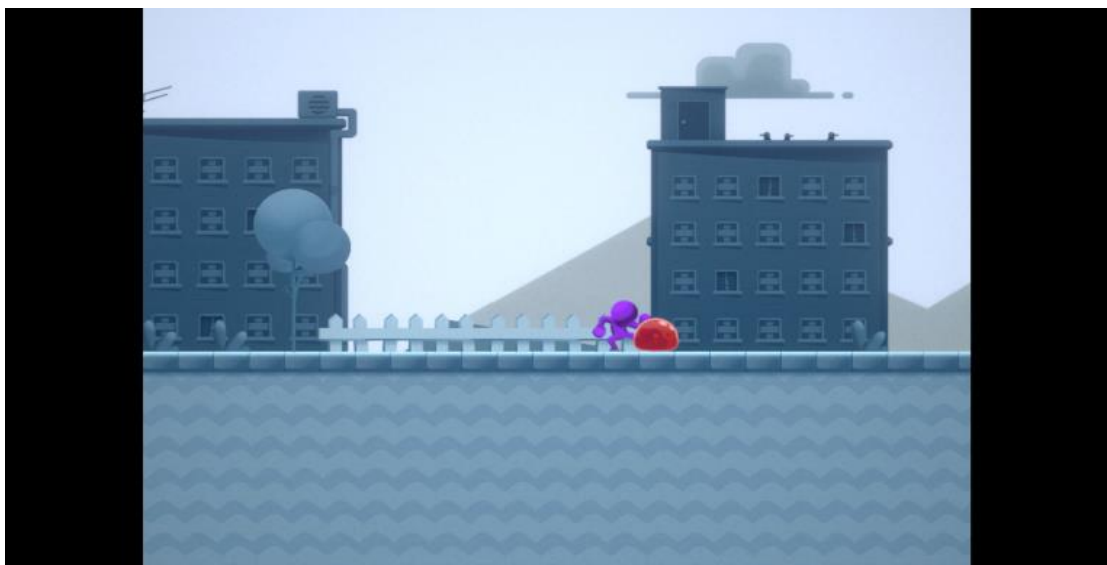
รูปที่ ข.12 Monster

13. เมื่อเราเดินไปเรื่อย ๆ จะเจอเพชร ให้เดินไปเก็บได้เลย



รูปที่ ข.13 เก็บเพชร

14. หากเราชนกับ Monster เราจะกลับไปจุดเริ่มต้น



รูปที่ ข.14 ชนกับ Monster

15. เมื่อเราเล่นไปสักพักเราก็จะเดินทางมาถึงบ้าน เพื่อไปด่านต่อไป



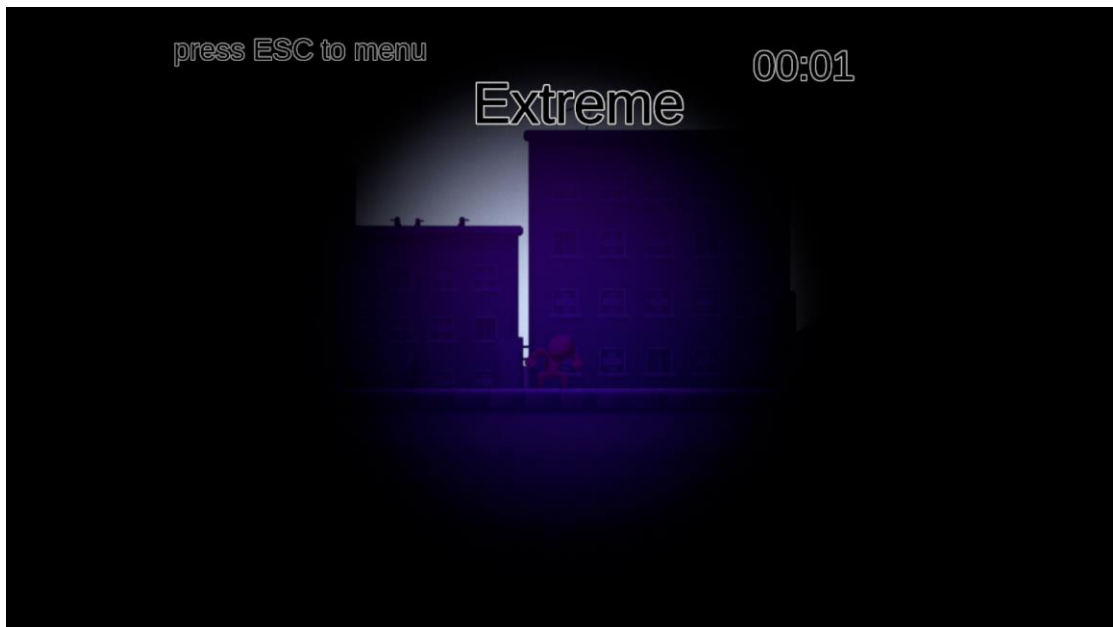
รูปที่ ข.15 จุดจบไปด่านต่อไป

16. เมื่อเราถึงด่านต่อไป



รูปที่ ข.16 ด่านต่อไป

17. เมื่อเราเล่นจบก็จะไปด่านต่อไป



รูปที่ ข.17 ด่านถัดไป

18. ฉากจบของเกมเมื่อเล่นจบแล้ว



รูปที่ ข.18 ฉากจบของเกม

19. กด Restart เพื่อไปหน้าเลือกด่าน



รูปที่ ข.19 Restart

20. กด Quit เพื่อออกจากเกม



รูปที่ ข.20 Quit เพื่อออกจากเกม

ภาคผนวก ค

เพื่อประเมินความพึงพอใจ และประสิทธิภาพของ Parkour Monster Map

เพื่อประเมินความพึงพอใจ และประสิทธิภาพของ Parkour Monster Map						
คำชี้แจง แบบสอบถามฉบับนี้ สร้างขึ้นเพื่อประเมินความพึงพอใจ และประสิทธิภาพของ Pakour Monster Map ซึ่งข้อมูล ที่ได้จากการตอบแบบสอบถามของท่าน จะเป็นแนวทางในการพัฒนาโครงการให้มี ประสิทธิภาพมากขึ้น						
ตอนที่ 1 ข้อมูลทั่วไปของผู้ตอบแบบสอบถาม (กรุณาทำเครื่องหมาย ✓ หน้าข้อความตามจริง) 1. เพศ ☐ ชาย ☐ หญิง 2. ช่วง ☐ ต่ำกว่า 16 ☐ 16-30 ☐ มากกว่า 60						
ตอนที่ 2 ให้ทำเครื่องหมาย ✓ ในช่องระดับความพึงพอใจตามสภาพจริง ซึ่งมีอยู่ 5 ระดับ คือ 5 = ระดับมากที่สุด 4 = ระดับมาก 3 = ระดับปานกลาง 2 = ระดับน้อย 1 = ระดับน้อยที่สุด						
ที่	รายการประเมิน	ระดับความพึงพอใจ				
		5	4	3	2	1
	ด้านการออกแบบ					
	1.1 ความสวยงาม ความทันสมัย น่าสนใจ					
	1.2 เกมความง่าย/ยากในการเล่น					
	1.3 เกมมีความราบรื่นในการเล่น (ไม่กระตุก/ค้าง)					
	ด้านประโยชน์และการนำไปใช้					
	2.1 การจัดวางมีความเรียบร้อย					
	2.2 ปุ่มต่างๆในเกมสามารถทำงานได้อย่างมีประสิทธิภาพ					

ตารางที่ ค.1 แบบประเมินความพึงพอใจ

.....

.....

.....

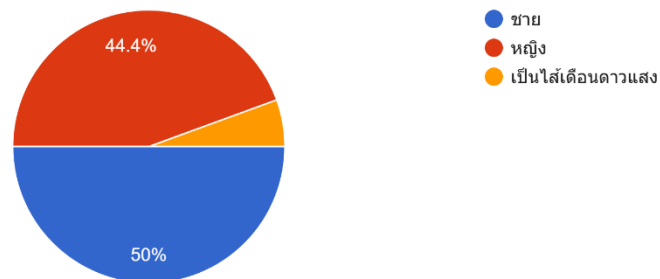
.....

.....

.....

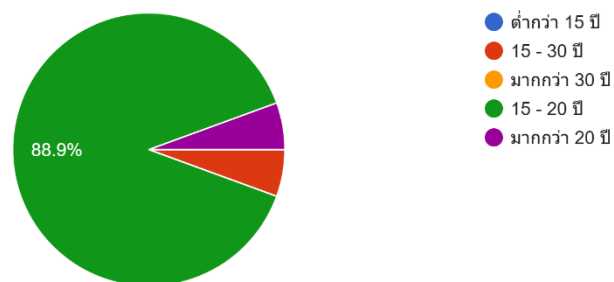
.....

เพศ
คำตอบ 18 ข้อ



แผนภูมิแสดงผลการตอบแบบสอบถามข้อมูลของผู้ตอบแบบสอบถามด้านเพศ

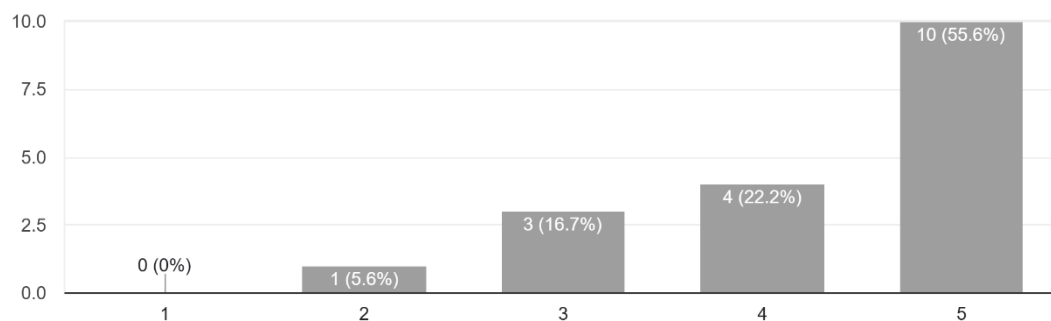
ช่วงอายุ
คำตอบ 18 ข้อ



แผนภูมิแสดงผลการตอบแบบสอบถามข้อมูลของผู้ตอบแบบสอบถามด้านอายุ

1. ความสวยงาม ความทันสมัย น่าสนใจ

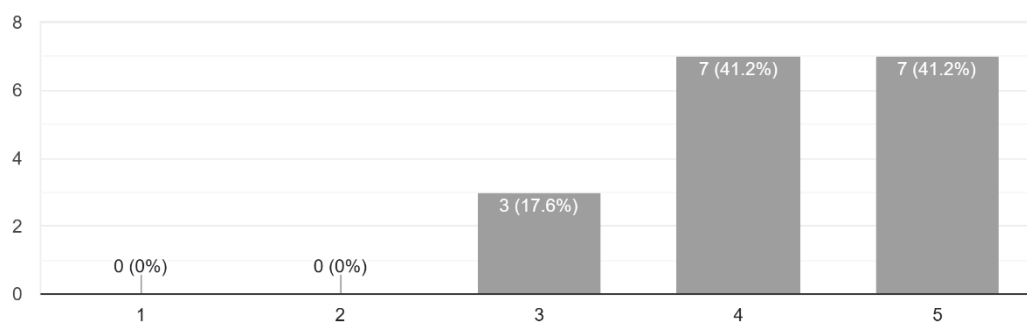
คำตอบ 18 ข้อ



แผนภูมิแสดงผลการตอบแบบสอบถามข้อมูลของผู้ตอบแบบสอบถาม

2. เกมความง่าย/ยากในการเล่นเกม

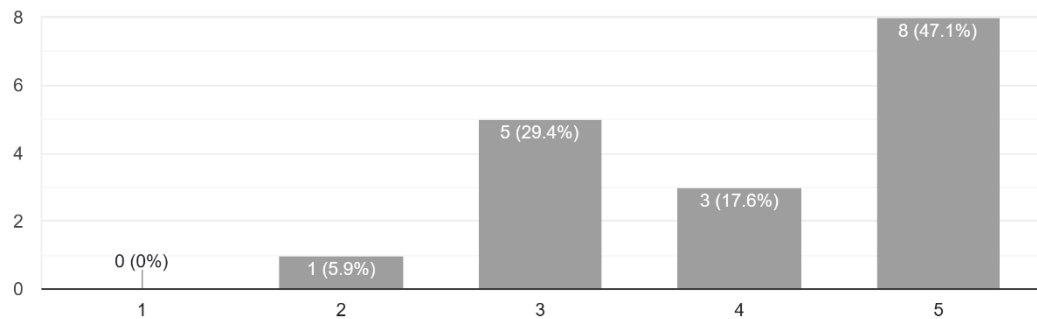
คำตอบ 17 ข้อ



แผนภูมิแสดงผลการตอบแบบสอบถามข้อมูลของผู้ตอบแบบสอบถาม

3.เกมมีความราบรื่นในการเล่น (ไม่กระตุก/ค้าง)

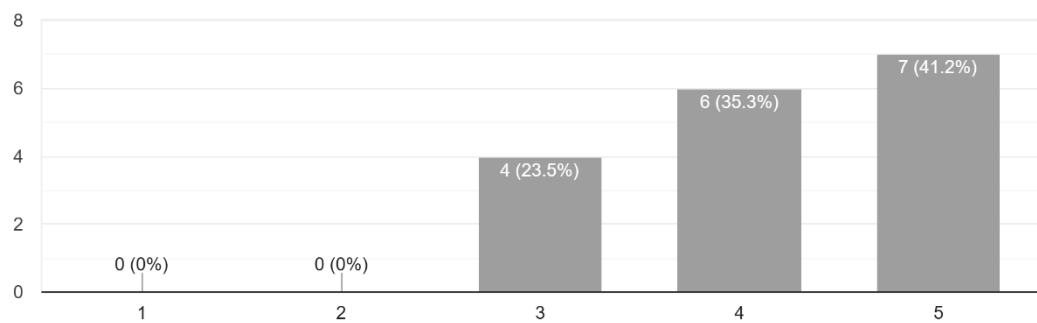
คำตอบ 17 ข้อ



แผนภูมิแสดงผลการตอบแบบสอบถามข้อมูลของผู้ตอบแบบสอบถาม

1. การจัดวางมีความเรียบร้อย

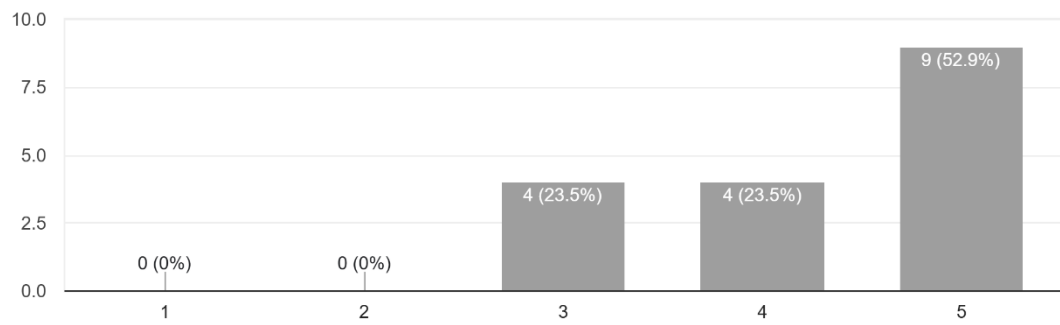
คำตอบ 17 ข้อ



แผนภูมิแสดงผลการตอบแบบสอบถามข้อมูลของผู้ตอบแบบสอบถาม

2. ปุ่มต่างๆในเกมสามารถทำงานได้อย่างมีประสิทธิภาพ

คำตอบ 17 ข้อ



แผนภูมิแสดงผลการตอบแบบสอบถามข้อมูลของผู้ตอบแบบสอบถาม

ประวัติผู้จัดทำโครงการ



ชื่อ-สกุล กฤษฎณะ ทวีธรรม
ภูมิลำเนา 104/14 ตำบลทุ่งเบญจา อำเภอท่าใหม่
จังหวัดจันทบุรี 22000
ประวัติการศึกษา
พ.ศ. 2564 ระดับมัธยมศึกษา (ม.3)
โรงเรียนประดิษฐ์ศึกษา
พ.ศ. 2567 ระดับประกาศนียบัตรวิชาชีพ (ปวช.)
แผนกเทคนิคคอมพิวเตอร์
วิทยาลัยเทคนิคจันทบุรี
Email Kitchaya.0605@gmail.com



ชื่อ-สกุล ธารทอง ชินอุดมพงศ์
ภูมิลำเนา 27/100 ซอยหลังแขวง ตำบลวัดใหม่
อำเภอเมือง จังหวัดจันทบุรี 22000
ประวัติการศึกษา
พ.ศ. 2563 ระดับมัธยมศึกษา (ม.3)
โรงเรียนเทศบาลเมืองจันทบุรี
พ.ศ. 2567 ระดับประกาศนียบัตรวิชาชีพ (ปวช.)
แผนกเทคนิคคอมพิวเตอร์
วิทยาลัยเทคนิคจันทบุรี
Email Trantong@gmail.com