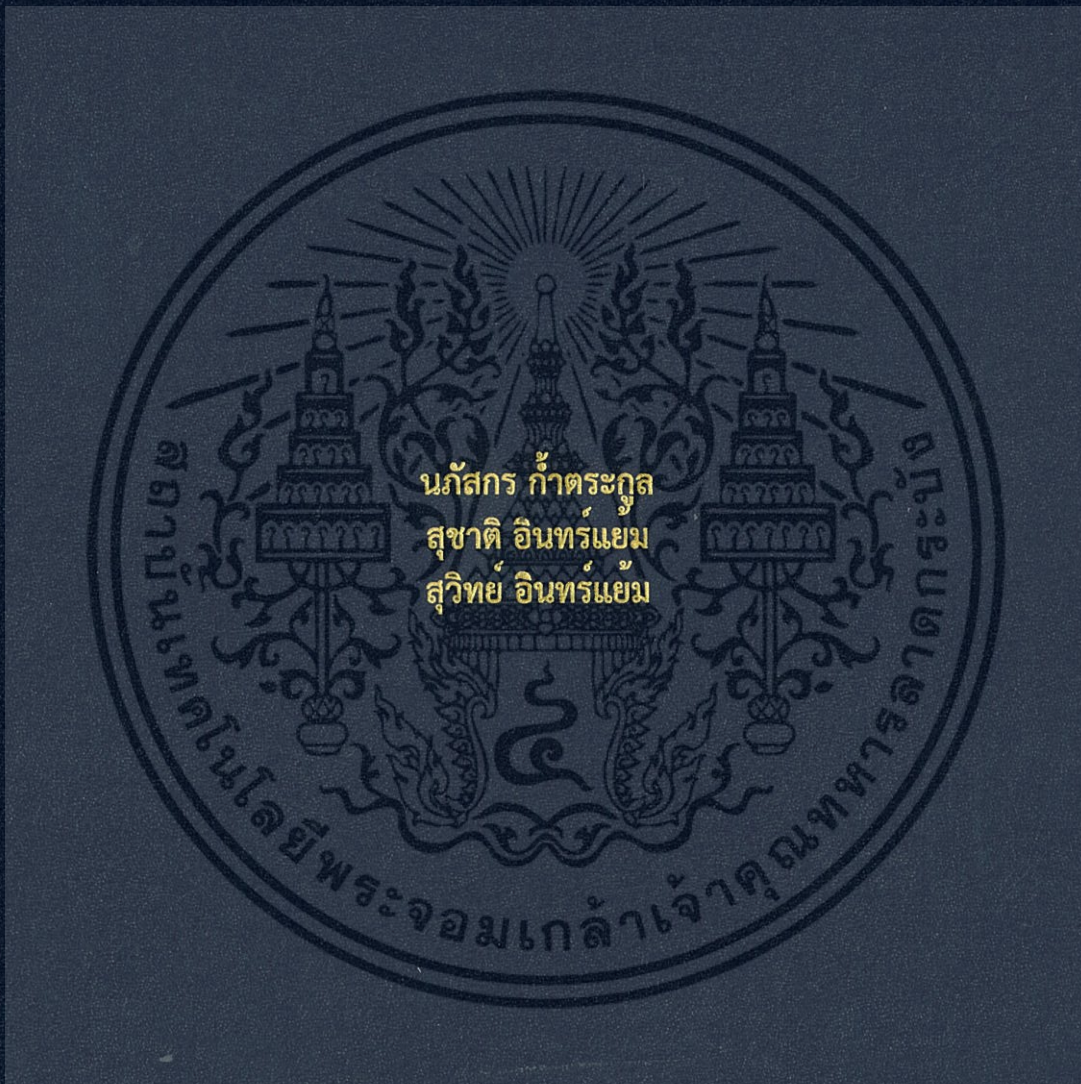


Quiz Quest Go แอปพลิเคชันเกมตอบคำถามนอกห้องเรียน

Quiz Quest Go : Game for doing quiz outside classroom



ปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิทยาศาสตรบัณฑิต สาขาวิทยาการคอมพิวเตอร์
ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2559

Quiz Quest Go แอปพลิเคชันเกมตอบคำถามนอกห้องเรียน

Quiz Quest Go : Game for doing quiz outside classroom



ปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิทยาศาสตรบัณฑิต สาขาวิทยาการคอมพิวเตอร์
ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2559

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Quiz Quest Go : Game for doing quiz outside classroom



SPECIAL PROBLEM IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF BACHELOR OF SCIENCE

IN COMPUTER SCIENCE

DEPARTMENT OF COMPUTER SCIENCE

FACULTY OF SCIENCE

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ACADEMIC YEAR 2016

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปัญหาพิเศษ
ชื่อนักศึกษา

Quiz Quest Go : แอปพลิเคชันตอบคำถามนอกห้องเรียน
นางสาวนภัสกร ก้ำตระกูล รหัสนักศึกษา 56050278
นายสุชาติ อินทร์แย้ม รหัสนักศึกษา 56050409
นายสุวิทย์ อินทร์แย้ม รหัสนักศึกษา 56050414

ปริญญา

วิทยาศาสตร์บัณฑิต

ภาควิชา

วิทยาการคอมพิวเตอร์

ปีการศึกษา

2559

อาจารย์ที่ปรึกษา

ผศ.ดร.ศรัณย์ อินทโกสุม

คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (สจล.) อนุมัติให้
ปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญา วิทยาศาสตร์บัณฑิต ประจำปี
การศึกษา 2559

คณะกรรมการสอบ	ลายมือชื่อ
ผศ.ดร.นันทิกา เบญจเทพานันท์ ประธานกรรมการ	
ผศ.สิริลักษณ์ อนันต์สถิตย์สิน กรรมการ	
ผศ.ดร.ศรัณย์ อินทโกสุม กรรมการและอาจารย์ที่ปรึกษา	

ลิขสิทธิ์ของคณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

T14959

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปัญหาพิเศษ	Quiz Quest Go : แอปพลิเคชันตอบคำถามนอกห้องเรียน	
ชื่อนักศึกษา	นางสาวนภัสกร ก้ำตระกูล	รหัสนักศึกษา 56050278
	นายสุชาติ อินทร์แย้ม	รหัสนักศึกษา 56050409
	นายสุวิทย์ อินทร์แย้ม	รหัสนักศึกษา 56050414
ปริญญา	วิทยาศาสตร์บัณฑิต	
ภาควิชา	วิทยาการคอมพิวเตอร์	
คณะ	วิทยาศาสตร์	
มหาวิทยาลัย	สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (สจล.)	
ปีการศึกษา	2559	
อาจารย์ที่ปรึกษา	ผศ.ดร.ศรัณย์ อินทโกสุม	

บทคัดย่อ

แอปพลิเคชันเกมเพื่อความบันเทิง และการศึกษาที่ใช้ระบบนำทางเข้าช่วยเพื่อสร้างแรงบรรดาลใจให้แก่ผู้เล่น และทำให้ผู้เล่นมีการเปลี่ยนแปลงพฤติกรรมตามที่ต้องการ และเพื่อให้เกิดการใช้งานแอปพลิเคชันนอกสถานที่มากขึ้น ส่งเสริมการใช้งานนอกสถานที่เพื่อสุขภาพของผู้เล่น และเปลี่ยนสภาพการเรียนการสอนให้เกิดขึ้นนอกสถานที่ โดยแอปพลิเคชันนี้สองส่วน ส่วนที่ใช้เล่น และส่วนที่ใช้สร้าง และปล่อยคำถามบนแผนที่ โดยสามารถกำหนดจุดที่ปล่อยคำถามได้เอง ผู้ใช้งานสามารถตรวจสอบคะแนน และนำค่าที่ได้มาใช้ในการประเมินคุณภาพของการเรียนการสอน ว่าส่งผลต่อผู้ใช้แอปพลิเคชันในการตอบคำถามนี้ได้หรือไม่

คำสำคัญ : เกมเพื่อการศึกษา, การเรียนรู้นอกห้องเรียน, อนิเมชัน

Title	Quiz Quest Go : Game for doing quiz outside classroom	
Students	Miss Napatsakorn Kumtragul	Student ID 56050278
	Mr.Suchart Inyam	Student ID 56050409
	Mr.Suwit Inyam	Student ID 56050414
Degree	Bachelor of Science	
Department	Computer Science	
Faculty	Science	
University	King Mongkut's Institute of Technology Ladkrabang (KMITL)	
Academic	2559	
Advisor	Asst.Prof.Dr.Sarun Intakosum	

Abstrat

Application for entertainment and education with GPS system for make inspiration to user and change user behavior. And increase user to use application at outside. Promote user to use application at outside for good healthy and change environment for learning to outside not only in classroom. This application have 2 part that for play and for make question on map that user can set question by themselves. User can check score for estimate the quality of learning that show aptitude of user.

Keywords : Game-Base Learnig, Learn Beyond Classroom, Animation

กิตติกรรมประกาศ

การทำปัญหาพิเศษหัวข้อเรื่อง Quiz Quest Go แอปพลิเคชันตอบคำถามนอกห้องเรียน สำเร็จ ลุล่วงไปได้ด้วยดี เนื่องจากบุคคลหลายท่านได้กรุณาช่วยเหลือให้คำปรึกษาแนะนำ เสนอแนะ และให้ กำลังใจ ดังนี้

ขอขอบคุณ ผู้ช่วยศาสตราจารย์ ดร.ศรัณย์ อินทโกสุม อาจารย์ที่ปรึกษาปัญหาพิเศษนี้ ที่ได้สละ เวลาให้คำปรึกษา คำแนะนำ และข้อคิดเห็นต่างๆ อันเป็นประโยชน์อย่างยิ่งในการพัฒนาระบบ อีกทั้งยัง ช่วยเสนอแนะแนวทางในการแก้ไขปัญหาต่างๆ ที่เกิดขึ้นระหว่างการดำเนินการ ให้สำเร็จลุล่วงไปได้ด้วยดี

สุดท้ายนี้ ขอขอบคุณเพื่อน และคณาจารย์ในคณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหารลาดกระบัง ที่ให้ข้อเสนอแนะในโครงการนี้ เพื่อปรับปรุงโครงการจนดี

นภัสกร ก้าตระกูล
สุชาติ อินทร์แย้ม
สุวิทย์ อินทร์แย้ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ.....	ง
สารบัญตาราง.....	จ
สารบัญรูป.....	ฉ
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์.....	2
1.3 ขอบเขตของงานวิจัย.....	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....	3
1.5 ขั้นตอนการดำเนินงาน.....	4
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง.....	5
2.1 แอปพลิเคชันที่นำมาศึกษา.....	5
2.1.1 Pokemon Go.....	5
2.1.2 Socrative.....	10
2.2 เอพีไอที่เกี่ยวข้อง.....	13
2.2.1 การติดต่อกับระบบอ้างอิงตำแหน่งบนพื้นโลกของสมาร์ทโฟนด้วย LocationService.....	13
2.2.2 การติดต่อเว็บไซต์ที่ให้บริการแผนที่ผ่านเอพีไอ OpenStreetMap.....	17
บทที่ 3 วิธีการดำเนินงานวิจัย.....	22
3.1 การออกแบบแอปพลิเคชัน.....	22
3.1.1 การออกแบบลักษณะของเกมโดยรวม.....	22
3.1.2 การออกแบบรูปแบบการเล่น และการใช้งาน.....	23
3.2 การวิเคราะห์ระบบ.....	24

สารบัญ (ต่อ)

3.2.1 การเก็บรวบรวมความต้องการของระบบ	24
3.2.2 แผนภาพยูสเคส.....	25
3.3 การออกแบบระบบ	38
3.3.1 การออกแบบแอปพลิเคชันฝั่งนักเรียน.....	38
3.3.1.1 การออกแบบตามยูสเคสการลงทะเบียน.....	38
3.3.1.2 การออกแบบตามยูสเคสการเข้าสู่ระบบ	51
3.3.1.3 การออกแบบตามยูสเคสการลงทะเบียนวิชา	55
3.3.1.4 การออกแบบตามยูสเคสการค้นหาคำถาม.....	61
3.3.1.5 การออกแบบตามยูสเคสการเล่นเก็มตอบคำถาม.....	66
3.3.1.6 การออกแบบตามยูสเคสการตรวจสอบสถานะมอนสเตอร์	74
3.3.1.7 การออกแบบตามยูสเคสการตรวจสอบคะแนน	80
3.3.1.8 การออกแบบตามยูสเคสการต่อสู่สถานประลอง	85
3.3.2 การออกแบบแอปพลิเคชันฝั่งอาจารย์.....	92
3.3.2.1 การออกแบบตามยูสเคสการลงทะเบียน.....	92
3.3.2.2 การออกแบบตามยูสเคสการเข้าสู่ระบบ	94
3.3.2.3 การออกแบบตามยูสเคสการสร้างรายวิชา.....	95
3.3.2.4 การออกแบบตามยูสเคสการสร้างคำถาม	98
3.3.2.5 การออกแบบตามยูสเคสการกำหนดตำแหน่งของคำถาม	101
3.3.2.6 การออกแบบตามยูสเคสการตรวจสอบคะแนน	103
บทที่ 4 ผลการดำเนินงาน	106
4.1 ผลการดำเนินงาน.....	106
4.1.1 การจัดการบัญชีผู้ใช้	106
4.1.1.1 การจัดการบัญชีผู้ใช้เว็บแอปพลิเคชัน.....	106
4.1.1.2 การจัดการบัญชีผู้ใช้ฝั่งแอปพลิเคชันเก็ม	108
4.1.2 แอปพลิเคชัน	110
4.2 การแข่งขันพัฒนาโปรแกรมคอมพิวเตอร์แห่งประเทศไทยครั้งที่ 19	125
4.2.1 รอบคัดเลือก.....	125
4.2.2 รอบรองชิงชนะเลิศ.....	125

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

บทที่ 5 สรุปผลการดำเนินงาน	126
5.1 สรุปผลการดำเนินงาน.....	126
5.2 อุปสรรคในการดำเนินงาน.....	127
5.3 ข้อเสนอแนะ.....	127
เอกสารอ้างอิง	128



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
3.1 ลงทะเบียน	26
3.2 เข้าสู่ระบบ	27
3.3 ลงทะเบียนวิชา	27
3.4 ค้นหาคำถาม	28
3.5 เล่นเกมตอบคำถาม	29
3.6 ตรวจสอบสถานะมอนสเตอร์	30
3.7 ตรวจสอบคะแนน	31
3.8 ต่อสู้อันตราย	32
3.9 ลงทะเบียน	34
3.10 เข้าสู่ระบบ	35
3.11 สร้างรายวิชา	35
3.12 สร้างคำถาม	36
3.13 กำหนดตำแหน่งของคำถาม	37
3.14 ตรวจสอบคะแนน	38
3.15 แสดงรายละเอียดของข้อมูลในตารางข้อมูล user	40
3.16 แสดงรายละเอียดของข้อมูลในตารางข้อมูล userbis	41
3.17 แสดงรายละเอียดของข้อมูลในตารางข้อมูล pet	41
3.18 แสดงรายละเอียดของข้อมูลในตารางข้อมูล monster	42
3.19 แสดงรายละเอียดของข้อมูลในคลาส RegisController	44
3.20 แสดงรายละเอียดของข้อมูลในคลาส RegisView	45
3.21 แสดงรายละเอียดของข้อมูลในคลาส PetRegisController	46
3.22 แสดงรายละเอียดของข้อมูลในคลาส PetSpritModel	47
3.23 แสดงรายละเอียดของข้อมูลในคลาส PetRegisView	47
3.24 แสดงรายละเอียดของข้อมูลในคลาส UserMdl	48
3.25 แสดงรายละเอียดของข้อมูลในคลาส PetMdl	49
3.26 แสดงรายละเอียดของข้อมูลในคลาส LoginController	53
3.27 แสดงรายละเอียดของข้อมูลในคลาส LoginView	54
3.28 แสดงรายละเอียดของข้อมูลในตารางข้อมูล usersubject	56

สารบัญตาราง (ต่อ)

3.29 แสดงรายละเอียดของข้อมูลในตารางข้อมูล subjects.....	56
3.30 แสดงรายละเอียดของข้อมูลในตารางข้อมูล teacher.....	56
3.31 แสดงรายละเอียดของข้อมูลในคลาส SubjectController1.....	58
3.32 แสดงรายละเอียดของข้อมูลในคลาส SubjectListMdl.....	59
3.33 แสดงรายละเอียดของข้อมูลในคลาส SubjectMdl.....	59
3.34 แสดงรายละเอียดของข้อมูลในคลาส SubjectView.....	60
3.35 แสดงรายละเอียดของข้อมูลในตารางข้อมูล exam.....	62
3.36 แสดงรายละเอียดของข้อมูลในตารางข้อมูล onmapteacher.....	62
3.37 แสดงรายละเอียดของข้อมูลในคลาส ScanController.....	64
3.38 แสดงรายละเอียดของข้อมูลในตารางข้อมูล question.....	67
3.39 แสดงรายละเอียดของข้อมูลในตารางข้อมูล answer.....	68
3.40 แสดงรายละเอียดของข้อมูลในตารางข้อมูล questionlist.....	68
3.41 แสดงรายละเอียดของข้อมูลในตารางข้อมูล userquestion.....	68
3.42 แสดงรายละเอียดของข้อมูลในตารางข้อมูล userplayed.....	69
3.43 แสดงรายละเอียดของข้อมูลในคลาส FightExamController.....	71
3.44 แสดงรายละเอียดของข้อมูลในคลาส QuizMdl.....	72
3.45 แสดงรายละเอียดของข้อมูลในคลาส QuestionMdl.....	73
3.46 แสดงรายละเอียดของข้อมูลในคลาส AnswerMdl.....	73
3.47 แสดงรายละเอียดของข้อมูลในคลาส PetController.....	77
3.48 แสดงรายละเอียดของข้อมูลในคลาส PetView.....	78
3.49 แสดงรายละเอียดของข้อมูลในคลาส ExamController.....	82
3.50 แสดงรายละเอียดของข้อมูลในคลาส ExamListMdl.....	83
3.51 แสดงรายละเอียดของข้อมูลในคลาส ExamMdl.....	83
3.52 แสดงรายละเอียดของข้อมูลในคลาส ExamView.....	84
3.53 แสดงรายละเอียดของข้อมูลในตาราง FightArenaController.....	87

สารบัญรูป

รูปที่	หน้า
2.1 หน้าจอหลักของ Pokemon Go	6
2.2 ภาพตัวอย่างลูกบอลที่ใช้จับโปเกมอน	6
2.3 ภาพตัวอย่างการจับโปเกมอน.....	6
2.4 ภาพตัวอย่างการต่อสู้.....	7
2.5 ภาพตัวอย่างกลุ่มผู้เล่น	7
2.6 ภาพตัวอย่างลูกกวาดที่ใช้วิวัฒนาการโปเกมอน.....	8
2.7 ภาพตัวอย่างการวิวัฒนาการโปเกมอน	8
2.8 การสร้างคำถามของโซคราทีฟ.....	10
2.9 การเข้าสู่ห้องเรียนของโซคราทีฟฝั่งนักเรียน.....	11
2.10 หน้าจอแสดงคำถามของโซคราทีฟฝั่งนักเรียน.....	11
2.11 ภาพตัวอย่างเมื่อตอบคำถามเสร็จ.....	12
2.12 แสดงการรายงานคำตอบ.....	12
2.13 ตัวอย่างโปรแกรมการใช้ API LocationService ของยูนิตี้.....	16
2.14 ภาพแสดงหน้าจอเว็บ openstreetmap.....	18
2.15 ภาพแสดงการสร้างออบเจกต์ และวัตถุบนออบเจกต์.....	18
2.16 ตัวอย่างโปรแกรมการใช้ API OpenStreetMap	20
3.1 ภาพรวมระบบของแอปพลิเคชันเกมควิสเคาสโก.....	23
3.2 แผนภาพยูสเคสส่วนของนักเรียน	25
3.3 แผนภาพยูสเคสส่วนของอาจารย์.....	33
3.4 ความสัมพันธ์ของตารางการลงทะเบียน	39
3.5 แผนภาพคลาสของการลงทะเบียน	43
3.6 หน้าจอการลงทะเบียน	50
3.7 หน้าจอลงทะเบียนมอนสเตอร์ประจำตัว	51
3.8 แผนภาพคลาสของการเข้าสู่ระบบ	52
3.9 หน้าจอการเข้าสู่ระบบ.....	54
3.10 ความสัมพันธ์ของตารางการลงทะเบียนวิชา.....	55
3.11 แผนภาพคลาสของการลงทะเบียนวิชา	57
3.12 หน้าจอการลงทะเบียนวิชา.....	60

สารบัญรูป (ต่อ)

3.13 ความสัมพันธ์ของตารางการค้นหาคำถาม	61
3.14 แผนภาพคลาสของการค้นหาคำถาม	63
3.15 หน้าจอการค้นหาคำถาม	65
3.16 ความสัมพันธ์ของตารางการเล่นเกมตอบคำถาม	67
3.17 แผนภาพคลาสของการเล่นเกมตอบคำถาม	70
3.18 หน้าจอการเล่นเกมตอบคำถาม	74
3.19 แผนภาพคลาสของการตรวจสอบสถานะมอนสเตอร์	76
3.20 หน้าจอตรวจสอบสถานะมอนสเตอร์	79
3.21 แผนภาพคลาสของการตรวจสอบคะแนน	81
3.22 หน้าจอการตรวจสอบรายละเอียดการตอบคำถาม	84
3.23 แผนภาพคลาสของการต่อสู้อันประหลาด	86
3.24 หน้าจอการต่อสู้อันประหลาด	91
3.25 ความสัมพันธ์ของตารางการลงทะเบียนของเว็บไซต์	92
3.26 หน้าต่างสมัครสมาชิก	93
3.27 หน้าต่างเข้าสู่ระบบ	95
3.28 หน้าต่างยืนยันตัวตน	95
3.29 ความสัมพันธ์ของตารางการสร้างรายวิชาของเว็บไซต์	96
3.30 หน้าจอแสดงแถบเมนู	96
3.31 หน้าจอแสดงแถบเมนู subject	97
3.32 หน้าจอแสดงแถบเมนู Quiz	97
3.33 หน้าต่างสร้างรายวิชา	98
3.34 ความสัมพันธ์ของตารางการสร้างคำถาม	99
3.35 หน้าต่างแสดงการสร้างคำถาม	100
3.36 หน้าจอแสดงคำถามที่สร้างไว้	100
3.37 ความสัมพันธ์ของตารางการกำหนดตำแหน่งของคำถามของเว็บไซต์	101
3.38 หน้าจอการกำหนดตำแหน่งของคำถามของเว็บไซต์	102
3.39 ความสัมพันธ์ของตารางการตรวจสอบคะแนน	103
3.40 หน้าจอเลือกวิชา และชุดคำถามที่ต้องการดูคะแนนของเว็บไซต์	105

สารบัญญรูป (ต่อ)

3.41 หน้าจอแสดงรายละเอียดคะแนนของเว็บไซต์	105
4.1 หน้าจอเริ่มเข้าใช้งานเว็บแอปพลิเคชันควิสควิสโก	106
4.2 หน้าจอป้อนชื่อเข้าใช้งานที่มีอยู่แล้วในระบบ	107
4.3 หน้าหลักของเว็บแอปพลิเคชันที่เข้าสู่ระบบแล้ว.....	107
4.4 หน้าจอเริ่มเข้าใช้งานแอปพลิเคชันเกมควิสควิสโก	108
4.5 หน้าจอแจ้งเตือน เมื่อชื่อผู้ใช้งานมีอยู่แล้วในระบบ	109
4.6 หน้าจอแจ้งเตือนเมื่อป้อนชื่อผู้ใช้งาน หรือรหัสผ่านไม่ถูกต้อง	109
4.7 หน้าหลักของแอปพลิเคชันเกมควิสควิสโก	110
4.8 หน้าจอสร้างวิชา.....	111
4.9 หน้าจอเมื่อสร้างวิชาสำเร็จ	112
4.10 หน้าจอใส่รหัสวิชาที่ต้องการลงทะเบียน.....	112
4.11 หน้าจอเมื่อเพิ่มรายวิชาแล้ว	113
4.12 หน้าจอสร้างคำถาม	114
4.13 หน้าจอแสดงผลลัพท์เมื่อสร้างคำถามเสร็จ	114
4.14 หน้าจอการเพิ่มคำถามลงในแบบทดสอบ	115
4.15 หน้าจอเพิ่มคำถามลงในแบบทดสอบครบแล้ว.....	115
4.16 หน้าจอกำหนดตำแหน่งของคำถามบนแผนที่.....	116
4.17 หน้าจอแจ้งเตือนเมื่อกำหนดตำแหน่งเสร็จ.....	116
4.18 หน้าจอกดปุ่มค้นหาคำถาม เมื่อลงทะเบียนรายวิชาแล้ว	117
4.19 หน้าจอเมื่อกดปุ่มค้นหาคำถาม เมื่อยังไม่ได้ลงทะเบียนรายวิชา.....	117
4.20 หน้าจอคำถาม	118
4.21 แสดงผลลัพท์เมื่อตอบคำถามถูกต้อง	118
4.22 แสดงผลลัพท์เมื่อตอบคำถามผิด	118
4.23 แสดงผลลัพท์เมื่อไม่ตอบคำถามตามเวลาที่กำหนด.....	119
4.24 หน้าจอแสดงรางวัล	119
4.25 หน้าการจัดการมอนสเตอร์.....	120
4.26 หน้าจอเมื่อเพิ่มค่าสถานะ	120
4.27 หน้ารายงานผล.....	121

สารบัญญรูป (ต่อ)

4.28 หน้าจอแสดงรายละเอียดของรายวิชา และแบบทดสอบที่เลือก.....	121
4.29 หน้าแสดงผลลัพธ์รายบุคคล.....	121
4.30 หน้าแสดงคะแนนของแต่ละแบบทดสอบ.....	122
4.31 หน้าแสดงคำถาม และคำตอบของแบบทดสอบที่เคยทำแล้ว.....	122
4.32 หน้าลานประลอง.....	123
4.33 หน้าสร้างห้องประลอง.....	123
4.34 หน้าลือบปี.....	124
4.35 หน้าจอการต่อสู้ลานประลอง.....	124



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันได้มีการใช้งานสมาร์ทโฟนมาพร้อมกับการเรียนการสอนเพื่อเพิ่มประสิทธิภาพการเรียนการสอน โดยได้มีการพัฒนาแอปพลิเคชัน และเว็บไซต์ที่สนับสนุนการเรียนการสอนขึ้นมามากมาย โดยแอปพลิเคชัน และเว็บไซต์ดังกล่าวก็อาจอยู่ในรูปแบบของอีเลิร์นนิ่ง หรือเกมการศึกษา โดยจุดประสงค์ของแอปพลิเคชันเหล่านี้ก็คืออำนวยความสะดวกให้กับทั้งผู้สอน และผู้เรียน เช่น การกระตุ้นให้ผู้เรียนมีส่วนร่วมในชั้นเรียน และสามารถทำแบบทดสอบออนไลน์ได้ ส่วนผู้สอนก็สามารถที่จะทราบได้ว่าผู้เรียนมีความเข้าใจในบทเรียนหรือไม่เป็นต้น อย่างไรก็ตามแอปพลิเคชันที่มีอยู่ในปัจจุบันส่วนใหญ่จะถูกพัฒนาขึ้นมาเพื่อการเรียนการสอนภายในห้องเรียน

เทคโนโลยีอ้างอิงตำแหน่งบนพื้นโลก (Global Positioning System) หรือเรียกสั้นๆ ว่าจีพีเอส (GPS) บนสมาร์ทโฟนมีความก้าวหน้าอย่างมาก ซึ่งการใช้จีพีเอสนั้นมีประโยชน์มากกับการใช้งานในกิจกรรมต่าง ๆ ที่อยู่ภายนอกอาคาร ไม่ว่าจะเป็นแอปพลิเคชันที่ใช้ในชีวิตประจำวันอย่างระบบนำทาง (Navigator) ในรถยนต์ หรือโปรแกรมเพื่อความบันเทิงอย่างเกม ซึ่งเกมที่ใช้ระบบจีพีเอส และได้รับความนิยมอย่างมากเกมหนึ่งคือ PokemonGo ซึ่งเป็นเกมที่ใช้ระบบจีพีเอสในการให้ผู้เล่นค้นหา และตามจับมอนสเตอร์ (monster) ตามสถานที่ต่างๆ ซึ่งทำให้ผู้ที่เล่นเกมมีกิจกรรมที่ออกมาทำร่วมกันภายนอกอาคาร อย่างไรก็ตามจุดประสงค์ของเกมนี้ก็เพื่อความบันเทิงอย่างเดียวเท่านั้น

จากที่กล่าวมาผู้วิจัยจึงมีแนวคิดที่จะพัฒนาแอปพลิเคชันด้านเกมการเรียนการสอนที่ใช้ได้กับกิจกรรมทั้งใน และนอกห้องเรียน โดยประยุกต์แนวคิดของการใช้จีพีเอส ผู้สอนสามารถวางคำถามไว้ในตำแหน่งใด ๆ ก็ได้ทั้งภายใน และภายนอกอาคาร ผู้เรียนมีหน้าที่ตามหา และตอบคำถาม ซึ่งถ้าตอบคำถามถูกต้องก็จะได้รับรางวัลในเกม นอกจากนี้ยังมีระบบที่จะช่วยกระตุ้นให้ผู้เรียนได้ทบทวนบทเรียนด้วยกัน ซึ่งอยู่ในรูปแบบการนำคำถามมาถามกันเองเพื่อแข่งขันกัน ในส่วนของผู้สอนจะสามารถติดตาม และประเมินได้ว่าผู้เรียนเข้าใจบทเรียนหรือไม่จากคำตอบของผู้เรียน แอปพลิเคชันนี้ยังสามารถนำไปประยุกต์กับกิจกรรมตอบคำถามในรูปแบบอื่น ๆ เช่น กิจกรรมทัศนศึกษา โดยอาจารย์สามารถส่งคำถามไปยังสถานที่ที่จะให้นักเรียนไปทัศนศึกษา เพื่อให้นักเรียนได้ค้นคว้าหาคำตอบของคำถามซึ่งเกี่ยวข้องกับสถานที่นั้น ยิ่งไปกว่านั้นจังหวัดที่ต้องการโปรโมทแหล่งท่องเที่ยว

ของตัวเองก็อาจจัดกิจกรรมตอบคำถามเกี่ยวกับแหล่งท่องเที่ยวเป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 วัตถุประสงค์ของงานวิจัย

พัฒนาแอปพลิเคชันเกมบนมือถือโดยใช้ระบบปฏิบัติการ Android ร่วมกับระบบ GPS เพื่อนำไปใช้กับการเรียนการสอน

1.3 ขอบเขตของงานวิจัย

1. เป็นแอปพลิเคชันในรูปแบบเกม และเว็บแอปพลิเคชัน
2. แอปพลิเคชันมีสองแอปพลิเคชัน คือสำหรับนักเรียน และสำหรับอาจารย์
 - 2.1 แอปพลิเคชันสำหรับนักเรียน มีขอบเขตดังนี้
 - 2.1.1 สามารถล็อกอินเพื่อใช้ในการยืนยันตัวตน และใช้งานข้อมูลจากฐานข้อมูลได้
 - 2.1.2 แสดงจุดที่ผู้ใช้อยู่บนแผนที่โดยใช้ระบบจีพีเอส (GPS) เข้าช่วย และแสดงแบบทดสอบในรัศมี 10 เมตร จากตัวผู้ใช้โดยอิงจากจีพีเอส (GPS)
 - 2.1.3 สามารถดูคะแนนย้อนหลังที่ทำได้ในแต่ละแบบทดสอบ
 - 2.1.4 มีระบบมอนสเตอร์ให้นักเรียนได้เลี้ยงเพื่อนำไปต่อสู้
 - 2.1.4.1 นักเรียนสามารถตรวจสอบสถานะความสามารถของมอนสเตอร์
 - 2.1.4.2 มอนสเตอร์สามารถพัฒนาความสามารถได้จากการทำแบบทดสอบที่อาจารย์สร้างขึ้นมา แล้วนักเรียนจะได้บิสกิตประเภทต่างๆเป็นการตอบแทน และนำบิสกิตนั้นมาให้มอนสเตอร์ของตนเอง
 - 2.1.5 มีระบบต่อสู้ ที่สามารถนำมอนสเตอร์ของผู้ใช้มาสู้กันเองได้
 - 2.1.6 ตรวจสอบอันดับของตนเองในแต่ละวิชาได้โดยยึดจากคะแนนที่ทำแบบทดสอบ
 - 2.2 แอปพลิเคชันสำหรับอาจารย์
 - 2.2.1 สามารถเปิดรายวิชาเพื่อให้นักเรียนเข้าไปลงทะเบียน
 - 2.2.2 สามารถสร้างแบบทดสอบเก็บไว้ได้
 - 2.2.3 สามารถกำหนดแบบทดสอบลงในจุดที่ต้องการได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.4 สามารถตรวจเช็คคะแนนของนักเรียนได้

3. ใช้เครื่องมือ Unity3D ในการพัฒนาแอปพลิเคชัน
4. ใช้ระบบจีพีเอส (GPS) ในการระบุตำแหน่งของผู้ใช้ เพื่อหาแบบทดสอบในบริเวณนั้น
5. พัฒนาเกมบนระบบปฏิบัติการ Android ตั้งแต่เวอร์ชัน 4.4 ขึ้นไป
6. มีการจัดอันดับของผู้ใช้งาน

1.4 ประโยชน์ที่คาดว่าจะได้รับ

- 1) เพื่อเป็นสื่อกลางในการเรียนการสอน และทำกิจกรรมต่างๆ ระหว่างนักเรียน และอาจารย์ ทั้งใน และนอกเวลาเรียน
- 2) เพื่อให้นักเรียนได้มีการทบทวนการเรียนการสอน
- 3) สามารถนำผลลัพธ์ที่ได้จากแอปพลิเคชันนี้ นำไปพัฒนา ปรับปรุงเปลี่ยนแปลง การเรียน การสอนเดิม ให้ดียิ่งขึ้น
- 4) เพื่อให้ได้รับความบันเทิง และความรู้ไปด้วยกัน
- 5) นอกจากจะใช้ในเรื่องการศึกษาแล้วยังสามารถนำไปใช้ในด้านอื่นๆได้อีกด้วย เช่น กิจกรรมการท่องเที่ยว โดยไกด์เป็นคนสร้าง และกำหนดตำแหน่งของคำถาม และให้นักท่องเที่ยวเป็นคนค้นหาคำถาม และเล่นเกมตอบคำถาม

1.5 ขั้นตอนการดำเนินงาน

- 1) สำรวจว่าในปัจจุบันมีแอปพลิเคชันหรือเกมใดที่นำระบบจีพีเอส (GPS) มาใช้ และมีแอปพลิเคชันใดเกี่ยวกับการศึกษาด้านการตอบคำถาม มีข้อดีข้อเสียอย่างไร
- 2) กำหนดวัตถุประสงค์ และขอบเขตของปัญหา
- 3) ศึกษาเนื้อหาข้อมูลที่ใช้ประกอบการทำแอปพลิเคชันเกมบนระบบปฏิบัติการแอนดรอยด์ ด้วยโปรแกรม Unity3D
- 4) ศึกษาการดึงข้อมูลแผนที่ Google Map API และการดึงข้อมูลจีพีเอส (GPS)
- 5) วิเคราะห์ปัญหา และความต้องการของระบบ
- 6) ออกแบบส่วนที่ติดต่อกันระหว่างผู้ใช้แอปพลิเคชัน และฐานข้อมูล ออกแบบระบบ
- 7) พัฒนาแอปพลิเคชันที่ได้ออกแบบไว้
- 8) ทดสอบการใช้งาน และปรับปรุงแก้ไข
- 9) สรุปผลการทำงาน ประเมิน และเขียนรายงาน
- 10) นำเสนอผลงาน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงผลการศึกษาคุณสมบัติต่างๆ ข้อดี และข้อเสียของแอปพลิเคชันเกม และเว็บแอปพลิเคชันสื่อการสอนบนอุปกรณ์เคลื่อนที่ที่มีอยู่ในปัจจุบัน เพื่อใช้เป็นข้อกำหนดความต้องการเบื้องต้น ในการพัฒนาแอปพลิเคชันเกม และเว็บแอปพลิเคชันของปัญหาพิเศษนี้

2.1 แอปพลิเคชันที่นำมาศึกษา

ปัจจุบันมีแอปพลิเคชันที่เกี่ยวกับการตอบคำถาม และใช้ในการศึกษานั้นมีอยู่มาก โดยแต่ละแอปพลิเคชันจะมีคุณสมบัติที่แตกต่างกันออกไป และเนื่องจากงานวิจัยนี้เป็นการพัฒนาแอปพลิเคชันเกม บนระบบอ้างอิงตำแหน่งบนพื้นโลกหรือระบบจีพีเอส (GPS) สำหรับการตอบคำถาม ทางผู้จัดทำจึงได้นำแอปพลิเคชันที่มีความเกี่ยวข้องกับเกมที่มีการใช้ระบบจีพีเอส และแอปพลิเคชันการตอบคำถามมาวิเคราะห์ และศึกษา ดังนี้

2.1.1 Pokemon Go

เป็นแอปพลิเคชันเกมมือถือที่ถูกพัฒนาโดยบริษัท Niantic สำหรับระบบปฏิบัติการแอนดรอยด์ ระบบปฏิบัติการไอโอเอส และอุปกรณ์นาฬิกาอัจฉริยะหรือสมาร์ทวอตช์ (Smart Watch) มีฟังก์ชันการทำงานคือ ผู้เล่นต้องใช้สมาร์ทโฟน ที่มีระบบอ้างอิงตำแหน่งบนพื้นโลกหรือจีพีเอส (GPS) เพื่อที่จะค้นหา ถ่ายรูป ต่อสู้ และฝึกมอนสเตอร์ที่เรียกว่า “โปเกมอน” ที่ปรากฏอยู่บนหน้าจอ โดยภายในเกม Pokemon Go นี้มีฟังก์ชันการทำงานหลักๆดังต่อไปนี้

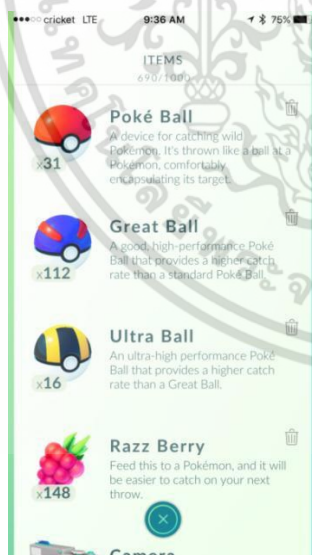
1.การค้นหาและจับโปเกมอน

ผู้เล่นสามารถค้นหาโปเกมอนที่อยู่ในบริเวณรอบในระยะเวลา 50 เมตร ซึ่งเมื่อเข้าใกล้บริเวณที่มีโปเกมอนอยู่ โปเกมอนจะปรากฏขึ้นบนหน้าจอ โดยประเภทโปเกมอนที่ปรากฏจะขึ้นอยู่กับสภาพแวดล้อมที่ผู้เล่นอยู่ เช่น โปเกมอนประเภทน้ำ จะปรากฏในบริเวณที่มีแม่น้ำ ทะเลสาบ มหาสมุทร หรือสถานที่ที่มีน้ำอยู่ในบริเวณนั้น ดังรูปที่ 2.1

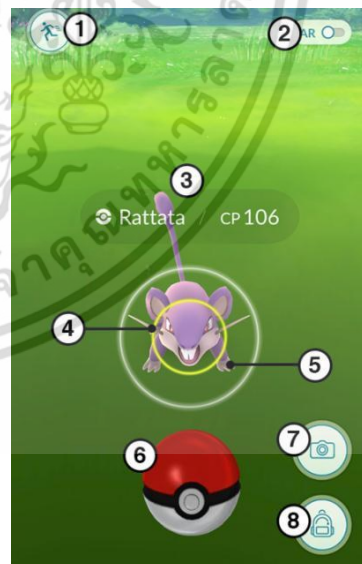


รูปที่ 2.1 หน้าจอหลักของ Pokemon Go

ในการจับโปเกมอน ผู้เล่นต้องใช้ลูกบอลที่ใช้จับโปเกมอนที่เรียกว่าโปเกบอล (Poke Balls) ซึ่งมีอยู่หลายประเภท และความสามารถของลูกบอลก็จะแตกต่างกันออกไป ทำให้เพิ่มโอกาสในการจับมากขึ้น และผู้เล่นสามารถปรับเปลี่ยนหน้าจอให้เป็น Augmented reality Mode (AR Mode) หรือก็คือเทคโนโลยีที่ผสมผสานโลกเสมือนเพิ่มเข้าไปในโลกจริง เพื่อทำให้เกิดความสมจริงมากที่สุดได้ ดังรูปที่ 2.2 และรูปที่ 2.3



รูปที่ 2.2 ภาพตัวอย่างลูกบอลที่ใช้จับโปเกมอน



รูปที่ 2.3 ภาพตัวอย่างการจับโปเกมอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. การฝึกและต่อสู้ของโปเกมอน

การฝึก และต่อสู้ของโปเกมอนจะใช้สถานที่ที่เรียกว่ายิม (Gym) ในส่วนนี้ผู้เล่นจะสามารถนำโปเกมอนในทีมมาต่อสู้กับโปเกมอนของทีมคู่แข่งได้ โดยต้องมีคุณสมบัติตามที่กำหนด ดังรูปที่ 2.4 และต้องเลือกกลุ่มก่อนเข้าการต่อสู้ โดยจะมี 3 กลุ่มให้เลือก ได้แก่ กลุ่มสีฟ้า (Mystic Team), กลุ่มสีแดง (Valor Team) และกลุ่มสีเหลือง (Instinct Team) เพื่อใช้ในการต่อสู้ แย่งชิงยิม และช่วยเหลือกันระหว่างกลุ่ม ดังรูปที่ 2.5



รูปที่ 2.4 ภาพตัวอย่างการต่อสู้



รูปที่ 2.5 ภาพตัวอย่างกลุ่มผู้เล่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

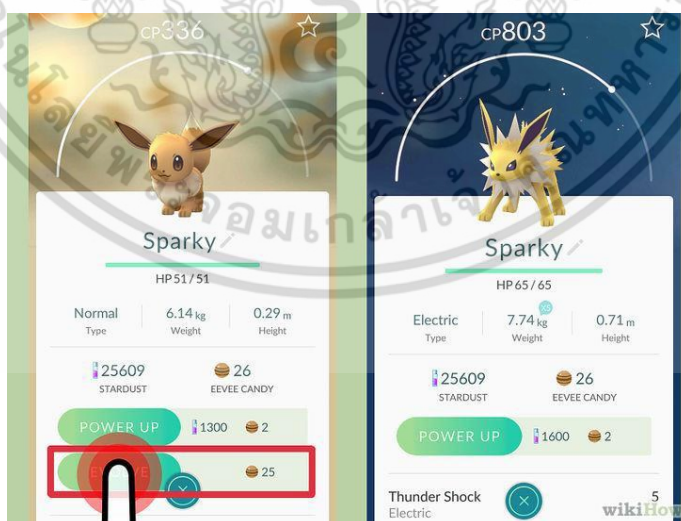
3.การวิวัฒนาการของโปเกมอน

การวิวัฒนาการโปเกมอนของผู้เล่น เวลาที่จับโปเกมอนของแต่ละประเภท จะได้รับลูกกวาด (Candy) สำหรับนำไปวิวัฒนาการ ในการวิวัฒนาการโปเกมอนแต่ละประเภท ต้องการจำนวน และประเภทของลูกกวาดแตกต่างกันออกไป เมื่อวิวัฒนาการแล้วค่าพลังโจมตี (CP) และค่าพลังชีวิต (HP) จะสูงขึ้น ดังรูปที่ 2.6 และรูปที่ 2.7

ผู้เล่นสามารถวิวัฒนาการร่างโปเกมอนได้จากการเก็บจำนวนลูกกวาดตามกำหนด ซึ่งโปเกมอนแต่ละตัวจะใช้ประเภท และจำนวนของลูกกวาดแตกต่างกันออกไป ดังรูปที่ 2.6 และเมื่อทำการวิวัฒนาการร่างโปเกมอนแล้ว จะได้รับค่าพลังโจมตี (CP) และค่าพลังชีวิต (HP) เพิ่มขึ้น ดังรูปที่ 2.7



รูปที่ 2.6 ภาพตัวอย่างลูกกวาดที่ใช้วิวัฒนาการโปเกมอน



รูปที่ 2.7 ภาพตัวอย่างการวิวัฒนาการโปเกมอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปข้อดี-ข้อเสียของแอปพลิเคชันเกม Pokemon Go

ข้อดี

1. เป็นแอปพลิเคชันที่สามารถใช้งานได้ฟรี
2. เป็นแอปพลิเคชันที่ใช้งานง่ายในการเล่น
3. ทำให้เกิดการออกกำลังกายไปในตัวขณะที่เล่น

ข้อเสีย

1. ไม่สามารถกำหนดจุดที่โปเกมอนเกิดเองได้
2. นำมาใช้ด้านการศึกษายาก เหมาะกับการท่องเที่ยว และเล่นเพื่อความบันเทิงมากกว่า



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 Socrative

โซคราทีฟ (Socrative) คือ แอปพลิเคชันสำหรับถาม-ตอบคำถามออนไลน์ เพื่อเป็นเครื่องมือช่วยเหลือผู้สอน ในการเพิ่มการตอบสนองของผู้เรียนผ่านแบบทดสอบ และเกมที่มีให้ใช้งาน ได้ทั้งบนเว็บเบราว์เซอร์ แอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ และระบบปฏิบัติการไอโอเอส โซคราทีฟมีแอปพลิเคชันสองฝั่งด้วยกันคือ แอปพลิเคชันฝั่งผู้สอน และแอปพลิเคชันฝั่งผู้เรียน ซึ่งเอาไว้ใช้งานที่ต่างกัน โดยภายในโซคราทีฟจะมีการทำงานหลักๆดังต่อไปนี้

1.การสร้างคำถาม

การสร้างคำถามเป็นการทำงานของแอปพลิเคชันฝั่งผู้สอน ผู้สอนสามารถสร้างคำถามเพื่อให้ผู้ใช้โซคราทีฟในฝั่งของนักเรียนตอบได้ โดยผู้สอนสามารถสร้างคำถามของตนเองขึ้นมาใหม่หรือนำคำถามของคนอื่นที่เคยถูกสร้างแล้วมาใช้ได้ และสามารถดูคำถามของตนเองที่สร้างไว้แล้ว และผลรายงานของคำถามที่ใช้ถามไปแล้วได้อีกด้วย ผู้สอนสามารถสร้างคำถามโดยรูปแบบของคำถามมีอยู่ 3 รูปแบบคือ คำถามปรนัย คำถามถูกหรือผิด และคำถามอัตนัย ดังรูปที่ 2.8 เป็นตัวอย่างของการสร้างคำถามในโซคราทีฟ

ANSWER CHOICE	CORRECT?
A	<input type="checkbox"/>
B	<input type="checkbox"/>
C	<input type="checkbox"/>
D	<input type="checkbox"/>
E	<input type="checkbox"/>

รูปที่ 2.8 การสร้างคำถามของโซคราทีฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

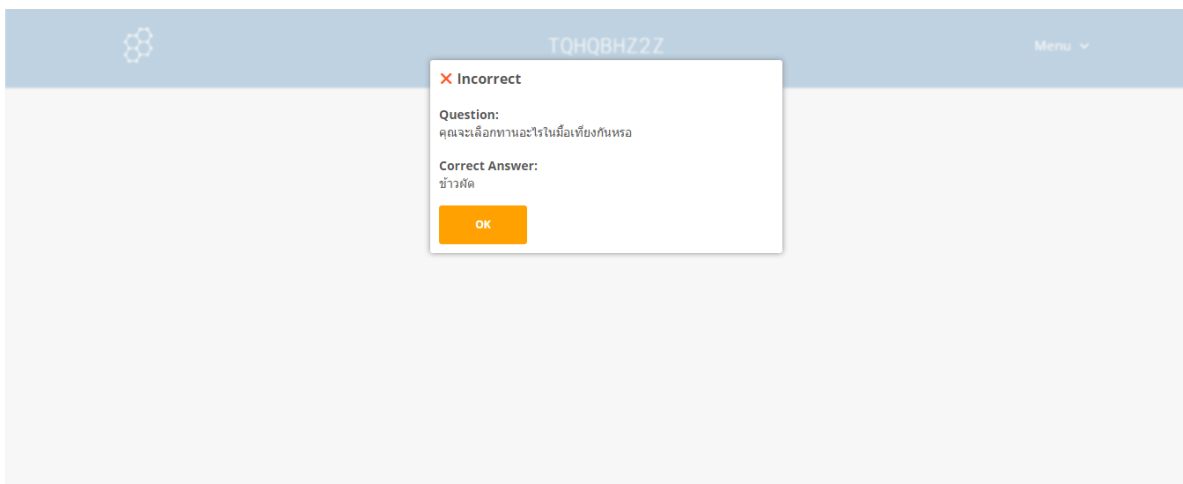
2.การตอบคำถาม

การตอบคำถามเป็นการทำงานของโซคราตีฟฝั่งนักเรียน การใช้งานโซคราตีฟฝั่งนักเรียนจะต้องใส่ชื่อของห้องที่ผู้สอนสร้างไว้เพื่อเข้าใช้งานดังรูป 2.9 เพื่อที่จะเข้าสู่ห้องเรียนของผู้สอนที่สร้างไว้ และรอคำถามที่ผู้สอนจะถามมา เมื่อมีคำถามถูกส่งมาคำถามจะแสดงที่ตรงกลางของหน้าจอ ดังรูป 2.10 หากตอบเสร็จแล้วจะมีข้อความแจ้งเตือนของเฉลยดังรูป 2.11

รูปที่ 2.9 การเข้าสู่ห้องเรียนของโซคราตีฟฝั่งนักเรียน

รูปที่ 2.10 หน้าจอแสดงคำถามของโซคราตีฟฝั่งนักเรียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.11 ภาพตัวอย่างเมื่อตอบคำถามเสร็จ

3.การแสดงผลคำตอบ

การแสดงผลคำตอบเป็นการทำงานของโซคราทีฟฝั่งผู้สอน โดยผู้สอนสามารถดูรายงานได้เมื่อผู้สอนนำคำถามที่สร้างไว้มาใช้งาน ซึ่งรายงานนี้จะแสดงผลคำตอบที่ผู้เรียนตอบคำถามที่ถูกใช้อยู่ตอนนั้น ดังรูปที่ 2.12 ซึ่งผู้สอนสามารถเก็บผลการตอบของผู้เรียนเป็นรายงานไว้ โดยสามารถเก็บในรูปแบบไฟล์เอ็กเซลล์ (.exel) หรือพีดีเอฟ (.pdf) ก็ได้

Name A-Z	Progress	#1
ADAM	100% ✓	A
julia	100% ✓	A
Class Total	0%	

รูปที่ 2.12 แสดงการรายงานคำตอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปข้อดี-ข้อเสียของแอปพลิเคชัน Socrative

ข้อดี

1. สามารถดูผลการตอบในเวลานั้นได้
2. เหมาะกับการใช้งานด้านการศึกษา
3. แบ่งแอปพลิเคชันเป็นสองแอปพลิเคชัน เพื่อแยกการใช้งานตามหน้าที่อย่างชัดเจน

ข้อเสีย

1. นักเรียนไม่สามารถดูคำถามที่เคยทำไปแล้วได้
2. เหมาะกับการเรียนการสอนภายในห้องเรียนเท่านั้น

2.2 เอพีไอที่เกี่ยวข้อง

เอพีไอ (Application Programming Interface: API) คือ วิธีการที่ระบบปฏิบัติการ ไลบรารี ชุดโค้ด หรือบริการอื่นๆ เปิดให้โปรแกรมคอมพิวเตอร์สามารถติดต่อเรียกใช้งานได้ เช่น การเรียกชุดคำสั่งผ่านทาง ไลบรารี เป็นต้น

ในหัวข้อนี้จะอธิบายถึงเอพีไอสองตัว ที่ถูกนำมาใช้ในโปรแกรมได้แก่ LocationService ซึ่งเป็นเอพีไอที่ทางยูนิตี้ (Unity) จัดเตรียมไว้เพื่อรองรับระบบอ้างอิงตำแหน่งบนพื้นโลก และ OpenStreetMap ซึ่งเป็นเอพีไอสำหรับการใช้งานแผนที่ซึ่งเรานำข้อมูลที่ได้จากเอพีไอนี้มาสร้างเป็นแผนที่บนแอปพลิเคชัน

2.2.1 การติดต่อกับระบบอ้างอิงตำแหน่งบนพื้นโลกของสมาร์ทโฟนด้วย

LocationService

ยูนิตี้นั้นได้จัดเตรียมฟังก์ชันการทำงานสำหรับการติดต่อกับระบบอ้างอิงตำแหน่งบนพื้นโลกสำหรับสมาร์ทโฟน ซึ่งบรรจุอยู่ในคลาส LocationService เพื่อให้ผู้พัฒนานั้นสามารถนำค่าที่ได้จากระบบอ้างอิงตำแหน่งบนพื้นโลก มาใช้ในแอปพลิเคชันเกมของตัวเองได้ โดย LocationService มีรายละเอียดดังต่อไปนี้

ตัวแปร

isEnabledByUser	คือตัวแปรที่ไว้สำหรับบอกว่าระบบอ้างอิงตำแหน่งบนพื้นโลกของผู้ใช้อนุญาตให้ใช้หรือไม่
lastData	คือตัวแปรที่ไว้สำหรับบอกค่าต่างๆทางภูมิศาสตร์ ซึ่งก่อนจะใช้ตัวแปรนี้ จะต้องเรียกใช้คำสั่งของคลาสนำเข้าข้อมูลก่อน Input.location.start()
status	คือ ตัวแปรที่ใช้บอกสถานะการทำงานของ LocationService

คำสั่ง

start	คือเริ่มใช้งาน LocationService เพื่อรับค่าของระบบอ้างอิงตำแหน่งบนพื้นโลก
stop	คือหยุดใช้งาน LocationService

การใช้งาน

การใช้งานระบบอ้างอิงตำแหน่งบนพื้นโลก เพื่อใช้ระบุตำแหน่ง ณ ปัจจุบันสำหรับการค้นหาคำถาม จำเป็นต้องใช้งานคลาส LocationService เพื่อนำค่าละติจูด ลองจิจูด จากระบบอ้างอิงตำแหน่งบนพื้นโลกมาใช้ในการค้นหาคำถามที่อยู่ใกล้ผู้ใช้งาน โดยจะทำการเขียนคำสั่งลงโปรแกรมเพื่อเก็บค่าละติจูดลองจิจูด และจึงนำค่าที่ได้ไปใช้อีกที โดยมีขั้นตอนการใช้งาน LocationService ดังนี้

เมื่อเริ่มการทำงานของโปรแกรม จะต้องทำการตรวจสอบว่าผู้ใช้ได้เปิดระบบอ้างอิงตำแหน่งบนพื้นโลกบนสมาร์ทโฟนหรือไม่ด้วยการเช็คเงื่อนไข ถ้าหากยังไม่เปิดการใช้งาน จะจบการทำงานลงหรือรอจนกว่าผู้ใช้งานจะเปิดระบบ เพราะถ้าหากไม่จบการทำงานหรือรอให้ผู้ใช้งานเปิดใช้งานระบบ จะไม่สามารถนำค่าจากระบบอ้างอิงตำแหน่งบนพื้นโลกมาใช้ได้ และเกิดความผิดพลาดขึ้น โดยเราจะตรวจสอบการใช้งานระบบอ้างอิงตำแหน่งบนพื้นโลกผ่านตัวแปร isEnabledByUser ดังคำสั่งต่อไปนี้

```
if(!Input.location.isEnabledByUser)
```

จากคำสั่ง เมื่อเราเรียกใช้ตัวแปร `isEnabledByUser` หาผู้ใช้งานเปิดใช้งานระบบอ้างอิงตำแหน่งบนพื้นโลก ตัวแปรจะให้ค่าเป็นจริง และข้ามขั้นตอนนี้ไป แต่หาผู้ใช้งานไม่ได้เปิดระบบอ้างอิงตำแหน่งบนพื้นโลกไว้จะคืนค่าเป็นเท็จ และเข้าสู่การทำงานในส่วนการตรวจสอบเงื่อนไขซึ่งอาจให้ทำการรอก่อนกว่าผู้ใช้งานเปิดระบบอ้างอิงตำแหน่งบนพื้นโลกหรือทำการปิดโปรแกรม หลังจากตรวจสอบแล้วว่ามีเปิดใช้งานระบบอ้างอิงตำแหน่งบนพื้นโลกแล้ว จะต้องทำการเรียกใช้คำสั่ง `start()` เพื่อให้สามารถอ้างอิงค่าละติจูด ลองจิจูด มาใช้ในโปรแกรมได้ ดังคำสั่งต่อไปนี้

```
Input.location.start()
```

จากคำสั่ง `Input.location.start()` จะเป็นการสั่งให้มีการเปิดระบบนำเข้าสู่ข้อมูลจากระบบอ้างอิงตำแหน่งบนพื้นโลกมาใช้งาน ซึ่งหลังจากเรียกคำสั่ง `start()` แล้ว จะต้องตรวจสอบว่าสามารถเปิดใช้งานการนำเข้าสู่ข้อมูลของระบบอ้างอิงตำแหน่งบนพื้นโลกได้หรือไม่ ด้วยการตรวจสอบผ่านตัวแปร `status` ถ้าหากทำงานได้จะแสดงสถานะ `Initializing` หากทำงานไม่ได้จะเป็น `Failed` โดยจะใช้คำสั่งดังนี้

```
if (Input.location.status == LocationService.Initializing)
```

จากคำสั่งจะเป็นการตรวจสอบค่าสถานะการนำเข้าสู่ข้อมูลโดยตรวจสอบว่าค่าที่อยู่ในตัวแปร `status` นั้นมีค่าเท่ากับ `Initializing` หรือไม่ ถ้าหากใช่แสดงว่าเราสามารถนำข้อมูลจากระบบอ้างอิงตำแหน่งบนพื้นโลกมาใช้งานได้ ซึ่งเมื่อตรวจสอบแล้วว่าการนำเข้าสู่ข้อมูลระบบอ้างอิงตำแหน่งบนพื้นโลกสามารถทำงานได้ เราจะสามารถนำค่า ละติจูด ลองจิจูด มาใช้กับระบบได้ โดยสามารถนำค่าอ้างอิงตำแหน่งบนพื้นโลกนี้มาใช้โดยผ่านตัวแปร `lastData` ดังคำสั่งต่อไปนี้

```
Input.location.lastData.latitude
```

```
Input.location.lastData.longitude
```

จากคำสั่ง จะได้รับค่าจากตัวแปร `lastData.latitude` เป็นค่าละติจูด และค่าของตัวแปร `lastData.longitude` เป็นค่าลองจิจูด ซึ่งเราสามารถนำไปเก็บในตัวแปร และใช้งานในการอ้างอิงตำแหน่งของผู้ใช้งานได้ และต่อไปจะเป็นตัวอย่างโปรแกรมซึ่งมีการใช้เอพีไอ `locationService`

ตัวอย่างโปรแกรม

```

1  using UnityEngine;
2  using System.Collections;
3
4  public class TestLocationService : MonoBehaviour
5  {
6      IEnumerator Start()
7      {
8          if (!Input.location.isEnabledByUser)
9              yield break;
10
11         Input.location.Start();
12
13         int maxWait = 20;
14         while (Input.location.status == LocationServiceStatus.Initializing && maxWait > 0)
15         {
16             yield return new WaitForSeconds(1);
17             maxWait--;
18         }
19
20         if (maxWait < 1)
21         {
22             print("Timed out");
23             yield break;
24         }
25
26         if (Input.location.status == LocationServiceStatus.Failed)
27         {
28             print("Unable to determine device location");
29             yield break;
30         }
31         else
32         {
33             print("Location: " + Input.location.lastData.latitude + " " + Input.location.lastData.longitude + " "
34                 + Input.location.lastData.altitude + " " + Input.location.lastData.horizontalAccuracy + " "
35                 + Input.location.lastData.timestamp);
36         }
37     }
38     Input.location.Stop();
39 }
40

```

รูปที่ 2.13 ตัวอย่างโปรแกรมการใช้ API LocationService ของยูนิตี้

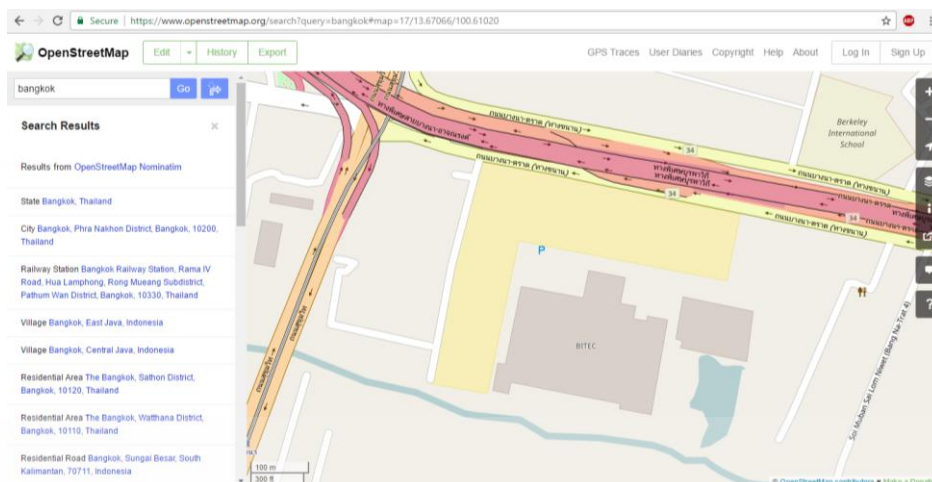
อธิบายโปรแกรม

จากตัวอย่างโปรแกรมที่แสดงในรูปที่ 2.13 เป็นโปรแกรมที่ใช้ในการตรวจสอบการใช้งานระบบอ้างอิงตำแหน่งบนพื้นโลก โดยหากมีการเปิดใช้จะแสดงตำแหน่งที่อยู่ในปัจจุบันเป็นค่า ละติจูดและลองจิจูด เริ่มต้นโปรแกรมจะเช็คก่อนว่ามีการเปิดระบบนำทางจีพีเอสหรือไม่ โดยตรวจสอบผ่านคำสั่ง `if(!Input.location.isEnabledByUser)` ถ้าหากไม่ได้ทำการเปิดระบบอ้างอิงตำแหน่งบนพื้นโลก จะเข้าไปทำตามคำสั่งในเงื่อนไขซึ่งคือ `yield break` ซึ่งจะทำให้จบการทำงานของโปรแกรมในทันที แต่หากผู้ใช้เปิดระบบนำทางไว้แล้ว จะทำการข้ามคำสั่ง `yield break` ไป และเมื่อตรวจสอบแล้วว่ามีการเปิดใช้งานระบบอ้างอิงตำแหน่งบนพื้นโลก บรรทัดต่อมาได้ทำการเปิดระบบการนำค่าจากระบบอ้างอิงบนพื้นโลกมาใช้ด้วยคำสั่ง `Input.location.Start()` ซึ่งหลังจากใช้คำสั่งเปิดระบบการนำค่าจากระบบอ้างอิงบนพื้นโลกแล้ว โปรแกรมได้ทำการเก็บค่าตัวเลขไว้ในตัวแปรชื่อ `maxWait` ซึ่งเก็บค่าเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เวลาในการรอให้เปิดระบบนำเข้าสู่ข้อมูลจากระบบอ้างอิงตำแหน่งบนพื้นโลกสำเร็จ โดยบรรทัดต่อมา โปรแกรมได้ทำการใช้คำสั่งวนซ้ำ ซึ่งจะวนจนกว่าระบบจะเปิดระบบนำเข้าสู่ข้อมูลจากระบบอ้างอิงตำแหน่งบนพื้นโลกสำเร็จ ดังโปรแกรมในบรรทัดที่ 14 ถึง 18 โดยหากยังเปิดไม่เสร็จจะหยุดรอ 1 วินาที และลดค่า maxWait ลง หากลดลงต่ำกว่า 0 และจะออกจากการทำซ้ำ และตรวจว่าเวลาในการรอน้อยกว่า 0 หรือไม่ หากน้อยกว่าจะแสดงข้อความว่าเกินเวลาที่กำหนด และจบการทำงาน โปรแกรม ดังคำสั่งในบรรทัดที่ 20 ถึง 24 แต่เมื่อตรวจสอบแล้วเวลาเหลือมากกว่า 0 จะทำการตรวจสอบสถานะการนำเข้าสู่ข้อมูลจากระบบอ้างอิงตำแหน่งบนพื้นโลกอีกครั้งหากด้วยคำสั่ง `if(Input.location.status == Input.location.Failed)` ซึ่งหากผิดพลาดจะแสดงข้อความว่าผิดพลาด และปิดโปรแกรมไปดังคำสั่งในบรรทัดที่ 28 และ 29 แต่หากการเปิดระบบนำเข้าสู่ข้อมูลจากระบบอ้างอิงตำแหน่งบนพื้นโลกไม่มีข้อผิดพลาดจะนำค่าละติจูด และลองจิจูดมาใช้งาน ผ่านคำสั่ง `Input.lastData.latitude` และ `Input.lastData.longitude` ดังคำสั่งบรรทัดที่ 33 ถึง 35 หลังจากนั้น คือการปิดการนำเข้าสู่ข้อมูลจากระบบอ้างอิงตำแหน่งบนพื้นโลก โดยใช้คำสั่ง `Input.location.Stop()` เพื่อปิดการนำเข้าสู่ข้อมูลจากระบบอ้างอิงตำแหน่งบนพื้นโลก

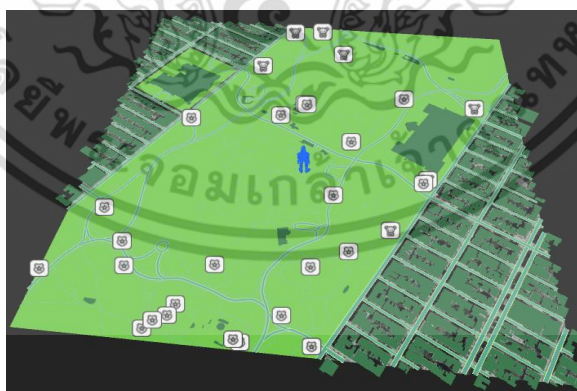
2.2.2 การติดต่อเว็บไซต์ที่ให้บริการแผนที่ผ่านเอพีไอ OpenStreetMap

โอเพนสตรีทแมพ (OpenStreetMap) คือ เว็บไซต์ที่ให้บริการข้อมูลแผนที่แบบโอเพนซอร์ส ซึ่งเกิดจากความร่วมมือของผู้ที่ต้องการใช้บริการแผนที่แบบฟรีร่วมมือกันสร้างขึ้น โดยจุดเด่นอยู่ที่การที่เปิดให้ผู้ที่ต้องการแก้ไขส่วนต่างๆบนแผนที่สามารถเพิ่มเติม หรือปรับปรุงแผนที่ได้ตลอดเวลา โอเพนสตรีทแมพเหมาะแก่การใช้งานในการทำโปรเจกของยูนิตี เนื่องจากมีการให้บริการข้อมูลแบบเวกเตอร์ไทล์ (Vector Tile) ที่ใช้สำหรับสร้างแอปเจกต์ในโปรแกรมยูนิตี โดยข้อมูลที่ส่งมาจะอยู่ในรูปแบบไฟล์ JSON ซึ่งเป็นไฟล์ที่ใช้เก็บข้อมูลประเภทหนึ่ง เราสามารถนำไฟล์ JSON ที่ได้รับจากโอเพนสตรีทแมพมาสร้างเป็นแอปเจกต์หรือพื้นผิวที่มีข้อมูลเหมือนพื้นที่บนโลกจริงๆได้บนโปรแกรมยูนิตี เช่น สร้างวัตถุเป็นพื้นที่มีอาคาร และถนนบนนั้นได้โดยใช้ข้อมูลที่อยู่บนไฟล์ JSON



รูปที่ 2.14 ภาพแสดงหน้าจอบริการ openstreetmap

บริการข้อมูลของโอเพนสตรีทแมปอยู่ในรูปแบบของบริการชิ้นส่วนแผนที่ (Tile Map Service : TMS) ซึ่งเป็นบริการที่จะให้ข้อมูลในรูปแบบของระบบตารางที่ใช้บอกตำแหน่งวัตถุ โดยจะแบ่งออกเป็นหลายระดับ แต่ละระดับจะบ่งบอกถึงข้อมูลข้างบนออบเจ็กต์ เช่น ระดับที่ 1 จะมี 4 เวกเตอร์บอกตำแหน่งของออบเจ็กต์ที่จะสร้างระดับที่ 2 จะมี 16 เวกเตอร์บอกวัตถุบนออบเจ็กต์ใน ระดับที่ 1 เช่น แม่น้ำ หรืออาคาร เป็นต้น โดยเวกเตอร์ที่ได้จะมีรูปแบบเป็น $(x, y, level)$ ค่า x และ y เป็นคู่อันดับในการบอกตำแหน่งวัตถุบนยูนิต์ และระดับ (level) คือระดับของเวกเตอร์ ตัวอย่างค่า เวกเตอร์ $(0,0,1)$, $(0,1,1)$, $(1,0,1)$, $(1,1,1)$ จะสร้างออบเจ็กต์สี่เหลี่ยมตามคู่อันดับ x และ y



รูปที่ 2.15 ภาพแสดงการสร้างออบเจ็กต์ และวัตถุบนออบเจ็กต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งาน

ในการใช้งานแผนที่ผ่านเอพีไอของโอเพนสตรีทแมพนั้น จะจำเป็นต้องทำการเรียก url ของข้อมูลที่ต้องการโดยใช้งานผ่าน url โดยใช้พารามิเตอร์ ได้แก่ ค่าละติจูด ลองจิจูด ระดับความลึกของแผนที่ ถ้าแผนที่มีความลึกมากจะได้ข้อมูลที่มีรายละเอียดมากขึ้น และรหัสของเอพีไอ ซึ่งรหัสของเอพีไอจำเป็นต้องสมัครสมาชิกบนเว็บไซต์ของโอเพนสตรีทแมพ ก่อนจึงจะได้รับ และเมื่อได้ข้อมูลทั้งหมดแล้วจะทำการใช้งาน url นี้เพื่อเรียกใช้งานข้อมูล ซึ่งจะรับกลับมาเป็นไฟล์ JSON และใช้ในการสร้างแผนที่ต่อไป ซึ่งจะยกตัวอย่างการใช้งานเอพีไอโอเพนสตรีทแมพผ่านตัวอย่างโปรแกรมซึ่งใช้การเรียก url ของโอเพนสตรีทแมพเพื่อรับไฟล์ JSON สำหรับสร้างแอปเจ็ท ดั้งนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างโปรแกรม

```

44     Vector2 realPos = tileCenter.tileCoordinates (zoom);
45
46     var tileurl = realPos.x + "/" + realPos.y;
47
48     var baseUrl = "https://tile.mapzen.com/mapzen/vector/v1/";
49     List <string> layerNames = map.layerNames();
50     layerNames.RemoveAll(str => String.IsNullOrEmpty(str));
51
52     var url = baseUrl + string.Join(",",layerNames.ToArray())+"/"+zoom+"/";
53
54     if (layers.Count() > 2) {
55         url = baseUrl + "all/" + zoom + "/";
56     }
57
58
59     var completeurl = url + tileurl + ".json";
60
61     if (map.mapzen_api_key != null && map.mapzen_api_key != "") {
62         completeurl = completeurl + "?api_key=" + map.mapzen_api_key;
63     }
64
65     if (Application.isPlaying) { //Runtime build
66
67         if (map.useCache && FileHandler.Exist(gameObject.name))
68         {
69             yield return StartCoroutine (ParseJson(FileHandler.LoadText (gameObject.name)));
70         }
71         else
72         {
73             Debug.Log (completeurl);
74             var www = new WWW(completeurl);
75             yield return www;
76             if (www.error == null && www.text.Length > 0) {
77                 FileHandler.SaveText (gameObject.name, www.text);
78             } else if (www.error != null && (www.error.Contains("429") || www.error.Contains("timed out"))) {
79                 Debug.LogWarning("Tile data reload "+www.error);
80                 yield return new WaitForSeconds(1);
81                 yield return StartCoroutine (LoadTileData(map,tilecenter,zoom,layers,delayedLoad));
82                 yield break;
83             } else if (www.error != null && (www.error.Contains("401"))) {
84                 Debug.LogWarning("[MapZen API KEY] "+www.error+ " - A Mapzen Api key is required to make GoMap v
85                 yield break;
86             }
87             else {
88                 Debug.LogWarning("Tile data missing "+www.error);
89                 ((GOMap)m).tiles.Remove(this);
90                 GameObject.Destroy(this.gameObject);
91                 yield break;
92             }
93             yield return StartCoroutine (ParseJson(www.text));
94         }
95     }
96
97     mapData = job.OutData;
98
99     yield return StartCoroutine(ParseTileData(map,tileCenter,zoom,layers,delayedLoad,layerNames));
100

```

รูปที่ 2.16 ตัวอย่างโปรแกรมการใช้ API OpenStreetMap

จากตัวอย่างโปรแกรมด้านบนเป็นการเรียกใช้เอพีไอของโอเพนสตรีทแมพผ่านทาง url ของโอเพนสตรีทแมพ โดยจะเก็บค่าพารามิเตอร์ได้แก่ ละติจูด ลองจิจูด ความลึก ระดับชั้นที่ต้องการรหัสเอพีไอ แล้วประกอบกันเพื่อเรียกใช้งานเอพีไอ และรับค่ากลับมาเป็น JSON โดยสามารถอธิบายขั้นตอนสำคัญได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เริ่มชุดคำสั่งจะมีการรับค่าคู่อันดับ ซึ่งเป็นค่าละติจูด และลองจิจูดจากระบบอ้างอิงตำแหน่งบนพื้นโลก มาแปลงเป็นส่วนหนึ่งของ url จากโปรแกรมชุดคำสั่งนี้ได้มีการรับค่าคู่อันดับละติจูด ลองจิจูด เก็บไว้ในตัวแปร tileCenter เรียบร้อยแล้ว และทำการแปลงค่าคู่ละติจูด ลองจิจูดที่ได้เป็นข้อความ เพื่อใช้เป็นพารามิเตอร์ส่งไปพร้อมกับ url โดยใช้คำสั่ง `Vector2 realPos = tileCenter.tileCoordinate (zoom) และ var tileurl = realPos.x + "/" + realPos.y` ในบรรทัดที่ 44 และ 46 เพื่อแปลงเป็นข้อความใช้สำหรับต่อท้าย url ในโปรแกรมก่อนใช้งาน หลังจากนั้นโปรแกรมได้สร้างตัวแปรสำหรับเก็บ url ของโอเพนสตรีทแมพ เพื่อให้ได้ข้อมูลของแผนที่โดยรอบ โดยใช้คำสั่งในบรรทัดที่ 52 คือ `var baseUrl = "https://tile.mapzen.com/mapzen/vector/v1/"` และเมื่อเก็บค่า url และเก็บพารามิเตอร์ตำแหน่งแล้ว ให้ทำการเชื่อมต่อ url เพื่อร้องขอข้อมูล โดยจะมีการสร้างตัวแปรสำหรับบอกว่าจะรับข้อมูลรายละเอียดอะไรบ้าง เช่น แม่น้ำ ถนน อาคาร ใช้ตัวแปรชื่อว่า `layerNames` และเชื่อมต่อ url ทั้งหมดเข้าด้วยกัน แล้วต่อท้ายด้วย `.json` เพื่อขอข้อมูลกลับมาเป็นไฟล์ JSON ดังคำสั่งในบรรทัด 52 ถึง 59 เมื่อเชื่อมต่อ url เข้ากับพารามิเตอร์ทั้งหมดแล้ว จะต้องทำการเชื่อมต่ออีกส่วนหนึ่งเข้าด้วย คือ รหัสเอพียู หากไม่มีรหัสเอพียูจะทำให้ไม่สามารถร้องขอข้อมูลแผนที่จากเว็บไซต์โอเพนสตรีทแมพได้ จึงต้องทำการเชื่อมต่อ url ด้วย รหัส เอ พียู ด้วย คำสั่ง `completeurl = completeurl + "?api_key=" + map.mapzen_api_key` ในบรรทัดที่ 62 และเมื่อได้ url ที่สมบูรณ์แล้ว จะต้องทำการเรียกใช้ url ผ่านคลาส WWW ซึ่งเป็นคลาสที่ใช้สำหรับติดต่อกับเว็บไซต์ภายนอกโปรแกรม ดังคำสั่งในบรรทัดที่ 74 และ 75 ซึ่งจะทำให้ได้รับข้อมูลจากโอเพนสตรีทแมพกลับมา ซึ่งข้อมูลที่ได้รับจะเป็นไฟล์ JSON ที่มีข้อมูลอยู่ในนั้น แต่เนื่องจากยูนิตันนั้นจะมองว่าค่าที่ได้รับมาเป็นข้อความอยู่ จึงจำเป็นต้องแปลงข้อมูลที่อยู่ในรูปของข้อความนั้นเป็นไฟล์ JSON จริงๆเสียก่อนด้วยคำสั่ง `ParseJson (www.text)` ดังคำสั่งในบรรทัดที่ 94 เพื่อให้โปรแกรมยูนิตี (Unity) อ่านข้อมูลได้ และนำไปใช้ต่อไป

บทที่ 3

การวิเคราะห์และออกแบบระบบ

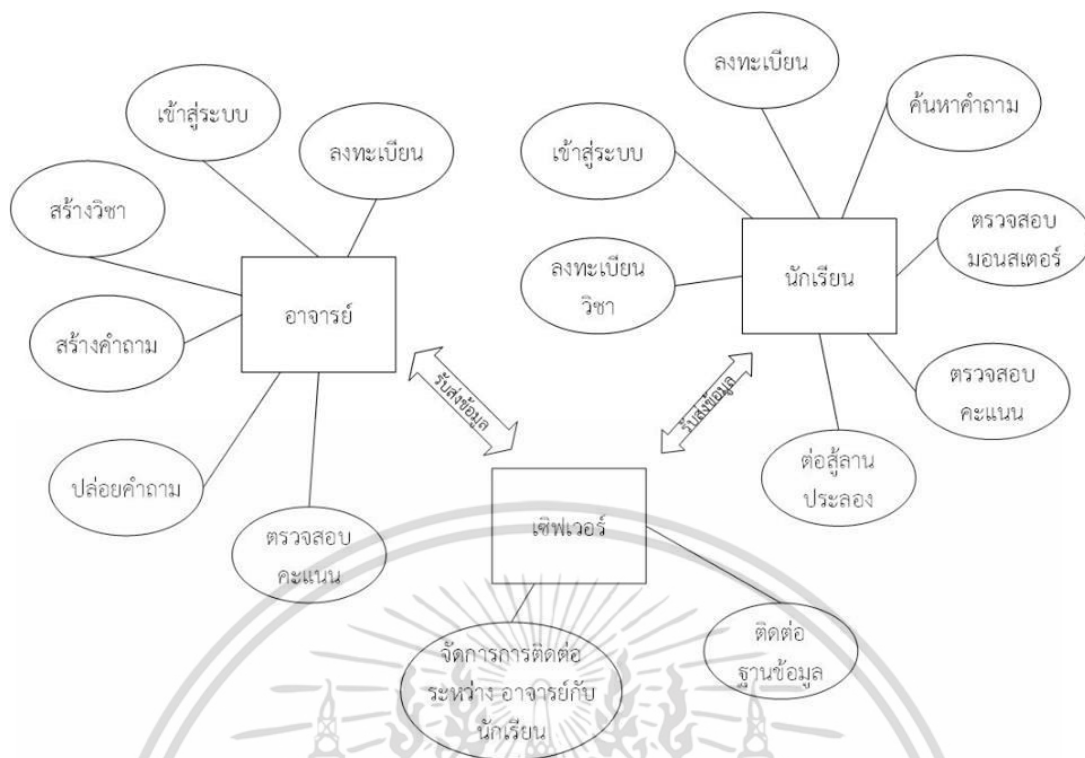
บทนี้จะเริ่มต้นโดยการออกแบบลักษณะเกมที่พัฒนาขึ้น โดยจะกล่าวถึงการออกแบบเกม และนำเอาการออกแบบเกมนั้นมาเขียนข้อกำหนดความต้องการเพื่อวิเคราะห์ และออกแบบระบบ และจะกล่าวถึงการวิเคราะห์ และออกแบบเว็บแอปพลิเคชันซึ่งเป็นอีกส่วนหนึ่งของระบบ

3.1 การออกแบบแอปพลิเคชัน

3.1.1 การออกแบบลักษณะของแอปพลิเคชันโดยรวม

แอปพลิเคชันเกมควิสแควสโกเป็นเกมตอบคำถามแบบออนไลน์ที่ผสมเข้ากับระบบอ้างอิงตำแหน่งบนพื้นโลก หรือจีพีเอส (GPS) โดยผู้เล่นจะได้เลือกมอนสเตอร์ 1 ตัวตอนเริ่มต้น และพัฒนาความสามารถของมอนสเตอร์ ผ่านการกินบิสกิตประเภทต่างๆ ซึ่งได้จากการตอบคำถามที่อาจารย์กำหนดตำแหน่งไว้บนแผนที่ โดยเป็นแผนที่จริงทางภูมิศาสตร์จึงต้องใช้ระบบจีพีเอสในการบอกตำแหน่ง และเมื่อมอนสเตอร์ของผู้เล่นมีค่าพลังต่างๆมากขึ้น ผู้เล่นสามารถนำมอนสเตอร์ของตัวเองไปประลองกับเพื่อนที่ลานประลอง โดยการสลักกันถามคำถามที่ตัวเองมีอยู่ เพื่อชิงของรางวัลได้ ซึ่งจะอธิบายเพิ่มเติมในหัวข้อที่ 3.4.1.5.3 นอกจากนี้เมื่อตอบคำถามแล้วผู้เล่นสามารถมาดูคะแนนภายหลังได้ว่าตนเองทำแบบทดสอบได้คะแนนเท่าไร เพื่อใช้ในการประเมินความรู้ของตัวเองได้

สาเหตุที่ออกแบบเกมในลักษณะนี้เนื่องจากต้องการให้ผู้ใช้ที่เป็นนักเรียนมีสุขภาพที่ดี และได้รับบรรยากาศในการเรียนแบบใหม่ ตัวเกมจึงออกแบบให้มีการเดินในสถานที่จริง หรือให้มีการเล่นนอกห้องเรียน เพื่อให้ผู้เรียนเล่นเกม และปรับเปลี่ยนพฤติกรรม ปรับเปลี่ยนความคิดว่าสามารถเรียน หาความรู้ได้จากข้างนอกได้อย่างสนุกสนาน และมีจุดมุ่งหมาย



รูปที่ 3.1 ภาพรวมระบบของแอปพลิเคชันเกมควิสแควสโก

3.1.2 การออกแบบรูปแบบการเล่นและใช้งาน

แอปพลิเคชันควิสแควสโกนั้นถูกแบ่งออกเป็น 2 แอปพลิเคชัน คือ แอปพลิเคชันส่วน
ของอาจารย์ และแอปพลิเคชันส่วน of นักเรียน ซึ่งมีวิธีการใช้งานแตกต่างกันออกไปดังนี้

1) แอปพลิเคชันส่วน of อาจารย์

อาจารย์จะต้องสมัครสมาชิก และเข้าสู่ระบบก่อนใช้งานในแอปพลิเคชัน
ส่วน of อาจารย์ จากนั้นอาจารย์ต้องสร้างรายวิชาเรียน เพื่อเป็นการจำกัดให้นักเรียนเห็นได้เฉพาะ
คำถามที่ลงทะเบียนวิชาไว้ และแยกนักเรียนตามรายวิชาเพื่อตรวจสอบผลคะแนน อาจารย์จะได้รับ
รหัสวิชาซึ่งไว้ให้นักเรียนลงทะเบียน เมื่ออาจารย์สร้างวิชาเรียบร้อยแล้วจะสามารถเข้าไปในรายวิชา
นั้นๆ เพื่อสร้างคำถาม โดยจะทำได้ 2 รูปแบบคือ 1.สร้างคำถามเป็นข้อๆเก็บไว้ และ 2.นำคำถามที่
สร้างไว้เก็บเป็นชุดคำถาม เมื่ออาจารย์ได้สร้างชุดคำถามเรียบร้อยแล้ว จะสามารถเลือกกำหนด
ตำแหน่ง of คำถามผ่านการกำหนดตำแหน่งบนแผนที่ได้ และอาจารย์สามารถตรวจสอบคะแนนของ
นักเรียนภายหลังได้โดยแบ่งตามรายวิชาเรียน และหัวข้อแบบทดสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) แอปพลิเคชันส่วนของนักเรียน

นักเรียนจะต้องสมัครสมาชิก และเข้าสู่ระบบก่อนใช้งานแอปพลิเคชันเกมควิซเคสโกลจากนั้นนักเรียนจะได้รับบทบาทเป็นผู้ฝึกสัตว์เลี้ยง โดยจะได้เลือกสัตว์เลี้ยง 1 ตัวเป็นพวกสัตว์เลี้ยงที่ได้จะมีค่าความสามารถเป็นค่าเริ่มต้นตามประเภทสัตว์เลี้ยงที่เลือก เมื่อตอบคำถามแล้วได้รับบิสกิต ก็จะสามารถนำไปเพิ่มค่าความสามารถตามจำนวนบิสกิตที่มี โดยบิสกิตแต่ละประเภทจะให้ค่าความสามารถที่แตกต่างกันออกไปตามประเภท นักเรียนจะต้องลงทะเบียนในรายวิชาผ่านการใส่รหัสวิชาที่ได้มาเพื่อที่จะได้เห็นคำถามของรายวิชานั้นได้ หลังจากนั้นเมื่อนักเรียนลงทะเบียนวิชาแล้วจะสามารถออกตามหาคำถามของรายวิชานั้นได้ โดยผ่านหน้าจอหลักซึ่งเป็นแผนที่สำหรับตรวจว่ามีคำถามบริเวณรอบตัวนักเรียนหรือไม่ ถ้าหากมีคำถามจะมีรูปมอนสเตอร์ตัวเล็กๆ ขึ้นมาที่หน้าจอ และเมื่อนักเรียนกดเข้าไปจะพบกับมอนสเตอร์ ซึ่งถามคำถามอยู่ ตามจำนวนที่อาจารย์กำหนด เมื่อตอบถูกจะได้รับของรางวัล เป็นบิสกิตประเภทต่างๆ ที่สามารถพัฒนาความสามารถมอนสเตอร์ที่เลี้ยงไว้ได้ และเมื่อมอนสเตอร์มีความสามารถมากขึ้น สามารถนำมอนสเตอร์ไปต่อสู้กับผู้เล่นคนอื่นๆ ได้ ในระบบลานประลองซึ่งจะได้รับของรางวัลอย่างเช่น บิสกิต เป็นต้น และหลังจากตอบคำถามต่างๆ แล้วนักเรียนสามารถเข้าไปดูคะแนนย้อนหลังของตนเองได้ โดยคะแนนของคำถามจะถูกแยกไว้เป็นชุดตามชุดคำถามที่ได้ทำ

3.2 การวิเคราะห์ระบบ

3.2.1 การเก็บรวบรวมความต้องการของระบบ

จากการศึกษา และวิเคราะห์โปรแกรมดังกล่าวในบทที่ 2 และการออกแบบเกมที่กล่าวมาแล้วข้างต้น ผู้จัดทำได้ทำการรวบรวมความต้องการของระบบซึ่งมีรายละเอียดดังนี้

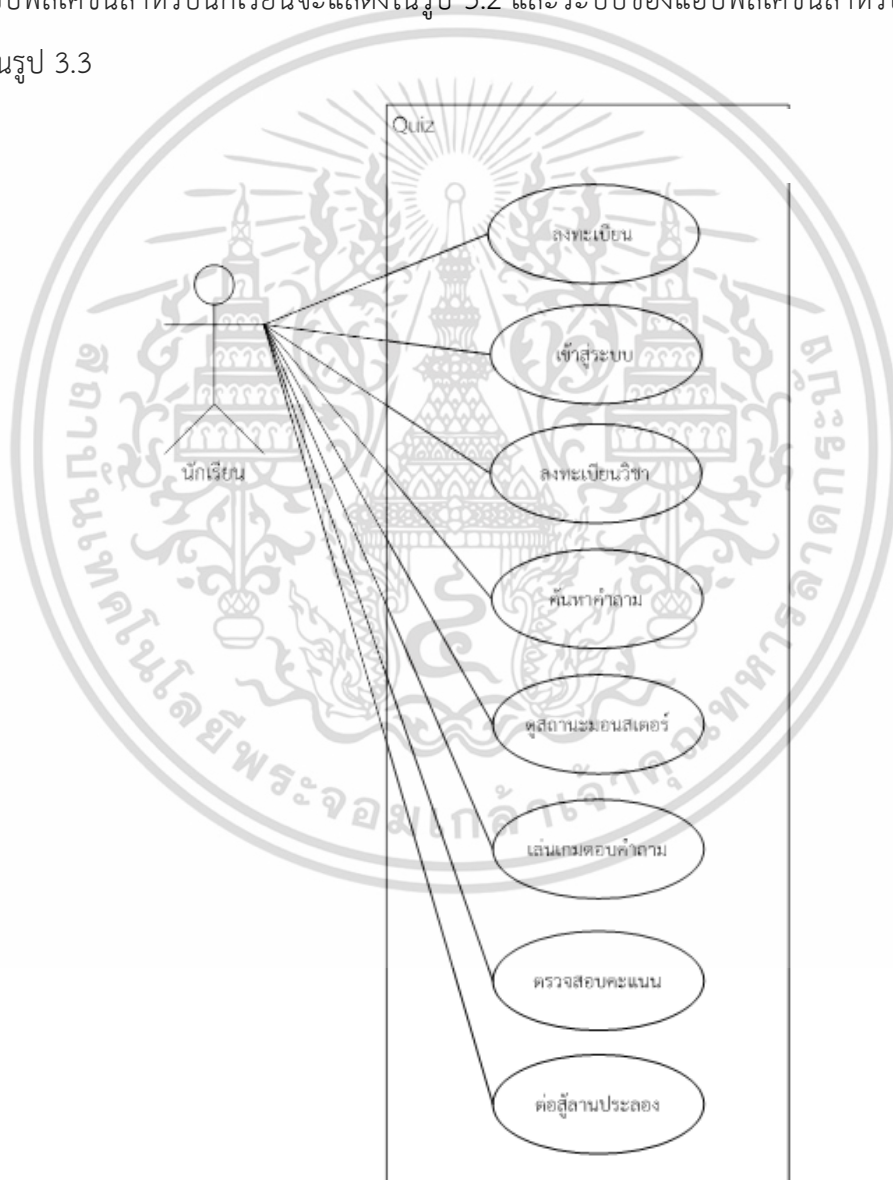
1. เป็นแอปพลิเคชันที่แบ่งออกเป็นสองส่วนอย่างชัดเจนตามการทำงาน คือ ส่วนของอาจารย์ที่ใช้ควบคุม และส่วนของนักเรียนที่ใช้เล่น
2. อาจารย์สามารถสร้างรายวิชาเพื่อให้นักเรียนลงทะเบียนได้
3. อาจารย์สามารถสร้าง และแก้ไขคำถามได้
4. อาจารย์สามารถกำหนดตำแหน่งของคำถามได้
5. นักเรียนจะเห็นคำถามเฉพาะวิชาที่ลงทะเบียนไว้
6. นักเรียนสามารถดูคะแนนย้อนหลังได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. เมื่อตอบคำถามถูกต้องจะต้องจะมีของตอบแทนเพื่อเพิ่มความสามารถให้กับมอนสเตอร์ของผู้เล่น
8. มีระบบลานประลองเพื่อให้นักเรียนสามารถนำมอนสเตอร์ของตัวเองมาประลองกับมอนสเตอร์ของคนอื่นได้โดยการตอบคำถามสลับข้อกัน

3.2.2 แผนภาพยูสเคส

ในส่วนของการวิเคราะห์ระบบจะแสดงในรูปแบบของแผนภาพยูสเคส โดยระบบของแอปพลิเคชันสำหรับนักเรียนจะแสดงในรูป 3.2 และระบบของแอปพลิเคชันสำหรับอาจารย์จะแสดงในรูป 3.3



รูปที่ 3.2 แผนภาพยูสเคสส่วนของนักเรียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.2 เป็นการแสดงแผนภาพยูสเคสของแอปพลิเคชันเกมควิสควอสโกส่วนนักเรียน โดยรายละเอียดยูสเคสแสดงในตารางตามลำดับต่อไปนี้

ตารางที่ 3.1 ลงทะเบียน

หัวข้อ	คำอธิบาย
Use Case Name	ลงทะเบียน
Brief Description	ลงทะเบียนผู้ใช้งานใหม่
Actor	นักเรียน
Trigger	นักเรียนเลือกลงทะเบียน
Preconditions	-
Postconditions	เข้าสู่ระบบ
Flow of Events	<ol style="list-style-type: none"> 1. นักเรียนกรอกข้อมูล ได้แก่ ชื่อเข้าใช้งาน รหัสผ่านเข้าใช้งาน ชื่อ นามสกุล รหัสนักเรียน และอีเมล จากนั้นคลิกตกลง เพื่อลงทะเบียน 2. ระบบตรวจสอบชื่อเข้าใช้งานว่ามีอยู่ในฐานข้อมูลหรือไม่ <ol style="list-style-type: none"> 2.1 หากชื่อเข้าใช้งานซ้ำกับที่มีอยู่ในฐานข้อมูลแล้วก็จะไม่สามารถลงทะเบียนได้ 2.2 หากชื่อเข้าใช้งานไม่ซ้ำกับที่มีอยู่ในฐานข้อมูลแล้วก็จะสามารถลงทะเบียนได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.2 เข้าสู่ระบบ

หัวข้อ	คำอธิบาย
Use Case Name	เข้าสู่ระบบ
Brief Description	ลงชื่อเข้าสู่ระบบ
Actor	นักเรียน
Trigger	นักเรียนเลือกเข้าสู่ระบบ
Preconditions	-
Postconditions	สมาชิกในระบบ
Flow of Events	<ol style="list-style-type: none"> 1. นักเรียนกรอกข้อมูล ได้แก่ ชื่อเข้าใช้งาน และรหัสผ่าน ผู้ใช้ จากนั้นคลิกเข้าสู่ระบบเพื่อเข้าใช้งานในระบบ 2. ระบบตรวจสอบชื่อเข้าใช้งาน และรหัสผ่าน ว่าตรงกับในฐานข้อมูลหรือไม่ <ol style="list-style-type: none"> 2.1 หากชื่อเข้าใช้งาน และรหัสผ่านตรงกับที่มีอยู่ในฐานข้อมูล ก็จะสามารถเข้าสู่ระบบได้ 2.2 หากชื่อเข้าใช้งาน และรหัสผ่านไม่ตรงกับที่มีอยู่ในฐานข้อมูล ก็จะไม่สามารถเข้าสู่ระบบได้

ตารางที่ 3.3 ลงทะเบียนวิชา

หัวข้อ	คำอธิบาย
Use Case Name	ลงทะเบียนวิชา
Brief Description	ลงทะเบียนวิชา
Actor	นักเรียน
Trigger	นักเรียนเลือกลงทะเบียนวิชา
Preconditions	เข้าสู่ระบบ
Postconditions	ระบบแสดงรายชื่อวิชาที่ลงทะเบียน
Flow of Events	<ol style="list-style-type: none"> 1. นักเรียนกรอกรหัสวิชาที่ต้องการลงทะเบียน จากนั้นคลิกปุ่มเพื่อเพิ่มวิชา 2. ระบบเพิ่มวิชาลงในฐานข้อมูลของผู้ใช้ที่ลงทะเบียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.4 ค้นหาคำถาม

หัวข้อ	คำอธิบาย
Use Case Name	ค้นหาคำถาม
Brief Description	ค้นหาคำถามที่ลงทะเบียนวิชาไว้
Actor	นักเรียน
Trigger	นักเรียนกดปุ่มเพื่อค้นหาคำถาม
Preconditions	เข้าสู่ระบบ และลงทะเบียนวิชา
Postconditions	คำถามปรากฏ
Flow of Events	<ol style="list-style-type: none"> 1. นักเรียนกดปุ่มค้นหา เพื่อค้นหาคำถามในบริเวณรอบๆตัว 2. ระบบจะตรวจสอบว่าในบริเวณรอบๆตัวผู้ใช้ มีข้อมูลของคำถามในฐานข้อมูลหรือไม่ <ol style="list-style-type: none"> 2.1 หากพบข้อมูลคำถามในฐานข้อมูล อยู่บริเวณรอบๆตัวผู้ใช้ ระบบจะแสดงข้อมูลคำถามในรูปแบบตัวมอนสเตอร์ 2.2 หากไม่พบข้อมูลคำถามในฐานข้อมูล อยู่บริเวณรอบๆตัวผู้ใช้ ระบบจะไม่แสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.5 เล่นเกมตอบคำถาม

หัวข้อ	คำอธิบาย
Use Case Name	เล่นเกมตอบคำถาม
Brief Description	นักเรียนตอบคำถามที่แสดงขึ้นมา
Actor	นักเรียน
Trigger	นักเรียนกดเลือกคำถาม
Preconditions	ค้นหาคำถาม
Postconditions	ระบบแสดงของรางวัล
Flow of Events	<ol style="list-style-type: none"> 1. นักเรียนกดเลือกคำถามที่ต้องการเล่น 2. นักเรียนเล่นเกมตอบคำถาม โดยคลิกเลือกคำตอบจากคำถาม 3. ระบบจะตรวจสอบตัวเลือกที่นักเรียนเลือก <ol style="list-style-type: none"> 3.1 หากนักเรียนเลือกคำตอบถูก ระบบจะลดระดับพลังของมอนสเตอร์ และแสดงผลออกมา 3.2 หากนักเรียนเลือกคำตอบผิด ระบบจะลดระดับพลังของมอนสเตอร์ของนักเรียน และแสดงผลออกมา 3.3 หากนักเรียนไม่เลือกคำตอบจนหมดเวลา ระบบจะลดระดับพลังของมอนสเตอร์ของนักเรียน และแสดงผลออกมา 4. ระบบบันทึกรายละเอียดของการเล่นเกมตอบคำถาม ได้แก่ คะแนน คำถาม และคำตอบ ที่เล่นผ่านไป 5. ระบบแสดงของรางวัล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.6 ตรวจสอบสถานะมอนสเตอร์

หัวข้อ	คำอธิบาย
Use Case Name	ตรวจสอบสถานะมอนสเตอร์
Brief Description	ตรวจสอบสถานะมอนสเตอร์ และบิสกิตต่างๆ
Actor	นักเรียน
Trigger	นักเรียนกดเลือกตรวจสอบมอนสเตอร์
Preconditions	-
Postconditions	ระบบแสดงข้อมูลของมอนสเตอร์
Flow of Events	<ol style="list-style-type: none"> 1. นักเรียนเลือกบิสกิตที่ต้องการเพิ่มสถานะมอนสเตอร์ จากนั้นคลิกปุ่มเพื่อเพิ่มค่าสถานะของมอนสเตอร์ 2. ระบบตรวจสอบว่า นักเรียนมีข้อมูลบิสกิตในฐานข้อมูลหรือไม่ <ol style="list-style-type: none"> 2.1 หากพบข้อมูลบิสกิตในฐานข้อมูล ระบบจะเพิ่มสถานะมอนสเตอร์ 2.2 หากไม่พบข้อมูลบิสกิตในฐานข้อมูล ระบบจะไม่เพิ่มสถานะมอนสเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.7 ตรวจสอบคะแนน

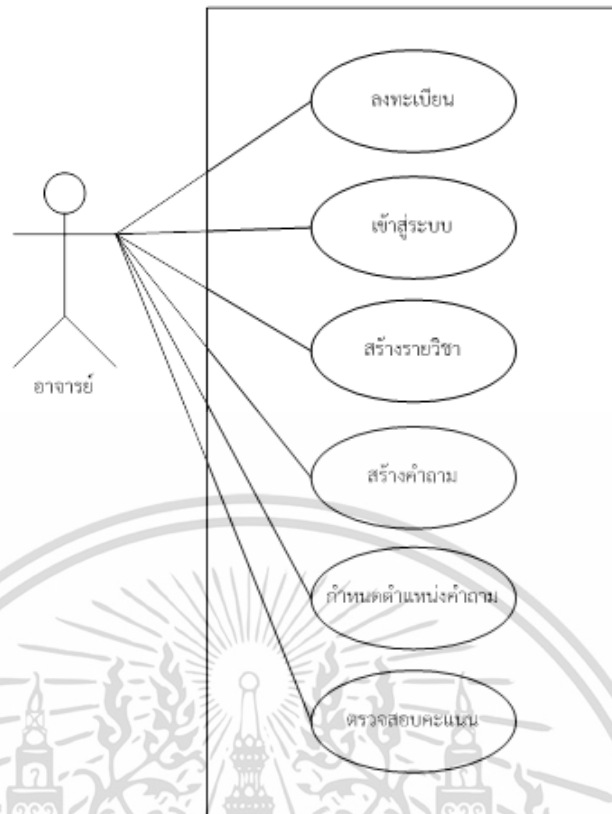
หัวข้อ	คำอธิบาย
Use Case Name	ตรวจสอบคะแนน
Brief Description	ตรวจสอบคะแนน คำตอบ และคำถามที่ได้ทำไปแล้ว
Actor	นักเรียน
Trigger	นักเรียนเลือกคะแนน หรือดูคำถาม และคำตอบ
Preconditions	-
Postconditions	ระบบแสดงข้อมูลคะแนน คำถาม และคำตอบ
Flow of Events	<ol style="list-style-type: none"> 1. นักเรียนเลือกรายวิชาที่ต้องการดูคะแนน จากนั้นคลิกที่ชื่อวิชาเพื่อดูคะแนน 2. ระบบแสดงคะแนนในแต่ละแบบทดสอบของวิชาที่เลือกไว้ 3. นักเรียนเลือกแบบทดสอบที่ต้องการจะดูคำถาม และคำตอบ จากนั้นคลิกชื่อแบบทดสอบเพื่อดูคำถาม และคำตอบ 4. ระบบจะแสดงคำถาม และคำตอบในของแบบทดสอบที่เลือกไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.8 ต่อสู้สถานประลอง

หัวข้อ	คำอธิบาย
Use Case Name	ต่อสู้สถานประลอง
Brief Description	นำมอนสเตอร์ของเจ้าของห้อง และผู้เข้าร่วมมาต่อสู้กัน โดย การตอบคำถามสลับข้อกัน
Actor	นักเรียน
Trigger	นักเรียนกดปุ่ม
Preconditions	-
Postconditions	ระบบจะแสดงผู้ชนะ และของรางวัล
Flow of Events	<ol style="list-style-type: none"> 1. นักเรียนกดปุ่มเพื่อเข้าสู่การต่อสู้ประลอง จากนั้นเลือกคลิกปุ่มสร้างห้อง หรือปุ่มเข้าร่วมการต่อสู้ 2. ระบบจะตรวจสอบว่านักเรียนเลือกฟังก์ชันไหน <ol style="list-style-type: none"> 2.1 หากเลือกสร้างห้อง ระบบจะยังไม่แสดงคำถาม และตัวเลือก จนกว่าจะมีผู้อื่นมาเข้าร่วมการต่อสู้ 2.2 หากเลือกเข้าร่วมการต่อสู้ ระบบจะแสดงคำถาม และคำตอบจากฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 แผนภาพยูสเคสส่วนของอาจารย์

จากรูปที่ 3.3 เป็นการแสดงแผนภาพยูสเคสของเว็บไซต์ควิสแควสโกส่วนของอาจารย์ โดยรายละเอียดยูสเคสแสดงในตารางตามลำดับต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.9 ลงทะเบียน

หัวข้อ	คำอธิบาย
Use Case Name	ลงทะเบียน
Brief Description	ลงทะเบียนผู้ใช้งานใหม่
Actor	อาจารย์
Trigger	อาจารย์เลือกลงทะเบียน
Preconditions	-
Postconditions	เข้าสู่ระบบ
Flow of Events	<ol style="list-style-type: none"> 1. อาจารย์กรอกข้อมูล ได้แก่ ชื่อเข้าใช้งาน รหัสผ่านเข้าใช้งาน ชื่อ นามสกุล และอีเมล จากนั้นคลิกตกลง เพื่อลงทะเบียน 2. ระบบตรวจสอบชื่อเข้าใช้งานว่ามีอยู่ในฐานข้อมูลหรือไม่ <ol style="list-style-type: none"> 2.1 หากชื่อเข้าใช้งานซ้ำกับที่มีอยู่ในฐานข้อมูลแล้วก็จะไม่สามารถลงทะเบียนได้ 2.2 หากชื่อเข้าใช้งานไม่ซ้ำกับที่มีอยู่ในฐานข้อมูลแล้วก็จะสามารถลงทะเบียนได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.10 เข้าสู่ระบบ

หัวข้อ	คำอธิบาย
Use Case Name	เข้าสู่ระบบ
Brief Description	ลงชื่อเข้าสู่ระบบ
Actor	อาจารย์
Trigger	อาจารย์เลือกเข้าสู่ระบบ
Preconditions	-
Postconditions	สมาชิกในระบบ
Flow of Events	<ol style="list-style-type: none"> 1. อาจารย์กรอกข้อมูล ได้แก่ ชื่อเข้าใช้งาน และรหัสผ่าน ผู้ใช้ จากนั้นคลิกเข้าสู่ระบบ เพื่อเข้าสู่ระบบ 2. ระบบตรวจสอบชื่อเข้าใช้งาน และรหัสผ่าน ว่าตรงกับในฐานข้อมูลหรือไม่ <ol style="list-style-type: none"> 2.1 หากชื่อเข้าใช้งาน และรหัสผ่านตรงกับที่มีอยู่ในฐานข้อมูล ก็จะสามารถเข้าสู่ระบบได้ 2.2 หากชื่อเข้าใช้งาน และรหัสผ่านไม่ตรงกับที่มีอยู่ในฐานข้อมูล ก็จะไม่สามารถเข้าสู่ระบบได้

ตารางที่ 3.11 สร้างรายวิชา

หัวข้อ	คำอธิบาย
Use Case Name	สร้างรายวิชา
Brief Description	สร้างรายวิชา
Actor	อาจารย์
Trigger	อาจารย์เลือกสร้างรายวิชา
Preconditions	-
Postconditions	ระบบแสดงรหัสวิชา และชื่อวิชา
Flow of Events	<ol style="list-style-type: none"> 1. อาจารย์กรอกชื่อวิชาที่ต้องการสร้าง จากนั้นคลิกปุ่มเพื่อสร้างวิชา 2. ระบบจะบันทึกลงฐานข้อมูล และแสดงชื่อวิชาที่สร้างไว้

ตารางที่ 3.12 สร้างคำถาม

หัวข้อ	คำอธิบาย
Use Case Name	สร้างคำถาม
Brief Description	สร้างคำถาม และคำตอบ
Actor	อาจารย์
Trigger	อาจารย์เลือกสร้างคำถาม
Preconditions	-
Postconditions	ระบบจะแสดงชื่อคำถาม คำถาม และคำตอบ
Flow of Events	<ol style="list-style-type: none"> 1. อาจารย์กรอกรายละเอียดของคำถาม ได้แก่ ชื่อคำถาม คำถาม คำตอบที่ถูก และคำตอบที่ผิด จากนั้นคลิกปุ่มเพื่อสร้างคำถาม 2. ระบบจะบันทึกข้อมูลคำถามลงฐานข้อมูล และแสดงชื่อคำถาม คำถาม และคำตอบที่ถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.13 กำหนดตำแหน่งของคำถาม

หัวข้อ	คำอธิบาย
Use Case Name	กำหนดตำแหน่งของคำถาม
Brief Description	กำหนดตำแหน่งของคำถามที่สร้างไว้ ลงบนแผนที่
Actor	อาจารย์
Trigger	อาจารย์เลือกกำหนดตำแหน่งของคำถาม
Preconditions	สร้างคำถาม
Postconditions	ระบบแสดงตำแหน่งที่คำถามถูกกำหนดตำแหน่งลงบนแผนที่
Flow of Events	<ol style="list-style-type: none"> 1. อาจารย์เลือกวิชา และคำถามที่ต้องการจะกำหนดตำแหน่ง จากนั้นคลิกเลือกตำแหน่งที่ต้องการลงในแผนที่ 2. ระบบจะตรวจสอบว่ามีการกดปุ่มปุ่มกำหนดตำแหน่งของคำถามใหม่ <ol style="list-style-type: none"> 2.1 หากมีการกดปุ่มปุ่มกำหนดตำแหน่งของคำถาม ระบบจะซูมการกำหนดตำแหน่งของคำถามในรัศมี 20 เมตร และบันทึกคำถาม และตำแหน่งที่ซูมลงในฐานข้อมูลของแผนที่ 2.2 หากไม่มีการกดปุ่มปุ่มคำถาม ระบบจะบันทึก คำถาม และตำแหน่งที่เลือกไว้ลงในฐานข้อมูลของแผนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.14 ตรวจสอบคะแนน

หัวข้อ	คำอธิบาย
Use Case Name	ตรวจสอบคะแนน
Brief Description	ตรวจสอบคะแนนของนักเรียนในแต่ละชุดคำถามได้
Actor	อาจารย์
Trigger	อาจารย์เลือกดูคะแนน
Preconditions	-
Postconditions	ระบบจะแสดงรายละเอียดของคะแนน
Flow of Events	1.อาจารย์เลือกชื่อวิชา และชื่อชุดคำถามที่ต้องการจะดูคะแนน 2.ระบบจะตรวจสอบชื่อวิชา และชื่อชุดคำถามตรงกับฐานข้อมูลตัวไหน ก็จะแสดงรายละเอียดของคะแนนออกมาจากฐานข้อมูล

3.3 การออกแบบระบบ

ในส่วนของการออกแบบ ได้นำแผนภาพยูสเคสจากหัวข้อ 3.2.2 มาออกแบบตารางข้อมูลโดยจะแสดงในรูปแบบแผนภาพอี-อาร์ ความสัมพันธ์ระหว่างคลาสด้วยแผนภาพคลาส และออกแบบส่วนติดต่อผู้ใช้งานดังต่อไปนี้

3.3.1 การออกแบบแอปพลิเคชันฝั่งนักเรียน

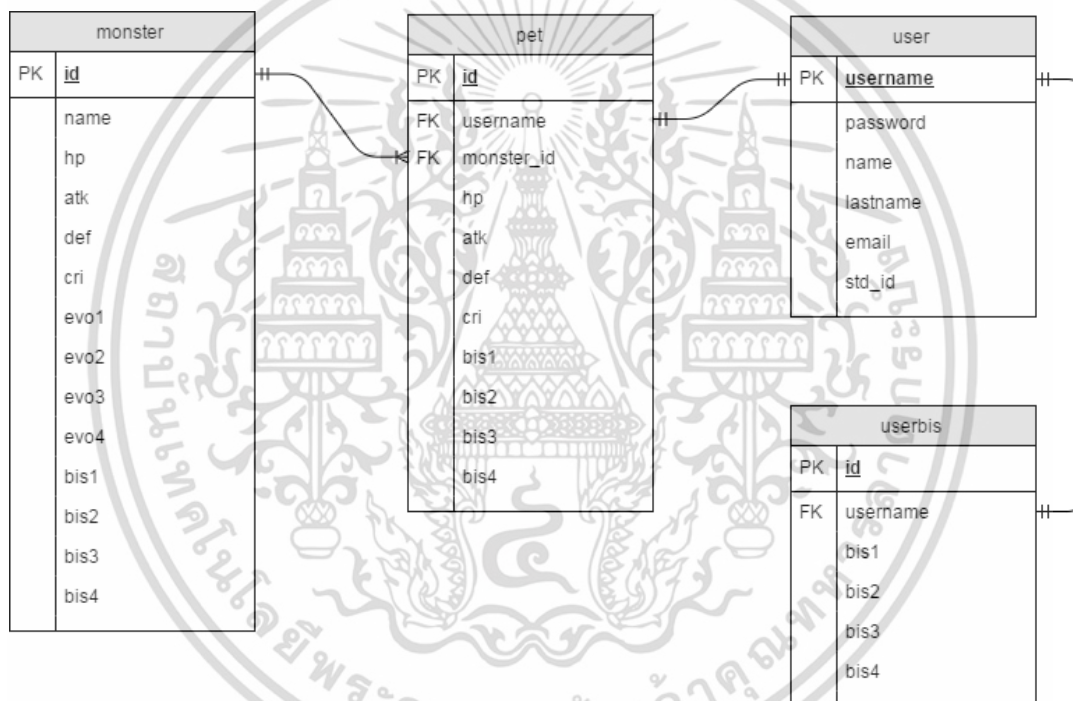
จากการวิเคราะห์แผนภาพยูสเคสในหัวข้อ 3.2.2 จะประกอบไปด้วยความสามารถหลักของแอปพลิเคชัน ซึ่งสามารถออกแบบโดยแยกตามหัวข้อดังนี้

3.3.1.1 การออกแบบตามยูสเคสการลงทะเบียน

จากการวิเคราะห์แผนภาพยูสเคสการลงทะเบียน ผู้เรียนต้องลงทะเบียนก่อนเพื่อเข้าใช้งานแอปพลิเคชัน จึงได้ออกแบบตารางข้อมูล ออกแบบแผนภาพคลาส และออกแบบส่วนติดต่อผู้ใช้งาน ดังต่อไปนี้

1) การออกแบบตารางข้อมูลของการลงทะเบียน

ในการลงทะเบียนเข้าสู่ระบบ การจัดการข้อมูลผู้ใช้ และการลงทะเบียนมอนสเตอร์ประจำตัว ในระบบเกมควิสควิสโก ดังที่ได้กล่าวในหัวข้อที่ 3.2.2 จำเป็นต้องเก็บข้อมูลส่วนบุคคลเช่น ชื่อ นามสกุล รหัสนักเรียน และข้อมูลที่ใช้ในการอ้างอิงข้อมูลส่วนบุคคล ได้แก่ ชื่อผู้ใช้ และรหัสผ่าน รวมถึงข้อมูลที่ใช้เล่นภายในเกมเช่น ข้อมูลมอนสเตอร์ จำนวนบิสิตของผู้เล่น จึงได้ออกแบบตารางข้อมูล user, userbis, pet และ monster ซึ่งสามารถเก็บข้อมูลผู้ใช้งาน เก็บจำนวนบิสิตของผู้ใช้งาน เก็บข้อมูลมอนสเตอร์ เก็บข้อมูลมอนสเตอร์ประจำตัวผู้ใช้งาน ดังตารางที่ 3.15, 3.16, 3.17 และ 3.18 โดยแต่ละตารางมีความสัมพันธ์ดังรูปที่ 3.4



รูปที่ 3.4 ความสัมพันธ์ของตารางการลงทะเบียน

ตารางที่ 3.15 แสดงรายละเอียดของข้อมูลในตารางข้อมูล user

คำอธิบายตาราง ตารางข้อมูล user เป็นตารางที่ใช้เก็บข้อมูลของผู้ใช้ฝั่งนักเรียน

User			
ชื่อคอลัมน์	ประเภทข้อมูล	ความหมาย	คีย์
username	varchar(40)	ชื่อเข้าใช้งานผู้ใช้ฝั่งนักเรียน	Primary Key
password	varchar(11)	รหัสผ่านผู้ใช้ฝั่งนักเรียน	
name	varchar(40)	ชื่อผู้ใช้ฝั่งนักเรียน	
lastname	varchar(45)	นามสกุลผู้ใช้ฝั่งนักเรียน	
email	varchar(45)	อีเมลผู้ใช้ฝั่งนักเรียน	
std_id	int(10)	รหัสประจำตัวนักเรียน	

ตารางที่ 3.16 แสดงรายละเอียดของข้อมูลในตารางข้อมูล userbis

คำอธิบายตาราง ตารางข้อมูล userbis เป็นตารางที่ใช้เก็บข้อมูลผู้ใช้ฝั่งนักเรียนเมื่อใช้บิสิตรูปแบบต่างๆ

Userbis			
ชื่อคอลัมน์	ประเภทข้อมูล	ความหมาย	คีย์
Id	Int(10)	รหัสผู้ใช้ฝั่งนักเรียนเมื่อใช้บิสิต	Primary Key
Username	Varchar(40)	ชื่อผู้ใช้ฝั่งนักเรียน	Foreign Key
Bis1	Int(3)	บิสิตรูปแบบที่ 1	
Bis2	Int(3)	บิสิตรูปแบบที่ 2	
Bis3	Int(3)	บิสิตรูปแบบที่ 3	
Bis4	Int(3)	บิสิตรูปแบบที่ 4	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.17 แสดงรายละเอียดของข้อมูลในตารางข้อมูล pet

คำอธิบายตาราง ตารางข้อมูล pet เป็นตารางที่ใช้เก็บข้อมูลมอนสเตอร์ประจำตัวผู้ใช้

Pet			
ชื่อคอลัมน์	ประเภทข้อมูล	ความหมาย	คีย์
Id	int(10)	รหัสมอนสเตอร์ประจำตัวผู้ใช้	Primary Key
username	varchar(45)	ชื่อผู้ใช้ฝั่งนักเรียน	Foreign Key
monster_id	int(10)	รหัสมอนสเตอร์	Foreign Key
hp	int(4)	ระดับพลังของมอนสเตอร์	
atk	int(3)	ค่าโจมตีมอนสเตอร์	
def	int(3)	ค่าป้องกันมอนสเตอร์	
cri	int(3)	ค่าความแม่นยำมอนสเตอร์	
bis1	int(3)	บิสกิตรูปแบบที่ 1	
bis2	int(3)	บิสกิตรูปแบบที่ 2	
bis3	int(3)	บิสกิตรูปแบบที่ 3	
bis4	int(3)	บิสกิตรูปแบบที่ 4	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.18 แสดงรายละเอียดของข้อมูลในตารางข้อมูล monster

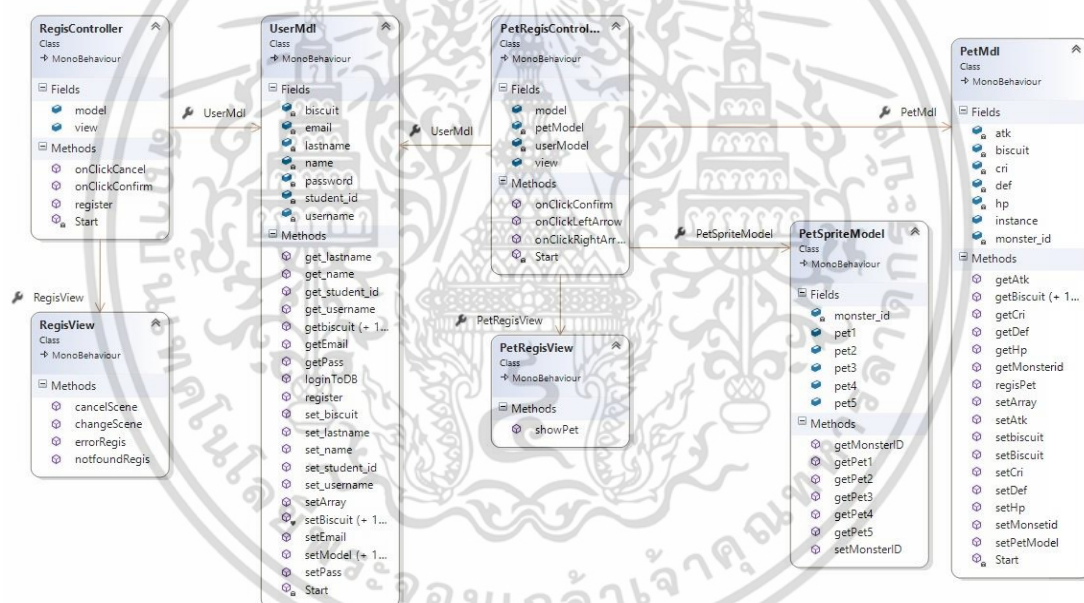
คำอธิบายตาราง ตารางข้อมูล monster เป็นตารางที่ใช้เก็บข้อมูลมอนสเตอร์

Monster			
ชื่อคอลัมน์	ประเภทข้อมูล	ความหมาย	คีย์
id	int(10)	รหัสมอนสเตอร์	Primary Key
name	varchar(40)	ชื่อมอนสเตอร์	
hp	int(4)	ระดับพลังของมอนสเตอร์	
atk	int(3)	ค่าโจมตีมอนสเตอร์	
def	int(3)	ค่าป้องกันมอนสเตอร์	
cri	int(3)	ค่าความแม่นยำมอนสเตอร์	
evo1	int(10)	วิวัฒนาการมอนสเตอร์รูปแบบที่ 1	
evo2	int(10)	วิวัฒนาการมอนสเตอร์รูปแบบที่ 2	
evo3	int(10)	วิวัฒนาการมอนสเตอร์รูปแบบที่ 3	
evo4	int(10)	วิวัฒนาการมอนสเตอร์รูปแบบที่ 4	
bis1	int(3)	บิสกิตรูปแบบที่ 1	
bis2	int(3)	บิสกิตรูปแบบที่ 2	
bis3	int(3)	บิสกิตรูปแบบที่ 3	
bis4	int(3)	บิสกิตรูปแบบที่ 4	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) การออกแบบแผนภาพคลาสของการลงทะเบียน

จากการออกแบบตารางข้อมูลของการลงทะเบียน สามารถนำมาออกแบบเป็นแผนภาพคลาสของการลงทะเบียน เพื่อแสดงความสัมพันธ์ระหว่างออบเจ็กต์ของคลาสที่เก็บข้อมูลจากตารางข้อมูล คลาสเพื่อการแสดงผล และคลาสสำหรับจัดการการทำงานของลงทะเบียน เพื่อใช้ในการพัฒนาแอปพลิเคชัน จึงได้ออกแบบคลาส RegisController , RegisView , PetRegisController , PetSpriteModel , PetRegisView , UserMdl และ PetMdl ซึ่งใช้ในการจัดการการทำงานหน้าลงทะเบียน แสดงผลหน้าลงทะเบียน จัดการการทำงานหน้าลงทะเบียนสัตว์เลี้ยง จัดเก็บรูปสัตว์เลี้ยง แสดงผลหน้าลงทะเบียนสัตว์เลี้ยง จัดเก็บ และจัดการข้อมูลผู้ใช้งานฐานข้อมูล และจัดเก็บ และจัดการข้อมูลสัตว์เลี้ยงกับฐานข้อมูล โดยจะแสดงรายละเอียดดังตารางที่ 3.19 , 3.20 , 3.21 , 3.22 , 3.23 และ 3.24 โดยแต่ละคลาสมีความสัมพันธ์กันดังรูปที่ 3.5



รูปที่ 3.5 แผนภาพคลาสของการลงทะเบียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.19 แสดงรายละเอียดของข้อมูลในคลาส RegisController

คำอธิบายคลาส คลาส RegisController เป็นคลาสที่ใช้เกี่ยวกับการควบคุมการทำงานการลงทะเบียน

RegisController		
ชื่อแอททริบิวต์	ประเภทข้อมูล	ความหมาย
View	RegisView	ออบเจ็กต์เกี่ยวกับการแสดงผลของหน้าลงทะเบียน
model	UserMdl	ออบเจ็กต์สำหรับเก็บ เรียกใช้ และจัดการข้อมูลของผู้เล่น

ชื่อคำสั่ง	พารามิเตอร์	หน้าที่ / การทำงาน
onClickCancel		คำสั่งที่ทำงานเมื่อมีการกดปุ่มยกเลิกบนหน้าจอ จะทำการเรียกใช้งานคำสั่ง cancelScene ของคลาส RegisView เพื่อเปลี่ยนหน้าจอไปหน้าจอเข้าสู่ระบบ
onClickConfirm		คำสั่งที่ทำงานเมื่อมีการกดปุ่มยืนยันบนหน้าจอ จะทำการเรียกคำสั่ง register ขึ้นมาใช้งาน
register		คำสั่งที่นำข้อมูลที่ผู้ใช้ป้อนเพื่อใช้สมัครสมาชิก เก็บไว้ในออบเจ็กต์ model และเรียกคำสั่ง register ของ model เพื่อนำข้อมูลที่เก็บไว้ใน model เพิ่มลงฐานข้อมูล หากไม่มีข้อมูลเข้าภายในฐานข้อมูล จะทำการย้ายหน้าจอไปเมนูเลือกสตั๊วเลี้ยง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.20 แสดงรายละเอียดของข้อมูลในคลาส RegisView

คำอธิบายคลาส คลาส RegisView เป็นคลาสที่ใช้เกี่ยวกับการแสดงผลหน้าจอของหน้าลงทะเบียน

RegisView		
ชื่อคำสั่ง	พารามิเตอร์	หน้าที่ / การทำงาน
cancelScene		คำสั่งที่ถูกเรียกหลังมีการกดปุ่มยกเลิก จะทำการเปลี่ยนหน้าจอไปหน้าจอเข้าสู่ระบบ
changeScene		คำสั่งที่ถูกเรียกเมื่อทำการลงทะเบียน และไม่เกิดปัญหาชื่อผู้ใช้งานมีอยู่ในฐานข้อมูลอยู่แล้ว จะทำการเปลี่ยนหน้าจอไปหน้าจอเลือกสัตว์เลี้ยง
errorRegis		คำสั่งสำหรับแสดงคำแจ้งเตือนเมื่อชื่อผู้ใช้งานมีอยู่ในฐานข้อมูลอยู่แล้ว
notfoundRegis		คำสั่งสำหรับแสดงคำแจ้งเตือนเมื่อผู้ใช้งาน ป้อนข้อมูลสำหรับการลงทะเบียนไม่ครบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.21 แสดงรายละเอียดของข้อมูลในคลาส PetRegisController

คำอธิบายคลาส คลาส PetRegisController เป็นตารางที่ใช้เกี่ยวกับการควบคุมการลงทะเบียนมอนสเตอร์ประจำตัว

PetRegisController		
ชื่อแอตทริบิวต์	ประเภทข้อมูล	หน้าที่ / การทำงาน
model	PetSpiritModel	ออบเจ็กต์สำหรับเก็บที่ภาพใช้แสดงของมอนสเตอร์
petModel	PetMdl	ออบเจ็กต์สำหรับเก็บ เรียกใช้ และจัดการข้อมูลสัตว์เลี้ยงของผู้ใช้
userModel	UserMdl	ออบเจ็กต์สำหรับเก็บ เรียกใช้ และจัดการข้อมูลของผู้เล่น
View	PetRegisView	ออบเจ็กต์เกี่ยวกับการแสดงผลหน้าลงทะเบียนมอนสเตอร์

ชื่อคำสั่ง	พารามิเตอร์	ความหมาย
onClickConfirm		คำสั่งที่ทำงานเมื่อมีการกดปุ่มยืนยันบนหน้าจอ จะทำการเรียกใช้งานคำสั่ง regisPet ของคลาส PetModel เพื่อเพิ่มข้อมูลสัตว์เลี้ยงที่ผู้ใช้งานเลือกลงในฐานข้อมูล
onClickLeftArrow		คำสั่งที่ทำงานเมื่อมีการกดปุ่มลูกศรซ้าย จะทำการแสดงรูปของสัตว์เลี้ยงที่มีรหัสก่อนหน้าสัตว์เลี้ยงตัวปัจจุบัน
onClickRightArrow		คำสั่งที่ทำงานเมื่อมีการกดปุ่มลูกศรขวา จะทำการแสดงรูปของสัตว์เลี้ยงที่มีรหัสถัดจากสัตว์เลี้ยงตัวปัจจุบัน

ตารางที่ 3.22 แสดงรายละเอียดของข้อมูลในคลาส PetSpriteModel

คำอธิบายคลาส คลาส PetSpriteModel เป็นตารางที่ใช้เก็บภาพที่ใช้แสดงของสัตว์เลี้ยง ซึ่งใช้ภายในโปรแกรมยูนิตี เพื่อแสดงรูปภาพของสัตว์เลี้ยงที่แตกต่างกันไป

PetSpriteModel		
ชื่อแอตทริบิวต์	ประเภทข้อมูล	ความหมาย
Monster_id	Int	รหัสของสัตว์เลี้ยงหรือมอนสเตอร์
pet1	sprite	เก็บรูปภาพสัตว์เลี้ยงแบบที่1
pet2	Sprite	เก็บรูปภาพสัตว์เลี้ยงแบบที่2
pet3	Sprite	เก็บรูปภาพสัตว์เลี้ยงแบบที่3
pet4	Sprite	เก็บรูปภาพสัตว์เลี้ยงแบบที่4
pet5	Sprite	เก็บรูปภาพสัตว์เลี้ยงแบบที่5

ตารางที่ 3.23 แสดงรายละเอียดของข้อมูลในคลาส PetRegisView

คำอธิบายคลาส คลาส PetRegisView เป็นคลาสที่ใช้เกี่ยวกับการแสดงผลหน้าจอของหน้าต่างทะเบียนสัตว์เลี้ยง

PetRegisView		
ชื่อคำสั่ง	พารามิเตอร์	หน้าที่ / การทำงาน
showPet	Sprite mon	แสดงภาพของสัตว์เลี้ยงในหน้าต่างทะเบียนสัตว์เลี้ยงตามข้อมูลที่ส่งมาทางพารามิเตอร์

ตารางที่ 3.24 แสดงรายละเอียดของข้อมูลในคลาส UserMdl

คำอธิบายคลาส คลาส UserMdl เป็นคลาสที่ใช้เก็บข้อมูลของผู้ใช้ ซึ่งออกแบบให้สัมพันธ์กับตาราง user และ ตาราง userbis โดยคลาสนี้จะออกแบบไว้สำหรับใช้ทั้งสมัครสมาชิก และรับข้อมูลผู้ใช้เมื่อเข้าสู่ระบบสำเร็จ จึงได้ออกแบบให้มีการเก็บข้อมูลซึ่งประกอบไปด้วยชื่อเข้าใช้ รหัส ชื่อ นามสกุล รหัสนักศึกษา อีเมลล์ และจำนวนบิสกิต

UserMdl		
ชื่อแอตทริบิวต์	ประเภทข้อมูล	ความหมาย
Biscuit	int[]	จำนวนบิสกิตแต่ละประเภท
Lastname	String	นามสกุลของผู้ใช้
Name	String	ชื่อของผู้ใช้
student_id	String	รหัสนักศึกษาของผู้ใช้
username	String	ชื่อเข้าใช้งาน
password	String	รหัสผ่าน
Email	String	อีเมลล์ของผู้ใช้

ชื่อคำสั่ง	พารามิเตอร์	หน้าที่ / การทำงาน
setModel		คำสั่งที่ถูกเรียกโดยคำสั่ง loginToDB ใช้ในการขอข้อมูลชื่อ นามสกุล รหัสนักศึกษา และจำนวนบิสกิตจากฐานข้อมูลโดยใช้ชื่อผู้ใช้อ้างอิง และเก็บลงในตัวแปร
loginToDB	String username, string passWord	คำสั่งสำหรับนำค่าชื่อผู้ใช้ และรหัสผ่าน ตรวจสอบกับฐานข้อมูลว่ามีข้อมูลอยู่หรือไม่หากมีจะเรียกใช้คำสั่ง loginToDB เพื่อขอข้อมูลผู้ใช้งานจากฐานข้อมูล หากไม่พบ จะเรียกคำสั่ง changeScene ของคลาส LoginView เพื่อแสดงข้อความผิดพลาด
register		คำสั่งสำหรับเพิ่มข้อมูลผู้ใช้งานบนฐานข้อมูล โดยจะใช้ข้อมูลภายในคลาส หากเพิ่มข้อมูลลงในฐานข้อมูลสำเร็จจะเรียกใช้คำสั่ง changScene() ของคลาส RegisView เพื่อไปหน้าเลือกสแตว์เล็ขง แต่หากชื่อผู้ใช้ซ้ำกับที่มีในฐานข้อมูล จะเรียก errorRegis ของคลาส RegisView เพื่อขึ้นข้อความเตือนว่าลงทะเบียนไม่สำเร็จแทน

ตารางที่ 3.25 แสดงรายละเอียดของข้อมูลในคลาส PetMdl

คำอธิบายคลาส คลาส PetMdl เป็นคลาสที่ใช้เก็บข้อมูลของมอนสเตอร์ประจำตัว ซึ่งออกแบบให้สัมพันธ์กับตาราง userpet จึงได้ออกแบบให้มีการเก็บข้อมูลซึ่งประกอบไปด้วยรหัสมอนสเตอร์, ค่าพลังโจมตี, พลังป้องกัน, ความแม่นยำ และได้ออกแบบให้เก็บค่าบิสกิตแต่ละประเภทที่มอนสเตอร์ประจำตัวกินให้เป็นอาเรย์เพื่อให้ง่ายต่อการเรียกใช้

PetMdl		
ชื่อแอตทริบิวต์	ประเภทข้อมูล	ความหมาย
Atk	int	ค่าพลังโจมตี
Biscuit	int[]	จำนวนบิสกิตแต่ละประเภทที่มอนสเตอร์ประจำตัวกิน
Cri	int	ค่าความแม่นยำ
Def	int	ค่าพลังป้องกัน
Hp	int	ค่าพลังชีวิต
monster_id	String	รหัสมอนสเตอร์ประจำตัว

ชื่อคำสั่ง	พารามิเตอร์	หน้าที่ / การทำงาน
regisPet	int monsterid	คำสั่งสำหรับนำค่ารหัสมอนสเตอร์ที่รับมา และชื่อผู้ใช้ เพิ่มเป็นข้อมูลส่วนตัวของผู้ใช้บนฐานข้อมูล
setPetModel		คำสั่งสำหรับขอข้อมูลส่วนตัวของผู้ใช้จากฐานข้อมูล โดยใช้ชื่อผู้ใช้ในการอ้างอิง และเก็บลงในตัวแปร

3) การออกแบบส่วนติดต่อผู้ใช้ของการลงทะเบียน

เมื่อผู้ใช้งานเข้าสู่แอปพลิเคชันเกมควิสควิสโก ได้ออกแบบให้หน้าจอแรกมีปุ่มเพื่อใช้งานสำหรับ 2 ฟังก์ชัน คือ ปุ่มเข้าสู่ระบบ (Login) สำหรับเข้าสู่ระบบของผู้ใช้งานที่ลงทะเบียนกับแอปพลิเคชันแล้ว และปุ่มลงทะเบียน (Register) ในการลงทะเบียนสำหรับผู้ใช้งานใหม่

รูปที่ 3.6 หน้าจอการลงทะเบียน

จากรูปที่ 3.6 เป็นการออกแบบหน้าจอให้ผู้ใช้งานใหม่ป้อนข้อมูลเพื่อใช้ลงทะเบียน โดยข้อมูลที่ใช้ในการลงทะเบียนเป็นข้อมูลทั่วไปของนักศึกษาที่มีทุกคน ซึ่งประกอบไปด้วย ชื่อเข้าใช้งานรหัสผ่าน ชื่อ นามสกุล รหัสนักศึกษา และอีเมล และได้ออกแบบให้สามารถยกเลิกหรือยืนยันการลงทะเบียนได้



รูปที่ 3.7 หน้าจอลงทะเบียนมอนสเตอร์ประจำตัว

จากรูปที่ 3.7 เป็นการออกแบบหน้าจอให้ผู้เลือกใช้มอนสเตอร์ประจำตัว เนื่องจากในเกมควิสเคสโก ผู้ใช้จะต้องมีมอนสเตอร์ประจำตัวอยู่หนึ่งตัว จึงได้ออกแบบให้มีการแสดงภาพของมอนสเตอร์เพื่อให้ผู้ใช้เลือกมอนสเตอร์ประจำตัวได้ ซึ่งมอนสเตอร์ที่ให้เลือกมีอยู่หลายชนิด ดังนั้นเพื่อให้ผู้ใช้สามารถเปลี่ยนมอนสเตอร์ที่จะเลือกได้ง่าย จึงได้ออกแบบให้มีปุ่มซ้าย และขวาเพื่อเปลี่ยนมอนสเตอร์

3.3.1.2 การออกแบบสำหรับยูสเคสการเข้าสู่ระบบ

จากการวิเคราะห์แผนภาพยูสเคสการเข้าสู่ระบบ หลังจากที่ผู้เรียนลงทะเบียนแล้ว ผู้เรียนต้องเข้าสู่ระบบเพื่อเข้าใช้งานแอปพลิเคชัน จึงได้ออกแบบตารางข้อมูลออกแบบแผนภาพคลาส และออกแบบส่วนติดต่อผู้ใช้ ดังต่อไปนี้

1) การออกแบบตารางข้อมูลของการเข้าสู่ระบบ

ในการเข้าสู่ระบบเกมควิสเคสโก ดังที่กล่าวในหัวข้อที่ 3.2.2 ระบบจะนำข้อมูลที่ผู้ใช้ป้อนเข้ามา นำไปตรวจสอบในฐานข้อมูลว่าข้อมูลตรงกันหรือไม่ ซึ่งใช้ชื่อผู้ใช้งาน และรหัสผ่านในการตรวจสอบ และเมื่อตรวจสอบเสร็จ และเข้าสู่ระบบแล้วจะทำการใช้ข้อมูลต่างๆเช่น มอนสเตอร์ จำนวนบิสกิต โดยอ้างอิงจากชื่อผู้ใช้ จึงได้ออกแบบตาราง user,

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

userbis, pet และ monster เพื่อตรวจสอบข้อมูลดังกล่าว โดยมีการแสดงรายละเอียดแต่ละตาราง ดังตารางที่ 3.15, 3.16, 3.17 และ 3.18 ที่กล่าวไว้ข้างต้น โดยแต่ละตารางมีความสัมพันธ์ดังรูปที่ 3.4 ที่กล่าวไว้ข้างต้น

โครงสร้างของตาราง user สามารถดูรายละเอียดได้ในตารางที่ 3.15

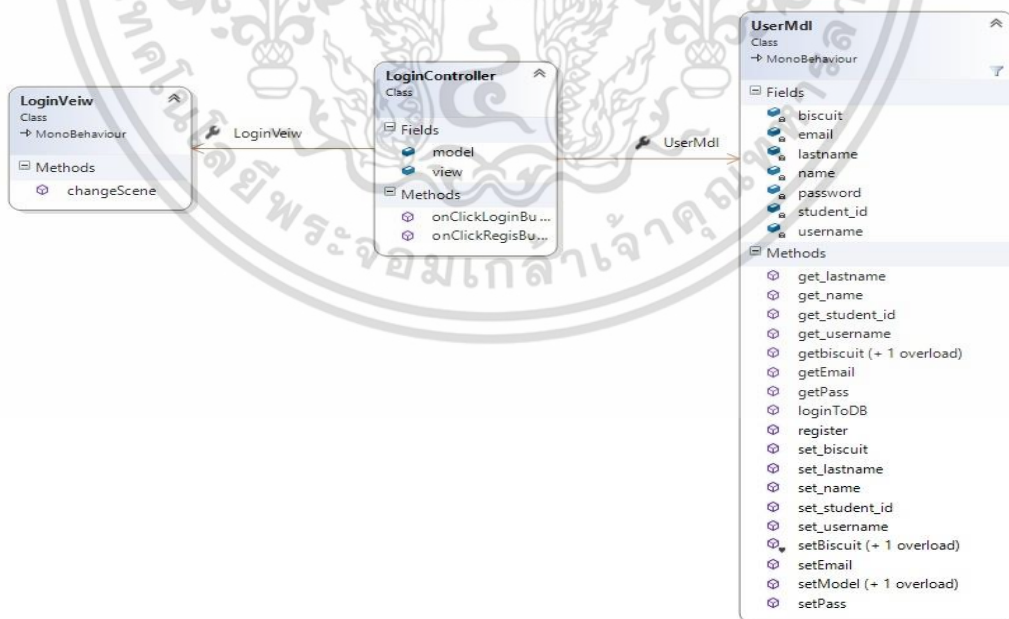
โครงสร้างของตาราง userbis สามารถดูรายละเอียดได้ในตารางที่ 3.16

โครงสร้างของตาราง pet สามารถดูรายละเอียดได้ในตารางที่ 3.17

โครงสร้างของตาราง monster สามารถดูรายละเอียดได้ในตารางที่ 3.18

2) การออกแบบแผนภาพคลาสของการเข้าสู่ระบบ

จากการออกแบบตารางข้อมูลของการเข้าสู่ระบบ สามารถนำมา ออกแบบเป็นแผนภาพคลาสของการเข้าสู่ระบบ เพื่อแสดงความสัมพันธ์ระหว่างออบเจ็กต์ของคลาสที่ เก็บข้อมูลจากตารางข้อมูลคลาสเพื่อการแสดงผล และคลาสสำหรับจัดการการทำงานของ การลงทะเบียน เพื่อใช้ในการพัฒนาแอปพลิเคชัน จึงได้ออกแบบคลาส UserMdl , LoginController และ LoginView ซึ่งใช้ในการจัดเก็บ และจัดการข้อมูลผู้ใช้งานฐานข้อมูล จัดการการทำงานของหน้า เข้าสู่ระบบ และแสดงผลหน้าเข้าสู่ระบบ โดยจะแสดงรายละเอียดดังตารางที่ 3.24 , 3.26 , 3.27 โดย แต่ละคลาสมีความสัมพันธ์กันดังรูปที่ 3.8



รูปที่ 3.8 แผนภาพคลาสของการเข้าสู่ระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.26 แสดงรายละเอียดของข้อมูลในคลาส LoginController

คำอธิบายคลาส คลาส LoginController เป็นคลาสที่ใช้เกี่ยวกับการควบคุมการทำงานการเข้าสู่ระบบ

LoginController		
ชื่อแอตทริบิวต์	ประเภทข้อมูล	ความหมาย
view	LoginView	ออบเจกต์เกี่ยวกับการแสดงผลของหน้าเข้าสู่ระบบ
model	UserMdl	ออบเจกต์สำหรับเก็บ เรียกใช้ และจัดการข้อมูลของผู้เล่น

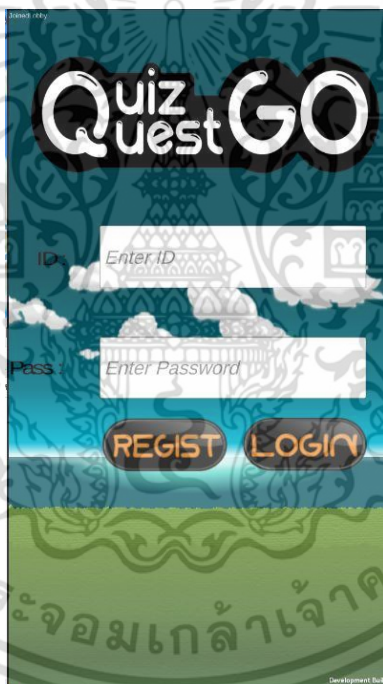
ชื่อคำสั่ง	พารามิเตอร์	หน้าที่ / การทำงาน
onClickLoginButton		คำสั่งที่ถูกเรียกใช้เมื่อผู้ใช้กดปุ่มเข้าสู่ระบบ โดยเมททอดนี้จะไปเรียกคำสั่ง loginToDB ของคลาส UserMdl เพื่อตรวจสอบว่าชื่อผู้ใช้ และรหัสผ่านตรงกับที่มีอยู่ในฐานข้อมูลหรือไม่ หากพบให้เปลี่ยนไปหน้าจอเมนูหลัก หากไม่พบจะเรียกคำสั่ง changeScene ของคลาส LoginView เพื่อแสดงข้อความผิดพลาด
onClickRegisButton		คำสั่งที่จะถูกเรียกใช้เมื่อผู้ใช้คลิกปุ่มลงทะเบียน ใช้สำหรับเปลี่ยนหน้าเข้าสู่ระบบเป็นหน้าลงทะเบียน

ตารางที่ 3.27 แสดงรายละเอียดของข้อมูลในคลาส LoginView

คำอธิบายคลาส คลาส LoginView เป็นคลาสที่ใช้เกี่ยวกับการควบคุมการทำงานการเข้าสู่ระบบ

LoginView		
ชื่อคำสั่ง	พารามิเตอร์	ความหมาย
changeScene	int i	เป็นคำสั่งที่จะรับค่าพารามิเตอร์เป็นเลขจำนวนเต็ม ซึ่งใช้บอกหน้าจอที่จะเปลี่ยนไป โดยถ้าพารามิเตอร์เป็น 0 จะแจ้งเตือนข้อผิดพลาดเมื่อเข้าสู่ระบบ หากเป็นเลขอื่นสำหรับเปลี่ยนหน้าจอไปตามหมายเลข

3) การออกแบบส่วนติดต่อผู้ใช้ของการเข้าสู่ระบบ



รูปที่ 3.9 หน้าจอการเข้าสู่ระบบ

จากรูปที่ 3.9 เป็นการออกแบบหน้าจอให้ผู้ป้อนข้อมูลเพื่อเข้าสู่ระบบของเกมควิสแควสโก ซึ่งข้อมูลที่ใช้เข้าสู่ระบบประกอบไปด้วยชื่อเข้าใช้งาน และรหัสผ่าน และได้มีการออกแบบให้ผู้ใช้งานใหม่ที่ยังไม่มีข้อมูลใช้เข้าสู่ระบบสามารถลงทะเบียนได้อีกด้วย

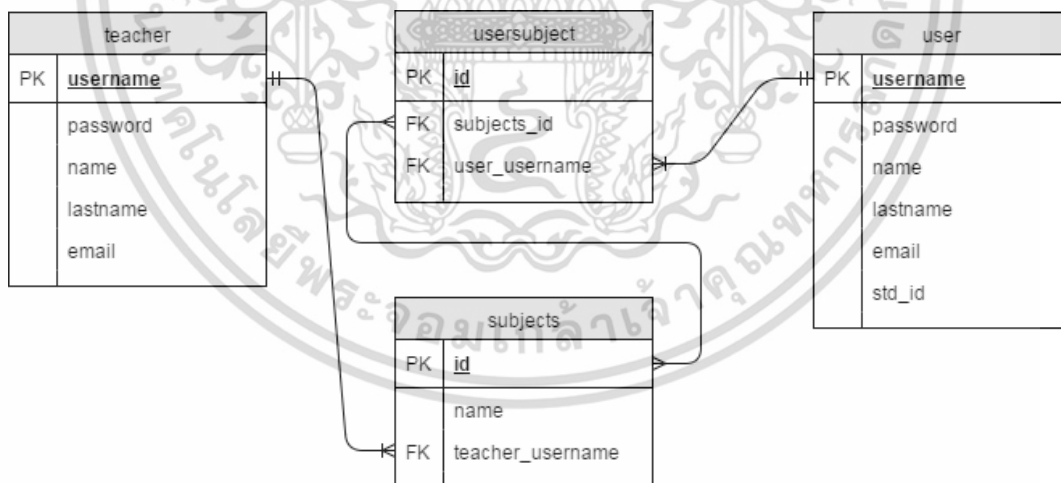
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.1.3 การออกแบบสำหรับยูสเคสการลงทะเบียนวิชา

จากการวิเคราะห์แผนภาพยูสเคสการลงทะเบียนวิชา หลังจากที่คุณเรียนเข้าสู่ระบบแล้ว ผู้เรียนต้องลงทะเบียนวิชาเพื่อที่จะเห็นคำถามในรายวิชาที่ลงทะเบียนไว้ได้ จึงได้ออกแบบตารางข้อมูล ออกแบบแผนภาพคลาส และออกแบบส่วนติดต่อผู้ใช้ ดังต่อไปนี้

1) การออกแบบตารางข้อมูลของการลงทะเบียนวิชา

ในการลงทะเบียนวิชาในระบบเกมควิสแควสโกล์ ดังที่กล่าวในหัวข้อที่ 3.2.2 จำเป็นต้องใช้รหัสวิชาในการอ้างอิงวิชาที่ต้องการตอบคำถาม ซึ่งวิชาแต่ละวิชานั้นจะถูกสร้างโดยอาจารย์ ทำให้ต้องเก็บข้อมูลส่วนบุคคลของอาจารย์ เพื่อใช้บอกว่าวิชานี้เป็นของอาจารย์คนใด และเมื่อลงทะเบียนแล้ว จำเป็นที่ต้องเก็บข้อมูลของการลงทะเบียนวิชาของนักเรียนแต่ละคนด้วย จึงทำให้การออกแบบตารางในหัวข้อการลงทะเบียนวิชาประกอบด้วยตารางข้อมูล user ,ตารางข้อมูล teacher, ตารางข้อมูล usersubject และตารางข้อมูล subjects ใช้เก็บข้อมูลในการลงทะเบียนวิชา โดยมีการแสดงรายละเอียดแต่ละตารางดังตารางที่ 3.15 ที่กล่าวไว้ข้างต้น และตารางที่ 3.28, 3.29 และ 3.30 โดยแต่ละตารางมีความสัมพันธ์ดังรูปที่ 3.10



รูปที่ 3.10 ความสัมพันธ์ของตารางการลงทะเบียนวิชา

ตารางที่ 3.28 แสดงรายละเอียดของข้อมูลในตารางข้อมูล usersubject

คำอธิบายตาราง ตารางข้อมูล usersubject เป็นตารางที่ใช้เก็บข้อมูลวิชาที่ผู้ใช้นักเรียนลงทะเบียนไว้แล้ว

Usersubject			
ชื่อคอลัมน์	ประเภทข้อมูล	ความหมาย	คีย์
Id	int(10)	รหัสวิชาที่ผู้ใช้นักเรียนลงทะเบียนไว้แล้ว	Primary Key
subjects_id	int(10)	รหัสวิชา	Foreign Key
user_username	varchar(45)	ชื่อเข้าใช้งานผู้ใช้นักเรียน	Foreign Key

ตารางที่ 3.29 แสดงรายละเอียดของข้อมูลในตารางข้อมูล subjects

คำอธิบายตาราง ตารางข้อมูล subjects เป็นตารางที่ใช้เก็บข้อมูลวิชา

Subjects			
ชื่อคอลัมน์	ประเภทข้อมูล	ความหมาย	คีย์
Id	int(10)	รหัสวิชา	Primary Key
Name	varchar(45)	ชื่อวิชา	

ตารางที่ 3.30 แสดงรายละเอียดของข้อมูลในตารางข้อมูล teacher

คำอธิบายตาราง ตารางข้อมูล teacher เป็นตารางที่ใช้เก็บข้อมูลของอาจารย์

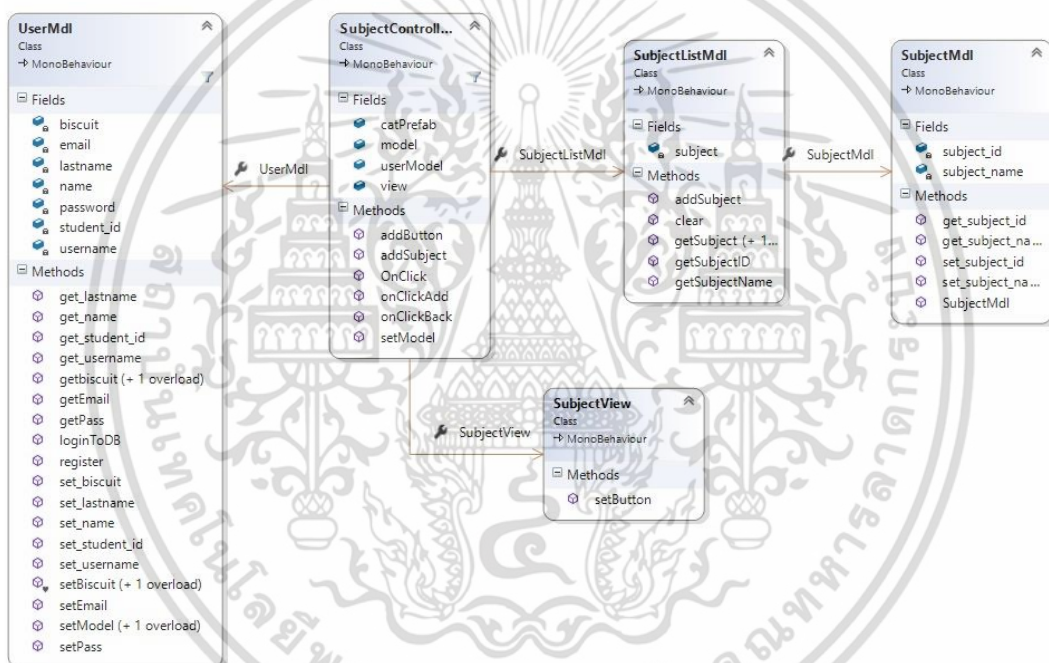
Teacher			
ชื่อคอลัมน์	ประเภทข้อมูล	ความหมาย	คีย์
Username	varchar(40)	ชื่อเข้าใช้งานผู้ใช้งานอาจารย์	Primary Key
Password	varchar(11)	รหัสผ่านผู้ใช้งานอาจารย์	
Name	varchar(40)	ชื่อผู้ใช้งานอาจารย์	
lastname	varchar(45)	นามสกุลผู้ใช้งานอาจารย์	
Email	varchar(45)	อีเมลผู้ใช้งานอาจารย์	

โครงสร้างของตาราง user สามารถดูรายละเอียดได้ในตารางที่ 3.15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) การออกแบบแผนภาพคลาสของการลงทะเบียนวิชา

จากการออกแบบตารางข้อมูลของการลงทะเบียนวิชา สามารถนำมาออกแบบเป็นแผนภาพคลาสของการลงทะเบียนข้อมูล เพื่อแสดงความสัมพันธ์ระหว่างออบเจกต์ของคลาสที่เก็บข้อมูลจากตารางข้อมูล คลาสเพื่อการแสดงผล และคลาสสำหรับจัดการการทำงานของ การลงทะเบียน เพื่อใช้ในการพัฒนาแอปพลิเคชัน จึงได้ออกแบบคลาส UserMdl, SubjectController, SubjectList, SubjectMdl, SubjectView ซึ่งใช้ในการจัดอ้างอิงชื่อผู้ใช้จัดการการทำงานในหน้าลงทะเบียนวิชา เก็บข้อมูลรายวิชาที่ผู้ใช้เป็นเจ้าของทั้งหมด เก็บข้อมูลรายละเอียดแต่ละวิชา จัดเก็บรูปสัต์ว์เลี้ยง และแสดงผลหน้าลงทะเบียนวิชา โดยจะแสดงรายละเอียดดังตารางที่ 3.24 , 3.31 , 3.32 , 3.33 และ 3.34 โดยแต่ละคลาสมีความสัมพันธ์กันดังรูปที่ 3.11



รูปที่ 3.11 แผนภาพคลาสของการลงทะเบียนวิชา

จากรูปที่ 3.11 เป็นการแสดงแผนภาพคลาสของการลงทะเบียนวิชา โดยรายละเอียดของแผนภาพคลาสจะอธิบายตามตารางที่ 3.31 ,3.32 ,3.33 และ 3.34 ดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.31 แสดงรายละเอียดของข้อมูลในคลาส SubjectController1

คำอธิบายคลาส คลาส SubjectController1 เป็นคลาสที่ใช้เกี่ยวกับการควบคุมการทำงานที่เกี่ยวข้องกับวิชา

SubjectController1		
ชื่อแอตทริบิวต์	ประเภทข้อมูล	ความหมาย
catPrefab	GameObject	ออบเจ็กต์สำหรับสร้างปุ่มกด
Model	SubjectListMdl	ออบเจ็กต์สำหรับเก็บข้อมูลรายวิชาที่ผู้ใช้มีทั้งหมด
View	SubjectView	ออบเจ็กต์เกี่ยวกับการแสดงผลหน้ารายวิชา
model	UserMdl	ออบเจ็กต์สำหรับเก็บ เรียกใช้ และจัดการข้อมูลของผู้เล่น

ชื่อคำสั่ง	พารามิเตอร์	ความหมาย
addButton	int id,String name	คำสั่งสำหรับการสร้างปุ่มกดของแต่ละรายวิชา โดยนำข้อมูลจาก model ได้แก่ชื่อรายวิชา และรหัสวิชา สร้างเป็นปุ่มกด
addSubject		คำสั่งสำหรับลงทะเบียนวิชา โดยนำรหัสวิชาที่ผู้ใช้ป้อนไว้ เทียบกับข้อมูลในฐานข้อมูล และลงทะเบียนรายวิชาให้ผู้ใช้งานหากไม่ผิดพลาด จะเรียกใช้คำสั่ง addButton เพื่อสร้างปุ่มแสดงรายวิชา
OnClick	int num,String name	คำสั่งที่ถูกเรียกเมื่อมีการกดปุ่มรายวิชาบนหน้าจอ จะทำการนำค่ารหัสวิชา และชื่อวิชา เก็บลงในตัวแปรเซสชัน และเปลี่ยนไปหน้าแบบทดสอบ
onClickAdd		คำสั่งที่ถูกเรียกเมื่อมีการกดปุ่มเพิ่มวิชา โดยจะทำการเรียกคำสั่ง addSubject ซึ่งใช้ในเพิ่มวิชาของผู้ใช้ลงฐานข้อมูลอีกที
onClickBack		คำสั่งที่ถูกเรียกเมื่อมีการกดปุ่มย้อนกลับ จะทำการเปลี่ยนหน้าไปหน้าเมนูหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

setModel		คำสั่งสำหรับขอข้อมูลรายวิชาของผู้ใช้จากฐานข้อมูล โดยใช้ชื่อผู้ใช้ในการอ้างอิง และเก็บลงในตัวแปร
----------	--	---

ตารางที่ 3.32 แสดงรายละเอียดของข้อมูลในคลาส SubjectListMdl

คำอธิบายคลาส คลาส SubjectListMdl เป็นคลาสที่ใช้เก็บวิชาเป็นรายการ และจัดการรายวิชา

SubjectListMdl		
ชื่อแอตทริบิวต์	ประเภทข้อมูล	ความหมาย
Subject	List<SubjectMdl>	รายการสำหรับเก็บออบเจกต์ของวิชา

ชื่อคำสั่ง	ประเภทข้อมูล	พารามิเตอร์	ความหมาย
addSubject	void	int id,String name	คำสั่งสำหรับเพิ่มข้อมูลวิชาลงในรายการวิชา โดยสร้างออบเจกต์ของคลาส SubjectMdl กำหนดรหัสวิชาและชื่อวิชา จากนั้นเพิ่มลงในรายการ
Clear	void		คำสั่งสำหรับลบทุกวิชาออกจากรายการวิชา โดยการเรียกคำสั่ง Clear ของคลาส List

ตารางที่ 3.33 แสดงรายละเอียดของข้อมูลในคลาส SubjectMdl

คำอธิบายคลาส คลาส SubjectMdl เป็นคลาสที่ใช้เก็บข้อมูลวิชา ซึ่งออกแบบให้สัมพันธ์กับตาราง subjects จึงได้ออกแบบให้มีการเก็บข้อมูลซึ่งประกอบไปด้วยรหัสวิชา และชื่อวิชา

SubjectMdl		
ชื่อแอตทริบิวต์	ประเภทข้อมูล	ความหมาย
subject_id	int	รหัสวิชา
subject_name	String	ชื่อวิชา

ตารางที่ 3.34 แสดงรายละเอียดของข้อมูลในคลาส SubjectView

คำอธิบายคลาส คลาส SubjectView เป็นคลาสที่ใช้เก็บข้อมูลวิชา

SubjectView			
ชื่อคำสั่ง	ประเภทคำสั่ง	พารามิเตอร์	ความหมาย
setButton	void	GameObject cat	แสดงปุ่มกดของรายวิชาที่ผู้ใช้ลงทะเบียนไว้

3) การออกแบบส่วนติดต่อผู้ใช้การลงทะเบียนวิชา



รูปที่ 3.12 หน้าจอการลงทะเบียนวิชา

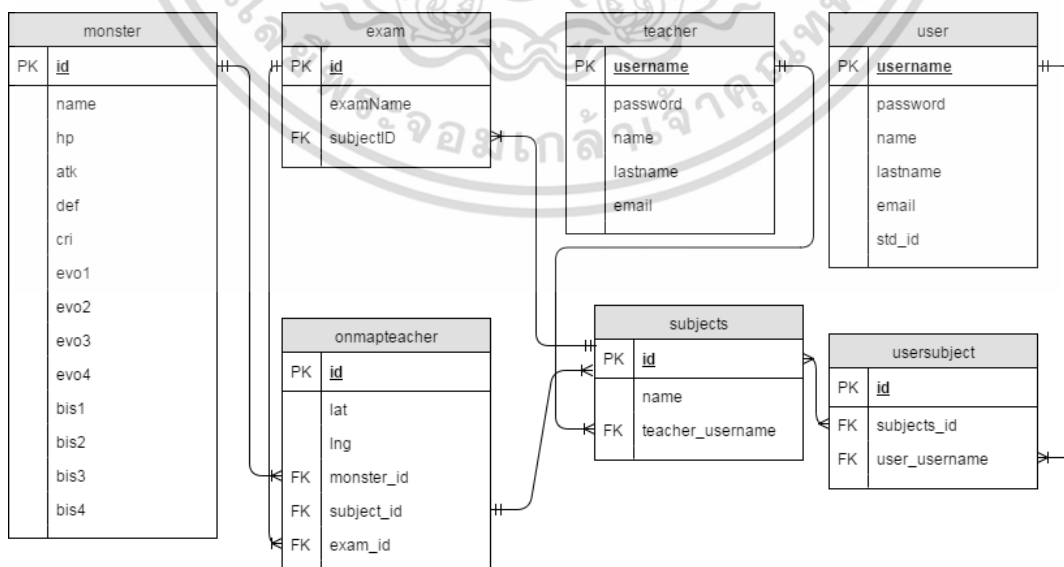
จากรูป 3.12 เป็นการออกแบบหน้าจอให้ผู้ใช้สามารถเห็นวิชาที่ลงทะเบียนวิชาแล้ว และเพิ่มรายวิชาได้ เนื่องจากเกมควิสแควสโก ผู้ใช้จะสามารถเล่นเกมตอบคำถามได้เฉพาะวิชาที่ผู้ใช้ลงทะเบียนไว้แล้วเท่านั้น ดังนั้นจะต้องป้อนข้อมูลเพื่อลงทะเบียนวิชา โดยข้อมูลที่ใช้ในการลงทะเบียนวิชาคือรหัสวิชา และเพื่อให้ผู้ใช้ทราบว่าได้ลงทะเบียนวิชาอะไรไว้แล้ว จึงออกแบบหน้าจอให้แสดงวิชาที่ผู้ใช้ลงทะเบียนไว้แล้ว

3.3.1.4 การออกแบบสำหรับยูสเคสการค้นหาคำถาม

จากการวิเคราะห์แผนภาพยูสเคสการค้นหาคำถาม หลังจากที่คุณผู้เรียนลงทะเบียนวิชาแล้ว ผู้เรียนจะต้องค้นหาคำถามในรายวิชาที่ลงทะเบียนไว้ก่อนเพื่อที่จะเข้าไปตอบคำถาม จึงได้ออกแบบตารางข้อมูล ออกแบบแผนภาพคลาส และออกแบบส่วนติดต่อผู้ใช้ ดังต่อไปนี้

1) การออกแบบตารางข้อมูลของการค้นหาคำถาม

ในการค้นหาคำถามในระบบเกมควิสแควสโก ดังที่กล่าวในหัวข้อที่ 3.2.2 เมื่อผู้ใช้งานป้อนค้นหาคำถาม ระบบจะตรวจสอบว่ามีคำถามอยู่บริเวณรอบตัวผู้ใช้หรือไม่ โดยคำถามที่ผู้ใช้พบได้จะเป็นคำถามหลายข้อในรูปแบบแบบทดสอบ และจะต้องเป็นคำถามที่อยู่ในวิชาที่ผู้ลงทะเบียนวิชาไว้แล้ว จึงได้มีการออกแบบตาราง exam เพื่อใช้เก็บข้อมูลอ้างอิงดังกล่าว และในการตรวจสอบว่าคำถามอยู่บริเวณรอบตัวผู้ใช้หรือไม่จำเป็นต้องใช้ตำแหน่งพิกัดบนแผนที่ที่อาจารย์ได้ปล่อยคำถามเอาไว้ จึงได้ออกแบบตารางข้อมูล onmapteacher ไว้เพื่อเก็บข้อมูลดังกล่าว ดังนั้นการออกแบบตารางในหัวข้อการค้นหาคำถามจะประกอบไปด้วยตารางข้อมูล monster, ตารางข้อมูล exam, ตารางข้อมูล subjects, ตารางข้อมูล teacher, ตารางข้อมูล onmapteacher, ตารางข้อมูล user และตารางข้อมูล usersubject โดยมีการแสดงรายละเอียดแต่ละตารางดังตารางที่ 3.15, 3.18, 3.28, 3.29 และ 3.30 ที่กล่าวไว้ข้างต้น และตารางที่ 3.35 และ 3.36 โดยแต่ละตารางมีความสัมพันธ์ดังรูปที่ 3.13



รูปที่ 3.13 ความสัมพันธ์ของตารางการค้นหาคำถาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.35 แสดงรายละเอียดของข้อมูลในตารางข้อมูล exam

คำอธิบายตาราง ตารางข้อมูล exam เป็นตารางที่ใช้เก็บแบบทดสอบ

Exam			
ชื่อคอลัมน์	ประเภทข้อมูล	ความหมาย	คีย์
Id	int(10)	รหัสแบบทดสอบ	Primary Key
examName	varchar(100)	ชื่อแบบทดสอบ	
subjectID	int(10)	รหัสวิชา	Foreign Key

ตารางที่ 3.36 แสดงรายละเอียดของข้อมูลในตารางข้อมูล onmapteacher

คำอธิบายตาราง ตารางข้อมูล onmapteacher เป็นตารางที่ใช้เก็บตำแหน่งของแบบทดสอบบน google map

Onmapteacher			
ชื่อคอลัมน์	ประเภทข้อมูล	ความหมาย	คีย์
Id	int(10)	รหัสตำแหน่ง	Primary Key
Lat	double	ค่าละติจูด	
Lng	double	ค่าลองจิจูด	
monster_id	int(10)	รหัสสมอนสเตอร์	Foreign Key
subject_id	int(10)	รหัสวิชา	Foreign Key
exam_id	int(10)	รหัสแบบทดสอบ	Foreign Key

โครงสร้างของตารางข้อมูล user สามารถดูรายละเอียดได้ในตารางที่ 3.15

โครงสร้างของตารางข้อมูล monster สามารถดูรายละเอียดได้ในตารางที่ 3.18

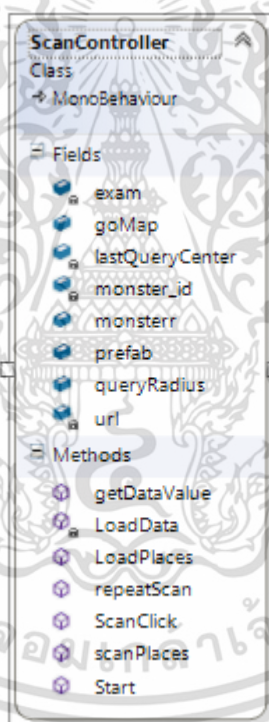
โครงสร้างของตารางข้อมูล usersubject สามารถดูรายละเอียดได้ในตารางที่ 3.28

โครงสร้างของตารางข้อมูล subjects สามารถดูรายละเอียดได้ในตารางที่ 3.29

โครงสร้างของตารางข้อมูล teacher สามารถดูรายละเอียดได้ในตารางที่ 3.30

2) การออกแบบแผนภาพคลาสของการค้นหาคำถาม

จากการออกแบบตารางข้อมูลของการค้นหาคำถาม สามารถนำมาออกแบบเป็นแผนภาพคลาสของการลงทะเบียนข้อมูล เพื่อแสดงความสัมพันธ์ระหว่างออบเจกต์ของคลาสที่เก็บข้อมูลจากตารางข้อมูล คลาสเพื่อแสดงผล และคลาสสำหรับจัดการการทำงานของลงทะเบียน เพื่อใช้ในการพัฒนาแอปพลิเคชัน จึงได้ออกแบบคลาส ScanController ซึ่งใช้ในการจัดอ้างอิงชื่อผู้ใช้ จัดการการทำงานในหน้าลงทะเบียนวิชา เก็บข้อมูลรายวิชาที่ผู้ใช้เป็นเจ้าของทั้งหมด เก็บข้อมูลรายละเอียดแต่ละวิชา จัดเก็บรูปสัตว์เลี้ยง และแสดงผลหน้าลงทะเบียนวิชา โดยจะแสดงรายละเอียดดังตารางที่ 3.24 , 3.31 , 3.32 , 3.33 และ 3.34 โดยแต่ละคลาสมีความสัมพันธ์กันดังรูปที่ 3.14



รูปที่ 3.14 แผนภาพคลาสของการค้นหาคำถาม

จากรูปที่ 3.14 เป็นการแสดงแผนภาพคลาสของการลงทะเบียนวิชา โดยรายละเอียดของแผนภาพคลาสจะอธิบายตามตารางดังที่ 3.35 ต่อไปนี้

ตารางที่ 3.37 แสดงรายละเอียดของข้อมูลในคลาส ScanController

คำอธิบายคลาส คลาส ScanController เป็นคลาสที่ใช้เกี่ยวกับการควบคุมการค้นหาคำถามบนแผนที่

ScanController		
ชื่อแอตทริบิวต์	ประเภทข้อมูล	ความหมาย
exam	String	รหัสแบบทดสอบ
goMap	GOMap	ออบเจ็กต์ของแผนที่ และข้อมูลอ้างอิงตำแหน่งผู้ใช้
lastQueryCenter	Coordinate	ตำแหน่งพิกัดผู้ใช้
monster_id	String	รหัสมอนสเตอร์
monsterr	String[]	อาเรียที่ใช้สำหรับเก็บข้อมูลของคำถามที่จะแสดง ชั่วคราวก่อนนำไปใส่ในคลาสโมเดลของข้อมูล
prefab	GameObject	ออบเจ็กต์มอนสเตอร์ที่ใช้แสดงบนแผนที่
queryRadius	float	ระยะรัศมีสูงสุดที่ใช้ค้นหาแบบทดสอบ
url	String	ที่อยู่ของไฟล์ที่ใช้ติดต่อฐานข้อมูลเพื่อเรียกข้อมูล ของแบบทดสอบที่พิกัดของผู้ใช้สามารถค้นหาเจอ ภายในรัศมี

ชื่อคำสั่ง	พารามิเตอร์	ความหมาย
LoadData	Coordinate currentloacation	คำสั่งที่ใช้ในการร้องขอไฟล์สำหรับสร้างแผนที่ ซึ่ง จะเรียกใช้ทุกครั้งที่มีการเปลี่ยนแปลงตำแหน่ง อ้างอิงบนพื้นโลกของผู้ใช้งาน
LoadPlace	String url	คำสั่งสำหรับเรียกข้อมูลแบบทดสอบที่พิกัดของ ผู้ใช้ภายในรัศมีที่กำหนดจากฐานข้อมูล โดยจะ ได้รับข้อมูลตำแหน่ง และรหัสแบบทดสอบ แล้ว สร้างออบเจ็กต์มอนสเตอร์เพื่อใช้เป็นปุ่มเข้าเล่น เกมตอบคำถามของแบบทดสอบ
repeatScan		คำสั่งสำหรับใช้เรียกคำสั่ง scanPlace ตามเวลาที่ กำหนดไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ScanClick		คำสั่งที่ถูกเรียกใช้เมื่อมีการกดปุ่มค้นหาคำถาม โดยจะเรียกคำสั่ง LoadPlace สำหรับแสดงมอนสเตอร์บนแผนที่
scanPlaces	String url	คำสั่งใช้ตรวจสอบว่ามีคำถามอยู่ในรัศมี10เมตรหรือไม่ ด้วยการตรวจสอบจากฐานข้อมูลโดยอ้างอิงจากตำแหน่งปัจจุบัน ถ้ามีจะแสดงเสียงเตือน

3.3.1.4.3 การออกแบบส่วนติดต่อผู้ใช้ของการค้นหาคำถาม



รูปที่ 3.15 หน้าจอการค้นหาคำถาม

จากรูปที่ 3.15 เป็นการออกแบบหน้าจอให้มีคำสั่งต่างๆไว้ให้ผู้ใช้ได้ใช้งาน ซึ่งคำสั่งที่รวบรวมไว้ประกอบไปด้วย การตรวจสอบสถานะมอนสเตอร์,จัดการรายวิชา,ค้นหาคำถาม,การต่อสู้อานประลอง และปรับแต่งหน้าจอ และเนื่องจากการค้นหาคำถามของผู้ใช้นั้นต้องใช้ตำแหน่งที่ผู้ใช้อยู่ปัจจุบันในการเล่น จึงได้ออกแบบให้มีการแสดงแผนที่เพื่อใช้อ้างอิงพื้นที่โดยรอบในปัจจุบัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

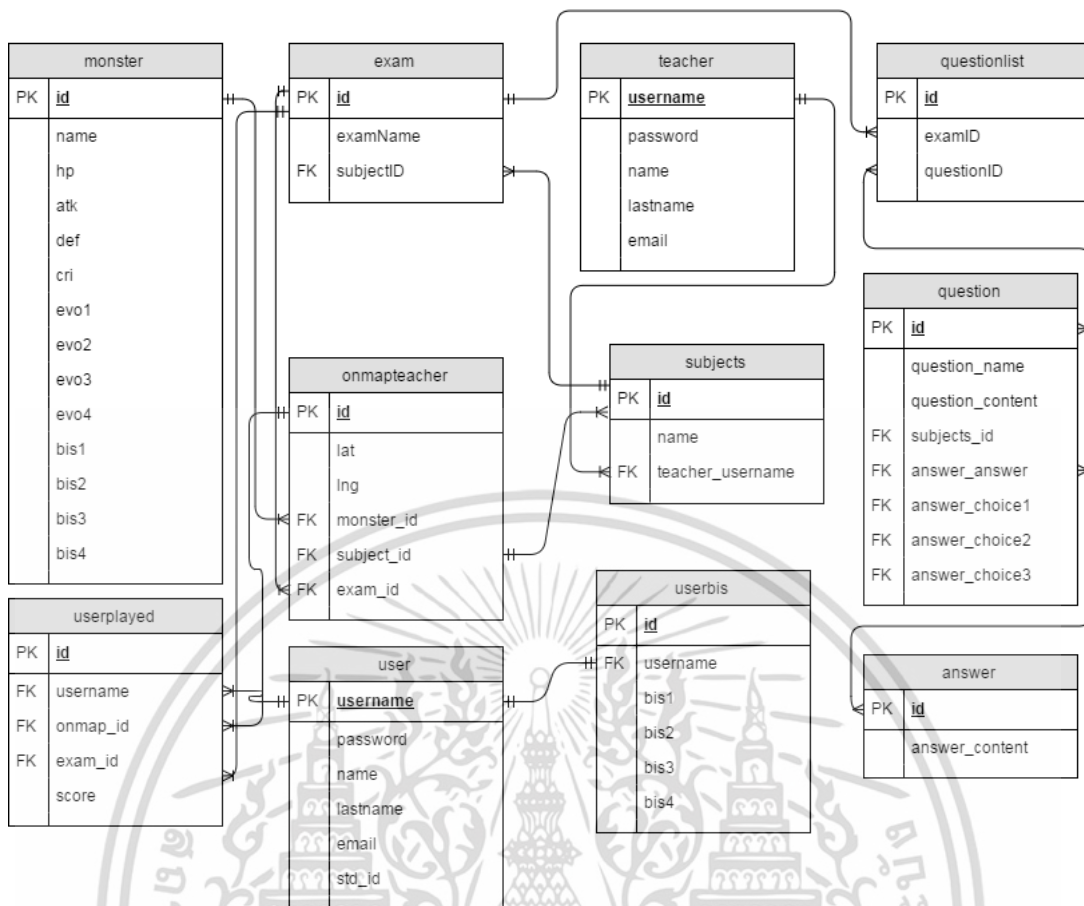
และมีตัวละครที่ใช้อ้างอิงจุดที่ผู้ใช้อยู่ และเนื่องจากผู้ใช้สามารถค้นหาคำถามได้ภายในรัศมี 20 เมตร เท่านั้น เพื่อให้เห็นขอบเขตในการค้นหาได้อย่างชัดเจนขึ้น จึงได้ออกแบบให้รอบตัวละครมีวงกลมรัศมี โดยคำถามที่ผู้ใช้จะเจอจะแสดงด้วยรูปของมอนสเตอร์ลอยจากพื้นเพื่อให้เห็นได้ชัด และกดได้ง่าย

3.3.1.5 การออกแบบสำหรับยูสเคสการเล่นเกมนตอบคำถาม

จากการวิเคราะห์แผนภาพยูสเคสการเล่นเกมนตอบคำถาม หลังจากที่ยุเรียนค้นหาคำถามที่ลงทะเบียนไว้เจอแล้ว ยูเรียนสามารถเข้าไปตอบคำถาม และรับของรางวัลหลังจากตอบคำถามเสร็จ จึงได้ออกแบบตารางข้อมูล ออกแบบแผนภาพคลาส และออกแบบส่วนติดต่อยุใช้ดังต่อไปนี้

1) การออกแบบตารางข้อมูลของการเล่นเกมนตอบคำถาม

ในส่วนเกมนตอบคำถามในระบบเกมนควิสควอสโก ดังที่กล่าวในหัวข้อที่ 3.2.2 เมื่อผู้ใช้เล่นเกมนตอบคำถาม ผู้ใช้จะพบมอนสเตอร์ และคำถามสี่ตัวเลือก ซึ่งมีคำตอบที่ถูกต้องเพียงหนึ่งคำตอบ โดยคำถามที่ผู้ใช้จะได้เล่นจะเป็นชุดคำถามในแบบทดสอบ และเมื่อผู้ใช้ตอบคำถามแล้ว ระบบจะต้องทำการเก็บข้อมูลที่ผู้ใช้ตอบคำถามไว้เพื่อดูคำถามย้อนหลัง และเมื่อผู้ใช้ตอบคำถามทั้งหมดเสร็จ ระบบจะต้องเก็บข้อมูลว่าเคยเล่นเกมนตอบคำถามของแบบทดสอบใดไปแล้วเพื่อไม่ให้เล่นซ้ำได้ และในแบบทดสอบที่เล่นนั้นได้คะแนนเท่าไร จึงได้ออกแบบตารางข้อมูล monster, ตารางข้อมูล userplayed, ตารางข้อมูล onmapteacher, ตารางข้อมูล question, ตารางข้อมูล userbis, ตารางข้อมูล exam, ตารางข้อมูล answer, ตารางข้อมูล teacher, ตาราง ข้อมูล subjects และตารางข้อมูล user โดยมีการแสดงรายละเอียดแต่ละตารางดังตารางที่ 3.15, 3.16, 3.18, 3.27, 3.28 และ 3.33 ที่กล่าวไว้ข้างต้น และตารางที่ 3.36, 3.37, 3.38, 3.39 และ 3.40 โดยแต่ละตารางมีความสัมพันธ์ดังรูปที่ 3.16



รูปที่ 3.16 ความสัมพันธ์ของตารางการเล่นเกมนตอบคำถาม

ตารางที่ 3.38 แสดงรายละเอียดของข้อมูลในตารางข้อมูล question

คำอธิบายตาราง ตารางข้อมูล question เป็นตารางที่ใช้เก็บคำถาม

Question			
ชื่อคอลัมน์	ประเภทข้อมูล	ความหมาย	คีย์
Id	int(10)	รหัสคำถาม	Primary Key
question_name	varchar(45)	ชื่อแบบทดสอบ	
question_content	varchar(500)	คำถาม	
subjects_id	int(10)	รหัสวิชา	Foreign Key
question_type	int(10)	รหัสรูปแบบคำถาม	Foreign Key
answer_answer	int(10)	รหัสคำตอบที่ถูกต้อง	Foreign Key
answer_choice1	int(10)	รหัสคำตอบตัวเลือกที่ 1	Foreign Key
answer_choice2	int(10)	รหัสคำตอบตัวเลือกที่ 2	Foreign Key
answer_choice3	int(10)	รหัสคำตอบตัวเลือกที่ 3	Foreign Key

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.39 แสดงรายละเอียดของข้อมูลในตารางข้อมูล answer

คำอธิบายตาราง ตารางข้อมูล answer เป็นตารางที่ใช้เก็บคำตอบ

Answer			
ชื่อคอลัมน์	ประเภทข้อมูล	ความหมาย	คีย์
Id	int(11)	รหัสคำตอบ	Primary Key
answer_content	varchar(500)	คำตอบ	

ตารางที่ 3.40 แสดงรายละเอียดของข้อมูลในตารางข้อมูล questionlist

คำอธิบายตาราง ตารางข้อมูล questionlist เป็นตารางที่ใช้เก็บชุดคำถาม

Questionlist			
ชื่อคอลัมน์	ประเภทข้อมูล	ความหมาย	คีย์
Id	int(10)	รหัสชุดคำถาม	Primary Key
examID	int(10)	รหัสแบบทดสอบ	Foreign Key
questionID	int(10)	รหัสคำถาม	Foreign Key

ตารางที่ 3.41 แสดงรายละเอียดของข้อมูลในตารางข้อมูล userquestion

คำอธิบายตาราง ตารางข้อมูล userquestion เป็นตารางที่ใช้เก็บข้อมูลคำถามของนักเรียนที่ทำไปแล้ว

Userquestion			
ชื่อคอลัมน์	ประเภทข้อมูล	ความหมาย	คีย์
Id	int(10)	รหัสคำถามที่ทำไปแล้ว	Primary Key
question_id	int(10)	รหัสคำถาม	Foreign Key
answer_id	int(10)	รหัสคำตอบ	Foreign Key
exam_id	int(10)	รหัสแบบทดสอบ	Foreign Key
username	int(10)	รหัสผู้ใช้นักเรียน	Foreign Key

ตารางที่ 3.42 แสดงรายละเอียดของข้อมูลในตารางข้อมูล userplayed

คำอธิบายตาราง ตารางข้อมูล userplayed เป็นตารางที่ใช้เก็บข้อมูลการเล่นของผู้ใช้ฝั่งนักเรียน

Userplayed			
ชื่อคอลัมน์	ประเภทข้อมูล	ความหมาย	คีย์
Id	int(10)	รหัสการเล่น	Primary Key
username	varchar(45)	ชื่อผู้ใช้ฝั่งนักเรียน	Foreign Key
onmap_id	int(10)	รหัสแผนที่	Foreign Key
exam_id	int(10)	รหัสแบบทดสอบ	Foreign Key
Score	int(3)	คะแนน	

โครงสร้างของตารางข้อมูล user สามารถดูรายละเอียดได้ในตารางที่ 3.15

โครงสร้างของตารางข้อมูล userbis สามารถดูรายละเอียดได้ในตารางที่ 3.16

โครงสร้างของตารางข้อมูล monster สามารถดูรายละเอียดได้ในตารางที่ 3.18

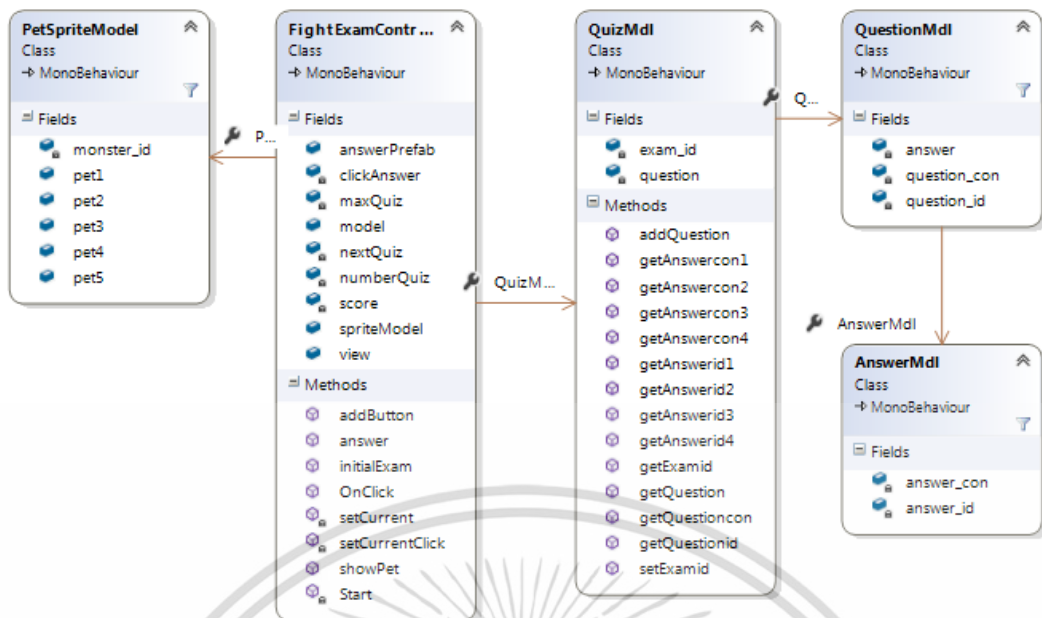
โครงสร้างของตารางข้อมูล subjects สามารถดูรายละเอียดได้ในตารางที่ 3.29

โครงสร้างของตารางข้อมูล teacher สามารถดูรายละเอียดได้ในตารางที่ 3.30

โครงสร้างของตารางข้อมูล exam สามารถดูรายละเอียดได้ในตารางที่ 3.35

2) การออกแบบแผนภาพคลาสของเกมตอบคำถาม

จากการออกแบบตารางข้อมูลของเกมตอบคำถาม สามารถนำมาออกแบบเป็นแผนภาพคลาสของเกม เพื่อแสดงความสัมพันธ์ระหว่างออบเจกต์ของคลาสที่เก็บข้อมูลจากตารางข้อมูล คลาสเพื่อการแสดงผล และคลาสสำหรับจัดการการทำงานของระบบ เพื่อใช้ในการพัฒนาแอปพลิเคชัน จึงได้ออกแบบคลาส FightExamController, QuizMdl , QuestionMdl และ AnswerMdl ซึ่งใช้ในการจัดการการเล่น และจัดการการทำงานในหน้าลงทะเบียนวิชา เก็บข้อมูลรายวิชาที่ผู้ใช้เป็นเจ้าของทั้งหมด เก็บข้อมูลรายละเอียดแต่ละวิชา จัดเก็บรูปสัตว์เลี้ยง และแสดงผลหน้าลงทะเบียนวิชา โดยจะแสดงรายละเอียดดังตารางที่ 3.24 , 3.31 , 3.32 , 3.33 และ 3.34 โดยแต่ละคลาสมีความสัมพันธ์กันดังรูปที่ 3.11



รูปที่ 3.17 แผนภาพคลาสของการเล่นเกมตอบคำถาม

จากรูปที่ 3.17 เป็นการแสดงแผนภาพคลาสของการเล่นเกมตอบคำถาม โดยรายละเอียดของแผนภาพคลาสดังตาราง 3.22 ที่กล่าวไว้ข้างต้น และตารางที่ 3.41, 3.42, 3.43 และ 3.44 ดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.43 แสดงรายละเอียดของข้อมูลในคลาส FightExamController

คำอธิบายคลาส คลาส FightExamController เป็นคลาสที่ใช้เกี่ยวกับการควบคุมการเล่นเกมนอบคำถาม

FightExamController		
ชื่อแอตทริบิวต์	ประเภทข้อมูล	ความหมาย
answerPrefab	GameObject	ออบเจกต์ปุ่มตอบคำถาม
clickAnswer	Bool	ใช้กำหนดว่าสามารถกดปุ่มตอบคำถามได้หรือไม่
maxQuiz	Int	จำนวนคำถามทั้งหมด
model	QuizMld	ออบเจกต์สำหรับเก็บคำถามที่ใช้เล่น
nextQuiz	bool	ใช้กำหนดว่าสามารถกดปุ่มเปลี่ยนคำถามได้หรือไม่
numberQuiz	Int	ชื่อของคำถามที่เล่นอยู่ในขณะนั้น
score	Int	คะแนนรวม
spriteModel	PetSpriteModel	ออบเจกต์สำหรับเก็บภาพที่ใช้แสดงของมอนสเตอร์
view	FightExamView	ออบเจกต์เกี่ยวกับการแสดงผลหน้าเล่นเกมตอบคำถาม

ชื่อคำสั่ง	พารามิเตอร์	หน้าที่ / การทำงาน
addButton	int pos	คำสั่งที่ถูกเรียกใช้งานภายในคำสั่ง OnClick ทำหน้าที่นำข้อมูลตัวเลือกจากออบเจกต์ model มาสร้างเป็นตัวเลือกในแต่ละข้อ
answer	int pos	คำสั่งที่ทำงานเมื่อมีการกดเลือกคำตอบ โดยจะได้รับค่าหมายเลขของคำตอบแล้วเช็คกับเงื่อนไขว่าเป็นคำตอบที่ถูกต้องหรือไม่ ถ้าถูกต้องให้เพิ่มคะแนน
initialExam		คำสั่งสำหรับขอข้อมูลคำถามของแบบทดสอบจากฐานข้อมูลโดยใช้รหัสแบบทดสอบในการอ้างอิง เมื่อได้รับข้อมูลคำถามจะเรียกใช้คำสั่งในการเพิ่มข้อมูลลงในออบเจกต์ model เพื่อเรียกใช้ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OnClick		คำสั่งที่ทำงานเมื่อมีการกดปุ่มคำถาม โดยจะเช็ค ว่าคำถามข้อปัจจุบันเกินกว่าจำนวนข้อคำถามใน model หรือไม่ ถ้าเกินจะย้ายไปหน้าแสดง ผลลัพธ์ ถ้าไม่เกินจะแสดงคำถามของข้อถัดไป และเรียกใช้คำสั่ง addButton เพื่อสร้างตัวเลือก
setCurrent	String x,String z	คำสั่งที่ถูกเรียกโดย setCurrentClick โดยจะนำ ค่ารหัสคำถาม และรหัสคำตอบ เพิ่มลงไปยัง ฐานข้อมูลเพื่อบอกว่าเราได้ตอบคำถามว่าอะไร
setCurrentClick	String x,String z	คำสั่งที่ทำงานเมื่อมีการกดปุ่มคำตอบ โดยจะ เรียกใช้คำสั่ง setCurrent และส่งค่า รหัสคำถาม (x) และ รหัสคำตอบ(y) เพื่อเพิ่มคำตอบที่เลือก ลงในฐานข้อมูล
showPet		คำสั่งสำหรับใช้สำหรับแสดงภาพของสัตว์เลี้ยง ประจำตัว เมื่อเข้าสู่หน้าจอเล่นเกม

ตารางที่ 3.44 แสดงรายละเอียดของข้อมูลในคลาส QuizMdl

คำอธิบายคลาส คลาส QuizMdl เป็นคลาสที่ใช้เก็บคำถามที่ใช้ในการเล่นเกมนตอบคำถาม ซึ่งได้
ออกแบบให้มีความสัมพันธ์กับตาราง questionlist โดยคลาสนี้ออกแบบให้เก็บรหัสแบบทดสอบ และ
นำคำถามที่เป็นของแบบทดสอบนั้นมาเก็บเป็นชุดคำถาม

QuizMdl		
ชื่อแอตทริบิวต์	ประเภทข้อมูล	ความหมาย
exam_id	String	รหัสแบบทดสอบ
question	List<QuestionMdl>	ชุดคำถามที่ใช้เล่น

ชื่อคำสั่ง	พารามิเตอร์	ความหมาย
addQuestion	String question_id, String question_con ,String answer_id1, String answer_id2, String answer_id3, String answer_id4	เพิ่มคำถามลงในชุดคำถาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.45 แสดงรายละเอียดของข้อมูลในคลาส QuestionMdl

คำอธิบายคลาส คลาส QuestionMdl เป็นคลาสที่ใช้เก็บคำถาม และคำตอบที่ใช้เป็นตัวเลือกในการเล่นเกมน ซึ่งได้ออกแบบให้มีความสัมพันธ์กับตาราง question จึงได้ออกแบบให้มีการเก็บข้อมูลซึ่งประกอบไปด้วยรหัสคำถาม, เนื้อหาคำถาม และชุดของคำตอบ

QuestionModel		
ชื่อแอตทริบิวต์	ประเภทข้อมูล	ความหมาย
answer	AnswerMdl[]	อาเรย์ใช้เก็บคำตอบ
question_con	String	เนื้อหาคำถาม
question_id	String	รหัสคำถาม

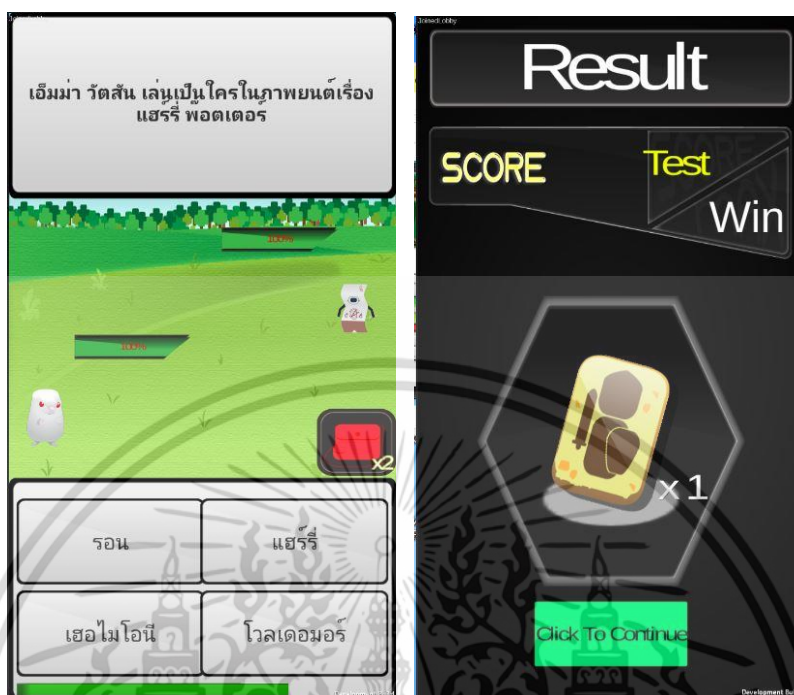
ตารางที่ 3.46 แสดงรายละเอียดของข้อมูลในคลาส AnswerMdl

คำอธิบายคลาส คลาส AnswerMdl เป็นคลาสที่ใช้เก็บคำตอบ ซึ่งได้ออกแบบให้มีความสัมพันธ์กับตาราง answer จึงได้ออกแบบให้มีการเก็บข้อมูลซึ่งประกอบไปด้วยรหัสคำตอบ และเนื้อหาคำตอบ

AnswerMdl		
ชื่อแอตทริบิวต์	ประเภทข้อมูล	ความหมาย
answer_id	String	รหัสตอบ
answer_con	String	เนื้อหาคำตอบ

โครงสร้างของตาราง PetSpriteModel สามารถดูรายละเอียดได้ในตารางที่ 3.24

3) การออกแบบส่วนติดต่อผู้ใช้ของการเล่นเกมนอบคำถาม



รูปที่ 3.18 หน้าจอการเล่นเกมนอบคำถาม

จากรูปที่ 3.18 เป็นการออกแบบให้ผู้ใช้กดปุ่มเลือกตอบคำถาม เนื่องจากเกมควิสแควสโกใช้คำถามในการเล่น เป็นแบบปรนัย 4 ตัวเลือก จึงได้ออกแบบให้ผู้ใช้กดตอบคำถามได้เพียงแค่ตัวเลือกเดียวเท่านั้น แล้วจึงเปลี่ยนคำถามได้ และได้ออกแบบเมื่อเล่นเกมนอบคำถามจบแล้วให้แสดงข้อมูลผลลัพธ์ของการเล่นเกมนอบคำถามโดยข้อมูลที่แสดงประกอบไปด้วยคะแนนที่ทำได้ และของรางวัล

3.3.1.6 การออกแบบสำหรับยูสเคสการตรวจสอบสถานะมอนสเตอร์

จากการวิเคราะห์แผนภาพยูสเคสการตรวจสอบสถานะมอนสเตอร์ หลังจากที่คุณเรียนเข้าสู่ระบบแล้ว ผู้เรียนสามารถตรวจสอบสถานะของมอนสเตอร์ สามารถเพิ่มค่าสถานะต่างๆ และสามารถวิวัฒนาการมอนสเตอร์ได้ จึงได้ออกแบบตารางข้อมูล ออกแบบแผนภาพคลาส และออกแบบส่วนติดต่อผู้ใช้ ดังต่อไปนี้

1) การออกแบบตารางข้อมูลตรวจสอบสถานะมอนสเตอร์

ในส่วนตรวจสอบสถานะมอนสเตอร์ในระบบเกมควิสควิสโก ดังที่กล่าวในหัวข้อที่ 3.2.2 ผู้ใช้สามารถตรวจสอบสถานะของมอนสเตอร์ประจำตัวอันได้แก่ ค่าพลังโจมตี ค่าพลังป้องกัน ค่าความแม่นยำ ค่า และพลังชีวิต ซึ่งผู้ใช้สามารถนำบิสกิตที่ผู้ใช้มีอยู่ นำมาเพิ่มสถานะให้กับสัตว์เลี้ยงได้ เมื่อให้มอนสเตอร์ประจำตัวกินบิสกิต ค่าสถานะนั้นๆจะเพิ่มขึ้น และค่าที่ใช้เก็บจำนวนบิสกิตในประเภทนั้นก็เพิ่มขึ้นเพื่อใช้ตรวจสอบการเข้าเงื่อนไขการวิวัฒนาการ จึงได้ออกแบบตารางข้อมูล monster, ตารางข้อมูล user, ตารางข้อมูล pet และตารางข้อมูล userbis โดยมีการแสดงรายละเอียดแต่ละตารางดังตารางที่ 3.15, 3.16, 3.17 และ 3.18 ที่กล่าวไว้ข้างต้น โดยแต่ละตารางมีความสัมพันธ์ดังรูปที่ 3.4

โครงสร้างของตารางข้อมูล user สามารถดูรายละเอียดได้ในตารางที่ 3.15

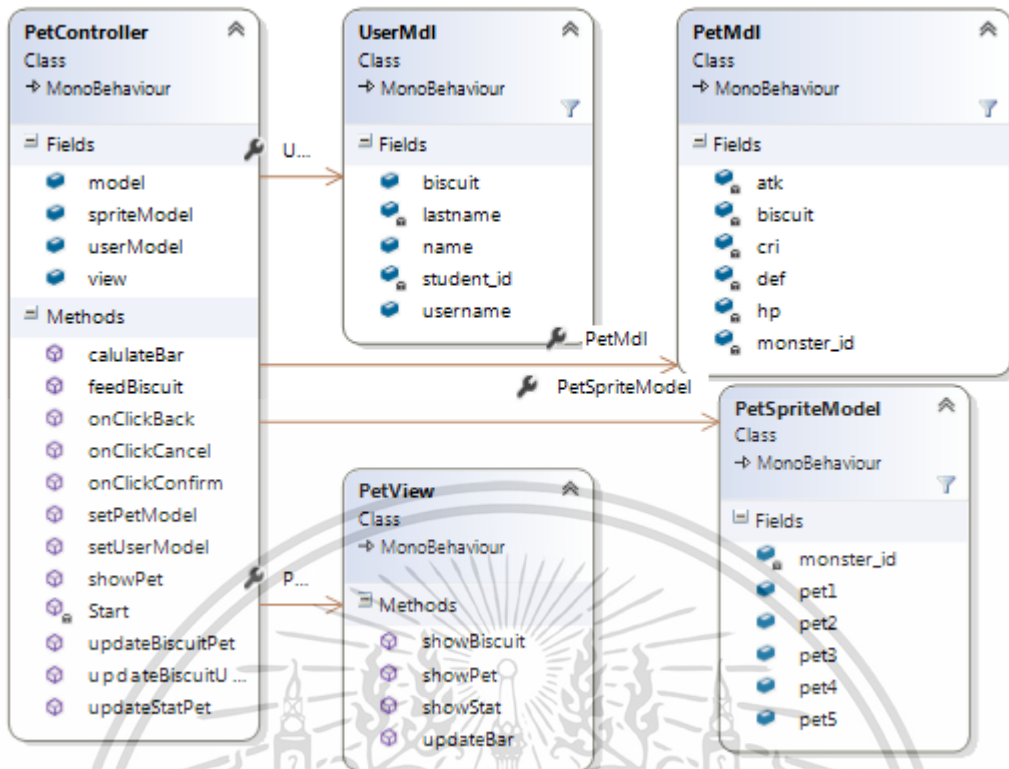
โครงสร้างของตารางข้อมูล userbis สามารถดูรายละเอียดได้ในตารางที่ 3.16

โครงสร้างของตารางข้อมูล pet สามารถดูรายละเอียดได้ในตารางที่ 3.17

โครงสร้างของตารางข้อมูล monster สามารถดูรายละเอียดได้ในตารางที่ 3.18

2) การออกแบบแผนภาพคลาสของการตรวจสอบสถานะมอนสเตอร์

จากการออกแบบตารางข้อมูลของการเล่นเกมนตอบคำถาม สามารถนำมาออกแบบเป็นแผนภาพคลาสของการเล่นเกมน เพื่อแสดงความสัมพันธ์ระหว่างออบเจกต์ของคลาสที่เก็บข้อมูลจากตารางข้อมูล คลาสเพื่อการแสดงผล และคลาสสำหรับจัดการการทำงานของ การลงทะเบียน เพื่อใช้ในการพัฒนาแอปพลิเคชัน จึงได้ออกแบบคลาส PetController, UserMdl , PetMdl , PetSpriteModel และ PetView ซึ่งใช้ในการจัดการการเล่นเกมน และจัดการการทำงานในหน้าลงทะเบียนวิชา เก็บข้อมูลรายวิชาที่ผู้ใช้เป็นเจ้าของทั้งหมด เก็บข้อมูลรายละเอียดแต่ละวิชา จัดเก็บรูปสัตว์เลี้ยง และแสดงผลหน้าลงทะเบียนวิชา โดยจะแสดงรายละเอียดดังตารางที่ 3.45 , 3.24 , 3.25 , 3.22 และ 3.46 โดยแต่ละคลาสมีความสัมพันธ์กันดังรูปที่ 3.19



รูปที่ 3.19 แผนภาพคลาสของการตรวจสอบสถานะมอนสเตอร์

จากรูปที่ 3.19 เป็นการแสดงแผนภาพคลาสของการตรวจสอบสถานะมอนสเตอร์ โดยรายละเอียดของแผนภาพคลาสจะอธิบายตามตารางที่ 3.45, 3.46, 4.47 และ 3.48 ดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.47 แสดงรายละเอียดของข้อมูลในคลาส PetController

คำอธิบายคลาส คลาส PetController เป็นคลาสที่ใช้เกี่ยวกับการควบคุมการตรวจสอบสถานะมอนสเตอร์

PetController		
ชื่อแอตทริบิวต์	ประเภทข้อมูล	ความหมาย
model	PetMdl	ออบเจ็กต์สำหรับเก็บข้อมูลมอนสเตอร์ประจำตัว
spriteModel	PetSpriteModel	ออบเจ็กต์สำหรับเก็บภาพที่ใช้แสดงของมอนสเตอร์
userModel	UserMdl	ออบเจ็กต์สำหรับเก็บข้อมูลผู้ใช้
view	PetView	ออบเจ็กต์เกี่ยวกับการแสดงผลหน้าจอนำตรวจสอบสถานะสัตว์เลี้ยง

ชื่อคำสั่ง	พารามิเตอร์	ความหมาย
calculateBar		คำสั่งสำหรับคำนวณความกว้างของหลอดแสดงบิสกิตที่สัตว์เลี้ยงกิน โดยนำค่าบิสกิตของสัตว์เลี้ยงหาร 100 เป็นความกว้าง แล้วเรียกคำสั่ง updateBar ของออบเจ็กต์ view
feedBiscuit	int pos	คำสั่งสำหรับเพิ่มจำนวนบิสกิตประเภทที่เลือกของสัตว์เลี้ยง โดยจะเพิ่มลงในออบเจ็กต์ model และลดบิสกิตประเภทที่เลือกของผู้ใช้ในออบเจ็กต์ userModel เพื่อใช้ในการปรับปรุงข้อมูลในฐานข้อมูลอีกที
onClickBack		คำสั่งที่ทำงานเมื่อกดปุ่มย้อนกลับ โดยจะเปลี่ยนหน้าไปหน้าจอเมนูหลัก
onClickCancel		คำสั่งซึ่งถูกเรียกเมื่อกดปุ่มยกเลิก โดยจะยกเลิกการกินบิสกิตที่ยังไม่ได้กดยืนยัน และปรับข้อมูลให้เป็นข้อมูลเดิมโดยการเรียกคำสั่ง setPetModel และ setUserModel

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

onCliclConfirm		คำสั่งซึ่งถูกเรียกเมื่อมีการกดปุ่มยืนยัน โดยจะไปเรียกคำสั่ง updateBiscuitPet , updateBiscuitUser และ UpdateStatPet เพื่อปรับปรุงข้อมูลในฐานข้อมูลให้เป็นข้อมูลใหม่
setPetModel		คำสั่งสำหรับขอข้อมูลสัตว์เลี้ยงของผู้ใช้จากฐานข้อมูล โดยใช้ชื่อผู้ใช้ในการอ้างอิง และเก็บลงในตัวแปร
setUserModel		คำสั่งสำหรับขอข้อมูลของผู้ใช้จากฐานข้อมูล โดยใช้ชื่อผู้ใช้ในการอ้างอิง และเก็บลงในตัวแปร
showPet		คำสั่งสำหรับแสดงภาพของสัตว์เลี้ยงเมื่อเข้าหน้าจอตระกูลสถานะสัตว์เลี้ยง
updateBiscuitPet		คำสั่งสำหรับปรับปรุงค่าบิสกิตที่สัตว์เลี้ยงกินลงในฐานข้อมูล
updateBiscuitUser		คำสั่งสำหรับปรับปรุงบิสกิตที่เหลืออยู่ของผู้ใช้ลงในฐานข้อมูล
updateStatPet		คำสั่งสำหรับปรับปรุงค่าสถานะของสัตว์เลี้ยงลงในฐานข้อมูล

ตารางที่ 3.48 แสดงรายละเอียดของข้อมูลในคลาส PetView

คำอธิบายคลาส คลาส PetView เป็นคลาสที่ใช้เกี่ยวกับการแสดงผลหน้าจอนำหน้าตรวจสอบสถานะมอนสเตอร์

PetView		
ชื่อคำสั่ง	พารามิเตอร์	ความหมาย
showBiscuit		คำสั่งสำหรับแสดงบิสกิตของผู้ใช้ที่มีอยู่
showPet	Sprite mon	คำสั่งสำหรับแสดงภาพของสัตว์เลี้ยง โดยรับค่าเป็นรูปภาพมาแสดง
showStat	int hp, int atk, int def, int cri	คำสั่งสำหรับแสดงค่าสถานะของสัตว์เลี้ยง โดยจะแสดงค่า พลังชีวิต พลังโจมตี พลังป้องกัน และค่าความแม่นยำ
updateBar	int[] biscuit	คำสั่งสำหรับแสดงหลอดบิสกิตที่มอนสเตอร์ประจำตัวกิน โดยรับค่าตัวเลขที่มากเพื่อแสดงความยาวหลอดบิสกิตตามตัวเลข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ในห้องเรียนเท่านั้น ไม่อนุญาตให้เผยแพร่ไปใช้ประโยชน์ในการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงสร้างของตาราง PetSpriteModel สามารถดูรายละเอียดได้ในตารางที่ 3.22

3) การออกแบบส่วนติดต่อผู้ใช้การตรวจสอบสถานะมอนสเตอร์



รูปที่ 3.20 หน้าจอตรวจสอบสถานะมอนสเตอร์

จากรูปที่ 3.20 เป็นการออกแบบหน้าจอให้ผู้ใช้ตรวจสอบ และเพิ่มสถานะให้มอนสเตอร์ประจำตัวเก่งขึ้น เพื่อนำไปสู่ในลานประลอง เนื่องจากการต่อสู้ลานประลองนั้นจะนำค่าสถานะของมอนสเตอร์ไปคำนวณระหว่างที่ต่อสู้กัน ดังนั้นยิ่งค่าสถานะในด้านต่างๆที่สูงขึ้นจะยิ่งทำให้ได้เปรียบในการเล่นประลอง ซึ่งค่าสถานะของมอนสเตอร์ประกอบไปด้วย พลังชีวิต,พลังโจมตี,พลังป้องกัน และความแม่นยำ ในหน้าจอนี้จะออกแบบให้ผู้ใช้สามารถเพิ่มค่าสถานะให้มอนสเตอร์ประจำตัวได้ด้วยการใช้ของรางวัลที่ได้รับจากการเล่นเกมตอบคำถาม และในการเพิ่มค่าสถานะได้ออกแบบให้มีการจำกัดการเพิ่มสถานะอีกด้วยเพื่อไม่ให้ค่าสถานะสูงเกินไป และได้ออกแบบให้สามารถยกเลิกหรือยืนยันการเพิ่มค่าสถานะได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.1.7 การออกแบบสำหรับยูสเคสการตรวจสอบคะแนน

จากการวิเคราะห์แผนภาพยูสเคสการตรวจสอบคะแนน หลังจากที่คุณผู้เรียนลงทะเบียนวิชา และเล่นเกมตอบคำถามแล้ว ผู้เรียนสามารถตรวจสอบคะแนนของแบบทดสอบที่ได้ตอบคำถามไปแล้วได้ จึงได้ออกแบบตารางข้อมูล ออกแบบแผนภาพคลาส และออกแบบส่วนติดต่อผู้ใช้ ดังต่อไปนี้

1) การออกแบบตารางข้อมูลตรวจสอบคะแนน

ในส่วนตรวจสอบคะแนน ในระบบเกมควิสแควสโก ดังที่กล่าวในหัวข้อที่ 3.2.2 ผู้ใช้สามารถตรวจสอบคะแนนจากการตอบคำถามย้อนหลังได้ โดยฐานข้อมูลจะตรวจสอบชื่อเข้าใช้งานของผู้ใช้ และวิชาที่ผู้ใช้เลือกจะดูคะแนน ระบบจะเรียกแบบทดสอบที่ผู้ใช้เคยเล่น และคะแนนของแบบทดสอบนั้นมาแสดง และเมื่อผู้ใช้เลือกแบบทดสอบ ระบบจะเรียกข้อมูลคำถาม และคำตอบที่ผู้ใช้เคยเล่นในแบบทดสอบนั้นเพื่อเป็นการดูคำตอบย้อนหลังได้ จึงได้มีการออกแบบตาราง user, ตาราง userplayed, ตาราง exam, ตาราง question, ตาราง answer, ตาราง userbis, ตาราง subjects, ตาราง monster, ตาราง teacher, ตาราง userquestion และตาราง onmapteacher โดยมีการแสดงรายละเอียดแต่ละตารางดังตารางที่ 3.15, 3.16, 3.18, 3.27, 3.28, 3.33, 3.34, 3.36, 3.37, 3.39 และ 3.40 ที่กล่าวไว้ข้างต้น โดยแต่ละตารางมีความสัมพันธ์ดังรูปที่ 3.16

โครงสร้างของตาราง user สามารถดูรายละเอียดได้ในตารางที่ 3.15

โครงสร้างของตาราง userbis สามารถดูรายละเอียดได้ในตารางที่ 3.16

โครงสร้างของตาราง monster สามารถดูรายละเอียดได้ในตารางที่ 3.18

โครงสร้างของตาราง subjects สามารถดูรายละเอียดได้ในตารางที่ 3.27

โครงสร้างของตาราง teacher สามารถดูรายละเอียดได้ในตารางที่ 3.28

โครงสร้างของตาราง exam สามารถดูรายละเอียดได้ในตารางที่ 3.33

โครงสร้างของตาราง onmapteacher สามารถดูรายละเอียดได้ในตารางที่ 3.34

โครงสร้างของตาราง question สามารถดูรายละเอียดได้ในตารางที่ 3.36

โครงสร้างของตาราง answer สามารถดูรายละเอียดได้ในตารางที่ 3.37

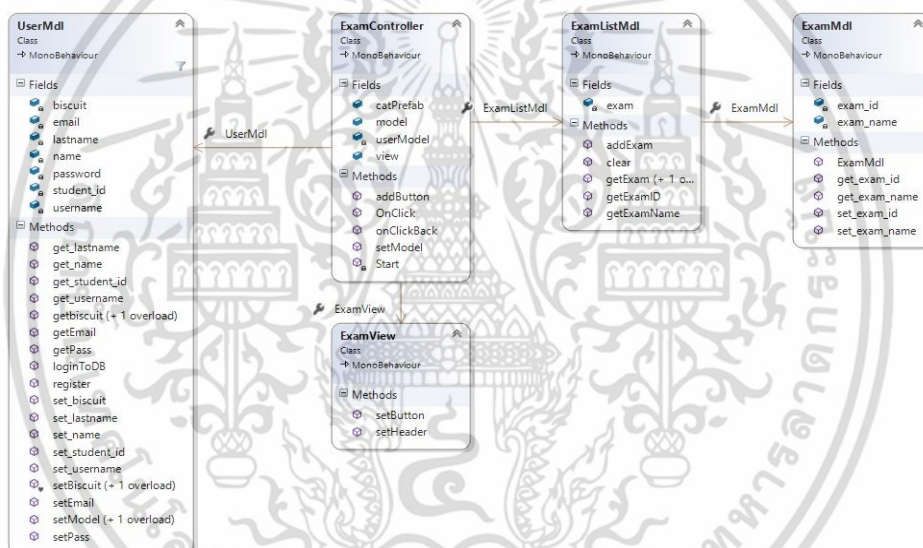
โครงสร้างของตาราง userquestion สามารถดูรายละเอียดได้ในตารางที่ 3.39

โครงสร้างของตาราง userplayed สามารถดูรายละเอียดได้ในตารางที่ 3.40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) การออกแบบแผนภาพคลาสของการตรวจสอบคะแนน

จากการออกแบบตารางข้อมูลของการตรวจสอบคะแนน สามารถนำมาออกแบบเป็นแผนภาพคลาสของการตรวจสอบคะแนน เพื่อแสดงความสัมพันธ์ระหว่างออบเจกต์ของคลาสที่เก็บข้อมูลจากตารางข้อมูล คลาสเพื่อการแสดงผล และคลาสสำหรับจัดการการทำงานของ การลงทะเบียน เพื่อใช้ในการพัฒนาแอปพลิเคชัน จึงได้ออกแบบคลาส ExamController, UserMdl , ExamListMdl , ExamMdl และ ExamView ซึ่งใช้ในการจัดการการทำงานของ การตรวจสอบคะแนน และแสดงแบบทดสอบ เก็บข้อมูลผู้ใช้งาน เก็บข้อมูลแบบทดสอบ เก็บข้อมูลรายละเอียดแต่ละแบบทดสอบ และแสดงผลหน้าตรวจสอบคะแนน โดยจะแสดงรายละเอียดดังตารางที่ 3.47 , 3.24 , 3.48 , 3.49 และ 3.50 โดยแต่ละคลาสมีความสัมพันธ์กันดังรูปที่ 3.21



รูปที่ 3.21 แผนภาพคลาสของการตรวจสอบคะแนน

จากรูปที่ 3.21 เป็นการแสดงแผนภาพคลาสของการตรวจสอบคะแนน โดยรายละเอียดของแผนภาพคลาสจะอธิบายตามตารางที่ 3.47, 3.48, 3.49 และ 3.50 ดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.49 แสดงรายละเอียดของข้อมูลในคลาส ExamController

คำอธิบายคลาส คลาส ExamController เป็นคลาสที่ใช้เกี่ยวกับการควบคุมเกี่ยวกับตรวจสอบคะแนน และแบบทดสอบที่เคยทำ

ExamController		
ชื่อแอตทริบิวต์	ประเภทข้อมูล	ความหมาย
catPrefab	GameObject	ออบเจ็กต์ปุ่มคำถาม
model	ExamListMdl	ออบเจ็กต์สำหรับเก็บรายการแบบทดสอบที่เคยทำ
userModel	UserMdl	ออบเจ็กต์สำหรับเก็บข้อมูลผู้ใช้
View	ExamView	ออบเจ็กต์เกี่ยวกับการแสดงผลหน้าตรวจสอบคะแนน

ชื่อคำสั่ง	พารามิเตอร์	ความหมาย
addButton	int id,String name	คำสั่งสำหรับการสร้างปุ่มกดของแต่ละแบบทดสอบที่ได้ทำไว้แล้ว โดยนำข้อมูลจาก model ได้แก่ชื่อแบบทดสอบ คะแนนที่ได้ และรหัสแบบทดสอบ สร้างเป็นปุ่มกด
setModel		คำสั่งสำหรับการขอข้อมูลแบบทดสอบ และคะแนนจากฐานข้อมูล โดยจะทำการส่งชื่อผู้ใช้ และเลขรหัสวิชา เพื่อใช้อ้างอิง และเมื่อได้รับข้อมูล จะเก็บข้อมูลลงในออบเจ็กต์ model
OnClick	int num,String name	คำสั่งที่ถูกเรียกเมื่อมีการกดปุ่มแบบทดสอบบนหน้าจอ จะทำการนำค่ารหัสแบบ ชื่อแบบทดสอบ เก็บลงในตัวแปรเซชัน และเปลี่ยนไปหน้าคำตอบที่เคยทำ
onClickBack		คำสั่งที่ถูกเรียกเมื่อมีการกดปุ่มย้อนกลับ จะทำการเปลี่ยนหน้าไปหน้าเมนูลงทะเบียนวิชา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.50 แสดงรายละเอียดของข้อมูลในคลาส ExamListMdl

คำอธิบายคลาส คลาส ExamListMdl เป็นคลาสที่ใช้เก็บรายการของแบบทดสอบ

ExamListMdl		
ชื่อแอตทริบิวต์	ประเภทข้อมูล	ความหมาย
Exam	List<ExamMdl>	รายการสำหรับเก็บคำถามที่เคยทำ

ชื่อคำสั่ง	ประเภทข้อมูล	พารามิเตอร์	ความหมาย
addExam	void	int id,String name	คำสั่งสำหรับเพิ่มข้อมูลแบบทดสอบลงในรายการแบบทดสอบ โดยสร้างออบเจ็กต์ของคลาส ExamMdl กำหนดรหัสแบบทดสอบ และชื่อแบบทดสอบ จากนั้นเพิ่มลงในรายการ
Clear	void		คำสั่งสำหรับลบทุกแบบทดสอบออกจากรายการแบบทดสอบ โดยการเรียกคำสั่ง Clear ของคลาส List

ตารางที่ 3.51 แสดงรายละเอียดของข้อมูลในคลาส ExamMdl

คำอธิบายคลาส คลาส ExamMdl เป็นคลาสที่ใช้เก็บข้อมูลแบบทดสอบที่เคยทำแล้ว ซึ่งได้ออกแบบให้มีความสัมพันธ์กับตาราง exam จึงได้ออกแบบให้มีการเก็บข้อมูลซึ่งประกอบไปด้วยรหัสแบบทดสอบ และชื่อแบบทดสอบ

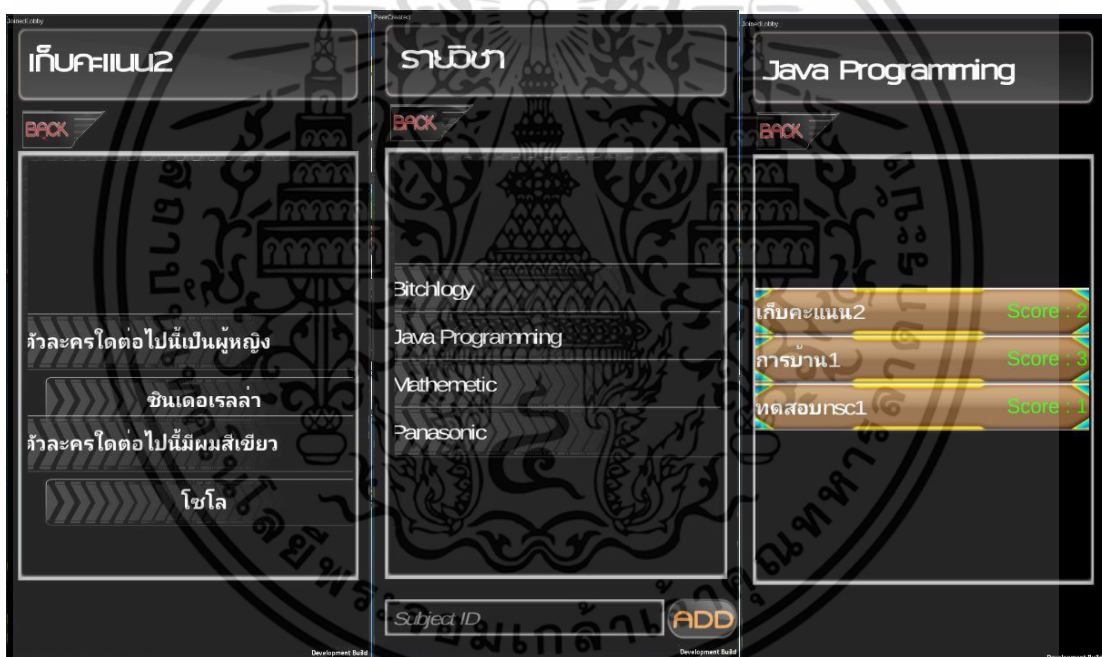
ExamMdl		
ชื่อแอตทริบิวต์	ประเภทข้อมูล	ความหมาย
exam_id	String	เก็บรหัสแบบทดสอบ
exam_name	String	เก็บชื่อแบบทดสอบ

ตารางที่ 3.52 แสดงรายละเอียดของข้อมูลในคลาส ExamView

คำอธิบายคลาส คลาส ExamView เป็นคลาสที่ใช้เกี่ยวกับการแสดงผลหน้าจอนำตรวจสอบคะแนน

QuestionView		
ชื่อคำสั่ง	พารามิเตอร์	ความหมาย
setButton	GameObjects cat	คำสั่งสำหรับแสดงปุ่มของแบบทดสอบที่เคยทำไว้แล้ว
setHeader		คำสั่งสำหรับใช้แสดงชื่อวิชาที่เข้ามาดูแบบทดสอบไว้ด้านบนสุด

3) การออกแบบส่วนติดต่อผู้ใช้การตรวจสอบคะแนน



รูปที่ 3.22 หน้าจอการตรวจสอบรายละเอียดการตอบคำถาม

จากรูปที่ 3.22 เป็นการออกแบบให้ผู้ใช้กดปุ่มเพื่อดูรายละเอียดการตอบคำถาม เนื่องจากผู้ใช้สามารถคะแนนรวมในแต่ละแบบทดสอบภายในวิชาได้ และยังสามารถดูคำถาม และคำตอบของแบบทดสอบที่ทำไปแล้วได้ ซึ่งข้อมูลที่ใช้ในหน้าตรวจสอบรายละเอียดการตอบคำถามจะมีโครงสร้างแบบลำดับขั้นดังนี้ รายวิชาซึ่งมีแบบทดสอบเป็นเนื้อหาย่อยลงมา และแบบทดสอบสอบซึ่งมีคำถาม และคำตอบเนื้อหาย่อยลงมา จึงได้ออกแบบให้ผู้ใช้กดปุ่มหัวข้อต่างๆเพื่อเข้าสู่ข้อมูลที่เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื้อหาย่อยข้างใน เช่น ผู้ใช้กดปุ่มวิชาการเขียนโปรแกรมพื้นฐานแล้วจะแสดงแบบทดสอบที่เคยทำ เมื่อผู้ใช้กดที่แบบทดสอบก็จะแสดงคำถาม และคำตอบของแบบทดสอบนั้นๆ

3.3.1.8 การออกแบบสำหรับยูสเคสการต่อสู้ลานประลอง

จากการวิเคราะห์แผนภาพยูสเคสการต่อสู้ลานประลอง หลังจากที่คุณเรียนเข้าสู่ระบบแล้ว ผู้เรียนสามารถนำมอนสเตอร์ของตนเองไปต่อสู้ที่ลานประลองกับเพื่อนได้ จึงได้ออกแบบตารางข้อมูล ออกแบบแผนภาพคลาส และออกแบบส่วนติดต่อผู้ใช้ ดังต่อไปนี้

1) การออกแบบตารางข้อมูลของการต่อสู้ลานประลอง

ในส่วนการต่อสู้ลานประลองในระบบเกมควิสควอสโก ดังที่กล่าวในหัวข้อที่ 3.2.2 ผู้ใช้สามารถทำประลองต่อสู้กับผู้ใช้คนอื่นได้ โดยฐานข้อมูลจะตรวจสอบชื่อผู้ใช้งานของทั้งสองฝั่งแล้วเรียกข้อมูลมอนสเตอร์ของผู้ใช้ทั้งสองฝั่ง จากนั้นระบบจึงเรียกคำถามและคำตอบที่ใช้ในการประลองจากฐานข้อมูล เพื่อให้ผู้ใช้ทั้งสองฝั่งประลองกันโดยการตอบคำถาม แต่ในการต่อสู้ลานประลองจะไม่มีกรเก็บคำตอบเหมือนกับการเล่นเกมตอบคำถาม ดังนั้นจึงได้ออกแบบตาราง user, ตาราง monster, ตาราง userbis และตาราง pet โดยมีการแสดงรายละเอียดแต่ละตารางดังตารางที่ 3.15, 3.16, 3.17 และ 3.18 ที่กล่าวไว้ข้างต้น โดยแต่ละตารางมีความสัมพันธ์ดังรูปที่ 3.4

โครงสร้างของตาราง user สามารถดูรายละเอียดได้ในตารางที่ 3.15

โครงสร้างของตาราง userbis สามารถดูรายละเอียดได้ในตารางที่ 3.16

โครงสร้างของตาราง pet สามารถดูรายละเอียดได้ในตารางที่ 3.17

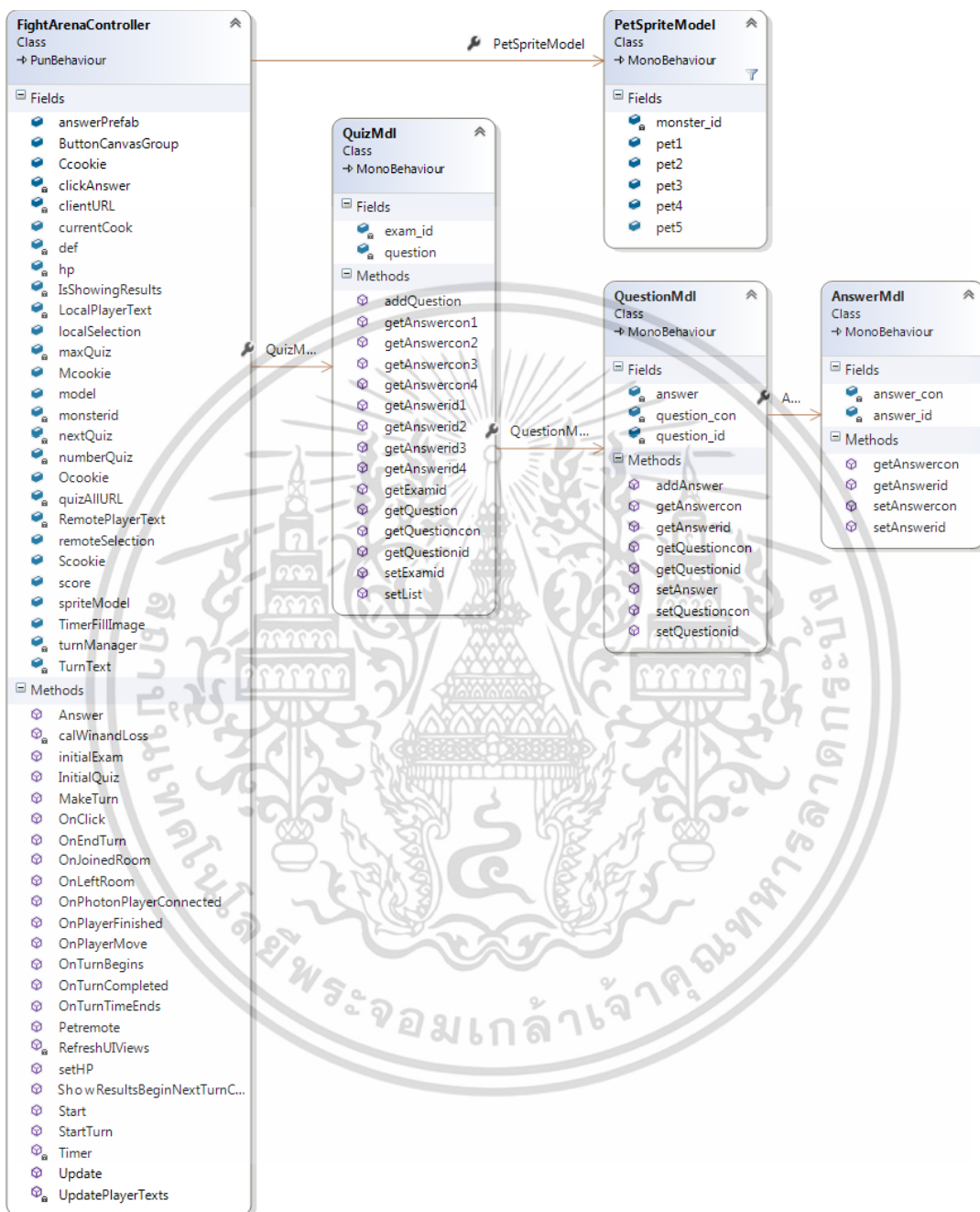
โครงสร้างของตาราง monster สามารถดูรายละเอียดได้ในตารางที่ 3.18

2) การออกแบบแผนภาพคลาสของการต่อสู้ลานประลอง

การออกแบบแผนภาพคลาสของการต่อสู้ลานประลองนั้น จะมีความคล้ายกันกับการออกแบบแผนภาพการเล่นเกม ซึ่งแตกต่างกันเพียงแค่การต่อสู้ลานประลองจะจัดการการทำงานของหน้าจอกด้วยคลาส FightArenaController ซึ่งต้องทำการสืบทอดคลาส PhotonNetworkManager เพื่อให้สามารถใช้งานเซิร์ฟเวอร์ของทางยูนิตีซึ่งใช้สำหรับการเล่นเกม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ออนไลน์แข่งขันกันได้ โดรนจะแสดงข้อมูลของคลาส FightArenaController ไว้ที่ตาราง 3.53 และแสดงแผนภาพคลาสการต่อสู้ลานประลองดังรูป 3.23



รูปที่ 3.23 แผนภาพคลาสของการต่อสู้ลานประลอง

จากรูปที่ 3.23 เป็นการแสดงแผนภาพคลาสของการต่อสู้ลานประลอง โดยรายละเอียดของแผนภาพคลาสจะอธิบายตามตารางที่ 3.53 ดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.53 แสดงรายละเอียดของข้อมูลในตาราง FightArenaController

คำอธิบายตาราง ตาราง FightArenaController เป็นตารางที่ใช้เกี่ยวกับการเก็บข้อมูลรูปภาพ และจัดการสถานะต่างๆของมอนสเตอร์ของผู้เล่นทั้งสองฝ่าย รวมถึงการสร้างคำถาม

FightArenaController		
ชื่อแอตทริบิวต์	ประเภทข้อมูล	ความหมาย
answerPrefab	GameObject	ออบเจกต์สำหรับสร้างปุ่มตัวเลือก
ButtonCavasGroup		ชุดของกลุ่มตัวเลือก เพื่อใช้ในการอ้างอิงตำแหน่งเมื่อมีการสร้างปุ่มตัวเลือก
Ccookie	Sprite	รูปบิสกิตรูปแบบที่2
clickAnswer	Bool	ตัวแปรตรรกศาสตร์ ใช้ตรวจสอบว่ามีการเลือกคำตอบหรือไม่
clickURL	String	ที่อยู่ของไฟล์ที่ใช้ติดต่อฐานข้อมูลเพื่อเรียกข้อมูลของมอนสเตอร์ประจำตัว
currentCook	Image	เป็นบิสกิตที่จะได้รับหลังจากเล่นเกมต่อสู่สถานทดลองเสร็จสิ้น
def	Int	ค่าพลังป้องกันฝ่ายตนเอง
Hp	Int	ค่าพลังมอนสเตอร์ฝ่ายตรงข้าม
isShowingResult	Bool	แสดงผลพัธอยู่หรือไม่
LocalPlayerText	Text	ชื่อผู้เล่น
localSelection	Text	คำตอบที่ผู้เล่นเลือก
maxQuiz	Int	จำนวนคำถามทั้งหมด
Mcookie	Sprite	รูปบิสกิตรูปแบบที่4
model	QuizMdl	ออบเจกต์สำหรับเก็บคำถามที่ใช้เล่น
monsterId	String	รหัสมอนสเตอร์ฝ่ายตนเอง
nextQuiz	Bool	สามารถไปข้อถัดไปได้หรือไม่
numberQuiz	Int	เลขข้อของคำถาม
Ocookie	Sprite	รูปบิสกิตรูปแบบที่3
quizAllURL	String	ที่อยู่ของไฟล์ที่ใช้ติดต่อฐานข้อมูลเพื่อเรียกข้อมูลของคำถามเพื่อใช้ในการทดลอง
remotePlayerText	Text	ชื่อผู้เล่นฝ่ายตรงข้าม
remoteSelection	Int	คำตอบที่ผู้เล่นฝ่ายตรงข้ามเลือก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Scookie	Sprite	รูปบิสกิตรูปแบบที่1
score	Int	คะแนนของตนเอง
spriteModel	PetSpriteModel	ออบเจ็กต์สำหรับเก็บภาพที่ใช้แสดงของมอนสเตอร์
TimeFillImage	Image	รูปภาพใช้อ้างอิงการจับเวลาในการเล่นเกมน ตอบคำถามในลานประลองแต่ละรอบ
turnManager	PUNTurnManager	ออบเจ็กต์สำหรับจัดการระบบเครือข่าย
turnText	Text	ข้อความแสดงรอบของผู้เล่นฝ่ายใด

ชื่อคำสั่ง	พารามิเตอร์	ความหมาย
Answer	string x	คำสั่งที่เรียกใช้งานเมื่อมีการกดปุ่มตัวเลือก โดยจะตรวจสอบคำตอบที่เลือกว่ามีรหัสคำตอบที่ถูกหรือผิด เมื่อถูกจะลดพลังสัตว์เลี้ยงฝ่ายตรงข้าม
calWinandLoss		คำสั่งที่ถูกเรียกใช้เมื่อผู้เล่นเลือกคำตอบ โดยจะคำนวณว่าผู้เล่นใดตอบถูกหรือผิด เป็นคำสั่งที่ใช้สำหรับคำนวณทั้งสองผู้เล่น
InitialQuiz		คำสั่งที่ถูกเรียกเมื่อเข้าลานประลองมาแล้ว เพื่อตั้งค่าเริ่มต้นสำหรับการเล่นเกมตอบคำถามได้แก่จำนวนข้อ พลังชีวิต ภาพสัตว์เลี้ยงฝ่ายตรงข้ามและตนเอง
MakeTurn	int attack	คำสั่งสำหรับส่งข้อมูลฝ่ายตรงข้าม โดยเมื่อเราตอบถูกจะส่งค่าพลังโจมตีไปให้ผู้เล่นฝ่ายตรงข้าม แต่ถ้าตอบผิดจะส่งเลขศูนย์ไป เพื่อบอกว่าตอบผิด และไม่มีการลดพลังชีวิตฝ่ายตรงข้าม
OnClick		คำสั่งที่ทำงานเมื่อมีการกดปุ่มคำถาม และจะทำงานได้เมื่อถึงรอบของผู้เล่นเท่านั้น หากถึงรอบตัวเองจะทำงานโดยจะเช็คค่าพลังชีวิตของฝ่ายใดฝ่ายหนึ่งหมดหรือไม่ หากหมดจะย้ายไปหน้าจอให้รางวัล ถ้าไม่จะทำการสุ่มคำถามของผู้เล่นฝ่ายนั้นขึ้นมาเพื่อใช้ในการเล่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OnEndTurn		คำสั่งสำหรับจัดการระบบเครือข่ายโดยบอกว่า หมดรอบของตนเอง และเริ่มรอบใหม่เป็นรอบ ของฝ่ายตรงข้าม
OnJoinedRoom		คำสั่งสำหรับแจ้งเซิร์ฟเวอร์ว่ามีผู้เล่นต่อเข้ามาใน ลานประลอง และส่งข้อมูลให้แก่ผู้เล่นฝ่ายตรง ข้าม เพื่อใช้ในการแสดงผลหน้าจอ
OnLeftRoom		คำสั่งสำหรับแจ้งเซิร์ฟเวอร์ว่ามีผู้เล่นออกจาก ลานประลอง โดยจะทำการหยุดการเล่นเมื่อ มีผู้เล่นออกจากลานประลอง
OnPhotonPlayerConne cted	PhotonPlayer newPlayer	คำสั่งสำหรับบอกเซิร์ฟเวอร์ว่ามีผู้เล่นใหม่เข้ามา ในลานประลอง และเรียกใช้คำสั่ง InitialQuiz สำหรับผู้เล่นใหม่ที่เข้ามา
OnPlayerFinished	PhotonPlayer player,int turn,object move	เป็นคำสั่งสำหรับรับ และส่งค่าการเล่นของผู้เล่น ในแต่ละรอบ โดยจะรับ และส่งค่าที่มาจากคำสั่ง MakeTurn ของผู้เล่นฝ่ายตรงข้าม และตนเอง
OnPlayerMove	PhotonPlayer player,int turn,object move	คำสั่งสำหรับรับ และส่งค่าการเล่นของผู้เล่นใน แต่ละรอบ โดยจำเรียกคำสั่ง MakeTurn เพื่อ บอกว่าผู้เล่นกระทำสิ่งใดไป
OnTurnBegins	int turn	คำสั่งที่ทำงานเมื่อเริ่มรอบของตนเอง โดยจัดการ เปลี่ยนข้อความ และบอกว่าเป็นผู้เล่นฝ่ายใดที่ ได้เล่นในรอบนี้
OnTurnComplete	int turn	คำสั่งที่เรียกเมื่อหมดรอบของการเล่น โดยจะ เรียกใช้คำสั่ง calWinAndLoss เพื่อตรวจสอบว่า ผู้เล่นฝ่ายตรงข้าม และตนเองตอบถูกหรือผิด หรือไม่
PetRemote		คำสั่งที่เรียกเมื่อมีผู้เล่นใหม่เข้ามา ทำหน้าที่ใช้ ตรวจสอบมอสต์วี่เลี้ยงทั้งสองฝ่าย และแสดง ภาพมอนสเตอร์ตัวนั้น
RefreshUIView		คำสั่งที่เรียกใช้งานเมื่อผู้เล่นเข้ามาในลาน ประลอง โดยจะทำการตั้งค่าพลังให้เต็มทั้งสอง ฝ่าย และแสดงชื่อของผู้เล่นทั้งสองฝ่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่สามารถนำออกเผยแพร่โดยไม่ได้รับอนุญาต

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

setHP		คำสั่งสำหรับกำหนดค่าพลังมอนสเตอร์ให้เต็ม โดยใช้ค่าพลังของผู้ในการกำหนด
ShowResultsBeginNext TurnCoroutine		คำสั่งที่แสดงผลพลังชีวิตโดยหากฝ่ายตรงข้ามตอบถูกลดพลังชีวิตผู้เล่น แต่หากผู้เล่นถูกลดเลือกฝ่ายตรงข้าม
StartTurn		คำสั่งสำหรับแจ้งเซิร์ฟเวอร์ว่าเริ่มรอบใหม่
Timer		คำสั่งสำหรับใช้จับเวลาในการตอบคำถาม โดยจะลดลงตามเวลาที่กำหนด และเมื่อหมดเวลาจะเรียกคำสั่ง Maketurn และส่งค่าศูนย์เพื่อบอกว่าตอบไม่ทันเวลา
Update		คำสั่งที่ทำงานตลอดเวลา ใช้สำหรับตรวจสอบว่ามีผู้เล่นเข้ามาในห้องหรือไม่ ถ้ามีแสดงชื่อผู้เล่นฝ่ายตรงข้ามเมื่อเข้ามา และใช้สำหรับแสดงข้อความว่ารอบนี้เป็นของผู้เล่นฝ่ายใด
UpdatePlayerText		คำสั่งที่ถูกเรียกโดยคำสั่ง Update ใช้ในการแสดงชื่อของผู้เล่น เมื่อมีผู้เล่นใหม่เข้ามาในลานประลอง

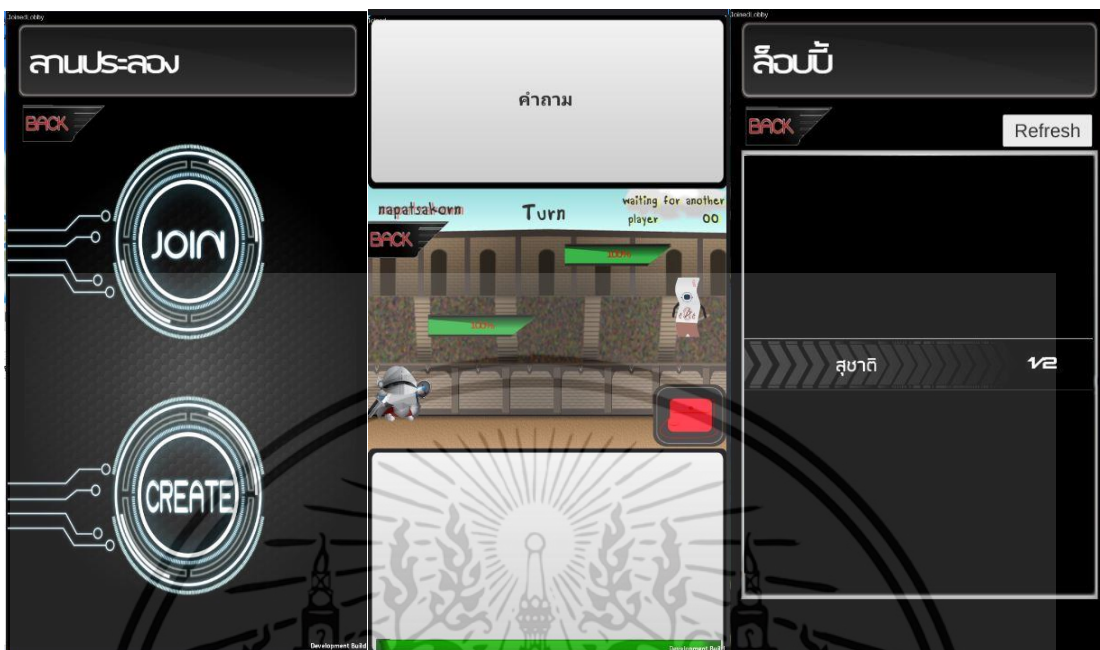
โครงสร้างของตาราง PetSpriteModel สามารถดูรายละเอียดได้ในตารางที่ 3.22

โครงสร้างของตาราง QuizMdl สามารถดูรายละเอียดได้ในตารางที่ 3.42

โครงสร้างของตาราง QuestionMdl สามารถดูรายละเอียดได้ในตารางที่ 3.43

โครงสร้างของตาราง AnswerMdl สามารถดูรายละเอียดได้ในตารางที่ 3.44

3) การออกแบบส่วนติดต่อผู้ใช้ของการต่อสู้อานประลอง



รูปที่ 3.24 หน้าจอการต่อสู้อานประลอง

จากรูปที่ 3.24 เป็นการออกแบบให้ผู้ใช้เล่นเกมตอบคำถามกับผู้อื่น เนื่องจากการต่อสู้อานประลองเป็นการเล่นแบบผลัดกันตอบคำถามระหว่างผู้ใช้สองคน จึงได้ออกแบบให้หน้าแรกของการต่อสู้อานประลองมีคำสั่งอยู่สองคำสั่งคือสร้างห้องแล้วไปรอที่ลานประลอง และเข้าร่วมห้อง ซึ่งเมื่อผู้ใช้เข้ามาในลานประลอง ได้ออกแบบให้ผู้เล่นที่สร้างห้องเล่นเกมตอบคำถามก่อน โดยจะมีข้อความบอกว่ารอบที่กำลังเล่นเป็นของใคร และเมื่อตอบเสร็จก็เปลี่ยนเป็นผู้ใช้อีกคนเล่นเกมตอบคำถาม

3.3.2 การออกแบบฝั่งอาจารย์

จากการวิเคราะห์แผนภาพยูสเคสในหัวข้อ 3.1.2 จะประกอบไปด้วยความสามารถหลักของแอปพลิเคชัน ได้แก่

3.3.2.1 การออกแบบสำหรับยูสเคสการลงทะเบียน

จากการวิเคราะห์แผนภาพยูสเคสการลงทะเบียน ผู้สอนต้องลงทะเบียนก่อนเพื่อใช้งานแอปพลิเคชัน จึงได้ออกแบบตารางข้อมูล ออกแบบแผนภาพคลาส และออกแบบส่วนติดต่อผู้ใช้ ดังต่อไปนี้

1) การออกแบบตารางข้อมูลของการลงทะเบียน

การลงทะเบียนเข้าสู่ระบบ ในเว็บไซต์ควิสแควสโก ดังที่ได้กล่าวในหัวข้อที่ 3.2.2 จำเป็นจะต้องเก็บข้อมูลส่วนบุคคลเช่น ชื่อ นามสกุล อีเมลล์ และข้อมูลที่ใช้อ้างอิงข้อมูลส่วนบุคคล ได้แก่ชื่อเข้าใช้ และรหัสผ่าน จึงได้ออกแบบตาราง teacher ซึ่งสามารถเก็บข้อมูลผู้ใช้ ดังตารางที่ 3.28 ที่กล่าวไว้ข้างต้น และมีการแสดงรายละเอียดความสัมพันธ์ของตารางดังรูปที่ 3.25

teacher	
PK	<u>username</u>
	password
	name
	lastname
	email

รูปที่ 3.25 ความสัมพันธ์ของตารางการลงทะเบียนของเว็บไซต์

โครงสร้างของตาราง teacher สามารถดูรายละเอียดได้ในตารางที่ 3.28

2) การออกแบบส่วนติดต่อผู้ใช้ของการลงทะเบียน

เมื่ออาจารย์เข้าสู่เว็บไซต์ควิสแควสโก ออกแบบให้มีหน้าจอ 2 ฟังก์ชัน คือ sign in สำหรับเข้าสู่ระบบของผู้ใช้งานที่ลงทะเบียนกับแอปพลิเคชันแล้ว และฟังก์ชัน Registration ในการลงทะเบียนสำหรับผู้ใช้งานใหม่

QuizQuestGO Home About Contact Sign in ▾

Registration

Username

Password

Confirm Password

Name

Lastname

Email

[SIGN UP](#)

© 2017 quizquestgo.com · [Home](#) · [About](#) · [Contact](#) [Back to Top](#)

รูปที่ 3.26 หน้าต่างสมัครสมาชิก

จากรูปที่ 3.26 เป็นการออกแบบหน้าจอให้ผู้ใช้งานใหม่ป้อนข้อมูลเพื่อใช้ลงทะเบียน โดยข้อมูลที่ใช้ในการลงทะเบียนเป็นข้อมูลทั่วไปของอาจารย์ซึ่งประกอบไปด้วย ชื่อใช้งาน รหัสผ่าน ชื่อนามสกุล และอีเมลล์

3.3.2.2 การออกแบบสำหรับยูสเคสการเข้าสู่ระบบ

จากการวิเคราะห์แผนภาพยูสเคสการเข้าสู่ระบบ หลังจากที่คุณผู้สอนลงทะเบียนแล้ว ผู้สอนต้องเข้าสู่ระบบเพื่อเข้าใช้งานเว็บแอปพลิเคชัน จึงได้ออกแบบตารางข้อมูล ออกแบบแผนภาพคลาส และออกแบบส่วนติดต่อผู้ใช้ ดังต่อไปนี้

1) การออกแบบตารางข้อมูลของการเข้าสู่ระบบ

ในการเข้าสู่ระบบ ในเว็บไซต์ควิสเคสโก ดังที่กล่าวในหัวข้อที่ 3.2.2 ระบบจะนำข้อมูลที่ผู้ใช้ป้อนเข้ามา นำไปตรวจสอบในฐานข้อมูลว่าข้อมูลตรงกันหรือไม่ ซึ่งใช้ชื่อผู้ใช้งาน และรหัสผ่านในการตรวจสอบ จึงได้ออกแบบตาราง teacher ดังตารางที่ 3.28 เพื่อตรวจสอบข้อมูลดังกล่าว และมีการแสดงรายละเอียดความสัมพันธ์ของตารางดังรูปที่ 3.25 ที่กล่าวไว้ข้างต้น

โครงสร้างของตาราง teacher สามารถดูรายละเอียดได้ในตารางที่ 3.28



2) การออกแบบส่วนติดต่อผู้ใช้ของการเข้าสู่ระบบ

QuizQuestGO Home About Contact Sign in ▾

Sign In

Username
teach1

Password

Remember me?

LOG IN Sign Up

© 2017 quizquestgo.com · Home · About · Contact [Back to Top](#)

รูปที่ 3.27 หน้าต่างเข้าสู่ระบบ

จากรูปที่ 3.27 เป็นการออกแบบหน้าจอให้ผู้ใช้งานป้อนข้อมูลเพื่อเข้าสู่ระบบของเว็บไซต์ควิสเควสโก ซึ่งข้อมูลที่ใช้เข้าสู่ระบบประกอบไปด้วยชื่อเข้าใช้งาน และรหัสผ่าน และได้มีการออกแบบให้ผู้ใช้งานใหม่ที่ยังไม่มีข้อมูลใช้เข้าสู่ระบบสามารถลงทะเบียนได้อีกด้วย และเมื่อเข้าสู่ระบบสำเร็จได้ ออกแบบให้มีเมนูสำหรับเข้าใช้งานคำสั่งต่างๆ และเมนูให้ผู้ใช้ออกจากระบบได้ ดังรูปที่ 3.28

QuizQuestGO Home Subject LOG OUT

Log In Success, Welcome teach1

© 2017 quizquestgo.com · Home · About · Contact [Back to Top](#)

รูปที่ 3.28 หน้าต่างยืนยันตัวตน

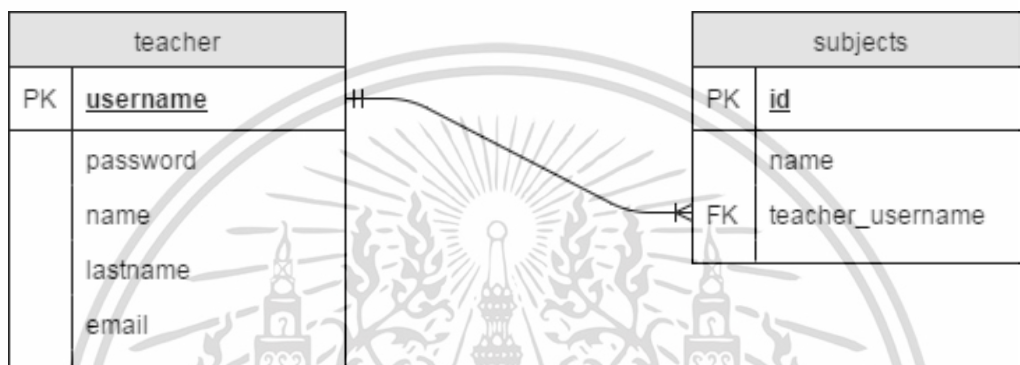
3.3.2.3 การออกแบบสำหรับยุคการสร้างรายวิชา

จากการวิเคราะห์แผนภาพยุคการสร้างรายวิชา หลังจากผู้สอนเข้าสู่ระบบแล้ว ผู้สอนจำเป็นต้องสร้างรายวิชาก่อน จึงจะสามารถสร้างคำถาม และกำหนดตำแหน่งคำถามได้ จึงได้ออกแบบตารางข้อมูล ออกแบบแผนภาพคลาส และออกแบบส่วนติดต่อผู้ใช้ ดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) การออกแบบตารางข้อมูลของการสร้างรายวิชา

ในการสร้างรายวิชาในเว็บไซต์ควิสแควสโก ดังที่กล่าวในหัวข้อที่ 3.2.2 ในการสร้างรายวิชา ผู้ใช้จะต้องป้อนชื่อวิชาที่ต้องการ แล้วระบบจะสร้างวิชาโดยใช้ชื่อเข้าใช้ของผู้ใช้เก็บในวิชานั้นด้วยเพื่อใช้อ้างอิงว่าเป็นวิชาของอาจารย์ท่านใด จึงได้ออกแบบตาราง teacher และ subjects เพื่อสร้างวิชาดังกล่าว ดังตารางที่ 3.27 และ 3.28 ที่กล่าวไว้ข้างต้น และมีการแสดงรายละเอียดความสัมพันธ์ของตารางดังรูปที่ 3.29

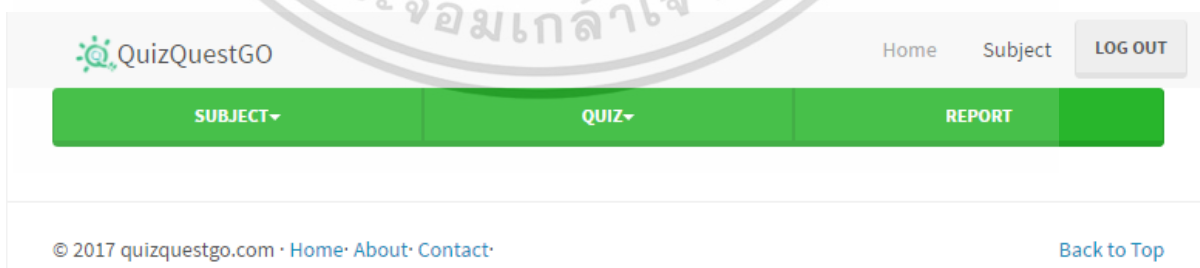


รูปที่ 3.29 ความสัมพันธ์ของตารางการสร้างรายวิชาของเว็บไซต์

โครงสร้างของตาราง subjects สามารถดูรายละเอียดได้ในตารางที่ 3.27

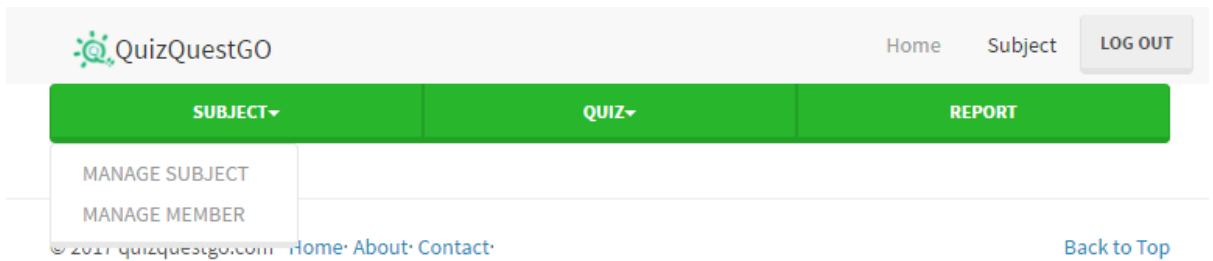
โครงสร้างของตาราง teacher สามารถดูรายละเอียดได้ในตารางที่ 3.28

2) การออกแบบส่วนติดต่อผู้ใช้ของการสร้างรายวิชา

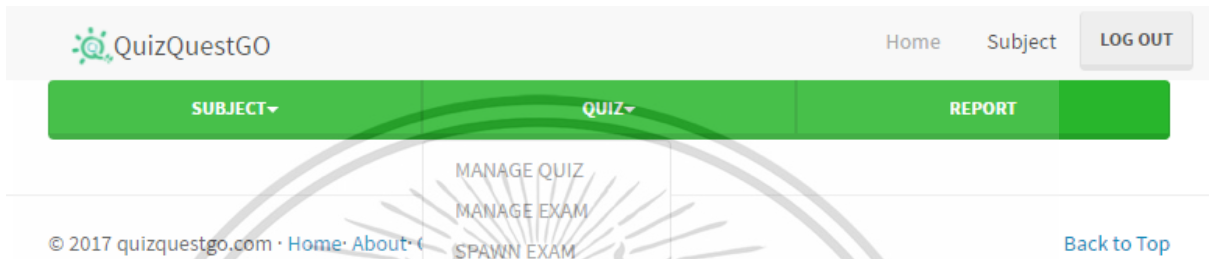


รูปที่ 3.30 หน้าจอแสดงแถบเมนู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.31 หน้าจอแสดงแถบเมนู subject



รูปที่ 3.32 หน้าจอแสดงแถบเมนู Quiz

จากรูปที่ 3.30 รูปที่ 3.31 และรูปที่ 3.32 เป็นการออกแบบหน้าจอให้ผู้ใช้เลือกคำสั่งเพื่อจัดการข้อมูล เพื่อให้ผู้ใช้ใช้งานง่าย จึงออกแบบให้แยกคำสั่งออกเป็นสามประเภทประกอบไปด้วยการจัดการรายวิชา, การจัดการคำถาม และรายงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

QuizQuestGO
Home Subject LOG OUT

Create Subject

Subject Name

Manage Subject

ID PIN	SUBJECT NAME
8	Java Programming
10	English For Com
13	Alkaline
14	Evergreen
17	Algorithm
22	การ์ตูน

รูปที่ 3.33 หน้าต่างสร้างรายวิชา

จากรูปที่ 3.33 เป็นการออกแบบเพื่อให้ผู้ใช้ป้อนข้อมูลเพื่อสร้างรายวิชา และแสดงวิชาที่เคยสร้างไว้แล้ว ซึ่งข้อมูลที่ใช้ในการสร้างรายวิชาคือชื่อวิชา และเมื่อสร้างเสร็จจะมีการแสดงข้อมูลวิชาในตารางด้วย จึงได้ออกแบบให้ข้อมูลวิชาที่แสดงในแต่ละแถวประกอบไปด้วยรหัสวิชาและชื่อวิชา เพื่อให้รหัสวิชาไปให้นักเรียนลงทะเบียน

3.3.2.4 การออกแบบสำหรับยูสเคสการสร้างคำถาม

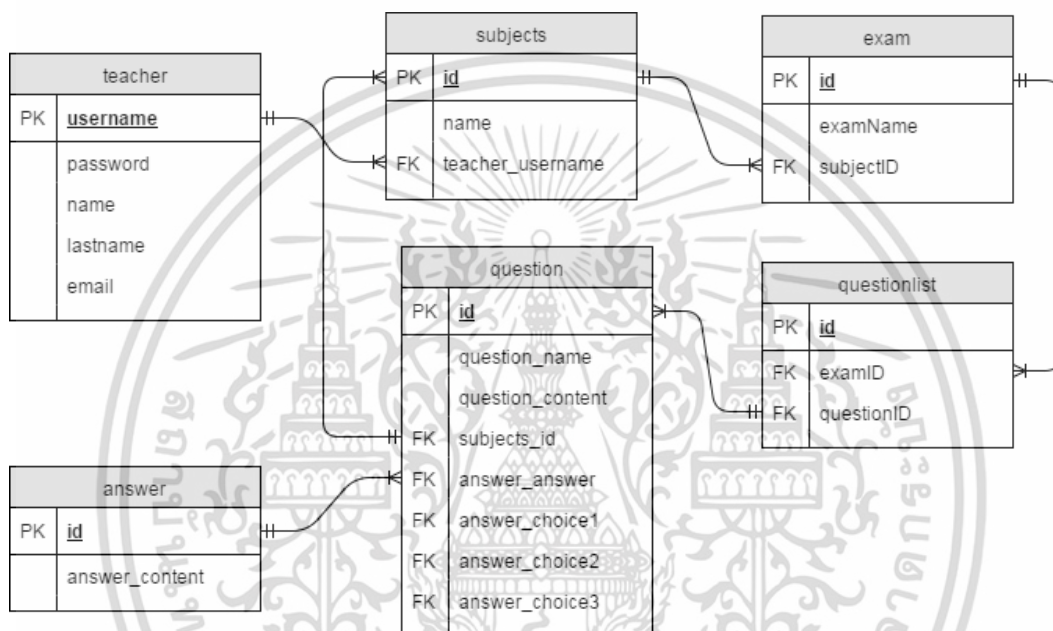
จากการวิเคราะห์แผนภาพยูสเคสการสร้างคำถาม หลังจากที่มีผู้สอนสร้างรายวิชาแล้ว ผู้สอนสามารถสร้างคำถามได้ โดยคำถามจะเป็นรูปแบบปรนัย 4 ตัวเลือก จึงได้ออกแบบตารางข้อมูล ออกแบบแผนภาพคลาส และออกแบบส่วนติดต่อผู้ใช้ ดังต่อไปนี้

1) การออกแบบตารางข้อมูลของการสร้างคำถาม

ในการสร้างคำถามในเว็บไซต์ควิสแควสโก ดังที่กล่าวในหัวข้อที่ 3.2.2 ผู้ใช้จะต้องป้อนข้อมูลคำถาม และคำตอบให้ครบ โดยคำตอบจะเป็นรูปแบบ 4 ตัวเลือก ซึ่งเมื่อผู้ใช้เพิ่มคำถาม ระบบจะตรวจสอบตัวเลือกแต่ละตัวมีอยู่ในระบบแล้วหรือไม่ ถ้ามีระบบจะนำรหัสของคำตอบมาเก็บเพื่อใช้อ้างอิงคำตอบ แต่ถ้าคำตอบนั้นไม่มีอยู่ในฐานข้อมูลระบบจะเพิ่มคำตอบนั้น

และนำรหัสคำตอบมาใช้ การแยกตารางคำตอบออกมาเพื่อที่จะได้ใช้งานร่วมกับแอปพลิเคชันฝั่งเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นักเรียน ทำให้สามารถอ้างอิงคำตอบมาแสดงได้ในหน้าดูคำถามย้อนหลังได้ จากนั้นเมื่อสร้างคำถามเสร็จแล้ว ผู้ใช้จะต้องนำคำถามไปรวมกันเป็นชุดคำถาม โดยผู้ใช้จะต้องสร้างหัวข้อแบบทดสอบของวิชาที่ต้องการ แล้วจึงเพิ่มวิชาลงในแบบทดสอบนั้นๆ จึงได้ออกแบบตาราง subjects, teacher, exam, question, answer และ questionlist เพื่อตรวจสอบข้อมูลดังกล่าว ดังตารางที่ 3.27, 3.28, 3.33, 3.36, 3.37 และ 3.38 ที่กล่าวไว้ข้างต้น และมีการแสดงรายละเอียดความสัมพันธ์ของตารางดังรูปที่ 3.34



รูปที่ 3.34 ความสัมพันธ์ของตารางการสร้างคำถาม

โครงสร้างของตาราง subjects สามารถดูรายละเอียดได้ในตารางที่ 3.22

โครงสร้างของตาราง teacher สามารถดูรายละเอียดได้ในตารางที่ 3.23

โครงสร้างของตาราง exam สามารถดูรายละเอียดได้ในตารางที่ 3.25

โครงสร้างของตาราง question สามารถดูรายละเอียดได้ในตารางที่ 3.28

โครงสร้างของตาราง answer สามารถดูรายละเอียดได้ในตารางที่ 3.29

โครงสร้างของตาราง questionlist สามารถดูรายละเอียดได้ในตารางที่ 3.30

2) การออกแบบส่วนติดต่อผู้ใช้ของการสร้างคำถาม

QuizQuestGO Home Subject LOG OUT

Create Question

Question Name

Question Content

Subject

Correct Answer

Wrong Answer

Wrong Answer

Wrong Answer

Quiz

รูปที่ 3.35 หน้าต่างแสดงการสร้างคำถาม

Quiz

Subject

ALL QUESTION

QUESTION ID	QUESTION NAME	QUESTION CONTENT	QUESTION ANSWER
32	การคูณคณนิยม	เจ้าหญิงอลิซาเบธ มีเคยมีเรื่องแยกจากเป็นเสกษนิยม	มูเตลาน
33	การคูณคณนิยม	เจ้าหญิงคณนิยมคณนิยม ที่ไม่ต้องแต่งงามก็มึงเป็นเจ้าหญิง	เอลซ่า
34	คณนิยม	เจ้าหญิงอลิซาเบธที่เสกษนิยมของนางฟ้าลงใต้	เบตส์
37	คณนิยม	เจ้าหญิงอลิซาเบธ ไซโรไนท์จิกคณาเสกษนิยม	ซินเดอเรลล่า
38	คณนิยม	คณนิยมคณนิยม นมเลียรทำถึงคณนิยมคณนิยมคณนิยม	คณนิยม
39	คณนิยม	คณนิยมคณนิยมคณนิยมคณนิยมคณนิยมคณนิยม	คณนิยมคณนิยมคณนิยม
40	คณนิยม	ซินเดอเรลล่า คณนิยมคณนิยมคณนิยมคณนิยม	จิกคณา

รูปที่ 3.36 หน้าจอแสดงคำถามที่สร้างไว้

จากรูปที่ 3.35 และรูปที่ 3.36 เป็นการออกแบบหน้าจอให้ผู้ใช้ป้อนข้อมูลเพื่อใช้สร้างคำถาม และดูคำถามที่สร้างในแต่ละวิชา โดยข้อมูลที่ใช้ในการสร้างคำถามจะเป็นข้อมูลที่จำเป็นในการสร้างคำถามแบบปรนัย 4 ตัวเลือกซึ่งประกอบไปด้วย ชื่อหัวข้อคำถาม คำถาม วิชา คำถามตอบที่ถูกต้อง และคำตอบที่ผิดอีก 3 ตัวเลือก และเมื่อสร้างคำถาม จะแสดงข้อมูลเกี่ยวกับคำถามที่จำเป็น จึงได้ออกแบบให้ข้อมูลในแต่ละแถวประกอบไปด้วยรหัสคำถาม ชื่อหัวข้อคำถาม คำถาม และคำตอบที่ถูกต้อง

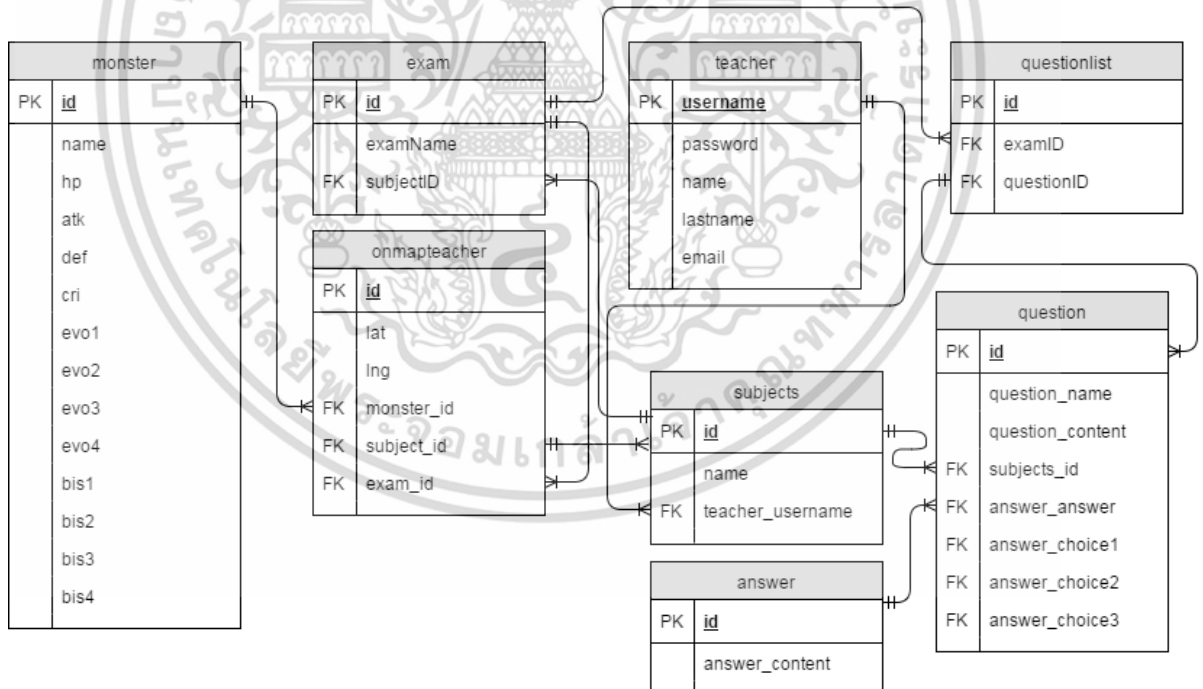
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2.5 การออกแบบสำหรับยูสเคสการกำหนดตำแหน่งของคำถาม

จากการวิเคราะห์แผนภาพยูสเคสการกำหนดตำแหน่งของคำถาม หลังจาก
ที่ผู้สอนสร้างคำถามแล้ว ผู้สอนสามารถนำคำถามที่สร้างไว้แล้วมากำหนดตำแหน่งของคำถามบนแผน
ที่ได้ จึงได้ออกแบบตารางข้อมูล ออกแบบแผนภาพคลาส และออกแบบส่วนติดต่อผู้ใช้ ดังต่อไปนี้

1) การออกแบบตารางข้อมูลของการกำหนดตำแหน่งของคำถาม

ในส่วนกำหนดตำแหน่งของคำถามในเว็บไซต์ควิสแควสโก ดังที่
กล่าวในหัวข้อที่ 3.2.2 ผู้ใช้จะต้องเลือกแบบทดสอบที่สร้างไว้มากำหนดตำแหน่งบนแผนที่โดยจะ
เก็บค่าระตืดจุดละติจูด และลองจิจูด ไว้เพื่อใช้อ้างอิงกับตำแหน่งแผนที่ในเกมควิสแควสโก จึงได้
ออกแบบตาราง user, monster, subjects, teacher, exam, onmapteacher, question, answer และ questionlist โดยรายละเอียดแต่ละตารางที่ 3.15, 3.18, 3.27, 3.28, 3.33, 3.34, 3.36, 3.37 และ 3.38 ดังที่กล่าวไว้ข้างต้น โดยแต่ละตารางมีความสัมพันธ์ดังรูปที่ 3.37



รูปที่ 3.37 ความสัมพันธ์ของตารางการกำหนดตำแหน่งของคำถามของเว็บไซต์

โครงสร้างของตาราง user สามารถดูรายละเอียดได้ในตารางที่ 3.15

โครงสร้างของตาราง monster สามารถดูรายละเอียดได้ในตารางที่ 3.18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงสร้างของตาราง subjects สามารถดูรายละเอียดได้ในตารางที่ 3.27

โครงสร้างของตาราง teacher สามารถดูรายละเอียดได้ในตารางที่ 3.28

โครงสร้างของตาราง exam สามารถดูรายละเอียดได้ในตารางที่ 3.33

โครงสร้างของตาราง onmapteacher สามารถดูรายละเอียดได้ในตารางที่ 3.34

โครงสร้างของตาราง question สามารถดูรายละเอียดได้ในตารางที่ 3.36

โครงสร้างของตาราง answer สามารถดูรายละเอียดได้ในตารางที่ 3.37

โครงสร้างของตาราง questionlist สามารถดูรายละเอียดได้ในตารางที่ 3.38

2) การออกแบบส่วนติดต่อผู้ใช้ของการกำหนดตำแหน่งของคำถาม

The screenshot shows the QuizQuestGO web application interface. At the top, there are navigation links for 'Home', 'Subject', and 'LOG OUT'. The main content area features a Google Maps satellite view of a campus. A red pin is placed on a building, with a popup window displaying the following coordinates: Latitude: 13.728509641825154 and Longitude: 100.77970147132874. Below the map, there are input fields for 'Subject' (with a dropdown menu) and 'Exam' (with a dropdown menu). A 'Random in Area' checkbox is also present. A 'SUBMIT' button is located at the bottom of the form. The footer contains copyright information for 2017 quizquestgo.com and links for 'Home', 'About', 'Contact', and 'Back to Top'.

รูปที่ 3.38 หน้าจอการกำหนดตำแหน่งของคำถามของเว็บไซต์

จากรูปที่ 3.38 เป็นการออกแบบให้ผู้ใช้กำหนดตำแหน่งที่จะปล่อยแบบทดสอบ เนื่องจากการกำหนดนั้นใช้ข้อมูลละเอียด และลองจุดซึ่งยากมากถ้ากำหนดด้วยตัวเอง จึงได้ออกแบบให้ใช้แผนที่ในการกำหนดจุด และได้ออกแบบให้สามารถค้นหาสถานที่ได้ด้วย

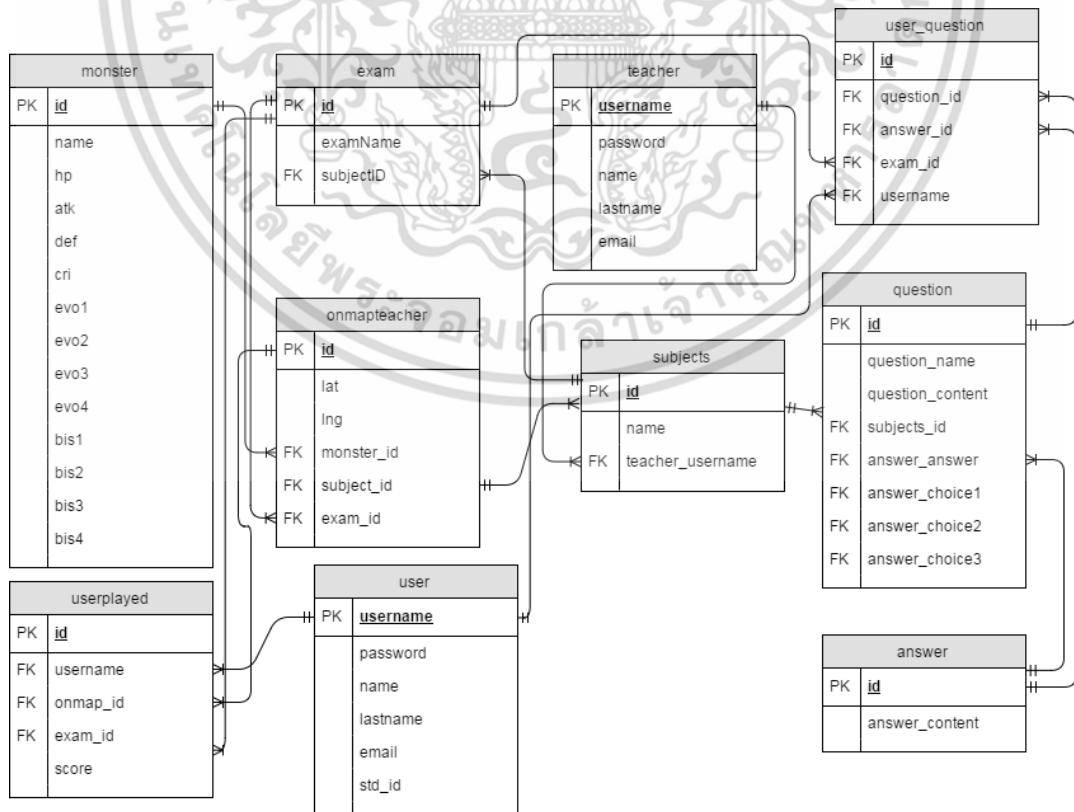
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2.6 การออกแบบสำหรับยูสเคสการตรวจสอบคะแนน

จากการวิเคราะห์แผนภาพยูสเคสการตรวจสอบคะแนน หลังจากที่มีผู้เรียนตอบคำถามที่ผู้สอนสร้างไว้แล้ว ผู้สอนสามารถตรวจสอบคะแนนของผู้เรียนได้ จึงได้ออกแบบตารางข้อมูล ออกแบบแผนภาพคลาส และออกแบบส่วนติดต่อผู้ใช้ ดังต่อไปนี้

1) การออกแบบตารางข้อมูลของการตรวจสอบคะแนน

ในส่วนตรวจสอบคะแนน ในระบบของเว็บไซต์ควิสแควสโก ดังที่กล่าวในหัวข้อที่ 3.2.2 ผู้ใช้สามารถตรวจสอบคะแนนของนักเรียนได้ โดยฐานข้อมูลจะตรวจสอบชื่อเข้าใช้งานของผู้ใช้,วิชา และแบบทดสอบ เพื่อเรียกข้อมูลผู้ใช้ฝั่งนักเรียนที่ตอบแบบทดสอบแล้วเพื่อดูคะแนน และชื่อของผู้ใช้ฝั่งนักเรียน เมื่อผู้ใช้เลือกดูเนื้อหาที่ผู้ใช้ฝั่งนักเรียนตอบ ระบบจะเรียกข้อมูลคำถาม และคำตอบของผู้ใช้ฝั่งนักเรียนคนนั้นที่ได้ทำในแบบทดสอบมาแสดง จึงได้มีการออกแบบตาราง user ,monster, subjects, teacher, exam, question, answer, onmapteacher, userquestion และ userplayed โดยรายละเอียดแต่ละตารางที่ 3.15, 3.18, 3.27, 3.28, 3.33, 3.34, 3.36, 3.37 ,3.39 และ 3.40 ดังที่กล่าวไว้ข้างต้น โดยแต่ละตารางมีความสัมพันธ์ดังรูปที่ 3.39



รูปที่ 3.39 ความสัมพันธ์ของตารางการตรวจสอบคะแนน

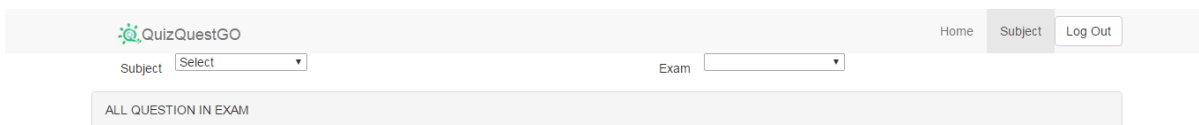
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษเท่านั้น เมื่อผู้ผู้ดูแลเห็นผู้ใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- โครงสร้างของตาราง user สามารถดูรายละเอียดได้ในตารางที่ 3.15
- โครงสร้างของตาราง monster สามารถดูรายละเอียดได้ในตารางที่ 3.18
- โครงสร้างของตาราง subjects สามารถดูรายละเอียดได้ในตารางที่ 3.27
- โครงสร้างของตาราง teacher สามารถดูรายละเอียดได้ในตารางที่ 3.28
- โครงสร้างของตาราง exam สามารถดูรายละเอียดได้ในตารางที่ 3.33
- โครงสร้างของตาราง onmapteacher สามารถดูรายละเอียดได้ในตารางที่ 3.34
- โครงสร้างของตาราง question สามารถดูรายละเอียดได้ในตารางที่ 3.36
- โครงสร้างของตาราง answer สามารถดูรายละเอียดได้ในตารางที่ 3.37
- โครงสร้างของตาราง userquestion สามารถดูรายละเอียดได้ในตารางที่ 3.39
- โครงสร้างของตาราง userplayed สามารถดูรายละเอียดได้ในตารางที่ 3.40



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) การออกแบบส่วนติดต่อผู้ใช้ของการตรวจสอบคะแนน



รูปที่ 3.40 หน้าจอเลือกวิชา และชุดคำถามที่ต้องการดูคะแนนของเว็บไซต์

Student Name	Student Lastname	Student ID	Student Score	Detail
สุวิทย์	อินทร์เม็ม	56050414	2	Open Detail
Ari	Wanda	56050121	1	Open Detail
สุชาติ	อินทร์เม็ม	56050409	2	Open Detail
สมนจัน	สุวิทย์	0	1	Open Detail
จันทรูป	ราคา	56050448	2	Open Detail
จันทรูป	ราคา	56050448	2	Open Detail
สุวิทย์		0	1	Open Detail
not		56050278	2	Open Detail

รูปที่ 3.41 หน้าจอแสดงรายละเอียดคะแนนของเว็บไซต์

จากรูปที่ 3.40 และ รูปที่ 3.41 เป็นการออกแบบหน้าจอให้แสดงข้อมูลรายงานการตอบคำถาม เนื่องจากผู้ใช้ฝั่งอาจารย์สามารถดูรายงานของการทำแบบฝึกหัดของนักเรียนซึ่งจะแสดงข้อมูลพื้นฐานของนักเรียน และการทำแบบทดสอบของนักเรียนได้ จึงได้ออกแบบให้ตารางแสดงข้อมูลดังต่อไปนี้ ชื่อ นามสกุล รหัสนักศึกษา คะแนนที่ทำได้ในแบบทดสอบ และได้ออกแบบให้ดูได้ด้วยว่านักเรียนตอบอะไร

บทที่ 4

ผลการดำเนินงาน

จากการวิเคราะห์ และออกแบบระบบ ผลลัพธ์ที่ได้จากการพัฒนาแสดงออกมาในรูปแบบของแอปพลิเคชันเกมสำหรับนักเรียน และเว็บแอปพลิเคชันสำหรับอาจารย์ ตามที่ได้วิเคราะห์ และออกแบบไว้ในบทที่ 3 ซึ่งสามารถอธิบายผลการดำเนินงานได้ดังนี้

4.1 ผลการดำเนินงาน

เมื่อนำส่วนของการออกแบบมาพัฒนาระบบงานโดยใช้โปรแกรมวิซวลสตูดิโอ (Visual Studio) ร่วมกับดอตเน็ตเฟรมเวิร์ค (.NET Framework) พัฒนาเว็บแอปพลิเคชันฝั่งอาจารย์ และโปรแกรมยูนิตีทรีดี (Unity3D) พัฒนาแอปพลิเคชันเกมฝั่งนักเรียน จากการพัฒนาแอปพลิเคชันจะได้ผลลัพธ์ซึ่งแสดงหน้าต่างการทำงานของแต่ละส่วนได้ดังนี้

4.1.1 การจัดการบัญชีผู้ใช้

4.1.1.1 การจัดการบัญชีผู้ใช้เว็บแอปพลิเคชัน

- ลงทะเบียน
- ลงชื่อเข้าใช้งาน

ในหน้าเริ่มต้นเข้าใช้งาน อาจารย์สามารถลงทะเบียน และเข้าใช้งานระบบได้ ดังรูปที่ 4.1

QuizQuestGO Home About Contact Sign In

Sign In

Username
teach1

Password
....

Remember me?

LOG IN Sign Up

© 2017 quizquestgo.com · Home · About · Contact · Back to Top

รูปที่ 4.1 หน้าจอเริ่มเข้าใช้งานเว็บแอปพลิเคชันควิสควิสโก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4.1 อาจารย์สามารถเข้าใช้งานระบบได้ ดังนี้

1) การเริ่มเข้าใช้งานระบบ

เมื่อเข้าสู่เว็บแอปพลิเคชันอาจารย์สามารถลงทะเบียน โดยการป้อนชื่อผู้ใช้งาน รหัสผ่าน ยืนยันรหัสผ่าน ชื่อ นามสกุล และอีเมล หากชื่อผู้เข้าใช้งานมีอยู่แล้ว จะไม่สามารถลงทะเบียนได้ และจะมีข้อความแสดงเพื่อแจ้งเตือน ดังรูปที่ 4.2

รูปที่ 4.2 หน้าจอป้อนชื่อเข้าใช้งานที่มีอยู่แล้วในระบบ

เมื่อเข้าสู่เว็บแอปพลิเคชัน อาจารย์สามารถเข้าสู่ระบบโดยการป้อนชื่อผู้ใช้งาน และรหัสผ่าน ดังรูปที่ 4.1 ที่ได้กล่าวไว้ข้างต้น

หากอาจารย์เคยลงทะเบียน และเข้าสู่ระบบได้แล้ว จะเข้าสู่หน้าหลักของเว็บแอปพลิเคชัน และแสดงข้อความเข้าสู่ระบบสำเร็จดังรูปที่ 4.3

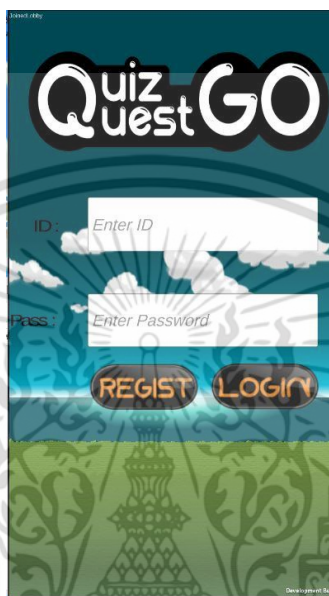
รูปที่ 4.3 หน้าหลักของเว็บแอปพลิเคชันที่เข้าสู่ระบบแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.1.2 การจัดการบัญชีผู้ใช้งานแอปพลิเคชันเกม

- ลงทะเบียน
- ลงชื่อเข้าใช้งาน

ในหน้าเริ่มต้นเข้าใช้งาน นักเรียนสามารถลงทะเบียน และเข้าใช้งานระบบได้ ดังรูปที่ 4.4



รูปที่ 4.4 หน้าจอเริ่มเข้าใช้งานแอปพลิเคชันเกมควิสควอสโก

จากรูปที่ 4.4 นักเรียนสามารถเข้าใช้งานระบบได้ ดังนี้

1) การเริ่มเข้าใช้งานระบบ

เมื่อเข้าสู่แอปพลิเคชันเกม นักเรียนสามารถลงทะเบียน โดยการป้อนชื่อผู้ใช้ รหัสผ่าน ชื่อ นามสกุล รหัสนักเรียน และอีเมล หากชื่อผู้ใช้งานมีอยู่แล้ว จะไม่สามารถลงทะเบียนได้ และกล่องรับข้อมูลของชื่อผู้ใช้จะเปลี่ยนเป็นสีแดงเพื่อแจ้งเตือน ดังรูปที่ 4.5

ID:
 Pass:
 Firstname:
 Lastname:
 StudentID:
 Email:

รูปที่ 4.5 หน้าจอแจ้งเตือน เมื่อชื่อผู้ใช้งานมีอยู่แล้วในระบบ

เมื่อเข้าสู่แอปพลิเคชันเกม นักเรียนสามารถเข้าสู่ระบบโดยการป้อนชื่อผู้ใช้ และรหัสผ่าน ดังรูปที่ 4.4 ที่กล่าวไว้ข้างต้น

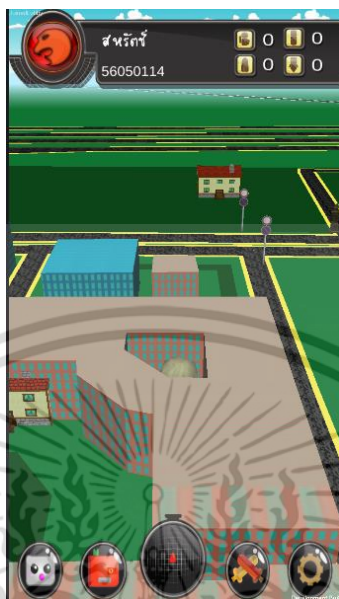
เมื่อนักเรียนป้อนชื่อผู้ใช้งาน หรือรหัสผ่านผิด จะไม่สามารถเข้าสู่ระบบได้ และกล่องรับข้อมูลของชื่อผู้ใช้ และรหัสผ่านจะเปลี่ยนเป็นสีแดงเพื่อแจ้งเตือน ดังรูปที่ 4.6

Quiz GO
 ID:
 Pass:

รูปที่ 4.6 หน้าจอแจ้งเตือนเมื่อป้อนชื่อผู้ใช้งาน หรือรหัสผ่านไม่ถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเข้าสู่ระบบสำเร็จจะเข้าสู่หน้าหลักของแอปพลิเคชัน และแสดงข้อมูลของผู้ใช้งานที่เก็บไว้บนฐานข้อมูลได้แก่ ชื่อ รหัสนักศึกษา และจำนวนบิสกิต ดังรูปที่ 4.7



รูปที่ 4.7 หน้าหลักของแอปพลิเคชันเกมควิสเคสโก

4.1.2 แอปพลิเคชัน

จะแบ่งออกเป็น 2 ส่วน ได้แก่

ฝั่งเว็บแอปพลิเคชัน(อาจารย์)

- สามารถสร้างรายวิชา
- สามารถสร้างคำถาม
- สามารถสร้างแบบทดสอบ
- สามารถกำหนดตำแหน่งของคำถาม
- สามารถดูผลลัพธ์ของนักเรียนแล้วนำมาสรุปผลการเรียนได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฝั่งแอปพลิเคชันเกม(นักเรียน)

- สามารถเพิ่มรายวิชา
- สามารถค้นหาคำถาม
- สามารถตอบคำถาม
- สามารถตรวจสอบสถานะมอนสเตอร์
- สามารถต่อสู้อุปกรณ์ประลอง
- สามารถดูผลลัพธ์จากการตอบคำถามได้

1) การลงทะเบียน

อาจารย์สามารถสร้างรายวิชาของตัวเอง ในหน้าจะมีด้วยกันสองส่วนคือ ส่วนของการสร้างรายวิชา (Create Subject) ดังรูปที่ 4.8 ซึ่งเมื่อเพิ่มรายวิชาใหม่เข้าไป จะถูกนำไปแสดงในส่วนที่สองคือ ส่วนของการจัดการรายวิชา (Manage Subject) ซึ่งแสดงรหัสวิชา และชื่อวิชาที่สร้างไว้แล้ว ดังรูปที่ 4.9

ในตัวอย่างจากรูปที่ 4.8 และรูปที่ 4.9 อาจารย์ได้สร้างวิชา Java Programming ขึ้นมา

QuizQuestGO Home Subject LOG OUT

Create Subject

Subject Name

Manage Subject

SUBJECT	
ID PIN	SUBJECT NAME

© 2017 quizquestgo.com · Home · About · Contact · [Back to Top](#)

รูปที่ 4.8 หน้าจอสร้างวิชา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

QuizQuestGO Home Subject LOG OUT

Create Subject

Subject Name

Registration Success

Manage Subject

SUBJECT	
ID PIN	SUBJECT NAME
33	Java Programming

รูปที่ 4.9 หน้าจอเมื่อสร้างวิชาสำเร็จ

เมื่ออาจารย์นักเรียนสามารถเพิ่มรายวิชาได้ด้วยรหัสวิชาที่ได้รับมาจากอาจารย์โดยป้อนรหัสลงในกล่องรับข้อมูลด้านล่าง ดังรูปที่ 4.10 เมื่อลงทะเบียนสำเร็จ จะแสดงวิชาที่นักเรียนได้นำรหัสวิชามาลงทะเบียน ดังรูปที่ 4.11 โดยนักเรียนจะสามารถดูคะแนน และดูคำถาม คำตอบย้อนหลังในแต่ละแบบทดสอบของวิชาที่ลงทะเบียนได้ ซึ่งจะอธิบายเพิ่มเติมในหัวข้อที่ 4.1.2.5

ในตัวอย่างจากรูปที่ 4.10 และรูปที่ 4.11 นักเรียนได้นำรหัสวิชา Java Programming จากอาจารย์มาลงทะเบียนในรายวิชาเรียน



รูปที่ 4.10 หน้าจอใส่รหัสวิชาที่ต้องการลงทะเบียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 หน้าจอเมื่อเพิ่มรายวิชาแล้ว

2) การเพิ่มคำถามและตอบคำถาม

เมื่ออาจารย์สร้างรายวิชาแล้ว อาจารย์สามารถสร้างคำถามได้ในหน้าจัดการคำถาม ในหน้านี้จะมีด้วยกันสองส่วนคือ ส่วนสร้างคำถาม (Create Question) ซึ่งอาจารย์สามารถสร้างคำถามในรายวิชาได้ โดยลักษณะของคำถามจะเป็นตัวเลือก 4 ตัวเลือก ดังรูปที่ 4.12 เมื่อสร้างคำถามแล้วจะแสดงคำถาม และคำตอบในส่วนที่สองคือ ส่วนคำถาม (Quiz) ซึ่งจะแสดงคำถามที่สร้างไว้ในรายวิชาที่ผู้ใช้เลือก ดังรูปที่ 4.13

ในตัวอย่างดังรูปที่ 4.12 อาจารย์ได้สร้างคำถามชื่อ “ตัวเลขจำนวนเต็ม” และถามคำถามว่า “ประเภทตัวแปรใดที่ใช้ในการเก็บเลขจำนวนเต็ม” ในรายวิชา Java Programming และสร้างตัวเลือกคำตอบ 4 ตัวเลือก คือ 1.int 2.String 3.char และ 4.ไม่มีข้อใดถูก

QuizQuestGO Home Subject LOG OUT

Create Question

Question Name

Question Content

Subject

Correct Answer

Wrong Answer

Wrong Answer

Wrong Answer

รูปที่ 4.12 หน้าจอสร้างคำถาม

QuizQuestGO Home Subject LOG OUT

Create Question

Question Name

Question Content

Subject

Correct Answer

Wrong Answer

Wrong Answer

Wrong Answer

Registration Success

Quiz

Subject

QUESTION ID	QUESTION NAME	QUESTION CONTENT	QUESTION ANSWER
188	ตัวเลขจำนวนเต็ม	ประเภทตัวแปรใดที่ใช้ในการเก็บเลขจำนวนเต็ม	int

รูปที่ 4.13 หน้าจอแสดงผลลัพธ์เมื่อสร้างคำถามเสร็จ

เมื่ออาจารย์สร้างคำถามเสร็จแล้วต้องการนำคำถามมารวมกันเป็นแบบทดสอบก่อนที่จะไปกำหนดตำแหน่งบนแผนที่ อาจารย์สามารถมาสร้างแบบทดสอบได้ในหน้าจัดการแบบทดสอบ ซึ่งจะแบ่งออกเป็นสองส่วนคือ ส่วนสร้างแบบทดสอบ (Create Exam) เมื่อสร้างแบบทดสอบแล้วจะสามารถเลือกจัดการคำถามของแบบทดสอบ และกำหนดตำแหน่งบนแผนที่เพื่อใช้ในการเล่นบนแอปพลิเคชันเกมได้ ในส่วนที่สองคือ ส่วนจัดการแบบทดสอบ (Manage Exam) ซึ่งสามารถนำคำถามที่สร้างเก็บไว้ไปเพิ่มในแบบทดสอบ โดยเมื่อเลือกวิชา และแบบทดสอบ จะมีคำถามที่สร้างไว้แสดงขึ้นที่ตารางคำถามทั้งหมดในรายวิชา (ALL QUESTION IN SUBJECT) ดังรูปที่ 4.14 เมื่อเพิ่มคำถามในรายวิชาลงในแบบทดสอบจะแสดงคำถามนั้นในตาราง คำถามทั้งหมดในแบบทดสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เพื่อการวิจัยเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้เผยแพร่ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(ALL QUESTION IN EXAM) ด้านบน ดังรูปที่ 4.15

ในตัวอย่างรูปที่ 4.14 อาจารย์เลือกคำถามที่สร้างไว้ในรายวิชา Java Programming เพิ่มไปยังแบบทดสอบ Exam 1

MANAGE EXAM

Subject Java Programming Exam Exam1

[Spawn Exam To Real World](#)

ALL QUESTION IN EXAM			
QUESTION NAME	QUESTION CONTENT	QUESTION ANSWER	DELETE
ตัวเลขจำนวนเต็ม	ประเภทตัวแปรใดที่ใช้ในการเก็บเลขจำนวนเต็ม	int	Button
ตัวเลขจำนวนเต็ม	ถ้าเราใส่ค่า 4.25 เก็บในตัวแปรประเภท int ตัวแปรนั้นจะมีค่าเป็นเท่าไร	4	Button
ตัวเลขจำนวนเต็ม	int เป็นประเภทตัวแปรที่มีขนาดเท่าไร	32	Button
ตัวเลขจำนวนเต็ม	ค่าที่น้อยที่สุดของ int คือเท่าไร	-2,147,483,648	Button
คำสั่งวน	คำสั่งต่อไปนี้ ไม่ใช่คำสั่งวน	stop	Button
คำสั่งวน	คำสั่งต่อไปนี้ ไม่ใช่คำสั่งวน	far	Button
คำสั่งวน	คำสั่งต่อไปนี้ ไม่ใช่คำสั่งวน	cat	Button

รูปที่ 4.14 หน้าจอการเพิ่มคำถามลงในแบบทดสอบ

MANAGE EXAM

Subject Java Programming Exam Exam1

[Spawn Exam To Real World](#)

ALL QUESTION IN EXAM			
QUESTION NAME	QUESTION CONTENT	QUESTION ANSWER	DELETE
ตัวเลขจำนวนเต็ม	ประเภทตัวแปรใดที่ใช้ในการเก็บเลขจำนวนเต็ม	int	Button
ตัวเลขจำนวนเต็ม	ถ้าเราใส่ค่า 4.25 เก็บในตัวแปรประเภท int ตัวแปรนั้นจะมีค่าเป็นเท่าไร	4	Button
ตัวเลขจำนวนเต็ม	int เป็นประเภทตัวแปรที่มีขนาดเท่าไร	32	Button
ตัวเลขจำนวนเต็ม	ค่าที่น้อยที่สุดของ int คือเท่าไร	-2,147,483,648	Button
คำสั่งวน	คำสั่งต่อไปนี้ ไม่ใช่คำสั่งวน	stop	Button
คำสั่งวน	คำสั่งต่อไปนี้ ไม่ใช่คำสั่งวน	far	Button
คำสั่งวน	คำสั่งต่อไปนี้ ไม่ใช่คำสั่งวน	cat	Button
คำสั่งวน	คำสั่งต่อไปนี้ ไม่ใช่คำสั่งวน	troll	Button

ALL QUESTION IN Subject

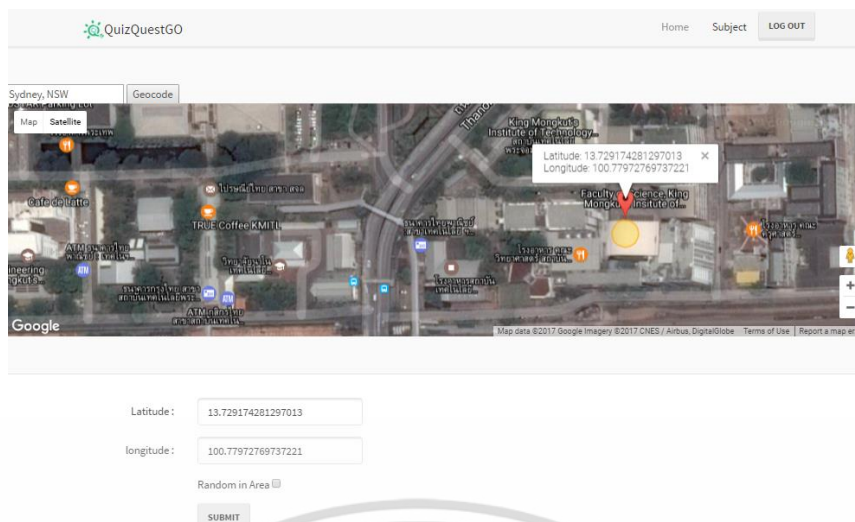
QUESTION NAME	QUESTION CONTENT	QUESTION ANSWER	ADD QUIZ
---------------	------------------	-----------------	----------

รูปที่ 4.15 หน้าจอเพิ่มคำถามลงในแบบทดสอบครบแล้ว

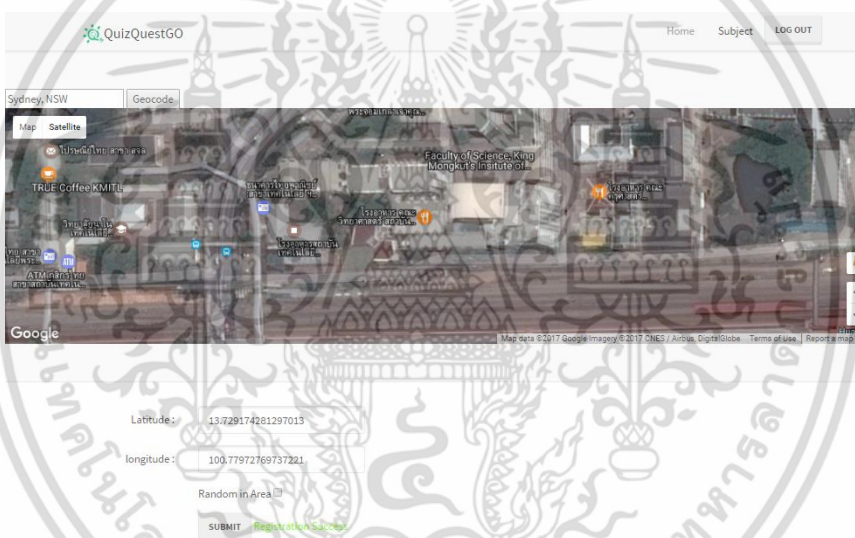
เมื่ออาจารย์สร้างแบบทดสอบแล้ว อาจารย์สามารถนำแบบทดสอบที่สร้างไว้แล้วไปกำหนดตำแหน่งของคำถามบนแผนที่เพื่อใช้เล่นบนแอปพลิเคชันเกมได้ ซึ่งเมื่อกดที่แผนที่จะมีวงกลมสีเหลืองแสดงขอบเขตที่นักเรียนสามารถพบแบบทดสอบที่กำหนดไว้ได้ ดังรูปที่ 4.12

ในตัวอย่างจากรูปที่ 4.16 อาจารย์นำแบบทดสอบ Exam 1 มาเพิ่มไว้ที่อาคารพระจอมเกล้าสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.16 หน้าจอกำหนดตำแหน่งของคำถามบนแผนที่



รูปที่ 4.17 หน้าจอแจ้งเตือนเมื่อกำหนดตำแหน่งเสร็จ

เมื่อนักเรียนลงทะเบียนรายวิชาเรียนแล้ว นักเรียนสามารถค้นหาคำถามได้เฉพาะรายวิชาที่ลงทะเบียนไว้ โดยลักษณะเมื่อเจอคำถามแล้วจะเป็นรูปมอนสเตอร์ ดังรูปที่ 4.18 และหากนักเรียนยังไม่ได้ลงทะเบียนรายวิชา นักเรียนก็จะไม่พบคำถาม ดังรูปที่ 4.19

ในตัวอย่างจากรูปที่ 4.18 นักเรียนได้ลงทะเบียนวิชา Java Programming และได้ค้นหาคำถามที่อาคารพระจอมเกล้า สถาบันเทคโนโลยีพระจอมเกล้าลาดกระบัง และเจอเป็นรูปมอนสเตอร์ ในบริเวณดังกล่าว 2 ตัว เนื่องจากอาจารย์เพิ่มแบบทดสอบ 2 ชุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.18 หน้าจอกดปุ่มค้นหาคำถาม เมื่อลงทะเบียนรายวิชาแล้ว



รูปที่ 4.19 หน้าจอเมื่อกดปุ่มค้นหาคำถาม เมื่อยังไม่ได้ลงทะเบียนรายวิชา

เมื่อนักเรียนกดที่รูปมอนเตอร์จะเข้าสู่หน้าตอบคำถามของแบบทดสอบ Exam1 ในรายวิชา Java Programming โดยจะมีคำถามอยู่ด้านบน และตัวเลือกจะอยู่ด้านล่างดังรูปที่ 4.20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.20 หน้าจอคำถาม

ในการเล่นเก้มตอบคำถามจะเป็นในลักษณะสัตว์เลี้ยง และมอนสเตอร์ต่อสู้กัน มีการแสดงพลังค่าชีวิตของสัตว์เลี้ยง และมอนสเตอร์เพื่อบ่งบอกการสิ้นสุดของเก้ม เมื่อตอบผิดปุ่มตัวเลือกจะเป็นสีแดง และระบบจะลดค่าพลังชีวิตของนักเรียน เมื่อตอบคำถามถูกปุ่มตัวเลือกจะเป็นสีเขียว และระบบจะลดค่าพลังชีวิตของมอนสเตอร์ ดังรูปที่ 4.21 และ 4.22



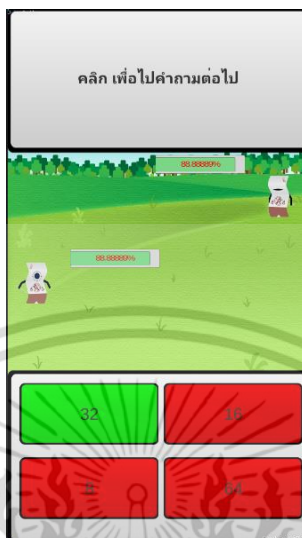
รูปที่ 4.21 แสดงผลลัพธ์เมื่อตอบคำถามถูกต้อง



รูปที่ 4.22 แสดงผลลัพธ์เมื่อตอบคำถามผิด

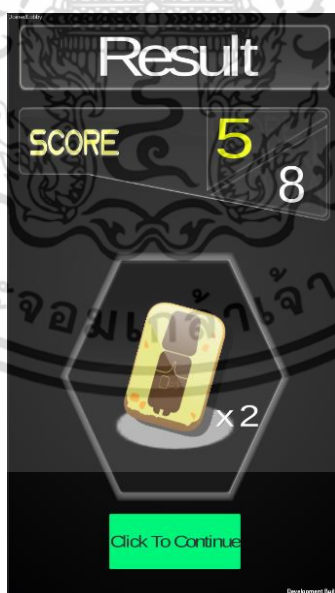
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อไม่ตอบคำถามตามเวลาที่กำหนด ระบบจะเฉลยข้อที่ถูกต้อง และลดค่าพลังชีวิตของสัตว์เลี้ยงของนักเรียน ดังรูปที่ 4.23



รูปที่ 4.23 แสดงผลลัพธ์เมื่อไม่ตอบคำถามตามเวลาที่กำหนด

เมื่อเสร็จจากการตอบแบบทดสอบแล้ว จะแสดงคะแนน และสุ่มรางวัลซึ่งเป็นบิสกิตประเภทต่างๆ ดังรูปที่ 4.24



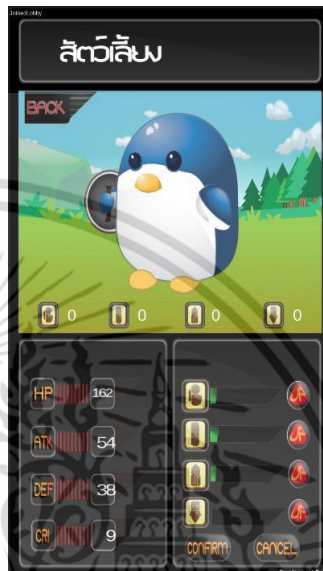
รูปที่ 4.24 หน้าจอแสดงรางวัล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อนักเรียนได้บิสกิตจากการตอบคำถามแล้ว นักเรียนสามารถนำมาเพิ่มให้กับมอนสเตอร์ประจำตัวได้ ในหน้านี้จะแสดงภาพสัตว์เลี้ยงที่ได้เลือกเมื่อตอนสมัคร จำนวนบิสกิต และค่าสถานะของสัตว์เลี้ยง ดังรูปที่ 4.25 ซึ่งเมื่อเราให้บิสกิตแล้วค่าสถานะทางซ้ายจะเปลี่ยนไปตามที่ให้ และค่าหลอดบิสกิตทางขวาจะเพิ่มขึ้นตามจำนวนบิสกิตที่เราให้กับสัตว์เลี้ยง ดังรูปที่ 4.26



รูปที่ 4.25 หน้าการจัดการมอนสเตอร์

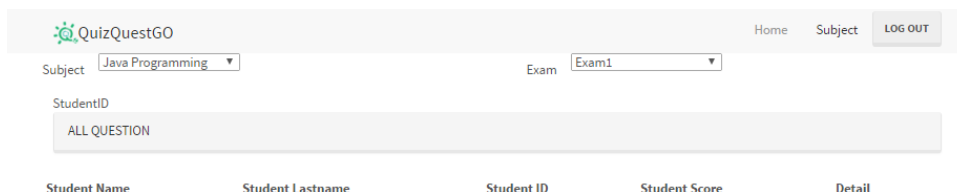


รูปที่ 4.26 หน้าจอเมื่อเพิ่มค่าสถานะ

3) การตรวจสอบคะแนน

ถ้าอาจารย์ต้องการจะดูคะแนนของนักเรียน อาจารย์เลือกรายวิชา และแบบทดสอบ เพื่อดูผลลัพธ์ของนักเรียนได้ ดังรูปที่ 4.27 ระบบจะแสดงรายละเอียดของผลลัพธ์ต่างๆ ดังรูปที่ 4.28 และสามารถดูผลลัพธ์รายบุคคลได้ ดังรูปที่ 4.29

ในตัวอย่างจากรูปที่ 4.27 อาจารย์เลือกรายวิชา Java Programming และเลือกดูคะแนนของแบบทดสอบ Exam 1 ซึ่งแสดงผลออกมาว่ามีนักเรียนชื่อ สุดารัตน์ ขำอรร้อย รหัส 56050432 ได้คะแนนในแบบทดสอบนี้ 5 คะแนน ดังรูปที่ 4.28 อาจารย์เลือกดูผลลัพธ์รายบุคคลของ สุดารัตน์ ขำอรร้อย ระบบแสดงคำถาม และคำตอบทั้งหมดที่นักเรียนทำได้ ดังรูปที่ 4.29



รูปที่ 4.27 หน้ารายงานผล

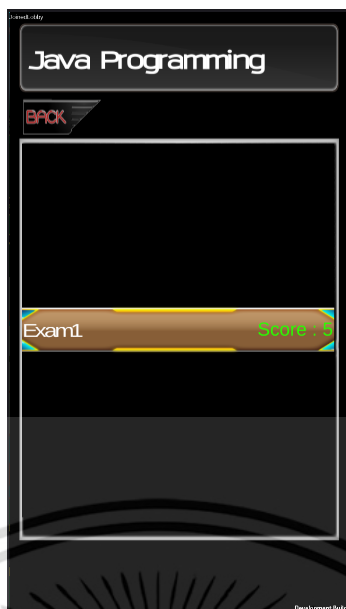


รูปที่ 4.28 หน้าจอแสดงรายละเอียดของรายวิชา และแบบทดสอบที่เลือก

รูปที่ 4.29 หน้าแสดงผลลัพธ์รายบุคคล

หลังจากเล่นเกมตอบคำถามแล้ว หากนักเรียนต้องการจะดูคะแนนที่ได้ตอบคำถามไปแล้ว นักเรียนสามารถดูคะแนนได้ที่รายวิชาที่ลงทะเบียนไว้แล้วเท่านั้น ดังรูปที่ 4.19 ที่กล่าวไว้ข้างต้น โดยระบบจะแสดงชื่อแบบทดสอบ และคะแนนที่ได้จากการตอบคำถามในแบบทดสอบนั้น ดังรูปที่ 4.30 ในตัวอย่างจากรูปที่ 4.30 นักเรียนเข้าไปดูคะแนนของแบบทดสอบ Exam1 ในรายวิชา Java Programming

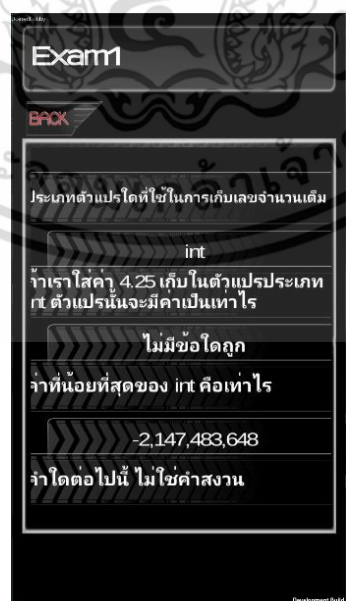
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.30 หน้าแสดงคะแนนของแต่ละแบบทดสอบ

ในหน้านี้นักเรียนสามารถดูคำถาม คำตอบย้อนหลังในแต่ละแบบทดสอบที่เคยตอบคำถามไปแล้วได้ โดยจะแสดงแค่คำถาม และคำตอบ แต่จะไม่บอกว่าข้อใดถูกหรือผิด ดังรูปที่ 4.31

ในตัวอย่างจากรูปที่ 4.31 นักเรียนเข้าไปดูคำถาม และคำตอบของแบบทดสอบ Exam 1 ในรายวิชา Java Programming ซึ่งคำถาม และคำตอบที่นักเรียนได้ทำไว้ในแบบทดสอบ Exam 1 ทั้งหมดจะปรากฏบนหน้าจอ



รูปที่ 4.31 หน้าแสดงคำถาม และคำตอบของแบบทดสอบที่เคยทำแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

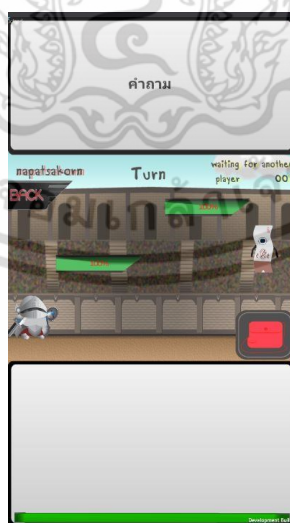
4.การต่อสู้อาณาประลอง

ถ้านักเรียนต้องการนำมอนสเตอร์ประจำตัวไปต่อสู้กับมอนสเตอร์ประจำตัวของคนอื่น นักเรียนสามารถเลือกเข้าสู่ลานประลองได้ โดยจะแบ่งออกเป็น 2 ส่วน ได้แก่ 1.สร้างห้องประลอง 2.เข้าร่วมห้องประลอง ดังรูปที่ 4.32



รูปที่ 4.32 หน้าลานประลอง

ส่วนที่ 1 สร้างห้องประลอง นักเรียนสามารถเลือกสร้างห้อง แล้วเข้ามารอผู้เล่นอื่นเข้ามาร่วมห้อง เพื่อเริ่มการแข่งขัน ดังรูปที่ 4.33

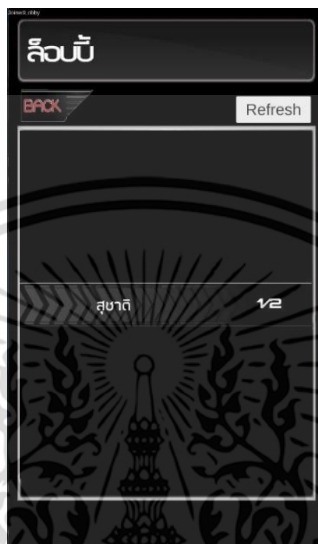


รูปที่ 4.33 หน้าสร้างห้องประลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

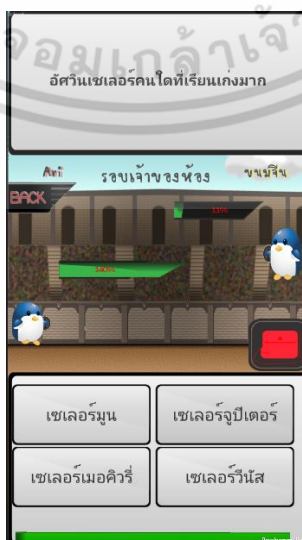
ส่วนที่ 2 เข้าร่วมห้องประลอง นักเรียนสามารถเลือกเข้าร่วมห้องที่มีอยู่แล้วในล็อบบี้ได้ ระบบ จะแสดงชื่อผู้สร้างห้อง และจำนวนคนที่อยู่ในห้อง ดังรูปที่ 4.34

ในตัวอย่างจากรูปที่ 4.34 นักเรียนเข้ามาหาห้องลานประลองในล็อบบี้ แล้วเจอห้อง “สุชาติ” มีจำนวนผู้เล่น 1 คน หมายความว่าห้องนี้ยังไม่เต็ม สามารถเข้าไปเล่นได้



รูปที่ 4.34 หน้าล็อบบี้

เมื่อผู้สร้างห้องกับผู้เข้าร่วมเข้ามาอยู่ในห้องประลองเดียวกันแล้ว จากรูปที่ 4.33 ที่กล่าวไว้ข้างต้น จะแสดงคำถามเพื่อเริ่มการแข่งขัน ดังรูปที่ 4.35 และเมื่อชนะจะได้รับรางวัลเป็นบิสกิตเช่นเดียวกับการเล่นตอบคำถามปกติ



รูปที่ 4.35 หน้าจอการต่อสู้อานประลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้วงเวียนเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การแข่งขันพัฒนาโปรแกรมคอมพิวเตอร์แห่งประเทศไทยครั้งที่ 19

ทางผู้จัดทำได้มีโอกาสนำโปรแกรมไปนำเสนอในงานแข่งขันพัฒนาโปรแกรมคอมพิวเตอร์แห่งประเทศไทยครั้งที่ 19 และได้ผ่านเข้ารอบคัดเลือก, รอบรองชนะเลิศ และเข้ารอบชิงชนะเลิศ โดยมีข้อติชม และข้อเสนอแนะดังต่อไปนี้

4.2.1 รอบคัดเลือก

1) จุดเด่นของโปรแกรม

- 1.ประยุกต์ใช้งานระบบนำทางGPSกับแอปพลิเคชันได้ดี
- 2.มีประโยชน์ในการใช้งาน

2) จุดด้อยของโปรแกรม

- 1.วิธีการเล่นเข้าใจได้ยาก
- 2.ส่วนติดต่อกับผู้ใช้ออกแบบได้น่าสนใจน้อยเกินไป

4.2.2 รอบรองชิงชนะเลิศ

1) จุดเด่นของโปรแกรม

- 1.มีความแปลกใหม่เนื่องจากตัวเกมได้มีการออกไปเล่นบนสถานที่จริง ไม่เหมือนเกมทั่วไปที่เล่นอยู่กับที่

2) จุดด้อยของโปรแกรม

- 1.ลานประลองควรมีการร่วมตอบหลายประเภทห้องเล่นควรรับรองทั้ง private และ public
- 2.มีการใช้แผนที่น้อยเกินไป ทำให้แผนที่ดูเป็นแค่พื้นหลังของคำถามเท่านั้น
- 3.ควรมีการตรวจสอบผู้ปล่อยคำถาม เพราะอาจเกิดการแฝงตัว และปล่อยคำถามในทางไม่ดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปผลการดำเนินงาน

5.1 สรุปผลการดำเนินงาน

แอปพลิเคชันควิสแควสโก เกมตอบคำถามแบบอิงตำแหน่งที่พัฒนาขึ้นนี้ สามารถใช้เป็นเครื่องมือในการสนับสนุนการเรียนการสอนที่มีกิจกรรมทั้งภายใน และภายนอกห้องเรียน โดยแอปพลิเคชันจะช่วยเหลืออาจารย์ในการใช้เป็นเครื่องมือในการกระตุ้นนักศึกษา และเป็นเครื่องมือช่วยให้การเรียนการสอนมีความน่าสนใจมากขึ้น แอปพลิเคชันนี้พัฒนาขึ้นโดยนำแนวคิดของการนำจีพีเอสมาใช้ในเกมของเกม PokemonGo และแนวคิดของการตอบคำถามแบบออนไลน์ของโปรแกรม Socrative มาใช้ โดยอาจารย์จะสร้างคำถาม และส่งคำถามไปยังตำแหน่งจริงทางภูมิศาสตร์ เพื่อให้ผู้เรียนไปทำกิจกรรมนอกห้องเรียน คำตอบของนักเรียนจะถูกส่งกลับมาในระบบเพื่อให้อาจารย์สามารถประเมินผลการเรียนรู้ของนักเรียนได้ ในส่วนของนักเรียนเนื่องจากโปรแกรมอยู่ในรูปแบบเกม และเป็นการทำกิจกรรมนอกสถานที่ดังนั้นจะกระตุ้นความสนใจได้มากขึ้น นอกจากนี้ยังมีระบบลานประลองที่ทำงานโดยให้นักเรียนนำคำถามที่ตัวเองตอบไปแล้วมาถามกันเองเพื่อแข่งขันกัน ซึ่งจะกระตุ้นให้นักเรียนได้ทบทวนความรู้ที่ได้เรียนไปอย่างสม่ำเสมอ แอปพลิเคชันนี้ยังสามารถนำไปประยุกต์ใช้กับกิจกรรมตอบคำถามประเภทอื่น ๆ ได้เช่นกิจกรรมทัศนศึกษาของโรงเรียน หรือการโปรโมทแหล่งท่องเที่ยวที่สำคัญของจังหวัดต่างๆ

และทางผู้จัดทำได้มีโอกาสนำโปรแกรมไปนำเสนอในงานแข่งขันพัฒนาโปรแกรมคอมพิวเตอร์แห่งประเทศไทยครั้งที่ 19 แล้วได้เข้ารอบชิงชนะเลิศ ได้ความเห็นจากกรรมการ และผู้เข้าร่วมงานเกี่ยวกับโปรแกรมว่า เป็นโปรแกรมที่มีความแปลกใหม่ น่าสนใจ รวมถึงสามารถสร้างแรงจูงใจได้มากโดยเฉพาะการที่มีของรางวัลให้อย่างเช่น การเข้าร่วมกับการศึกษาที่มีการให้คะแนนตอบแทน หรือการที่ห้างร้านนำมาใช้เพื่อโปรโมท และมีของรางวัลให้ และยังได้รับคำชมว่าเป็นโปรแกรมที่สามารถพัฒนาต่อยอดสู่การใช้งานจริงได้

5.2 อุปสรรคในการดำเนินงาน

เนื่องจากโปรแกรมนี้ต้องมีการเก็บข้อมูล และมีการใช้ข้อมูลร่วมกัน จึงจำเป็นต้องใช้เซิร์ฟเวอร์ที่มีประสิทธิภาพเพื่อให้รองรับการใช้งานของผู้ใช้จำนวนมาก ซึ่งต้องใช้ค่าใช้จ่ายราคาสูงในการหาเซิร์ฟเวอร์ที่มีประสิทธิภาพมาใช้ แต่เนื่องจากทางผู้พัฒนาไม่มีงบประมาณในส่วนนี้ ทำให้ต้องใช้งานเซิร์ฟเวอร์ที่ไม่เสียค่าใช้จ่าย ทำให้ระบบไม่สามารถทำงานได้เต็มที่ รองรับผู้เล่นได้ไม่เกิน 20 คนในระบบลานประลองซึ่งต้องใช้เซิร์ฟเวอร์ในการติดต่อ

5.3 ข้อเสนอแนะ

จากการศึกษา และพัฒนาระบบที่ผ่านมา รวมถึงการรับความเห็นจากคณะกรรมการ และผู้ใช้งานในงานแข่งขันพัฒนาโปรแกรมคอมพิวเตอร์แห่งประเทศไทยครั้งที่ 19 ทางผู้พัฒนาจึงนำความเห็นมาปรับปรุงแก้ไข

1. ควรพัฒนาระบบเซิร์ฟเวอร์ของลานประลองให้สามารถรองรับได้ทั้งห้องส่วนตัว และห้องสาธารณะ
2. ควรแยกประเภทคำถาม เป็น ง่าย กลาง ยาก เพื่อความเหมาะสมของผู้ใช้งาน
3. มอนสเตอร์ควรมีความแปลกใหม่ และหลากหลายมากขึ้น เพื่อทำให้เกมน่าสนใจมากขึ้น
4. การสมัครเว็บไซต์ควรมีการตรวจสอบข้อมูลผู้ใช้เพื่อป้องกันบุคคลทั่วไปที่นำไปใช้ในทางที่ไม่ดี เช่น การกำหนดตำแหน่งของคำถามว่าร้ายผู้อื่นบริเวณบ้านของผู้นั้น
5. ควรมีคำถามมากกว่า 1 ประเภท เช่น คำถามปลายเปิด
6. ควรมีการจำกัดคนตอบคำถาม เช่น ในรายวิชา java programming ให้มีคนตอบได้ 20 คน

เอกสารอ้างอิง

[1] 2016. **Pokemon Go**. [Online]

Available : <http://www.pokemongo.com>

[2] 2017. **Socartive**. [Online]

Available : <https://www.socrative.com>

[3] **OpenStreetMap**. [Online]

Available : <https://www.openstreetmap.org/#map=5/51.500/-0.100>

[4] 2017. **LocationService.Start**. [Online]

Available : <https://docs.unity3d.com/ScriptReference/LocationService.Start.html>



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้