

ระบบจัดการตารางเวลา: กรณีศึกษาภาควิชาคอมพิวเตอร์
โรงเรียนนายเรืออากาศนวมินทกษัตริยาธิราช

Scheduling Management System: A Case Study of department of Computer NKRAFA

^{1*}พนิดา แก้วชนิด ^{2#}ภูมินันท์ บัวงาม

^{1,2}ภาควิชาคอมพิวเตอร์ กองวิชาคณิตศาสตร์และคอมพิวเตอร์
กองการศึกษา โรงเรียนนายเรืออากาศนวมินทกษัตริยาธิราช

^{1*}Panida Kaewchanid and ^{2#}Phuminun Bua-ngam

^{1,2}Department of Computer Science, Navaminda Kasatriyadhiraj Royal Air Force Academy

*panida_ka@rtaf.mi.th, #phuminun@rtaf.mi.th

Received : June 28, 2022

Revised : October 25, 2022

Accepted : June 11, 2023

บทคัดย่อ

ภาควิชาคอมพิวเตอร์ กองวิชาคณิตศาสตร์และคอมพิวเตอร์ กองการศึกษา โรงเรียนนายเรืออากาศนวมินทกษัตริยาธิราช เป็นสาขาที่จัดการเรียนการสอนโดยมุ่งเน้นความรู้ทั้งภาคทฤษฎีและภาคปฏิบัติทางคอมพิวเตอร์เป็นหลัก โดยเน้นการนำความรู้ไปประยุกต์เพื่อใช้งานจริง มีการสนับสนุนการเรียนการสอนโดยจัดห้องปฏิบัติการสำหรับการเรียนรู้เพิ่มเติม เพื่อให้นักศึกษาได้พัฒนาความรู้ ทักษะ นอกเหนือจากการเรียนในห้องเรียน โดยการจัดการเรียนการสอนในคาบปฏิบัติการนั้น มีความจำเป็นต้องมีอาจารย์พิเศษเพื่อทำการช่วยเหลือนักศึกษา และแบ่งเบาภาระของอาจารย์ผู้สอน การจัดการอาจารย์พิเศษโดยพิจารณาข้อจำกัดด้านทรัพยากร อันได้แก่ จำนวนผู้สมัคร เวลาในการเรียนการสอน รวมถึงคุณสมบัติในการปฏิบัติงานของอาจารย์พิเศษ ถือเป็นปัญหาที่มีความยุ่งยากซับซ้อน โดยเฉพาะเมื่อเป็นการจัดการโดยมนุษย์ มีขั้นตอนวิธีมากมายที่ถูกนำเสนอขึ้นเพื่อใช้ในการแก้ปัญหาการจัดการตารางเวลา เช่น ขั้นตอนวิธีทางพันธุกรรม (Genetic Algorithm) การค้นหาแบบทาบู (Tabu Search) ในงานวิจัยชิ้นนี้ นำเสนอการแก้ปัญหาการจัดการตาราง โดยพิจารณาเงื่อนไขและหาเซตคำตอบที่สอดคล้องกับเงื่อนไข (Constraint Based) ร่วมกับเทคนิคการจัดกลุ่ม (Cluster Method) และหลักการกำหนดเส้นตาย (Earliest Deadline First) งานวิจัยชิ้นนี้ได้พัฒนาระบบจัดการตารางสำหรับอาจารย์พิเศษผ่านเว็บ โดยมีกรณีศึกษา คือ ภาควิชาคอมพิวเตอร์ กองวิชาคณิตศาสตร์และคอมพิวเตอร์ กองการศึกษา โรงเรียนนายเรืออากาศนวมินทกษัตริยาธิราช

คำสำคัญ: การจัดการตารางเวลา, ปัญหาความพึงพอใจภายใต้ข้อจำกัด, การแบ่งกลุ่ม

Abstract

The department of Computer Science belong to Science Faculty of Navaminda Kasatriyadhiraj Royal Air Force Academy (NKRAFA) has a main mission to improve students in both theoretical and practical skills. For each enrolled subject, the computer science students have laboratory for practicing. Therefore, special instructors are needed for supporting the students in each subject. The problem of selecting a set of appropriate special instructor and scheduling a right person in a right timeslot are challenging. Since the problem is large, complex and consumed time when solved by manual. Even there are many techniques have been proposed such as Tabu search, Genetic Algorithm, Constraint Based Approach. This research proposes a combined technique between Constraint Based Approach, Cluster method and Earliest Deadline First for timetabling special instructor problem. The implemented system works like a web-based application. Case study of the department of Computer Science, Navaminda Kasatriyadhiraj Royal Air Force Academy (NKRAFA)

Keywords: Scheduling, Constraint Satisfaction Problem, Clustering

1. บทนำ

ภาควิชาคอมพิวเตอร์ กองวิชาคณิตศาสตร์และคอมพิวเตอร์ กองการศึกษา โรงเรียนนายเรืออากาศนวมินทกษัตริยาธิราช เป็นภาควิชาที่จัดการเรียนการสอนโดยมุ่งเน้นความรู้ทั้งภาคทฤษฎีและภาคปฏิบัติทางคอมพิวเตอร์เป็นหลัก โดยเน้นการนำความรู้ไปประยุกต์เพื่อใช้งานจริง มีการสนับสนุนการเรียนการสอนโดยจัดห้องปฏิบัติการสำหรับการเรียนรู้เพิ่มเติม เพื่อให้ นักศึกษาได้พัฒนาความรู้ ทักษะ นอกเหนือจากการเรียนในห้องเรียน โดยการจัดการเรียนการสอนในคาบปฏิบัติการณ์นั้น มีความ จำเป็นต้องมีอาจารย์พิเศษเพื่อทำการช่วยเหลือนักศึกษา และแบ่งเบาภาระของอาจารย์ผู้สอน

การจัดตารางเวลา คือ การกำหนดวัน เวลา โดยทำการจัดสรรทรัพยากรภายใต้เงื่อนไขต่างๆ ลงในช่วงเวลาที่มีเพื่อให้ได้ ตารางเวลาที่ตรงตามวัตถุประสงค์ ภายใต้ทรัพยากรที่มีอยู่อย่างจำกัด โดยจะต้องไม่เกิดความขัดแย้งกันในเวลาดังกล่าว [1] การ จัดตารางอาจารย์พิเศษโดยพิจารณาข้อจำกัดด้านทรัพยากรอื่น ได้แก่ จำนวนผู้สมัคร เวลาในการเรียนการสอน รวมถึงคุณสมบัติ ในการปฏิบัติงานของอาจารย์พิเศษ ถือเป็นปัญหาที่มีความซับซ้อน โดยเฉพาะการจัดการโดยมนุษย์ ปัญหาที่อาจเกิดขึ้นระหว่าง การดำเนินการจัดตารางเวลา เช่น มีการเปลี่ยนแปลงตารางเวลาของอาจารย์ผู้สอนหรือนักศึกษา การชนกันของเวลาเมื่อมีการจัด ตารางให้อาจารย์ผู้สอนมากกว่า 1 วิชา หรือมากกว่า 1 ห้อง ณ ช่วงเวลาใดเวลาหนึ่ง และอีกปัญหาหนึ่งซึ่งเกิดจากการเกิดความ ต้องการอาจารย์ผู้สอน ณ ช่วงเวลาใดเวลาหนึ่งพร้อมกัน ก็ทำให้การจัดตารางอาจารย์พิเศษมีความยุ่งยากมากขึ้น

เนื่องจากความซับซ้อนของปัญหาการจัดตาราง เป็นความท้าทายของนักวิจัยมายาวนาน มีขั้นตอนวิธีมากมายที่ถูกเสนอขึ้น เพื่อใช้ในการแก้ปัญหาการจัดตาราง เช่น ขั้นตอนวิธีทางพันธุกรรม (Genetic Algorithm) -การค้นหาลำดับ (Tabu Search)

ส่วนในงานวิจัยนี้ นำเสนอการแก้ปัญหาการจัดตารางโดยพิจารณาเงื่อนไขและหาเซตคำตอบที่สอดคล้องกับเงื่อนไข (Constraint Based Approach) ร่วมกับเทคนิคการจัดกลุ่ม (Cluster Method) การค้นคืนฐานกรณี (Retriever) [2] และ หลักการกำหนดเส้นตาย (Earliest Deadline First)

งานวิจัยนี้ได้นำเสนอขั้นตอนวิธีในการจัดตารางสำหรับอาจารย์พิเศษ และพัฒนาระบบจัดตารางอาจารย์พิเศษผ่านเว็บแอปพลิเคชัน โดยอาจารย์ผู้สอนสามารถลงทะเบียนเพื่อสมัครเป็นอาจารย์พิเศษ อาจารย์ผู้รับผิดชอบในแต่ละรายวิชาสามารถลงทะเบียนความต้องการอาจารย์พิเศษ และระบบจะทำการจัดตารางเวลาสำหรับอาจารย์พิเศษแต่ละคนโดยอัตโนมัติ ซึ่งสามารถแก้ปัญหาค่าความยุ่งยากในระหว่างการจัดตาราง ทำให้ประหยัดทรัพยากรด้านเวลา ค่าใช้จ่ายในการดำเนินงาน และก่อให้เกิดความสะดวกเมื่อเทียบกับการจัดตารางด้วยแรงงานมนุษย์

2. ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

กว่าครึ่งศตวรรษของงานวิจัยที่เกี่ยวกับปัญหาการจัดตารางเวลา มีงานวิจัยจำนวนหนึ่งนำเสนอวิธีการแก้ปัญหา โดยปัญหาการจัดตารางการเรียนการสอนในระดับมหาวิทยาลัยเป็นกรณีศึกษาที่ถูกยกมาศึกษามากที่สุดเพราะเป็นปัญหาที่มีความยุ่งยากซับซ้อน [3] ส่วนปัญหาการจัดตารางให้ตรงตามความเหมาะสมของรายวิชานั้นยังไม่ปรากฏในงานวิจัยมากนัก โดยการพัฒนาวิธีการเพื่อนำมาใช้แก้ปัญหาในการจัดตาราง สามารถแบ่งกลุ่มวิธีการในการจัดตาราง ออกเป็น 5 ประเภท ดังนี้

2.1 Sequential Method

วิธีการนี้มีหลักในการจัดตารางตามลำดับเหตุการณ์ และทำการกำหนดเหตุการณ์ลงในช่วงเวลาตามลำดับที่ได้จัดเอาไว้ ซึ่งจะต้องไม่มีเหตุการณ์ใดที่เกิดความขัดแย้งกับเหตุการณ์อื่น ๆ โดยเมื่อนำวิธีการแบบ Sequential มาใช้กับการจัดตารางส่วนใหญ่จะนำเสนอโดยการใช้โหนดเป็นตัวแทนของแต่ละเหตุการณ์ และเส้นที่เชื่อมระหว่างโหนดจะแทนค่าของระยะทางหรือความขัดแย้งของแต่ละเหตุการณ์

2.2 Cluster Method

วิธีการในกลุ่มนี้จะใช้การแบ่งเหตุการณ์ออกเป็นกลุ่มๆ จากนั้นในขั้นตอนวิธีของการจัดตาราง จะทำการจัดแต่ละกลุ่มลงในช่วงเวลาที่ได้รับการจัดสรร โดยพิจารณาถึงความเหมาะสมและเงื่อนไขข้อบังคับต่างๆ

2.3 Meta-heuristic Method

วิธีการในกลุ่มนี้จะเริ่มต้นจากการสร้างตัวแทนของปัญหาขึ้นมา จากนั้นจะนำมาผ่านกระบวนการที่จะพัฒนาให้ตัวแทนปัญหามีคุณภาพที่ดีขึ้นในแต่ละรอบ โดยผลลัพธ์ที่ได้จากวิธีการนี้มักจะมีคุณภาพสูง แต่ในแต่ละรอบของการพัฒนาจะใช้ทรัพยากรในการประมวลผลสูงด้วย

2.4 Constraint Based Approach

วิธีการนี้จะเป็นการแก้ปัญหาโดยมีการกำหนดเงื่อนไขและพิจารณาหาคำตอบที่สอดคล้องกับเงื่อนไขนั้น ส่วนของเงื่อนไขต้องกำหนดตัวแปรขึ้นรองรับ เช่น ถ้ามีตัวแปรเพียงตัวเดียว (Unary) เงื่อนไขก็จะเกี่ยวข้องกับตัวแปรเพียงตัวเดียว ถ้ามี 2 ตัวแปร (Binary) เงื่อนไขก็จะเกี่ยวข้องกับ 2 ตัวแปร คือ จะต้องพิจารณาคู่ของตัวแปรนั้นๆ และในการแก้ปัญหาจะต้องทำตาม

ข้อบังคับต่าง ๆ ให้ได้ตามกฎเกณฑ์ของแต่ละตัวแปร ซึ่งความยุ่งยากในการจัดตารางจะแปรผันตรงกับเงื่อนไขที่ถูกกำหนดขึ้น โดยทั่วไปมีการแบ่งเงื่อนไขที่นำมาใช้เพื่อพิจารณาในการจัดตาราง โดยสามารถแบ่งออกเป็น 2 ประเภท ดังนี้

เงื่อนไขหลัก (Hard Constraint) เป็นข้อบังคับพื้นฐานในการจัดตารางโดยจะต้องไม่มีการละเมิดเงื่อนไขนี้ หากมีการละเมิดเงื่อนไข จะถือว่าตารางเวลาที่ได้นั้นไม่สามารถนำไปใช้ได้

เงื่อนไขรอง (Soft Constraint) เป็นเงื่อนไขที่ไม่ส่งผลกระทบต่อการจัดตารางเวลา แต่จะช่วยให้ตารางเวลามีความเหมาะสมหรือเป็นที่น่าพอใจสำหรับผู้ที่น่าตารางเวลาไปใช้งานการจัดตารางเวลาให้ออกมาตรงตามเงื่อนไขรองในทุกกรณีนั้น อาจเป็นไปได้ยากเนื่องจากทรัพยากรที่มีอยู่อย่างจำกัด ซึ่งถึงแม้ว่าตารางเวลาที่ออกมาจะไม่ผ่านเงื่อนไขรองทุกข้อ แต่ตารางเวลาที่ได้นั้นสามารถรองรับการใช้งานได้

2.5 Earliest Deadline First

เป็นเทคนิคการจัดตารางเวลาแบบไดนามิกที่ใช้ในระบบปฏิบัติการ (Operating Systems) ซึ่งใช้ในการจัดคิวลำดับความสำคัญของโพรเซส โดยเมื่อมีการจัดตารางจะมีการค้นหาโพรเซสที่ใกล้เคียงกับเส้นตายหรือเร่งด่วนที่สุด และจะกำหนดให้โพรเซสดังกล่าวถูกดำเนินการในคิวถัดไป [4]

EDF เป็นวิธีการจัดการตารางงานใน Preemptive Uniprocessor โดยในการดำเนินการสำหรับงานใดๆ ในแต่ละโพรเซสย่อยที่เป็นอิสระต่อกัน มีคุณสมบัติที่แตกต่างกันทั้งเวลาในการดำเนินการ (Execution Time) และกำหนดเส้นตาย (Deadline) ซึ่งระบบปฏิบัติการจะทำการจัดลำดับความสำคัญเพื่อช่วยให้งานทั้งหมดเสร็จสมบูรณ์ตามกำหนดเวลาของแต่ละโพรเซส

2.6 Constraint Satisfaction Problem

ปัญหาที่ถูกกำหนดโดยเงื่อนไขความพึงพอใจสามารถจำลองปัญหาผ่านโมเดลทางคณิตศาสตร์ เพื่อใช้แก้ปัญหาที่ถูกกำหนดด้วยเงื่อนไข โดยมีสามสิ่ง ที่ต้องกำหนด คือ ตัวแปร โดเมนของตัวแปร และข้อจำกัดหรือเงื่อนไข การแก้ปัญหาแบบ CSP นั้นเป็นการค้นหาคำตอบในลักษณะที่ต้องค้นหาคำตอบไปเรื่อย ๆ โดยรูปแบบของเซตคำตอบที่ถูกต้องจะถูกจำกัดด้วยข้อกำหนดหรือเงื่อนไขที่กำหนดขึ้น และการค้นหาคำตอบใน CSP นั้นสามารถใช้ขั้นตอนวิธีในการค้นหาทั่วไป (General - Purpose Search Algorithms) หากแต่จะใช้เวลานานในการแก้ปัญหา การใช้ขั้นตอนวิธี Backtracking หรือ Forward Checking จะทำให้ค้นหาคำตอบได้เร็วขึ้น [5]

Backtracking เป็นวิธีการหาคำตอบของปัญหา โดยการค้นหาคำตอบไปที่ละส่วน (Partial Solution) โดยเมื่อได้คำตอบในส่วนหนึ่งแล้ว ก็ทำการค้นหาคำตอบในส่วนอื่นๆ ต่อไป เพื่อให้ได้เซตของคำตอบที่สมบูรณ์ โดยในระหว่างการค้นหาคำตอบนั้นเมื่อพบว่าคำตอบที่ได้นั้นขัดแย้งกับเงื่อนไข หรือไม่สามารถนำไปใช้ได้จะทำการย้อนกลับไป (Backtrack) เพื่อดำเนินการค้นหาคำตอบใหม่ในรูปแบบอื่นๆ ที่เป็นไปได้ต่อไป

Forward Checking เป็นวิธีการหาคำตอบของปัญหา ในลักษณะการตรวจสอบไปข้างหน้า คือ ในระหว่างการดำเนินการเพื่อหาคำตอบ จะมีการตัดตัวเลือกที่ไม่ใช่คำตอบออกทันที โดยอาจจะมีการตัดตัวเลือกดังกล่าวตั้งแต่แรก หรือตัดทิ้งทีละส่วนในแต่ละขอบเขตที่พิจารณา โดยในงานวิจัยชิ้นนี้จะใช้วิธีการหาคำตอบแบบ Forward Checking

2.6 งานวิจัยที่เกี่ยวข้อง

จากการศึกษางานวิจัยที่เกี่ยวข้องกับการจัดตาราง พบว่ามีการศึกษา และนำวิธีการต่างๆ มาใช้สำหรับการจัดตาราง ซึ่งในแต่ละงานวิจัยมีวิธีการและเทคนิคที่แตกต่างกันออกไป โดยผู้วิจัยแต่ละคนจะพิจารณาขั้นตอนวิธีที่เหมาะสมกับปัญหา และนำเทคนิคต่างๆ มาประยุกต์ใช้เพื่อให้ได้คำตอบที่ตรงตามวัตถุประสงค์

ปี ค.ศ. 2003 Valouxis และ Housos [6] ได้นำเสนอการแก้ปัญหาการจัดตารางเรียนในโรงเรียนมัธยม โดยใช้ Constraint Based Programming และใช้เทคนิค Local Search ร่วมด้วย เพื่อลดพื้นที่และเพิ่มประสิทธิภาพในการค้นหาเซตคำตอบ

ปี ค.ศ. 2011 Hussin และคณะ [7] ได้นำเสนอการแก้ปัญหาการจัดตารางสอบของมหาวิทยาลัยในมาเลเซียโดยใช้วิธีการแบบ Heuristic และประยุกต์ใช้เทคนิค Cluster Heuristic กับ Sequential Heuristic ร่วมกัน โดยการจัดกลุ่มรายวิชาทั้งหมด 39 วิชานั้น มีเพียง 30 วิชาที่ได้รับการจัดกลุ่ม และ 9 วิชาซึ่งไม่สามารถจัดกลุ่ม โดยผู้วิจัยได้แยกทั้ง 9 วิชานี้ออกมาเป็นกลุ่มพิเศษ แล้วใช้วิธีการ Graph Coloring เข้ามาจัดการปัญหานี้

ปี ค.ศ. 2013 Mousa และ El-Sisi [8] ได้นำเสนอการแก้ปัญหาการจัดตารางเรียนในมหาวิทยาลัย โดยใช้ Genetic Algorithm ซึ่งผู้วิจัยได้ทำการทดสอบ โดยวิธีการคัดเลือกกลุ่มประชากร 2 วิธี คือ Tournament Selection และ Roulette Wheel Selection ในการทดสอบกับปัญหขนาดเล็กลง และขนาดใหญ่นั้น พบว่าการเลือกกลุ่มประชากรโดย Tournament Selection ให้ผลลัพธ์ที่ดีกว่าทั้งสองการทดสอบ และมีการใช้ Mutation Error Method เพื่อเพิ่มประสิทธิภาพในการหาคำตอบในกรณีที่เกิดความขัดแย้งหรือไม่สามารถหาคำตอบได้

ปี ค.ศ. 2015 Nagareda และ Camacho [9] ได้นำเสนอ Constraint Based Model สำหรับการแก้ปัญหาการจัดตารางเรียนในโรงเรียนมัธยม ในประเทศสวิตเซอร์แลนด์ โดยทำการวิเคราะห์ เงื่อนไขข้อบังคับต่างๆ และกำหนดตัวแปรมารองรับ กฎเกณฑ์ เงื่อนไข ข้อบังคับ ที่กำหนดขึ้น ซึ่งทำให้สะดวกต่อการนำไปใช้งาน และง่ายต่อการยุบรวมกฎหรือเงื่อนไขที่มีความคล้ายคลึงกันเพื่อแก้ปัญหาไปพร้อมกัน

3. การดำเนินการวิจัย

3.1 วิเคราะห์ปัญหาการจัดตารางเวลา

ปัญหาการจัดตารางเวลาอาจารย์พิเศษนั้น เป็นปัญหาที่เกี่ยวข้องกับข้อมูลผู้สมัครเป็นอาจารย์ผู้สอน และข้อมูลรายวิชา โดยเป็นการค้นหาคำตอบที่สอดคล้องกับเงื่อนไข และไม่มี ความขัดแย้งกันในด้านตารางเวลา เมื่อพิจารณาปัญหาการจัดตารางเวลาอาจารย์พิเศษ สามารถวิเคราะห์เป็นปัญหาในเชิงคณิตศาสตร์ และเขียนให้อยู่ในรูปแบบของแบบจำลองทางคณิตศาสตร์ตามหลักการ Constraint Satisfaction Problem ได้ดังนี้

$T = \{T_1, ..., T_i\}$	แทนเซตของผู้สมัครเป็นอาจารย์ผู้สอน (Teacher)
$S = \{S_1, ..., S_j\}$	แทนเซตของรายวิชาที่ต้องการอาจารย์พิเศษ (Subject)
$F = \{F_1, ..., F_m\}$	แทนเซตของชั่วโมงแรกใน (First Hour) ในคาบปฏิบัติการ

$$L = \{L_1, \dots, L_m\}$$

แทนเซตชั่วโมงสุดท้าย (Last Hour) ในคาบปฏิบัติการ

$$C = \{C_1, \dots, C_q\}$$

แทนเซตของเงื่อนไข (Condition) ที่กำหนดขึ้นเพื่อใช้ประกอบการพิจารณา

โดยคำตอบที่เป็นไปได้ทั้งหมด คือ

$$W = T \times S \times F \times L$$

$$W = \{(T_1, S_1, F_1, L_1), \dots, (T_i, S_j, F_m, L_m)\} \text{ โดยที่ } |W| = i \times j \times m \times m$$

เมื่อทำการคัดกรองตามเงื่อนไขที่กำหนดขึ้น จะได้เซตคำตอบที่ตรงตามวัตถุประสงค์ และสามารถนำไปใช้งานได้จริง โดยในกระบวนการจัดตารางเวลานั้น ผู้วิจัยได้พัฒนาระบบจัดตาราง ด้วยวิธี Constraint Based Approach ร่วมกับเทคนิค Cluster Method และ Earliest Deadline First โดยนำเงื่อนไขข้อบังคับที่กำหนดขึ้น มาพิจารณาเพื่อคัดกรองข้อมูลตั้งแต่กระบวนการรับข้อมูลเข้า ไปจนถึงกระบวนการในการจัดตารางเวลา เพื่อลดปัญหาที่เกิดจากการได้เซตคำตอบที่ไม่สามารถนำไปใช้ได้ ซึ่งส่งผลกระทบโดยตรงต่อระยะเวลาในการประมวลผล พร้อมทั้งจัดกลุ่มปัญหาเพื่อให้ง่ายต่อการหาคำตอบ และเทคนิค Earliest Deadline First ช่วยในการจัดตารางเวลาในกรณีที่มีทรัพยากรจำกัด

ในกระบวนการจัดตารางเวลาสำหรับอาจารย์พิเศษนั้น จำเป็นต้องพิจารณาเงื่อนไขในการจัดตาราง และตารางเวลาที่ได้จะต้องสอดคล้องกับเงื่อนไขที่กำหนดไว้ โดยแบ่งเงื่อนไข ออกเป็น 2 ประเภท ประกอบด้วย เงื่อนไขหลัก และเงื่อนไขรอง ดังนี้

เงื่อนไขหลัก (Hard Constraint) ได้แก่

1. เวลาทำงานของอาจารย์พิเศษจะต้องมีจำนวนชั่วโมงครบถ้วน
2. จำนวนอาจารย์พิเศษจะต้องครบถ้วนตามความต้องการในแต่ละวิชา
3. ช่วงเวลาในการปฏิบัติงานของอาจารย์ผู้สอนจะต้องไม่ขัดแย้งกัน โดย ณ ช่วงเวลาใดเวลาหนึ่งอาจารย์ผู้สอนจะมีคาบปฏิบัติงาน เพียงรายวิชาเดียว ห้องเดียว
4. ตารางเวลาในการปฏิบัติงานของอาจารย์ผู้สอน จะต้องสอดคล้องกับตารางเวลาของอาจารย์ผู้สอน โดยจะต้องจัดตารางปฏิบัติงาน ในช่วงเวลาที่อาจารย์ผู้สอนมีเวลาว่างสามารถปฏิบัติงานได้จริง

เงื่อนไขรอง (Soft Constraint) คือ

ในรายวิชาที่มีความสำคัญ (พิจารณาจาก priority ของแต่ละรายวิชา) กำหนดให้มีการคัดเลือกคุณสมบัติของอาจารย์ผู้สอน โดยพิจารณาจากวุฒิการศึกษา ผลการเรียนโดยเฉลี่ย หรือระดับผลการเรียนในรายวิชานั้น ๆ

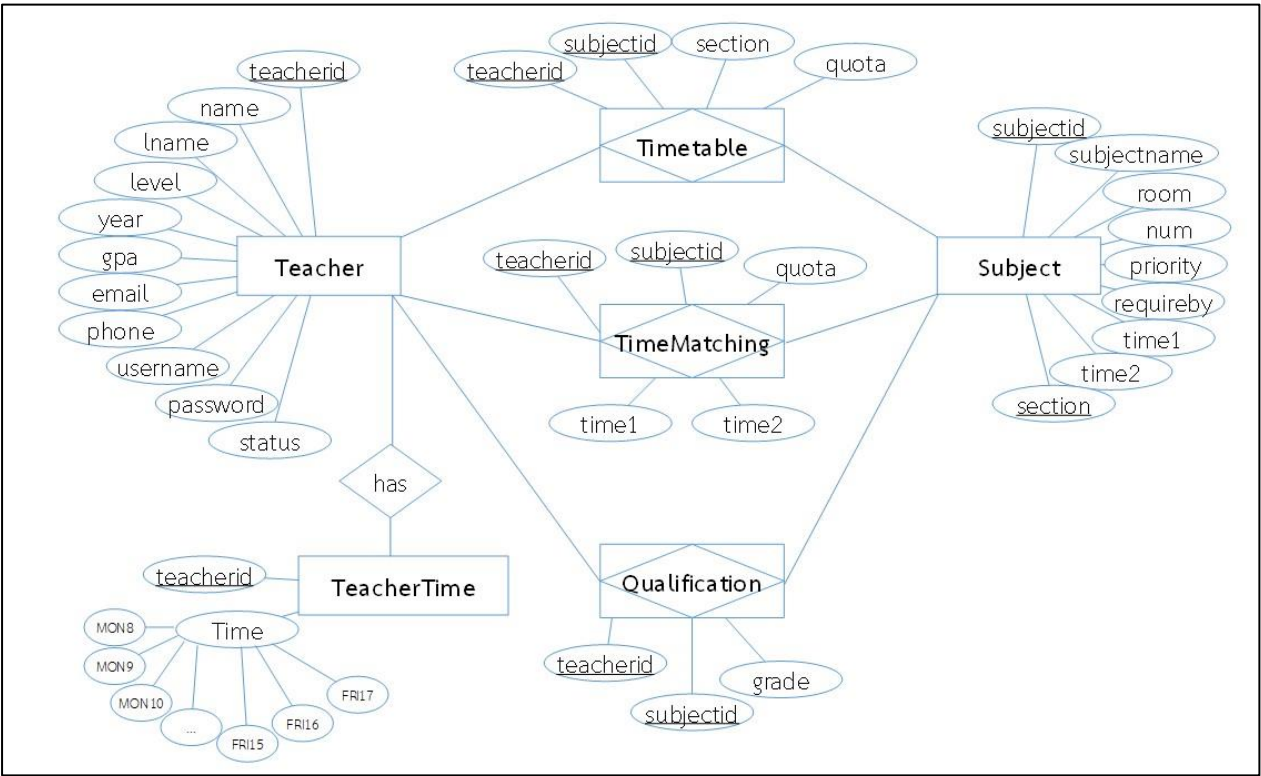
3.2 ฐานข้อมูลสำหรับระบบจัดตารางเวลา

โครงสร้างของฐานข้อมูลภายในระบบจัดตารางเวลา ใช้สำหรับจัดเก็บข้อมูลที่เกี่ยวข้องกับการจัดตารางทั้งหมด โดยจะเก็บข้อมูลเกี่ยวกับ รายวิชาที่มีการร้องขออาจารย์พิเศษ และข้อมูลเกี่ยวกับผู้สมัครอาจารย์พิเศษทั้งหมด โดยออกแบบให้

อยู่ในรูปของโครงสร้างฐานข้อมูลเชิงสัมพันธ์ (Relational Database Structure) ซึ่งอธิบายตาม Entity Relationship Diagram หรือ ER Diagram ใน รูปที่ 1 (โดยมีคำอธิบายตามตารางที่ 1)

ตารางที่ 1 คำอธิบายตารางในฐานข้อมูลสำหรับระบบจัดตารางเวลา

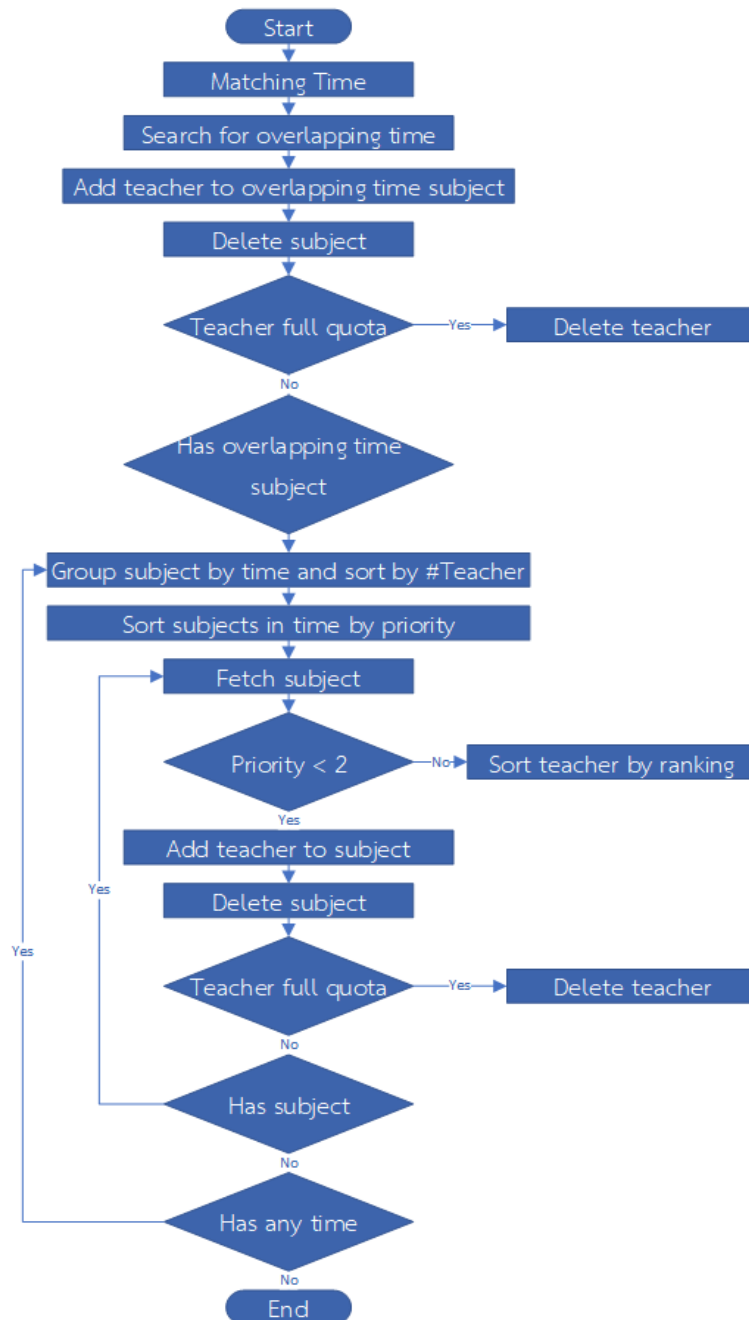
ชื่อตาราง	คำอธิบาย
Teacher	สำหรับเก็บข้อมูลทั่วไปและข้อมูลการเข้าสู่ระบบของผู้สมัครอาจารย์พิเศษ
TeacherTime	สำหรับเก็บข้อมูลตารางเวลาของผู้สมัครอาจารย์พิเศษ
Qualification	สำหรับเก็บข้อมูลผลการเรียนของผู้สมัครอาจารย์พิเศษ
Subject	สำหรับเก็บข้อมูลทั่วไปและข้อมูลความต้องการอาจารย์พิเศษในแต่ละรายวิชา
TimeMatching	สำหรับเก็บข้อมูลเวลาที่อาจารย์พิเศษสามารถทำงานได้ในรายวิชาต่างๆ
Timetable	สำหรับเก็บผลลัพธ์หรือเซตคำตอบสำหรับระบบจัดตารางเวลา



รูปที่ 1 แผนภาพความสัมพันธ์ของตารางในฐานข้อมูลสำหรับระบบจัดตาราง

3.3 วิเคราะห์ขั้นตอนวิธีในการจัดตารางเวลา

เมื่อทำการวิเคราะห์และออกแบบระบบจัดตารางเวลาสำหรับอาจารย์พิเศษเสร็จครบถ้วนแล้ว ขั้นตอนวิธีในกระบวนการจัดตารางเวลาแสดงยังรูปที่ 2



รูปที่ 2 ขั้นตอนวิธีในการการจัดตารางเวลา

- Matching Time: เป็นการจับคู่กันระหว่างเวลาว่างของผู้สมัครอาจารย์พิเศษและเวลาเรียนในแต่ละรายวิชา โดยผลลัพธ์ที่ได้จะบอกว่าผู้สมัครแต่ละคนสามารถทำงานในช่วงเวลาใด
- Search for overlapping time: เป็นขั้นตอนในการค้นหาวิชาที่มีเวลาในการเรียนการสอนคาบเกี่ยวกับรายวิชาอื่น ๆ หรือกลุ่มผู้เรียนอื่น ๆ โดยจะเป็นกรณีพิเศษที่จะได้รับการจัดตารางให้เป็นอันดับแรก
- Add teacher to overlapping time subject: ในขั้นตอนนี้จะทำการพิจารณาจัดอาจารย์พิเศษให้แต่ละรายวิชาตามเงื่อนไขลำดับความสำคัญของแต่ละรายวิชา
- Delete Subject: เป็นขั้นตอนการลบรายวิชาที่ได้รับการจัดอาจารย์พิเศษเรียบร้อยแล้วออก
- Condition (Teacher full quota?): เป็นการตรวจสอบชั่วโมงการทำงานของอาจารย์พิเศษแต่ละคน หากคนใดมีชั่วโมงการทำงานครบตามกำหนดแล้ว จะทำการลบข้อมูลออกจาก Working Space เนื่องจากอาจารย์คนนั้น ๆ ไม่จำเป็นต้องได้รับการพิจารณาเพื่อเพิ่มภาระงานแล้ว
- Condition (Has overlapping time subject?): เป็นการตรวจสอบเพื่อทำการจัดอาจารย์พิเศษในรายวิชาที่มีเวลาคาบเกี่ยวกับ โดยจะทำการจัดอาจารย์พิเศษจนครบทุกรายวิชาที่อยู่ในช่วงเวลาที่ถูกลัดกรอง
- Group subject by time and sort by #teacher: ในขั้นตอนนี้จะทำการจัดกลุ่มรายวิชา โดยรวมรายวิชาที่สอน ณ เวลาเดียวกันมาจัดเป็นกลุ่มเดียวกัน และจัดเรียงตามจำนวนอาจารย์พิเศษ ที่มีเวลาว่างในแต่ละช่วงเวลา
- Sort subject in time by priority: ในแต่ละช่วงเวลาที่ถูกแบ่งเป็นกลุ่ม ๆ นั้น อาจมีหนึ่งหรือหลายวิชา ณ ช่วงเวลาเดียวกัน โดยในขั้นตอนนี้จะทำการจัดเรียงลำดับความสำคัญของแต่ละวิชา เพื่อทำการพิจารณาจัดอาจารย์พิเศษให้ในขั้นตอนถัดไป
- Fetch subject: เป็นการหยิบแต่ละรายวิชาในแต่ละช่วงเวลามาทำการจัดการเรียนการสอน
- Condition (Priority < 2): ในขั้นตอนนี้จะเป็นการตรวจสอบเงื่อนไขความสำคัญในแต่ละรายวิชา โดยหากรายวิชานั้น ๆ มี Priority มากกว่าหรือเท่ากับ 2 จะทำการพิจารณาคุณสมบัติของอาจารย์พิเศษเพื่อจัดให้รายวิชาดังกล่าวต่อไป
- Sort teacher by ranking: เป็นการตรวจสอบคุณสมบัติของอาจารย์พิเศษแต่ละคน เพื่อทำการพิจารณาจัดตารางงานในแต่ละรายวิชา
- Add TA to subject: ในขั้นตอนนี้จะทำการจัดอาจารย์พิเศษให้รายวิชาที่ถูกเลือก และวนซ้ำที่ละวิชา จนกระทั่งครบทุกรายวิชา

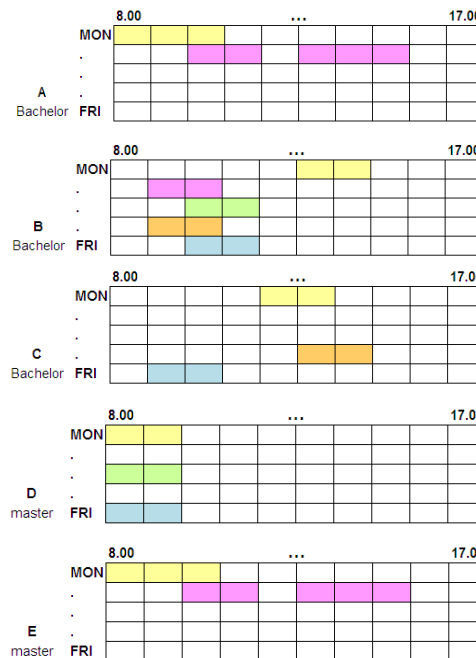
4. ผลการวิจัย

จากการวิเคราะห์ระบบจัดตารางเวลาสำหรับอาจารย์พิเศษ ผู้วิจัยได้ทำการพัฒนาโปรแกรมประยุกต์ สำหรับระบบจัดตาราง ซึ่งประกอบด้วยส่วนการทำงานหลัก ได้แก่ 1) การลงทะเบียนเพื่อเข้าใช้งานระบบ 2) การเพิ่ม ลบ แก้ไข รายละเอียดของผู้สมัคร และอาจารย์พิเศษ 3) การจัดการผู้สมัครอาจารย์พิเศษ 4) การลงทะเบียนความต้องการในแต่ละรายวิชา และ 5) การแสดงผลข้อมูลตารางอาจารย์พิเศษ ซึ่งผ่านกระบวนการจัดตารางเรียบร้อยแล้ว ซึ่งในงานวิจัยนี้ได้ดำเนินการจัดตารางสำหรับอาจารย์พิเศษ โดยการจัดสรรทรัพยากรในระบบให้สอดคล้องกับเงื่อนไขที่ถูกกำหนดขึ้น

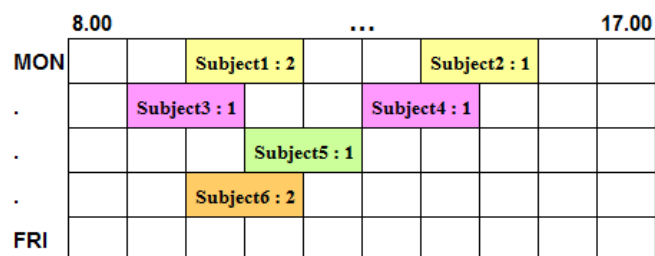
4.1 การทดสอบระบบ

เพื่อทำการทดสอบระบบจัดตาราง ผู้วิจัยได้ทำการสร้างกรณีทดสอบในรูปแบบต่าง ๆ ขึ้น โดยมีการออกแบบสภาพแวดล้อมของระบบ ซึ่งประกอบด้วย ผู้ช่วยสอน ตารางเวลาของผู้สมัคร ตารางเวลาชั่วโมงเรียน และความต้องการอาจารย์พิเศษในแต่ละรายวิชา

โดยในการทดสอบระบบนั้นจะทำการเพิ่มปริมาณความหนาแน่นในตัวแปรต่าง ๆ คือ ความหนาแน่นของตารางเวลา ผู้สมัครอาจารย์พิเศษ ความหนาแน่นของความต้องการอาจารย์พิเศษ เพื่อทำการทดสอบระบบการจัดตาราง และพิจารณาคุณภาพของตารางเวลาที่ได้



รูปที่ 3 ตารางเวลาผู้สมัครอาจารย์พิเศษ



รูปที่ 4 ตารางความต้องการอาจารย์พิเศษ

จากรูปประกอบที่ 3 แสดงตารางเวลาผู้สมัครอาจารย์พิเศษ โดยความหนาแน่นของตารางเวลาผู้สมัครอาจารย์พิเศษ จะพิจารณาจากจำนวนชั่วโมงที่ผู้สมัครจะต้องปฏิบัติงานตามเงื่อนไขที่กำหนด ต่อ จำนวนชั่วโมงที่ผู้สมัครมีเวลาว่างและสามารถปฏิบัติงานได้ตามตารางปฏิบัติการ เขียนเป็นสมการ ดังนี้

$$D_{ta} = \frac{TA_{work}}{TA_{free}}$$

โดยที่ D_{ta} แทนความหนาแน่นของตารางเวลาผู้สมัครอาจารย์พิเศษ

TA_{work} แทนจำนวนชั่วโมงที่ผู้สมัครอาจารย์พิเศษจะต้องปฏิบัติงานตามเงื่อนไขที่กำหนด

TA_{free} แทนจำนวนชั่วโมงที่ผู้สมัครอาจารย์พิเศษมีเวลาว่างและสามารถปฏิบัติงานได้ตามตาราง

จากรูปประกอบที่ 4 แสดงตารางเวลาคาบปฏิบัติการและจำนวนความต้องการอาจารย์พิเศษในแต่ละวิชาโดยความหนาแน่นของความต้องการอาจารย์พิเศษในแต่ละวิชา จะพิจารณาจาก จำนวนความต้องการอาจารย์พิเศษ ต่อ จำนวนอาจารย์พิเศษที่สามารถปฏิบัติงานได้ ณ ช่วงเวลาใด ๆ โดยช่วงเวลาที่มีความหนาแน่นสูงที่สุดจะเป็นตัวกำหนดความหนาแน่นของทั้งตาราง เขียนเป็นสมการได้ดังสมการที่ (1)

$$D_{timetable} = \max \left(\frac{TA_{requirement}}{TA_{amount}} \right) \quad (1)$$

โดยที่ $D_{timetable}$ แทนความหนาแน่นของตารางเวลาคาบปฏิบัติการ

$TA_{requirement}$ แทนจำนวนความต้องการผู้ช่วยสอน ณ ช่วงเวลาใด ๆ

TA_{amount} แทนจำนวนผู้ช่วยสอนที่สามารถปฏิบัติงานได้ในแต่ละช่วงเวลาใด ๆ

จากตัวอย่างกรณีทดสอบข้างต้น ประกอบด้วย ผู้สมัครอาจารย์พิเศษทั้งหมด 5 คน เป็นอาจารย์พิเศษในระดับปริญญาตรี 3 คน และระดับปริญญาโท 2 คน ในส่วนของความต้องการอาจารย์พิเศษในแต่ละรายวิชา จากตัวอย่างประกอบด้วยรายวิชาทั้งหมด 6 วิชา โดยแต่ละวิชาจะแสดงจำนวนความต้องการอาจารย์พิเศษหลังเครื่องหมาย “ : ” เช่น Subject1 : 2 หมายความว่า Subject1 ต้องการอาจารย์พิเศษ 2 คน

การใช้งานระบบจะมี 2 ส่วนหลัก ๆ คือ ส่วนของผู้ดูแลระบบและผู้ใช้ทั่วไปซึ่งก็คืออาจารย์พิเศษ ที่จะเข้ามาทำการลงทะเบียนเข้ารับการศึกษา โดยจะมีการเก็บข้อมูลประวัติ ผลการเรียนในรายวิชาที่สนใจ และตารางเวลาของผู้สมัคร เพื่อเป็นข้อมูลในการประกอบการพิจารณา ตัวอย่างหน้าจอของระบบ แสดงดัง รูปที่ 5, รูปที่ 6 และ รูปที่ 7

Profile	
ID	001
First name	Poramapom
Last name	Lporamapom
Level	Bachelor's degree ▾
Year	4 ▾
GPA	2.84
Email	P@e.c
Phone	0
Save Change	

รูปที่ 5 ประวัติส่วนตัวผู้สมัคร

Subject ID	Subject Name	Grade
คณ233	คณ233	A ▾
คณ331	คณ331	A ▾
คณ332	คณ332	A ▾
คณ432	คณ432	Non Regist ▾
คค231	คค231	Non Regist ▾
คค232	คค232	Non Regist ▾
คค233	คค233	Non Regist ▾
คค331	คค331	Non Regist ▾
คค333	คค333	Non Regist ▾
คค437	คค437	A ▾
Save Change		

รูปที่ 6 ระดับผลการเรียน

Day/Time	8.00-9.00	9.00-10.00	10.00-11.00	11.00-12.00	12.00-13.00	13.00-14.00	14.00-15.00	15.00-16.00	16.00-17.00	17.00-18.00
Mon	☐	☐	☐	☐	☐	☐	☒	☐	☐	☐
Tue	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
Wed	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
Thurs	☐	☐	☐	☐	☐	☒	☒	☐	☐	☐
Fri	☐	☐	☒	☒	☐	☐	☐	☒	☒	☐
Save Change										

รูปที่ 7 ตารางเวลาของผู้สมัคร

เมื่อทำการ Submit ข้อมูลเข้าสู่ระบบเรียบร้อยแล้ว ผู้ดูแลระบบจะสามารถเรียกดูข้อมูลทั้งหมดได้ผ่าน View ของ Administrator ตัวอย่างมุมมองการใช้งานของผู้ดูแลระบบแสดงดังรูปที่ 8 แสดงภาพรวมของการสมัครอาจารย์พิเศษทั้งหมด โดยผู้ดูแลระบบสามารถ Disable ผู้สมัครที่คุณสมบัติไม่ผ่านเกณฑ์ เพื่อไม่ให้ระบบนำข้อมูลดังกล่าวไปใช้ในการประมวลผล เช่น วุฒิการศึกษาไม่ตรง ผู้สมัคร ไม่แสดงหลักฐานประกอบการสมัคร และในรูปที่ 9 แสดงข้อมูลรายบุคคลของผู้สมัคร

ID	Name	Level	Detail	Enable
001	Poramaporn	Bachelor's degree		
002	Wachiraya	Bachelor's degree		
003	Nattharika	Bachelor's degree		
004	Naphat	Bachelor's degree		
005	Jirasak	Bachelor's degree		
007	Patcharee	Bachelor's degree		
008	Kanokwan	Bachelor's degree		
011	Tanaporn	Bachelor's degree		
015	Poranat	Master's degree		
016	Siripanga	Master's degree		
017	Pawin	Master's degree		
018	Jakkrit	Master's degree		
019	Panida	Master's degree		
020	Natsarin	Bachelor's degree		
021	Paweena	Bachelor's degree		
024	Nattapon	Bachelor's degree		
006	Jiraporn	Bachelor's degree		
009	Noppadon	Bachelor's degree		
010	Methavee	Bachelor's degree		
012	Somchai	Bachelor's degree		
013	Sofia	Bachelor's degree		
014	Narongrit	Bachelor's degree		
022	Suchanart	Bachelor's degree		
023	Pattaraporn	Bachelor's degree		

รูปที่ 8 ภาพรวมผลการสมัครอาจารย์พิเศษผ่านระบบ

Profile		
ID	013	
First name	Sofia	
Last name	Lsofia	
Level	Bachelor's degree	
Year	3	
GPA	3.23	
Email	S@e.c	
Phone	0	

Result		
คต331	คต331	Non Regist
คต232	คต232	Non Regist
คต231	คต231	Non Regist
คณ432	คณ432	Non Regist
คณ332	คณ332	A
คณ331	คณ331	A
คณ233	คณ233	A
คต437	คต437	A
คต233	คต233	Non Regist
คต333	คต333	Non Regist

013	8.00-9.00	9.00-10.00	10.00-11.00	11.00-12.00	12.00-13.00	13.00-14.00	14.00-15.00	15.00-16.00	16.00-17.00	17.00-18.00
Mon										
Tue										
Wed										
Thurs										
Fri										

รูปที่ 9 แสดงข้อมูลรายบุคคลของผู้สมัคร

สำหรับความต้องการอาจารย์พิเศษในแต่ละรายวิชา ผู้ดูแลระบบจะทำการนำข้อมูลความต้องการทั้งหมดเข้าสู่ระบบ โดยจะบันทึกข้อมูลรายละเอียดของแต่ละรายวิชา และช่วงเวลาที่ จะจัดให้มีการเรียนการสอนเพิ่มเติมแสดงดังรูปที่ 10

Subject Detail

Subject ID

344-141

Subject Name

344-141

Room

BSc606

Teacher

Supaporn

Section

01

Quantity

2

Priority

Very Important

Study Time

day/ time	8.00 9.00	9.00 10.00	10.00 11.00	11.00 12.00	12.00 13.00	13.00 14.00	14.00 15.00	15.00 16.00	16.00 17.00	17.00 18.00
Mon	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
Tue	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
Wed	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
Thurs	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
Fri	☐	☐	☒	☒	☐	☐	☐	☐	☐	☐

Submit

รูปที่ 10 แสดงความต้องการอาจารย์พิเศษ

เมื่อทำการบันทึกความต้องการทั้งหมดเข้าสู่ระบบเรียบร้อยแล้ว และการรับสมัครอาจารย์พิเศษมีจำนวนเพียงพอต่อการจัดการตาราง ระบบจะสามารถดำเนินการจัดการตารางเวลา เพื่อคัดเลือกอาจารย์พิเศษที่เหมาะสมในแต่ละรายวิชา ภายใต้ทรัพยากรที่มีทั้งหมด ผลลัพธ์ที่ได้จะแสดงผลในรูปของตารางเวลาที่เสร็จสมบูรณ์แล้ว แสดงดังรูปที่ 11

	8.00-9.00	9.00-10.00	10.00-11.00	11.00-12.00	12.00-13.00	13.00-14.00	14.00-15.00	15.00-16.00	16.00-17.00	17.00-18.00
Mon								คณ.432(01) 009 015	คณ.432(01) 009 015	
								คณ.231(04) 001 010 011	คณ.231(04) 001 010 011	
								คณ.437(02) 003 008	คณ.437(02) 003 008	
Tue						คณ.333(01) 017 019	คณ.333(01) 017 019	คณ.231(01) 015 019	คณ.231(01) 015 019	
								คณ.231(02) 016 018	คณ.231(02) 016 018	
								คณ.331(04) 003 017	คณ.331(04) 003 017	
Wed	คณ.331(01) 005	คณ.331(01) 005	คณ.231(05) 007 021	คณ.231(05) 007 021	คณ.232(01) 013 024	คณ.437(03) 007 024	คณ.437(03) 007 024	คณ.331(02) 011 015	คณ.231(03) 012 019	คณ.231(03) 012 019
Thurs	คณ.332(01) 016 017	คณ.332(01) 016 017		คณ.231(08) 005					คณ.331(02) 011 015	
Fri	คณ.231(07) 001 019 020	คณ.231(07) 001 019 020	คณ.231(06) 012 020	คณ.231(06) 012 020	คณ.233(01) 016	คณ.331(03) 006 018	คณ.331(03) 006 018	คณ.233(01) 006 018	คณ.233(01) 006 018	คณ.232(01) 013 024
			คณ.331(01) 008 019	คณ.331(01) 008 019						คณ.233(01) 016
			คณ.437(01) 008 020	คณ.437(01) 008 020						

รูปที่ 11 ตารางเวลาที่ ได้จากการประมวลผลของระบบ

4.2 ผลการประเมินประสิทธิภาพของระบบ

การประเมินประสิทธิภาพของระบบจัดการเวลาอาจารย์พิเศษ โดยการเพิ่มความหนาแน่นของตาราง เวลาของผู้สมัครอาจารย์พิเศษ ความหนาแน่นของความต้องการอาจารย์พิเศษ เพื่อทำการทดสอบระบบการจัดการ และพิจารณาคุณภาพของตารางเวลาที่ได้ พบว่าระบบทำงานได้อย่างถูกต้องและมีประสิทธิภาพภายใต้ทรัพยากรที่มีอยู่ โดยตารางที่ได้มีอาจารย์พิเศษประจำในแต่ละวิชาอย่างเพียงพอต่อความต้องการ

การประเมินโดยการสอบถามจากผู้ทดลองใช้ระบบ ซึ่งเป็นอาจารย์ในกองวิชาคณิตศาสตร์และคอมพิวเตอร์ โรงเรียนนายเรืออากาศนวมินทกษัตริยาธิราช จำนวน 10 คน มีรายการประเมินผลทั้งหมด 3 ด้าน ได้แก่ ด้านประสิทธิภาพ ด้านความถูกต้อง และด้านการใช้งานพบว่า อยู่ในเกณฑ์ดี ดังตารางที่ 2

ตารางที่ 2 ผลการประเมินความพึงพอใจของผู้ทดลองใช้ระบบ

รายการประเมินผล	$\bar{X}$	S.D.	แปลผล
1.ด้านประสิทธิภาพ	4.2	0.63	ดี
2.ด้านความถูกต้อง	4.8	0.42	ดี
3.ด้านการใช้งาน	4.3	0.48	ดี
รวมเฉลี่ย	4.43	0.51	ดี

5. สรุปผลการวิจัย

จากการพัฒนาระบบจัดการเวลาอาจารย์พิเศษ สามารถลดระยะเวลาในการจัดการตารางได้อย่างมีนัยสำคัญ เมื่อเปรียบเทียบกับการใช้วิธีการจัดการด้วยมนุษย์ ในกรณีที่มีการเปลี่ยนแปลงของตารางเวลา ระบบสามารถทำการจัดการใหม่ได้ทันที นอกจากจะลดระยะเวลาในการจัดการใหม่ ระบบยังสามารถลดค่าใช้จ่ายที่อาจเกิดขึ้นในระหว่างการจัดการใหม่ ตารางที่ได้จากการจัดด้วยระบบจัดการอัตโนมัติ เป็นตารางที่เสร็จสมบูรณ์ครบถ้วนตามเงื่อนไขหลัก และสามารถนำไปใช้งานได้จริง สอดคล้องกับวัตถุประสงค์ของงานวิจัย

6. ข้อเสนอแนะ

ในงานวิจัยนี้มีการพัฒนาระบบโดยแบ่งกลุ่มผู้ใช้ออกเป็น 2 ส่วน คือ ผู้ดูแลระบบและผู้ใช้งานทั่วไป ในส่วนของผู้ใช้งานทั่วไปจะเป็นการลงทะเบียนของผู้สมัครอาจารย์พิเศษ ผู้วิจัยพบว่า สามารถทำการพัฒนาต่อในส่วนนี้โดยการเพิ่มเติมการลงทะเบียน ความความต้องการผู้ช่วยสอนหรืออาจารย์พิเศษ ซึ่งระบบปัจจุบันจำกัดสิทธิ์ในการจัดการนี้โดยผู้ดูแลระบบเท่านั้น หากสามารถจัดการความต้องการจากผู้ใช้งานตารางโดยตรง จะช่วยให้การทำงานสะดวกและมีประสิทธิภาพมากขึ้น

7. เอกสารอ้างอิง

- [1] Yu, E. and Sung, K. (2002). A genetic algorithm for a university weekly courses timetabling problem. *International Federation of Operational Research Societies (IFORS)*. 703-717.
- [2] ภูมินันท์ บัวงาม. (2019). ระบบ แก้ไข ปัญหา ไอ พี ทีวี อัตโนมัติ โดยใช้ วิธี การ ให้ เหตุผล ด้วย ฐาน กรณี. *วารสารวิทยาศาสตร์และเทคโนโลยีนายเรืออากาศ*. 15(1): 99-111.
- [3] Burke, E. K., Newall, J. P. and Weare, R. F. (1998). A Simple Heuristically Guided Search for the Time-table Problem. *International ICSC Symposium on Engineering of Intelligent Systems*. 98(1): 574-579.
- [4] Singh, J., Patra, B. and Singh, S. (2011). An Algorithm to Reduce the Time Complexity of Earliest Deadline First Scheduling Algorithm in Real-Time System. *International Journal of Advanced Computer Science and Applications (IJACSA)*. 31-37.
- [5] Müller, T. (2005). Constraint-based Timetabling. Ph.D. thesis, Charles University.
- [6] Valouxis C. and Housos, E. (2003). *Computers & Operations Research*. 1555-1572.
- [7] Hussin, B., Basari, A., Shibghatullah, A. and Asmai, S. (2011). *Open Systems (ICOS)*. 133-138.
- [8] Mousa, H. and Elsis, A. (2013). *Computer Engineering & Systems (ICCES)*. 167-171.
- [9] Nogareda, A. and Camacho, D. (2015). Constraint-based model design for Timetabling problems in secondary schools. *Innovations in Intelligent Systems and Applications (INISTA)*. 1-6.