

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

โครงการเรื่อง ระบบคำนวณภาระงานนอกเวลาและเปลี่ยนแปลงเวลาเข้างาน (Overtime Duration Computed System with Time Attendance Changing Function) เพื่อให้ได้งานวิจัยที่สมบูรณ์จึงได้ทำการศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้องเพื่อนำมากำหนดขอบเขตแนวคิดในการศึกษาวิจัย โดยประกอบด้วยเนื้อหา ดังนี้

- 2.1 หลักการทำงานของระบบคำนวณภาระงานนอกเวลาและเปลี่ยนแปลงเวลาเข้างาน
- 2.2 เทคโนโลยีที่ใช้
- 2.3 งานวิจัยที่เกี่ยวข้อง

2.1 หลักการทำงานของระบบคำนวณภาระงานนอกเวลาและเปลี่ยนแปลงเวลาเข้างาน

ระบบคำนวณภาระงานนอกเวลาและเปลี่ยนแปลงเวลาเข้างาน แต่เดิมนั้นในการส่งคำร้องขอต่าง ๆ เพื่อขอการอนุมัติของบริษัททั่วไปจะเป็นระบบการอนุมัติเป็นลำดับขั้น ซึ่งในการขอทำงานล่วงเวลา หรือ ขอเปลี่ยนเวลาทำงานแต่ละครั้งนั้น พนักงานของบริษัทจำเป็นต้องเขียนคำร้องเพื่อส่งให้หัวหน้าพนักงานพิจารณาทำการอนุมัติ หลังจากหัวหน้าพนักงานทำการอนุมัติคำร้องแล้ว หัวหน้าจำเป็นต้องส่งคำร้องของพนักงานให้กับหัวหน้าแผนกเพื่อพิจารณาและทำการอนุมัติต่อ ซึ่งในการอนุมัติในลักษณะนี้จำเป็นต้องใช้เวลาในการอนุมัตินาน เนื่องจากบางครั้งหัวหน้าของพนักงานหรือหัวหน้าแผนกไม่ได้เข้าบริษัท จะทำให้การอนุมัติล่าช้า ทางผู้พัฒนาจึงคิดที่จะเปลี่ยนระบบคำนวณภาระงานนอกเวลาและเปลี่ยนแปลงเวลาเข้างานให้เป็นระบบแบบออนไลน์ ดังนั้นทีมผู้จัดทำได้เสนอระบบคำนวณภาระงานนอกเวลาและเปลี่ยนแปลงเวลาเข้างาน ในรูปแบบของเว็บไซต์เพื่อแก้ปัญหาการอนุมัติคำขอของพนักงาน โดยระบบยังคงใช้หลักการอนุมัติแบบเป็นลำดับขั้นเหมือนกับการทำงานแบบเก่า เนื่องจากทางบริษัทมีตำแหน่งของพนักงานในแต่ละแผนก ซึ่งจำเป็นต้องมีการอนุมัติแบบเป็นลำดับขั้น ระบบจะมีหน้าการเขียนคำร้องของพนักงานเพื่อทำการขอทำงานล่วงเวลาหรือขอเปลี่ยนวันเวลาการทำงาน เมื่อพนักงานได้ทำการเขียนคำร้องเสร็จ ระบบจะทำการบันทึกคำร้องลงฐานข้อมูลเพื่อไม่ให้ข้อมูลสูญหาย หลังจากนั้นระบบจะทำการส่งอีเมลให้กับหัวหน้าพนักงานและหัวหน้าแผนกโดยอ้างอิงจากตำแหน่งงานภายในบริษัท ซึ่งระบบนี้ได้พัฒนาขึ้นโดยใช้ภาษา C#.net และ MVC เข้ามาใช้ในการพัฒนาหน้าเว็บไซต์ และระบบฐานข้อมูลใช้เป็น SQL Server ในการเก็บคำร้องของพนักงานภายในบริษัท

2.2 เทคโนโลยีที่ใช้

2.2.1 GitLab (Mongkon Jakit, 2018) GitLab ซอฟต์แวร์จัดการโครงการชื่อดัง ประกาศออกเวอร์ชันใหม่ 10.6 ที่สามารถทำงานร่วมกับซอร์สโค้ดที่เก็บบน GitHub ได้แล้ว GitLab เป็นซอฟต์แวร์ที่พัฒนาขึ้นจาก Git โดยมีหน้าที่สองส่วนคือเก็บซอร์สโค้ด (repository) และการจัดการโครงการ (CI/CD ย่อมาจาก continuous integration and continuous delivery) ที่ผ่านมา GitLab ต้องการให้ลูกค้าเก็บซอร์สโค้ดไว้บนโฮสต์ของตัวเอง แต่ในความเป็นจริง ผู้ใช้อาจเก็บซอร์สโค้ดไว้กับผู้ให้บริการรายอื่น ๆ อยู่แล้ว และไม่อยากเปลี่ยนมาเก็บบน GitLab ทำให้สุดท้าย GitLab ต้องยอมเปิดกว้าง ทำงานร่วมกับผู้ให้บริการรายอื่นมากขึ้น แนวทางของ GitLab จะเริ่มจากโฮสต์ที่ยอดนิยมอย่าง GitHub ก่อน แต่เปิดกว้างให้เชื่อมกับรายอื่น ๆ (เช่น BitBucket) ผ่าน API ได้เช่นกัน



ภาพที่ 2-1 ซอฟต์แวร์ GitLab

ที่มา: (<https://about.gitlab.com/>, 2020)

การเปิดกว้างของ GitLab ทำให้โครงการโอเพนซอร์สที่อยู่บน GitHub สามารถใช้บริการ CI/CD เวอร์ชันฟรีของ GitLab ได้ทันที ส่วนลูกค้าองค์กรที่ใช้ GitHub เวอร์ชันเสียเงินก็จะได้ประโยชน์จากการเชื่อมต่อนี้เช่นกันฟีเจอร์อีกอย่างใน GitLab 10.6 คือการเชื่อมต่อกับ Kubernetes ที่แน่นหนาขึ้น ช่วยให้การ deploy โค้ดจาก GitLab ไปยังเครื่องคลัสเตอร์ที่รันด้วย Kubernetes สะดวกมากเพียงแค่คลิกเดียว

2.2.2 C#.Net (Marcus, 2016) C# คือ ภาษาคอมพิวเตอร์ประเภท object-oriented programming พัฒนาโดย Microsoft โดยมีจุดมุ่งหมายในการรวมความสามารถการคำนวณของ C++ ด้วยการโปรแกรมง่ายกว่าของ Visual Basic โดย C# มีพื้นฐานจาก C++ และเก็บส่วนการทำงานคล้ายกับ Java C# ได้รับการออกแบบให้ทำงานกับ .NET platform ของ Microsoft จุดมุ่งหมายคือ อำนวยความสะดวกในการแลกเปลี่ยนสารสนเทศและบริการผ่านเว็บ และทำให้ผู้พัฒนาสร้างโปรแกรมประยุกต์ในขนาดกระทัดรัด C# ทำให้โปรแกรมง่ายขึ้นผ่านการใช้ Extensible Markup Language (XML) และ Simple Object Access Protocol (SOAP) ซึ่งยอมให้เข้าถึงอ็อบ

เจคของโปรแกรมหรือเมธอด โดยปราศจากความต้องการให้ผู้เขียนโปรแกรมเขียนคำสั่งเพิ่มในแต่ละขั้นตอน เนื่องจากผู้เขียนโปรแกรมสามารถสร้างบนคำสั่งที่มีอยู่ แทนที่การคัดลอกซ้ำ C# ภาษา C# ถูกพัฒนาขึ้นโดยเป็นส่วนหนึ่งในการพัฒนาโครงสร้างพื้นฐานของ .NET Framework เป็นการการนำข้อดีของภาษาต่าง ๆ (เช่นภาษา Delphi , ภาษา C++) มาปรับปรุงเพื่อให้มีความเป็น OOP (โปรแกรมเชิงวัตถุ) มากขึ้น ขณะเดียวกันก็ลดความซับซ้อนในโครงสร้างของภาษาลง (เรียบง่ายกว่า ภาษา C++) และมีสิ่งที่เกินความจำเป็นน้อยลง (เมื่อเทียบกับ Java) C# ถูกรับรองจากหน่วยงาน ECMA (หน่วยงานกำหนดมาตรฐานสากลด้านสารสนเทศ) และ ISO และปัจจุบันไมโครซอฟท์ยังพัฒนาภาษานี้อย่างต่อเนื่อง (ปัจจุบันเป็นเวอร์ชัน 3.0)



รูปภาพที่ 2-2 โปรแกรม Visual C#.net

ที่มา: (<https://bit.ly/2Om3dcv>, 2017)

2.2.2.1 เปรียบเทียบภาษา C# กับภาษาอื่น ๆ

ก) ถ้าพูดถึงความใกล้เคียงกับภาษาอื่น ๆ ภาษา C# ใกล้เคียงกับภาษา Java มากที่สุด โดยมีความเหมือนกันถึง 70% ดังนั้นนักเขียนโปรแกรมภาษา Java จึงอาจย้ายมาเขียนภาษา C# ได้โดยศึกษาว่ามีสิ่งใดที่แตกต่างกันบ้าง ภาษา C# ยังมีความคล้ายคลึงกับภาษ C++.NET และภาษา VB.NET เป็นอย่างมาก ทำให้นักเขียนโปรแกรมภาษา C# สามารถอ่าน-เขียนโค้ดในภาษากลุ่มนี้ได้เมื่อฝึกฝนเพียงเล็กน้อย

ข) C# และภาษา Java ทั้งคู่เป็นแบบสืบจากคลาสหลักได้คลาสเดียว ขณะที่ภาษา C++ สามารถสืบจากคลาสหลักได้มากกว่าหนึ่ง (Multiple inheritance) โดยภาษา C# และภาษา Java ใช้ Interface มาทดแทน Multiple inheritance เหมือนกันทั้งคู่

ค) สิ่งที่ภาษา C# และ Java มีร่วมกันคือเรื่อง Garbage Collection แต่ไม่มีใน C++ จึงทำให้ดูเหมือนว่าภาษา Java ต่อยอดมาจากภาษา C++ และ C# ต่อยอดมาจาก Java อีกที ที่เป็นเช่นนั้นเพราะทั้ง Java และ C# มีต้นสายมาจาก C++ ทำให้สองภาษานี้ดูคล้ายกัน แต่

ภาษา C# ไม่ใช่ภาษา Java มันมีกลไกที่เป็นเอกลักษณ์หลายอย่าง เช่น พารามิเตอร์แบบ reference และ output การจัดเก็บ object ไว้ใน stack (struct) การทำ Versioning และยังมีสิ่งใหม่ ๆ ที่เป็นข้อดี เช่น delegate, properties และ operator overloading ซึ่งจะไม่พบในภาษา Java

2.2.2.2 จุดเด่นหลัก ๆ ของภาษา C# มีดังนี้

ก) Component oriented – เป็นภาษาที่เน้นขึ้นส่วนโดยถูกออกแบบมาเป็นอย่างดีทำให้สามารถนำมาใช้ต่อกันเป็นอะไรก็ได้

ข) สิ่งต่าง ๆ ใน C# เป็นอ็อบเจกต์ทั้งหมด

ค) เป็นภาษา ที่ทนทาน (robust) - ทนต่อความผิดพลาด ไม่ทำให้ระบบแองก์หรือระบบทำงานช้า เพราะ C# มีข้อดีคือ garbage collection , exception , type-safety และ versioning

ง) ภาษา C# จัดเตรียมกลไกไว้หลายอย่างที่ช่วยให้ผู้เขียนโปรแกรมสามารถนำโค้ดที่เขียนไว้ใน โปรเจค หนึ่งไปใช้กับอีกโปรเจคหนึ่งได้ง่าย นอกจากนั้นภาษา C# ยังสามารถเรียกใช้คลาสหลายพันคลาสใน .NET Framework ได้โดยตรง ทำให้ลดเวลาการพัฒนาซอฟต์แวร์ได้มาก

2.2.3 Bootstrap Framework v.3 (Suranart Niamcome, 2013)

2.2.3.1 Bootstrap นั้น ถือว่าเป็น front-end framework ที่ได้รับความนิยมมากที่สุดขณะนี้เลยก็ว่าได้ โดยจุดเด่นของ bootstrap จะอยู่ที่ การใช้งานที่ง่าย เรียนรู้ได้เร็ว แล้วยังรวบรวมสิ่งที่เราใช้บ่อย ๆ มาให้เกือบหมดเลย อย่างไรก็ตาม bootstrap ยังคงมีการพัฒนามาอย่างต่อเนื่อง

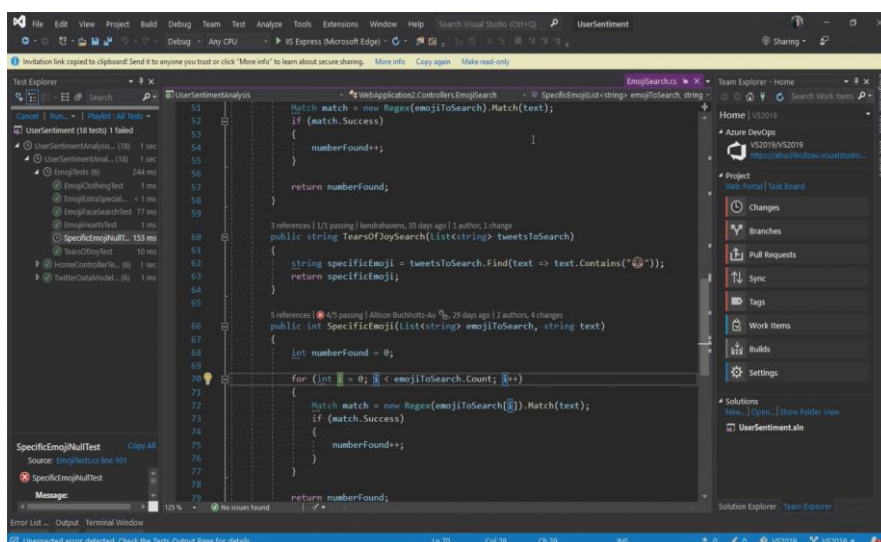
2.2.3.2 Theme default theme ของ bootstrap 3 ได้เปลี่ยนมาเป็นแบบ flat design Responsive Design bootstrap 3 เปลี่ยนมาใช้ mobile-first approach หรือการพัฒนาเว็บโดยโฟกัสไปที่หน้าจอขนาดเล็กก่อน ซึ่งวิธีนี้ ถือเป็น best practice ของการทำ responsive web ในปัจจุบัน นอกจากนี้ css ของ bootstrap 3 จะรวม responsive features มาให้ในตัวเลย

2.2.3.3 Grid System grid system ใน bootstrap 3 จะถูกยุบรวมให้เหลือแบบเดียว คือ mobile-first responsive fluid grid system โดยยังคงมีจำนวน 12 คอลัมน์ เหมือนเดิม แต่จะมีคุณสมบัติใหม่ที่เพิ่มเข้ามาก็คือ เราจะสามารถกำหนดขนาดของคอลัมน์ให้กับ devices ในแต่ละแบบได้

2.2.3.4 Components bootstrap 3 ได้เพิ่ม components เข้ามาใหม่ 2 อันด้วยกัน ได้แก่ list group และ panel แต่ในขณะเดียวกัน ก็ได้เอา components ออกไปหลายอันด้วยกัน ไม่ว่าจะเป็น submenus, typeahead รวมไปถึง accordion นอกจากนี้ bootstrap 3 ยังได้ปรับปรุง components ต่างๆ ที่เคยมีอยู่เดิมให้ดีขึ้นอีกด้วย

2.2.3.5 Browser support bootstrap 3 ยกเลิกการรองรับ Internet Explorer 7 แล้ว ส่วน Internet Explorer 8 นั้นยังคงรองรับอยู่ แต่เราต้องใส่ respond.js เพิ่มเข้าไปด้วย เพื่อให้ใช้ media queries ได้

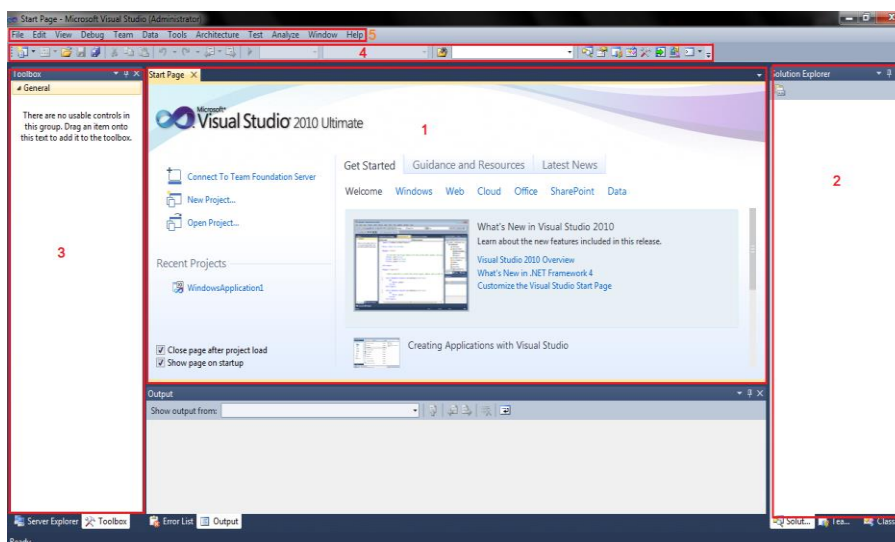
2.2.4 Visual Studio (Kachapot, 2019) Visual Studio คือ โปรแกรมตัวหนึ่งที่เป็นเครื่องมือที่ช่วยพัฒนาซอฟต์แวร์และระบบต่างๆ ซึ่งสามารถติดต่อสื่อสารพูดคุยกับคอมพิวเตอร์ได้ในระดับหนึ่งแล้ว แต่ยังไม่สามารถพัฒนาเป็นระบบเองได้ เหมาะสมสำหรับภาษา VB และ VB.NET เนื่องจากไมโครซอฟต์ได้พัฒนาโปรแกรมและภาษาขึ้นมาควบคู่กันเพื่อให้ใช้งานได้ซึ่งกันและกัน ซึ่งนักโปรแกรมเมอร์จะนำเครื่องมือมาใช้ในการพัฒนาต่อยอดให้เกิดเป็นระบบต่าง ๆ หรือเป็นเว็บไซต์ และแอปพลิเคชันต่าง ๆ Visual Studio ทำอะไรได้บ้าง เป็นเครื่องมือที่ใช้สร้างโปรแกรม ต่าง ๆ เช่น - โปรแกรมที่รันบน ระบบปฏิบัติการ windows เช่น โปรแกรมคำนวณเลข - โปรแกรมฐานข้อมูล เช่น Microsoft access, Microsoft SQL server - คอมโพเนนต์ทางด้าน Active X - โปรแกรมที่รันบน อินเทอร์เน็ต



ภาพที่ 2-3 โปรแกรม Visual Studio

ที่มา: (<https://bit.ly/2po6DCZ>, 2019)

ส่วนประกอบของ Visual Basic โดยทั่วไป เราจะ ใช้ Project Standard. EXE ซึ่งเป็นการเขียนโปรแกรมที่รันบนวินโดว Project คือ กลุ่มของ File ที่เราจะ นำมารวมกันเพื่อสร้างโปรแกรมรายละเอียดของส่วนประกอบต่าง ๆ ของหน้าจอ - Menu bar - Tool bar - Tool box - Project explorer - Properties window - Form



ภาพที่ 2-4 ส่วนประกอบของ Visual Studio

ที่มา: (<https://bit.ly/33UOiwe>, 2013)

2.2.5 TortoiseGit (Ake Exocist, 2017) Git คือการทำงานแบบ Distributed Version Control Systems Git ออกแบบมาให้ทำงานกระจายแบบไม่มีศูนย์กลาง ทุกเครื่องทำงานเป็น VCS ด้วยตัวเองได้นั้นหมายความว่าไม่จำเป็นต้องมี Server กลางก็ได้ สามารถใช้เครื่องส่วนตัวทำเป็น VCS ได้เลย แต่ถ้าต้องทำงานร่วมกันหลายๆเครื่อง ก็ต้องใช้ Server เป็นตัวกลางในการรวมข้อมูล

2.3 งานวิจัยที่เกี่ยวข้อง

2.3.1 (วรรณลักษณ์ เหล่าทวิทรัพย์, 2545) “การจัดตารางการทำงานของพนักงานแบบ สับเปลี่ยนการทำงานสำหรับโรงภาพยนตร์” งานวิจัยนี้ได้พัฒนาวิธีการแก้ปัญหาการจัดตารางการทำงานของพนักงานแบบสับเปลี่ยนหน้าที่การทำงานสำหรับโรงภาพยนตร์ โดยได้พัฒนาวิธีเชิงพันธุกรรม (Genetic Algorithm) และ วิธีฮิวริสติก (Heuristic Method) ร่วมกัน เพื่อให้สามารถตอบสนองต่อเป้าหมายหลัก 3 เป้าหมาย ซึ่งประกอบด้วย งบประมาณในการจ้างงานของพนักงานประจำ งบประมาณในการจ้างงานของพนักงานชั่วคราว และความสามารถในการตอบสนองความต้องการจำนวนพนักงานขั้นต่ำใน แต่ละช่วงเวลา วิธีฮิวริสติกจะทำการสร้างประชากรเบื้องต้น (Population Size) ที่สามารถ ตอบสนองต่อความต้องการจำนวนพนักงานขั้นต่ำในแต่ละช่วงเวลา แล้วส่งผลไปยังวิธีเชิง พันธุกรรมที่พัฒนาขึ้นเพื่อค้นหาคำตอบที่ดีที่สุดสำหรับการจัดตารางการทำงานแบบสับเปลี่ยนหน้าที่ การจัดตารางงานถูกแบ่งออกเป็น 4 กรณีตามช่วงระดับความนิยมคือระดับความนิยมต่ำปานกลาง สูง และสูงมาก ซึ่งผลการเปรียบเทียบค่าใช้จ่ายพบว่าในระดับความนิยมต่ำสามารถประหยัดค่าใช้จ่ายได้ 72,192 บาทต่อปี ระดับความนิยมปานกลางสามารถประหยัดค่าใช้จ่ายได้ 11,616 บาทต่อปี ระดับความนิยมสูงสามารถประหยัดค่าใช้จ่ายได้ 36,336 บาทต่อปีและระดับความ

นิยมสูงมากสามารถประหยัดค่าใช้จ่ายได้ 43,488 บาทต่อปี ผลสรุปจากการวิจัยแสดงให้เห็นว่าการพัฒนาวิธีเชิงพันธุกรรมร่วมกับวิธีฮิวริสติกสำหรับจัดตารางการทำงานของพนักงานแบบสลับเปลี่ยนหน้าที่การทำงาน ช่วยปรับปรุงตารางการทำงานและสามารถลดค่าใช้จ่ายในการจัดจ้างพนักงาน

2.3.2 (ลาวิเนีย ราชราม, 2562) “คนไทยทำงานหนักเสี่ยง Burnout “ อาการอยากหยุดงาน แต่หยุดไม่ได้เพราะติดงาน ดูเหมือนจะเป็นสิ่งที่มนุษย์ออฟฟิศทุกคนต้องเผชิญกัน ซึ่งภาวะดังกล่าวมีผลกระทบต่อสุขภาพทางจิตใจได้เช่นกัน ยังมีผลต่อประสิทธิภาพของการทำงานด้วย โดยจากงานศึกษาของคณะจิตวิทยา จุฬาลงกรณ์มหาวิทยาลัย เรื่องการติดงานและความผูกใจมั่นในงาน: สาเหตุเชิงแรงจูงใจ และผลด้านความเหนื่อยหน่ายในงาน ระบุถึงการติดงาน (Workaholism) เป็นสภาวะที่คนมีแนวโน้มในการทำงานหนักมากเกินไปจนเกิดความจำเป็น ซึ่งการทำงานดังกล่าวเกิดขึ้นโดยไม่สามารถต้านทานได้ ทำให้คิดหรือรู้สึกหมกมุ่นกับงาน และแสดงออกในรูปของการทำงานอย่างหมกมุ่น

2.3.3 (วิมลรัตน์ ดำรงสันติพิทักษ์, 2546) “การจัดพนักงานทำงานล่วงเวลาและทำโปรแกรมคอมพิวเตอร์เพื่อลดเวลาในการจัดตารางการทำงานล่วงเวลาและลดค่าใช้จ่ายในการทำงานดังกล่าวของพนักงานฝ่ายซ่อมบำรุงอุปกรณ์ภาคพื้น บริษัท การบินไทย จำกัด (มหาชน)” ซึ่งมีพนักงาน 61 คน แยกเป็นช่างที่มีความชำนาญเฉพาะอุปกรณ์ 23 คน และช่างทั่วไป 38 คน วิธีการที่ใช้จัดตารางการทำงานล่วงเวลาในปัจจุบันจะมีเจ้าหน้าที่ 1 คน และใช้เวลา 5 วัน ในการจัดพนักงานให้ทำงานล่วงเวลาใน 1 เดือน และเมื่อพิจารณาค่าล่วงเวลาที่จ่ายจริงประมาณเดือนละ 579,750 บาท ตามแผนที่กำหนดโดยเจ้าหน้าที่เปรียบเทียบกับงบประมาณที่ตั้งไว้ เดือนละประมาณ 259,000 บาท ปรากฏว่าจำนวนเงินที่จ่ายจริงสูงกว่างบประมาณที่ตั้งไว้เดือนละ 320,750 บาท หรือ 124% ในช่วงเวลาดังตั้งแต่วันที่ 1 ตุลาคม 2544 ถึง 30 กันยายน 2545 เพื่อแก้ไขปัญหาดังกล่าวข้างต้นจึงได้นำเอาหลักของโปรแกรมเชิงเส้นตรง มาประยุกต์ใช้เพื่อสร้างแบบจำลองทางคณิตศาสตร์และสร้างเป็นโปรแกรมคอมพิวเตอร์จัดให้พนักงานทำงานล่วงเวลาเพื่อที่จะลดเวลาในการจัดและค่าใช้จ่ายในการทำงานล่วงเวลาที่เกิดขึ้น โปรแกรมที่ได้จัดทำขึ้นใช้ภาษา Visual Basic 6.0 และใช้ LINGO 5.0 ในการประมวลผล จากการทดลองใช้โปรแกรมที่ได้จัดทำขึ้นในการจัดตารางการทำงานล่วงเวลาของพนักงานในแต่ละเดือนลดลงเหลือประมาณ 10 นาที และเมื่อคำนวณค่าล่วงเวลาในช่วงเวลาดังกล่าวข้างต้น ถ้าใช้โปรแกรมคอมพิวเตอร์จัดตาราง การทำงานล่วงเวลาพบว่าการทำงานล่วงเวลาเฉลี่ยในแต่ละเดือนเป็น 279,609 บาท ลดลงจากการจัดโดยวิธีปัจจุบัน 300,141 บาท หรือลดลง 51.7% จึงสรุปได้ว่าโปรแกรมคอมพิวเตอร์ที่ได้จัดทำขึ้นสามารถลดเวลาในการจัดพนักงานได้อย่างมาก และลดการจ่ายเงินค่าทำงานล่วงเวลา