การสอนการพัฒนาซอฟต์แวร์ตามแนวคิดแบบ Agile
Teaching Software Development According the Agile Concept

นภารัตน์ ชูไพร[1*] สุภาณี เส็งศรี[2] และกอบสุข คงมนัส[3]
Napharat Chooprai[1*] Supanee Sengsri[2] and Kobsook Kongmanus[3]

Received: (November 7, 2023) Revised: (February 7, 2024) Accepted: (February 21, 2024)

บทคัดย่อ

การสอนการพัฒนาซอฟต์แวร์โดยใช้แนวคิด Agile เป็นแนวคิดในการพัฒนาซอฟต์แวร์ที่ทำให้เกิดความรวดเร็ว สามารถปรับเปลี่ยนการทำงานได้ตลอดเวลา มีความยืดหยุ่น และมีการทำงานเป็นทีม เป็นวิธีในการบริหารจัดการเพื่อตอบสนองความต้องการของลูกค้าให้ดีที่สุด ระเบียบวิธีแบบ Agile มีลักษณะการทำงานแบบวนซ้ำ เน้นย้ำการพัฒนา การทำงานร่วมกัน การยอมปรับการเปลี่ยนแปลง คำติชมของลูกค้า และมีการนำไปใช้อย่างกว้างขวางเพื่อความยืดหยุ่นและการปรับตัว ซึ่งการพัฒนาซอฟต์แวร์ตามแนวคิดแบบ Agile ประกอบด้วย 1) การศึกษาความต้องการของผู้มีส่วนได้ส่วนเสีย 2) การวางแผนและออกแบบ 3) การพัฒนาซอฟต์แวร์ 4) การทดสอบการทำงานของซอฟต์แวร์ 5) การปรับปรุงซอฟต์แวร์ 6) การให้ข้อเสนอแนะความคิดเห็น และ 7) การประเมินผล โดยผู้สอนต้องเป็นผู้นำทางการเรียนรู้ มีการวางแผนการเรียนที่มีความยืดหยุ่น มีการทำงานเป็นทีม ส่งเสริมการเรียนรู้ด้วยตนเองให้กับผู้เรียน มีการสร้างบรรยากาศที่เป็นกันเอง เพื่อให้ผู้เรียนได้เรียนรู้เข้าใจหลักการของ Agile ที่สามารถนำไปใช้ในการพัฒนาซอฟต์แวร์ ซึ่งเป็นทักษะที่มีความจำเป็นในศตวรรษที่ 21 ที่ต้องพัฒนาสมรรถนะผู้เรียนให้ทันสมัยกับการเปลี่ยนแปลงที่เกิดขึ้นรอบด้าน กว้างขวาง รู้เท่าทันโลก ก้าวทันความเจริญก้าวหน้าทางวิทยาการและเทคโนโลยี ต่อการเรียนรู้และการพัฒนาตนเองอย่างต่อเนื่องตลอดชีวิต สู่การเป็นคนไทยที่มีทักษะสูง เป็นนวัตกร นักคิด ผู้ประกอบการ เกษตรกรยุคใหม่

คำสำคัญ: การสอน, การพัฒนาซอฟต์แวร์, แนวคิดแบบ Agile

➢ บทความวิชาการ (Academic Articles)

[1,2,3] คณะศึกษาศาสตร์ มหาวิทยาลัยนเรศวร จังหวัดพิษณุโลก
Faculty of Education, Naresuan University, Phitsanulok Province
*Corresponding Author E-mail: napharatc66@nu.ac.th

Abstract

Widely adopted for its inherent speed, flexibility, adaptability, and collaborative teamwork, Agile serves as a robust framework for navigating the dynamic landscape of software development. For management, it stands out as an optimal approach that promptly responds to evolving customer needs, places a strong emphasis on development and team collaboration, embraces changes, and incorporates customer feedback. Software development according to Agile concepts consists of 1) hearing steak holders' needs 2) planning and designing 3) software developing 4) trialing 5) redesigning 6) Reviewing and 7) evaluating. Teachers, in this context, assume the role of learning leaders, facilitating the learning process with flexible learning plan, learning-friendly atmosphere, and teamwork-oriented mind. This instructional approach promotes self-directed learning among students and enables them to apply the realm of software development which is an imperative skill for 21st century as the students are encouraged to cultivate competencies that enable them to stay abreast of global changes and advancements in science and technology. This is essential for individuals aspiring to become highly skilled professionals, Innovators, entrepreneurs, and modern farmers in the context of contemporary Thailand.

Keywords: Teaching, Software Development, Agile Concept

บทนำ

การเปลี่ยนแปลงที่เกิดขึ้นอย่างรวดเร็ว ส่งผลทำให้เกิดการเปลี่ยนแปลงอย่างมากต่อพฤติกรรมในการใช้ชีวิตรวมถึงการเรียนรู้โลกแห่งการเรียนรู้ได้พัฒนาไปอย่างมากจากการที่มีระบบอินเทอร์เน็ตและการพัฒนาของเทคโนโลยีดิจิทัล เทคโนโลยีเป็นปัจจัยสำคัญในการส่งเสริมการเรียนรู้ของผู้เรียนและเข้ามามีส่วนสำคัญในด้านข้อมูล เครื่องมือในการเรียนรู้ รวมถึงการสร้างองค์ความรู้และประสบการณ์ให้กับผู้เรียน การจัดการทรัพยากรการเรียนรู้ร่วมกันและการสื่อสารการแลกเปลี่ยนเรียนรู้ ในการจัดการเรียนรู้เพื่อพัฒนาผู้เรียนในยุคดิจิทัล ผู้สอนต้องศึกษาและทำความเข้าใจในองค์ประกอบที่สำคัญ ได้แก่ การเรียนรู้ในยุคดิจิทัล ผู้เรียนและผู้สอนในยุคดิจิทัล การรู้ดิจิทัล การออกแบบการจัดการเรียนรู้ การจัดสภาพแวดล้อม ของการเรียนรู้ดิจิทัล และการประเมินผลการเรียนรู้ในยุคดิจิทัล เพื่อที่จะบริหารจัดการให้เอื้อต่อการเรียนรู้ของผู้เรียนให้มีประสิทธิภาพสูงสุด (พงษ์พัชรินทร์ พุธวัฒนะ, 2564) ท่ามกลางกระแสการเปลี่ยนแปลงของโลกที่เป็นไปอย่างรวดเร็ว รวมถึงการเข้ามาของเทคโนโลยีและนวัตกรรมใหม่ ๆ ส่งผลให้สภาพแวดล้อม ของธุรกิจ ตลาด และพฤติกรรมของลูกค้าเปลี่ยนไปด้วยเช่นกัน ทำให้องค์กรต้องตื่นตัวอยู่ตลอดเวลา การดำเนินงานต้องมีความง่ายและรวดเร็วมากกว่าเดิม เพื่อจะสามารถปรับตัวให้ทันต่อความเปลี่ยนแปลงทั้งหลายได้ทันท่วงที องค์กรต่าง ๆ จึงได้นำแนวทาง Agile Organization คือ องค์กรที่สามารถปรับตัวและตอบสนองต่อสถานการณ์ที่เปลี่ยนแปลงได้อย่างรวดเร็ว ผ่านการปรับเปลี่ยนแนวคิดและสร้างทัศนคติใหม่ (New Mindset) เพื่อการทำงานที่รวดเร็ว และคล่องตัวกว่าเดิม ซึ่งจะต้องมีการยึดลูกค้าเป็นศูนย์กลาง (Customer Centric) ลดขั้นตอนการทำงานที่ไม่จำเป็นออกไป เพื่อการทำงานที่ง่ายและรวดเร็วกว่าเดิม (รวิศ หาญอุตสาหะ, 2566)

Agile เป็นแนวคิดและวิธีการบริหารงานขององค์กรยุคดิจิทัลซึ่งเป็นยุคที่มีการเปลี่ยนแปลงอย่างรวดเร็ว องค์กรจะต้องปรับตัวเพื่อสอดรับกับการสภาวะแวดล้อมที่เปลี่ยนไปได้ทันท่วงที ดังนั้น “ความคล่องตัว” ในการปรับเปลี่ยนการทำงานจึงมีความสำคัญเป็นอย่างยิ่ง (พิชูร ธนบดีกิจ และจีราภา อินธิแสง โธณีม, 2565) ข้อสำคัญหลักของ Agile คือการทำงานร่วมกันเป็นทีม การทำงานนั้นจะตั้งอยู่บนเหตุและผลที่เป็นข้อเท็จจริง เพื่อให้แต่ละฝ่ายช่วยกันหาทางออกที่ดีที่สุดให้กันและกัน Agile นำไปสู่การพัฒนากรอบงาน รูปแบบ กระบวนการทำงานที่เป็นระบบ อีกทั้งยังมีการร่วมกับปรับเปลี่ยนพฤติกรรมให้ดีขึ้นอย่างต่อเนื่องอีก ซึ่งจะทำให้การพัฒนาสินค้า บริการและบุคลากรมีต่อไปเรื่อย ๆ อย่างไม่หยุดสิ้น การที่องค์กรนำ Agile ไปใช้จริง ๆ นั้นเป็นอีกทางเลือกหนึ่งที่ทำให้การทำงานเป็นทีมมีประสิทธิภาพมากขึ้น (Upplychainguru, 2023)

จากที่กล่าวมาข้างต้น ผู้เขียนได้เล็งเห็นความสำคัญของการนำแนวคิดการพัฒนาซอฟต์แวร์ตามแนวคิดของ Agile มาช่วยส่งเสริมกิจกรรมการเรียนการสอนให้ผู้เรียนมีความรู้ความสามารถในการพัฒนาซอฟต์แวร์ จึงรวบรวมเอกสารและงานวิจัยที่เกี่ยวข้องเพื่อนำเสนอกิจกรรมการสอนการพัฒนาซอฟต์แวร์ตามแนวคิดแบบ Agile

ความหมายของแนวคิดแบบ Agile

ชนัญชิดา เลิศจะบก และจงกล จันทร์เรือง (2563) ได้ให้ความหมายว่า Agile เป็นแนวคิดในการพัฒนาระบบสารสนเทศที่มุ่งเน้น การทำงานที่รวดเร็ว สามารถปรับเปลี่ยนแก้ไขการทำงานที่ตรงตามความต้องการของผู้ใช้งานที่มีความยืดหยุ่น เน้นการทำงานเป็นทีมที่มีการกำหนดเป้าหมายระยะสั้น ทำงานทีละส่วนงานแบบมีส่วนร่วมของผู้พัฒนาระบบ กับทีมของผู้ใช้งานระบบ เมื่อความต้องการของผู้ใช้งานเปลี่ยนแปลงก็สามารถปรับแก้ได้ตลอดเวลา

López-Alcarria, Olivares-Vicente and Poza-Vilches (2019) ได้ให้ความหมายว่า Agile เป็นความสามารถในการสร้างและตอบสนองต่อการเปลี่ยนแปลง เป็นวิธีในการจัดการและประสบความสำเร็จในสภาพแวดล้อมที่ไม่แน่นอนจริง ๆ แล้วเป็นเรื่องเกี่ยวกับการคิดว่าคุณจะเข้าใจสิ่งที่เกิดขึ้นในสภาพแวดล้อมที่คุณเผชิญอยู่ทุกวันนี้ได้อย่างไร ระบุความไม่แน่นอนที่คุณกำลังเผชิญอยู่ และหาคำตอบว่าคุณจะปรับตัวเข้ากับสิ่งนั้นได้อย่างไรในขณะที่คุณดำเนินไป วิธีการแบบ Agile เป็นกลุ่มแนวทางการจัดการโครงการที่ขัดแย้งกับเทคนิคการจัดการโครงการแบบ Waterfall เป็นวิธีการแบบดั้งเดิมในหลาย ๆ ด้าน ซึ่งมีการกำหนดโครงการไว้ทั้งหมดก่อนที่จะเริ่มดำเนินการ หมายความว่าในรูปแบบการจัดการโครงการแบบดั้งเดิม ข้อกำหนดเฉพาะ ทรัพยากร งาน และเวลาทั้งหมดจำเป็นต้องได้รับการกำหนดก่อนที่จะเริ่มงานจริง

Suksomboon (2023) ได้ให้ความหมายว่า Agile เป็นหลักการทางวิศวกรรมซอฟต์แวร์ที่รวมเอาแนวทางพัฒนาเข้าด้วยกันเพื่อมุ่งสร้างความพึงพอใจให้กับลูกค้าและสามารถส่งมอบซอฟต์แวร์แบบค่อยเพิ่มขึ้นแก่ลูกค้าโดยใช้ทีมงานขนาดเล็กที่กระตือรือร้น ใช้วิธีการแบบไม่เป็นทางการผลิตงานด้านวิศวกรรมซอฟต์แวร์ถ้าจำเป็น และใช้วิธีการแบบเรียบง่าย ส่วนแนวทางการพัฒนาเน้น การส่งมอบมากกว่าการวิเคราะห์ออกแบบ และการสื่อสารอย่างต่อเนื่องกับลูกค้า

Phongphet, Chumpa and Thumpanit (2023) ได้ให้ความหมายว่า Agile เป็นแนวคิดพัฒนาระบบสารสนเทศที่มุ่งเน้นการทำงานที่รวดเร็ว สามารถปรับเปลี่ยนแก้ไขการทำงานที่ตรงตามความต้องการ

ของผู้ใช้งาน ที่มีความยืดหยุ่น เน้นการทำงานเป็นทีมที่มีการกำหนดเป้าหมายทำงานทีละส่วนงานแบบมีส่วนร่วมของผู้พัฒนา ระบบกับทีมของผู้ใช้งานระบบ

สรุปได้กว่า Agile เป็นแนวคิดในการพัฒนาซอฟต์แวร์ที่มุ่นเน้นทำให้เกิดความรวดเร็ว สามารถปรับเปลี่ยนแก้ไขการทำงานที่ตรงตามความต้องการของลูกค้าได้ตลอดเวลา มีความยืดหยุ่น และมีการทำงานเป็นทีม มีการดำหนดเป้าหมายของงานในระยะเวลาสั้น ๆ โดยมีการทำงานทีละส่วน มีการทำงานร่วมระหว่างผู้พัฒนาซอฟต์แวร์กับลูกค้า เป็นวิธีในการบริหารจัดการเพื่อตอบสนองความต้องการของลูกค้าให้ดีที่สุด

หลักการสำคัญของแนวคิด Agile

การศึกษาหลักการที่สำคัญของแนวคิดแบบ Agile ผู้เขียนได้สรุปการนำเสนอหลักการของนักวิชาการ (Al-Saqqa, Sawalha and AbdelNabi, 2020; López-Alcarria, Olivares-Vicente and Poza-Vilches, 2019; Pócsová, Bednárová, Bogdanovská and Mojžišová, 2020) ที่ได้กล่าวถึงหลักการสำคัญของแนวคิด Agile ซึ่งมี 4 หลักการ ดังนี้

1. การให้ความสำคัญกับตัวผู้ทำงานและการมีปฏิสัมพันธ์ระหว่างผู้ทำงาน มากกว่าขั้นตอนวิธีการหรือเครื่องมือ

2. การสร้างซอฟต์แวร์หรือผลิตภัณฑ์ที่สามารถนำไปใช้งานได้จริง มากกว่าเอกสาร คู่มือ หรือการใช้เวลาในการวางแผนเอกสารงาน

3. การทำงานร่วมกับลูกค้า มากกว่าการต่อรองสัญญากับลูกค้า

4. การยอมรับปรับเปลี่ยนซอฟต์แวร์หรือผลิตภัณฑ์ ตามความเปลี่ยนแปลงของความต้องการของลูกค้า มากกว่าการทำตามแผนการเพียงอย่างเดียว

12 คำแถลงอุดมการณ์ของ Agile

คำแถลงอุดมการณ์ของ Agile 12 ประการ โดยโปรแกรมเมอร์ชั้นแนวหน้า 17 คนที่ประสบปัญหาการบริหารโครงการแบบเก่าไม่สามารถตอบโจทย์กับการพัฒนาซอฟต์แวร์ได้อีกต่อไป เนื่องจากช่องว่างระหว่างความต้องการทางธุรกิจซึ่งอ้างอิงความต้องการของลูกค้าที่เปลี่ยนอย่างรวดเร็ว จากการทบทวนวรรณกรรมและงานวิจัยที่เกี่ยวข้อง มีรายละเอียดดังนี้ (Al-Saqqa, Sawalha and AbdelNabi, 2020; Gheorghe, Gheorghe and Iatan, 2020; Saltz and Heckman, 2020; Sharp, Mitchell and Lang, 2020)

หลักการที่ 1 การสร้างความพึงพอใจให้กับลูกค้า ด้วยการส่งมอบซอฟต์แวร์ตั้งแต่ต้น และต่อเนื่องจะสร้างความไว้วางใจและความยืดหยุ่นระหว่างลูกค้าและทีมพัฒนา ทีมพัฒนาจะได้รับผลตอบรับเกี่ยวกับความถูกต้องของความเข้าใจในความต้องการของผลิตภัณฑ์ ผ่านการให้ผลตอบรับของลูกค้า

หลักการที่ 2 การยอมรับความเปลี่ยนแปลง เนื่องจากการเปลี่ยนแปลงเกิดขึ้นตลอด กระบวนการ Agile มีการควบคุมการเปลี่ยนแปลง เพื่อความได้เปรียบทางการแข่งขันของลูกค้า

หลักการที่ 3 การส่งมอบซอฟต์แวร์บ่อย ๆ การส่งมอบงานที่ใช้งานได้จริงอย่างสม่ำเสมอ จากสองสามสัปดาห์ถึงสองสามเดือน โดยเลือกใช้ระยะเวลาที่สั้น ๆ จะทำให้ได้ซอฟต์แวร์ที่ตรงกับความต้องการของลูกค้ามากที่สุด

หลักการที่ 4 การทำงานร่วมกัน นักธุรกิจและนักพัฒนาต้องทำงานร่วมกันทุกวันตลอดทั้งโครงการ มีจุดมุ่งหมายเพื่อให้ข้อเสนอแนะและตอบคำถามของทีมพัฒนาให้มีความเข้าใจตรงกันในการทำงาน

หลักการที่ 5 การสนับสนุนซึ่งกันและกัน เพื่อสร้างแรงบันดาลใจ สร้างสภาพแวดล้อมและการสนับสนุนที่ดีในการทำงานของทีม และไว้วางใจให้ทีมทำงานให้สำเร็จ สมาชิกในทีมเป็นปัจจัยที่สำคัญที่สุดในความสำเร็จและบรรลุเป้าหมาย

หลักการที่ 6 การปฏิสัมพันธ์คุยกันซึ่ง ๆ หน้า เป็นวิธีการที่มีประสิทธิภาพและประสิทธิผลมากที่สุดในการถ่ายทอดข้อมูลภายในทีมพัฒนาที่มีการสื่อสารแบบเห็นหน้ากัน หลักการนี้เน้นการสื่อสารโดยตรงของบุคคลในทีม

หลักการที่ 7 ความคืบหน้าของงานวัดผลจากงานที่ใช้ได้จริง ซอฟต์แวร์ที่สามารถทำงานได้จริงจะเป็นตัววัดความก้าวหน้าของการทำงาน การแบ่งผลิตภัณฑ์ออกเป็นชิ้นเล็ก ๆ และการส่งมอบงานบ่อย ๆ เป็นวิธีการวัดความคืบหน้าที่ดี ซึ่งให้ผลที่เที่ยงตรงมากขึ้นผ่านโค้ดที่กำลังทำงานอยู่

หลักการที่ 8 การพัฒนางานที่ยั่งยืน กระบวนการแบบ Agile ส่งเสริมการพัฒนาที่ยั่งยืน ทุกฝ่ายจะรักษาอัตราเร็วในการทำงานร่วมกันให้คงที่ การก้าวที่คงที่ได้อย่างไม่มีกำหนด การพัฒนาที่ยั่งยืนหมายความว่าทีมควรรักษาจังหวะให้คงที่

หลักการที่ 9 พัฒนาความเป็นเลิศอย่างต่อเนื่อง การเอาใจใส่ความเป็นเลิศด้านเทคนิคและการออกแบบที่ดีอย่างต่อเนื่องจะช่วยเพิ่มความเป็น Agile การส่งมอบโค้ดที่มีคุณภาพสูงสุดเป็นสิ่งจำเป็นที่ควรได้รับเสมอ แม้ว่าจะต้องปรับโครงสร้างโค้ดใหม่เพื่อตอบสนองต่อการเปลี่ยนแปลงข้อกำหนดก็ตาม

หลักการที่ 10 ทำในสิ่งที่สำคัญและกล้าละสิ่งที่ไม่สำคัญ ความเรียบง่ายเป็นศิลปะในการเพิ่มปริมาณงานที่เป็นสิ่งจำเป็น จุดมุ่งหมายคือการผลิตผลิตภัณฑ์ที่เรียบง่ายและสามารถรับมือกับการเปลี่ยนแปลงกะทันหันและในขณะเดียวกันก็ตอบสนองความต้องการของลูกค้าได้

หลักการที่ 11 ให้อำนาจการตัดสินใจกับทีม การออกแบบที่ดีที่เกิดขึ้นจากการบริหารจัดการของทีม ซึ่งแบ่งปันความรับผิดชอบของโครงการและกำหนดวิธีที่ดีที่สุดในการจัดการพวกเขาผ่านความรู้ที่ได้รับตลอดกระบวนการพัฒนาเพื่อให้ได้โครงสร้างที่เหมาะสมที่สุด

หลักการที่ 12 ร่วมกันปรับเปลี่ยนพฤติกรรมให้ดีขึ้นต่อเนื่อง ทีมจะต้องย้อนกลับไปตรองดูสิ่งที่ผ่านมาเพื่อหาทางที่จะพัฒนาความมีประสิทธิผลของทีม แล้วนำสิ่งเหล่านั้นมาปรับปรุงและเปลี่ยนแปลงพฤติกรรมของทีม

เนื่องจากสภาพแวดล้อมโดยรอบอาจมีการเปลี่ยนแปลง ทีมจึงต้องปรับโครงสร้าง ความสัมพันธ์และพฤติกรรมใหม่โดยยึดตามนั้น กล่าวคือ ในแต่ละรอบ ทีมยังคงปฏิบัติตามแนวทางปฏิบัติที่ตอบสนองความต้องการของตน และปรับเปลี่ยนแนวทางปฏิบัติที่เป็นอุปสรรคขวางทาง

การพัฒนาระบบซอฟต์แวร์แบบ Agile

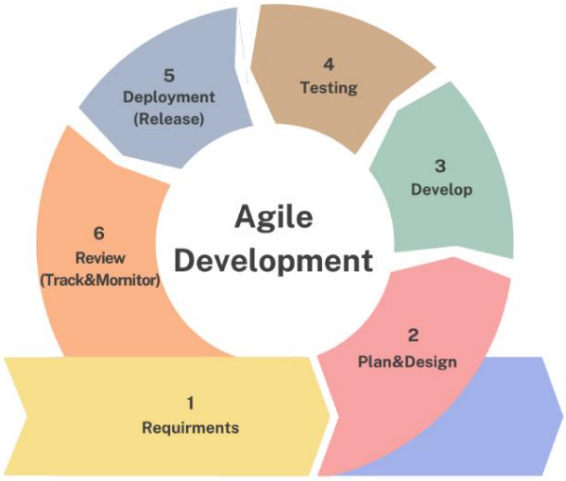
Paijit Suksomboon (2023) ได้กล่าวถึงแนวคิดการพัฒนาระบบซอฟต์แวร์แบบ Agile เป็นแนวคิดพัฒนาระบบสารสนเทศที่มุ่งเน้นการทำงานที่รวดเร็ว สามารถปรับเปลี่ยนแก้ไขการทำงานที่ตรงตามความต้องการของผู้ใช้งานที่มีความยืดหยุ่น เน้นการทำงานเป็นทีมที่มีการกำหนดเป้าหมายทำงานทีละส่วนงานแบบมีส่วนร่วมของผู้พัฒนาระบบกับทีมของผู้ใช้งานระบบ

Al-Saqqa, Sawalha and AbdelNabi (2020) ได้กล่าวถึงการพัฒนาซอฟต์แวร์แบบ Agile เป็นหนึ่งในแนวทางที่ประสบความสำเร็จ การพัฒนาซอฟต์แวร์แบบ Agile เป็นแนวทางแบบน้ำหนักเบาที่ถูกเสนอเพื่อเอาชนะข้อจำกัดของวิธีการพัฒนาแบบบิดเบี้ยว และเพื่อลดค่าใช้จ่ายและต้นทุน ในขณะเดียวกันก็ให้ความยืดหยุ่นในการปรับใช้การเปลี่ยนแปลงข้อกำหนดในทุกขั้นตอน ซึ่งทำได้โดยการจัดการงานและการประสานงานผ่านค่านิยมและหลัก วิธีการแบบ Agile ที่ได้รับความนิยมมากที่สุด

Singh, Kumar and Sagar (2019) ได้กล่าวถึงการพัฒนาซอฟต์แวร์ด้วยวิธีการแบบ Agile มุ่งเน้นไปที่ความพึงพอใจของลูกค้า สนับสนุนกลุ่มที่จัดระเบียบตนเองซึ่งทำงานเพื่อความเป็นเลิศด้านเทคนิคเพื่อเพิ่มความคล่องตัว แต่จำนวนสมาชิกในทีมเป็นปัญหาที่ถูกควบคุมโดยมุมมองของทีม เทคนิคที่ใช้ในการวิจัยนี้มุ่งเน้นไปที่กลุ่ม Agile สามกลุ่มที่มีขนาดต่างกัน ซึ่งทำให้ซอฟต์แวร์ใช้เทคโนโลยีเดียวกัน มีการใช้ทั้งการกระทำที่มีวัตถุประสงค์และเชิงอัตนัย และผลลัพธ์ได้รับการเสริมด้วยการศึกษา ผลลัพธ์แสดงให้เห็นอย่างชัดเจนว่าเพื่อให้ได้ผลลัพธ์ที่ดีขึ้นในการพัฒนาซอฟต์แวร์แบบคล่องตัว การเลือกบุคคลที่เหมาะสมเพื่อพิจารณาว่าเป็นทีมที่ดีเป็นสิ่งสำคัญ

Dudhat and Abbasi (2021) ได้กล่าวถึงการพัฒนาซอฟต์แวร์แบบ Agile แนวทางแบบคล่องตัวเป็นทางเลือกที่เป็นไปได้ในการวางแผนวิธีการขับเคลื่อน (เช่น UML หรือ Waterfall Model) ในการสร้างซอฟต์แวร์คุณภาพสูง นอกจากนี้ยังเหมาะสำหรับโปรเจกต์ขนาดเล็กถึงขนาดกลางด้วย เนื่องจากโดยทั่วไปแล้วนักพัฒนาไม่ได้เน้นไปที่โมเดลซึ่งเป็นองค์ประกอบหลักของโปรแกรมขั้นสุดท้าย ภาพรวมของระเบียบวิธีการพัฒนาซอฟต์แวร์แบบ Agile และความเกี่ยวข้องกับวิศวกรรมซอฟต์แวร์ เทคนิคแบบ Agile มีแนวทางที่แตกต่างกันไปในแต่ละขั้นตอนของวิศวกรรมซอฟต์แวร์ที่มุ่งเน้นไปที่ผลตอบรับและการเปลี่ยนแปลง ด้วยเหตุนี้จึงคิดว่าเทคนิคนี้สามารถเพิ่มประสิทธิภาพการทำงานของโปรแกรมเมอร์ได้

Trivedi (2021) ได้กล่าวถึงการพัฒนาซอฟต์แวร์แบบ Agile ให้ความสำคัญกับการทำงานร่วมกันและทีมที่จัดระเบียบตนเองเป็นอย่างมาก อย่างไรก็ตามนั่นไม่ได้หมายความว่าไม่มีผู้จัดการ แต่ทีมสามารถคิดได้ว่าพวกเขาจะจัดการกับสิ่งต่าง ๆ ได้อย่างอิสระอย่างไร หมายความว่าทีมเหล่านั้นมีการทำงานแบบข้ามสายงาน ทีมเหล่านั้นไม่ได้กำหนดบทบาทเฉพาะที่เกี่ยวข้องมากนัก เมื่อคุณรวมทีมพัฒนาเข้าด้วยกัน คุณจะต้องแน่ใจว่าคุณมีทักษะที่เหมาะสมทั้งหมดในทีม มีข้อกำหนดสำหรับผู้จัดการที่ต้องแน่ใจว่าสมาชิกในทีมมีหรือได้รับชุดทักษะที่เหมาะสม ผู้จัดการยังจัดให้มีสภาพแวดล้อมที่ช่วยให้ทีมประสบความสำเร็จ ตัวอย่างเช่น ผู้จัดการปล่อยให้ทีมของตนทราบวิธีส่งมอบผลิตภัณฑ์และเข้ามาแก้ไขปัญหา เมื่อองค์กรเริ่มทำ Agile Development จะเน้นแนวปฏิบัติที่ช่วยในเรื่องการทำงานร่วมกันและองค์กรได้ดีเยี่ยม อย่างไรก็ตาม แนวปฏิบัติที่สำคัญอีกชุดหนึ่งที่ไม่ได้ปฏิบัติตามบ่อยนักคือแนวปฏิบัติทางเทคนิคเฉพาะที่เกี่ยวข้องกับการพัฒนาซอฟต์แวร์ในลักษณะที่ช่วยให้ทีมจัดการกับความไม่แน่นอน



ภาพที่ 1 Agile Methodology
ที่มา: Trivedi (2021)

1. Requirements การวิเคราะห์ความต้องการของผู้มีส่วนได้ส่วนเสีย เพื่อรวมรวมข้อมูลความต้องการซอฟต์แวร์และการทำงานของระบบ เพื่อระบุข้อกำหนดทางธุรกิจ ทีมงานจำเป็นต้องรวบรวมข้อมูล เช่น ใครเป็นผู้ใช้ซอฟต์แวร์ จะใช้ซอฟต์แวร์อย่างไร ซอฟต์แวร์ที่จำเป็นจะต้องมีอะไรบ้างเป็นอินพุต ซอฟต์แวร์ที่จำเป็นจะให้อะไรเป็นผลลัพธ์สุดท้าย กำหนดเป้าหมายที่บรรลุได้ กรอบเวลา ฟังก์ชันการทำงานของผลิตภัณฑ์ หลังจากนั้นทีมพัฒนาจะดึงแนวทางการแก้ปัญหาโดยการวิเคราะห์ความต้องการ

2. Plan & Design การออกแบบเกี่ยวกับฟังก์ชันการทำงานของซอฟต์แวร์ ส่วนต่อประสานผู้ใช้ และการประมวลผล ภาษาการเขียนโปรแกรม รายละเอียดคุณลักษณะอุปกรณ์ที่ใช้ เทคโนโลยีที่ใช้ ชนิดฐานข้อมูลการออกแบบ ลักษณะของการนำข้อมูลเข้า ลักษณะรูปแบบรายงานที่เกิดและผลลัพธ์ที่ได้

3. Develop การพัฒนาซอฟต์แวร์ให้เป็นระบบตามคุณลักษณะที่กำหนดไว้ เป็นขั้นตอนการการเขียนโปรแกรม (Coding) เพื่อพัฒนาซอฟต์แวร์ให้เป็นระบบตามคุณลักษณะที่กำหนดไว้ จากนั้นทำการทดสอบหาข้อผิดพลาด (Testing) เพื่อตรวจสอบความถูกต้อง

4. Testing การทดสอบทีมพัฒนาทำการทดสอบด้วยตนเอง เป็นการตรวจสอบข้อบกพร่องของซอฟต์แวร์ เพื่อหาข้อผิดพลาดและตรวจสอบว่าตรงตามความต้องการของลูกค้าหรือไม่ ขั้นตอนการทดสอบมักจะทำงานคู่ขนานกับขั้นตอนการพัฒนา

5. Deployment หลังจากการทดสอบข้อบกพร่องของซอฟต์แวร์เรียบร้อยแล้ว ต้องมีการพัฒนาปรับปรุงซอฟต์แวร์ จากนั้นจัดส่งมอบให้กับลูกค้าเพื่อใช้งาน เมื่อลูกค้าเริ่มใช้งานซอฟต์แวร์แล้ว อาจจะเจอปัญหาหรือข้อผิดพลาดเกิดขึ้น ทีมพัฒนาจะต้องดำเนินการปรับแก้ไขซอฟต์แวร์ให้เหมาะสมกับการใช้งานจริงของลูกค้าตามขอบเขตงาน

6. Review เมื่อส่งมอบซอฟต์แวร์ให้กับลูกค้าเรียบร้อยแล้ว ทีมพัฒนาจะต้องมีการบริการหลังการส่งมอบ เพื่อรับฟังความคิดเห็นของผู้ใช้ และรายงานผลลัพธ์ของซอฟต์แวร์ให้กับลูกค้า

การสอนการพัฒนาซอฟต์แวร์ตามแนวคิดแบบ Agile

บทบาทของผู้สอนตามแนวคิดแบบ Agile

1. เป็นผู้นำทางการเรียนรู้ โดยให้คำแนะนำสนับสนุนการเรียนรู้ สร้างสภาพแวดล้อมที่ส่งเสริมการทำงานร่วมกัน และเป็นตัวกลางในการสื่อสารระหว่างผู้เรียน

2. การวางแผนการเรียนที่มีความยืดหยุ่น มีการออกแบบและวางแผนการเรียนรู้ที่สามารถปรับเปลี่ยนได้ตามความต้องการและความก้าวหน้าของผู้เรียน

3. การทำงานร่วมกันเป็นทีม ทำการส่งเสริมการทำงานเป็นทีมและทำงานร่วมกันระหว่างผู้เรียนให้มีการเรียนรู้ร่วมกัน อำนวยความสะดวกในกิจกรรมกลุ่มเพื่อให้ผู้เรียนได้เรียนรู้ซึ่งกันและกัน

4. การส่งเสริมการเรียนรู้ด้วยตนเอง ส่งเสริมให้ผู้เรียนมีทักษะกระบวนการเรียนรู้ สามารถกำหนดเป้าหมายการเรียนรู้ ตัดสินใจเลือก มีทักษะการเรียนรู้

5. การสร้างบรรยากาศที่เป็นกันเอง มีการสร้างบรรยากาศที่สนับสนุนการทำงานเป็นกันเองและส่งเสริมให้ทีมมีความรับผิดชอบต่อผลงานและการทำงานของตนเอง

6. การให้คำติชมและกำลังใจ ให้คำแนะนำและติชมเชิงสร้างสรรค์ที่ช่วยให้ผู้เรียนปรับปรุงตนเองเพื่อให้ผู้เรียนมีส่วนร่วมและมีแรงบัลดาลใจในการเรียนรู้

บทบาทของผู้เรียนแนวคิดแบบ Agile

1. การเรียนรู้ร่วมกับผู้อื่น การทำงานร่วมกันและการแบ่งปันความรู้ภายในทีม ผู้เรียนทำงานร่วมกัน แบ่งปันความเชี่ยวชาญ และเรียนรู้จากประสบการณ์ของกันและกัน ในการส่งเสริมสภาพแวดล้อมที่สนับสนุนซึ่งกันในทีมให้สามารถเติบโตร่วมกันได้

2. ความสามารถในการปรับตัว ผู้เรียนต้องสามารถปรับตัวในแนวทางการทำงานได้ ยอมรับการเปลี่ยนแปลงจากข้อมูลใหม่และสถานการณ์ที่เปลี่ยนแปลงไป ความสามารถในการปรับตัวนี้เป็นกุญแจสำคัญในการตอบสนองต่อการเปลี่ยนแปลงข้อกำหนด เทคโนโลยี หรือสภาวะตลาดอย่างมีประสิทธิภาพ

3. การยอมรับการเปลี่ยนแปลง ผู้เรียนควรเตรียมพร้อมที่จะเรียนรู้ทักษะใหม่ ๆ เปิดรับมุมมองที่แตกต่าง และปรับแนวทางตามความต้องการที่เปลี่ยนแปลงของสังคมทางการเรียนรู้

4. การยอมรับกระบวนการทำซ้ำ Agile ส่งเสริมแนวทางการพัฒนาแบบวนซ้ำ โดยงานจะแบ่งออกเป็นส่วนเล็กๆ ที่สามารถจัดการได้ ผู้เรียนยอมรับกระบวนการทำซ้ำ โดยใช้การทำซ้ำแต่ละครั้งเป็นโอกาสในการเรียนรู้ รวบรวมคำติชม และทำการปรับเปลี่ยนที่จำเป็นเพื่อส่งซอฟต์แวร์ที่มีคุณภาพสูง

5. การทำงานร่วมกันข้ามสายงาน ผู้เรียนสามารถทำงานร่วมกับทีมข้ามสายงาน ส่งเสริมวัฒนธรรมการแบ่งปันความรู้และการเรียนรู้ร่วมกัน พยายามทำความเข้าใจมุมมองที่แตกต่างกันและใช้ประโยชน์จากทักษะและความเชี่ยวชาญที่หลากหลายของสมาชิกในทีมเพื่อให้บรรลุเป้าหมายของโครงการ

การออกแบบกิจกรรมการสอนการพัฒนาซอฟต์แวร์ตามแนวคิดแบบ Agile

ตารางที่ 1 กิจกรรมการสอนการพัฒนาซอฟต์แวร์ตามแนวคิดแบบ Agile

ระเบียนวิธีแบบ Agile	การเรียนการสอนตามแนวคิดแบบ Agile	กิจกรรม
1. ความต้องการ (Requirements)	**ขั้นตอนที่ 1** เป็นขั้นเริ่มต้นของการพัฒนาซอฟต์แวร์ ผู้เรียนต้องทำการศึกษาข้อมูลความต้องการในการพัฒนาซอฟต์แวร์ของลูกค้าหรือผู้ที่เกี่ยวข้อง เช่น ปัญหาการทำงาน	สัมภาษณ์ผู้ใช้งาน และผู้ที่เกี่ยวข้อง ศึกษาข้อมูลจากเอกสาร ศึกษาข้อมูลจากซอฟต์แวร์เดิม โดยการบันทึกข้อมูล ทำการประชุมเพื่อหาข้อสรุปของข้อมูล

ระเบียนวิธีแบบ Agile	การเรียนการสอนตามแนวคิดแบบ Agile	กิจกรรม
	งานปัจจุบัน ปัญหาของการทำงานในระบบเดิม ความต้องการของซอฟต์แวร์ใหม่	
	ขั้นตอนที่ 2 รวมรวมข้อมูลความต้องการซอฟต์แวร์และการทำงานซอฟต์แวร์ เพื่อทำการวิเคราะห์ข้อมูล ใครเป็นผู้ใช้ซอฟต์แวร์ จะใช้ซอฟต์แวร์อย่างไร ซอฟต์แวร์ที่จำเป็นจะต้องมีอะไรบ้างเป็นอินพุต ซอฟต์แวร์ที่จำเป็นจะให้อะไรเป็นผลลัพธ์สุดท้าย กำหนดเป้าหมายที่บรรลุได้ กรอบเวลา ฟังก์ชัน การทำงานของผลิตภัณฑ์	ทีมงานทำการประชุมวิเคราะห์ประเด็นความต้องการในการพัฒนาซอฟต์แวร์
	ขั้นตอนที่ 3 ทีมพัฒนาจะดึงแนวทางการแก้ปัญหาโดยการวิเคราะห์ความต้องการของผู้ใช้งาน	แนวทางในแก้ปัญหา การพัฒนาซอฟต์แวร์
2. การวางแผนและออกแบบ (Plan & Design)	**ขั้นตอนที่ 1** นำข้อมูลที่ได้จากการวิเคราะห์ความต้องการมาทำการวางแผน กำหนดระยะเวลาในการพัฒนาซอฟต์แวร์ตามกรอบเวลา ซึ่งระยะเวลาดำเนินงานจะขึ้นอยู่กับขนาดของซอฟต์แวร์	ทีมพัฒนาซอฟต์แวร์ช่วยกันกำหนดระยะเวลาในการดำเนินงานให้อยู่ในกรอบระยะเวลาที่ผู้ใช้งานต้องการ
	ขั้นตอนที่ 2 ทำการออกแบบการทำงานของซอฟต์แวร์ เป็นการฟังก์ชันการทำงานของซอฟต์แวร์ ส่วนต่อประสานผู้ใช้ และการประมวลผล ภาษาการเขียนโปรแกรม	ทีมพัฒนาซอฟต์แวร์ทำการออกแบบฟังก์ชันการทำงานแต่ละฟังก์ชัน ออกแบบ UX, UI และกำหนดภาษาที่ใช้ในการพัฒนาซอฟแวร์ เช่น ภาษา PHP, JAVA, CSS, Python เป็นต้น
	ขั้นตอนที่ 3 กำหนดรายละเอียดคุณลักษณะอุปกรณ์ที่ใช้ เทคโนโลยีที่ใช้ ชนิดฐานข้อมูลการออกแบบ ลักษณะของการนำข้อมูลเข้า ลักษณะรูปแบบรายงานและผลลัพธ์ที่ได้	- กำหนดเทคโนโลยีที่ใช้ในการพัฒนาซอฟแวร์ ฐานข้อมูล เช่น MySQL, Oracle, FoxPro, Clipper เป็นต้น - กำหนดลักษณะการนำเข้าข้อมูลจากผู้ใช้ ให้มีความสะดวก และไม่ซับซ้อน - กำหนดลักษณะรูปแบบรายงานและผลลัพธ์ที่ได้ ให้แสดงผลในรูปแบบรายงาน กราฟ ตาราง
3. การพัฒนา (Develop)	**ขั้นตอนที่ 1** การเขียนโปรแกรม (Coding) ซึ่งจะต้องแบ่งซอฟต์แวร์ออกเป็นส่วนย่อยๆ การแบ่งผลิตภัณฑ์ออกเป็นชิ้นเล็ก ๆ และการส่งมอบงานบ่อย ๆ ซึ่งใช้เวลาในการพัฒนาในแต่ละส่วนอยู่ที่ 2-4 สัปดาห์ เป็นการทำงานในลักษณะของ Sprint	พัฒนาซอฟต์แวร์ตามที่ได้ออกแบบไว้ และมีการประชุมการดำเนินงานกับทีมในทุก ๆ 2-4 สัปดาห์
	ขั้นตอนที่ 2 การส่งมอบงานในส่วนที่พัฒนาเสร็จแล้วให้กับผู้ใช้ เป็นระยะ จะทำให้ได้ซอฟต์แวร์ที่ตรงกับความต้องการของลูกค้ามากที่สุด	ส่งมอบซอฟต์แวร์เพื่อตรวจสอบความถูกต้อง
	ขั้นตอนที่ 3 การประชุมทีมพัฒนาซอฟต์แวร์ เพื่อดูความคืบหน้าของงาน การส่งมอบงานบ่อย ๆ เป็นวิธีการวัดความคืบหน้าที่ดี	สรุปการพัฒนาซอฟต์แวร์ในแต่ละครั้ง ตรวจตวามคืบหน้าของงาน เก็บข้อมูลการประชุม เพื่อนำไปพัฒนาการทำงานของทีม
4. การทดสอบ (Testing)	**ขั้นตอนที่ 1** การทดสอบการทำงานของซอฟต์แวร์ด้วยผู้พัฒนา การตรวจสอบข้อบกพร่องของซอฟต์แวร์ เพื่อหาข้อผิดพลาดและตรวจสอบว่าตรงตามความต้องการของลูกค้าหรือไม่	นำซอฟต์แวร์ไปทดสอบ พูดคุยกับทีมเพื่อตรวจสอบข้อบกพร่องและข้อผิดพลาดของซอฟต์แวร์

ระเบียนวิธีแบบ Agile	การเรียนการสอนตามแนวคิดแบบ Agile	กิจกรรม
	ขั้นตอนที่ 2 การประชุมทีมพัฒนา เพื่อวิเคราะห์ข้อสรุปเกี่ยวกับผลการทำงานของซอฟต์แวร์หลังการทดสอบ	กำหนดวิธีการ แนวทางการแก้ไขข้อผิดพลาดของซอฟต์แวร์
5. การพัฒนาปรับปรุง (Deployment)	**ขั้นตอนที่ 1** หลังจากการทดสอบข้อบกพร่องของซอฟต์แวร์เรียบร้อยแล้ว ต้องมีการพัฒนาปรับปรุงซอฟต์แวร์	นำข้อผิดพลาดมาปรับปรุง
	ขั้นตอนที่ 2 จัดส่งมอบให้กับลูกค้าเพื่อใช้งาน เมื่อลูกค้าเริ่มใช้งานซอฟต์แวร์แล้ว อาจจะเจอปัญหาหรือข้อผิดพลาดเกิดขึ้น ทีมพัฒนาจะต้องดำเนินการปรับแก้ไขซอฟต์แวร์ให้เหมาะสมกับการใช้งานจริงของผู้ใช้งานตามขอบเขตงาน	นำซอฟต์แวร์ไปใช้งานจริง
6. ความคิดเห็น (Review)	หลังจากส่งมอบซอฟต์แวร์ให้กับผู้ใช้งานเรียบร้อยแล้ว ทีมพัฒนาจะต้องมีการบริการหลังการส่งมอบ เพื่อรับฟังความคิดเห็นของผู้ใช้ และรายงานผลลัพธ์ของซอฟต์แวร์ให้กับผู้ใช้งาน	รวบรวมข้อเสนอแนะ ข้อคิดเห็นจากผู้ใช้งานมาวิเคราะห์ สรุป ปรับปรุงและทำการแนะนำข้อมูลให้กับผู้ใช้งานต่อไป
7. การประเมินผล (Evaluation)	เมื่อมีการพัฒนาซอฟต์แวร์เรียบร้อยแล้ว ผู้เรียนต้องทำการนำเสนอซอฟต์แวร์ที่ทำการพัฒนาขึ้น เพื่อให้ผู้สอนทำการประเมินผลการเรียนรู้	นำเสนอซอฟต์แวร์ที่พัฒนาขึ้น

สรุป

แนวทางการสอนการพัฒนาซอฟต์แวร์ตามแนวคิดแบบ Agile เป็นแนวคิดที่เน้นไปยังซอฟต์แวร์ที่พัฒนาขึ้นอย่างรวดเร็ว เพื่อรองรับการเปลี่ยนแปลงที่เกิดขึ้นตลอดเวลา มีความยืดหยุ่น สามารถปรับได้ตามความต้องการของลูกค้า ซึ่ง Agile ให้ความสำคัญกับคนและการปฏิสัมพันธ์กับลูกค้า มีการกำหนดการทำงานร่วมกันกับลูกค้าสำคัญเป็นสำคัญ และการตอบสนองที่จะเปลี่ยนแปลงไปตามแผน ในส่วนของการนำแนวคิดแบบ Agile มีปรับใช้กับการจัดการเรียนรู้ในยุคปัจจุบันจำเป็นต้องมีการปรับวิธีการในการจัดการเรียนรู้ การสร้างโอกาสทางการเรียนรู้ เพื่อรองรับสภาพแวดล้อมที่มีการเปลี่ยนแปลงอย่างรวดเร็ว ผู้สอนต้องเป็นผู้นำทางการเรียนรู้ โดยให้คำแนะนำสนับสนุนการเรียนรู้ สร้างสภาพแวดล้อมที่ส่งเสริมการทำงานร่วมกัน และเป็นตัวกลางในการสื่อสารระหว่างผู้เรียน ในส่วนของผู้เรียนต้องมีการปรับตัวใน การเรียนรู้ร่วมกับผู้อื่น การทำงานร่วมกันและการแบ่งปันความรู้ภายในทีม ผู้เรียนทำงานร่วมกัน แบ่งปันความเชี่ยวชาญ และเรียนรู้จากประสบการณ์ของกันและกัน ในการส่งเสริมสภาพแวดล้อมที่สนับสนุนซึ่งกันในทีมให้สามารถเติบโตร่วมกันได้

บรรณานุกรม

ภาษาไทย

ชนัญชิดา เลิศจะบก และจงกล จันทร์เรือง. (2563). *การประยุกต์ใช้อไจล์เพื่อออกแบบและพัฒนาระบบนำเสนอหนังสือมีชีวิตดิจิทัล กรณีศึกษาสำนักวิทยบริการและเทคโนโลยีสารสนเทศ มหาวิทยาลัยเทคโนโลยีราชมงคลอีสาน.* รายงานสืบเนื่องจากงานประชุมวิชาการระดับชาติและนานาชาติ.

พงษ์พัชรินทร์ พุธวัฒนะ. (2564). การจัดการเรียนรู้เพื่อพัฒนาผู้เรียนในยุคดิจิทัล. *วารสารนวัตกรรมการเรียนรู้และเทคโนโลยี*, *1*(2), 1-11.

พิฑูร ธนบดีกิจ และจีราภา อินธิแสง โธณีม. (2565). *หลักสูตรแนวคิดแบบ Agile (Agile Thinking).* https://www.lifelong.cmu.ac.th/video/assets/files/skills4life/files/4c8d2c1ef63240b3a97637db5ec0079f.pdf

รวิศ หาญอุตสาหะ. (2566). *ปรับองค์กรอย่างไรให้คล่องตัวและทันโลก.* https://www.krungsri.com/th/plearn-plearn/adjust-agile-organization

ภาษาอังกฤษ

Al-Saqqa, S., Sawalha, S., & AbdelNabi, H. (2020). Agile software development: Methodologies and trends. *International Journal of Interactive Mobile Technologies*, *14*(11), 246-270.

Dudhat, A., & Abbasi, M. A. (2021). Discussion of Agile Software Development Methodology and its Relevance to Software Engineering. *ADI Journal on Recent Innovation*, *3*(1), 105-114.

Gheorghe, A. M., Gheorghe, I. D., & Iatan, I. L. (2020). Agile Software Development. *Informatica Economica*, *24*(2), 90-100.

López-Alcarria, A., Olivares-Vicente, A., & Poza-Vilches, F. (2019). A systematic review of The use of agile methodologies in education to foster sustainability competencies. *Sustainability*, *11*(10), 2915.

Phongphet, W., Chumpa, A., & Thumpanit, A. (2023). การประยุกต์ใช้แนวคิดแบบอไจล์เพื่อพัฒนาระบบรับสมัครนักศึกษาออนไลน์ มหาวิทยาลัยนครพนม. *วารสารศิลปศาสตร์ (วังนางเลิ้ง) มหาวิทยาลัยเทคโนโลยีราชมงคลพระนคร*, *3*(1), 49-61.

Pócsová, J., Bednárová, D., Bogdanovská, G., & Mojžišová, A. (2020). Implementation of Agile methodologies in an engineering course. *Education Sciences*, *10*(11), 333.

Saltz, J., & Heckman, R. (2020). Exploring which agile principles students internalize when using a Kanban process methodology. *Journal of Information Systems Education*, *31*(1), 51.

Sharp, J. H., Mitchell, A., & Lang, G. (2020). Agile Teaching and learning in information systems education: an analysis and categorization of literature. *Journal of Information Systems Education*, *31*(4), 269-281.

Singh, R., Kumar, D., & Sagar, B. B. (2019). *Analytical study of agile methodology in information technology sector*. In 2019 4th International Conference on Information Systems and Computer Networks (ISCON), 422-426.

Suksomboon, P. (2023). *Software Engineer*. http://paijit.lpru.ac.th/cgi-bin/S5672601/SoftEng4.pdf

Trivedi, D. (2021). Agile methodologies. *International Journal of Computer Science & Communicat.*

Upplychainguru. (2023). *หลักการของอไจล์ 12 ข้อ.* https://supplychainguru.co.th/articles/soft-skills/12-agile-principles/