

นิพนธ์ต้นฉบับ
(Original article)การศึกษากราฟิกยูสเซอร์อินเตอร์เฟสข้ามแพลตฟอร์มของระบบปฏิบัติการ
โทรศัพท์มือถือA study on graphic user interface (GUI) of
cross-platform operating system for mobile phone

สกล ธีระวรวิญญู*

Sakol Teeravarunyoo*

คณะสถาปัตยกรรมศาสตร์และการออกแบบ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี บางขุนเทียน กรุงเทพฯ 10150

School of Architecture and Design, Bangkhuntien, Bangkok 10150 Thailand (sakol.tee@mail.kmutt.ac.th)

*ผู้พิมพ์หลัก

Received: May 14, 2020/ Revised: May 28, 2020/ Accepted: June 19, 2020

บทคัดย่อ: นักพัฒนาซอฟต์แวร์ลดระยะเวลาในการผลิตซอฟต์แวร์ของระบบปฏิบัติการที่มีความแตกต่างกันด้วยการใช้ซอฟต์แวร์ที่สามารถเขียนครั้งเดียวได้สองระบบปฏิบัติการบนโทรศัพท์มือถือ แต่อย่างไรก็ตามกราฟิกยูสเซอร์อินเตอร์เฟสที่เขียนออกมาไม่สามารถที่จะแปลงให้ออกมาเป็นสองระบบปฏิบัติการได้อย่างแตกต่างกันให้เหมือนกับระบบปฏิบัติการแบบดั้งเดิม จึงเป็นปัญหาในการวิจัยที่ว่าผู้ใช้งานที่ใช้ระบบปฏิบัติการเดียวกันโดยตลอดสามารถใช้งานกราฟิกข้ามแพลตฟอร์มได้หรือไม่เพราะอาจจะเกิดปัญหาด้านความไม่สอดคล้องกันระหว่างการออกแบบแอปพลิเคชันและระบบความคิดของสมอง โดยการศึกษาครั้งนี้จะเน้นเรื่องการทดสอบ 3 ส่วน คือ การทดลองแรกเป็นการนำทางที่ต้องใช้กราฟิกยูสเซอร์อินเตอร์เฟสสำหรับข้อมูลปริมาณมาก โดยเปรียบเทียบระหว่าง Navigation drawer bar ในระบบปฏิบัติการ Android กับ Table ของ iOS การทดลองที่สองคือการนำทางสำหรับข้อมูลปริมาณน้อยระหว่าง Tab bar ที่นิยมใช้กันมากใน iOS เปรียบเทียบกับ Navigation drawer bar ส่วนสุดท้ายจะศึกษาองค์ประกอบด้านกราฟิกที่มีรูปลักษณ์แตกต่างกันโดยสิ้นเชิงเช่นปฏิทินของ Android เปรียบเทียบกับ Date picker ของ iOS โดยการศึกษาครั้งนี้จะทำการทดสอบผู้ใช้ระบบปฏิบัติการเดียวกันโดยตลอด ผลจากการศึกษาพบว่า ไม่มีความแตกต่างอย่างมีนัยสำคัญระหว่าง Navigation drawer bar กับ Table สำหรับการค้นหาข้อมูลปริมาณมาก ในขณะที่มีความแตกต่างอย่างมีนัยสำคัญของข้อมูลปริมาณน้อย ผู้ใช้ต้องการ Tab bar มากกว่า Navigation drawer bar ในด้านของรูปลักษณ์ที่ต่างกันก็ไม่มีปัญหาด้านการทำงานข้ามแพลตฟอร์มเนื่องจากผู้ใช้คุ้นเคยกับการใช้กราฟิกยูสเซอร์อินเตอร์เฟสนั้นในแอปพลิเคชันทั่วไป การศึกษาครั้งนี้ทำให้ทราบว่ากราฟิกยูสเซอร์อินเตอร์เฟสของแอปพลิเคชันข้ามแพลตฟอร์มสามารถพัฒนาต่อไปในอนาคต

ABSTRACT: Software developers reduce the time of development by using the cross platform mobile app. However, the Graphic User Interface (GUI) designed by using the cross-platform app was nearly the same for both Android and iOS platform. This inspires the study of how users can use the GUI across the platform. The objective of the study is to investigate how users develop their mental models to use the cross-platform app. This study covers three experiments. The first experiment is on the navigation menu for a lot of data such as Navigation drawer bar in Android and Table in iOS. Second investigation is the navigation for small amount of data such as Tab bar in iOS comparing with Navigation drawer bar on Android. The last experiment is the GUI that was designed different look and feel, for example, the Android calendar comparing with Date picker in iOS. The result shows that there is no significant difference of effectiveness between Navigation drawer bar and Table in the first test. For the second test, subjects preferred the Tab bar over the Navigation drawer bar when searching small amount of data. For the last experiment, there is no significant difference between Android Calendar and iOS date picker. Subject experienced those GUI from other apps. This study suggests the GUI of cross-platform can be used for further development.

คำสำคัญ: ข้ามแพลตฟอร์ม การนำทาง กราฟิกยูสเซอร์อินเตอร์เฟส ปัจจัยมนุษย์ ระบบปฏิบัติการ

Keywords: Cross-platform, Navigation, GUI, Human Factor, Operation system

1. บทนำ

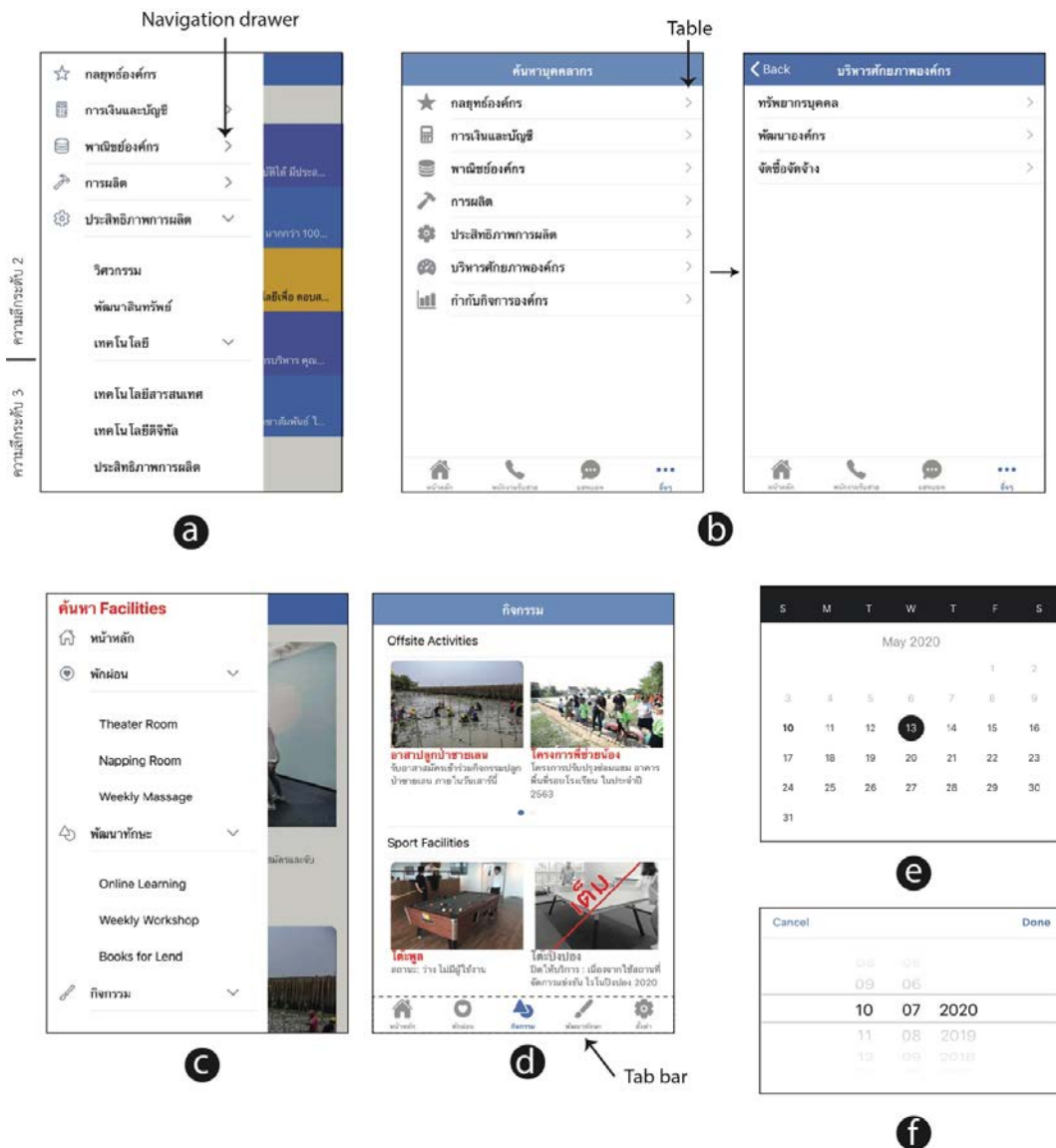
การพัฒนาแอปพลิเคชันโทรศัพท์มือถือมักจะใช้เวลาในการพัฒนาค่อนข้างมาก อีกทั้งยังต้องการทักษะที่หลากหลายจากนักพัฒนาในการเขียนโปรแกรมทั้งระบบปฏิบัติการ iOS และ Android ดังนั้นจึงมีหลายผู้ผลิตซอฟต์แวร์ได้ออกแบบวิธีการเขียนโปรแกรมครั้งเดียวแต่ได้สองระบบปฏิบัติการที่เรียกกันว่า Cross-platform ที่มีการออกแบบตัวเฟรมเวิร์คเช่น Ionic framework, React Native, Titanium และ Flutter ประโยชน์ของการทำ

Cross-platform ก็คือ ค่าใช้จ่ายในการพัฒนาที่มีประสิทธิภาพมากกว่าเพราะไม่ต้องจ้างนักพัฒนาที่มีทักษะการเขียนโปรแกรมทั้งสองระบบปฏิบัติการ และสามารถนำโค้ดที่เขียนกลับมาใช้ใหม่ได้ ลดระยะเวลาในการส่งผลิตภัณฑ์เข้าสู่ตลาด อุปสรรคด้านเทคนิคก็น้อยกว่าเพราะนักพัฒนาแอปพลิเคชันไม่ต้องเรียนรู้ภาษาโปรแกรมมิ่ง เช่น Objective-C ของระบบปฏิบัติการ iOS หรือโปรแกรม Java ของระบบปฏิบัติการ Android [1] ในปัจจุบันองค์กรหลายแห่งในประเทศไทยก็เริ่มการใช้ Cross-platform เพื่อเพิ่มขีดความสามารถในการแข่งขัน

นิยามคำว่า Cross-platform หมายถึงกลุ่มของยูสเซอร์อินเทอร์เฟซสำหรับบริการหนึ่ง ๆ ที่สามารถเล่นได้บนสองแพลตฟอร์มหรือมากกว่า ผู้ใช้สามารถที่จะเข้าถึง Cross-platform ได้ด้วยการใช้ระบบปฏิบัติการบนโทรศัพท์มือถือที่ตนเองใช้อยู่ ปัญหาที่พบส่วนใหญ่ในการทำ Cross-platform ก็คือ ผู้ใช้พบความยากลำบากในการค้นหาคำประกอบของยูสเซอร์อินเทอร์เฟซที่วางไว้อยู่ในตำแหน่งที่แตกต่างจากสิ่งที่ผู้ใช้เคยรู้มา [2] งานวิจัยด้านความสามารถในการใช้งานและประสบการณ์ของผู้ใช้ในเรื่องของ Cross-platform ได้มีรูปแบบที่ทำอยู่สองด้านด้วยกัน คือ ด้านที่เป็น Cross-platform ระหว่างเครื่องมือที่ต่างกัน เช่น ระหว่างคอมพิวเตอร์กับโทรศัพท์มือถือ [3] อีกด้านหนึ่งของการศึกษาเป็นระหว่างระบบปฏิบัติการที่ต่างกันคือ ได้มีการใช้คอมพิวเตอร์ในการวิเคราะห์การร้องเรียนของการใช้แอปพลิเคชันจำนวน 50 ซอฟต์แวร์ จากผลการวิเคราะห์พบว่าผู้ใช้งาน iOS มีการร้องเรียนของ Native แอปพลิเคชันมากกว่า Cross-platform แอปพลิเคชันทั้งในเรื่องการใช้งานยกเว้นประสิทธิภาพของแอปพลิเคชัน [4] เรื่องนี้อาจจะชี้แนะว่า Cross-platform อาจจะมีประสิทธิภาพเหนือ Native แอปพลิเคชันถ้ามีการผลิตที่ดีพอ อีกงานวิจัยคือได้มีการทดสอบ ซอฟต์แวร์ที่ใช้ทำ Cross-platform ที่ชื่อ Titanium mobile development environment ผลปรากฏว่าไม่ค่อยมีความแตกต่างกับ Native version ในระบบปฏิบัติการ Android ในทางตรงข้ามผู้ใช้ต้องการเลือก Native version บนระบบปฏิบัติการ iOS มากกว่า Cross-platform [5] ดังนั้นควรมีการศึกษาถึงสาเหตุปัญหาของการใช้งาน Cross-platform

การใช้ Cross-platform ก็ยังมีปัญหาในด้านการเขียนกราฟิกยูสเซอร์อินเทอร์เฟซ (Graphic User Interface – GUI) ที่สามารถเปลี่ยนไปตามมาตรฐานของ Human Interface Design Guidelines [6] และ Material Design Guideline [7] ได้พร้อมกัน เช่น ถ้าเขียนโปรแกรมให้ปฏิทินมีรูปลักษณะเหมือนตัวหมุน (Spinner) สำหรับการเลือกตัวเลขของ iOS ก็ไม่สามารถเปลี่ยนรูปเป็นปฏิทินตั้งโต๊ะของ Android ได้ ทั้งสองแนวทางการออกแบบของระบบปฏิบัติการมีลักษณะบางอย่างที่คล้ายกันและบางอย่างที่ต่างกันออกไปอย่างสิ้นเชิง จากการเปรียบเทียบทั้งสองระบบ ในทางปฏิบัติถ้าออกแบบได้ดำเนินการออกแบบแอปพลิเคชันตามมาตรฐานการออกแบบที่ผู้ผลิตระบบปฏิบัติการได้ชี้แนะไว้ ก็จะทำให้ผู้ใช้งานเข้าใจง่าย และเกิดคุณภาพของตัวแอปพลิเคชัน ดังจะเห็นได้ว่าผู้บริโภคเมื่อซื้อโทรศัพท์ใหม่ข้ามระบบปฏิบัติการจาก Android มาเป็น iOS จะมีการเปลี่ยนโทรศัพท์กลับไปใช้ระบบปฏิบัติการเดิมหลังจากใช้งานโทรศัพท์ใหม่เพียงไม่กี่สัปดาห์ ทั้งนี้จะเป็นปัญหาเรื่องความคุ้นเคยกับระบบเดิมหรือความยากต่อการใช้งาน

ปัญหาที่พบมากสำหรับผู้ที่ใช้โทรศัพท์มือถือบนระบบปฏิบัติการหนึ่งไม่สามารถใช้กับอีกระบบปฏิบัติการหนึ่งได้ จากการสำรวจที่บริษัทแห่งหนึ่งของผู้ใช้ทั้งหมด 119 คนพบว่าผู้ใช้ที่เป็น iOS อยู่ 39 คน (33%) และคนที่ใช้ Android อยู่ที่ 34 คน (28.5%) ในขณะที่มีคนใช้ทั้งสองระบบอยู่ที่ 46 คน (38%) สาเหตุที่พบว่าคนส่วนใหญ่ไม่สามารถใช้สองระบบได้เพราะเรื่องของ Mental model [8] ซึ่งเป็นแผนภาพในสมองของผู้ใช้คุ้นเคยกับระบบหนึ่งและเมื่อ Conceptual model หรือสิ่งที่นักออกแบบวางแผนระบบไม่ตรงกับแผนภาพในสมองของผู้ใช้จะทำให้เกิดปัญหาของการใช้งาน



รูปที่ 1: เปรียบเทียบระหว่าง GUI ที่ใช้กับ Android และ iOS ตามลำดับเริ่มจาก a) Navigation drawer bar, b) Table เมื่อกดก็จะเปิดหน้าต่างไป c) Navigation drawer bar, d) Tab bar, e) Calendar date picker, f) Date picker

รูปที่ 1 เป็นการเปรียบเทียบตัวอย่างของกราฟิกยูสเซอร์อินเตอร์เฟสระหว่าง Android และ iOS โดยการเปรียบเทียบระหว่าง a) และ b) คือ การนำทางสำหรับข้อมูลปริมาณมาก ทั้งนี้ a) Navigation drawer bar ของ Android มีลักษณะเหมือนแผนภูมิต้นไม้ (Hierarchical navigation) ที่มีลูกศรชี้ลงสามารถขยายเพื่อดูรายละเอียดเพิ่มเติมได้ เมื่อเปรียบเทียบกับ b) Table ของ iOS ที่มีการวิธีเข้าถึงข้อมูลโดยการกดลูกศรชี้ไปทางขวาเพื่อเปลี่ยนหน้าเพจไปที่ละหน้าเมื่อคลิกถึงข้อมูลต่างระดับ ข้อสมมุติฐานนี้คือ Navigation drawer bar แตกต่างจาก Table อย่างไรในเชิงประสิทธิภาพการใช้งาน

ข้อสมมุติฐานที่ 2 เป็นเรื่อง การนำทางสำหรับข้อมูลปริมาณน้อย เราต้องการทราบว่าระหว่าง c) Navigation drawer bar ที่ใช้ใน Android (แถบเมนูด้านซ้ายมือ) สามารถที่จะมาใช้นำทางเทียบกับ d) Tab bar ที่ใช้กันมากใน iOS (แถบเมนูด้านล่างของจอ) ข้อสมมุติฐานในส่วนนี้คือ การนำทางที่แสดงลำดับชั้นของหัวข้อ (Navigation drawer) และการนำทางที่แสดงหัวข้อด้านล่าง (Tab bar) แบบใดจะมีประสิทธิภาพมากกว่ากัน

ข้อสมมติฐานที่ 3 คือ รูปลักษณ์ที่ต่างกันมีผลต่อการใช้งานระหว่างผู้ใช้ iOS และ Android อย่างไร e) Calendar date picker ที่นิยมใช้กันมากใน Android เทียบกับ f) Date picker สำหรับ iOS สามารถแสดงวัน เดือน ปี ซึ่งปฏิทินของ Android เป็นรูปลักษณ์เหมือนปฏิทินตั้งโต๊ะ ในขณะที่รูปลักษณ์ของ iOS จะเหมือนกับวงล้อหมุนของตัวเลข

การศึกษาคั้งนี้มีวัตถุประสงค์อยู่ 3 ประการด้วยกัน คือ

- เพื่อศึกษาประสิทธิภาพในการค้นหาข้อมูลเชิงลึกระหว่าง Navigation drawer bar และ Table
- เพื่อศึกษาประสิทธิภาพและความแตกต่างในการใช้ Navigation drawer bar เมื่อเปรียบเทียบกับ Tab bar ในการหาข้อมูลปริมาณน้อย
- เพื่อศึกษาประสิทธิภาพในการใช้งานรูปลักษณ์กราฟิกยูสเซอร์อินเทอร์เฟซที่แตกต่างกันระหว่าง Calendar date picker และ Picker
- เพื่อศึกษาผลกระทบเมื่อผู้ใช้ระบบปฏิบัติการเดียวมาโดยตลอดที่เปลี่ยนมาใช้กราฟิกยูสเซอร์อินเทอร์เฟซของอีกระบบ

2. วิธีการศึกษา

2.1 กลุ่มตัวอย่าง

ผู้ร่วมวิจัยทั้งหมด 16 คนที่ถูกคัดกรองโดยต้องมีคุณสมบัติคือ มีประสบการณ์การใช้ระบบปฏิบัติการ iOS หรือ Android อย่างใดอย่างหนึ่งมาอย่างน้อย 3-5 ปี ครั้งนี้มีผู้เข้าร่วมที่ใช้ระบบปฏิบัติการ iOS จำนวน 8 ท่านและ Android จำนวน 8 ท่าน โดยแต่ละคนจะใช้เวลาในการทดสอบทั้งสิ้น 40 นาทีต่อบุคคล รวมเวลาการสัมภาษณ์หลังทดสอบ

2.2 เครื่องมือในการทดสอบ

การวิจัยครั้งนี้ได้ใช้ซอฟต์แวร์ในการบันทึกเวลาและความผิดพลาดจากพฤติกรรมที่ชื่อ Boris [9] และใช้โปรแกรมคำนวณทางสถิติ SPSS [10] ในการวิเคราะห์ความแตกต่างอย่างมีนัยสำคัญโดยจะคำนวณด้านเวลาและปริมาณความผิดพลาดที่เกิดขึ้นในเรื่องของการจำแนกความผิดพลาดของมนุษย์ [11] จะถูกวิเคราะห์ผลโดยหลักการของ Human Factors Design Process for Medical device [12]

2.3 งานที่ใช้ในการทดสอบ

การทดสอบจะมีการบันทึกหน้าจอโทรศัพท์มือถือและเสียงโดยจะมีการจำลองแอปพลิเคชันที่ทำงานอยู่บน Browser ที่เขียนด้วย Ionic framework โดยจะใช้ Web browser ในการเปิดแอปพลิเคชันบนโทรศัพท์มือถือทั้งระบบปฏิบัติการ iOS และ Android มีการจับเวลาและความผิดพลาด หลังจากการทดลองให้ผู้ใช้ใส่คะแนนความพึงพอใจตั้งแต่ 1 ถึง 5 โดย 1 คือ ไม่พอใจ ส่วน 5 คือ พอใจมาก ได้มีการสลับการทดลองระหว่างระบบปฏิบัติการทั้งสองเพื่อไม่ให้เกิดความได้เปรียบเสียเปรียบ ตารางที่ 1 เป็นงานที่ใช้ในการทดสอบซึ่งจะมีงานหลัก ๆ อยู่ 3 งาน โดยงานที่ 1 และ 2 เป็นงานเกี่ยวกับการนำทาง ซึ่งงานที่ 1 เป็นการทดสอบการนำทางระดับลึกด้วย Navigation drawer bar และ Table งานนี้จะให้ผู้เข้าร่วมทดสอบค้นหาข้อมูลรายชื่อบุคลากรในองค์กร งานที่สองจะเป็นการนำทางของข้อมูลปริมาณน้อยโดยจะมีงานย่อยจำนวน 3 งาน ผู้เข้าร่วมทดสอบต้องทำการจองห้องกิจกรรมหรือบริการ และเมื่อผู้ทดสอบหาข้อมูลพบให้ตอบว่าห้องหรือกิจกรรมนั้นว่างหรือเต็ม งานที่ 3 เป็นงานทดสอบรูปลักษณ์ของอินเทอร์เฟซที่มีความแตกต่างกันทั้งสองระบบปฏิบัติการว่าจะมีผลต่อการใช้งานอย่างไร

ตารางที่ 1: งานสำหรับ GUI บน Android รุ่น 10 และ iOS รุ่น 13.5

งาน	Android	iOS
1. ค้นหาชื่อคนในองค์กร	Navigation Drawer ฝ่ายประสิทธิภาพการผลิต แผนก เทคโนโลยี หน่วย เทคโนโลยีสารสนเทศ ชื่อ กรรยา มาธิตนันท์ (นามสมมุติ)	Table ฝ่ายบริหารศักยภาพองค์กร และอยู่แผนก ทรัพยากรบุคคล ชื่อคุณปราณี เลหาสุข (นาม สมมุติ)
2. จองห้องกิจกรรมหรือ บริการแล้วเช็คค่าห้อง เต็มหรือว่าง	Navigation Drawer 1. บริการนวดรอบเย็น ที่หมวดพักผ่อน 2. โต๊ะปิงปอง ที่หมวดกิจกรรม 3. หนังสือ Creative Confidence ที่หมวด พัฒนา ทักษะ	Tab Bar 1. ห้องดูหนัง สำหรับ 2 คน ที่หมวดพักผ่อน (เต็ม) 2. Board Game Werewolf ที่หมวดกิจกรรม (ว่าง) 3. Workshop empathize 101 ที่หมวด พัฒนา ทักษะ (ว่าง)
3. ให้ตั้ง วัน เดือน ปี สำหรับการโพสต์ ประกาศ	Calendar date picker งดการใช้ห้อง ประชุมทั้งหมด เนื่องจากจำเป็นต้องพญา ฆ่าเชื้อ ป้องกัน Covid-19 ในช่วงวันที่กำหนด ดังนี้ เริ่มต้น วันที่ 7 June 2020 (เดือน 6) และ สิ้นสุด วันที่ 7 Aug 2020 (เดือน 8) และ ให้ดำเนินการ ลงโพสต์ประกาศให้สำเร็จ	Date Picker

3. ผลการศึกษา

3.1 ประสิทธิภาพ

ตารางที่ 2 แสดงค่าเฉลี่ยและค่าเบี่ยงเบนมาตรฐานในแต่ละงานของผู้ใช้ Android และ iOS โดยใช้ Paired sample t-test จากข้อสมมุติฐานข้อแรกคือเรื่องการนำทางเชิงลึก พบว่าไม่มีความแตกต่างอย่างมีนัยสำคัญ เช่น การค้นหาชื่อคนในองค์กร จะสังเกตได้ว่าเวลาเฉลี่ยของการใช้ Navigation drawer bar ($M = 33.35$ วินาที, $SD = 11.24$) ของ Android ที่เป็นเมนูออกมาจากด้านซ้ายเมื่อเปรียบเทียบกับ Table ของ iOS ($M = 31.24$ วินาที, $SD = 10.83$) ไม่มีความแตกต่างอย่างมีนัยสำคัญ ถ้าสังเกตจะพบว่าเวลาที่เมนูรวมศูนย์เป็นเมนูเดียวของ Navigation drawer bar น่าจะได้เปรียบเพราะไม่มีการเปลี่ยนหน้าจอไปเรื่อย ๆ เหมือนการนำทางแบบ Table ที่ทำให้เสียเวลาแต่ท้ายที่สุดแล้วเวลาหรือความผิดพลาดที่ใช้ไม่แตกต่างกันมากนักสำหรับการหาข้อมูลที่มีปริมาณมาก

ในข้อสมมุติฐานข้อที่สองพบว่ามีความแตกต่างอย่างมีนัยสำคัญของการจองห้องกิจกรรมโดยที่การใช้ Tab menu เป็นเมนูอยู่ด้านล่างที่ใช้กันมากใน iOS ($M = 38.63$ วินาที, $SD = 11.63$) มีอัตราความเร็วเฉลี่ยน้อยกว่าการใช้ Navigation drawer bar ของ Android ($M = 50.11$ วินาที, $SD = 23.11$), $t(15) = -2.15$, $p = 0.049^*$ ซึ่งหมายความว่า ถ้าใช้เมนูที่มาจากด้านล่างในการหาข้อมูลปริมาณน้อยจะได้ประสิทธิภาพมากกว่าในเรื่องของความผิดพลาดพบว่ามีความแตกต่างอย่างมีนัยสำคัญเช่นเดียวกับค่าเฉลี่ยเรื่องเวลา ซึ่งงานจองห้องกิจกรรมมีค่าเฉลี่ยของความผิดพลาด Navigation drawer bar ($M = .75$ วินาที, $SD = .93$) สูงกว่า Tab bar และมีความแตกต่างอย่างมีนัยสำคัญ $t(15) = -3.22$, $p = 0.006^{**}$ ในงานอื่น ๆ เช่น การค้นหาชื่อคนในองค์กรและตั้งวันเดือนปี ไม่มีความแตกต่างอย่างมีนัยสำคัญ ถ้ามาดูเรื่องจำนวนครั้งที่ผิดพลาดของ Navigation drawer bar จะพบว่าทั้งหมดถึง 13 ครั้ง ก็คือความผิดพลาดที่เกิดจาก Mistake เช่น การตีความผิดถึง 11 ครั้ง เนื่องผู้ใช้ตีความผิดจากชื่อหมวด ความผิดพลาดที่เกิดจาก Lapse จำนวน 1 ครั้งเช่น การลืมเป้าหมายที่จะทำ และความผิดพลาดที่เป็น Slip จำนวน 1 ครั้ง คือ การกดผิดพลาด

ในข้อสมมุติฐานข้อที่สามพบว่าไม่มีความแตกต่างอย่างมีนัยสำคัญในเรื่องค่าเฉลี่ยของเวลาที่ใช้ของ Picker ของ iOS ($M=14.44$ วินาที, $SD = 7.64$) และ Calendar date picker ของ Android ($M=15.74$ วินาที, $SD = 7.49$) เช่นเดียวกันกับค่าความผิดพลาด

ตารางที่ 2: เปรียบเทียบเวลาเฉลี่ยของการใช้ Android GUI และ iOS GUI

Tasks	ค่าเฉลี่ยเวลา(วินาที)		ค่านัยสำคัญ (2 tailed)	ค่าเฉลี่ยความผิดพลาด		ค่านัยสำคัญ (2 tailed)
	Android GUI	iOS GUI		Android GUI	iOS GUI	
1. ค้นหาชื่อคนในองค์กร	33.35(11.24)	31.24(10.83)	0.642	0.25(0.58)	0.38(0.72)	0.16
2. จองห้องกิจกรรม	50.11(23.11)	38.63(11.65)	0.049*	.75(.93)	0(0)	0.006**
3. ตั้งวันเดือนปี	15.74(7.49)	14.44(7.64)	0.378	0.71(0.81)	0.35(0.34)	0.18

* ค่าความแตกต่างอย่างมีนัยสำคัญที่ 95% (p value < 0.05) ** ค่าความแตกต่างอย่างมีนัยสำคัญที่ 99% (p value < 0.01)

3.2 การทดสอบผู้ใช้ระบบปฏิบัติการเดียว

การศึกษาค้นคว้าครั้งนี้มีคำถามวิจัยว่าผู้ใช้ที่ใช้ระบบปฏิบัติการเดิมตลอดของแต่ละระบบจะมีปัญหาในการใช้งานหรือไม่ เมื่อพบกราฟิกยูสเซอร์อินเทอร์เฟซที่ข้ามระบบปฏิบัติการมา จากการทดสอบด้วย Independent sample t-test พบว่า ไม่มีความแตกต่างของค่าเฉลี่ยระหว่างผู้ใช้ทั้งสอง ในงานแรกที่เป็นการค้นหาชื่อคนในองค์กรด้วย Navigation drawer bar พบว่าค่าเวลาเฉลี่ยของผู้ใช้ Android ($M = 40.35$ วินาที, $SD = 11.89$) ไม่ต่างจาก iOS มากนัก ($M = 36.90$ วินาที, $SD = 11.95$) เช่นเดียวกับการค้นหาพบว่าผู้ใช้ iOS ($M = 46.31$ วินาที, $SD = 21.35$) มีค่าความเร็วเฉลี่ยมากกว่าผู้ใช้ Android ไม่มากนัก ($M = 53.90$ วินาที, $SD = 26.03$) ในส่วนที่เป็น Tab bar ผู้ใช้ iOS น่าจะได้เปรียบกว่าผู้ใช้ Android แต่คะแนนค่าเฉลี่ยของก็ไม่ได้แตกต่างกันมากนักโดยค่าเฉลี่ยเวลาของผู้ใช้ iOS ($M = 16.38$ วินาที, $SD = 8.81$) น้อยกว่าของผู้ใช้ Android เล็กน้อย ($M = 15.10$ วินาที, $SD = 6.44$) ในการตั้งวันเดือนปี ปฏิทินของผู้ใช้ Android มีค่าเฉลี่ยความเร็ว ($M = 34.48$ วินาที, $SD = 13.10$) เกือบเท่ากับผู้ใช้ iOS ($M = 32.22$ วินาที, $SD = 9.82$) ในการใช้ Date Picker ของ iOS ค่าเฉลี่ยเวลาของผู้ใช้ iOS อยู่ที่ ($M = 30.62$ วินาที, $SD = 13.08$) ซึ่งไม่ต่างมากกับผู้ใช้ Android ($M = 31.86$ วินาที, $SD = 8.92$)

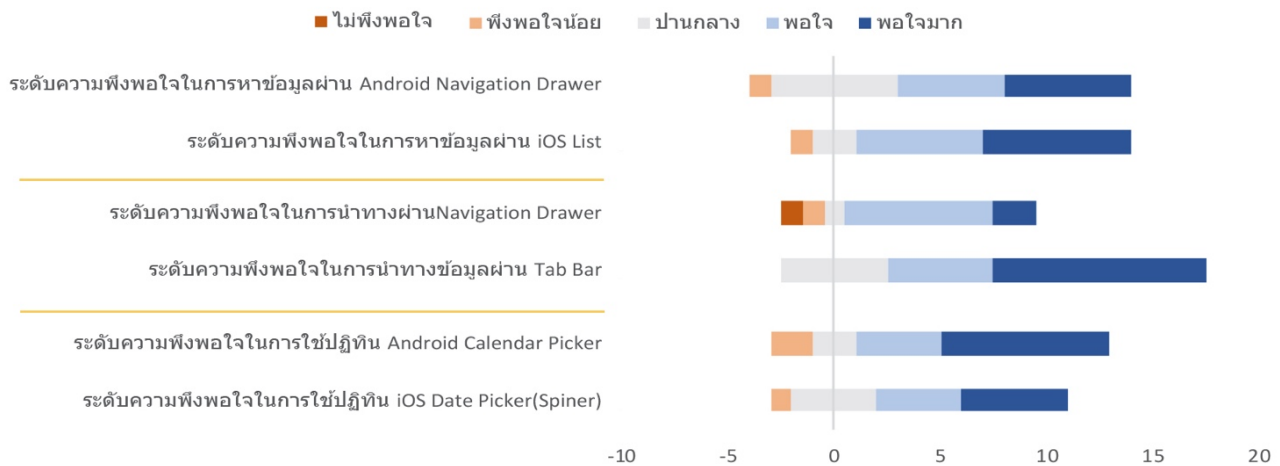
3.3 ความพึงพอใจของผู้ใช้

จากการสัมภาษณ์หลังการทดสอบในด้านความพึงพอใจโดยการให้คะแนนระหว่าง 1 ถึง 5 ในรูปที่ 2 เป็นปริมาณคะแนนความพึงพอใจของ 3 การทดลอง

ในข้อสมมุติฐานที่ 1 ผู้เข้าร่วมทดสอบส่วนใหญ่ให้คะแนนความพึงพอใจที่สอดคล้องกับประสิทธิภาพการใช้งาน กล่าวคือระดับคะแนนความพึงพอใจของงานจองห้องกิจกรรมไม่มีความแตกต่างอย่างมีนัยสำคัญระหว่าง Navigation drawer bar ($M = 4.00$, $SD = 0.97$) และ Table ($M = 4.19$, $SD = 0.91$)

ในข้อสมมุติฐานที่ 2 มีความแตกต่างอย่างมีนัยสำคัญระหว่างค่าเฉลี่ยของ Navigation drawer bar ($M = 3.50$, $SD = 1.03$) ที่น้อยกว่า Tab bar ($M = 4.56$, $SD = 0.63$); $p\text{ value} = 0.08^*$ ผู้ใช้ให้คะแนน Tab bar เนื่องจากความง่ายในการใช้งานและสามารถมองเห็นข้อมูลได้ทันที ไม่จำเป็นกดเข้าไปดูแต่ละหัวข้อเหมือน Navigation Drawer

ในข้อสมมุติฐานที่ 3 ไม่มีความแตกต่างอย่างมีนัยสำคัญระหว่างผู้ใช้ที่เป็น iOS และ Android มากนัก ค่าเฉลี่ยความพึงพอใจขึ้นกับการใช้งานที่ง่ายมากกว่าความคุ้นเคย ถึงแม้ว่าจะมีผู้ใช้บางคนที่ถนัดกราฟิกยูสเซอร์อินเตอร์เฟซที่มีลักษณะเฉพาะ เช่น Date picker ของระบบปฏิบัติการ iOS ค่าเฉลี่ยความพึงพอใจของผู้ใช้ iOS ($M = 3.63$, $SD = 1.06$) ไม่แตกต่างจากค่าเฉลี่ยความพึงพอใจของผู้ใช้ที่เป็น Android ($M = 4.00$, $SD = 0.93$)



รูปที่ 2: ปริมาณคะแนนความพึงพอใจในการทดลองทั้งสามการทดลอง

4. อภิปรายผลการศึกษา

มาตรฐานการออกแบบของ Apple ได้กล่าวถึงการนำทางที่เป็นระดับชั้น Hierarchical navigation [2] และ Google ได้พูดถึง Flat navigation [3] แต่ทั้งสองบริษัทไม่ได้เปรียบเทียบวิธีการที่ใช้งานกับกราฟิกยูสเซอร์อินเตอร์เฟซระหว่างกัน ผลจากการทดลองพบว่าทั้งสองระบบปฏิบัติการสามารถใช้ทดแทนกันได้ ยกตัวอย่างเช่น Navigation drawer bar ของ Android เหมาะสำหรับการหาข้อมูลปริมาณมาก สามารถทดแทนด้วยการใช้ Table ของ iOS ข้อแตกต่างคือ Navigation drawer bar สามารถเห็นข้อมูลทั้งหมดในแถบเดียวกัน ในขณะที่ Table จะมีการจำกัดปริมาณการแสดงผลข้อมูลที่เสนอข้อมูลที่เสนอให้ผู้ใช้ไม่สับสนกับปริมาณข้อมูลที่มาก

ในการเปรียบเทียบการใช้งานระหว่าง Navigation drawer bar และ Tab bar พบว่ามีความแตกต่างกันอย่างมีนัยสำคัญในเชิงประสิทธิภาพ เพราะ Tab bar ของ iOS เหมาะกับการค้นหาข้อมูลที่มีปริมาณน้อย และประเด็นที่สำคัญคือ Visibility ที่ผู้ใช้สามารถเห็นว่าเนื้อหาด้านในของแต่ละหัวข้อมีอะไรบ้างและสามารถหาข้อมูลได้เร็วกว่า Navigation drawer bar ในขณะที่ถ้าใช้ Navigation drawer bar ของ Android จะพบว่าผู้ใช้ไม่สามารถเห็นเนื้อหาในหัวข้อทันทีจนกว่าจะเปิดดูข้างในทำให้เกิดความผิดพลาดในการตีความได้ง่าย (Mistake)

การออกแบบกราฟิกยูสเซอร์อินเตอร์เฟซที่มีลักษณะเฉพาะที่ไม่เหมือนกันทั้งสองระบบปฏิบัติการพบว่าประสิทธิภาพการใช้งานและความพึงพอใจไม่ต่างกันมาก เพราะว่าผู้ใช้มีประสบการณ์ในการเชื่อมโยง Mental model [6] ของประสบการณ์ในการใช้แอปพลิเคชันอื่น ๆ ในชีวิตประจำวัน เช่น ประสบการณ์สั่งซื้อของออนไลน์ทำให้ไม่เกิดปัญหาในการใช้งาน ข้อค้นพบนี้สวนทางกับผลการศึกษาของงานวิจัยที่มีมาก่อนหน้า [5] ที่ผู้ใช้ของ iOS ต้องการยูสเซอร์อินเตอร์เฟซแบบดั้งเดิมมากกว่าผู้ใช้ Android ทั้งนี้เกิดขึ้นเพราะเทคโนโลยีของซอฟต์แวร์การสร้าง Cross-platform สมัยนั้นยังไม่พัฒนาถึงขั้นที่จะเลียนแบบให้เหมือนกับแอปพลิเคชันแบบดั้งเดิมได้

นอกจากนี้กราฟิกยูสเซอร์อินเตอร์เฟซของทั้งสองระบบปฏิบัติการมีข้อดีและข้อเสียแตกต่างกัน เช่น ตัวปฏิทินของ Android สามารถมองเห็นภาพรวมของวันเดือนปีได้ดีกว่า iOS date picker ในทางตรงกันข้าม Date picker ของ iOS สามารถเปลี่ยนระยะเวลาระดับรายปีได้เร็วกว่า ดังนั้นการใช้กราฟิกยูสเซอร์อินเตอร์เฟซที่เป็นเฉพาะทางต้องคำนึงถึงบริบทของการใช้งานเป็นสำคัญ

อีกส่วนที่ผู้ใช้แนะนำและน่าสนใจคือการทำให้เกิดส่วนผสมระหว่างกราฟิกยูสเซอร์อินเตอร์เฟซของทั้งสองระบบยกตัวอย่างเช่น การใช้ Table ของ iOS ผสมกับระดับชั้นของหัวข้อแบบ Navigation Drawer ของ Android ส่วนนี้ทำให้การหาข้อมูลได้ง่ายขึ้นเพราะนำจุดดีของทั้งสองระบบมารวมกัน

คำถามต่อไปคือถ้าประสิทธิภาพในการใช้งานบนแอปพลิเคชันไม่ต่างกันแล้วเหตุผลใดผู้ใช้ไม่ซื้อโทรศัพท์ของระบบปฏิบัติการใด ๆ มาใช้ก็ได้ สาเหตุเนื่องจากการออกแบบที่นอกเหนือจากแอปพลิเคชันที่ไม่ใช่ Cross-platform เช่น การตั้งค่าในระบบปฏิบัติการ ตำแหน่งการวางในที่ต่าง ๆ ของทั้งสองระบบมีวิธีการใช้งานที่แตกต่างกันและไม่ได้อยู่ที่เดียวกันเสมอทำให้ผู้ใช้ไม่สามารถค้นหาสิ่งที่ต้องการพบได้ง่าย

5. สรุปผล

การศึกษครั้งนี้พบว่าผู้ใช้งานที่เป็น iOS หรือ Android ไม่มีความแตกต่างกันในการใช้ Cross-platform แอปพลิเคชันทั้งในด้านประสิทธิภาพและความพึงพอใจ ผู้ใช้งานจำนวนมากแยกไม่ออกว่ากราฟิกยูสเซอร์อินเตอร์เฟซใดเป็นของระบบปฏิบัติการใดเนื่องจากว่าแอปพลิเคชันในปัจจุบันได้ออกแบบโดยการใช้ส่วนประกอบของกราฟิกยูสเซอร์อินเตอร์เฟซทั้งสองระบบทำให้ผู้ใช้มี Mental model ของอีกระบบอยู่ในโทรศัพท์มือถือของตน ความคุ้นเคยนี้ทำให้ผู้ใช้ไม่รู้สึกรู้สถึงการใช้งานที่ยาก ส่วนที่มีความแตกต่างกันมากก็จะเป็นในเรื่องของระบบนำทางที่แต่ละระบบปฏิบัติการนำเสนอการนำทางที่เป็นลักษณะเฉพาะ Hierarchical navigation ของ Android สามารถตอบสนองความต้องการในการหาข้อมูลปริมาณมาก ส่วนการนำทางด้วย Tab bar เหมาะสำหรับการนำทางแบบ Flat navigation ที่มีปริมาณข้อมูลน้อยจะทำให้มีประสิทธิภาพการใช้งานได้ดีกว่า Hierarchical navigation ของ Android นอกจากนี้แล้วการเลือกใช้กราฟิกยูสเซอร์อินเตอร์เฟซที่มีเอกลักษณ์เฉพาะของแต่ละระบบปฏิบัติการจำเป็นต้องคำนึงถึงบริบทของการใช้งาน ดังนั้นการศึกษานี้ช่วยให้เราสามารถแนะนำกราฟิกยูสเซอร์อินเตอร์เฟซที่เหมาะสมสำหรับการใช้งานของ Cross-platform หรือใช้กราฟิกยูสเซอร์อินเตอร์เฟซแบบดั้งเดิมเพื่อเป็นแนวทางให้นักออกแบบได้เลือกใช้ให้เหมาะสมกับการพัฒนาซอฟต์แวร์

เอกสารอ้างอิง

- [1] Manchada A. Where Do Cross-Platform App Frameworks Stand in 2020. [Cited on 11 May 2020]. Available from: <https://www.netsolutions.com/insights/cross-platform-app-frameworks-in-2019/>
- [2] Denis C, Karsenty L. Inter-usability of multi-device systems: A conceptual framework. In: Amhmed S, Homa J, editors. Multiple user Interfaces. New York: Wiley; 2003. P. 373-384.
- [3] Majrashi K, Hamilton M, Uitdenboger AL. Correlating Cross-Platform Usability Problems with Eye Tracking Patterns. In: Proceedings of the 30th International BCS Human Computer Interaction Conference: Fusion! [Internet]. Swindon, GBR: BCS Learning & Development Ltd.; 2016. (HCI '16). Available from: <https://doi.org/10.14236/ewic/HCI2016.40>

- [4] Mercado IT, Munaiah N, Meneely A. The Impact of Cross-Platform Development Approaches for Mobile Applications from the User's Perspective. In: Proceedings of the International Workshop on App Market Analytics 43–49 (Association for Computing Machinery, 2016).
- [5] Angulo E, Ferre X. A Case Study on Cross-Platform Development Frameworks for Mobile Applications and UX. In: Proceedings of the XV International Conference on Human Computer Interaction [Internet]. New York, NY, USA: Association for Computing Machinery; 2014. Available from: <https://doi.org/10.1145/2662253.2662280>
- [6] Apple Inc. Human Interface Guidelines. [Cited on 11 May 2020]. Available from: <https://developer.apple.com/design/human-interface-guidelines/>
- [7] Google. Material. [Cited on 11 May 2020]. Available from: <https://material.io>
- [8] Nersession NJ. In: The Theoretician's Laboratory: Thought Experimenting as Mental Modeling. PSA: Proceedings of the Biennial Meeting of the Philosophy of Science Association. 1992; 2:291-301
- [9] Friard O, Gamba M. BORIS: a free, versatile open-source event-logging software for video/audio coding and live observations. Methods in Ecology and Evolution 2016; Nov;7(11):1325-30
- [10] Pfaff B. PSPP Users' Guide. GNU PSPP Statistical Analysis Software. Release 0.8.5. Boston, MA: Free Software Foundation. 2015. Available from: <http://www.gnu.org/software/pspp/documentation.html>
- [11] Reason J. Human Error. Cambridge, UK: Cambridge University Press, 1990.
- [12] American National Standards Institute. Human Factors Design Process for Medical Devices, ANSI/AAMI HE74:2001.