ระบบยืนยันตัวตนและกำหนดสิทธิ์การใช้งานผ่านเครือข่ายกลาง

Identity Server System

สิรวิชญ์ ฐิติสุนทรลักษณ์ และ อภิสิทธิ์ แสงใส*

สาขาวิชาวิศวกรรมซอฟต์แวร์ คณะวิทยการสารสนเทศ มหาวิทยาลัยบูรพา

Emails: 62160162@go.buu.ac.th, apisit.sa@buu.ac.th*

บทคัดย่อ

การยืนยันตัวตน (Authentication) เป็นวิธีในการระบุหรือทราบตัวตนของผู้เข้าใช้งานระบบ โดยวิธีการยืนยันตัวตนโดยใช้ชื่อผู้ใช้ (Username) และรหัสผ่าน (Password) ถือเป็นวิธีที่ได้รับความนิยมมากที่สุด อย่างไรก็ตาม ผู้วิจัยพบปัญหาผู้ใช้งานต้องมีชื่อผู้ใช้งานและรหัสผ่านหลายชุดในการเข้าใช้งานซอฟต์แวร์ที่แตกต่างดัน ดังนั้น งานวิจัยนี้ จึงพัฒนาระบบยืนยันตัวตนและกำหนดสิทธิ์การใช้งานผ่านเครือข่ายกลาง (Identity Server System: IDS) โดยมีวัตถุประสงค์คือช่วยให้ผู้ใช้งานสามารถเข้าใช้งานซอฟต์แวร์ที่แตกต่างกันโดยใช้เพียงชื่อผู้ใช้งานและรหัสผ่านเพียงชุดเดียว โดยระบบมีมอดูลหลักๆ ได้แก่ มอดูลจัดการผู้ใช้งาน มอดูลจัดการไคลเอนต์ และมอดูลดูบันทึกการใช้งานระบบ โดยระบบดังกล่าว ถูกทดสอบใช้งานจริงโดยผู้เชี่ยวชาญ โดยมีผลความพึงพอใจอยู่ที่ 4.60 (ดีมาก)

คำสำคัญ **–** Authorization, Authentication, OAuth2.0, OpenID Connect, Identity Server

ABSTRACT

Authentication is the process of identifying users who request access to a system. Most access control often determines user identity according to credentials like username and password. However, It is very difficult to have complex & unique usernames and passwords for as many sites as required. Therefore, this paper aims to present the identity server system (IDS). The purpose of this research was to allow users to access various software with only one username and password. IDS has three main modules, i.e. 1) User management 2) Client management, and 3) System logging. IDS has been unit-tested and expert-tested. From the experimental results by using the satisfaction questionnaire. The results showed that the average satisfaction level of this research was 4.60 (Very good).

Keywords -- Authorization, Authentication, OAuth2.0, OpenID Connect, Identity Server

1. บทนำ

ในปัจจุบันเทคโนโลยีสารสนเทศได้เข้ามามีบทบาทสำคัญในการดำเนินชีวิตประจำวัน ซึ่งนอกจากจะช่วยอำนวยความสะดวกในการดำเนินกิจกรรมต่างๆแล้ว ยังช่วยให้ผู้ใช้งานสามารถควบคุมระเบียบการใช้ชีวิตได้อย่างมีประสิทธิภาพมากยิ่งขึ้น ดังนั้น ในมุมมองของความปลอดภัยของข้อมูลของผู้ใช้งานจึงเป็นปัจจัยสำคัญที่จะช่วยส่งเสริมให้ผู้ใช้งานสามารถใช้งานเทคโนโลยีสารสนเทศได้ โดยไม่ต้องคำนึงถึงความปลอดภัยของข้อมูลของตนเองหรือการถูกผู้ไม่ประสงค์ดีทำการเข้าถึงข้อมูล

ของตน ซึ่งช่วยให้ผู้ใช้งานมีความมั่นใจในการเข้าใช้งานเทคโนโลยีสารสนเทศ ได้อย่างสะดวกและปลอดภัยมากยิ่งขึ้น

การพัฒนาระบบยืนยันตัวตนและกำหนดสิทธิ์การใช้งานผ่านเครือข่ายกลาง (Identity Server System) ซึ่งเปรียบเสมือนระบบตัวกลางที่คอยให้บริการซอฟต์แวร์อื่นๆ ในด้านของการยืนยันตัวตนของผู้ใช้งาน (Authentication) และการมอบหมายสิทธิ์ในการเข้าถึงบริการส่วนต่างๆ ของซอฟต์แวร์ให้กับผู้ใช้งาน (Authorization) โดยระบบที่พัฒนาขึ้นนี้จะช่วยส่งเสริมให้การยืนยันตัวตนของผู้ใช้งานมีความสะดวกต่อตัวผู้ใช้งานมากยิ่งขึ้น ซึ่งโครงสร้างของงานวิจัยนี้ ประกอบด้วย บทที่ 1 กล่าวถึงที่มาของงานวิจัย บทที่ 2 กล่าวถึงวัตถุประสงค์ในการจัดทำงานวิจัย บทที่ 3 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง บทที่ 4 การวิเคราะห์และออกแบบระบบ บทที่ 5 ผลการดำเนินงาน และบทที่ 6 สรุปผลการวิจัย

2. วัตถุประสงค์ในการจัดทำงานวิจัย

2.1) เพื่อเพิ่มความปลอดภัยในการเข้าใช้งานซอฟต์แวร์ของผู้ใช้งาน โดยข้อมูลของผู้ใช้งานที่เข้าใช้งานซอฟต์แวร์จะได้รับการยืนยันตัวตนของผู้ใช้งาน (Authentication) และจะถูกกำหนดขอบเขตเพื่อเป็นการป้องกันข้อมูลส่วนตัวของผู้ใช้งานรั่วไหล

2.2) เพื่อให้ผู้ใช้งานที่ต้องการเข้าใช้งานซอฟต์แวร์ สามารถเข้าใช้งานซอฟต์แวร์ที่ได้ลงทะเบียนไว้กับระบบที่พัฒนาขึ้น โดยผู้ใช้งานไม่จำเป็นต้องทำการสมัครสมาชิกที่ซอฟต์แวร์โดยตรง อันเป็นการเพิ่มความสะดวกในการเข้าใช้งานซอฟต์แวร์ให้กับผู้ใช้งาน

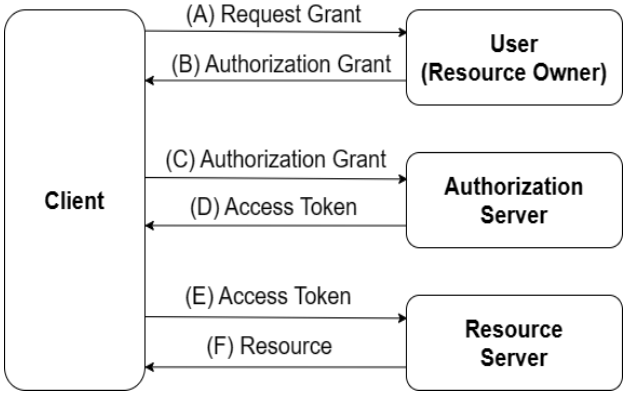
2.3) เพื่อให้ซอฟต์แวร์ที่ได้ลงทะเบียนไว้กับระบบที่พัฒนาขึ้นสามารถบริหารจัดการการยืนยันตัวตนของผู้ใช้งาน (Authentication) และกำหนดสิทธิ์การใช้งานให้กับผู้ใช้งาน (Authorization) ได้อย่างมีประสิทธิภาพ

3. ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

3.1 OAuth 2.0

OAuth 2.0 [1] คือมาตรฐานในการจัดการสิทธิ์การเข้าใช้งานแอปพลิเคชันของผู้ใช้งานในกรณีที่ผู้ใช้งานต้องการเข้าใช้งานแอปพลิเคชัน ซึ่งผู้ใช้งานแต่ละคนอาจมีสิทธิ์ในการเข้าใช้งานแอปพลิเคชัน ทั้งในส่วนที่เหมือนกันและแตกต่างกัน ดังนั้นโปรโตคอล OAuth 2.0 จึงถูกออกแบบมาให้จัดการในส่วนของการจัดการสิทธิ์การใช้งานแอปพลิเคชันให้กับผู้ใช้งานอย่างมีโครงสร้างและมีประสิทธิภาพ โดย OAuth 2.0 มีขั้นตอนการทำงานที่ใช้ Access Token ซึ่งเปรียบเสมือนรหัสที่ใช้ในการเข้าใช้งานแอปพลิเคชัน แทนการที่ผู้ใช้งานจะต้องกรอกชื่อผู้ใช้งานและรหัสผ่านในการเข้าใช้งานแอปพลิเคชันโดยตรง

สถาปัตยกรรมของ OAuth 2.0 มีองค์ประกอบซึ่งประกอบไปด้วยไคลเอนต์ (Client) ผู้ใช้งานหรือเจ้าของทรัพยากร (Resource Owner) และเซิร์ฟเวอร์ OAuth (Authorization Server) และเซิร์ฟเวอร์จัดเก็บทรัพยากร (Resource Server) โดยมีลำดับการทำงาน ดังนี้



ภาพ 1 สถาปัตยกรรมของ OAuth 2.0

3.1.1) ขั้นตอน A ไคลเอนต์ร้องขอสิทธิ์ในการเข้าถึงทรัพยากรที่มีผู้ใช้งานเป็นเจ้าของทรัพยากรนั้น

3.1.2) ขั้นตอน B ผู้ใช้งานอนุมัติให้ไคลเอนต์เข้าถึงทรัพยากรของตน ด้วยวิธีการเลือกยินยอมให้เข้าถึงทรัพยากร (Consent)

3.1.3) ขั้นตอน C ไคลเอนต์ติดต่อไปยังเซิร์ฟเวอร์ OAuth และแสดงหลักฐานการอนุมัติการเข้าถึงทรัพยากร (Authorization Grant)

3.1.4) ขั้นตอน D เซิร์ฟเวอร์ OAuth ตรวจสอบ Authorization Grant และส่ง Access Token กลับไปยังไคลเอนต์

3.1.5) ขั้นตอน E หลังจากที่ไคลเอนต์ได้รับ Access Token จากเซิร์ฟเวอร์ OAuth ไคลเอนต์จะติดต่อไปยังเซิร์ฟเวอร์จัดเก็บทรัพยากร โดยใช้ Access Token ที่ได้รับมา

3.1.6) ขั้นตอน F เซิร์ฟเวอร์จัดเก็บทรัพยากร ตรวจสอบ Access Token และส่งทรัพยากรของผู้ใช้งานกลับไปยังไคลเอนต์

3.2 OpenID Connect

เทคโนโลยี OpenID Connect [2] คือโปรโตคอลที่ช่วยในการตรวจสอบการยืนยันสิทธิ์ ในการเข้าถึงทรัพยากร (Authentication) [3] โดยในขั้นตอนที่ Authorization Server ยืนยันสิทธิ์ในการเข้าถึงทรัพยากรจะเป็นการเข้าสู่การทำงานของ OIDC ซึ่งจะทำหน้าที่ตรวจสอบการยืนยันสิทธิ์ในการเข้าถึงทรัพยากร หากพบว่าการยืนยันสิทธิ์ในการเข้าถึงทรัพยากรนั้นถูกต้องจึงจะสร้าง ID Token แล้วส่งกลับไปยัง Client

3.3 งานวิจัยที่เกี่ยวข้อง

Jeff H. (2000) ได้มองเห็นถึงช่องโหว่ ซึ่งอาจส่งผลให้ข้อมูลของผู้ใช้งานที่ทำการยืนยันตัวตนแล้วในเครือข่ายคอมพิวเตอร์หลุดรอดออกไปได้ ผู้วิจัยจึงได้ทำการออกแบบและกำหนดนโยบายความปลอดภัยของข้อมูลของผู้ใช้งาน ในขณะที่ใช้งานอยู่ในเครือข่ายคอมพิวเตอร์นั้นๆ โดยใช้ Secure Socker Layer (SSL) และ Transport Layer Security (TLS) ซึ่งเป็นโพรโตคอลสำหรับเข้ารหัสข้อมูลของผู้ใช้งานในขณะที่ใช้งานอยู่บนเครือข่ายโดยใช้ Public Key ซึ่งข้อมูลของผู้ใช้งานที่ผ่านการเข้ารหัสแล้วนั้น จะมีเพียงแต่เครื่องเซิร์ฟเวอร์ในเครือข่ายที่มี Public Key ที่ได้รับอนุญาตจากผู้ใช้งานแล้วเท่านั้น จึงจะสามารถถอดรหัสและเข้าถึงข้อมูลของผู้ใช้งานได้ ส่งผลให้ข้อมูลของผู้ใช้งานบนเครือข่ายมีความปลอดภัยมากยิ่งขึ้น [3]

Yu S. และ Zhu l. (2008) ได้ทำการออกแบบวิธีการป้องกันเว็บไซต์ จากการสวมรอยเป็นผู้ใช้งานของเว็บไวต์นั้นๆ โดยผู้โจมตี (Attacker) ในกรณีที่ผู้โจมตีได้ทำการเข้าถึงข้อมูลชื่อผู้ใช้งานและรหัสผ่านของผู้ใช้งานด้วยวิธีการที่ไม่พึงประสงค์ โดยใช้ Trusted Computing (TC) ซึ่งเป็นวิธีการในการตรวจสอบฮาร์ดแวร์หรือข้อมูลเครื่องคอมพิวเตอร์ของผู้ใช้งานว่าเครื่องคอมพิวเตอร์ที่ใช้ในการยืนยันตัวตนของผู้ใช้งานในการเข้าใช้งานเว็บไซต์อยู่ ณ ขณะนี้ เป็นเครื่องคอมพิวเตอร์ที่ใช้ในการยืนยันตัวตนของผู้ใช้งานอยู่เป็นประจำหรือไม่ ซึ่งช่วยให้เว็บไซต์สามารถคัดกรองได้ว่า ผู้ใช้งานที่เข้าใช้งานเว็บไซต์อยู่นั้น เป็นผู้ใช้งานที่เป็นเจ้าของข้อมูลชื่อผู้ใช้งานและรหัสผ่านชุดนั้นจริงหรือไม่ [4]

Namzul H. และคณะ (2018) ได้เสนอวิธีการเพิ่มความปลอดภัยให้กับวิธีการยืนยันตัวตนของผู้ใช้งานโดยใช้โพรโตคอล OAuth ผู้วิจัยและคณะได้มองเห็นว่า Access Token ซึ่งเป็น Token ที่ใช้สำหรับยืนยันตัวตนของผู้ใช้งานในการเข้าใช้งานแอปพลิเคชันต่างๆนั้น มีความปลอดภัยที่ต่ำและมีความเสี่ยงสูงที่ Access Token จะถูกผู้ไม่ประสงค์ดีทำการเข้าถึงและขโมย Access Token ไป ผู้วิจัยและคณะจึงได้มีการเสนอให้มีการเข้ารหัส Access Token โดยใช้เครื่องมือสำหรับเข้ารหัสข้อมูลต่างๆ เพื่อเพิ่มความปลอดภัยในการป้องกันและการเข้าถึง Access Token ของผู้ใช้งาน [5]

3.4 ภาษาที่ใช้การพัฒนาระบบ

ภาษาที่ใช้ในการพัฒนาระบบเป็นองค์ประกอบสำคัญที่จะช่วยให้ระบบสามารถทำงานได้ตรงตามวัตถุประสงค์ ซึ่งภาษาที่ใช้ในการพัฒนาระบบ ประกอบไปด้วย C# (C Sharp), TypeScript [6], HTML, CSS, JavaScript และ SQL [7] รวมถึงเฟรมเวิร์กที่ใช้ในการพัฒนาระบบ โดยระบบฝั่ง Front-End Framework คือ Angular [8] และเฟรมเวิร์กที่ใช้ในการพัฒนาระบบ โดยระบบฝั่ง Back-End Framework คือ ASP.NET Core [9]

3.5 เครื่องมือที่ใช้ในการพัฒนาระบบ

การใช้เครื่องมือและเทคโนโลยีต่างๆ เข้ามาช่วยในการพัฒนาระบบ เพื่ออำนวยคสามสะดวกในการพัฒนาระบบ โดยเครื่องมือที่ใช้ ประกอบไปด้วย Visual Studio Code, Visual Studio 2022, GitLab, Google Chrome และ Microsoft SQL Server Management Studio 18 [10]

4. การวิเคราะห์และออกแบบระบบ

ระบบยืนยันตัวตนและกำหนดสิทธิ์การใช้งานผ่านเครือข่ายตัวกลาง (Identity Server System) เริ่มต้นจากการออกแบบโครงสร้างและลำดับการทำงานของระบบ จนถึงขั้นตอนของการพัฒนาระบบตามที่ได้ออกแบบไว้ในเอกสารการออกแบบ

4.1 ขั้นตอนการวางแผนการดำเนินงานวิจัย

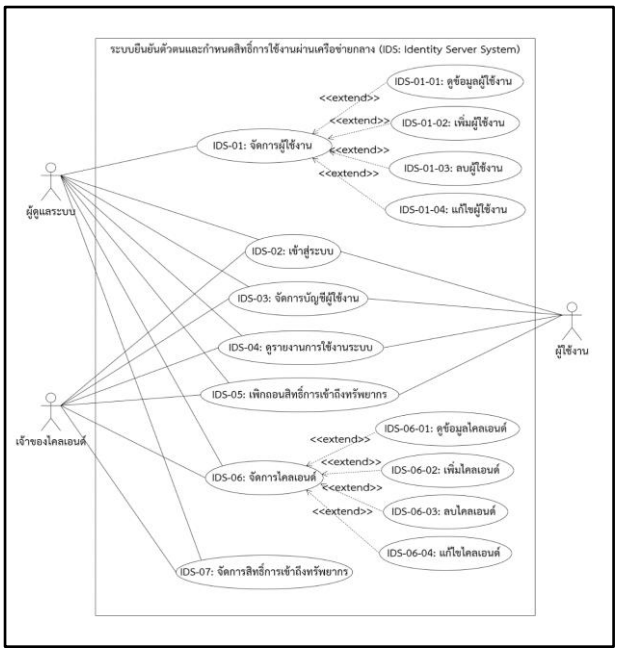
ขั้นตอนของการวางแผนการดำเนินงานวิจัย ได้มีการวางแผนการทำงาน โดยแบ่งหัวข้อการทำงานออกเป็นหัวข้อดังนี้

1) เก็บรวบรวมความต้องการ
2) วิเคราะห์ความต้องการ
3) วิเคราะห์และออกแบบกระบวนการทำงานของระบบ
4) ศึกษาเครื่องมือและวิธีการใช้งานเครื่องมือ
5) พัฒนาระบบยันยันตัวตนและกำหนดสิทธิ์การใช้งาน
6) ทดสอบกระบวนการทำงานของระบบ
7) ส่งมอบระบบ

4.2 วิเคราะห์กระบวนการทำงานของระบบ

การวิเคราะห์และออกแบบการทำงานของระบบ ผู้จัดทำได้ทำการการออกแบบลำดับการทำงานของมอดูลต่างๆ อันประกอบไปด้วย แผนภาพยูสเคส (Use Case Diagram), แผนภาพกิจกรรม (Activity Diagram) และ แผนภาพคลาส (Class Diagram)

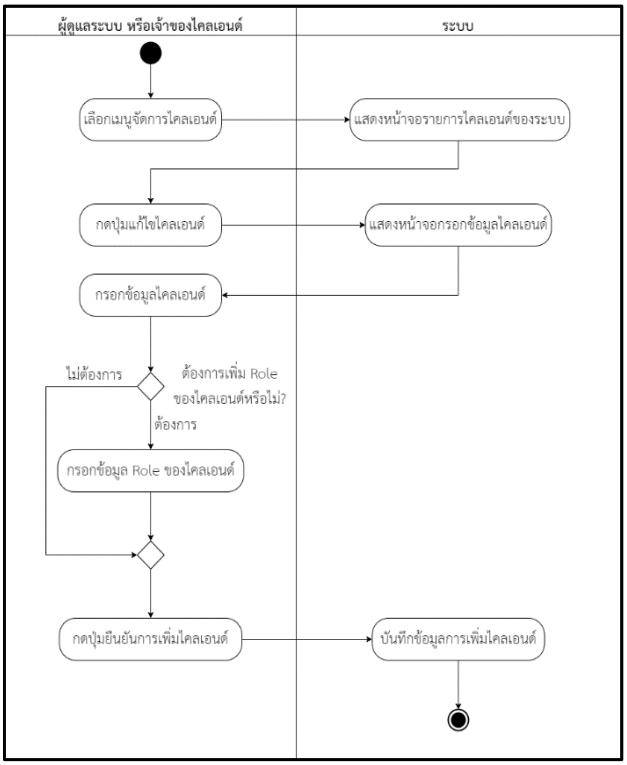
4.2.1) แผนภาพยูสเคส (Use Case Diagram)



ภาพ 2 แผนภาพยูสเคส

ภาพที่ 2 แสดงภาพองค์ประกอบต่างๆในการพัฒนาระบบ โดยทำการจำแนกมอดูลการทำงานของระบบออกเป็น 13 มอดูล และผู้มีส่วนเกี่ยวข้องกับระบบ ทั้งหมด 3 ประเภท ได้แก่ ผู้ดูแลระบบ เจ้าของไคลเอนต์ และผู้ใช้งาน

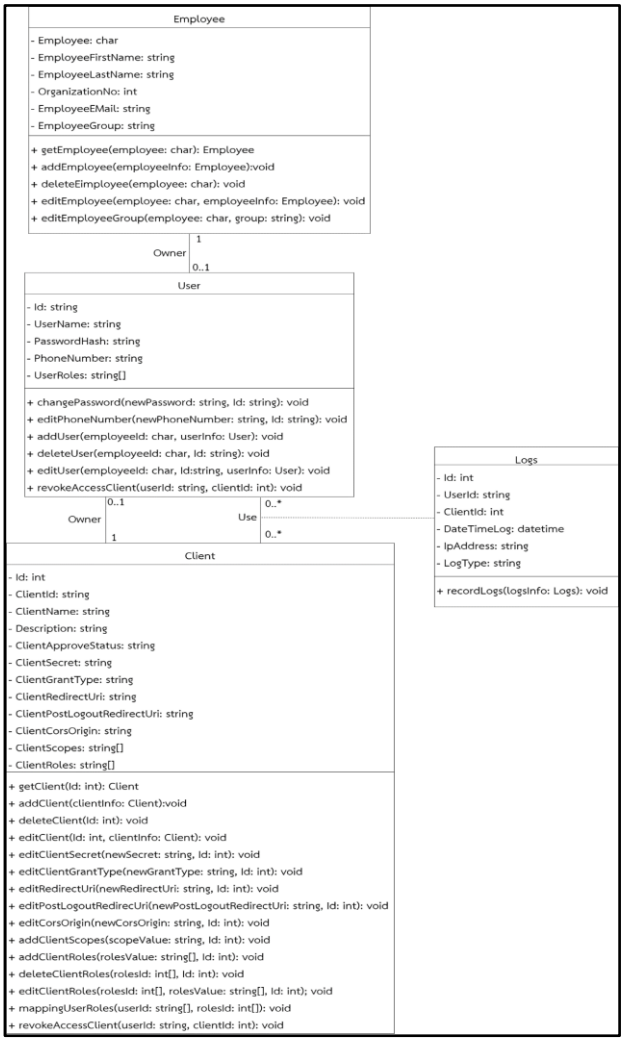
4.2.2) แผนภาพกิจกรรม (Activity Diagram)



ภาพ 3 แผนภาพกิจกรรม

ภาพที่ 3 แผนภาพอธิบายลำดับการทำงานของมอดูลลงทะเบียนไคลเอนต์หรือซอฟต์แวร์ เป็นมอดูลสำหรับเพิ่มข้อมูลของไคลเอนต์หรือซอฟต์แวร์ที่ต้องการเข้ามาใช้บริการของระบบยืนยันตัวตนและกำหนดสิทธิ์การใช้งานผ่านเครือข่ายกลาง โดยมีผู้ดูแลระบบและเจ้าของไคลเอนต์ เป็นผู้กระทำกับมอดูลนี้

4.2.3) แผนภาพคลาส (Class Diagram)



ภาพ 4 แผนภาพคลาส

ภาพที่ 4 แสดงองค์ประกอบของระบบ โดยทำการจำแนกองค์ประกอบหลักของระบบออกเป็นทั้งหมด 4 ส่วน ได้แก่ ส่วนของข้อมูลพนักงาน ส่วนของข้อมูลผู้ใช้งาน ส่วนของไคลเอนต์ และส่วนของบันทึกการทำงานของระบบ จากนั้นจึงนำองค์ประกอบดังกล่าวบันทึกลงบนแผนภาพคลาสเพื่อใช้อธิบายถึงคุณลักษณะ (Attribute) และความสามารถ (Method) ของแต่ละองค์ประกอบ

4.2.4) รหัสเทียมการทำงานในส่วนของ OAuth 2.0 (Pseudo Code)

อัลกอริทึม การยืนยันตัวตนของผู้ใช้งาน (User authentcation) ผ่านทางโพรโทคอล OAuth2.0

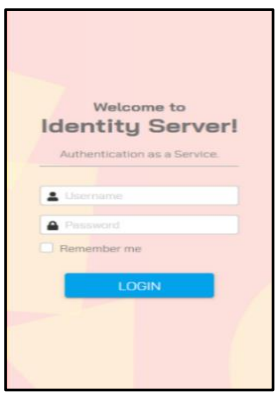
```
username_input = " "
password_input = " "
username_read = " "
password_read = " "
count = 0
user_data = " "
account_count = 0
if user is authenticated then
    read user data
    redirect to application homepage
else
    redirect to Identity Server System (IDS)
    login page
    input username_input
    input password_input
    read account_count
    while count < account_count do
        read username_read[count]
        read password_read[count]
        If username_input = username_read[count]
        and password_input = password_read[count]
        then
            read user data
            redirect to application homepage
        else
            count = count + 1
        end if
    end while
    output message "username or password invalid."
end if
```

รหัสเทียมการทำงานของ OAuth 2.0 ซึ่งแสดงการนำเอาหลักการของ OAuth 2.0 มาใช้ในส่วนของการยืนยันตัวตนของผู้ใช้งาน (Authentication) โดยเริ่มจากการเข้าใช้งานไคลเอนต์หรือแอปพลิเคชันที่ทำการลงทะเบียนเพื่อขอใช้บริการของระบบยืนยันตัวตนและกำหนดสิทธิ์การใช้งานผ่านเครือข่ายกลาง (Identity Server System) ซึ่งเมื่อเข้าใช้งานไคลเอนต์จะทำการตรวจสอบว่าผู้ใช้งานทำการยืนยันตัวตนไว้ก่อนหน้าแล้วหรือไม่ หากผู้ใช้งานยังไม่ทำการยืนยันตัวตน ไคลเอนต์จะทำการเปลี่ยนเส้นทางไประบบยืนยันตัวตนและกำหนดสิทธิ์การใช้งานผ่านเครือข่ายกลาง และแสดงหน้าจอลงชื่อเข้าใช้งาน จากนั้นให้ผู้ใช้งานกรอกชื่อผู้ใช้งาน (Username) และรหัสผ่าน (Password) จากนั้นทำการเข้าสู่ระบบ ระบบจะทำการตรวจสอบชื่อผู้ใช้งานและรหัสผ่าน หากตรวจสอบแล้วพบว่าถูกต้อง ระบบจะทำการอ่านข้อมูลของผู้ใช้งาน และส่งข้อมูลของผู้ใช้กลับไปยังไคลเอนต์หรือแอปพลิเคชันต้นทาง เป็นอันเสร็จสิ้นการทำงานของ OAuth 2.0 ในส่วนของการยืนยันตัวตนของผู้ใช้งาน

5. ผลการดำเนินงาน

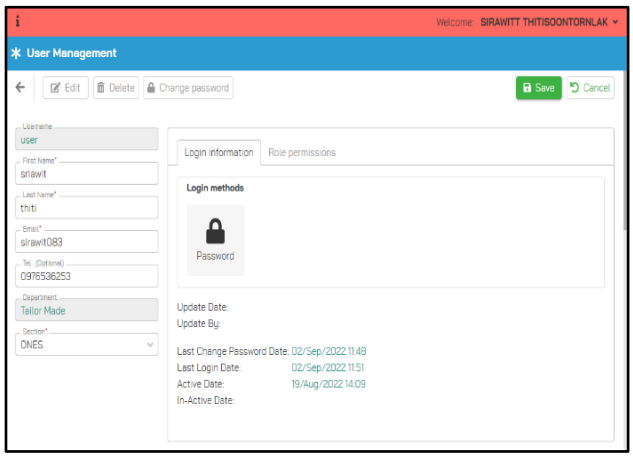
การดำเนินการพัฒนาระบบยืนยันตัวตนและกำหนดสิทธิ์การใช้งานผ่านเครือข่ายกลาง โดยมีผลลัพธ์ของการพัฒนาระบบมีรายละเอียดดังนี้

5.1) มอดูลเข้าสู่ระบบ ผู้ใช้งานซึ่งในที่นี้ประกอบไปด้วย ผู้ดูแลระบบ เจ้าของไคลเอนต์ และผู้ใช้งาน ทำการกรอกชื่อผู้ใช้งานและรหัสผ่าน เพื่อเข้าใช้บริการในส่วนต่างๆ ของระบบ โดยผู้ใช้งานแต่ละประเภทจะมีสิทธิ์การใช้บริการส่วนต่างๆ ของระบบที่แตกต่างกัน



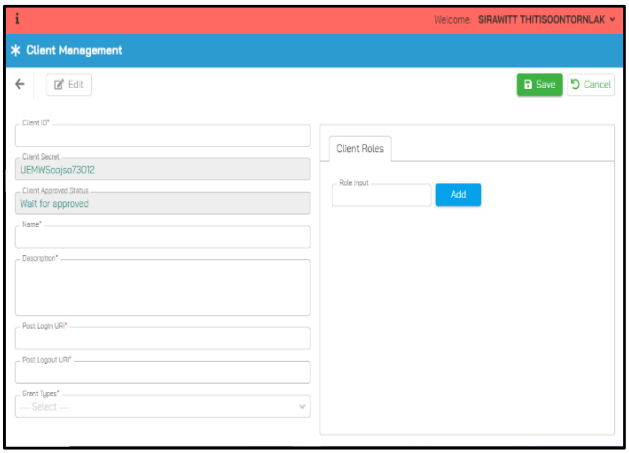
ภาพ 5 หน้าจอมอดูลเข้าสู่ระบบ

5.2) มอดูลจัดการผู้ใช้งาน เป็นมอดูลสำหรับบริหารจัดการผู้ใช้งานประเภทต่างๆ ในระบบ โดยผู้ดูแลระบบ ซึ่งประกอบไปด้วยการดูข้อมูลของผู้ใช้งาน การเพิ่มผู้ใช้งาน การลบผู้ใช้งาน การแก้ไขข้อมูลของผู้ใช้งาน และการจัดการรหัสผ่านของผู้ใช้งานในระบบ



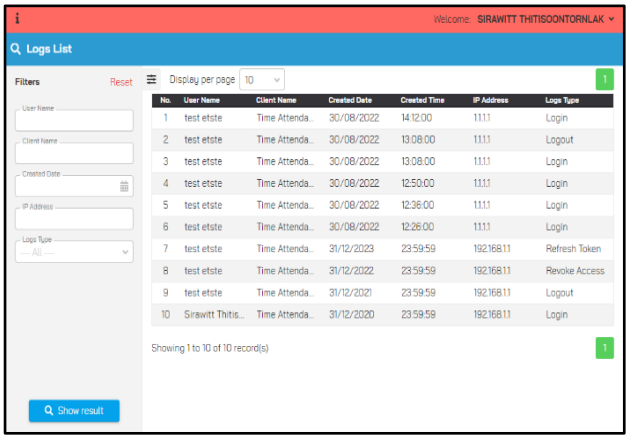
ภาพ 6 หน้าจอมอดูลจัดการผู้ใช้งาน

5.3) มอดูลการจัดการไคลเอนต์ สำหรับบริหารจัดการไคลเอนต์หรือแอปพลิเคชันที่ต้องการใช้บริการ ของระบบโดยผู้ดูแลระบบและผู้เป็นเจ้าของไคลเอนต์ ซึ่งประกอบไปด้วย การดูข้อมูลของไคลเอนต์ การเพิ่มไคลเอนต์ การลบไคลเอนต์ และการแก้ไขข้อมูลของไคลเอนต์



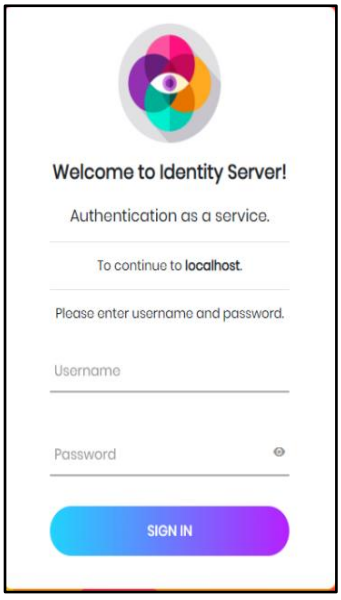
ภาพ 7 หน้าจอมอดูลการจัดการไคลเอนต์

5.4) มอดูลดูรายงานการใช้งานระบบ มอดูลดูรายงานการใช้งานระบบเป็นมอดูลสำหรับการดูรายงานการดำเนินการกับระบบในส่วนต่างๆ ประกอบไปด้วย วันที่และเวลาที่ผู้ใช้งานทำการเข้าใช้งานและการออกจากระบบของไคลเอนต์หรือซอฟต์แวร์ รวมถึงวันที่และเวลาที่ผู้ใช้งานทำการเพิกถอนการเข้าถึงทรัพยากร



ภาพ 8 หน้าจอมอดูลดูรายงานการใช้งานระบบ

5.5) มอดูลเข้าสู่ระบบในฝั่งของผู้ใช้งานที่ต้องการเข้าใช้งานไคลเอนต์หรือซอฟต์แวร์ โดยเป็นการเข้าใช้งานไคลเอนต์หรือซอฟต์แวร์ โดยใช้ระบบระบบยืนยันตัวตนและกำหนดสิทธิ์การใช้งานเป็นทางผ่านในการยืนยันตัวตนของผู้ใช้งาน



ภาพ 9 หน้าจอมอดูลเข้าสู่ระบบของผู้ใช้งานที่ต้องการเข้าใช้งานไคลเอนต์หรือซอฟต์แวร์

โดยหลังจากเสร็จสิ้นกระบวนการพัฒนาแล้ว ระบบได้รับการทดสอบการใช้งานจากผู้เชี่ยวชาญ ไม่ว่าจะเป็นในด้านของความสะดวกสบายในการเข้าใช้งานซอฟต์แวร์ ประสิทธิภาพการทำงานของระบบ ความพร้อมใช้งานของระบบ ความเร็วในการตอบสนองของระบบ และความน่าใช้งานของระบบ ซึ่งได้ผลลัพธ์ในการประเมินความพึงพอใจในการใช้งานระบบโดยผู้เชี่ยวชาญมีผลคะแนนการประเมินอยู่ที่ 4.60 (ดีมาก)

ตาราง 1 แบบประเมินความพึงพอใจในการใช้งานระบบ

หัวข้อประเมินความพึงพอใจในการใช้งานระบบ	ระดับความพึงพอใจ				
	5	4	3	2	1
1. ความสะดวกสบายในการเข้าใช้งานซอฟต์แวร์	✓				
2. ประสิทธิภาพการทำงานของระบบ		✓			
3. ความพร้อมใช้งานของระบบ	✓				
4. ความเร็วในการตอบสนองของระบบ		✓			
5.ความน่าใช้งานของระบบ	✓				

6. สรุปผลการวิจัย

ในการพัฒนาระบบยืนยันตัวตนและกำหนดสิทธิ์การใช้งานผ่านเครือข่ายกลาง ได้เริ่มจากการมองเห็นปัญหาที่ผู้ใช้งานจำเป็นต้องมีชื่อผู้ใช้งานและรหัสผ่านหลายชุดในการเข้าใช้งานซอฟต์แวร์ที่แตกต่างกัน ดังนั้น การดำเนินงานวิจัยฉบับนี้จึงมีวัตถุประสงค์เพื่อแก้ไขปัญหาที่ได้กล่าวถึงนี้ โดยได้เริ่มพัฒนาระบบจากมอดูลจัดการผู้ใช้งาน ซึ่งเป็นมอดูลในการสร้างบัญชีผู้ใช้ให้กับผู้ใช้งานที่ต้องการเข้าใช้งานซอฟต์แวร์ ถัดมาจึงเป็นการพัฒนามอดูลจัดการไคลเอนต์ ซึ่งเป็นมอดูลสำหรับลงทะเบียนไคลเอนต์หรือซอฟต์แวร์ที่ต้องการเข้าใช้บริการยืนยันตัวตนของผู้ใช้งานผ่านระบบยืนยันตัวตนและกำหนดสิทธิ์การใช้งาน จากนั้นจึงเป็นการพัฒนามอดูลบันทึกการใช้งานระบบ ซึ่งจะเป็นมอดูลสำหรับบันทึกข้อมูลการเข้าใช้งานงานระบบ (Login) และบันทึกข้อมูลการออกจากระบบ (Logout)

ของผู้ใช้งาน เมื่อสิ้นสุดขั้นตอนของการพัฒนาระบบแล้วจึงเข้าสู่ขั้นตอนของการทดสอบและการประเมินความพึงพอใจในการใช้งานระบบโดยผู้เชี่ยวชาญ ซึ่งผลลัพธ์จากการประเมินอยู่ในระดับดีมาก

เนื่องจากในการพัฒนาระบบยืนยันตัวตนและกำหนดสิทธิ์การใช้งานผ่านเครือข่ายกลาง ได้พัฒนาขึ้นตามความต้องการที่ได้รับมาจากผู้ให้ความต้องการ แต่ในหลักการลำดับการทำงานและแนวคิดของไลบรารีต่างๆ ที่ใช้ในการตั้งค่าระบบยังคงมีส่วนของการทำงานของระบบในบางส่วนที่ได้พัฒนา ณ ปัจจุบันยังขาดไป ไม่ว่าจะเป็นการติดต่อระหว่างระบบเพื่อร้องขอข้อมูลที่เฉพาะเจาะจงของระบบนั้นๆ และการเข้ามาใช้บริการของระบบยืนยันตัวตนและกำหนดสิทธิ์การใช้งานผ่านเครือข่ายกลางโดยระบบที่ไม่ได้ทำงานอยู่บนเว็บไซต์ ดังนั้น ระบบยืนยันตัวตนและกำหนดสิทธิ์ การใช้งานผ่านเครือข่ายกลางยังสามารถถูกพัฒนาเพิ่มเติมความสามารถเพื่อต่อยอดการทำงานในส่วนที่ได้กล่าวไปข้างต้น อันเป็นการเพิ่มความสามารถของระบบให้ครอบคลุมการทำงานด้านต่างๆ มากยิ่งขึ้นในอนาคต

เอกสารอ้างอิง

[1] OAuth 2.0 (นิยามและความหมาย). เข้าถึงได้จาก: https://oauth.net/2/ (วันที่ค้นข้อมูล: 1 มิถุนายน 2565)

[2] OpenID Connect (นิยามและความหมาย). เข้าถึงได้จาก: https://openid.net/connect/ (วันที่ค้นข้อมูล: 6 มิถุนายน 2565)

[3] J. Hayes, "Policy-based authentication and authorization: secure access to the network infrastructure," Proceedings 16th Annual Computer Security Applications Conference (ACSAC'00), New Orleans, LA, USA, 2000, pp. 328-333, doi: 10.1109/ACSAC.2000.898887. (วันที่ค้นข้อมูล: 8 มิถุนายน 2565)

[4] Y. Sheng and Z. Lu, "An Online User Authentication Scheme for Web-Based services," 2008 International Seminar on Business and Information Management, Wuhan, China, 2008, pp. 173-176, doi: 10.1109/ISBIM.2008.217. (วันที่ค้นข้อมูล: 9 มิถุนายน 2565)

[5] N. Hossain, M. A. Hossain, M. Z. Hossain, M. H. I. Sohag and S. Rahman, "OAuth-SSO: A Framework to Secure the OAuth-Based SSO Service for Packaged Web Applications," 2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE), New York, NY, USA, 2018, pp. 1575-1578, doi: 10.1109/TrustCom/BigDataSE.2018.00227. (วันที่ค้นข้อมูล: 15 มิถุนายน 2565)

[6] ภาษา TypeScript (นิยามและความหมาย). เข้าถึงได้จาก: https://www.typescriptlang.org/docs/handbook//typescript-from-scratch.html (วันที่ค้นข้อมูล: 11 กรกฎาคม 2565)

[7] SQL Stored Procedures. เข้าถึงได้จาก: https://www.w3schools.com/sql/sql_

[8] Angular (นิยามและความหมาย). เข้าถึงได้จาก: https://angular.io/docs (วันที่ค้นข้อมูล: 11 กรกฎาคม 2565)

[9] ASP.NET Core (นิยามและความหมาย). เข้าถึงได้จาก: https://docs.microsoft.com/enus/aspnet/core/?view=aspnetcore-6.0 (วันที่ค้นข้อมูล: 25 กรกฎาคม 2565)

stored_procedures.asp/ (วันที่ค้นข้อมูล: 1 สิงหาคม 2564)

[10] Microsoft SQL Server Management Studio 18. เข้าถึงได้จาก: https://docs.microsoft .com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sqlserver-ver16 (วันที่ค้นข้อมูล: 27 มิถุนายน 2565)