



Monate

จัดทำโดย

| | | |
|------------------------------|-----------------------|-----------------------|
| นางสาวเกศราณี ไตรสวัสดิ์วงศ์ | รหัสประจำตัว 61070018 | กลุ่มการเรียนรู้ที่ 1 |
| นายคุณากร โสสิตสกุล | รหัสประจำตัว 61070020 | กลุ่มการเรียนรู้ที่ 1 |
| นายบรรณจา โสวรรณ | รหัสประจำตัว 61070097 | กลุ่มการเรียนรู้ที่ 1 |

เสนอ

อ.พัฒนพงษ์ ฉันทมิตรโอภาส,
ดร.สุพัฒน์ดา โชติพันธ์

รายงานนี้เป็นส่วนหนึ่งของวิชา การโปรแกรมอุปกรณ์เคลื่อนที่ (06016323)
หลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
ภาคเรียนที่ 1 ปีการศึกษา 2563
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

สารบัญ

| | |
|---|----|
| บทนำ | 6 |
| ที่มาและความสำคัญของโครงการ..... | 6 |
| วัตถุประสงค์ | 6 |
| หลักการทำงานของโครงการ | 7 |
| ขอบเขตของโครงการ..... | 7 |
| ประโยชน์ที่คาดว่าจะได้รับ | 7 |
| การออกแบบแอปพลิเคชัน | 8 |
| Use case Diagram..... | 8 |
| Use case Description | 9 |
| Anonymous User | 9 |
| User | 9 |
| User Interface Design | 12 |
| หน้าธุรกรรมทางการเงิน (Transaction Screen)..... | 12 |
| หน้ารายละเอียดธุรกรรมทางการเงิน (Transaction List Screen)..... | 14 |
| Edit Transaction Modal..... | 16 |
| หน้ารายงานข้อมูลทางสถิติของธุรกรรมทางการเงิน (Report Screen)..... | 22 |
| Time Range Modal..... | 24 |
| หน้าหนี้สิน (Debt Screen)..... | 26 |
| หน้ารายละเอียดลูกหนี้ (Debtor Detail Screen)..... | 29 |
| Debtor Debt Detail Modal | 31 |
| Debt Detail Modal | 33 |
| หน้าบัญชีผู้ใช้ (Account Screen) | 35 |
| หน้ารายละเอียดบัญชีผู้ใช้ (Account Detail Screen)..... | 37 |
| Edit Account Detail Modal..... | 39 |
| Notification Modal..... | 41 |

| | |
|---|-----------|
| Wallet List Modal | 43 |
| Edit Wallet Detail Modal..... | 45 |
| Create Wallet Modal..... | 48 |
| Add Wallet Member Modal..... | 50 |
| Create Modal..... | 52 |
| การเปลี่ยนแปลงแนวคิดจากแบบเสนอหัวข้อ | 55 |

สารบัญรูปภาพ

| | |
|--|----|
| รูปภาพที่ 1 หลักการทำงานของโครงการ | 7 |
| รูปภาพที่ 2 Use case Diagram..... | 8 |
| รูปภาพที่ 3 Transaction Screen | 12 |
| รูปภาพที่ 4 Transaction List Screen | 14 |
| รูปภาพที่ 5 Edit Transaction Modal..... | 16 |
| รูปภาพที่ 6 Picker for Type..... | 18 |
| รูปภาพที่ 7 Text Input for amount..... | 19 |
| รูปภาพที่ 8 Date Picker for edit transaction..... | 20 |
| รูปภาพที่ 9 Picker for category | 21 |
| รูปภาพที่ 10 Report Screen..... | 22 |
| รูปภาพที่ 11 Time Range Modal | 24 |
| รูปภาพที่ 12 Debt Screen | 26 |
| รูปภาพที่ 13 Debtor Detail Screen | 29 |
| รูปภาพที่ 14 Debtor Debt Detail Modal | 31 |
| รูปภาพที่ 15 Debt Detail Modal | 33 |
| รูปภาพที่ 16 Account Screen..... | 35 |
| รูปภาพที่ 17 Account Detail Screen..... | 37 |
| รูปภาพที่ 18 Edit Account Detail Modal | 39 |
| รูปภาพที่ 19 Notification Modal..... | 41 |
| รูปภาพที่ 20 Wallet List Modal..... | 43 |
| รูปภาพที่ 21 Edit Wallet Detail Modal..... | 45 |
| รูปภาพที่ 22 Delete Members from Edit Wallet Detail Modal..... | 47 |
| รูปภาพที่ 23 Create Wallet Modal..... | 48 |
| รูปภาพที่ 24 Add Wallet Member Modal | 50 |
| รูปภาพที่ 25 Create Modal | 52 |

สารบัญตาราง

| | |
|--|----|
| ตารางที่ 1 Use case Description ของ Anonymous User | 9 |
| ตารางที่ 2 Use case Description ของ User | 11 |

บทนำ

ที่มาและความสำคัญของโครงการ

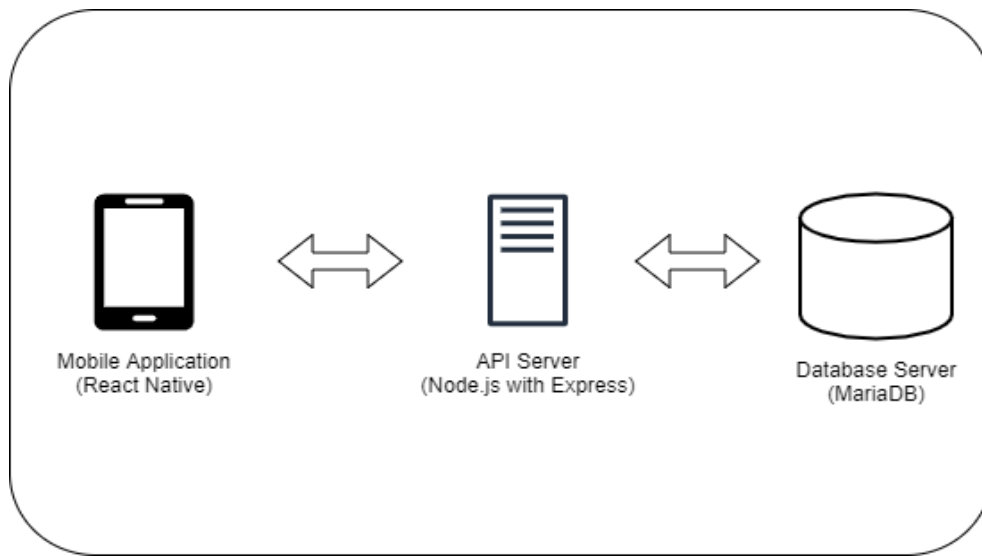
โดยทั่วไปแล้ว ผู้คนส่วนใหญ่มักจะไม่ทำบัญชีรายรับ-รายจ่ายทั้งแบบส่วนตัวและแบบเป็นกลุ่ม เช่น บัญชีรายรับ-รายจ่ายของครอบครัว เป็นต้น เพราะอาจจะมีความยุ่งยากและวุ่นวาย ซึ่งการที่ไม่มีการทำบัญชีรายรับ-รายจ่ายนั้นอาจจะส่งผลให้ประสบปัญหาเกี่ยวกับการวางแผนทางการเงินในภายหลังได้ เช่น ไม่มีเงินเหลือพอใช้จ่ายในช่วงปลายของเดือน ไม่มีเงินไว้ใช้ในยามวิกฤต เป็นต้น หรืออาจจะประสบปัญหา อย่างเช่น การให้คนที่เรารู้จักหยิบยืมเงินแล้วเกิดหลงลืมขึ้นมา ทำให้เราจำไม่ได้ว่าเงินในส่วนนี้ถูกใช้จ่ายไปกับสิ่งใด

คณะผู้จัดทำจึงมีความประสงค์ที่จะจัดทำแอปพลิเคชันบัญชีรายรับ-รายจ่ายทั้งแบบส่วนตัวและแบบเป็นกลุ่ม สำหรับโทรศัพท์มือถือ เพื่อช่วยในการจัดการบัญชีรายรับ-รายจ่ายทั้งแบบส่วนตัวและแบบเป็นกลุ่มให้มีความสะดวกสบายมากขึ้นและเข้าถึงได้โดยง่าย อีกทั้งยังมีระบบที่จะช่วยในการบันทึก และจัดการกับหนี้สินของเราได้ดียิ่งขึ้น

วัตถุประสงค์

- เพื่อให้จัดการและวางแผนการใช้จ่ายเงินได้ง่ายยิ่งขึ้น
- เพื่อให้จัดการและวางแผนการชำระหนี้สินได้ง่ายยิ่งขึ้น
- สามารถทำบัญชีรายรับ-รายจ่ายร่วมกับผู้อื่นได้ง่าย ผ่านการใช้แอปพลิเคชันของเรา
- ปลุกฝังลักษณะนิสัยการบันทึกรายรับ-รายจ่ายให้เป็นกิจจะลักษณะ

หลักการทำงานของโครงการ



รูปภาพที่ 1 หลักการทำงานของโครงการ

Client Side (Mobile Application)

- React Native: ใช้สำหรับสร้าง User Interface ของแอปพลิเคชัน

Server Side

- Node.js (NestJS): ใช้สำหรับเป็น API Service เพื่อใช้ในการติดต่อสื่อสาร จัดการข้อมูลภายในฐานข้อมูล
- MariaDB: ใช้เป็นฐานข้อมูลของแอปพลิเคชัน

ขอบเขตของโครงการ

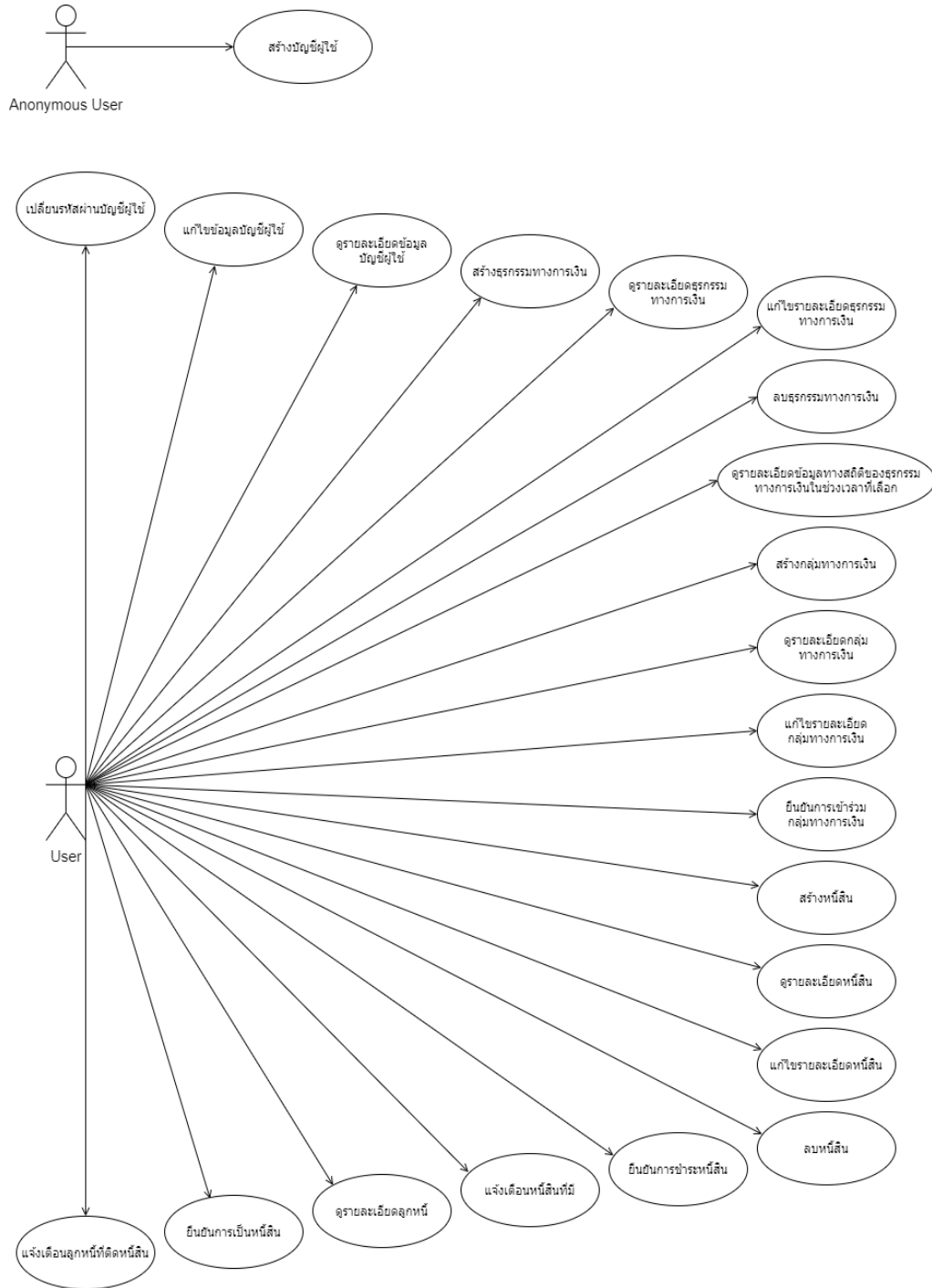
- โทรศัพท์ของผู้ใช้แอปพลิเคชันต้องสามารถใช้งานอินเทอร์เน็ตได้
- แอปพลิเคชันนี้เหมาะสำหรับผู้ที่ต้องการทำบัญชีรายรับ-รายจ่าย ทั้งแบบส่วนตัวและแบบเป็นกลุ่ม รวมไปถึงผู้ที่มีความต้องการที่จะจัดการเกี่ยวกับหนี้สินของตนเอง

ประโยชน์ที่คาดว่าจะได้รับ

- ผู้ใช้แอปพลิเคชันสามารถจัดการบัญชีรายรับ-รายจ่ายได้ง่ายยิ่งขึ้น
- ผู้ใช้แอปพลิเคชันสามารถวางแผนทางการเงินได้ดียิ่งขึ้น
- ผู้ใช้แอปพลิเคชันสามารถวางแผนชำระหนี้สินของตนเองได้ดียิ่งขึ้น
- ผู้ใช้แอปพลิเคชันสามารถตัดสินใจเกี่ยวกับการเงินของตนเองได้ดียิ่งขึ้น
- ผู้ใช้แอปพลิเคชันเห็นถึงแนวโน้มและภาพรวมการใช้จ่ายเงินของตนเอง

การออกแบบแอปพลิเคชัน

Use case Diagram



รูปภาพที่ 2 Use case Diagram

Use case Description

Anonymous User

| No. | Use case name | Use case Description |
|-----|------------------|--|
| 1 | สร้างบัญชีผู้ใช้ | ผู้ใช้แอปพลิเคชันสามารถสร้างบัญชีผู้ใช้ใหม่ที่ไม่อยู่ในระบบได้ เพื่อที่จะสามารถเข้าใช้งานระบบได้ |

ตารางที่ 1 Use case Description ของ Anonymous User

User

| No. | Use case name | Use case Description |
|-----|----------------------------------|---|
| 2 | เปลี่ยนรหัสผ่านบัญชีผู้ใช้ | ผู้ใช้แอปพลิเคชันสามารถเปลี่ยนรหัสผ่านบัญชีผู้ใช้ของตนเองที่มีอยู่ในระบบได้ เพื่อป้องกันการเข้าถึงข้อมูลทางการเงินของตนเอง |
| 3 | แก้ไขข้อมูลบัญชีผู้ใช้ | ผู้ใช้แอปพลิเคชันสามารถแก้ไขข้อมูลส่วนบุคคลของบัญชีผู้ใช้ตนเองที่มีอยู่ในระบบได้ เพื่ออัปเดตข้อมูลส่วนบุคคลให้มีความเป็นปัจจุบัน |
| 4 | ดูรายละเอียดข้อมูลบัญชีผู้ใช้ | ผู้ใช้แอปพลิเคชันสามารถดูรายละเอียดข้อมูลบัญชีผู้ใช้ของบัญชีผู้ใช้ตนเองที่มีอยู่ในระบบได้ เพื่อตรวจสอบข้อมูลส่วนบุคคลของตนเอง |
| 5 | สร้างธุรกรรมทางการเงิน | ผู้ใช้แอปพลิเคชันสามารถสร้างธุรกรรมทางการเงิน เพื่อบันทึกการรับ-รายจ่ายที่ทำการใช้จ่ายไปในแต่ละกลุ่มทางการเงินที่ตนเองเป็นสมาชิกอยู่ |
| 6 | ดูรายละเอียดธุรกรรมทางการเงิน | ผู้ใช้แอปพลิเคชันสามารถดูรายละเอียดธุรกรรมทางการเงินของกลุ่มทางการเงินที่ตนเองเป็นสมาชิกได้ เพื่อที่จะทราบข้อมูลธุรกรรมทางการเงินที่เกิดขึ้นภายในกลุ่มทางการเงินที่ตนเองเป็นสมาชิกได้ |
| 7 | แก้ไขรายละเอียดธุรกรรมทางการเงิน | ผู้ใช้แอปพลิเคชันสามารถแก้ไขรายละเอียดธุรกรรมทางการเงินที่ตนเองเป็นผู้สร้างขึ้นได้ |

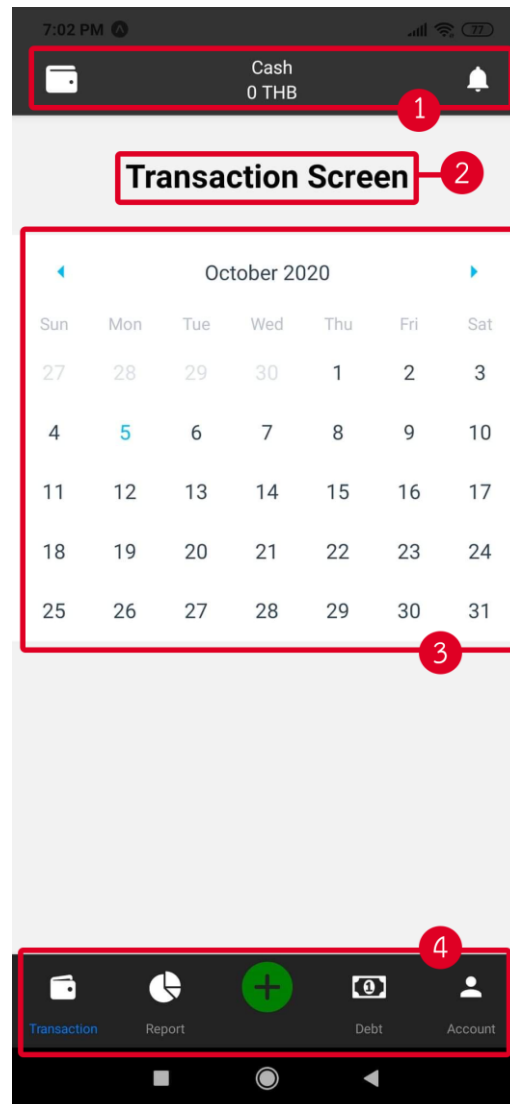
| | | |
|----|--|--|
| 8 | ลบธุรกรรมทางการเงิน | ผู้ใช้แอปพลิเคชันสามารถลบรายละเอียดธุรกรรมทางการเงินที่ตนเองเป็นผู้สร้างขึ้นได้ |
| 9 | ดูรายละเอียดข้อมูลทางสถิติของธุรกรรมทางการเงินในช่วงเวลาที่เลือก | ผู้ใช้แอปพลิเคชันสามารถดูรายละเอียดข้อมูลทางสถิติของธุรกรรมทางการเงินในช่วงเวลาที่เลือกได้ จากแต่ละกลุ่มทางการเงินที่ตนเองเป็นสมาชิกอยู่ |
| 10 | สร้างกลุ่มทางการเงิน | ผู้ใช้แอปพลิเคชันสามารถสร้างกลุ่มทางการเงิน เพื่อแบ่งกลุ่มทางการเงินและชักชวนผู้ใช้แอปพลิเคชันรายอื่น ๆ เข้าร่วมการทำบัญชีรายรับ-รายจ่ายแบบกลุ่มได้ |
| 11 | ดูรายละเอียดกลุ่มทางการเงิน | ผู้ใช้แอปพลิเคชันสามารถดูรายละเอียดกลุ่มทางการเงิน เพื่อที่จะทราบข้อมูลต่าง ๆ ของกลุ่มทางการเงินนั้น ๆ อย่างเช่น กลุ่มทางการเงินนี้มีใครเป็นสมาชิก |
| 12 | แก้ไขรายละเอียดกลุ่มทางการเงิน | ผู้ใช้แอปพลิเคชันสามารถแก้ไขรายละเอียดกลุ่มทางการเงิน เพื่อที่จะแก้ไขรายละเอียดข้อมูลกลุ่มทางการเงิน ออกจากกลุ่มทางการเงินที่ตนเองเป็นสมาชิกอยู่ หรือชักชวนผู้ใช้แอปพลิเคชันรายอื่น ๆ เพื่อเพิ่มสมาชิกกลุ่มทางการเงิน (กรณีที่เป็นผู้สร้างกลุ่มทางการเงินเท่านั้น) ได้ |
| 13 | ยืนยันการเข้าร่วมกลุ่มทางการเงิน | ผู้ใช้แอปพลิเคชันสามารถยืนยันการเข้าร่วมกลุ่มทางการเงิน เพื่อเข้าร่วมกลุ่มทางการเงินที่ตนเองถูกชักชวนได้ |
| 14 | สร้างหนี้สิน | ผู้ใช้แอปพลิเคชันสามารถสร้างหนี้สินได้ โดยแบ่งออกเป็น หนี้สินที่สร้างเพื่อเตือนตนเอง เช่น ค่าผ่อนบ้าน ค่าน้ำประปา ค่าไฟฟ้า และหนี้สินที่สร้างโดยมีผู้ใช้แอปพลิเคชันรายอื่น ๆ เกี่ยวข้อง (เป็นหนี้สินระหว่างผู้ใช้แอปพลิเคชัน 2 ราย) |
| 15 | ดูรายละเอียดหนี้สิน | ผู้ใช้แอปพลิเคชันสามารถดูรายละเอียดของหนี้สินได้ เพื่อตรวจสอบข้อมูลของหนี้สิน |
| 16 | แก้ไขรายละเอียดหนี้สิน | ผู้ใช้แอปพลิเคชันสามารถแก้ไขรายละเอียดของหนี้สินที่อยู่ในประเภทหนี้สินที่สร้างเพื่อเตือนตนเองได้ เพื่อแก้ไขรายละเอียดของหนี้สินให้มีความถูกต้อง |
| 17 | ลบหนี้สิน | ผู้ใช้แอปพลิเคชันสามารถลบหนี้สินที่อยู่ในประเภทหนี้สินที่สร้างเพื่อเตือนตนเองได้ เพื่อบ่งบอกว่าหนี้สินนี้ถูกชำระแล้ว |

| | | |
|----|-------------------------------|--|
| 18 | ยืนยันการชำระหนี้สิน | ผู้ใช้แอปพลิเคชันที่เป็นเจ้าหนี้สามารถยืนยันการชำระหนี้สินของลูกหนี้ได้ เพื่อบ่งบอกว่าหนี้สินนี้ได้รับการชำระแล้ว |
| 19 | แจ้งเตือนหนี้สินที่มี | ผู้ใช้แอปพลิเคชันสามารถแจ้งเตือนหนี้สินที่มีได้ ผ่านการตั้งวันสิ้นสุดหนี้สิน เพื่อให้ตนเองสามารถรับรู้ถึงหนี้สินและจัดการกับหนี้สินของตนเองได้อย่างมีประสิทธิภาพ |
| 20 | ดูรายละเอียดลูกหนี้ | ผู้ใช้แอปพลิเคชันสามารถดูรายละเอียดของลูกหนี้ได้ เพื่อตรวจสอบว่าผู้ใช้แอปพลิเคชันรายไหนเป็นผู้ที่มีภาระหนี้สินติดค้างกับตนเองอยู่ และมีรายการหนี้สินอะไรบ้าง |
| 21 | ยืนยันการเป็นหนี้สิน | ผู้ใช้แอปพลิเคชันสามารถสามารถยืนยันการเป็นหนี้สินได้ โดยทำการตรวจสอบรายละเอียดของหนี้สินที่เจ้าหนี้ได้ทำการสร้างขึ้นมา และเมื่อทำการยืนยันว่ารายละเอียดของหนี้สินนั้นถูกต้องแล้ว หนี้สินนั้นจะถูกบันทึกในหนี้สินของผู้ใช้แอปพลิเคชันที่เป็นผู้ยืนยัน |
| 22 | แจ้งเตือนลูกหนี้ที่ติดหนี้สิน | ผู้ใช้แอปพลิเคชันสามารถแจ้งเตือนลูกหนี้ที่มีภาระหนี้สินติดค้างกับตนเองได้ เพื่อให้ลูกหนี้ได้รับรู้ว่ามีหนี้สินคงเหลืออยู่ |

ตารางที่ 2 Use case Description ของ User

User Interface Design

หน้าธุรกรรมทางการเงิน (Transaction Screen)



รูปภาพที่ 3 Transaction Screen

หมายเลข 1: Top bar เป็น Component ที่ประกอบไปด้วย

- Wallet Icon เมื่อกดแล้วจะแสดง Wallet List Modal ที่หน้าจอต้อนรับของผู้ใช้แอปพลิเคชัน
- ส่วนของ Text Component ที่แสดงชื่อของกลุ่มทางการเงินที่ผู้ใช้แอปพลิเคชันเลือกดูอยู่ รวมไปถึงยอดเงินคงเหลือของกลุ่มทางการเงินนั้น
- Notification Icon เมื่อกดแล้วจะแสดง Notification Modal ที่หน้าจอต้อนรับของผู้ใช้แอปพลิเคชัน

หมายเลข 2: Text Component ที่มีการตั้งข้อความว่า “Transaction Screen” เพื่อบ่งบอกถึงหน้าจอที่กำลังแสดงอยู่ว่า ปัจจุบันอยู่ที่หน้าธุรกรรมทางการเงิน (Transaction Screen)

หมายเลข 3: Calendar เป็น Component ที่ import มาจาก react-native-calendars ซึ่งสามารถกำหนด Properties ได้หลากหลาย เช่น กำหนดวันที่สามารถกดได้ สามารถเปลี่ยนเดือนได้ด้วยการลากปฏิทินไปทางซ้ายหรือขวา รวมไปถึง onDayPress ที่สามารถกำหนดได้ว่าเมื่อกดที่วันในปฏิทินแล้วจะเกิดเหตุการณ์อะไรขึ้น

หมายเลข 4: Bottom bar เป็น Bottom Tab Navigator ที่ import มาจาก @react-navigation/bottom-tabs โดยสามารถสร้าง Bottom Tab Navigator ได้โดยการใช้คำสั่ง createBottomTabNavigator() และกำหนดค่าต่าง ๆ เพื่อใช้ในการเปลี่ยนหน้าจอโทรศัพท์ของผู้ใช้แอปพลิเคชัน ซึ่งจะมีไอคอนและคำอธิบายต่าง ๆ ที่เมื่อกดแล้วจะนำไปสู่หน้านั้น ๆ โดยจะประกอบไปด้วย

- Transaction เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้าธุรกรรมทางการเงิน (Transaction Screen)
- Report เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้ารายงานข้อมูลทางสถิติของธุรกรรมทางการเงิน (Report Screen)
- ปุ่มเพิ่มธุรกรรมทางการเงินหรือหนี้สิน เมื่อกดแล้วจะมี Create Modal ปรากฏขึ้นมาให้กรอกข้อมูลเพื่อเพิ่มธุรกรรมทางการเงินหรือหนี้สินใหม่
- Debt เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้าหนี้สิน (Debt Screen)
- Account เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้าบัญชีผู้ใช้ (Account Screen)

หน้ารายละเอียดธุรกรรมทางการเงิน (Transaction List Screen)



รูปภาพที่ 4 Transaction List Screen

หมายเลข 1: Top bar เป็น Component ที่ประกอบไปด้วย

- Wallet Icon เมื่อกดแล้วจะแสดง Wallet List Modal ที่หน้าจอโทรศัพท์ของผู้ใช้แอปพลิเคชัน
- ส่วนของ Text Component ที่แสดงชื่อของกลุ่มทางการเงินที่ผู้ใช้แอปพลิเคชันเลือกดูอยู่ รวมไปถึงยอดเงินคงเหลือของกลุ่มทางการเงินนั้น
- Notification Icon เมื่อกดแล้วจะแสดง Notification Modal ที่หน้าจอโทรศัพท์ของผู้ใช้แอปพลิเคชัน

หมายเลข 2: TouchableOpacity Component ที่กำหนดให้เป็นไอคอนลูกศร เมื่อกดแล้วจะสามารถเปลี่ยนวันที่ต้องการดูรายละเอียดธุรกรรมทางการเงินได้

หมายเลข 3: Text Component ที่รับค่าวันที่ต้องการจะดูรายละเอียดทางธุรกรรมจาก onPress ของ Calendar Component ซึ่งจะแสดงวันเดียวกับที่ผู้ใช้แอปพลิเคชันกดในปฏิทินที่หน้าธุรกรรมทางการเงิน (Transaction Screen)

หมายเลข 4: Text Component ที่อยู่ภายในซึ่งแสดงถึงยอดสรุปของรายรับในวันนั้น ๆ โดยถูกรอบคลุมด้วย View Component เพื่อจัดแต่งให้สวยงาม

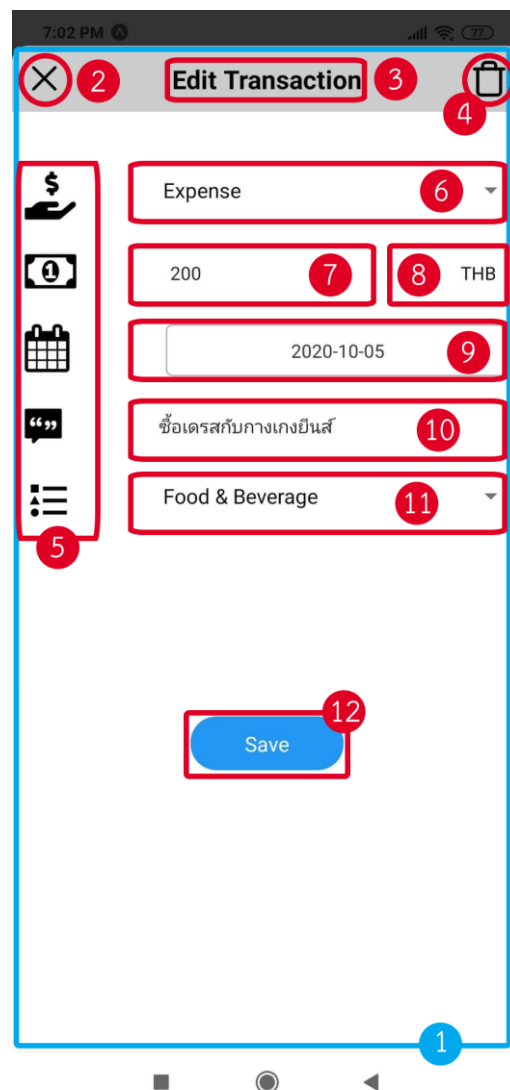
หมายเลข 5: Text Component ที่อยู่ภายในซึ่งแสดงถึงยอดสรุปของรายจ่ายในวันนั้น ๆ โดยถูกรอบคลุมด้วย View Component เพื่อจัดแต่งให้สวยงาม

หมายเลข 6: TouchableOpacity Component ที่จะแสดงตามจำนวนของธุรกรรมทางการเงินที่เกิดขึ้นในวันนั้น ๆ ยกตัวอย่างเช่น วันที่ 5 ตุลาคม พ.ศ.2563 มีธุรกรรมทางการเงินเกิดขึ้นแค่ 1 รายการ คือ ใช้จ่ายค่า Shopping โดยการซื้อเครื่องสำอางยี่ห้อเป็นจำนวนเงิน 200 บาท หน้าจอแอปพลิเคชันก็จะแสดงถึงจำนวนรายการธุรกรรมทางการเงินทั้งหมดที่เกิดขึ้นในวันนั้น ๆ ซึ่งก็คือ 1 รายการ โดยเมื่อกดที่ TouchableOpacity Component นั้น ๆ จะมี Modal Component ปรากฏขึ้นมา ในกรณีที่ผู้ใช้แอปพลิเคชันมีความต้องการที่จะแก้ไขรายละเอียดของธุรกรรมทางการเงินนั้น ๆ (Edit Transaction Modal) ดังรูปภาพที่ 5

หมายเลข 7: Bottom bar เป็น Bottom Tab Navigator ที่ import มาจาก @react-navigation/bottom-tabs โดยสามารถสร้าง Bottom Tab Navigator ได้โดยการใช้คำสั่ง createBottomTabNavigator() และกำหนดค่าต่าง ๆ เพื่อใช้ในการเปลี่ยนหน้าจอโทรศัพท์ของผู้ใช้แอปพลิเคชัน ซึ่งจะมีไอคอนและคำอธิบายต่าง ๆ ที่เมื่อกดแล้วจะนำไปสู่หน้านั้น ๆ โดยจะประกอบไปด้วย

- Transaction เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้าธุรกรรมทางการเงิน (Transaction Screen)
- Report เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้ารายงานข้อมูลทางสถิติของธุรกรรมทางการเงิน (Report Screen)
- ปุ่มเพิ่มธุรกรรมทางการเงินหรือหนี้สิน เมื่อกดแล้วจะมี Create Modal ปรากฏขึ้นมาให้กรอกข้อมูลเพื่อเพิ่มธุรกรรมทางการเงินหรือหนี้สินใหม่
- Debt เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้าหนี้สิน (Debt Screen)
- Account เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้าบัญชีผู้ใช้ (Account Screen)

Edit Transaction Modal



รูปภาพที่ 5 Edit Transaction Modal

ในการที่จะเข้าสู่ Edit Transaction Modal ได้นั้น ผู้ใช้แอปพลิเคชันจำเป็นต้องกดที่หมายเลข 6 ในรูปภาพที่ 4 ก่อน Edit Transaction Modal จึงจะปรากฏ โดยใน Modal นี้จะเป็น Modal สำหรับการแก้ไขรายละเอียดของธุรกรรมทางการเงินที่ผู้ใช้แอปพลิเคชันกด ซึ่งจะประกอบไปด้วย Component ที่ใช้ในการรับค่า หรือการกระทำพื้นฐานต่าง ๆ เช่น การปิด การลบ การบันทึก เป็นต้น

หมายเลข 1: Modal Component เป็น Container ที่ใช้สำหรับแสดงข้อมูลที่จะให้แสดงผลหรือไม่แสดงผลก็ได้ โดยการตั้งค่า Properties เหมือน Pop up ที่จะปรากฏขึ้นเมื่อผู้ใช้กดที่อะไรบางอย่าง แต่มีความแตกต่างจากการเปลี่ยนหน้าจอโทรศัพท์ของผู้ใช้แอปพลิเคชันในจุดที่ Modal นั้นเมื่อถูกเรียกใช้จะยังคงอยู่ในหน้าเดิมที่ถูกเรียกใช้เสมอ

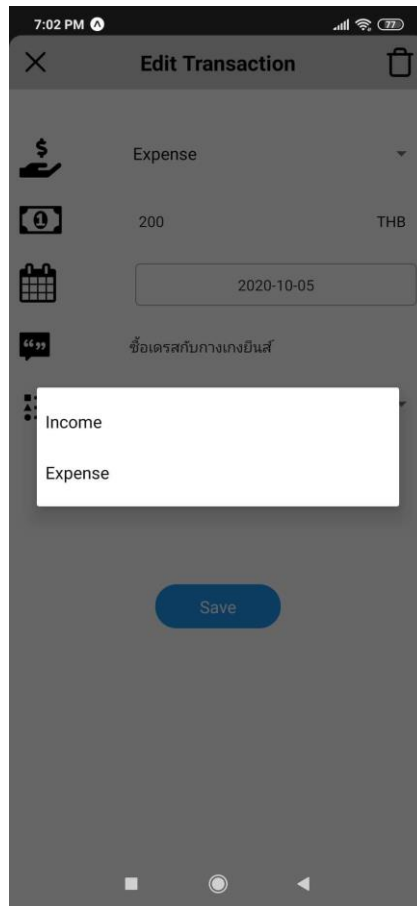
หมายเลข 2: TouchableOpacity Component ที่กำหนดให้เป็นไอคอนกากบาท เมื่อกดแล้ว Modal Component นี้จะไม่แสดงผลอีกต่อไป

หมายเลข 3: Text Component ที่มีการตั้งข้อความว่า “Edit Transaction” เพื่อบ่งบอกถึงหน้าจอที่กำลังแสดงว่าปัจจุบันอยู่ที่ Edit Transaction Modal

หมายเลข 4: TouchableOpacity Component ที่กำหนดให้เป็นไอคอนถังขยะ เมื่อกดแล้วจะมี Alert แจ้งเตือนขึ้นมาให้ผู้ใช้แอปพลิเคชันยืนยันว่าต้องการลบรายการธุรกรรมทางการเงินนี้หรือไม่

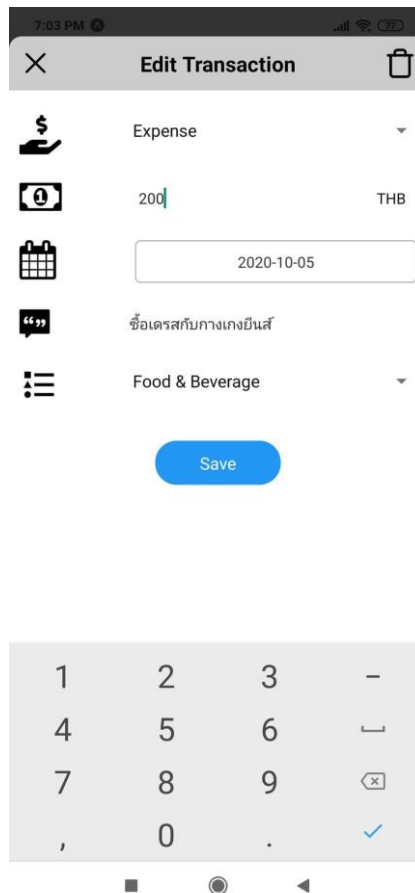
หมายเลข 5: Icon ซึ่งเป็น Component ที่ import มาจาก @expo/vector-icons ใช้สำหรับการแสดงผลให้สวยงาม หรือทำให้ผู้ใช้แอปพลิเคชันเข้าใจได้มากขึ้น

หมายเลข 6: Picker Component ที่ import มาจาก @react-native-community/picker ซึ่งใช้สำหรับเลือกประเภทของธุรกรรมทางการเงินนั้น ๆ เหมือนกับ Select Tag ของภาษา HTML เพียงแต่จะไม่แสดงในลักษณะ Drop down list แต่ปรากฏรายการให้ผู้ใช้แอปพลิเคชันเลือกขึ้นบนหน้าจอโทรศัพท์แทน ตัวอย่างการรับค่าตัวเลือกจากผู้ใช้แอปพลิเคชัน ดังรูปภาพที่ 6



รูปภาพที่ 6 Picker for Type

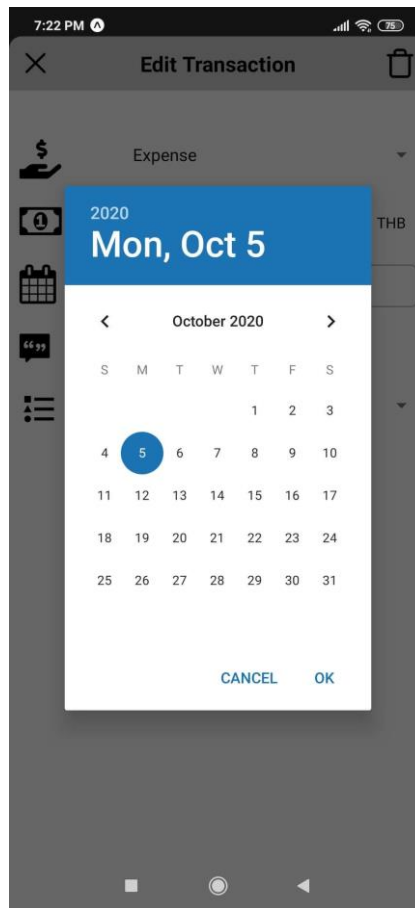
หมายเลข 7: TextInput Component ที่กำหนดประเภทของแป้นพิมพ์ให้เป็นประเภทที่พิมพ์ได้แค่ตัวเลขเท่านั้น (keyboardType="numeric") ซึ่งใช้สำหรับการพิมพ์จำนวนเงินของรายรับหรือรายจ่ายในธุรกรรมทางการเงินนั้น ๆ ดังรูปภาพที่ 7



รูปภาพที่ 7 Text Input for amount

หมายเลข 8: Text Component ที่มีการตั้งข้อความว่า “THB” เพื่อบ่งบอกให้ผู้ใช้แอปพลิเคชันเข้าใจว่าหน่วยเงินที่ให้ผู้ใช้อปพลิเคชันระบุนั้น คือ บาท

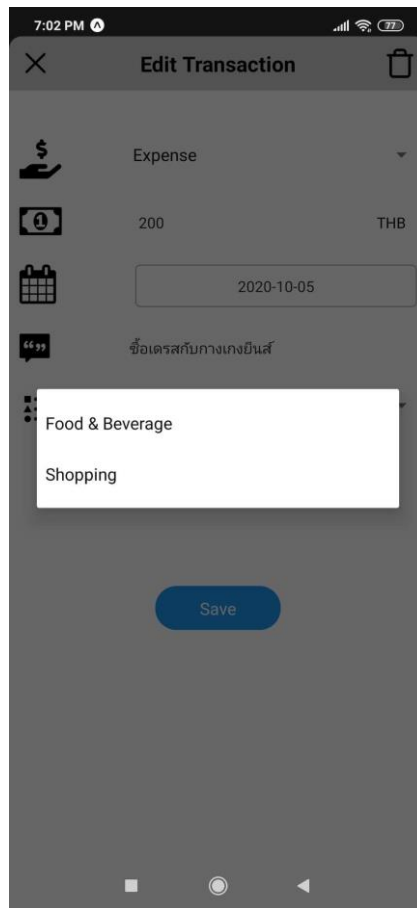
หมายเลข 9: DatePicker Component ที่ import มาจาก react-native-datepicker ซึ่งใช้สำหรับการกำหนดวันที่ผู้ใช้แอปพลิเคชันทำธุรกรรมทางการเงินนั้น โดยที่เราสามารถกำหนด Properties ได้หลากหลายขึ้นอยู่กับการใช้งาน เช่น รูปแบบของการแสดง วันในอดีตที่สามารถเลือกได้ หรือวันในอนาคตที่สามารถเลือกได้ ตัวอย่างการรับค่าวันที่จากผู้ใช้แอปพลิเคชันนั้นจะเป็นการกดปฏิทิน ดังรูปภาพที่ 8



รูปภาพที่ 8 Date Picker for edit transaction

หมายเลข 10: TextInput Component ที่รับค่าจากแป้นพิมพ์ ในกรณีที่ผู้ใช้แอปพลิเคชันต้องการจดบันทึกเกี่ยวกับรายการธุรกรรมทางการเงินนั้น ๆ เช่น ธุรกรรมทางการเงินนี้ใช้ทำอะไร เป็นต้น

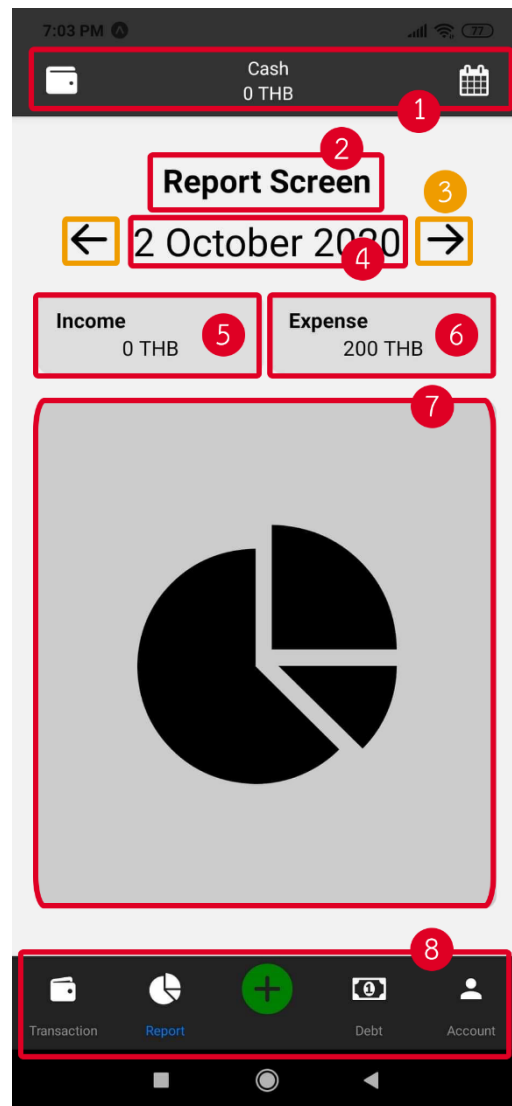
หมายเลข 11: Picker Component ที่ import มาจาก @react-native-community/picker ซึ่งใช้สำหรับเลือกหมวดหมู่ของธุรกรรมทางการเงินนั้น ๆ เหมือนกับ Select Tag ของภาษา HTML เพียงแต่จะไม่แสดงในลักษณะ Drop down list แต่ปรากฏรายการให้ผู้ใช้แอปพลิเคชันเลือกขึ้นบนหน้าจอโทรศัพท์แทนตัวอย่างการรับค่าตัวเลือกจากผู้ใช้แอปพลิเคชันดังรูปภาพที่ 9



รูปภาพที่ 9 Picker for category

หมายเลข 12: TouchableOpacity Component ที่มีการกำหนดรูปแบบให้เป็นปุ่มสำหรับบันทึกการแก้ไขธุรกรรมทางการเงินนั้น เมื่อผู้ใช้แอปพลิเคชันกดจะทำการบันทึกการแก้ไขทั้งหมดที่เกิดขึ้น

หน้ารายงานข้อมูลทางสถิติของธุรกรรมทางการเงิน (Report Screen)



รูปภาพที่ 10 Report Screen

หมายเลข 1: Top bar เป็น Component ที่ประกอบไปด้วย

- Wallet Icon เมื่อกดแล้วจะแสดง Wallet List Modal ที่หน้าจอโทรศัพท์ของผู้ใช้แอปพลิเคชัน
- ส่วนของ Text Component ที่แสดงชื่อของกลุ่มทางการเงินที่ผู้ใช้แอปพลิเคชันเลือกดูอยู่ รวมไปถึงยอดเงินคงเหลือของกลุ่มทางการเงินนั้น
- Calendar Icon เมื่อกดแล้วจะแสดง Time Range Modal ที่หน้าจอโทรศัพท์ของผู้ใช้แอปพลิเคชัน

หมายเลข 2: Text Component ที่มีการตั้งข้อความว่า “Report Screen” เพื่อบ่งบอกถึงหน้าจอที่กำลังแสดงอยู่ว่า ปัจจุบันอยู่ที่หน้ารายงานข้อมูลทางสถิติของธุรกรรมทางการเงิน (Report Screen)

หมายเลข 3: TouchableOpacity Component ที่กำหนดให้เป็นไอคอนลูกศร เมื่อกดแล้วจะสามารถเปลี่ยนช่วงเวลาที่ต้องการดูรายงานข้อมูลทางสถิติของธุรกรรมทางการเงินได้

หมายเลข 4: Text Component ที่รับค่าช่วงเวลาที่ต้องการจะดูรายงานข้อมูลทางสถิติของธุรกรรมทางการเงิน

หมายเลข 5: Text Component ที่อยู่ภายในซึ่งแสดงถึงยอดสรุปของรายรับในช่วงเวลานั้น ๆ โดยถูกรอบคลุมด้วย View Component เพื่อจัดแต่งให้สวยงาม

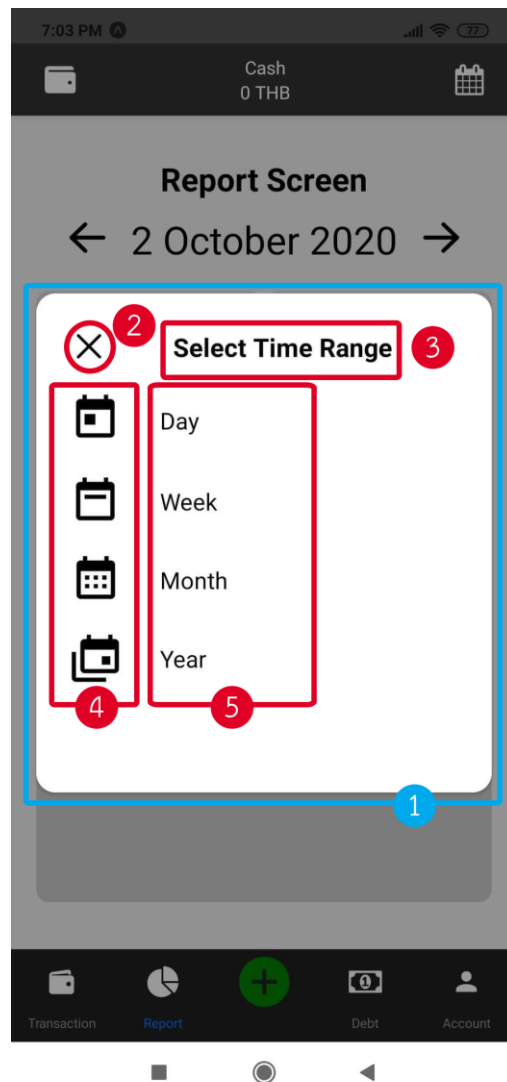
หมายเลข 6: Text Component ที่อยู่ภายในซึ่งแสดงถึงยอดสรุปของรายจ่ายในช่วงเวลานั้น ๆ โดยถูกรอบคลุมด้วย View Component เพื่อจัดแต่งให้สวยงาม

หมายเลข 7: View Component ที่ครอบคลุมแผนภาพทางสถิติที่สร้างขึ้นมาจากข้อมูลธุรกรรมทางการเงินทั้งหมดที่เกิดขึ้นในช่วงเวลาที่ผู้ใช้แอปพลิเคชันเลือก

หมายเลข 8: Bottom bar เป็น Bottom Tab Navigator ที่ import มาจาก @react-navigation/bottom-tabs โดยสามารถสร้าง Bottom Tab Navigator ได้โดยการใช้คำสั่ง createBottomTabNavigator() และกำหนดค่าต่าง ๆ เพื่อใช้ในการเปลี่ยนหน้าจอโทรศัพท์ของผู้ใช้แอปพลิเคชัน ซึ่งจะมีไอคอนและคำอธิบายต่าง ๆ ที่เมื่อกดแล้วจะนำไปสู่หน้านั้น ๆ โดยจะประกอบไปด้วย

- Transaction เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้าธุรกรรมทางการเงิน (Transaction Screen)
- Report เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้ารายงานข้อมูลทางสถิติของธุรกรรมทางการเงิน (Report Screen)
- ปุ่มเพิ่มธุรกรรมทางการเงินหรือหนี้สิน เมื่อกดแล้วจะมี Create Modal ปรากฏขึ้นมาให้กรอกข้อมูล เพื่อเพิ่มธุรกรรมทางการเงินหรือหนี้สินใหม่
- Debt เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้าหนี้สิน (Debt Screen)
- Account เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้าบัญชีผู้ใช้ (Account Screen)

Time Range Modal



รูปภาพที่ 11 Time Range Modal

ในการที่จะเข้าสู่ Time Range Modal ได้นั้น ผู้ใช้แอปพลิเคชันจำเป็นต้องกดที่ไอคอนปฏิทินที่อยู่ภายในหน้ารายงานข้อมูลทางสถิติของธุรกรรมทางการเงิน (Report Screen) ก่อน Time Range Modal จึงจะปรากฏ โดยใน Modal นี้จะเป็น Modal สำหรับเลือกช่วงเวลาที่ใช้แอปพลิเคชันต้องการจะดูรายงานข้อมูลทางสถิติของธุรกรรมทางการเงิน

หมายเลข 1: Overlay Component ที่ import มาจาก react-native-elements ซึ่งมีลักษณะคล้ายคลึงกับ View Component แต่จะมีความแตกต่างกันตรงที่ Overlay Component นั้นจะลอยจากเนื้อหาของแอปพลิเคชันและในกรณีที่ผู้ใช้แอปพลิเคชันกดที่หน้าจอตีพิมพ์ภายนอกบริเวณของ Overlay Component นั้นจะทำการปิด Modal นั้นโดยอัตโนมัติ ซึ่งเป็นวิธีที่ง่ายในกรณีที่ต้องการแจ้งสิ่งที่ต้องการให้กับผู้ใช้แอปพลิเคชัน

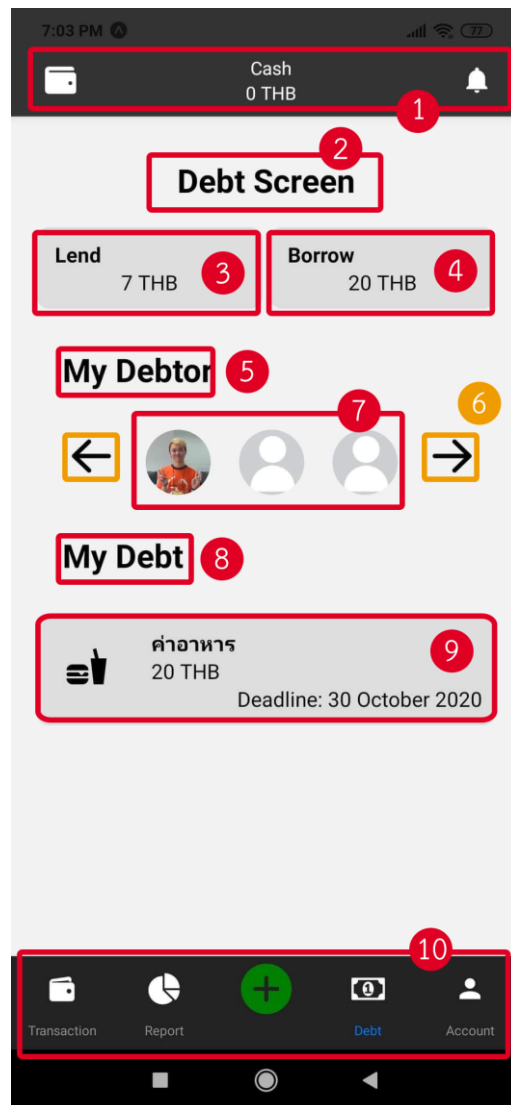
หมายเลข 2: TouchableOpacity Component ที่กำหนดให้เป็นไอคอนกากบาท เมื่อกดแล้ว Modal Component นี้จะไม่แสดงผลอีกต่อไป

หมายเลข 3: Text Component ที่มีการตั้งข้อความว่า “Selece Time Range” เพื่อบ่งบอกถึงหน้าจอที่กำลังแสดงว่าต้องการให้ผู้ใช้แอปพลิเคชันทำการเลือกช่วงเวลาที่ต้องการจะดูรายงานข้อมูลทางสถิติของธุรกรรมทางการเงิน

หมายเลข 4: Icon ซึ่งเป็น Component ที่ import มาจาก @expo/vector-icons ใช้สำหรับการแสดงผลให้สวยงาม หรือทำให้ผู้ใช้แอปพลิเคชันเข้าใจได้มากขึ้น

หมายเลข 5: Text Component ที่มีการตั้งข้อความตามแต่ละช่วงเวลา โดยมีจุดประสงค์ให้ผู้ใช้แอปพลิเคชันสามารถเลือกช่วงเวลาที่ต้องการจะดูรายงานข้อมูลทางสถิติของธุรกรรมทางการเงินได้

หน้าหนี้สิน (Debt Screen)



รูปภาพที่ 12 Debt Screen

หมายเลข 1: Top bar เป็น Component ที่ประกอบไปด้วย

- Wallet Icon เมื่อกดแล้วจะแสดง Wallet List Modal ที่หน้าจอตีพิมพ์ของผู้ใช้แอปพลิเคชัน
- ส่วนของ Text Component ที่แสดงชื่อของกลุ่มทางการเงินที่ผู้ใช้แอปพลิเคชันเลือกดูอยู่ รวมไปถึงยอดเงินคงเหลือของกลุ่มทางการเงินนั้น
- Notification Icon เมื่อกดแล้วจะแสดง Notification Modal ที่หน้าจอตีพิมพ์ของผู้ใช้แอปพลิเคชัน

หมายเลข 2: Text Component ที่มีการตั้งข้อความว่า “Debt Screen” เพื่อบ่งบอกถึงหน้าจอที่กำลังแสดงอยู่ว่าปัจจุบันอยู่ที่หน้าหนี้สิน (Debt Screen)

หมายเลข 3: Text Component ที่อยู่ภายในซึ่งแสดงถึงยอดสรุปของหนี้สินที่ผู้ใช้แอปพลิเคชันติดค้าง โดยถูกครอบคลุมนด้วย View Component เพื่อจัดแต่งให้สวยงาม

หมายเลข 4: Text Component อยู่ภายในซึ่งแสดงถึงยอดสรุปของหนี้สินที่ผู้ใช้แอปพลิเคชันรายอื่น ๆ ติดค้างผู้ใช้แอปพลิเคชัน โดยถูกครอบคลุมนด้วย View Component เพื่อจัดแต่งให้สวยงาม

หมายเลข 5: Text Component ที่มีการตั้งข้อความว่า “My Debtor” เพื่อบ่งบอกว่าผู้ใช้แอปพลิเคชันรายอื่น ๆ เหล่านี้คือบุคคลที่มีหนี้สินติดค้างกับผู้ใช้แอปพลิเคชัน

หมายเลข 6: TouchableOpacity Component ที่กำหนดให้เป็นไอคอนลูกศร เมื่อกดแล้วจะสามารถเลื่อนบุคคลที่มีหนี้สินติดค้างกับผู้ใช้แอปพลิเคชันได้ ในกรณีที่ผู้ใช้แอปพลิเคชันรายอื่นติดค้างหนี้สินมากกว่าที่หน้าจอโทรศัพท์จะรองรับการแสดงผลได้ทั้งหมด

หมายเลข 7: TouchableOpacity Component ที่ครอบคลุม Avatar Component ซึ่ง import มาจาก react-native-elements ใช้สำหรับแสดงรูปภาพประจำตัวของผู้ใช้แอปพลิเคชัน สามารถกำหนดรูปแบบได้หลากหลาย โดยเมื่อผู้ใช้แอปพลิเคชันกดที่รูปภาพประจำตัวของผู้ใช้แอปพลิเคชันรายอื่น ๆ แล้ว จะทำการเปลี่ยนหน้าจอโทรศัพท์ของผู้ใช้แอปพลิเคชันให้ไปที่หน้ารายละเอียดลูกหนี้ (Debtor Detail Screen) ดังรูปภาพที่ 13

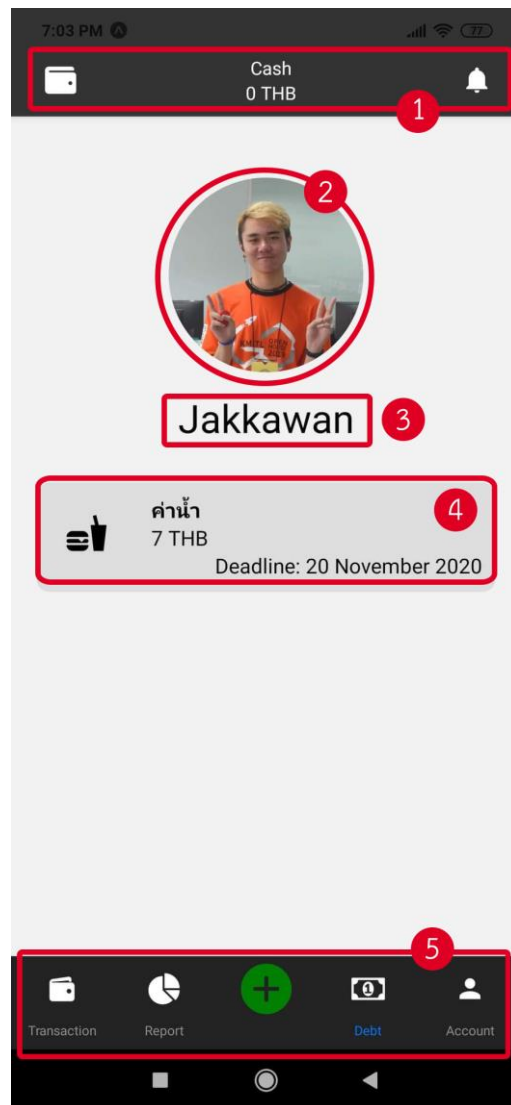
หมายเลข 8: Text Component ที่มีการตั้งข้อความว่า “My Debt” เพื่อบ่งบอกว่ารายการหนี้สินเหล่านี้คือหนี้สินที่ผู้ใช้แอปพลิเคชันติดค้างอยู่

หมายเลข 9: TouchableOpacity Component ที่จะแสดงตามจำนวนของหนี้สินที่ผู้ใช้แอปพลิเคชันติดค้าง โดยเมื่อกดที่ TouchableOpacity Component นั้น ๆ จะมี Modal Component ปรากฏขึ้นมา เพื่อแสดงรายละเอียดของหนี้สินนั้น ๆ (Debt Detail Modal) ดังรูปภาพที่ 15

หมายเลข 10: Bottom bar เป็น Bottom Tab Navigator ที่ import มาจาก @react-navigation/bottom-tabs โดยสามารถสร้าง Bottom Tab Navigator ได้โดยการใช้คำสั่ง createBottomTabNavigator() และกำหนดค่าต่าง ๆ เพื่อใช้ในการเปลี่ยนหน้าจอโทรศัพท์ของผู้ใช้แอปพลิเคชัน ซึ่งจะมีไอคอนและคำอธิบายต่าง ๆ ที่เมื่อกดแล้วจะนำไปสู่หน้านั้น ๆ โดยจะประกอบไปด้วย

- Transaction เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้าธุรกรรมทางการเงิน (Transaction Screen)
- Report เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้ารายงานข้อมูลทางสถิติของธุรกรรมทางการเงิน (Report Screen)
- ปุ่มเพิ่มธุรกรรมทางการเงินหรือหนี้สิน เมื่อกดแล้วจะมี Create Modal ปรากฏขึ้นมาให้กรอกข้อมูล เพื่อเพิ่มธุรกรรมทางการเงินหรือหนี้สินใหม่
- Debt เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้าหนี้สิน (Debt Screen)
- Account เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้าบัญชีผู้ใช้ (Account Screen)

หน้ารายละเอียดลูกหนี้ (Debtor Detail Screen)



รูปภาพที่ 13 Debtor Detail Screen

หมายเลข 1: Top bar เป็น Component ที่ประกอบไปด้วย

- Wallet Icon เมื่อกดแล้วจะแสดง Wallet List Modal ที่หน้าจอโทรศัพท์ของผู้ใช้แอปพลิเคชัน
- ส่วนของ Text Component ที่แสดงชื่อของกลุ่มทางการเงินที่ผู้ใช้แอปพลิเคชันเลือกดูอยู่ รวมไปถึงยอดเงินคงเหลือของกลุ่มทางการเงินนั้น
- Notification Icon เมื่อกดแล้วจะแสดง Notification Modal ที่หน้าจอโทรศัพท์ของผู้ใช้แอปพลิเคชัน

หมายเลข 2: Avatar Component ที่ import มาจาก react-native-elements ซึ่งใช้สำหรับแสดงรูปภาพประจำตัวของผู้ใช้แอปพลิเคชัน สามารถกำหนดรูปแบบได้หลากหลาย

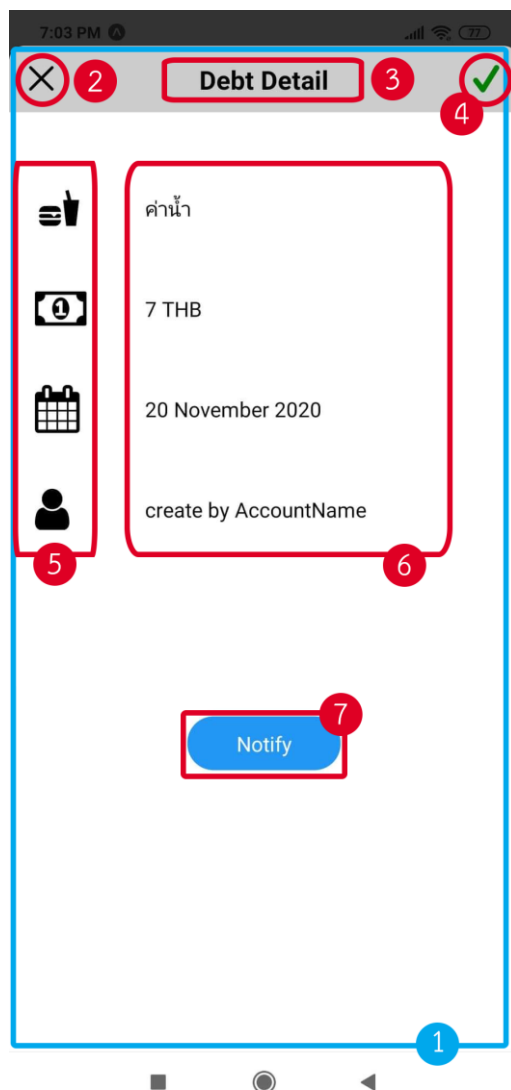
หมายเลข 3: Text Component ที่มีการตั้งข้อความตามชื่อผู้ใช้งานของผู้ใช้แอปพลิเคชันรายอื่น เพื่อบ่งบอกถึงตัวตนของผู้ใช้แอปพลิเคชันรายนั้น ๆ

หมายเลข 4: TouchableOpacity Component ที่จะแสดงตามจำนวนของหนี้สินที่ผู้ใช้แอปพลิเคชันรายนั้น ๆ ติดค้างผู้ใช้แอปพลิเคชัน โดยเมื่อกดที่ TouchableOpacity Component นั้น ๆ จะมี Modal Component ปรากฏขึ้นมา เพื่อแสดงรายละเอียดของหนี้สินที่ผู้ใช้แอปพลิเคชันรายนั้นติดค้าง (Debtor Debt Detail Modal) ดังรูปภาพที่ 14

หมายเลข 5: Bottom bar เป็น Bottom Tab Navigator ที่ import มาจาก @react-navigation/bottom-tabs โดยสามารถสร้าง Bottom Tab Navigator ได้โดยการใช้คำสั่ง createBottomTabNavigator() และกำหนดค่าต่าง ๆ เพื่อใช้ในการเปลี่ยนหน้าจอโทรศัพท์ของผู้ใช้แอปพลิเคชัน ซึ่งจะมีไอคอนและคำอธิบายต่าง ๆ ที่เมื่อกดแล้วจะนำไปสู่หน้านั้น ๆ โดยจะประกอบไปด้วย

- Transaction เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้าธุรกรรมทางการเงิน (Transaction Screen)
- Report เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้ารายงานข้อมูลทางสถิติของธุรกรรมทางการเงิน (Report Screen)
- ปุ่มเพิ่มธุรกรรมทางการเงินหรือหนี้สิน เมื่อกดแล้วจะมี Create Modal ปรากฏขึ้นมาให้กรอกข้อมูลเพื่อเพิ่มธุรกรรมทางการเงินหรือหนี้สินใหม่
- Debt เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้าหนี้สิน (Debt Screen)
- Account เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้าบัญชีผู้ใช้ (Account Screen)

Debtor Debt Detail Modal



รูปภาพที่ 14 Debtor Debt Detail Modal

ในการที่จะเข้าสู่ Debtor Debt Detail Modal ได้นั้น ผู้ใช้แอปพลิเคชันจำเป็นต้องกดที่หมายเลข 4 ในรูปภาพที่ 13 Debtor Detail Screen ก่อน Debtor Debt Detail Modal จึงจะปรากฏ โดยใน Modal นี้จะเป็น Modal สำหรับแสดงรายละเอียดของหนี้สินที่ผู้ใช้แอปพลิเคชันรายอื่นติดค้าง

หมายเลข 1: Modal Component เป็น Container ที่ใช้สำหรับแสดงข้อมูลที่จะให้แสดงผลหรือไม่แสดงผลก็ได้ โดยการตั้งค่า Properties เหมือน Pop up ที่จะปรากฏขึ้นเมื่อผู้ใช้กดที่อะไรบางอย่าง แต่มีความแตกต่างจากการเปลี่ยนหน้าจอโทรศัพท์ของผู้ใช้แอปพลิเคชันในจุดที่ Modal นั้นเมื่อถูกเรียกใช้จะยังคงอยู่ในหน้าเดิมที่ถูกเรียกใช้เสมอ

หมายเลข 2: TouchableOpacity Component ที่กำหนดให้เป็นไอคอนกากบาท เมื่อกดแล้ว Modal Component นี้จะไม่แสดงผลอีกต่อไป

หมายเลข 3: Text Component ที่มีการตั้งข้อความว่า “Debt Detail” เพื่อป้องกันหน้าจอที่กำลังแสดงว่าปัจจุบันอยู่ที่ Debtor Debt Detail Modal

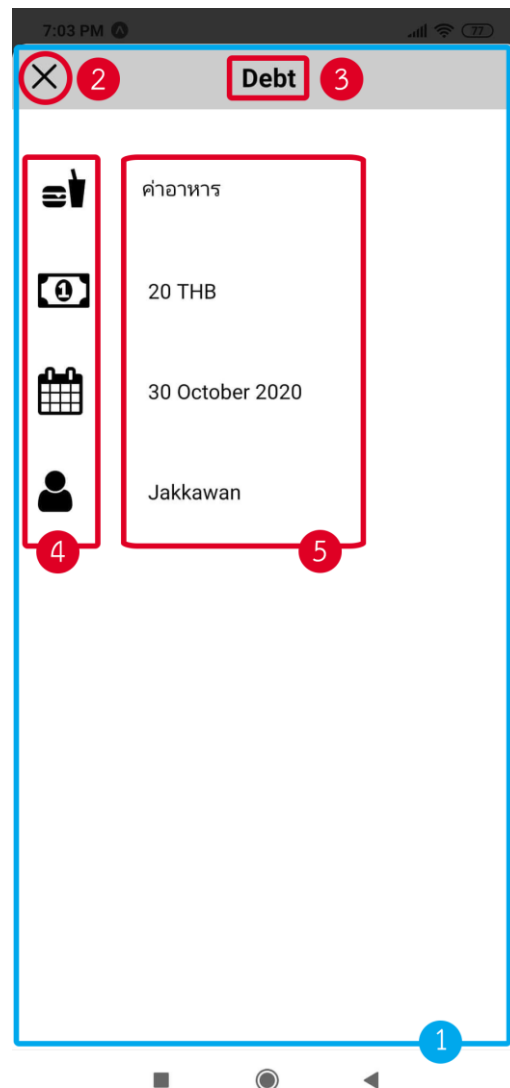
หมายเลข 4: TouchableOpacity Component ที่กำหนดให้เป็นไอคอนเครื่องหมายถูก เมื่อกดแล้วจะทำการยืนยันว่าหนี้สินนี้ได้รับการชำระจากผู้ใช้ออปพลิเคชันรายนั้น ๆ แล้ว

หมายเลข 5: Icon ซึ่งเป็น Component ที่ import มาจาก @expo/vector-icons ใช้สำหรับการแสดงผลให้สวยงาม หรือทำให้ผู้ใช้ออปพลิเคชันเข้าใจได้มากขึ้น

หมายเลข 6: Text Component ที่มีการตั้งข้อความตามข้อมูลของรายการหนี้สินนั้น ๆ

หมายเลข 7: TouchableOpacity Component ที่มีการกำหนดรูปแบบให้เป็นปุ่มสำหรับแจ้งเตือนหนี้สินกับผู้ใช้ออปพลิเคชันรายนั้น ๆ เมื่อผู้ใช้ออปพลิเคชันกดจะทำการส่งการแจ้งเตือนไปยังผู้ใช้ออปพลิเคชันรายนั้น ๆ

Debt Detail Modal



รูปภาพที่ 15 Debt Detail Modal

ในการที่จะเข้าสู่ Debt Detail Modal ได้นั้น ผู้ใช้แอปพลิเคชันจำเป็นต้องกดที่หมายเลข 9 ในรูปภาพที่ 12 ก่อน Debt Detail Modal จึงจะปรากฏ โดยใน Modal นี้จะเป็น Modal สำหรับแสดงรายละเอียดของหนี้สินที่ผู้ใช้แอปพลิเคชันกด

หมายเลข 1: Modal Component เป็น Container ที่ใช้สำหรับแสดงข้อมูลที่จะให้แสดงผลหรือไม่แสดงผลก็ได้ โดยการตั้งค่า Properties เหมือน Pop up ที่จะปรากฏขึ้นเมื่อผู้ใช้กดที่อะไรบางอย่าง แต่มีความแตกต่างจากการเปลี่ยนหน้าจอโทรศัพท์ของผู้ใช้แอปพลิเคชันในจุดที่ Modal นั้นเมื่อถูกเรียกใช้จะยังคงอยู่ในหน้าเดิมที่ถูกเรียกใช้เสมอ

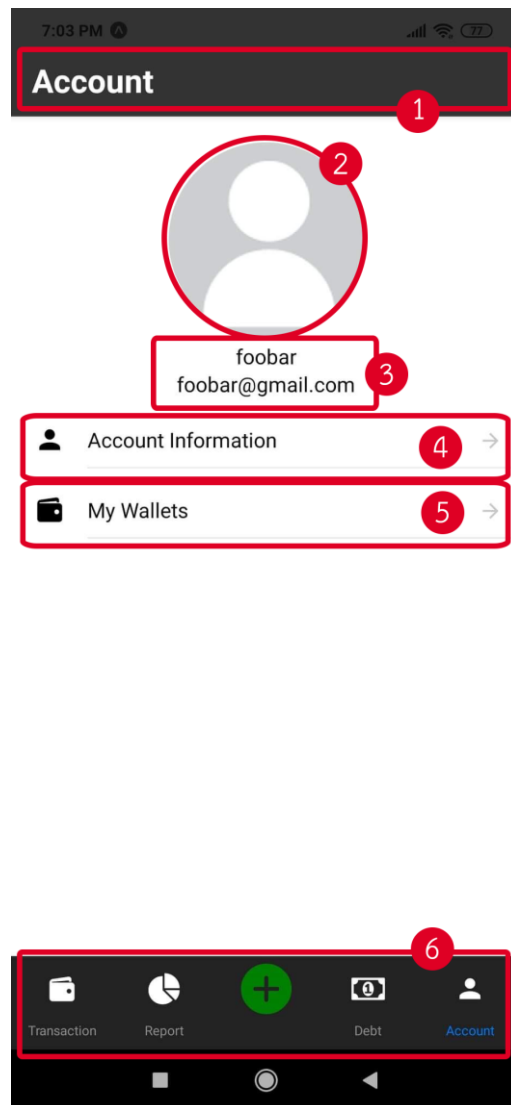
หมายเลข 2: TouchableOpacity Component ที่กำหนดให้เป็นไอคอนกากบาท เมื่อกดแล้ว Modal Component นี้จะไม่แสดงผลอีกต่อไป

หมายเลข 3: Text Component ที่มีการตั้งข้อความว่า “Debt” เพื่อบ่งบอกถึงหน้าจอที่กำลังแสดงว่าปัจจุบันอยู่ที่ Debt Detail Modal

หมายเลข 4: Icon ซึ่งเป็น Component ที่ import มาจาก @expo/vector-icons ใช้สำหรับการแสดงผลให้สวยงาม หรือทำให้ผู้ใช้แอปพลิเคชันเข้าใจได้มากขึ้น

หมายเลข 5: Text Component ที่มีการตั้งข้อความตามข้อมูลของรายการหนี้สินนั้น ๆ

หน้าบัญชีผู้ใช้ (Account Screen)



รูปภาพที่ 16 Account Screen

หมายเลข 1: Header Bar ที่มี Text Component อยู่ภายใน โดยกำหนดให้มีการตั้งข้อความว่า “Account” เพื่อบ่งบอกถึงหน้าจอที่กำลังแสดงอยู่ว่าปัจจุบันอยู่ที่หน้าบัญชีผู้ใช้ (Account Screen)

หมายเลข 2: Avatar Component ที่ import มาจาก react-native-elements ซึ่งใช้สำหรับแสดงรูปภาพประจำตัวของผู้ใช้แอปพลิเคชัน สามารถกำหนดรูปแบบได้หลากหลาย

หมายเลข 3: Text Component ที่มีการตั้งข้อความตามชื่อผู้ใช้งานและอีเมลของผู้ใช้แอปพลิเคชัน เพื่อบ่งบอกถึงตัวตนของผู้ใช้แอปพลิเคชัน

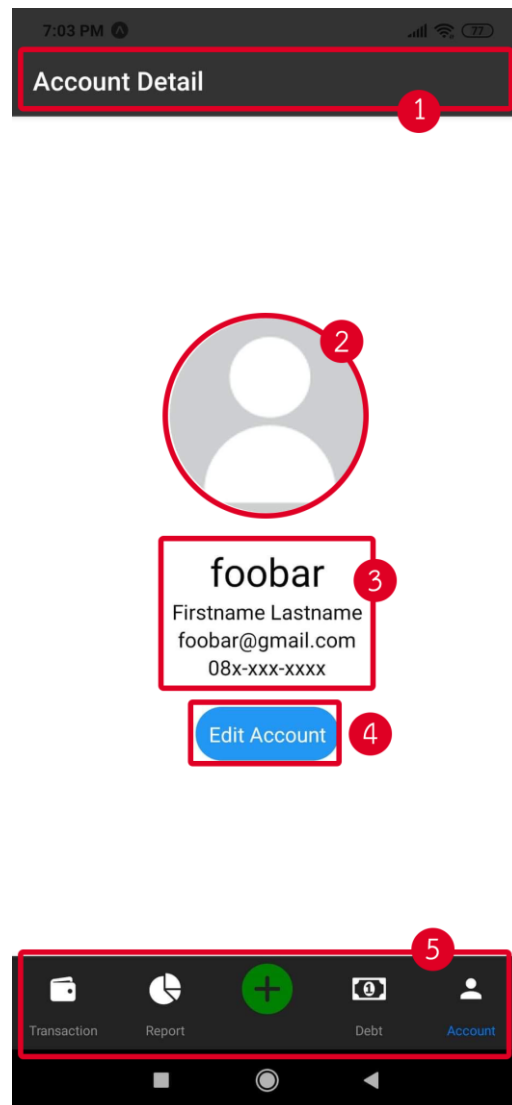
หมายเลข 4: ListItem Component ที่ import มาจาก native-base ใช้สำหรับแสดงข้อมูลในรูปแบบของลิสต์ ซึ่งภายใน ListItem Component จะมี Icon Component และ Text Component อยู่ภายใน และยังมีการตั้งค่าให้ในกรณีที่ผู้ใช้แอปพลิเคชันกดที่ ListItem Component นี้ จะทำการเปลี่ยนหน้าจอโทรศัพท์ของผู้ใช้แอปพลิเคชันให้เข้าสู่หน้ารายละเอียดบัญชีผู้ใช้ (Account Detail Screen) ดังรูปภาพที่ 17

หมายเลข 5: ListItem Component ที่ import มาจาก native-base ใช้สำหรับแสดงข้อมูลในรูปแบบของลิสต์ ซึ่งภายใน ListItem Component จะมี Icon Component และ Text Component อยู่ภายใน และยังมีการตั้งค่าให้ในกรณีที่ผู้ใช้แอปพลิเคชันกดที่ ListItem Component นี้ จะมี Wallet List Modal ปรากฏบนหน้าจอโทรศัพท์ของผู้ใช้แอปพลิเคชัน ดังรูปภาพที่ 20

หมายเลข 6: Bottom bar เป็น Bottom Tab Navigator ที่ import มาจาก @react-navigation/bottom-tabs โดยสามารถสร้าง Bottom Tab Navigator ได้โดยการใช้คำสั่ง createBottomTabNavigator() และกำหนดค่าต่าง ๆ เพื่อใช้ในการเปลี่ยนหน้าจอโทรศัพท์ของผู้ใช้แอปพลิเคชัน ซึ่งจะมีไอคอนและคำอธิบายต่าง ๆ ที่เมื่อกดแล้วจะนำไปสู่หน้านั้น ๆ โดยจะประกอบไปด้วย

- Transaction เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้าธุรกรรมทางการเงิน (Transaction Screen)
- Report เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้ารายงานข้อมูลทางสถิติของธุรกรรมทางการเงิน (Report Screen)
- ปุ่มเพิ่มธุรกรรมทางการเงินหรือหนี้สิน เมื่อกดแล้วจะมี Create Modal ปรากฏขึ้นมาให้กรอกข้อมูลเพื่อเพิ่มธุรกรรมทางการเงินหรือหนี้สินใหม่
- Debt เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้าหนี้สิน (Debt Screen)
- Account เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้าบัญชีผู้ใช้ (Account Screen)

หน้ารายละเอียดบัญชีผู้ใช้ (Account Detail Screen)



รูปภาพที่ 17 Account Detail Screen

หมายเลข 1: Header Bar ที่มี Text Component อยู่ภายใน โดยกำหนดให้มีการตั้งข้อความว่า “Account Detail” เพื่อบ่งบอกถึงหน้าจอที่กำลังแสดงอยู่ว่าปัจจุบันอยู่ที่หน้ารายละเอียดบัญชีผู้ใช้ (Account Detail Screen)

หมายเลข 2: Avatar Component ที่ import มาจาก react-native-elements ซึ่งใช้สำหรับแสดงรูปภาพประจำตัวของผู้ใช้แอปพลิเคชัน สามารถกำหนดรูปแบบได้หลากหลาย

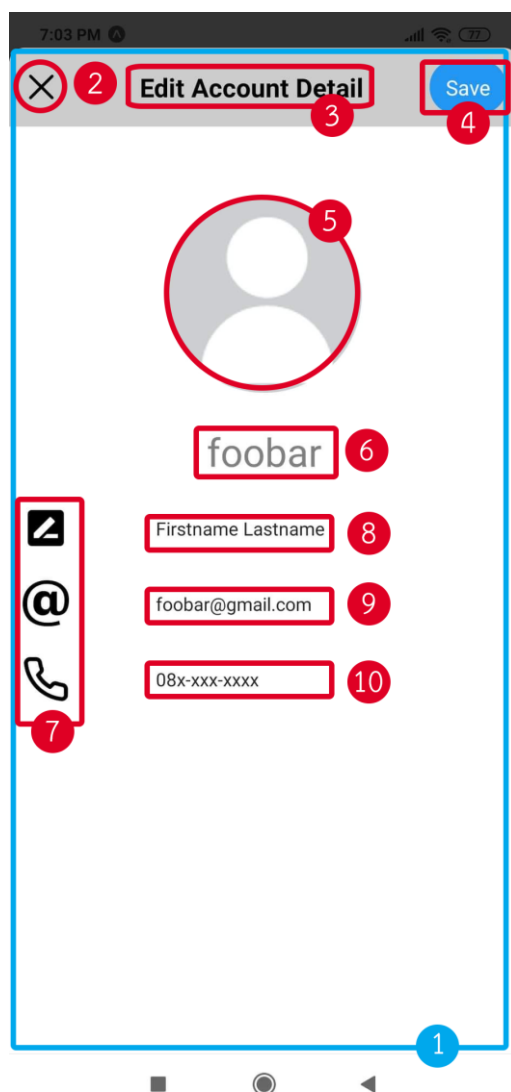
หมายเลข 3: Text Component ที่มีการตั้งข้อความตามข้อมูลของผู้ใช้แอปพลิเคชันที่มีในระบบ เพื่อบ่งบอกถึงตัวตนของผู้ใช้แอปพลิเคชัน

หมายเลข 4: TouchableOpacity Component ที่มีการกำหนดรูปแบบให้เป็นปุ่มสำหรับแก้ไขรายละเอียดบัญชีผู้ใช้ เมื่อผู้ใช้แอปพลิเคชันกดจะมี Edit Account Detail Modal ปรากฏขึ้นบนหน้าจอโทรศัพท์ของผู้ใช้แอปพลิเคชัน ดังรูปภาพที่ 18

หมายเลข 5: Bottom bar เป็น Bottom Tab Navigator ที่ import มาจาก @react-navigation/bottom-tabs โดยสามารถสร้าง Bottom Tab Navigator ได้โดยการใช้คำสั่ง createBottomTabNavigator() และกำหนดค่าต่าง ๆ เพื่อใช้ในการเปลี่ยนหน้าจอโทรศัพท์ของผู้ใช้แอปพลิเคชัน ซึ่งจะมีไอคอนและคำอธิบายต่าง ๆ ที่เมื่อกดแล้วจะนำไปสู่หน้านั้น ๆ โดยจะประกอบไปด้วย

- Transaction เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้าธุรกรรมทางการเงิน (Transaction Screen)
- Report เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้ารายงานข้อมูลทางสถิติของธุรกรรมทางการเงิน (Report Screen)
- ปุ่มเพิ่มธุรกรรมทางการเงินหรือหนี้สิน เมื่อกดแล้วจะมี Create Modal ปรากฏขึ้นมาให้กรอกข้อมูล เพื่อเพิ่มธุรกรรมทางการเงินหรือหนี้สินใหม่
- Debt เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้าหนี้สิน (Debt Screen)
- Account เมื่อกดแล้วจะนำทางหน้าจอโทรศัพท์ของผู้ใช้ให้เข้าสู่หน้าบัญชีผู้ใช้ (Account Screen)

Edit Account Detail Modal



รูปภาพที่ 18 Edit Account Detail Modal

ในการที่จะเข้าสู่ Edit Account Detail Modal ได้นั้น ผู้ใช้แอปพลิเคชันจำเป็นต้องกดที่หมายเลข 4 ในรูปภาพที่ 17 ก่อน Edit Account Detail Modal จึงจะปรากฏ โดยใน Modal นี้จะเป็น Modal สำหรับการแก้ไขรายละเอียดของบัญชีผู้ใช้แอปพลิเคชัน ซึ่งจะประกอบไปด้วย Component ที่ใช้ในการรับค่า หรือการกระทำพื้นฐานต่าง ๆ เช่น การปิด การบันทึก เป็นต้น

หมายเลข 1: Modal Component เป็น Container ที่ใช้สำหรับแสดงข้อมูลที่จะให้แสดงผลหรือไม่แสดงผลก็ได้ โดยการตั้งค่า Properties เหมือน Pop up ที่จะปรากฏขึ้นเมื่อผู้ใช้กดที่อะไรบางอย่าง แต่มีความแตกต่างจากการเปลี่ยนหน้าจอ โทรศัพท์ของผู้ใช้แอปพลิเคชันในจุดที่ Modal นั้นเมื่อถูกเรียกใช้จะยังคงอยู่ในหน้าเดิมที่ถูกเรียกใช้เสมอ

หมายเลข 2: TouchableOpacity Component ที่กำหนดให้เป็นไอคอนกากบาท เมื่อกดแล้ว Modal Component นี้จะไม่แสดงผลอีกต่อไป

หมายเลข 3: Text Component ที่มีการตั้งข้อความว่า “Edit Account Detail” เพื่อบ่งบอกถึงหน้าจอที่กำลังแสดงว่าปัจจุบันอยู่ที่ Edit Account Detail Modal

หมายเลข 4: TouchableOpacity Component ที่มีการกำหนดรูปแบบให้เป็นปุ่มสำหรับบันทึกการแก้ไขรายละเอียดบัญชีผู้ใช้นั้น เมื่อผู้ใช้แอปพลิเคชันกดจะทำการบันทึกการแก้ไขทั้งหมดที่เกิดขึ้น

หมายเลข 5: ImagePicker Component ที่ import มาจาก react-native-image-picker ซึ่งใช้สำหรับเปลี่ยนแปลงหรือแก้ไขรูปภาพประจำตัวของผู้ใช้แอปพลิเคชัน

หมายเลข 6: Text Component ที่มีการตั้งข้อความตามชื่อผู้ใช้งานของผู้ใช้แอปพลิเคชัน เพื่อบ่งบอกถึงตัวตนของผู้ใช้แอปพลิเคชัน

หมายเลข 7: Icon ซึ่งเป็น Component ที่ import มาจาก @expo/vector-icons ใช้สำหรับการแสดงผลให้สวยงาม หรือทำให้ผู้ใช้แอปพลิเคชันเข้าใจได้มากขึ้น

หมายเลข 8: TextInput Component ที่รับค่าจากแป้นพิมพ์ ซึ่งใช้สำหรับเปลี่ยนแปลงหรือแก้ไขชื่อและนามสกุลของผู้ใช้แอปพลิเคชัน

หมายเลข 9: TextInput Component ที่รับค่าจากแป้นพิมพ์ ซึ่งใช้สำหรับเปลี่ยนแปลงหรือแก้ไขอีเมลของผู้ใช้แอปพลิเคชัน

หมายเลข 10: TextInput Component ที่รับค่าจากแป้นพิมพ์ ซึ่งใช้สำหรับเปลี่ยนแปลงหรือแก้ไขหมายเลขโทรศัพท์มือถือของผู้ใช้แอปพลิเคชัน

Notification Modal



รูปภาพที่ 19 Notification Modal

ในการที่จะเข้าสู่ Notification Modal ได้นั้น ผู้ใช้แอปพลิเคชันจำเป็นต้องกดที่ไอคอนกระดิ่ง ภายในหน้าใดก็ได้ที่มี Top Bar นอกจากหน้ารายงานข้อมูลทางสถิติของธุรกรรมทางการเงิน (Report Screen) ก่อน Notification Modal จึงจะปรากฏ โดยใน Modal นี้จะเป็น Modal สำหรับการแจ้งเตือน และยืนยันการเป็นหนี้สินหรือการเข้าร่วมกลุ่มทางการเงินของผู้ใช้แอปพลิเคชัน

หมายเลข 1: Modal Component เป็น Container ที่ใช้สำหรับแสดงข้อมูลที่จะให้แสดงผลหรือไม่แสดงผลก็ได้ โดยการตั้งค่า Properties เหมือน Pop up ที่จะปรากฏขึ้นเมื่อผู้ใช้กดที่อะไรบางอย่าง แต่มีความแตกต่างจากการเปลี่ยนหน้าจอโทรศัพท์ของผู้ใช้แอปพลิเคชันในจุดที่ Modal นั้นเมื่อถูกเรียกใช้จะยังคงอยู่ในหน้าเดิมที่ถูกเรียกใช้เสมอ

หมายเลข 2: TouchableOpacity Component ที่กำหนดให้เป็นไอคอนกากบาท เมื่อกดแล้ว Modal Component นี้จะไม่แสดงผลอีกต่อไป

หมายเลข 3: Text Component ที่มีการตั้งข้อความว่า “Notification” เพื่อบ่งบอกถึงหน้าจอที่กำลังแสดงว่าปัจจุบันอยู่ที่ Notification Modal

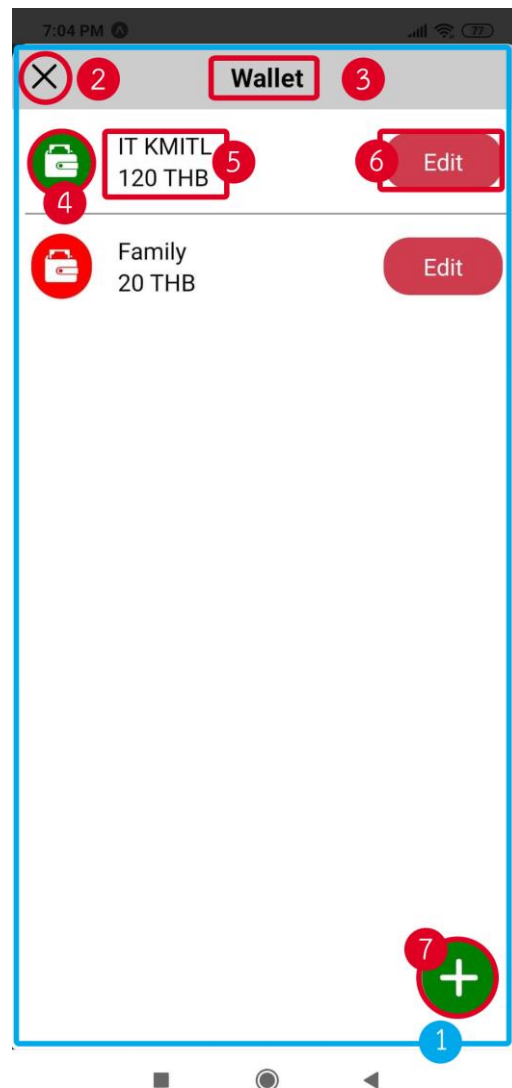
หมายเลข 4: Avatar Component ที่ import มาจาก react-native-elements ซึ่งใช้สำหรับแสดงรูปภาพประจำตัวของผู้ใช้แอปพลิเคชัน สามารถกำหนดรูปแบบได้หลากหลาย

หมายเลข 5: Text Component ที่มีการตั้งข้อความตามการแจ้งเตือนที่ผู้ใช้แอปพลิเคชันได้รับ

หมายเลข 6: TouchableOpacity Component ที่มีการกำหนดรูปแบบให้เป็นปุ่มสำหรับยืนยันการแจ้งเตือน เมื่อผู้ใช้แอปพลิเคชันกดจะทำการยืนยันการเป็นหนี้สินหรือการเข้าร่วมกลุ่มทางการเงิน

หมายเลข 7: TouchableOpacity Component ที่มีการกำหนดรูปแบบให้เป็นปุ่มสำหรับปฏิเสธการแจ้งเตือน เมื่อผู้ใช้แอปพลิเคชันกดจะทำการปฏิเสธการเป็นหนี้สินหรือการเข้าร่วมกลุ่มทางการเงิน

Wallet List Modal



รูปภาพที่ 20 Wallet List Modal

ในการที่จะเข้าสู่ Wallet List Modal ได้นั้น ผู้ใช้แอปพลิเคชันจำเป็นต้องกดที่ไอคอนกระเป๋าเงิน ภายในหน้าใดก็ได้ที่มี Top Bar ก่อน Wallet List Modal จึงจะปรากฏ โดยใน Modal นี้จะเป็น Modal สำหรับการเปลี่ยน แก้ไข และเพิ่มกลุ่มทางการเงินใหม่ของผู้ใช้แอปพลิเคชัน

หมายเลข 1: Modal Component เป็น Container ที่ใช้สำหรับแสดงข้อมูลที่จะให้แสดงผลหรือไม่แสดงผลก็ได้ โดยการตั้งค่า Properties เหมือน Pop up ที่จะปรากฏขึ้นเมื่อผู้ใช้กดที่อะไรบางอย่าง แต่มีความแตกต่างจากการเปลี่ยนหน้าจอโทรศัพท์ของผู้ใช้แอปพลิเคชันในจุดที่ Modal นั้นเมื่อถูกเรียกใช้จะยังคงอยู่ในหน้าเดิมที่ถูกเรียกใช้เสมอ

หมายเลข 2: TouchableOpacity Component ที่กำหนดให้เป็นไอคอนกากบาท เมื่อกดแล้ว Modal Component นี้จะไม่แสดงผลอีกต่อไป

หมายเลข 3: Text Component ที่มีการตั้งข้อความว่า “Wallet” เพื่อบ่งบอกถึงหน้าจอที่กำลังแสดงว่าปัจจุบันอยู่ที่ Wallet List Modal

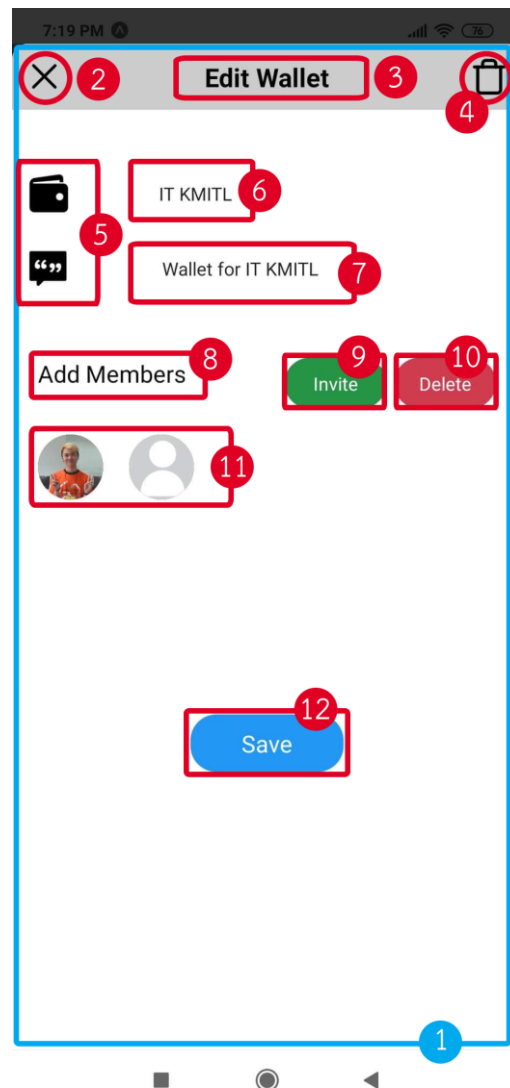
หมายเลข 4: Avatar Component ที่ import มาจาก react-native-elements ซึ่งใช้สำหรับแสดงรูปภาพประจำตัวของผู้ใช้แอปพลิเคชัน ซึ่งสามารถกำหนดรูปแบบได้หลากหลาย

หมายเลข 5: Text Component ที่มีการตั้งข้อความตามข้อมูลของกลุ่มทางการเงิน

หมายเลข 6: TouchableOpacity Component ที่มีการกำหนดรูปแบบให้เป็นปุ่มสำหรับแก้ไขรายละเอียดกลุ่มทางการเงิน เมื่อผู้ใช้แอปพลิเคชันกดจะมี Edit Wallet Detail Modal ปรากฏขึ้นมาบนหน้าจอโทรศัพท์ของผู้ใช้แอปพลิเคชัน ดังรูปภาพที่ 21

หมายเลข 7: TouchableOpacity Component ที่มีการกำหนดรูปแบบให้เป็นปุ่มสำหรับเพิ่มกลุ่มทางการเงิน เมื่อผู้ใช้แอปพลิเคชันกดจะมี Create Wallet Modal ปรากฏขึ้นมาบนหน้าจอโทรศัพท์ของผู้ใช้แอปพลิเคชัน ดังรูปภาพที่ 23

Edit Wallet Detail Modal



รูปภาพที่ 21 Edit Wallet Detail Modal

ในการที่จะเข้าสู่ Edit Wallet Detail Modal ได้นั้น ผู้ใช้แอปพลิเคชันจำเป็นต้องกดที่หมายเลข 6 ในรูปภาพที่ 20 ก่อน Edit Wallet Detail Modal จึงจะปรากฏ โดยใน Modal นี้จะเป็น Modal สำหรับการแก้ไขรายละเอียดของกลุ่มทางการเงินที่ผู้ใช้แอปพลิเคชันรายนั้น ๆ อยู่

หมายเลข 1: Modal Component เป็น Container ที่ใช้สำหรับแสดงข้อมูลที่จะให้แสดงผลหรือไม่แสดงผลก็ได้ โดยการตั้งค่า Properties เหมือน Pop up ที่จะปรากฏขึ้นเมื่อผู้ใช้กดที่อะไรบางอย่าง แต่มีความแตกต่างจากการเปลี่ยนหน้าจอโทรศัพท์ของผู้ใช้แอปพลิเคชันในจุดที่ Modal นั้นเมื่อถูกเรียกใช้จะยังคงอยู่ในหน้าเดิมที่ถูกเรียกใช้เสมอ

หมายเลข 2: TouchableOpacity Component ที่กำหนดให้เป็นไอคอนกากบาท เมื่อกดแล้ว Modal Component นี้จะไม่แสดงผลอีกต่อไป

หมายเลข 3: Text Component ที่มีการตั้งข้อความว่า “Edit Wallet” เพื่อบ่งบอกถึงหน้าจอที่กำลังแสดงว่าปัจจุบันอยู่ที่ Edit Wallet Detail Modal

หมายเลข 4: TouchableOpacity Component ที่กำหนดให้เป็นไอคอนถังขยะ เมื่อกดแล้วจะมี Alert แจ้งเตือนขึ้นมาให้ผู้ใช้แอปพลิเคชันยืนยันว่าต้องการลบกลุ่มทางการเงินนี้หรือไม่

หมายเลข 5: Icon ซึ่งเป็น Component ที่ import มาจาก @expo/vector-icons ใช้สำหรับการแสดงผลให้สวยงาม หรือทำให้ผู้ใช้แอปพลิเคชันเข้าใจได้มากขึ้น

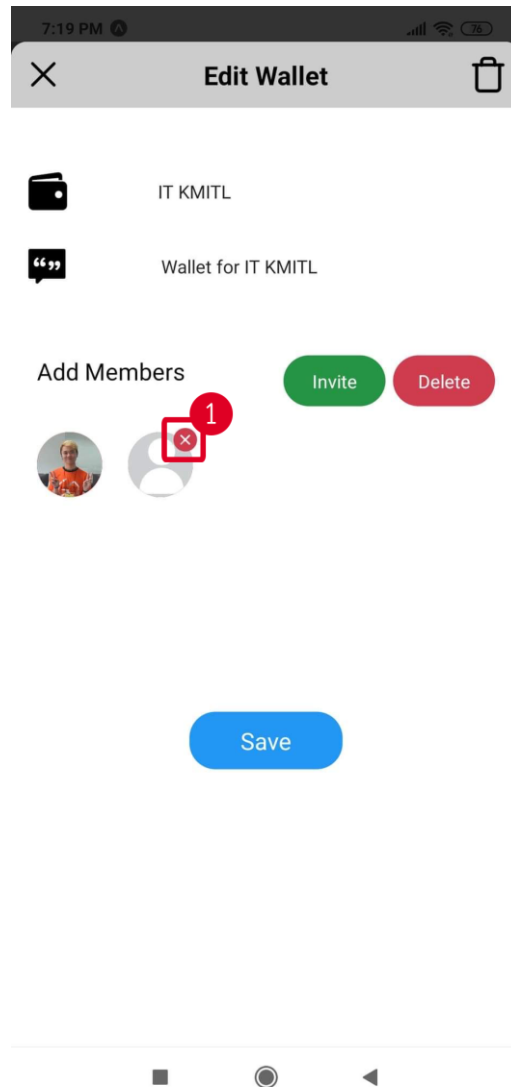
หมายเลข 6: TextInput Component ที่รับค่าจากแป้นพิมพ์ ซึ่งใช้สำหรับเปลี่ยนแปลงหรือแก้ไขชื่อของกลุ่มทางการเงินที่ผู้ใช้แอปพลิเคชันรายนั้น ๆ เป็นผู้สร้างขึ้น

หมายเลข 7: TextInput Component ที่รับค่าจากแป้นพิมพ์ ซึ่งใช้สำหรับเปลี่ยนแปลงหรือแก้ไขรายละเอียดของกลุ่มทางการเงินที่ผู้ใช้แอปพลิเคชันรายนั้น ๆ เป็นผู้สร้างขึ้น

หมายเลข 8: Text Component ที่มีการตั้งข้อความว่า “Add Members” เพื่อบ่งบอกให้ผู้ใช้แอปพลิเคชันเข้าใจว่าส่วนนี้คือส่วนที่เกี่ยวข้องกับการเพิ่มหรือลบสมาชิกในกลุ่มทางการเงิน

หมายเลข 9: TouchableOpacity Component ที่มีการกำหนดรูปแบบให้เป็นปุ่มสำหรับเพิ่มสมาชิกในกลุ่มทางการเงิน เมื่อผู้ใช้แอปพลิเคชันกดจะมี Add Wallet Member Modal ปรากฏขึ้นมาบนหน้าจอโทรศัพท์ของผู้ใช้แอปพลิเคชัน ดังรูปภาพที่ 24

หมายเลข 10: TouchableOpacity Component ที่มีการกำหนดรูปแบบให้เป็นปุ่มสำหรับลบสมาชิกในกลุ่มทางการเงิน เมื่อผู้ใช้แอปพลิเคชันกดจะมีเครื่องหมายกากบาทปรากฏขึ้นมาบนรูปภาพประจำตัวของผู้ใช้แอปพลิเคชันรายอื่น ๆ ที่อยู่ในกลุ่มทางการเงินนั้น ๆ ดังรูปภาพที่ 22

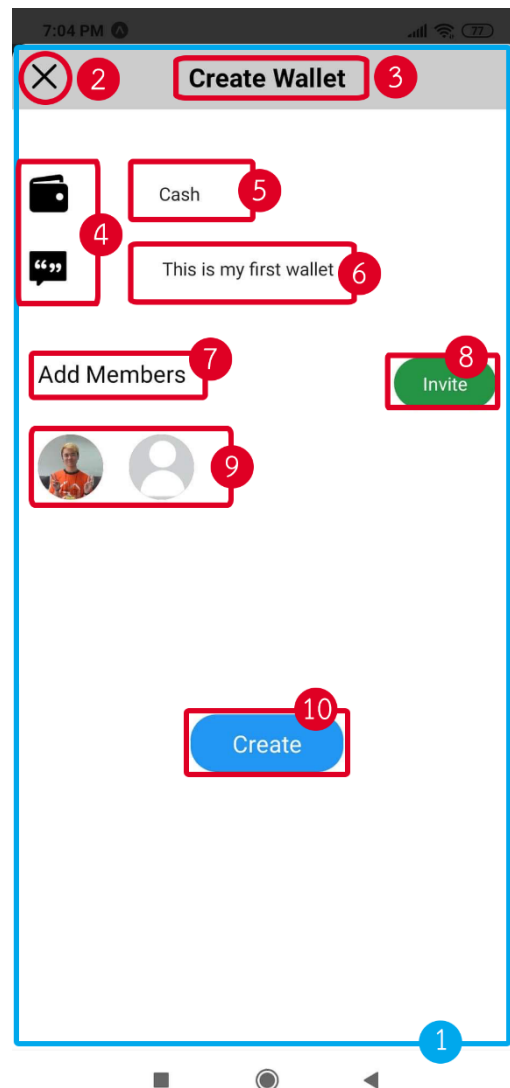


รูปภาพที่ 22 Delete Members from Edit Wallet Detail Modal

หมายเลข 11: Avatar Component ที่ import มาจาก react-native-elements ซึ่งใช้สำหรับแสดงรูปภาพประจำตัวของผู้ใช้แอปพลิเคชันรายอื่น ๆ ที่อยู่ในกลุ่มทางการเงินนั้น ๆ สามารถกำหนดรูปแบบได้หลากหลาย

หมายเลข 12: TouchableOpacity Component ที่มีการกำหนดรูปแบบให้เป็นปุ่มสำหรับบันทึกการแก้ไขรายละเอียดกลุ่มทางการเงินนั้น เมื่อผู้ใช้แอปพลิเคชันกดจะทำการบันทึกการแก้ไขทั้งหมดที่เกิดขึ้น

Create Wallet Modal



รูปภาพที่ 23 Create Wallet Modal

ในการที่จะเข้าสู่ Create Wallet Modal ได้นั้น ผู้ใช้แอปพลิเคชันจำเป็นต้องกดที่หมายเลข 7 ในรูปภาพที่ 20 ก่อน Create Wallet Modal จึงจะปรากฏ โดยใน Modal นี้จะเป็น Modal สำหรับการสร้างกลุ่มทางการเงินใหม่

หมายเลข 1: Modal Component เป็น Container ที่ใช้สำหรับแสดงข้อมูลที่จะให้แสดงผลหรือไม่แสดงผลก็ได้ โดยการตั้งค่า Properties เหมือน Pop up ที่จะปรากฏขึ้นเมื่อผู้ใช้กดที่อะไรบางอย่าง แต่มีความแตกต่างจากการเปลี่ยนหน้าจอ โทรศัพท์ของผู้ใช้แอปพลิเคชันในจุดที่ Modal นั้นเมื่อถูกเรียกใช้จะยังคงอยู่ในหน้าเดิมที่ถูกเรียกใช้เสมอ

หมายเลข 2: TouchableOpacity Component ที่กำหนดให้เป็นไอคอนกากบาท เมื่อกดแล้ว Modal Component นี้จะไม่แสดงผลอีกต่อไป

หมายเลข 3: Text Component ที่มีการตั้งข้อความว่า “Create Wallet” เพื่อบ่งบอกถึงหน้าจอที่กำลังแสดงว่าปัจจุบันอยู่ที่ Create Wallet Modal

หมายเลข 4: Icon ซึ่งเป็น Component ที่ import มาจาก @expo/vector-icons ใช้สำหรับการแสดงผลให้สวยงาม หรือทำให้ผู้ใช้แอปพลิเคชันเข้าใจได้มากขึ้น

หมายเลข 5: TextInput Component ที่รับค่าจากแป้นพิมพ์ ซึ่งใช้สำหรับกำหนดชื่อของกลุ่มทางการเงินที่ผู้ใช้แอปพลิเคชันต้องการจะสร้างขึ้นใหม่

หมายเลข 6: TextInput Component ที่รับค่าจากแป้นพิมพ์ ซึ่งใช้สำหรับกำหนดรายละเอียดของกลุ่มทางการเงินที่ผู้ใช้แอปพลิเคชันต้องการจะสร้างขึ้นใหม่

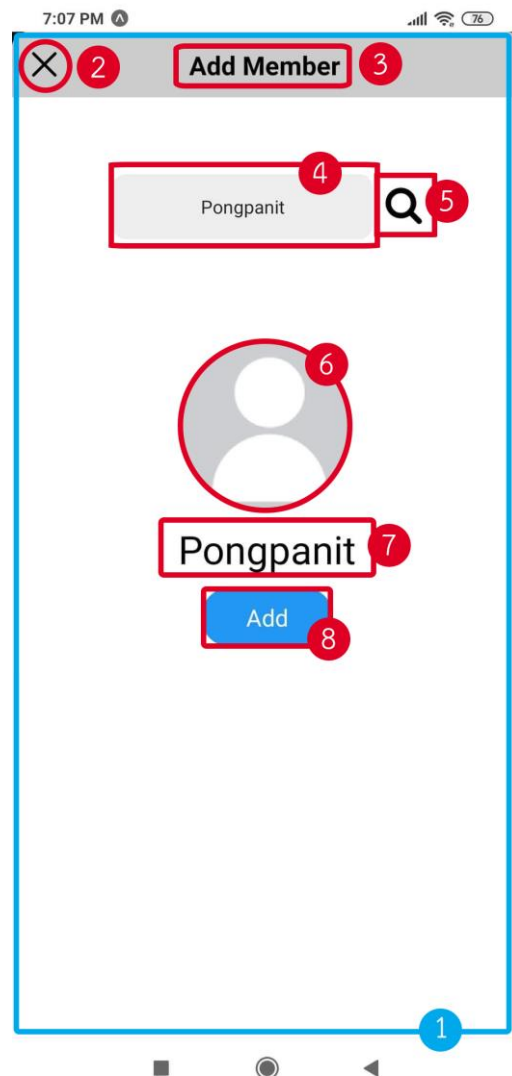
หมายเลข 7: Text Component ที่มีการตั้งข้อความว่า “Add Members” เพื่อบ่งบอกให้ผู้ใช้แอปพลิเคชันเข้าใจว่าส่วนนี้คือส่วนที่เกี่ยวข้องกับการเพิ่มสมาชิกในกลุ่มทางการเงิน

หมายเลข 8: TouchableOpacity Component ที่มีการกำหนดรูปแบบให้เป็นปุ่มสำหรับเพิ่มสมาชิกในกลุ่มทางการเงิน เมื่อผู้ใช้แอปพลิเคชันกดจะมี Add Wallet Member Modal ปรากฏขึ้นมาบนหน้าจอโทรศัพท์ของผู้ใช้แอปพลิเคชัน ดังรูปภาพที่ 24

หมายเลข 9: Avatar Component ที่ import มาจาก react-native-elements ซึ่งใช้สำหรับแสดงรูปภาพประจำตัวของผู้ใช้แอปพลิเคชันรายอื่น ๆ ที่ต้องการจะเพิ่มเข้ากลุ่มทางการเงินนั้น ๆ สามารถกำหนดรูปแบบได้หลากหลาย

หมายเลข 10: TouchableOpacity Component ที่มีการกำหนดรูปแบบให้เป็นปุ่มสำหรับสร้างกลุ่มทางการเงิน เมื่อผู้ใช้แอปพลิเคชันกดจะทำการสร้างกลุ่มทางการเงินใหม่โดยอาศัยข้อมูลที่ผู้ใช้แอปพลิเคชันกรอกเข้ามา

Add Wallet Member Modal



รูปภาพที่ 24 Add Wallet Member Modal

ในการที่จะเข้าสู่ Add Wallet Member Modal ได้นั้น ผู้ใช้แอปพลิเคชันจำเป็นต้องกดที่หมายเลข 9 ในรูปภาพที่ 21 หรือหมายเลข 8 ในรูปภาพที่ 23 ก่อน Add Wallet Member Modal จึงจะปรากฏ โดยใน Modal นี้จะเป็น Modal สำหรับการเพิ่มผู้ใช้แอปพลิเคชันรายอื่น ๆ เข้ากลุ่มทางการเงิน

หมายเลข 1: Modal Component เป็น Container ที่ใช้สำหรับแสดงข้อมูลที่จะให้แสดงผลหรือไม่แสดงผลก็ได้ โดยการตั้งค่า Properties เหมือน Pop up ที่จะปรากฏขึ้นเมื่อผู้ใช้กดที่อะไรบางอย่าง แต่มีความแตกต่างจากการเปลี่ยนหน้าจอ โทรศัพท์ของผู้ใช้แอปพลิเคชันในจุดที่ Modal นั้นเมื่อถูกเรียกใช้จะยังคงอยู่ในหน้าเดิมที่ถูกเรียกใช้เสมอ

หมายเลข 2: TouchableOpacity Component ที่กำหนดให้เป็นไอคอนกากบาท เมื่อกดแล้ว Modal Component นี้จะไม่แสดงผลอีกต่อไป

หมายเลข 3: Text Component ที่มีการตั้งข้อความว่า “Add Member” เพื่อบ่งบอกถึงหน้าจอที่กำลังแสดงว่าปัจจุบันอยู่ที่ Add Wallet Member Modal

หมายเลข 4: TextInput Component ที่รับค่าจากแป้นพิมพ์ ซึ่งใช้สำหรับการค้นหาผู้ใช้แอปพลิเคชันรายอื่น ๆ จากชื่อผู้ใช้งาน

หมายเลข 5: Icon ซึ่งเป็น Component ที่ import มาจาก @expo/vector-icons ใช้สำหรับการแสดงผลให้สวยงาม หรือทำให้ผู้ใช้แอปพลิเคชันเข้าใจได้มากขึ้น

หมายเลข 6: Avatar Component ที่ import มาจาก react-native-elements ซึ่งใช้สำหรับแสดงรูปภาพประจำตัวของผู้ใช้แอปพลิเคชันที่ค้นพบ สามารถกำหนดรูปแบบได้หลากหลาย

หมายเลข 7: Text Component ที่มีการตั้งข้อความตามชื่อผู้ใช้งานของผู้ใช้แอปพลิเคชันรายอื่น ๆ ที่ค้นหา เพื่อบ่งบอกถึงตัวตนของผู้ใช้แอปพลิเคชันรายนั้น ๆ

หมายเลข 8: TouchableOpacity Component ที่มีการกำหนดรูปแบบให้เป็นปุ่มสำหรับเพิ่มสมาชิกในกลุ่มทางการเงิน เมื่อผู้ใช้แอปพลิเคชันกดจะทำการเพิ่มผู้ใช้แอปพลิเคชันรายนั้น ๆ เข้ากลุ่มทางการเงิน

Create Modal



รูปภาพที่ 25 Create Modal

ในการที่จะเข้าสู่ Create Modal ได้นั้น ผู้ใช้แอปพลิเคชันจำเป็นต้องกดที่ไอคอนเครื่องหมายบวก ภายในหน้าใดก็ได้ ที่มี Bottom Bar ก่อน Create Modal จึงจะปรากฏ โดยใน Modal นี้จะเป็น Modal สำหรับการเพิ่มธุรกรรมทางการเงิน หรือหนี้สินใหม่ของผู้ใช้แอปพลิเคชัน

หมายเลข 1: Modal Component เป็น Container ที่ใช้สำหรับแสดงข้อมูลที่จะให้แสดงผลหรือไม่แสดงผลก็ได้ โดยการตั้งค่า Properties เสมือน Pop up ที่จะปรากฏขึ้นเมื่อผู้ใช้กดที่อะไรบางอย่าง แต่มีความแตกต่างจากการเปลี่ยนหน้าจอ โทรศัพท์ของผู้ใช้แอปพลิเคชันในจุดที่ Modal นั้นเมื่อถูกเรียกใช้จะยังคงอยู่ในหน้าเดิมที่ถูกเรียกใช้เสมอ

หมายเลข 2: TouchableOpacity Component ที่กำหนดให้เป็นไอคอนกากบาท เมื่อกดแล้ว Modal Component นี้จะไม่แสดงผลอีกต่อไป

หมายเลข 3: Text Component ที่มีการตั้งข้อความว่า “Create” เพื่อบ่งบอกถึงหน้าจอที่กำลังแสดงว่าปัจจุบันอยู่ที่ Create Modal

หมายเลข 4: Icon ซึ่งเป็น Component ที่ import มาจาก @expo/vector-icons ใช้สำหรับการแสดงผลให้สวยงาม หรือทำให้ผู้ใช้แอปพลิเคชันเข้าใจได้มากขึ้น

หมายเลข 5: Picker Component ที่ import มาจาก @react-native-community/picker ซึ่งใช้สำหรับเลือกประเภทของธุรกรรมทางการเงินหรือหนี้สินนั้น ๆ เหมือนกับ Select Tag ของภาษา HTML เพียงแต่จะไม่แสดงในลักษณะ Drop down list แต่ปรากฏรายการให้ผู้ใช้แอปพลิเคชันเลือกขึ้นบนหน้าจอโทรศัพท์แทน

หมายเลข 6: TextInput Component ที่กำหนดประเภทของแป้นพิมพ์ให้เป็นประเภทที่พิมพ์ได้แค่ตัวเลขเท่านั้น (keyboardType=“numeric”) ซึ่งใช้สำหรับการพิมพ์จำนวนเงินของรายรับหรือรายจ่ายในธุรกรรมทางการเงินหรือหนี้สินนั้น ๆ

หมายเลข 7: Text Component ที่มีการตั้งข้อความว่า “THB” เพื่อบ่งบอกให้ผู้ใช้แอปพลิเคชันเข้าใจว่าหน่วยเงินที่ผู้ใช้แอปพลิเคชันระบุนั้น คือ บาท

หมายเลข 8: DatePicker Component ที่ import มาจาก react-native-datepicker ซึ่งใช้สำหรับการกำหนดวันที่ผู้ใช้แอปพลิเคชันทำธุรกรรมทางการเงินหรือหนี้สินนั้น โดยที่เราสามารถกำหนด Properties ได้หลากหลายขึ้นอยู่กับการใช้งาน เช่น รูปแบบของการแสดง วันในอดีตที่สามารถเลือกได้ หรือวันในอนาคตที่สามารถเลือกได้

หมายเลข 9: TextInput Component ที่รับค่าจากแป้นพิมพ์ ในกรณีที่ผู้ใช้แอปพลิเคชันต้องการจดบันทึกเกี่ยวกับรายการธุรกรรมทางการเงินหรือหนี้สินนั้น ๆ เช่น ธุรกรรมทางการเงินนี้ใช้ทำอะไร เป็นต้น

หมายเลข 10: Picker Component ที่ import มาจาก @react-native-community/picker ซึ่งใช้สำหรับเลือกหมวดหมู่ของธุรกรรมทางการเงินหรือหนี้สินนั้น ๆ เหมือนกับ Select Tag ของภาษา HTML เพียงแต่จะไม่แสดงในลักษณะ Drop down list แต่ปรากฏรายการให้ผู้ใช้แอปพลิเคชันเลือกขึ้นบนหน้าจอโทรศัพท์แทน

หมายเลข 11: TouchableOpacity Component ที่มีการกำหนดรูปแบบให้เป็นปุ่มสำหรับสร้างธุรกรรมทางการเงินหรือหนี้สิน เมื่อผู้ใช้แอปพลิเคชันกดจะทำการสร้างกลุ่มธุรกรรมทางการเงินหรือหนี้สินใหม่โดยอาศัยข้อมูลที่ผู้ใช้แอปพลิเคชันกรอกเข้ามา

การเปลี่ยนแปลงแนวคิดจากแบบเสนอหัวข้อ

1. เพิ่มแนวคิดระบบ “หนี้สิน” กล่าวคือ ผู้ใช้แอปพลิเคชันสามารถสร้างรายการหนี้สินเพื่อแจ้งเตือนรายจ่ายของตนเองได้ โดยมีสองประเภท คือ หนี้สินของตนเอง กับ หนี้สินที่ผู้ใช้แอปพลิเคชันรายอื่นติดค้าง
 - หนี้สินของตนเอง เป็นการแจ้งเตือนตามปกติ
 - หนี้สินที่ผู้ใช้แอปพลิเคชันรายอื่นติดค้าง ผู้ใช้แอปพลิเคชันรายอื่นที่ติดค้างจะได้รับข้อความเพื่อยืนยันจำนวนเงินที่ติดค้าง เมื่อผู้ใช้แอปพลิเคชันรายนั้น ๆ ทำการยืนยันแล้ว รายการหนี้สินที่ถูกยืนยันจะถูกเพิ่มไปยัง My debtor ของเจ้าหนี้ และ Debt ของลูกหนี้ โดยลูกหนี้สามารถดูรายละเอียดของหนี้สินได้ในส่วนเจ้าหนี้สามารถแจ้งเตือนลูกหนี้ได้ผ่านแอปพลิเคชัน
2. เปลี่ยนแปลงหลักการทำงานของโครงการจากแบบเสนอหัวข้อ ดังรูปภาพที่ 1