

โครงการพัฒนาเว็บแอปพลิเคชัน

แคมเปญบริจาคเพื่อการกุศล

ข้อกำหนดความต้องการของระบบ

Version: 0.0.5

Date: 12/02/2023

By

กลุ่มที่ 3

นายเฉลิมชัย กำลังเดช 6320502363

นายพีรวัส วงศ์เมฆ 6320501596

นายชนกานต์ ศรีศรีติพร 6320502371

นายธิตินันท์ สว่างศรี 6320501588

นายกฤษกรณ์ ขวัญยืน 6020551750

นายนิภัทร์ ใจซื่อ 6120501720

สารบัญ

Contents

1. บทนำ	4
1.1 วัตถุประสงค์	4
1.2 หลักการและเหตุผล	4
1.3 นิยามศัพท์และตัวย่อ	5
1.4 เอกสารอ้างอิง	5
1.5 ภาพรวม	5
2. คุณลักษณะทั่วไป.....	6
2.1 ภาพรวมระบบ	6
2.2 คุณลักษณะของผู้ใช้.....	6
2.3 คุณสมบัติของระบบ	7
2.4 ข้อจำกัด.....	7
3. ข้อกำหนดความต้องการ	8
3.1 ข้อกำหนดส่วนต่อประสานภายนอก.....	8
3.1.1 ส่วนต่อประสานกับผู้ใช้.....	8
3.1.2 ส่วนต่อประสานของฮาร์ดแวร์.....	8
3.1.3 ส่วนต่อประสานของซอฟต์แวร์.....	8

3.1.4 ส่วนต่อประสานทางการสื่อสาร	9
3.2 ความต้องการแบบ functional	9
3.3 ความต้องการแบบ Non-functional	11
3.4 ความต้องการอื่นๆ	12
4. แผนภาพการวิเคราะห์ระบบ	12
5. ภาคผนวก	13

Revision History

Version Number	จัดทำโดย	Revision Date	การเปลี่ยนแปลง
0.0.1	คณะทีมงาน	07/02/2023	เริ่ม
0.0.5	คณะทีมงาน	12/02/2023	อัปเดตข้อมูลในหัวข้อ ต่างๆ

1. บทนำ

1.1 วัตถุประสงค์

1.1.2. เอกสารฉบับนี้จัดทำขึ้นเพื่อเป็นข้อกำหนดความต้องการของระบบให้กับโครงการ

1.2.2. เอกสารฉบับนี้นั้นจัดทำขึ้นให้กับ นักพัฒนา และผู้บริหารโครงการได้เข้าใจถึงความต้องการของระบบในรูปแบบต่าง

1.2 หลักการและเหตุผล

การบริจาค้นั้นถือเป็นหนึ่งในประเพณีการทำบุญอย่างหนึ่งที่ซึ่งอยู่คู่กับสังคมไทยมาอย่างช้านาน อันซึ่งแสดงถึงน้ำใจอันประเสริฐของผู้ให้ ไปจนถึงความสุขใจของผู้รับ ในไทยนั้นการบริจาคถือว่าพบเห็นได้ทั่วไปตามองค์กร หรือแคมเปญต่างๆที่จัดขึ้นตามวาระโอกาสต่างๆ ซึ่งในหลายครั้งนั้นการประชาสัมพันธ์ตลอดจนช่องทางในการบริจาค้นั้นไม่ทั่วถึงเท่าที่ควร

ผู้จัดทำเล็งเห็นถึงปัญหาในข้างต้นจึงได้จัดทำโครงการการพัฒนาเว็บแอปพลิเคชันแคมเปญบริจาคเพื่อการกุศล ที่ซึ่งจะเป็นแพลตฟอร์มออนไลน์ที่ให้ผู้บริจาคและผู้เปิดรับบริจาค้นั้นสามารถสื่อสารกันได้อย่างรวดเร็ว อันช่วยกันแก้ไขปัญหาการขาดช่องทางการสื่อสาร ตลอดไปจนถึงผู้บริจาค้นั้นสามารถบริจาคได้อย่างสะดวกและรวดเร็ว โดยเว็บแอปพลิเคชันนั้นจะทำงานผ่านระบบอินเทอร์เน็ต ผู้ใช้สามารถสมัครสมาชิกและเข้าสู่ระบบได้ทั้งในฐานะผู้บริจาคหรือผู้เปิดแคมเปญบริจาค โดยมีระบบ story ของแคมเปญต่างๆที่ซึ่งสามารถให้ผู้บริจาค้นั้นติดตามความเคลื่อนไหวของแคมเปญได้ผ่านทางหน้าเว็บไซต์ตลอดไปจนถึงการร่วมบริจาคสมทบทุนให้แคมเปญได้ ซึ่งช่วยให้ทั้งผู้บริจาคและผู้รับบริจาคเกิดความสะดวกสบายและส่งเสริมกิจกรรมนี้ให้คงอยู่สืบต่อไป

1.3 นิยามศัพท์และตัวย่อ

- MVC คือ Model View Controller ใช้เรียกรูปแบบการพัฒนาซอฟต์แวร์ที่มีโครงสร้างซึ่งแบ่งออกมา
- Framework คือ ชุดคำสั่ง เครื่องมือ หรือโครงสร้างอย่างใดอย่างหนึ่ง ที่สร้างขึ้นมาเพื่ออำนวยความสะดวกแก่โปรแกรมเมอร์และนักพัฒนา
- Web Server คือซอฟต์แวร์และฮาร์ดแวร์ของคอมพิวเตอร์ที่สามารถรับคำขอผ่านทาง HTTP ซึ่งเป็นโปรโตคอลเครือข่ายที่สร้างขึ้นเพื่อเผยแพร่เนื้อหาของเว็บ
- Front - End คือ ส่วนที่สามารถมองเห็นได้ของเว็บไซต์หรือแอปพลิเคชัน
- Back - End คือ ส่วนของการทำงานเบื้องหลังจำพวก ฐานข้อมูลและโครงสร้างพื้นฐาน
- JSON คือ ฟอแมตสำหรับแลกเปลี่ยนข้อมูลคอมพิวเตอร์ ฟอแมต JSON นั้นอยู่ในรูปข้อความธรรมดา ที่ทั้งมนุษย์และโปรแกรมคอมพิวเตอร์สามารถอ่านเข้าใจได้
- CRUD คือ แนวคิดพื้นฐานที่ใช้ในการเขียนโปรแกรม หรือเว็บแอปพลิเคชัน เพื่อบ่งบอกลักษณะการกระทำ (เพิ่ม อ่าน แก้ไข ลบ) ข้อมูลนั่นเอง

1.4 เอกสารอ้างอิง

-

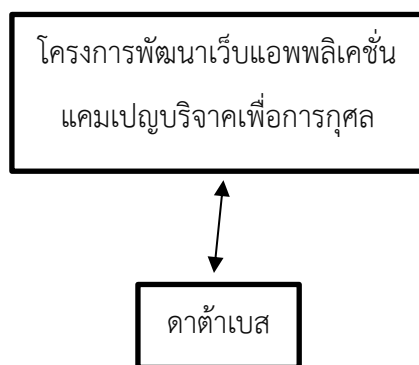
1.5 ภาพรวม

1.5.1 เอกสารประกอบไปด้วย 5 หัวข้อใหญ่

- 1.5.1.1 หัวข้อ บทนำ เหมาะสำหรับผู้ใช้และผู้บริหารโครงการ
- 1.5.1.2 หัวข้อ คุณลักษณะทั่วไป เหมาะสำหรับผู้พัฒนาและผู้บริหารโครงการ
- 1.5.1.3 หัวข้อ ข้อกำหนดความต้องการ เหมาะสำหรับผู้พัฒนา
- 1.5.1.4 หัวข้อ แผนภาพวิเคราะห์ระบบ เหมาะสำหรับผู้พัฒนาและผู้บริหารโครงการ
- 1.5.1.4 หัวข้อ ภาคผนวก เหมาะสำหรับผู้บริหารโครงการ

2. คุณลักษณะทั่วไป

2.1 ภาพรวมระบบ



2.2 คุณลักษณะของผู้ใช้

a. ผู้ใช้ทั่วไป

คุณสมบัติ : เป็นผู้มีความคุ้นเคยกับการใช้เว็บแอปพลิเคชันบนพีซีหรือแล็ปท็อป โดยมีเป้าหมายในการบริจาคเงินเพื่อการกุศลต่างๆ

b. ผู้เปิดแคมเปญ

คุณสมบัติ : เป็นผู้มีความคุ้นเคยกับการใช้เว็บแอปพลิเคชันบนพีซีหรือแล็ปท็อป โดยมีเป้าหมายในการเปิดแคมเปญรับบริจาคเงินเพื่อการกุศลต่างๆ

c. ผู้ดูแลระบบ

คุณสมบัติ : จบการศึกษาระดับปริญญาตรีหรือสายอาชีพที่มีความเกี่ยวข้องกับระบบคอมพิวเตอร์ สามารถเขียนโค้ดในเบื้องต้นได้ สามารถตรวจสอบข้อมูลเอกสารราชการหรือที่เกี่ยวข้องได้ สามารถที่จะเข้ามาอัปเดตระบบได้ทุกวัน

2.3 คุณสมบัติของระบบ

- a. ผู้ใช้ทั่วไปสามารถดูรายการแคมเปญได้
- b. ผู้ใช้ทั่วไปสามารถบริจาคเงินให้กับแคมเปญได้
- c. ผู้ใช้ทั่วไปสามารถติดตามความคืบหน้าของแคมเปญได้
- d. ผู้เปิดแคมเปญสามารถเปิดแคมเปญรับบริจาคได้
- e. ผู้เปิดแคมเปญสามารถอัปเดตความคืบหน้าของแคมเปญที่ตนเปิดได้
- f. ผู้เปิดแคมเปญสามารถดูรายละเอียดของผู้ให้บริจาคได้ เช่น ชื่อ หรือยอดเงิน
- g. ระบบสามารถตรวจสอบการชำระเงินบริจาคได้

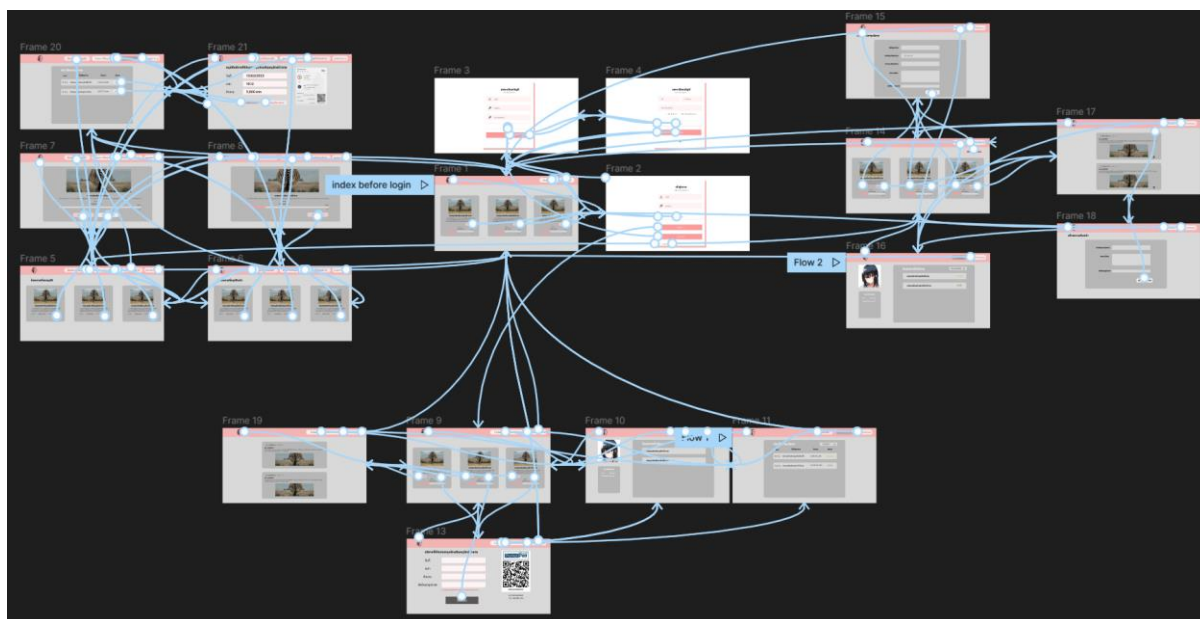
2.4 ข้อจำกัด

- a. ระบบรองรับภาษาไทยเต็มรูปแบบสำหรับผู้ใช้งานและผู้ดูแลระบบ
- b. ระบบรองรับการใช้งานบนพีซีหรือแล็ปท็อปเต็มรูปแบบ
- c. นโยบายความปลอดภัยที่ทุกแคมเปญจะต้องมีการตรวจสอบโดยผู้ดูแลระบบ

3. ข้อกำหนดความต้องการ

3.1 ข้อกำหนดส่วนต่อประสานภายนอก

3.1.1 ส่วนต่อประสานกับผู้ใช้



3.1.2 ส่วนต่อประสานของฮาร์ดแวร์

เนื่องจากเว็บแอปพลิเคชันไม่มีการใช้ฮาร์ดแวร์ใดๆ จึงไม่มีอินเทอร์เฟซสำหรับฮาร์ดแวร์ และการเชื่อมต่อกับฐานข้อมูลได้รับการจัดการโดยระบบปฏิบัติการเว็บเซิร์ฟเวอร์ อุปกรณ์ที่ต้องใช้จึงมีเพียงแคคอมพิวเตอร์, เซิร์ฟเวอร์สำหรับเว็บไซต์และเซิร์ฟเวอร์สำหรับฐานข้อมูล

3.1.3 ส่วนต่อประสานของซอฟต์แวร์

เว็บแอปพลิเคชันเพื่อสร้างการบริจาค, แสดงผลลัพธ์การบริจาคของผู้ใช้ที่ทำงานร่วมกับฐานข้อมูล เพื่อนำเงินที่รับบริจาคมาใช้ในการทำการกุศลและสามารถอัปเดตกระบวนการทำการกุศลให้กับผู้บริจาคผ่านเว็บแอปพลิเคชันนี้ได้ โดยเว็บแอปพลิเคชันของเราได้ใช้สถาปัตยกรรม MVC ในการพัฒนาดังนั้นจึงเลือกใช้ Laravel framework ในการพัฒนาระบบ โดย Framework นี้นำมาพัฒนาระบบทางด้านหลังบ้านและส่วนหน้าบ้านใช้เป็นภาษา php ในการเขียนและได้ใช้ xampp เป็นโปรแกรมจำลอง Web

Server เพื่อใช้ในการเชื่อมต่อกับฐานข้อมูลและเว็บแอปพลิเคชัน โดยข้อมูลที่จะส่งผ่าน Front-End และ Back-End ซึ่งมีรูปแบบของข้อมูลเป็น JSON เพื่อให้ง่ายต่อการทำ CRUD

3.1.4 ส่วนต่อประสานทางการสื่อสาร

เว็บเราใช้โปรโตคอลในการสร้างเว็บแอปพลิเคชันเป็น โปรโตคอล HTTP เนื่องจากตัว xampp หรือ โปรแกรมจำลองเว็บเซิร์ฟเวอร์ ได้ใช้โปรโตคอลเป็น HTTP แต่ระบบได้มีการป้องกันระดับหนึ่งโดยการใช้ JWT มาช่วยในเรื่องของความปลอดภัย

3.2 ความต้องการแบบ functional

User Class
<p>3.2.1 User Class 1 – ผู้บริจาคทั่วไป</p> <p>3.2.1.1 Functional requirement 1.1</p> <ol style="list-style-type: none"> 1) description : ผู้บริจาคทั่วไปสามารถสร้างบัญชีได้ 2) inputs : ข้อมูลของผู้บริจาคทั่วไป 3) process : ระบบสร้างบัญชีของผู้บริจาคทั่วไป 4) outputs : มีบัญชีของผู้บริจาคทั่วไปในฐานข้อมูล 5) errors handling : - <p>3.2.1.2 Functional requirement 1.2</p> <ol style="list-style-type: none"> 1) description : ผู้บริจาคทั่วไปสามารถบริจาคให้กับโครงการบริจาคได้ 2) inputs : ข้อมูลและหลักฐานการบริจาค 3) process : สร้างคำร้องขออนุมัติการบริจาค 4) outputs : คำร้องขออนุมัติการบริจาค 5) errors handling : - <p>3.2.1.3 Functional requirement 1.3</p> <ol style="list-style-type: none"> 1) description : ผู้บริจาคทั่วไปสามารถดูความคืบหน้าของโครงการบริจาคได้ 2) inputs : ผู้บริจาคทั่วไปเข้าไปยังหน้าอัปเดตความคืบหน้าของโครงการบริจาค 3) process : ค้นหาความคืบหน้าของโครงการบริจาคจากฐานข้อมูล 4) outputs : แสดงผลความคืบหน้าของโครงการบริจาค 5) errors handling : -

3.2.1.4 Functional requirement 1.4

- 1) description : ผู้บริจาคทั่วไปสามารถตรวจสอบสถานะการบริจาคโครงการบริจาคได้
- 2) inputs : ผู้บริจาคทั่วไปเข้าไปยังหน้าตรวจสอบสถานะการบริจาค
- 3) process : แอดมินทำการอนุมัติหรือยกเลิกการบริจาค
- 4) outputs : แสดงผลการอนุมัติหรือยกเลิกการบริจาค
- 5) errors handling : -

3.2.2 User Class 2 – ผู้เปิดโครงการบริจาค

3.2.2.1 Functional requirement 2.1

- 1) description : ผู้เปิดโครงการบริจาคสามารถสร้างบัญชีได้
- 2) inputs : ข้อมูลของผู้เปิดโครงการบริจาค
- 3) process : ระบบสร้างบัญชีของผู้เปิดโครงการบริจาค
- 4) outputs : มีบัญชีของผู้เปิดโครงการบริจาคในฐานข้อมูล
- 5) errors handling : -

3.2.2.2 Functional requirement 2.2

- 1) description : ผู้เปิดโครงการสามารถสร้างโครงการบริจาคได้
- 2) inputs : ข้อมูลการบริจาค เช่น รูปภาพ หัวข้อโครงการ เป็นต้น
- 3) process : ผู้เปิดโครงการสร้างโครงการบริจาค
- 4) outputs : ระบบมีโครงการบริจาคในฐานข้อมูล
- 5) errors handling : -

3.2.2.3 Functional requirement 2.3

- 1) description : ผู้เปิดโครงการสามารถอัปเดตความคืบหน้าโครงการบริจาคได้
- 2) inputs : ข้อมูลการบริจาค เช่น รูปภาพ เนื้อหาของการบริจาคต่างๆ เป็นต้น
- 3) process : ผู้เปิดโครงการอัปเดตความคืบหน้าโครงการบริจาค
- 4) outputs : ระบบมีความคืบหน้าของโครงการในฐานข้อมูล
- 5) errors handling : -

3.2.2.4 Functional requirement 2.4

- 1) description : ผู้เปิดโครงการสามารถตรวจสอบสถานะของโครงการบริจาคว่าถูกอนุมัติหรือไม่
- 2) inputs : โครงการบริจาคที่รออนุมัติ
- 3) process : ทำการอนุมัติหรือยกเลิกโครงการบริจาค
- 4) outputs : สถานะโครงการบริจาค
- 5) errors handling : -

3.2.3 User Class 3 – แอดมิน

3.2.3.1 Functional requirement 3.1

- 1) description : แอดมินสามารถอนุมัติหรือยกเลิกโครงการบริจาคที่รออนุมัติได้
- 2) inputs : สิทธิการเป็นแอดมิน
- 3) process : แอดมินอนุมัติหรือยกเลิกโครงการบริจาคที่รออนุมัติ
- 4) outputs : โครงการบริจาคที่รออนุมัติถูกอนุมัติหรือยกเลิก
- 5) errors handling : -

3.2.3.2 Functional requirement 3.2

- 1) description : แอดมินสามารถยกเลิกโครงการบริจาคที่อนุมัติแล้วได้
- 2) inputs : สิทธิการเป็นแอดมิน
- 3) process : แอดมินยกเลิกโครงการบริจาคที่อนุมัติแล้วได้
- 4) outputs : โครงการบริจาคที่อนุมัติแล้วถูกยกเลิก
- 5) errors handling : -

3.2.3.3 Functional requirement 3.3

- 1) description : แอดมินสามารถอนุมัติหรือยกเลิกการบริจาคของผู้ใช้ทั่วไปที่ได้
- 2) inputs : สิทธิการเป็นแอดมิน
- 3) process : แอดมิน อนุมัติหรือยกเลิกการบริจาคของผู้ใช้ทั่วไป
- 4) outputs : การบริจาคของผู้ใช้ทั่วไปถูกอนุมัติหรือยกเลิก
- 5) errors handling : -

3.3 ความต้องการแบบ Non-functional

3.3.1 Performance: ระบบสามารถรองรับคำขอในการตรวจสอบการโอนเงินพร้อมกันได้มากถึง 600 คำขอ โดยหลังจบแคมเปญที่ร่วมกันบริจาคไปแล้ว ภายใน 1 ปี ระบบจะทำการล้างข้อมูลของแคมเปญนั้นออกไปจากระบบ

3.3.2 Reliability: ระบบไม่ควรเปิดเผยรายละเอียดข้อมูลส่วนตัวของผู้ใช้ระบบแต่จะสามารถเปิดเผยเป็นนามสมมุติแทนได้ หากไม่ได้กรอกก็จะขึ้นชื่อเป็นผู้ไม่ประสงค์ออกนาม หากระบบล่มต้องสามารถกู้คืนได้ภายใน 1 ชั่วโมง

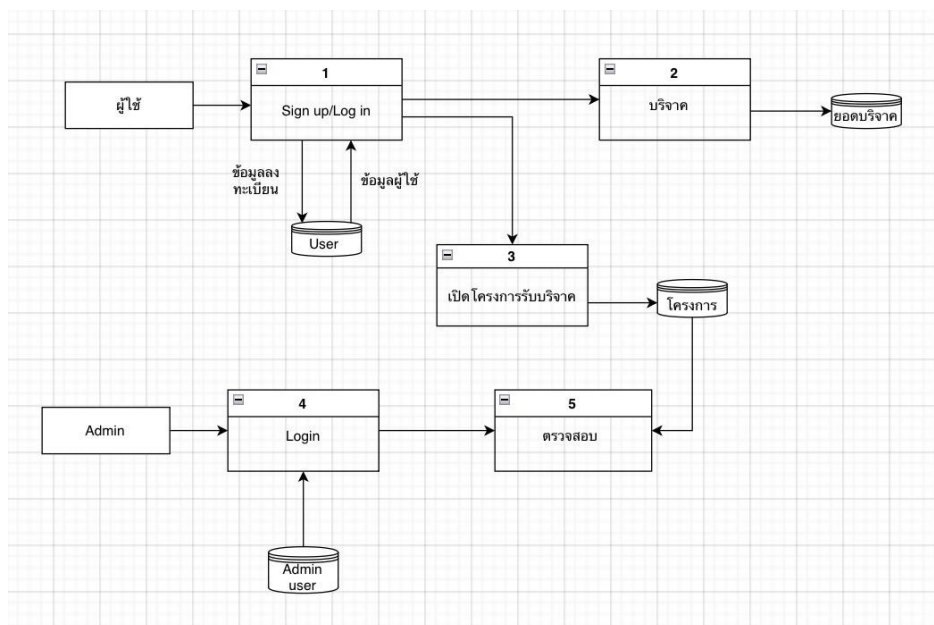
3.3.3 Availability: ระบบการโอนเงินสามารถใช้งานได้ 06:00-24:00น. เนื่องจากว่าในช่วงเวลา 00:00 - 06:00 มักจะเป็นเวลาที่ธนาคารมักจะปิดปรับปรุงโดยใน 1 สัปดาห์ ระบบต้องสามารถใช้งานได้มากถึงร้อยละ 90 ต่อ สัปดาห์

3.4 ความต้องการอื่นๆ

-

4. แผนภาพการวิเคราะห์ระบบ

Data Flow Diagram



5. ภาคผนวก

Gantt Chart แผนการดำเนินงาน

