

หาบ้านให้น้อง

ข้อกำหนดความต้องการของระบบ

Version: 01

Date: 12/02/2566

By

กลุ่ม 8

นางสาวศศิชา สลับแก้ว รหัสนิต 6320502509

นายพชร พึ่งอวยผล รหัสนิต 6120503927

นายกิตติพัทธ์ ช่อฟ้า รหัสนิต 6320502983

นางสาวสุกัลยา ขำเจริญ รหัสนิต 6320503106

นายณเรศ เฟื่องเวโรจน์สกุล รหัสนิต 6320503041

นางสาวอลิสรา กระจ่างแจ่ม รหัสนิต 6320500697

สารบัญ

1. บทนำ

1.1. วัตถุประสงค์

-เพื่อพัฒนาระบบการหาบ้านให้สัตว์ไร้บ้านที่สามารถส่งต่อสัตว์และรับดูแลสัตว์

-เพื่อเป็นอีกทางเลือกให้กับบุคคลที่กำลังหาสัตว์เลี้ยง

-เพื่อลดการนำสัตว์ไปปล่อยในที่ต่างๆ

1.2. หลักการและเหตุผล

เนื่องด้วยปัจจุบันปัญหาการรับเลี้ยงสัตว์และสัตว์ไร้บ้าน

ทำให้เกิดปัญหาการควบคุมประชากรสัตว์ที่ตามมา

อันเนื่องมาจากผู้เลี้ยงสัตว์ไม่สามารถเลี้ยงสัตว์ของตนเองได้ และไม่มีใครมารับช่วงต่อในการรับเลี้ยงสัตว์

นำไปสู่การปล่อยไว้ข้างทางทำให้จำนวนประชากรของสัตว์จรจัดมีจำนวนมากขึ้น

การพัฒนาระบบหาบ้านให้น้องน จึงเป็นส่วนช่วยให้การแก้ไขปัญหาดังกล่าว อาทิเช่น

ผู้เลี้ยงสัตว์มีความจำเป็นไม่สามารถเลี้ยงสัตว์ของตนได้อีกต่อก็สามารถใช้ระบบหาบ้านให้น้องนเพื่อเป็นการหาผู้
รับดูแลสัตว์ที่มีการผ่านการคัดกรองเบื้องต้นมาแล้ว

การใช้งานเบื้องต้นเมื่อผู้ที่จะประกาศหาบ้านให้สัตว์สามารถโพสต์หาบ้านได้ในระบบ

โดยผู้ใช้ที่ต้องการรับสัตว์ไปเลี้ยงสามารถเห็นข้อมูลเบื้องต้นได้ เช่น อายุ พันธุ์สัตว์ โรคประจำตัว

1.3. นิยามศัพท์และตัวย่อ

favorite = บันทึกโพสต์ความชอบ

1.4. เอกสารอ้างอิง

ไม่มี

1.5. ภาพรวม

หัวข้อที่ 2 เหมาะกับบุคคลทั่วไป

หัวข้อที่ 3 เหมาะกับนักพัฒนาเว็บ

2. คุณลักษณะทั่วไป

2.1 ภาพรวมของระบบ

ต้องมีอินเทอร์เน็ต

2.2 คุณลักษณะของผู้ใช้

ผู้ใช้ทั่วไป

- ผู้ใช้ทั่วไปที่อายุมากกว่า 18 ปี ขึ้นไป
- ศูนย์พักพิงสัตว์
- ต้องมีการลงทะเบียนกับระบบเพื่อเป็นuserในการใช้งาน web application
- ต้องการหาสัตว์เลี้ยงหรือส่งต่อสัตว์เลี้ยงกรณีเลี้ยงไม่ไหวหรือพบเจอสัตว์ที่ไม่มีเจ้าของโดยบังเอิญตามสถานที่ต่างๆและอยากช่วยเหลือให้มีที่พิงพิง

ผู้ดูแลระบบ

- ได้รับการอนุญาตจากผู้ดูแลระบบโดยตรง

2.3 คุณสมบัติของระบบ

ผู้ใช้ทั่วไป

- ผู้ใช้สามารถแสดงความคิดเห็นเมื่อสนใจหรือต้องการรับเลี้ยงได้
- ผู้ใช้สามารถโพสต์ส่งต่อสัตว์ของตนได้
- สามารถกดปุ่ม favorite เพื่อเก็บไว้เป็นตัวเลือกในการหาสัตว์เลี้ยง
- ผู้ใช้ที่โพสต์ส่งต่อสัตว์ของตนสามารถเปลี่ยนสถานะยังไม่ถูกรับเลี้ยงเป็นถูกรับเลี้ยงแล้ว
- ผู้ใช้สามารถกดดูหน้าโปรไฟล์ของอื่นแล้วตนเองได้
- ผู้ใช้สามารถเลือกชนิดของสัตว์ที่ต้องการหาได้
- ผู้ใช้สามารถแก้ไขโปรไฟล์ของตัวเองได้
- ผู้ใช้ต้องลงทะเบียนก่อนเข้าใช้ระบบทุกครั้ง
- ผู้ใช้สามารถupdate สถานะรับเลี้ยงทุก 1 วัน, 1 เดือน, 3 เดือนหลังจากรับเลี้ยงไปแล้ว

ผู้ดูแลระบบ

- ต้องลงทะเบียนก่อนเข้าใช้ระบบทุกครั้ง
- สามารถรายงานผู้ใช้ได้กรณีรับเลี้ยงสัตว์ไม่สำเร็จหรือมีปัญหาในการเลี้ยงหรือโพสต์เนื้อหาที่ไม่เกี่ยวข้องกับจุดประสงค์ของระบบ

- ผู้ดูแลสามารถลบโพสต์ที่มีการซื้อขายสัตว์เลี้ยงได้

2.4 ข้อจำกัด

- ไม่มีการตรวจสอบทางกฎหมายคุ้มครองสัตว์ (กรณีที่มีผู้ทรมานสัตว์)
- ไม่มีการซื้อขายสัตว์ในเว็บ
- ใช้งานในรูปแบบของเว็บเท่านั้น

3. ข้อกำหนดความต้องการ

3.1 ข้อกำหนดส่วนต่อประสานภายนอก

3.1.1 ส่วนต่อประสานกับผู้ใช้

A hand-drawn login form on a light yellow background. It contains a text input field labeled 'User name', another text input field labeled 'Password', an oval button labeled 'Login', and a link at the bottom that says 'not registered? Click here'.

ผู้ใช้เว็บเบราว์เซอร์เป็นครั้งแรกจะเห็นหน้าเข้าสู่ระบบ

หากผู้ใช้อย่างไม่ได้ลงทะเบียน ผู้ใช้ควรจะสามารถดำเนินการดังกล่าวได้ในหน้าเข้าสู่ระบบ

A hand-drawn user profile page. The header has three buttons: 'multiselect', 'query', and 'query'. Below the header is a profile card for a user named 'tam' with a heart icon and a timestamp '12:00'. Below the card is a section labeled 'นางสาวไนน์นัย' and '3 comment'. At the bottom right is a 'write' button.

เมื่อผู้ใช้ต้องการโพสต์เพื่อส่งต่อสัตว์เลี้ยงเมื่อเลี้ยงไม่ไหวหรือพบเจอตามสถานที่ต่างๆ

A hand-drawn post creation form. It has a profile section with a user icon and name 'tam', and a text input field. Below this is a list of categories: 'jijee', 'kanun', and 'new'. There is also a 'save' button at the bottom right.

เมื่อมีผู้ใช้มีความสนใจในการรับเลี้ยงสัตว์ต้องมีคอมเม้นหรือแสดงความคิดเห็นในโพสต์นั้นๆ

3.1.2 ส่วนต่อประสานของฮาร์ดแวร์

- ใช้ฮาร์ดแวร์ที่เชื่อมกับweb sever
- เชื่อมต่อระบบฐานข้อมูลกับผู้ใช้ด้วยweb sever

3.1.3 ส่วนต่อประสานของซอฟต์แวร์

- ระบบฐานข้อมูล:MySQL เวอร์ชัน 5.7.21
- ภาษาที่ใช้เขียน web application : PHP
- Web framework:Laravelเวอร์ชัน 9

3.1.4 ส่วนต่อประสานทางการสื่อสาร

- ใช้ HTTP ในการใช้งานเว็บเบราว์เซอร์ที่เป็นมาตรฐานโปรโตคอล

3.2ความต้องการแบบ functional

3.2.1 The User

3.2.1.1 เข้าเว็บ

คำบรรยายและความสำคัญ: ผู้ใช้สามารถใช้งานผ่าน web server ได้

ระดับความสำคัญคือ Mandatory

ลำดับการทำงาน: ผู้ใช้สามารถเข้า webapplication ผ่านlinkได้

ความต้องการ:

- 1) description:ผู้ใช้งานสามารถเข้า webapplication ผ่านlinkได้
- 2) Input:link
- 3) process: การเข้าเว็บแอปพลิเคชัน
- 4) outputs: หน้าลงทะเบียนของเว็บ
- 5) errors handling: แจ้งผู้ดูแลระบบ

3.2.1.2 ลงทะเบียน

คำบรรยายและความสำคัญ: ผู้ใช้ทุกคนต้องลงทะเบียนก่อนเข้าใช้ระบบ

ระดับความสำคัญคือ Mandatory

ลำดับการทำงาน:

- 1) กรอกข้อมูลส่วนบุคคล
- 2) กดปุ่มลงทะเบียน

ความต้องการ:

- 1) description: ผู้ใช้ต้องการกรอก ชื่อ นามสกุล เบอร์โทรศัพท์ ที่อยู่ อายุ อีเมล และ รหัสผ่าน ให้ครบ
- 2) Input: ชื่อ นามสกุล เบอร์โทรศัพท์ ที่อยู่ อายุ อีเมล และ รหัสผ่าน
- 3) process: ระบบจะเพิ่มข้อมูลผู้ใช้ลงฐานข้อมูล
- 4) outputs: ลงทะเบียนสำเร็จสามารถใช้อีเมลและรหัสผ่านที่กรอกไว้ loginเข้าระบบได้
- 5) errors handling:แจ้งผู้ดูแลระบบ

3.2.1.3 เข้าสู่ระบบ

คำบรรยายและความสำคัญ:ผู้ใช้ทุกคนต้องเข้าสู่ระบบก่อนเข้าใช้ระบบ

ระดับความสำคัญคือ Mandatory

ลำดับการทำงาน:

- 1) กรอกอีเมล
- 2) กรอกรหัสผ่าน
- 3) กดปุ่มเข้าสู่ระบบ

ความต้องการ:

- 1) description:ผู้ต้องใช้อีเมลและรหัสผ่านเดียวกันกับอีเมลและรหัสผ่านที่ใช้ลงทะเบียน
- 2) Input:อีเมลและรหัสผ่าน
- 3) process:ระบบจะตรวจสอบกับฐานข้อมูลว่าตรงกันหรือไม่
- 4) outputs:เข้าสู่ระบบสำเร็จ
- 5) errors handling:กรอกรหัสผ่านหรืออีเมลใหม่

3.2.1.4 ยืนยันตัวตนในe-mail

คำบรรยายและความสำคัญ:ใช้ยืนยันว่าผู้ใช้มีตัวตนอยู่จริง ระดับความสำคัญคือ

Mandatory

ลำดับการทำงาน:

- 1) ระบบส่งอีเมลยืนยันตัวตนไปยังอีเมลที่ใช้ลงทะเบียน
- 2) กดยืนยันตัวตนทางอีเมลจากอีเมลที่ได้รับจากทางระบบ

ความต้องการ:

- 1) description: ผู้ใช้กดยืนยันอีเมลที่ส่งไปจึงจะถือว่าลงทะเบียนสำเร็จ
- 2) Input: อีเมลของผู้ใช้ที่ใช้ลงทะเบียน
- 3) process: ระบบส่งอีเมลยืนยันตัวตนไปยังอีเมลที่ใช้ลงทะเบียน
- 4) outputs: การลงทะเบียนสำเร็จ
- 5) errors handling: เช็คว่าอีเมลถูกหรือไม่

· 3.2.1.5 โฟสต์

คำบรรยายและความสำคัญ: ผู้ใช้สามารถใช้การโฟสต์ในกรณีของการส่งต่อสัตว์เลี้ยงเท่านั้น ระดับความสำคัญคือ Desirable

ลำดับการทำงาน:

- 1) สร้างโฟสต์
- 2) ระบุชนิด ชื่อ อายุ สถานะการฉีดวัคซีน สถานที่อยู่/พบ ของสัตว์เลี้ยง
- 3) กดโฟสต์

ความต้องการ:

- 1) description: ผู้ระบุชนิด ชื่อ อายุ สถานะการฉีดวัคซีน สถานที่อยู่/พบ ของสัตว์เลี้ยงให้ครบ หากไม่ทราบให้กรอกว่าไม่ทราบ
- 2) Input: ชนิด ชื่อ อายุ สถานะการฉีดวัคซีน สถานที่อยู่/พบ idของผู้ใช้ที่โฟสต์
- 3) process: ระบบจะเก็บข้อมูลลงฐานข้อมูล
- 4) outputs: โฟสต์จะถูกโพสต์ลงหน้าหลัก
- 5) errors handling: กรอกข้อมูลสำคัญให้ครบ

3.2.1.6 แก้ไขโฟสต์

คำบรรยายและความสำคัญ: ผู้ใช้สามารถแก้ไขโฟสต์ในกรณีที่กรอกข้อมูลผิดได้ ระดับความสำคัญคือ Desirable

ลำดับการทำงาน:

- 1) กดแก้ไขโฟสต์
- 2) กรอกข้อมูลที่ต้องการแก้ไขใหม่
- 3) กดบันทึกข้อมูล

ความต้องการ:

- 1) description: ผู้ใช้สามารถแก้ไขข้อมูลที่กรอกผิดได้แต่จะต้องกรอกข้อมูลให้ครบตามเดิม
- 2) Input: ข้อมูลที่ต้องการแก้ไขใหม่
- 3) process: แก้ไขข้อมูลในฐานข้อมูล
- 4) outputs: ข้อมูลเปลี่ยนแปลงเป็นข้อมูลที่กรอกใหม่
- 5) errors handling: กรอกข้อมูลให้ครบตามเดิม

3.2.1.7 ดูโปรไฟล์ของตัวเอง

คำบรรยายและความสำคัญ: ผู้ใช้สามารถเข้าไปดูโปรไฟล์ตัวเองได้

ระดับความสำคัญคือ Desirable

ลำดับการทำงาน: กดที่โปรไฟล์หรือรูปไอคอนของตนเองที่หน้าหลัก

ความต้องการ:

- 1) description: โปรไฟล์จะประกอบด้วยข้อมูลทั้งหมดของผู้ใช้และสถานะการติดตามของสัตว์เลี้ยงที่เคยรับเลี้ยง(ในกรณีที่เคยรับเลี้ยง)
- 2) Input: โปรไฟล์ของผู้ใช้คนนั้น
- 3) process: ระบบจะหาข้อมูลของผู้ใช้จากฐานข้อมูล
- 4) outputs: ระบบจะแสดงข้อมูลทั้งหมดของผู้ใช้และสถานะการติดตามของสัตว์เลี้ยงที่เคยรับเลี้ยง(ในกรณีที่เคยรับเลี้ยง)
- 5) errors handling: ติดต่อผู้ดูแลระบบ

3.2.1.8 ดูโปรไฟล์ของคนอื่น

คำบรรยายและความสำคัญ: ผู้ใช้สามารถเข้าไปดูโปรไฟล์ของคนที่จะรับเลี้ยงสัตว์

ของเราหรือคนโพสต์ได้ เพื่อตรวจสอบความแน่ใจในการส่งต่อสัตว์เลี้ยง

ระดับความสำคัญคือ Desirable

ลำดับการทำงาน: กดที่รูปโปรไฟล์ของคนอื่นที่โพสต์หรือคอมเมนต์

ความต้องการ:

- 1) description: ผู้ใช้สามารถเข้าไปดูโปรไฟล์ของคนที่จะรับเลี้ยงสัตว์ของเราหรือคนโพสต์ได้
- 2) Input: โปรไฟล์ของผู้ใช้คนนั้น
- 3) process: ระบบจะหาข้อมูลของผู้ใช้จากฐานข้อมูล

- 4) outputs:ระบบจะแสดงข้อมูลทั้งหมดของผู้ใช้และสถานะการติดตามของสัตว์เลี้ยงที่เคยรับเลี้ยง(ในกรณีที่เคยรับเลี้ยง)
- 5) errors handling:ติดต่อผู้ดูแลระบบ

3.2.1.9 แสดงหมวดหมู่สัตว์

คำบรรยายและความสำคัญ:เพื่อลดเวลาในการหาสัตว์เลี้ยงโดยการกำหนดชนิดที่

อยากหา ระดับความสำคัญคือ Desirable

ลำดับการทำงาน:กดเลือกชนิดที่อยากหา

ความต้องการ:

- 1) description:ผู้ใช้งานสามารถกดเลือกชนิดที่อยากหาได้จากชนิดpet,exotic pet และสามารถแยกย่อยได้อีก เช่นexotic pet และหาเพิ่มเป็น กิ้งก่า งู เป็นต้น หรือถ้าสนใจอยากจะเลื่อนดูทั้งหมดให้กดall
- 2) Input:ชนิดที่ผู้ใช้เลือก
- 3) process:ระบบจะหาข้อมูลของชนิดสัตว์ที่เลือกจากฐานข้อมูล
- 4) outputs:ระบบจะแสดงโพสต์ของชนิดสัตว์ที่เลือก
- 5) errors handling:ติดต่อผู้ดูแลระบบ

3.2.1.10 แสดงความคิดเห็น

คำบรรยายและความสำคัญ:ผู้ใช้งานสามารถแสดงความสนใจในการอยากรับเลี้ยงได้

ระดับความสำคัญคือ Desirable

ลำดับการทำงาน:

- 1) เลือกโพสต์ที่สนใจ
- 2) กดแสดงความคิดเห็น
- 3) พิมพ์ความคิดเห็นที่ต้องการแสดง
- 4) กดโพสต์

ความต้องการ:

- 1) description:ผู้ใช้งานสามารถแสดงความสนใจในการอยากรับเลี้ยงได้เพื่อที่คนโพสจะพิจารณาส่งสัตว์เลี้ยงให้
- 2) Input:ความคิดเห็นที่ต้องการแสดง
- 3) process:ระบบจะเพิ่มการแสดงความคิดเห็นลงโพสต์และฐานข้อมูลเกี่ยวกับโพสนั้น

- 4) outputs:แสดงความคิดเห็นจะแสดงได้โพสต์นั้น
- 5) errors handling:ติดต่อผู้ดูแลระบบ

3.2.1.11 Update สถานการณ์รับเลี้ยง

คำบรรยายและความสำคัญ:ผู้ใช้สามารถบอกได้ว่าสัตว์เลี้ยงที่ตนโพสต์ไปถูกรับเลี้ยงไปแล้วหรือยังไม่ถูกรับเลี้ยง ระดับความสำคัญคือ Desirable

ลำดับการทำงาน:

- 1) กดที่โพสต์ที่ตนเคยโพสต์
- 2) กดอัปเดตสถานะ
- 3) เลือกสถานะที่ต้องการ
- 4) กดบันทึก

ความต้องการ:

- 1) description:ผู้ใช้สามารถบอกได้ว่าสัตว์เลี้ยงที่ตนโพสต์ไปถูกรับเลี้ยงไปแล้วหรือยังไม่ถูกรับเลี้ยง
- 2) Input:สถานะของสัตว์เลี้ยง
- 3) process:ระบบจะเปลี่ยนข้อมูลในฐานข้อมูล
- 4) outputs:โพสต์จะถูกลบออกจากหน้าหลัก
- 5) errors handling:ติดต่อผู้ดูแลระบบ

3.2.1.12 Favorite post

คำบรรยายและความสำคัญ:ผู้ใช้สามารถกดถูกใจโพสต์ที่สนใจเพื่อประกอบการตัดสินใจรับเลี้ยงสัตว์ได้ ระดับความสำคัญคือ Optional

ลำดับการทำงาน: กดที่ไอคอนรูปหัวใจ

ความต้องการ:

- 1) description:ผู้ใช้สามารถกดถูกใจโพสต์ที่สนใจเพื่อประกอบการตัดสินใจรับเลี้ยงสัตว์ได้ และสามารถกดเลิกถูกใจได้ที่ไอคอนเดิม
- 2) Input:การกดที่ไอคอนรูปหัวใจ
- 3) process:ระบบจะเพิ่มโพสต์นี้ลงในฐานข้อมูลที่ใช้คนนั้นถูกใจ
- 4) outputs:โพสต์ถูกเก็บไว้ในหน้าโพสต์ที่ฉันถูกใจ
- 5) errors handling:ติดต่อผู้ดูแลระบบ

3.2.2 The Admin

3.2.2.1 เพิ่มข้อมูลชนิดของสัตว์

คำบรรยายและความสำคัญ: สัตว์บนโลกนี้มีมากมายหลายชนิดมาก

ระบบนี้จัดทำขึ้นมาเพื่อเพิ่มชนิดของสัตว์ที่ต้องการโพสต์ส่งต่อแต่ผู้ใช้หาชนิดของสัตว์ชนิดนี้ไม่เจอ ระดับความสำคัญคือ Desirable

ลำดับการทำงาน:

- 1) ผู้ใช้ส่งอีเมลไปให้ผู้ดูแลระบบ
- 2) ผู้ดูแลระบบเพิ่มชนิดของสัตว์
- 3) กดบันทึก

ความต้องการ:

- 1) description: เพิ่มชนิดของสัตว์ที่ต้องการโพสต์ส่งต่อแต่ผู้ใช้หาชนิดของสัตว์ชนิดนี้ไม่เจอ
- 2) Input: ชนิดของสัตว์ที่ต้องการเพิ่ม
- 3) process: ระบบเพิ่มข้อมูลลงฐานข้อมูล
- 4) outputs: ชนิดของสัตว์มีให้userเลือกมากขึ้น
- 5) errors handling: ติดต่อผู้ดูแลระบบ

3.2.2.2 รายงานผู้ใช้

คำบรรยายและความสำคัญ: เนื่องจากเว็บจัดทำขึ้นเพื่อการกุศล

จึงต้องป้องกันในกรณีผู้ใช้ใช้เว็บแอปพลิเคชันในเชิงพาณิชย์

โพสต์หาบ้านให้สัตว์สงวน หรือโพสต์เนื้อหาที่ไม่เกี่ยวข้องกับจุดประสงค์ของระบบ

ระดับความสำคัญคือ Desirable

ลำดับการทำงาน: กดที่โพสต์นั้นและรายงานผู้ใช้ที่โพสต์

ความต้องการ:

- 1) description:
ผู้ใช้ที่ถูกรายงานจะได้รับการแจ้งเตือนทางอีเมลว่าถูกรายงานเนื่องจากอะไร
- 2) Input: โพสต์ที่มีเนื้อหาไม่เหมาะสม
- 3) process: ผู้ดูแลระบบกดปุ่มรายงานและระบบจะส่งอีเมลไปหาผู้ใช้

- 4) outputs: ผู้ใช้ที่ถูกรายงานจะได้รับการแจ้งเตือนทางอีเมลว่าถูกรายงานเนื่องจากอะไร
- 5) errors handling: ติดต่อผู้ดูแลระบบ

3.3. ความต้องการแบบ Non-functional

3.3.1 ข้อมูลส่วนบุคคลของ user จะต้องเป็นความลับตามพรบ. ข้อมูลส่วนบุคคล

3.3.2 ระบบห้องสมุดออนไลน์จะต้องให้บริการได้ตลอดเวลา แต่อาจล่มเหลวได้ไม่เกิน ร้อยละ 1 ในรอบ 1 สัปดาห์

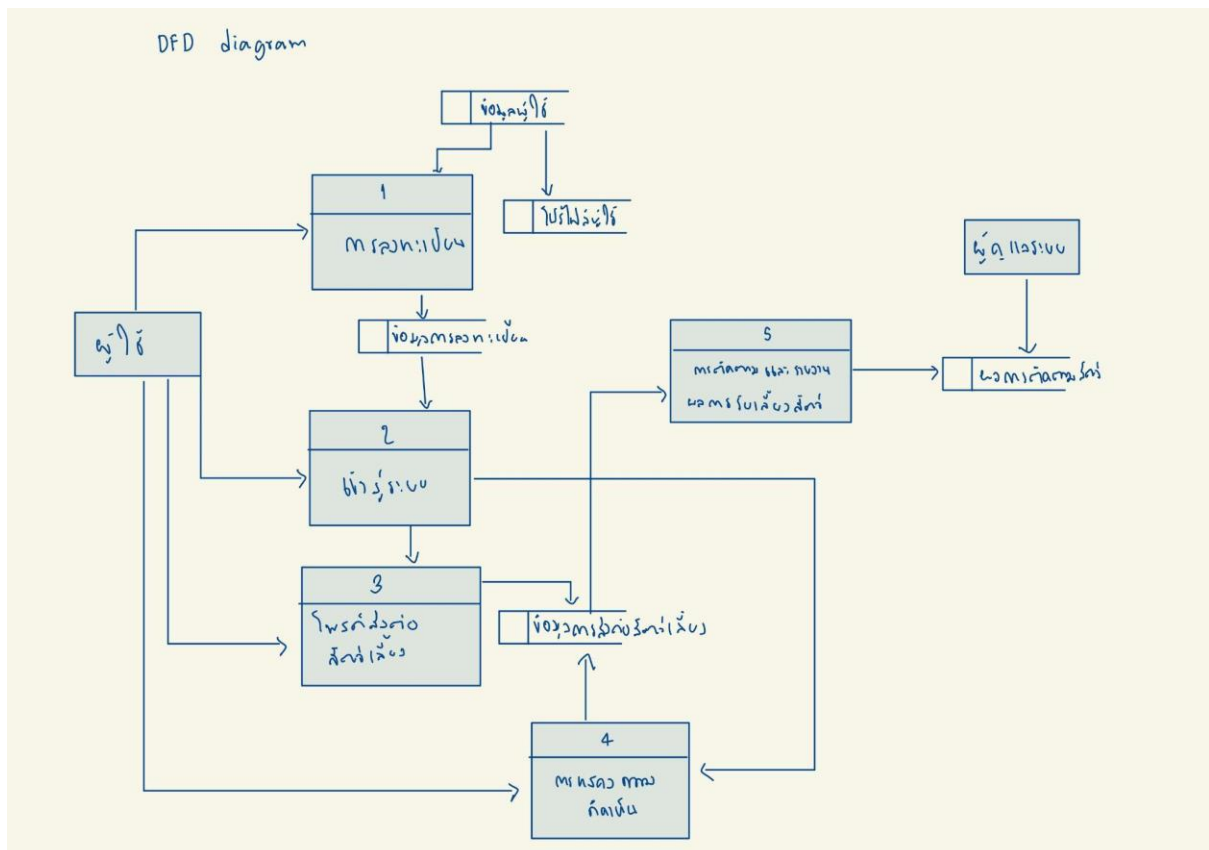
3.3.3 หากระบบล่มเหลว จะต้องกลับมาใช้งานได้ภายใน 20 นาที

3.4. ความต้องการอื่น ๆ

3.4.1 ในระบบต้องไม่มีการซื้อขายสัตว์เลี้ยง

4. แผนภาพการวิเคราะห์ระบบ

DFD diagram :



```

graph TD
    user((user))
    Admin((Admin))
    admin((admin))
    ban((ban))
    login((login))
    register((register))
    password((password))
    forgot((forgot))
    reset((reset))
    change((change))
    update((update))
    delete((delete))

    user --- admin
    user --- ban
    user --- login
    user --- register
    user --- password
    user --- forgot
    user --- reset
    user --- change
    user --- update
    user --- delete

    Admin --- admin
    Admin --- ban
    Admin --- login
    Admin --- register
    Admin --- password
    Admin --- forgot
    Admin --- reset
    Admin --- change
    Admin --- update
    Admin --- delete

    admin -.->|extends| ban
    admin -.->|extends| login
    admin -.->|extends| register
    admin -.->|extends| password
    admin -.->|extends| forgot
    admin -.->|extends| reset
    admin -.->|extends| change
    admin -.->|extends| update
    admin -.->|extends| delete

    ban -.->|extends| login
    ban -.->|extends| register
    ban -.->|extends| password
    ban -.->|extends| forgot
    ban -.->|extends| reset
    ban -.->|extends| change
    ban -.->|extends| update
    ban -.->|extends| delete

    login -.->|extends| register
    login -.->|extends| password
    login -.->|extends| forgot
    login -.->|extends| reset
    login -.->|extends| change
    login -.->|extends| update
    login -.->|extends| delete

    register -.->|extends| password
    register -.->|extends| forgot
    register -.->|extends| reset
    register -.->|extends| change
    register -.->|extends| update
    register -.->|extends| delete

    password -.->|extends| forgot
    password -.->|extends| reset
    password -.->|extends| change
    password -.->|extends| update
    password -.->|extends| delete

    forgot -.->|extends| reset
    forgot -.->|extends| change
    forgot -.->|extends| update
    forgot -.->|extends| delete

    reset -.->|extends| change
    reset -.->|extends| update
    reset -.->|extends| delete

    change -.->|extends| update
    change -.->|extends| delete

    update -.->|extends| delete
  
```

[illegible]