Database Design V1

Last updated:
August 20th

Sprint:
Sprint 1

Version Description:

- This is our very first design made after collecting requirements from client
- Due to the nature of MongoDB, the tables actually don't have physical relations, including multiplicities. However, here we drew lines between them to represent logical relations.

Database Design (names of tables could be discussed further):

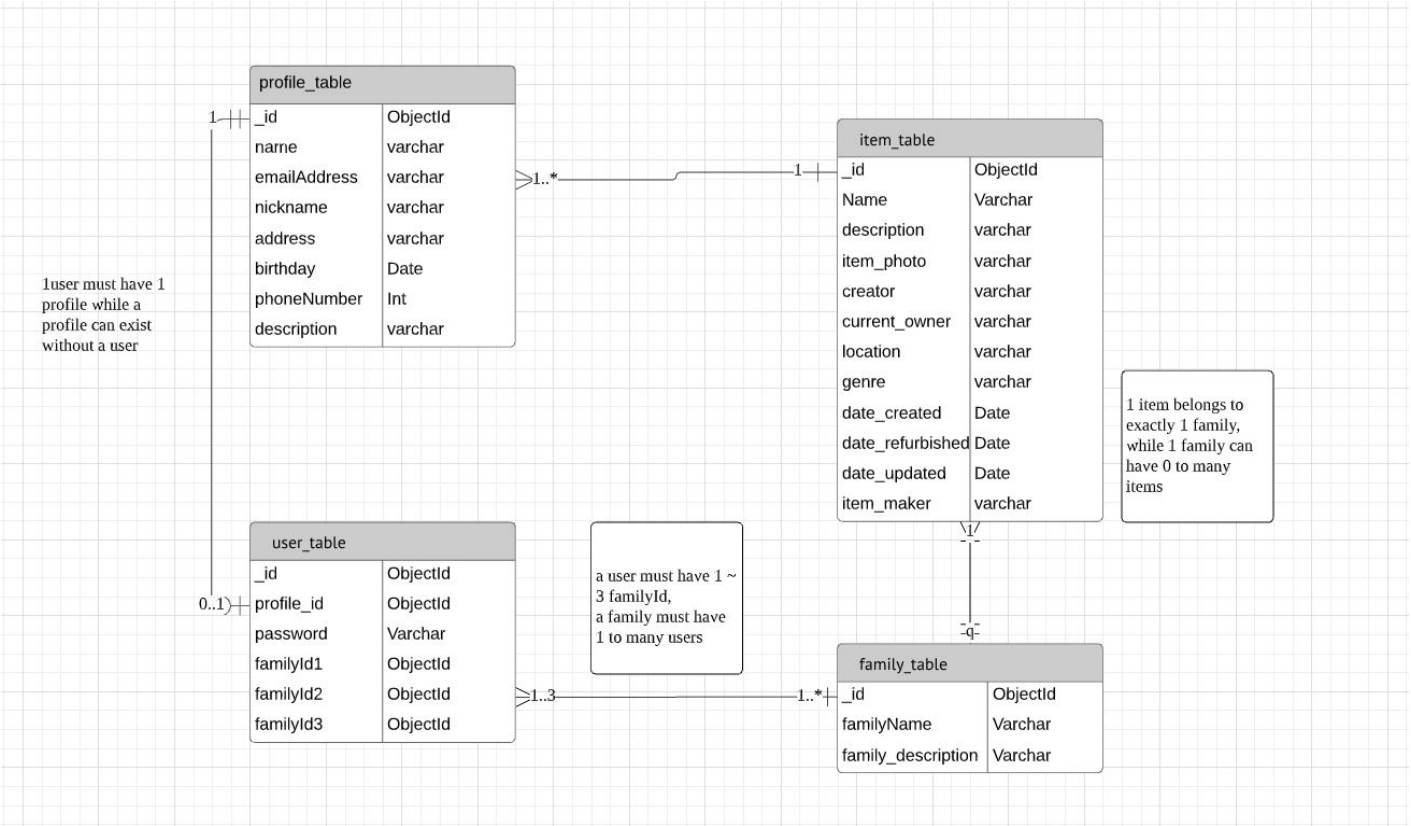


Figure 1 Database design tables

Database Table Details

1. **Item_tables**: all treasure items, items of the same family are grouped by family_id
   id int notnull pk
   Name varchar
   Description varchar
   Item_photo url varchar
   date _added date default notnull The date that item is added to the database
   Date_updated date default
   Date_created data default notnull The date that the item was created(a picture shot by client/ an item purchased by client)
   Date_refurbished data default The data the the item was refurbished
   Creator varchar notnull
   Current_owner varchar notnull
   Location varchar notnull (str 'online' for digital treasure)
   Genre varchar notnull
   Family_id int notnull

2. **profile_tables**: all users, users of the same family are grouped by family_id(here we assume that a person who has place in more than one family would have several entries in the database, with different family_id and user_id)
   *shall we include pets here? No, as asked client
   profile_table and item_table can join by i.creator = u.name or i.current_owner=u.name
   profile_id int notnull pk
   email_address pk,which can be used to login
   name varchar notnull
   nickname varchar notnull
   Family_id int notnull
   Address varchar
   Phone_number int
   Birthday Date
   Authorization bool

3. **family_tables**:
   Family_id int notnull
   Family_name varchar notnull

4. **account_tables**:
   All users have profile while not all profiles have corresponding user(e.g passed

people)
profile_table and user_table can join by p.user_id = u.user_id
User_id int
Password varchar
Family_id1 int
Family_id2 int
Family_id3 int

Non Relational Database Design V2

Last updated:
September 10th

Last version:
File ‘Non Relational Database Design v1’, under the same folder
https://docs.google.com/document/d/1cb-047RQjQOZZ-SqPYB8viTdjZNY4KYNVQmI2OxwZgI/edit?usp=sharing

New Features:

1. All table names now end with ‘s’ to be suitable for mongoDB
2. Changes of attributes including fields, names, and data types have been updated
3. Addition of person_tables, to generalise fields that profile_tables and account_tables share in common

Database Design (names of tables could be discussed further):

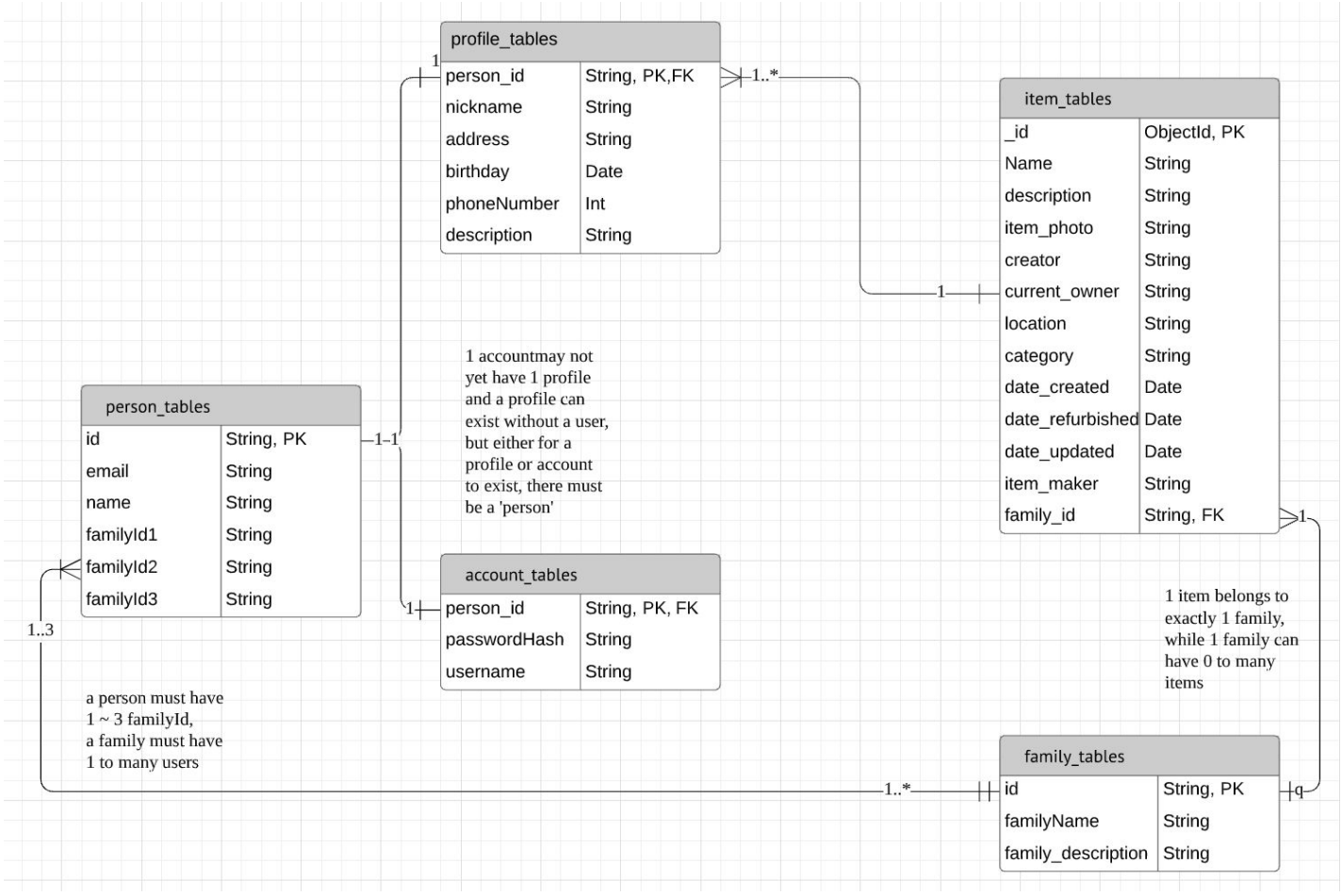


Figure 1 Database design tables

System Flow Design(substitutes SSD)

Flow Chart Link

- Flow chart of the website's supposed jumping between pages(substitution of SSD): https://www.lucidchart.com/invitations/accept/264c3994-ede6-4af0-bd95-cef3ac3fecba

Last Updated

September 10th, 9:07 p.m.

Notations:

1. an arrow means jumpable from one page to the other
2. pages on navigation bar can be reached by clicking on the navigation bar
3. dash line refers to corresponding functions
4. colors: orange = error, blue = user input page, green = information showing page
5. green line = when operation successful, red line = when operation failed
6. arrow from left and top= in, arrow from right and bottom = out

Overall look:

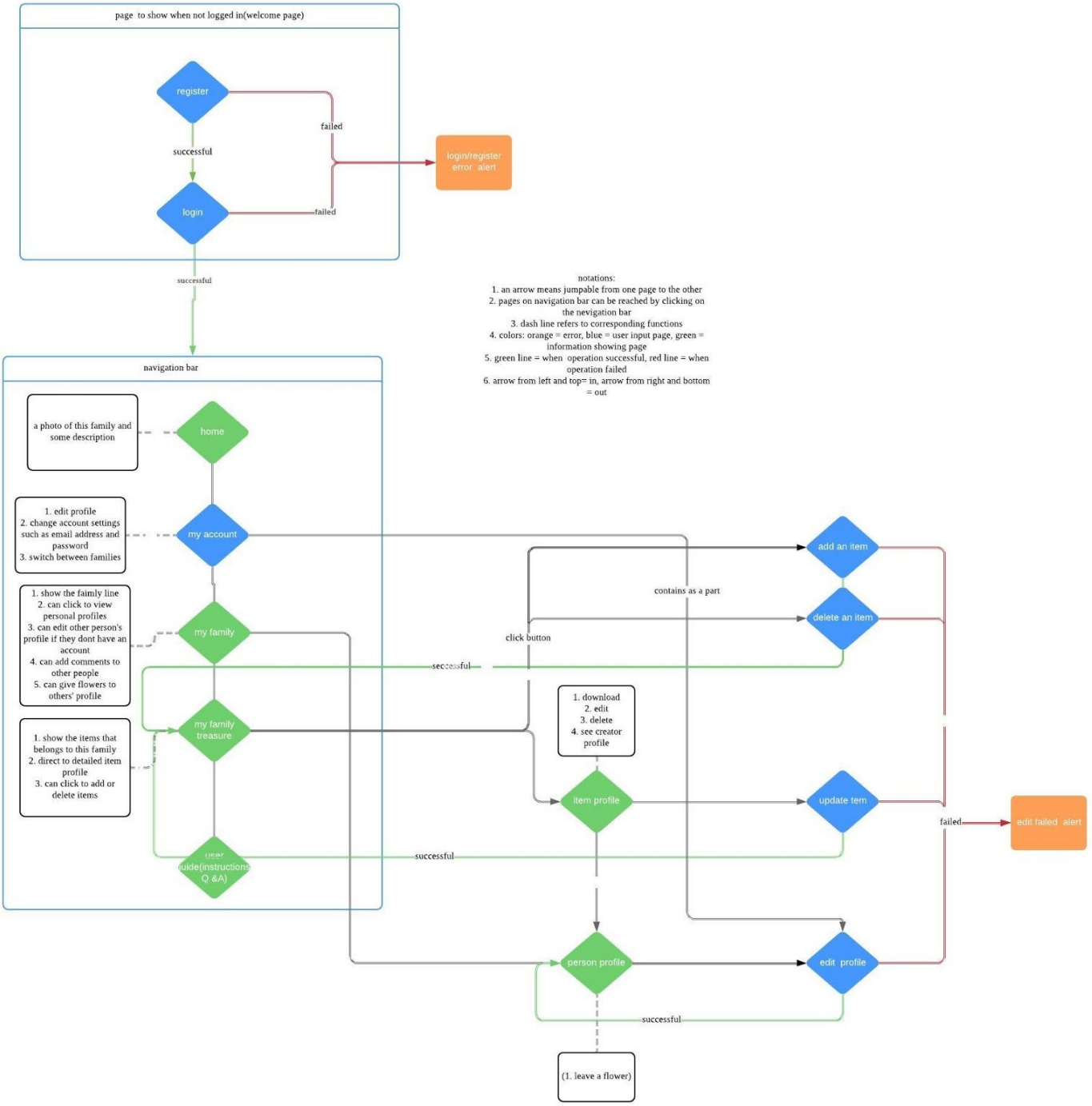


Figure 1: the whole diagram

Welcome page zoom-in:

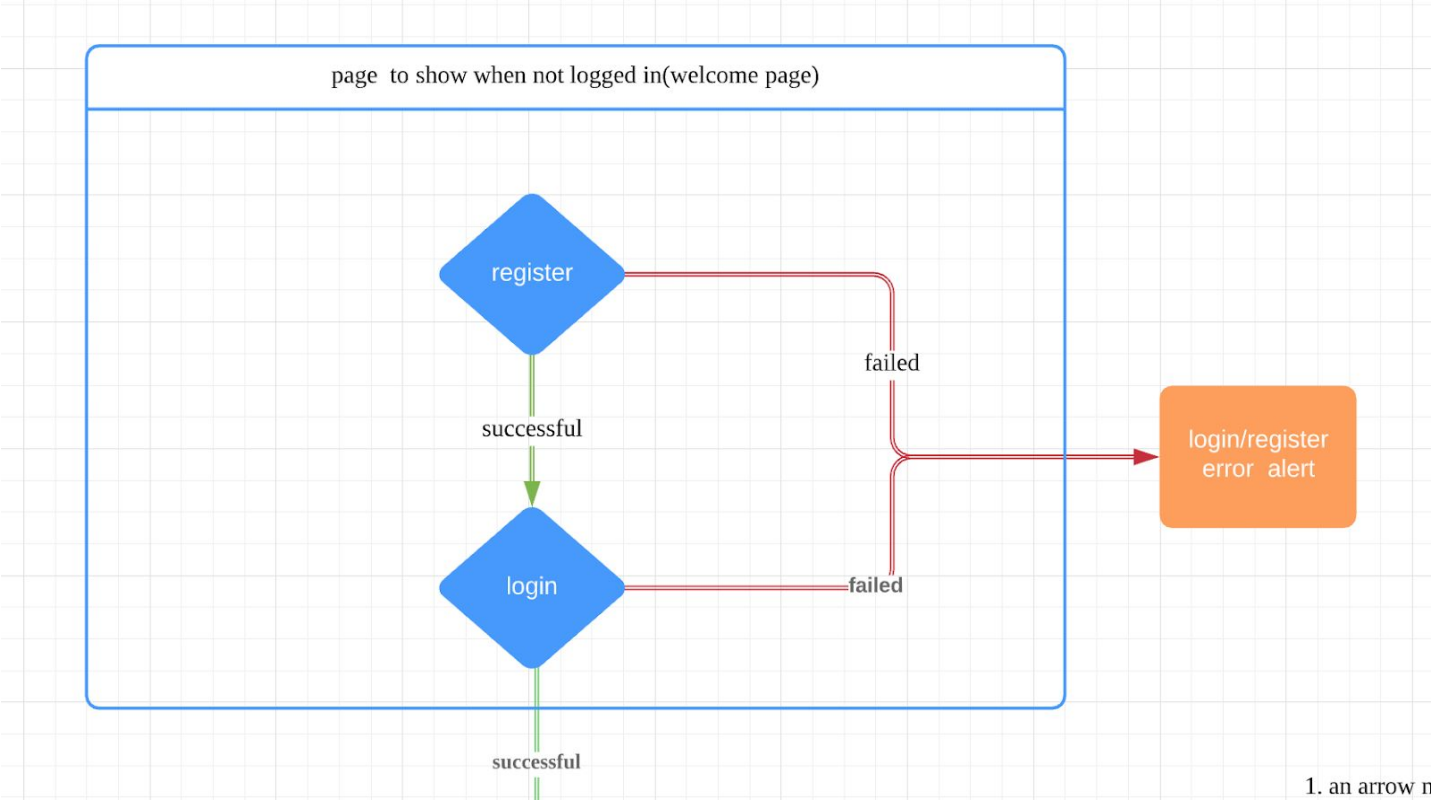


Figure 2: welcome page zoom-in

Navigation bar zoom-in:

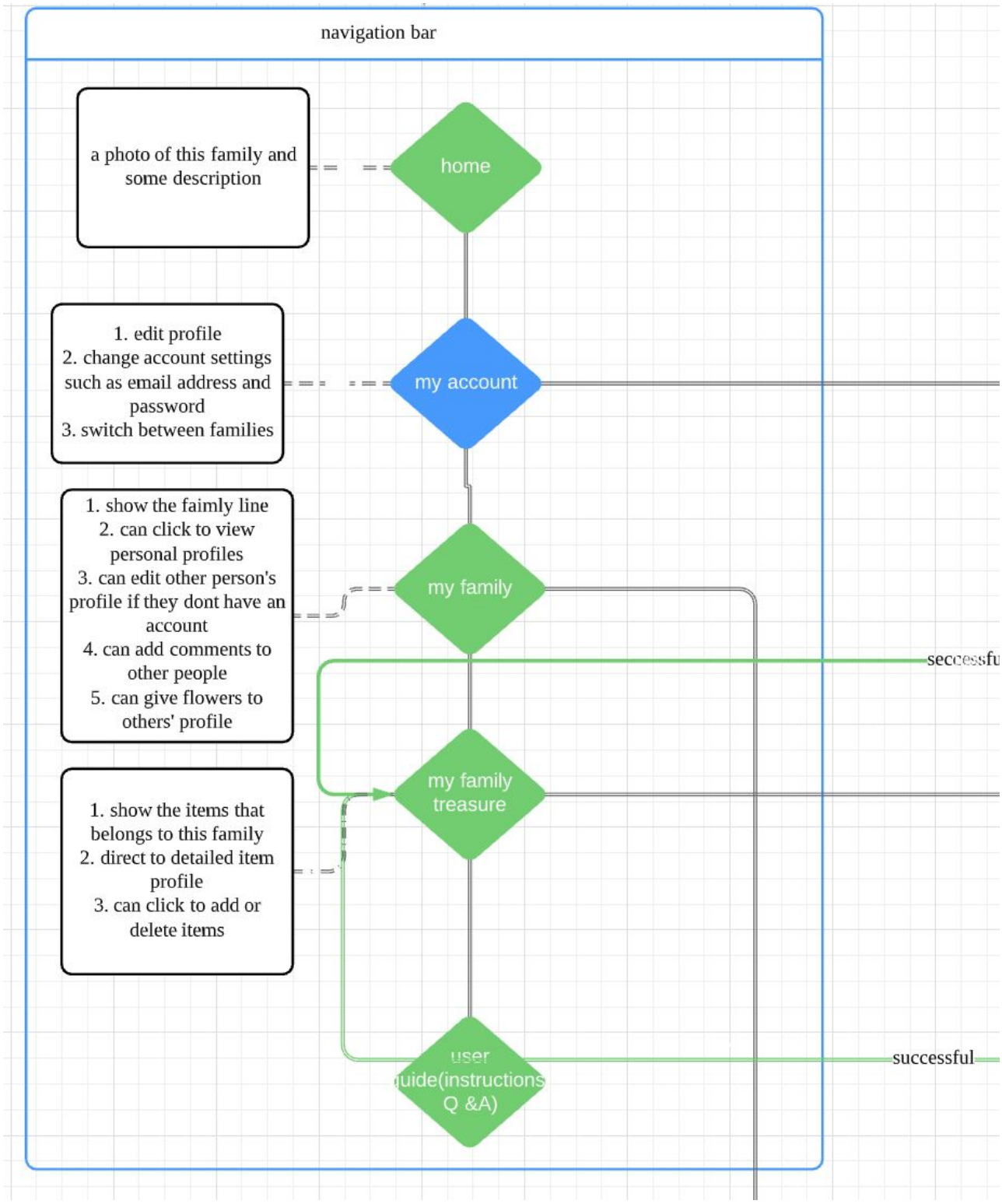


Figure 3: navigation bar zoom-in

Function Flow Zoom-in:

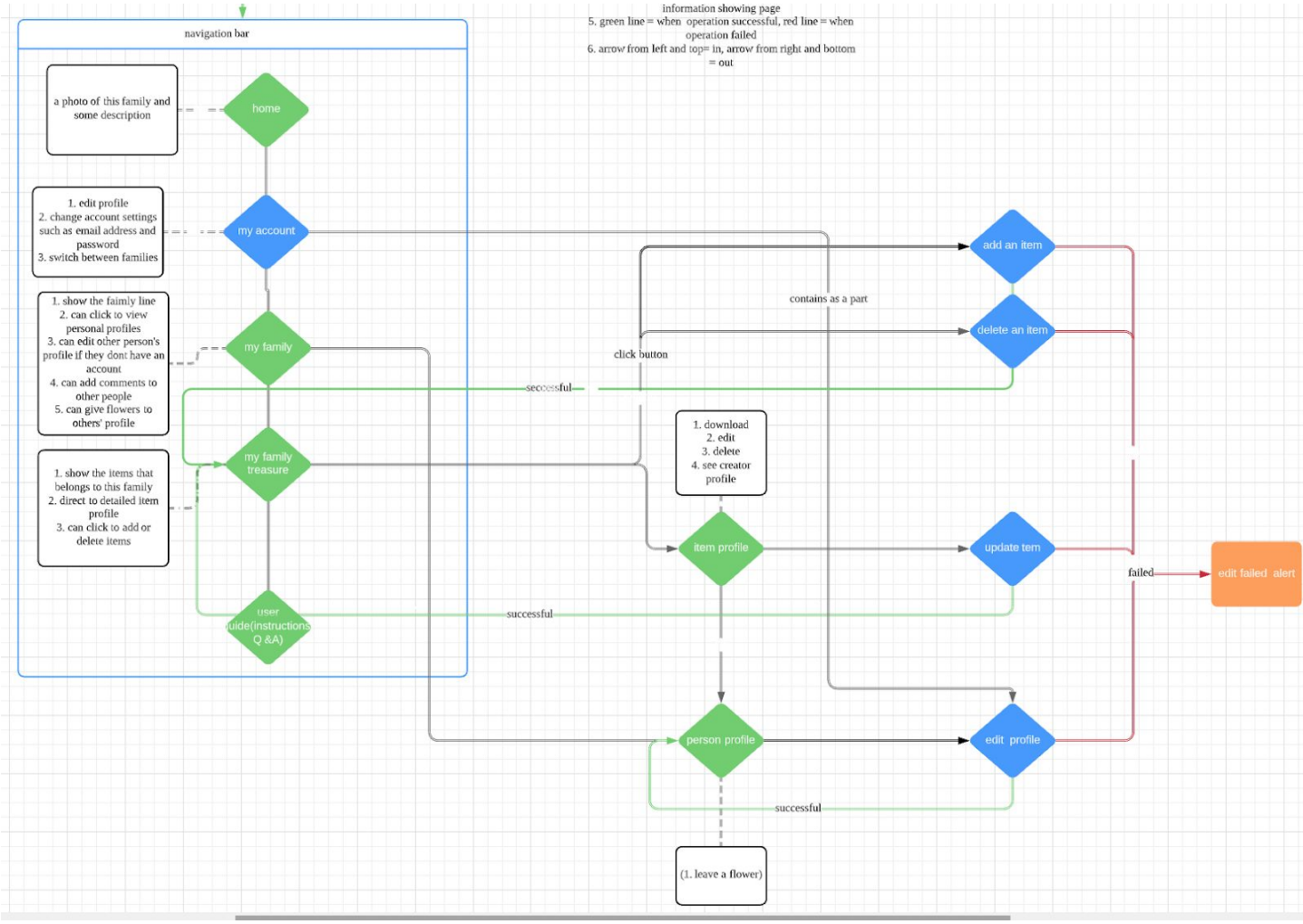
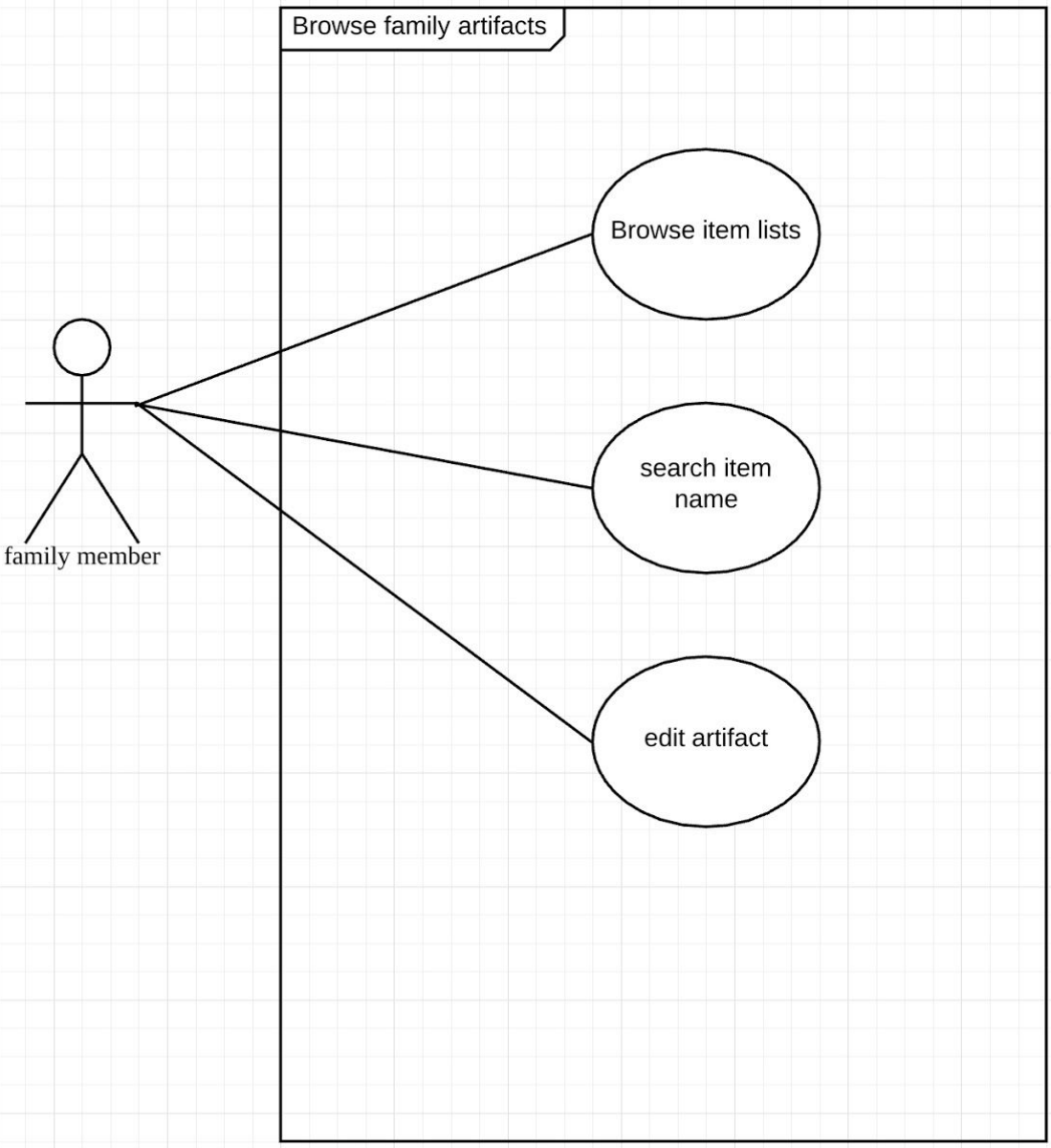
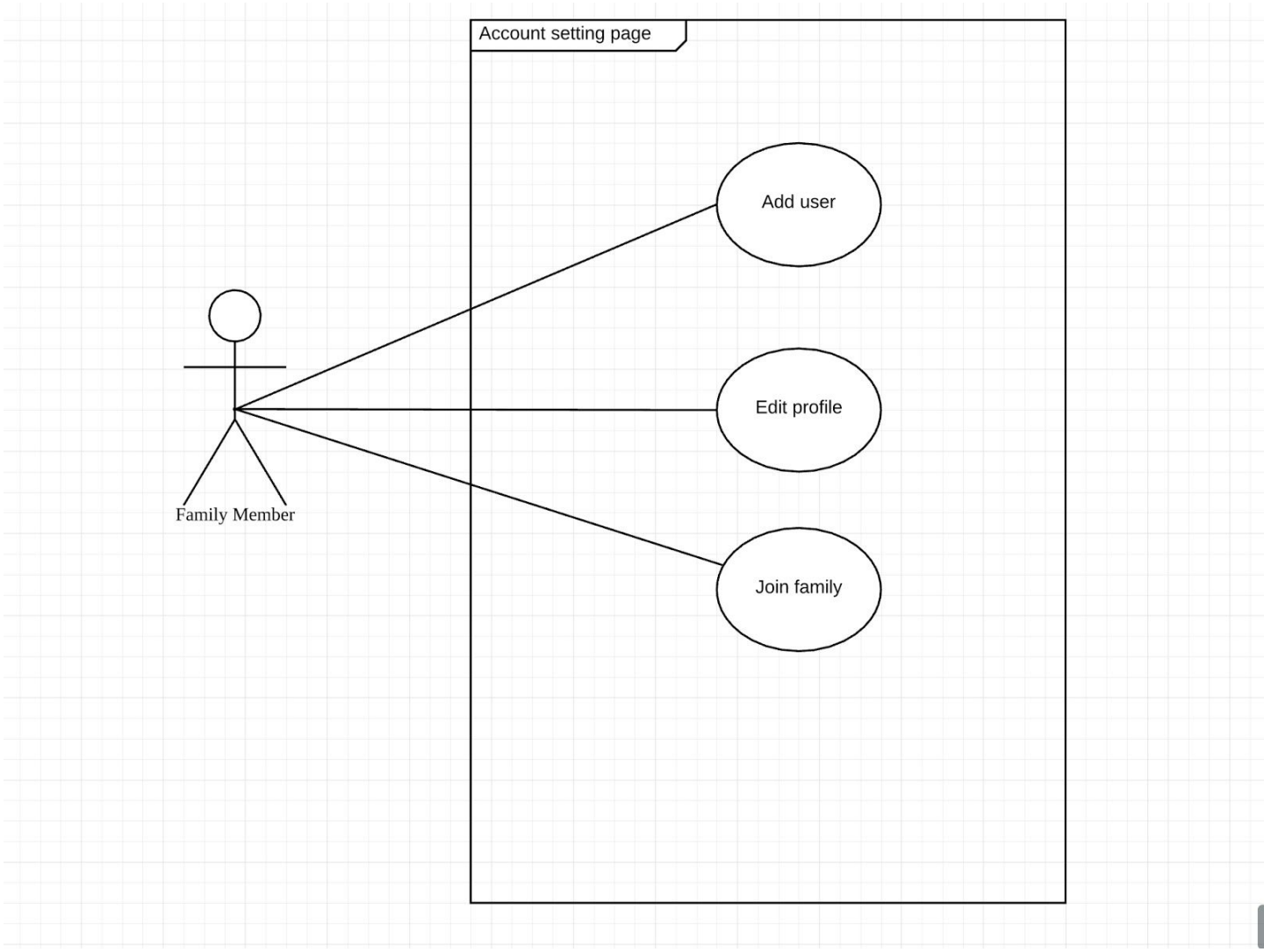


Figure 4: function flow zoom-in

Use Case Diagram




Account setting page
Add user
Edit profile
Join family
Family Member