**HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY**

SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

**REPORT**

**SOFTWARE ENGINEERING**

Project ICT Blog

\*\*\* Group 2:

- Thandar Shwe Sin - 2017T012

- Aye Myat Thu - 2017T014

- Phạm Quang Huy - 20151703

- Phạm Minh Tư – 20154293

- Đoàn Sỹ Hùng- 20151776

Hanoi, 2017

**1. Requirement Specification:**

In the realm of technology, we all need a platform where we can share our thoughts. Facebook seems to be a good platform, but contains too many ads and it is not the place for people who need a quiet place.

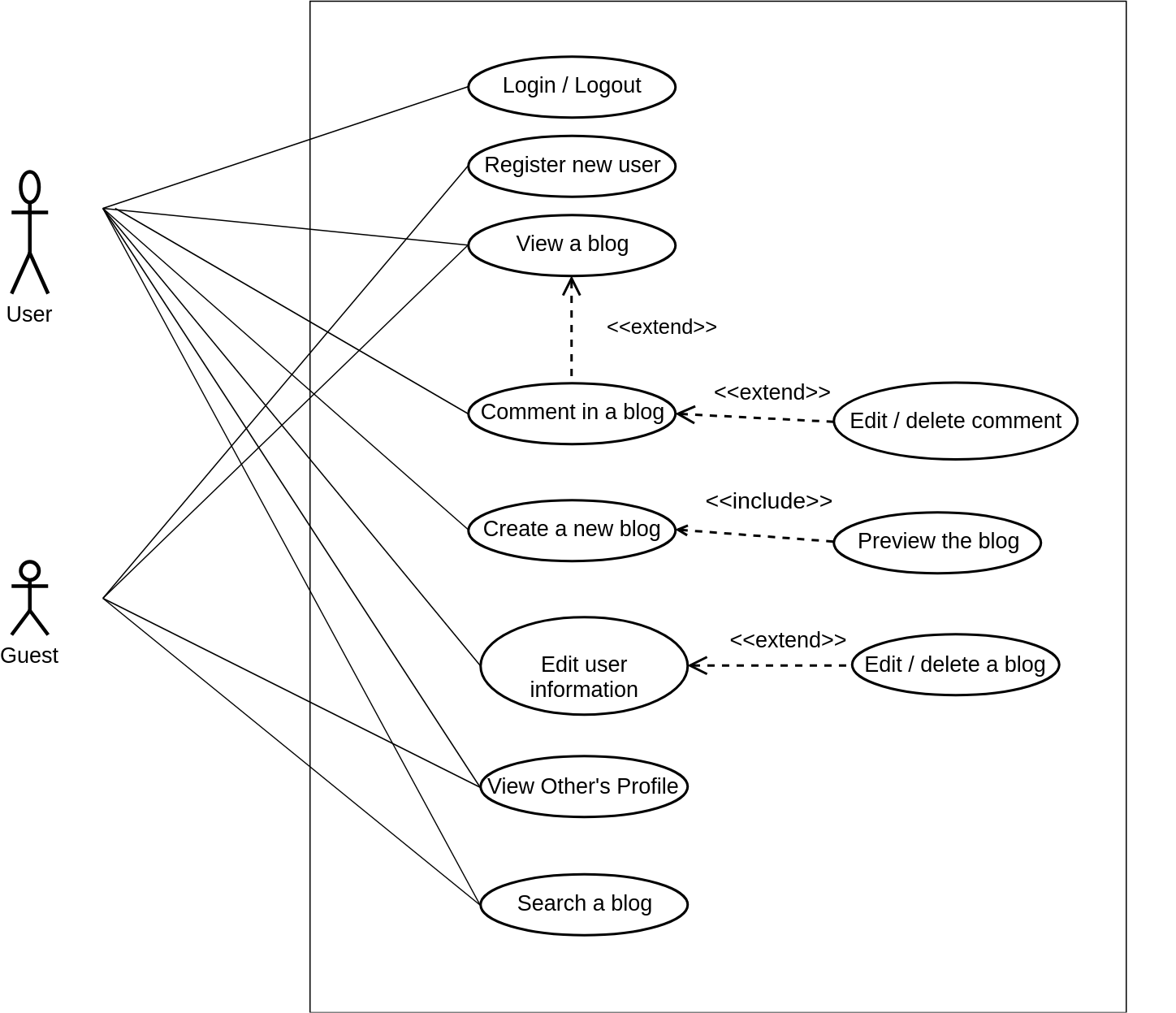
We are thinking of a platform where users can share their thoughts, their feelings and others can post some comments on that.

This idea, although we all know that it has been quite a mainstream idea, we are hoping that in the future, we are able to develop some outstanding features.

The main feature of our project is that blogs of users are accessible among others and they can submit comments to any open posts.

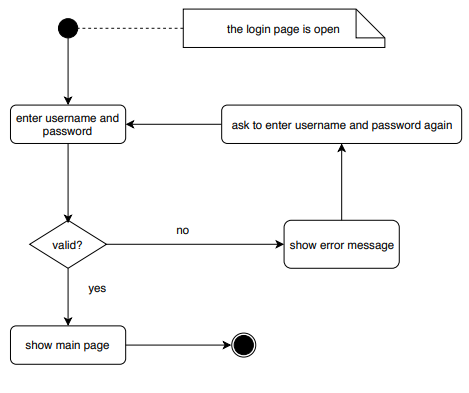
**2. Design diagram**

**2.1. Use-case diagram**

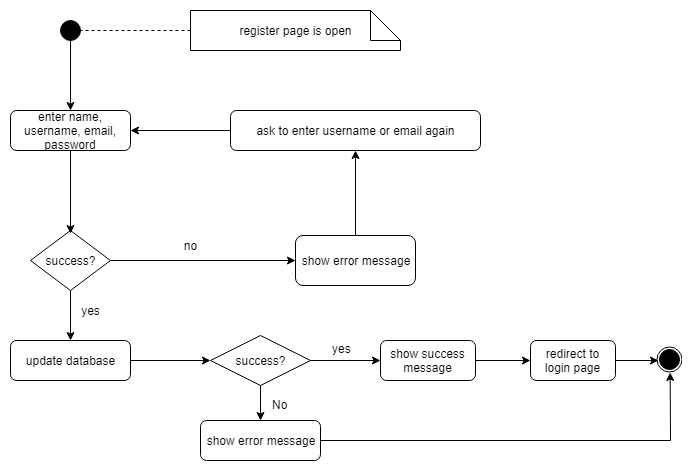


**2.2. Activity diagram**

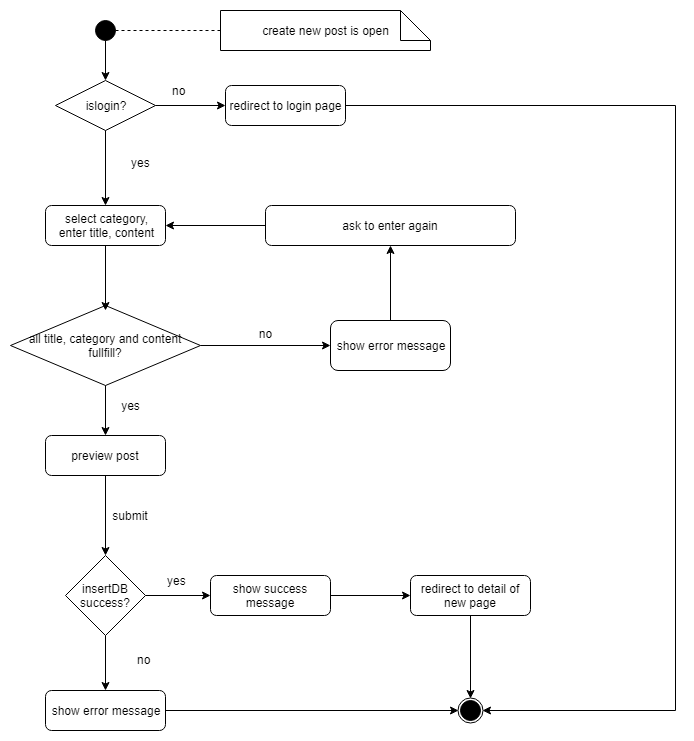
**LOGIN**

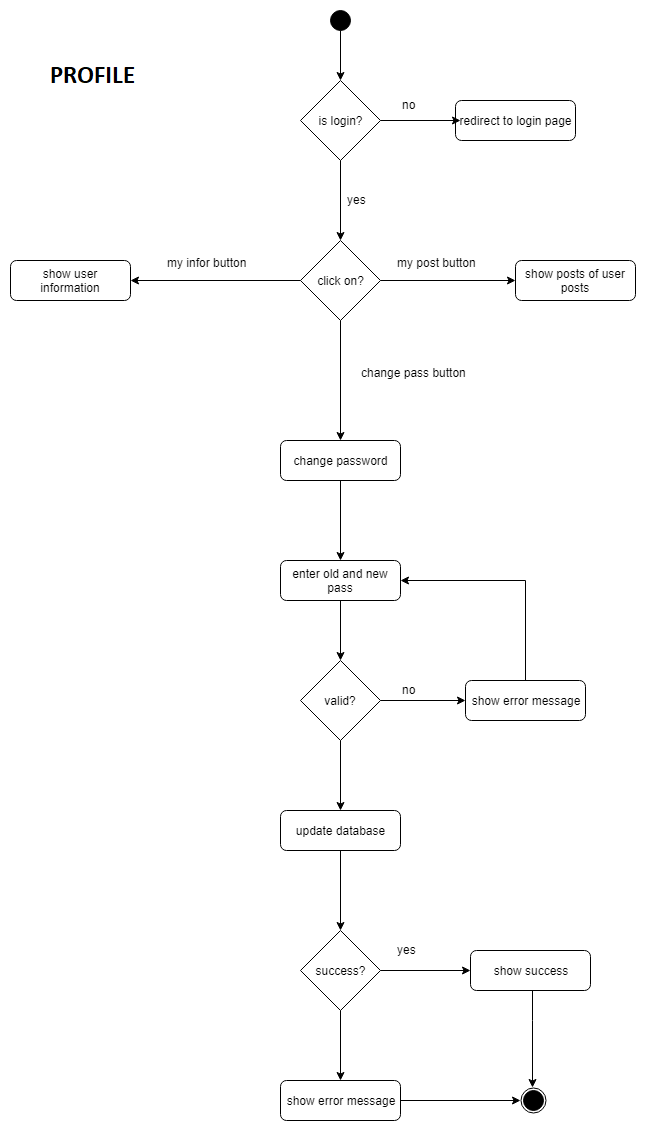


**REGISTER**

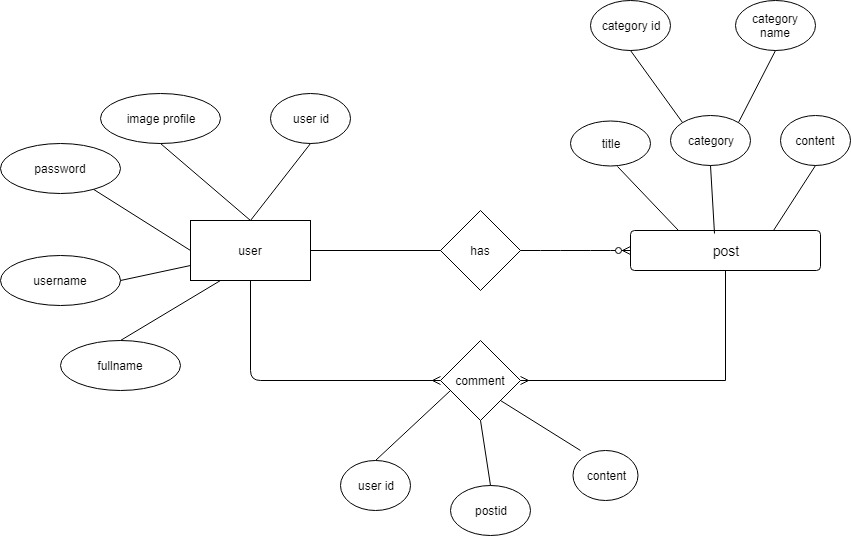


**CREATE A NEW POST**

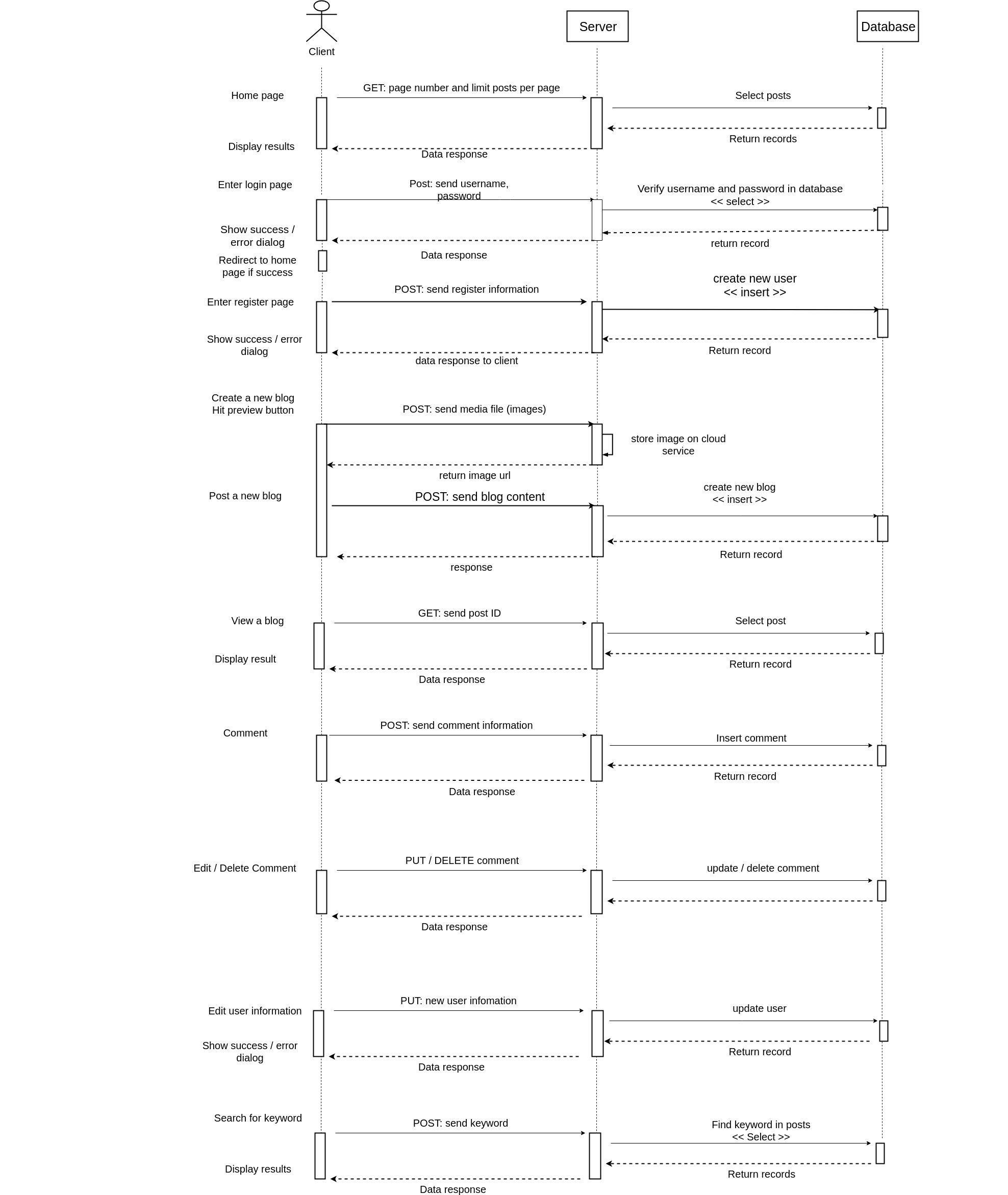




**2.3. Entity Diagram**



**2.4. Sequence Diagram**



**3. Technology**

We build our project on 2 platforms: Web and Mobile Application.

Web:

- Front End: Angular 2+, Bootstrap

Achieve the maximum speed possible on the Web Platform today, and take it further, via Web Workers and server-side rendering.

Angular puts you in control over scalability. Meet huge data requirements by building data models on RxJS, Immutable.js or another push-model.

- Back End: Node.js Express Sequelize

Mobile:

- Front End: Ionic Angular

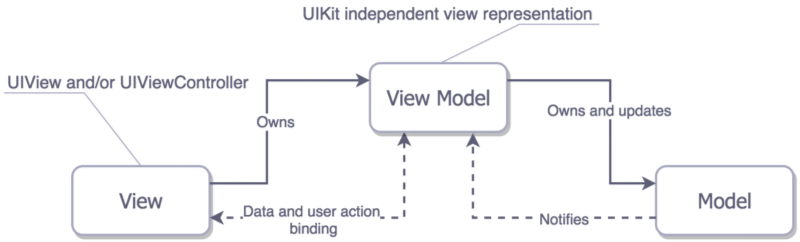
- Back End: Node.js Express Sequelize

Testing:

- Chai, Mocha to test server Node.js

**4. Coding**

We use MVVM design pattern to write codes as this is the pattern that Angular 4 supports.



The **Model**, **View**, **ViewModel** (**MVVM** pattern) is all about guiding you in how to organize and structure your code to write maintainable, testable and extensible applications.

**Model** − It simply holds the data and has nothing to do with any of the business logic.

**ViewModel** − It acts as the link/connection between the Model and View and makes stuff look pretty.

**View** − It simply holds the formatted data and essentially delegates everything to the Model.

**5. Teamwork**

**Pham Minh Tu**

- Construct the base structure of Web version

- Design UI, and implement logic in client side

- Take charge of building Android application from scratch

**Pham Quang Huy**

- Design database

- Implement RESTful APIs in server side

- Contribute to design UI

**Doan Sy Hung**

- Write testing code

- Draw diagrams

- write documentations

**Thandar Shwe Sin**

- Specify requirements

- Experience and evaluate UI/UX

**Aye Myat Thu**

- Specify requirements

- Experience and evaluate UI/UX

Github repositories:

Client: <https://github.com/trachpro/SE-project>

Server: <https://github.com/quanghuy219/SE-server>

Mobile: <https://github.com/trachpro/SE-Mobile>

Resources:

- <https://startbootstrap.com/template-overviews/clean-blog/> - we base on this UI to develope our application

- <https://nodejs.org/en/docs/>- Nodejs documentation

- <https://angular.io/docs>– Angular documentation

- <http://docs.sequelizejs.com/>- Sequelize

- <https://ionicframework.com/> - Ionic