

| **MINISTRY OF EDUCATION AND TRAINING**  **FPT UNIVERSITY** |
| --- |
| Capstone Project Document |
| **LAF - FPTU Lost And Found** |

| **LAF Team** | |
| --- | --- |
| **Group Members** | Trần Thiên Quý - SE63199  Vũ Lâm Bảo Vân - SE130253  Phạm Nguyễn Hoài Linh - SE63552  Trần Vĩnh Phát - SE130678  Võ Quốc Việt - SE63467 |
| **Supervisor** | Nguyễn Nguyên Bình |

- HoChiMinh, <05>/<2022> -

# Table of Contents

[Table of Contents](#_heading=h.gjdgxs) **2**

[List of Tables](#_heading=h.1ksv4uv) **6**

[List of Firgures](#_heading=h.2jxsxqh) **7**

[Acknowledgement](#_heading=h.gd2mgtianyy5) **7**

[Definition and Acronyms](#_heading=h.1fob9te) **8**

[**I. Project Introduction**](#_heading=h.3znysh7) **8**

[1. Overview](#_heading=h.2et92p0) 8

[1.1 Project Information](#_heading=h.tyjcwt) 8

[1.2 Project Team](#_heading=h.3dy6vkm) 8

[a. Supervisors](#_heading=h.bne5b1xivmcj) 8

[b. Team Members](#_heading=h.2rx0ksxftk0f) 9

[2. Product Background](#_heading=h.1t3h5sf) 9

[3. Existing Systems](#_heading=h.4d34og8) 9

[3.1 System name:](#_heading=h.w022t17th3q9) 9

[4. Business Opportunity](#_heading=h.2s8eyo1) 9

[5. Software Product Vision](#_heading=h.17dp8vu) 9

[6. Project Scope & Limitations](#_heading=h.3rdcrjn) 10

[**II. Project Management Plan**](#_heading=h.26in1rg) **10**

[1. Overview](#_heading=h.lnxbz9) 10

[1.1 WBS & Estimation](#_heading=h.35nkun2) 10

[1.3 Project Risks](#_heading=h.44sinio) 11

[2. Management Approach](#_heading=h.s65jtdc2dz3b) 11

[2.1 Project Process](#_heading=h.4du1wux) 11

[2.2 Quality Management](#_heading=h.184mhaj) 12

[2.3 Training Plan](#_heading=h.m0aw20iolbef) 12

[3. Project Organization](#_heading=h.4i7ojhp) 12

[3.2 Role and Responsibilities](#_heading=h.yzorocet68ex) 12

[4. Project Communication](#_heading=h.2xcytpi) 13

[4.1 Communication Plan](#_heading=h.9kwlkjsdbbwc) 13

[a. FU Contacts](#_heading=h.dtx16r3hri3r) 13

[5. Configuration Management](#_heading=h.1ci93xb) 13

[5.1 Tools & Infrastructures](#_heading=h.3whwml4) 13

[**III. Software Requirement Specification**](#_heading=h.3as4poj) **14**

[1. Overall Description](#_heading=h.1pxezwc) 14

[1.1 Product Overview](#_heading=h.49x2ik5) 14

[1.2 Business Rules](#_heading=h.2p2csry) 15

[2. User Requirements](#_heading=h.147n2zr) 16

[2.1 Overview](#_heading=h.cwekxmykwi84) 16

[2.1.1 Use Case Diagram](#_heading=h.1entob1efnfk) 16

[2.1.2 System Actors](#_heading=h.hkweg2p4x2tz) 16

[2.1.3 Use Cases List](#_heading=h.s3q8bbuuuewj) 17

[2.2 Use Case Specifications](#_heading=h.9tecc3pfjs20) 18

[2.2.1 < User> Login](#_heading=h.r7siprfjqcau) 18

[2.2.2 < User> Logout](#_heading=h.3jrr117p36zd) 18

[2.2.3 < User> Create post](#_heading=h.yyd8w99ii2er) 19

[2.2.4 < User> Change status post](#_heading=h.n41h7ealyp05) 20

[2.2.5 <User, Admin> View list posts](#_heading=h.gshk0riqq7c4) 20

[2.2.6 < User, Admin> View detail post](#_heading=h.2dpho8yo6qs8) 21

[2.2.7 <User, Admin> Search post](#_heading=h.p0jnzv7gwbvd) 21

[2.2.8 <User, Admin> Comment on post](#_heading=h.qhmpcnklfe54) 22

[2.2.9 <Admin> Approve a post](#_heading=h.txqtekf8jr1e) 23

[2.2.10 <Admin> Reject a post](#_heading=h.4odbyfstgub2) 23

[2.2.11 <User> Update post](#_heading=h.wnneiouyx9ux) 24

[3. Functional Requirements](#_heading=h.utku04jcikaf) 25

[3.1 System Functional Overview](#_heading=h.23ckvvd) 25

[a. Screen Flow](#_heading=h.u8r676ij0t49) 25

[b. Non-Screen Functions](#_heading=h.dhwznxurfquc) 25

[c. Entity Relationship Diagram](#_heading=h.qnb5ydggie5k) 25

[4. Non-Functional Requirements](#_heading=h.32hioqz) 26

[4.1 External Interfaces](#_heading=h.1hmsyys) 26

[4.2 Quality Attributes](#_heading=h.41mghml) 26

[5. Other Requirements](#_heading=h.2grqrue) 27

[**IV. Software Design Description**](#_heading=h.vx1227) **28**

[1. Overall Description](#_heading=h.3fwokq0) 28

[1.1 Assumptions](#_heading=h.1v1yuxt) 28

[2. System Architecture Design](#_heading=h.19c6y18) 28

[2.1 Overall System Architecture](#_heading=h.3tbugp1) 28

[2.3 Package Diagram](#_heading=h.nmf14n) 29

[3. System Detailed Design](#_heading=h.37m2jsg) 29

[3.1 Class Diagram](#_heading=h.1mrcu09) 29

[3.2 Class Specification](#_heading=h.uxhvt8pexvi0) 30

[a. User](#_heading=h.xt00x4scsuyw) 30

[b. Post](#_heading=h.pjy5x091yssr) 30

[c. Comment](#_heading=h.ymb8yr2ljv7y) 31

[3.3 Sequence Diagrams](#_heading=h.9neo4gttcp74) 32

[a. Create Post](#_heading=h.islgqa4qa2lt) 32

[b. Search Post](#_heading=h.9w4dc2nrvoq3) 33

[c. Update Post](#_heading=h.p3ey5zf7rovg) 33

[d. Comment On Post](#_heading=h.zib6ue50ukau) 34

[4. Data & Database Design](#_heading=h.ccos3z6zy7ha) 34

[4.1 Database Design](#_heading=h.2lwamvv) 34

[4.1.1 Post](#_heading=h.z8sa3nynl2h) 35

[4.1.2 User](#_heading=h.2ciohcrh81ot) 36

[4.1.3 Role](#_heading=h.3fjy47e) 36

[4.1.4 Comment](#_heading=h.owlk1x1opy4g) 36

[**V. Software Testing Documentation**](#_heading=h.206ipza) **37**

[1. Overall Description](#_heading=h.4k668n3) 37

[1.1 Test Model](#_heading=h.2zbgiuw) 37

[1.2 Testing Levels](#_heading=h.1egqt2p) 37

[a. Unit testing](#_heading=h.kvmugb6hv3l4) 37

[b. Integration testing](#_heading=h.mt7onaqfee8x) 37

[c. System testing](#_heading=h.op6yd09v5sfx) 37

[1.3 Testing Types](#_heading=h.elbgojbdzqnv) 37

[2. Test Plan](#_heading=h.q7mfafvzlb7g) 38

[2.1 Test Stages](#_heading=h.sqyw64) 38

[2.2 Resources](#_heading=h.y5evjz2ndzfa) 38

[a. Human resources](#_heading=h.eknfwq1isyy9) 38

[b. Environment](#_heading=h.7v7ogglsn380) 38

[2.3 Test Milestones](#_heading=h.kzjmga1qr5sw) 39

[2.4 Deliverables](#_heading=h.bg5rkysa6q9q) 39

[3. Test Cases](#_heading=h.2r0uhxc) 39

[4. Test Reports](#_heading=h.1664s55) 40

[**VI. Release Package & User Guides**](#_heading=h.3q5sasy) **40**

[1. Deliverable Package](#_heading=h.25b2l0r) 40

[1.1 Source codes & documents](#_heading=h.u5cohb5fgav2) 40

[1.2 Known Issues, Limitations & Restrictions](#_heading=h.wd5ruhesyvxe) 41

[2. Installation Guides](#_heading=h.kgcv8k) 41

[2.1 System Requirements](#_heading=h.31ucoouilx5v) 41

[2.1.1 Hardware requirements](#_heading=h.ynhb4sb2cfwa) 41

[2.1.2 Software requirements](#_heading=h.fnt47v6im3mr) 41

[2.2 Setup Files](#_heading=h.t7wy7mnuvdwl) 41

[2.3 Installation Instruction](#_heading=h.yqsgx9ejlku6) 42

[a. NodeJS](#_heading=h.y388vj6t4aqy) 42

[3. User Manual](#_heading=h.vrzlxpmoaawl) 43

[3.1 Terms and definitions](#_heading=h.na8vj46w4zuw) 43

[3.2 System requirements](#_heading=h.43ky6rz) 44

[3.3 Application Usage](#_heading=h.2iq8gzs) 44

[3.3.1 Login](#_heading=h.m3uxtdkv6pre) 44

[3.3.1.1 Admin login](#_heading=h.rlxpyw12bmm2) 44

[3.3.1.2 User login](#_heading=h.sq61oj23ded1) 45

[3.3.2 Admin feature](#_heading=h.vcmv8968eopt) 48

[3.3.2.1 Admin view recent post feature](#_heading=h.cosy3g239xd6) 48

[3.3.2.2 Admin search post feature](#_heading=h.c1wi7ayscycr) 48

[3.3.2.3 Admin view a post feature](#_heading=h.fdde701p6k) 49

[3.3.2.4 Admin view pending post feature](#_heading=h.sgpz11wdaftz) 50

[3.3.2.5 Admin approve/reject a pending post feature](#_heading=h.d0285be3c2f) 51

[a. Approve](#_heading=h.2nshz985kncy) 51

[b. Reject](#_heading=h.wtq0w0xtktvj) 51

[3.3.2.6 Comment feature](#_heading=h.j3oagy1omalf) 52

[3.3.3 User feature](#_heading=h.u9j2qglixiyk) 53

[3.3.3.1 User view their own post feature](#_heading=h.tsvtahxxbj36) 53

[3.3.3.2 User view recent post feature](#_heading=h.wcjyf3a8nk96) 54

[3.3.3.3 View a post feature](#_heading=h.lyig0vlwla5n) 54

[3.3.3.4 User Search post feature](#_heading=h.heulmjpx9xls) 55

[3.3.3.5 Comment feature](#_heading=h.g8ib3edxnmjr) 55

[3.3.3.6 User Edit their own Post (only posts with "Waiting" status)](#_heading=h.ejunwgmx80pl) 57

[3.4 Troubleshooting](#_heading=h.xvir7l) 58

[**VII. Appendix**](#_heading=h.3hv69ve) **59**

[1. Glossary](#_heading=h.4hk9nu4j4o0h) 59

[2. References](#_heading=h.bted7nynq95) 59

[3. Message](#_heading=h.q25dkrhqov9x) 59

# List of Tables

[Table 1. Definitions and Acronyms 8](#_heading=h.z337ya)

[Table 2. Supervisors 9](#_heading=h.3j2qqm3)

[Table 3. Team members 9](#_heading=h.1y810tw)

[Table 4. WBS & Estimation 10](#_heading=h.2bn6wsx)

[Table 5. Project Risks 11](#_heading=h.qsh70q)

[Table 6. Training Plan 12](#_heading=h.3o7alnk)

[Table 7. Team and Structures 13](#_heading=h.ihv636)

[Table 8. Roles and Responsibilities 13](#_heading=h.4f1mdlm)

[Table 9. Communication Plan 13](#_heading=h.2u6wntf)

[Table 10. FU contacts 14](#_heading=h.4f55kduw9okw)

[Table 11. Tools & Infrastructures 14](#_heading=h.28h4qwu)

[Table 12. Business rules 15](#_heading=h.46r0co2)

[Table 13. System Actors 17](#_heading=h.3ygebqi)

[Table 14. Use Case List 18](#_heading=h.1rvwp1q)

[Table 15. <Use Case Specifications> Login 19](#_heading=h.2dlolyb)

[Table 16. <Use Case Specifications> Logout 19](#_heading=h.3cqmetx)

[Table 17. <Use Case Specifications> Create post 20](#_heading=h.4bvk7pj)

[Table 18. <Use Case Specifications> Change status post 21](#_heading=h.1jlao46)

[Table 19. <Use Case Specifications> View list posts 21](#_heading=h.1x0gk37)

[Table 20. <Use Case Specifications> View detail posts 22](#_heading=h.4h042r0)

[Table 21. <Use Case Specifications> Search post 22](#_heading=h.2w5ecyt)

[Table 22. <Use Case Specifications> Comment on post 23](#_heading=h.1baon6m)

[Table 23. <Use Case Specifications> Approve a post 24](#_heading=h.3vac5uf)

[Table 24. <Use Case Specifications> Reject a post 24](#_heading=h.2afmg28)

[Table 25. <Use Case Specifications> Update post 25](#_heading=h.pkwqa1)

[Table 26. <Class Specification> User 31](#_heading=h.2250f4o)

[Table 27. <Class Specification> Post 31](#_heading=h.haapch)

[Table 28. <Class Specification> Comment 32](#_heading=h.319y80a)

[Table 29. <Database Design> Post 36](#_heading=h.3ep43zb)

[Table 30. <Database Design> User 37](#_heading=h.1tuee74)

[Table 31. <Database Design> Role 37](#_heading=h.3s49zyc)

[Table 32. <Database Design> Comment 37](#_heading=h.1er0t5e)

[Table 33. Type of Test 39](#_heading=h.meukdy)

[Table 34. Human resources 39](#_heading=h.36ei31r)

[Table 35. Environment specifications 40](#_heading=h.1ljsd9k)

[Table 36. Test Milestones 40](#_heading=h.45jfvxd)

[Table 37. Deliverables 40](#_heading=h.2koq656)

[Table 38. Message 62](#_heading=h.zu0gcz)

# List of Firgures

[Figure 1. Scrum Framework 12](#_heading=h.2szc72q)

[Figure 2. Use Case Diagram 17](#_heading=h.111kx3o)

[Figure 3. <Screen Flow> Admin 25](#_heading=h.39kk8xu)

[Figure 4. <Screen Flow> User 25](#_heading=h.1opuj5n)

[Figure 5. Entity Relationship Diagram 26](#_heading=h.48pi1tg)

[Figure 6. Overall Architecture 28](#_heading=h.2nusc19)

[Figure 7. Package diagram 29](#_heading=h.1302m92)

[Figure 8. Class diagram 29](#_heading=h.3mzq4wv)

[Figure 9. <Sequence Diagram> Create post 32](#_heading=h.1gf8i83)

[Figure 10. <Sequence Diagram> Search post 33](#_heading=h.40ew0vw)

[Figure 11. <Sequence Diagram> Update post 33](#_heading=h.2fk6b3p)

[Figure 12. <Sequence Diagram> Comment on post 34](#_heading=h.fvife6y9mr1e)

[Figure 13. Database Design 35](#_heading=h.upglbi)

[Figure 14. Scrum Model 37](#_heading=h.279ka65)

# Acknowledgement

First of all, we would like to express our sincere and deepest gratitude to our supervisor, Mr. Nguyen Nguyen Binh, for giving us the precious opportunity to participate in this project. His guidance helped us in all the time of research, designing, building, and developing of this thesis. No matter day or night, as long as we have problems, her always ready to support us without complaining, once again we would like to sincerely thank you, Mr. Binh.

Besides our advisor, we owe deep thanks to Ms. Nguyen Thi Cam Huong for taking the valuable time to help us review this project. Furthermore, all of you also gave very sincere and accurate suggestions to help us realize where the problem is and what we should do to make this project more complete.

We would like to thank each individual in our group for our enthusiasm, hard-working, dedication and non-stop effort. Sometimes our group fell into a state of deadlock, stress, and fatigue, but luckily, we all tried to overcome them to reach today. Special thanks to Tran Thien Quy for being a great leader, always being the one who giving the best effort and always tolerating group members' mistakes. Thank you Tran Vinh Phat for your research ability and expertise. Much appreciation to Vo Quoc Viet and Pham Nguyen Hoai Linh for his silent contributions and for always actively taking care of other members. Thank you Vu Lam Bao Van for not giving up and always trying to care about the smallest things.

Finally, we would like to extend our gratitude to our families and friends who have provided emotional and practical support for this project. Thanks for everything all of you give us, we will forever be engraved in our hearts.

# Definition and Acronyms

*Table 1. Definitions and Acronyms*

| **Acronym** | **Definition** |
| --- | --- |
| LAF | Lost and Found |
| BR | Business Rule |
| ERD | Entity Relationship Diagram |
| GUI | Graphical User Interface |
| PM | Project Manager |
| SDD | Software Design Description |
| SPMP | Software Project Management Plan |
| SRS | Software Requirement Specification |
| UC | Use Case |
| API | Application Program Interface |
| MSG | Message |

# I. Project Introduction

## 1. Overview

### 1.1 Project Information

* Project name: FPTU Lost And Found
* Group name: LAF Team
* Software type: Web App

### 1.2 Project Team

#### a. Supervisors

*Table 2. Supervisors*

| **Full Name** | **Email** | **Phone Number** | **Title** |
| --- | --- | --- | --- |
| Nguyen Nguyen Binh | BinhNN7@fe.edu.vn |  | Mr. |

#### b. Team Members

*Table 3. Team members*

| **Full Name** | **Email** | **Mobile** | **Role** |
| --- | --- | --- | --- |
| Tran Thien Quy | quyttse63199@fpt.edu.vn | 033.5657.532 | Member |
| Vu Lam Bao Van | vanvlbse130253@fpt.edu.vn | 077.7055.884 | Member |
| Pham Nguyen Hoai Linh | linhpnhse63552@fpt.edu.vn |  | Member |
| Tran Vinh Phat | phattvse130678@fpt.edu.vn |  | Member |
| Vo Quoc Viet | vietvqse63467@fpt.edu.vn |  | Member |

## 2. Product Background

In the current situation lost and dropped items in school is more and more common. Such as lost keys, lost car cards, forgot laptop chargers, forgot things,... With such a situation, there are communication channels to help us find such things as FPTU confession, or contact the school's janitors or guards directly. However, those ways are not always quick and convenient, it will take a lot of time to find lost items through the aunts and uncles at school. Or through the FPTU Confession website, there is too much information, which will make it more difficult to find items, not always items can be posted immediately. Therefore, FPTU Lost and Found was born to help us find things quickly, conveniently, and extremely suitable for students at FPT.

## 3. Existing Systems

#### 3.1 System name:

https://www.facebook.com/FPTUHCMConfessions

Advantage:

* Information channel of FPT school
* The web helps you find things so far
* A lot of students follow

Disadvantage:

* Too much information in one day
* Can't find the item quickly
* If you want to post, you have to wait too long for approval
* Not everyone can know how to post

## 4. Business Opportunity

Currently, the current map search pages are not available at FPT school. Therefore, FPTU Lost and Found were built to provide information, a place to post lost items, find lost items, and help post quickly, so the app was born.

## 5. Software Product Vision

FPTU Lost and Found app focuses on finding things, so it helps users to find information about their items accurately and post quickly. In addition, we will aggregate lost items according to a reasonable and convenient catalogue so that everyone can find them.

## 6. Project Scope & Limitations

N/A

# II. Project Management Plan

## 1. Overview

### 1.1 WBS & Estimation

*Table 4. WBS & Estimation*

| **#** | **WBS Item** | **Complexity** | **Est. Effort** |
| --- | --- | --- | --- |
| ***1*** | ***Specification*** |  |  |
| 1.1 | Project Planning | Medium |  |
| 1.2 | Deliver Report 1 - Project Introduction | Simple |  |
| 1.3 | Deliver Report 2 - Project Management Plan | Simple |  |
| 1.4 | Deliver Report 3 - System Requirement Specification | Simple |  |
| ***2*** | ***Design*** |  |  |
| 2.1 | Design User Interfaces (UI) prototypes | Medium |  |
| 2.2 | Deliver Report 4 - Software Design Document | Medium |  |
| ***3*** | ***Training*** |  |  |
| 3.1 | Learn backend, frontend and frameworks | Medium |  |
| 3.2 | Learn to use branching model | Simple |  |
| ***4*** | ***Implement*** |  |  |
| ***4.1*** | ***Common functions*** |  |  |
| 4.1.1 | Login | Simple |  |
| 4.1.2 | Logout | Simple |  |
| 4.1.3 | View list post | Simple |  |
| 4.1.4 | View detail a post | Simple |  |
| 4.1.5 | Search post | Simple |  |
| 4.1.6 | Comment on post | Medium |  |
| ***4.2*** | ***Web Application for Admin*** |  |  |
| 4.2.1 | Approve a post | Simple |  |
| 4.2.2 | Reject a post | Simple |  |
| ***4.3*** | ***Web Application for User*** |  |  |
| 4.3.1 | Create a lost post | Simple |  |
| 4.3.2 | Change status a post | Simple |  |
| 4.3.3 | Update post | Simple |  |

### 1.3 Project Risks

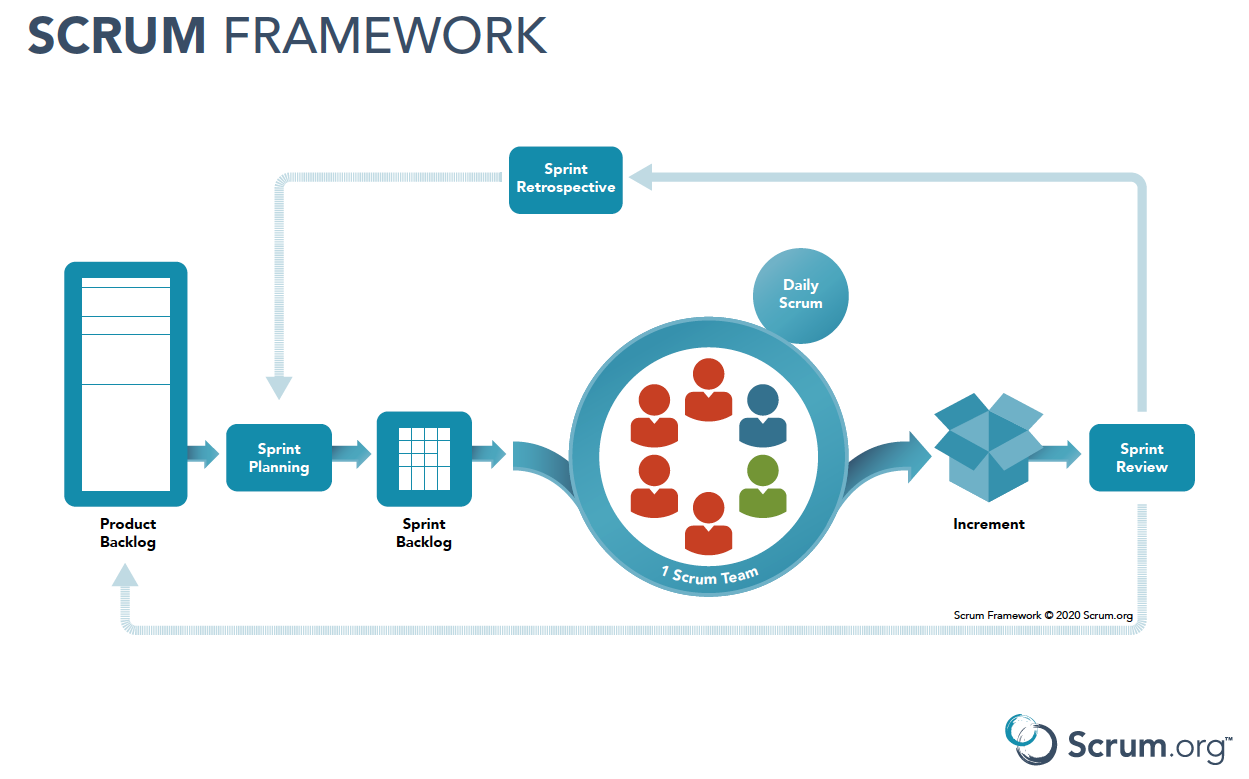
*Table 5. Project Risks*

| **#** | **Risk Description** | **Impact** | **Possibility** | **Response Plans** |
| --- | --- | --- | --- | --- |
| 1 | Team member drops out | Critical | Low | Reduce scope |
| 2 | Team member gets COVID-19 | Critical | Low | Reduce scope |
| 3 | Host server die | Medium | Low | Change host server |
| 4 | Hardware devices get broken | Medium | Medium | Spare devices |
| 5 | Team communications is disrupted due to COVID-19 Pandemic | Medium | High | Use online tools for communication and task tracking |

## 2. Management Approach

Implementing Scrum as an Agile methodology enables IPSP team under a common organization for decision making, allowing us to see products evolve in real time as we build and adjust them.

### 2.1 Project Process



*Figure 1. Scrum Framework*

LAF team apply Scrum Framework for managing project process for the following reasons:

* It’s suitable for small or medium size project as LAF System.
* Project processes are done by the development team simultaneously, that help everything to be flexible and changeable during the life of the project and even after.
* Better visibility into project and minimize risks.

### 2.2 Quality Management

To improve the project quality, LAF team apply the following schemes:

* Following available application’s design.
* Cohering supervisors’ guidance and researching to achieve the best solution.
* Holding daily meeting to follow project’s tasks and support each other.

### 2.3 Training Plan

*Table 6. Training Plan*

| **Training Area** | **Participants** | **When, Duration** | **Waiver Criteria** |
| --- | --- | --- | --- |
| NodeJS Express | Everyone | 3 days | Mandatory |
| ReactJS | Everyone | 3 days | Mandatory |
| AntDesign | Everyone | 3 days | Mandatory |
| MongoDB | Everyone | 1 day | Mandatory |
| Git, Github | Everyone | 1 day | Mandatory |
| Draw.io | Everyone | 1 day | Mandatory |

## 3. Project Organization

**3.1 Team and Structures**

*Table 7. Team and Structures*

| **Role** | **Full name** |
| --- | --- |
| Project Mentor | Nguyen Nguyen Binh |
| Technical Master | Tran Thien Quy |
| Technical Members | Vu Lam Bao Van  Pham Nguyen Hoai Linh  Tran Vinh Phat  Vo Quoc Viet |

### 3.2 Role and Responsibilities

*Table 8. Roles and Responsibilities*

| **Role** | **Responsibilities** |
| --- | --- |
| Project Mentor | Help team finish the project with advice on business and technical issues |
| Technical Master | Help members to finish their technical task on time and decide the technical solution for the issue |
| Technical Members | Implement the task |

## 4. Project Communication

### 4.1 Communication Plan

*Table 9. Communication Plan*

| **Communitication** | **Who/Target** | **Purpose** | **When,Frequency** | **Type,Tool,Method** |
| --- | --- | --- | --- | --- |
| Messenger Texting | Team Members | Internal conversations, internal reports between members | Always | Messenger |
| Meeting offline | Team Members + Supervisor | Process Reports, Show Demo, Report Problem and receive reviews from Supervisor | Always | On Classroom |
| Meeting Online | Team Members | Process Reports, Show Demo, Report Problem and receive reviews from Supervisor | Always | Google Meet |

**4.2 External Interface**

#### a. FU Contacts

*Table 10. FU contacts*

| **Function** | **Contact Person**  **(name, position)** | **Contact address**  **(email, telephone)** | **Responsibility** |
| --- | --- | --- | --- |
| Supervisor | Nguyễn Nguyên Bình | BinhNN7@fe.edu.vn | - Provide document template - Give instruction to project team - Review deliverables - Supervise project status  - Receive project reports  - Answer questions about the project |

## 5. Configuration Management

### 5.1 Tools & Infrastructures

*Table 11. Tools & Infrastructures*

| **Programming languages** | NodeJS Express, ReactJS |
| --- | --- |
| **Framework** | AntDesign |
| **DBMS** | MongoDB |
| **IDEs/Editors** | Visual Code |
| **UML tools** | Draw.io |
| **Version Control** | Github |
| **Project management tool** | Google Drive, Excel |

**5.2. Document Management**

Google Drive is used to manage project’s documents.

**5.3. Source Code Management**

Git is used to manage project’s source code.

# III. Software Requirement Specification

## 1. Overall Description

### 1.1 Product Overview

LostAndFound.com is a new website that helps students in the school to find their lost or lost items. The website was designed with new languages which are NodeJS for a website. The website also uses google APIs such as Google Mail, Google Firebase Cloud Messaging, and Google Firebase Storage to support user experience while using an application LostAndFound.

**1.1.1. User**

A user who is the one who lost or finds an item

* Login
* Logout
* Create a post
* Change status a post
* View list post
* View detail a post
* Search item
* Comment on post

**1.1.2. Admin**

* Login
* Logout
* View list post
* View detail a post
* Search item
* Comment on post
* Approve a post
* Reject a post

### 1.2 Business Rules

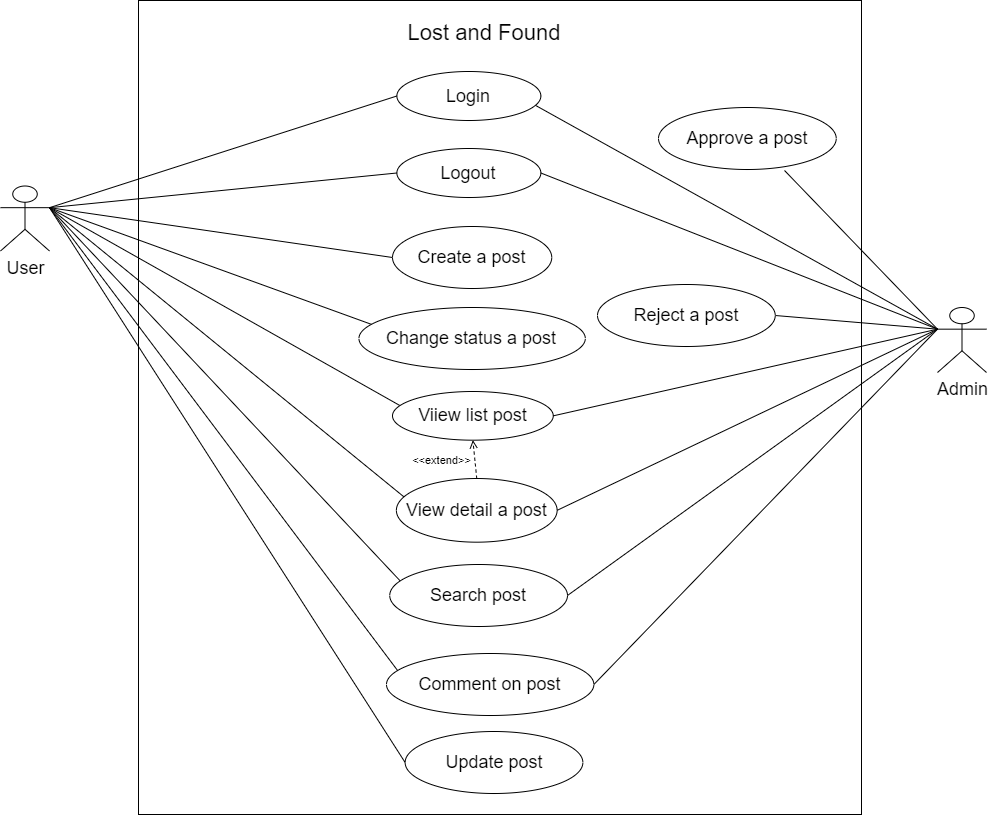
*Table 12. Business rules*

| ID | Rule Definition |
| --- | --- |
| BR-01 | Email can only be logged in by school account |
| BR-02 | Password must be greater than 6 characters |
| BR-03 | Username must be greater than 4 characters |
| BR-04 | The post can be hidden by admin |
| BR-05 | The post name cannot be duplicated |
| BR-06 | Max length of a topic name is 200 characters |
| BR-07 | Min length of a topic is 10 characters |
| BR-08 | Only admins handle reports |
| BR-09 | Change status when the post has been approved |
| BR-10 | Only administrator can create policy |
| BR-11 | The lost item must be verified before the lost person can get it back. |
| BR-12 | The user must exist in the system |
| BR-13 | The Maximum length of the search text field is 125 characters. |
| BR-14 | Account must be on the system |
| BR-15 | Users cannot delete a post once it's posted |
| BR-16 | Users cannot edit post once it's posted |
| BR-17 | One post has maximum 50mb pictures |
| BR-18 | Image limited 50mb |
| BR-19 | Every post on the web must be of these 3 statuses: “Waiting”, “Approved”, “Found” |
| BR-20 | Posts cannot be deleted from the system, but the posted information of the post “Found” will be displayed |
| BR-21 | Lost items will be posted on the website within 30 days. If no one accepts it, it will be removed. |

## 2. User Requirements

### 2.1 Overview

#### 2.1.1 Use Case Diagram



*Figure 2. Use Case Diagram*

#### 2.1.2 System Actors

*Table 13. System Actors*

| **#** | **Actor** | **Description** |
| --- | --- | --- |
| 1 | User | User is the one lost or find an item and has email account belonging to FPT University |
| 2 | Admin | Admin can process approve, reject a post |

#### 2.1.3 Use Cases List

*Table 14. Use Case List*

| **ID** | **Use Case** | **Primary Actors** | **Secondary Actors** |
| --- | --- | --- | --- |
| UC01 | Login | User, Admin | N/A |
| UC02 | Logout | User, Admin | N/A |
| UC03 | Create a post | User | N/A |
| UC04 | Change status a post | User | N/A |
| UC05 | View list post | User | N/A |
| UC06 | View detail a post | User | N/A |
| UC07 | Search post | User | N/A |
| UC08 | Comment on post | User | N/A |
| UC09 | Update post | User | N/A |
| UC10 | View list post | Admin | N/A |
| UC11 | View detail a post | Admin | N/A |
| UC12 | Search post | Admin | N/A |
| UC13 | Comment on post | Admin | N/A |
| UC14 | Approve a post | Admin | N/A |
| UC15 | Reject a post | Admin | N/A |

### 

### 2.2 Use Case Specifications

#### 2.2.1 < User> Login

*Table 15. <Use Case Specifications> Login*

| ID and Name: | **UC-01 Login** | | |
| --- | --- | --- | --- |
| Created By: | VanVLB | Date Created: | 18/07/2022 |
| Primary Actor: | User, Admin | Secondary Actors: | N/A |
| Description: | User login to LAF to verify role, give them access to more features in the system. | | |
| Trigger: | User request login into LAF. | | |
| Pre-conditions: | PRE-01. User is logged in. | | |
| Post-conditions: | POST-01.   * Success: LAF login for user and shows success messages. * Fail: LAF shows error messages. | | |
| Normal Flow: | 1. **Login into IPSB** **System** (Exception 1.0.E3) 2. User send request to log into the LAF system. 3. User fill to login form:  * Email: text input, blank * Password: text input, blank  1. User click to “Login” button. 2. LAF validate the email, password (Exception 1.0.E1, E2) 3. LAF determines user’s identity, role and permission then redirects user to home page and shows message. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | 1. **E1 User fails to fill in all the required fields** 2. LAF shows messages MSG-001, MSG-002 | | |
| Priority: | Medium. | | |
| Frequency of Use: | Sometimes. | | |
| Business Rules: | BR-01, BR-02, BR-03, BR-10 | | |
| Other Information: | In case of internet connection failure, User cannot login into LAF. | | |
| Assumptions: | N/A | | |

#### 2.2.2 < User> Logout

*Table 16. <Use Case Specifications> Logout*

| ID and Name: | **UC-02 Logout** | | |
| --- | --- | --- | --- |
| Created By: | VanVLB | Date Created: | 18/07/2022 |
| Primary Actor: | User | Secondary Actors: | N/A |
| Description: | This feature allows user log out from the LAF system | | |
| Trigger: | The actor sends logout request. | | |
| Pre-conditions: | Success: User will be redirected to login screen of the application. | | |
| Post-conditions: | POST-01.   * Success: LAF logged out user and shows success messages. * Fail: LAF shows error messages. | | |
| Normal Flow: | **2.0 Logout the LAF System** (Exception 2.0.E1)   1. User clicks the “Logout” button. 2. LAF System redirects to login screen and delete user session. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | N/A | | |
| Priority: | Medium. | | |
| Frequency of Use: | Sometimes. | | |
| Business Rules: | BR-01, BR-10 | | |
| Other Information: | In case of internet connection failure, User cannot log out of LAF. | | |
| Assumptions: | N/A | | |

#### 2.2.3 < User> Create post

*Table 17. <Use Case Specifications> Create post*

| ID and Name: | **UC-03 Create post** | | |
| --- | --- | --- | --- |
| Created By: | VanVLB | Date Created: | 18/07/2022 |
| Primary Actor: | User | Secondary Actors: | N/A |
| Description: | This feature allows user create a new post in system | | |
| Trigger: | The actor sends request to create post. | | |
| Pre-conditions: | PRE-01. User has logged in the web as user role.  PRE-02. Current screen is “Post”. | | |
| Post-conditions: | POST-01:   * Success: LAF create new post and shows success messages. * Fail: LAF shows error messages. | | |
| Normal Flow: | **3.0 Create post**(Exception 3.0.E3)   1. User clicks Create button icon to send request to create post. 2. LAF will show a screen with nine fields:  * Title of post: text input, length 10 - 200 * Location: text input, length 1-100 * Image: file picker, limited 50mb * Phone number : text input, length 1-10 * Lost: text input, length 1-100 * Found: text input length 1-100 * Address: text input length 1-100 * Identity mark : text input length 1-100 * Secret information : text input 1-200  1. User fills and click “Upload” button. 2. LAF create new post and shows message MSG06. (Exception 3.0.E1, 3.0.E2) | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **3.0.E1 Missing of required fields**   1. LAF shows messages MSG-005, MSG-006, MSG-007, MSG-008. | | |
| Priority: | Medium. | | |
| Frequency of Use: | Sometimes. | | |
| Business Rules: | BR-05, BR-06, BR-07, BR-08, BR-10, BR-16, BR-17 | | |
| Other Information: | In case of internet connection failure, User create a new post. | | |
| Assumptions: | N/A | | |

#### 2.2.4 < User> Change status post

*Table 18. <Use Case Specifications> Change status post*

| ID and Name: | **UC-04 Change status post** | | |
| --- | --- | --- | --- |
| Created By: | VanVLB | Date Created: | 18/07/2022 |
| Primary Actor: | User | Secondary Actors: | N/A |
| Description: | This feature allows user change status of post | | |
| Trigger: | The actor sends request to change status of post. | | |
| Pre-conditions: | PRE-01. User has logged in the web as user role.  PRE-02. Current screen is “Post”. | | |
| Post-conditions: | POST-01:   * Success: LAF change status of post * Fail: LAF shows error messages. | | |
| Normal Flow: | **4.0 Change status post**(Exception 4.0.E3)   1. User clicks Change button icon to send request to change status of post. 2. LAF will click “Close post”. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **4.0.E1 Missing of required fields**   1. LAF shows messages MSG01, MSG03 , MSG04, MSG05.   **4.0.E2 Input’s length is larger than specified**   1. LAF shows messages MSG02.   **4.0.E3 Internet connection lost**   1. LAF forwards to the Lost Internet page. | | |
| Priority: | Medium. | | |
| Frequency of Use: | Sometimes. | | |
| Business Rules: | BR-10, BR-18 | | |
| Other Information: | In case of internet connection failure, User can not change status of post. | | |
| Assumptions: | N/A | | |

#### 2.2.5 <User, Admin> View list posts

*Table 19. <Use Case Specifications> View list posts*

| ID and Name: | **UC-05 View list post** | | |
| --- | --- | --- | --- |
| Created By: | VanVLB | Date Created: | 18/07/2022 |
| Primary Actor: | User, Admin | Secondary Actors: | N/A |
| Description: | This feature allows user to get all to view . | | |
| Trigger: | The actor sends request to get list of posts | | |
| Pre-conditions: | PRE-01. User has logged in the web as user role. | | |
| Post-conditions: | POST-01:   * Success: LAF shows list of posts * Fail: LAF shows error messages. | | |
| Normal Flow: | **5.0 View list posts**(Exception 5.0.E3)   1. User logged LAF will shows the list on page | | |
| Alternative Flows: | N/A | | |
| Exceptions: | N/A | | |
| Priority: | Medium. | | |
| Frequency of Use: | Sometimes. | | |
| Business Rules: | BR-10 | | |
| Other Information: | In case of internet connection failure.User can not view posts. | | |
| Assumptions: | N/A | | |

#### 2.2.6 < User, Admin> View detail post

*Table 20. <Use Case Specifications> View detail posts*

| ID and Name: | **UC-06 View detail post** | | |
| --- | --- | --- | --- |
| Created By: | VanVLB | Date Created: | 18/07/2022 |
| Primary Actor: | User | Secondary Actors: | N/A |
| Description: | This feature allows user to get view detail of post . | | |
| Trigger: | The actor sends request to get detail of post | | |
| Pre-conditions: | PRE-01. User has logged in the web as user role. | | |
| Post-conditions: | POST-01:   * Success: LAF shows detail of posts * Fail: LAF shows error messages. | | |
| Normal Flow: | **6.0 View list posts**(Exception 6.0.E3)   1. User logged LAF will shows the list on page | | |
| Alternative Flows: | N/A | | |
| Exceptions: | N/A | | |
| Priority: | Medium. | | |
| Frequency of Use: | Sometimes. | | |
| Business Rules: | BR-10 | | |
| Other Information: | In case of internet connection failure, User can not view detail post. | | |
| Assumptions: | N/A | | |

#### 2.2.7 <User, Admin> Search post

*Table 21. <Use Case Specifications> Search post*

| ID and Name: | **UC-07 Search post** | | |
| --- | --- | --- | --- |
| Created By: | VietVQ | Date Created: | 18/07/2022 |
| Primary Actor: | User, Admin | Secondary Actors: | N/A |
| Description: | This feature allows the user or admin to search post items in LAF. | | |
| Trigger: | The actor sends a request to search for a post item. | | |
| Pre-conditions: | PRE-01. User has logged in the web as user role.  PRE-02. Current screen is “View recent item”. | | |
| Post-conditions: | POST-01: Success: Show list item recent by key search. | | |
| Normal Flow: | **7.0 Search post item**(Exception 7.0.E1)   1. User clicks the 'View recent item’ button to send a request to the search post. 2. Enter key search to textfield 3. Click button ‘Search’ 4. LAF show list post recent item. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | N/A | | |
| Priority: | Medium. | | |
| Frequency of Use: | Sometimes. | | |
| Business Rules: | BR-10, BR-11 | | |
| Other Information: | In case of internet connection failure, users can not view search posts. | | |
| Assumptions: | N/A | | |

#### 2.2.8 <User, Admin> Comment on post

*Table 22. <Use Case Specifications> Comment on post*

| ID and Name: | **UC-08 Comment on post** | | |
| --- | --- | --- | --- |
| Created By: | VietVQ | Date Created: | 18/07/2022 |
| Primary Actor: | User, Admin | Secondary Actors: | N/A |
| Description: | This feature allows user comment on post in system | | |
| Trigger: | The actor sends a request to comment on the post. | | |
| Pre-conditions: | PRE-01. User has logged in the web as user role.  PRE-02. Current screen is “View Recent item”.  PRE-03. Current screen is view detail post. | | |
| Post-conditions: | POST-01:   * Success: LAF create new comment and show success messages. * Fail: LAF shows error messages. | | |
| Normal Flow: | **8.0 Comment on post**(Exception 8.0.E1)   1. User fills and click “Comment” button 2. LAF create new comment on post and shows message MSG009. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | N/A | | |
| Priority: | Medium. | | |
| Frequency of Use: | Sometimes. | | |
| Business Rules: | BR-10 | | |
| Other Information: | In case of internet connection failure, users can not comment on post. | | |
| Assumptions: | N/A | | |

#### 2.2.9 <Admin> Approve a post

*Table 23. <Use Case Specifications> Approve a post*

| ID and Name: | **UC-09 Approve a post** | | |
| --- | --- | --- | --- |
| Created By: | VietVQ | Date Created: | 18/07/2022 |
| Primary Actor: | Admin | Secondary Actors: | N/A |
| Description: | This feature allows admin to approve posts of users in LAF. | | |
| Trigger: | The actor sends a request to approve a post of the user. | | |
| Pre-conditions: | PRE-01. User has logged in the web as an admin role.  PRE-02. Current screen is “View Recent item”. | | |
| Post-conditions: | POST-01:   * Success: The post of the user will appear on ‘View Recent item’. * Fail: LAF shows error messages. | | |
| Normal Flow: | **9.0** **Approve the pending post of user**(Exception 9.0.E1)  1. Admin click checkbox ‘Show the pending item’.  2. Admin click view detail post  3. Admin click button “Approve”. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **9.0.E1 Internet connection lost**   1. LAF forwards to the Lost Internet page. | | |
| Priority: | Medium. | | |
| Frequency of Use: | Sometimes. | | |
| Business Rules: | BR-10, BR-17 | | |
| Other Information: | In case of internet connection failure, approve post failure. | | |
| Assumptions: | N/A | | |

#### 2.2.10 <Admin> Reject a post

*Table 24. <Use Case Specifications> Reject a post*

| ID and Name: | **UC-010 Reject a post** | | |
| --- | --- | --- | --- |
| Created By: | VietVQ | Date Created: | 18/07/2022 |
| Primary Actor: | User | Secondary Actors: | N/A |
| Description: | This feature allows admin to reject posts of users in LAF. | | |
| Trigger: | The actor sends a request to reject a post of the user. | | |
| Pre-conditions: | PRE-01. User has logged in the web as an admin role.  PRE-02. Current screen is “View Recent item”. | | |
| Post-conditions: | POST-01:   * Success: The post of the user cannot appear on ‘View Recent item’. * Fail: LAF shows error messages. | | |
| Normal Flow: | **10.0** **Reject the pending post of user**(Exception 10.0.E1)  1. Admin click checkbox ‘Show the pending item’.  2. Admin click view detail post  3. Admin click button “Reject”. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **10.0.E1 Internet connection lost**   1. LAF forwards to the Lost Internet page. | | |
| Priority: | Medium. | | |
| Frequency of Use: | Sometimes. | | |
| Business Rules: | BR-10, BR-17 | | |
| Other Information: | In case of internet connection failure, the post is still pending. | | |
| Assumptions: | N/A | | |

#### 2.2.11 <User> Update post

*Table 25. <Use Case Specifications> Update post*

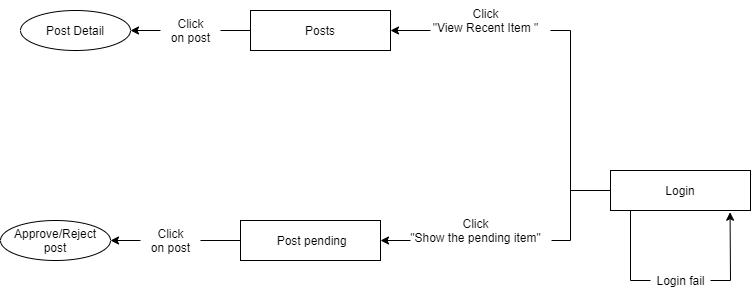
| ID and Name: | **UC-011 Update post** | | |
| --- | --- | --- | --- |
| Created By: | VietVQ | Date Created: | 18/07/2022 |
| Primary Actor: | User | Secondary Actors | N/A |
| Description: | This feature allows user update item posted | | |
| Trigger: | The actor sends request to get list of posts | | |
| Pre-conditions: | PRE-01. User has logged in the web as user role.  PRE-02. LAF created a post and shows success messages. | | |
| Post-conditions: | POST-01:   * Success: The posted will be update * Fail: LAF shows error messages. | | |
| Normal Flow: | **11. 0 Update post**(Exception 11.0.E1)   1. User clicks the “Update” button icon to send a request to update the post. 2. LAF will show a screen allow updating some fields:   ·         Title of post:  text input, length 10 - 200  ·         Location: text input, length 1-100  ·         Image:  file picker, limited 50mb  ·         Phone number : text input, length 1-10  ·         Lost: text input, length 1-100  ·         Found: text input length 1-100  ·         Address: text input length 1-100  ·         Identity mark : text input length 1-100  ·         Secret information : text input 1-200   1. User changes the value and clicks the “Upload” button.   LAF update post and show message MSG11. | | |
| Alternative Flows: | N/A | | |
| Exceptions: | **11.0.E1 Internet connection lost**   1. LAF  forwards to the Lost Internet page. | | |
| Priority: | Medium. | | |
| Frequency of Use: | Sometimes. | | |
| Business Rules: | BR-05, BR-06. BR-07, BR-10, BR-15, BR-16, | | |
| Other Information: | In case of internet connection failure,  users can not update posts. | | |
| Assumptions: | N/A | | |

## 3. Functional Requirements

### 3.1 System Functional Overview

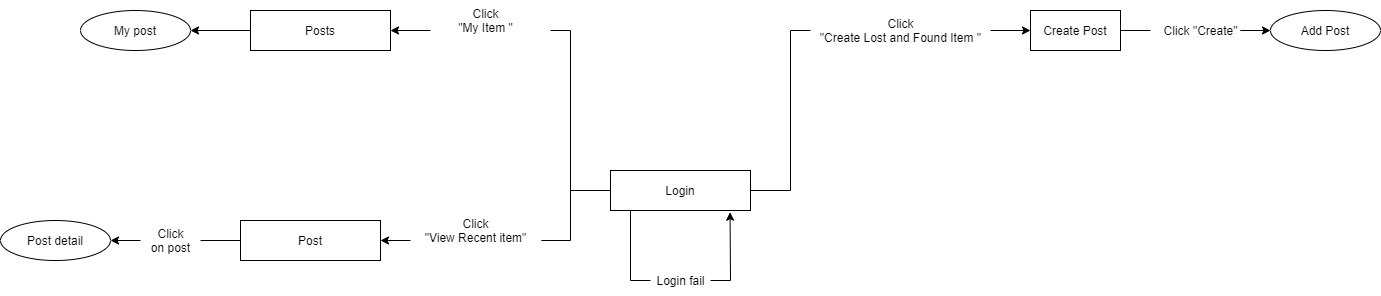
#### a. Screen Flow

* **Admin**

****

*Figure 3. <Screen Flow> Admin*

* **User**

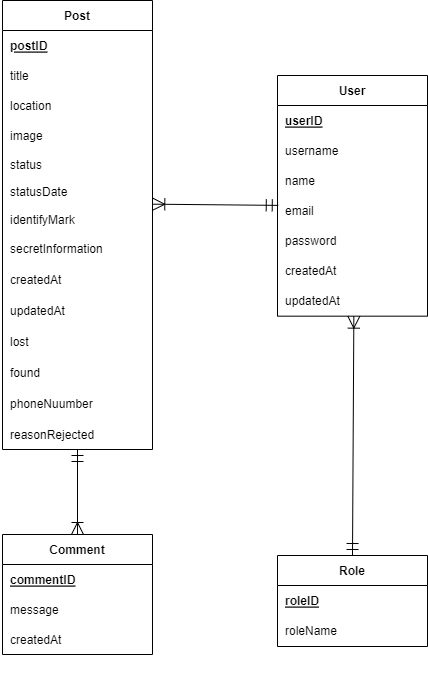
****

*Figure 4. <Screen Flow> User*

#### b. Non-Screen Functions

N/A

#### c. Entity Relationship Diagram



*Figure 5. Entity Relationship Diagram*

## 4. Non-Functional Requirements

### 4.1 External Interfaces

System will communicate properly with users and with external hardware or software elements. Specifically, the project was created for 3 objects: admin, lost and recovered. The software is built to run on computers with a network connection, and users need an FTP email account to be able to access and use the functions.

### 4.2 Quality Attributes

* Security: The system is well secured with its own organization email
* Performance: fast loading ability
* Usability and Reliability: Friendly interface, easy to understand, easy to use
* Availability: ready to access from anywhere with an internet connection

## 5. Other Requirements

N/A

# IV. Software Design Description

## 1. Overall Description

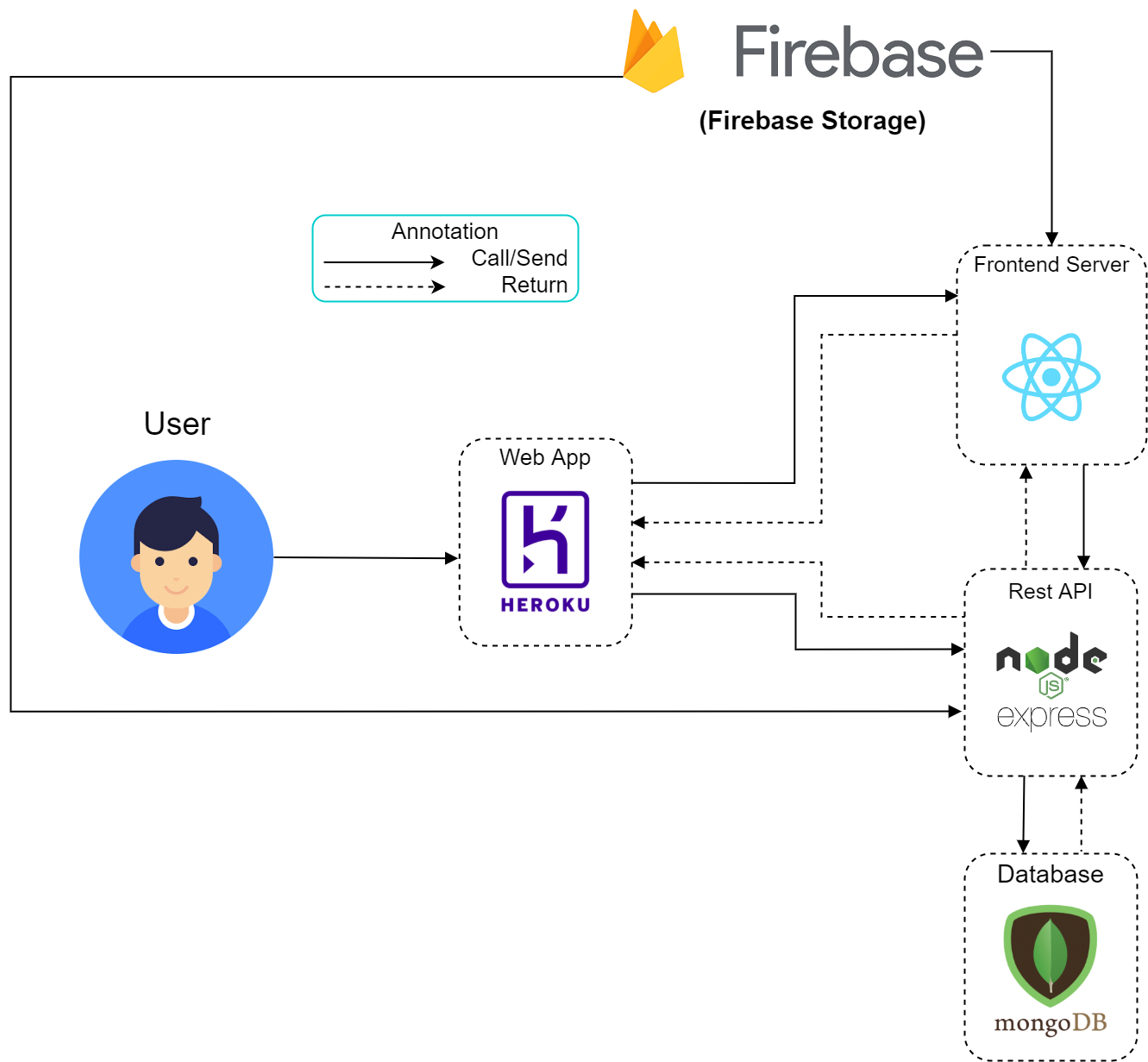
### 1.1 Assumptions

This system is designed basing on these following assumptions:

* Windows 10
* Chrome
* MongoDB
* End-User: FPT University Students, Admin
* Security requirement: devices with confidential information to avoid accidental recipients (eg: product series, identification)
* Network communication: This system depends on networks connection

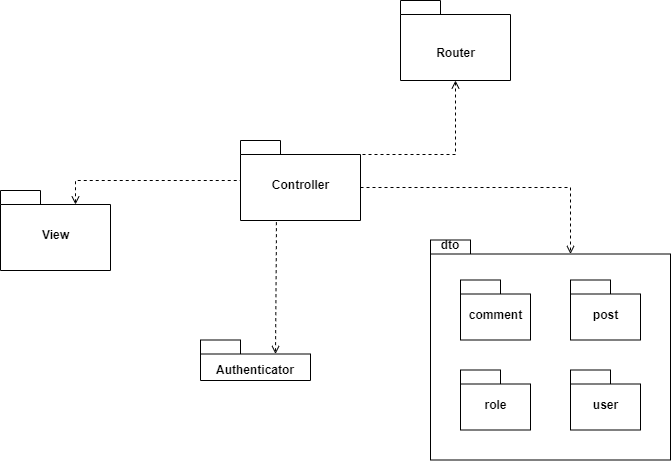
## 2. System Architecture Design

### 2.1 Overall System Architecture



*Figure 6. Overall Architecture*

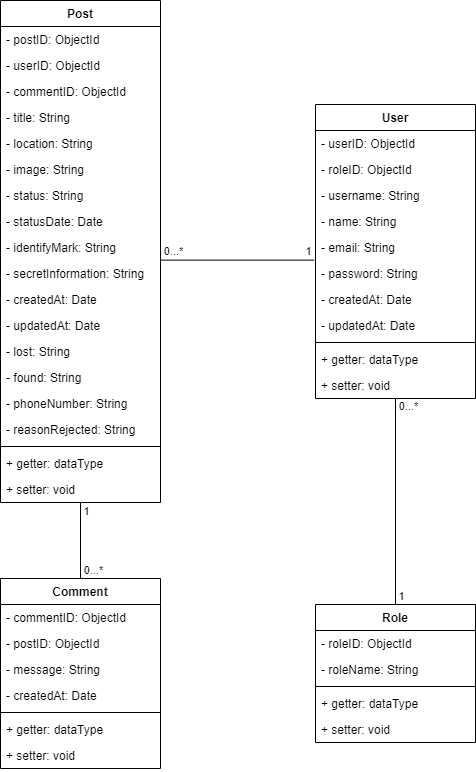
### 2.3 Package Diagram



*Figure 7. Package diagram*

## 3. System Detailed Design

### 3.1 Class Diagram



*Figure 8. Class diagram*

### 3.2 Class Specification

#### a. User

*Table 26. <Class Specification> User*

| **Field name** | **Type** | **Description** |
| --- | --- | --- |
| userID | int | Unique identifier of user |
| roleID | int | The identifier of role |
| userName | string | Contains username information of user |
| email | string | Unique email of user |
| password | string | Contains password information of user |
| gender | string | Contains gender information of user |
| phoneNumber | string | Contains phone number information of user |

| **Method** | **Return type** | **Description** |
| --- | --- | --- |
| get | data type | Get attribute value |
| set | void | Set value of attribute |

#### b. Post

*Table 27. <Class Specification> Post*

| **Field name** | **Type** | **Description** |
| --- | --- | --- |
| postID | ObjectId | Unique identifier of user |
| userID | ObjectId | Contains username information of user |
| commentID | ObjectId | Contains identifier of comment |
| title | String | Contains title of post |
| location | String | Contains location of item |
| image | String | Contains image of item |
| status | String | Contains status of post |
| statusDate | Date | Contains status date of post |
| identificationMark | String | Contains identification mark of item |
| secretInformation | String | Contains secret information of item |
| createdAt | Date | Contains date when the post created |
| updatedAt | Date | Contains date when the post updated |
| lost | String | Contains type lost post |
| found | String | Contains type found post |
| phoneNumber | String | Contains phone number of user |
| reasonRejected | String | Contains reason reject post |

| **Method** | **Return type** | **Description** |
| --- | --- | --- |
| get | data type | Get attribute value |
| set | void | Set value of attribute |

#### c. Comment

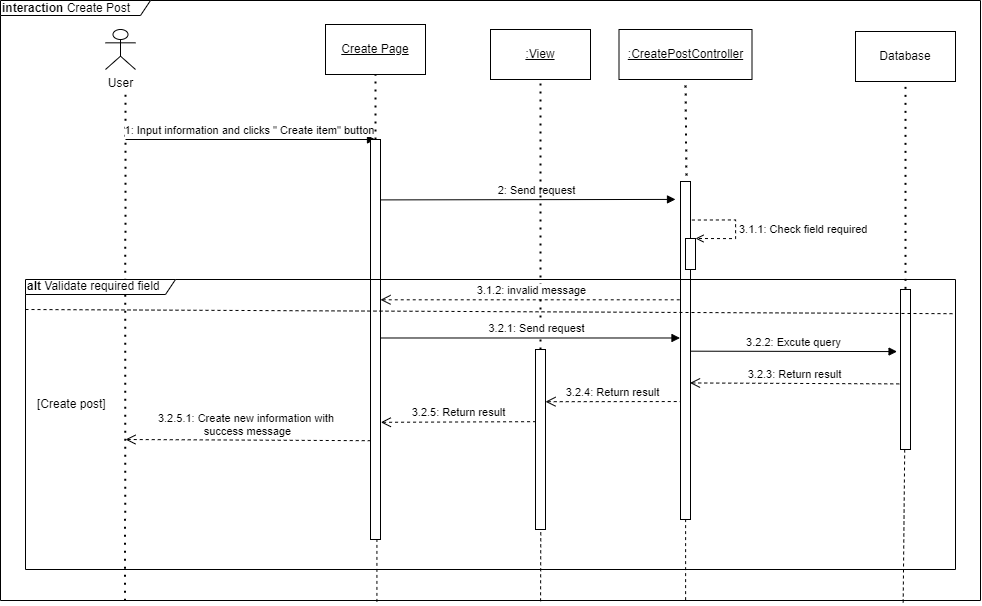
*Table 28. <Class Specification> Comment*

| **Field name** | **Type** | **Description** |
| --- | --- | --- |
| commentID | ObjectId | Unique identifier of comment |
| postID | ObjectId | Contain identifier of comment |
| message | String | Contains message of comment |
| timeComment | Date | Contains time comment of comment |

| **Method** | **Return type** | **Description** |
| --- | --- | --- |
| get | data type | Get attribute value |
| set | void | Set value of attribute |

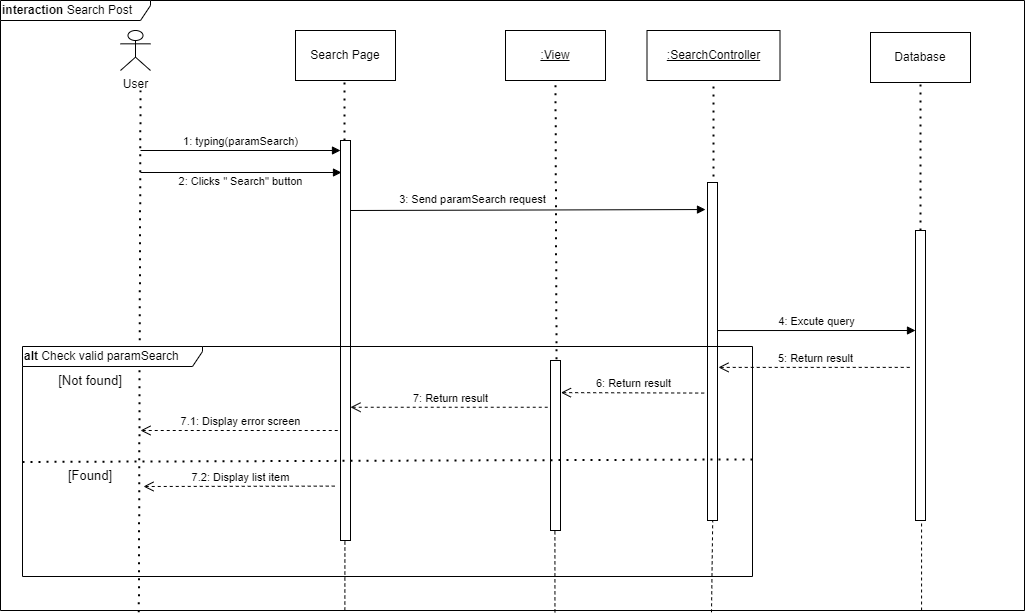
### 3.3 Sequence Diagrams

#### a. Create Post



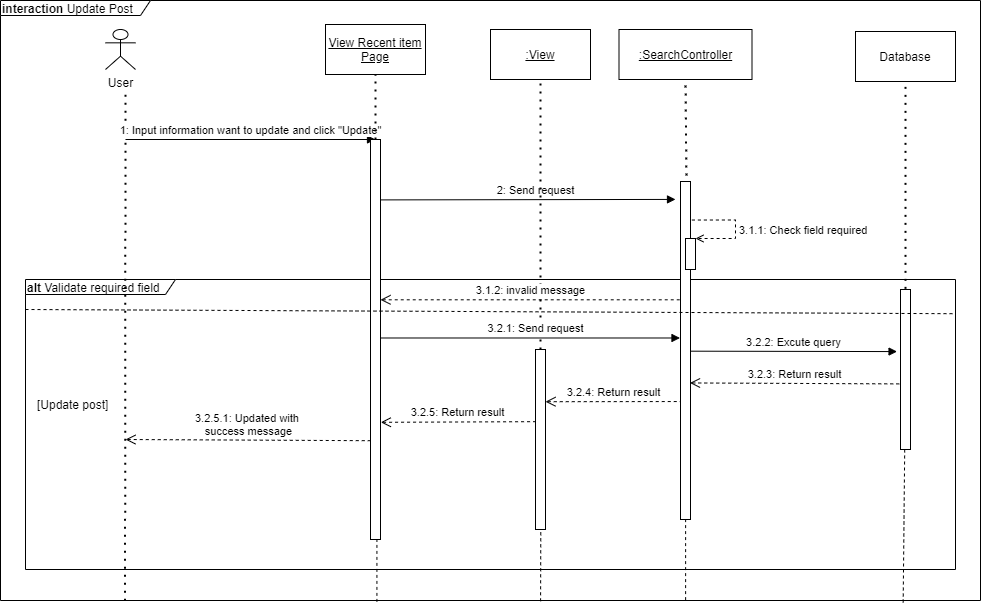
*Figure 9. <Sequence Diagram> Create post*

#### b. Search Post



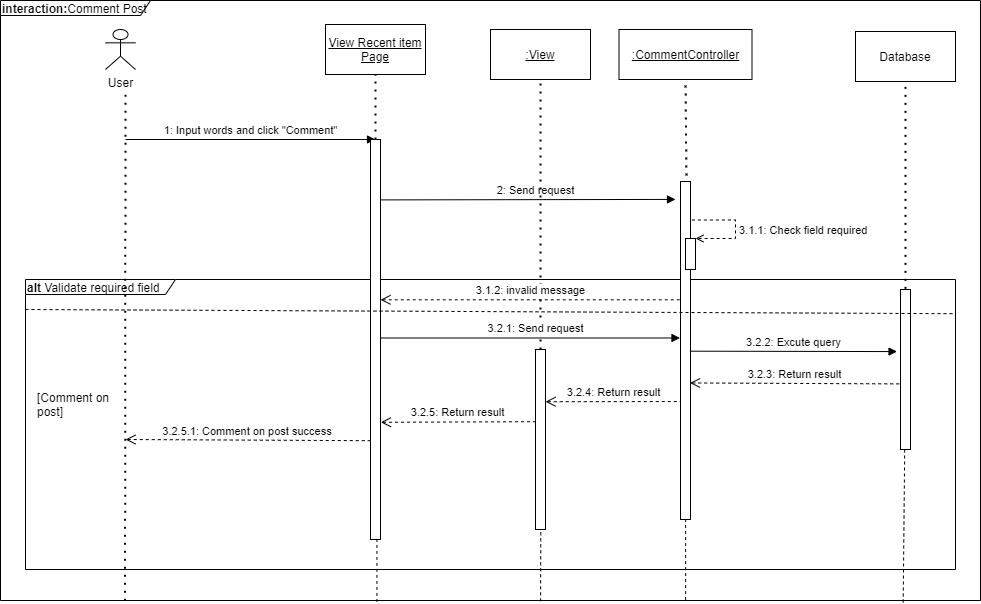
*Figure 10. <Sequence Diagram> Search post*

#### c. Update Post



*Figure 11. <Sequence Diagram> Update post*

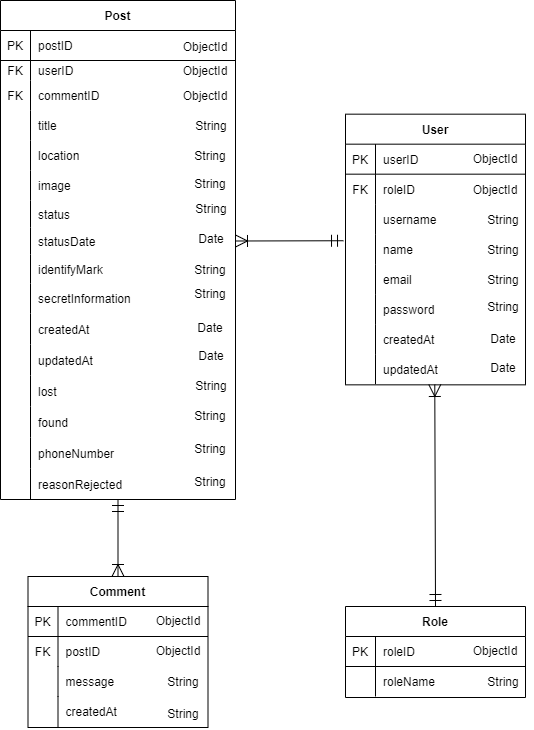
#### d. Comment On Post



*Figure 12. <Sequence Diagram> Comment on post*

## 4. Data & Database Design

### 4.1 Database Design



*Figure 13. Database Design*

#### 4.1.1 Post

*Table 29. <Database Design> Post*

| **Field name** | **Type** | **Unique** | **Allow Nulls** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- |
| postID | ObjectId | X |  | PK |  |
| userID | ObjectId |  |  | FK |  |
| commentID | ObjectId |  |  | FK |  |
| title | String |  |  |  |  |
| location | String |  |  |  |  |
| image | String |  |  |  |  |
| status | String |  |  |  |  |
| statusDate | Date |  |  |  |  |
| identificationMark | String |  |  |  |  |
| secretInformation | String |  |  |  |  |
| createdAt | Date |  |  |  |  |
| updatedAt | Date |  |  |  |  |
| lost | String |  |  |  |  |
| found | String |  |  |  |  |
| phoneNumber | String |  |  |  |  |
| reasonRejected | String |  |  |  |  |

#### 4.1.2 User

*Table 30. <Database Design> User*

| **Field name** | **Type** | **Unique** | **Allow Nulls** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- |
| userID | ObjectId | X |  | PK |  |
| roleID | ObjectId |  |  | FK |  |
| name | String |  |  |  |  |
| email | String | X |  |  |  |
| password | String |  |  |  |  |
| createdAt | Date |  |  |  |  |
| updatedAt | Date |  |  |  |  |

#### 4.1.3 Role

*Table 31. <Database Design> Role*

| **Field name** | **Type** | **Unique** | **Allow Nulls** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- |
| roleID | ObjectId | X |  | PK |  |
| roleName | String |  |  |  |  |

#### 4.1.4 Comment

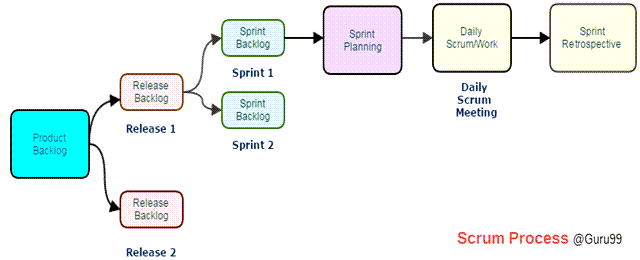
*Table 32. <Database Design> Comment*

| **Field name** | **Type** | **Unique** | **Allow Nulls** | **PK/FK** | **Notes** |
| --- | --- | --- | --- | --- | --- |
| commentID | ObjectId | X |  | PK |  |
| postID | ObjectId |  |  | FK |  |
| message | String |  |  |  |  |
| createdAt | Date |  |  |  |  |

# V. Software Testing Documentation

## 1. Overall Description

### 1.1 Test Model



*Figure 14. Scrum Model*

**Source**: [*Scrum Testing Methodology Tutorial: What is, Process, Artifacts, Sprint*](https://www.guru99.com/scrum-testing-beginner-guide.html)

We apply Scrum Testing because of the following reasons:

* It makes the interaction between developers and customers more fast and simple.
* It can quickly adapt to the rate of change required.
* It helps developers pay more attention to quality, performance and usability.

### 1.2 Testing Levels

#### a. Unit testing

Unit testing applies test conditions for individual function, which is the first level for locating defects in the LAF system.

#### b. Integration testing

Integration testing validates the workings of individual modules/components in the system in a unified manner.

#### c. System testing

System testing tests the LAF system as a whole validates the functionalities and verifies if the system meets the end-to-end specifications and requirements.

### 1.3 Testing Types

**a. Function testing**

● Testing the functionalities individually to verify the purpose as intended of each function under specific conditions.

● For detailed descriptions of each test case please refer to [Report5\_Test Case Document.xlsx](https://docs.google.com/spreadsheets/d/1rOwDvvV-KgT38Gixmhjn2zl8zFXBrfiH/edit#gid=1530594792) document.

**b. UI testing**

● Testing the UI controls functionalities to verify the purpose as intended of each UI control.

● Additionally, test if the interaction flows correctly conforms to requirements and business rules/processes.

● For detailed descriptions of each test case please refer to [Report5\_Test Case Document.xlsx document](https://docs.google.com/spreadsheets/d/1rOwDvvV-KgT38Gixmhjn2zl8zFXBrfiH/edit#gid=1530594792).

## 2. Test Plan

### 2.1 Test Stages

*Table 33. Type of Test*

| **Type of Tests** | **Stage of Test** | | | |
| --- | --- | --- | --- | --- |
| **Unit** | **Integration** | **System** | **Acceptance** |
| Function Test | X | X | X | X |
| User Interface Test | X |  | X |  |

### 2.2 Resources

#### a. Human resources

*Table 34. Human resources*

| **Worker/Doer** | **Role** | **Specific Responsibilities/Comments** |
| --- | --- | --- |
| Tran Thien Quy | Leader | Verify test result, final review of test cases, test plan, test reports, test data |
| Pham Nguyen Hoai Linh | Member | Review test cases and reports as planned |
| Vo Quoc Viet | Member | Execute test and reports as planned |
| Vu Lam Bao Van | Member | Planning, assign test function to members |

#### b. Environment

*Table 35. Environment specifications*

| **Purpose** | **Tool** | **Provider** | **Version** |
| --- | --- | --- | --- |
| Test case & bug logs management, test reports | Excel  Word | Microsoft  Microsoft | >= 2013  >= 2013 |
| Screen capture | Windows Snipping Tool | Microsoft | 6.4.5 |
| Execution browser | Edge  Chrome | Microsoft  Google | >=80.0  >=80.0 |
| Communication | Notion  Messenger | Atlassian  Facebook | N/A  N/A |

### 2.3 Test Milestones

*Table 36. Test Milestones*

| **Milestone Task** | **Effort (md)** | **Start Date** | **End Date** |
| --- | --- | --- | --- |
| Create test plan | 3 | 01/06/2022 | 01/06/2022 |
| Create integration test cases for Web App | 2 | 04/06/2022 | 20/06/2022 |
| Create system test cases for Web App | 2 | 10/06/2022 | 20/06/2022 |
| Execute tests for Web App | 1 | 22/06/2022 | 10/06/2022 |
| Create reports for testing | 2 | 13/07/2022 | 15/07/20022 |

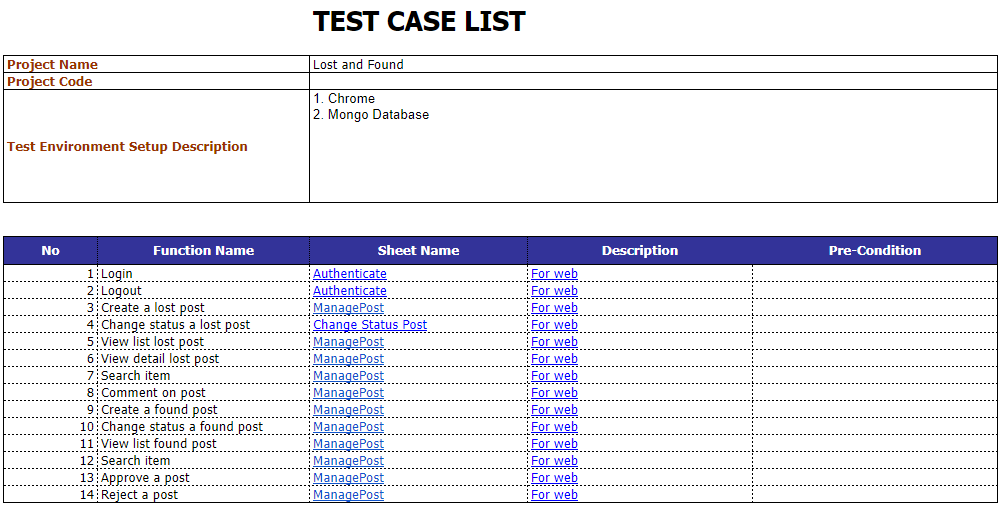
### 2.4 Deliverables

*Table 37. Deliverables*

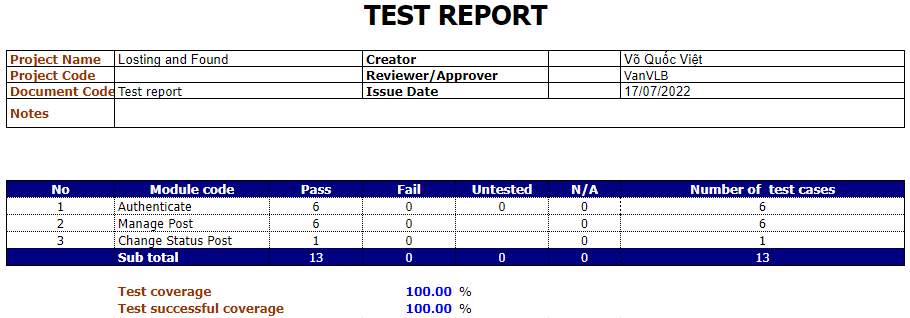
| **No.** | **Deliverables** | **Due Date** |
| --- | --- | --- |
| 1 | Test plan | 01/06/2022 |
| 2 | Test Case Suite for web app | 27/06/2022 |
| 3 | Report on testing | 15/07/2022 |

## 3. Test Cases

* Detailed test cases will be described in [LAF\_Test Case Document.xlsx](https://docs.google.com/spreadsheets/d/1rOwDvvV-KgT38Gixmhjn2zl8zFXBrfiH/edit#gid=1530594792) file.



## 4. Test Reports

**

# VI. Release Package & User Guides

## 1. Deliverable Package

### 1.1 Source codes & documents

| **No.** | **Items** | **Sub Item** |
| --- | --- | --- |
| **Code Package** | | |
| 1. | LAF.CapstoneBE.zip |  |
| 2. | LAF.CapstoneFE.zip |  |
| **Documents** | | |
| 1. | Test Documentation | LAF\_Test Case Document.xlsx |
| 2. | Final Project Report | LAF\_Project Document.docx |

### 1.2 Known Issues, Limitations & Restrictions

Applicant Process API server has system requirements of at least 1GB RAM but our free deployed Heroku only offers a maximum of 500MB RAM. Due to the shortage of budget, we cannot afford a charged plan to implement the system properly.

## 2. Installation Guides

### 2.1 System Requirements

Any Computer included keyboard and mouse or Phone with a browser with an internet connection

#### 2.1.1 Hardware requirements

| **PC** | **Minimum** | **Recommended** |
| --- | --- | --- |
| Internet connection | Cable, Wi-fi (4 Mbps) | Cable, Wi-fi (8 Mbps) |
| Processor | Intel Core i3 1.4Ghz | Intel Core i7 2.5Ghz |
| Memory | 4GB RAM | 8GB RAM |
| Storage | HDD 100GB | SSD 200GB |

#### 2.1.2 Software requirements

| **Component** | **Name and Version** | **Description** |
| --- | --- | --- |
| Operating system | Windows 7 sp1/8.1/10/window server 2016 or above | Operating system for building production |
| Dbms | MongoDB 6.0.0 | Used to manage database |
| ReactJS | [java 8 or 9](https://www.java.com/) and [spring framework 5.0.4](https://docs.spring.io/spring/docs/5.0.4.RELEASE/spring-framework-reference/) or above | Used to run frontend application |
| Nodejs Express | V14.17.6 or above | Used to run backend server |

### 2.2 Setup Files

- *LAF\_BE.zip*: API server of the system.

- *LAF\_FE.zip*: UI web administration application.

### 2.3 Installation Instruction

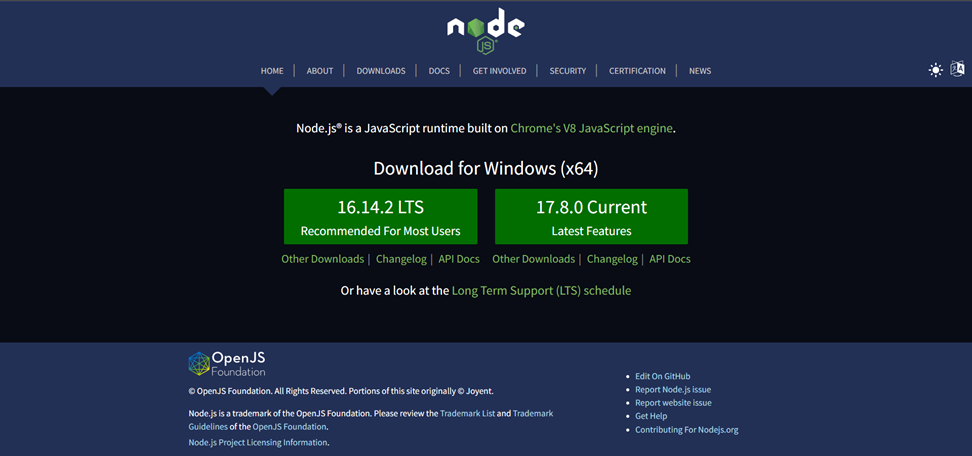
Setup and start a Windows OS (Windows 10 or Windows Server 2016):

a. For further information: Download Windows 10 | Microsoft

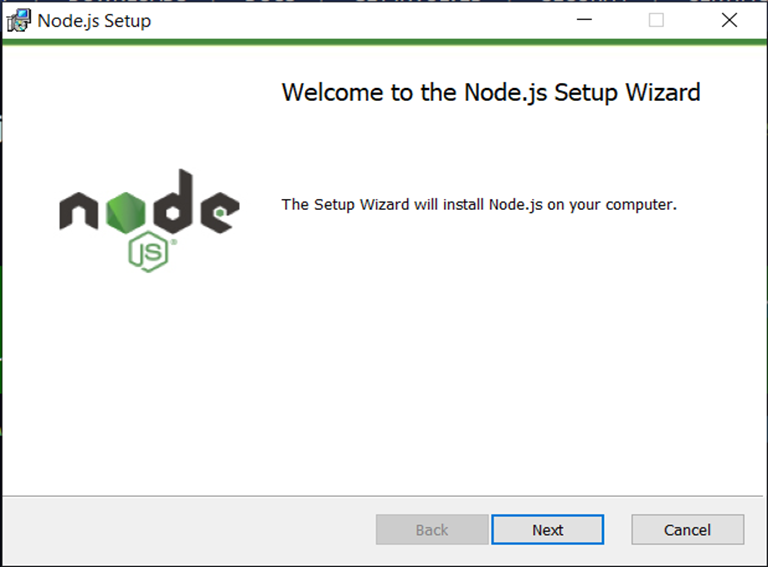
b. Setup a Windows Server 2016

##### **a. NodeJS**

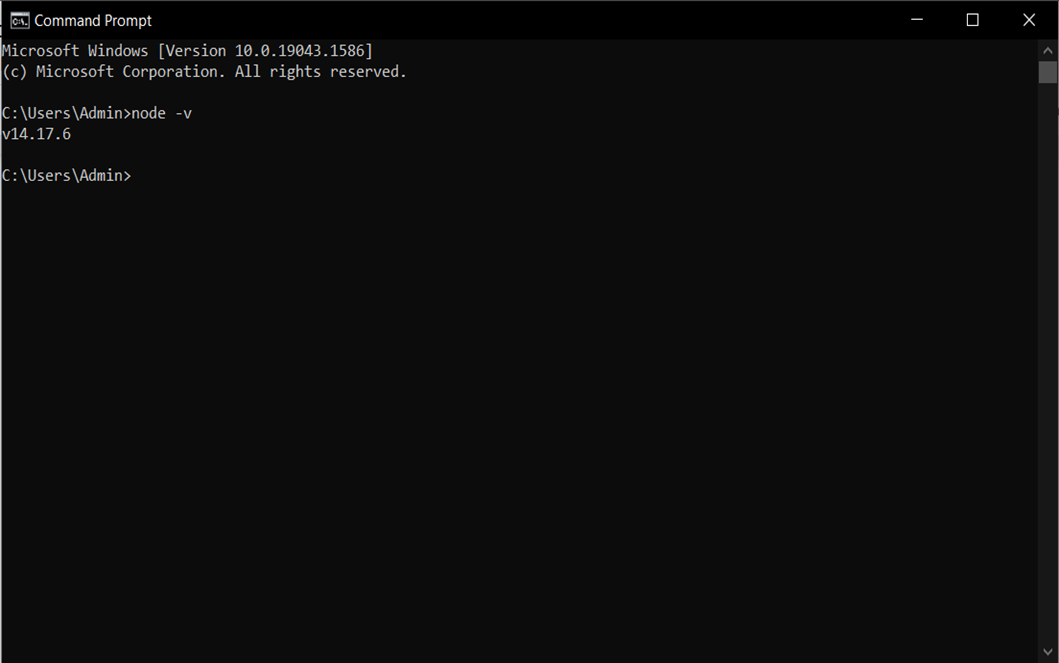
1. Go to [Node.js](https://nodejs.org/en/) and download the latest version.



2. Start the NodeJS downloaded setup file and go through install.



3. Open cmd and check the node version by run command *node-v.*



## 3. User Manual

### 3.1 Terms and definitions

N/A

### 3.2 System requirements

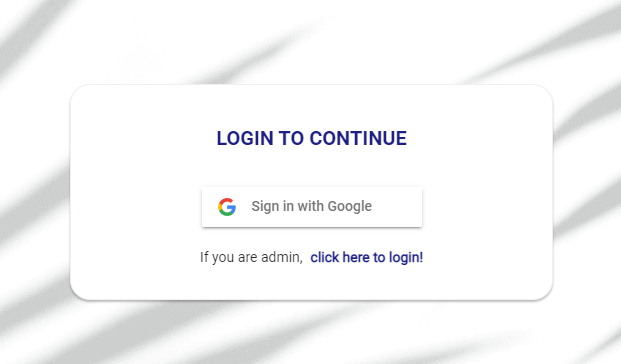
See to [*section 2.1*](#_heading=h.31ucoouilx5v)

### 3.3 Application Usage

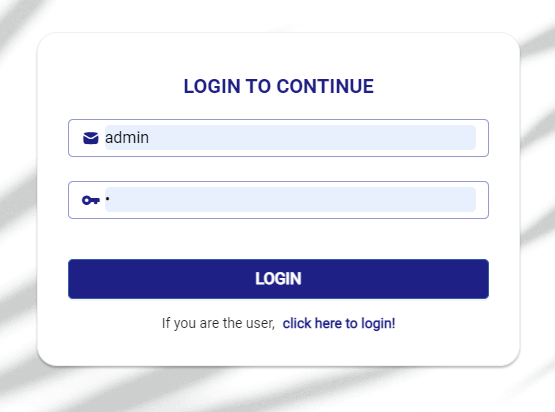
#### 3.3.1 Login

##### **3.3.1.1 Admin login**

Click on the link in bold "click here to login" in the text "If you are admin, click here to login!"



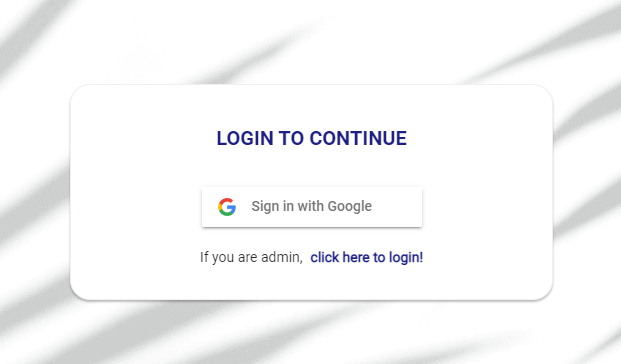
Input your admin "Username" and "Password"

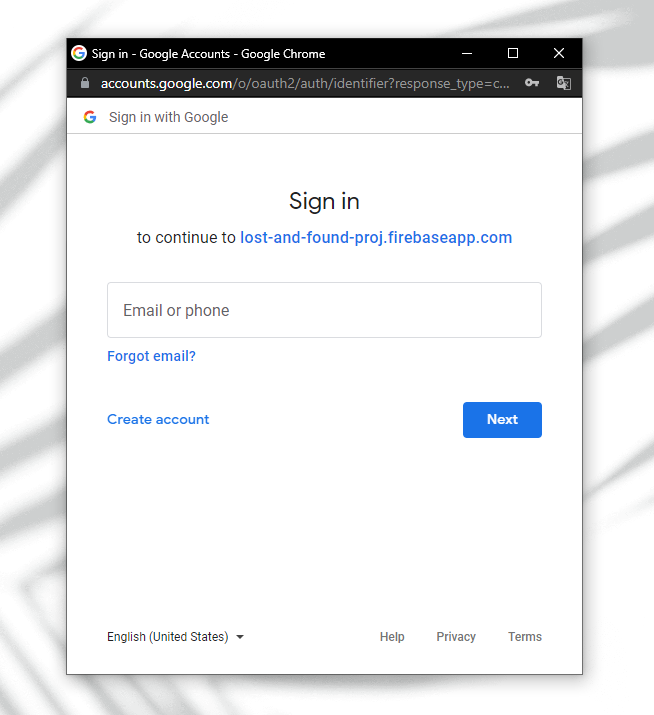


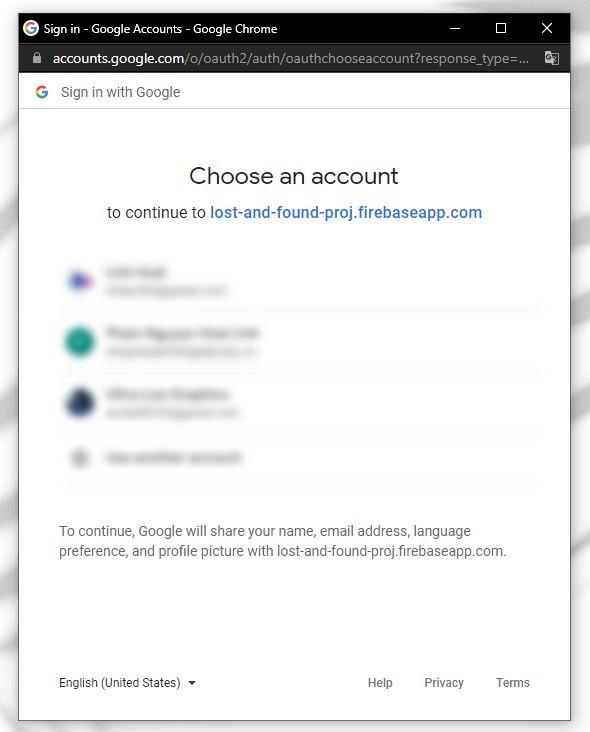
Click "LOGIN" button, if your admin username and password is correct you will access to admin page.

##### **3.3.1.2 User login**

Click the "Sign in with Google" button

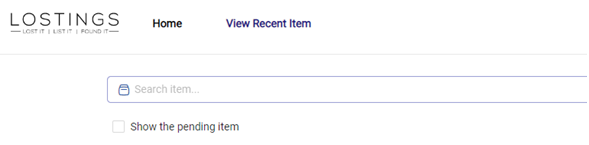


Input your "Email" and "Password"

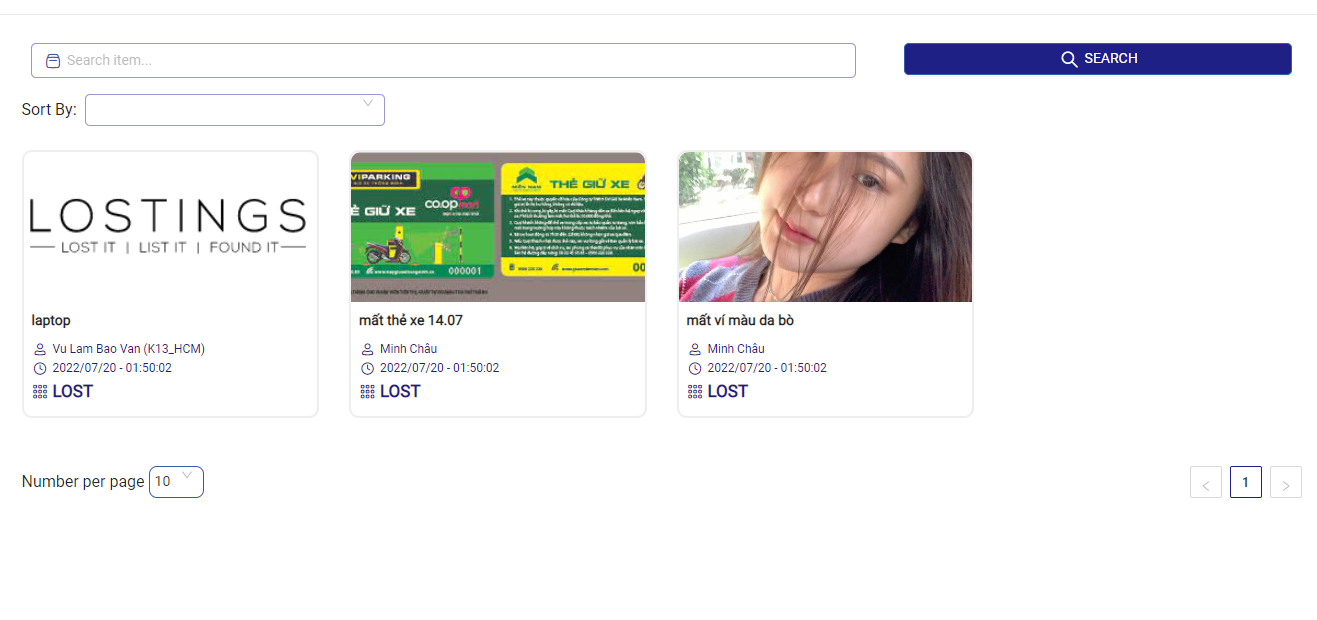
Select the google account you're signed in with ****

#### 3.3.2 Admin feature

##### **3.3.2.1 Admin view recent post feature**

Click on the "View Recent Item" in the navigation bar at the top of the website 

Recent Item list will be displayed

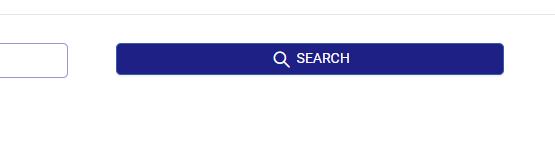


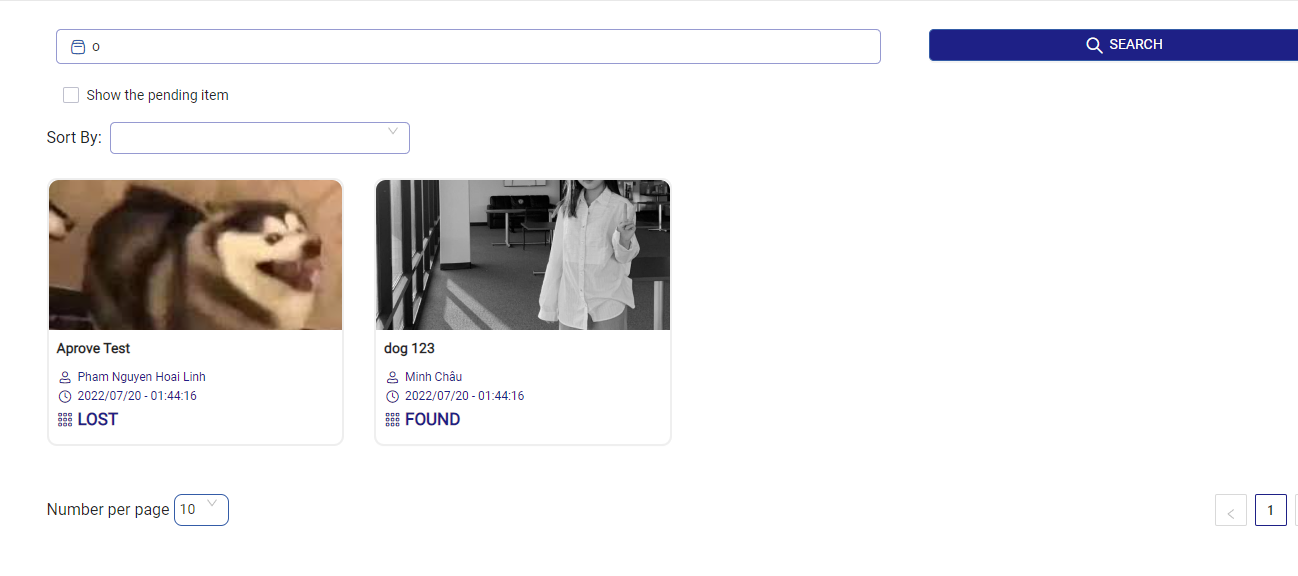
##### **3.3.2.2 Admin search post feature**

On View recent items, click on Search bar

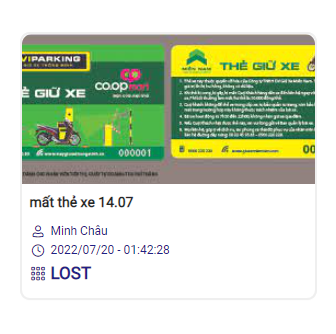


Input a item name you want to search and click “Search” button right next to the input bar, all items matching the search will be displayed

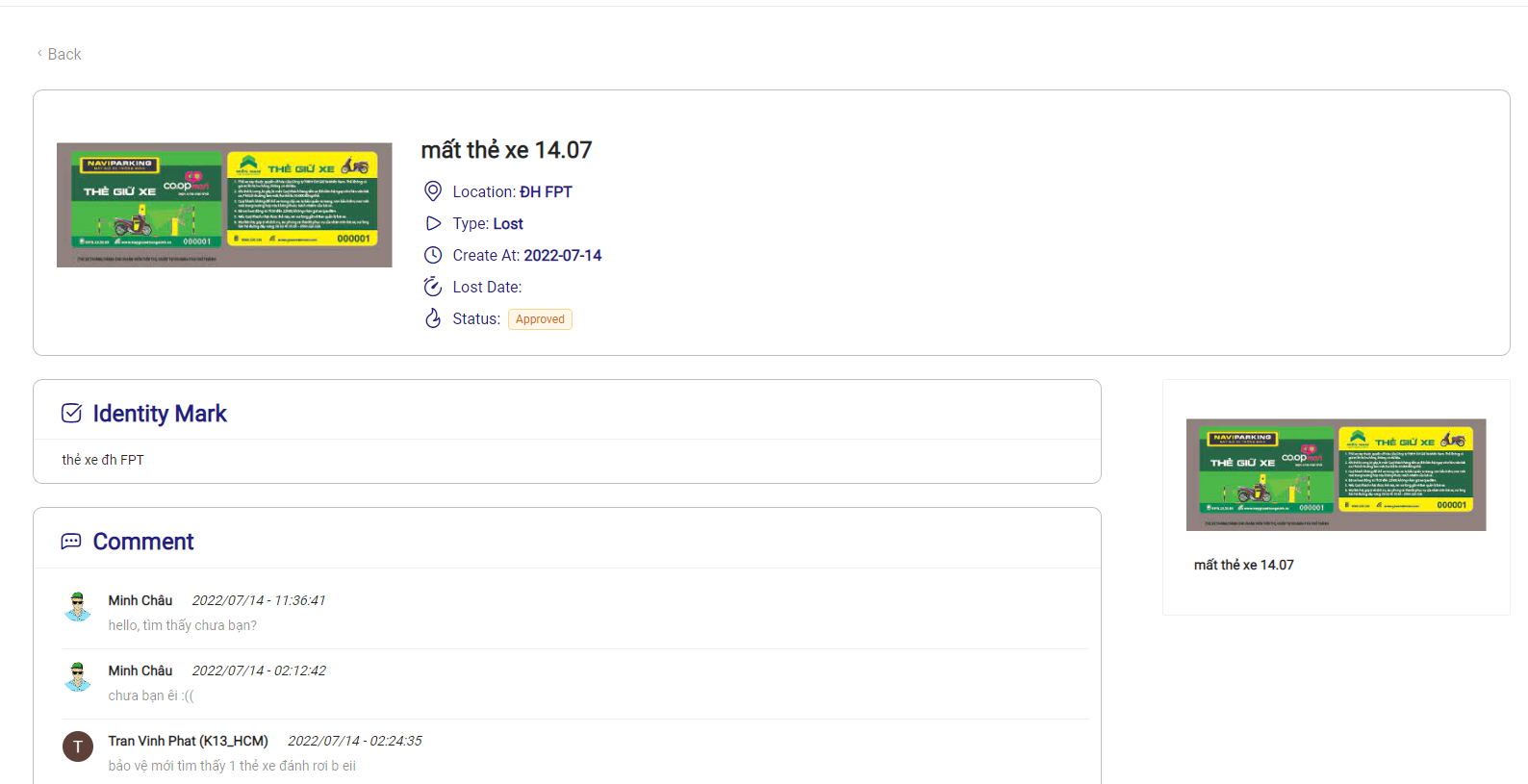




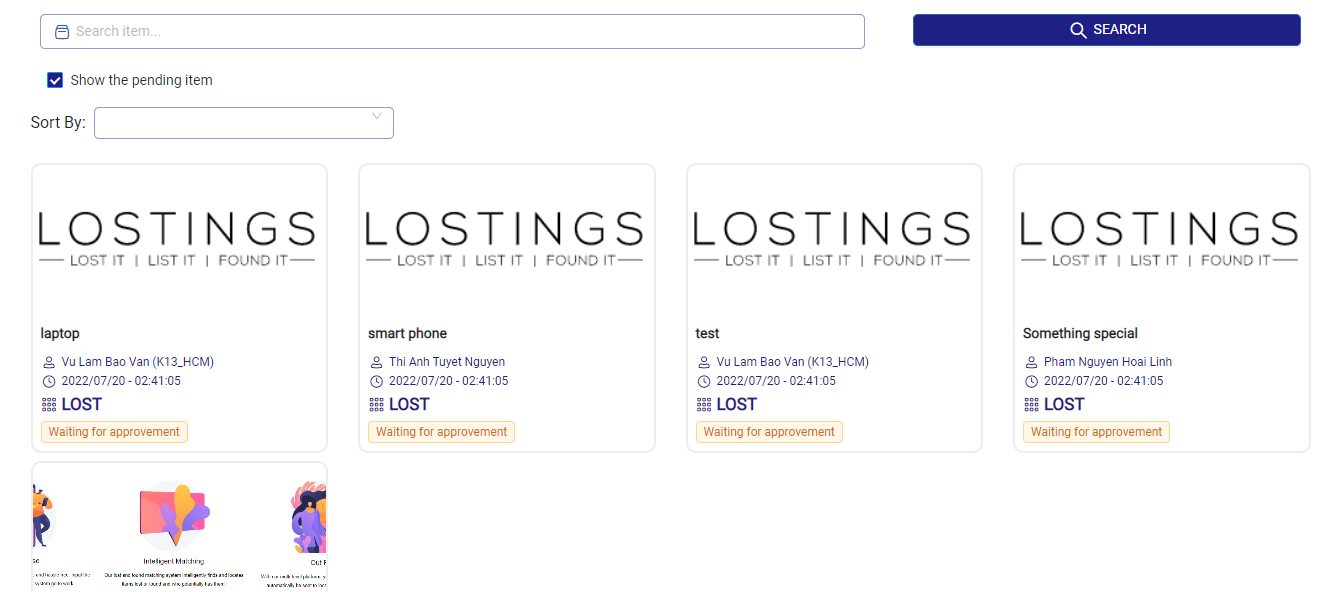
##### **3.3.2.3 Admin view a post feature**

Click on a post as a tag containing the post's image and general information 

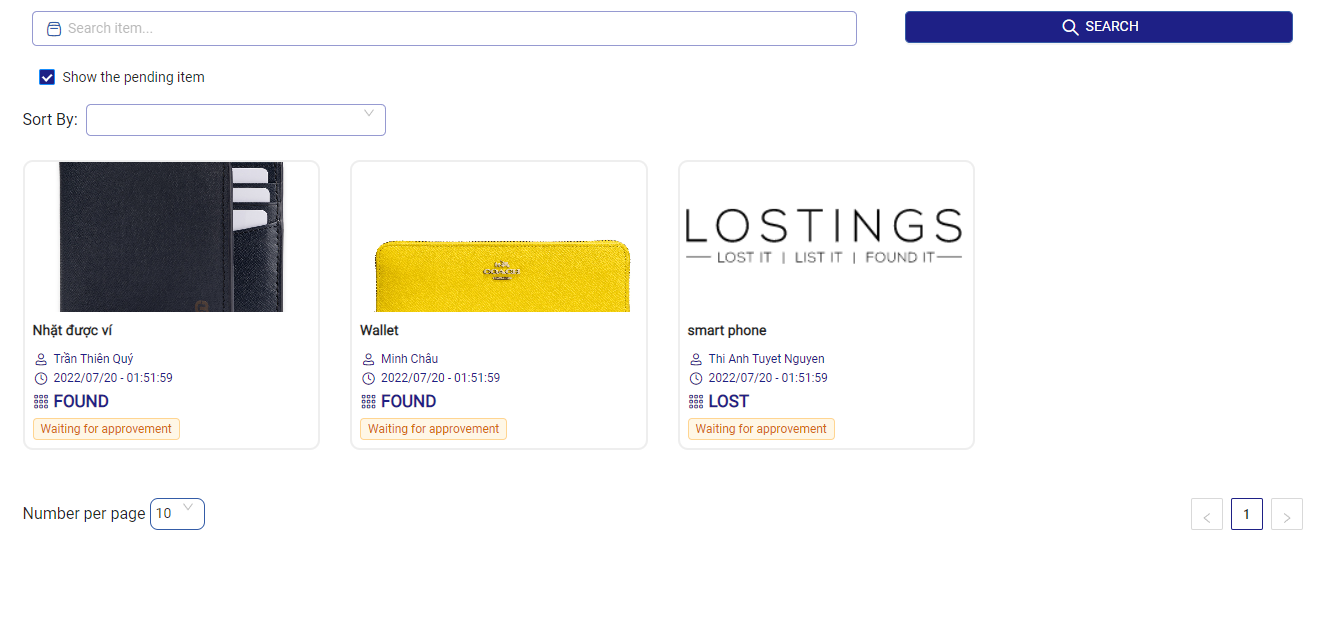
Post detail will be display



##### **3.3.2.4 Admin view pending post feature**

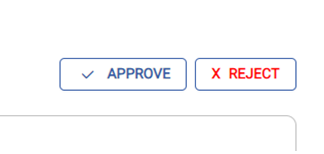
Tick the "Show the pending item" box at the top of the Recent Posts 

Pending posts will display



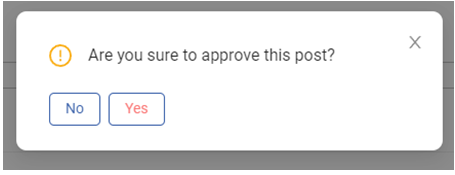
##### **3.3.2.5 Admin approve/reject a pending post feature**

On a pending post a click Approve/Reject button



###### a. Approve

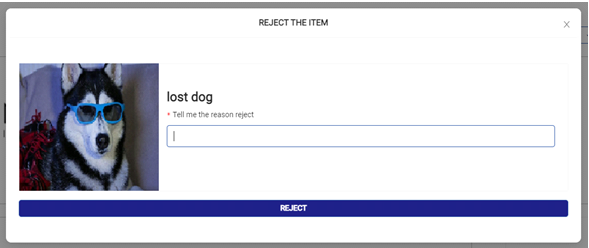
Confirmation window will displayed



Click "Yes" button and the post will be approve, Success Notification will appear in the upper right corner of the website.

Click "No" button to cancel approved and close Confirmation window

###### b. Reject

Reject Post window will displayed 

Input reject reason in the input field

* Click "X" button in the upper right corner to cancel approved and close Reject Post window
* Click "Reject" button and the post will be approve, Success Notification will appear in the upper right corner of the website.

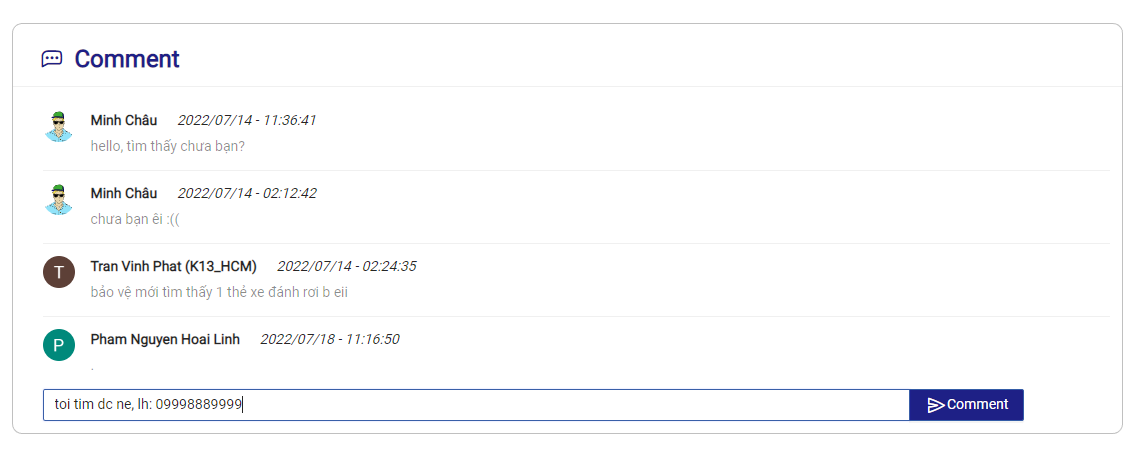
##### 

##### 

##### 

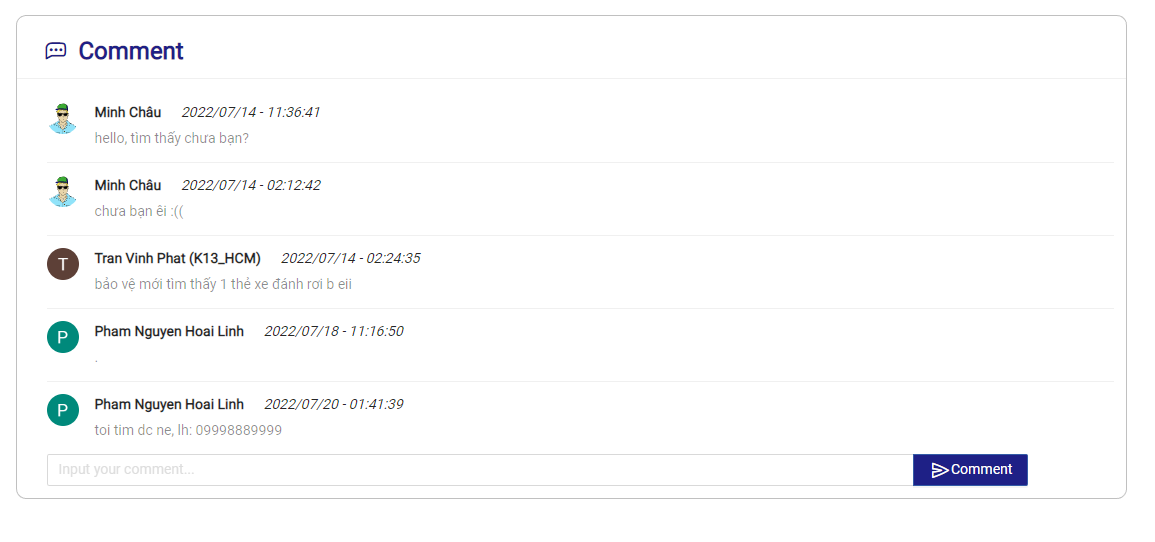
##### **3.3.2.6 Comment feature**

Input a comment in the comment input field that suggests "Input your comment"



Press the "Comment" button by mouse or press enter (if using a computer with a keyboard)

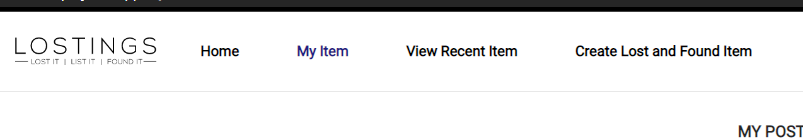
Comment will be appear on comment list.



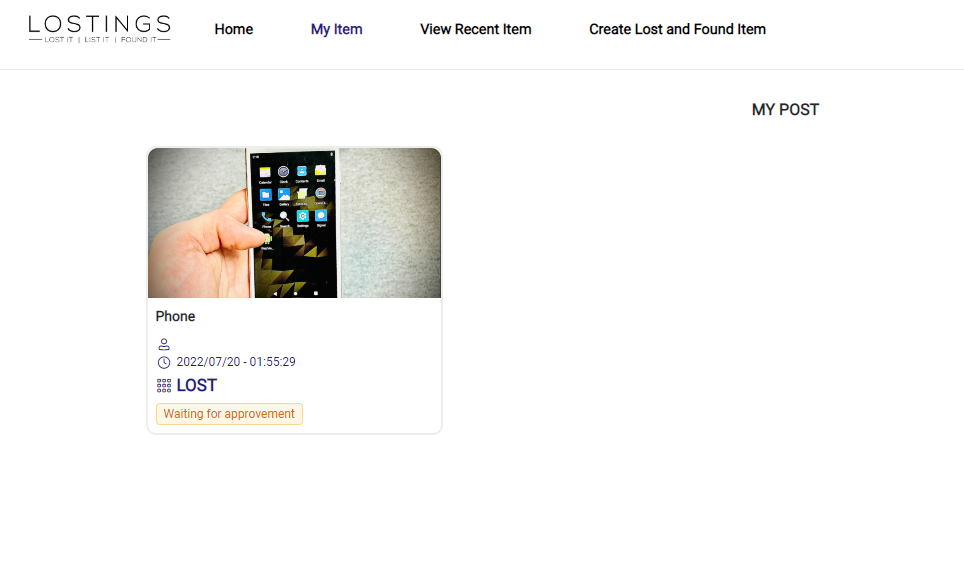
#### 3.3.3 User feature

##### **3.3.3.1 User view their own post feature**

Click on the "My Item" in the navigation bar at the top of the website



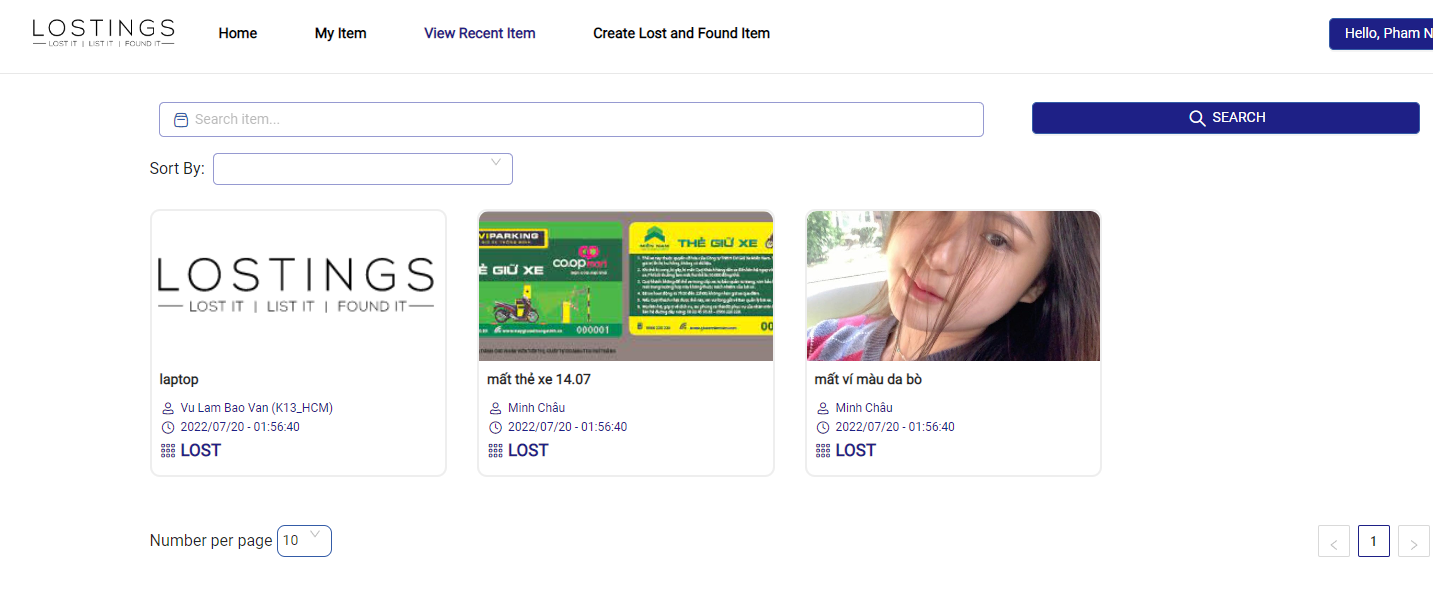
Their own post list will be display



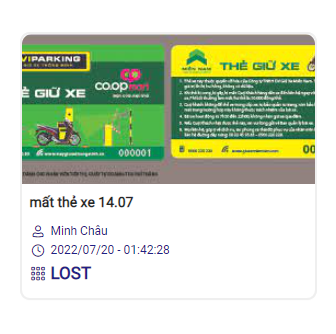
##### **3.3.3.2 User view recent post feature**

Click on the "View Recent Item" in the navigation bar at the top of the website 

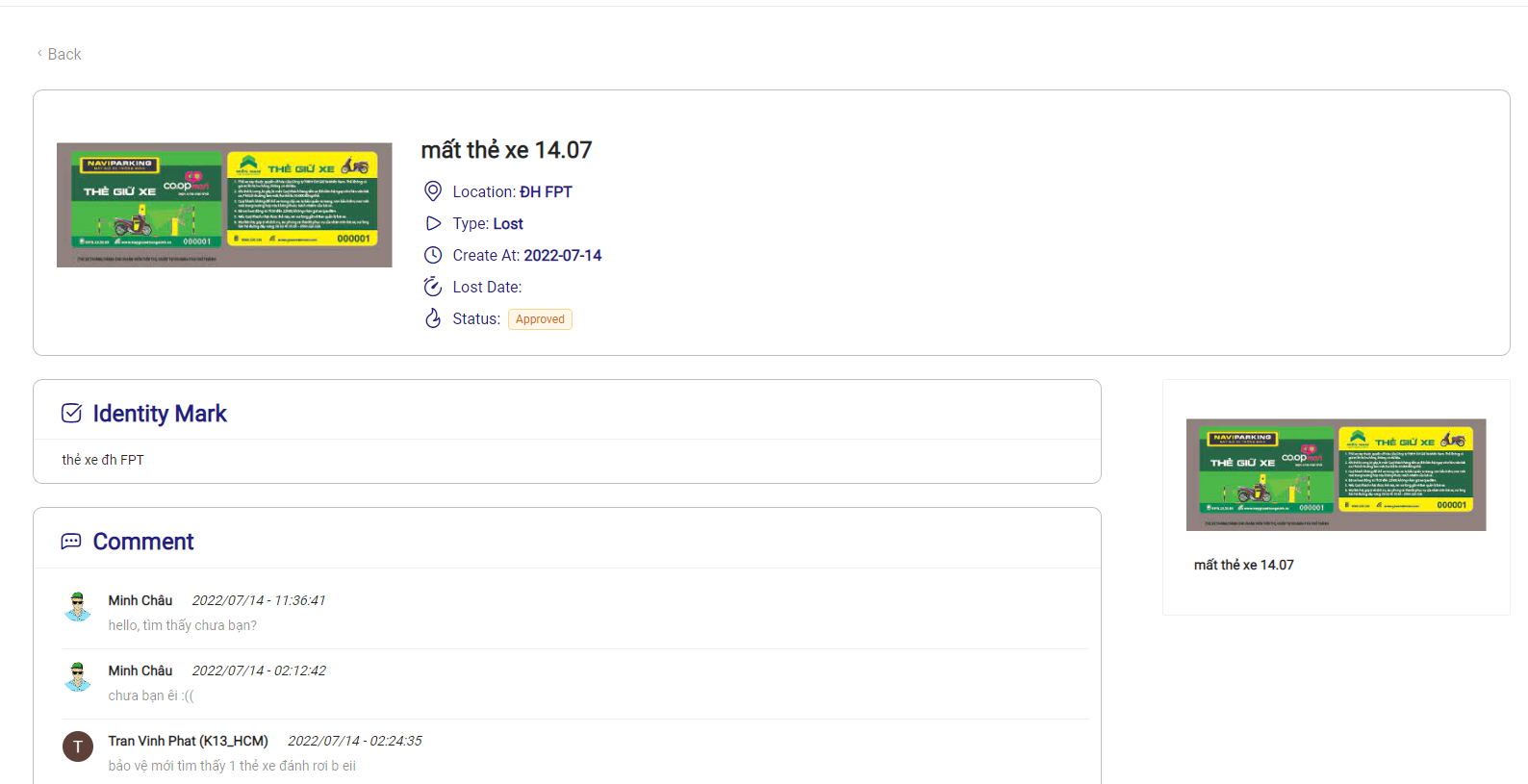
Recent posts will be displayed



##### **3.3.3.3 View a post feature**

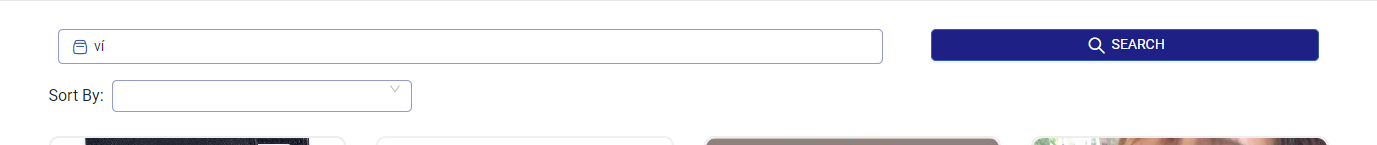
Click on a post as a tag containing the post's image and general information 

Post detail will be display

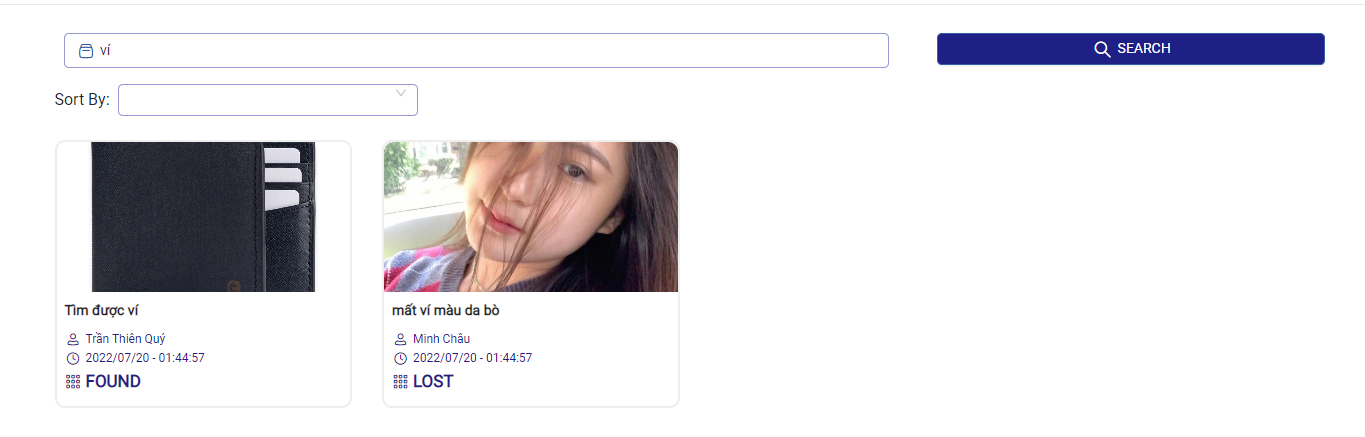


##### **3.3.3.4 User Search post feature**

On View recent items, click on Search bar

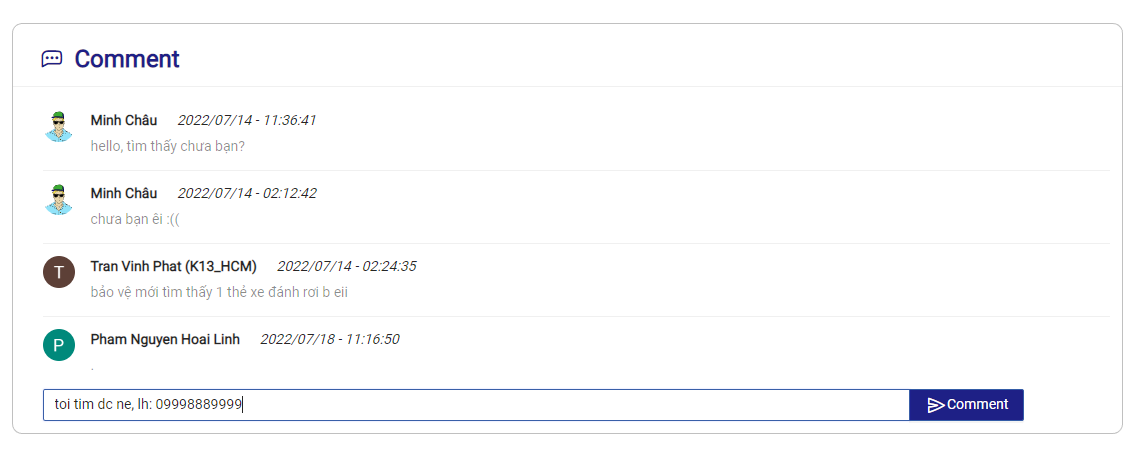


Input a item name you want to search and click “Search” button right next to the input bar, all items matching the search will be displayed

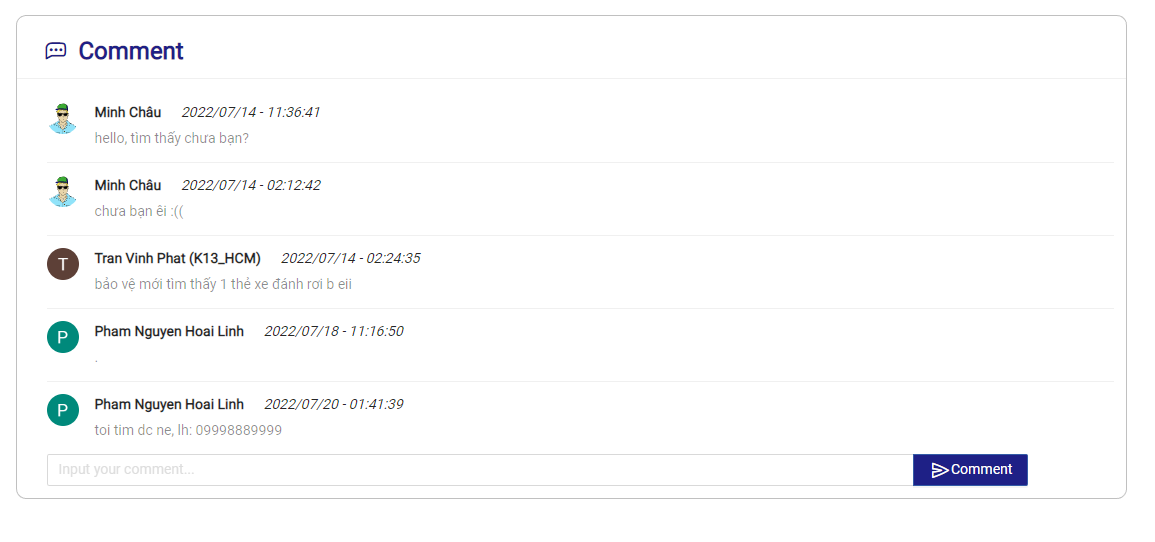


##### **3.3.3.5 Comment feature**

Input a comment in the comment input field that suggests "Input your comment"

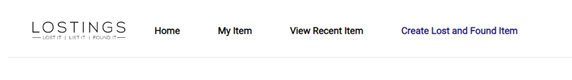


Press the "Comment" button by mouse or press enter (if using a computer with a keyboard)

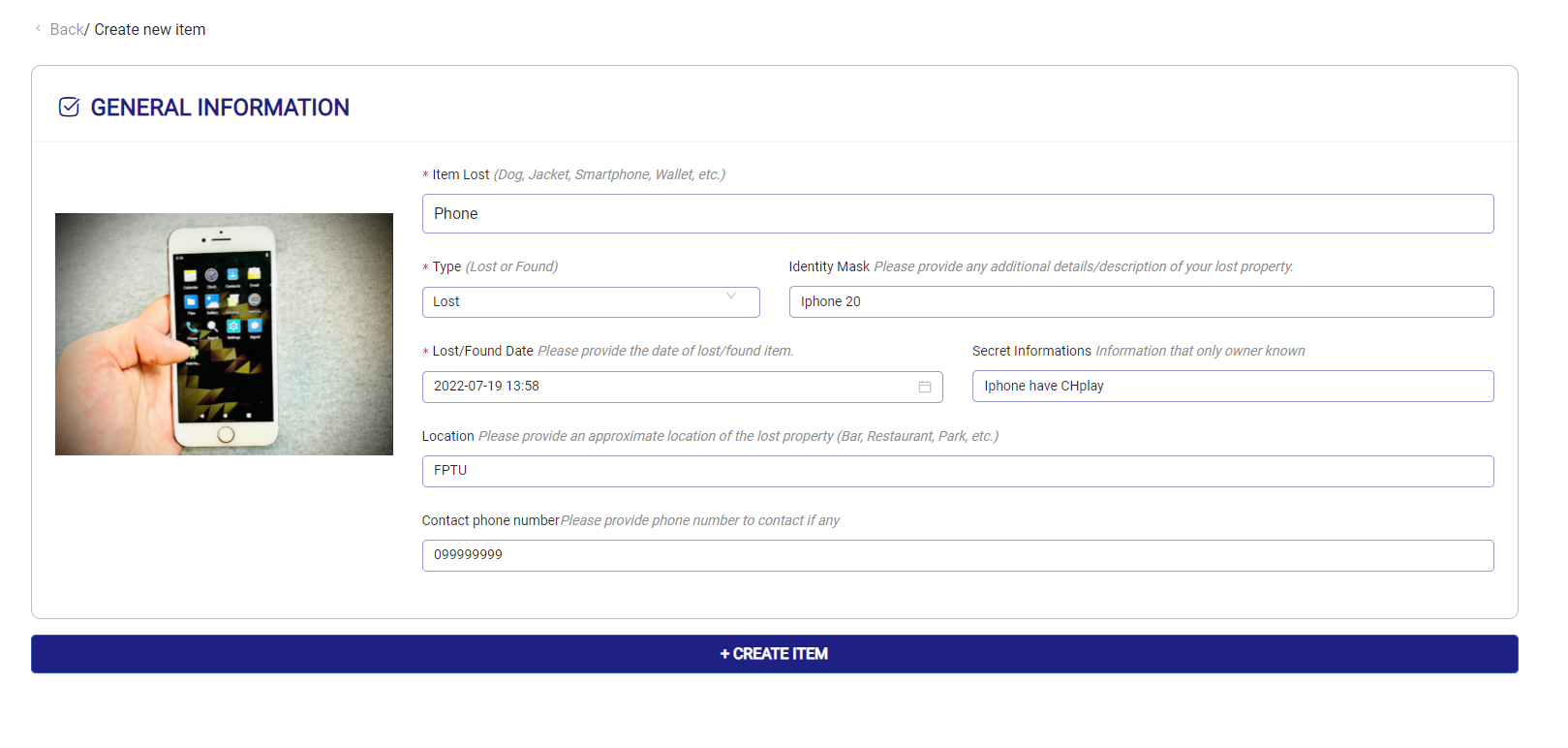
Comment will be appear on comments list.

**3.3.3.5 Create post feature**

Click on the "Create Lost and Found Item" in the navigation bar at the top of the website

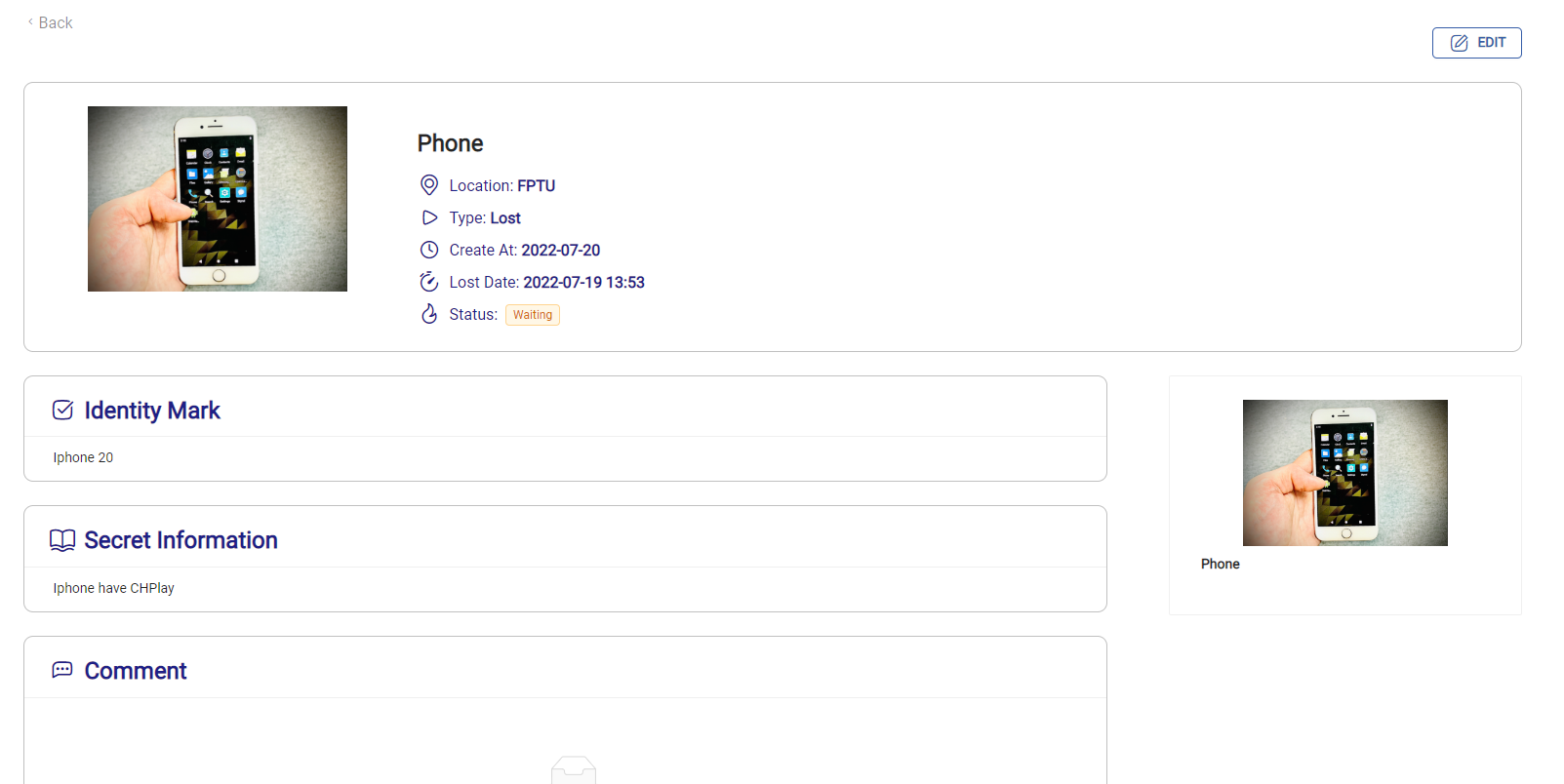


A post form will be displayed



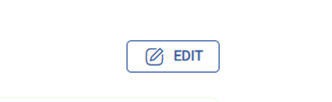
Input all require information

Click "Create Item" and Success Notification will appear in the upper right corner of the website and you will return to the view post page to view the post you just created

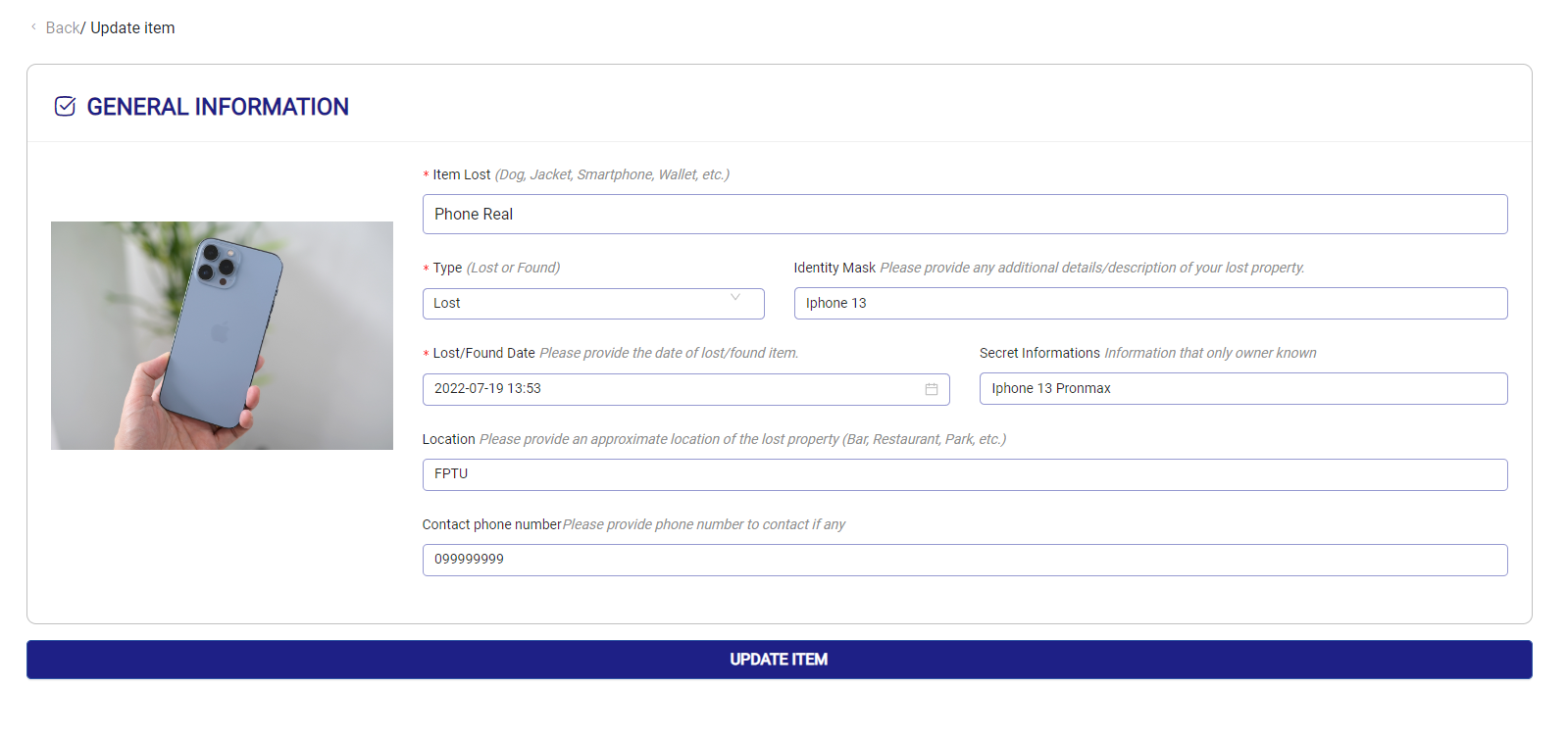


##### **3.3.3.6 User Edit their own Post (only posts with "Waiting" status)**

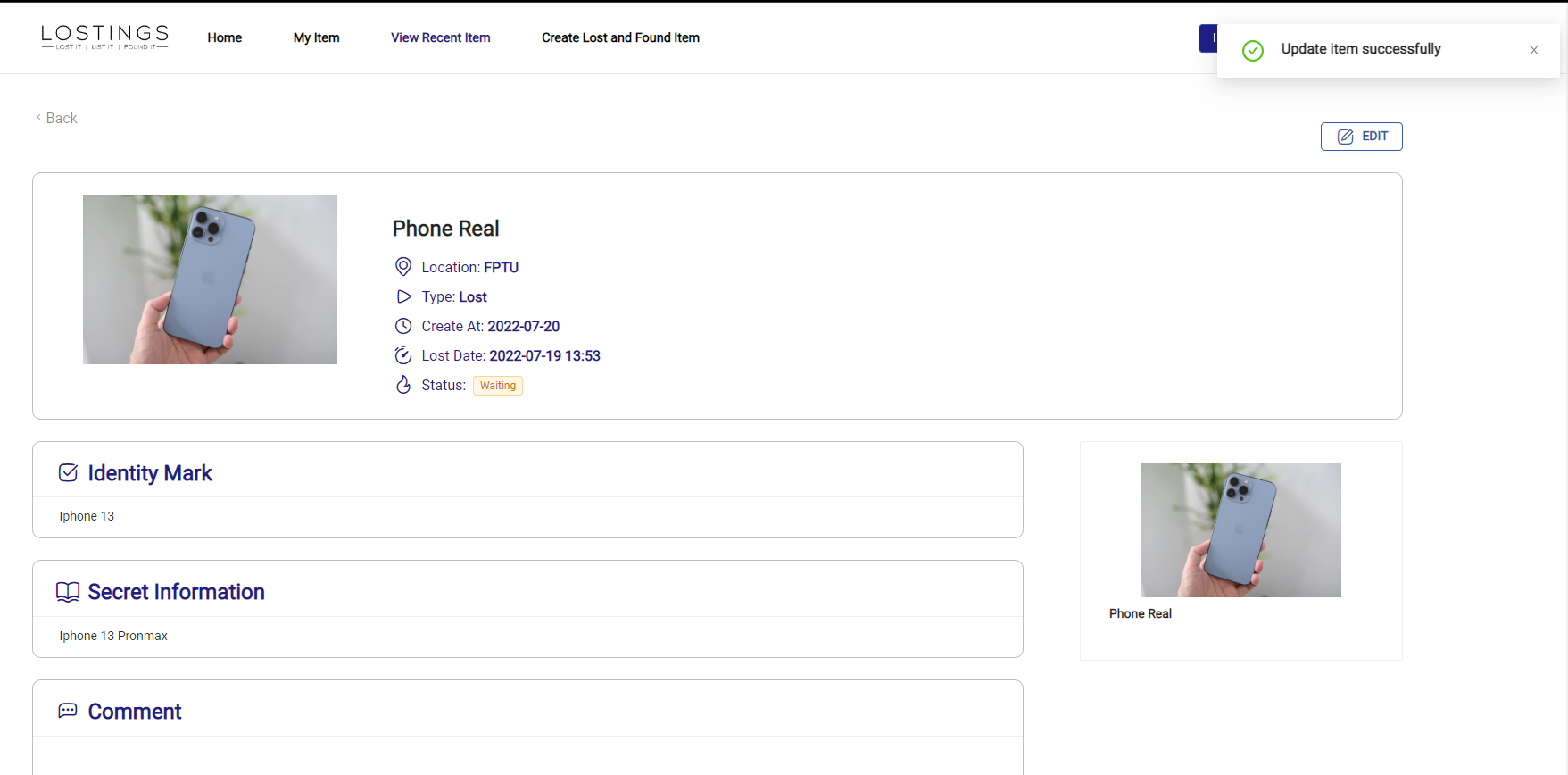
Click the "Edit" button in the upper right corner of the post detail page



Input all require information



Click "Update Item" and Success Notification will appear in the upper right corner of the website and you will return to the view post page to view the post you just edited



**3.3.4 Logout (Admin/User)**

Hover on your name, the logout option will appear



Click "Logout" option and your account will be logged out, returning to the login page

### 3.4 Troubleshooting

N/A

# VII. Appendix

## 1. Glossary

*See section* [*Definition and Acronyms*](#_heading=h.1fob9te)

## 2. References

Ant Design, <https://ant.design/>

MongoDB, [https://www.mongodb.com](https://www.mongodb.com/)/

Node Express JS, <https://expressjs.com/>

ReactJS, <https://reactjs.org/>

## 3. Message

*Table 38. Message*

| **STT** | **Message Code** | **Message type** | **Context** |
| --- | --- | --- | --- |
| 1 | MSG-001 | In red, under the text box | Please input username |
| 2 | MSG-002 | In red, under the text box | Please input password |
| 3 | MSG-003 | In red | Wrong authentication, Please try again |
| 4 | MSG-004 | Pop-up, icon successfully | Approve successfully |
| 5 | MSG-005 | In red, under the text box | Please input Item lost |
| 6 | MSG-006 | In red, under the text box | Please select type |
| 7 | MSG-007 | In red, under the text box | Please pick the date |
| 8 | MSG-008 | In red, under the text box | Please pick the date less than current |
| 9 | MSG-009 | Pop-up, icon successfully | Comment successfully |
| 10 | MSG-0010 | Pop-up, icon successfully | Create item successfully |
| 11 | MSG-0011 | Pop-up, icon successfully | Update item successfully |