|  |  |
| --- | --- |
|  | **MINISTRY OF EDUCATION AND TRAINING** |

|  |
| --- |
| **FPT UNIVERSITY** |
| Capstone Project Document |
| Photographer Booking System |

|  |  |
| --- | --- |
| **FA20SE21** | |
| **Group Members** | Trần Thiên Thảo – Leader – SE62888  Trương Ngọc Mỹ - Member – SE62050  Bồ Công Đạt – Member – SE63154  Đào Sỹ Trung Kiên – Member – SE63208 |
| **Supervisor** | Lại Đức Hùng |
| **Ext Supervisor** | N/A |
| **Capstone Project code** | PBS |

- HoChiMinh, September 2020 -

**Table of Contents**

[Acknowledgement 4](#_heading=h.30j0zll)

[Definition and Acronyms 4](#_heading=h.1fob9te)

[I. Project Introduction 5](#_heading=h.3znysh7)

[1. Overview 5](#_heading=h.2et92p0)

[2. Product Background 5](#_heading=h.2s8eyo1)

[3. Existing Systems 6](#_heading=h.17dp8vu)

[4. Business Opportunity 7](#_heading=h.35nkun2)

[5. Software Product Vision 8](#_heading=h.1ksv4uv)

[6. Project Scope & Limitations 8](#_heading=h.44sinio)

[II. Project Management Plan 10](#_heading=h.3j2qqm3)

[1. Overview 10](#_heading=h.1y810tw)

[2. Management Approach 15](#_heading=h.2bn6wsx)

[3. Master Schedule 16](#_heading=h.49x2ik5)

[4. Project Organization 17](#_heading=h.2p2csry)

[5. Project Communication 18](#_heading=h.23ckvvd)

[6. Configuration Management 19](#_heading=h.32hioqz)

[III. Software Requirement Specification 19](#_heading=h.3fwokq0)

[1. Overall Description 19](#_heading=h.1v1yuxt)

[2. User Requirements 21](#_heading=h.4f1mdlm)

[3. Functional Requirements 90](#_heading=h.2u6wntf)

[4. Non-Functional Requirements 97](#_heading=h.3tbugp1)

[5. Other Requirements 99](#_heading=h.111kx3o)

[IV. Software Design Description 101](#_heading=h.206ipza)

[1. Overall Description 101](#_heading=h.4k668n3)

[2. System Architecture Design 101](#_heading=h.2zbgiuw)

[3. System Detailed Design 106](#_heading=h.1egqt2p)

[4. Class Specifications 116](#_heading=h.2r0uhxc)

[5. Data & Database Design 128](#_heading=h.48pi1tg)

[6. Algorithm 136](#_heading=h.1yyy98l)

[V. Software Testing Documentation 138](#_heading=h.1qoc8b1)

[1. Overall Description 138](#_heading=h.4anzqyu)

[2. Test Plan 139](#_heading=h.2pta16n)

[3. Test Cases 141](#_heading=h.14ykbeg)

[4. Test Reports 141](#_heading=h.3oy7u29)

[VI. Release Package & User Guides 141](#_heading=h.243i4a2)

[1. Deliverable Package 141](#_heading=h.j8sehv)

[2. Installation Guides 142](#_heading=h.42ddq1a)

[3. User Manual 144](#_heading=h.1vsw3ci)

[First occurrence 150](#_heading=h.odc9jc)

[Problem 150](#_heading=h.38czs75)

[First occurrence 151](#_heading=h.1nia2ey)

[Problem 151](#_heading=h.47hxl2r)

[VII. Appendix 151](#_heading=h.2mn7vak)

[1. Glossary [Optional] 151](#_heading=h.11si5id)

[2. References [Optional] 152](#_heading=h.3ls5o66)

[3. Others [Optional] 152](#_heading=h.20xfydz)

# Acknowledgement

*[Fill team’s acknowledgement here…]*

# Definition and Acronyms

|  |  |
| --- | --- |
| **Acronym** | **Definition** |
| PWM | Psychology website |
| AWS | Amazon Web Services |
| BA | Business Analysis |
| 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 |
| UAT | User Acceptance Test |
| UC | Use Case |
| API | Application Program Interface |

# Project Introduction

## 1. Overview

### 1.1 Project Information

* Project name: Photographer Booking System
* Project code: PBS
* Group name: FA20SE21
* Software type: Mobile Application, Web Application

### 1.2 Project Team

#### a. Supervisor

|  |  |  |  |
| --- | --- | --- | --- |
| **Full Name** | **Email** | **Phone Number** | **Title** |
| Lại Đức Hùng | Hungld5@fe.edu.vn |  | Lecturer |

**Table 1 - Supervisor**

#### b. Team Members

|  |  |  |  |
| --- | --- | --- | --- |
| **Full Name** | **Email** | **Mobile** | **Role** |
| Trần Thiên Thảo | Thaottse62888@fpt.edu.vn | 0903917982 | Leader |
| Đào Sỹ Trung Kiên | Kiendstse63208@fpt.edu.vn | 0769606867 | Member |
| Bồ Công Đạt | Datbcse63154@fpt.edu.vn | 0915223623 | Member |
| Trương Ngọc Mỹ | Mytnse62050@fpt.edu.vn | 0906969071 | Member |

**Table 2 - Team members**

## 2. Product Background

Despite the ever-increasingly fast-paced development of science and technology, there are still difficulties in the connections between customers and photographers.

As for customers:

* They find it difficult to contact skilful photographers living nearby or near where they want to shoot.
* They cannot view photographers’ working schedule, so they are not able to book the ones they want.
* Without viewing photographers’ albums beforehand, they are not sure if photographers’ style meets their requirements before booking.
* They are unlikely to find the appropriate rates quickly.

As for photographers:

* They are likely to miss potential customers living nearby or near their current shooting sites because customers do not know where they are and how to contact them.
* Applications that allow them to arrange a working schedule and receive bookings from customers are scarce.

## 3. Existing Systems

Currently, there are a myriad of platforms that help connect freelancers from various industries to the customers. However, there are only a few platforms specializing in photography. In addition,

there are no systems or applications that provide photographers with time and business management.

Below are some employment and job-search systems that the team takes into consideration while doing research. Learning from their strengths and weaknesses can help better the team’s product/system.

### 3.1 WeHelp

WeHelp is a platform which allows freelancers from a wide range of industries to market their skills and services that companies, organizations and others may need.

Link:

<https://wehelp.vn/>

[https://apps.apple.com/us/app/wehelp-tìm-người-tìm-việc/id1447795502/](https://apps.apple.com/us/app/wehelp-t%C3%ACm-ng%C6%B0%E1%BB%9Di-t%C3%ACm-vi%E1%BB%87c/id1447795502/)

System actors:

* Customer.
* Freelancer.
* Admin.

Features:

* Allow freelancers to market their skills and services.
* Allow customers to state their requirements and book photographers.

Pros

* Good design, user-friendly

Cons

* Covering a wide range of job sectors, hence not well-support any sector.
* No time management for freelancers.
* No payment method.

### 3.2 vLance

A job-search website for freelancers, which helps companies or individuals reach the ones that can fulfil their needs.

Link: <https://www.vlance.vn/>

System actors:

* Customer.
* Freelancer.
* Admin.

Features:

* Allow customers to add job postings and desired commissions.
* E-wallet is supported.

Pros

* Allow freelancers to propose competitive commissions.
* E-wallet helps ensure both sides’ benefits.

Cons

* Complex layouts, not user-friendly.

### 3.3 Tripo

Tripo is a young developed web application for photographer booking. Provide photographer and studio ways to approach potential customers and express their skills.

Link: <https://tripo.vn/>

System actors:

* Customer.
* Photographer.
* Admin.

Features:

* Allows customers to post their request as auction format.
* Photographers can offer the price after viewing customers’ requests.

Pros

* Specialize in the photography domain.
* Easy to use design.

Cons

* Don’t support mobile applications, lack of interactivity.
* Don’t support schedule managing features for photographers.
* Customers can’t book directly from their favoured photographer.

## 4. Business Opportunity

The model that we want to build is not the one emerging in the current market, which freelancers seek websites such as vLance, FreeLancerViet in Vietnam or foreign ones such as Upwork, PeoplePerHour and so on. Despite their long-time existence, they still have limitations, which is a huge opportunity for the research team. Researching comparison websites and users’ feedback, the team realized that most job search websites allow customers to hire photographers, but they cannot arrange shooting time effectively. In addition, photography has various branches such as cosplay photography, portrait photography, product photography, etc. Each of these branches would have its techniques, which means that shooting prices vary. The team would like to improve the drawbacks of those job-seeking websites via this project.

In fact, there is also a need for contact between customers and photographers to help minimize difficulties in finding the most appropriate one. With the team’s product, customers could save a lot of time in searching and thanks to other users’ feedback, they no longer need to ask for advice from friends, family or social networks. Customers would be able to view photographers’ working schedule and then easily arrange desired shooting time. Photographers would not miss potential customers and have the chance to market their abilities via profile and users’ feedback. They could also be more flexible in their working time management.

## 5. Software Product Vision

Intending to help customers easily contact skilful photographers, the researchers came up with the idea of building two mobile applications, which serve as a platform for connecting customers and photographers and vice versa.

All things considered, this project is conducted to provide customers with a faster, more convenient and effective way to contact photographers. They now can view photographers’ working schedules and make their reservations easily without double-booking. Similarly, photographers can also efficiently arrange their schedule and manage their bookings.

Thanks to those applications, customers can reach the most appropriate photographers based on criteria such as shooting time, rates, and feedback from other users. Customers would also be provided with a pricing list and can avoid being overcharged. On the other hand, photographers can easily get booked, upload their album, efficiently manage their working schedule and develop their brand thanks to users’ feedback.

## 6. Project Scope & Limitations

### 6.1 Major Features

FE-01: Customer can search for photographers based on requirements

FE-02: Customer can view photographers’ album, services and profile

FE-03: Customer can like photographers’ album

FE-04: Customer can book photographers that meet requirements and be able to cancel bookings due to inconvenient time and weather

FE-05: Customer and Photographer can view booking history

FE-06: Customer and Photographer can view direction to booking history

FE-07: Customer and Photographer can edit and cancel booking

FE-08: Customers can rate and write review for photographers

FE-09: Photographers can manage their schedule

FE-10: Photographers can create and edit their album

FE-11: Photographers can manage services they provide

FE-12: Photographers can accept or decline customers’ request

FE-13: Photographers and Admin can view customers’ review

FE-14: Admin can disable customers’ review

FE-15: Admin, Photographer, Customer sign up and sign in

FE-16: Admin, Photographer, Customer add, delete and edit profile and change password

FE-17: Admin, Photographer, Customer có thể xem forum

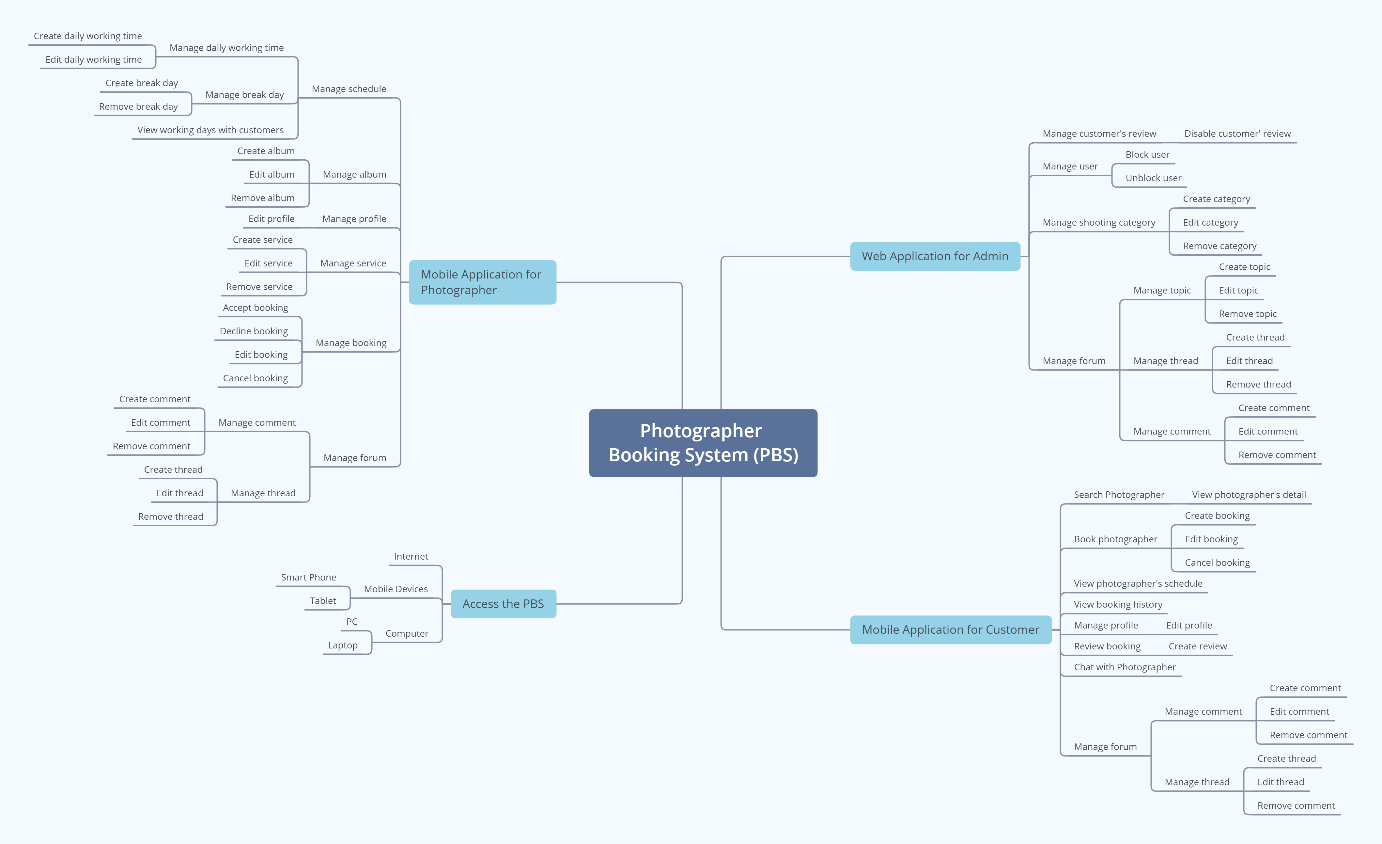
FE-18: Admin có thể tạo, chỉnh sửa và xóa topic của mình

FE-19: Admin, Photographer, Customer có thể tạo và chỉnh sửa và xóa thread của mình

FE-20: Admin, Photographer, Customer có thể tạo, chỉnh sửa và xóa comment của mình trong thread

FE-21: Admin có thể xóa comment và thread

FE-22: Customer và Photographer có thể chat với nhau



**Figure 1 - Mind Map**

### 6.2 Limitations & Exclusions

LI-01: The application has not supported payment.

LI-02: The system has not had a photo verification feature.

LI-03: The system has not provided a website version to both customers and photographers. Users may find it uncomfortable to use applications.

# Project Management Plan

## 1. Overview

### 1.1 WBS & Estimation

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **WBS Item** | **Complexity** | **Est. Effort**  **(man-days)** |
| **1** | **Initiating** |  | 5 |
| 1.1 | Kick-off project   * Identify Project Goals & Vision * Get the Right Team Together * Define the Right Key Performance Indicators (KPIs) or Timeline * Project Tools & Methodologies * Project Kick-off Meeting Planning * Project Kick-off Meeting | Medium | 1 |
| 1.2 | Develop business case   * Confirm the opportunity * Analyse and develop shortlisted options * Evaluate the options * Implementation strategy * Recommendation | Medium | 2 |
| 1.3 | Assign Project Manager | Simple | 1 |
| 1.4 | Identify stake-holder   * Identifying * Analysing  1. Stakeholder Type 2. Contribution (value) 3. Legitimacy 4. Willingness to engage 5. Influence 6. Involvement  * Mapping | Medium | 1 |
| **2** | **Planning** |  | 14 |
| 2.1 | Create business plan   * Executive summary * Opportunity * Execution * Company and management summary * Financial plan * Appendix | Medium | 2 |
| 2.2 | Collect requirement | Complex | 7 |
| 2.3 | Define scope   * Identify the project needs * Confirm the objectives and goals of the Project * Project Scope description * Expectations and acceptance * Identify constraints * Identify necessary changes | Medium | 1 |
| 2.4 | Define vision   * Define outcome * Define what unique brings to the above outcome * Apply Add relatable, human, aspects | Medium | 1 |
| 2.5 | Create WBS   * Identify the most important pieces of scope * Break down the project * Identify the known attributes for each activity | Medium | 1 |
| 2.6 | Create Risk Management Plan   * Identify the Risk * Analyse the Risk * Evaluate or Rank the Risk * Treat the Risk * Monitor and Review the Risk | Medium | 1 |
| 2.7 | Create Human Resource Plan   * Analyse the objectives * Make an inventory of current human resources * Forecast your HR demand * Determine the number and extent of skills gaps * Draw up an action plan * Integrate and implement the plan * Monitoring, measurement, and feedback | Simple | 1 |
| **3** | **Executing** |  | 22 |
| 3.1 | Analysis |  |  |
| 3.1.1 | Analyse Requirements   * Identify customer's needs. * Evaluate system for feasibility. * Perform economic and technical analysis. * Allocate functions to system elements. * Establish schedule and constraints. * Create system definitions. | Medium | 2 |
| 3.1.2 | Create Use case diagram   * Identify the Actors * For each category of users, identify all roles played by the users relevant to the system * Identify what are the users required the system to be performed to achieve these goals * Create use cases for every goal * Structure the use cases | Medium | 1 |
| 3.1.3 | Create Use case specification   * Create a use case model showing the use cases and actors * Create an overview of the steps (content) of the use case * Write the use case specification | Complex | 1 |
| 3.1.4 | Create SRS   * Create an Outline * Start with a Purpose * Give an Overview * Detail Specific Requirements * Get Approval for the SRS | Medium | 2 |
| **3.2** | Design |  |  |
| 3.2.1 | Design System Architecture   * Analyse the requirements * Define use cases for the system * Identify processor/modules to implement the use cases * Select operating system and hardware platform * Define sequence diagrams | Medium | 2 |
| 3.2.2 | Design System Detail (Diagrams) | Medium | 3 |
| 3.2.3 | Design Database   * Determine the purpose * Find and organize the information required * Divide the information into tables * Turn information items into columns * Specify primary keys * Set up the table relationships * Refine your design * Apply the normalization rules | Medium | 3 |
| 3.2.4 | Design User Interface   * Briefing * Analytics * Wireframing * UX prototype * UI creation | Complex | 3 |
| **3.3** | Prototype |  |  |
| 3.3.1 | Define storyboard | Medium | 1 |
| 3.3.2 | Create paper prototype | Simple | 1 |
| 3.3.3 | Create digital prototype | Complex | 3 |
| **3.4** | Implementation |  |  |
| 3.4.1 | Coding/ implement every module defined in SRS | Medium | 30 |
| **3.5** | Perform quality assurance |  | 7 |
| 3.5.1 | Perform unit testing | Simple | 1 |
| 3.5.2 | Perform integration testing | Medium | 2 |
| 3.5.3 | Perform system testing | Medium | 2 |
| 3.5.4 | Perform acceptance testing | Medium | 2 |
| **3.6** | Support |  | 7 |
| 3.6.1 | Training | Medium | 2 |
| 3.6.2 | Documentation | Medium | 2 |
| 3.6.3 | User Support | Simple | 1 |
| 3.6.4 | Enhancement | Medium | 2 |
| **4** | **Monitoring** |  | 3 |
| 4.1 | Validate Scope | Medium | 2 |
| 4.2 | Monitor project work | Medium | 1 |
| **5** | **Controlling** |  | 5 |
| 5.1 | Control scope | Medium | 1 |
| 5.2 | Control quality | Medium | 1 |
| 5.3 | Control risks | Medium | 1 |
| 5.4 | Control schedule | Medium | 1 |
| 5.5 | Control communication | Simple | 1 |
| **6** | **Closing** |  | 1 |
| 6.1 | Close project | Simple | 1 |
| Total Estimated Effort (man-days) | | | FINAL PRESENTATION |
|  | | | 94 |

**Table 3 - WBS & Estimation**

### 1.2 Project Objectives

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Quality Stage** | **No. of Defects** | **% of Defect** | **Notes** |
| 1 | Initiating | 2 | 8.7 | Must be completed in the first stage. |
| 2 | Planning | 4 | 17.4 | Plan to complete at least 80% of the project, and time distribution. |
| 3 | Analysis | 4 | 17.4 | Detailed project analysis. |
| 4 | Design | 2 | 8.8 | Design diagram and database. |
| 5 | Prototype | 4 | 17.4 | Make a prototype of a project |
| 6 | Implementation | 1 | 4.3 | Implementation of the plans. |
| 7 | Perform quality assurance | 2 | 8.8 | Write test cases to check the quality of the project. |
| 8 | Support | 1 | 4.3 | Discuss unclear issues with the instructor. |
| 9 | Monitoring | 1 | 4.3 | Monitoring the steps taken. |
| 10 | Controlling | 1 | 4.3 | Track the progress of the project |
| 11 | Closing | 1 | 4.3 | Review code and document. |
| ***Total*** | | ***23*** | ***100%*** |  |

**Table 4 - Project Objectives**

### 1.3 Project Risks

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Risk Description** | **Impact** | **Possibility** | **Response Plans** |
| 1 | Most members find it unfamiliar to use flutter or spring framework | High | 30% | Hold training session to get familiar with them |
| 2 | Scrum model is applied for the first time hence there may be difficulties in system development | High | 30% | Hold daily meeting to avoid misunderstandings among members |
| 3 | Wrong estimation of time needed | Medium | 10% | Regularly track progress and update time estimation |
| 4 | Unexpected scope change | High | 10% | Be careful in planning to minimize impact of scope change |
| 5 | Poor code quality | Medium | 20% | Strictly follow code convention, carry out acceptance test to ensure quality source code |

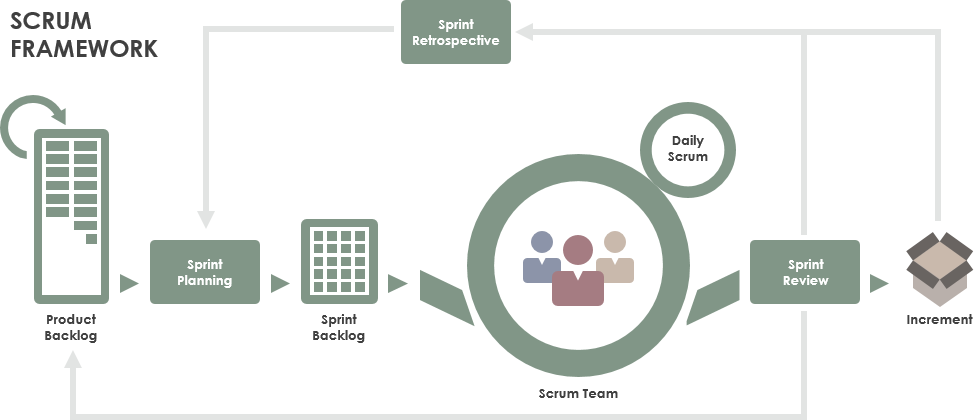
**Table 5 - Project Risks**

## 2. Management Approach

Scrum model would be implemented in the development of this project because it is appropriate for the process.

### 2.1 Project Process

We choose Scrum model to do this project because of the following reasons:

* The feature can be changed during the developing process to be suitable with the Scrum model we can adapt to change it easily and help reduce risk.
* Scrum allows us to learn new frameworks or technology and develop projects at the same time.
* Because dividing the project allows the team to check partial inspection, identify and fix problems faster.

**Figure 2 - Scrum Framework**

*Reference:* [*https://www.scrum.org/resources/what-is-scrum*](https://www.scrum.org/resources/what-is-scrum)

### 2.2 Quality Management

To increase the pace of the project, the team take following measures:

* Observe similar available applications.
* Discuss all requirements together to achieve the best quality.
* Hold daily meetings so that all members are acknowledged of the progress and others’ ability.

### 2.3 Training Plan

|  |  |  |  |
| --- | --- | --- | --- |
| **Training Area** | **Participants** | **When, Duration** | **Waiver Criteria** |
| Java Spring Boot | All team members | 3 days | Mandatory |
| Dart, Flutter | All team members | 4 days | Mandatory |
| Staruml | All team members | 3 days | Mandatory |
| MySql | All team members | 3 days | Mandatory |
| Git, Github | All team members | 1 days | Mandatory |

**Table 6 - Training Plan**

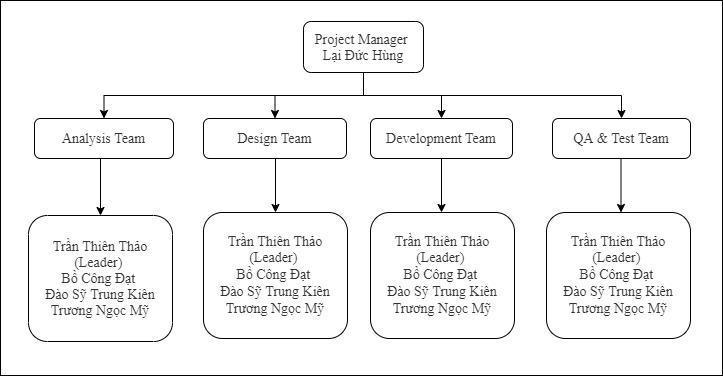
## 3. Master Schedule

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Deliverable** | **Due Date** | **Deliverable Scope** |
| 1 | Project Plan | 17/09/2020 | Make a goal plan, list tasks to do, time distribution. |
| 2 | Use case diagram | 24/09/2020 | Design use case diagram |
| 3 | Conceptual diagram | 24/09/2020 | Design conceptual diagram |
| 4 | ER diagram | 29/09/2020 | Design ER diagram |
| 5 | Database | 29/09/2020 | Design database |
| 6 | Design | 08/10/2020 | Architecture Design, Detailed Design |
| 7 | Screen prototype | 13/10/2020 | Design screen of application |
| 8 | Report 1 | 20/10/2020 | Project introduction |
| 9 | Report 2 | 20/10/2020 | Project management plan |
| 10 | Report 3 | 20/10/2020 | Software requirement specification |
| 11 | Report 4 | 20/10/2020 | Software design document |
| 12 | Report 5 | 26/11/2020 | Test case document, test documentation |
| 13 | Report 6 | 26/11/2020 | Software user guides |
| 14 | Report 7 | 26/11/2020 | Final project report |
| 15 | Code package 1 | 15/10/2020 | Code & Unit test, System test cases |
| 16 | Code package 2 | 05/11/2020 | Code & Unit test, System test cases |
| 17 | Code package 3 | 10/11/2020 | Code & Unit test, System test cases |
| 18 | Code front-end 1 | 15/10/2020 | Code & design screen |
| 19 | Code front-end 2 | 10/11/2020 | Code & design screen |
| 20 | Code front-end 3 | 24/11/2020 | Code & design screen |
| 21 | Final Package | 08/12/2020 | Final Codes & documents, User manual |

**Table 7 - Master Schedule**

## 4. Project Organization

### 4.1 Team & Structures

***Figure 3 - Team & Structures***

### 4.2 Roles & Responsibilities

|  |  |
| --- | --- |
| **Role** | **Responsibility** |
| Project Manager | Give guidance, check, and help correct mistakes in project |
| Analysis Leader | Give guidance and adjust projects’ requirement |
| Analysis Member | Create and update projects’ requirement |
| Design Leader | Give guidance, design and write code for website layouts |
| Design Member | Design and write code for website layouts |
| Technical Leader | Introduce new techniques to the team |
| Test Leader | Write test case and test plan for the project |
| Test Member | Write test case and test plan for the project |

**Table 8 - Roles & Responsibilities**

## 5. Project Communication

### 5.1 Communication Plan

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Communication Item** | **Who/ Target** | **Purpose** | **When, Frequency** | **Type, Tool, Method(s)** |
| Kick off meeting | Project supervisor, project team | Introduce the project | Once | Meeting |
| Project status report | Project supervisor, Project team | Discuss project status, resolve potential issues or delays | Weekly | Meeting |
| Daily stand up meeting | Project team | Review each member work status or any happening problems | Daily | Meeting |
| Meeting minute reports | Project supervisor | Inform supervisor about meeting content | Weekly | Email |

**Table 9 - Communication Plan**

### 5.2 External Interface

#### a. FU Contacts

|  |  |  |  |
| --- | --- | --- | --- |
| **Function** | **Contact Person**  **(name, position)** | **Contact address**  **(email, telephone)** | **Responsibility** |
| Supervisor | Lại Đức Hùng | Hungld5@fe.edu.vn | - Provide document template - Give instruction to project team - Review deliverables - Supervise project status  - Answer questions about the project |

**Table 10 - FU Contacts**

#### b. Customer Contacts

|  |  |  |  |
| --- | --- | --- | --- |
| **Function** | **Contact Person**  **(name, position)** | **Contact address**  **(email, telephone)** | **Responsibility** |
| Supervisor | Lại Đức Hùng | Hungld5@fe.edu.vn | - Provide document template - Give instruction to project team - Review deliverables - Supervise project status  - Answer questions about the project |

**Table 11 - Customer Contacts**

## 6. Configuration Management

### 6.1 Tools & Infrastructures

|  |  |
| --- | --- |
| **Programming languages** | Java, Dart |
| **Framework** | Spring framework, Flutter |
| **API** | API |
| **DBMS** | MySQL |
| **IDEs/Editors** | Visual Studio Code, Android Studio, IntelliJ |
| **UML tools** | StarUML, Draw.io |
| **Version Control** | Github, Git |
| **Deployment server** | Heroku, AWS EC2 (backup) |
| **Project management tool** | Microsoft Excel |

**Table 12 - Tools & Infrastructures**

### 6.2 Document Management

*https://github.com/thienthao/pbs/tree/master/Documents*

### 6.3 Source Code Management

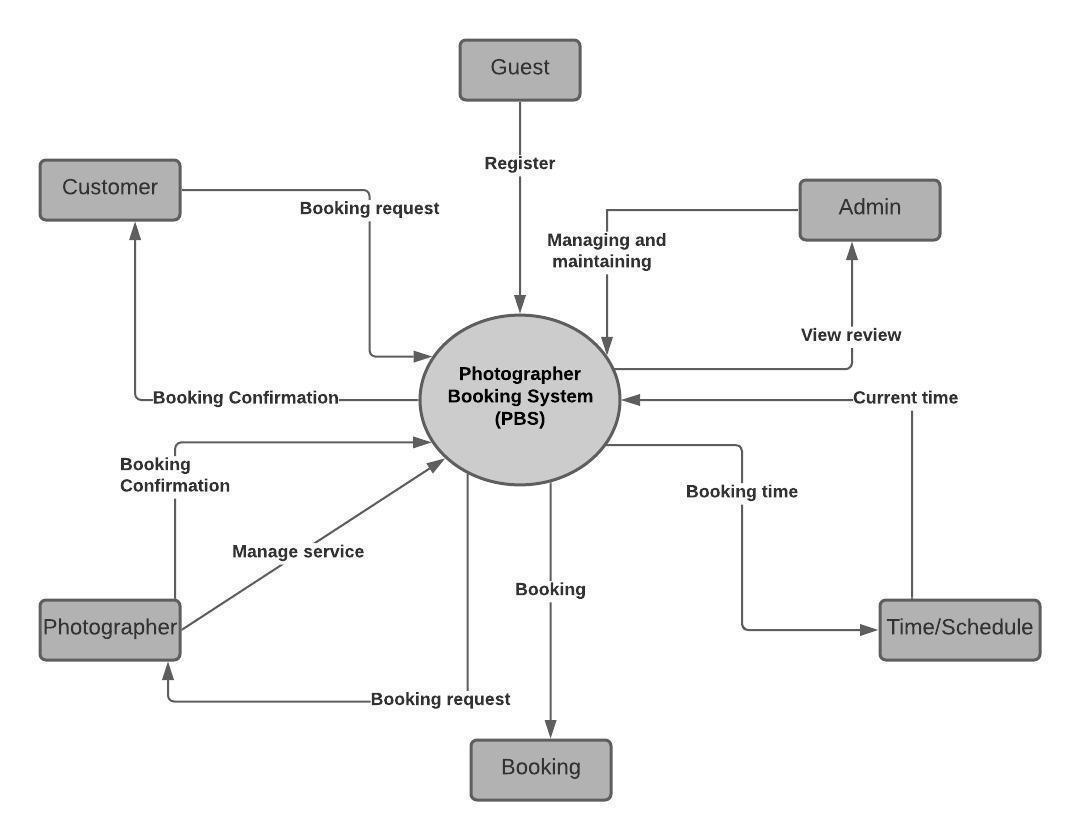
*https://github.com/thienthao/pbs*

# Software Requirement Specification

## 1. Overall Description

### 1.1 Product Overview

Photographer Booking System is a new system that helps users find nearby photographers effectively. The context diagram below describes the 1.0 version of the system and external entities that interact with it. The system is intended to be used in the city area and has access to many photographers and studios.

**Figure 4 – Product Overview**

### 1.2 Business Rules

|  |  |
| --- | --- |
| **ID** | **Rule Definition** |
| BR-01 | Photographers have to update their customers’ request status within 24 hours. |
| BR-02 | Customers can only book photographers within 3 months. |
| BR-03 | Customers cannot remove the reviews they have made. |
| BR-04 | After making a request, the customer can update information related to that request before the photographer confirms it. |
| BR-05 | Customers can cancel a booking request when the request’s status is “Pending” or “Upcoming”. |
| BR-06 | If a customer cancels a request with status “Upcoming”, he/she must give a reason and wait for the photographer to confirm. |
| BR-07 | The time for a new request cannot be in the past. |
| BR-08 | Photographers cannot remove their customers’ reviews. |
| BR-09 | The admin locks users when they violate business rules. |
| BR-10 | Administrators are not permitted to write reviews and rate for photographers. |
| BR-11 | Administrators are not permitted to remove photographers’ comments. |
| BR-12 | Only the admin can add a shooting category. |
| BR-13 | Only the admin can edit the shooting category. |
| BR-14 | Only the admin can remove the shooting category. |
| BR-15 | After a new request is made for a photographer, other customers cannot make a new request to that photographer with the same schedule. |
| BR-16 | If customers want to cancel a request with status “Upcoming”, they have to do it within 24 hours. |

**Table 13 - Business Rules**

## 2. User Requirements

### 2.1 Overview

#### a. Use Case Diagram

**Figure 5 - Overview Use Case Diagram**

#### b. System Actors

|  |  |  |
| --- | --- | --- |
| **#** | **Actor** | **Description** |
| 1 | Administrator | The role who manages the whole system |
| 2 | Photographer | The person who receives customers’ requests and their reviews. Photographers will confirm the received requests, inform the customers about shooting day, photos returning day. They will provide services, photos for the system. |
| 3 | Customer | The person who searches for a photographer and sends a request to him. They will receive notifications about the request confirmation, shooting day, photos returning day. They can write reviews about photographers’ service quality after the requests are done. |
| 4 | Guest | The person who can register an account and will receive a response from the system. |

**Table 14 - System Actors**

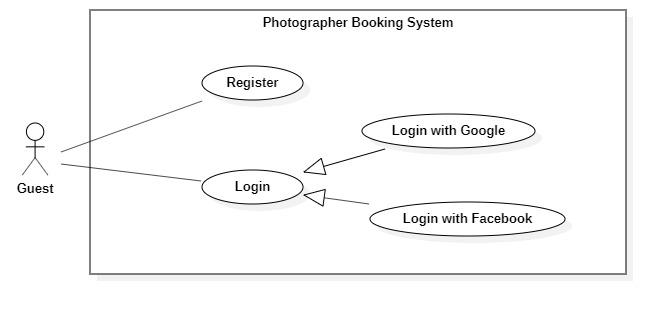
#### c. Use Cases List

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Use Case** | **Primary Actors** | **Secondary Actors** |
| 01 | Register | Guest |  |
| 02 | Login  (Login with google, Login with facebook) | Guest |  |
| 03 | Search photographer | Customer |  |
| 04 | Get photographer’s detail | Customer |  |
| 05 | Book Photographer | Customer |  |
| 06 | Review photographer | Customer |  |
| 07 | Make notification  (Make booking notification) | Customer, Photographer, Handler |  |
| 08 | Get booking  (Cancel, Edit, Accept, Decline) | Customer, Photographer |  |
| 09 | Chat | Customer, Photographer |  |
| 10 | Discuss on forum  (Open thread, Edit thread, Comment thread, Follow thread) | Customer, Photographer, Admin |  |
| 11 | Edit working calendar | Photographer |  |
| 12 | Get customer’s review | Photographer, Admin |  |
| 13 | Manage service  (Add, Edit, Remove) | Photographer |  |
| 14 | Manage album  (Add, Edit, Remove) | Photographer |  |
| 15 | Manage delivery method | Admin |  |
| 16 | Manage variables | Admin |  |
| 17 | Manage user  (Get user’s activity, Unblock user, Block user) | Admin |  |
| 18 | Manage shooting category  (Add, Edit, Remove) | Admin |  |
| 19 | Get profile  (Edit profile) | Authenticated User |  |
| 20 | Logout | Authenticated User |  |
| 21 | Change password | Authenticated User |  |

**Table 15 - Use Cases List**

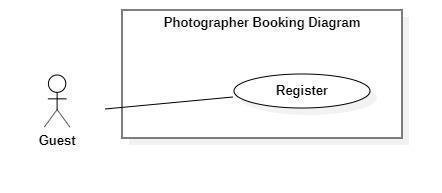
### 2.2 List of use case

#### a. <Guest > Overview use case



**Figure 6 - <Guest> Overview Use Case**

1. ***<Guest> Register***

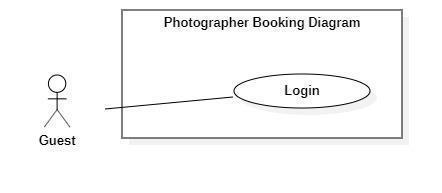


**Figure 7 - <Guest> Register**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-01 Register** | | |
| Created By: | Truong Ngoc My | Date Created: | 08/10/2020 |
| Primary Actor: | Guest | Secondary Actors: |  |
| Trigger: | The Guest taps “Đăng ký” on “Đăng nhập” screen | | |
| Description: | The guest wants to register an account. | | |
| Preconditions: | - Guests launch mobile applications.  - Guest is at the “Đăng nhập” screen.  - Guests have connected to the Internet. | | |
| Post-conditions: | Success: The guest registers an account as a customer or photographer. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The Guest taps “Đăng kí” on “Đăng nhập” screen. | The system shows a “Thông tin đăng ký” screen with the following inputs.  - “Họ và tên”: text input, required.  - “Tên đăng nhập”: text input, required  - “Mật khẩu”: password input, required.  - “Xác nhận mật khẩu”: password input, required.  - “Số điện thoại”: phone number input.  - “Email”: text input, required. | | 2 | The guest taps the “Đăng kí” button. | - The system validates inputted data [Exception 1,2,3,4,5,6].  - The system shows the “Đăng nhập” screen. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | - “Họ và tên” is empty  - “Tên đăng nhập” is empty  - “Mật khẩu” is empty  - “Xác nhận mật khẩu” is empty.  - “Số điện thoại” is empty  - “Email” is empty | System shows an error message: “Thông tin này là bắt buộc” | | 2 | Password and Confirm password are not matched. | System shows an error message: “Mật khẩu nhập lại không khớp”. | | 3 | Register successfully | System shows an error message: “Đăng ký thành công”. | | 4 | “Email” Is invalid | System shows an error message:  “Email mail này không hợp lệ. Vui lòng nhập email khác.” | | 5 | - “Tên đăng nhập” not enough character.  - “Mật khẩu” not enough character. | System shows an error message:  “Bạn vui lòng nhập đủ 8 – 16 ký tự.” | | 6 | “Tên đăng nhập” is exist. | System shows an error message:  “Tên đăng nhập của bạn đã có người sử dụng. Vui lòng nhập tên khác.” | | | |
| Priority: | Medium | | |
| Frequency of Use: | Always | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

**Table 16 - <Guest> Register use case specification**

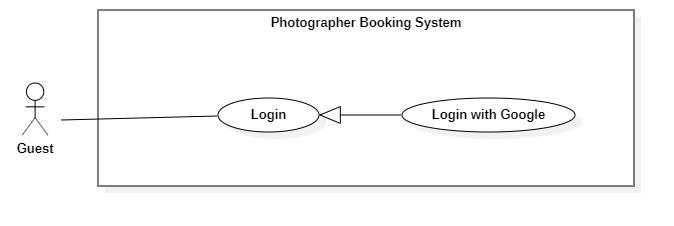
1. ***<Guest> Login***



**Figure 8 - <Guest> Login**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-02 Login** | | |
| Created By: | Truong Ngoc My | Date Created: | 08/10/2020 |
| Primary Actor: | Guest | Secondary Actors: |  |
| Trigger: | The Guest taps “Đăng nhập” on “Đăng nhập” screen | | |
| Description: | The guest wants to login to the system. | | |
| Preconditions: | - Guests launch mobile applications.  - Guest is at the “Đăng nhập” screen.  - Guests have connected to the Internet. | | |
| Post-conditions: | Success: The guest login to the system. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The Guest opened a mobile app. | “Đăng nhập” screen is show with the following inputs:  - “Tên đăng nhập”: text input, required  - “Mật khẩu”: password input, required.  - “Đăng nhập”: button  - “Đăng ký”: button  - “Login with google”: button  - “Login with facebook”: button | | 2 | The guest inputs the name and password and taps the “Đăng nhập” button. | - The system validates inputted data [Exception 1,2]  - The system shows the main screen. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | - “Tên đăng nhập” is empty  - “Mật khẩu” is empty | System shows an error message: “Thông tin này là bắt buộc” | | 2 | Name and Password is incorrect. | System shows an error message: “Thông tin đăng nhập không chính xác”. | | | |
| Priority: | Medium | | |
| Frequency of Use: | Always | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

**Table 17 - <Guest> Login use case specification**

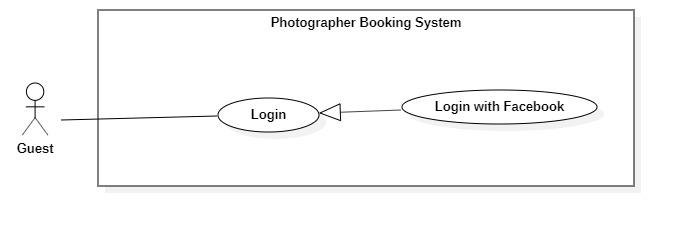
1. ***<Guest> Login with Google***

**Figure 9 - <Guest> Overview login with Google**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-03 Login with google** | | |
| Created By: | Truong Ngoc My | Date Created: | 08/10/2020 |
| Primary Actor: | Guest | Secondary Actors: |  |
| Trigger: | The Guest taps “Login with google” on “Đăng nhập” screen | | |
| Description: | The guest wants to login to the system. | | |
| Preconditions: | - Guests launch mobile applications.  - Guest is at the “Đăng nhập” screen.  - Guests have connected to the Internet. | | |
| Post-conditions: | Success: The guest login with google to the system. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The Guest opened a mobile app. | “Đăng nhập” screen is show with the following inputs:  - “Tên đăng nhập”: text input, required  - “Mật khẩu”: password input, required.  - “Đăng nhập”: button  - “Đăng ký”: button  - “Login with google”: button  - “Login with facebook”: button | | 2 | The guest click “Login with google”. | - The system show popup “Choose an account”, the following inputs: [Exception 1, 2]  + “Enter password”: text, hiden  + “Next”: button | | 3 | The guest click “Next” button. | - The system shows the main screen. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | - “Enter password” is empty | System shows an error message: “Thông tin này là bắt buộc” | | 2 | Password is incorrect. | System shows an error message: “Mật khẩu không chính xác”. | | | |
| Priority: | High | | |
| Frequency of Use: | Always | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

**Table 18 - <Guest> Login with google use case specification**

1. ***<Guest> Login with facebook***

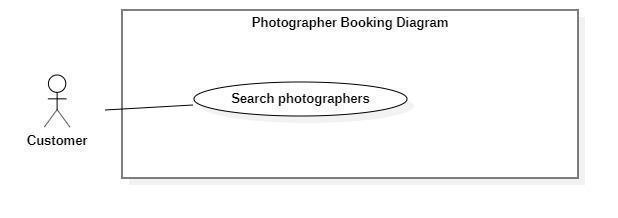
**Figure 10 - <Guest> Overview login with facebook**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-03 Login with google** | | |
| Created By: | Truong Ngoc My | Date Created: | 08/10/2020 |
| Primary Actor: | Guest | Secondary Actors: |  |
| Trigger: | The Guest taps “Login with google” on “Đăng nhập” screen | | |
| Description: | The guest wants to login to the system. | | |
| Preconditions: | - Guests launch mobile applications.  - Guest is at the “Đăng nhập” screen.  - Guests have connected to the Internet. | | |
| Post-conditions: | Success: The guest login with google to the system. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The Guest opened a mobile app. | “Đăng nhập” screen is show with the following inputs:  - “Tên đăng nhập”: text input, required  - “Mật khẩu”: password input, required.  - “Đăng nhập”: button  - “Đăng ký”: button  - “Login with google”: button  - “Login with facebook”: button | | 2 | The guest click “Login with facebook”. | - The system show popup “facebook name”, the following inputs: [Exception 1, 2]  + “Enter password”: text, hiden  + “Next”: button | | 3 | The guest click “Next” button. | - The system shows the main screen. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | - “Enter password” is empty | System shows an error message: “Thông tin này là bắt buộc” | | 2 | Password is incorrect. | System shows an error message: “Mật khẩu không chính xác”. | | | |
| Priority: | High | | |
| Frequency of Use: | Always | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

**Table 19 - <Guest> Login with facebook use case specification**

#### b. <Customer> Overview use case

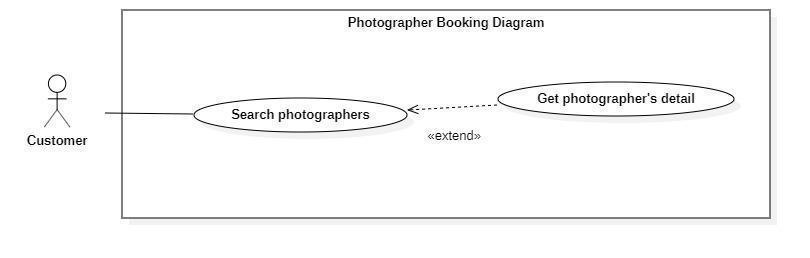
**Figure 11 - <Customer> Overview use case**

1. ***<Customer> Search Photographer***

**Figure 12 - <Customer> Search Photographer**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-04 Search photographer** | | |
| Created By: | Truong Ngoc My | Date Created: | 08/10/2020 |
| Primary Actor: | Customer | Secondary Actors: |  |
| Trigger: | The customer sends request to find photographers | | |
| Description: | The customer needs to find photographers. | | |
| Preconditions: | - The customer has been authenticated.  - The customer has accessed the Internet. | | |
| Post-conditions: | Success: A list of photographers is shown.  Fail: System shows error messages. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The customer input name photographer or service name after clicking the search icon button. | - The system validates inputted data.  - The system shows a list photographer or service name. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | N/A | | |
| Priority: | Medium | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: | N/A | | |
| Other Information: | The customer can view a list of available photographers and categories of the main screen. | | |
| Assumptions: |  | | |

**Table 20 - <Customer> Search photographer use case specification**

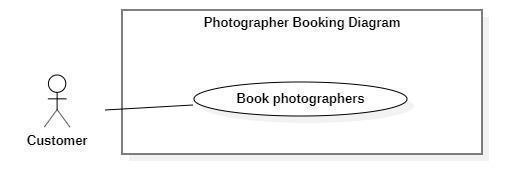
1. ***<Customer> Get photographer’s detai***

**Figure 13 - <Customer> Get photographer's Detail**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-05 Get photographer’s detail** | | |
| Created By: | Truong Ngoc My | Date Created: | 08/10/2020 |
| Primary Actor: | Customer | Secondary Actors: |  |
| Trigger: | The customer taps the photographer or photographer icon. | | |
| Description: | The customer needs to view the profile of the photographer. | | |
| Preconditions: | - The customer has been authenticated.  - The customer has opened location service.  - The customer has accessed the Internet.  - The customer is at the “Danh sách photographer” screen. | | |
| Post-conditions: | Success: The system showed the photographer’s page. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The customer taps on the photographer on the result of a list photographer search or photographer view main screen. | The system shows the photographer profile screen with the following information:  - “Mô tả”: introduction or a brief description.  - “Đánh giá” number of people and number of stars: number of bookings.  - “Các album”: album photo of photographer.  - “Nhận xét”: list of feedbacks.  - “Lịch của photographer”: schedule of photographer.  - “Chọn gói dịch vụ bạn muốn”: list of service of photographer, choose option, required.  - “tiếp tục”: button | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | N/A | | |
| Priority: | High | | |
| Frequency of Use: | Always | | |
| Business Rules: | BR-04 | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

**Table 21 - <Customer> Get photographer's detail use case specification**

1. ***<Customer> Book photographer***

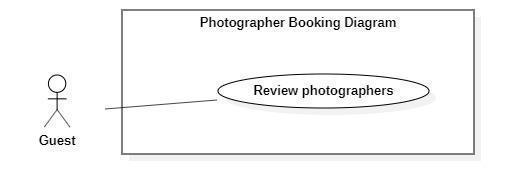


**Figure 14 - <Customer> Book photographer**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-06 Book photographer** | | |
| Created By: | Truong Ngoc My | Date Created: | 08/10/2020 |
| Primary Actor: | Customer | Secondary Actors: |  |
| Trigger: | The customer taps the “Tiếp tục” button. | | |
| Description: | The customer can book a photographer. | | |
| Preconditions: | - The customer has been authenticated.  - The customer has selected a photographer and viewed their profile photographer.  - The customer has accessed the Internet. | | |
| Post-conditions: | Success: The system sends requests to the photographer. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The customer taps “Tiếp tục” button after click choose “Gói chụp trong ngày”. | The system shows the “Đặt lịch với photographer” screen. The customer input following:  - “Photographer”: default.  - “Thời gian chụp”: The customer can choose date and start time.  - “Thời gian nhận ảnh”: choose date.  - “Địa điểm”: input text, required.  - “Phương thức nhận ảnh”: choose a method.  - “Gói dịch vụ”: the customer can change.  - “Tổng cộng”: price.  - “Đặt dịch vụ”: button | | 2 | The customer taps “Đặt dịch vụ” button | - The system validates inputted data  - The system creates a new request and notifies the chosen photographer. [Exception 1,2] | | | |
| Alternative Flows: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The customer taps “Tiếp tục” button after click choose “Gói chụp nhiều ngày”. | The system shows the “Đặt lịch với photographer” screen. The customer input following:  - “Photographer”: default.  - “Thời gian chụp”: The customer can choose date and start time.  - “Thời gian nhận”: choose date.  - “Địa điểm”: input text, required.  - “Gói dịch vụ”: the customer can change.  - “Phương thức nhận ảnh”: choose a method.  - “Tổng cộng”: price.  - “Đặt dịch vụ”: button | | 2 | The customer taps “Đặt dịch vụ” button | - The system validates inputted data  - The system creates a new request and notifies the chosen photographer. [Exception 1,2] | | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | The customer book photographer successfully. | System shows an error message: “Gửi yêu cầu thành công” | | 2 | The customer do not choose “Thời gian chụp”, “Thời gian nhận” and “Địa điểm”. | System shows an error message: “Bạn chưa chọn thông tin” | | | |
| Priority: | High | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: | BR-01, BR-02, BR-07, BR-15 | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

**Table 22 - <Customer> Book photographer use case specification**

1. ***<Customer> Review photographers***

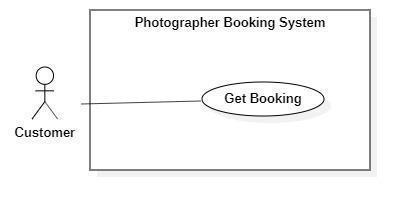


**Figure 15 - <Customer> Review photographers**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-07 Review photographers** | | |
| Created By: | Truong Ngoc My | Date Created: | 08/10/2020 |
| Primary Actor: | Customer | Secondary Actors: |  |
| Trigger: | The customer taps “Đánh giá” to request successfully history. | | |
| Description: | The customer needs to write a comment and rating to the photographer service. | | |
| Preconditions: | - The customer has been authenticated.  - The customer book photographer successfully.  - The customer has accessed the Internet. | | |
| Post-conditions: | Success: The system shows comments and ratings of the customer on the photographer page. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 |  | After the request’s status is updated to “Đã hoàn thành”, “Đánh giá” button and “Bỏ qua” button is shown. | | 2 | The customer taps “Đánh giá” button | The system will go to “Nhận xét” on the photographer page. The customer input with the following:  - star icon: The customer choose star  - “Bình luận”: Text input, optional | | 3 | The customer taps “Gửi” button | - The system validates inputted data [Exception1,2]  - The system shows comments and ratings of customers. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | Enter many characters. | System shows an error message:  “Bạn chỉ được nhập tối đa 150 ký tự” | | 2 | Enter less character. | System shows an error message:  “Bạn phải nhập tối thiểu 50 ký tự”. | | | |
| Priority: | High | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: | BR-03 | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

**Table 23 - <Customer> Review photographers use case specification**

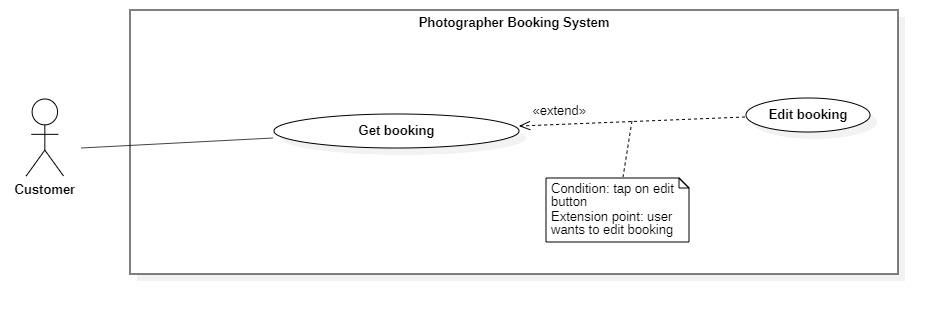
1. ***<Customer> Get booking***



**Figure 16 - <Customer> Get booking**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-08 Get booking** | | |
| Created By: | Tran Thien Thao | Date Created: | 08/10/2020 |
| Primary Actor: | Customer | Secondary Actors: | Photographer |
| Trigger: | The customer navigates to the “Hoat động” screen. | | |
| Description: | The customer can view all requests of theirs. | | |
| Preconditions: | - The customer has been authenticated.  - The customer has accessed the Internet.  - The customer book photographer. | | |
| Post-conditions: | Success: The customer can view request history. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The customer navigates to the “Lịch sử” screen. | The system shows a list of requests the customer has made. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | N/A | | |
| Priority: | Low | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: | N/A | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

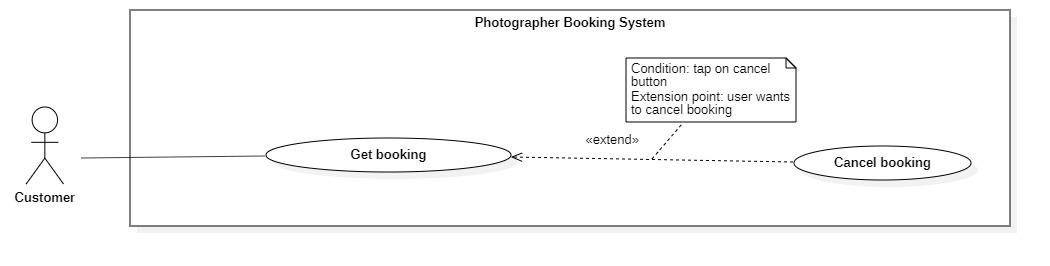
**Table 24 - <Customer> Get booking use case specification**

1. ***<Customer> Edit Booking***

**Figure 17 - <Customer> Edit booking**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-09 Edit booking** | | |
| Created By: | Tran Thien Thao | Date Created: | 08/10/2020 |
| Primary Actor: | Customer | Secondary Actors: |  |
| Trigger: | The customer taps on “Thay đổi” button in the request “Chờ xác nhận”. | | |
| Description: | The customer needs to change information on request. | | |
| Preconditions: | - The customer has been authenticated.  - The customer has accessed the Internet.  - The status of request is “Chờ xác nhận”. | | |
| Post-conditions: | Success: The customer changes a request successfully and the system send request to the photographer. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The customer taps “Chờ xác nhận” request. | The system shows the “Thông tin hoạt động” screen:  - “Trạng thái, thời gian & địa điểm” to include: “Trạng thái”, “Thời gian”, “Thời gian tác nghiệp dự kiến”, “Thời gian nhận ảnh”, “Địa điểm”.  - “Thông tin photographer”: name photographer and rating.  - “Thông tin gói dịch vụ” to include: “tên gói”, “Mô tả”, “Bao gồm các dịch vụ”, “Tổng cộng”  - “Phương thức nhận ảnh”  - “Thay đổi”: button  - “Hủy”: button | | 2 | The customer taps “Thay đổi” button on the item. | The system shows the “Thông tin đặt lịch” screen. The customer input following:  - “Photographer”: default.  - “Thời gian chụp”: The customer can choose date and start time.  - “Thời gian nhận”: choose date.  - “Địa điểm”: input text, required.  - “Gói dịch vụ”: the customer can change.  - “Phương thức nhận ảnh”  - “Tổng cộng”: price.  - “Xác nhận”: button  - “Hủy”: button | | 3 | The customer taps the “Xác nhận” button. | - The system validates inputted data [Exception 2]  - The system creates a new request and notifies the chosen photographer. [Exception 1] | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | The customer book photographer successfully. | System shows an error message: “Gửi yêu cầu thành công” | | 2 | The customer does not choose “Thời gian” and “Địa điểm”. | System shows an error message: “Bạn chưa chọn thông tin” | | | |
| Priority: | Medium | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: | BR-02, BR-04, BR-07, BR-15 | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

**Table 25 - <Customer> Edit booking use case specification**

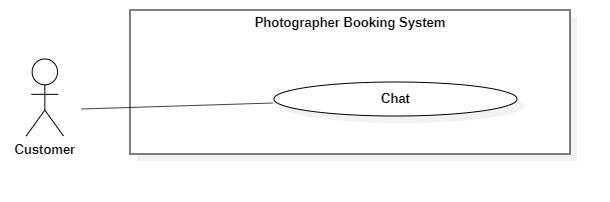
1. ***<Customer> Cancel booking***

**Figure 18 - <Customer> Cancel booking**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-11 Cancel booking** | | |
| Created By: | Tran Thien Thao | Date Created: | 08/10/2020 |
| Primary Actor: | Customer | Secondary Actors: |  |
| Trigger: | The customer taps “Hủy” button on the booking detail. | | |
| Description: | The customer needs to cancel their booking. | | |
| Preconditions: | - The customer has been authenticated.  - The customer has accessed the Internet.  - The status of request is “Chờ xác nhận” or “Sắp diễn ra”. | | |
| Post-conditions: | Success: the customer cancels booking successfully. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The customer taps “Chờ xác nhận” request. | The system shows the “Thông tin hoạt động” screen:  - “Trạng thái, thời gian & địa điểm” to include: “Trạng thái”, “Thời gian”, “Thời gian tác nghiệp dự kiến”, “Thời gian nhận ảnh”, “Địa điểm”.  - “Thông tin photographer”: name photographer and rating.  - “Thông tin gói dịch vụ” to include: “tên gói”, “Mô tả”, “Bao gồm các dịch vụ”, “Tổng cộng”  - “Phương thức nhận ảnh”  - “Thay đổi”: button  - “Hủy”: button | | 2 | The customer taps “Hủy” button. | The system shows notifies the chosen customer: [Exception 2]  - “Có”: button  - “Không”: button | | 3 | The customer taps the “Có” button. | - The system update requests the chosen photographer. | | | |
| Alternative Flows: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The customer taps the “Sắp diễn ra” request. | The system shows the “Thông tin hoạt động” screen:  - “Trạng thái, thời gian & địa điểm” to include: “Trạng thái”, “Thời gian”, “Thời gian tác nghiệp dự kiến”, “Thời gian nhận ảnh”, “Địa điểm”.  - “Thông tin photographer”: name photographer and rating.  - “Thông tin gói dịch vụ” to include: “tên gói”, “Mô tả”, “Bao gồm các dịch vụ”, “Tổng cộng”  - “Phương thức nhận ảnh”  - “Thay đổi”: button  - “Hủy”: button | | 2 | The customer taps “Hủy” button. | The system shows notifies the chosen customer: [Exception 2]  - “Có”: button  - “Không”: button | | 3 | The customer taps the “Có” button. | - The system update requests the chosen photographer. | | 4 |  | The system shows “Hủy cuộc hẹn” popup. The customer input following:  -” Hủy cuộc hẹn”: input text “Lý do hủy cuộc hẹn của bạn là gì”, required.  - “xác nhận”: button | | 5 | The customer input reason and taps “Xác nhận” button. | The system update requests the chosen photographer. | | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | The customer taps “Hủy” button. | System shows an error message: “Bạn có muốn hủy lần đặt photographer này không?” | | | |
| Priority: | High | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: | BR-05, BR-06, BR-16 | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

**Table 26 - <Customer>Cancel booking use case specification**

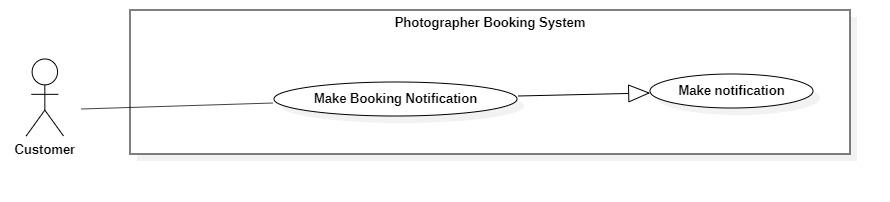
1. ***<Customer> Chat***

**Figure 19 - <Customer> Chat**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-12 Chat** | | |
| Created By: | Tran Thien Thao | Date Created: | 08/10/2020 |
| Primary Actor: | Customer, photographer | Secondary Actors: | Photographer |
| Trigger: | The customer click “Chat” icon. | | |
| Description: | The customer chat with photographer. | | |
| Preconditions: | - The customer has been authenticated.  - The customer has accessed the Internet.  - The customer book photographer. | | |
| Post-conditions: | Success: The customer can chat with photographer. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The customer click “Chat” icon. | The system shows “Chat” screen. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | N/A | | |
| Priority: | Low | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: | N/A | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

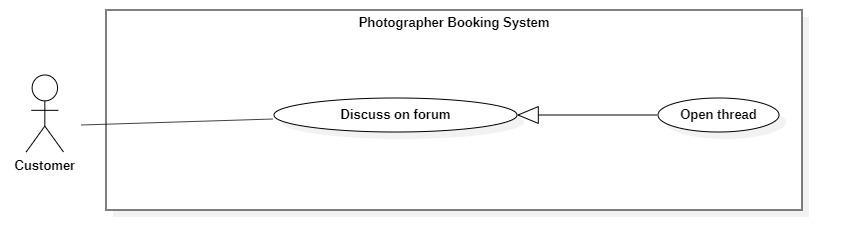
**Table 27 - <Customer> Chat use case specification**

1. ***<Customer> Make booking notification***

**Figure 20 - <Customer> Overview make booking notification**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-13 Make booking notification** | | |
| Created By: | Dao Sy Trung Kien | Date Created: | 08/10/2020 |
| Primary Actor: | Customer, Photographer | Secondary Actors: | System handler |
| Trigger: | The customer send request to the photograper.  The photographers respond to requests for customers. | | |
| Description: | The customer receives notices of his request sent to the photographer and also is the photographer. | | |
| Preconditions: | - The customer send request successfully. | | |
| Post-conditions: | Success: The system make booking notification and sends to the customer | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | System handler retrieves make booking notification. | The system stores request and makes notification information to return to systems handler. | | 2 | System handler retrieves notification information and sends to customers. |  | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Make notification process encountered an internal error. | Error detail will be tracked in log file. System outputs error message and sends to user. | | | |
| Priority: | Low | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: | N/A | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

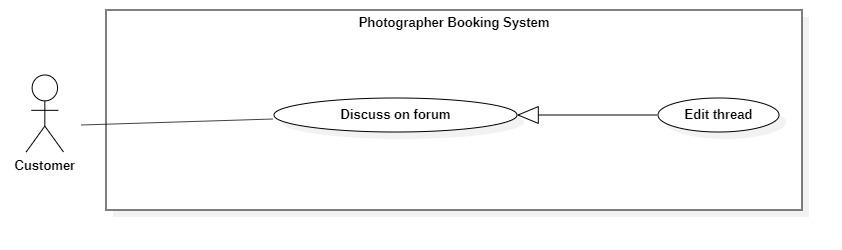
**Table 28 - <Customer> Make booking notification use case specification**

1. ***<Customer> Open thread***

**Figure 21 - <Customer> Overview open thread**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-15 Open thread** | | |
| Created By: | Tran Thien Thao | Date Created: | 08/10/2020 |
| Primary Actor: | Customer, Photographer, Admin | Secondary Actors: |  |
| Trigger: | The customer choose navigates to the “Diễn đàn” screen after choose add icon. | | |
| Description: | The customer open thread. | | |
| Preconditions: | - The customer has been authenticated.  - The customer has accessed the Internet. | | |
| Post-conditions: | Success: The customer can open thread. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The customer choose navigates to the “Diễn đàn” screen. | The system shows “Diễn đàn” screen”.  - “Xu hướng gần đây”: main topic.  - “Gần đây nhất”: main topic.  - “Danh mục”: category.  - “Add” icon. | | 2 | The customer click “Add” icon. | The system shows “Chủ đề mới” screen. The following inputs:  - “Chủ đề”: choose a method.  - “Tên chủ đề”: text, requid.  - “Bình luận của bạn”: text, requid  - “Hình ảnh”: choose the picture  - “V”: button | | 3 | The customer click “V” button. | - The system shows message.  - The system shows thread. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | The customer open thread successfully. | System shows an error message: “Thêm chủ đề thành công” | | 2 | “Chủ đề”, “Tên chủ đề” and “Bình luận của bạn” is empty. | System shows an error message: “Bạn chưa chọn thông tin” | | | |
| Priority: | Low | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: | N/A | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

**Table 29 - <Customer> Open thread use case specification**

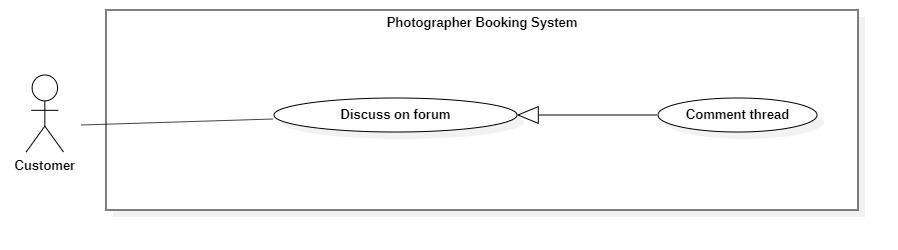
1. ***<Customer> Edit thread***

**Figure 22 - <Customer> Overview edit thread**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-16 Edit thread** | | |
| Created By: | Tran Thien Thao | Date Created: | 08/10/2020 |
| Primary Actor: | Customer, Photographer, Admin | Secondary Actors: |  |
| Trigger: | The customer choose navigates to the “Diễn đàn” screen after choose edit thread. | | |
| Description: | The customer edit thread. | | |
| Preconditions: | - The customer has been authenticated.  - The customer has accessed the Internet.  - The customer open thread. | | |
| Post-conditions: | Success: The customer can edit thread. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The customer choose navigates to the “Diễn đàn” screen. | The system shows “Diễn đàn” screen”.  - “Xu hướng gần đây”: main topic.  - “Gần đây nhất”: main topic.  - “Danh mục”: category.  - “Add” icon. | | 2 | The customer click thread want to edit after click “Điều hành” button. | The system shows thread screen. The following inputs:  - “Chủ đề”: choose a method.  - “Tên chủ đề”: text, requid.  - “Bình luận của bạn”: text, requid  - “Hình ảnh”: choose the picture  - “V”: button | | 3 | The customer click “V” button. | - The system shows message.  - The system shows thread. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | The customer edit thread successfully. | System shows an error message: “Chỉnh sửa thành công” | | 2 | “Chủ đề”, “Tên chủ đề” and “Bình luận của bạn” is empty. | System shows an error message: “Bạn chưa chọn thông tin” | | | |
| Priority: | Low | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: | N/A | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

**Table 30 - <Customer> Edit thread use case specification**

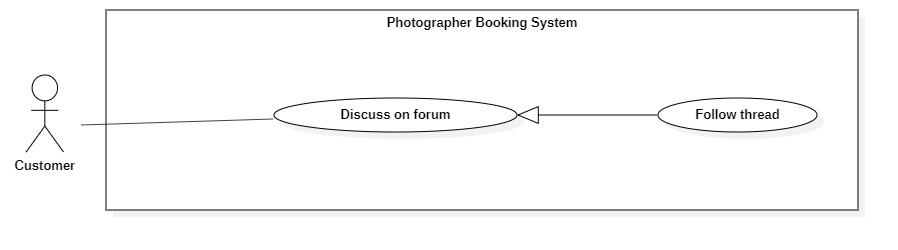
1. ***<Customer> Comment thread***



**Figure 23 - <Customer> Overview comment thread**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-17 Comment thread** | | |
| Created By: | Tran Thien Thao | Date Created: | 08/10/2020 |
| Primary Actor: | Customer, Photographer, Admin | Secondary Actors: |  |
| Trigger: | The customer choose navigates to the “Diễn đàn” screen after choose thread want to comment. | | |
| Description: | The customer comment thread. | | |
| Preconditions: | - The customer has been authenticated.  - The customer has accessed the Internet. | | |
| Post-conditions: | Success: The customer can comment thread. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The customer choose navigates to the “Diễn đàn” screen. | The system shows “Diễn đàn” screen”.  - “Xu hướng gần đây”: main topic.  - “Gần đây nhất”: main topic.  - “Danh mục”: category.  - “Add” icon. | | 2 | The customer click thread want to comment. | The system shows thread screen. The following inputs:  - “Chủ đề”: choose a method.  - “Tên chủ đề”: text, requid.  - “Nội dung”: text.  - “Bình luận”: text.  - “Bình luận”: button | | 3 | The customer click “Bình luận” button. | - The system shows “Trả lời: screen. The following input:  + “Viết bình luận của bạn”: text, requid.  + “Hình ảnh”: choose the picture.  + “V”: button. | | 4 | The customer click “V” button. | - The system shows message.  - The system shows comment into the thread. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | The customer comment thread successfully. | System shows an error message: “Bình luận thành công” | | 2 | “Viết bình luận của bạn” is empty | System shows an error message: “Thông tin này là bắt buộc” | | | |
| Priority: | Low | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: | N/A | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

**Table 31 - <Customer> Comment thread use case specification**

1. ***<Customer> Follow thread***

**Figure 24 - <Customer> Overview follow threar**

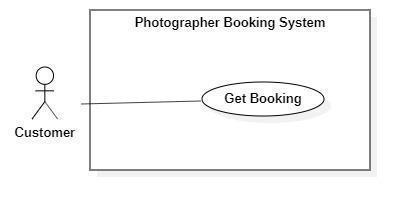
|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-18 Follow thread** | | |
| Created By: | Tran Thien Thao | Date Created: | 08/10/2020 |
| Primary Actor: | Customer, Photographer, Admin | Secondary Actors: |  |
| Trigger: | The customer choose navigates to the “Diễn đàn” screen after choose thread want to follow | | |
| Description: | The customer follow thread. | | |
| Preconditions: | - The customer has been authenticated.  - The customer has accessed the Internet. | | |
| Post-conditions: | Success: The customer can follow thread. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The customer choose navigates to the “Diễn đàn” screen. | The system shows “Diễn đàn” screen”.  - “Xu hướng gần đây”: main topic.  - “Gần đây nhất”: main topic.  - “Danh mục”: category.  - “Add” icon. | | 2 | The customer click thread want to follow. | The system shows thread screen. The following inputs:  - “Chủ đề”: choose a method.  - “Tên chủ đề”: text, requid.  - “Bình luận của bạn”: text, requid  - “Hình ảnh”: choose the picture  - “V”: button | | 3 | The customer click “Follow” button. | - The system shows message.  - The system shows thread. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | N/A | | |
| Priority: | Low | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: | N/A | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

**Table 32 - <Customer> Follow thread use case specification**

#### c. <Photographer> Overview Use Case

**Figure 25 - <Photographer> Overview use case**

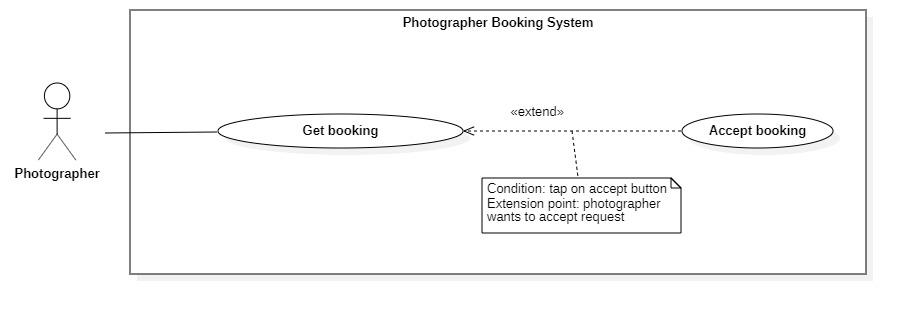
1. ***<Photographer> Get Booking***



**Figure 26 - <Photographer> Get booking**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-19 Get booking** | | |
| Created By: | Bo Cong Dat | Date Created: | 08/10/2020 |
| Primary Actor: | Photographer | Secondary Actors: |  |
| Trigger: | Photographers navigates to the “Trang chủ” in Photographer’s application | | |
| Description: | The photographer can view a list of their bookings including all status. | | |
| Preconditions: | - The photographer has been authenticated.  - The photographer has access to the Internet. | | |
| Post-conditions: | Success: the system shows lists of requests on the “Lịch làm việc của tôi” screen. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 |  | The system shows all requests of customer to the main screen of photographer. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | N/A | | |
| Priority: | High | | |
| Frequency of Use: | Always | | |
| Business Rules: | N/A | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

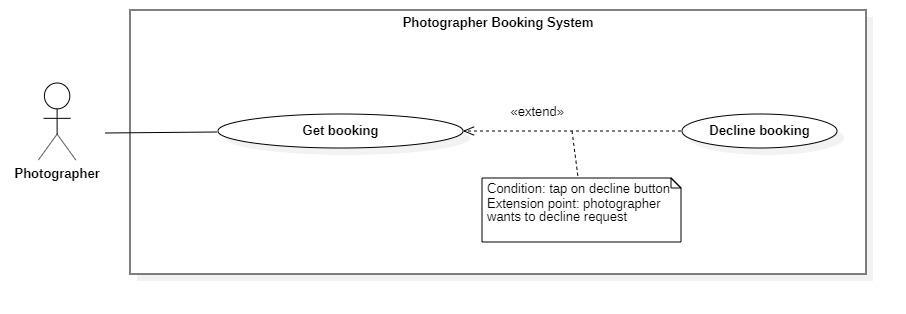
**Table 33 - <Photographer>Get booking use case specification**

1. ***<Photographer> Accept Booking***

**Figure 27 - <Photographer> Accept Booking**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-20 Accept booking** | | |
| Created By: | Bo Cong Dat | Date Created: | 08/10/2020 |
| Primary Actor: | Photographer | Secondary Actors: |  |
| Trigger: | The photographer taps the “Đồng ý” button. | | |
| Description: | The photographer can accept requests from the customer. | | |
| Preconditions: | - The photographer has been authenticated.  - The photographer has access to the Internet.  - The status of request is “Chờ xác nhận”. | | |
| Post-conditions: | Success: the system change status of request is “Sắp diễn ra”. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The photographer taps the “Đồng ý” button when dragging the request to the left. | - The system will change the status of request to “Sắp diễn ra”.  - The system sends notifications to the customer. [Exception 1] | | | |
| Alternative Flows: | |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | The photographer tabs the request “Chờ xác nhận” status. | The system shown detail of request:  - “Trạng thái, thời gian & địa điểm”: text  - “Thông tin khách hàng”: text  - “Thông tin gói dịch vụ”: text  - “Đồng ý”: button  - “Hủy”: button | | 2 | The photographer taps the “Đồng ý” button | - The system will change the status of request to “Sắp diễn ra”.  - The system sends notifications to the customer. [Exception 1] | | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | The system sends notifications to the customer. | System shows an error message: “Yêu cầu của bạn đã được xác nhận”. | | | |
| Priority: | High | | |
| Frequency of Use: | Always | | |
| Business Rules: | BR-01 | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

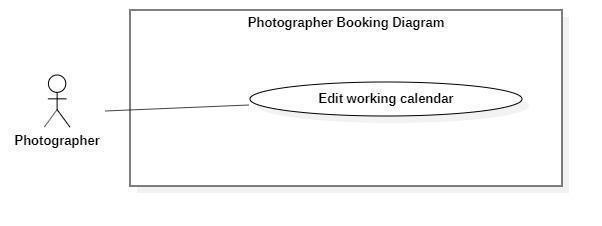
**Table 34 - <Photographer> Accept Booking use case specification**

1. ***<Photographer> Decline Booking***

**Figure 28 - <Photographer> Decline Booking**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-21 Decline booking** | | |
| Created By: | Bo Cong Dat | Date Created: | 08/10/2020 |
| Primary Actor: | Photographer | Secondary Actors: |  |
| Trigger: | The photographer taps “Hủy” button. | | |
| Description: | The photographer can decline a request from the customer. | | |
| Preconditions: | - The photographer has been authenticated.  - The photographer has access to the Internet.  - The status of request is “Chờ xác nhận”. | | |
| Post-conditions: | Success: the system sends notifications to the customer.F | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The photographer taps the “Hủy” button when dragging the request to the left. | - The system sends notification to the customer. [Exception 1] | | | |
| Alternative Flows: | |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | The photographer tabs the request “Chờ xác nhận” status. | The system shown detail of request:  - “Trạng thái, thời gian & địa điểm”: text  - “Thông tin khách hàng”: text  - “Thông tin gói dịch vụ”: text  - “Đồng ý”: button  - “Hủy”: button | | 2 | The photographer taps the “Hủy” button | The system sends notification to the customer. [Exception 1] | | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | The system sends notification to the customer. | System shows an error message: “Yêu cầu của bạn đã bị từ chối” | | | |
| Priority: | High | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: | BR-01 | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

**Table 35 - <Photographer> Cancel Booking Use Case Specification**

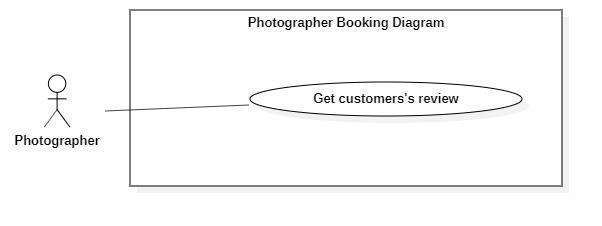
1. ***<Photographer> Edit working calendar***

**Figure 29 - <Photographer> Edit working calendar**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-22 Edit working calendar** | | |
| Created By: | Bo Cong Dat | Date Created: | 08/10/2020 |
| Primary Actor: | Photographer | Secondary Actors: |  |
| Trigger: | The photographer taps the date in the calendar. | | |
| Description: | The photographer can edit the working calendar. | | |
| Preconditions: | - The photographer has been authenticated.  - The photographer has access to the Internet. | | |
| Post-conditions: | Success: the system updates the calendar of photographers. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The photographer taps day. | The system shows details of day. Photographer can view:  - A list request of customer following start time | | 2 | The photographer taps day in the future and makes an empty request. |  | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | N/A | | |
| Priority: | Medium | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

**Table 36 - <Photographer>Edit Working Calendar Use Case Specification**

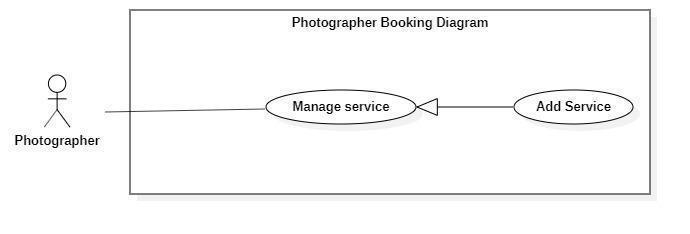
1. ***<Photographer> Get customer’s review***



**Figure 30 - <Photographer> Get customer's review**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-23 Get customer’s review** | | |
| Created By: | Bo Cong Dat | Date Created: | 08/10/2020 |
| Primary Actor: | Photographer | Secondary Actors: |  |
| Trigger: | The photographer taps “Trang cá nhân” page. | | |
| Description: | The photographer can view the customer's review. | | |
| Preconditions: | - The photographer has been authenticated.  - The photographer has access to the Internet. | | |
| Post-conditions: | Success: the system sends notifications to the customer. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The photographer taps “Trang cá nhân” page. | - The system shows the “Trang cá nhân” page of the photographer. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | N/A | | |
| Priority: | Medium | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: | BR-08 | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

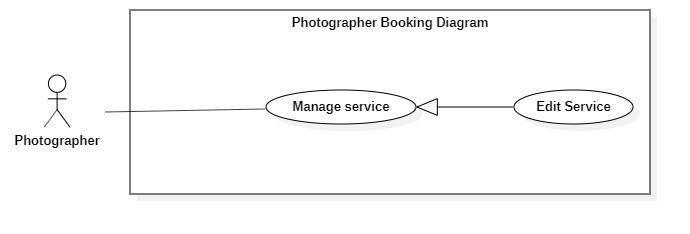
**Table 37 - <Photographer> Get Customer’s Review Use Case Specification**

1. ***<Photographer> Add service***

**Figure 31 - <Photographer> Add service**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-24 Add Service** | | |
| Created By: | Bo Cong Dat | Date Created: | 08/10/2020 |
| Primary Actor: | Photographer | Secondary Actors: |  |
| Trigger: | The photographer taps add icon on the “Trang cá nhân” screen. | | |
| Description: | The photographer can add service. | | |
| Preconditions: | - The photographer has been authenticated.  - The photographer has access to the Internet. | | |
| Post-conditions: | Success: the photographer adds service to the “Chọn gói dịch vụ bạn muốn” the detail screen. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The photographer taps navigates “Dịch vụ” screen and taps add icon. | The system shows list of services the photographer has added with the following information:  - “Kiểu dịch vụ”: choose a method, requid.  - “Tên dịch vụ”: input text, required.  - “Mô tả”: text, requid.  - “Giá thành dịch vụ”: input text, required  - “Thời gian tác nghiệp”: input text, required  - “Chi tiết dịch vụ”: input text, required.  - “Phương thức giao hàng”: choose a method.  - “Thêm dịch vụ”: button | | 2 | The photographer taps “Thêm dịch vụ” button | - The system validates inputted data [Exception 1].  - The system sends notifications to the customer. [Exception 2]. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | “Tên dịch vụ”, “Giá thành dịch vụ”, “Chi tiết dịch vụ” is empty | System shows an error message: “Thông tin này bắt buộc” | | 2 | Add service successfully | System shows an error message:  “Thêm dịch vụ thành công”. | | | |
| Priority: | High | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

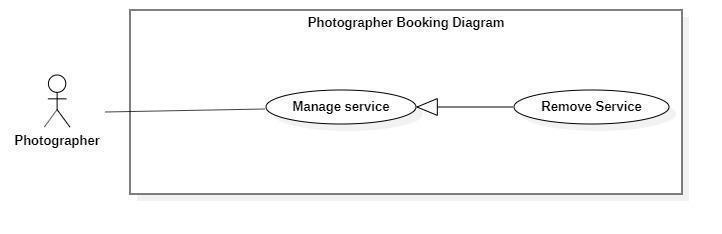
**Table 38 - <Photographer> Add Service Use Case Specification**

1. ***<Photographer> Edit service***

**Figure 32 - <Photographer> Edit Service**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-25 Edit service** | | |
| Created By: | Bo Cong Dat | Date Created: | 08/10/2020 |
| Primary Actor: | Photographer | Secondary Actors: |  |
| Trigger: | The photographer taps “Thay đổi” button on “Gói dịch vụ” detail page. | | |
| Description: | The photographer can edit the service available. | | |
| Preconditions: | - The photographer has been authenticated.  - The photographer has access to the Internet.  - The service pack available. | | |
| Post-conditions: | Success: the photographer edit service successfully and update “Các gói dịch vụ bạn muốn” to the detail photographer screen. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The photographer taps “Dịch vụ của tôi”. | The system shows list of services the photographer has view with the following information:  - “Kiểu dịch vụ”: choose a method, requid.  - “Tên dịch vụ”: input text, required.  - “Mô tả”: text, requid.  - “Giá thành dịch vụ”: input text, required  - “Thời gian tác nghiệp”: input text, required  - “Chi tiết dịch vụ”: input text, required.  - “Phương thức giao hàng”: choose a method.  - “Chỉnh sửa”: button  - “Xóa”: button | | 2 | The photographer taps “Chỉnh sửa” button | The system shows list of services the photographer input with the following information:  - “Kiểu dịch vụ”: choose a method, requid.  - “Tên dịch vụ”: input text, required.  - “Mô tả”: text, requid.  - “Giá thành dịch vụ”: input text, required  - “Thời gian tác nghiệp”: input text, required  - “Chi tiết dịch vụ”: input text, required.  - “Phương thức giao hàng”: choose a method.  - “Xác nhận”: button  - Hủy”: button | | 3 | The photographer taps “Xác nhận” button | - The system validates inputted data [Exception 1].  - The system sends nontifies to the customer. [Exception 2]. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 2 | “Tên dịch vụ”, “Giá thành dịch vụ”, “Chi tiết dịch vụ” is empty | System shows an error message: “Thông tin này bắt buộc” | | 3 | Edit service successfully | System shows an error message:  “Thay đổi thành công”. | | | |
| Priority: | Medium | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

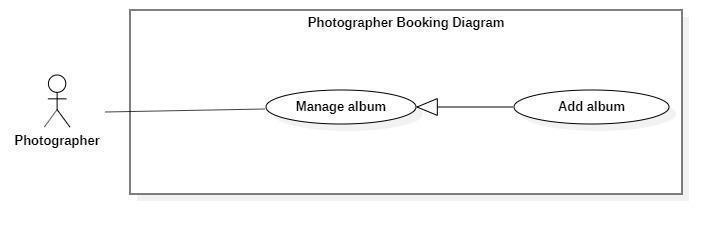
**Table 39 - <Photographer>Edit service use case specification**

1. ***<Photographer> Remove Service***

**Figure 33 - <Photographer> Remove Service**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-26 Remove service** | | |
| Created By: | Bo Cong Dat | Date Created: | 08/10/2020 |
| Primary Actor: | Photographer | Secondary Actors: |  |
| Trigger: | The photographer taps “Xóa” button in the “Dịch vụ của tôi” screen. | | |
| Description: | The photographer can remove service. | | |
| Preconditions: | - The photographer has been authenticated.  - The photographer has access to the Internet.  - The service pack available. | | |
| Post-conditions: | Success: the photographer removes service successfully and update “Các gói dịch vụ bạn muốn” on the detail screen. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The photographer taps navigate to “Dịch vụ”. | The system shows list of services the photographer has view with the following information:  - “Kiểu dịch vụ”: choose a method, requid.  - “Tên dịch vụ”: input text, required.  - “Mô tả”: text, requid.  - “Giá thành dịch vụ”: input text, required  - “Thời gian tác nghiệp”: input text, required  - “Chi tiết dịch vụ”: input text, required.  - “Phương thức giao hàng”: choose a method.  - “Xác nhận”: button  - Hủy”: button | | 2 | The photographer taps the “Xóa” button. | The system shows notifications to the photographer. [Exception 1] | | 3 | The photographer taps “Xác nhận” button | The system sends notifications to the customer. [Exception 2]. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | The photographer taps the “Xóa” button. | System shows an error message: “Bạn có muốn xóa dịch vụ này?” | | 2 | Remove service successfully | System shows an error message:  “Xóa dịch vụ thành công”. | | | |
| Priority: | Medium | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

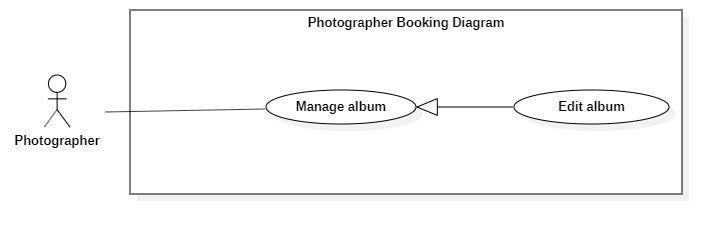
**Table 40 - <Photographer>Remove service use case specification**

1. ***<Photographer>Add album***

**Figure 34 - <Photographer> Add Album**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-27 Add Album** | | |
| Created By: | Bo Cong Dat | Date Created: | 08/10/2020 |
| Primary Actor: | Photographer | Secondary Actors: |  |
| Trigger: | The photographer taps the add album icon on the “Album” screen. | | |
| Description: | The photographer can add albums. | | |
| Preconditions: | - The photographer has been authenticated.  - The photographer has access to the Internet. | | |
| Post-conditions: | Success: the photographer adds the album successfully and update “Các album của photographer” on the detail screen. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The photographer taps the “Album” screen and taps the added album icon. | The system shows list of albums available the photographer has added photo with the following information:  - “Ngày tạo”: choose date, requid.  - “Mô tả”: text, requid.  - “Tên album”: input text, required.  - “Hình ảnh”: choose the device.  - “Category”: choose method, required  - “Địa điểm”: input text, required.  - “Thêm album”: button | | 2 | The photographer taps “Thêm album” button | - The system validates inputted data [Exception 1].  - The system sends notifications to the customer. [Exception 2]. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | “Tên album” and “Hình ảnh”, “thể loại” is empty | System shows an error message: “Thông tin này bắt buộc” | | 2 | Add album successfully | System shows an error message:  “Thêm album thành công”. | | | |
| Priority: | Medium | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: | . | | |

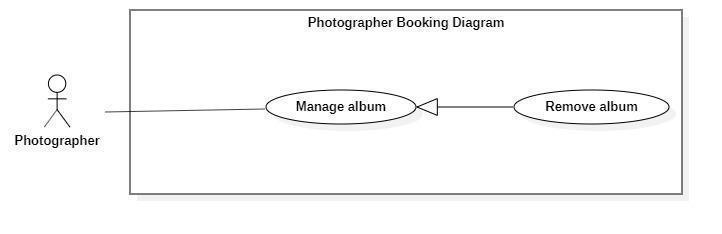
**Table 41 - <Photographer> Add album use case specification**

1. ***<Photographer> Edit album***

**Figure 35 - <Photographer> Edit Album**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-28 Edit Album** | | |
| Created By: | Bo Cong Dat | Date Created: | 08/10/2020 |
| Primary Actor: | Photographer | Secondary Actors: |  |
| Trigger: | The photographer taps “Album” detail and taps “Chỉnh sửa” button. | | |
| Description: | The photographer can edit the album of them. | | |
| Preconditions: | - The photographer has been authenticated.  - The photographer has access to the Internet.  - The album available. | | |
| Post-conditions: | Success: the photographer edited the album successfully and update “Các album của photographer” on the detail screen. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The photographer taps “Album”. | The system shows a list of photos available. The photographer has view photo with the following information:  - “Ngày tạo”: choose date, requid.  - “Mô tả”: text, requid.  - “Tên album”: input text, required.  - “Hình ảnh”: choose the device.  - “Category”: choose method, required  - “Địa điểm”: input text, required.  - “Xóa”: button  - “Chỉnh sửa”: button | | 2 | The photographer taps “Chỉnh sửa” button. | The system shows a detailed photo screen. The photographer has edit album with the following information:  - “Ngày tạo”: choose date, requid.  - “Mô tả”: text, requid.  - “Tên album”: input text, required.  - “Hình ảnh”: choose the device.  - “Category”: choose method, required  - “Địa điểm”: input text, required.  - “Xác nhận”: button  -”Hủy”: button | | 3 | The photographer taps “Xác nhận” button | - The system validates inputted data [Exception 1].  - The system sends notifications to the customer. [Exception 2]. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | “Tên album” and “Hình ảnh”, “thể loại” is empty | System shows an error message: “Thông tin này bắt buộc” | | 2 | The photographer taps the “Xác nhận” button. | System shows an error message:  “Thay đổi album thành công”. | | | |
| Priority: | Medium | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

**Table 42 - <Photographer> Edit Album Use Case Specification**

1. ***<Photographer> Remove album***

**Figure 36 - <Photographer> Remove Album**

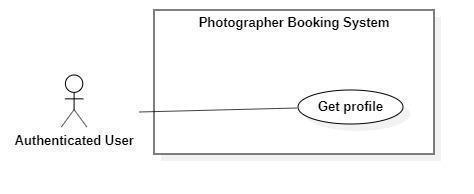
|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-29 Remove Album** | | |
| Created By: | Bo Cong Dat | Date Created: | 08/10/2020 |
| Primary Actor: | Photographer | Secondary Actors: |  |
| Trigger: | The photographer taps the “Xóa” button. | | |
| Description: | The photographer can remove the album. | | |
| Preconditions: | - The photographer has been authenticated.  - The photographer has access to the Internet.  - The album available. | | |
| Post-conditions: | Success: the photographer removed the album successfully and update “Các album của photographer” on the detail screen. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The photographer taps the “Xóa” button. | The system shows notifies to the photographer. [Exception 1] | | 2 | The photographer taps the “Xác nhận” button. | The system shows notifications to the photographer. [Exception 2]. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | The photographer taps the “Xóa” button. | System shows an error message: “Bạn có muốn xóa album này?” | | 2 | The photographer taps the “Xác nhận” button. | System shows an error message:  “Xóa album thành công”. | | | |
| Priority: | Medium | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

**Table 43 - Photographer> Remove Album Use Case Specification**

#### d. <Authenticated User> Overview Use Case

**Figure 37 - <Authenticated User> Overview Use Case**

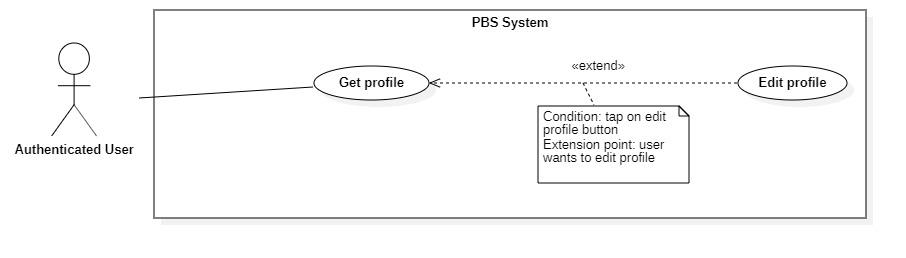
1. ***<Authenticated User> Get profile***



**Figure 38 - <Authenticated User> Get profile**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-30 Get profile** | | |
| Created By: | Tran Thien Thao | Date Created: | 08/10/2020 |
| Primary Actor: | Authenticated User | Secondary Actors: |  |
| Trigger: | The Authenticated User navigates to “Cài đặt” and taps the “Thông tin cá nhân” screen to view the profile. | | |
| Description: | The authenticated user can view their profile. | | |
| Preconditions: | The authenticated user has accessed the Internet. | | |
| Post-conditions: | Success: The Authenticated User can view his/her profile successfully. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The Authenticated User navigates to the “Cài đặt” and taps the “Thông tin cá nhân” screen to view the profile. | The system shows “Thông tin cá nhân” screen with the following information:  - “Avatar”: image.  - “Họ và tên”: text input, editable  - “Địa chỉ”: text input, editable  - “Số điện thoại”: phone number input  - “Email”: text input, required.  - “Cập nhật” button | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | N/A | | |
| Priority: | Medium | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: | N/A | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

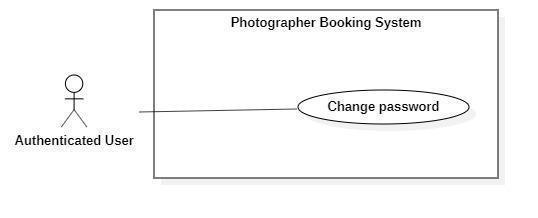
**Table 44 - <Authenticated User> Get profile use case specification**

1. ***<Authenticated User> Edit profile***

**Figure 39 - <Authenticated User> Edit profile**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-31 Edit profile** | | |
| Created By: | Tran Thien Thao | Date Created: | 08/10/2020 |
| Primary Actor: | Authenticated User | Secondary Actors: |  |
| Trigger: | The Authenticated User taps “Cập nhật” of “Thông tin cá nhân” screen to view profile. | | |
| Description: | The authenticated user can update their profile. | | |
| Preconditions: | The authenticated user has accessed the Internet. | | |
| Post-conditions: | Success: The Authenticated User can update his/her profile successfully. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The Authenticated User taps the “Thông tin cá nhân” screen to view the profile. | The system shows “Thông tin cá nhân” screen and view with the following information:  - “Avatar”: image  - “Họ và tên”: text input, editable  - “Địa chỉ: text input, editable  - “Số điện thoại”: phone number input.  - “Email”: text input, required  - “Cập nhật” button: is used to update information. | | 2 | The Authenticated User updates information and clicks “Cập nhật” to confirm | The system shows “Thông tin cá nhân” screen and view with the following information:  - “Avatar”: image  - “Họ và tên”: text input, editable  - “Địa chỉ: text input, editable  - “Số điện thoại”: phone number input.  - “Xác nhận” button  - “Hủy”: button | | 3 | The Authenticated User updates information and clicks “Xác nhận” to confirm | The system validates inputted data  The system updates the profile and shows message  [Exception 1, 2] | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | “Họ và tên”, “địa chỉ”, “Số điện thoại” is empty | System shows an error message: “Thông tin này bắt buộc”. | | 2 | Update successfully | System shows an error message: “Cập nhật thông tin thành công”. | | | |
| Priority: | Medium | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

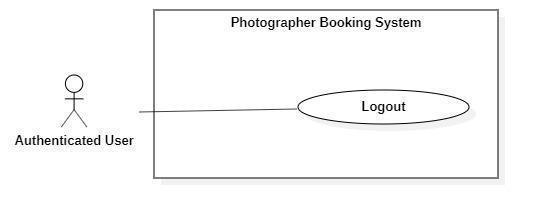
**Table 45 - <Authenticated User> Edit profile use case specification**

1. ***<Authenticated User> Change password***

**Figure 40 - <Authenticated User> Change password**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-32 Change password** | | |
| Created By: | Tran Thien Thao | Date Created: | 08/10/2020 |
| Primary Actor: | Authenticated User | Secondary Actors: |  |
| Trigger: | The Authenticated User navigates to “Cài đặt” screen and taps the “Thay đổi mật khẩu” to change password. | | |
| Description: | The authenticated user can change their password. | | |
| Preconditions: | The authenticated user has accessed the Internet. | | |
| Post-conditions: | Success: The Authenticated User can change password successfully and logout the system. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The Authenticated User navigates to “Cài đặt” screen and taps the “Thay đổi mật khẩu” | The system shows “Đổi mật khẩu” screen with the following inputs:  - “Mật khẩu hiện tại”: current password, password input, required  - “Mật khẩu mới”: new password, password input, required  - “Xác nhận mật khẩu mới”: confirm new password, password input, required  - “xác nhận” button.  - “Hủy”: button | | 2 | The Authenticated User inputs information and taps “Xác nhận” to confirm. | The system validates inputted data  [Exception 1, 2]  The system updates the profile and shows message  [Exception 3] | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | “Mật khẩu hiện tại”, “Mật khẩu mới”, “xác nhận mật khẩu mới” is empty | System shows an error message: “Thông tin này bắt buộc”. | | 2 | New password and confirm password do not match. | System shows an error message: “Mật khẩu nhập lại chưa chính xác” | | 3 | Current password is invalid | System shows an error message: “Mật khẩu hiện tại chưa chính xác” | | | |
| Priority: | Medium | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: |  | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

**Table 46 - <Authenticated User> Change Password Use Case specification**

1. ***<Authenticated User> Logout***

**Figure 41 - <Authenticated User> Logout**

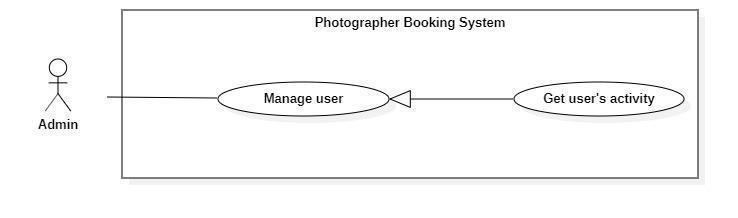
|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-33 Logout** | | |
| Created By: | Tran Thien Thao | Date Created: | 08/10/2020 |
| Primary Actor: | Authenticated User | Secondary Actors: |  |
| Trigger: | The Authenticated User navigates to “Cài đặt” screen and taps the “Đăng xuất” to log out. | | |
| Description: | The system allows the Authenticated User to log out | | |
| Preconditions: | The authenticated user has accessed the Internet. | | |
| Post-conditions: | Success: The Authenticated User can log out successfully. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The Authenticated User navigates to “Cài đặt” screen to log out | The system deletes the login session and navigates the back to “Đăng nhập” screen. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | N/A | | |
| Priority: | Medium | | |
| Frequency of Use: | Always | | |
| Business Rules: | N/A | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

**Table 47 - <Authenticated User> Logout use case specification**

#### e. <Admin> Overview Use Case

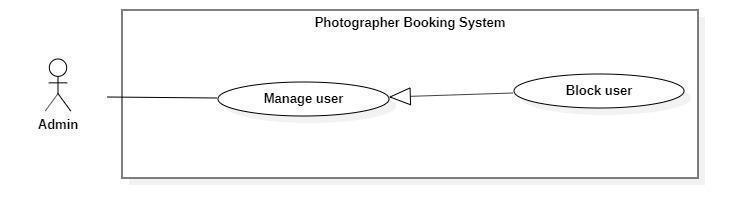
**Figure 42 - <Admin> Overview Use Case**

1. ***<Admin> Get user’s activity***

**Figure 43 - <Admin> Get user's activity**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-34 Get user’s activity** | | |
| Created By: | Dao Sy Trung Kien | Date Created: | 08/10/2020 |
| Primary Actor: | Admin | Secondary Actors: |  |
| Trigger: | The admin taps navigate to “Hoạt động” screen. | | |
| Description: | The admin can view a list of the user's activity. | | |
| Preconditions: | - User must be an Admin.  - User has logged into the website. | | |
| Post-conditions: | Success: The admin can view a table and a list of the user's activity successfully. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The admin taps navigate to “Hoạt động” screen. | The system showed a table a list of the user's activity. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | N/A | | |
| Priority: | Medium | | |
| Frequency of Use: | Always | | |
| Business Rules: | BR-10, BR-11 | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

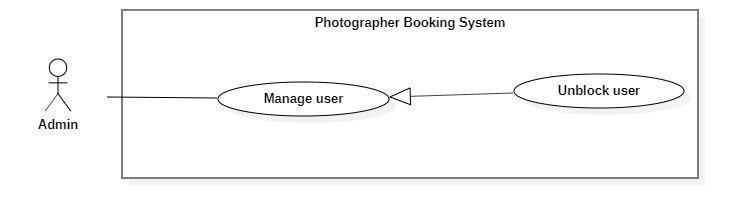
**Table 48 - <Admin> Get user's activity use case specification**

1. ***<Admin> Block user***

**Figure 44 - <Admin> Block user**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-35 Block user** | | |
| Created By: | Dao Sy Trung Kien | Date Created: | 08/10/2020 |
| Primary Actor: | Admin | Secondary Actors: |  |
| Trigger: | The admin tabs “Khóa” button on the “Danh sách người dùng” screen. | | |
| Description: | The admin can block users if the user violates business rules. | | |
| Preconditions: | - User must be an Admin.  - User has logged into the website. | | |
| Post-conditions: | Success: The admin can block users successfully. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The admin tabs “Khóa” button on the “Danh sách người dùng” screen. | The system shows notifications to the admin. [Exception 1]. | | 2 | The admin taps the “Xác nhận” button. | The system shows notifications to the admin. [Exception 2] | | 3 |  | The system sends notifications to the user. [Exception 3]. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | The system shows notifications to the admin. | System shows an error message: “Bạn có muốn khóa người dùng này?”. | | 2 | The admin taps the “Xác nhận” button. | System shows an error message: “Người dùng đã được khóa thành công”. | | 3 | The system sends notifications to the user. | System shows an error message: “Tài khoản của bạn đã bị vi phạm. Vui lòng liên hệ với quản trị viên”. | | | |
| Priority: | High | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: | BR-09 | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

**Table 49 - <Admin> Block user use case specification**

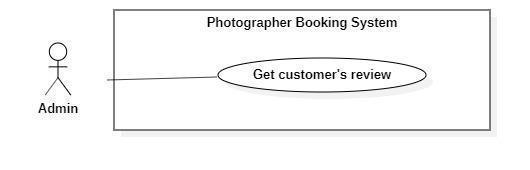
1. ***<Admin> Unblock user***

**Figure 45 - <Admin> Unblock user**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-36 Unblock user** | | |
| Created By: | Dao Sy Trung Kien | Date Created: | 08/10/2020 |
| Primary Actor: | Admin | Secondary Actors: |  |
| Trigger: | The admin tabs “Mở khóa” button on the “Danh sách người dùng” screen. | | |
| Description: | The admin can unblock users. | | |
| Preconditions: | - User must be an Admin.  - User has logged into the website.  - The user locked. | | |
| Post-conditions: | Success: The admin can unblock users successfully. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The admin tabs “Mở khóa” button on the “Danh sách người dùng” screen. | The system shows notifications to the admin. [Exception 1] | | 2 | The admin taps the “Xác nhận” button. | The system shows notifications to the admin. [Exception 3] | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | The system shows notifications to the admin. | System shows an error message: “Bạn có muốn mở khóa người dùng này?”. | | 2 | The admin taps the “Xác nhận” button. | System shows an error message: “Người dùng đã được mở khóa thành công”. | | | |
| Priority: | High | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: | BR-09 | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

**Table 50 - <Admin> Unblock user use case specification**

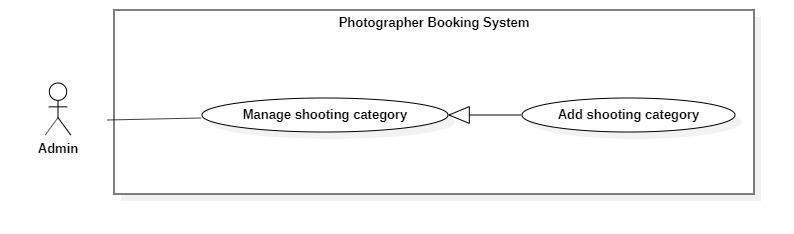
1. ***<Admin> Get customer’s review***

******

**Figure 46 - <Admin> Get customer's review**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-37 Get customer’s review** | | |
| Created By: | Dao Sy Trung Kien | Date Created: | 08/10/2020 |
| Primary Actor: | Admin | Secondary Actors: |  |
| Trigger: | The admin taps the “Trang cá nhân” page of the photographer. | | |
| Description: | The admin can view customer’s review. | | |
| Preconditions: | - User must be an Admin.  - User has logged into the website. | | |
| Post-conditions: | Success: the admin views the customer’s review. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The admin taps “Trang cá nhân” page. | - The system shows the “Trang cá nhân” page of the photographer. [Exception 1] | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | N/A | | |
| Priority: | Medium | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: | BR-23, BR-24 | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

**Table 51 - <Admin> Get customer's review use case specification**

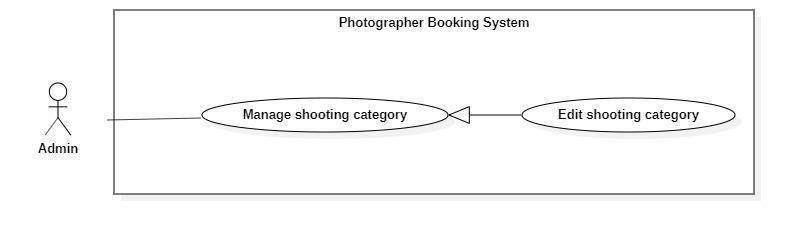
1. ***<Admin> Add shooting category***

**Figure 47 - <Admin> Add shooting category**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-38 Add shooting category** | | |
| Created By: | Dao Sy Trung Kien | Date Created: | 08/10/2020 |
| Primary Actor: | Admin | Secondary Actors: |  |
| Trigger: | The admin taps “Thêm thể loại” button. | | |
| Description: | The admin can add a shooting category. | | |
| Preconditions: | - User must be an Admin.  - User has logged into the website. | | |
| Post-conditions: | Success: The admin can add shooting successfully and update “Trang chủ” of customer’s application. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The admin taps “Thêm thể loại” button. | The system shown “Thêm thể loại chụp ảnh” page. The admin input following:  - “Tên thể loại”: input text, required.  - “Xác nhận”: button.  - “Hủy”: button | | 2 | The photographer taps the “Xác nhận” button. | - The system validates inputted data  [Exception 1, 2]  - The system shows notifications to the photographer. [Exception 3]. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | “Tên thể loại” is empty | System shows an error message: “Thông tin này bắt buộc” | | 2 | “Tên thể loại” is duplicate | System shows an error message: “Thể loại này đã tồn tại” | | 3 | The photographer taps the “Xác nhận” button. | System shows an error message:  “Thêm thể loại thành công”. | | | |
| Priority: | Medium | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: | BR-12 | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

**Table 52 - <Admin> Add shooting category use case specification**

1. ***<Admin> Edit shooting category***

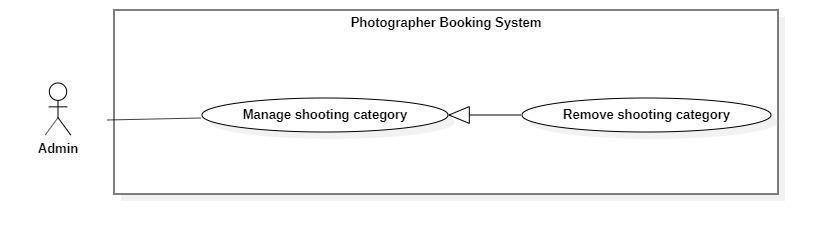


**Figure 48 - <Admin> Edit shooting category**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-39 Edit shooting category** | | |
| Created By: | Dao Sy Trung Kien | Date Created: | 08/10/2020 |
| Primary Actor: | Admin | Secondary Actors: |  |
| Trigger: | The admin tabs “Chỉnh sửa” button on the “Thể loại” detail screen. | | |
| Description: | The admin can edit the shooting category. | | |
| Preconditions: | - User must be an Admin.  - User has logged into the website.  - The category available. | | |
| Post-conditions: | Success: The admin can edit shooting successfully and update “Trang chủ” of customer’s application. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The admin taps “Thể loại” to the view detail. | The system showed a list of category pages.  - “Tên thể loại”: text  - “Xóa”: button  - “Chỉnh sửa”: button. | | 2 | The photographer taps the “Cập nhật” button. | The system shown “Thể loại” detail screen. The admin input following:  - “Tên thể loại”: input text, required.  - “Xác nhận”: button. | | 3 | The photographer taps the “Xác nhận” button. | - The system validates inputted data [Exception 1, 2]  - The system shows notifications to the photographer. [Exception 3]. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | “Tên thể loại” is empty | System shows an error message: “Thông tin này bắt buộc” | | 2 | “Tên thể loại” is duplicate | System shows an error message: “Thể loại này đã tồn tại” | | 3 | The photographer taps the “Xác nhận” button. | System shows an error message:  “Thay đổi thể loại thành công”. | | | |
| Priority: | Medium | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: | BR-13 | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

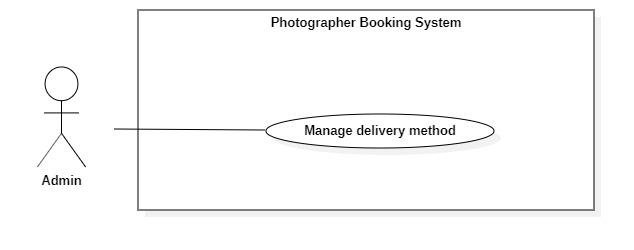
**Table 53 - <Admin> Edit shooting category use case specification**

1. ***<Admin> Remove shooting category***

**Figure 49 - <Admin> Remove shooting category**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-40 Edit shooting category** | | |
| Created By: | Dao Sy Trung Kien | Date Created: | 08/10/2020 |
| Primary Actor: | Admin | Secondary Actors: |  |
| Trigger: | The admin tabs “Xóa” button on the “Danh sách thể loại” screen. | | |
| Description: | The admin can remove the shooting category. | | |
| Preconditions: | - User must be an Admin.  - User has logged into the website.  - The category available. | | |
| Post-conditions: | Success: The admin can remove shooting successfully and update “Trang chủ” of customer’s application. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The admin taps “Thể loại” to the view detail. | The system showed a list of category pages.  - “Tên thể loại”: text  - “Xóa”: button  - “Chỉnh sửa”: button. | | 2 | The admin taps “Xóa” button | The system shows notifications to the photographer. [Exception 1] | | 3 | The photographer taps the “Xác nhận” button. | - The system shows notifications to the photographer. [Exception 2]. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | The admin taps remove the category icon. | System shows an error message: “Bạn có muốn xóa thể loại này ra khỏi thư mục?” | | 2 | The photographer taps the “Xác nhận” button. | System shows an error message:  “Xóa thể loại thành công”. | | | |
| Priority: | Medium | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: | BR-14 | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

**Table 54 - <Admin> Remove shooting category use case specification**

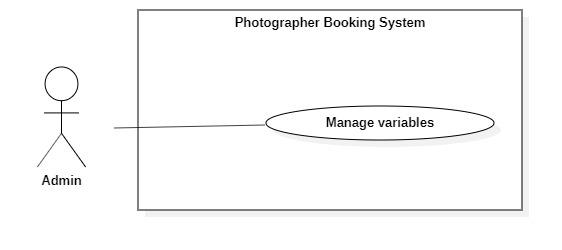
1. ***<Admin> Manage delivery method***

**Figure 50 - <Admin> Overview manage delivery method**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-41 Manage delivery method** | | |
| Created By: | Dao Sy Trung Kien | Date Created: | 08/10/2020 |
| Primary Actor: | Admin | Secondary Actors: |  |
| Trigger: | The admin taps “Thêm phương thức giao hàng” button. | | |
| Description: | The admin can add a shooting category. | | |
| Preconditions: | - User must be an Admin.  - User has logged into the website. | | |
| Post-conditions: | Success: The admin can add delivery method successfully and update “Phương thức giao hàng” of photographer’s application. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The admin taps “Thêm thể loại” button. | The system shown “Phương thức giao hàng” page. The admin input following:  - “Phương thức giao hàng”: input text, required.  - “Xác nhận”: button.  - “Hủy”: button | | 2 | The photographer taps the “Xác nhận” button. | - The system validates inputted data  [Exception 1, 2]  - The system shows notifications to the photographer. [Exception 3]. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | **No** | **Actor Action** | **System Response** | | 1 | “Phương thức giao hàng” is empty | System shows an error message: “Thông tin này bắt buộc” | | 2 | “Phương thức giao hàng” is duplicate | System shows an error message: “Thể loại này đã tồn tại” | | 3 | The photographer taps the “Xác nhận” button. | System shows an error message:  “Thêm thể loại thành công”. | | | |
| Priority: | Medium | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: | N/A | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

**Table 55 - <Admin> Manage delivery method use case specification**

1. ***<Admin> Manage variables***



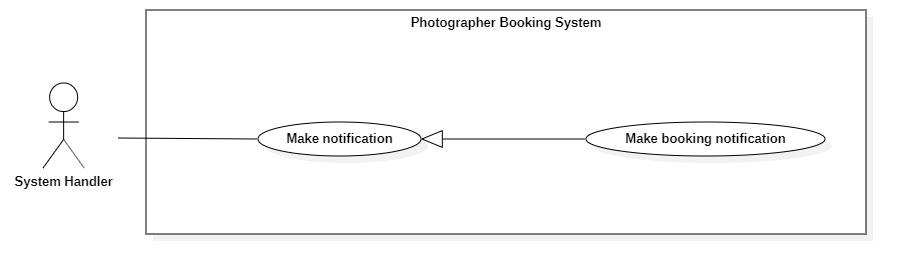
**Figure 51 - <Admin> Overview manage variables**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-42 Manage variables** | | |
| Created By: | Dao Sy Trung Kien | Date Created: | 08/10/2020 |
| Primary Actor: | Admin | Secondary Actors: |  |
| Trigger: | The admin taps “Variables” button. | | |
| Description: | The admin can manage variables. | | |
| Preconditions: | - User must be an Admin.  - User has logged into the website. | | |
| Post-conditions: | Success: The admin can manage variables. | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | The admin taps “Variables” button. | The system shown “Variables” page. | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | N/A | | |
| Priority: | Low | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: | N/A | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

**Table 56 - <Admin> Manage variables use case specification**

#### f. <System handler> Overview use case

1. **<System handler> Make booking notification**

**Figure 52 - <System handler> Make booking notification**

|  |  |  |  |
| --- | --- | --- | --- |
| UC ID and Name: | **PBS-UC-43 Make booking notification** | | |
| Created By: | Dao Sy Trung Kien | Date Created: | 08/10/2020 |
| Primary Actor: | System handler | Secondary Actors: |  |
| Trigger: | The customer send request to the photograper.  The photographers respond to requests for customers. | | |
| Description: | The customer receives notices of his request sent to the photographer and also is the photographer. | | |
| Preconditions: | - The customer send request successfully. | | |
| Post-conditions: | Success: The system make booking notification and sends to the customer | | |
| Normal Flow: | |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | System handler retrieves make booking notification. | The system stores request and makes notification information to return to systems handler. | | 2 | System handler retrieves notification information and sends to customers. |  | | | |
| Alternative Flows: | N/A | | |
| Exceptions: | |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Make notification process encountered an internal error. | Error detail will be tracked in log file. System outputs error message and sends to user. | | | |
| Priority: | Low | | |
| Frequency of Use: | Sometimes | | |
| Business Rules: | N/A | | |
| Other Information: | N/A | | |
| Assumptions: |  | | |

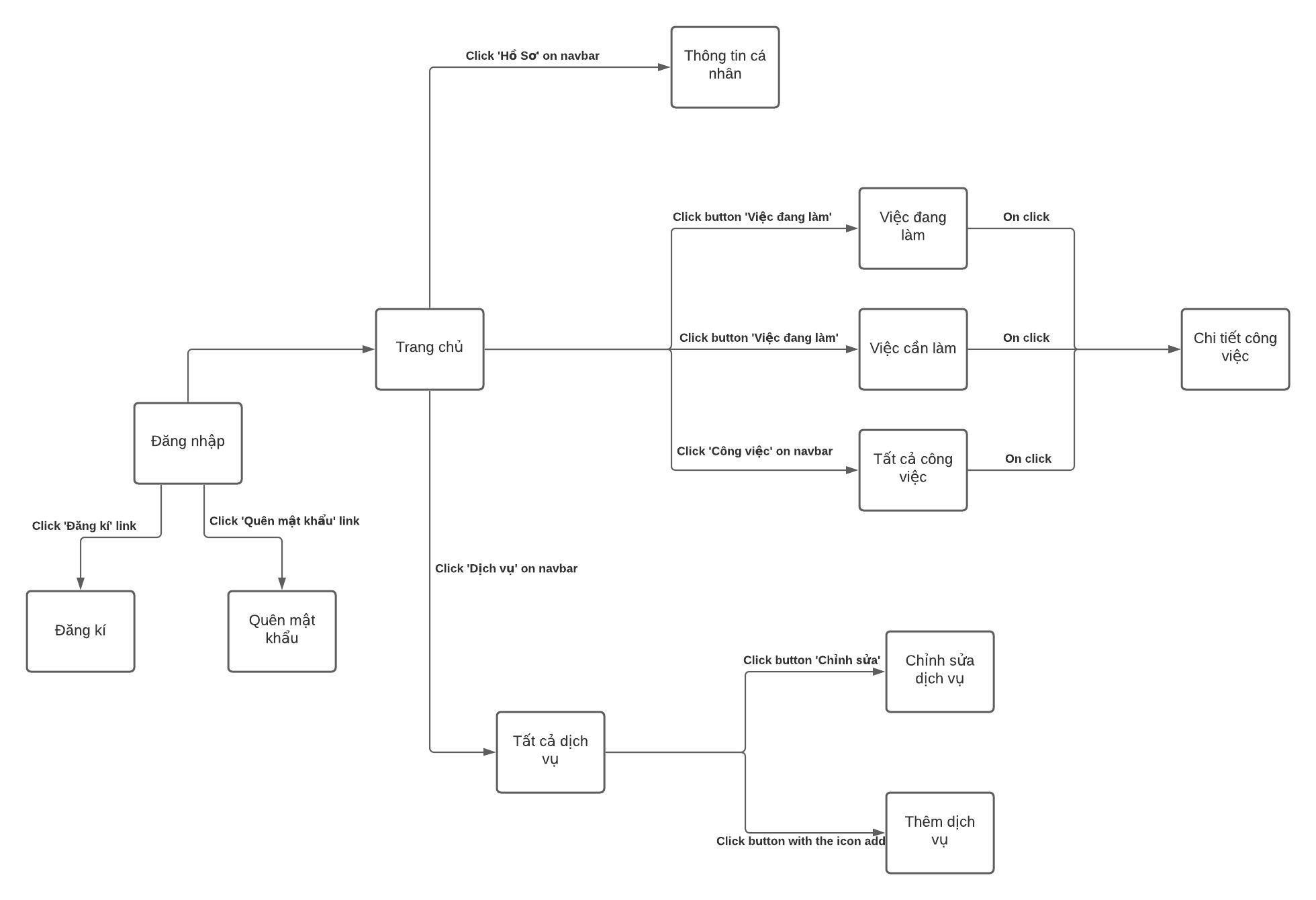
**Table 57 - <System handler> Make booking notification use case specification**

## 3. Functional Requirements

### 3.1 System Functional Overview

#### a. Screen Flow

1. **Customer’s screen flow**
2. **Photographer’s screen flow**

**Figure 53 - Customer's Screen Flow**

#### b. Screen Details

1. ***Customer’s screen Details***

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Feature** | **Screen** | **Description** |
| 1 | Register | Register | Register screen included: avatar, username, fullname, password, re-password, phone number, address, email. |
| 2 | Login | Login | Login screen included: username, password |
| 3 | Login | Forgot password | Forgot password screen included: Old password, new password, re-password. |
| 4 | View application | Main screen | Main screen included: Search icon, category, photographer is appreciated, Album is appreciated… |
| 5 | View profile | Profile | Profile screen included: fullname, username, phone number, address, email. |
| 6 | Update profile | Edit profile | Edit profile screen included: fullname, phone number, address, email. |
| 7 | View request | Request | Request screen included: Status, name photographer, schedule/time, package service, address. |
| 8 | View request | Request detail | Detail request screen included: Status, name photographer, schedule/time, package service, address. |
| 9 | Update request | Change schedule/time | Change booking time screen included: calendar |
| 10 | Update request | Change delivery schedule/time | Change delivery time screen included: calendar |
| 11 | Update request | Change address | Change booking address screen included: map |
| 12 | View album | Album detail | Detail album screen include: like, photographer, description |
| 13 | View album | Like album | “Like” screen include: number of hearts |
| 14 | View profile photographer | Photographer’s profile | Profile photographer screen included: name, review, description, album, package service, schedule. |
| 15 | Book photographer | Create request | Create request screens include: photographer, schedule/time, delivery schedule, address, package service, price. |
| 16 | Book photographer | Create schedule/time | Create booking time screen included: calendar |
| 17 | Book photographer | Create delivery schedule/time | Create delivery time screen included: calendar |
| 18 | Book photographer | Create address | Create booking address screen included: map |

**Table 58 - Customer’s screen Details**

1. ***Photographer’s screen Details***

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Feature** | **Screen** | **Description** |
| 1 | Register | Register | Register screen included: avatar, username, fullname, password, re-password, phone number, address, email. |
| 2 | Login | Login | Login screen included: username, password |
| 3 | Login | Forgot password | Forgot password screen included: Old password, new password, re-password. |
| 4 | View application | Main screen | Main screen included: Search icon, category, photographer is appreciated, Album is appreciated… |
| 5 | View profile | Profile | Profile screen included: fullname, username, phone number, address, email. |
| 6 | Update profile | Edit profile | Edit profile screen included: fullname, phone number, address, email. |
| 7 | View request | Request | Request screen included: Status, name customer, schedule/time, package service, address |
| 8 | View request | Request detail | Detail request screen included: Status, name customer, schedule/time, package service, address. |
| 9 | View request | “Việc đang làm” | “Việc đang làm” screen included: Customer requests are awaiting confirmation. |
| 10 | View request | “Việc cần làm” | “Việc cần làm” screen included: calendar and customer request has been confirmed. |
| 11 | View request | “Tất cả công việc” | “Tất cả công việc” screen included: all status of the customer requests. |
| 12 | Add service | Service | Service screen included: all package service and add service button. |
| 13 | Edit service | Service detail | Service detail included: Name, service, price. |
| 14 | Remove service | Service detail | Service detail included: Name, service, price. |

**Table 59 - Photographer’s screen Details**

#### c. Screen Authorization

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Screen** | **Guest** | **Customer** | **Photographer** | **Admin** | **Authenticated User** |
| Register | X |  |  |  |  |
| Photographer’s login |  |  | X |  | X |
| Customer’s login |  | X |  |  |  |
| Forgot password |  | X | X | X |  |
| Photographer’s home screen |  |  | X |  |  |
| Customer’s home screen |  | X |  |  |  |
| Search photographer |  | X |  |  |  |
| Category |  |  |  | X |  |
| Photographer is appreciated |  | X |  |  |  |
| Album is appreciated |  | X |  |  |  |
| Photographer’s detail |  | X |  | X |  |
| Package service |  | X |  |  |  |
| Book photographer |  | X |  |  |  |
| Calendar |  | X |  |  |  |
| View review |  | X | X | X |  |
| Booking detail |  | X |  |  |  |
| Choose start time |  | X |  |  |  |
| Choose delivery time |  | X |  |  |  |
| Choose Address |  | X |  |  |  |
| Activity screen |  | X |  |  |  |
| Pending |  |  | X |  |  |
| Upcoming |  |  | X |  |  |
| Post production |  |  | X |  |  |
| Done |  |  | X |  |  |
| Review |  | X |  |  |  |
| Cancel |  | X | X |  |  |
| Setting screen |  | X | X |  |  |
| Profile |  | X | X |  |  |
| Edit profile |  | X | X |  |  |
| Change password |  | X | X |  |  |
| Log out |  |  |  |  | X |
| Service |  |  | X |  |  |
| Service detail |  |  | X |  |  |
| Edit service |  |  | X |  |  |
| Remove Service |  |  | X |  |  |
| Add service |  |  | X |  |  |
| Album |  | X | X |  |  |
| Add album |  |  | X |  |  |
| Edit album |  |  | X |  |  |
| Remove album |  |  | X |  |  |
| Forum |  | X | X | X |  |
| Open thread |  | X | X | X |  |
| Edit thread |  | X | X | X |  |
| Comment thread |  | X | X | X |  |
| Follow thread |  | X | X | X |  |
| Notification |  | X | X |  |  |

**Table 60 - Screen Authorization**

In which:

* Guest: A person who cannot use the system’s services and functionalities.
* Customer: A person who can make a request to a photographer.
* Photographer: A person who can update a booking request’s status.
* Admin: A person who manages the whole system.
* Authenticated user: A person who has logged in to the system successfully and can update their profile.

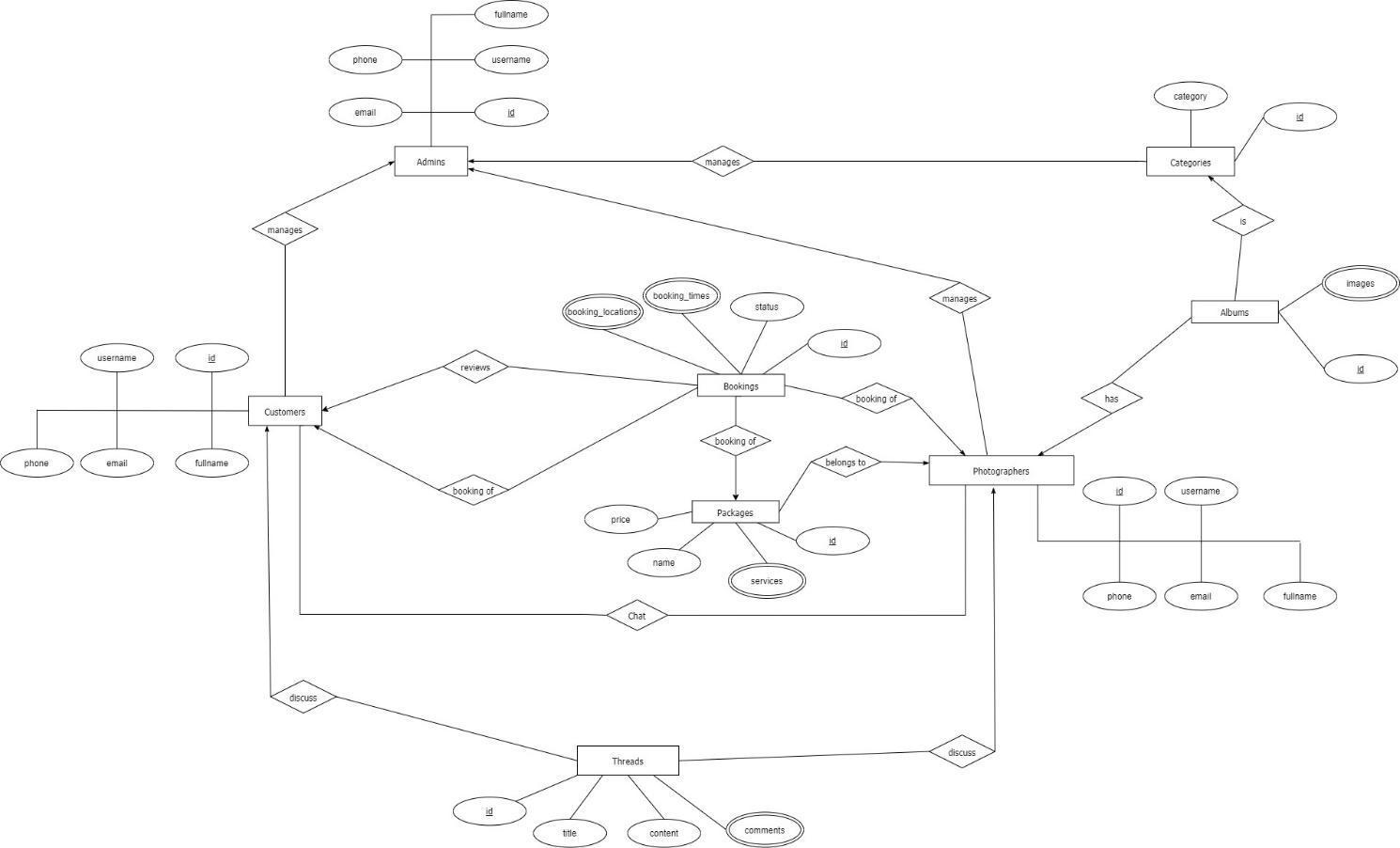
#### d. Non-Screen Functions

*[Provide the descriptions for the non-screen system functions, i.e batch/cron job, service, API, etc.]*

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Feature** | **System Function** | **Description** |
| 1 | <<Feature Name>> | <<Function Name1>> | <<Function Name1 Description>> |
| 2 | … |  |  |

**Table 61 - Non-Screen Functions**

#### e. Entity Relationship Diagram



**Figure 54 - Enity Relationship Diagram**

**Entities List**

|  |  |  |
| --- | --- | --- |
| **#** | **Entity** | **Description** |
| 1 | Customer | Customer is an entity representing the person who purchases bookings from photographers: id, username, fullname, phone, email. |
| 2 | Admins | Admin is an entity represents the person who works for pbs system and manages activities of the system: id, username, fullname, phone, email |
| 3 | Booking | Booking is an entity represents the photography deal between photographer and customer: id, status, booking\_times, booking\_locations |
| 4 | Photographer | Photographer is an entity that represents the person who provides the photography services and takes customers’ deals. |
| 5 | Category | Category is an entity represents the type of albums (is it wedding or look book or anything else): id, category |
| 6 | Album | Album is an entity represents the images which photographer wants to show the customers to attract them: id, images |
| 7 | Package | Package is an entity represents the photography services which photographer provided: id, name, price, services |
| 8 | Thread | Thread is an entity represents a post of photographer or customer or admin on forum: id, title, content, comments |

**Table 62 - Entities List**

## 4. Non-Functional Requirements

### 4.1 External Interfaces

#### a. User Interfaces

For web application:

UI-1: GUI has to be simple and easy to use.

UI-2: The language used in the web application is Vietnamese.

UI-3: The web application has to be responsive on multiple devices: laptop, tablet and mobile.

For mobile application:

UI-1: User interface is pleasant to look at and easy to use.

UI-2: The language is Vietnamese.

UI-3: The user interface can be responsive on multiple screen sizes.

UI-4: The functionalities are arranged clearly.

#### b. Software Interfaces

This system will have required software MySQL to program and create and manage the database for the system. Use for source control is Git and GitHub. Test browsers use Chrome 96 or above. Specification for developing the web application is Java EE 11.0.8. Program tools are Visual Studio Code, Android Studio, IntelliJ.

The system will require software window 8 and window 10 to program and manage photographer booking systems. Booking process of the system that organizes the distribution of users in the mobile app and able to do this manage review.

#### c. Hardware Interfaces

Just like any other system, this system requires a basic computer that consists of CPU, monitor, keyboard and mouse or laptop and smartphone for input and output. Smartphones could be used to do booking and use the system. Besides, for direction the system connected with a GPS device in every smartphone to get the location detail.

#### d. Communications Interfaces

The system communication interface is completely dependent on correctly sending and retrieving data from the database. Other than that is an online protocol to connect between GPS and the mobile.

### 4.2 Quality Attributes

#### a. Usability

* The system does not require its users to have any technical knowledge before using its functionalities.
* Layout and navigation are clear and easy to use.
* Customers can use PSB easily even if they have never used it before. They can find a photographer and make a request within 5 minutes.
* Photographers can use PSB easily even if they have never used it before. They can view and confirm requests received from the customers within 5 minutes. They can also use other functionalities quickly and effectively with easy-to-understand instructions and simple operation.

#### b. Reliability

* The availability of the system is 80%. The system can be used within 60 minutes and will automatically log the user out if there is no operation within 5 minutes.

#### c. Performance

All requests will be processed within 10 seconds on an Internet connection above 25Mbps.

#### d. Dependability

##### d1. Security

* All user’s passwords will be encrypted before being saved into the database.
* Accounts will be verified by email.
* Username must be unique.
* Photographers’ email and phone number will be secured.

##### d2. Safety

* A service price a photographer has posted cannot be changed when a customer chooses that service. If there is a surcharge, it must be negotiated with the customer.
* When the photos return day has come, if the photographer does not return the photos, the request will be cancelled automatically.

#### e. Supportability

* Online bookings in real time: The customer will be able to see all the available dates directly, and make a booking. If there is no further availability on one date, this will be blocked in real time to prevent further bookings.
* Intelligent calendar: An intelligent and easy-to-understand booking or occupancy calendar provides the user with a summary of all bookings, capacity and turnover for the required periods at a glance.
* Online and chat support for booking systems.
* Booking management via smartphone.

#### f. Design Constraints

* The software language is Vietnamese.
* The software process is Scrum.

#### g. Support Documents

The system does not use any references to support document writing.

#### h. Purchased Components

To support the system operation, we have used some third-party services such as:

* Google Map API
* AWS S3

## 5. Other Requirements

### Messages List

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| # | Message code | Message Type | Context | Content |
| 1 | MSG01 | In line | There is not any search result | Không có kết quả. |
| 2 | MSG02 | In red, under the text box | Input-required fields are empty | Thông tin này là bắt buộc. |
| 3 | MSG03 | Toast message | Updating asset(s) information successfully | Cập nhật thành công. |
| 4 | MSG04 | Toast message | Adding new asset successfully | Thêm \* thành công. |
| 5 | MSG05 | Toast message | Confirming email of asset hand-over is sent successfully | A confirmation email has been sent to {email\_address}. |
| 6 | MSG06 | Toast message | Resetting asset information successfully | Thay đổi \* thành công |
| 7 | MSG07 | Toast message | Deleting asset information successfully | Xóa \* thành công. |
| 8 | MSG08 | In red, under the text box | Input value length > max length | Bạn vui lòng nhập đủ{max\_length} ký tự. |
| 9 | MSG09 | In line | Username or password is not correct when clicking sign-in | Thông tin đăng nhập không chính xác. |
| 10 | MSG10 | In line | Password and Confirm password are not matched. | Mật khẩu nhập lại không khớp. |
| 11 | MSG11 | Toast message | Register successfully | Đăng ký thành công |
| 12 | MSG12 | In red, under the text box | Email is invalid | Email mail này không hợp lệ. Vui lòng nhập email khác |
| 13 | MSG13 | In red, under the text box | Username is existing | Tên đăng nhập của bạn đã có người sử dụng. Vui lòng nhập tên khác |
| 14 | MSG14 | Toast message | The customer book photographer successfully. | Gửi yêu cầu thành công. |
| 15 | MSG15 | In red, under the text box | The customer does not choose a calendar and address. | Bạn chưa chọn thông tin. |
| 16 | MSG16 | Notifications | The system sends notifications to the customer when photographers accept requests. | Yêu cầu của bạn đã được xác nhận |
| 17 | MSG17 | Notifications | The system sends notification to the customer when a photographer declines a request. | Yêu cầu của bạn đã bị từ chối. |
| 18 | MSG18 | In red, under the text box | Current password is invalid | Mật khẩu nhập lại chưa chính xác |
| 19 | MSG19 | Toast message | The system shows notifications to the admin when blocking users. | Bạn có muốn khóa người dùng này? |
| 20 | MSG20 | Toast message | The admin block user successfully. | Người dùng đã được khóa thành công. |
| 21 | MSG21 | Toast message | The system sends notifications to the user when the account has been blocked. | Tài khoản của bạn đã bị vi phạm. Vui lòng liên hệ với quản trị viên |
| 22 | MSG22 | Toast message | The system shows notifications to the admin when unblocking the user. | Bạn có muốn mở khóa người dùng này? |
| 23 | MSG23 | Toast message | The system sends notifications to the user when the account has unblocked. | Người dùng đã được mở khóa thành công |
| 24 | MSG24 | In red, under the text box | Information is duplicated | \* đã tồn tại. |

**Table 63 - Messages List**

# Software Design Description

## 1. Overall Description

### 1.1 Assumptions

Assuming all the users for this system have basic mobile knowledge and knows how to operate the system smoothly.

The process and algorithm for calculating available schedules are assumed to be given prior.

### 1.2 Design Constraints

Hard drive space: The system will need at least 8GB storage space and minimum 1GB Storage space in order to run the booking system.

## 2. System Architecture Design

### 2.1 Overall Architecture

**Figure 55 - Overall Architecture**

PBS system consists of three components, the spring boot web server provides APIs, two mobile applications for customers and photographers, and the web application for admin roles.

We also use several third-party services such as Aws S3 for image storage, Firebase cloud messaging for push notification functions and Amazon RDS for MySQL instances.

### 2.2 System Architecture

**2.2.1 Web server architecture**

We build our web server following n-tier architecture. We decided to use this pattern for several reasons:

* It has been proven to be a scalable architecture model.
* Provide reusability, maintainability and fault tolerance.
* Helps our members to create reusable and flexible functionality.

**2.2.2 Mobile app architecture**





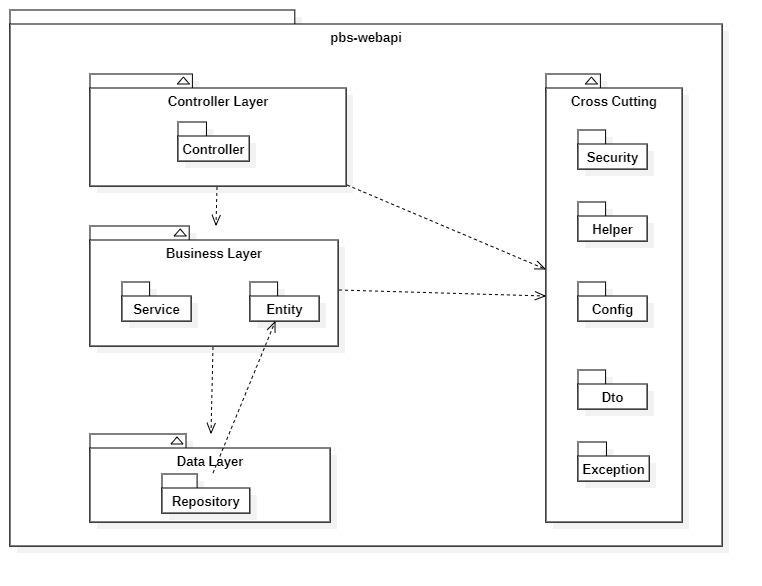
**Figure 56 - Mobile app architecture**

Bloc pattern allow us to separate our mobile app into 3 layers:

* Presentation: UI
* Business Logic Component: bloc (where we can manage events – click, tap, etc. - send a request to data layer, then retrieve data from data layer and response to input from UI through states).
* Data Layer: Repository (where we can retrieve data from sources).

### 2.3 Package Diagram

**2.3.1 Webapi**

****

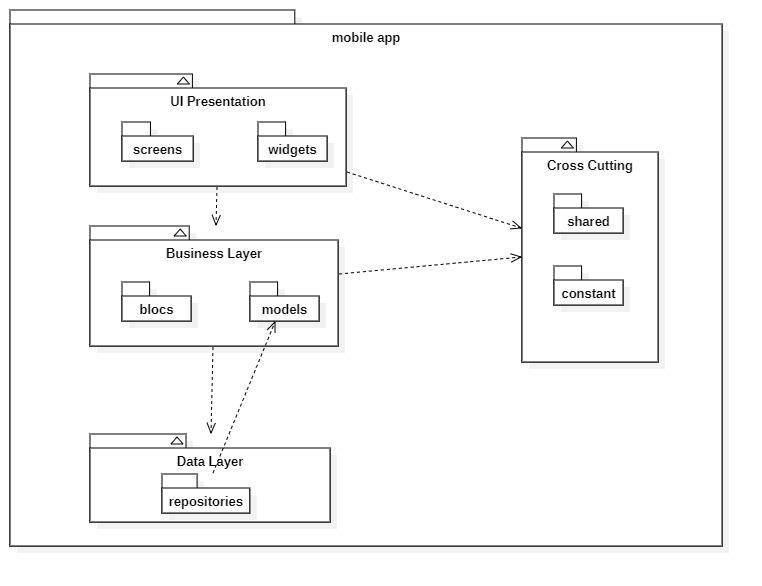
**Figure 57 - Webapi Package Diagram**

|  |  |  |
| --- | --- | --- |
| **No** | **Package** | **Description** |
| 01 | Controller | Package contains classes responsible for handling Http request and return responses with values |
| 02 | Service | Package contains classes responsible for handling business logic, validation and communicate with repository |
| 03 | Entity | Package contains classes represents the business logic entities, also responsible for code first approach with hibernate |
| 04 | Repository | Package contains classes responsible for executing query and communicate with database |
| 05 | Security | Package contains classes responsible for performing security includes jwt handling, authentication and authorization |
| 06 | Helper | Package contains classes responsible for utilizing like map helper, datetime helper, String helper |
| 07 | Config | Package contains classes responsible for configuring spring boot app, from security, application properties to firebase notification |
| 08 | Exception | Package contains classes responsible for handling exception, including status code mapping or error messages returning |
| 09 | Dto | Package contains classes responsible for sharping the data to return to clients |

**Table 64 - Webapi Package Diagram list**

**2.3.2 Mobile App**

Both mobile apps share a same project structuring, so we use one diagram to represents both of them here:

**Figure 58 - Mobile Package Diagram**

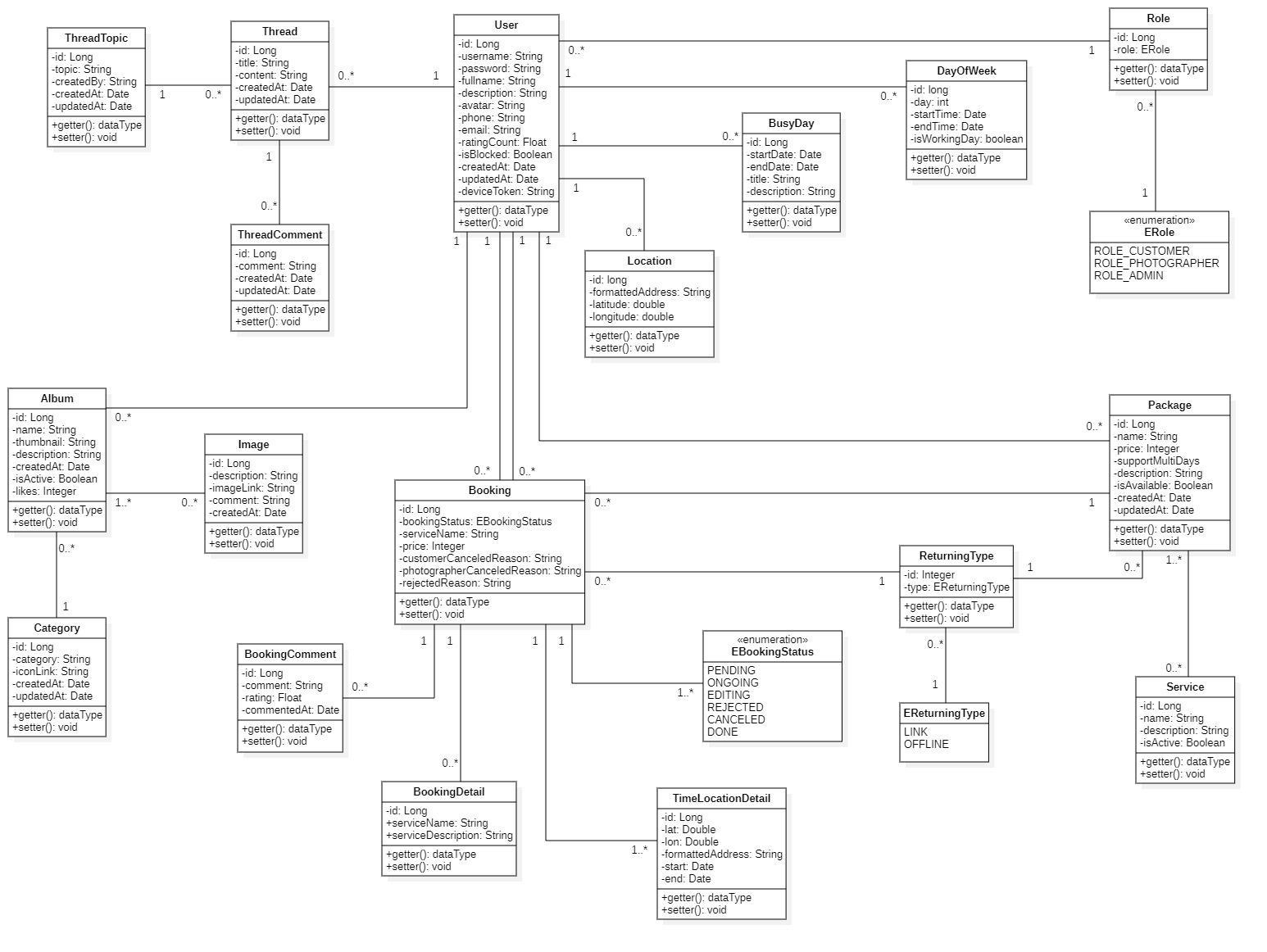
|  |  |  |
| --- | --- | --- |
| **No** | **Package** | **Description** |
| 01 | screens | Package contains classes for UI representation |
| 02 | widgets | Package contains classes to form the widgets used by screens |
| 03 | blocs | Package contains classes to implement the bloc architecture including bloc’s state, bloc’s event |
| 04 | models | Package contains classes to form an object extracted from api |
| 05 | repositories | Package contains classes responsible for communicating with api server |
| 06 | shared | Package contains classes responsible for utilizing and environment variables |
| 07 | constant | Package contains classes represents the constants used by application |

**Table 65 - Mobile Package Diagram List**

## 3. System Detailed Design

### 3.1 Overall

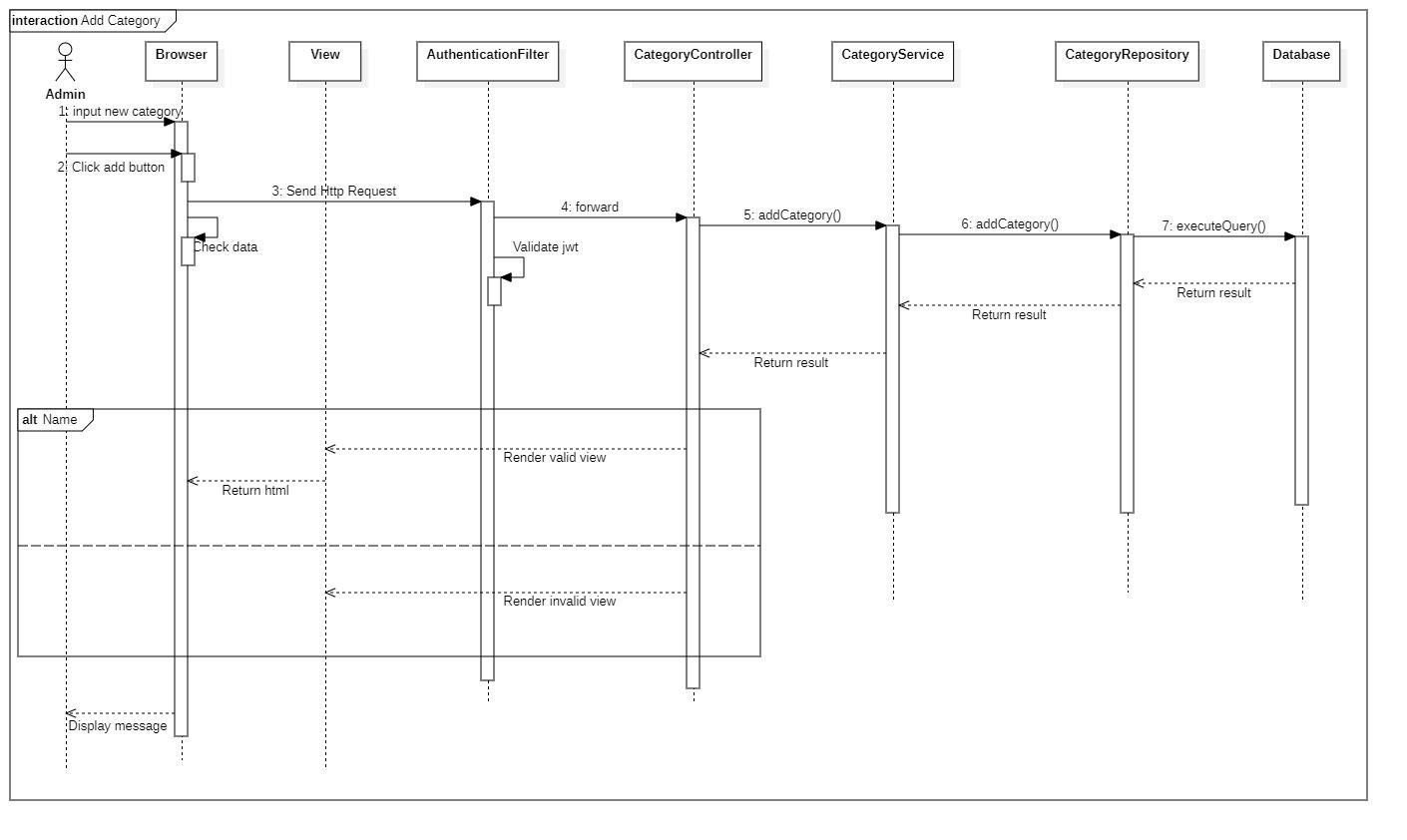
#### a. Class Diagram



**Figure 59 - Class Diagram**

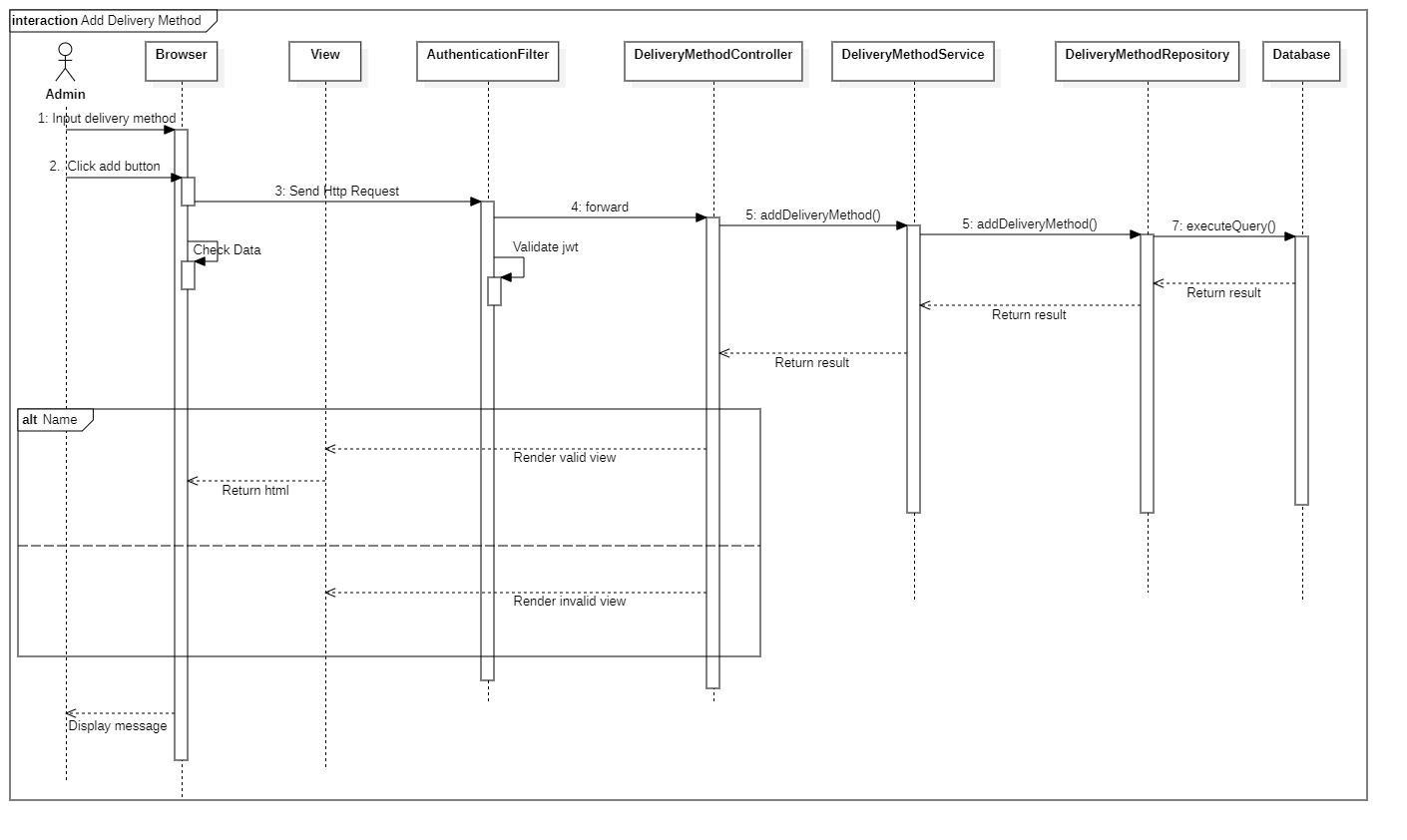
#### b. Sequence Diagram(s)

1. **Add Category**



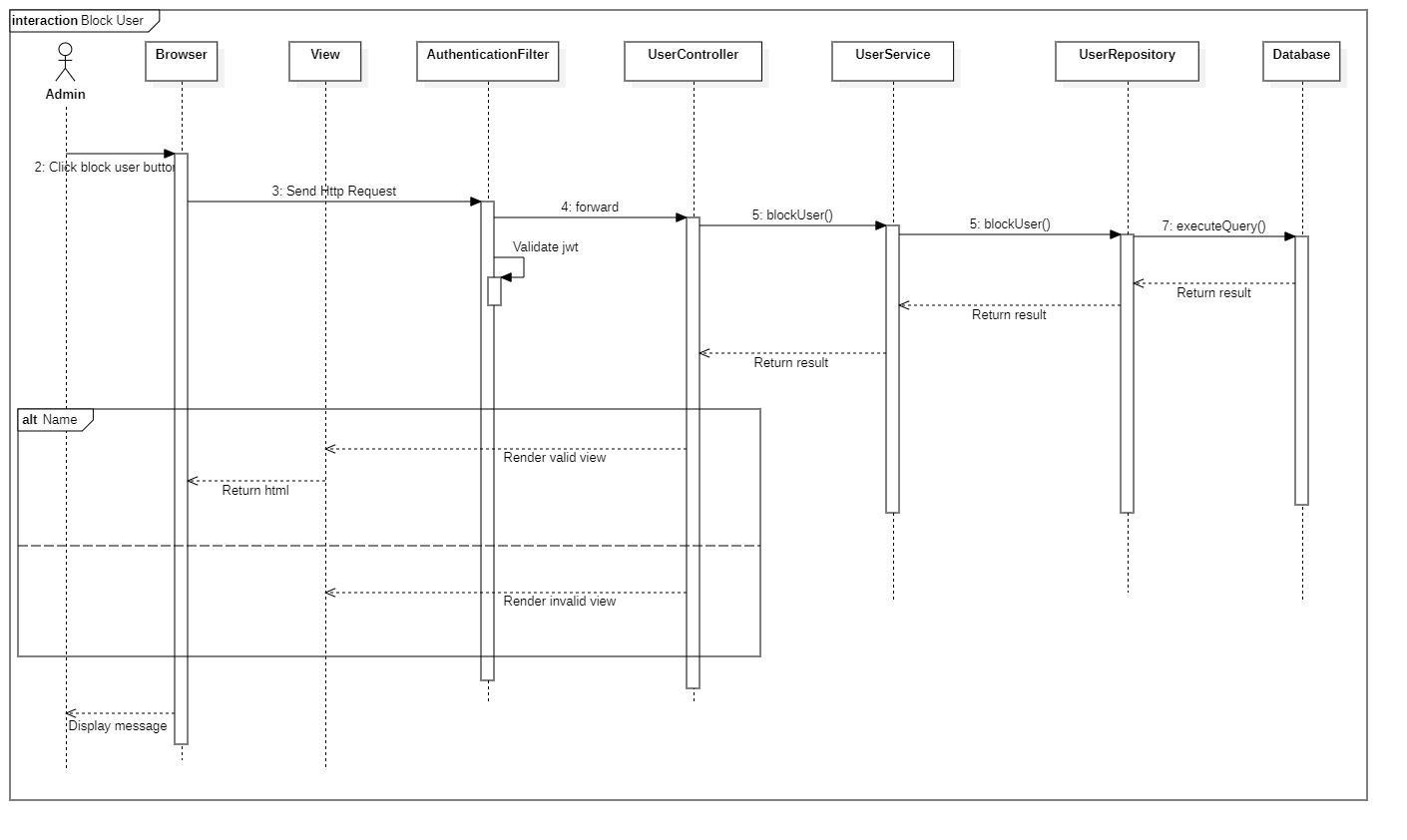
**Figure 60 - Sequence Diagram - Add Category**

1. **Add delivery method**



**Figure 61 - Sequence Diagram - Add Delivery Method**

1. **Block user**

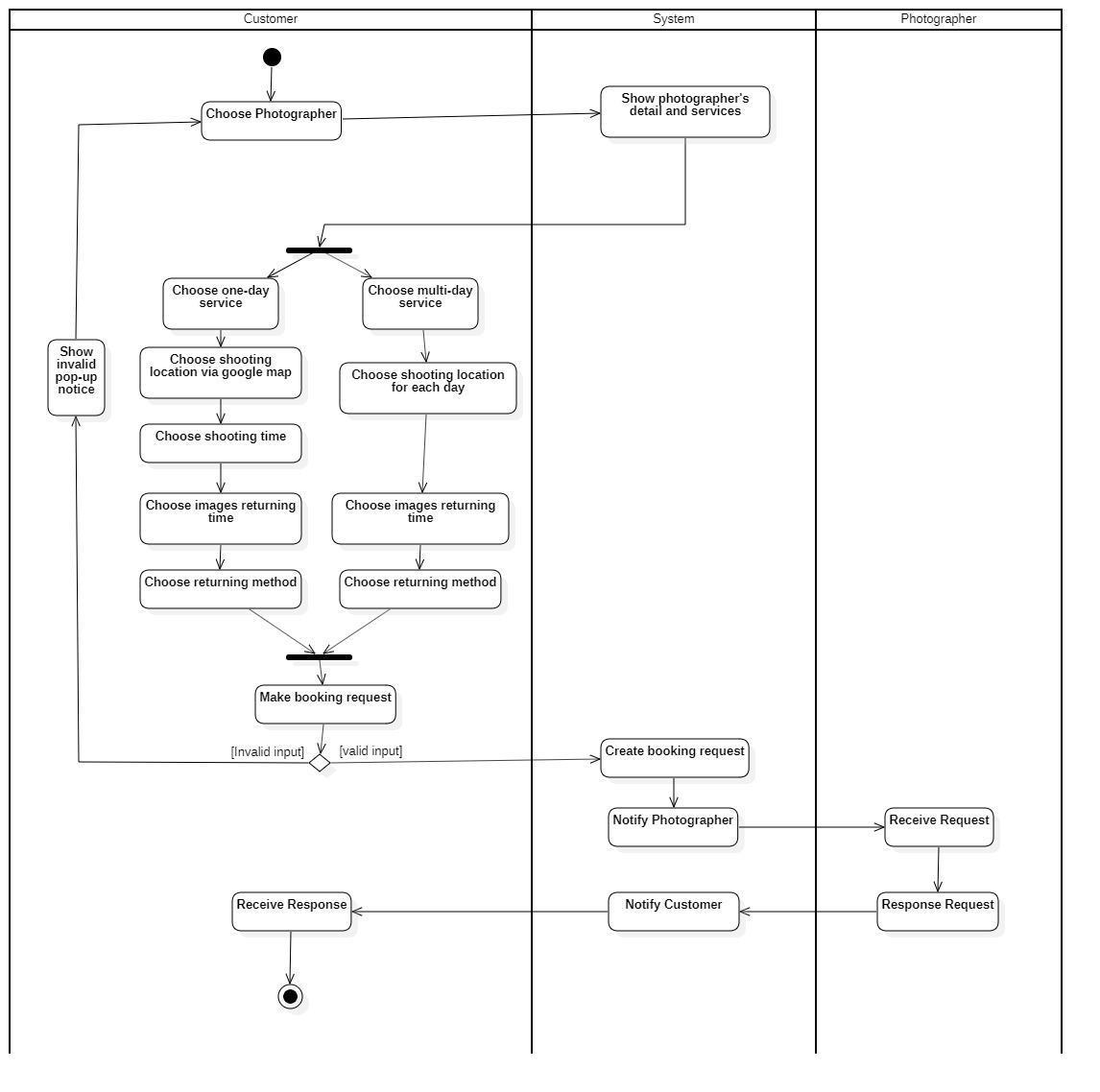


**Figure 62 - Sequence Diagram - Block User**

#### c. Activity Diagram(s)

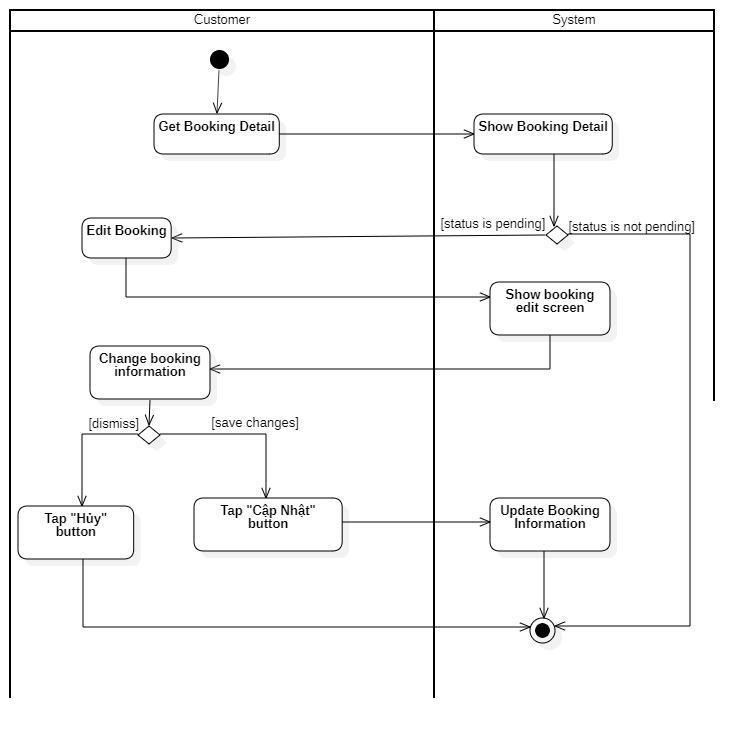
1. **<Customer>** **Make request**

Summary: This diagram shows the process when a customer makes booking request.

**Figure 63 - <Customer> Make request- Activity Diagram** 

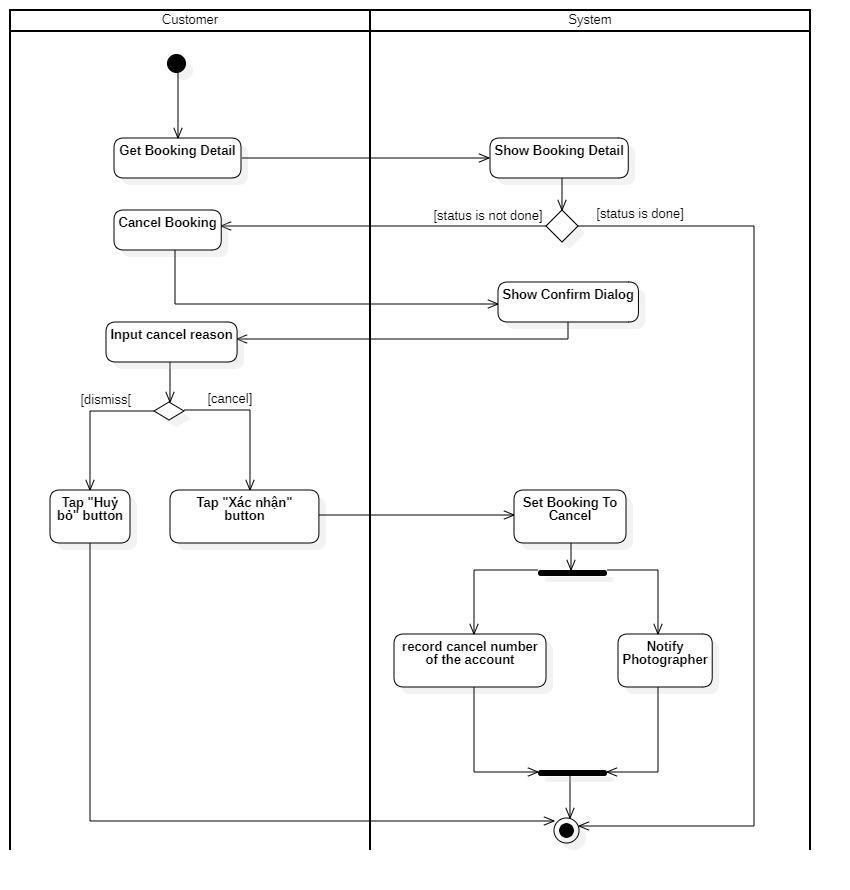
1. **<Customer> Edit request**

Summary: This diagram shows the process when customer edit request.

**Figure 64 - <Customer> Edit request- Activity Diagram** 

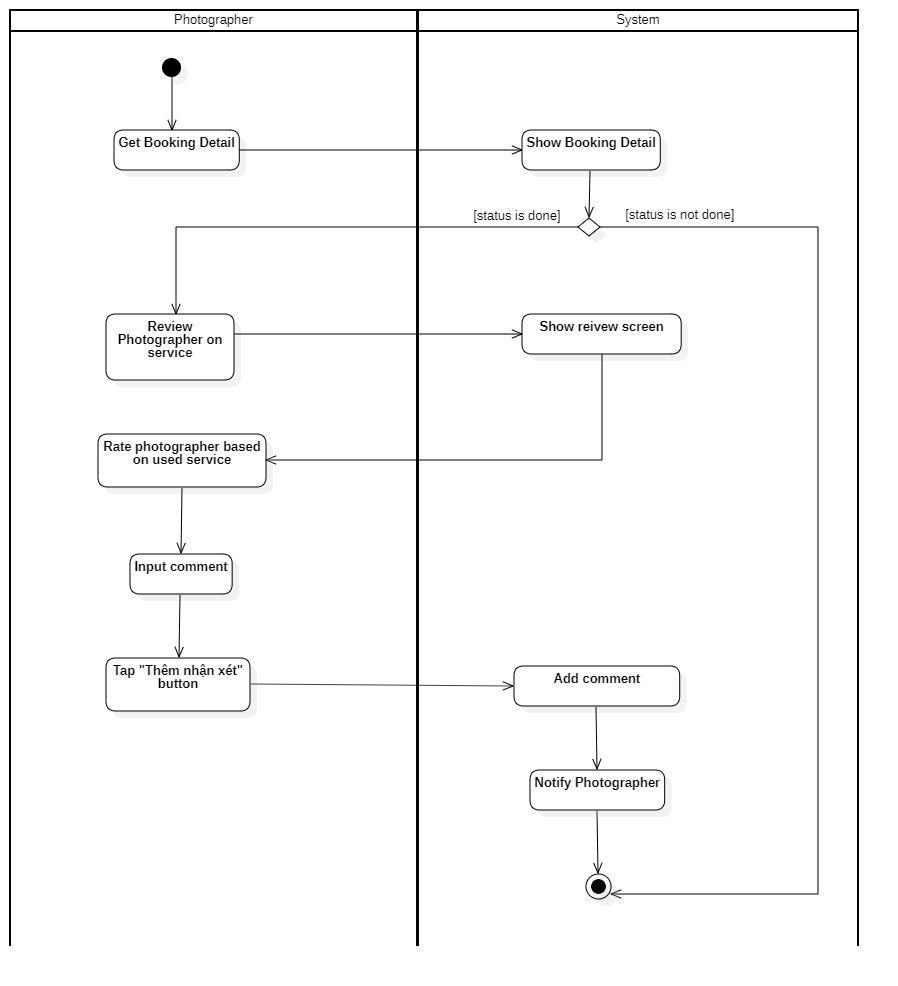
1. **<Customer> Cancel booking**

Summary: This diagram shows the process when customer cancels a request.

**Figure 65 - <Customer> Cancel booking - Activity Diagram**

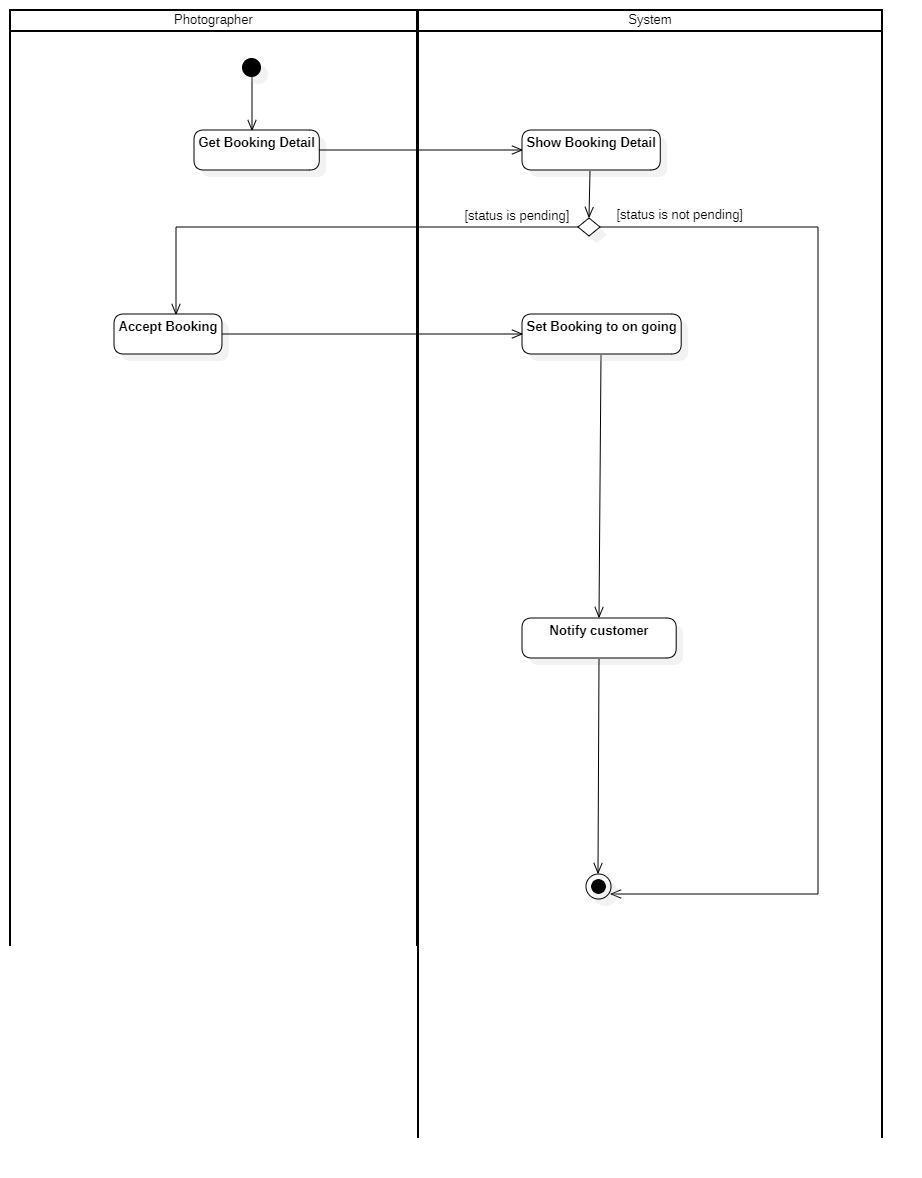
1. **<Customer> Rating**

Summary: This diagram shows the process when a customer rates photographer after booking finish.

**Figure 66 - <Customer> Rating - Activity Diagram**

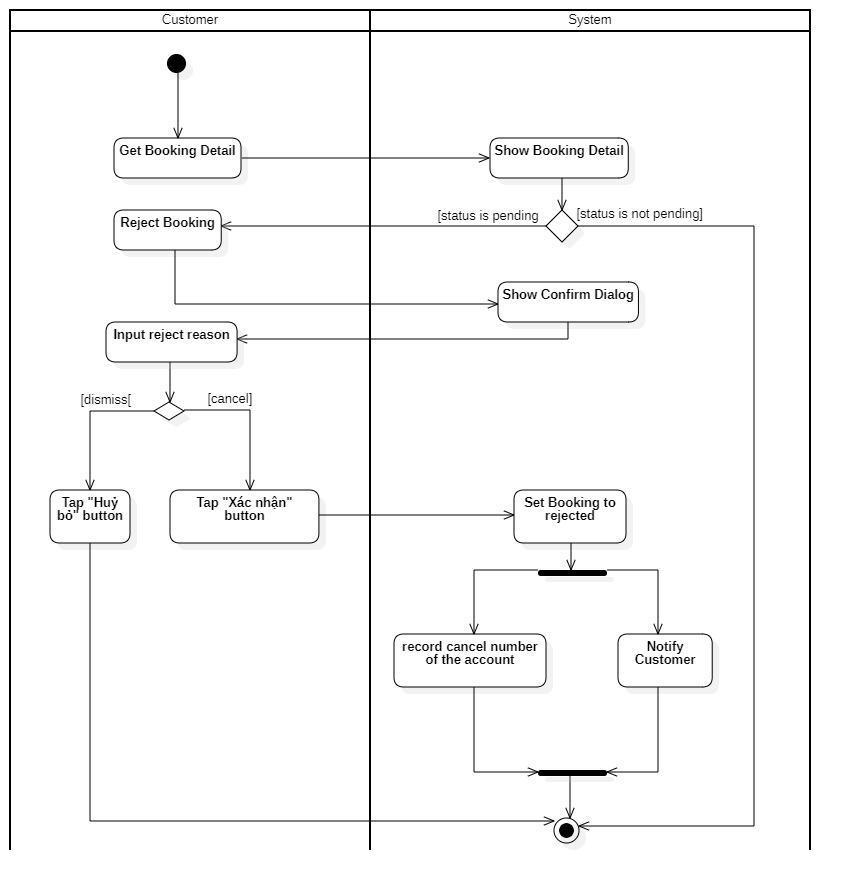
1. **<Photographer> Accept booking**

Summary: This diagram shows the process when photographer accepts a request.

**Figure 67 - <Photographer> Accept booking - Activity Diagram**

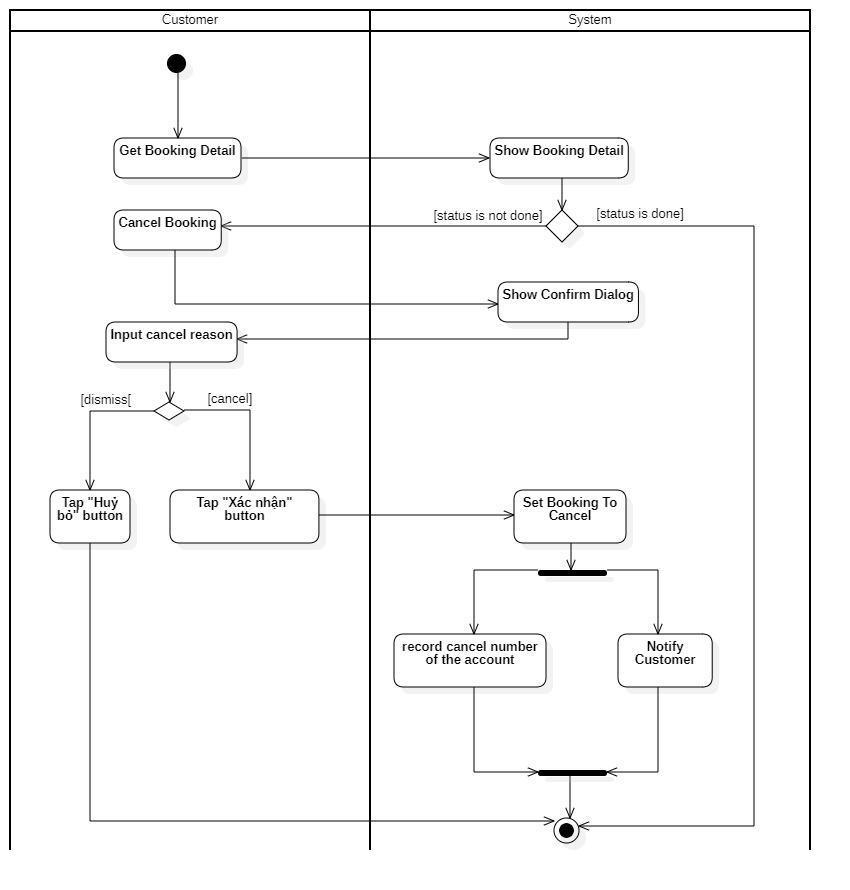
1. **<Photographer> Reject booking**

Summary: This diagram shows the process when photographer rejects a request.

**Figure 68 - <Photographer> Reject booking - Activity Diagram**

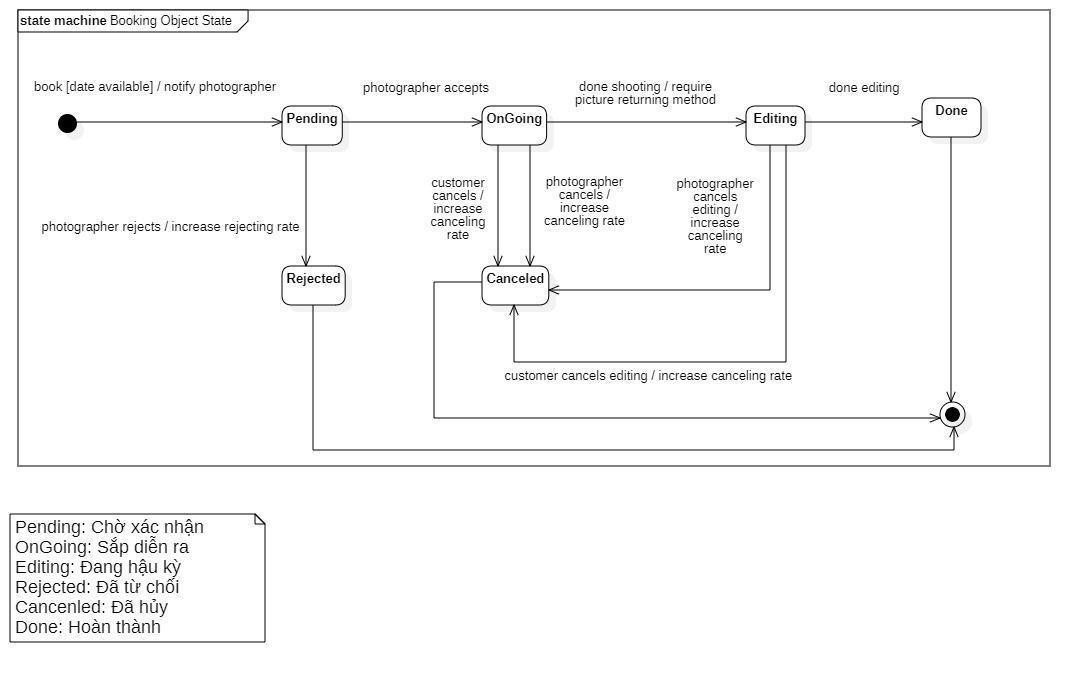
1. **<Photographer> Cancel booking**

Summary: This diagram shows the process when photographer cancels a request.

**Figure 69 - <Photographer> Cancel booking - Activity Diagram**

#### d. State Machine Diagram(s)

This state machine diagram describes states of a booking from start to done.



**Figure 70 - State Machine Diagram of Booking**

## 4. Class Specifications

### 4.1 User

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Long | private | Unique identifier of a user |
| username | String | private | The attribute represents username of a user |
| password | String | private | The attribute represents password of a user |
| fullname | String | private | The attribute represents full name of a user |
| description | String | private | The attribute represents user self-description |
| avatar | String | private | The attribute represents avatar link of a user |
| phone | String | private | The attribute represents phone of a user |
| email | String | private | The attribute represents email of an user |
| ratingCount | Float | private | The attribute represents the rating count of a user whose role is photographer |
| isBlocked | Boolean | private | The attribute represents if user has been blocked |
| createdAt | Date | private | The attribute represents created date of a user object |
| updatedAt | Date | private | The attribute represents updated date of a user object |
| deviceToken | String | private | The attribute represents user’s firebase device token |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | Attribute Type | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.2 Role

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Long | private | Unique identifier of a role |
| role | ERole | private | The attribute represents role name |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | Attribute Type | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.3 ERole

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| ROLE\_CUSTOMER |  |  | Enum attribute |
| ROLE\_PHOTOGRAPHER |  |  | Enum attribute |
| ROLE\_ADMIN |  |  | Enum attribute |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | Attribute Type | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.4 Package

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Integer | private | Unique identifier of a package |
| name | String | private | The attribute represents the package name |
| price | Integer | private | The attribute represents the package price |
| supportMultiDays | Boolean | private | The attribute represents if this package support multi days deal |
| description | String | private | The attribute represents the package description |
| isAvailable | Boolean | private | The attribute represents if the package still available |
| createdAt | Date | private | The attribute represents created date of a package object |
| updatedAt | Date | private | The attribute represents updated date of a package object |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | Attribute Type | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.5 Service

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Long | private | Unique identifier of a package |
| name | String | private | The attribute represents the name of a service |
| description | String | private | The attribute represents the service’s description |
| isActive | Boolean | private | The attribute represents if the service is still available |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | Attribute Type | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.6 Booking

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Long | private | Unique identifier of a booking |
| bookingStatus | EBookingStatus | private | The attribute represents the status of a booking |
| serviceName | String | private | The attribute represents the service of a booking |
| price | Integer | private | The attribute represents the service price of a booking |
| customerCancelledReason | String | private | The attribute represents the reason if customer decided to cancel a booking |
| photographerCancelledReason | String | private | The attribute represents the reason if photographer decided to cancel a booking |
| rejectedReason | String | private | The attribute represents the reason if photographer decided to reject a booking |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | Attribute Type | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.7 EBookingStatus

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| Pending |  |  | Enum attribute |
| OnGoing |  |  | Enum attribute |
| Editing |  |  | Enum attribute |
| Rejected |  |  | Enum attribute |
| Cancelled |  |  | Enum attribute |
| Done |  |  | Enum attribute |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | Attribute Type | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.8 Album

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Long | private | Unique identifier of album |
| name | String | private | The attribute represents the name album |
| thumbnail | String | private | The attribute represents the album’s thumbnail link |
| description | String | private | The attribute represents the album’s description |
| createdAt | Date | private | The attribute represents the created date of album object |
| isActive | Boolean | private | The attribute represents if album still active |
| likes | Integer | private | The attribute represents the album like count |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | Attribute Type | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.9 Image

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Long | private | Unique identifier of image |
| description | String | private | The attribute represents the description of album |
| imageLink | String | private | The attribute represents the album’s image link |
| comment | String | private | The attribute represents the album’s comment |
| createdAt | Date | private | The attribute represents the created date of image object |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | Attribute Type | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.10 Category

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Long | private | Unique identifier of category |
| category | String | private | The attribute represents the category name |
| iconLink | String | private | The attribute represents the category icon link |
| createdAt | Date | private | The attribute represents the created date of category object |
| updatedAt | Date | private | The attribute represents the updated date of category object |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | Attribute Type | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.11 DayOfWeek

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Long | private | Unique identifier of day of week |
| day | Int | private | The attribute represents the day of the week |
| startTime | Date | private | The attribute represents the start time of photographer’s working time |
| endTime | Date | private | The attribute represents the end time of photographer’s working time |
| isWorkingDay | Boolean | private | The attribute represents if this day of week is working day |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | Attribute Type | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.12 Location

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Long | private | Unique identifier of location |
| formattedAddress | String | private | The attribute represents the location full address |
| longitude | String | private | The attribute represents the longitude of a location |
| latitude | Double | private | The attribute represents the latitude of a location |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | Attribute Type | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.13 BookingComment

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Long | private | Unique identifier of booking comment |
| comment | String | private | The attribute represents the comment of user |
| rating | Float | private | The attribute represents the rating of user |
| commentedAt | Date | private | The attribute represents the commented date of a booking |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | Attribute Type | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.14 BookingDetail

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Long | private | Unique identifier of booking detail |
| serviceName | String | private | The attribute represents the booked service’s name |
| serviceDescription | String | private | The attribute represents the booked service’s description |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | Attribute Type | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.15 BusyDay

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Long | private | Unique identifier of busy day |
| startDate | Date | private | The attribute represents the start date of photographer’s busy event |
| endDate | Date | private | The attribute represents the end date of photographer’s busy event |
| title | String | private | The attribute represents the title of photographer’s busy event |
| description | String | private | The attribute represents the description of this busy event |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | Attribute Type | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.16 Thread

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Long | private | Unique identifier of thread |
| title | String | private | The attribute represents the title of thread |
| content | String | private | The attribute represents the content of a thread |
| createdAt | Date | private | The attribute represents the created date of a thread |
| updatedAt | Date | private | The attribute represents the updated date of a thread |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | Attribute Type | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.17 ThreadTopic

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Long | private | Unique identifier of thread topic |
| topic | String | private | The attribute represents the topic name of a thread |
| createdBy | String | private | The attribute represents the name of the person who created this topic |
| createdAt | Date | private | The attribute represents the created date of a thread |
| updateAt | Date | private | The attribute represents the updated date of a thread |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | Attribute Type | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.18 ThreadComment

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Long | private | Unique identifier of thread comment |
| comment | String | private | The attribute represents the comment of a thread |
| createdAt | Date | private | The attribute represents the created date of this record |
| updateAt | Date | private | The attribute represents the updated date of this record |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | Attribute Type | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.19 ReturningType

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Long | private | Unique identifier of returning type |
| type | EReturningType | private | The attribute represents the type |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | Attribute Type | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.20 EReturningType

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| LINK |  |  | Enum attribute |
| OFFLINE |  |  | Enum attribute |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | Attribute Type | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.21 TimeLocationDetail

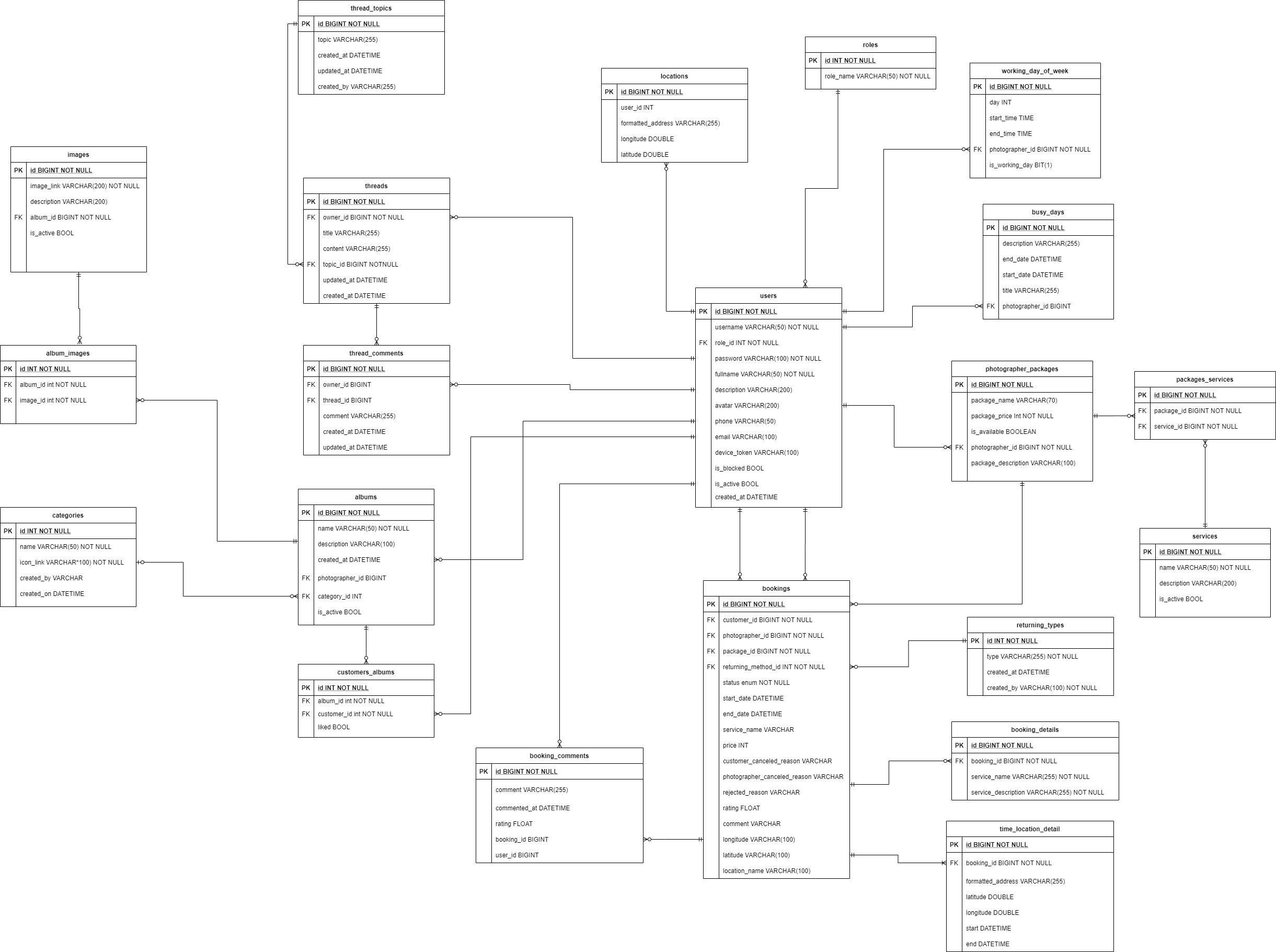
This table used for saving time and location details of a booking

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | Long | private | Unique identifier of time location detail |
| lat | Double | private | The attribute represents the latitude of a user |
| lon | Double | private | The attribute represents the longitude of a user |
| formattedAddress | String | private | The attribute represents the formatted address of a user |
| start | Date | private | The attribute represents start time of a booking |
| end | Date | private | The attribute represents end time of a booking |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | Attribute Type | get attribute value |
| 2 | setter | public | void | set attribute value |

## 5. Data & Database Design

### 5.1 Database Design



*Figure 16 Database Design*

#### 5.1.1 users

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | id | bigint |  | X | X | X |  |
| 2 | username | varchar | 50 | X | X |  |  |
| 3 | password | varchar | 100 |  | X |  |  |
| 4 | fullname | varchar | 50 |  | X |  |  |
| 5 | description | varchar | 200 |  |  |  |  |
| 6 | avatar | varchar | 200 |  |  |  | Avatar image link of user |
| 7 | phone | varchar | 50 |  |  |  |  |
| 8 | email | varchar | 100 |  |  |  |  |
| 9 | device\_token | varchar | 100 |  |  |  | The firebase token of device, using for firebase messaging. |
| 9 | is\_blocked | boolean |  |  |  |  |  |
| 10 | is\_active | boolean |  |  |  |  |  |
| 11 | created\_at | datetime |  |  |  |  |  |
| 12 | role\_id | int |  |  | X | X |  |

#### roles

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | id | int |  | X | X | X |  |
| 2 | role\_name | varchar | 50 |  | X |  |  |

#### photographer\_packages

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | id | bigint |  | X | X | X |  |
| 2 | package\_name | varchar | 70 |  | X |  |  |
| 3 | package\_price | int |  |  | X |  |  |
| 4 | is\_available | boolean |  |  |  |  |  |
| 5 | package \_description | varchar | 100 |  |  |  |  |
| 6 | photographer \_id | int |  |  | X | X |  |

#### packages\_services

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | id | bigint |  | X | X | X |  |
| 2 | package\_id | int |  |  | X | X |  |
| 3 | service\_id | int |  |  | X | X |  |

#### services

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | id | bigint |  | X | X | X |  |
| 2 | name | varchar | 50 |  | X |  |  |
| 3 | description | varchar | 200 |  |  |  |  |
| 4 | is\_active | boolean |  |  |  |  |  |

#### bookings

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | id | bigint |  | X | X | X |  |
| 2 | status | enum |  |  | X |  | There will be 6 status: Pending, Ongoing, Editing, Done, Rejected, Canceled |
| 3 | start\_date | datetime |  |  | X |  |  |
| 4 | end\_date | datetime |  |  | X |  |  |
| 5 | service\_name | varchar | 100 |  | X |  | Package’s name |
| 6 | price | int |  |  | X |  | Package’s price |
| 7 | customer  \_canceled  \_reason | varchar | 100 |  |  |  | In case customers cancel and provide their reason |
| 8 | photographer \_canceled \_reason | varchar | 100 |  |  |  | In case photographers cancel and provide their reason |
| 9 | rejected\_reason | varchar | 100 |  |  |  | In case photographers reject the request and provide reason |
| 10 | rating | float |  |  |  |  | After booking is done, customer can rate the booking |
| 11 | comment | varchar | 100 |  |  |  | After booking is done, customer can comment on the booking |
| 12 | longitude | varchar | 100 |  |  |  |  |
| 13 | latitude | varchar | 100 |  |  |  |  |
| 14 | location\_name | varchar | 100 |  |  |  | The shooting location picked by customer |
| 15 | customer\_id | int |  |  | X | X |  |
| 16 | photographer \_id | int |  |  | X | X |  |
| 17 | package\_id | int |  |  | X | X |  |

#### albums

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | id | bigint |  | X | X | X |  |
| 2 | name | varchar | 50 |  | X |  |  |
| 3 | description | varchar | 100 |  |  |  |  |
| 4 | created\_at | datetime |  |  |  |  |  |
| 5 | is\_active | boolean |  |  |  |  |  |
| 6 | photographer \_id | int |  |  | X | X |  |
| 7 | category\_id | int |  |  |  | X |  |

#### locations

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | id | bigint |  | X | X | X |  |
| 2 | Formatted\_address | varchar | 255 |  | X |  |  |
| 3 | longitude | double | 100 |  | X |  |  |
| 4 | latitude | double | 100 |  | X |  |  |
| 5 | user\_id | int |  |  | X | X |  |

#### categories

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | id | int |  | X | X | X |  |
| 2 | name | varchar | 50 |  | X |  |  |
| 3 | icon\_link | varchar | 100 |  | X |  | Link for the icon of category |
| 4 | created\_by | varchar | 100 |  |  |  | Name of the admin who created the category |
| 5 | Created\_on | datetime |  |  |  |  |  |

#### album\_images

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | id | int |  | X | X | X |  |
| 2 | album\_id | int |  |  | X | X |  |
| 3 | image\_id | int |  |  | X | X |  |

#### images

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | id | bigint |  | X | X | X |  |
| 2 | image\_link | varchar | 200 |  | X |  |  |
| 3 | description | varchar | 200 |  |  |  |  |
| 4 | is\_active | boolean |  |  |  |  |  |
| 5 | album\_id | int |  |  | X | X |  |

#### customers\_albums

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | id | int |  | X | X | X |  |
| 2 | liked | boolean |  |  |  |  | represent if customer has liked the album |
| 3 | album\_id | int |  |  | X | X |  |
| 4 | customer\_id | int |  |  | X | X |  |

#### working\_day\_of\_week

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | id | bigint |  | X | X | X |  |
| 2 | day | int |  |  |  |  |  |
| 3 | start\_time | datetime |  |  |  |  |  |
| 4 | end\_time | datetime |  |  |  |  |  |
| 5 | is\_working\_day | bit |  |  |  |  |  |
| 6 | photographer \_id | bigint |  |  | X | X |  |

#### busy\_days

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | id | bigint |  | X | X | X |  |
| 2 | description | varchar | 255 |  |  |  |  |
| 3 | start\_date | datetime |  |  |  |  |  |
| 4 | end\_date | datetime |  |  |  |  |  |
| 5 | title | varchar | 255 |  |  |  |  |
| 6 | photographer \_id | bigint |  |  | X | X |  |

#### returning\_types

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | id | int |  | X | X | X |  |
| 2 | type | varchar | 255 |  | X |  |  |
| 3 | created\_at | datetime |  |  |  |  |  |
| 4 | created\_by | varchar | 100 |  | X |  |  |

#### booking\_details

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | id | bigint |  | X | X | X |  |
| 2 | service\_name | varchar | 255 |  | X |  |  |
| 3 | service\_description | varchar | 255 |  | X |  |  |
| 4 | booking\_id | bigint |  |  | X | X |  |

#### time\_location\_details

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | id | bigint |  | X | X | X |  |
| 2 | formatted\_address | varchar | 255 |  |  |  |  |
| 3 | laitude | double |  |  |  |  |  |
| 4 | longitude | double |  |  |  |  |  |
| 5 | start | datetime |  |  |  |  |  |
| 6 | end | datetime |  |  |  |  |  |
| 7 | booking\_id | bigint |  |  | X | X |  |

#### booking\_comments

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | id | bigint |  | X | X | X |  |
| 2 | comment | varchar | 255 |  |  |  |  |
| 3 | commented\_at | datetime |  |  |  |  |  |
| 4 | rating | float |  |  |  |  |  |
| 5 | user\_id | bigint |  |  |  |  |  |
| 6 | booking\_id | bigint | 100 |  |  |  |  |

#### threads

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | id | bigint |  | X | X | X |  |
| 2 | title | varchar | 255 |  |  |  |  |
| 3 | content | varchar | 255 |  |  |  |  |
| 4 | update\_at | datetime |  |  |  |  |  |
| 5 | created\_at | datetime |  |  |  |  |  |
| 6 | topic\_id | bigint |  |  | X | X |  |
| 7 | owner\_id | bigint |  |  | X | X |  |

#### thread\_comments

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | id | bigint |  | X | X | X |  |
| 2 | comment | varchar | 255 |  |  |  |  |
| 3 | update\_at | datetime |  |  |  |  |  |
| 4 | created\_at | datetime |  |  |  |  |  |
| 5 | thread\_id | bigint |  |  |  | X |  |
| 6 | owner\_id | bigint |  |  |  | X |  |

#### thread\_topics

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | id | bigint |  | X | X | X |  |
| 2 | topic | varchar | 255 |  |  |  |  |
| 3 | created\_by | varchar | 255 |  |  |  |  |
| 4 | update\_at | datetime |  |  |  |  |  |
| 5 | created\_at | datetime |  |  |  |  |  |

### 5.2 Data File Design

We use Aws S3 to store files and images.

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **File Name** | **Type** | **Notes** |
| 1 | User avatar | Image | uri: “{baseUrl}/api/users/{userId}/{userId}-avatar” |
| 2 | Photographer album thumbnail | Image | uri: “{baseUrl}/api/photographers/{ptgId}/{albumid}/{albumId}-thumbnail” |
| 3 | Photographer album image | Image | uri: “{baseUrl}/api/photographers /{ptgId}/{albumId}/{imageId}-image” |
| 4 | Photographer cover image | Image | uri: “{baseUrl}/api/photographers/{ptgId}/{ptgId}-cover” |
| 5 | Category icon image | Image | uri: ““{baseUrl}/api/categories/{categoryId}-category” |

## 6. Algorithm

### 6.1 Multiple-factors weighted sorting

#### 6.1.1 Definition

Multiple-factors weighted ranking is considered to be a simple yet effective algorithm.

#### 6.1.2 Define problem

A suggestion method is always needed in every kind of commercial application. Since it gives the customers an objective way to decide whether a photographer is trustable or not.

It also boosts the photographers’ motivation to give the best of them in order to connect to more customers.

#### 6.1.3 Solution

**Step 1: Choose suitable factors**

We will rank the photographers based on the following factors:

* Customers’ ratings.
* Photographer’s average service price.
* Distance (from photographer’s working location to the booked location).

**Step 2: Specify weight for each factor**

Admin can change the weights.

|  |  |
| --- | --- |
| **Factor** | **Weight** |
| Rating | 0.3 |
| Distance | 0.3 |
| Price | 0.4 |

**Step 3: Set a common scale for factors**

Common scale = from 0 (worst) to 100 (best)

**Step 4: Transform each object’s factors score to the common scale**

* Rating: 1 rating point on scale of 5 will equal 20 points on scale of 100
* Distance:

Calculate total distances of all photographers in the list

Calculate score of a photographer’s distance factor = (1 – distance / sum) \* max scale

(photographers with higher distance will get lower score)

* Photographer’s average service price

Calculate total prices of all photographers in the list

Calculate score of a photographer’s price factor = (1 – price / sum) \* max scale

(Photographer with higher price will get lower score)

**Step 5: Sum the scores**

Score of a photographer = 0.3 \* rating score + 0.3 \* distance score + 0.3 \* price score

**Step 6: Sort photographers based on the score.**

#### 6.1.4 Complexity

The complexity of this algorithm is O(n) for finding n photographers.

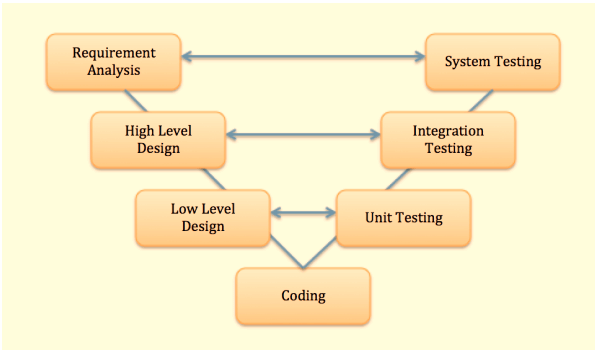
#### 6.1.5 Flowchart

*Figure 17 Flowchart - Multiple factors sorting*

# Software Testing Documentation

## 1. Overall Description

### 1.1 Test Model

We use a software development process model is V-model because this model is suitable for project conditions. For each stage, there will be a corresponding check it helps us to detect errors from the beginning.

**Figure 71 - V-model**

References: *https://viblo.asia/p/v-model-trong-kiem-thu-phan-mem-la-gi-tim-hieu-voi-vi-du-sdlc-stlc-Qbq5QMEL5D8*

### 1.2 Testing Levels

**1.2.1 Unit test**

**1.2.2 Integration test**

**1.2.3 System test**

**1.2.4 Acceptance test**

**1.2.4.1 Alpha test**

**1.2.4.2 Beta test**

### 1.3 Testing Types

**1.3.1 Functional testing**

* Register
* Login
* Forgot password
* Update profile
* Change password
* Book photographer
* Edit request
* Remove request
* Review photographer (comment and rating)
* Accept request the customer
* Decline to request the customer.
* Change status request
* Add working day
* Add busy day
* The photographer edit request
* Add service
* Edit service
* Remove service
* Add album
* Edit album
* Delete album
* Open thread
* Comment thread

**1.3.2 Non-functional testing**

* Session timeout in 30 minutes.

## 2. Test Plan

### 2.1 Test Stages

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Type of Test** | **Stage of Test** | | | |
| ***Unit*** | ***Integration*** | ***System*** | ***Acceptance*** |
| Interface test |  | X | X | X |
| Functional test |  | X | X | X |
| Load test |  |  | X | X |
| Usability test |  |  | X | X |
| Security test | X |  | X |  |

**Table 66 - Test Stages**

### 2.2 Resources

#### a. Human Resources

|  |  |  |
| --- | --- | --- |
| **Worker/Doer** | **Role** | **Specific Responsibilities/Comments** |
| Trần Thiên Thảo | Leader | * Managing process * Designing database * Clarifying requirements * Prepare documents * GUI Design * Create test plan * Coding * Testing * Arrange Meeting * Risk Management |
| Bồ Công Đạt | Member | * Designing database * Clarifying requirements * Prepare documents * GUI Design * Create test plan * Coding * Testing |
| Đào Sỹ Trung Kiên | Member | * Designing database * Clarifying requirements * Prepare documents * GUI Design * Create test plan * Coding * Testing |
| Trương Ngọc Mỹ | Member | * Clarifying requirements * Prepare documents * GUI Design * Create test plan * Coding * Testing |

**Table 67 - Human Resources**

#### b. Environment

|  |  |  |  |
| --- | --- | --- | --- |
| **Purpose** | **Tool** | **Provider** | **Version** |
| Postman |  |  |  |
| Excel microsoft |  |  |  |
| Chrome |  |  |  |
| Mobile phone |  |  |  |

**Table 68 - Environment**

### 2.3 Test Milestones

|  |  |  |  |
| --- | --- | --- | --- |
| **Milestone Task** | **Effort (md)** | **Start Date** | **End Date** |
| Unit test |  | 20/11/2020 | 15/12/2020 |
| Test case |  | 20/11/2020 | 15/12/2020 |
| Test procedures |  | 05/12/2020 | 13/12/202 |
| Defect reports |  | 15/12/2020 | 20/12/2020 |

**Table 69 - Test Milestones**

### 2.4 Deliverables

|  |  |  |
| --- | --- | --- |
| **No** | **Deliverables** | **Due Date** |
| 1 | Unit test | 15/12/2020 |
| 2 | Test case | 15/12/2020 |

**Table 70 - Deliverables**

## 3. Test Cases

* Unit Test Cases: Report5\_Unit Test Case.xls
* Other Test Cases: Report5\_Test Case Document.xlsx

## 4. Test Reports

Other Test Cases: Report5\_Test Case Document.xlsx

# Release Package & User Guides

## 1. Deliverable Package

### 1.1 Source codes & documents \*chưa fix

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Items** | **Sub-Items** | **Type** | **Version** |
| ***Code Package*** | | | | |
| 1 | Web API | pbs\_webapi | New | 1.0.1 |
| 2 | Mobile Apps | photographer\_mobile\_app | New | 1.0.1 |
| customer\_mobile\_app | New | 1.0.1 |
| ***Database*** | | | | |
| 1 | Tables | pbs\_users.sql | New | 1.0.1 |
| pbs\_roles.sql | New | 1.0.1 |
| pbs\_photographer\_packages.sql | New | 1.0.1 |
| pbs\_packages\_services.sql | New | 1.0.1 |
| pbs\_services.sql | New | 1.0.1 |
| pbs\_returning\_types.sql | New | 1.0.1 |
| pbs\_bookings.sql | New | 1.0.1 |
| pbs\_booking\_details.sql | New | 1.0.1 |
| pbs\_booking\_time\_location\_detail.sql | New | 1.0.1 |
| pbs\_booking\_comments.sql | New | 1.0.1 |
| pbs\_albums.sql | New | 1.0.1 |
| pbs\_customers\_albums.sql | New | 1.0.1 |
| pbs\_album\_images.sql | New | 1.0.1 |
| pbs\_images.sql | New | 1.0.1 |
| pbs\_categories.sql | New | 1.0.1 |
| pbs\_threads.sql | New | 1.0.1 |
| pbs\_thread\_comments.sql | New | 1.0.1 |
| pbs\_thread\_topics.sql | New | 1.0.1 |
| pbs\_locations.sql | New | 1.0.1 |
| pbs\_working\_day\_of\_week.sql | New | 1.0.1 |
| pbs\_busy\_days.sql | New | 1.0.1 |
| ***Documents*** | | | | |
| 1 | Introduction | Report1\_Project-Introduction.docx | New | 1.0.1 |
| 2 | Management Plan | Report2\_Project-Management-Plan.docx | New | 1.0.1 |
| 3 | Requirement | Report3\_Software Requirement Specification.docx | New | 1.0.1 |
| 4 | Design | Report4\_Software Design Document.docx | New | 1.0.1 |
| 5 | Testing | Report5\_Test Documentation.docx | New | 1.0.1 |
| Report5\_Test Case Document.xlsx | New | 1.0.1 |
| Report5\_Unit Test Case.xls | New | 1.0.1 |
| 6 | User guides | Report6\_Software User Guides.docx | New | 1.0.1 |
| 7 | Final | Report7\_Final Project Report.docx | New | 1.0.1 |

**Table 71 - Source codes & documents**

### 1.2 Known Issues, Limitations & Restrictions \*bàn lại sau

* The application has not supported payment.
* The system does not have a photo verification feature.
* The system has not provided a website version to both customers and photographers. Users may find it uncomfortable to use applications.

## 2. Installation Guides

### 2.1 System Requirements

**2.1.1 Hardware requirements**

For PC

|  |  |  |
| --- | --- | --- |
| **PC** | **Minimum Requirements** | **Recommended** |
| Internet Connection | Cable, Wi-Fi (4 Mbps) | Cable, Wi-Fi (8 Mbps) |
| Operating System | Windows 7 | Windows 7 or more. |
| Computer Processor | Intel® Core i3 1.4GHz | Intel® Core i5 2.50GHz |
| Computer Memory | 1GB RAM | 2GB RAM or more |
| Web Browser | Chromes (v42 or higher) | Chrome latest stable version |

For Android Devices

|  |  |
| --- | --- |
| **Operating system** | Android 4.0 or higher |
| **Processor** | Intel Atom® Processor Z2520 1.2 GHz, or faster processor |
| **Storage** | Between 850 MB and 1.2 GB, depending on the language version |
| **RAM** | Minimum of 512 MB, 2 GB is recommended |
| **Hard Disk** | * 2 GB of available hard-disk space for installation; extra free space is required during installation. * You cannot install using a removable flash storage device. |
| **Video** | 1920 x 1080 pixels or higher |
| **Software** |  |
| **Browser/Internet** | * This application is designed to work online. * To download and launch Google Play Store apps within the application, a high-speed Internet connection is recommended. |

**2.1.2 Software requirements**

|  |  |  |
| --- | --- | --- |
| **Software** | **Name / Version** | **Description** |
| Operation System | Windows 7, Windows 10 | Operating system and platform for development |
| Web browser | Chrome 69 or above, Mozilla Firefox | For web app |
| Android System | Android 4.0 or higher | For mobile app |

### 2.2 Setup Files

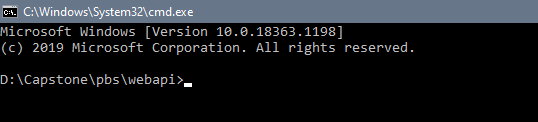
* PBS for photographer.apk
* PBS for customer.apk

### 2.3 Installation Instruction

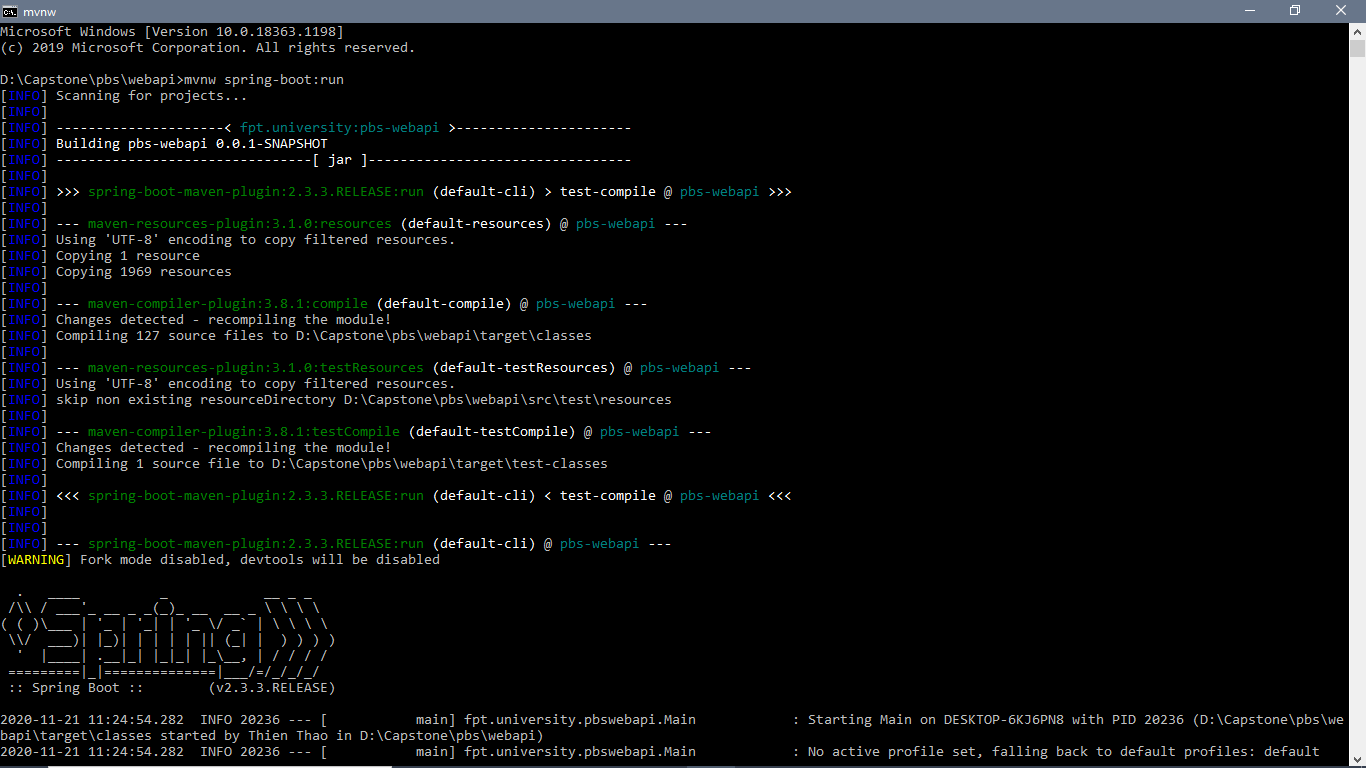
1. Open Google Play on your mobile device.
2. Press the loupe icon to search.
3. Search for “PBS”.
4. Select the PBS App from the list of found items.
5. Install PBS by pressing the “Install” button.
6. Press “Accept” to accept application permissions.
7. Wait for installation of the PBS App.
8. Wait for the initial setup.
9. Press the PBS icon to start the main App.
10. Read the manual for more information.

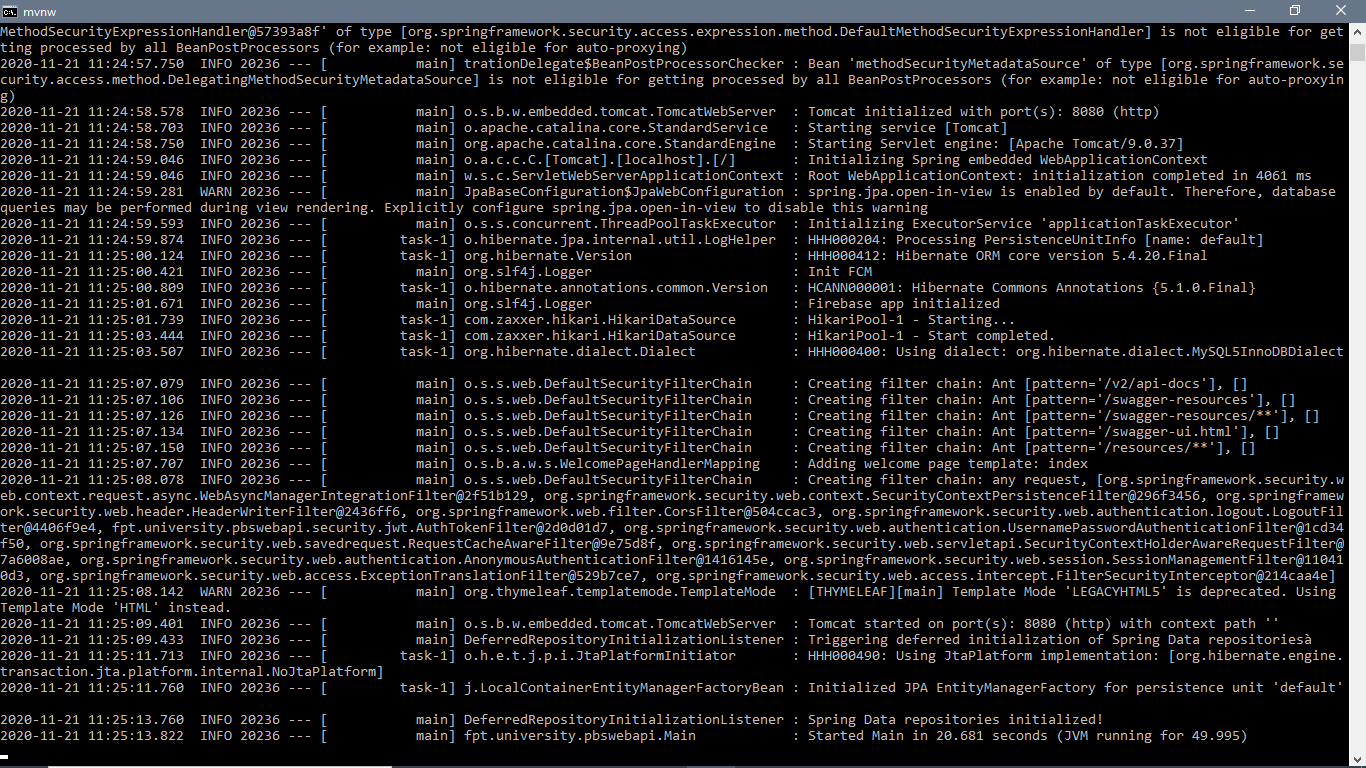
### 2.4 Server-side deployment

Step 1: Navigate to webapi folder



Step 2: Using command “mvnw spring-boot:run” to run server api





## 3. User Manual

### 3.1 Terms and definitions

* Tài Khoản có nghĩa là tài khoản đã đăng ký mà Bạn nhận được để truy cập Hệ Thống với tư cách là Người Dùng.
* Ứng Dụng Photographer có nghĩa là ứng dụng điện tử do PBS cung cấp cho các Thợ chụp ảnh để kết nối với Người Dùng
* Ứng Dụng Customer có nghĩa là ứng dụng điện tử do PBS cung cấp cho Người Dùng để kết nối với Thợ chụp ảnh.
* Thiết Bị Di Động nghĩa là điện thoại thông minh, máy tính bảng hoặc thiết bị di động khác do Bạn sở hữu hoặc kiểm soát mà Bạn sử dụng để kết nối với Ứng Dụng Customer và Ứng dụng Photographer
* Hệ Thống có nghĩa là hệ thống do PBS cung cấp để kết nối Thợ chụp ảnh với Người Dùng, bao gồm Ứng dụng Photographer, Ứng Dụng Customer và trang điện tử, nền tảng.
* Thợ chụp ảnh có nghĩa là bên cung cấp dịch vụ thứ ba độc lập cung cấp Dịch Vụ Chụp Ảnh đã được kết nối với Bạn thông qua Hệ Thống.
* Người Dùng có nghĩa là người dùng cuối đã đăng ký của Ứng Dụng Customer.
* Trang Điện Tử có nghĩa là bất kỳ trang điện tử nào do PBS điều hành.

### 3.2 System requirements

**3.2.1 Hardware requirements**

**For PC**

|  |  |  |
| --- | --- | --- |
| **PC** | **Minimum Requirements** | **Recommended** |
| Internet Connection | Cable, Wi-Fi (4 Mbps) | Cable, Wi-Fi (8 Mbps) |
| Operating System | Windows 7 | Windows 7 or more. |
| Computer Processor | Intel® Core i3 1.4GHz | Intel® Core i5 2.50GHz |
| Computer Memory | 1GB RAM | 2GB RAM or more |
| Web Browser | Chromes (v42 or higher) | Chrome latest stable version |

|  |  |
| --- | --- |
| **Operating system** | Android 4.0 or higher |
| **Processor** | Intel Atom® Processor Z2520 1.2 GHz, or faster processor |
| **Storage** | Between 850 MB and 1.2 GB, depending on the language version |
| **RAM** | Minimum of 512 MB, 2 GB is recommended |
| **Hard Disk** | * 2 GB of available hard-disk space for installation; extra free space is required during installation. * You cannot install using a removable flash storage device. |
| **Video** | 1920 x 1080 pixels or higher |
| **Software** |  |
| **Browser/Internet** | * This application is designed to work online. * To download and launch Google Play Store apps within the application, a high-speed Internet connection is recommended. |

**For Android Devices**

**3.2.2 Software requirements**

|  |  |  |
| --- | --- | --- |
| **Software** | **Name / Version** | **Description** |
| Operation System | Windows 7, Windows 10 | Operating system and platform for development |
| Web browser | Chrome 69 or above, Mozilla Firefox | For web app |
| Android System | Android 4.0 or higher | For mobile app |

### 3.3 Application Usage

#### 3.3.1. Overview

Ứng dụng đặt thợ chụp ảnh giúp cho khách hàng có thể tìm kiếm và đặt thợ chụp ảnh thông qua ứng dụng. Sau đó khách hàng có thể đánh giá thợ chụp ảnh đó với mức độ hài lòng của họ về người thợ đó sẽ được thể hiện bằng số sao họ đánh giá và những bình luận của họ sẽ được hiển thị trên trang của thợ chụp ảnh. Ngoài ra, nó giúp cho khách hàng có thể chỉnh sửa thông tin cá nhân của mình trên ứng dụng. Và để có thể tìm hiểu rõ hơn về chụp ảnh cũng như xu hướng hệ thống còn hỗ trợ diễn đàn để mọi người có thể học hỏi hoặc chia sẻ thông tin, hình ảnh của mình.

Ứng dụng đặt thợ chụp ảnh giúp cho thợ chụp ảnh có thể tìm kiếm được nhiều khách hàng tiềm năng. Ngoài ra, thợ chụp ảnh có thể quản lý lịch làm việc của mình và nhận được thông báo về lịch làm việc. Họ có thể chỉnh sửa thông tin cá nhân, thêm các album ảnh cũng như các gói dịch vụ mà họ có thể cung cấp cho khách hàng. Và họ cũng sẽ được tham gia vào diễn đàn như khách hàng.

Ứng dụng đặt thợ chụp ảnh còn cung cấp trang web dành cho admin để họ dễ dàng quản lý được hệ thống. Admin có thể thêm xóa sửa các category, thay đổi các thông số giá trị đánh giá thợ chụp ảnh,, quản lý các cách thức giao nhận ảnh. Họ có thể block các user có hành vi xấu trên app.

#### 3.3.2. Register

* **Purpose**
  + Tạo tài khỏan cho người dùng sử dụng hệ thống
* **Function**
  + Đăng kí bằng tài khoản
  + Đăng kí bằng Google
  + Đăng kí bằng Facebook
* **User Interfaces**

#### 3.3.3. Login

* **Purpose**
  + Giúp người dùng đăng nhập vào hệ thống
* **Function**
  + Đăng nhập bằng tài khoản
  + Đăng nhập bằng Google
  + Đăng nhập bằng Facebook
* **User Interfaces**

#### 3.3.4. Logout

* **Purpose**
  + Giúp người dùng đăng xuất khỏi hệ thống
* **Function**
  + Đăng xuất
* **User Interfaces**

#### 3.3.5. Forgot password

* **Purpose**
  + Giúp người dùng lấy lại mật khẩu đã quên
* **Function**
  + Đăng nhập bằng tài khoản
  + Đăng nhập bằng Google
  + Đăng nhập bằng Facebook
* **User Interfaces**

***3.3.6. Search photographer***

* **Purpose**
  + Giúp người dùng tìm thợ chụp ảnh và gói dịch vụ ưng ý
* **Function**
  + Tìm kiếm thợ chụp ảnh
  + Tìm kiếm gói dịch vụ
* **User Interfaces**

***3.3.7. Manage request***

* **Purpose**
  + Giúp người dùng quản lý yêu cầu chụp ảnh với thợ chụp ảnh
* **Function**
  + Tạo yêu cầu nhiều ngày
  + Tạo yêu cầu một ngày
  + Chỉnh sửa yêu cầu
  + Hủy yếu
  + Chấp nhận yêu cầu (chỉ dành cho photographer)
  + Từ chối yêu cầu (chỉ dành cho photographer)
* **User Interfaces**

***3.3.8. Manage review***

* **Purpose**
  + Giúp người dùng nhận xét dịch vụ chụp ảnh
  + Theo dõi nhận xét dịch vụ chụp ảnh
* **Function**
  + Xem nhận xét
  + Tạo nhận xét (chỉ customer)
* **User Interfaces**

***3.3.9. Manage status request***

* **Purpose**
  + Giúp thợ chụp ảnh thay đổi trạng thái cuộc hẹn
* **Function**
  + Thay đổi trạng thái cuộc hẹn
* **User Interfaces**

***3.3.10. Manage calendar***

* **Purpose**
  + Giúp thợ chụp ảnh thay đổi thay đổi được lịch làm việc
* **Function**
  + Theo dõi thời gian làm việc hằng ngày
  + Thay đổi thời gian làm việc hằng ngày
  + Thêm thời giời gian nghỉ đội xuất
* **User Interfaces**

***3.3.11. Manage album***

* **Purpose**
  + Giúp thợ chụp quản lý thêm album của mình
  + Giúp customer theo dõi album mình thích
* **Function**
  + Thêm album (chỉ photographer)
  + Chỉnh sửa album (chỉ photoprapher)
  + Xóa album (chỉ photogrpher)
  + Theo dõi album
* **User Interfaces**

***3.3.12. Manage service***

* **Purpose**
  + Giúp thợ chụp ảnh quản gói dịch vụ
  + Người dùng xem được gói dịch vụ
* **Function**
  + Thêm gói dịch vụ (chỉ photographer)
  + Chỉnh sửa gói dịch vụ (chỉ photographer)
  + Xóa gói dịch vụ (chỉ photographer)
  + Theo dõi gói dịch vụ
* **User Interfaces**

***3.3.13. Photographer edit service***

* **Purpose**
  + Giúp thợ chụp ảnh chỉnh sửa gói dịch vụ
* **Function**
  + Chỉnh sửa gói dịch vụ
* **User Interfaces**

***3.3.14. Manage category***

* **Purpose**
  + Giúp admin quản lý category
  + Giúp người dùng xem và chọn category
* **Function**
  + Thêm category (chỉ admin)
  + Chỉnh sửa category (chỉ admin)
  + Xóa category (chỉ admin)
  + Xem category
  + Chọn category (chỉ photographer)
* **User Interfaces**

***3.3.15. Chat***

* **Purpose**
  + Giúp người dùng và thợ chụp ảnh có thể liên lạc dễ dàng hơn với nhau
* **Function**
  + Chat
* **User Interfaces**

***3.3.16. Forum***

* **Purpose**
  + Giúp tất cả người dùng trong hệ thống có thể xem được nội dung trong forum
  + Comment trong thread
  + Quản lý thread
  + Quản lý category
* **Function**
  + Xem thread
  + Xem comment
  + Thêm thread
  + Chỉnh sửa thread
  + Xóa thread
  + Thêm comment
  + Chỉnh sửa comment
  + Xóa comment
  + Thêm category (chỉ admin)
  + Chỉnh sửa category (chỉ admin)
  + Xóa category (chỉ admin)
* **User Interfaces**

***3.3.17. Quản lý delivery method***

* **Purpose**
  + Giúp admin quản lý delivery method
  + Giúp người dùng và photographer chọn delivery method
* **Function**
  + Thêm delivery method (chỉ admin)
  + Chỉnh sửa delivery method (chỉ admin)
  + Xóa delivery method (chỉ admin)
  + Chọn delivery method (photographer và customer)
* **User Interfaces**

***3.3.18. Manage profile***

* **Purpose**
  + Giúp người dùng chỉnh sửa thông tin cá nhân
* **Function**
  + Chỉnh sửa thông tin cá nhân
  + Chỉnh sửa ảnh đại diện và ảnh bìa
* **User Interfaces**

### 3.4 Troubleshooting

1. **Đôi khi ko lấy được current location**

## First occurrence

* Trong phiên bản test

## Problem

* Không mở Vị Trí trong điện thoại

**Workaround**

Hoàn thành các bước sau:

1. Vào mục **Setting**
2. Chọn mục **Vị Trí**
3. Chọn **On**
4. **Đôi khi up ảnh xong, ảnh ko hiện**

## First occurrence

* Trong phiên bản test

## Problem

* Do dung lượng ảnh quá lớn

**Workaround**

* Up ảnh nhỏ hơn … size

# VII. Appendix

## 1. Glossary [Optional]

|  |  |
| --- | --- |
| **Acronym** | **Definition** |
| PWM | Psychology website |
| AWS | Amazon Web Services |
| BA | Business Analysis |
| 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 |
| UAT | User Acceptance Test |
| UC | Use Case |
| API | Application Program Interface |

**Table 72 - Glossary Table**

## 2. References [Optional]

1. **MySQL** [Online] Available: <https://www.mysql.com/>**.**
2. **Android Studio** [Online] Available: <https://developer.android.com/docs>.

## 3. Others [Optional]

*[Provide any other information here…]*