MINISTRY OF EDUCATION AND TRAINING



**Capstone Project Document**

**Photographic Studio Management**

**Project Code: PSM**

|  |  |  |
| --- | --- | --- |
| PSM Team | | |
| Group members | Nguyen Manh Huy | SE03199 |
| Nguyen Sy Bach | SE03704 |
| Pham Tuan Dung | SE03655 |
| Cao Lam Vu | SE03800 |
| Nguyen Hoang Trung | SE03608 |
| Supervisor | **Mr. Phan Truong Lam** | |
| Capstone Project Code | **PSM** | |

*Acknowledgements*

*We wish to express our deepest gratitude to our supervisor: Mr. Phan Truong Lam for his tutorial, sharing and motivating throught the project. Under the instructions of Mr. Phan Truong Lam, the project team has always complete at doing all assigned tasks and solving all problems. Apart from all the technologies and methodologies, Mr. Phan Truong Lam shared with us the philosophy of keeping the right attitude towards problems. We believe that was the most important factor that led to the success of this project.*

*We would also like to thank the instructors at FPT University for all the classes. We hope you will find this project as a reflection of the knowledge and experiences you have given us.*

*Finally, we truly appreciate all of our friends who has discussed with us. With your help, we have been recognized and solve every arised problem.*

Table of Contents 目次

[1 INTRODUCTION はじめに 8](#_Toc469436384)

[1.1 Purpose　目的 8](#_Toc469436385)

[1.2 Project information　プロジェクトの情報 8](#_Toc469436386)

[1.3 Project members　プロジェクトメンバー 8](#_Toc469436387)

[1.4 Our proposal　提案 9](#_Toc469436388)

[1.4.1 Problem　問題 9](#_Toc469436389)

[1.4.2 Solution　解決 9](#_Toc469436390)

[1.4.3 Idea　アイディア 9](#_Toc469436391)

[1.5 Product　商品 9](#_Toc469436392)

[1.5.1 Basic Functions　基本的な機能 9](#_Toc469436393)

[1.6 Benefit from project　プロジェクトからベネフィット 11](#_Toc469436394)

[1.6.1 For our group　グループのために 11](#_Toc469436395)

[1.6.2 For user　ユーザーのために 11](#_Toc469436396)

[2 SOFTWARE PROJECT MANAGEMENT　ソフトウェアプロジェクト管理 12](#_Toc469436397)

[2.1 Software development process　ソフトウェア開発のプロセス 12](#_Toc469436398)

[2.2 Project Organization　プロジェクト組織 13](#_Toc469436399)

[2.2.1 Organizational Structure　組織構造 13](#_Toc469436400)

[2.2.2 Project Team　プロジェクトチーム 14](#_Toc469436401)

[2.3 Project schedule　プロジェクトスケジュール 15](#_Toc469436402)

[2.4 Meeting minutes　打ち合わせ議事録 17](#_Toc469436403)

[2.5 Project Millstones　プロジェクトマイルストーン 18](#_Toc469436404)

[2.6 Project Deliverables　プロジェクトの成果物 19](#_Toc469436405)

[2.7 Resource　リソース 20](#_Toc469436406)

[2.7.1 Human resource　人的リソース 20](#_Toc469436407)

[2.7.2 Non-human resource　機器やツール 20](#_Toc469436408)

[2.8 Risk Management　リスク管理 20](#_Toc469436409)

[2.9 Communication Management　コミュニケーション管理 22](#_Toc469436410)

[2.9.1 Communication between Team Members　チームメンバーの間のコミュニケーション 22](#_Toc469436411)

[2.9.2 Communication with Supervisor　指導教員とコミュニケーション 22](#_Toc469436412)

[2.10 Configuration Management Process　コンフィグレーション管理 22](#_Toc469436413)

[2.10.1 CI Identification and Naming Convention　構成アイテムと命名規則 22](#_Toc469436414)

[2.10.2 Project Infrastructure　プロジェクトインフラストラクチャ 24](#_Toc469436415)

[2.10.3 Directory structure　ディレクトリ構造 24](#_Toc469436416)

[2.10.4 Coding convention　コーディング規約 26](#_Toc469436417)

[3 SOFTWARE REQUIREMENTS SPECIFICATIONソフトウェア要件仕様 29](#_Toc469436418)

[3.1 Introduction　はじめに 29](#_Toc469436419)

[3.1.1 Purpose　目的 29](#_Toc469436420)

[3.1.2 Scope　範囲 29](#_Toc469436421)

[3.1.3 References　参照資料 29](#_Toc469436422)

[3.1.4 Overview　概要 29](#_Toc469436423)

[3.2 Overall Description　概要説明 30](#_Toc469436424)

[3.2.1 Use Case Diagram　ユースケース図 30](#_Toc469436425)

[3.3 Product Features 製品の特徴 32](#_Toc469436426)

[3.3.1 Use Characteristics 35](#_Toc469436427)

[3.3.2 User Documentationユーザードキュメンテーション 36](#_Toc469436428)

[3.3.3 Assumption 仮定 36](#_Toc469436429)

[3.4 Entity Relation Modelエンティティ関係モデル 37](#_Toc469436430)

[3.4.1 Entity-Relationship Diagram エンティティ- 関係図 37](#_Toc469436431)

[3.4.2 Entity descriptionエンティティの説明 42](#_Toc469436432)

[3.4.3 Relationship 関係 42](#_Toc469436433)

[3.5 Specific Requirements 　 要求仕様 44](#_Toc469436434)

[3.5.1 Business Rules　ビジネスルール 44](#_Toc469436435)

[3.5.2 Functionality　機能要求 45](#_Toc469436436)

[3.6 Usability 　ユーザビリティ 132](#_Toc469436437)

[3.7 Reliability 　信頼性 132](#_Toc469436438)

[3.8 Performance　性能 133](#_Toc469436439)

[3.9 Maintainability　保全性 133](#_Toc469436440)

[3.10 Supporting Information　サポート情報 133](#_Toc469436441)

[4 ARCHITECTURE DESIGN　アーキテクチャ設計書 134](#_Toc469436442)

[4.1 Introduction　はじめに 134](#_Toc469436443)

[4.1.1 Purpose　目的 134](#_Toc469436444)

[4.1.2 References　参照資料 134](#_Toc469436445)

[4.2 Architectural Overview　アーキテクチャ概要 134](#_Toc469436446)

[4.2.1 System Architecture　システムアーキテクチャ 134](#_Toc469436447)

[4.3 System Architecture Explanation　システムアーキテクチャについての説明 135](#_Toc469436448)

[4.4 Design of Jobs Finding System　Jobs finding system の設計書 136](#_Toc469436449)

[4.4.1 Database Design データベース設計 136](#_Toc469436450)

[4.4.2 Database detail　データベースの詳細 136](#_Toc469436451)

[4.5 Component Diagramコンポーネント図 143](#_Toc469436452)

[4.6 CRC CARDS CRC カード 143](#_Toc469436453)

[4.6.1 Modelモデル 144](#_Toc469436454)

[4.6.2 Viewビュー 146](#_Toc469436455)

[4.6.3 Controllerコントローラ 151](#_Toc469436456)

[4.7 Detail Design　詳細設計 154](#_Toc469436457)

[4.7.1 Customerゲスト 154](#_Toc469436458)

[4.7.2 Managerメンバー 164](#_Toc469436459)

[4.7.3 Admin　管理 211](#_Toc469436460)

[4.7.4 System　システム 255](#_Toc469436461)

[5 SOFWARE TESTING ソフトウェアテスト 256](#_Toc469436462)

[5.1 Introduction　はじめに 256](#_Toc469436463)

[5.1.1 Purpose　目的 256](#_Toc469436464)

[5.1.2 References　参照資料 256](#_Toc469436465)

[5.1.3 Background information　背景情報 256](#_Toc469436466)

[5.1.4 Scope of testing　テスト範囲 257](#_Toc469436467)

[5.2 Constraints　制約事項 257](#_Toc469436468)

[5.2.1 Risk listリスク一覧 257](#_Toc469436469)

[5.3 Requirements for Test　テスト要求 258](#_Toc469436470)

[5.3.1 Test items　テスト頭目 258](#_Toc469436471)

[5.3.2 Acceptance Test Criteria　テスト受入基準 261](#_Toc469436472)

[5.3.3 Feature not to be tested　テストされない特徴 262](#_Toc469436473)

[5.4 Test strategy　テスト戦略 262](#_Toc469436474)

[5.4.1 Test Model　テストモデル 262](#_Toc469436475)

[5.4.2 Test types テストタイプ 264](#_Toc469436476)

[5.4.3 Function Testing　機能テスト 264](#_Toc469436477)

[5.4.4 User Interface Testing　ユーザーインターフェーステスト 265](#_Toc469436478)

[5.4.5 Data and Database Integrity Testing　でーたおよびデータベース整合性テスト 266](#_Toc469436479)

[5.4.6 Test stages　テストステージ 266](#_Toc469436480)

[5.5 Resource　リソース 267](#_Toc469436481)

[5.5.1 Human Resource　人的リソース 267](#_Toc469436482)

[5.5.2 Test management　テスト管理 267](#_Toc469436483)

[5.5.3 Defect management　欠陥管理 268](#_Toc469436484)

[5.6 Test environment　たテスト環境 269](#_Toc469436485)

[5.6.1 Hardware ハードウェア 269](#_Toc469436486)

[5.6.2 Software　ソフトウェア 269](#_Toc469436487)

[5.7 Test Milestones　テストマイルストーン 270](#_Toc469436488)

[5.8 Deliverables　納品物 271](#_Toc469436489)

[5.9 Test Case　テストケース 271](#_Toc469436490)

[5.10 Test Report　テストレポート 272](#_Toc469436491)

[5.10.1 Unit Test Report 単体テスト 272](#_Toc469436492)

[5.10.2 Common Test Report　一般テストレポート 273](#_Toc469436493)

[5.10.3 Integration Test Report　統合テストレポート 273](#_Toc469436494)

[5.10.4 System Test Report　システムテストレポート 273](#_Toc469436495)

[5.10.5 Test Report　テストレポート 274](#_Toc469436496)

[6 USER MANUAL　ユーザーマニュアル 274](#_Toc469436497)

[6.1 Install Guide　インストールガイド 274](#_Toc469436498)

[6.1.1 Introduction　はじめに 274](#_Toc469436499)

[6.1.2 Installation Guideline　インストールのガイドライン 275](#_Toc469436500)

[6.1.3 Other considerations 他の検討事項 279](#_Toc469436501)

[6.2 User Guide　ユーザーガイド 279](#_Toc469436502)

[6.2.1 Customerゲスト 279](#_Toc469436503)

[6.2.2 Managerメンバー 284](#_Toc469436504)

[6.2.3 Admin　管理 314](#_Toc469436505)

[6.2.4 System　システム 338](#_Toc469436506)

Definition and acronyms

|  |  |  |
| --- | --- | --- |
| Acronym | Definition | Note |
| PSM | Website Studio Media | Project’s name |
| SRS | Software Requirement Specifications |  |
| CI | Configuration Item |  |
| CM | Configuration Management |  |
| Admin | Administrator |  |
| Manager | Photo Studio Manager |  |
| PM | Project Manager |  |
| PTL | Project Technical Leader |  |
| CRC | Class-Responsibility-Collaborators |  |
| AT | Acceptance Test |  |
| IT | Integration Test |  |
| KLOC | 1000 lines of code |  |
| PM | Project Manager |  |
| PTL | Project Technical Leader |  |
| PT/TT | Program Test/ Total Test |  |
| QA | Quality Assurance |  |
| UT | Unit Test |  |
| PC | Personal Computer |  |
| UI | User Interface |  |

# INTRODUCTION はじめに

## Purpose　目的

This chapter provides an overview of Website Studio Media system, it includes background information, literature review of existing system and raising a proposal for ideas of improvement.

## Project information　プロジェクトの情報

|  |  |
| --- | --- |
| Project name: | Website Studio Media |
| Project code: | PSM |
| Project type: | Website |
| Project category: | New development |
| Business type: | Website Service |
| Project manager: | Nguyen Manh Huy |
| Timeline: | *Work in progress* |

Table 1.1 - Project Information

## Project members　プロジェクトメンバー

* Supervisor:

|  |  |  |
| --- | --- | --- |
| Full name | E-mail | Title |
| Phan Truong Lam | Lampt2@fe.edu.vn | Lecturer |

* Team member:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Full name | Roll number | E-mail | Phone | Position |
| Nguyen Manh Huy | SE03199 | HuynmSE03199@fpt.edu.vn | 01689597078 | Leader |
| Nguyen Sy Bach | SE03704 | BachnsSE03704@fpt.edu.vn | 01228245737 | Member |
| Cao Lam Vu | SE03800 | VuclSE03800@fpt.edu.vn | 01689181825 | Member |
| Pham Tuan Dung | SE03655 | DungptSE03655@fpt.edu.vn | 01646426345 | Member |
| Nguyen Hoang Trung | SE03608 | TrungnhSE03608@fpt.edu.vn | 01635105330 | Member |

## Our proposal　提案

### Problem　問題

The rapid change in technology, knowledge explosion, the increased and complex demands for a good place with the best price to make their own memory with an album book full of picture of important moments in their life. Moreover, there are so much information about unknow photo studio on the internet, customer find it difficult to find their ideal photo studio because of the number and confusing information. Therefore, the posed problems are making an interaction between customer and photo studio, and help the customer to search for the best studio as quick as possible.

### Solution　解決

Create a website that have all the information about a coporation of photo studio to help the customer find the best one.

Provide a lot of service to help the customer to keep track of the photo contract and printing detail.

By the automatic searching function, cost and time are minimized.

### Idea　アイディア

Our idea is creating a website not only to link Customer and Photo Studio but also to help customer manage information about the photo conctract and printing requirement quickly and efficiently.

## Product　商品

After researching carefully about the market needs of photo conctract, we have chosen main functions which are necessary and important to our system. They will help the manager easy to use and operate. Besides, we also compare to other website and research about user’s habits to design good and intuitive interface. We want to reduce all limitation and improve our own advantages.

The objective user of our Website is Vietnamese people who living in VietNam. Therefore, our website using Vietnamese as primary language, to meet the criteria of user-friendly.

### Basic Functions　基本的な機能

**Customer:**

* Search for the nearest studio location.
* View studio basic informations and sample portfolio.
* See and pick a photo session contract package.
* Tracking the contract.
* See photo uploaded after photo shoot.
* Pick and choose size, material for the photo to print.

**Photo Studio Manager:**

**Photo Studio Manager can only access to these function after login.**

* View own photo studio information.
* Edit photo studio information and sample portfolio.
* Add photo studio information and sample portfolio.
* Delete photo studio information and sample portfolio.
* Search photo session package.
* View photo session package
* Edit photo session package.
* Add photo session package.
* Delete photo session package.
* See customer contract package register.
* Search customer contract package.
* Accept customer contract and add to contract list.
* Edit a contract.
* Upload photos after photo shoot.
* Accept printing order if customer request.
* Search priting size and material.
* View priting size and material.
* Edit priting size and material.
* Add priting size and material.
* Delete priting size and material

**Admin**

* **Manage Photo Studio Manager**
* Search Manager profile.
* View Manager profile
* Edit a Manager profile.
* Add a Manager.
* Delete a Manager.
* **Manage Province and District**
* Search Province and District information.
* View Province and District information.
* Edit Province and District information.
* Add a Province or District.
* Delete a Province or District.

**System**

* Send notification to Customer’s email in the following cases:
* Cutomer leave email information for photo studio to contact.
* Customer order a photo sessions package.
* The Studio Manager had accepted the customer’s order and awaiting the customer confirm.
* The Photo Taking Session is done and Photo Studio send customer a receipt

Out of scope

* **Manage Website’s payment activities**

By using website, when order a photo session or photo extra priting require customer to pay. Therefore, website will use online payment service .

## Benefit from project　プロジェクトからベネフィット

### For our group　グループのために

After developing and implementing this project, our group will get some benefits:

* Have more experiences of software project management: how to manage plan, time, member and risk.
* Have more knowledge and skill about Node.js, and HTML, MySQL.
* Know how to communicate with team members and how to make teamwork more effective.

### For user　ユーザーのために

* Improving chances to finding the best Photo Studio.
* Saving time and costs for Studio.
* Customer can track Photo Contract and order extra printing.

# SOFTWARE PROJECT MANAGEMENT　ソフトウェアプロジェクト管理

In this section, we will describe our organization and plan of project. Each member of team must use this section as a guideline tracking assigned tasks and deadlines. In addition, we introduce our team and overview about project. Meeting minute also present in this section. Moreover, we will describe risks of project and the way communication of our team.

## Software development process　ソフトウェア開発のプロセス

This figure below will describe the information and products waterfall process model.



Figure 2.1Waterfall software process model

PSM project uses the Waterfall Software Process Model. We have chosen the Waterfall Software Process Model because the scope of our project is small, the technical documentation is a necessary part of the initial requirements pahse, this mean that everyone understands the objectives. By using this software process model, the phased development cycle enforces discipline. Each step has a clearly defined starting poin and conclusion, which makes progress easy to monitor. This model make testing easy, test scenarios are already detailed in the function specification phased. We easier to manage risk because risky pieces are identified and handled. Besides, the iterative model is easier than other models when the issues are discovered. All issue will be raised to the team. They will be solved while the project is still developed.

## Project Organization　プロジェクト組織

### 2.2.1 Organizational Structure　組織構造

Figure 2.2.1Organizational structure chart

### 2.2.2 Project Team　プロジェクトチーム

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Role | Responsibility | Full name | % Effort | Start date | End date |
| Instructor | * Provide templates & tools * Review deliverables * Review project status * Resolve escalated issues * Consultant | Pham Truong Lam |  |  |  |
| PM | * Planning and defining scope * Developing schedule * Reviewing and Tracking document * Identify resources and assign individual responsibilities * Assign task to team members * Coordinating communication | Nguyen Manh Huy |  |  |  |
| Analysis and Design Team | | | | | |
| Team Leader | * Reviewing and Tracking design * Identify main UI * Draw prototype, Architecture Design * Database Design | Cao Lam Vu |  |  |  |
| Team member #1 | * Database Design * Draw prototype, Architecture Design | Nguyen Hoang Trung |  |  |  |
| Team member #2 | * Database Design * Draw prototype, Architecture Design | Nguyen Sy Bach |  |  |  |
| Team member #3 | * Screen Design | Pham Tuan Dung |  |  |  |
| Team member #4 | * Screen Design | Pham Tuan Dung |  |  |  |
| Documentation Team | | | | | |
| Team Leader | * Reviewing and Tracking Requirement | Nguyen Sy Bach |  |  |  |
| Team member #1 | * Create SRS document | Nguyen Sy Bach |  |  |  |
| Team member #2 | * Create SRS document | Nguyen Sy Bach |  |  |  |
| Development Team | | | | | |
| PTL | * Developing * Solve issue of project | Nguyen Manh Huy |  |  |  |
| Developer #1 | * Developing | Cao Lam Vu |  |  |  |
| Developer #2 | * Developing | Nguyen Hoang Trung |  |  |  |
| QA and Test Team | | | | | |
| Test Leader | * Responsible for test execution * Test results recording * Evaluation of test run and error recovery | Pham Tuan Dung |  |  |  |
| Tester #1 | * Testing | Pham Tuan Dung |  |  |  |
| Tester #2 | * Testing | Nguyen Sy Bach |  |  |  |

Table 2.2.2 Project Team

## Project schedule　プロジェクトスケジュール

## *ÔNG ADD CÁI HÌNH SCHEDULE ÔNG LÀM VÀO ĐÂY.*

## Meeting minutes　打ち合わせ議事録

All meeting minutes will be written follow this template:

|  |  |  |  |
| --- | --- | --- | --- |
| Meeting Minutes 1 | | | |
| Subject |  | **Date** |  |
| Conductor |  | **Time** |  |
| Location |  | **Secretary** |  |
| Attendees |  | | |
| Absent |  | | |

|  |  |  |
| --- | --- | --- |
| Key Points Discussed | | |
| **No.** | Topic | Highlights |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |
| 5 |  |  |
| 6 |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Action Plan | | | |
| No. | Action Item(s) | Owner | Target Date |
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |
| 4 |  |  |  |
| 5 |  |  |  |

Table 2.4 Meeting minutes

## Project Millstones　プロジェクトマイルストーン

|  |  |  |  |
| --- | --- | --- | --- |
| No | Milestone | Completion Date | Verification |
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |
| 4 |  |  |  |
| 5 |  |  |  |
| 6 |  |  |  |
| 7 |  |  |  |
| 9 |  |  |  |

Table 2.5 Project Millstones

## Project Deliverables　プロジェクトの成果物

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Deliverable** | **Committed Delivery date** | **Description of Deliverable** | **Delivery media** |
| 1 |  |  |  |  |
| **Iteration 1** | | | | |
| 2 |  |  |  |  |
| 3 |  |  |  |  |
| 4 |  |  |  |  |
| 5 |  |  |  |  |
| 6 |  |  |  |  |
| 7 |  |  |  |  |
| 8 |  |  |  |  |
| 9 |  |  |  |  |
| 10 |  |  |  |  |
| 11 |  |  |  |  |
| 12 |  |  |  |  |
| 13 |  |  |  |  |
| 14 |  |  |  |  |
| 15 |  |  |  |  |
| 16 |  |  |  |  |
| 17 |  |  |  |  |
| **Iteration 2** | | | | |
| 18 |  |  |  |  |
| 19 |  |  |  |  |
| 20 |  |  |  |  |

Table 2.6 Project Deliverables

## Resource　リソース

### Human resource　人的リソース

* Team members
* Supervisor

### Non-human resource　機器やツール

* Equipment: Desktop, Laptop.
* Building: ALFA & BETA Building - FPT University.

## Risk Management　リスク管理

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Name | Root Cause | Probability | Avoidance plan | Contingency plan | Impact |
| 1 | Team don’t keep the deadline on time. | Members lack of responsibility, have bad habits.  Lack of management. | Medium | Make rules and penalties for people who doesn’t meet the deadline. | Working overtime to complete on time  Punish member by money... | High |
| 2 | Requirement changed while project is processing. | SRS is not provide all information about user requirement such as: lack of customer’s requirement, misunderstand requirement of customer. | High | SRS should be review carefully by PM and the Supervisor.  Always do brainstorming carefully to design detail system. Hold all meeting and make minimize effort to change and improve design by requirement changes. | Team members will have meetings with supervisor to analyze the requirement change, and make a specific action to resolve that problem. | High |
| 3 | Delivery does not meet the deadline. | Team member work late, lack of responsible causing not meet project schedule. | High | Assign tasks to suitable team members depending on their strengths and weakness. | Team leader should regularly care about deadline and project schedule to inform members.  Break down task to smaller tasks and assign new tasks to team members. | Medium |
| 4 | Lack of knowledge about technology needs for project. | With some member, they want to study new technology through this project. | Medium | Each team member must study to understand the technology, framework, which is needed for project. | Technical leader need support carefully for team members in group to develop required skills and knowledge.  Review complete task to make comment for team member. | Medium |
| 5 | Conflict among team members. | Team member doesn't understand each other. | Medium | Plan some games or team building activities to improve teammate’s relationship and to understand each other.  Clear role and responsibility between members. | Have a meeting to resolve conflict problems. | High |
| 6 | Data or Source Code Lost. | Delete Brand in GitHub.  Delete wrong source code. | Medium | Back up source code carefully. Commit all small tasks after complete to GitHub. | Restore data from backed up data.  Restore source code from history versions. | High |

Table 2.8 Risk Management

## Communication Management　コミュニケーション管理

### Communication between Team Members　チームメンバーの間のコミュニケーション

* **Face-to-face meeting:** at least twice a week at Monday and Thursday. This is the fastest way of communication to solve big problems in the project and members can help out each other easily.
* **E-mail and message:** Email, Skype and Facebook are used for members to keep tracking other team member’s progress and team members can also help each other online.
* **Mobile Phone:** is using for emergency to directly contact to others.
* **Collaboration tool:** GitHub is used for document and source code management.

### Communication with Supervisor　指導教員とコミュニケーション

* **Face-to-face meeting:** Weekly on every Thursday afternoons to make sure that supervisor can keep tracking of the team’s progress.
* **E-mail:** Gmail is the fastest way to get advice and document checking from supervisor.
* **Mobile phone:** is used to get time and place arranged for the meeting every weeks.

## Configuration Management Process　コンフィグレーション管理

### CI Identification and Naming Convention　構成アイテムと命名規則

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Deliverable | Committed Delivery date | Description of Deliverable | Delivery media |
| 1 |  |  |  |  |
| Iteration 1 |
| 2 |  |  |  |  |
| 3 |  |  |  |  |
| 4 |  |  |  |  |
| 5 |  |  |  |  |
| 6 |  |  |  |  |
| 7 |  |  |  |  |
| 8 |  |  |  |  |
| 9 |  |  |  |  |
| 10 |  |  |  |  |
| 11 |  |  |  |  |
| 12 |  |  |  |  |
| 13 |  |  |  |  |
| 14 |  |  |  |  |
| 15 |  |  |  |  |
| 16 |  |  |  |  |
| 17 |  |  |  |  |
| Iteration 2 |
| 18 |  |  |  |  |
| 19 |  |  |  |  |
| 20 |  |  |  |  |

Table 2.10.1 Naming Convention

### Project Infrastructure　プロジェクトインフラストラクチャ

Below is the list of tools and infrastructure requirements needed for development environment.

#### Software　ソフトウェア

|  |  |
| --- | --- |
| Programming languages | **Node.js , HTML, CSS, Javascript** |
| Framework |  |
| Version Control | **Github** |
| IDEs |  |
| DBMS | **MySQL v5.7, MySQL Workbench v6.3.7** |
| Deployment tools |  |
| Communication tools | **Trello, Skype, Facebook, Phone, Email.** |
| UML tools |  |
| Management tools | **Microsoft Project Professional 2013** |
| Web server |  |
| Other | **Microsoft Office 2013** |

Table 2.10.2 Tools

#### Hardware　ハードウェア

* Personal computer for developing and testing with the configuration: 16GB RAM, 1TB of hard disk, Intel Core i7 Processor.
* Other computers and phone for testing.
* Internet network connection with minimum speech 512KByte/s.

#### Other infrastructure　その他のインフラストラクチャ

* Rooms for meeting and working.
* Internet and mobile phone services for communication.

### Directory structure　ディレクトリ構造

|  |  |  |
| --- | --- | --- |
| Main folder | Sub-folder | Purpose |
| Documents |  |  |
|  |  |

### Coding convention　コーディング規約

The following rules follow the standard rule of developing applications using PHP. It refers to the coding convention on websites:

# SOFTWARE REQUIREMENTS SPECIFICATIONソフトウェア要件仕様

## Introduction　はじめに

### Purpose　目的

This document is created as the introduction for project JFS – our Capstone Project at FPT University. In this document, we provide the describing of general requirement and non-functional requirements. These requirements will assure that the system will correctly and reliably perform its intended functionality. This specification will provide general, as well as specific requirements to be used in the design, testing and validation of the system. It is intended for both stakeholders and the project team of the system.

### Scope　範囲

PSM is a system that build on web platform to create the connect between Photo Studio Manager and Customers by provide all necessary information that all Managers and Customers need, and help them to find each other in a fastest way. To do this, we build our system a special feature to help Manager and Customers keep track od their contract.

### References　参照資料

|  |  |  |
| --- | --- | --- |
| No. | Document | Source/Web Address |
| 1 | System Requirements Specification Template | FPT Software Cop. |

### Overview　概要

The rest of this document contains 4 sections:

* **Introduction**: provides an overview of this document. It includes the purpose, scope, definitions, acronyms, abbreviations, references, and overview.
* **Overall Description**: describes the general factors that affect to the product and their requirements.
* **Specific Requirement**: Contains all requirements to a level of detail sufficient to help designers and testers satisfy those requirements.
* **Supporting Information**: Provide some other reference information that can help people understand easier.

## Overall Description　概要説明

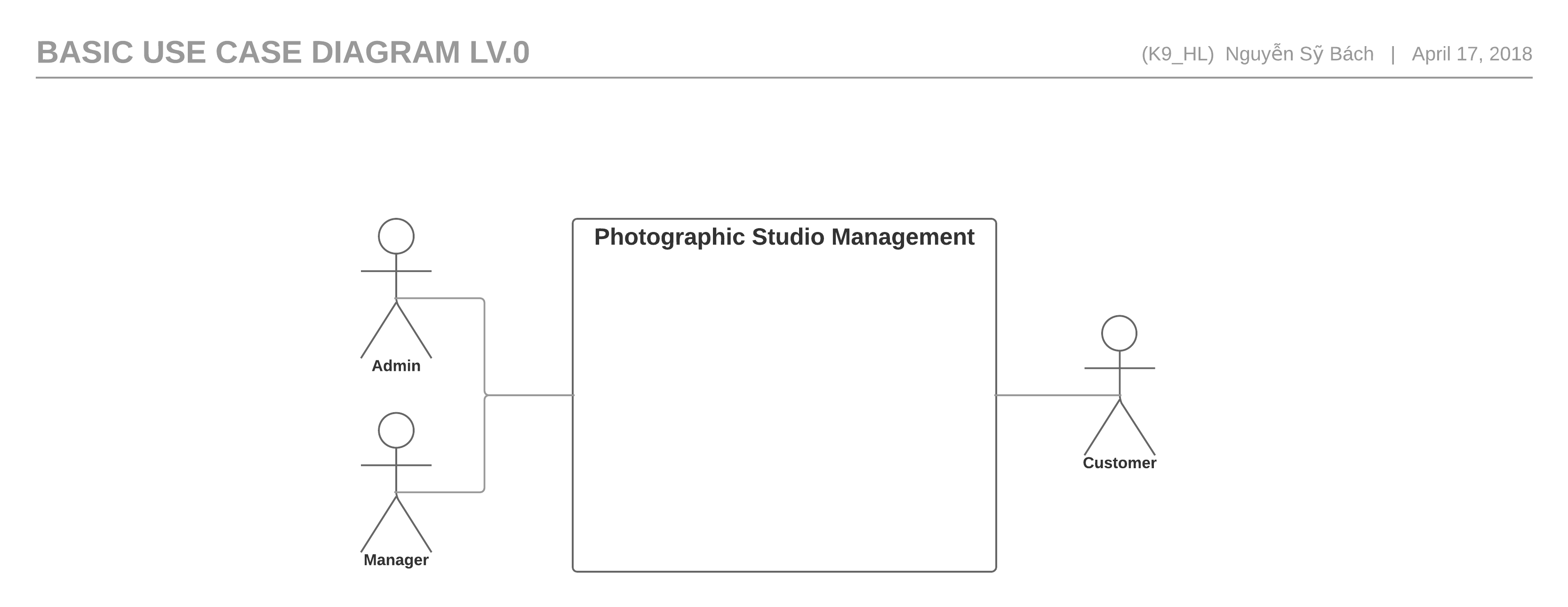


Figure 3.2.1 :Use Case of PSM Lv.0

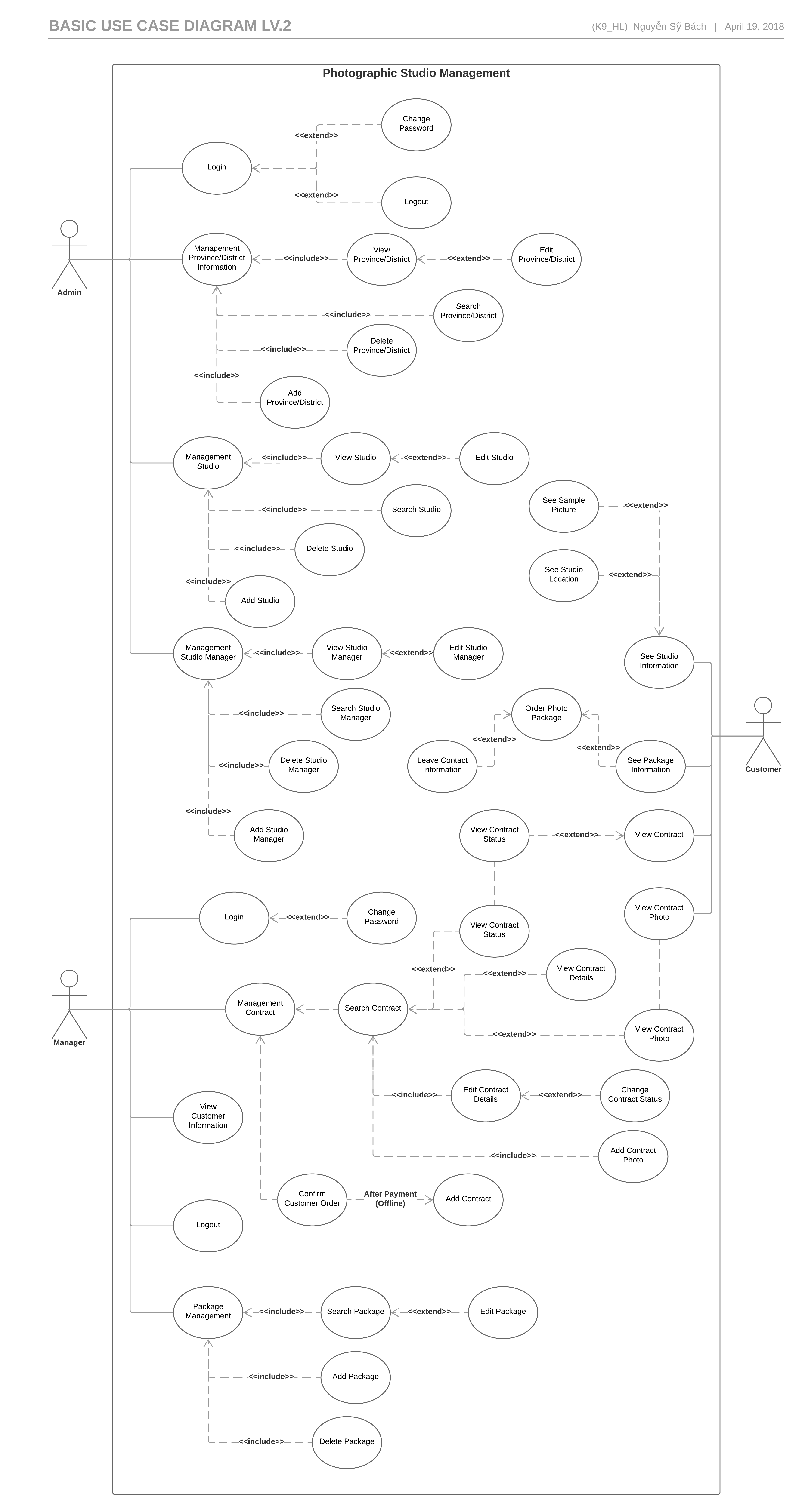


Figure 3.2.1: List Of Use Case of PSM

## Entity Relation Modelエンティティ関係モデル

### Entity-Relationship Diagram エンティティ- 関係図

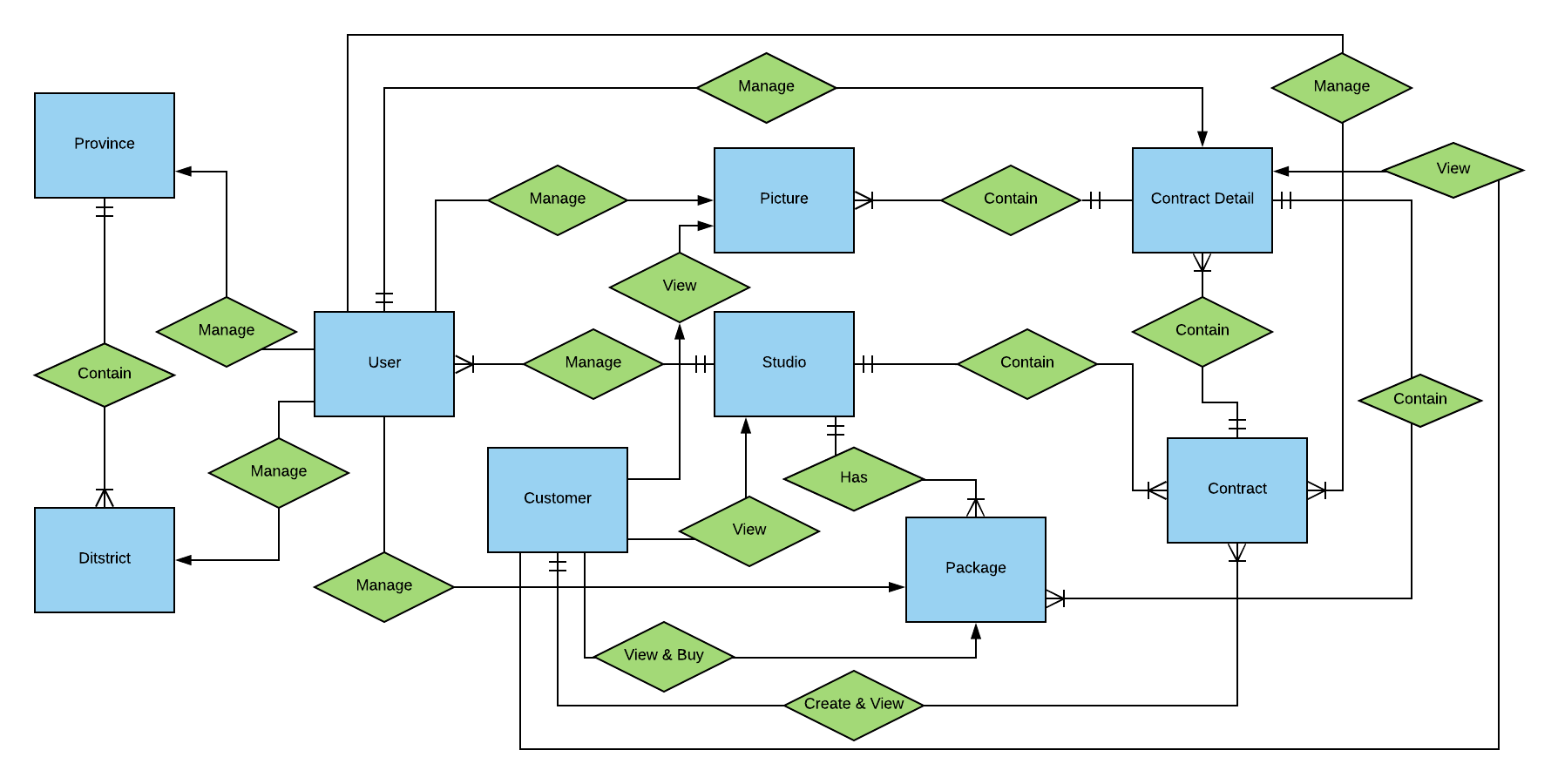


Figure 3.4.1**:** Entity Relationship Diagram

#### User(Manager)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Attribute** | **Type** | **Mandatory** | **Description** |
| 1 | User\_ID | String | Y | User’s username |
| 2 | Studio\_ID | int | Y | Studio’s ID |
| 3 | User\_Role | String | Y | User’s role |
| 4 | User\_Password | String | Y | User’s password |

Table 3.4.1.1**:** User(Manager)

#### Studio

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Attribute** | **Type** | **Mandatory** | **Description** |
| 1 | Studio\_ID | Int | Y | Studio’s ID |
| 2 | District\_ID | String | Y | District’s ID. |
| 3 | Studio\_Name | String | Y | Studio’s name |
| 4 | Studio\_Address | String | Y | Studio’s physical address |
| 5 | Studio\_Email | String | Y | Studio’s email address |
| 6 | Studio\_Number | Number | Y | Studio’s phone number |
| 7 | Studio\_Coordinate | String | Y | Studio’s location coordinate |
| 8 | Studio\_Icon | String | Y | Studio’s icon |

Table 3.4.1.2**:** Studio

#### District

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Attribute | Type | Mandatory | Description |
| 1 | District\_ID | Int | Y | District’s ID |
| 2 | Province\_ID | Int | Y | Province that have the current district |
| 3 | District\_Name | String | Y | District’s Name |

Table 3.4.1.3**:** District

#### Province

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Attribute | Type | Mandatory | Description |
| 1 | Province\_ID | Int | Y | Province’s ID |
| 2 | Province\_Name | String | Y | Province’s Name |

Table 3.4.1.4: Provice

#### Contract

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Attribute | Type | Mandatory | Description |
|  | Contract\_ID | String | Y | Contract’s ID |
|  | Customer\_ID | Int | Y | Customer’s ID |
|  | Studio\_ID | Int | Y | Studio’s ID |
|  | Contract\_Description | String | N | The Description about the current contract |
|  | Contract\_cDate | Date | Y | The create date of the contract |
|  | Contract\_sDate | Date | N | The start the contract start have effectness |
|  | Contract\_eDate | Date | N | The date the contract end lost effectness |

Table 3.4.1.5**:** Contract

#### Customer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Attribute | Type | Mandatory | Description |
| 1 | Customer\_ID | Int | Y | Customer’s ID |
| 2 | Customer\_Name | String | Y | Customer’s name |
| 3 | Customer\_Gender | String | Y | Customer’s gender |
| 4 | Customer\_Address | String | Y | Customer’s address |
| 5 | Customer\_Email | String | Y | Customer’s email |
| 6 | Customer\_Number | String | Y | Customer’s number |
| 7 | Customer\_Note | String | N | Customer’s note |
| 8 | Customer\_Other | String | N | Customer’s social media |

Table 3.4.1.6**:** Customer

#### Contract Detail

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Attribute | Type | Mandatory | Description |
| 1 | ConDetail\_ID | Int | Y | Feedback’s ID |
| 2 | Contract\_ID | String | Y | Contract’s ID that relate to the contract’s detail |
| 3 | Package\_ID | Int | Y | Real time that member create feedback |
| 4 | Package\_Name | String | Y | Name of the current available package |
| 5 | Package\_Detail | String | Y | Detail of the current available package |
| 6 | Package\_Price | Int | Y | Price of the current available package |
| 7 | Package\_Note | String | N | Note about package |

Table 3.4.1.7**:** Contract Detail

#### Package

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Attribute | Type | Mandatory | Description |
|  | Package\_ID | Int | Y | package’s ID |
|  | Package\_Name | String | Y | Package’s Name |
|  | Package\_Detail | String | Y | Package’s Detail |
|  | Package\_Price | Int | Y | Package’s price |
|  | Package\_Available | Boolean | Y | Define that the package is available or not |

Table 3.4.1.8**:** Package

#### Picture

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Attribute** | **Type** | **Mandatory** | **Description** |
| 1 | Picture\_ID | Int | Y | Picrure’s ID |
| 2 | ConDetail\_ID | Int | Y | Comtract’s ID |
| 3 | Picture\_Detail | String | N | Picture’s detail |
| 4 | Picture\_Url | String | N | Picture’s Url link |

Table 3.4.1.10**:** Picture

### Entity descriptionエンティティの説明

|  |  |  |
| --- | --- | --- |
| No | Table Name | Description |
|  | User | Account of member of Studio. Manage Users, Studios, Provinces, Districts, Package, Contract, Contract Detail, Picture, Customer depend on role of the User. |
|  | Picture | The Picture of the current Contract Detail. |
|  | Province | The information of the Province. |
|  | District | The information of the District where it belong in a Province. |
|  | Package | The information of the Contract . |
|  | Studio | The information of the Studio. |
|  | Contract | A contract of a Studio where Customer order Packages. |
|  | Contract Detail | Details of the current Contract that contain the information of the Packages that the Customer order. |
|  | Customer | Customer who buy Packages. |
|  | User | Account of member of Studio. Manage Users, Studios, Provinces, Districts, Package, Contract, Contract Detail, Picture, Customer depend on role of the User. |
|  | Picture | The Picture of the current Contract Detail. |
|  | Province | The information of the Province. |
|  | District | The information of the District where it belong in a Province. |

Table 3.4.2: Entity Description

### Relationship 関係

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Relation | Properties | From | To | Description |
|  | CONTAIN | Province\_ID | Province | District | A Province has one or many District. |
|  | MANAGE | District\_ID | User | District | User role Admin manages all district at District management page. |
|  | MANAGE | Province\_ID | User | Province | User role Admin manages all Province at Province management page. |
|  | MANAGE | Package\_ID | User | Package | User role manager manages all packages of the User’s current Studio at Package management page. |
|  | MANAGE | Contract\_ID | User | Contract | User role manager manages all their contracts of the user’s current Studio at Contract management page. |
|  | MANAGE | ConDetail\_ID | User | Contract Detail | User role manager manages all thier contract’s details of it current contract at Contract management page. |
|  | MANAGE | Picture\_ID | User | Picture | User role manager manages all pictures of the current contract’s detail page at view picture page. |
|  | MANAGE | Studio\_ID | User | Studio | User role Admin manages all Studios at Studio Management page. User role manager manage their own Studio at Profile management page. |
|  | MANAGE | Customer\_ID | User | Customer | User role manager can manages their own Customer at Contract management page. |
|  | VIEW | None | Customer | Picture | A customer can view picture of the contract they order at Contract view page. |
|  | VIEW | None | Customer | Contract Detail | A customer can view contract’s detail at Contract view page. |
|  | VIEW | None | Customer | Studio | A customer can view Studio information and Packages’s information inside the Studio view page. |
|  | VIEW & CREATE | Contract\_ID | Customer | Contract | A customer can view contract’s information at Contract view page. Customer can also create new Contract by buying one or many packages at checkout page |
|  | VIEW & BUY | None | Customer | Package | A customer can buy packages by add to cart at the Studio view page. Customer can also view them at Studio view page |
|  | HAS | Package\_ID | Studio | Package | A studio can have one or many Packages. |
|  | CONTAIN | Studio\_ID | Studio | Contract | A studio can have one or many Contracts. |

Table 3.4.3 Relationship

## Specific Requirements 　 要求仕様

### Business Rules　ビジネスルール

|  |  |
| --- | --- |
| No | Description |
|  | Search textbox cannot be empty |
|  | Text field cannot be empty |
|  | Content of post is limited to 1000 characters |
|  | Chat box cannot be empty |
|  | If user accesses to nonexistent page, PSM displays error message “Page not found” |
|  | Email must characters ‘@’ and ‘.’content |
|  | Date of birth must be chosen |
|  | Displayed date is mm/dd/yyyy format |
|  | Date time input format is date time |
|  | Number is rounded to 2 decimal number |
|  | Unit of money is VND |
|  | Displayed money is rounded to 2 decimal number |
|  | Allowed file extension: .jpg, .png, .bmp |
|  | File size is limited to 100 MB |
|  | Message displayed like a Gmail format |

Table 3.5: Business Rules

### Actor

|  |  |
| --- | --- |
| Actor | Description |
| Admin | Manage all Studio and Studio Manager. |
| Manager | Manage work of one Studio include package, photo and contract with customer. Need an account from Admin. |
| Customer | People who browsing the web and order photo package. |

Table 3.5.1: Actor

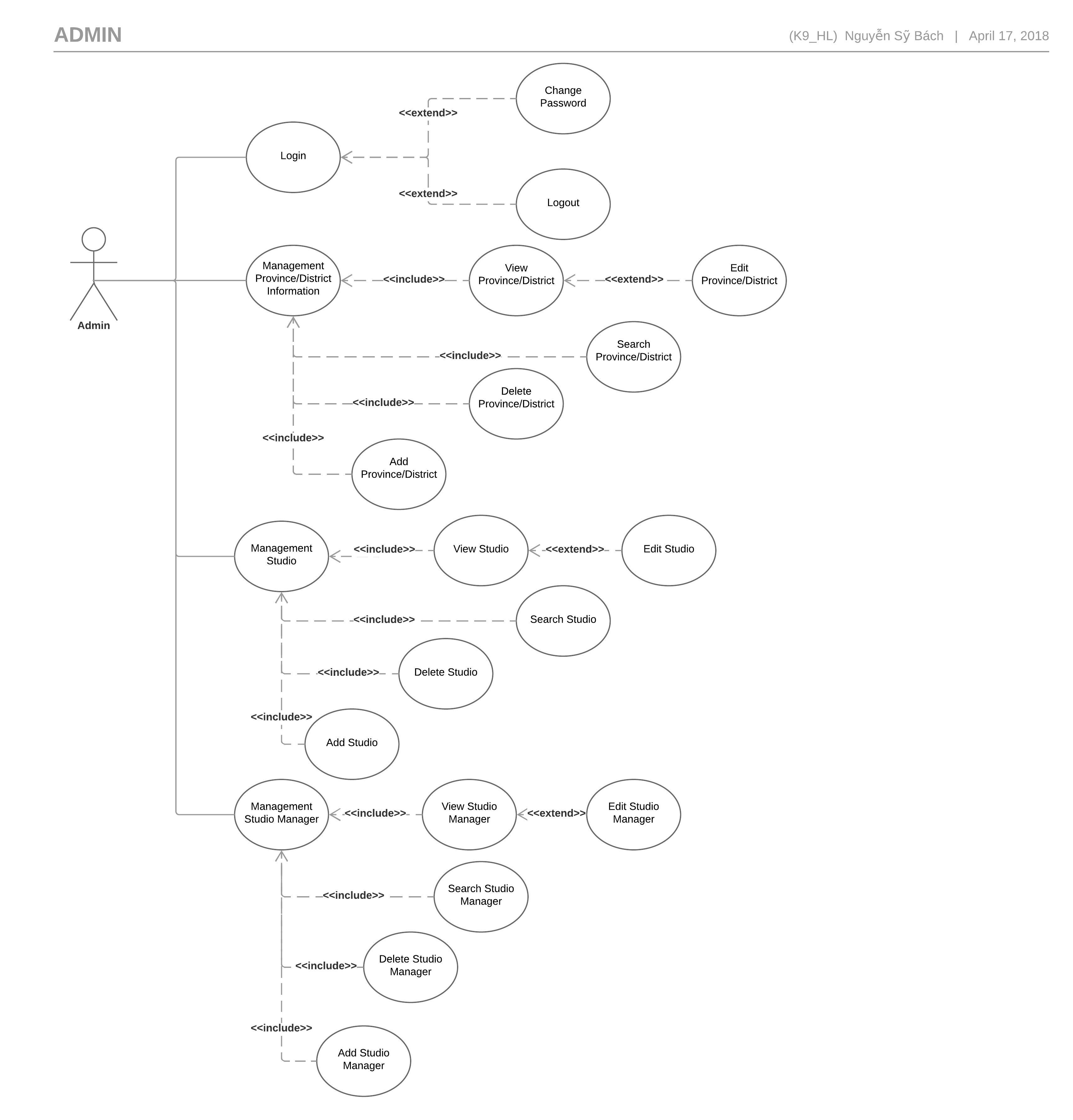
### Use Case List

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Primary Actor | Secondary Actor | Name |
| UC-001 |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table 3.5.1: Use Case List

### Functionality　機能要求

#### Admin

Figure 3.5.2**:** Admin Use Case Diagram

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID UC-001 | | Version | 1.0 | |
| Use Case Name | Login PSM | | | |
| Created by: | HuyNM | | **Date Created:** | 12/04/2018 |
| Primary Actor | Admin | | **Secondary Actor** |  |
| Description: | Login to website PSM. | | | |
| Preconditions: | Go website PSM  Already have account Admin. | | | |
| Trigger: | Click to “Đăng Nhập” hyper link. | | | |
| Post conditions: | Show page of Admin. | | | |
| Normal Flow | | | | |
| Step | **Actor’ event** | | **System response** | |
| 1 | From the website header, clicks “Đăng Nhập”. | |  | |
| 2 |  | | Loads the Login page | |
| 3 | Enter account in textbox “Tài khoản” and “Mật khẩu”then click “Đăng nhập”. | |  | |
| 4 |  | | Popup “Đăng nhập thành công” appear. | |
| Alternative Flows: N/A | | | | |
|  | | | | |
| Exceptions 1: At step 3 of normal flow, user leaves email and/or password blank, then proceeds to step 4.  Exceptions 2: At step 3 of normal flow, user enters invalid login credentials, then proceeds to step 4. | | | | |
| Priority | | Normal | | |
| Frequency of Use: | | Normal | | |
| Business Rules: | | B2,B17 | | |
| Other Information: | | N/A | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID UC-002 | | Version |  | 1.0 |
| Use Case Name | Logout PSM | | |  |
| Created by: | HuyNM | | **Date Created:** |  |
| Primary Actor | Admin | | **Secondary Actor** |  |
| Description: | Logout website PSM | | | |
| Preconditions: | Login by account admin PSM. | | | |
| Trigger: | Click button “Đăng xuất” . | | | |
| Post conditions: | Come back Home page as customer | | | |
| Normal Flow | | | | |
| Step | **Actor’ event** | | **System response** | |
| 1 | Click on “Đăng xuất” quick menu. | |  | |
| 2 |  | | 1. Log user out of the systems. 2. Redirect user back to the home page as customer. | |
| Alternative Flows: N/A   |  | | --- | |  | | | | | |
| Exceptions: N/A | | | | |
|  | | | | |
| Priority | | | Normal | |
| Frequency of Use: | | | Normal | |
| Business Rules: | | | N/A | |
| Other Information: | | | N/A | |

##### Change Password Admin

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Use Case ID | UC- 003 | Version | | |  |  | 1.0 | | |
| Use Case Name | Change Password | | | | | | | | |
| Created by: | HuyNM | | **Date Created:** | | | | | | 12/04/2018 |
| Primary Actor | Manager | | **Secondary Actor** | | | | | |  |
| Description: | This function allows Manager to change password or forgot their password. | | | | | | | | |
| Preconditions: | Systems must be connected to the Internet. | | | | | | | | |
| Trigger: | Click on “Forgot Password” hyperlink. | | | | | | | | |
| Post conditions: | Systems redirect user to “Change Password” page. | | | | | | | | |
|  |  | | |
| Step | **Actor Event** | | **System Response** | | | | | | |
| 1 | Click on “Login” quick menu at the homepage. | |  | | | | | | |
| 2 |  | | Display “Login” page include:   1. Username textbox. 2. Password textbox. 3. “Login” button. 4. “Cancel” button.   “Forgot password” hyperlink. | | | | | | |
| 3 | Click on “Forgot password” hyperlink. | |  | | | | | | |
| 4 |  | | Redirect user to “Change Password” page include:   1. Username textbox. 2. “Please enter your username” message.   “Enter” button. | | | | | | |
| 5 | Enter Username. | |  | | | | | | |
| 6 | Click “Enter” button. | |  | | | | | | |
| 7 |  | | Change “Change Password” page to:   1. “Old password has been send to your email” message. 2. “Old password” textbox. 3. “New password” textbox. 4. “OK” button.   “Cancel” button. | | | | | | |
| 8 | Enter “Old password”, “New password”. | |  | | | | | | |
| 9 | Click “OK” button. | |  | | | | | | |
| 10 |  | | Change user password, redirect user to login page. | | | | | | |
|  |  | | |
| AT1 | At step 6, Manager click “Cancel” button. | | | | | | | | |
| Step | **Actor Event** | | **System Response** | | | | | | |
| 6.1 | Click “Cancel” button | |  | | | | | | |
| 6.2 |  | | Redirect user back to “Login” page | | | | | | |
|  |  | | | | | | |
|  |  | | | | | | |
| EC1 | At step 8 in main flow, if user entered wrong User name or Password | | | | | | | | |
| Step | **Actor Event** | | **System Response** | | | | | | |
| 8.1 | User entered wrong old password | |  | | | | | | |
| 8.2 |  | | Pop error message. | | | | | | |
|  |  | | |
| Priority | High | | | | | | | | |
| Frequency of Use: | Low | | | | | | | | |
| Business Rules: | B2 | | | | | | | | |
| Other Information: | N/A | | | | | | | | |

##### View Province/District

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Use Case ID UC-004 | | | | Version | | 1.0 |
| Use Case Name | | View province/District | | | | |
| Created by: | | HuyNM | | **Date Created:** | | 12/04/2018 |
| Primary Actor | | Admin | | **Secondary Actor** | |  |
| Description: | | Show information about province and district. | | | | |
| Preconditions: | | Login by account admin PSM. | | | | |
| Trigger: | | Click District Management hyper link | | | | |
| Post conditions: | | Information of Studio need to exist. | | | | |
| Normal Flow | | | | | | |
| Step | **Actor’ event** | | | | **System response** | |
| 1 | Select Province in combo box then click Search. | | | |  | |
| 2 |  | | | | Display all districts of the province | |
| 3 | Click to the name of District hyperlink. | | | |  | |
| 4 |  | | | | Display district details | |
| Alternative Flows: N/A | | | | | | |
|  | | | | | | |
| Exceptions: N/A | | | | | | |
|  | | | | | | |
| Priority | | | Normal | | | |
| Frequency of Use: | | | Normal | | | |
| Business Rules: | | | N/A | | | |
| Other Information: | | | N/A | | | |

##### Edit Province/District

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use Case ID UC-005 | | | | Version | 1.0 |
| Use Case Name | | Edit Province/District | | | |
| Created by: | | HuyNM | | **Date Created:** | 12/04/2018 |
| Primary Actor | | Admin | | **Secondary Actor** |  |
| Description: | | Edit province and district. | | | |
| Preconditions: | | Login by account admin PSM.  Province/District data has been added to the database.  Display district details. | | | |
| Trigger: | | Click one District hyper link. | | | |
| Post conditions: | | Information of Studio need to exist. | | | |
| Normal Flow | | | | | |
| Step | **Actor’ event** | | | **System response** | |
| 1 | Write the textbox in Update District Name then click button Update Name. | | |  | |
| 2 |  | | | Display popup Edit successful. | |
| 3 | Click to the “OK”. | | |  | |
| 4 |  | | | Display district details | |
| Alternative Flows: N/A | | | | | |
|  | | | | | |
| Exceptions: N/A | | | | | |
|  | | | | | |
| Priority | | | Normal | | |
| Frequency of Use: | | | Normal | | |
| Business Rules: | | | B2,B4. | | |
| Other Information: | | | N/A | | |

##### Search Province/District

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Use Case ID UC-006 | | | | Version | | 1.0 |
| Use Case Name | | View province/District | | | | |
| Created by: | | HuyNM | | **Date Created:** | | 112/04/2018 |
| Primary Actor | | Admin | | **Secondary Actor** | |  |
| Description: | | Show information about province and district. | | | | |
| Preconditions: | | Login by account admin PSM. | | | | |
| Trigger: | | Click District Management hyper link | | | | |
| Post conditions: | | Information of Studio need to exist. | | | | |
| Normal Flow | | | | | | |
| Step | **Actor’ event** | | | | **System response** | |
| 1 | Select Province in combo box then click Search. | | | |  | |
| 2 |  | | | | Display all districts of the province | |
| 3 | Click to the name of District hyperlink. | | | |  | |
| 4 |  | | | | Display district details | |
| Alternative Flows: N/A | | | | | | |
|  | | | | | | |
| Exceptions: N/A | | | | | | |
|  | | | | | | |
| Priority | | | Normal | | | |
| Frequency of Use: | | | Normal | | | |
| Business Rules: | | | N/A | | | |
| Other Information: | | | N/A | | | |

##### Delete Province/District

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use Case ID UC-007 | | | | Version | 1.0 |
| Use Case Name | | Delete Province/District | | | |
| Created by: | | HuyNM | | **Date Created:** | 12/04/2018 |
| Primary Actor | | Admin | | **Secondary Actor** |  |
| Description: | | Delete province and district. | | | |
| Preconditions: | | Login by account admin PSM.  Province/District data has been added to the database.  Display district details. | | | |
| Trigger: | | Click one District hyper link. | | | |
| Post conditions: | | District removed in database. | | | |
| Normal Flow | | | | | |
| Step | **Actor’ event** | | | **System response** | |
| 1 | Select Province in combo box then click Search. | | |  | |
| 2 |  | | | Display name of district. | |
| 3 | Click to the button “Delete” then click “Ok”. | | |  | |
| 4 |  | | | Displayed popup “Successful Delete”. | |
| Alternative Flows: N/A | | | | | |
|  | | | | | |
| Exceptions: N/A | | | | | |
|  | | | | | |
| Priority | | | Normal | | |
| Frequency of Use: | | | Normal | | |
| Business Rules: | | | N/A | | |
| Other Information: | | | N/A | | |

##### Add Province/District

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use Case ID UC-008 | | | | Version | 1.0 |
| Use Case Name | | Add Province/District | | | |
| Created by: | | HuyNM | | **Date Created:** | 12/04/2018 |
| Primary Actor | | Admin | | **Secondary Actor** |  |
| Description: | | Add province and district. | | | |
| Preconditions: | | Login by account admin PSM.  Province/District data has been added to the database.  Display district details. | | | |
| Trigger: | | Click one District hyper link. | | | |
| Post conditions: | | Information of new District have to exist. | | | |
| Normal Flow | | | | | |
| Step | **Actor’ event** | | | **System response** | |
| 1 | Select Province in table “Add new district” then write in the text box District after that click “Add”. | | |  | |
| 2 |  | | | Display popup add district successful. | |
| 3 | Click to the “OK”. | | |  | |
| 4 |  | | | New district appear in province. | |
| Alternative Flows: N/A | | | | | |
|  | | | | | |
| Exceptions: N/A | | | | | |
|  | | | | | |
| Priority | | | Normal | | |
| Frequency of Use: | | | Normal | | |
| Business Rules: | | | B2,B3. | | |
| Other Information: | | | N/A | | |

##### View Studio

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Use Case ID UC-009 | | | | | Version | 1.0 |
| Use Case Name | | View Studio | | | | |
| Created by: | | HuyNM | | **Date Created:** | | 12/04/2018 |
| Primary Actor | | Admin | | **Secondary Actor** | |  |
| Description: | | Show all information about Studio in PSM. | | | | |
| Preconditions: | | Login by account admin PSM.  Studio data has been added to the database. | | | | |
| Trigger: | | Click to “Quản lý Studio” hyper link. | | | | |
| Post conditions: | | Information of Studio need to exist. | | | | |
| Normal Flow | | | | | | |
| Step | **Actor’ event** | | | **System response** | | |
| 1 | Select “hiển thị tất cả”. | | |  | | |
| 2 |  | | | PSM will display:   1. Number of Packagers found: 2. Studio ID 3. Studio Name 4. Package Detail 5. Studio Email 6. Studio Number 7. Studio Coordinate 8. Province 9. District | | |
| Alternative Flows: Select button “Search”. | | | | | | |
|  | | | | | | |
| Exceptions: N/A | | | | | | |
|  | | | | | | |
| Priority | | | Normal | | | |
| Frequency of Use: | | | Normal | | | |
| Business Rules: | | | N/A | | | |
| Other Information: | | | N/A | | | |

##### Edit Studio

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use Case ID UC-010 | | | | Version | 1.0 |
| Use Case Name | | Edit Studio | | | |
| Created by: | | HuyNM | | **Date Created:** | 12/04/2018 |
| Primary Actor | | Admin | | **Secondary Actor** |  |
| Description: | | Edit information of Studio. | | | |
| Preconditions: | | Login by account admin PSM.  Studio data has been added to the database.  Display Studio details. | | | |
| Trigger: | | Click to “Edit” hyper link. | | | |
| Post conditions: | | Information of Studio need to edited. | | | |
| Normal Flow | | | | | |
| Step | **Actor’ event** | | | **System response** | |
| 1 | Click Button ”Edit” | | |  | |
| 2 |  | | | Display table “Edit Studio”   1. Studio Name 2. Studio Address 3. Province 4. District 5. Studio email 6. Studio Number 7. Studio Coordinate | |
| 3 | Write to the textbox then click Edit | | |  | |
| 4 |  | | | Popup appear ”Bạn có chắc chắc muốn thay đổi”. | |
| 5 | Click “Ok” | | |  | |
| 6 |  | | | Popup appear “Thêm thành công”. | |
| Alternative Flows: N/A | | | | | |
|  | | | | | |
| Exceptions: N/A | | | | | |
|  | | | | | |
| Priority | | | Normal | | |
| Frequency of Use: | | | Normal | | |
| Business Rules: | | | N/A | | |
| Other Information: | | | N/A | | |

##### Search Studio

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use Case ID UC-011 | | | | Version | 1.0 |
| Use Case Name | | Search Province/District | | | |
| Created by: | | HuyNM | | **Date Created:** | 12/04/2018 |
| Primary Actor | | Admin | | **Secondary Actor** |  |
| Description: | | Search by Studio Name. | | | |
| Preconditions: | | Login by account admin PSM.  Studio show in formation. | | | |
| Trigger: | | Click one “Search” hyper link. | | | |
| Post conditions: | | Information of Studio need to exist. | | | |
| Normal Flow | | | | | |
| Step | **Actor’ event** | | | **System response** | |
| 1 | Write the textbox name or character of Studio want to show. | | |  | |
| 2 |  | | | Display all Studio have name or character same that. | |
| 3 | Click to the “OK”. | | |  | |
| 4 |  | | | Display district details | |
| Alternative Flows: N/A | | | | | |
|  | | | | | |
| Exceptions: N/A | | | | | |
|  | | | | | |
| Priority | | | Normal | | |
| Frequency of Use: | | | Normal | | |
| Business Rules: | | | N/A | | |
| Other Information: | | | N/A | | |

##### Delete Studio

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use Case ID UC-012 | | | | Version | 1.0 |
| Use Case Name | | Delete Studio | | | |
| Created by: | | HuyNM | | **Date Created:** | 12/04/2018 |
| Primary Actor | | Admin | | **Secondary Actor** |  |
| Description: | | Delete Studio information. | | | |
| Preconditions: | | Login by account admin PSM.  Studio information data has been added to the database. | | | |
| Trigger: | | Click to “Delete” hyper link. | | | |
| Post conditions: | | Information of Studio remove in the data. | | | |
| Normal Flow | | | | | |
| Step | **Actor’ event** | | | **System response** | |
| 1 | Click to delete in table Studio information. | | |  | |
| 2 |  | | | Display popup “Bạn có muốn xoá Studio này”. | |
| 3 | Click to the “OK”. | | |  | |
| 4 |  | | | Display new table information. | |
| Alternative Flows: N/A | | | | | |
|  | | | | | |
| Exceptions: N/A | | | | | |
|  | | | | | |
| Priority | | | Normal | | |
| Frequency of Use: | | | Normal | | |
| Business Rules: | | | N/A | | |
| Other Information: | | | N/A | | |

##### Add Studio

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Use Case ID UC-013 | | | | Version | 1.0 | |
| Use Case Name | | Add Studio | | | | |
| Created by: | | HuyNM | | **Date Created:** | | 12/04/2018 |
| Primary Actor | | Admin | | **Secondary Actor** | |  |
| Description: | | Add Studio information | | | | |
| Preconditions: | | Login by account admin PSM.  Province/District data has been added to the database.  Display district details. | | | | |
| Trigger: | | Click “Add New Studio” hyper link. | | | | |
| Post conditions: | | Updated new Studio in database. | | | | |
| Normal Flow | | | | | | |
| Step | **Actor’ event** | | | **System response** | | |
| 1 | Click to Add new Studio hyperlink. | | |  | | |
| 2 |  | | | Display all information   1. Studio Name 2. Studio Address 3. Province 4. District 5. Studio Email 6. Studio Number 7. Studio Coordinate | | |
| 3 | Click to the “Add”. | | |  | | |
| 4 |  | | | Add Studio successful | | |
| Alternative Flows: At step 4 click “Reset”. | | | | | | |
|  | | | | | | |
| Exceptions: N/A | | | | | | |
|  | | | | | | |
| Priority | | | Normal | | | |
| Frequency of Use: | | | Normal | | | |
| Business Rules: | | | B3,B4,B6 | | | |
| Other Information: | | | N/A | | | |

##### View Studio Manager

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use Case ID UC-014 | | | | Version | 1.0 |
| Use Case Name | | View Studio Manager | | | |
| Created by: | | HuyNM | | **Date Created:** | 12/04/2018 |
| Primary Actor | | Admin | | **Secondary Actor** |  |
| Description: | | Show all information about Studio manager in PSM. | | | |
| Preconditions: | | Login by account admin PSM.  Studio data has been added to the database. | | | |
| Trigger: | | Click to “Quản lý Studio” hyper link. | | | |
| Post conditions: | | Information of Studio need to exist. | | | |
| Normal Flow | | | | | |
| Step | **Actor’ event** | | | **System response** | |
| 1 | Select “hiển thị tất cả”. | | |  | |
| 2 |  | | | PSM will display:   1. Number of Packagers found: 2. Studio ID 3. Studio Name 4. Package Detail 5. Studio Email 6. Studio Number 7. Studio Coordinate 8. Province 9. District | |
| Alternative Flows: Select button “Search”. | | | | | |
|  | | | | | |
| Exceptions: N/A | | | | | |
|  | | | | | |
| Priority | | | Normal | | |
| Frequency of Use: | | | Normal | | |
| Business Rules: | | | N/A | | |
| Other Information: | | | N/A | | |

##### Edit Studio

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use Case ID UC-015 | | | | Version | 1.0 |
| Use Case Name | | Edit Studio Manager | | | |
| Created by: | | HuyNM | | **Date Created:** | 12/04/2018 |
| Primary Actor | | Admin | | **Secondary Actor** |  |
| Description: | | Edit information of Studio. | | | |
| Preconditions: | | Login by account admin PSM.  Studio data has been added to the database.  Display Studio details. | | | |
| Trigger: | | Click to “Edit” hyper link. | | | |
| Post conditions: | | Information of Studio need to edited. | | | |
| Normal Flow | | | | | |
| Step | **Actor’ event** | | | **System response** | |
| 1 | Click Button ”Edit” | | |  | |
| 2 |  | | | Display table “Edit User”   1. User Name 2. Password 3. User Role 4. Studio | |
| 3 | Write to the textbox then click Edit | | |  | |
| 4 |  | | | Popup appear ”Bạn có chắc chắc muốn thay đổi”. | |
| 5 | Click “Ok” | | |  | |
| 6 |  | | | Popup appear “Thêm thành công”. | |
| Alternative Flows: N/A | | | | | |
|  | | | | | |
| Exceptions: N/A | | | | | |
|  | | | | | |
| Priority | | | Normal | | |
| Frequency of Use: | | | Normal | | |
| Business Rules: | | | N/A | | |
| Other Information: | | | N/A | | |

##### Search Studio Manager

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Use Case ID UC-016 | | | | | Version | 1.0 |
| Use Case Name | | Search Studio Manager | | | | |
| Created by: | | HuyNM | | | **Date Created:** | 12/04/2018 |
| Primary Actor | | Admin | | | **Secondary Actor** |  |
| Description: | | Search by Studio Name Manager | | | | |
| Preconditions: | | Login by account admin PSM.  Studio Manager show in formation. | | | | |
| Trigger: | | Click one “Search” hyper link. | | | | |
| Post conditions: | | Information of Studio Manager need to exist. | | | | |
| Normal Flow | | | | | | |
| Step | **Actor’ event** | | | **System response** | | |
| 1 | Write the textbox name or character of Studio Manager want to show. | | |  | | |
| 2 |  | | | Display all Studio Manager have name or character same that. | | |
| 3 | Click to the “OK”. | | |  | | |
| 4 |  | | | Display district details | | |
| Alternative Flows: N/A | | | | | | |
|  | | | | | | |
| Exceptions: N/A | | | | | | |
|  | | | | | | |
| Priority | | | Normal | | | |
| Frequency of Use: | | | Normal | | | |
| Business Rules: | | | N/A | | | |
| Other Information: | | | N/A | | | |

##### Delete Studio Manager

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use Case ID UC-017 | | | | Version | 1.0 |
| Use Case Name | | Delete Studio Manager | | | |
| Created by: | | HuyNM | | **Date Created:** | 12/04/2018 |
| Primary Actor | | Admin | | **Secondary Actor** |  |
| Description: | | Delete Studio Manager information. | | | |
| Preconditions: | | Login by account admin PSM.  Studio manager information data has been added to the database. | | | |
| Trigger: | | Click to “Delete” hyper link. | | | |
| Post conditions: | | Information of Studio manager remove in the data. | | | |
| Normal Flow | | | | | |
| Step | **Actor’ event** | | | **System response** | |
| 1 | Click to delete in table Studio manager information. | | |  | |
| 2 |  | | | Display popup “Bạn có muốn xoá người quản lý Studio này”. | |
| 3 | Click to the “OK”. | | |  | |
| 4 |  | | | Display new table Studio manager information | |
| Alternative Flows: N/A | | | | | |
|  | | | | | |
| Exceptions: N/A | | | | | |
|  | | | | | |
| Priority | | | Normal | | |
| Frequency of Use: | | | Normal | | |
| Business Rules: | | | N/A | | |
| Other Information: | | | N/A | | |

##### Add Studio Information

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use Case ID UC-018 | | | | Version | 1.0 |
| Use Case Name | | Add Studio Information | | | |
| Created by: | | HuyNM | | **Date Created:** | 12/04/2018 |
| Primary Actor | | Admin | | **Secondary Actor** |  |
| Description: | | Add Studio Manager information | | | |
| Preconditions: | | Login by account admin PSM. | | | |
| Trigger: | | Click “Add New Studio” hyper link. | | | |
| Post conditions: | | Updated new Studio Manager in database. | | | |
| Normal Flow | | | |  | |
| Step | **Actor’ event** | | | **System response** | |
| 1 | Click to Add new Studio Manager hyperlink. | | |  | |
| 2 |  | | | Display all information   1. User Name 2. Password 3. User Role 4. Studio | |
| 3 | Click to the “Add”. | | |  | |
| 4 |  | | | Add new User successful | |
| Alternative Flows: At step 4 click “Reset”. | | | | | |
|  | | | | | |
| Exceptions: N/A | | | | | |
|  | | | | | |
| Priority | | | Normal | | |
| Frequency of Use: | | | Normal | | |
| Business Rules: | | | B3,B4,B6 | | |
| Other Information: | | | N/A | | |

#### Manager

##### 

Figure 3.5.2.2.1**:** Manager Use Case Diagram

##### Login

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID | | UC-009 | Version | 1.0 |
| Use Case Name | | Login | | |
| Created by: | | BachNS | **Date Created:** | 12/04/2018 |
| Primary Actor | | Manager | **Secondary Actor** |  |
| Description: | | When Manager want to login by registered account to use more functions in website. | | |
| Preconditions: | | Systems must be connected to the Internet. | | |
| Trigger: | | Click on “Login” quick menu. | | |
| Post conditions: | | 1. Log Manager into system. 2. Redirect manager to Studio Management page. | | |
| Normal Flow | | | | |
| Step | **Actor Event** | **System Response** | | |
|  | Click on “Login” quick menu at the homepage. |  | | |
|  |  | Display “Login” page include:   1. Username textbox. 2. Password textbox. 3. “Login” button. 4. “Always remember me” checkbox. 5. “Forgot password” hyperlink. | | |
|  | Enter Username and Password. |  | | |
|  | Click “Login” button. |  | | |
|  |  | Log Manager in to the systems. | | |
|  |  | Display Studio Management page with message: “Login Successfully”. | | |
| Alternative Flows: N/A | | | | |
| Exceptions | | | | |
| EC1 | At step 3 in main flow, if user entered wrong User name or Password | | | |
| Step | **Actor Event** | **System Response** | | |
| 3.1 | User entered wrong User name or Password |  | | |
|  |  | Pop error message. | | |
|  | | | | |
| Priority | | High | | |
| Frequency of Use: | | High | | |
| Business Rules: | | B2 | | |
| Other Information: | | N/A | | |

##### Logout

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID | | UC-010 | Version | 1.0 |
| Use Case Name | | Logout | | |
| Created by: | | BachNS | **Date Created:** | 12/04/2018 |
| Primary Actor | | Manager | **Secondary Actor** |  |
| Description: | | This function allows a manager to logout | | |
| Preconditions: | | 1. Systems must be connected to the Internet. 2. Manager has login to the systems. | | |
| Trigger: | | Click on “Logout” quick menu. | | |
| Post conditions: | | 1. Log user out of the systems. 2. Redirect user back to the home page. | | |
| Normal Flow | | | | |
| Step | **Actor Event** | **System Response** | | |
| 1 | Click on “Logout” quick menu. |  | | |
| 2 |  | 1. Log user out of the systems. 2. Redirect user back to the home page as customer. | | |
| Alternative Flows: N/A | | | | |
| Exceptions: N/A | | | | |
|  | | | | |
| Priority | | High | | |
| Frequency of Use: | | Normal | | |
| Business Rules: | | N/A | | |
| Other Information: | | N/A | | |

##### Change Password

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use Case ID | | | UC-011 | Version | 1.0 |
| Use Case Name | | | Change Password | | |
| Created by: | | | BachNS | **Date Created:** | 12/04/2018 |
| Primary Actor | | | Manager | **Secondary Actor** |  |
| Description: | | | This function allows Manager to change password or forgot their password. | | |
| Preconditions: | | | Systems must be connected to the Internet. | | |
| Trigger: | | | Click on “Forgot Password” hyperlink. | | |
| Post conditions: | | | Systems redirect user to “Change Password” page. | | |
| Normal Flow | | | | | |
| Step | **Actor Event** | | **System Response** | | |
| 1 | Click on “Login” quick menu at the homepage. | |  | | |
| 2 |  | | Display “Login” page include:   1. Username textbox. 2. Password textbox. 3. “Login” button. 4. “Always remember me” checkbox. 5. “Forgot password” hyperlink. | | |
| 3 | Click on “Forgot password” hyperlink. | |  | | |
| 4 |  | | Redirect user to “Change Password” page include:   1. Username textbox. 2. “Please enter your username” message. 3. “Enter” button. | | |
| 5 | Enter Username. | |  | | |
| 6 | Click “Enter” button. | |  | | |
| 7 |  | | Change “Change Password” page to:   1. “Old password has been send to your email” message. 2. “Old password” textbox. 3. “New password” textbox. 4. “OK” button. 5. “Cancel” button. | | |
| 8 | Enter “Old password”, “New password”. | |  | | |
| 9 | Click “OK” button. | |  | | |
| 10 |  | | Change user password, redirect user to login page. | | |
| Alternative Flows | | | | | |
| AT1 | At step 6, Manager click “Cancel” button. | | | | |
| Step | **Actor Event** | | **System Response** | | |
| 6.1 | Click “Cancel” button | |  | | |
| 6.2 |  | | Redirect user back to “Login” page | | |
|  | | | | | |
| Exceptions | | | | | |
| EC1 | | At step 8 in main flow, if user entered wrong User name or Password | | | |
| Step | | **Actor Event** | **System Response** | | |
| 8.1 | | User entered wrong old password |  | | |
| 8.2 | |  | Pop error message. | | |
|  | | | | | |
| Priority | | | High | | |
| Frequency of Use: | | | Low | | |
| Business Rules: | | | B2 | | |
| Other Information: | | | N/A | | |

##### Search Contract

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use Case ID | | | UC-012 | Version | 1.0 |
| Use Case Name | | | Search Contract | | |
| Created by: | | | BachNS | **Date Created:** | 12/04/2018 |
| Primary Actor | | | Manager | **Secondary Actor** |  |
| Description: | | | Manager can search old Contract | | |
| Preconditions: | | | Manager must login to the systems. | | |
| Trigger: | | | Click on “Contract” hyperlink in the menu. | | |
| Post conditions: | | | Open table of detail information of the search contract. | | |
| Normal Flow | | | | | |
| Step | **Actor Event** | | **System Response** | | |
| 1 | Click on “Contract” hyperlink menu. | |  | | |
| 2 |  | | Show search table include:   1. Dropdown menu include: “ContractID, CutomerName, CustomerPhone”. 2. Textbox field for dropdown menu choice 3. “Search” button. | | |
| 3 | Pick dropdown menu options, enter textbox field. | |  | | |
| 4 | Click “Search” button. | |  | | |
| 5 |  | | Show table with fitted Contract information include:   1. Number of Contract found. 2. Contract ID 3. Customer ID 4. Studio ID 5. Contract Description. 6. Create Date. 7. Start Date. 8. End Date. 9. “Edit” Button. 10. “More Information” Button. | | |
| Alternative Flows: N/A | | | | | |
| Exceptions: | | | | | |
| EC1 | | At step 3 in main flow, if user entered invalid Contract information | | | |
| Step | | **Actor Event** | **System Response** | | |
| 8.1 | | User entered invalid information |  | | |
| 8.2 | |  | Pop error message. | | |
|  | | | | | |
| Priority | | | Normal | | |
| Frequency of Use: | | | Normal | | |
| Business Rules: | | | B3, B8 | | |
| Other Information: | | | N/A | | |

##### Edit Contract

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID | | UC-013 | Version | 1.0 |
| Use Case Name | | Edit Contract | | |
| Created by: | | BachNS | **Date Created:** | 12/04/2018 |
| Primary Actor | | Manager | **Secondary Actor** |  |
| Description: | | Manager can change old contract information. | | |
| Preconditions: | | 1. Manager must log-in to the systems. 2. Case UC-012 “Search Contract” must be executed successful first. | | |
| Trigger: | | Manager click on “Edit” Button in the Contract Information table. | | |
| Post conditions: | | New edit information saved in the Contract. | | |
| Normal Flow | | | | |
| Step | **Actor Event** | **System Response** | | |
| 1 | Click on “Edit” Button. |  | | |
| 2 |  | Open “Edit Contract” table include:   1. “Họ và Tên” textbox. 2. “Số điện thoại” textbox. 3. “Email” textbox. 4. “Ghi Chú” textbox. 5. “Save” Button. 6. “Reset” Button. | | |
| 3 | Enter new edit information |  | | |
| 4 |  | Save new edit information. | | |
| Alternative Flows | | | | |
| AT1 | At step 2, Manager click “Reset” button. | | | |
| Step | **Actor Event** | **System Response** | | |
| 2.1 | Click “Reset” Button. |  | | |
| 2.2 |  | Discard all textbox. Keep old information. | | |
| Exceptions: N/A | | | | |
| Priority | | Normal | | |
| Frequency of Use: | | Normal | | |
| Business Rules: | | B2, B3, B11 | | |
| Other Information: | | N/A | | |

##### Add Contract

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID | | UC-014 | Version | 1.0 |
| Use Case Name | | Add Contract | | |
| Created by: | | BachNS | **Date Created:** | 12/04/2018 |
| Primary Actor | | Manager | **Secondary Actor** |  |
| Description: | | Manager can add new Contract with customer. | | |
| Preconditions: | | Manager must log-in to the systems. | | |
| Trigger: | | Manager click on “Add Contract” hyperlink. | | |
| Post conditions: | | New Contract added to the systems. | | |
| Normal Flow | | | | |
| Step | **Actor Event** | **System Response** | | |
|  | Manager click on “Add Contract” hyperlink. |  | | |
|  |  | Open “Add Contract” table include:   1. “Họ và Tên” textbox. 2. “Số điện thoại” textbox. 3. “Email” textbox. 4. “Địa chỉ khách hàng” textbox. 5. “Studio làm việc” dropdown box. 6. “Tên sản phẩm” dropdown box. 7. “Chi tiết sản phẩm” textbox. 8. “Ghi chú” textbox. 9. “Chi tiết hợp đồng” textbox. 10. “Ngày lập hợp đồng” datetime. 11. “Ngày kết thúc hợp đồng” datetime. 12. “Ngày chụp” datetime. 13. “Add” Button. 14. “Reset” Button. | | |
|  | Enter new Contract information |  | | |
|  | Click “Add” |  | | |
|  |  | Added new Contract to the systems. | | |
| Alternative Flows | | | | |
| AT1 | At step 4, Manager click “Reset” button. | | | |
| Step | **Actor Event** | **System Response** | | |
| 4.1 | Manager click “Reset” button |  | | |
| 4.2 |  | Discard all textbox. | | |
| Exceptions: N/A | | | | |
| Priority | | Normal | | |
| Frequency of Use: | | Normal | | |
| Business Rules: | | B2, B3, B11 | | |
| Other Information: | | N/A | | |

##### Search Package

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use Case ID | | | UC-015 | Version | 1.0 |
| Use Case Name | | | Search Package | | |
| Created by: | | | BachNS | **Date Created:** | 12/04/2018 |
| Primary Actor | | | Manager | **Secondary Actor** |  |
| Description: | | | Manager can search Photo Package | | |
| Preconditions: | | | Manager must login to the systems. | | |
| Trigger: | | | Click on “Package Manager” hyperlink in the menu. | | |
| Post conditions: | | | Open table of detail information of the search package. | | |
| Normal Flow | | | | | |
| Step | **Actor Event** | | **System Response** | | |
| 1 | Click on “Package Manager” hyperlink menu. | |  | | |
| 2 |  | | Show search table include:   1. Package name textbox. 2. Search button. | | |
| 3 | Enter search package name. | |  | | |
| 4 | Click Search button. | |  | | |
| 5 |  | | Show search package table include:   1. Number of Package found. 2. Package ID. 3. Package name. 4. Package Detail. 5. Package Price. 6. Available. 7. “Show all” Hyperlink 8. “Add new Package” Hyperlink. 9. “Edit” button. 10. “Delete” button. | | |
| Alternative Flows: N/A | | | | | |
| Exceptions | | | | | |
| EC1 | | At step 3 in main flow, if user entered invalid Contract information | | | |
| Step | | **Actor Event** | **System Response** | | |
| 8.1 | | User entered invalid information |  | | |
| 8.2 | |  | Pop error message. | | |
|  | | | | | |
| Priority | | | Normal | | |
| Frequency of Use: | | | Normal | | |
| Business Rules: | | | B3 | | |
| Other Information: | | | N/A | | |

##### Add Package

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID | | UC-016 | Version | 1.0 |
| Use Case Name | | Add Package | | |
| Created by: | | BachNS | **Date Created:** | 12/04/2018 |
| Primary Actor | | Manager | **Secondary Actor** |  |
| Description: | | Manager can add new package for customer. | | |
| Preconditions: | | Manager must log-in to the systems. | | |
| Trigger: | | Manager click on “Add Package” hyperlink. | | |
| Post conditions: | | New Contract added to the systems. | | |
| Normal Flow | | | | |
| Step | **Actor Event** | **System Response** | | |
| 1 | Manager click on “Add Contract” hyperlink. |  | | |
| 2 |  | Open “Add Contract” table include:   1. Package name textbox. 2. Package details textbox. 3. Package price textbox. 4. Package availability drop check box. 5. “Add” button. 6. “Reset” button. | | |
| 3 | Enter new Package information |  | | |
| 4 | Click “Add” |  | | |
| 5 |  | Added new Package to the systems. | | |
| Alternative Flows | | | | |
| AT1 | At step 4, Manager click “Reset” button. | | | |
| Step | **Actor Event** | **System Response** | | |
| 4.1 | Manager click “Reset” button |  | | |
| 4.2 |  | Discard all textbox. | | |
| Exceptions: N/A | | | | |
| Priority | | Normal | | |
| Frequency of Use: | | Normal | | |
| Business Rules: | | B2, B3 | | |
| Other Information: | | N/A | | |

##### Edit Package

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID | | UC-017 | Version | 1.0 |
| Use Case Name | | Edit Package | | |
| Created by: | | BachNS | **Date Created:** | 12/04/2018 |
| Primary Actor | | Manager | **Secondary Actor** |  |
| Description: | | Manager can edit old package information. | | |
| Preconditions: | | 1. Manager must log-in to the systems. 2. Case UC-015 “Search Package” must be executed successful first. | | |
| Trigger: | | Manager click on “Edit” Button in the Package Information table. | | |
| Post conditions: | | New edit information saved in the Package. | | |
| Normal Flow | | | | |
| Step | **Actor Event** | **System Response** | | |
| 1 | Click on “Edit” Button. |  | | |
| 2 |  | Open “Edit package” table include:   1. Package Id (Cannot edit). 2. Package Name textbox. 3. Package Detail textbox. 4. Package price textbox. 5. Package availability drop check box. 6. “Add” button. 7. “Reset” button | | |
| 3 | Enter new edit information |  | | |
| 4 |  | Save new edit information. | | |
| Alternative Flows | | | | |
| AT1 | At step 3, Manager click “Reset” button. | | | |
| Step | **Actor Event** | **System Response** | | |
| 2.1 | Click “Reset” Button. |  | | |
| 2.2 |  | Discard all textbox. Keep old information. | | |
| Exceptions: N/A | | | | |
| Priority | | Normal | | |
| Frequency of Use: | | Normal | | |
| Business Rules: | | B2, B3, B11 | | |
| Other Information: | | N/A | | |

##### View Picture

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID | | UC-018 | Version | 1.0 |
| Use Case Name | | View Picture | | |
| Created by: | | BachNS | **Date Created:** | 12/04/2018 |
| Primary Actor | | Manager | **Secondary Actor** |  |
| Description: | | Manager can view Contract uploaded picture. | | |
| Preconditions: | | Manager must login to the systems. | | |
| Trigger: | | Click on “Picture” hyperlink in the menu. | | |
| Post conditions: | | Open all Contract picture. | | |
| Normal Flow | | | | |
| Step | **Actor Event** | **System Response** | | |
| 1 | Click on “Picture” hyperlink in the menu. |  | | |
| 2 |  | Show Picture table:   1. Pictures. 2. Contract name of the pictures. | | |
| Alternative Flows: N/A | | | | |
| Exceptions: N/A | | | | |
| Priority | | Normal | | |
| Frequency of Use: | | Normal | | |
| Business Rules: | | N/A | | |
| Other Information: | | N/A | | |

##### Upload Picture

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use Case ID | | UC-019 | | Version | 1.0 |
| Use Case Name | | Upload Picture | | | |
| Created by: | | BachNS | | **Date Created:** | 12/04/2018 |
| Primary Actor | | Manager | | **Secondary Actor** |  |
| Description: | | Manager can upload Contract uploaded picture. | | | |
| Preconditions: | | 1. Manager must login to the systems. 2. Case UC-012 “Search Contract” must be executed successful first. | | | |
| Trigger: | | Click on “Upload Pictures” button. | | | |
| Post conditions: | | Uploaded all picture to the Contract. | | | |
| Normal Flow | | | | | |
| Step | **Actor Event** | **System Response** | | | |
| 1 | Click on “Upload” button |  | | | |
| 2 |  | Show Upload pop-up:   1. Upload button. 2. Cancel button. | | | |
| 3 | Choose picture to upload. |  | | | |
| 4 | Click “Upload” button. |  | | | |
|  |  | Upload picture to the systems. | | | |
| Alternative Flows | | | | | |
| AT1 | At step 4, Manager click “Cancel” button. | | | | |
| Step | **Actor Event** | | **System Response** | | |
| 4.1 | Manager click “Cancel” button | |  | | |
| 4.2 |  | | Discard all upload picture. Close upload pop-up. | | |
| Exceptions: N/A | | | | | |
| Priority | | Normal | | | |
| Frequency of Use: | | Normal | | | |
| Business Rules: | | B14, B13. | | | |
| Other Information: | | N/A | | | |

##### View Customer Information

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID | | UC-020 | Version | 1.0 |
| Use Case Name | | View Customer Information | | |
| Created by: | | BachNS | **Date Created:** | 12/04/2018 |
| Primary Actor | | Manager | **Secondary Actor** |  |
| Description: | | Manager can view Customers information. | | |
| Preconditions: | | Manager must log-in to the systems. | | |
| Trigger: | | Manager click on “View Customer Information” hyperlink. | | |
| Post conditions: | | Manager can view table of customer information. | | |
| Normal Flow | | | | |
| Step | **Actor Event** | **System Response** | | |
| 1 | Click on “View Customer Information” hyperlink. |  | | |
| 2 |  | Open “Customer Information” table include:   1. Họ và Tên. 2. Số điện thoại. 3. Email. 4. Giới tính. 5. Địa chỉ. 6. Lịch sử mua hàng. | | |
| Alternative Flows: N/A | | | | |
| Exceptions: N/A | | | | |
| Priority | | Normal | | |
| Frequency of Use: | | Normal | | |
| Business Rules: | | B2, B3, B11 | | |
| Other Information: | | N/A | | |

##### **Customer**

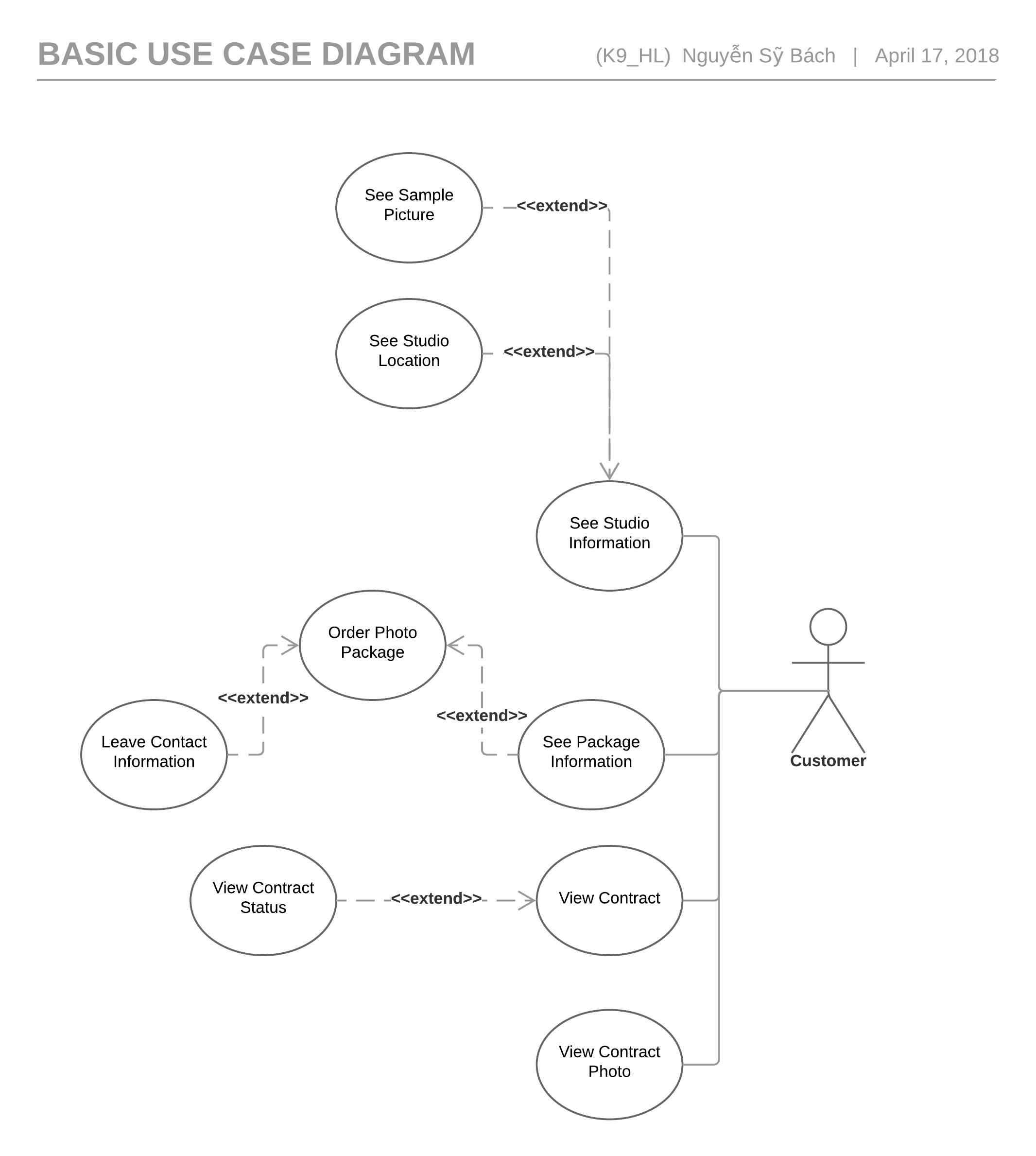


Figure 3.5.2.2.2: Customer Use Case Diagram

##### See Studio Information

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use Case ID UC-019 | | | | Version | 1.0 |
| Use Case Name | | See Studio Information | | | |
| Created by: | | HuyNM | | **Date Created:** | 12/04/2018 |
| Primary Actor | | Customer | | **Secondary Actor** |  |
| Description: | | Add Studio Manager information | | | |
| Preconditions: | | Customer must to go PSM website. | | | |
| Trigger: | | Customer clock to btton”Studio”. | | | |
| Post conditions: | | Display all information Studio. | | | |
| Normal Flow | | | | | |
| Step | **Actor’ event** | | | **System response** | |
| 1 | Click to button “Studio” | | |  | |
| 2 |  | | | Display above “Studio” is list of Studio. | |
| 3 | Click on name of Studio. | | |  | |
| 4 |  | | | Display all Information studio.   1. Name Studio 2. Studio Phone number 3. Studio Email 4. Studio address 5. Studio map location | |
| Alternative Flows: N/A | | | | | |
|  | | | | | |
| Exceptions: N/A | | | | | |
|  | | | | | |
| Priority | | | Normal | | |
| Frequency of Use: | | | Normal | | |
| Business Rules: | | | B3,B6 | | |
| Other Information: | | | N/A | | |

##### See Sample Picture

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use Case ID UC-020 | | | | Version | 1.0 |
| Use Case Name | | See sample Picture | | | |
| Created by: | | HuyNM | | **Date Created:** | 12/04/2018 |
| Primary Actor | | Customer | | **Secondary Actor** |  |
| Description: | | Someone want to see sample picture of studio. | | | |
| Preconditions: | | Customer must to go PSM website. | | | |
| Trigger: | | Customer clock to button ”Studio”. | | | |
| Post conditions: | | Picture is showed | | | |
| Normal Flow | | | | | |
| Step | **Actor’ event** | | | **System response** | |
| 1 | Click on to Button ”Studio”. | | |  | |
| 2 |  | | | Display sample picture of some Studio. | |
| 3 | 1. Move to the Studio want to see sample picture and click to this. 2. Click to picture want to show. | | |  | |
| 4 |  | | | Show the picture. | |
| Alternative Flows: N/A | | | | | |
|  | | | | | |
| Exceptions: N/A | | | | | |
|  | | | | | |
| Priority | | | Normal | | |
| Frequency of Use: | | | Normal | | |
| Business Rules: | | | N/A | | |
| Other Information: | | | N/A | | |

##### See Studio Location

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use Case ID UC-021 | | | | Version | 1.0 |
| Use Case Name | | See Studio Location | | | |
| Created by: | | HuyNM | | **Date Created:** | 12/04/2018 |
| Primary Actor | | Customer | | **Secondary Actor** |  |
| Description: | | Show for customer location of Studio | | | |
| Preconditions: | | Customer must to go PSM website. | | | |
| Trigger: | | Customer clock to button ”Studio”. | | | |
| Post conditions: | | Location of Studio will show on the map. | | | |
| Normal Flow | | | | | |
| Step | **Actor’ event** | | | **System response** | |
| 1 | Click on the button Studio then lick name of Studio. | | |  | |
| 2 |  | | | Information of Studio will show on the table “Studio Information”. | |
| 3 | Click on Link of “Studio map location” hyperlink. | | |  | |
| 4 |  | | | Show map studio on the googlemap. | |
| Alternative Flows: N/A | | | | | |
|  | | | | | |
| Exceptions: N/A | | | | | |
|  | | | | | |
| Priority | | | Normal | | |
| Frequency of Use: | | | Normal | | |
| Business Rules: | | | N/A | | |
| Other Information: | | | N/A | | |

##### See Package Information

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Use Case ID UC-022 | | | | Version | | 1.0 |
| Use Case Name | | See Package Information | | | | |
| Created by: | | HuyNM | | **Date Created:** | | 12/04/2018 |
| Primary Actor | | Customer | | **Secondary Actor** | |  |
| Description: | | Show for customer packages of studio. | | | | |
| Preconditions: | | Customer must to go PSM website. | | | | |
| Trigger: | | Customer selected to name of Studio. | | | | |
| Post conditions: | | Show all package Studio. | | | | |
| Normal Flow | | | | | | |
| Step | **Actor’ event** | | | | **System response** | |
| 1 | Click to the “read more”. | | | |  | |
| 2 |  | | | | Detail of package will show all. | |
| Alternative Flows: N/A | | | | | | |
|  | | | | | | |
| Exceptions: N/A | | | | | | |
|  | | | | | | |
| Priority | | | Normal | | | |
| Frequency of Use: | | | Normal | | | |
| Business Rules: | | | N/A | | | |
| Other Information: | | | N/A | | | |

##### Order photo Package and leave Information

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use Case ID UC-021 | | | | Version | 1.0 |
| Use Case Name | | Order photo Package and leave Information | | | |
| Created by: | | HuyNM | | **Date Created:** | 12/04/2018 |
| Primary Actor | | Customer | | **Secondary Actor** |  |
| Description: | | Customer Order package. | | | |
| Preconditions: | | Customer must to go PSM website.  Customer need choose the Studio want to take picture. | | | |
| Trigger: | | Customer click to icon “Bag” for order package. | | | |
| Post conditions: | | PSM will send email for customer. | | | |
| Normal Flow | | | | | |
| Step | **Actor’ event** | | | **System response** | |
| 1 | After Customer select the cart. Click to the button “Buy”. | | |  | |
| 2 |  | | | PSM will move to checkout page and show price of package. | |
| 3 | Write in table Customer information:   1. Customer Name 2. Customer Phone Number 3. Customer Email   Then click “Buy”. | | |  | |
| 4 |  | | | PSM send email for Customer and Manager of package. | |
| Alternative Flows: N/A | | | | | |
|  | | | | | |
| Exceptions: N/A | | | | | |
|  | | | | | |
| Priority | | | Normal | | |
| Frequency of Use: | | | Normal | | |
| Business Rules: | | | B2,B6,B11 | | |
| Other Information: | | | N/A | | |

##### View contract

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Use Case ID UC-024 | | | | Version | | 1.0 |
| Use Case Name | | View contract | | | | |
| Created by: | | HuyNM | | **Date Created:** | 12/04/2018 | |
| Primary Actor | | Customer | | **Secondary Actor** |  | |
| Description: | | Show contract for customer | | | | |
| Preconditions: | | Customer must to go PSM website. | | | | |
| Trigger: | | Customer click to “Contract” hyperlink. | | | | |
| Post conditions: | | All information displayed. | | | | |
| Normal Flow | | | | | | |
| Step | **Actor’ event** | | | **System response** | | |
| 1 | Click to contract hyperlink | | |  | | |
| 2 |  | | | PSM will display table contract. | | |
| 3 | Write character of contract to text box in header then hit Enter. | | |  | | |
| 4 |  | | | PSM will display detail of your contract. | | |
| Alternative Flows: N/A | | | | | | |
|  | | | | | | |
| Exceptions: N/A | | | | | | |
|  | | | | | | |
| Priority | | | Normal | | | |
| Frequency of Use: | | | Normal | | | |
| Business Rules: | | | B1 | | | |
| Other Information: | | | N/A | | | |

##### View contract status

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use Case ID UC-025 | | | | Version | 1.0 |
| Use Case Name | | View contract status | | | |
| Created by: | | HuyNM | | **Date Created:** | 12/04/2018 |
| Primary Actor | | Customer | | **Secondary Actor** |  |
| Description: | | Show contract for customer | | | |
| Preconditions: | | Customer must to go PSM website. | | | |
| Trigger: | | Customer click to “Contract” hyperlink. | | | |
| Post conditions: | | All information displayed. | | | |
| Normal Flow | | | | | |
| Step | **Actor’ event** | | | **System response** | |
| 1 | Click to contract hyperlink | | |  | |
| 2 |  | | | PSM will display table contract. | |
| 3 | Write character of contract to text box in header then hit Enter. | | |  | |
| 4 |  | | | PSM will display detail of your contract. | |
| Alternative Flows: N/A | | | | | |
|  | | | | | |
| Exceptions: N/A | | | | | |
|  | | | | | |
| Priority | | | Normal | | |
| Frequency of Use: | | | Normal | | |
| Business Rules: | | | B1 | | |
| Other Information: | | | N/A | | |

##### View contract photo

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Use Case ID UC-026 | | | | | Version | 1.0 |
| Use Case Name | | View contract photo | | | | |
| Created by: | | HuyNM | | | **Date Created:** | 12/04/2018 |
| Primary Actor | | Customer | | | **Secondary Actor** |  |
| Description: | | Show photo in contract for Customer. | | | | |
| Preconditions: | | Customer must to go PSM website. | | | | |
| Trigger: | | Customer click to “Contract” hyperlink. | | | | |
| Post conditions: | | All information displayed. | | | | |
| Normal Flow | | | | | | |
| Step | **Actor’ event** | | | **System response** | | |
| 1 | Click to contract hyperlink | | |  | | |
| 2 |  | | | PSM will display table contract. | | |
| 3 | Write character of contract to text box in header then hit Enter. | | |  | | |
| 4 |  | | | PSM will display detail of your contract. | | |
| Alternative Flows: N/A | | | | | | |
|  | | | | | | |
| Exceptions: N/A | | | | | | |
|  | | | | | | |
| Priority | | | Normal | | | |
| Frequency of Use: | | | Normal | | | |
| Business Rules: | | | B1 | | | |

#### System　システム

Figure 3.5.2.4 System use case diagram

## Usability 　ユーザビリティ

* Interface should be beauty, simple, easy-to-use, clear.
* Provide an introduction to support users.

## Reliability 　信頼性

* Search result return must be correctly.
* Use’s information must be reliable.
* The database must be backed up daily and can be recovered if necessary.

## Performance　性能

* Maximum response time for display information must be smaller than 1 second.
* Waiting time of processing any functions will be 2 to 5 seconds.

## Maintainability　保全性

* The output of the project must include naming conventions and coding standards documentations. Code implementation must be easy to maintain.
* All code must be comment clearly, including class and method documentations.
* The design of the system must be loosely coupled that chances on some module will not affect others.

## Supporting Information　サポート情報

N/A

# ARCHITECTURE DESIGN　アーキテクチャ設計書

## Introduction　はじめに

### Purpose　目的

The purpose of this chapter is to give developers the overview and the detailed technical design of Jobs Finding System. This chapter consist of:

* Architecture overview
* Component diagram
* Detailed design
* Detailed description of components
* Database design

### References　参照資料

|  |  |  |
| --- | --- | --- |
| No. | Document | Source/Web Address |
| 1 | Architecture Design Template | FPT Software Cop. |
| 2 | JFS\_Software\_Requirements\_Specification\_v1.0 | Internal Resource |

## Architectural Overview　アーキテクチャ概要

### System Architecture　システムアーキテクチャ

In our project we use MVC architectural model.

[INSERT DIAGRAM HERE]

***Figure 4.1****: System Overview*

## System Architecture Explanation　システムアーキテクチャについての説明

The following is the explanation of the function and mechanism of each unit in the System Design.

**Azure Cloud Service:** The entire system will be deployed to Azure Cloud. Our target is developing a highly available and responsive system.

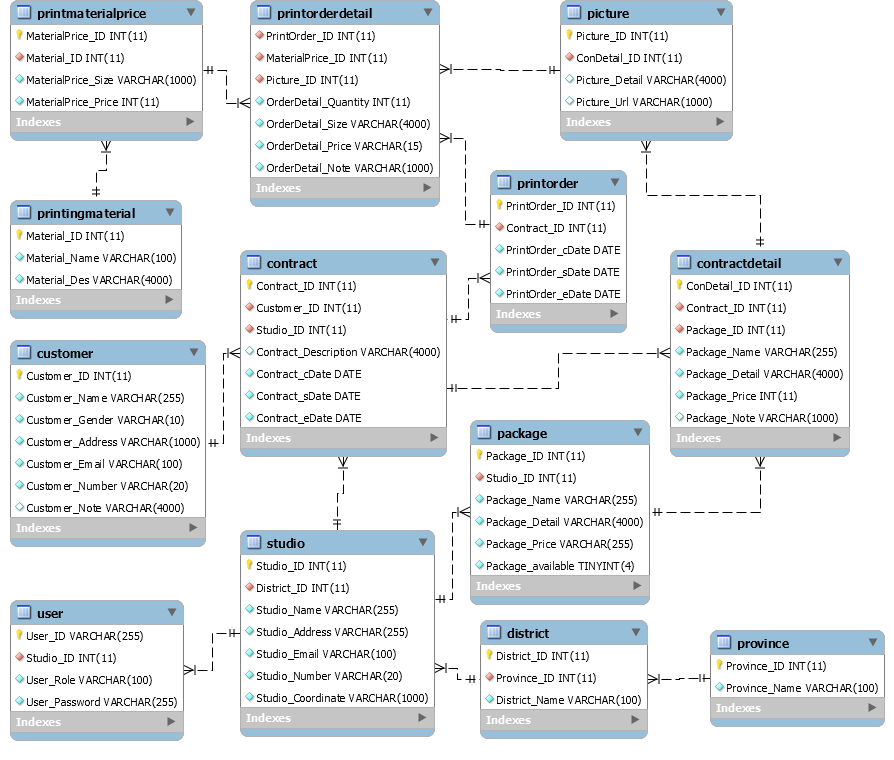
**Azure SendGrid**: We use SendGrid to send information to Users by email. SendGrid is a cloud-based email service that provides reliable transactional email delivery and real time analytics.

**Tawk.to API**: This is a free live chat API which we use to chat with all visitor on our website.

**MySQL**: MySQL is an open-source relational database management system. We use MySQL for adding, accessing and managing content of our system’s database.

## Design of Jobs Finding System　Jobs finding system の設計書

### Database Design データベース設計

**Figure 4.4.1:** Table Diagram

### Database detail　データベースの詳細

#### Manager

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Attribute** | **Type** | **Mandatory** | **Description** |
| 1 | User\_ID | String | Y | User’s username |
| 2 | Studio\_ID | int | Y | Studio’s ID |
| 3 | User\_Role | String | Y | User’s role |
| 4 | User\_Password | String | Y | User’s password |

#### Studio

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Attribute** | **Type** | **Mandatory** | **Description** |
| 1 | Studio\_ID | Int | Y | Studio’s ID |
| 2 | District\_ID | String | Y | District’s ID. |
| 3 | Studio\_Name | String | Y | Studio’s name |
| 4 | Studio\_Address | String | Y | Studio’s physical address |
| 5 | Studio\_Email | String | Y | Studio’s email address |
| 6 | Studio\_Number | Number | Y | Studio’s phone number |
| 7 | Studio\_Coordinate | String | Y | Studio’s location coordinate |
| 8 | Studio\_Icon | String | Y | Studio’s icon |

#### District

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Attribute | Type | Mandatory | Description |
| 1 | District\_ID | Int | Y | District’s ID |
| 2 | Province\_ID | Int | Y | Province that have the current district |
| 3 | District\_Name | String | Y | District’s Name |

#### Province

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Attribute | Type | Mandatory | Description |
| 1 | Province\_ID | Int | Y | Province’s ID |
| 2 | Province\_Name | String | Y | Province’s Name |

#### Contract

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Attribute | Type | Mandatory | Description |
|  | Contract\_ID | String | Y | Contract’s ID |
|  | Customer\_ID | Int | Y | Customer’s ID |
|  | Studio\_ID | Int | Y | Studio’s ID |
|  | Contract\_Description | String | N | The Description about the current contract |
|  | Contract\_cDate | Date | Y | The create date of the contract |
|  | Contract\_sDate | Date | N | The start the contract start have effectness |
|  | Contract\_eDate | Date | N | The date the contract end lost effectness |

#### Customer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Attribute | Type | Mandatory | Description |
| 1 | Customer\_ID | Int | Y | Customer’s ID |
| 2 | Customer\_Name | String | Y | Customer’s name |
| 3 | Customer\_Gender | String | Y | Customer’s gender |
| 4 | Customer\_Address | String | Y | Customer’s address |
| 5 | Customer\_Email | String | Y | Customer’s email |
| 6 | Customer\_Number | String | Y | Customer’s number |
| 7 | Customer\_Note | String | N | Customer’s note |
| 8 | Customer\_Other | String | N | Customer’s social media |

#### Contract Detail

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Attribute | Type | Mandatory | Description |
| 1 | ConDetail\_ID | Int | Y | Feedback’s ID |
| 2 | Contract\_ID | String | Y | Contract’s ID that relate to the contract’s detail |
| 3 | Package\_ID | Int | Y | Real time that member create feedback |
| 4 | Package\_Name | String | Y | Name of the current available package |
| 5 | Package\_Detail | String | Y | Detail of the current available package |
| 6 | Package\_Price | Int | Y | Price of the current available package |
| 7 | Package\_Note | String | N | Note about package |

#### Package

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Attribute | Type | Mandatory | Description |
|  | Package\_ID | Int | Y | package’s ID |
|  | Package\_Name | String | Y | Package’s Name |
|  | Package\_Detail | String | Y | Package’s Detail |
|  | Package\_Price | Int | Y | Package’s price |
|  | Package\_Available | Boolean | Y | Define that the package is available or not |

#### Picture

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Attribute** | **Type** | **Mandatory** | **Description** |
| 1 | Picture\_ID | Int | Y | Picrure’s ID |
| 2 | ConDetail\_ID | Int | Y | Comtract’s ID |
| 3 | Picture\_Detail | String | N | Picture’s detail |
| 4 | Picture\_Url | String | N | Picture’s Url link |

## Component Diagramコンポーネント図

[INSERT DIAGRAM HERE]

Figure 4.5 Commpoent Diagram

The **Model** manages behavior of application in terms of its problem domain including data, logic and rules of application. It takes request form controller layer and makes changes to database of the system.

The **View** provides the user interface element of the application. It takes information form model and display in a form that suitable for users.

The **Controller** receives user action and makes calls to model object.

## CRC CARDS CRC カード

CRC for Model - View – Controller

[INSERT DIAGRAM HERE]

### Modelモデル

# SOFWARE TESTING ソフトウェアテスト

## Introduction　はじめに

### Purpose　目的

This is the comprehensive test plan of the Jobs Finding System project. The purpose of this chapter describes scopes of test plan and activities which need to be taken during test process of project. It addresses the following items:

* Scope of Testing
* Requirement for Testing
* Test Strategy
* Test Resources
* Test Milestones
* Test Deliverables

### References　参照資料

| Title/File name | Author | Version |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

### Background information　背景情報

The target of testing is ensured all functions will be run correctly as SRS description. In addition, restrict maximum of defect during the using of Website. To do this target, Website have to:

* Pass all of the following stages of testing: Unit Testing, Integration Testing, System Testing and Acceptance Testing.
* Pass all of the following types of testing: Function Testing, User Interface Testing, Data and Data Integrity Testing
* Run normally in popular browsers and many types of device or display resolution.

### Scope of testing　テスト範囲

The scope of test will be limited to testing on 3 browser: Google Chrome, Mozilla Firefox, Microsoft Edge.

Functional items and non-functional items will be verified and passed by eJob development team, then be validated and approved by project manager and supervisor of JFS project.

## Constraints　制約事項

* Deadline for testing only can be met if development progress is on time.
* Working process must be reported daily to PTL.
* Have more environments should be tested: Window XP, Window 8 … and more browsers: Chrome 53, Firefox, Microsoft Edge 38 …

### Risk listリスク一覧

|  |  |  |  |
| --- | --- | --- | --- |
| No | Risk | Mitigation | Contingencies |
|  | PC have problem during test | Fix a problem | Find a other PC |
|  | Tester may be sick in test phase or may leave the project before completion | + Discussing  + Committing  + Assigning tasks appropriately | + Persuading  + Reviewing and re-planning the whole project |
|  | Scope: SRS/SAD may be changed | + Discussing  + Reviewing the whole project frequently | + Re-planning test phase  + Re-creating test case |
|  | Time of Coding phase may last too long | + Committing  + Assigning tasks | Working over time |

Table 5.2.1 Risk list

## Requirements for Test　テスト要求

### Test items　テスト頭目

|  |  |  |  |
| --- | --- | --- | --- |
| No | Name of features and functions | Outline of features and functions | Note |
|  |  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |  |
|  |  |  |
|  |  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |  |
|  |  |  |
|  |  |  |  |
|  |  |  |
|  |  |  |  |
|  |  |  |
|  |  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Table 5.3.1

### Acceptance Test Criteria　テスト受入基準

List of criteria to define what levels of test quality are sufficient to move to the next testing phase:

* Test coverage : 100%
* Successful Test coverage : 90%
* Number of defects/Weighted defects : 3~4 defects/KLOC

List of criteria which are based on to accept the products, for Test team to accept source code after Unit test of Development team:

* Number of UTC/KLOC: 40 UTC/KLOC
* Number or Weighted defects/KLOC : 3~4 defects/KLOC
* Statement coverage: 100%
* Branch coverage: 100%
* Path coverage: 100%

List of criteria which are based on to accept the products, for Integration test:

* Number of UTC/KLOC: 30 UTC/KLOC
* Number or Weighted defects/KLOC : 2~3 defects/KLOC
* Statement coverage: 100%
* Branch coverage: 100%
* Path coverage: 100%

List of criteria which are based on to accept the products, for System test:

* Number of UTC/KLOC: 20 UTC/KLOC
* Number or Weighted defects/KLOC : 5~6 defects/KLOC
* Statement coverage: 100%
* Branch coverage: 100%
* Path coverage: 100%

List of criteria which are based on to accept the products, for Acceptance test:

* Number or Weighted defects/KLOC : 1~2 defects/KLOC
* Statement coverage: 100%
* Branch coverage: 100%
* Path coverage: 100%

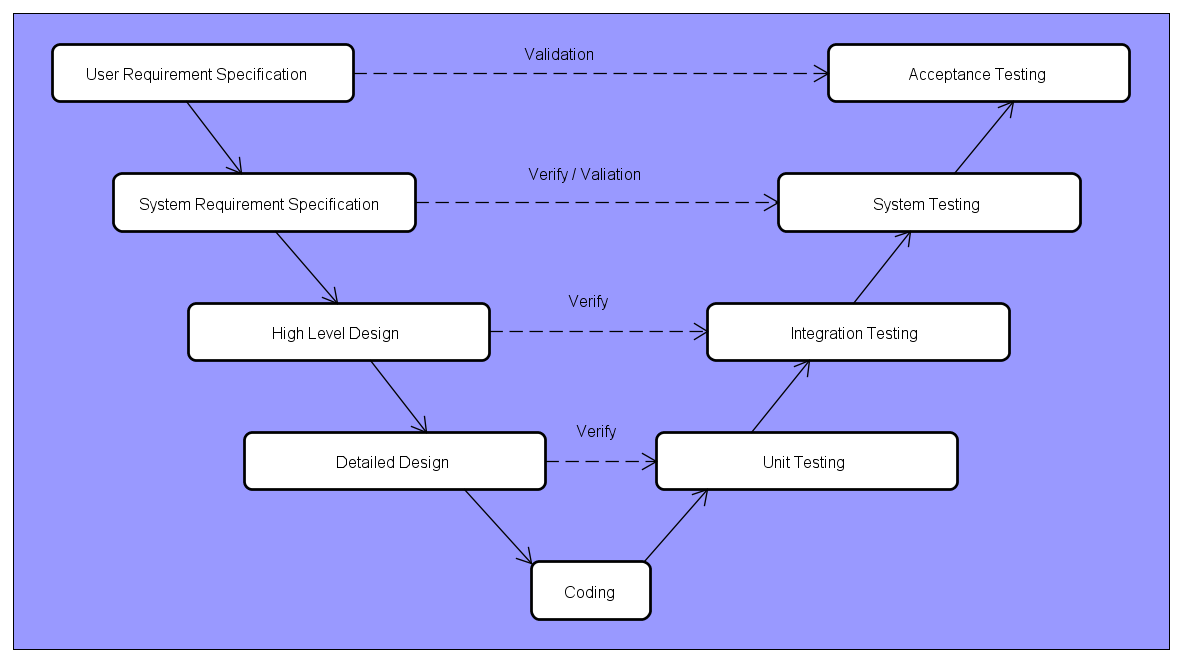
### Feature not to be tested　テストされない特徴

* The stable of website when do no connect internet.
* Manage Website’s payment activities**.** Manage the activity that user pay for system when they use paid-required service.

## Test strategy　テスト戦略

### Test Model　テストモデル

JFS follows V-Model process: The entire process of our Project is divided into 2 corresponding stages: Developing and Testing. Each developing process will be performed in parallel with the corresponding testing process. Bugs will be detected early.



JFS will be tested by 4 phases:

**Phase 1: Unit testing**

* Unit testing will be performed by developers
* Developers user White Box Testing technique to do
* When executing unit testing, if any bugs are found, developers have to log bug on “Defect Log Management” file and fix it until it is correct.

*Rule for filling test result:*

|  |  |
| --- | --- |
| Test result pass | Pass |
| Test result fail | Fail |
| Do not test | Untested |
| Cannot test | N/A (Not available) |

**Phase 2: Integration testing**

* After finishing component testing, integration testing will be performed by testers.
* Material are integration test cases, high- level design and test tools.
* Integration test focuses on specific areas of use cases when all requirements are completed.
* Integration test should be performed to ensure all components incorporate well.
* When executing integration testing, if any bugs are found, testers have to log on “Defect Log Management” file and assign to developer fix it and redo this process until it is correct.

*Rule for filling test result:*

|  |  |
| --- | --- |
| Test result pass | Pass |
| Test result fail | Fail |
| Do not test | Untested |
| Cannot test | N/A (Not available) |

**Phase 3: System testing**

* After finishing integration testing and developers collect all functions and items, testers will be performed system testing, it means doing test whole system.
* Material area system test case, SRS
* If any bugs are found, developers have to fix and testers will verify them. System test is ended only when test cases are passed and no bug is found.

*Rule for filling test result:*

|  |  |
| --- | --- |
| Test result pass | Pass |
| Test result fail | Fail |
| Do not test | Untested |
| Cannot test | N/A (Not available) |

**Phase 4: Acceptance testing**

* Acceptance testing is performed after the system testing
* Base on business/user requirement specification, system is tested again, for ensure there is not lacking or mistake any requirement.
* If there is any problem, developers have to fix/update and tester will verify them.
* Acceptance testing is ended only when whole system met requirement specification.

### Test types テストタイプ

### Function Testing　機能テスト

* Function Testing is a type of software testing whereby the system is tested against the functional requirements/ specifications.
* Functions are tested by feeding them input and examining the output. Functional testing ensures that the requirements are properly satisfied by application. This type of testing is not concerned with how processing occurs, but rather, with results of processing.
* During functional testing, **Black Box Testing** technique is used in which the internal logic of the system being tested is not known to the tester.

|  |  |
| --- | --- |
| Test Objective: | Ensure proper target-of-test functionality, including navigation, data entry, processing, and retrieval. |
| Technique: | Execute each use case, use-case flow, or function, using valid and invalid data, to verify the following:  -    The expected results occur when valid data is used.  -    The appropriate error or warning messages are displayed when invalid data is used.  -    Each business rule is properly applied.  - Use Test tool |
| Completion Criteria: | -   All planned tests have been executed.  -     All identified defects have been addressed and closed |
| Special Considerations: | Identify or describe those items or issues (internal or external) that impact the implementation and execution of function test |

Table 5.4.3 Function Testing

### User Interface Testing　ユーザーインターフェーステスト

* User interface testing, a testing technique used to identify the presence of defects is a product/software under test by using GUI.
* GUI testing evaluates design elements such as layout, colors, [fonts](http://whatis.techtarget.com/definition/font), font sizes, labels, text boxes, text formatting, captions, buttons, lists, icons, links and content.

|  |  |
| --- | --- |
| Test Objective: | Verify the following:  Navigation through the target-of-test properly reflects business functions and requirements, including window-to-window, field-to-field, and use of access methods (tab keys, mouse movements, accelerator keys)  Window objects and characteristics, such as menus, size, position, state, and focus conform to standards. |
| Technique: | Create or modify tests for each window to verify proper navigation and object states for each application window and objects. |
| Completion Criteria: | Each window successfully verified to remain consistent with benchmark version or within acceptable standard. |
| Special Considerations: | Not all properties for custom and third party objects can be accessed. |

Table 5.4.4 User Interface Testing

### Data and Database Integrity Testing　でーたおよびデータベース整合性テスト

Data integrity corresponds to the quality of data in the databases and to the level by which users examine data quality, integrity and reliability. Data integrity testing verifies that the data in the database is accurate and functions as expected within a given application.

|  |  |
| --- | --- |
| Test Objective: | Ensure database access methods and processes function properly and without data corruption. |
| Technique: | * Invoke each database access method and process, seeding each with valid and invalid data or requests for data. * Inspect the database to ensure the data has been populated as intended, all database events occurred properly, or review the returned data to ensure that the correct data was retrieved for the correct reasons |
| Completion Criteria: | All database access methods and processes function as designed and without any data corruption. |
| Special Considerations: | * Testing may require a DBMS development environment or drivers to enter or modify data directly in the databases. * Processes should be invoked manually. * Small or minimally sized databases (limited number of records) should be used to increase the visibility of any non-acceptable events. |

Table 5.4.5 Data and Database Integrity Testing

### Test stages　テストステージ

| Type of Tests | Stage of Test | | | |
| --- | --- | --- | --- | --- |
| Unit | Integration | System | Acceptance |
| Function Test | X | X | X | X |
| User Interface test | X |  | X |  |
| Date integrity test |  | X | X |  |

Table 5.4.6 Test Stages

## Resource　リソース

### Human Resource　人的リソース

|  |  |  |  |
| --- | --- | --- | --- |
| Worker/Doer | Role | Specific Responsibilities/Comments | Location |
|  |  |  | FPT University Vietnam |
|  |  |  | FPT University Vietnam |
|  |  |  | FPT University Vietnam |

Table 5.5.1 Human Resource

### Test management　テスト管理

#### Test management　テスト管理

Test management is practice of organizing and controlling the process and artifacts required for testing effort.

The general goal of test management is to allow teams to plan, develop, execute, and assess all testing activities within the overall software development effort. This includes coordinating efforts of all those involved in the testing effort, tracking dependencies and relationships among test assets and, most importantly, defining, measuring, and tracking quality goals.

* **Test planning** is the overall set of tasks that address the questions of why, what, where, and when to test. The reason why a given test is created is called a test motivator (for example, a specific requirement must be validated). What should be tested is broken down into many test cases for a project. Where to test is answered by determining and documenting the needed software and hardware configurations. When to test is resolved by tracking iterations (or cycles, or time period) to the testing.
* **Test authoring** is a process of capturing the specific steps required to complete a given test. This addresses the question of how something will be tested. This is where somewhat abstract test cases are developed into more detailed test steps, which in turn will become test scripts (either manual or automated).
* **Test execution** entails running the tests by assembling sequences of test scripts into a suite of tests. This is a continuation of answering the question of how something will be tested (more specifically, how the testing will be conducted).
* **Test reporting** is how the various results of the testing effort are analyzed and communicated. This is used to determine the current status of project testing, as well as the overall level of quality of the application or system.

### Defect management　欠陥管理

Defect management is crucial to closing the loop between requirements, implementation and verification and validation. Traditional defect tracking management, implemented in a standalone fashion, can no longer address the complexity and pace of change in modern software development. Defect management processes must be tightly interlinked with all of the other software development processes. The defect management process contains the following elements:

* **Defect Discovery** – Identification and reporting of potential defects. The defect tracking software must be simple enough so that people will use it, but ensure that the minimum necessary information is captured. The information captured here should be enough to reproduce the defect and allow development to determine root cause and impact.
* **Defect Analysis & Prioritization** – The development team determines if the defect report corresponds to an actual defect, if the defect has already been reported, and what the impact and priority of the defect is. Prioritization and scheduling of the defect resolution is often part of the overall change management process for the software development organization.
* **Defect Resolution –** Here the development team determines the root cause, implements the changes needed to fix the defect, and documents the details of the resolution in the defect management software, including suggestions on how to verify the defect is fixed. In organizations using [Software Product Lines](http://www.mks.com/challenges/product-lines) approaches, or other shared component approaches, defect resolution may need to be coordinated across multiple branches of development.
* **Defect Verification** – The build containing the resolution to the defect is identified, and testing of the build is performed to ensure the defect truly has been resolved, and that the resolution has not introduced side effects or regressions. Once all affected branches of development have been verified as resolved, the defect can be closed.
* **Defect Communication** – This encompasses automatic generation of defect metrics for management reporting and process improvement purposes, as well as visibility into the presence and status of defects across all disciplines of the software development team.

## Test environment　たテスト環境

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Screen Resolution | Browser | Purpose | Detail |
|  |  | Google Chrome, Mozilla Firefox, Microsoft Edge | Device for create and execute test | Microsoft Windows 10 Professional |
|  |  | Google Chrome, Mozilla Firefox, Internet Explorer | Device for create and execute test | Microsoft Windows 8.1 Professional |

### Hardware ハードウェア

Table 5.6.1 Hardware

### Software　ソフトウェア

|  |  |  |
| --- | --- | --- |
| Name | Purpose | Detail |
| Chrome | Executing test | Chrome 53.0 |
| Firefox | Executing test | Firefox |
| Microsoft Edge | Executing test | Microsoft Edge 38 |

# 

## Test Milestones　テストマイルストーン

|  |  |  |  |
| --- | --- | --- | --- |
| Milestone Task | Effort (pd) | Start Date | End Date |
| Create Test Plan | 6 |  |  |
| Review & update TP | 2 |  |  |
| Create Unit Test case | 2 |  |  |
| Review & update UTC | 1 |  |  |
| Create Integration Test case | 3 |  |  |
| Review & Update Integration TC | 1 |  |  |
| Create System Test case | 3 |  |  |
| Review & Update System TC | 1 |  |  |
| Execute Unit Test Iterative 1 | 3 |  |  |
| Execute Integration test Iterative 1 | 3 |  |  |
| Execute System test Iterative 1 | 3 |  |  |
| Execute Unit Test Iterative 2 | 3 |  |  |
| Execute Integration test Iterative 2 | 3 |  |  |
| Execute System test Iterative 2 | 3 |  |  |

Table 5.7 Test Milestones

## Deliverables　納品物

| No | Deliverables | Responsibilities | Delivered Date |
| --- | --- | --- | --- |
|  | Test Plan | Test Lead |  |
|  | Unit Test cases | Tester |  |
|  | Common Test cases | Tester |  |
|  | Integration Test Cases | Tester |  |
|  | System Test cases | Tester |  |
|  | Defect log Management | Tester |  |
|  | Test reports | PM |  |

Table 5.8 Deliverables

## Test Case　テストケース

Refer to:

* + PSM\_Integration Test Case\_v1.0.xlsx
  + PSM \_System Test Case\_v1.0.xlsx
  + PSM \_Common Test Case\_v1.0.xlsx
  + PSM \_UnitTestCase\_GuestCommon\_v1.0.xlsx
  + PSM \_UnitTestCase\_MemberCommon\_v1.0.xlsx
  + PSM \_UnitTestCase\_MemberManagePersonalPage\_v1.0.xlsx
  + PSM \_UnitTestCase\_MemberManagePost\_v1.0.xlsx
  + PSM \_UnitTestCase\_AdminStatistic\_v1.0.xlsx
  + PSM \_UnitTestCase\_AdminManageReports\_v1.0.xlsx
  + PSM \_UnitTestCase\_AdminManagePosts\_v1.0.xlsx
  + PSM \_UnitTestCase\_AdminManageMembers\_v1.0.xlsx
  + PSM \_UnitTestCase\_AdminManageFeedback\_v1.0.xlsx
  + PSM \_UnitTestCase\_AdminManageCategory\_v1.0.xlsx

## Test Report　テストレポート

### Unit Test Report 単体テスト

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Function code** | **Passed** | **Failed** | **Untested** | **N** | **A** | **B** | **Total Test Cases** |
| 1 |  |  |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |  |  |
|  | **Sub total** |  |  |  |  |  |  |  |

Table 5.10.1

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Function code** | **Passed** | **Failed** | **Untested** | **N** | **A** | **B** | **Total Test Cases** |
| 1 |  |  |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |  |  |
| 6 |  |  |  |  |  |  |  |  |
| 7 |  |  |  |  |  |  |  |  |
|  | **Sub total** |  |  |  |  |  |  |  |

Table 5.10.2

### Common Test Report　一般テストレポート

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No** | **Module code** | **Pass** | **Fail** | **Untested** | **N/A** | **Number of**  **test cases** |
| 1 |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |
|  | **Sub total** |  |  |  |  |  |

Table 5.10.3

### Integration Test Report　統合テストレポート

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No** | **Module code** | **Pass** | **Fail** | **Untested** | **N/A** | **Number of**  **test cases** |
| 1 |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |
|  | **Sub total** |  |  |  |  |  |

Table 5.10.4

### System Test Report　システムテストレポート

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No** | **Module code** | **Pass** | **Fail** | **Untested** | **N/A** | **Number of test cases** |
| 1 |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |
| 6 |  |  |  |  |  |  |
| 7 |  |  |  |  |  |  |
| 8 |  |  |  |  |  |  |
|  | **Sub total** |  |  |  |  |  |

Table 5.10.5

### Test Report　テストレポート

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Phase** | **Round 1** | | | | **Round 2** | | | |
| Pass | Fail | Untested | N/A | Pass | Fail | Untested | N/A |
| **Unit Test** |  |  |  |  |  |  |  |  |
| **Integration Test** |  |  |  |  |  |  |  |  |
| **System Test** |  |  |  |  |  |  |  |  |
| **Total** |  |  |  |  |  |  |  |  |

Table 5.10.5

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Defects** | **Logged** | **%Closed** | **%Invalid** | **%Open** |
| **Critical** |  |  |  |  |
| **High** |  |  |  |  |
| **Medium** |  |  |  |  |
| **Low** |  |  |  |  |
| **Total** |  |  |  |  |

Table 5.10.5.1

# USER MANUAL　ユーザーマニュアル