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

|  |
| --- |
| **FPT UNIVERSITY** |
| Capstone Project Document |
| Drop It |
|  |
| |  |  | | --- | --- | | **Group 08** | | | **Group Members** | Lê Khôi Phong  Đào Như Tùng  Công Minh Hiếu  Nguyễn Thanh Tùng  Đặng Quốc Duy | | **Supervisor** | Lâm Hữu Khánh Phương | |
| - Ho Chi Minh City, 05/2013 - |

Table of Contents

Table of Contents 3

1.1 Introduction 7

1.1.1 Project Information 7

1.1.2 Purpose 7

1.1.3 The People 7

1.1.4 Background 7

1.1.5 Problems 7

1.1.6 Our Proposal 8

1.1.7 Benefits 8

2.1 Problem Definition 9

2.1.1 Name of this Capstone Project 9

2.1.2 Problem Abstract 9

2.1.3 Project Overview 9

2.1.3.1 The Current System 9

2.1.3.2 The Proposed System 9

2.1.3.3 Boundaries of the System 10

2.1.3.4 Development Environment 10

2.2 Project organization 11

2.2.1 Software Process Model 11

2.2.2 Roles and Responsibilities 11

2.2.3 Tools and Techniques 12

2.3 Project management Plan 13

2.3.1 Tasks 13

2.3.2 Task Sheet: Assignment and Timetable 14

2.4 Coding Convention 16

2.4.1 Naming Convention 16

2.4.2 Lengths 16

3.1 User Requirement Specification 17

3.1.1 Common Features 17

3.1.2 Authorized User 17

3.1.3 Admin 17

3.1.4 Guest 17

3.2 System Requirement Specification 17

3.2.1 External Interface Requirements 17

3.2.3.1 User Interfaces 17

3.2.1.2 Hardware Interfaces 18

3.2.1.3 Software Interfaces 18

3.2.1.4 Communications Protocol 18

3.2.2 System Features 19

3.2.2.1 Admin 20

1. Create Category 21

2. Update Category 22

3. Delete Category 23

4. Hide Category 24

5. Show Category 24

6. Active User 25

7. Deactive User 26

8. Allow User Selling 27

9. Disallow User Selling 28

10. Create Event 28

11. Update Event 30

12. Delete Event 31

13. Approve Event 32

14. Disapprove Event 33

15. Close Request 34

16. Approve Ticket 35

17. Update Ticket 36

18. Disapprove Ticket 37

19. Delete Ticket 38

20. Create Venue 39

21. Update Venue 40

22. Delete Venue 41

23. Approve Venue 42

24. Disapprove Venue 43

25. Update Transaction Status 44

3.2.2.2 Customer 45

1. View Buy History 46

2. View Sell History 47

3. Create Request 48

4. Close Request 49

5. Response to Request 50

6. Follow Buy 51

7. Unfollow Buy 52

8. Follow Sell 53

9. Unfollow Sell 54

10. Follow Buy and Sell 55

11. Unfollow Buy and Sell 56

12. View Follow List 57

13. Update Transaction Status 58

14. Post Ticket 59

15. Edit Posted Ticket 60

16. Delete Posted Ticket 61

3.2.2.3 Authorized User 63

1. Change Personal Information 64

2. Change Password 65

3. View Event Detail 66

4. Search Event by Name 67

5. Logout 68

3.2.2.4 Guest 69

1. Register 70

2. View Event Detail 71

3. Search Event by Name 72

3.2.2.5 Unauthorized User 73

1. Login 74

3.2.3 Software System Attributes 75

3.2.3.1 Reliability 75

3.2.3.2 Availability 75

3.2.3.3 Security 75

3.2.3.4 Maintainability 75

3.3 Entity Relationship Diagram 76

## Introduction

### Project Information

* Project name: **Drop-It**
* Project code: **DIC2C**
* Product type: **Website**
* Timeline: **From 14 May 2013 To 14 Aug 2013**

### Purpose

This project is developed and registered as the capstone project for group number 8. The purpose is fulfilling the requirement from FPT University training program.

### The People

**Supervisor:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Full name** | **Phone** | **E-Mail** | **Title** |
| Lâm Hữu Khánh Phương | 0915353001 | phuonglhk@fpt.edu.vn | Teacher |

**Team members:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Student** | **Full name** | **Student code** | **Phone** | **E-Mail** | **Role in Group** |
| 1 | Lê Khôi Phong | 60473 | 0903045572 | phonglk60473@fpt.edu.vn | Leader |
| 2 | Đào Như Tùng | 60408 | 0909364008 | tungdn60408@fpt.edu.vn | Member |
| 3 | Công Minh Hiếu | 60535 | 0918125726 | hieucm60535@fpt.edu.vn | Member |
| 4 | Nguyễn Thanh Tùng | 60513 | 0934779004 | tungnt60513@fpt.edu.vn | Member |
| 5 | Đặng Quốc Duy | 00276 | 0943578179 | duydq00276@fpt.edu.vn | Member |

### Background

In addition to its economic growth, HCMC's cultural and artistic activities have been on the rise to form the habit of enjoying arts every night in the local residents. Together with technological demands and developments, a large number of online banking services for business transactions and ticket selling sprung up, making it more convenient for the buyers. However, whether the transactions are online or offline, there are always situations in which the ticket buyers cannot use the tickets and would have to find a way to resell the tickets to other people, often at a lower price. They can either advertise their tickets online or sell the tickets on the spot of the events.

### Problems

Most of online ticket selling businesses have these two common characteristics:

* Generality - Different types of tickets are sold and those tickets are not inclined towards any particular artistic activities.
* Most of them are B2C businesses, direct selling and are the distributors of programs to the end users.

Most of the Classified advertising websites are also general in nature and low in efficiency

Moreover, reliability and online security should be taken note of because those transactions are likely to have high risks.

### Our Proposal

DropIt is born with an aim to become a safe and convenient platform for users to resell their tickets. Although the market for second-hand tickets are smaller than that of first-hand tickets, second-hand tickets hold many potentials if DropIt can tap on to this and create a safe and anti-phish platforms.

Main Features:

* Allowing users to publish and advertise the tickets they want to resell
* Allowing users to buy second-hand tickets
* Holding money until buyer confirm that ticket is ok
* Keeping statistics and transaction records

### Benefits

* Sellers can have their tickets resold to the desired customers and are guaranteed to receive the money from the transaction.
* Buyers are able to buy their desired tickets, and only lose the money upon receiving the desired tickets.

## Problem Definition

### Name of this Capstone Project

Project Full name: **Drop It**

Project Code: **DIC2C**

### Problem Abstract

In addition to its economic growth, HCMC's cultural and artistic activities have been on the rise to form the habit of enjoying arts every night in the local residents. Together with technological demands and developments, a large number of online banking services for business transactions and ticket selling sprung up, making it more convenient for the buyers. However, whether the transactions are online or offline, there are always situations in which the ticket buyers cannot use the tickets and would have to find a way to resell the tickets to other people, often at a lower price. They can either advertise their tickets online or sell the tickets on the spot of the events.

### Project Overview

#### The Current System

Most of online ticket selling businesses have these two common characteristics:

* Generality - Different types of tickets are sold and those tickets are not inclined towards any particular artistic activities.
* Most of them are B2C businesses, direct selling and are the distributors of programs to the end users.
* Most of the Classified advertising websites are also general in nature and low in efficiency
* Moreover, reliability and online security should be taken note of because those transactions are likely to have high risks.

#### The Proposed System

DropIt is born with an aim to become a safe and convenient platform for users to resell their tickets. Although the market for second-hand tickets are smaller than that of first-hand tickets, second-hand tickets hold many potentials if DropIt can tap on to this and create a safe and anti-phish platforms.

Main Features:

* Allowing users to publish and advertise the tickets they want to resell
* Allowing users to buy second-hand tickets
* Holding money until buyer confirm that ticket is ok
* Keeping statistics and transaction records

#### Boundaries of the System

The system under development of this Capstone Project will include:

* Based on web application
* Real payment is not included in this version
* Real event also is not included in this version

#### Development Environment

Below is the list of hardware and software requirements needed for development environments:

**Hardware requirements:**

* Personal computers for developing with the minimum configuration: CPU Core 2 Duo 2.0GHz, 2GB of RAM, 30GB of hard disk, and internet.

**Software requirements:**

* Operating system: Windows 7
* Web Server: IIS
  + Microsoft Windows 7: operating system and platform for development
  + Microsoft SQL Server 2012 Express: used to create and manage the database for system
  + Assembla: used to control source code ,documents and task management project
  + IIS: web server
  + Idea Software Modeler: used to create models and diagrams
  + Microsoft Project 2010: used to manage process and work schedules.
  + Skype: used for communication and meeting
  + DBMS: Microsoft SQL Server 2012
* Source Control: Tortoise Subversion (SVN) and Assembla code server

## Project organization

### Software Process Model



**Figure 2.2.1.1 - Agile Development Model**

All of the phases in the implemented modal are included in the scope of this project and are the responsibilities of the team.

### Roles and Responsibilities

|  |  |  |  |
| --- | --- | --- | --- |
| No | Full name | Role in Group | Responsibilities |
| 1 | Lam Huu Khanh Phuong | Supervisor | * Tracking & managing progress * Advising Idea & solutions * Suggesting & supporting in technologies |
| 2 | Le Khoi Phong | Team Leader | * Tracking & managing progress * Designing database * Creating coding framework * Analyzing requirements * Planning & scheduling * Coding * GUI Design * Testing * Writing documents & reports |
| 3 | Dao Nhu Tung | Team member | * Analyzing requirements * Coding * Writing documents * Testing |
| 4 | Cong Minh Hieu | Team member | * Analyzing requirements * Coding * Writing documents * Testing |
| 5 | Nguyen Thanh Tung | Team member | * Analyzing requirements * Coding * Writing documents * Testing |
| 6 | Dang Quoc Duy | Team member | * Analyzing requirements * Coding * Writing documents * Testing |

Table 1: Roles and Responsibilities

### Tools and Techniques

**Tools:**

* *Microsoft Visual Studio 2012*: Used to implement software modules.
* *Microsoft SQL server 2008 R2 Express*: Used as the database of the system.
* *Microsoft Excel*: For the team leader to manage tasks of the members and the progress of the project.
* *Microsoft Project 2010*: for team tracking
* *TortoiseSVN*: Control Source code of the whole project.
* *VisualSVN*: extension for using subversion (SVN) inside Visual Studio.
* *Assembla:* SVN Repository
* *Idea Software Modeler:* Draw Diagram and Use case
* *moqups.com:* for prototyping
* *Crystal Report 13:*for reporting function
* *Google Chrome, Firefox*: Used to test the system

**Technologies:**

* ASP.NET MVC 4
* LINQ
* HTML 5, CSS 3, AJAX, jQuery, Knockout, Lesscss, coffescript, Bootstrap

## Project management Plan

### Tasks

|  |  |
| --- | --- |
| **Scope Study and Technology** | |
| Description | General requirements analysis, technology & business process study |
| Deliverables | The feasibility report and decisions for the project |
| Resources Needed | 25 man-days |
| Dependencies and Constraints | N/A |
| Risks | * The project or the chosen technology is not feasible. * Team members don’t |
| **Documentation and review** | |
| Description | Create all the necessary documents for research and delivery |
| Deliverables | 1. Project Management Plan (PMP) 2. Software Requirements Specification (SRS) 3. Create Software Architecture Design (SAD) 4. Software Design Description (SDD) 5. Software Test Documentation (STD) 6. Software User’s Manual (SUM) |
| Resources Needed | FPT templates,  75 man-day |
| Dependencies and Constraints | Follow FPT templates |
| Risks | * Not follow FPT Templates * Team members lack experience in creating documents, * Bootle-Neck in review because lack of resources that have experience in review * Requirements are changing so quickly, CR appears |
| **GUI design and implementation** | |
| Description | Design user interface |
| Deliverables | Prototype (in HTML & PNG format) |
| Resources Needed | 25 man-days |
| Dependencies and Constraints | Web Application |
| Risks | * CR appears frequently * Lack of Desinger and HCI experience * Not all team members can design front-end (2/4 Members can design) |
| **Implementation** | |
| Description | Create the executable files |
| Deliverables | Solution (SLN) with all needed project files. |
| Resources Needed | Visual Studio 2012, .NET framework 4.5, Web browsers  200 man-days |
| Dependencies and Constraints | * All Teammembers should understand their module precisely * All Teammember have to experience the MVC4.0, and EF5 |
| Risks | * Team members don’t have experience with Technologies * Some function and modules are underestimate so can keep the deadline * Coding not follow conventions. |
| **Release and deployment** | |
| Description | Release the complete application and deploy it on the server |
| Deliverables | DropIt installation file and all the related documents (SUM) |
| Resources Needed | Installation package  10 man-days |
| Dependencies and Constraints | Meet the user requirements |
| Risks | * Can’t baseline and release on time |
| **Quality control** | |
| Description | Testing application’s performance and usability |
| Deliverables | STD |
| Resources Needed | QA, testers (team member),members), FPT template test case  75 man-days |
| Dependencies and Constraints | Follow FPT template test case |
| Risks | * Bug rate is very high * Some bugs can detect from Implementation by Perform UT not by IT * Test cases doesn’t meet the LOCs |
| **Human resource management** | |
| Description | Manage human resource, task assignments and member’s performances |
| Deliverables | Project Task List – Assignment Table sheet |
| Resources Needed | Project Task List |
| Dependencies and Constraints | N/A |
| Risks | * First time teamwork together * Communication Risk |

### Task Sheet: Assignment and Timetable

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Task Name | Duration | Start | Finish | Resource Names |
| **Initiating** | **0.5 days** | **Fri 5/3/13** | **Fri 5/3/13** |  |
| **Planning** | **2 days** | **Sat 5/4/13** | **Tue 5/7/13** |  |
| Prepare project introduction | 1 day | Sat 5/4/13 | Sat 5/4/13 |  |
| Develop Software Management Plan | 2 days | Sun 5/5/13 | Mon 5/6/13 |  |
| Review Project Plan | 1 day | Tue 5/7/13 | Tue 5/7/13 |  |
| **Analyzing** | **7 days** | **Wed 5/8/13** | **Thu 5/16/13** |  |
| Define Problem | 2 days | Wed 5/8/13 | Thu 5/9/13 |  |
| Analyzing Ticket.st | 5 days | Fri 5/10/13 | Thu 5/16/13 |  |
| **Mockup & HTML** | **10 days** | **Fri 5/17/13** | **Thu 5/30/13** |  |
| Seller's functions mockup | 3 days | Fri 5/17/13 | Tue 5/21/13 | N.Tùng |
| Buyer's functions mockup | 3 days | Fri 5/17/13 | Tue 5/21/13 | Hiếu |
| User's function mockup | 3 days | Fri 5/17/13 | Tue 5/21/13 | T.Tùng |
| Seller's functions HTML | 7 days | Wed 5/22/13 | Thu 5/30/13 | N.Tùng,Phong |
| Buyer's functions HTML | 7 days | Wed 5/22/13 | Thu 5/30/13 | Hiếu,Phong |
| User's function HTML | 7 days | Wed 5/22/13 | Thu 5/30/13 | T.Tùng |
| **Design** | **11 days** | **Wed 5/22/13** | **Thu 6/6/13** |  |
| ERD Design | 0 days | Wed 5/22/13 | Wed 5/22/13 | N.Tùng |
| Database design | 7 days | Thu 5/23/13 | Fri 5/31/13 | Team |
| Graphic design | 11 days | Thu 5/23/13 | Thu 6/6/13 | Phong |
| **Coding** | **20 days** | **Mon 6/3/13** | **Fri 6/28/13** |  |
| **Buyer's functions** | **20 days** | Mon 6/3/13 | **Fri 6/28/13** | **Hiếu,Phong** |
| Create Basic Design (Architecture Design) | 6 days | Mon 6/3/13 | Mon 6/10/13 |  |
| Review Basic Design | 2 days | Mon 6/10/13 | Tue 6/11/13 |  |
| Create Detail Design | 4 days | Wed 6/12/13 | Mon 6/17/13 |  |
| Review Detail Design | 1 day | Tue 6/18/13 | Tue 6/18/13 |  |
| Coding | 5 days | Wed 6/19/13 | Tue 6/25/13 |  |
| Coding Review | 1 day | Wed 6/26/13 | Wed 6/26/13 |  |
| Unit Testing | 2 days | Thu 6/27/13 | Fri 6/28/13 |  |
| **Seller's functions** | **20 days** | **Mon 6/3/13** | **Fri 6/28/13** | **N.Tùng,Phong** |
| Create Basic Design (Architecture Design) | 6 days | Mon 6/3/13 | Mon 6/10/13 |  |
| Review Basic Design | 2 days | Mon 6/10/13 | Tue 6/11/13 |  |
| Create Detail Design | 4 days | Wed 6/12/13 | Mon 6/17/13 |  |
| Review Detail Design | 1 day | Tue 6/18/13 | Tue 6/18/13 |  |
| Coding | 5 days | Wed 6/19/13 | Tue 6/25/13 |  |
| Coding Review | 1 day | Wed 6/26/13 | Wed 6/26/13 |  |
| Unit Testing | 2 days | Thu 6/27/13 | Fri 6/28/13 |  |
| **User's functions** | **20 days** | **Mon 6/3/13** | **Fri 6/28/13** | **T.Tùng,Duy,Phong** |
| Create Basic Design (Architecture Design) | 6 days | Mon 6/3/13 | Mon 6/10/13 |  |
| Review Basic Design | 2 days | Mon 6/10/13 | Tue 6/11/13 |  |
| Create Detail Design | 4 days | Wed 6/12/13 | Mon 6/17/13 |  |
| Review Detail Design | 1 day | Tue 6/18/13 | Tue 6/18/13 |  |
| Coding | 5 days | Wed 6/19/13 | Tue 6/25/13 |  |
| Coding Review | 1 day | Wed 6/26/13 | Wed 6/26/13 |  |
| Unit Testing | 2 days | Thu 6/27/13 | Fri 6/28/13 |  |
| **Testing** | **30 days** | **Mon 7/1/13** | **Fri 8/9/13** | **Team** |
| Project Scope Studying | 5 days | Mon 7/1/13 | Fri 7/5/13 |  |
| Create Test View Point Document | 5 days | Sun 7/14/13 | Thu 7/18/13 |  |
| Review Test Viewpoint Document | 1 day | Fri 7/19/13 | Fri 7/19/13 |  |
| Create Integration Test Case Document | 7 days | Sat 7/20/13 | Mon 7/29/13 |  |
| Review Integration Test Case Document | 1 day | Tue 7/30/13 | Tue 7/30/13 |  |
| Perform Test | 7 days | Wed 7/31/13 | Thu 8/8/13 |  |
| Review Test Report | 1 day | Thu 8/8/13 | Thu 8/8/13 |  |
| Deliverable: Test Report | 0 days | Fri 8/9/13 | Fri 8/9/13 |  |
| Monitoring and Controlling | 57.04 days | Wed 5/22/13 | Fri 8/9/13 | Phong |

## Coding Convention

### Naming Convention

* Private Fields: underscore followed by lowerCamelCase. ( ex: int \_privateField)
* Non-private Fields and properties: UpperCamelCase. (ex: int PublicField)
* Local variables: lowerCamelCase. (ex: int localVariable)
* Do not use consecutive underscores in name.
* Do not use Hungarian style.

### Lengths

* Public name should not be longer than 32 characters or 7 words.
* Methods should contain no more than 70 lines of code (if it is, it must be divided into methods).
* Methods should contain no more than 5 levels of indentation (if it is, it must be divided into methods).
* A line of code should contain no more than 80 characters.

## User Requirement Specification

### Common Features

* Only authenticated users can access the system. Users can log in and log out using their own accounts.
* Users can change their password.
* All guests or users can view event, artist. And they can search for the event or artist, and view ticket or request ticket.
* Only authorized users can use specific functions of the system.

### Authorized User

* Users can follow category (they’re interested in their own preference).
* Users can view history their transaction, and claim for transaction (in 5 days from transaction’s day).
* If user wants to sell a ticket, he/she have to provide their bank account, and must be approved by admin.
* Show the amount of money the seller receives after subtracting 7% for website transaction fee and round up/ down amount of money.
* When user creates a request or ticket, he/she can choose default event of website or he/she can create new event.
* User is rated theirs prestige by other users.

### Admin

* Admin have to manage category, event, user, transaction, venue, request, tickets.

### Guest

* They only can view and search but they can nott use authorized user’ function.

## System Requirement Specification

### External Interface Requirements

#### User Interfaces

* The design should be simple and user-friendly. White and dark will be 2 main colors of the website while charts may have more color to visually express data more effectively.
* The menu bar should be on the left and not take too much space of the screen. It consists of the list of main functions of the system that users can access.
* The design should be responsive. It means that the web components should be scaled according to a range of resolutions and devices to provide a consistent experience, no matter what.

#### Hardware Interfaces

* To access to the system, users only need any type of computer, tablet, or mobile phone with a fair internet connection.

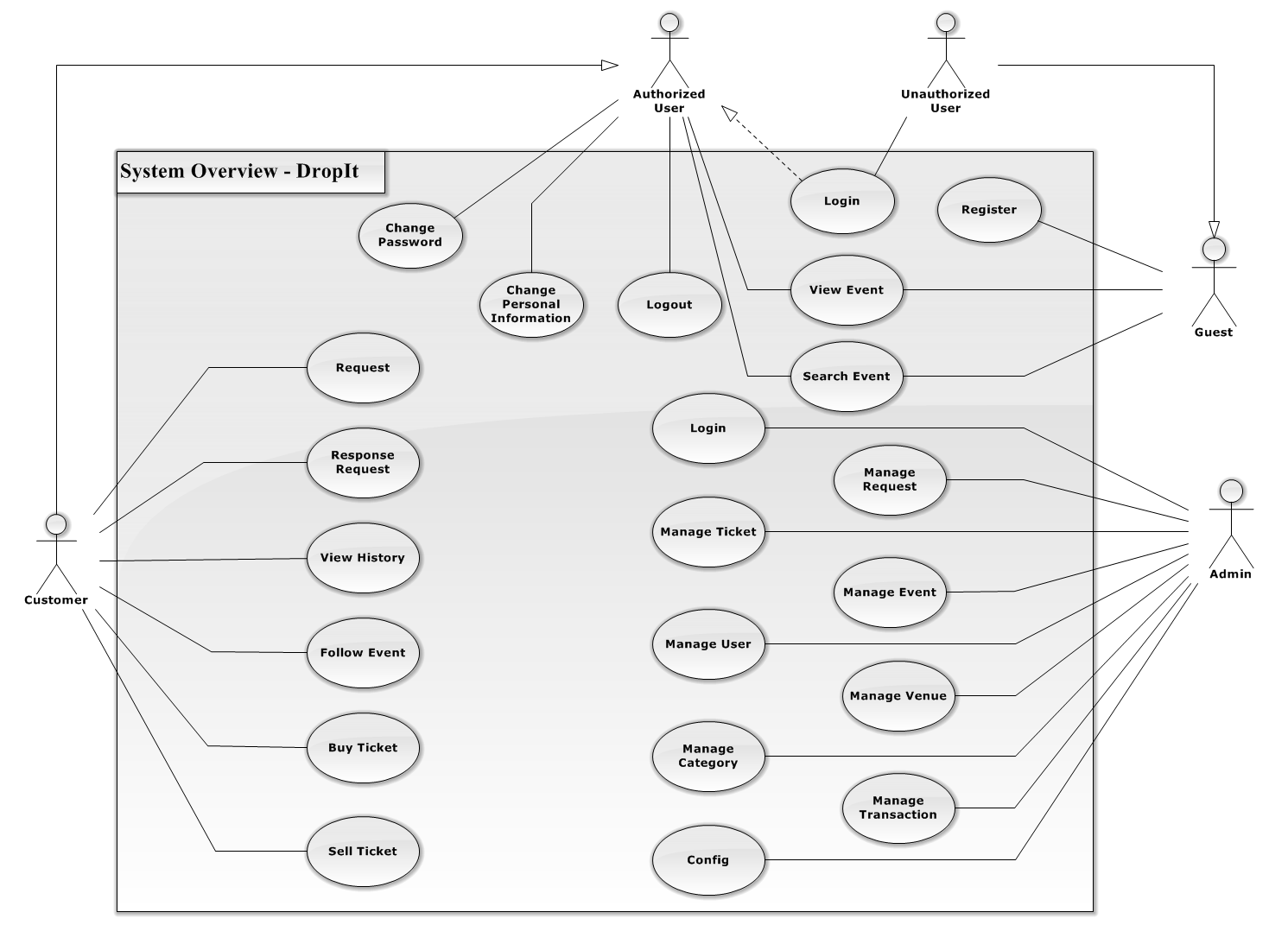
#### Software Interfaces

* At the server side, the system should run on top of Windows 7, Windows Server 2012 or later versions of Windows Server. Besides, Microsoft .NET Framework 4.5 and MVC 3 should be installed on the server. The database management system use for SMDH is SQL Server 2012.
* At the client side, users can use any modern browser that supports JavaScript and HTML 5 to access to the system.

#### Communications Protocol

HTTP is the protocol used for loading the web site in browsers.

### System Features



**Figure 3.2.2.1 - General Use Case Diagram**

#### Admin



**Figure 2 - Admin Overview Diagram**

##### **3.2.2.1.1 Create Category**



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – CREATE CATEGORY | | | |
| Use-case No. | UC01.1 | Use-case Version | 1.0 |
| Use-case Name | Create category | | |
| Author | TungNT | | |
| Date | 06/01/2013 | Priority | Normal |
| **Actor:** Admin  **Summary:** This use case is about how to create category  **Goal:** Create category successful  **Triggers:** N/A  **Pre-conditions:** N/A  **Post-conditions:** Category is created successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Danh mục” tab | Navigate to Manage Category page | | 2 | Click “Tạo danh mục” button | Show create new category page | | 3 | Fill mandatory fields and click “Tạo mới” button | - Validate, check conflict and save data  - Navigate to manage Category page |   **Alternative Scenario:**  N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Same as Main Success Scenario | | | 2 | Same as Main Success Scenario | | | 3 | Does not fill all required field or enters invalid data | Display warning message to notify the missing field or invalid data |   **Relationships:**  N/A  **Business Rules:** N/A | | | |

##### **Update Category**



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – UPDATE CATEGORY | | | |
| Use-case No. | UC01.2 | Use-case Version | 1.0 |
| Use-case Name | Update category | | |
| Author | TungNT | | |
| Date | 06/01/2013 | Priority | Normal |
| **Actor:** Admin  **Summary:** This use case is about how to update category after created  **Goal:** Category is updated  **Triggers:** N/A  **Pre-conditions:** Category is already created  **Post-conditions:** Update category successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Danh mục” tab | Navigate to Manage Category page | | 2 | Click on “Hiển thị” tab | Navigate to list of shown Category | | 3 | Click on a category | Show update category form | | 4 | Fill mandatory fields and click “Cập nhật” button | Does validation, checks conflict and save data  Navigate back to manage Category page |   **Alternative Scenario:**  N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Same as Main Success Scenario | | | 2 | Same as Main Success Scenario | | | 3 | Same as Main Success Scenario | | | 4 | Does not fill all required field or enters invalid data | Display warning message to notify the missing field or invalid data |   **Relationships:**  N/A  **Business Rules:** N/A | | | |

##### **Delete Category**



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – DELETE CATEGORY | | | |
| Use-case No. | UC01.3 | Use-case Version | 1.0 |
| Use-case Name | Delete category | | |
| Author | TungNT | | |
| Date | 06/01/2013 | Priority | Normal |
| **Actor:** Admin  **Summary:** This use case is about how to delete category after created  **Goal:** Category is deleted  **Triggers:** N/A  **Pre-conditions:** N/A  **Post-conditions:** Delete category successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Danh mục” tab | Navigate to Manage Category page | | 2 | Choose Category and click “Xóa” button | Show confirm message | | 3 | Click “Yes” button | Selected category is deleted successful |   **Alternative Scenario:**  N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | |

##### **Hide Category**



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – HIDE CATEGORY | | | |
| Use-case No. | UC01.4 | Use-case Version | 1.0 |
| Use-case Name | Hide category | | |
| Author | TungNT | | |
| Date | 06/01/2013 | Priority | Normal |
| **Actor:** Admin  **Summary:** This use case is about how to hide category after created  **Goal:** Category is hidden  **Triggers:** N/A  **Pre-conditions:** Category is shown  **Post-conditions:** Hide category successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Danh mục” tab | Navigate to Manage Category page | | 2 | Choose Category in shown category and click “Ẩn” button | Selected category is hidden successfully |   **Alternative Scenario:**  N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | |

##### **Show Category**



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – SHOW CATEGORY | | | |
| Use-case No. | UC01.5 | Use-case Version | 1.0 |
| Use-case Name | Show category | | |
| Author | TungNT | | |
| Date | 06/01/2013 | Priority | Normal |
| **Actor:** Admin  **Summary:** This use case is about how to show category after hidden  **Goal:** Category is shown  **Triggers:** N/A  **Pre-conditions:** Category is hidden  **Post-conditions:** Show category successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Danh mục” tab | Navigate to Manage Category page | | 2 | Choose Category in hidden category and click “Hiện” button | Selected category is shown successfully |   **Alternative Scenario:**  N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | |

##### **Active User**



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – INSERT CATEGORY | | | |
| Use-case No. | UC01.6 | Use-case Version | 1.0 |
| Use-case Name | Active user | | |
| Author | TungNT | | |
| Date | 06/01/2013 | Priority | Normal |
| **Actor:** Admin  **Summary:** This use case is about how to active a user  **Goal:** Active user successful  **Triggers:** N/A  **Pre-conditions:** User is deactivate  **Post-conditions:** User is activate  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Người dùng” tab | Navigate to Manage User page | | 2 | Choose user want to active and tick on “Kích hoạt” box | Active user successful |   **Alternative Scenario:**  N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | |

##### **Deactive User**



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – INSERT CATEGORY | | | |
| Use-case No. | UC01.7 | Use-case Version | 1.0 |
| Use-case Name | Deactive user | | |
| Author | TungNT | | |
| Date | 06/01/2013 | Priority | Normal |
| **Actor:** Admin  **Summary:** This use case is about how to deactive a user  **Goal:** Deactive user successful  **Triggers:** N/A  **Pre-conditions:** User is activate  **Post-conditions:** User is deactivate  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “User” button | Navigate to Manage User page | | 3 | Choose user want to deactive tick on “Kích hoạt” checkbox | Deactive user successful |   **Alternative Scenario:**  N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | |

##### **Allow User Selling**



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – ALLOW USER SELLING | | | |
| Use-case No. | UC01.8 | Use-case Version | 1.0 |
| Use-case Name | Allow user selling | | |
| Author | TungNT | | |
| Date | 06/01/2013 | Priority | Normal |
| **Actor:** Admin  **Summary:** This use case is about how to allow user selling  **Goal:** Allow user selling successful  **Triggers:** N/A  **Pre-conditions:** User is disallowed to sell  **Post-conditions:** User is allowed to sell  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “User” button | Navigate to Manage User page | | 3 | Choose user want to allow selling and tick on “Được bán” checkbox | Allow user selling successful |   **Alternative Scenario:**  N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | |

##### **Disallow User Selling**



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – DISALLOW USER SELLING | | | |
| Use-case No. | UC01.9 | Use-case Version | 1.0 |
| Use-case Name | Disallow user selling | | |
| Author | TungNT | | |
| Date | 06/01/2013 | Priority | Normal |
| **Actor:** Admin  **Summary:** This use case is about how to disallow user selling  **Goal:** Disallow user selling successful  **Triggers:** N/A  **Pre-conditions:** User is allowed to sell  **Post-conditions:** User is disallowed to sell  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “User” button | Navigate to Manage User page | | 3 | Choose user want to allow selling tick on “Được bán” checkbox | Allow user selling successful |   **Alternative Scenario:**  N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | |

##### **Create Event**



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – CREATE EVENT | | | |
| Use-case No. | UC01.10 | Use-case Version | 1.0 |
| Use-case Name | Create event | | |
| Author | TungNT | | |
| Date | 06/01/2013 | Priority | Normal |
| **Actor:** Admin  **Summary:** This use case is about how to create new event  **Goal:** Create event successful  **Triggers:** N/A  **Pre-conditions:** N/A  **Post-conditions:** Event is created  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Sự kiện” tab | Navigate to Manage Event page | | 2 | Click “Tạo mới” buton | Navigate to Create new event page | | 3 | Fill mandatory fields and click “Tạo mới” button | - Does validation, check conflict and save data  - Navigate back to Event page |   **Alternative Scenario:**  N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Same as Main Success Scenario | | | 2 | Same as Main Success Scenario | | | 3 | Does not fill all required field or enters invalid data | Display warning message to notify the missing field or invalid data |   **Relationships:**  N/A  **Business Rules:** N/A | | | |

##### **Update Event**



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – UPDATE EVENT | | | |
| Use-case No. | UC01.11 | Use-case Version | 1.0 |
| Use-case Name | Update event | | |
| Author | TungNT | | |
| Date | 06/01/2013 | Priority | Normal |
| **Actor:** Admin  **Summary:** This use case is about how to update event after created  **Goal:** Event is updated  **Triggers:** N/A  **Pre-conditions:** Event is already created  **Post-conditions:** Update event successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Event” tab | Navigate to Manage Event page | | 2 | Click on an event | Show Update event page | | 3 | Fill mandatory fields and click “Cập nhật” button | - Does validation, check conflict and save data  - Navigate back to Event page |   **Alternative Scenario:**  N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Same as Main Success Scenario | | | 2 | Same as Main Success Scenario | | | 3 | Does not fill all required field or enters invalid data | Display warning message to notify the missing field or invalid data |   **Relationships:**  N/A  **Business Rules:** N/A | | | |

##### **Delete Event**



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – DELETE EVENT | | | |
| Use-case No. | UC01.12 | Use-case Version | 1.0 |
| Use-case Name | Delete event | | |
| Author | TungNT | | |
| Date | 06/01/2013 | Priority | Normal |
| **Actor:** Admin  **Summary:** This use case is about how to delete an event after created  **Goal:** Event is deleted  **Triggers:** N/A  **Pre-conditions:** Event is already disapproved  **Post-conditions:** Delete event successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Event” tab | Navigate to Manage Event page | | 2 | Click on “Chưa duyệt” tab | Show list of disapproved events | | 3 | Choose event and click “Xóa” button | Show confirm message | | 4 | Click “Yes” button | Selected event is deleted successfully |   **Alternative Scenario:**  N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | |

##### **Approve Event**



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – APPROVE EVENT | | | |
| Use-case No. | UC01.13 | Use-case Version | 1.0 |
| Use-case Name | Approve event | | |
| Author | TungNT | | |
| Date | 06/01/2013 | Priority | Normal |
| **Actor:** Admin  **Summary:** This use case is about how to approve an event after created  **Goal:** Event is approved  **Triggers:** N/A  **Pre-conditions:** Event is already disapproved  **Post-conditions:** Approved event successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Event” tab | Navigate to Manage Event page | | 2 | Click on “Chưa duyệt” tab | Show list of disapproved events | | 3 | Choose event and click “Duyệt” button | Selected event is approved successfully |   **Alternative Scenario:**  N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | |

##### **Disapprove Event**



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – DISAPPROVE EVENT | | | |
| Use-case No. | UC01.14 | Use-case Version | 1.0 |
| Use-case Name | Disapprove event | | |
| Author | TungNT | | |
| Date | 06/01/2013 | Priority | Normal |
| **Actor:** Admin  **Summary:** This use case is about how to disapprove an event after created  **Goal:** Event is disapproved  **Triggers:** N/A  **Pre-conditions:** Event is already approved  **Post-conditions:** Disapproved event successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Event” tab | Navigate to Manage Event page | | 2 | Click on “Đã duyệt” tab | Show list of approved events | | 3 | Choose event and click “Bỏ duyệt” button | Selected event is disapproved successfully |   **Alternative Scenario:**  N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | |

##### **Close Request**



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – CLOSE REQUEST | | | |
| Use-case No. | UC01.15 | Use-case Version | 1.0 |
| Use-case Name | Close request | | |
| Author | TungNT | | |
| Date | 06/01/2013 | Priority | Normal |
| **Actor:** Admin  **Summary:** This use case is about how to close request after created  **Goal:** N/A  **Triggers:** N/A  **Pre-conditions:** Request is created  **Post-conditions:** Close request successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Yêu cầu” tab | Navigate to Manage Request page | | 2 | Choose request want to close and click on “Đóng” button | Display confirm message | | 3 | Click on “Yes” button | Selected request is closed successful |   **Alternative Scenario:**  N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | |

##### **Approve Ticket**



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – APPROVE TICKET | | | |
| Use-case No. | UC01.16 | Use-case Version | 1.0 |
| Use-case Name | Approve ticket | | |
| Author | TungNT | | |
| Date | 06/01/2013 | Priority | Normal |
| **Actor:** Admin  **Summary:** This use case is about how to approve ticket  **Goal:** Approve ticket successful  **Triggers:** N/A  **Pre-conditions:** Ticket is disapproved  **Post-conditions:** Ticket is approved  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Vé” tab | Navigate to Manage ticket page | | 2 | Click on “Tạm hoãn” tab | Show list of disapproved ticket | | 3 | Choose a ticket and click on “Duyệt” button | Approve ticket successful |   **Alternative Scenario:**  N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | |

##### **Update Ticket**



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – UPDATE TICKET | | | |
| Use-case No. | UC01.17 | Use-case Version | 1.0 |
| Use-case Name | Update ticket | | |
| Author | TungNT | | |
| Date | 06/01/2013 | Date | 07/31/2013 |
| **Actor:** Admin  **Summary:** This use case is about how to update ticket  **Goal:** Update ticket successful  **Triggers:** N/A  **Pre-conditions:** N/A  **Post-conditions:** Ticket is updated  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Vé” tab | Navigate to Manage Ticket page | | 2 | Click “Chi tiết” button | Show update ticket page | | 3 | Fill mandatory fields and click “Cập nhật” button | Does validation, check conflict and save data  Navigate back to Manage Ticket page |   **Alternative Scenario:**  N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Same as Main Success Scenario | | | 2 | Same as Main Success Scenario | | | 3 | Does not fill all required field or enters invalid data | Display warning message to notify the missing field or invalid data |   **Relationships:**  N/A  **Business Rules:** N/A | | | |

##### **Disapprove Ticket**



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – DISAPPROVE TICKET | | | |
| Use-case No. | UC01.18 | Use-case Version | 1.0 |
| Use-case Name | Disapprove ticket | | |
| Author | TungNT | | |
| Date | 06/01/2013 | Priority | Normal |
| **Actor:** Admin  **Summary:** This use case is about how to disapprove ticket  **Goal:** Disapprove ticket successful  **Triggers:** N/A  **Pre-conditions:** N/A  **Post-conditions:** Ticket is disapproved  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Vé” tab | Navigate to Manage ticket page | | 2 | Click on “Sẵn sàng” tab | Show list of approved ticket | | 3 | Choose a ticket and click on “Bỏ duyệt” button | Disapprove ticket successful |   **Alternative Scenario:**  N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | |

##### **Delete Ticket**



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – DELETE TICKET | | | |
| Use-case No. | UC01.19 | Use-case Version | 1.0 |
| Use-case Name | Delete ticket | | |
| Author | TungNT | | |
| Date | 06/01/2013 | Priority | Normal |
| **Actor:** Admin  **Summary:** This use case is about how to delete ticket  **Goal:** Delete ticket successful  **Triggers:** N/A  **Pre-conditions:** N Ticket is disapproved  **Post-conditions:** Ticket is deleted  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Vé” tab | Navigate to Manage ticket page | | 2 | Click on “Tạm hoãn” tab | Show list of disapproved ticket | | 3 | Choose a ticket and click on “Xóa” button | Display confirm message | | 4 | Click “Yes” button | Delete ticket successful |   **Alternative Scenario:**  N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | |

##### **Create Venue**



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – CREATE VENUE | | | |
| Use-case No. | UC01.20 | Use-case Version | 1.0 |
| Use-case Name | Create venue | | |
| Author | TungNT | | |
| Date | 06/01/2013 | Priority | Normal |
| **Actor:** Admin  **Summary:** This use case is about how to create venue  **Goal:** Create venue successful  **Triggers:** N/A  **Pre-conditions:** N/A  **Post-conditions:** Venue is created  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Địa điểm” tab | Navigate to Manage Venue page | | 2 | Click on “Tạo địa điểm” button | Navigate to create new venue page | | 3 | Fill mandatory fields and click “Tạo mới” button | - Does validation, check conflict and save data  - Navigate back to Manage Venue page |   **Alternative Scenario:**  N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Same as Main Success Scenario | | | 2 | Same as Main Success Scenario | | | 3 | Does not fill all required field or enters invalid data | Display warning message to notify the missing field or invalid data |   **Relationships:**  N/A  **Business Rules:** N/A | | | |

##### **Update Venue**



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – UPDATE VENUE | | | |
| Use-case No. | UC01.21 | Use-case Version | 1.0 |
| Use-case Name | Update venue | | |
| Author | TungNT | | |
| Date | 06/01/2013 | Priority | Normal |
| **Actor:** Admin  **Summary:** This use case is about how to update venue  **Goal:** Update venue successful  **Triggers:** N/A  **Pre-conditions:** Venue is not in transaction  **Post-conditions:** Venue is updated  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Địa điểm” tab | Navigate to Manage Venue page | | 2 | Click on a venue | Navigate to Update Venue page | | 3 | Fill mandatory fields and click “OK” button | Does validation, check conflict and save data  Navigate back to Venue page |   **Alternative Scenario:**  N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Same as Main Success Scenario | | | 2 | Click on a venue is in transaction | Display warning message that venue is in transaction, cannot edit | | 3 | Does not fill all required field or enters invalid data | Display warning message to notify the missing field or invalid data |   **Relationships:**  N/A  **Business Rules:** N/A | | | |

##### **Delete Venue**



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – DELETE VENUE | | | |
| Use-case No. | UC01.22 | Use-case Version | 1.0 |
| Use-case Name | Delete venue | | |
| Author | TungNT | | |
| Date | 06/01/2013 | Priority | Normal |
| **Actor:** Admin  **Summary:** This use case is about how to delete venue after disapproved  **Goal:** Venue is deleted  **Triggers:** N/A  **Pre-conditions:** Venue is already disapproved  **Post-conditions:** Delete venue successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Địa điểm” tab | Navigate to Venue page | | 2 | Click on “Chưa duyệt” tab | Show list of disapproved venues | | 3 | Choose venue want to delete and click on “Xóa” button | Show confirm message | | 4 | Click “Yes” button | Selected venue is deleted successful |   **Alternative Scenario:**  N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | |

##### **Approve Venue**



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – APPROVE VENUE | | | |
| Use-case No. | UC01.23 | Use-case Version | 1.0 |
| Use-case Name | Approve venue | | |
| Author | TungNT | | |
| Date | 06/01/2013 | Priority | Normal |
| **Actor:** Admin  **Summary:** This use case is about how to approve venue  **Goal:** Venue is approved  **Triggers:** N/A  **Pre-conditions:** Venue is already disapproved  **Post-conditions:** Approve venue successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Địa điểm” tab | Navigate to Venue page | | 2 | Click on “Chưa duyệt” tab | Show list of disapproved venues | | 3 | Choose venue want to approve and click on “Duyệt” button | Approve venue successful |   **Alternative Scenario:**  N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | |

##### **Disapprove Venue**



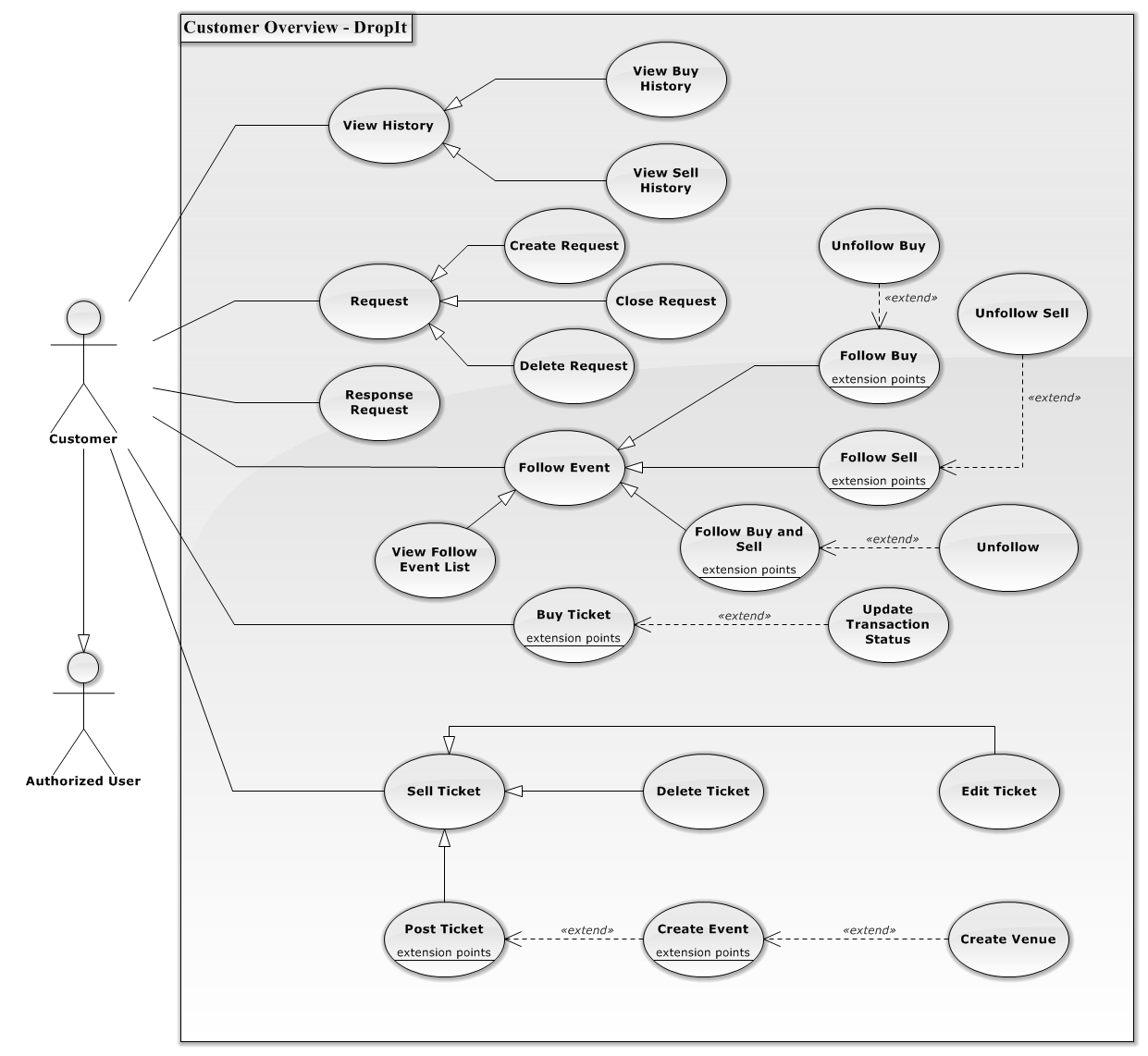
|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – DISAPPROVE VENUE | | | |
| Use-case No. | UC01.24 | Use-case Version | 1.0 |
| Use-case Name | Disapprove venue | | |
| Author | TungNT | | |
| Date | 06/01/2013 | Priority | Normal |
| **Actor:** Admin  **Summary:** This use case is about how to disapprove venue  **Goal:** Venue is approved  **Triggers:** N/A  **Pre-conditions:** Venue is already approved  **Post-conditions:** Disapprove venue successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Địa điểm” tab | Navigate to Venue page | | 2 | Click on “Đã duyệt” tab | Show list of approved venues | | 3 | Choose venue want to disapprove and click on “Bỏ duyệt” button | Disapprove venue successful |   **Alternative Scenario:**  N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | |

##### **Update Transaction Status**



|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – UPDATE TRANSACTION | | | |
| Use-case No. | UC01.25 | Use-case Version | 1.0 |
| Use-case Name | Update transaction status | | |
| Author | TungNT | | |
| Date | 06/01/2013 | Priority | Normal |
| **Actor:** Admin  **Summary:** This use case is about how to update transaction  **Goal:** Update transaction successful  **Triggers:** N/A  **Pre-conditions:** N/A  **Post-conditions:** Transaction is updated  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Giao dịch” tab | Navigate to Manage Venue page | | 2 | Click on a “Đã thanh toán” tab | Show paid transaction list | | 3 | Choose a transaction and click “Giao hàng” button | - Transaction disappeared  - Transaction status changed to delivering | | 4 | Click on a “Giao hàng” tab | Show delivering transaction list | | 5 | Choose a transaction and click “Nhận hàng” button | - Transaction disappeared  - Transaction status changed to received | | 6 | Click on a “Đã nhận hàng” tab | Show received transaction list |   **Alternative Scenario:**  N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | |

#### Customer



**Figure 3 – Customer Overview Diagram**

##### **View Buy History**

****

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE – VIEW BUY HISTORY | | | | |
| Use-case No. | UC02.01 | Use-case Version | | 1.0 |
| Use-case Name | View Buy History | | | |
| Author | DuyDQ | | | |
| Date | 06/01/2013 | Priority | Normal | |
| **Actor:** Customer  **Summary:** This use case is about how user view buy history  **Goal:** User can view buy history successful  **Triggers:** N/A  **Pre-conditions:**  User has logged into system  **Post-conditions:** View buy history successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Mua vé” tab | Navigate to Buy ticket page | | 2 | Click on “Lịch sử giao dịch” link | Navigate to Buy history page |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | | |

##### **View Sell History**

****

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE – VIEW SELL HISTORY | | | | |
| Use-case No. | UC02.02 | Use-case Version | | 1.0 |
| Use-case Name | View Sell History | | | |
| Author | DuyDQ | | | |
| Date | 06/01/2013 | Priority | Normal | |
| **Actor:** Customer  **Summary:** This use case is about how user view sell history  **Goal:** User can view sell history successful  **Triggers:** N/A  **Pre-conditions:**  User has logged into system  **Post-conditions:** View sell history successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Bán vé” tab | Navigate to Sell ticket page | | 2 | Click on “Lịch sử giao dịch” link | Navigate to Sell history page |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | | |

##### **Create Request**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE – CREATE REQUEST | | | | |
| Use-case No. | UC02.03 | Use-case Version | | 1.0 |
| Use-case Name | Create Request | | | |
| Author | DuyDQ | | | |
| Date | 06/01/2013 | Priority | Normal | |
| **Actor:** Customer  **Summary:** This use case is about how to create request  **Goal:** User can create new request successful  **Triggers:** N/A  **Pre-conditions:**  User has logged into system  **Post-conditions:** Create request successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Mua vé” tab | Navigate to Buy ticket page | | 2 | Click on “Đăng rao mua” link | Navigate to Create request page | | 3 | Fill mandatory fields and click “Đăng yêu cầu” button | Navigate to Home page |   **Alternative Scenario:** N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Same as Main Success Scenario | | | 1 | Same as Main Success Scenario | 1 | | 3 | Does not fill all required field or enters invalid data | Display warning message to notify the missing field or invalid data |   **Relationships:**  N/A  **Business Rules:** N/A | | | | |

##### **Close Request**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE – CLOSE REQUEST | | | | |
| Use-case No. | UC02.04 | Use-case Version | | 1.0 |
| Use-case Name | Close Request | | | |
| Author | DuyDQ | | | |
| Date | 06/01/2013 | Priority | Normal | |
| **Actor:** Customer  **Summary:** This use case is about how close request  **Goal:** User can close request successful  **Triggers:** N/A  **Pre-conditions:**  User has logged into system  **Post-conditions:** Close request successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Mua vé” tab | Navigate to Buy ticket page | | 2 | Click on “Quản lí rao mua” link | Navigate to Request page | | 3 | Click on “Đóng” button | Display confirm message | | 4 | Click on “OK” button | Close request successful |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | | |

##### **Response to Request**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE –RESPONSE TO REQUEST | | | | |
| Use-case No. | UC02.05 | Use-case Version | | 1.0 |
| Use-case Name | Response to request | | | |
| Author | DuyDQ | | | |
| Date | 06/01/2013 | Priority | Normal | |
| **Actor:** Customer  **Summary:** This use case is about how user response to request  **Goal:** User can response request successful  **Triggers:** N/A  **Pre-conditions:**  User has logged into system  **Post-conditions:** Response request successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Bán vé” tab | Navigate to Sell ticket page | | 2 | Click on “Danh sách rao mua” link | Navigate to Request page | | 3 | Click on “Phản hồi” button | Response to request successful |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | | |

##### **Follow Buy**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE – FOLLOW BUY | | | | |
| Use-case No. | UC02.06 | Use-case Version | | 1.0 |
| Use-case Name | Follow Buy | | | |
| Author | DuyDQ | | | |
| Date | 06/01/2013 | Priority | Normal | |
| **Actor:** Customer  **Summary:** This use case is about how user follow buy an event  **Goal:** User can follow buy event successful  **Triggers:** N/A  **Pre-conditions:**  User has logged into system  **Post-conditions:** Follow buy event successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on an event | Navigate to Event page | | 2 | Click on dropdown button next to “Theo dõi” button | Display follow dropdown list | | 3 | Click on “Theo dõi mua” button | “Theo dõi” button turns into “Bỏ theo dõi mua” button |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | | |

##### **Unfollow Buy**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE – UNFOLLOW BUY | | | | |
| Use-case No. | UC02.07 | Use-case Version | | 1.0 |
| Use-case Name | Unfollow Buy | | | |
| Author | DuyDQ | | | |
| Date | 06/01/2013 | Priority | Normal | |
| **Actor:** Customer  **Summary:** This use case is about how user unfollow buy an event  **Goal:** User can unfollow buy event successful  **Triggers:** N/A  **Pre-conditions:**  User has already follow-buy event  **Post-conditions:** Unfollow buy event successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on an event has been buy-followed | Navigate to Event page | | 2 | Click on “Bỏ theo dõi mua” button | “Bỏ theo dõi mua” button turns into “Theo dõi” button |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | | |

##### **Follow Sell**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE – FOLLOW SELL | | | | |
| Use-case No. | UC02.08 | Use-case Version | | 1.0 |
| Use-case Name | Follow Sell | | | |
| Author | DuyDQ | | | |
| Date | 06/01/2013 | Priority | Normal | |
| **Actor:** Customer  **Summary:** This use case is about how user follow sell an event  **Goal:** User can follow sell event successful  **Triggers:** N/A  **Pre-conditions:**  User has logged into system  **Post-conditions:** Follow sell event successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on an event | Navigate to Event page | | 2 | Click on dropdown button next to “Theo dõi” button | Display follow dropdown list | | 3 | Click on “Theo dõi bán” button | “Theo dõi” button turns into “Bỏ theo dõi bán” button |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | | |

##### **Unfollow Sell**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE – UNFOLLOW SELL | | | | |
| Use-case No. | UC02.09 | Use-case Version | | 1.0 |
| Use-case Name | Unfollow Sell | | | |
| Author | DuyDQ | | | |
| Date | 06/01/2013 | Priority | Normal | |
| **Actor:** Customer  **Summary:** This use case is about how user unfollow sell an event  **Goal:** User can unfollow sell event successful  **Triggers:** N/A  **Pre-conditions:**  User has already follow-sell event  **Post-conditions:** Unfollow sell event successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on an event has been buy-followed | Navigate to Event page | | 2 | Click on “Bỏ theo dõi bán” button | “Bỏ theo dõi bán” button turns into “Theo dõi” button |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | | |

##### **Follow Buy and Sell**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE – FOLLOW BUY AND SELL | | | | |
| Use-case No. | UC02.010 | Use-case Version | | 1.0 |
| Use-case Name | Follow Buy and Sell | | | |
| Author | DuyDQ | | | |
| Date | 06/01/2013 | Priority | Normal | |
| **Actor:** Customer  **Summary:** This use case is about how user follow buy and sell an event  **Goal:** User can follow buy and sell event successful  **Triggers:** N/A  **Pre-conditions:**  User has logged into system  **Post-conditions:** Follow buy and sell event successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on an event | Navigate to Event page | | 2 | Click on “Theo dõi” button | “Theo dõi” button turns into “Bỏ theo dõi” button |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | | |

##### **Unfollow Buy and Sell**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE – UNFOLLOW BUY AND SELL | | | | |
| Use-case No. | UC02.11 | Use-case Version | | 1.0 |
| Use-case Name | Unfollow Buy and Sell | | | |
| Author | DuyDQ | | | |
| Date | 06/01/2013 | Priority | Normal | |
| **Actor:** Customer  **Summary:** This use case is about how user unfollow buy and sell an event  **Goal:** User can unfollow buy and sell event successful  **Triggers:** N/A  **Pre-conditions:**  User has already follow-buy-and-sell event  **Post-conditions:** Unfollow buy and sell event successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on an event has been buy-followed | Navigate to Event page | | 2 | Click on “Bỏ theo dõi” button | “Bỏ theo dõi” button turns into “Theo dõi” button |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | | |

##### **View Follow List**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE – VIEW FOLLOW EVENT LIST | | | | |
| Use-case No. | UC02.12 | Use-case Version | | 1.0 |
| Use-case Name | View Follow Event List | | | |
| Author | DuyDQ | | | |
| Date | 06/01/2013 | Priority | Normal | |
| **Actor:** Customer  **Summary:** This use case is about how user view list of follow events  **Goal:** User can view list of follow events successful  **Triggers:** N/A  **Pre-conditions:**  User has logged into system  **Post-conditions:** View list of follow events successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Bán vé” or “Mua vé” tab | Navigate to corresponding page | | 2 | Click on “Danh sách theo dõi” link | Navigate to List follow event page |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | | |

##### **Update Transaction Status**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE – UPDATE TRANSACTION STATUS | | | | |
| Use-case No. | UC02.13 | Use-case Version | | 1.0 |
| Use-case Name | Update Transaction Status | | | |
| Author | DuyDQ | | | |
| Date | 06/01/2013 | Priority | Normal | |
| **Actor:** Customer  **Summary:** This use case is about how user updates status of transaction  **Goal:** User can change status transaction successful  **Triggers:** click “lịch sử giao dịch” tab at the left side  **Pre-conditions:**  User login into system already.  **Post-conditions:** N/A  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Select “lịch sử giao dịch” tab at the left side | System will display list of transaction of user in database. | | 2 | Click “chi tiết” button of a transaction | System will show detail transaction | | 3 | Choose status in text box | System will validate input. | | 4 | Click “Lưu” | System will validate, save to database, and change status of transaction. |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | | |

##### **Post Ticket**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE – POST TICKET | | | | |
| Use-case No. | UC02.14 | Use-case Version | | 1.0 |
| Use-case Name | Post Ticket | | | |
| Author | DuyDQ | | | |
| Date | 06/01/2013 | Priority | Normal | |
| **Actor:** Customer  **Summary:** This use case is about how user posts a ticket  **Goal:** User can post ticket successful  **Triggers:** N/A  **Pre-conditions:**  User has logged into system  **Post-conditions:** Ticket is posted successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Bán vé” tab | Navigate to Sell ticket page | | 2 | Click “Bán vé” link | Navigate to Create ticket page | | 3 | Fill mandatory fields and click “Đăng vé” button | - Does validation, check conflict and save data  - Navigate back to Home page |   **Alternative Scenario:** N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Same as Main Success Scenario | | | 2 | Same as Main Success Scenario | | | 3 | Does not fill all required field or enters invalid data | Display warning message to notify the missing field or invalid data |   **Relationships:**  N/A  **Business Rules:** N/A | | | | |

##### **Edit Posted Ticket**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE – EDIT POSTED TICKET | | | | |
| Use-case No. | UC02.15 | Use-case Version | | 1.0 |
| Use-case Name | Edit Posted Ticket | | | |
| Author | DuyDQ | | | |
| Date | 06/01/2013 | Priority | Normal | |
| **Actor:** Customer  **Summary:** This use case is about how user edit ticket information  **Goal:** User can edit ticket information successful  **Triggers:** N/A  **Pre-conditions:**  Ticket want to edit is not in any transaction  **Post-conditions:** Ticket is updated successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Bán vé” tab | Navigate to Sell ticket page | | 2 | Click on “Danh sách rao bán” link | Navigate to List posted ticket page | | 3 | Click on a posted ticket | Display ticket information | | 4 | Click on “Chỉnh sửa” button | Navigate to Edit ticket page | | 5 | Fill mandatory fields and click “Cập nhật” button | - Does validation, check conflict and save data  - Navigate back to Home page |   **Alternative Scenario:** N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | ystem Response | | 1 | Same as Main Success Scenario | | | 2 | Same as Main Success Scenario | | | 3 | Same as Main Success Scenario | | | 4 | Same as Main Success Scenario | | | 5 | Does not fill all required field or enters invalid data | Display warning message to notify the missing field or invalid data |   **Relationships:**  N/A  **Business Rules:** N/A | | | | |

##### **Delete Posted Ticket**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE – DELETE POST TICKET | | | | |
| Use-case No. | UC02.16 | Use-case Version | | 1.0 |
| Use-case Name | Delete Posted Ticket | | | |
| Author | DuyDQ | | | |
| Date | 06/01/2013 | Priority | Normal | |
| **Actor:** Customer  **Summary:** This use case is about how user delete a posted ticket  **Goal:** User can delete a sell ticket successful  **Triggers:** N/A  **Pre-conditions:**  Ticket want to delete is not in any transaction  **Post-conditions:** N/A.  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Bán vé” tab | Navigate to Sell ticket page | | 2 | Click on “Danh sách rao bán” link | Navigate to List posted ticket page | | 3 | Click on a posted ticket | Display ticket information | | 4 | Click on “Xóa” button | Show confirm message | | 5 | Click “Yes” button | Ticket is deleted successful |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | | |

#### Authorized User



**Figure 4 – Authorized User Overview Diagram**

##### **Change Personal Information**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE – CHANGE PERSONAL INFORMATION | | | | |
| Use-case No. | UC03.2 | Use-case Version | | 1.0 |
| Use-case Name | Change Personal Information | | | |
| Author | DuyDQ | | | |
| Date | 06/01/2013 | Priority | Normal | |
| **Actor:** Customer, Authorized User  **Summary:** This use case is about how user change personal information  **Goal:** User can change personal information successful  **Triggers:** N/A  **Pre-conditions:**  User has logged into system  **Post-conditions:** User’s information is changed successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Bán vé” or “Mua vé” tab | Navigate to corresponding page | | 2 | Click on “Thông tin tài khoản” link | Navigate to User information page | | 3 | Fill mandatory fields and click “Xác nhận” button | - Does validation, check conflict and save data  - Navigate back to Home page |   **Alternative Scenario:** N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | Systm Response | | 1 | Same as Main Success Scenario | | | 2 | Same as Main Success Scenario | | | 3 | Does not fill all required field or enters invalid data | Display warning message to notify the missing field or invalid data |   **Relationships:**  N/A  **Business Rules:** N/A | | | | |

##### **Change Password**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE – CHANGE PASSWORD | | | | |
| Use-case No. | UC03.3 | Use-case Version | | 1.0 |
| Use-case Name | Change Password | | | |
| Author | DuyDQ | | | |
| Date | 06/01/2013 | Priority | Normal | |
| **Actor:** Customer, Authorized User  **Summary:** This use case is about how user change password  **Goal:** User can change password successful  **Triggers:** N/A  **Pre-conditions:**  User has logged into system  **Post-conditions:** User’s password is changed successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on user name at the top right | Navigate to change password page | | 2 | Fill mandatory fields and click “Xác nhận” button | - Does validation, check conflict and save data  - Navigate to change password page with success message at top of page |   **Alternative Scenario:** N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Same as Main Success Scenario | | | 2 | Same a Main Success Scenario | | | 3 | Does not fill all required field or enters invalid data | Display warning message to notify the missing field or invalid data |   **Relationships:**  N/A  **Business Rules:** N/A | | | | |

##### **View Event Detail** C:\Users\duydq00276\Desktop\usecase\GuestOverview\ViewEventDetail.png

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE – VIEW EVENT DETAIL | | | | |
| Use-case No. | UC03.03 | Use-case Version | | 1.0 |
| Use-case Name | View Event Detail | | | |
| Author | DuyDQ | | | |
| Date | 01/06/2013 | Priority | Normal | |
| **Actor:** Guest, Admin, User, Authorized User, Unauthorized User  **Summary:** This use case is about how actor views event detail  **Goal:** Actor can view event detail successful  **Triggers:** N/A  **Pre-conditions:** N/A  **Post-conditions:** N/A  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on an event | Navigate to Event page |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | | |

##### **Search Event by Name**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE – SEARCH EVENT BY NAME | | | | |
| Use-case No. | UC03.04 | Use-case Version | | 1.0 |
| Use-case Name | Search Event by Name | | | |
| Author | DuyDQ | | | |
| Date | 01/06/2013 | Priority | Normal | |
| **Actor:** Admin, User, Authorized User, Guest, Unauthorized User  **Summary:** This use case is about how actor search event by name  **Goal:** Actor can search event successful  **Triggers:** N/A  **Pre-conditions:** N/A  **Post-conditions:** N/A  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Input information want to find in text box | Make suggestion when actor input more than 1 character | | 2 | Click on search label or press Enter key | Navigate to search result page |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | | |

##### **Logout**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE – LOGOUT | | | | |
| Use-case No. | UC03.05 | Use-case Version | | 1.0 |
| Use-case Name | Logout | | | |
| Author | DuyDQ | | | |
| Date | 01/06/2013 | Priority | Normal | |
| **Actor:** Admin, Customer, Authorized User  **Summary:** This use case is about how user change personal information  **Goal:** Logout from system successful  **Triggers:** N/A  **Pre-conditions:** User has logged into system  **Post-conditions:** Logout successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on the “Đăng Xuất” button at the top right | - Remove user’s session  - Navigate to Home page |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | | |

#### Guest

****

Figure 5 – Guest Overview Diagram

##### **Register**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE – REGISTER | | | | |
| Use-case No. | UC03.01 | Use-case Version | | 1.0 |
| Use-case Name | Register | | | |
| Author | DuyDQ | | | |
| Date | 01/06/2013 | Priority | Normal | |
| **Actor:** Guest  **Summary:** This use case is about how guest register to website  **Goal:** Register successful  **Triggers:** N/A  **Pre-conditions:** Register with username not existed in database  **Post-conditions:** User is registered successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Đăng kí thành viên” button | Navigate to register page | | 2 | Fill mandatory fields and click “Đăng ký” button | Navigate to Home page with username at the top right |   **Alternative Scenario:** N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Same as Main Success Scenario | | | 2 | Does not fill all required field | Display warning message to notify the missing field | | 2 | Enter invalid data (wrong confirm password or register with username existed in database) | Display warning message to notify in each field |   **Relationships:**  N/A  **Business Rules:** N/A | | | | |

##### **View Event Detail** C:\Users\duydq00276\Desktop\usecase\GuestOverview\ViewEventDetail.png

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE – VIEW EVENT DETAIL | | | | |
| Use-case No. | UC04.01 | Use-case Version | | 1.0 |
| Use-case Name | View Event Detail | | | |
| Author | DuyDQ | | | |
| Date | 01/06/2013 | Priority | Normal | |
| **Actor:** Guest, Admin, User, Authorized User, Unauthorized User  **Summary:** This use case is about how actor views event detail  **Goal:** Actor can view event detail successful  **Triggers:** N/A  **Pre-conditions:** N/A  **Post-conditions:** N/A  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on an event | Navigate to Event page |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | | |

##### **Search Event by Name**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE – SEARCH EVENT BY NAME | | | | |
| Use-case No. | UC04.02 | Use-case Version | | 1.0 |
| Use-case Name | Search Event by Name | | | |
| Author | DuyDQ | | | |
| Date | 01/06/2013 | Priority | Normal | |
| **Actor:** Admin, User, Authorized User, Guest, Unauthorized User  **Summary:** This use case is about how actor search event by name  **Goal:** Actor can search event successful  **Triggers:** N/A  **Pre-conditions:** N/A  **Post-conditions:** N/A  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Input information want to find in text box | Make suggestion when actor input more than 1 character | | 2 | Click on search label or press Enter key | Navigate to search result page |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:**  N/A  **Business Rules:** N/A | | | | |

#### Unauthorized User



Figure 6 – Unauthorized User Overview Diagram

##### **Login**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USE CASE – LOGIN | | | | |
| Use-case No. | UC05.01 | Use-case Version | | 1.0 |
| Use-case Name | Login | | | |
| Author | DuyDQ | | | |
| Date | 01/06/2013 | Priority | Normal | |
| **Actor:** Admin, Customer, Authorized User, Unauthorized User  **Summary:** This use case is about how actor login to website  **Goal:** Login to system successful  **Triggers:** N/A  **Pre-conditions:** No user has logged into system when access to home page  **Post-conditions:** Actor is logged in successful  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Đăng nhập” button | Navigate to login page | | 2 | Fill mandatory fields and click “Đăng nhập” button | Navigate to Home page with username at the top right |   **Alternative Scenario:** N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Same as Main Success Scenario | | | 2 | Does not fill all required field | Display warning message to notify the missing field | | 2 | Enter invalid data (wrong password or wrong username) | Display warning message “Tên đăng nhập hoặc mật khẩu không đúng” |   **Relationships:**  N/A  **Business Rules:** N/A | | | | |

### Software System Attributes



#### Reliability

* The database can be backed up easily and recovered if necessary.
* The system should never crash or hang, except for the cause from an operating system or network error.
* Mean Time Between Failures (MTBF): The acceptable failure is once a year. The failure should not be because of the database, or else the data may be lost and cannot be recovered.
* Mean Time To Repair (MTTR): When the failure occurs, it should take as little time as possible to repair. The acceptable mean time for a particular failure must be less than 8 hours.
* Maximum Bugs or Defect Rate: 05 bugs / KLOC

#### Availability

* The uptime percentage should be at least 99.95%.
* The acceptable time for maintenance or backup should not be more than 8 hours per month.
* When the system goes in under-maintenance, the website should display a message to inform that.

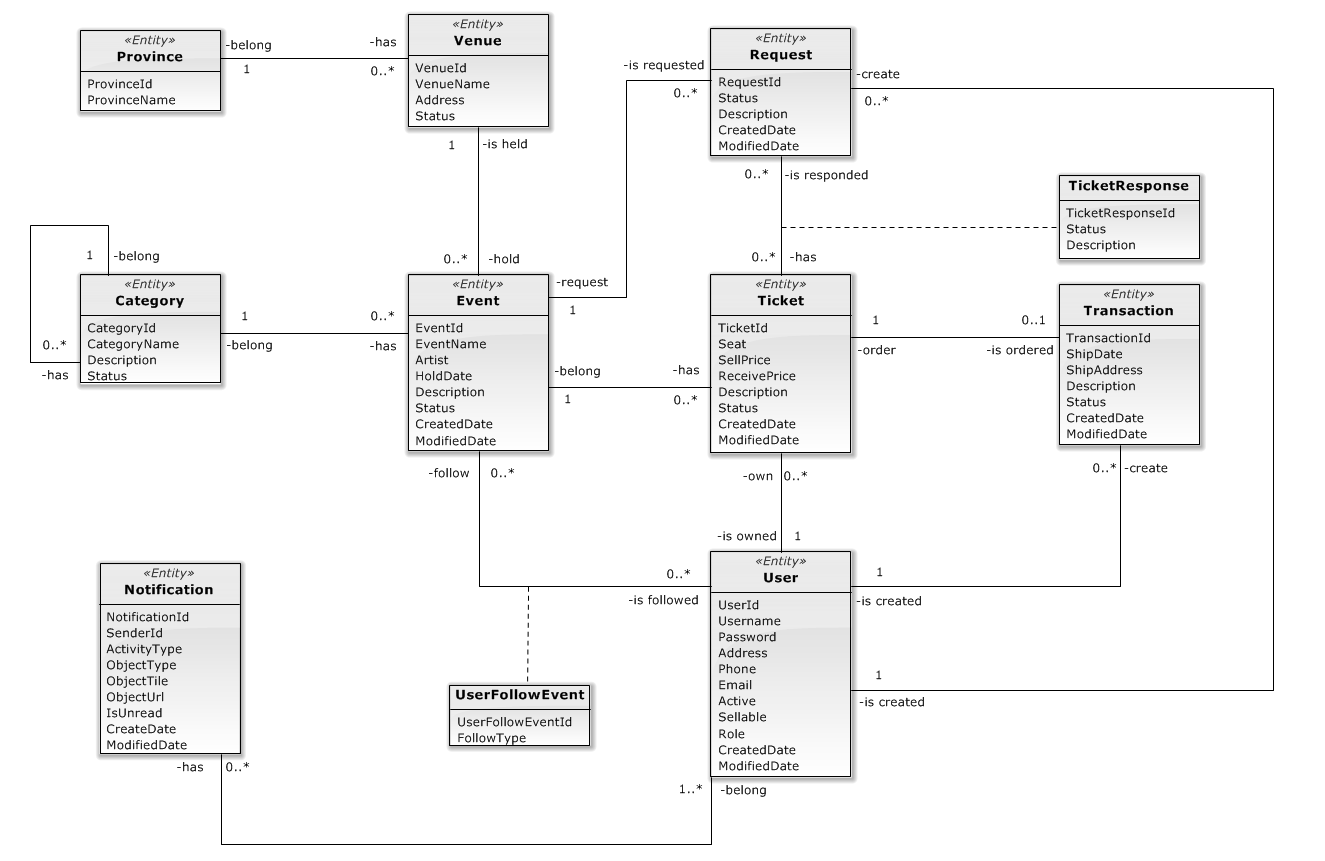
#### Security

* Only users with proper account can access certain information of the system. All the information of users must not be available for anyone or software that is not part of the system. User password is also encrypted and not available to the system administrators.

#### Maintainability

* The code must follow C#.NET coding and naming convention.
* There should be comments in code files that explain the functions of each code segment.
* All the errors should be logged, which supports bug fixing and maintenance.

## Entity Relationship Diagram

****

**Figure 3.3.1 Entity Relationship Diagram**