**MINISTRY OF EDUCATION AND TRAINING**

**FPT UNIVERSITY**

Capstone Project Document

**Site Builder for Product Catalogue**

|  |  |
| --- | --- |
| **Group 05** | |
| **Group members** | Phạm Thị Thu Hòa – Team Leader – SE60962  Trần Hoàn Anh Nguyên – Team member – SE61166  Trương Minh Tiến – Team member – SE61018  Lê Trần Thịnh – Team member – SE61152 |
| **Supervisor** | Mr. Nguyễn Huy Hùng |
| **Ext. Supervisor** | N/A |
| **Capstone Project code** | SBPC |

-Ho Chi Minh City, 05/11/2015-

*This page is intentionally left blank*

Contents

[Contents 3](#_Toc427473522)

[List of tables 6](#_Toc427473523)

[List of figures 10](#_Toc427473524)

[A. Introduction 14](#_Toc427473525)

[1. Project Information 14](#_Toc427473526)

[2. Introduction 14](#_Toc427473527)

[3. Current Situation 14](#_Toc427473528)

[4. Problem Definition 14](#_Toc427473529)

[5. Proposed Solution 15](#_Toc427473530)

[5.1. Feature functions 15](#_Toc427473531)

[5.2. Advantage and disadvantage 15](#_Toc427473532)

[6. Functional Requirements 15](#_Toc427473533)

[7. Role and Responsibility 16](#_Toc427473534)

[B. Software Project Management Plan 17](#_Toc427473535)

[1. Problem Definition 17](#_Toc427473536)

[1.1. Name of this Capstone Project 17](#_Toc427473537)

[1.2. Problem Abstract 17](#_Toc427473538)

[1.3. Project Overview 17](#_Toc427473539)

[2. Project organization 19](#_Toc427473540)

[2.1. Software Process Model 19](#_Toc427473541)

[2.2. Role and responsibilities 19](#_Toc427473542)

[2.3. Tools and Techniques 20](#_Toc427473543)

[3. Project Manage Plan 21](#_Toc427473544)

[3.1. Software development life cycle 21](#_Toc427473545)

[3.2. Product backlog 21](#_Toc427473546)

[3.3. Sprint backlog 23](#_Toc427473547)

[3.4. All Meeting Minutes 23](#_Toc427473548)

[4. Coding Convention 23](#_Toc427473549)

[C. Software Requirement Specification 24](#_Toc427473550)

[1. User Requirement Specification 24](#_Toc427473551)

[1.1. Guest Requirement 24](#_Toc427473552)

[1.2. User Requirement 24](#_Toc427473553)

[1.3. Admin Requirement 24](#_Toc427473554)

[2. System Requirement Specification 24](#_Toc427473555)

[2.1. External user requirement 24](#_Toc427473556)

[2.2. System Overview Use case 26](#_Toc427473557)

[2.3. List of Use Case 27](#_Toc427473558)

[3. Software System Attribute 91](#_Toc427473559)

[3.1. Usability 91](#_Toc427473560)

[3.2. Reliability 91](#_Toc427473561)

[3.3. Availability 91](#_Toc427473562)

[3.4. Security 91](#_Toc427473563)

[3.5. Maintainability 91](#_Toc427473564)

[3.6. Portability 92](#_Toc427473565)

[3.7. Performance 92](#_Toc427473566)

[4. Conceptual diagram 92](#_Toc427473567)

[D. Software Design Description 94](#_Toc427473568)

[1. Design Overview 94](#_Toc427473569)

[2. System Architecture Design 94](#_Toc427473570)

[3. Component Diagram 95](#_Toc427473571)

[4. Detailed Description 97](#_Toc427473572)

[4.1. Class Diagram 97](#_Toc427473573)

[4.2. Class Diagram Explanation 98](#_Toc427473574)

[4.3. Interaction Diagram 106](#_Toc427473575)

[5. User Interface Design 120](#_Toc427473576)

[5.1. Guest Interface Design 120](#_Toc427473577)

[5.2. User Interface Design 122](#_Toc427473578)

[5.3. Admin Interface Design 132](#_Toc427473579)

[6. Database Design 139](#_Toc427473580)

[6.1. Logical Diagram 139](#_Toc427473581)

[6.2. Entity Dictionary 139](#_Toc427473582)

[7. Algorithms 145](#_Toc427473583)

[7.1. Widget’s structure in system 145](#_Toc427473584)

[7.2. Template’s structure in system 146](#_Toc427473585)

[E. System Implementation & Test 147](#_Toc427473586)

[1. Introduction 147](#_Toc427473587)

[1.1. Overview 147](#_Toc427473588)

[1.2. Test Approach 147](#_Toc427473589)

[2. Database Relationship Diagram 147](#_Toc427473590)

[2.1. Physical Diagram 147](#_Toc427473591)

[2.2. Data Dictionary 149](#_Toc427473592)

[3. Performance Measures 156](#_Toc427473593)

[3.1. Definition 156](#_Toc427473594)

[3.2. Test environment 156](#_Toc427473595)

[3.3. Test case 157](#_Toc427473596)

[3.4. Test result 157](#_Toc427473597)

[4. Test plan 158](#_Toc427473598)

[4.1. Features to be tested 158](#_Toc427473599)

[4.2. Features not to be tested 158](#_Toc427473600)

[5. System Testing Test Case 158](#_Toc427473601)

[5.1. User’s core flow 159](#_Toc427473602)

[5.2. Admin’ core flow 177](#_Toc427473603)

[F. Software User’s Manual 189](#_Toc427473604)

[1. Installation Guide 189](#_Toc427473605)

[1.1. Setting up environment at server side 189](#_Toc427473606)

[1.2. Deployment at server side 189](#_Toc427473607)

[1.3. Setting up the environment at client side 191](#_Toc427473608)

[2. User Guide 191](#_Toc427473609)

[2.1. User 192](#_Toc427473610)

[2.2. Admin 199](#_Toc427473611)

[G. Appendix 202](#_Toc427473612)

List of tables

[Table 1: Definitions, Acronyms, and Abbreviations 13](#_Toc427474021)

[Table 2 : Roles and Responsibility 16](#_Toc427474022)

[Table 3: Hardware Requirement for Server 18](#_Toc427474023)

[Table 4: Project Role and Responsibilities 20](#_Toc427474024)

[Table 5: Product backlog 23](#_Toc427474025)

[Table 6: Coding Convention 23](#_Toc427474026)

[Table 7: <Admin> View Template use case specification table 29](#_Toc427474027)

[Table 8: <Admin> Create Template use case specification table 30](#_Toc427474028)

[Table 9: <Admin> Update Template use case specification table 32](#_Toc427474029)

[Table 10: <Admin> Delete Template use case specification table 34](#_Toc427474030)

[Table 11: <Admin> View Widget use case specification table 35](#_Toc427474031)

[Table 12: <Admin> Create Widget use case specification table 37](#_Toc427474032)

[Table 13: <Admin> Update Widget use case specification table 39](#_Toc427474033)

[Table 14: <Admin> Delete Widget use case specification table 40](#_Toc427474034)

[Table 15: <Admin> View Statistic use case specification table 42](#_Toc427474035)

[Table 16: <Admin> View Permission use case specification table 44](#_Toc427474036)

[Table 17: : <Admin> Update Permission use case specification table 46](#_Toc427474037)

[Table 18: <Admin> Change Status Account use case specification table 47](#_Toc427474038)

[Table 19: <Admin> View Business Type use case specification table 49](#_Toc427474039)

[Table 20: <User> View Website use case specification table 52](#_Toc427474040)

[Table 21: <User> Create Website use case specification table 54](#_Toc427474041)

[Table 22: <User> Delete Website use case specification table 55](#_Toc427474042)

[Table 23: <User> Update Website use case specification table 58](#_Toc427474043)

[Table 24: <User> Save Customization use case specification table 60](#_Toc427474044)

[Table 25: <User> View Product use case specification table 61](#_Toc427474045)

[Table 26: <User> Create Product use case specification table 63](#_Toc427474046)

[Table 27: <User> Update Product use case specification table 65](#_Toc427474047)

[Table 28: <User> Delete Product use case specification table 67](#_Toc427474048)

[Table 29: <User> Search Product use case specification table 69](#_Toc427474049)

[Table 30: <User> View Statistic use case specification table 70](#_Toc427474050)

[Table 31: <User> View Domain use case specification table 72](#_Toc427474051)

[Table 32: <User> Change Status Domain use case specification table 74](#_Toc427474052)

[Table 33: <User> Upgrade Account use case specification table 75](#_Toc427474053)

[Table 34: <User> View Article use case specification table 77](#_Toc427474054)

[Table 35: <User> Create Article use case specification table 78](#_Toc427474055)

[Table 36: <User> Update Article use case specification table 80](#_Toc427474056)

[Table 37: <User> Delete Article use case specification table 81](#_Toc427474057)

[Table 38: <User> View Image use case specification table 83](#_Toc427474058)

[Table 39: <User> Create Image use case specification table 84](#_Toc427474059)

[Table 40: <User> View Menu use case specification table 85](#_Toc427474060)

[Table 41: <User> Create Menu use case specification table 87](#_Toc427474061)

[Table 42: <User> View Category use case specification table 89](#_Toc427474062)

[Table 43: <Guest> Register use case specification table 91](#_Toc427474063)

[Table 44: Component Dictionary 96](#_Toc427474064)

[Table 45: Class Dictionary 98](#_Toc427474065)

[Table 46: User Class Attribute 98](#_Toc427474066)

[Table 47: User Class Method 98](#_Toc427474067)

[Table 48: Role Class Attribute 99](#_Toc427474068)

[Table 49: Role Class Method 99](#_Toc427474069)

[Table 50: Permission Class Attribute 99](#_Toc427474070)

[Table 51: Permission Class Method 99](#_Toc427474071)

[Table 52: RolePermission Class Attribute 99](#_Toc427474072)

[Table 53: RolePermission Class Method 99](#_Toc427474073)

[Table 54: Website Class Attribute 100](#_Toc427474074)

[Table 55: Website Class Method 100](#_Toc427474075)

[Table 56: Page Class Attribute 100](#_Toc427474076)

[Table 57: Page Class Method 100](#_Toc427474077)

[Table 58: BusinessType Class Attribute 100](#_Toc427474078)

[Table 59: BusinessType Class Method 101](#_Toc427474079)

[Table 60: Template Class Attribute 101](#_Toc427474080)

[Table 61: Template Class Method 101](#_Toc427474081)

[Table 62: BusinessTypeTemplate Class Attribute 101](#_Toc427474082)

[Table 63: BusinessTypeTemplate Class Method 101](#_Toc427474083)

[Table 64: PageTemplate Class Attribute 101](#_Toc427474084)

[Table 65: PageTemplate Class Method 102](#_Toc427474085)

[Table 66: WidgetType Class Attribute 102](#_Toc427474086)

[Table 67: WidgetType Class Method 102](#_Toc427474087)

[Table 68: Domain Class Attribute 102](#_Toc427474088)

[Table 69: Domain Class Method 102](#_Toc427474089)

[Table 70: Category Class Attribute 102](#_Toc427474090)

[Table 71: Category Class Method 103](#_Toc427474091)

[Table 72: Product Class Attribute 103](#_Toc427474092)

[Table 73: Product Class Method 103](#_Toc427474093)

[Table 74: Media Class Attribute 103](#_Toc427474094)

[Table 75: Media Class Method 103](#_Toc427474095)

[Table 76: MediaProduct Class Attribute 103](#_Toc427474096)

[Table 77: MediaProduct Class Method 104](#_Toc427474097)

[Table 78: Article Class Attribute 104](#_Toc427474098)

[Table 79: Article Class Method 104](#_Toc427474099)

[Table 80: PageWidget Class Attribute 104](#_Toc427474100)

[Table 81: PageWidget Class Method 104](#_Toc427474101)

[Table 82: WidgetCategory Class Attribute 104](#_Toc427474102)

[Table 83: WidgetCategory Class Method 105](#_Toc427474103)

[Table 84: GeneralReport Class Attribute 105](#_Toc427474104)

[Table 85: GeneralReport Class Method 105](#_Toc427474105)

[Table 86: ProductReport Class Attribute 105](#_Toc427474106)

[Table 87: ProductReport Class Method 105](#_Toc427474107)

[Table 88: Payment Class Attribute 106](#_Toc427474108)

[Table 89: Payment Class Method 106](#_Toc427474109)

[Table 90: MenuItem Class Attribute 106](#_Toc427474110)

[Table 91: MenuItem Class Method 106](#_Toc427474111)

[Table 92: Login - Fields 120](#_Toc427474112)

[Table 93: Login - Buttons/Links 121](#_Toc427474113)

[Table 94: Register - Fields 122](#_Toc427474114)

[Table 95: Register - Buttons/Links 122](#_Toc427474115)

[Table 96: User Manage Website - Fields 123](#_Toc427474116)

[Table 97: User Manage Website - Buttons/Links 123](#_Toc427474117)

[Table 98: User Manage Product - Fields 124](#_Toc427474118)

[Table 99: User Manage Product - Buttons/Links 125](#_Toc427474119)

[Table 100: User Create Product - Fields 126](#_Toc427474120)

[Table 101: User Create Product - Buttons/Links 126](#_Toc427474121)

[Table 102: User Manage Media - Fields 127](#_Toc427474122)

[Table 103: User Manage Media - Buttons/Links 128](#_Toc427474123)

[Table 104: User Manage Article - Fields 128](#_Toc427474124)

[Table 105: User Manage Article - Buttons/Links 129](#_Toc427474125)

[Table 106: User Create Article - Fields 129](#_Toc427474126)

[Table 107: User Create Article - Buttons/Links 130](#_Toc427474127)

[Table 108: User View Statistic - Fields 131](#_Toc427474128)

[Table 109: User View Statistic - Buttons/Links 131](#_Toc427474129)

[Table 110: Admin Manage Template - Fields 132](#_Toc427474130)

[Table 111: Admin Manage Template - Buttons/Links 133](#_Toc427474131)

[Table 112: Admin Create Template - Fields 133](#_Toc427474132)

[Table 113: Admin Create Template - Buttons/Links 134](#_Toc427474133)

[Table 114: Admin Manage Widget - Fields 134](#_Toc427474134)

[Table 115: Admin Manage Widget - Buttons/Links 135](#_Toc427474135)

[Table 116: Admin Create Widget - Fields 135](#_Toc427474136)

[Table 117: Admin Create Widget - Buttons/Links 136](#_Toc427474137)

[Table 118: Admin View Statistic - Fields 137](#_Toc427474138)

[Table 119: Admin View Statistic - Buttons/Links 137](#_Toc427474139)

[Table 120: Admin Manage Permission - Fields 138](#_Toc427474140)

[Table 121: Admin Manage Permission - Buttons/Links 138](#_Toc427474141)

[Table 122: Entity Dictionary 140](#_Toc427474142)

[Table 123: Detail Entity Dictionary 145](#_Toc427474143)

[Table 124: Data Dictionary 149](#_Toc427474144)

[Table 125: Attribute Data Dictionary 156](#_Toc427474145)

[Table 126: List page to test 157](#_Toc427474146)

[Table 127: Manage Website Step 191](#_Toc427474147)

[Table 128: Choose template Step 192](#_Toc427474148)

[Table 129: Create Website Step 192](#_Toc427474149)

[Table 130: Custom page Step 193](#_Toc427474150)

[Table 131: Website detail Step 194](#_Toc427474151)

[Table 132: Manage media Step 194](#_Toc427474152)

[Table 133: Add Product Step 195](#_Toc427474153)

[Table 134: Manage Article Step 196](#_Toc427474154)

[Table 135: Create Article Step 197](#_Toc427474155)

[Table 136: View Statistic Step 197](#_Toc427474156)

[Table 137: Manage Account Type and Permission Step 198](#_Toc427474157)

[Table 138: Manage Template Step 199](#_Toc427474158)

[Table 139: Manage Widget Step 199](#_Toc427474159)

[Table 140: View Statistic Step 200](#_Toc427474160)

List of figures

[Figure 1: Scrum Overview Diagram 19](#_Toc427473924)

[Figure 2: System Overview Use Case 26](#_Toc427473925)

[Figure 3: Admin Overview Use Case 27](#_Toc427473926)

[Figure 4: <Admin> View Template use case diagram 27](#_Toc427473927)

[Figure 5: <Admin> Create Template use case diagram 29](#_Toc427473928)

[Figure 6: <Admin> Update Template use case diagram 31](#_Toc427473929)

[Figure 7: <Admin> Delete Template use case diagram 32](#_Toc427473930)

[Figure 8: <Admin> View Widget use case diagram 34](#_Toc427473931)

[Figure 9: <Admin> Create Widget use case diagram 36](#_Toc427473932)

[Figure 10: <Admin> Update Widget use case diagram 37](#_Toc427473933)

[Figure 11: <Admin> Delete Widget use case diagram 39](#_Toc427473934)

[Figure 12: <Admin> View Statistic use case diagram 41](#_Toc427473935)

[Figure 13: <Admin> View Permission use case diagram 43](#_Toc427473936)

[Figure 14: <Admin> Update Permission use case diagram 44](#_Toc427473937)

[Figure 15: <Admin> Change Status Account use case diagram 46](#_Toc427473938)

[Figure 16: <Admin> View Business Type use case diagram 48](#_Toc427473939)

[Figure 17: User Overview Use Case 50](#_Toc427473940)

[Figure 18: <User> View Website use case diagram 51](#_Toc427473941)

[Figure 19: <User> Create Website use case diagram 52](#_Toc427473942)

[Figure 20: <User> Delete Website use case diagram 54](#_Toc427473943)

[Figure 21: <User> Update Website use case diagram 56](#_Toc427473944)

[Figure 22: <User> Save Customization use case diagram 58](#_Toc427473945)

[Figure 23: <User> View Product use case diagram 60](#_Toc427473946)

[Figure 24: <User> Create Product use case diagram 61](#_Toc427473947)

[Figure 25: <User> Update Product use case diagram 63](#_Toc427473948)

[Figure 26: <User> Delete Product use case diagram 66](#_Toc427473949)

[Figure 27: <User> Search Product use case diagram 67](#_Toc427473950)

[Figure 28: <User> View Statistic use case diagram 69](#_Toc427473951)

[Figure 29: <User> View Domain use case diagram 71](#_Toc427473952)

[Figure 30: <User> Change Status Domain use case diagram 72](#_Toc427473953)

[Figure 31: <User> Upgrade Account use case diagram 74](#_Toc427473954)

[Figure 32: <User> View Article use case diagram 75](#_Toc427473955)

[Figure 33: <User> Create sArticle use case diagram 77](#_Toc427473956)

[Figure 34: <User> Update Article use case diagram 78](#_Toc427473957)

[Figure 35: <User> Delete Article use case diagram 80](#_Toc427473958)

[Figure 36: <User> View Image use case diagram 81](#_Toc427473959)

[Figure 37: <User> Create Image use case diagram 83](#_Toc427473960)

[Figure 38: <User> View Menu use case diagram 84](#_Toc427473961)

[Figure 39: <User> Create Menu use case diagram 86](#_Toc427473962)

[Figure 40: <User> View Category use case diagram 87](#_Toc427473963)

[Figure 41: Guest Overview User Case 89](#_Toc427473964)

[Figure 42: <Guest> Register use case diagram 89](#_Toc427473965)

[Figure 43: Conceptual Diagram 92](#_Toc427473966)

[Figure 44: System Architecture Design Diagram 95](#_Toc427473967)

[Figure 45: Component Diagram 96](#_Toc427473968)

[Figure 46: Class Diagram 97](#_Toc427473969)

[Figure 47: <Admin> Create Template sequence diagram 107](#_Toc427473970)

[Figure 48: <Admin> Create Widget sequence diagram 108](#_Toc427473971)

[Figure 49: <Admin> Delete Template sequence diagram 109](#_Toc427473972)

[Figure 50: <Admin> Update Role and Permission sequence diagram 110](#_Toc427473973)

[Figure 51: <Admin> View Statistic sequence diagram 110](#_Toc427473974)

[Figure 52: <Admin> Delete Widget sequence diagram 111](#_Toc427473975)

[Figure 53: <User> Create Website sequence diagram 112](#_Toc427473976)

[Figure 54: <User> Delete Website sequence diagram 113](#_Toc427473977)

[Figure 55: <User> Create Product sequence diagram 114](#_Toc427473978)

[Figure 56: <User> Create Media sequence diagram 115](#_Toc427473979)

[Figure 57: <User> Create Article sequence diagram 115](#_Toc427473980)

[Figure 58: <User> Save Page Custom sequence diagram 116](#_Toc427473981)

[Figure 59: <User> Create Page sequence diagram 117](#_Toc427473982)

[Figure 60: <User> Delete Page sequence diagram 117](#_Toc427473983)

[Figure 61: <User> Write Widget sequence diagram 118](#_Toc427473984)

[Figure 62: <User> View Statistic sequence diagram 119](#_Toc427473985)

[Figure 63: Guest Login interface design 120](#_Toc427473986)

[Figure 64: Register interface design 121](#_Toc427473987)

[Figure 65: User Manage Website interface design 122](#_Toc427473988)

[Figure 66: User Manage Product interface design 124](#_Toc427473989)

[Figure 67: User Create Product interface design 125](#_Toc427473990)

[Figure 68: User Manage Media interface design 127](#_Toc427473991)

[Figure 69: User Manage Article interface design 128](#_Toc427473992)

[Figure 70: User Create Article interface design 129](#_Toc427473993)

[Figure 71: User View Statistic interface design 130](#_Toc427473994)

[Figure 72: Admin Manage Template interface design 132](#_Toc427473995)

[Figure 73: Amin Create Template interface design 133](#_Toc427473996)

[Figure 74: Admin Manage Widget interface design 134](#_Toc427473997)

[Figure 75: Admin Create Widget interface design 135](#_Toc427473998)

[Figure 76: Admin View Statistic interface design 136](#_Toc427473999)

[Figure 77: Admin Manage Permission interface design 137](#_Toc427474000)

[Figure 78: Logical Diagram 139](#_Toc427474001)

[Figure 79: Widget’s Structure description 146](#_Toc427474002)

[Figure 80: Template’ structure description 147](#_Toc427474003)

[Figure 81: Physical Diagram 148](#_Toc427474004)

[Figure 82: User Test case flow overview 158](#_Toc427474005)

[Figure 83: Admin Test case flow overview 159](#_Toc427474006)

[Figure 84: User – Manage Website Page 191](#_Toc427474007)

[Figure 85: User- Choose Template 191](#_Toc427474008)

[Figure 86: User – Create Website Page 192](#_Toc427474009)

[Figure 87: User – Custom Page 193](#_Toc427474010)

[Figure 88: User – Website detail Page 193](#_Toc427474011)

[Figure 89: User – Manage Media Page 194](#_Toc427474012)

[Figure 90: User – Add New Product Popup 195](#_Toc427474013)

[Figure 91: User – Manage Article Page 196](#_Toc427474014)

[Figure 92: User - Create Article Popup 196](#_Toc427474015)

[Figure 93: User – View Statistic Page 197](#_Toc427474016)

[Figure 94: Admin – Manage Account Type and Permission Page 198](#_Toc427474017)

[Figure 95: Admin – Manage Template Page 198](#_Toc427474018)

[Figure 96: Admin – Manage Widget Page 199](#_Toc427474019)

[Figure 97: Admin – View Statistic Page 200](#_Toc427474020)

**Definitions, Acronyms, and Abbreviations**

|  |  |
| --- | --- |
| Name | Definition |
| SBPC | Site Builder for Product Catalogue |
|  |  |

Table 1: Definitions, Acronyms, and Abbreviations

# Introduction

## Project Information

* Project name: **Site Builder for Product Catalogue**
* Project Code: **SPBC**
* Product Type: Website Application
* Start Date: May 11th, 2015
* End Date: August 24th, 2015

## Introduction

Nowadays, many businesses want to promote their product to customers with lowest cost. The expenses for advertising via television, newspaper or direct mail are usually expensive. As a result of fact, most of small and medium businesses promote and deliver their product to customer for free or by cheap methods such as: social networking (Facebook, Twitter), flyers, which are not always marketing effective. Due to the rapid growing of the Internet and related technologies, more and more businesses need to build website to connect with customers in the easier and more effective way. However, owners of small businesses such as: fashion boutiques, flower shops and cake shops are limited in knowledge of information technology, they must hire web developers which usually causes to high business costs. As a matter of fact, the Site Builder Product Catalogue (SBPC) system is developed to support enterprises to build their websites quickly and conveniently.

## Current Situation

Currently there are many sites that help business companies build website with pre-defined templates. However, almost all of these website builders only support users to custom their website for general purposes. When users want to configure more various functions, they must contact web developers to be supported. In addition, the cost for privately supported custom is very high that only big companies can afford.

## Problem Definition

We are researched on some current websites in Vietnam such as: sieuweb.com, webbnc.net… We also researched on some current overseas websites such as: weebly.com, wix.com, godaddy.com... Some of them have some limit.

* Not supported drag and drop interface.
* Not supported function create free website.
* Not supported function subscribe website that user created.
* Not supported design responsive website.
* Not supported website background music.

## Proposed Solution

### Feature functions

With the target to provide a reliable website for not only enterprises but also small traders to introduce their product to consumers the system is developed on tools to support customers to create their own websites quickly and easily. Moreover, users also can custom provided templates to adapt to each product specification. Furthermore, user will be able to get suggestions for templates that are suitable with their business areas.

### Advantage and disadvantage

The advantages and disadvantages of the proposed solution:

* Advantages:
* Designed specifically for product catalogue.
* Support drag and drop interface.
* Support function subscribe website.
* Support users create websites with cheaper cost.
* Disadvantages:
* Not support order and shipping online.
* Support online payment using only PayPal.
* Not support marketing such as: SEO on Google, SMS marketing.

## Functional Requirements

Admin:

* + - Manage Account
    - Manage Template
    - Manage Widget
    - View Website Statistic

Users:

* Manage their Websites
* Manage Product
* Manage Domain
* Custom their Websites
* View Websites Statistic

## Role and Responsibility

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Full Name** | **Role** | **Position** | **Contact** |
| 1 | Nguyễn Huy Hùng | Project Supervisor | Instructor | [hungnh@fpt.edu.vn](mailto:hungnh@fpt.edu.vn) |
| 2 | Phạm Thị Thu Hòa | Scrum Master | Team Leader | [hoapttse60962@fpt.edu.vn](mailto:hoapttse60962@fpt.edu.vn) |
| 3 | Trần Hoàn Anh Nguyên | Developer | Team Member | [nguyenthase61166@fpt.edu.vn](mailto:nguyenthase61166@fpt.edu.vn) |
| 4 | Trương Minh Tiến | Developer | Team Member | [tientmse61018@fpt.edu.vn](mailto:tientmse61018@fpt.edu.vn) |
| 5 | Lê Trần Thịnh | Developer | Team Member | [thinhltse61152@fpt.edu.vn](mailto:thinhltse61152@fpt.edu.vn) |

Table 2 : Roles and Responsibility

# Software Project Management Plan

## Problem Definition

### Name of this Capstone Project

* Project name:Site Builder for Product Catalogue
* Project Code: SPBC

### Problem Abstract

Nowadays, besides growing of internet and technologies, many businesses need a website to promote their products. However, creating website is not only cost but also complex with people who do not knowledge about information technology. In the market, many websites and technologies support businesses to solve these problems but most of them are not meet all demands of business. For implementation, we developing a system that businesses can learn and use easily to create their website. Business can promote their products, save cost and make more profit.

### Project Overview

#### Current Situation and Disadvantages

Many websites support creating free websites, but some services just are CMS (Content Management System). Therefore, these services are inconsistent with product introduction for business. Complexity also be obstruct for building website. Lack of functions, widgets to building site or do not have responsive design make websites have little access numbers.

#### The Proposed System

With disadvantages above, this project builds up a site builder for product catalogue. The system has some functions as:

Admin:

* Manage Account: admin can change status of user.
* Manage Template
* Manage Widget
* View Website Statistic: admin can view statistic of website such as number of following, top websites are visited).
* Manage Business Category.
* Manage Permission.
* Manage Account Type.

Users:

* Manage their Website: user can create and init website domain, user can change their website domains and update information of website.
* Manage page: user can create, delete, update page content and change page layout.
* Manage Product.
* Manage Article.
* Manage Media (Images and Videos).
* Manage Category Item.
* Custom their Websites: user can custom layout and theme each, change contain of each pages in websites.
* View Websites Statistic: user can view statistic of website such a number of product, number of website, number of view.
* View and update profile.
* Upgrage account and payment.

#### Boundaries of System

SPBC is a web-based service that everyone with a PC/laptop or smart phone with internet connection can use.

This system does not intent to implement these functions:

* Shopping cart.
* Manage order.
* Drag and drop are on PC/laptop only.

#### Development Environment

##### Hardware requirement

|  |  |  |
| --- | --- | --- |
|  | Minimum Requirement | Recommended |
| Internet Connection | Cable, Wi-Fi (4Mbps) | Cable, Wi-Fi (8Mbps) |
| Computer Processor | Intel® Xeon® Quad Core 2.50 GHz | Intel® Core™ i3 (2.50GHz) |
| Operating System | Window Server 2008 | Window Server 2008 |
| Computer Memory | 1 GB Ram | 2 GB or more |

Table 3: Hardware Requirement for Server

##### Software requirement

* + - Java 8.
    - Window 7.
    - Web browser: Firefox, Internet Explorer 10 or higher

## Project organization

### Software Process Model

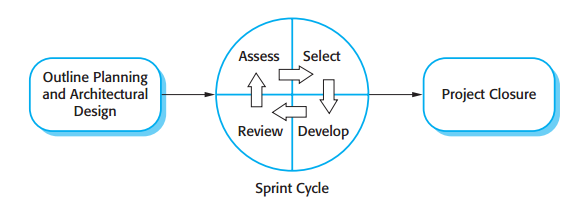


Figure 1: Scrum Overview Diagram

Reference: SOFTWARE ENGINEERING Ninth Edition, by Ian Sommerville (page 73).

Because of the project characteristics, we decide to make progress in series of sprints which are time boxed iterations in one week. At the beginning of one sprint, through sprint planning meeting, all team members will discuss together to define sprint backlog which is suitable to be completed within a week. Daily meeting and online conference are used to manage all activities and issues troubleshoot. At the end of sprint, product owner and developer team will review completed product to figure out the necessary changes for products.

### Role and responsibilities

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| * **No.** | **Full name** | * **Team Role** | * **Scrum Team Role** | **Responsibilities** |
| 1 | Nguyễn Huy Hùng | Supervisor | Product Owner | * Outline work in scrum backlog * Answer question and deliver direction |
| 2. | Phạm Thị Thu Hòa | Team Leader | Scrum Master | * Facilitate productivity – maximize team performance * Complete all individual work * Support each other to complete team work |
| 3. | Trần Hoàn Anh Nguyên | Team Member | Scrum Team Member | * Commit individual product on time * Support each other to complete team work |
| 4. | Trương Minh Tiến | Team Member | Scrum Team Member | * Commit individual product on time * Support each other to complete team work |
| 5. | Lê Trần Thịnh | Team Member | Scrum Team Member | * Commit individual product on time * Support each other to complete team work |

Table 4: Project Role and Responsibilities

**Product Owner** in this project is also Project Manager who supports the whole team by outlining works in the [scrum backlog](http://scrummethodology.com/the-scrum-backlog/), prioritizing it and being available to answer questions and deliver direction.

**Scrum Master** has responsibility to do everything he or she can do to facilitate productivity – address needs of the team and contribute to Product Owner to maximize team performance. As a real team player, he is required to complete all individual work.

**Scrum Team Member** is expected to be self-organized, cross-functional with high adaptability. All members have the right to do anything within the boundaries of project to meet the sprint goals and have responsibility not only to commit their individual product on time but also to support other members to ensure team can complete its work.

### Tools and Techniques

* OS: Window 7.
* DBMS: MySQL 5.5.44.
* Back-end: Netbeans 8.0.2, JDK 8 update 45.
* Front-end: CSS3, HTML5, Angular JS, JavaScript, jQuery, Ajax.
* Server: Apache Tomcat 8.0.15
* Repository: Bitbucket.
* Modeling tool: StarUML 2.2.0
* Document control: Google Drive, Microsoft Word 2010
* Communication and meeting: Skype.

## Project Manage Plan

### Software development life cycle

The starting point for planning is the product backlog, which is the list of work to be done on the project. During the sprint planning meeting, the team selects some number of product backlog items, usually in the form of user stories, and identifies the tasks necessary to complete each user story.

Every sprint begins with sprint planning meeting, in which the Product Owner and team discuss the prioritized tasks in product backlog then add that tasks to sprint backlog. Once the time team commit sprint backlog, Product Owner can’t add more task.

During daily meeting, team has online meeting to update task status, discusses solution to challenges. It happens each day of the sprint.

At the end of the sprint, that has a sprint review meeting, in which team present it works to the Product Owner. Project Owner decides each task is met acceptance criteria or not. If a task isn’t accepted, it’s rejected as incomplete.

All team also gather after end of each sprint to share what worked, what didn’t, and how processes could be improved. This meeting is called the sprint retrospective meeting.

### Product backlog

|  |  |  |
| --- | --- | --- |
| **ID** | **Feature** | **User story** |
| 1 | Register | Registering to become member of website |
| 2 | Login/Logout | Accessing the account to use their features |
| 3 | View list website | User can view list websites created before |
| 4 | Create website | User insert new website. |
| 5 | Update website | User edits information of website. |
| 6 | Delete website | User delete existed website |
| 7 | View list product | User can view list products created before |
| 8 | Create product | User insert new product |
| 9 | Update product | User edit product information |
| 10 | Delete product | User delete existed website |
| 11 | Search product | User search product by name |
| 12 | View list category | User can view list categories created before |
| 13 | Create category | User insert new category |
| 14 | Update category | User edit category |
| 15 | Delete category | User delete existed category |
| 16 | View list image | User can view list images created before |
| 17 | Create image | User insert new image |
| 18 | Delete image | User delete category name |
| 19 | View list video | User can view list videos created before |
| 20 | Create video | User insert new video |
| 21 | Delete video | User delete video |
| 22 | View list articles | User can view list articles created before. |
| 23 | Create article | User insert new article |
| 24 | Update article | User edit article |
| 25 | Delete article | User delete existed article |
| 26 | View list domains | User can view list domains created before |
| 27 | Create domain | User insert new domain |
| 28 | Update domain | User edit domain |
| 29 | Delete domain | User delete existed domain |
| 30 | Change status domain | User can change status existed domain |
| 31 | View statistic website | User can view existed website statistic |
| 32 | Upgrade account | User payment to upgrade account |
| 33 | View profile | User view their profile |
| 34 | Update profile | User update their profile |
| 35 | User custom website | User can create new pages, delete pages in website |
| 36 | Create new widget in page | User can drag drop widget |
| 37 | Delete widget in page | User remove widget from website |
| 38 | Custom widget | User change content in widget, how to widget is displayed |
| 39 | View list templates | Admin can view list templates created before |
| 40 | Create template | Admin insert new template |
| 41 | Update template | Admin edit template |
| 42 | Delete template | Admin delete existed template |
| 43 | View list widgets | Admin can view list widgets in system |
| 44 | Create widget in system | Admin insert new widget in system |
| 45 | Update widget in system | Admin edit widget in system |
| 46 | Delete widget in system | Admin delete existed widget in system |
| 47 | View list account | Admin view list users in system |
| 48 | Change status account | Admin change status account is registered into system. |
| 49 | View list business type | Admin can view list business type |
| 50 | Create business type | Admin insert new business type |
| 51 | Update business type | Admin edit business type |
| 52 | Delete business type | Admin delete existed business type |
| 53 | View account type and permission | Admin view permission corresponding with each account type |
| 54 | Update permission | Admin edit permission corresponding with each account type |

Table 5: Product backlog

### Sprint backlog

Refer to “Project Management” folder.

### All Meeting Minutes

Refer to “Meeting Minutes” folder.

## Coding Convention

This project is applied some coding convention rules as listed below:

|  |  |  |
| --- | --- | --- |
| No. | Type | Description |
| 1. | Naming convention | * User camel case for naming variables and functions. * User Pascal case for naming classes. |
| 2. | Indentation and braces | * Avoid line length more than 120 characters. * Principles to breaking lines that too long. |
| 3. | Declaration | * Use one declaration per line |
| 4. | Comment | * Comment should use to give overviews of code and additional information that not readily available. |

Table 6: Coding Convention

# Software Requirement Specification

## User Requirement Specification

### Guest Requirement

Guest is a person who does not have access to the system. To use all functions, guest must login. These are some functions guests can use:

* Register
* Login

### User Requirement

Users are guesses who use account with user’s role to login to the system. Users can additional functions such as:

* View profile, update profile.
* Manage Product
* Manage Article
* Manage Item Category
* Manage Website
* View statistic
* Upgrade account
* Manage Media
* Custom Website

### Admin Requirement

Admin is guest who use account with admin’s role to login to the system. Admin can additional functions such as:

* Manage Account
* Manage Template
* Manage Widget
* Manage Permission
* Manage Account Type
* View statistic
* Manage Business Type

## System Requirement Specification

### External user requirement

#### User Interface

The user interface uses language Vietnamese

#### Hardware Interface

N/A

#### Software Interface

Firefox (v38 or above), Chromes (v41 or above) browser, enable JavaScript.

#### Communication protocol

Website uses HTTP protocol for communication between the web browser and the web server.

### System Overview Use case

System overview use case only provides an overview on main features of the system. It mainly expressed in the form of in abstract use cases (abstract use case is not function in system). The detail functions of the system are represented in overview use cases of each actor.



Figure 2: System Overview Use Case

### List of Use Case

#### <Admin> Overview Use Case

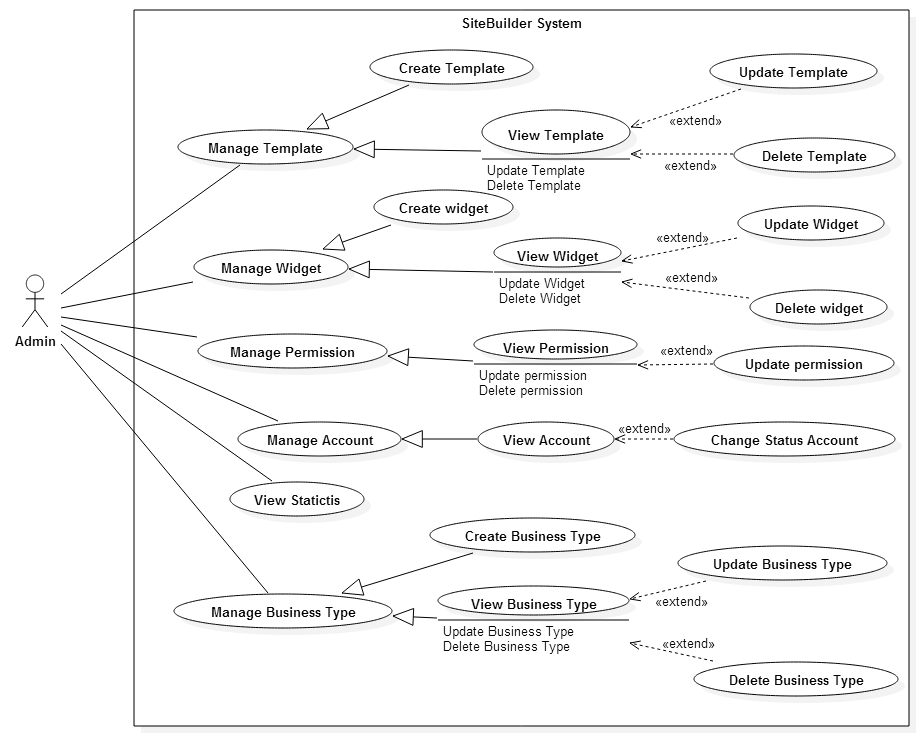


Figure 3: Admin Overview Use Case

##### <Admin> View Template

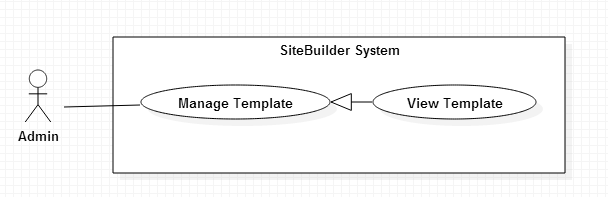


Figure 4: <Admin> View Template use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UCA01** | | | |
| **Use Case No.** | UCA01 | **Use Case Version** | 1.1 |
| **Use Case Name** | View Template | | |
| **Author** | Trương Minh Tiến | | |
| **Date** | 01/06/2015 | **Priority** | Normal |
| **Actor:**   * Admin   **Summary:**   * This use case allows admin to view all templates in system.   **Goal:**   * Admin can view all templates in system.   **Triggers:**   * From the menu bar: Choose “Template” menu.   **Preconditions:**   * User must log in the system with admin role.   **Post Conditions:**   * **Success:** List templates displayed. * **Fail:** Cannot load list templates.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on “Template” menu | Display list of templates:   * Thumbnail: image * Name: text * Path: text * Description: text * Create date: text * Update: button * Delete: button   [Alternative 1] [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Clicks on “Template ” menu, with no templates | Display message: “Do not have any template in system.” |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Cannot view list templates. | Display error message: “Cannot load list templates. Please try later.” |   **Relationships:** Manage Template  **Business Rules:**   * When users create new website, users can choose a template in this list. * A template can belong to many business types. Depends on business of users, system can suggest templates in this list that suitable with user needs. | | | |

Table 7: <Admin> View Template use case specification table

##### <Admin> Create Template

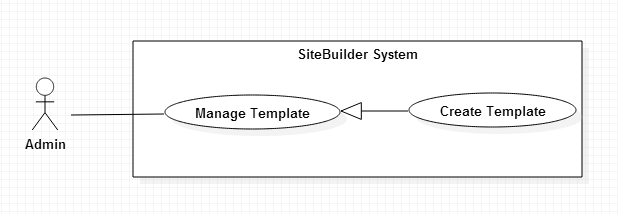


Figure 5: <Admin> Create Template use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCA02** | | | |
| **Use Case No.** | UCA02 | **Use Case Version** | 1.1 |
| **Use Case Name** | Create Template | | |
| **Author** | Trương Minh Tiến | | |
| **Date** | 01/06/2015 | **Priority** | High |
| **Actor:** Admin  **Summary:**   * This use case is used to create a new template.   **Goal:**   * Admin can add new template.   **Triggers:**   * Click “Template” menu. * Click “Add” button.   **Preconditions:**   * User must be logged in the system with Admin role.   **Post Conditions:**   * **Success:** New template is added to system * **Fail:** Cannot add new template to system.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on “Create new template” button. | Display a new popup to add template:   * Choose file button: required * Description: Textbox * Business type: Dropdown list * Submit: button * Cancel: button | | 3 | Fill forms and clicks on “Submit” button  [Alternative 1] | Create new template’s folder and create new record to database.  Display message notify new template is created.  Current popup is closed.  [Exception 1, 2, 3] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Clicks on “Cancel” button | Close current popup. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Not input required fields | Display message to notify user input required fields. | | 2 | Choose file is not a zip file | Display message to notify that file upload must be zip format. | | 3 | Choose file with size more than 20MB | Display message to notify maximum of size is 20MB. |   **Relationships:** View Template  **Business Rules:**   * Template must be a zip file with structure is defined by rule mapping with structure of templates are used in system. * A Template is uploaded can include one or more pages. Page with name as “index” is home page of template. * Name of template based on name of file upload. | | | |

Table 8: <Admin> Create Template use case specification table

##### <Admin> Update Template

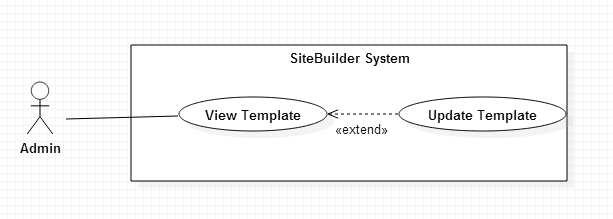


Figure 6: <Admin> Update Template use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UCA03** | | | |
| **Use Case No.** | UCA03 | **Use Case Version** | 1.0 |
| **Use Case Name** | Update Template | | |
| **Author** | Trương Minh Tiến | | |
| **Date** | 01/06/2015 | **Priority** | Normal |
| **Actor:** Admin  **Summary:**   * This use case is used to edit template.   **Goal:**   * Admin can edit information of template.   **Triggers:**   * Click “Template” menu. * Click on edit button   **Preconditions:**   * User must be logged in the system with Admin role. * Template is updated is existed in system.   **Post Conditions:**   * **Success:** New information of template is updated. * **Fail:** Cannot update information of template.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on edit button | Redirect to update page and show information of template:   * Add file: button * Name: Textbox, required, length 3-255 characters * Description: Textbox * Business type: Dropdown list * List files of template * Content file: text * Submit: button * Back: button | | 2 | [Alternative 1]  User clicks on “Submit” button | [Exception 1]  Show message “Template was successfully updated” and redirect to View Template page. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 5 | User click “Back” button | Redirect to View Template page. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Respond | | 1 | Not input required fields | Display error message to notify user input required fields. | | 2 | Cannot update template due to database connection. | Show message cannot update template. |   **Relationships:** View Template  **Business Rules:**   * Template is updated must be matched the defined template structure in system. | | | |

Table 9: <Admin> Update Template use case specification table

##### <Admin> Delete Template

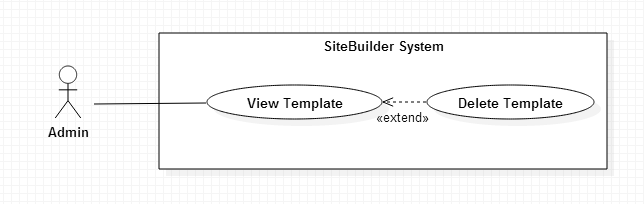


Figure 7: <Admin> Delete Template use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCA04** | | | |
| **Use Case No.** | UCA04 | **Use Case Version** | 1.1 |
| **Use Case Name** | Delete Template | | |
| **Author** | Trương Minh Tiến | | |
| **Date** | 01/06/2015 | **Priority** | Normal |
| **Actor:**   * Admin   **Summary:**   * This use case is used to remove template of system.   **Goal:**   * Admin can remove template of system   **Triggers:**   * Click “Template” menu. * Click on delete button.   **Preconditions:**   * User must be logged in the system with Admin role.   **Post Conditions:**   * **Success:** Template is removed from database * **Fail:** Cannot remove template and display error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on “Delete” button in manage template page. | Show popup:   * “Are you sure to delete” * “Yes” button. * “No” button” | | 2 | Click “Yes” button  [Alternative 1] | Delete folder template and delete record template in database.  Show message delete successful.  [Exception 1,2] | | 3 | Click “close” button | Close current popup. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Clicks on “No” button | Close current popup |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Cannot remove template due to database connection | Display error message: “Cannot delete template. Please try later.” | | 2 | Cannot remove template because it is used to create website. | Display message to notify that template is used, cannot remove. |   **Relationships:** View Template  **Business Rules:**   * Templates is using to created website cannot remove. * Delete record in database and delete folder is used to save template. | | | |

Table 10: <Admin> Delete Template use case specification table

##### <Admin> View Widget

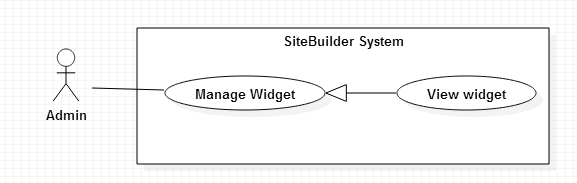


Figure 8: <Admin> View Widget use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCA05** | | | |
| **Use Case No.** | UCA05 | **Use Case Version** | 1.1 |
| **Use Case Name** | View Widget | | |
| **Author** | Phạm Thị Thu Hòa | | |
| **Date** | 01/06/2015 | **Priority** | Normal |
| **Actor:**   * Admin   **Summary:**   * This use case allows admin to view all widgets in system.   **Goal:**   * Admin can view all widgets in system.   **Triggers:**   * From the menu bar: Choose “Widget” menu.   **Preconditions:**   * User must log in the system with Admin role.   **Post Conditions:**   * **Success:** List widgets displayed. * **Fail:** Cannot load list widgets.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on “Widget” menu | Display list of widgets:  Thumbnail: image  Name: text  Path: text  Create date: path  Update: button  Delete: button  View: button  [Alternative 1] [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | List of widgets is empty | Display error message: “Do not have any widget in system.” |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Cannot view list widgets. | Display error message: “Cannot load list widgets. Please try later.” |   **Relationships:** Manage Widget  **Business Rules:**   * When users create new website, users can drag and drop widgets in this list. | | | |

Table 11: <Admin> View Widget use case specification table

##### <Admin> Create Widget

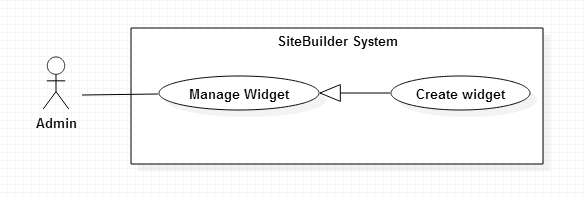


Figure 9: <Admin> Create Widget use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCA06** | | | |
| **Use Case No.** | UCA06 | **Use Case Version** | 1.1 |
| **Use Case Name** | Create Widget | | |
| **Author** | Phạm Thị Thu Hòa | | |
| **Date** | 01/06/2015 | **Priority** | High |
| **Actor:** Admin  **Summary:**   * This use case is used to create a new widget.   **Goal:**   * Admin can add new widget.   **Triggers:**   * Click “Widget” menu. * Click “Add” button.   **Preconditions:**   * User must be logged in the system with Admin role.   **Post Conditions:**   * **Success:** New widget is added to system * **Fail:** Cannot add new widget to system.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on “Add” button. | Display Add Widget popup:   * Choose file button: required * Description: Text * Category: Dropdown list, required. * Submit: button * Cancel: button | | 3 | Fill forms and clicks on “Submit” button  [Alternative 1] | Create new widget’s folder and create new record to database  Display message “Create successful new widget”.  [Exception 1, 2, 3] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Clicks on “Cancel” button | Close current popup. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Not input required fields | Display message to notify user input required fields. | | 2 | Choose file is not a zip file | Display message to notify that file upload must be zip format. | | 3 | Choose file with size more than 20MB | Display message to notify maximum of size is 20MB. |   **Relationships:** View Widget  **Business Rules:**   * Widget must be a zip file with structure is defined by rule mapping with structure of widgets are used in system. * Name of widget based on name of file upload. | | | |

Table 12: <Admin> Create Widget use case specification table

##### <Admin> Update Widget

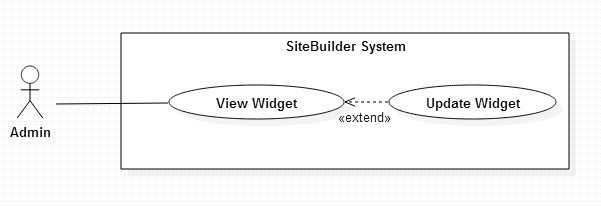


Figure 10: <Admin> Update Widget use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCA07** | | | |
| **Use Case No.** | UCA07 | **Use Case Version** | 1.1 |
| **Use Case Name** | Update Widget | | |
| **Author** | Phạm Thị Thu Hòa | | |
| **Date** | 01/06/2015 | **Priority** | Normal |
| **Actor:** Admin  **Summary:**   * This use case is used to edit widget.   **Goal:**   * Admin can edit information of widget.   **Triggers:**   * Click “Widget” menu. * Click on edit button   **Preconditions:**   * User must be logged in the system with Admin role. * Widget is updated is existed in system.   **Post Conditions:**   * **Success:** New information of widget is updated. * **Fail:** Cannot update information of widget.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on “Edit” button | Redirect to update page and show information of widget:   * Thumbnail: image * Upload image: button * Name: Textbox, required, length 4-32 characters * Type: Dropdown list * Content file: text * Submit: button * Back: button | | 2 | [Alternative 1]  User clicks on “Submit” button | [Exception 1]  Show message “Widget was successfully updated” and redirect to page View Widget |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | User click “Back” button | Redirect to page View Widget page |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Respond | | 1 | Not input required fields | Display error message to notify user input required fields. | | 2 | Cannot update widget due to database connection. | Show message cannot update widget. |   **Relationships:** View Widget  **Business Rules:**   * Widget is updated must be matched the defined widget structure. | | | |

Table 13: <Admin> Update Widget use case specification table

##### <Admin> Delete Widget

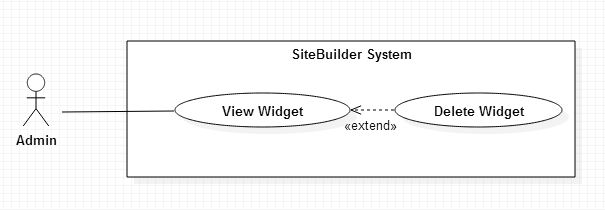


Figure 11: <Admin> Delete Widget use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCA08** | | | |
| **Use Case No.** | UCA08 | **Use Case Version** | 1.0 |
| **Use Case Name** | Delete Widget | | |
| **Author** | Phạm Thị Thu Hòa | | |
| **Date** | 01/06/2015 | **Priority** | Normal |
| **Actor:**   * Admin   **Summary:**   * This use case is used to remove widget of system.   **Goal:**   * Admin can remove widget of system   **Triggers:**   * Click “Widget” menu. * Click on delete button.   **Preconditions:**   * User must be logged in the system with Admin role.   **Post Conditions:**   * **Success:** Widget is removed from database * **Fail:** Cannot remove widget and display error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on “Delete” button | Show popup:   * “Are you sure to delete?” * “Yes” button. * “No” button” | | 2 | Click “Yes” button  [Alternative 1] | Delete folder widget and delete record widget in database.  Show message delete successful.  [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Clicks on “No” button | Close current popup |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Cannot remove widget due to database connection | Display error message: “Cannot delete widget. Please try later.” |   **Relationships:** View Widget  **Business Rules:**   * Delete record in database and delete folder is used to save widget. * When user custom page, system do not show widget deleted. | | | |

Table 14: <Admin> Delete Widget use case specification table

##### <Admin> View Statistic

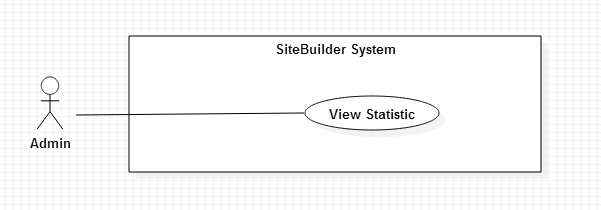


Figure 12: <Admin> View Statistic use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCA09** | | | |
| **Use Case No.** | UCA09 | **Use Case Version** | 1.0 |
| **Use Case Name** | View Statistic | | |
| **Author** | Trương Minh Tiến | | |
| **Date** | 01/06/2015 | **Priority** | Normal |
| **Actor:**  Admin  **Summary:**   * This use case is used to view statistic of system.   **Goal:**   * Admin can view statistic of system   **Triggers:**   * Click “Statistic” menu.   **Preconditions:**   * User must be logged in the system with Admin role.   **Post Conditions:**   * **Success:** Statistic of system is displayed. * **Fail:** Cannot display statistic of system and display error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click “Statistic” menu. | Display statistic of system:  +System information:   * Number of account: text * Number of free user: text * Number of silver user: text * Number of gold user: text * Number of website: text   [Exception 1]  +List Top website is accessed:   * Number of product: text * Storage: text * Number of views per day: Line chart * Number of views per mouth: Column chart   [Exception 2] |   **Alternative Scenario:** N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | List website is empty | Display error message: “List of website is empty.” | | 2 | Cannot display information | Display error message: “Cannot view statistic.” |   **Relationships:** N/A  **Business Rules:**   * Daily statistics will retrieve data during the last 30 days. * Monthly statistics will retrieve data during the last 12 months. * Storage of website is calculated based on size of website folder. * The number products of website include products with status is Disable. | | | |

Table 15: <Admin> View Statistic use case specification table

##### <Admin> View Permission

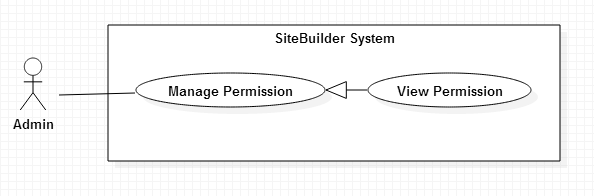


Figure 13: <Admin> View Permission use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCA10** | | | |
| **Use Case No.** | UCA10 | **Use Case Version** | 1.1 |
| **Use Case Name** | View Permission | | |
| **Author** | Lê Trần Thịnh | | |
| **Date** | 01/06/2015 | **Priority** | Medium |
| **Actor:**   * Admin   **Summary:**   * This use case allows admin to view all permissions in system.   **Goal:**   * Admin can view all permissions in system.   **Triggers:**   * From the menu bar: choose “Permission” menu.   **Preconditions:**   * User must log in the system with admin role.   **Post Conditions:**   * **Success:** List permissions displayed. * **Fail:** Cannot load list permissions.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on “Permission” menu | Display list of permissions correspond with membership:  Maximum websites user can create : text  Maximum megabyte storage: text  Support view statistic: checkbox  [Exception 1] |   **Alternative Scenario:** N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Cannot view list permissions. | Display error message: “Cannot load list permission. Please try later.” |   **Relationships:** Manage Permission  **Business Rules:**   * Membership includes three types: free user, silver user and gold user. * Free users, silver users and gold users are limited number of websites, storage. * Number of websites and storage of free users are smaller than silver users. * Number of websites and storage of silver users are smaller than gold users. * Free users can not view statistic. | | | |

Table 16: <Admin> View Permission use case specification table

##### <Admin> Update Permission

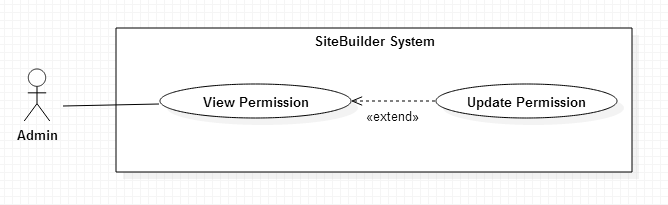


Figure 14: <Admin> Update Permission use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UCA11** | | | |
| **Use Case No.** | UCA11 | **Use Case Version** | 1.0 |
| **Use Case Name** | Update Permission | | |
| **Author** | Lê Trần Thịnh | | |
| **Date** | 01/06/2015 | **Priority** | Normal |
| **Actor:** Admin  **Summary:**   * This use case is used to edit permission.   **Goal:**   * Admin can edit information of template.   **Triggers:**   * Click “Permission” menu. * Click on “Edit” button.   **Preconditions:**   * User must be logged in the system with Admin role. * Accounts type existed in system.   **Post Conditions:**   * **Success:** Permission is updated. * **Fail:** Cannot update information of permission.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on edit button | Display popup:   * Membership name: text input, required, max length 3-22 characters. * Membership price: number input, required, value from 0 to 1 billion. * Number of website: number input, required, value from 0 to 500. * View Statistic: checkbox. * Update: button. * Cancel: button. | | 2 | Click “Update” button | Permission is updated to database and show message update success.  [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Click “Cancel” button | Close current popup. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Respond | | 1 | Cannot update permission of membership | Display error message cannot update permission. |   **Relationships:** View Permission  **Business Rules:**   * With checkbox is described true or false * The number of website will bound the number of sites the role user can create. | | | |

Table 17: : <Admin> Update Permission use case specification table

##### <Admin> Change Status Account

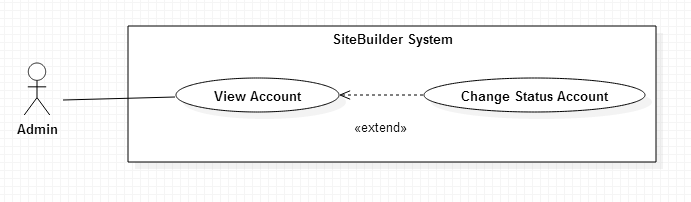


Figure 15: <Admin> Change Status Account use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCA13** | | | |
| **Use Case No.** | UCA13 | **Use Case Version** | 1.0 |
| **Use Case Name** | Change status account | | |
| **Author** | Phạm Thị Thu Hòa | | |
| **Date** | 01/06/2015 | **Priority** | Normal |
| **Actor:**   * Admin   **Summary:**   * This use case is changed status account.   **Goal:**   * Admin can change status of account.   **Triggers:**   * Click “Account” menu. * Click on “Active”/”Inactive” button.   **Preconditions:**   * User must be logged in the system with Admin role.   **Post Conditions:**   * **Success:** Status of use is changed. * **Fail:** Cannot change status of account and display error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on “Active” button.  [Alternative 1] | Show popup:   * “Are you sure to change status of account” * “Yes” button. * “No” button” | | 2 | Click “Yes” button in popup  [Alternative 2] | Status is changed from Inactive to Active.  Show message change status successful.  [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Clicks on “Inactive” button  Click “Yes” button in popup  [Alternative 2] | Show popup:   * “Are you sure to change status of account” * “Yes” button.   “No” button”  Status is changed from Active to Inactive.  Show message change status successful.  [Exception 1] | | 2 | Click “No” button | Popup is closed |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Cannot change status due to database connection | Display error message: “Cannot change status. Please try later.” |   **Relationships:** View Account  **Business Rules:**   * If admin click on “Active” button, status is changed from “Inactive” to “Active”. * If admin click on “Inactive” button, status is changed from “Active” to “Inactive”. * When status is “Active”, button “Active” cannot access. When status is “Inactive”, button “Inactive” cannot access. * When user account has status “Inactive”. This user cannot login system. (account pending). | | | |

Table 18: <Admin> Change Status Account use case specification table

##### <Admin> View Business Type

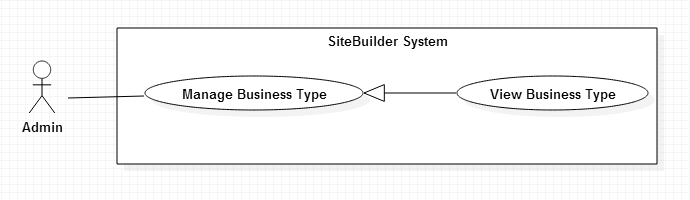


Figure 16: <Admin> View Business Type use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UCA14** | | | |
| **Use Case No.** | UCA14 | **Use Case Version** | 1.0 |
| **Use Case Name** | View Business Type | | |
| **Author** | Trần Hoàn Anh Nguyên | | |
| **Date** | 01/06/2015 | **Priority** | Normal |
| **Actor:**   * Admin   **Summary:**   * This use case allows admin to view all business type in system.   **Goal:**   * Admin can view all business type in system.   **Triggers:**   * From the menu bar:choose “Business Type” menu.   **Preconditions:**   * User must log in the system with admin role.   **Post Conditions:**   * **Success:** List business type displayed. * **Fail:** Cannot load list business type.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on “Permission” menu | Display list of business type:  Name: text  Description: text  Create date: date time  Update: button  Delete: button  [Alternative 1] [Exception 1] | |  |  |  |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Clicks on “Business type” menu, with no permission | Display error message: “Do not have any permission in system.” |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Cannot view list business type. | Display error message: “Cannot load list business type. Please try later.” |   **Relationships:** Manage Business Type  **Business Rules:**   * List of business types are defined by admin accordance with common type of business in society. Based on this list the users can easily find the suitable template for their websites. | | | |

Table 19: <Admin> View Business Type use case specification table

#### <User> Overview Use Case

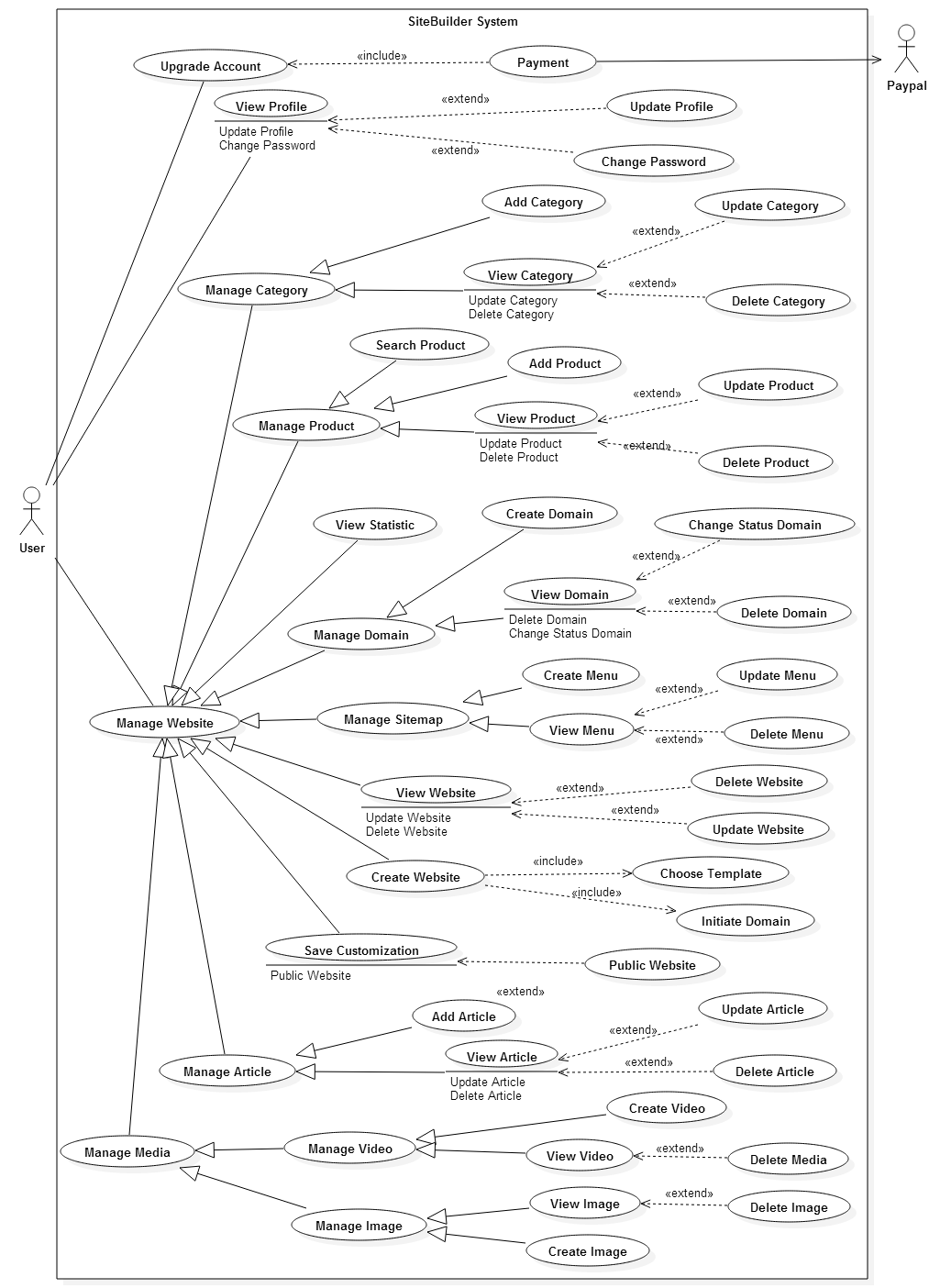


Figure 17: User Overview Use Case

##### <User> View Website

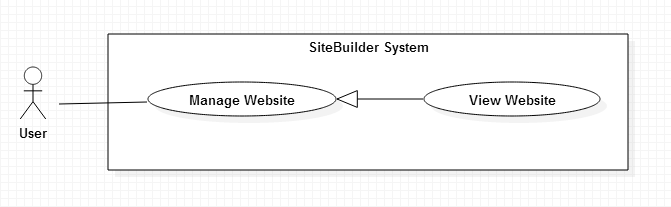


Figure 18: <User> View Website use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCU01** | | | |
| **Use Case No.** | UCU01 | **Use Case Version** | 1.0 |
| **Use Case Name** | View Website | | |
| **Author** | Trương Minh Tiến | | |
| **Date** | 01/06/2015 | **Priority** | Normal |
| **Actor:**   * User   **Summary:**   * This use case allows user to view their websites.   **Goal:**   * User can view all websites of user.   **Triggers:**   * From the menu bar: Choose “Website” menu.   **Preconditions:**   * User must log in the system with User role.   **Post Conditions:**   * **Success:** List websites displayed. * **Fail:** Cannot load list websites.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on “Website” menu | Display list of websites:  Thumbnail: image  Name: text  Business type: text  Status: text  Create date: text  Edit: button  Delete: button  View: button  [Alternative 1] [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Clicks on “Website” menu, with no website | Display error message: “Do not have any website in system.” |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Cannot view list website. | Display error message: “Cannot load list website. Please try later.” |   **Relationships:** Manage Website  **Business Rules:**   * Show list website of user. | | | |

Table 20: <User> View Website use case specification table

##### <User> Create Website

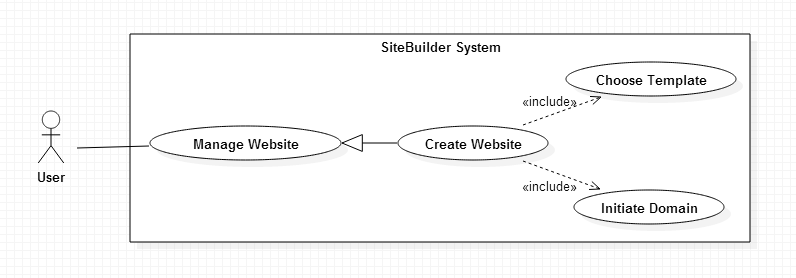


Figure 19: <User> Create Website use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCU02** | | | |
| **Use Case No.** | UCU02 | **Use Case Version** | 1.0 |
| **Use Case Name** | Create Website | | |
| **Author** | Trương Minh Tiến | | |
| **Date** | 01/06/2015 | **Priority** | High |
| **Actor:** User  **Summary:**   * This use case is used to create a new website.   **Goal:**   * Admin can add new website.   **Triggers:**   * Click “Website” menu. * From mange Website page click “Add” button.   **Preconditions:**   * User must be logged in the system with User role.   **Post Conditions:**   * **Success:** New website is added to system * **Fail:** Cannot add new website to system.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on “Add” button. | Redirect to Choose template page:   * Business type: dropdown list * List of template corresponding with business type. | | 2 | Click on Choose template button  [Alternative 1] | Redirect to Create website page:   * Website name: textbox, required. * Website domain: required   + Default domain: radio button, checked; Name of domain: text.  +Custom domain: radio button, Name of domain: text.   * Create: button * Back: button   [Exception 1] | | 4 | Clicks on “Create” button | New website is added to database and redirect to Manage Website page.  [Exception 2, 3] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Clicks on “Back” button | Redirect to Choose template page. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Not input all required fields | Display “Please input all required information” message | | 2 | Cannot create new website due to database connection. | Display message Cannot create new website. Please try again. | | 3 | Cannot create new website due to limit permission | Display message Cannot create new website. Please upgrade account to continue. |   **Relationships:** Initiate Domain, Choose Template  **Business Rules:**   * When user create new website, system check permission of user: number of websites user can create and storage. * System will suggest a default domain, if user input another domain different with suggested domain, system will check domain inputted to make sure that it is not exist in system. * System will clone folder template to folder website after that check number and name page of template to insert database. | | | |

Table 21: <User> Create Website use case specification table

##### <User> Delete Website

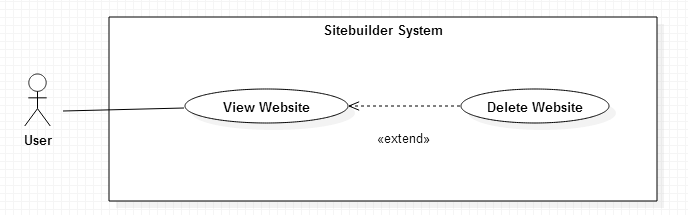


Figure 20: <User> Delete Website use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UCU03** | | | |
| **Use Case No.** | UCU03 | **Use Case Version** | 1.0 |
| **Use Case Name** | Delete Business Type | | |
| **Author** | Trương Minh Tiến | | |
| **Date** | 01/06/2015 | **Priority** | Medium |
| **Actor:**   * User   **Summary:**   * This use case is used to remove website.   **Goal:**   * User can remove business type.   **Triggers:**   * Click “Website” menu. * Click on “Delete” button.   **Preconditions:**   * User must be logged in the system with User role.   **Post Conditions:**   * **Success:** Website is removed from database * **Fail:** Cannot remove website and display error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on “Delete” button | Show popup:   * “Are you sure to delete” * “Yes” button. * “No” button” | | 2 | Click “Yes” button  [Alternative 1] | Website is deleted and show message delete successful.  [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Clicks on “No” button | Close current popup |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Cannot remove Website due to database connection | Display error message: “Cannot delete Website. Please try later.” |   **Relationships:** View Website  **Business Rules:**   * When delete website, system will check items of website (page, domain, media, article) and also delete all of them. * After delete items of website, system will delete record website in database and folder is used to save website. | | | |

Table 22: <User> Delete Website use case specification table

##### <User> Update Website



Figure 21: <User> Update Website use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UCU04** | | | |
| **Use Case No.** | UCU04 | **Use Case Version** | 1.0 |
| **Use Case Name** | Delete Website | | |
| **Author** | Trương Minh Tiến | | |
| **Date** | 01/06/2015 | **Priority** | Medium |
| **Actor:**   * User   **Summary:**   * This use case is used to update information of website.   **Goal:**   * User can update information of website.   **Triggers:**   * Click “Website” menu. * Click on Website name.   **Preconditions:**   * User must be logged in the system with User role.   **Post Conditions:**   * **Success:** Information of website is updated. * **Fail:** Cannot update information of website and display error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on Website name | Redirect to Website detail page and display:  Manage website:   * Product link: redirect to manage Product page. * Media link: redirect to manage Media page. * Domain link: redirect to manage Domain page. * Article link: redirect to manage Article page. * Statistic link: redirect to view Statistic page. * Modify content website link: redirect to Custom page.   Information of website:   * Logo website: button choose file * Website name: textbox, required, length 3-32 characters. * Email: textbox, required, length 3-250 characters, email format. * Address: textbox, length 3-250 characters. * Website’s footer: text area, length 3-250 characters. * Business type: dropdown list. * Status: dropdown list, default is Active. * Home page: dropdown list. * Save change: button. * View: button. | | 2 | User input information and click Save button | [Exceptions 1,2 3]  Information of website is changed and show message. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Clicks on “View” button. | Redirect to page review website. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Not input all required fields. | Display message to notify user input all required field. |   **Relationships:** View Website  **Business Rules:**   * Website information is loaded form database. * Status of website include: Active (website is publishing), Inactive (website has not published yet) and Block (website is cannot modify due to membership change). * Logo can use to binding data on header widget of website. * Footer information can use to binding data on footer widget of website. * Email and address can use to binding data on widget contact of website. | | | |

Table 23: <User> Update Website use case specification table

##### <User> Save Customization

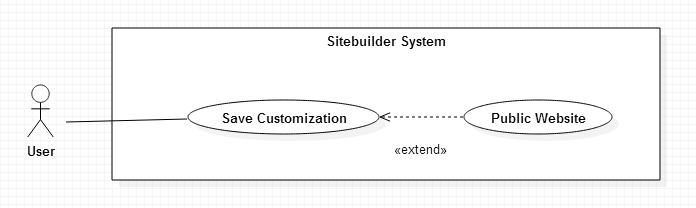


Figure 22: <User> Save Customization use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCU05** | | | |
| **Use Case No.** | UCU05 | **Use Case Version** | 1.0 |
| **Use Case Name** | Save Customization | | |
| **Author** | Trương Minh Tiến | | |
| **Date** | 01/06/2015 | **Priority** | High |
| **Actor:**   * User   **Summary:**   * This use case is used to save and public website.   **Goal:**   * User can save or save and public website.   **Triggers:**   * From Manage Website Detail page: click on “Customization” button. * From Manage Website page: click on “Customization” link.   **Preconditions:**   * User must be logged in the system with User role. * System must be exist at least one widget.   **Post Conditions:**   * **Success:** Website is public. * **Fail:** Cannot public website and display error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on “Customization” link | Show page custom:   * List widget * “Drag and drop” checkbox * “Change content” checkbox * “Custom detail” checkbox * “Save website” button. * “Save and public” button * “View” button | | 2 | Click “Save” button  [Alternative 1, 2] | Popup is closed.  Customizable website is saved and system show message.  [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Clicks on “View” button | View demo before public website. | | 2 | Clicks on “Save and Public” button | Popup is closed.  Customizable website is saved and status of website change to public  Show message success.  [Exception 2] | | 3 | Click on Close button | Close current popup |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Cannot save Website | Display error message to notify that cannot save website. | | 2 | Cannot save and public Website | Display error message to notify that cannot save and public website. |   **Relationships:** Public website  **Business Rules:**   * When user choose “Drag and drop” checkbox, list widgets are displayed in left menu and user can drag and drop widget to current page. * When user choose ““Change content” checkbox, user can click on each item in widget and change display text. * When user choose ““Custom detail” checkbox, user can change how widget is displayed such as color, font, size….In this mode, user also can bind data from media, products, articles to widget. * When user save and public successful, status of website is changed to Public and website can access by domain from out site. | | | |

Table 24: <User> Save Customization use case specification table

##### <User> View Product

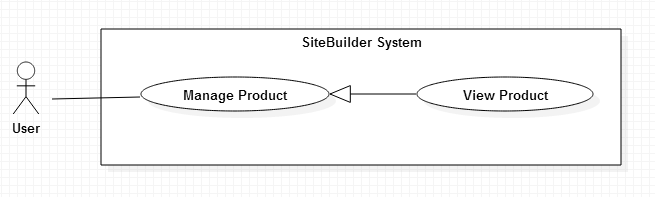


Figure 23: <User> View Product use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UCU06** | | | |
| **Use Case No.** | UCU06 | **Use Case Version** | 1.0 |
| **Use Case Name** | View Product | | |
| **Author** | Phạm Thị Thu Hòa | | |
| **Date** | 01/06/2015 | **Priority** | High |
| **Actor:**   * User   **Summary:**   * This use case allows user to view all product in website.   **Goal:**   * User can view to view products in system.   **Triggers:**   * From the menu bar: Choose “Product” menu.   **Preconditions:**   * User must log in the system with User role.   **Post Conditions:**   * **Success:** List product is displayed. * **Fail:** Cannot load list product.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on “Product” menu | Display list of product:   * Thumbnail: image * Name: text * Price: text * Edit: button * Delete: button   [Alternative 1] [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | List of product is empty | Display error message: “No product in system”. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Cannot view list product. | Display error message: “Cannot load list product .Please try later.” |   **Relationships:** Manage Product.  **Business Rules:**   * List product of website with thumbnail of product. * Products are sorted by create date. | | | |

Table 25: <User> View Product use case specification table

##### <User> Create Product

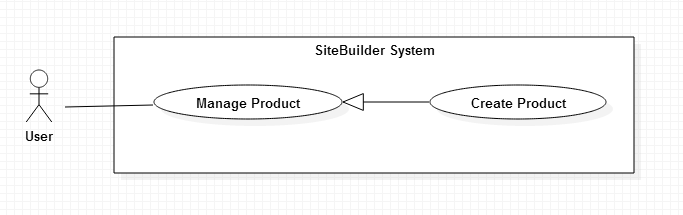


Figure 24: <User> Create Product use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCU07** | | | |
| **Use Case No.** | UCU07 | **Use Case Version** | 1.0 |
| **Use Case Name** | Create Product | | |
| **Author** | Phạm Thị Thu Hòa | | |
| **Date** | 01/06/2015 | **Priority** | High |
| **Actor:** User  **Summary:**   * This use case is used to create a new product.   **Goal:**   * User can add new product.   **Triggers:**   * From menu bar choose “Product” in menu * Click “Add” button.   **Preconditions:**   * User must be logged in the system with User role. * User must be created website before. * Website exist at least one category.   **Post Conditions:**   * **Success:** New product is added to system * **Fail:** Cannot add new product to system.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on “Add” button. | Display popup:   * Product name: Textbox, required, length 3-35. * Product price: Textbox, required, number format, value from 0 to 1 billion. * Product category: Dropdown list * Product status:   +Disable button  +Enable button   * Product image: required   + Upload tab: choose image button, upload button.  + Library image tab: list image in website (image and checkbox)   * Submit: button * Cancel: button | | 2 | [Alternative 1]  [Alternative 2]  Inputs information into all fields and clicks “Add” button  [Alternative 3] | Add new product to database display message create successful new product.  [Exception 1] [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | User clicks “Upload image”  tab  User select images from popup and click “Open”  [Alternative 4] | Open pop up [Select Image]  Images are inserted to Media and move to tab “Library picture” | | 2 | User clicks “Library picture” tab  User clicks in checkboxes to select image | Show all images in media.  Selected Images are checked. | | 3 | User clicks “Cancel” button. | Pop up [Add Product] is closed. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | User does not input all required field. | Display message “Please input required information.” | | 2 | Cannot create new product | Display message “Cannot create new product. Please try again.” |   **Relationships:** Manage Product  **Business Rules:**   * User can upload new images or choose images exist in website. * After user upload images, all images are uploaded is add to media of website and move to Library image tab. * Data of products used to binding data to create user’s website. * If product status is Disable, product is not displayed in list of product when use “Product gallery” Widget. | | | |

Table 26: <User> Create Product use case specification table

##### <User> Update Product

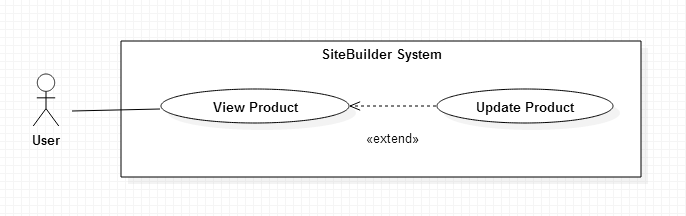


Figure 25: <User> Update Product use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCU08** | | | |
| **Use Case No.** | UCU08 | **Use Case Version** | 1.0 |
| **Use Case Name** | Update Product | | |
| **Author** | Phạm Thị Thu Hòa | | |
| **Date** | 01/06/2015 | **Priority** | High |
| **Actor:** User  **Summary:**   * This use case is used to update product.   **Goal:**   * User can update product.   **Triggers:**   * From view Product page * Click “Update” button.   **Preconditions:**   * User must be logged in the system with User role. * Product is updated exist in system.   **Post Conditions:**   * **Success:** Information of product is updated to database. * **Fail:** Cannot update product to database.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on “Update” button. | Display popup:   * Product name: Textbox, required, length 3-35. * Product price: Textbox, required, number format, value from 0 to 1 billion. * Product category: Dropdown list * Product status:   +Disable button  +Enable button   * Product image: required   + Upload tab: choose image button, upload button.  + Library image tab: list image in website (image and checkbox)   * Submit: button * Cancel: button | | 2 | [Alternative 1]  [Alternative 2]  Inputs information into all fields and clicks “Add” button  [Alternative 3] | Update information of product to database and display message: update successful product.  [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | User clicks “Upload image”  tab  User select images from popup and click “Open”  [Alternative 4] | Open pop up [Select Image]  Images are inserted to Media and move to tab “Library picture” | | 2 | User clicks “Library picture” tab  User clicks in checkboxes to select image | Show images of product and image in Library picture.  Selected Images are checked. | | 3 | User clicks “Cancel” button. | Pop up [Update Product] is closed. | | 4 | User clicks “Cancel” button. | Pop up [Select Image] is closed. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | User does not input all required field. | Display message “Please input required information.” | | 2 | Cannot update product | Display message “Cannot update product. Please try again.” |   **Relationships:** View Product  **Business Rules:**   * When popup update image open, Library Image tab display list images of product and list images in website. * Product name should be different from existed product of system. | | | |

Table 27: <User> Update Product use case specification table

##### <User> Delete Product

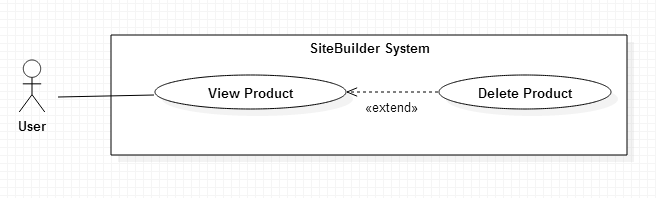


Figure 26: <User> Delete Product use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UCU09** | | | |
| **Use Case No.** | UCU09 | **Use Case Version** | 1.0 |
| **Use Case Name** | Delete Product | | |
| **Author** | Phạm Thị Thu Hòa | | |
| **Date** | 01/06/2015 | **Priority** | Medium |
| **Actor:**   * User   **Summary:**   * This use case is used to remove product.   **Goal:**   * User can remove product.   **Triggers:**   * From View product menu * Click on “Delete” button.   **Preconditions:**   * User must be logged in the system with User role.   **Post Conditions:**   * **Success:** Product is removed from database * **Fail:** Cannot remove product and display error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on “Delete” button | Show popup:   * “Are you sure to delete?” * “Yes” button. * “No” button” | | 2 | Click “Yes” button  [Alternative 1] | Product is deleted from database.  Show message delete successful.  [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Clicks on “No” button | Close current popup |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Cannot remove product due to database connection | Display error message cannot delete product. |   **Relationships:** View Product  **Business Rules:**   * Before system delete product, system will find relationship between product and images and delete it. But those images still exits in Media of website, user can delete it at page Manage Media. | | | |

Table 28: <User> Delete Product use case specification table

##### <User> Search Product

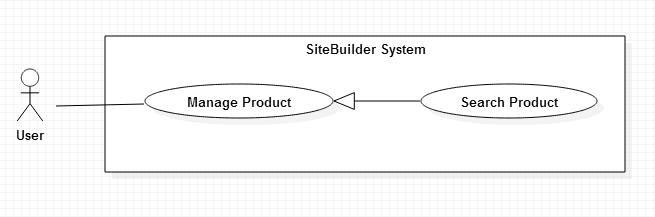


Figure 27: <User> Search Product use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCU10** | | | |
| **Use Case No.** | UCU10 | **Use Case Version** | 1.0 |
| **Use Case Name** | View Product | | |
| **Author** | Phạm Thị Thu Hòa | | |
| **Date** | 01/06/2015 | **Priority** | High |
| **Actor:**   * User   **Summary:**   * This use case allows user to view all product in website.   **Goal:**   * User can search to view products in system.   **Triggers:**   * From the menu bar: Choose “Product” menu.   **Preconditions:**   * User must log in the system with User role.   **Post Conditions:**   * **Success:** List product is displayed. * **Fail:** Cannot load list product.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on “Product” menu  [Alternative 1]  Input search text and press Enter  [Alternative 2] | Display textbox search: Textbox  Display message the number of products are found and list of product.  [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | User click on Category name | Display list products belong to category. | | 2 | User do not input search text and press Enter. | Display all products in current category. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Cannot search list product due to database connection. | Display error message cannot search product. |   **Relationships:** Manage Product.  **Business Rules:**   * Search textbox corresponding with search in category of product. After user search on category and input search text, result return is products that belong to this category. * User can sort list products by: price, create date and alphabet name. | | | |

Table 29: <User> Search Product use case specification table

##### <User> View Statistic

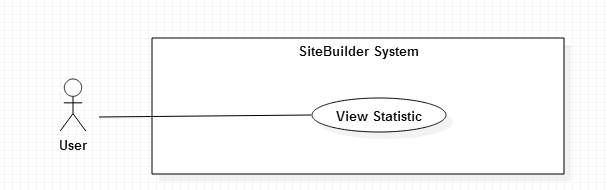


Figure 28: <User> View Statistic use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCU11** | | | |
| **Use Case No.** | UCU11 | **Use Case Version** | 1.0 |
| **Use Case Name** | View Statistic | | |
| **Author** | Lê Trần Thịnh | | |
| **Date** | 01/06/2015 | **Priority** | High |
| **Actor:**   * User   **Summary:**   * This use case is used to view statistic of website.   **Goal:**   * User can view statistic of website.   **Triggers:**   * User send view statistic request.   **Preconditions:**   * User must be logged in the system with User role. * Website is created before view statistic.   **Post Conditions:**   * **Success:** Statistic of website is displayed. * **Fail:** Cannot display statistic of website and display error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | User send view statistic request. | Display statistic of system:  +Storage of website  [Exception 1]  +Number of product  [Exception 2]  +Monthly statistics: number of number of visits per month.  [Exception 3]  +Daily statistics: number of number of visits per day.  [Exception 4]  +Top product: list of the most viewed products:  Name: text  Icon: image  Views: number  [Exception 5] |   **Alternative Scenario:** N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Cannot count storage of website | Display error message to notify system cannot count storage of website. | | 2 | Cannot count number of products of website. | Display error message to notify system cannot count number of products of website. | | 3 | Cannot count number of visits per mouth. | Display error message to notify system cannot count number of visits of website. | | 4 | Cannot count number of visits per day. | Display error message to notify system cannot count number of visits of website. | | 5 | Cannot show list of product | Display error message to notify system cannot show list most viewed products. |   **Relationships:** N/A  **Business Rules:**   * Daily statistics will retrieve data during the last 30 days. * Monthly statistics will retrieve data during the last 12 months. * Display top 10 products have most views. * Products of website include products with status is Disable. * Storage website is calculated based on size of folder is used to save website. | | | |

Table 30: <User> View Statistic use case specification table

##### <User> View Domain

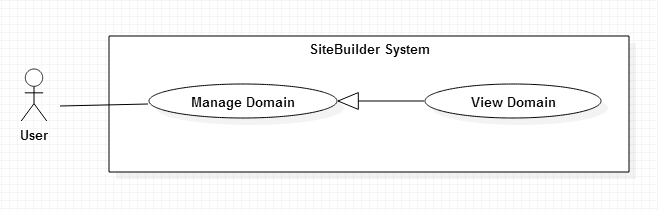


Figure 29: <User> View Domain use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UCU12** | | | |
| **Use Case No.** | UCU12 | **Use Case Version** | 1.0 |
| **Use Case Name** | View Domain | | |
| **Author** | Trương Minh Tiến | | |
| **Date** | 01/06/2015 | **Priority** | Medium |
| **Actor:**   * User   **Summary:**   * This use case allows user to view all domains in websites.   **Goal:**   * User can view domains in websites.   **Triggers:**   * User send request view domain websites.   **Preconditions:**   * User must log in the system with User role.   **Post Conditions:**   * **Success:** List product is displayed. * **Fail:** Cannot load list product.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | User send request view domain websites | Display list of domains:   * Name: text * Type: text * Status: text * Name of website: text * Delete: button * Change to public: button   [Alternative 1] [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | List of domain is empty | Display message to notify user do not have any domain in system. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Cannot view list domain. | Display error message cannot load list domains |   **Relationships:** Manage Domain.  **Business Rules:**   * Each domain has two statuses: Active (domain is used to public website) and Inactive (domain is not used to public website). | | | |

Table 31: <User> View Domain use case specification table

##### <User> Change Status Domain

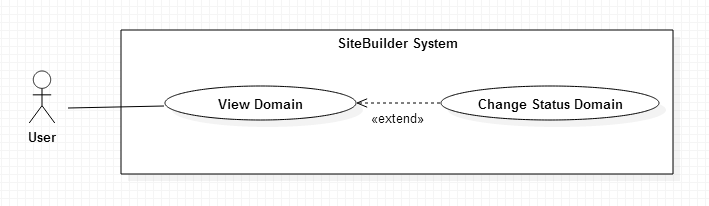


Figure 30: <User> Change Status Domain use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCU15** | | | |
| **Use Case No.** | UCU15 | **Use Case Version** | 1.0 |
| **Use Case Name** | Change status domain | | |
| **Author** | Trương Minh Tiến | | |
| **Date** | 01/06/2015 | **Priority** | High |
| **Actor:**   * User   **Summary:**   * This use case is changed status domain.   **Goal:**   * User can change status of domain.   **Triggers:**   * User send change status domain request.   **Preconditions:**   * User must be logged in the system with User role. * Domain must be exits in the system.   **Post Conditions:**   * **Success:** Status of domain is changed. * **Fail:** Cannot change status of domain and display error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | User send change status domain request. | Show popup confirm:   * “Yes” button. * “No” button” | | 2 | Click “Yes” button in popup  [Alternative 1] | Status is changed from Not Public to Public of changed from Public to Not Public.  Show message change status successful.  [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Click “No” button | Popup is closed |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Cannot change domain’s status | Display error message cannot change status domain. |   **Relationships:** View Domain  **Business Rules:**   * Each domain has two statuses: Public (domain is used to public website) and not public (domain is not used to public website). * Each website has only one active domain. If website exits an active domain and user want to change status of another domain, system display a message to notify. * Domains have two types: default domain (domain provided by the system) and custom domain (domain bought by user). Each site must have at least one default domain. | | | |

Table 32: <User> Change Status Domain use case specification table

##### <User> Upgrade Account

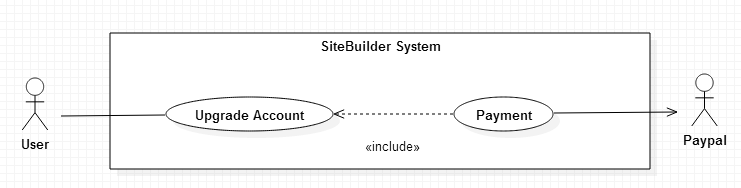


Figure 31: <User> Upgrade Account use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCU16** | | | |
| **Use Case No.** | UCU16 | **Use Case Version** | 1.0 |
| **Use Case Name** | Upgrade Account | | |
| **Author** | Trương Minh Tiến | | |
| **Date** | 01/06/2015 | **Priority** | Medium |
| **Actor:**   * User   **Summary:**   * This use case is used to upgrade account.   **Goal:**   * User can upgrade account.   **Triggers:**   * User send request upgrade account.   **Preconditions:**   * User must be logged in the system with User role. * User has not upgraded account yet.   **Post Conditions:**   * **Success:** Account’s status is changed. * **Fail:** Cannot change account’s status and display error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | User send request upgrade account | Display membership types with permissions:   * Name: text * Price: text, Unit: USD * Number of websites: text * Storage: text, unit: MB * View statistic: checkbox | | 2 | Click “Upgrade” button  After user payment in PayPal page  [Alternative 1] | Redirect to PayPal page to payment  Redirect to Upgrade account and show message membership of user is changed. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Clicks on cancel payment and back to Upgrade account page. | Display Upgrade account page. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Cannot remove product due to database connection | Display error message: “Cannot delete product. Please try later.” |   **Relationships:**  N/A  **Business Rules:**   * When user payment success, permission of user is changed in a month from payment date. Upon expiration, system will notify user payment for next month or enforce user choose features are limited. * Next month, if user does not payment in PayPal to keep new role account. User will down levels role account. * With number of website user created exceed the limit of role account, system will restrict edit some websites. | | | |

Table 33: <User> Upgrade Account use case specification table

##### <User> View Article

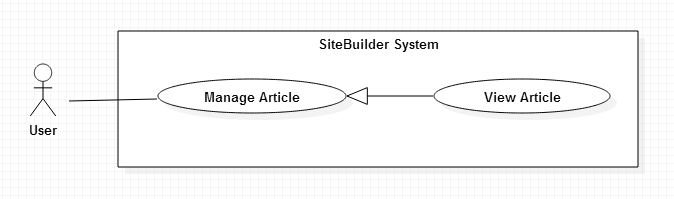


Figure 32: <User> View Article use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UCU17** | | | |
| **Use Case No.** | UCU17 | **Use Case Version** | 1.0 |
| **Use Case Name** | View Article | | |
| **Author** | Lê Trần Thịnh | | |
| **Date** | 01/06/2015 | **Priority** | Medium |
| **Actor:**   * User   **Summary:**   * This use case allows user to view list articles in a website.   **Goal:**   * User can view all article in a website.   **Triggers:**   * From the Website detail page: Click “Manage Article” link.   **Preconditions:**   * User must log in the system with User role.   **Post Conditions:**   * **Success:** List articles are displayed. * **Fail:** Cannot load list articles.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click “Manage Article” link. | Display list of article include:  Article thumbnail: image  Name: text  Create date: Date time  Update: button  Delete: button  [Alternative 1] [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | List article is empty. | Display message notify that website do not have any article. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Cannot view list articles. | Display error message cannot load list articles. |   **Relationships:** Manage Article  **Business Rules:**   * N/A | | | |

Table 34: <User> View Article use case specification table

##### <User> Create Article

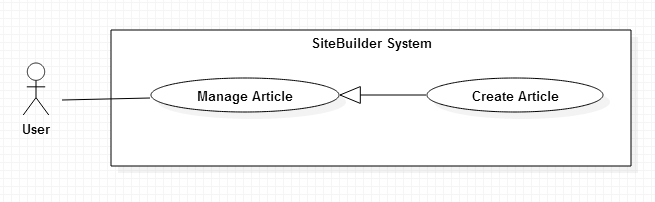


Figure 33: <User> Create sArticle use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCU18** | | | |
| **Use Case No.** | UCU18 | **Use Case Version** | 1.0 |
| **Use Case Name** | Create article | | |
| **Author** | Lê Trần Thịnh | | |
| **Date** | 01/06/2015 | **Priority** | High |
| **Actor:**   * User   **Summary:**   * This use case is used to create new article in a website.   **Goal:**   * User can create new article.   **Triggers:**   * From the Manage Article page: Click “Add” button   **Preconditions:**   * User must be logged in the system with User role.   **Post Conditions:**   * **Success:** New page add to website * **Fail:** Cannot add new page to website and display error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on “Add” button. | Display popup:   * Name: Textbox, required, length 3-255 characters. * Content: text area, required. * Thumbnail: image. * Submit button. | | 2 | Click Submit button. | New article is added to website and show message.  [Exception 1, 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Click Close button | Popup is closed. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Not input required input. | Display message to notify user input required information. | | 2 | Cannot create new article due to database connection | Display error message: “Cannot create new article. Please try later.” |   **Relationships:** Manage Article.  **Business Rules:**   * Data of article used to binding data to create user’s website. * Users can add pictures or videos into an article. * Thumbnail will get first pictures in article. | | | |

Table 35: <User> Create Article use case specification table

##### <User> Update Article



Figure 34: <User> Update Article use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCU19** | | | |
| **Use Case No.** | UCU19 | **Use Case Version** | 1.0 |
| **Use Case Name** | Update article | | |
| **Author** | Lê Trần Thịnh | | |
| **Date** | 01/06/2015 | **Priority** | High |
| **Actor:**   * User   **Summary:**   * This use case is used to update article in a website.   **Goal:**   * User can update article.   **Triggers:**   * From manage Article: User click Add button   **Preconditions:**   * User must be logged in the system with User role. * Article is existed in system.   **Post Conditions:**   * **Success:** Article add to website * **Fail:** Cannot add new page to website and display error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on “Add” button. | Display popup:   * Name: Textbox, required, length 3-255 characters. * Content: text area, required. * Thumbnail: image. * Submit button. | | 2 | Click Submit button. | New article is added to website and show message.  [Exception 1, 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Click Close button | Popup is closed. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Not input required input. | Display message to notify user input required information. | | 2 | Cannot update article due to database connection | Display error message: “Cannot update article. Please try later.” |   **Relationships:** View Article.  **Business Rules:**   * User can update title and content of article. * All change in article is updated also change on article Widget. | | | |

Table 36: <User> Update Article use case specification table

##### <User> Delete Article

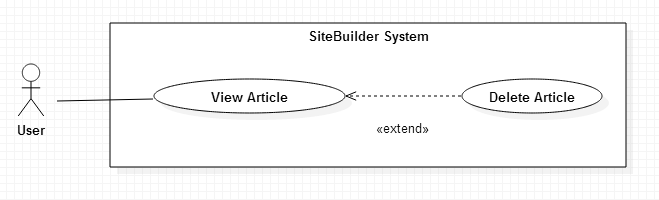


Figure 35: <User> Delete Article use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCU20** | | | |
| **Use Case No.** | UCU20 | **Use Case Version** | 1.0 |
| **Use Case Name** | Delete Article | | |
| **Author** | Lê Trần Thịnh | | |
| **Date** | 01/06/2015 | **Priority** | Medium |
| **Actor:**   * User   **Summary:**   * This use case is used to remove article.   **Goal:**   * User can remove article.   **Triggers:**   * From Manage Article page: click on “Delete” button   **Preconditions:**   * User must be logged in the system with User role.   **Post Conditions:**   * **Success:** Article is removed from database * **Fail:** Cannot remove article and display error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on “Delete” button | Show popup confirm:   * “Are you sure to delete?” * “Yes” button. * “No” button” | | 2 | Click “Yes” button  [Alternative 1] | Article is deleted from database.  Show message delete successful.  [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Clicks on “No” button | Close current popup |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Cannot remove article due to database connection | Display error message: “Cannot delete article. Please try later.” |   **Relationships:** View Article  **Business Rules:**   * When delete article, thumbnail is deleted but images and videos in content still belong to Media of website. | | | |

Table 37: <User> Delete Article use case specification table

##### <User> View Image

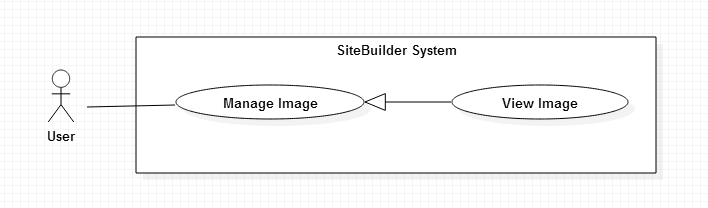


Figure 36: <User> View Image use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UCU22** | | | |
| **Use Case No.** | UCU22 | **Use Case Version** | 1.0 |
| **Use Case Name** | View Image | | |
| **Author** | Lê Trần Thịnh | | |
| **Date** | 01/06/2015 | **Priority** | Medium |
| **Actor:**   * User   **Summary:**   * This use case allows user to view list image in a website.   **Goal:**   * User can view all image in a website.   **Triggers:**   * From Manage Media page: click on “Image” tab   **Preconditions:**   * User must log in the system with User role.   **Post Conditions:**   * **Success:** List image are displayed. * **Fail:** Cannot load list image.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click “Image” tab.  [Alternative 1] | Display list image  [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | List image is empty. | Display message notify that website do not have any image. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Cannot view list image. | Display error message cannot load list image. |   **Relationships:** Manage Media  **Business Rules:**  Media of website includes: images and videos.   * Images are used to binding data to widget image of website. | | | |

Table 38: <User> View Image use case specification table

##### <User> Create Image



Figure 37: <User> Create Image use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCU23** | | | |
| **Use Case No.** | UCU23 | **Use Case Version** | 1.0 |
| **Use Case Name** | Create image | | |
| **Author** | Lê Trần Thịnh | | |
| **Date** | 01/06/2015 | **Priority** | High |
| **Actor:**   * User   **Summary:**   * This use case is used to create new image in a website.   **Goal:**   * User can create new image.   **Triggers:**   * User click on “Manage Media” link: choose “Add” button on Image tab   **Preconditions:**   * User must be logged in the system with User role. * User must be created website before.   **Post Conditions:**   * **Success:** New image is added to website * **Fail:** Cannot add new image to website and display error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on “Add” button. | Display popup:   * Choose file:button, required. * Cancel: button. * Submit button. | | 2 | Click Submit button. | New image is added to website and show message.  [Exception 1, 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Click Cancel button | Popup is closed. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Not input required input. | Display message to notify user input required information. |   **Relationships:** Manage Image.  **Business Rules:**   * When create new image, image file is uploaded to system and saved in media folder of each website. | | | |

Table 39: <User> Create Image use case specification table

##### <User> View Menu

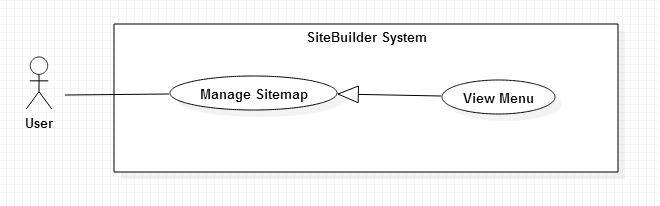


Figure 38: <User> View Menu use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCA27** | | | |
| **Use Case No.** | UCU27 | **Use Case Version** | 1.0 |
| **Use Case Name** | View Menu | | |
| **Author** | Trần Hoàn Anh Nguyên | | |
| **Date** | 01/06/2015 | **Priority** | Normal |
| **Actor:**   * User   **Summary:**   * This use case is used to view website’s sitemap.   **Goal:**   * User can view website’s sitemap.   **Triggers:**   * From Manage Website page: click on “View sitemap” link.   **Preconditions:**   * User must be logged in the system with User role.   **Post Conditions:**   * **Success:** List sitemap is displayed. * **Fail:** Cannot show list sitemap and display error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click on “Sitemap” link | Display list menu item:   * Name: text * Edit: button * Delete: button |   **Alternative Scenario:** N/A  **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Cannot display list sitemap database connection. | Display error message: “Cannot view sitemap. Please try later.” |   **Relationships:** Manage Sitemap.  **Business Rules:**   * This sitemap is used to binding data to “Menu Widget” on user’s website. * User can link to existed page in website or link to another website. * Order list menu item also is order list in “Menu Widget”, user can drag and drop to change order of menu item in this list. | | | |

Table 40: <User> View Menu use case specification table

##### <User> Create Menu



Figure 39: <User> Create Menu use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCU28** | | | |
| **Use Case No.** | UCU28 | **Use Case Version** | 1.0 |
| **Use Case Name** | Create Menu | | |
| **Author** | Trần Hoàn Anh Nguyên | | |
| **Date** | 01/06/2015 | **Priority** | Normal |
| **Actor:**   * User   **Summary:**   * This use case is used to create new menu item.   **Goal:**   * User can view website’s sitemap.   **Triggers:**   * From Manage Sitemap page: click on “Create” button.   **Preconditions:**   * User must be logged in the system with User role. * User must be created website before.   **Post Conditions:**   * **Success:** List sitemap is displayed. * **Fail:** Cannot show list sitemap and display error message.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on “Create” button | Show popup :   * Name menu: text, required, length 3-50 characters. * Type of link: * Page in website: checkbox, default checked. * Link to other website: checkbox * Pages in website: dropdown list * Name URL: text, required, length 3-255 characters. * Submit: button * Cancel: button | | 2 | Click “Submit” button  [Alternative 1] | [Exception 1,2]  New menu is created and display message created successful. |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Clicks on “Cancel” button | Close current popup |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Not input required input. | Display message to notify user input required information. | | 2 | Cannot create new menu due to database connection | Display error message: “Cannot create new menu. Please try later.” |   **Relationships:** View Video.  **Business Rules:**   * User can choose menu item link to a page existed in website or link to another website. * Each menu item can have one or more sub menu item but sub menu item cannot have their child. | | | |

Table 41: <User> Create Menu use case specification table

##### <User> View Category

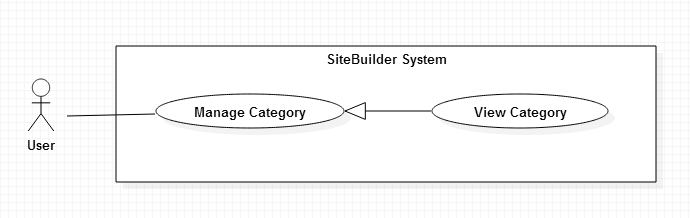


Figure 40: <User> View Category use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE - UCU33** | | | |
| **Use Case No.** | UCU33 | **Use Case Version** | 1.0 |
| **Use Case Name** | View Category | | |
| **Author** | Trần Hoàn Anh Nguyên | | |
| **Date** | 01/06/2015 | **Priority** | Normal |
| **Actor:**   * User   **Summary:**   * This use case allows user to view all category in system.   **Goal:**   * User can view to view categories in system.   **Triggers:**   * From the menu bar: Choose “Product” menu.   **Preconditions:**   * User must log in the system with User role.   **Post Conditions:**   * **Success:** List category is displayed. * **Fail:** Cannot load list category.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Clicks on “Product” menu | Display list of category in left bar:   * Name: text   [Alternative 1] [Exception 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | List of category is empty | Display message: “No category in system”. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Cannot view list category. | Display error message: “Cannot load list category. Please try later.” |   **Relationships:** Manage Category.  **Business Rules:**   * List category help to manage products more efficiently. * Each category includes many products. Users can search products by list category. | | | |

Table 42: <User> View Category use case specification table

#### <Guest> Overview Use Case

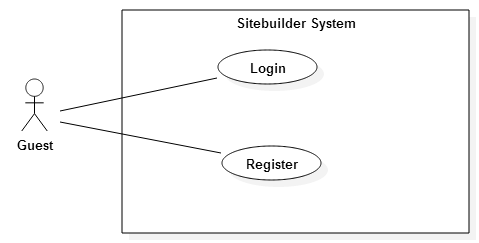


Figure 41: Guest Overview User Case

##### <Guest> Register

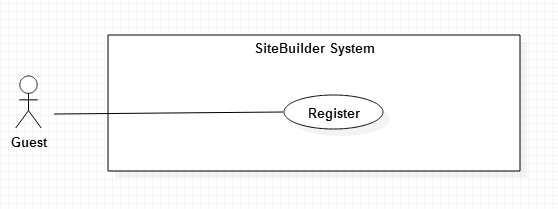


Figure 42: <Guest> Register use case diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – UCG02** | | | |
| **Use-case No.** | UCG02 | **Use-case Version** | 2.0 |
| **Use-case Name** | Register | | |
| **Author** | Trần Hoàn Anh Nguyên | | |
| **Date** | 01/06/2015 | **Priority** | Normal |
| **Actor:**   * Guest   **Summary:**   * This use case allows guest create account to use advanced functions of BTS system.   **Goal:**   * Help guest to create new account.   **Trigger:**   * Click “Register” button.   **Pre-conditions:**   * User want to create new account   **Post conditions:**   * **Success**: The new account will be added into database. * **Failure**: Cannot create account. System displays error message on register page.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click “Register” button | System redirect to [Register Page], contain a register form :   * Profile picture: button upload image * Username: textbox, required, length 6 – 255 characters. * Email: textbox, required, in format email. * Password: password field, required, length 6 – 25 characters. * Confirm password: password field, required, length 3 – 255 characters. * Address: Textbox, required, length 6 – 255 characters. * Register :button * Reset: button | | 2 | Input information.  Clicks “Register” button. | [Exceptions 1,2 3,4]  Redirect to User Dashboard view and displays popup:” You have successfully registered!” |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | No. | Actor Action | System Response | | 1 | Click on the “Reset” button. | All information in register form is cleared. |   **Exceptions:**   |  |  |  | | --- | --- | --- | | No. | Actor Action | System Response | | 1 | Username, Email, Password, Confirm password, Address are blank. | Display error message:” Please enter the required information!” | | 2 | Email is already exists. | Display error message:” This email already exists. Please input another email!” | | 3 | Password is less than 6 or more than 255 characters. | Display error message “Password 6-255 characters!” | | 4 | Guest fills password and confirm password different. | Display error message” Confirm password incorrect!” |   **Relationships:** N/A  **Business Rules**:   * Register email must be different from email of existed account of system. * Email will be used as contact information. | | | |

Table 43: <Guest> Register use case specification table

## Software System Attribute

### Usability

* Users need less than 8 hours to become familiar with the system.
* Notices, labels and text are displayed in Vietnamese.
* User can follow user manual guideline and can get support from system administrator if they have any problem.

### Reliability

N/A

### Availability

N/A

### Security

* All input are validated before inserted to database.
* Role and permission specify clearly.
* Users are authenticated and authorized when accessing to the system.

### Maintainability

* The system is divided into modules and easy to maintain and extend.

### Portability

* Admin and user can use application on every PC/laptop with supported web browser.

### Performance

* All requests are responded in no more than 1 minute at 8Mpbs bandwidth speech.

## Conceptual diagram

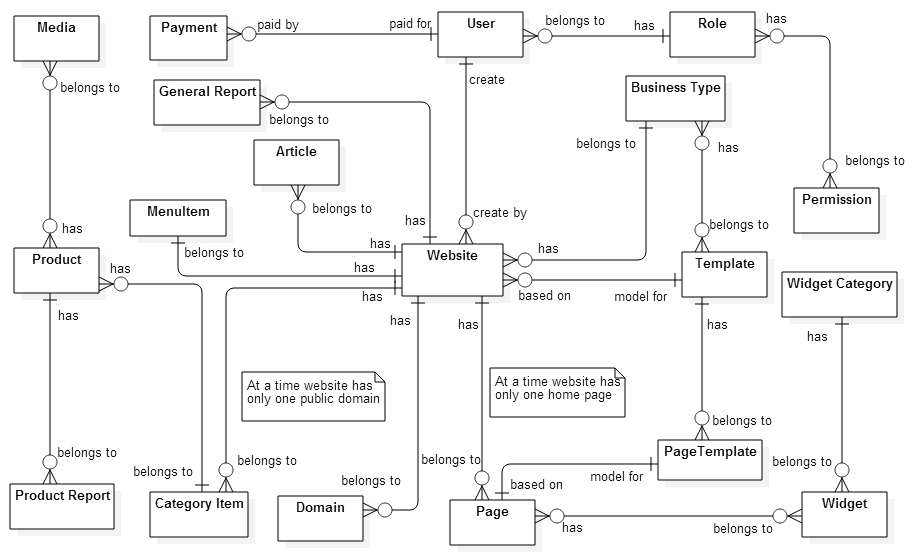


Figure 43: Conceptual Diagram

|  |  |
| --- | --- |
| Entity Data dictionary: describe content of all entities | |
| Entity Name | **Description** |
| Website | Describe website information in system. |
| Product | Describe product of website. |
| CategoryItem | Describe category item of website. |
| Article | Describe article of website. |
| Media | Describe media (image and video) of website. |
| Domain | Describe articles of website. |
| User | Describe users in system. |
| Page | Describe page of website. |
| Role | Describe roles in system. |
| Permission | Describe permission in system. |
| GeneralReport | Describe total view in a day of website. |
| ProductReport | Describe total view in a day of product. |
| Widget | Describe widget in system. |
| WidgetCategory | Describe widget categories in system. |
| Template | Describe templates in system. |
| BusinessType | Describe business type in system. |
| PageTemplate | Describe page of template. |
| Payment | Describe information of payment. |
| MenuItem | Describe the composition of the menu in website. |

# Software Design Description

## Design Overview

* This document describes the technical and user interface design of SBPC system. It includes the architectural design, the detailed design of common functions and business functions and the design of database model.
* The architectural design describes the overall architecture of the system and the architecture of each main component and subsystem.
* The detailed design describes static and dynamic structure for each component and functions. It includes class diagrams, class explanations and sequence diagrams for each use cases.
* The database design describes the relationships between entities and details of each entity.
* Document overview:
  + Section 2: gives an overall description of the system architecture design.
  + Section 3: gives component diagrams that describe the connection and integration of the system.
  + Section 4: gives the detail design description, which includes class diagram, class explanation, and sequence diagram to details the application functions.
  + Section 5: describe a fully attributed Entity Relationship Diagram.

## System Architecture Design

The system SBPC is developed from Java EE MVC Architecture style.

MVC consists of three kinds of objects. The Model is the application object, the View is its screen presentation, and the Controller defines the way the user interface reacts to user input. Based on the features of SBPC web application we choose this model for some of reasons:

**Efficient modularity**: SBPC system focus on the development user interface so we can develop business logic and user interface screen at the same time. Before, User interfaces tend to change more frequently than business rules, we can update user interface without slowing down the business logic process.

**Multiple views** **and re-use**: The separation the three components, allows we the re-use of the business logic. Multiple User Interfaces can be developed without concerning the code.

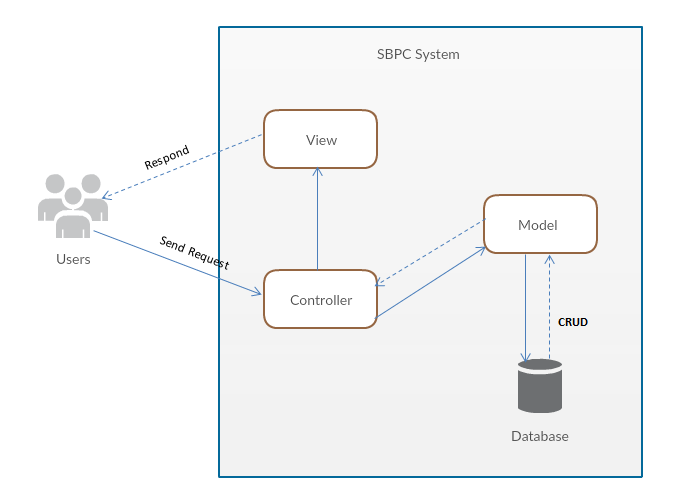


Figure 44: System Architecture Design Diagram

This project follows MVC architecture with following components:

**Controller**: Controller receives requests from users, after processing (call business logic from model) then select view return to users. Controller act like event handler connect between Model and View.

**Model**: Model represents the data and implements business logic.

**View**: View is html, Jsp files presents data to the users and how users interact with the system.

## Component Diagram

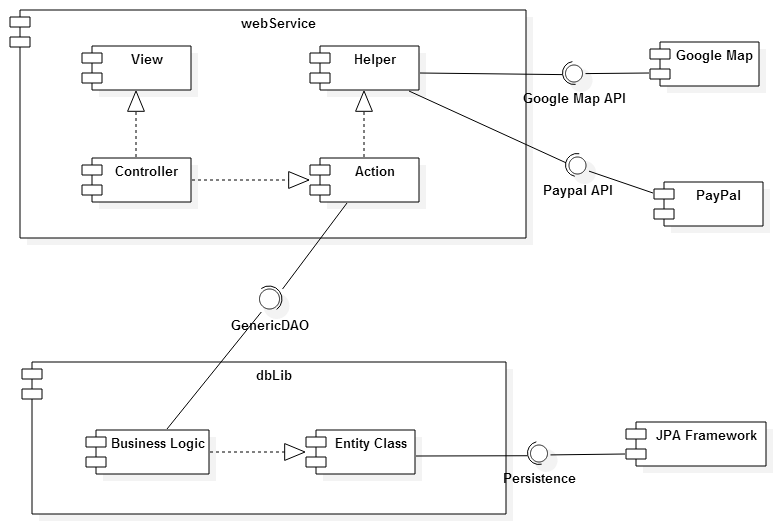


Figure 45: Component Diagram

|  |  |
| --- | --- |
| Component dictionary: describe components | |
| Component Name | **Description** |
| webService | Component include: View, Controller, Action, Helper |
| dbLib | Component include: Entity Class, Data Access |
| View | Component handle data represented |
| Controller | Component handle request from users |
| Action | Component processing requests from users |
| Helper | Component to handle processing with PayPal, Google Map. |
| Data Access | Component to handle business logic in system. |
| Entity Class | Component represents data is mapped from database. |
| JPA Framework | Handle mapping entity class from database |
| PayPal | Handle payment process with PayPal API |
| Google Map | Handle location process with Google Map API |

Table 44: Component Dictionary

## Detailed Description

### Class Diagram

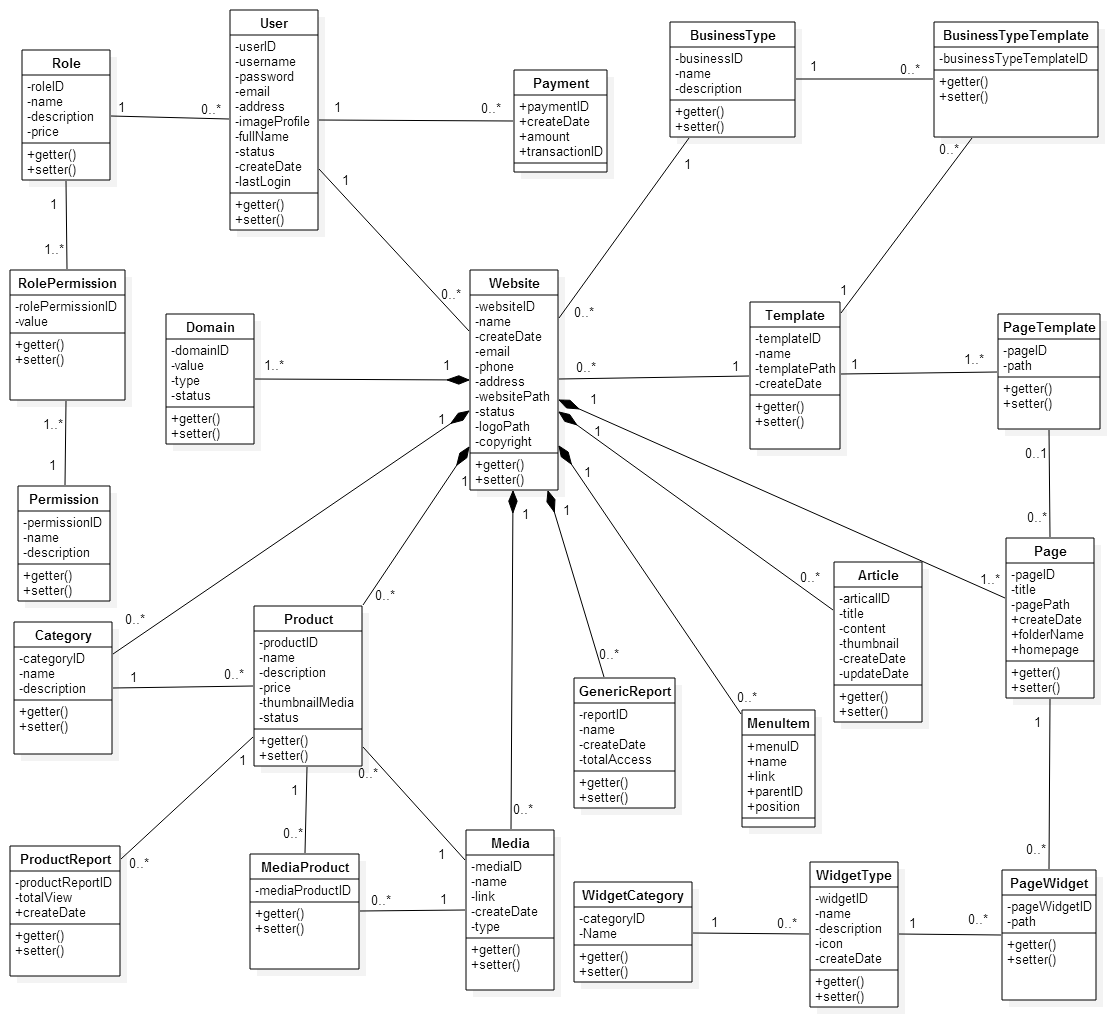


Figure 46: Class Diagram

|  |  |
| --- | --- |
| Class dictionary: describe class | |
| Class Name | **Description** |
| User | User that register or have an admin account |
| Role | Role of user in system |
| Permission | All permission that system provide |
| RolePermission | Present for relationship between role and permission. Present for relationship that each role they have many permission. |
| Website | Website that user create and publish |
| Page | Page in website |
| BusinessType | Business type of website |
| Template | Template for website |
| BusinessTypeTemplate | Present for relationship between business type and template. One business type have some coincide template |
| PageTemplate | Present for relationship between template and page. A template have some default |
| WidgetType | Some widget that allow user drag and drop for build they website |
| Domain | Website’s domain |
| Category | Category of product in website. |
| Product | Product in website. |
| Media | Media contain two types are image and video. |
| MediaProduct | Present for relationship between media and product. One product may have some media. |
| Article | Article of website. |
| PageWidget | Widget that used for a page. |
| WidgetCategory | Widget’s category. |
| GeneralReport | Report for system or for website |
| ProductReport | Report for product in website |
| Payment | Payment information of users. |
| MenuItem | Present sitemap of website. |

Table 45: Class Dictionary

### Class Diagram Explanation

#### User

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| userID | Integer | Private | Unique identifier of a user. |
| username | String | Private | Username of user. |
| password | String | Private | Password for login. |
| email | String | Private | Email of user. |

Table 46: User Class Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Type | Visibility | Description |
| getter | Attribute type | Public | Get attribute value. |
| setter | Void | Public | Set value of attribute. |

Table 47: User Class Method

#### Role

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| roleID | Integer | Private | Unique identifier of a role. |
| name | String | Private | Name of role. |
| description | String | Private | More information about that role. |
| price | Float | Private | Price of service package in a month. |

Table 48: Role Class Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Type | Visibility | Description |
| getter | Attribute type | Public | Get attribute value. |
| setter | Void | Public | Set value of attribute. |

Table 49: Role Class Method

#### Permission

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| permissionID | Integer | Private | Unique identifier of permission. |
| name | String | Private | Name of description. |
| description | String | Private | More information of permission. |

Table 50: Permission Class Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Type | Visibility | Description |
| getter | Attribute type | Public | Get attribute value. |
| setter | Void | Public | Set value of attribute. |

Table 51: Permission Class Method

#### RolePermission

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| rolePermissionID | Integer | Private | Unique identifier of role permission. |
| value | Integer | Private | Permission value of role. |

Table 52: RolePermission Class Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Type | Visibility | Description |
| getter | Attribute type | Public | Get attribute value. |
| setter | Void | Public | Set value of attribute. |

Table 53: RolePermission Class Method

#### Website

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| websiteID | Integer | Private | Unique identifier of a website |
| name | String | Private | Name of website. |
| createDate | Date | Private | Time that website created. |
| email | String | Private | Website’s email. |
| phone | String | Private | Website’s phone. |
| address | String | Private | Website’s address. |
| websitePath | String | Private | Website’s path or domain. |

Table 54: Website Class Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Type | Visibility | Description |
| getter | Attribute type | Public | Get attribute value. |
| setter | Void | Public | Set value of attribute. |

Table 55: Website Class Method

#### Page

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| pageID | Integer | Private | Unique identifier of a page in website. |
| title | String | Private | Page’s title. |
| pagePath | String | Private | Path to going to page. |
| createDate | Date | Private | Date created page. |
| folderName | String | Private | Folder name that include page. |
| homePage | Bit | Private | Define page is home page or not. |

Table 56: Page Class Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Type | Visibility | Description |
| getter | Attribute type | Public | Get attribute value. |
| setter | Void | Public | Set value of attribute. |

Table 57: Page Class Method

#### BusinessType

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| businessID | Integer | Private | Unique identifier of a business type. |
| name | String | Private | Name of business type. |
| description | String | Private | More information about business type. |

Table 58: BusinessType Class Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Type | Visibility | Description |
| getter | Attribute type | Public | Get attribute value. |
| setter | Void | Public | Set value of attribute. |

Table 59: BusinessType Class Method

#### Template

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| templateID | Integer | Private | Unique identifier of a template. |
| name | String | Private | Template’s name |
| templatePath | String | Private | Template’s path |
| createDate | Date | Private | Template’s created date |

Table 60: Template Class Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Type | Visibility | Description |
| getter | Attribute type | Public | Get attribute value. |
| setter | Void | Public | Set value of attribute. |

Table 61: Template Class Method

#### BusinessTypeTemplate

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| businessTypeTemplateID | Integer | Private | Unique identifier of a business type template. |

Table 62: BusinessTypeTemplate Class Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Type | Visibility | Description |
| getter | Attribute type | Public | Get attribute value. |
| setter | Void | Public | Set value of attribute. |

Table 63: BusinessTypeTemplate Class Method

#### PageTemplate

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| pageID | Integer | Private | Unique identifier of a page in template. |
| path | String | Private | Page’s path in template |

Table 64: PageTemplate Class Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Type | Visibility | Description |
| getter | Attribute type | Public | Get attribute value. |
| setter | Void | Public | Set value of attribute. |

Table 65: PageTemplate Class Method

#### WidgetType

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| widgetID | integer | Private | Unique identifier of a widget |
| name | String | Private | Widget’s name |
| description | String | Private | More information about widget |
| icon | String | Private | Link to icon image of widget |
| createDate | Date | Private | Widget’s create date |

Table 66: WidgetType Class Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Type | Visibility | Description |
| getter | Attribute type | Public | Get attribute value. |
| setter | Void | Public | Set value of attribute. |

Table 67: WidgetType Class Method

#### Domain

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| domainID | Integer | Private | Unique identifier of a domain |
| value | String | Private | Domain |
| type | String | Private | To know website using subdomain or normal domain |
| status | String | Private | Domain’s status |

Table 68: Domain Class Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Type | Visibility | Description |
| getter | Attribute type | Public | Get attribute value. |
| setter | Void | Public | Set value of attribute. |

Table 69: Domain Class Method

#### Category

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| categoryID | Integer | Private | Unique identifier of a category |
| name | String | Private | Category’s name |
| description | String | Private | More information about category |

Table 70: Category Class Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Type | Visibility | Description |
| getter | Attribute type | Public | Get attribute value. |
| setter | Void | Public | Set value of attribute. |

Table 71: Category Class Method

#### Product

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| productID | Integer | Private | Unique identifier of a product |
| name | String | Private | Product’s name |
| description | String | Private | More information about product |
| price | Float | Private | Product’s price |
| status | String | Private | Product’s status |

Table 72: Product Class Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Type | Visibility | Description |
| getter | Attribute type | Public | Get attribute value. |
| setter | Void | Public | Set value of attribute. |

Table 73: Product Class Method

#### Media

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| mediaID | Integer | Private | Unique identifier of a media |
| name | String | Private | Media’s name |
| link | String | Private | Media’s link |
| createDate | Date | Private | Media’s create date |
| type | String | Private | Determine media is video or image |

Table 74: Media Class Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Type | Visibility | Description |
| getter | Attribute type | Public | Get attribute value. |
| setter | Void | Public | Set value of attribute. |

Table 75: Media Class Method

#### MediaProduct

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| mediaProductID | String | Private | Unique identifier of a product’s media |

Table 76: MediaProduct Class Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Type | Visibility | Description |
| getter | Attribute type | Public | Get attribute value. |
| setter | Void | Public | Set value of attribute. |

Table 77: MediaProduct Class Method

#### Article

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| articleID | Integer | Private | Unique identifier of an article |
| title | String | Private | Article’s title |
| content | String | Private | Article’s main body |
| createDate | Date | Private | Article’s created date |
| updateDate | Date | Private | Last time article updated |
| thumbnail | Integer | Private | Thumbnail of article |

Table 78: Article Class Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Type | Visibility | Description |
| getter | Attribute type | Public | Get attribute value. |
| setter | Void | Public | Set value of attribute. |

Table 79: Article Class Method

#### PageWidget

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| pageWidgetID | Integer | Private | Unique identifier of a widget in page |
| Path | String | Private | Path to the widget |

Table 80: PageWidget Class Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Type | Visibility | Description |
| getter | Attribute type | Public | Get attribute value. |
| setter | Void | Public | Set value of attribute. |

Table 81: PageWidget Class Method

#### WidgetCategory

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| categoryID | integer | Private | Unique identifier of a category |
| name | String | Private | Category’s name |

Table 82: WidgetCategory Class Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Type | Visibility | Description |
| getter | Attribute type | Public | Get attribute value. |
| setter | Void | Public | Set value of attribute. |

Table 83: WidgetCategory Class Method

#### GeneralReport

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| reportID | Integer | Private | Unique identifier of a report |
| name | String | Private | Report’s name |
| createDate | Date | Private | The date that report’s created |
| totalAccess | Integer | Private | Access number |

Table 84: GeneralReport Class Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Type | Visibility | Description |
| getter | Attribute type | Public | Get attribute value. |
| setter | Void | Public | Set value of attribute. |

Table 85: GeneralReport Class Method

#### ProductReport

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| productReportID | Integer | Private | Unique identifier of a product report |
| totalView | Integer | Private | Access number |
| createDate | Date | Private | The date that report’s created |

Table 86: ProductReport Class Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Type | Visibility | Description |
| getter | Attribute type | Public | Get attribute value. |
| setter | Void | Public | Set value of attribute. |

Table 87: ProductReport Class Method

#### Payment

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| paymentID | Integer | Private | Unique identifier of an payment |
| amount | Double | Private | Total amount of payment |
| createDate | Date | Private | The date that report’s created |
| transactionID | String | Private | The PayPal transaction id |

Table 88: Payment Class Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Type | Visibility | Description |
| getter | Attribute type | Public | Get attribute value. |
| setter | Void | Public | Set value of attribute. |

Table 89: Payment Class Method

#### MenuItem

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | Type | Visibility | Description |
| menuID | Integer | Private | Unique identifier of an menu item |
| name | Double | Private | Name of menu item |
| link | Date | Private | Link of menu item |
| parentID | String | Private | Menu item parent that menu item belongs to |
| position | Integer | Private | Position of menu item in sitemap |

Table 90: MenuItem Class Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Type | Visibility | Description |
| getter | Attribute type | Public | Get attribute value. |
| setter | Void | Public | Set value of attribute. |

Table 91: MenuItem Class Method

### Interaction Diagram

#### Admin

##### <Admin> Create Template

Summary: This diagram shows how admin upload a template for system.

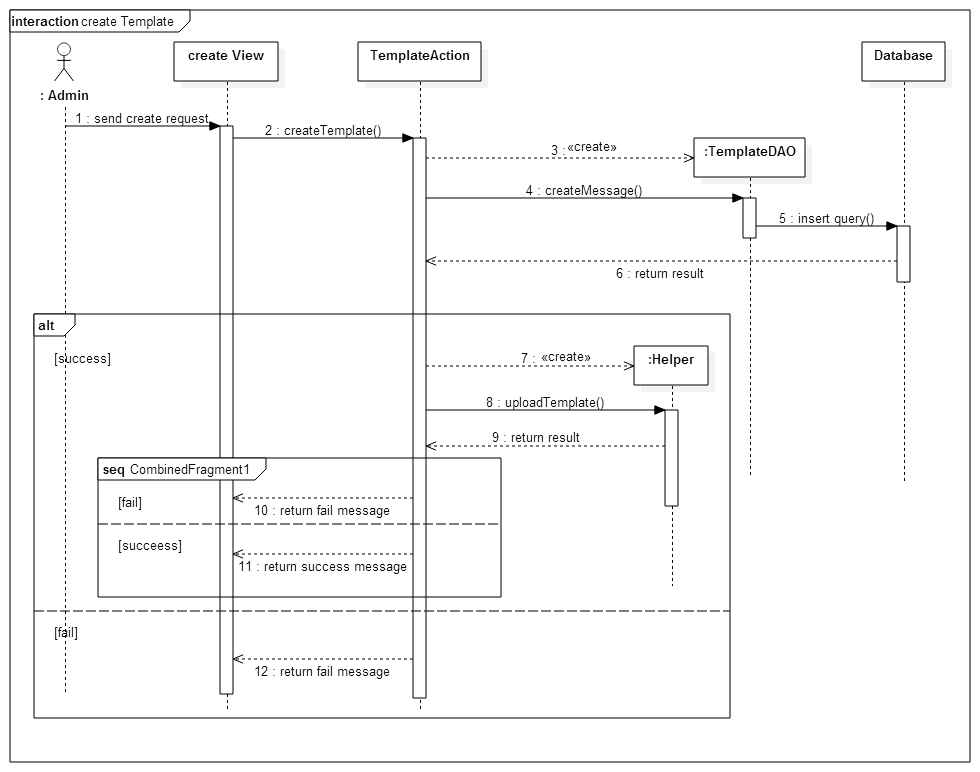


Figure 47: <Admin> Create Template sequence diagram

##### <Admin> Create Widget

Summary: This diagram shows how admin upload a widget for system.

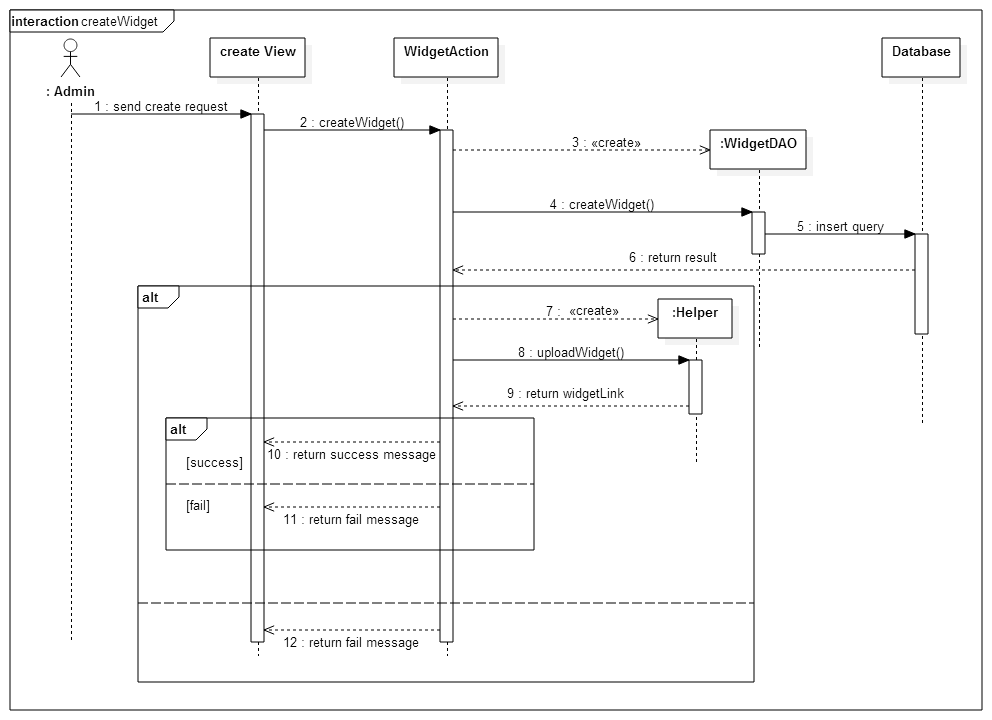


Figure 48: <Admin> Create Widget sequence diagram

##### <Admin> Delete Template

Summary: This diagram shows how admin update a widget that existed in system.

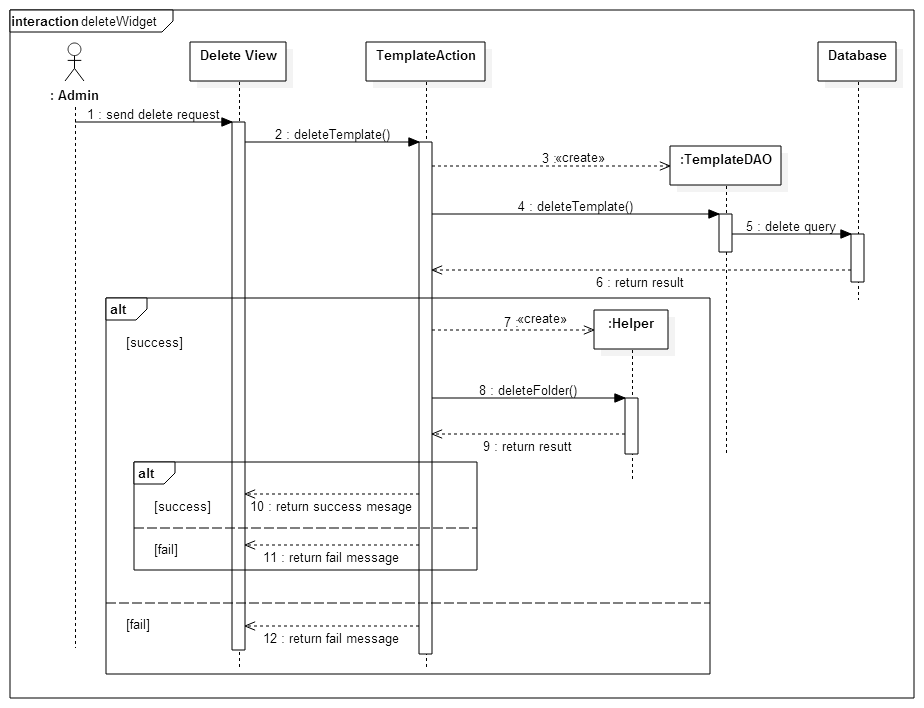


Figure 49: <Admin> Delete Template sequence diagram

##### <Admin> Update Role and Permission

Summary: This diagram shows how admin update money needed to upgrade account and permission of account.

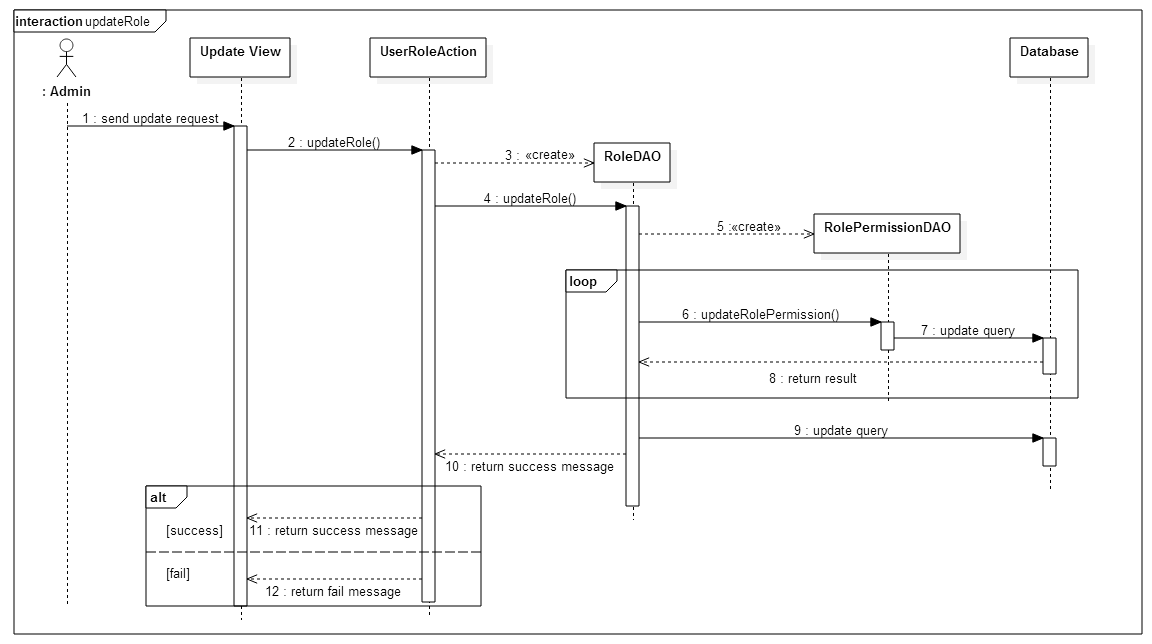


Figure 50: <Admin> Update Role and Permission sequence diagram

##### <Admin> View Statistic

Summary: This diagram shows how admin view information statistic of system.

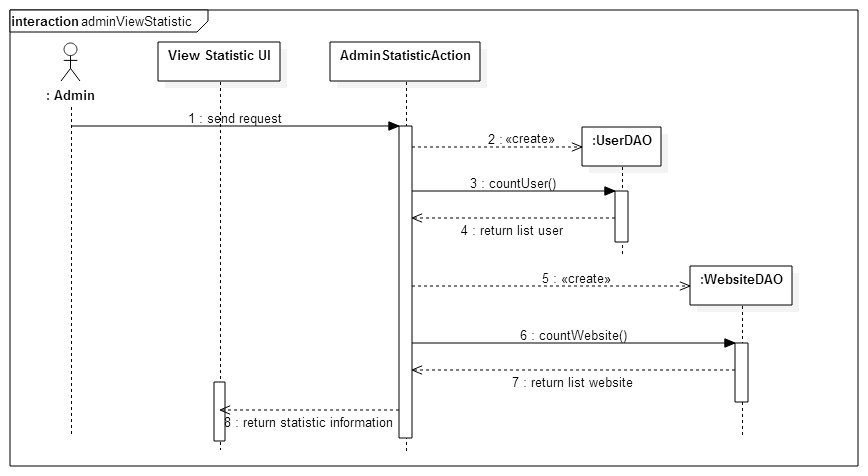


Figure 51: <Admin> View Statistic sequence diagram

##### <Admin > Delete Widget

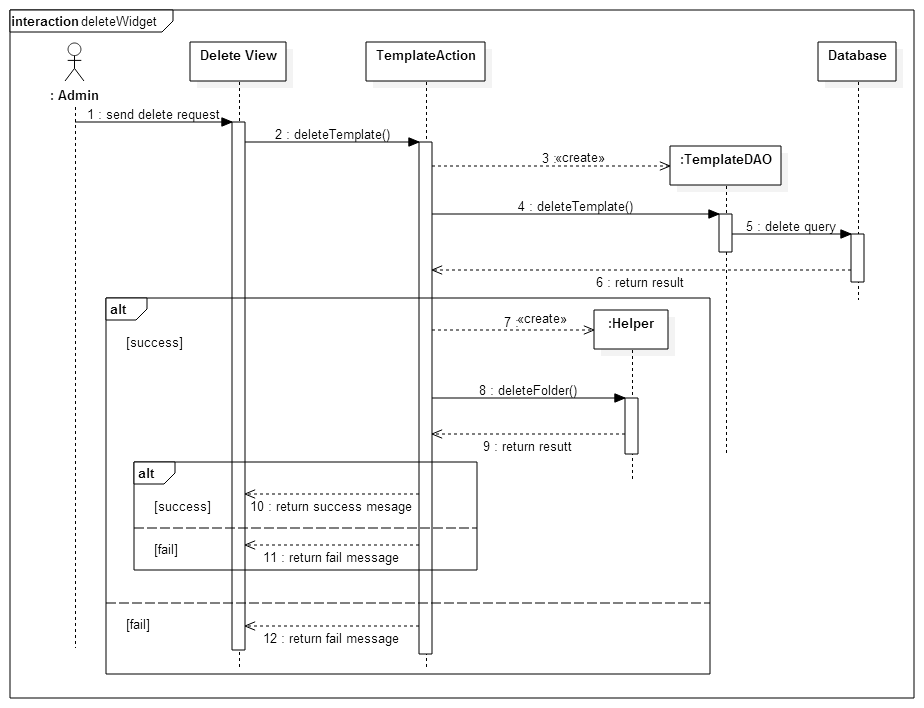


Figure 52: <Admin> Delete Widget sequence diagram

#### User

##### <User> Create Website

Summary: This diagram shows how users add a new website.

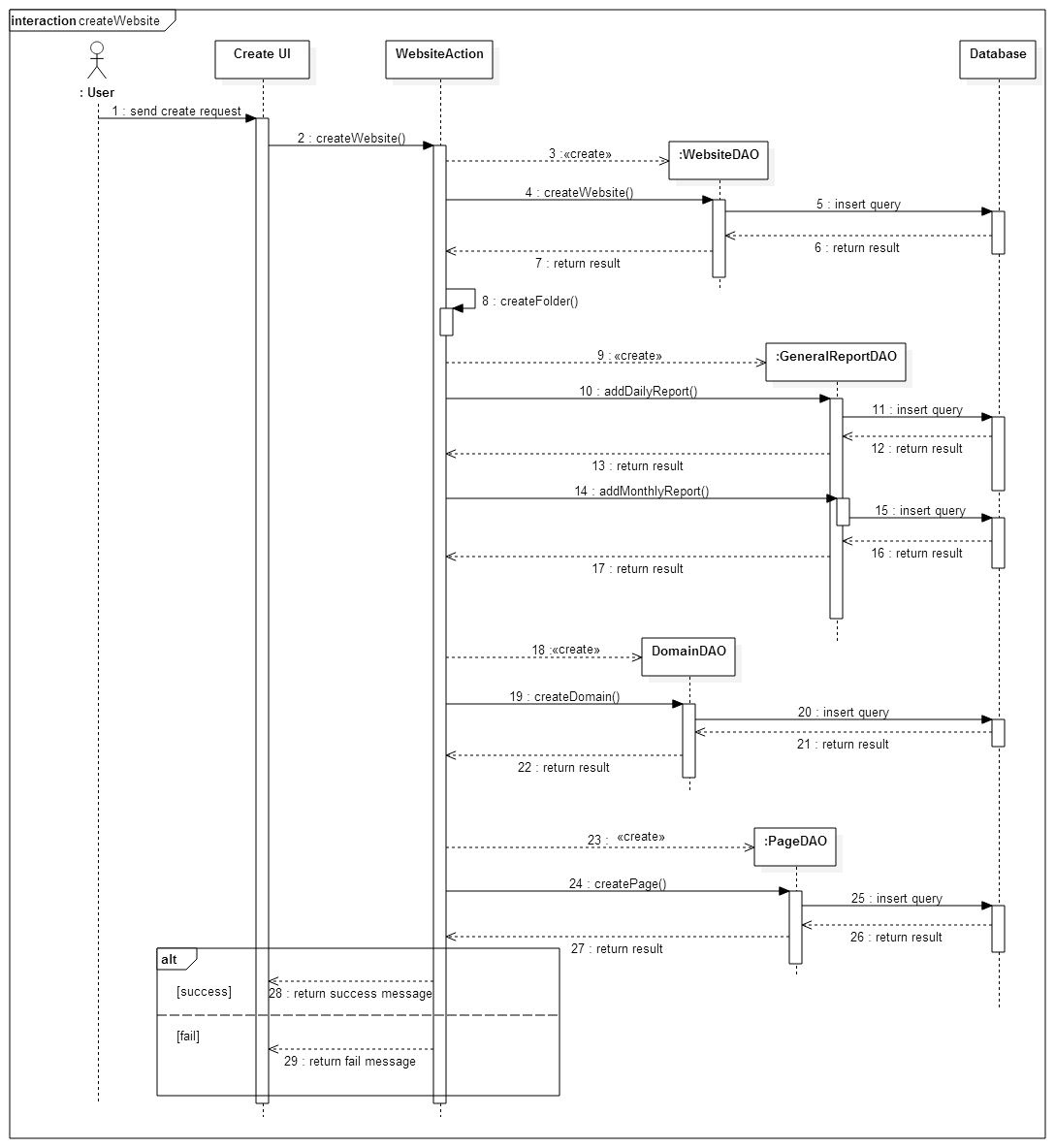


Figure 53: <User> Create Website sequence diagram

##### <User> Delete Website

Summary: This diagram shows how users delete a website.



Figure 54: <User> Delete Website sequence diagram

##### <User> Create Product

Summary: This diagram shows how users add a new product to website.

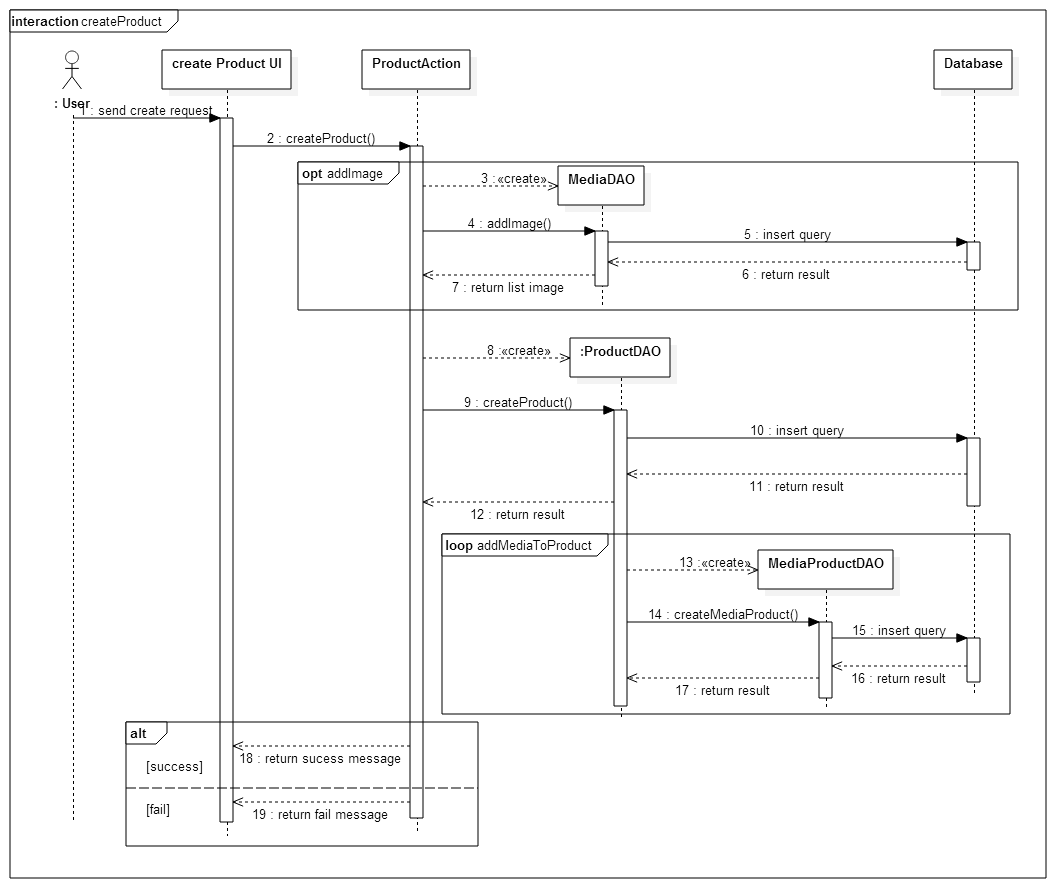


Figure 55: <User> Create Product sequence diagram

##### <User> Create Media

Summary: This diagram shows how users create media to website.

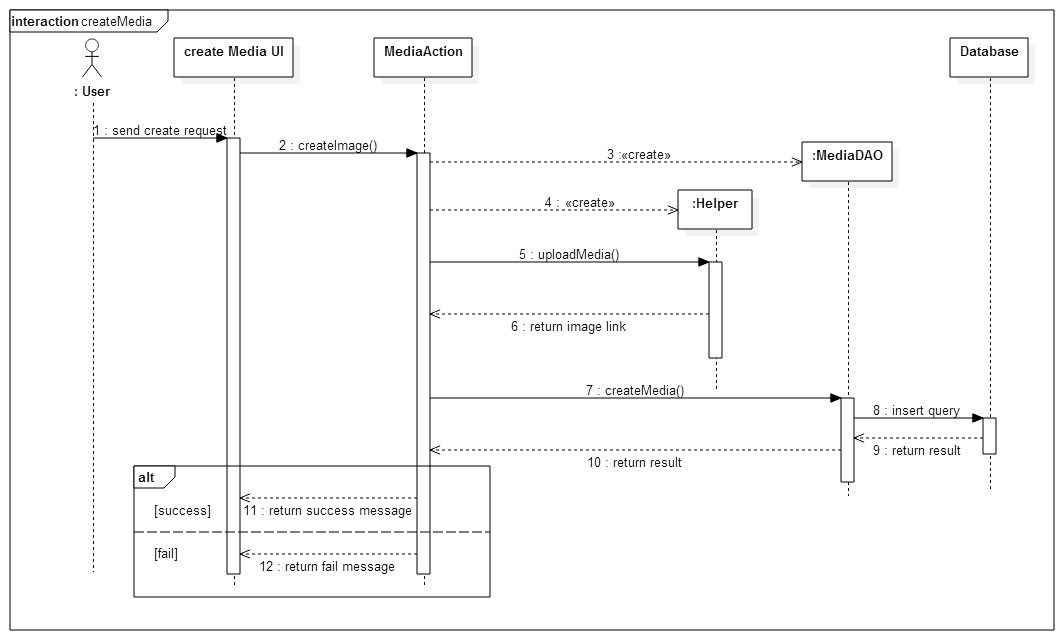


Figure 56: <User> Create Media sequence diagram

##### <User> Create Article

Summary: This diagram shows how users create a new article to website.

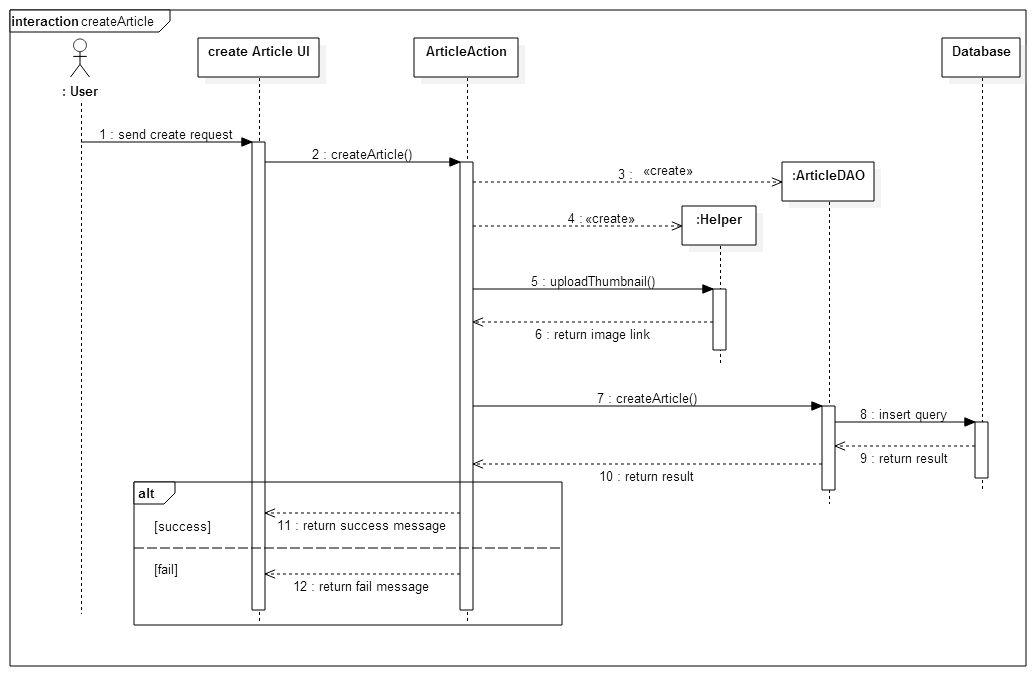


Figure 57: <User> Create Article sequence diagram

##### <User> Save Customization

Summary: This diagram shows how to save changes after user customization website’s interface. In custom actions users can add new pages, delete pages or customize a page already exists

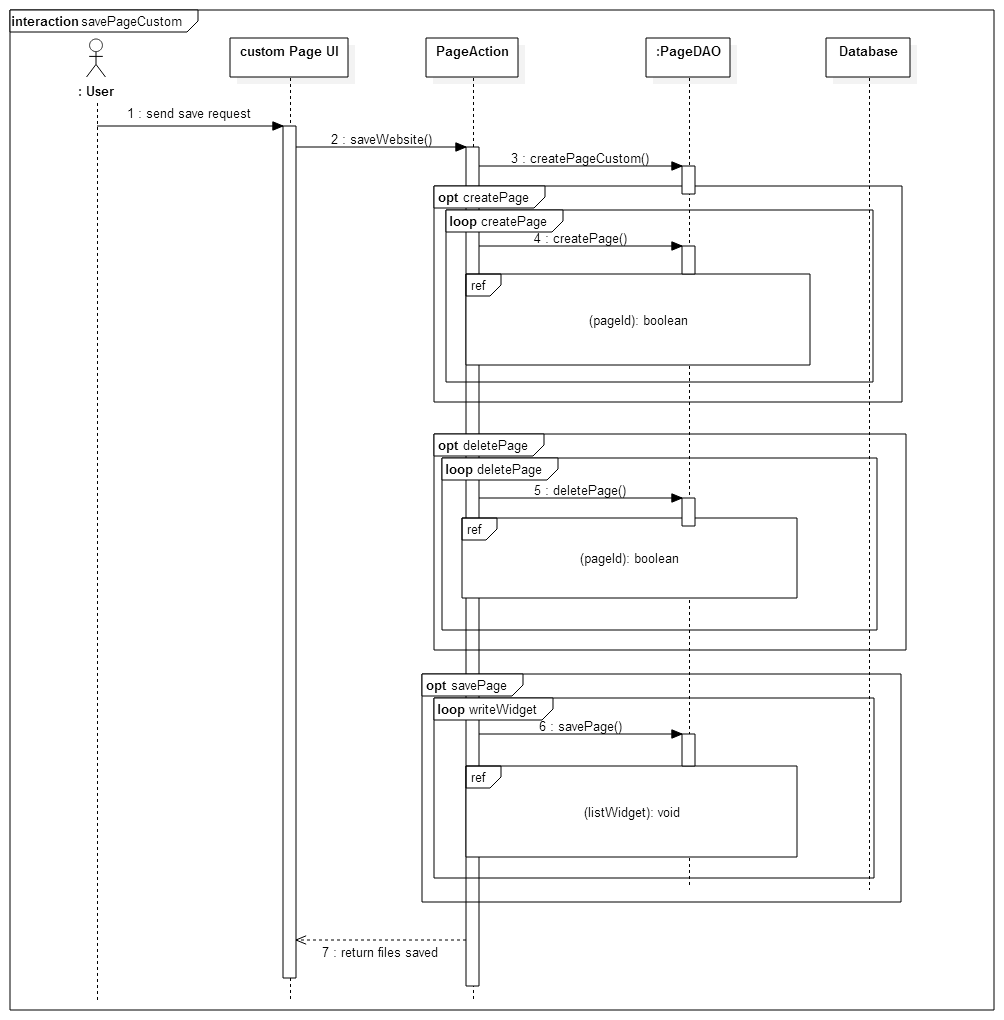


Figure 58: <User> Save Page Custom sequence diagram

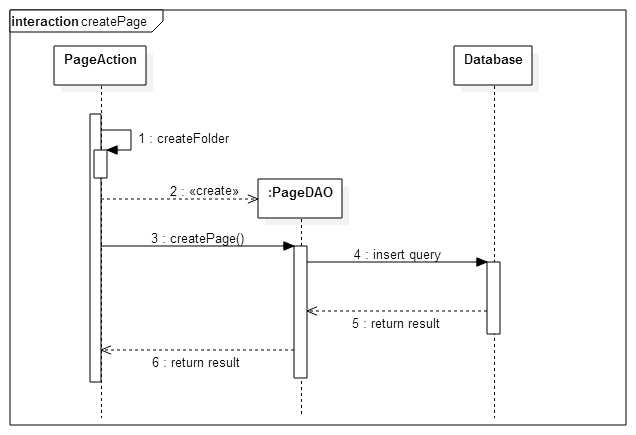


Figure 59: <User> Create Page sequence diagram

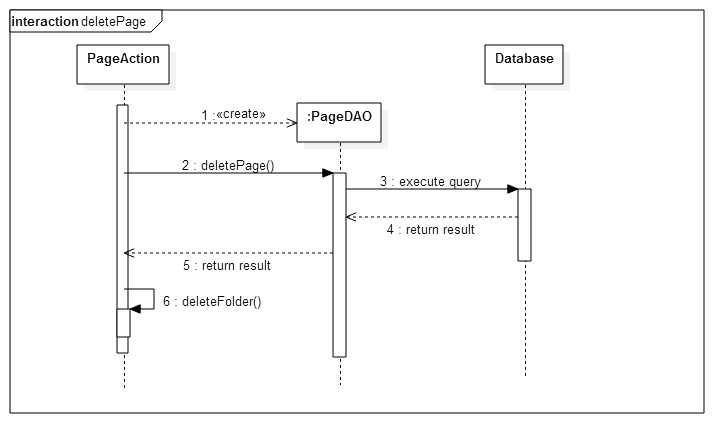


Figure 60: <User> Delete Page sequence diagram

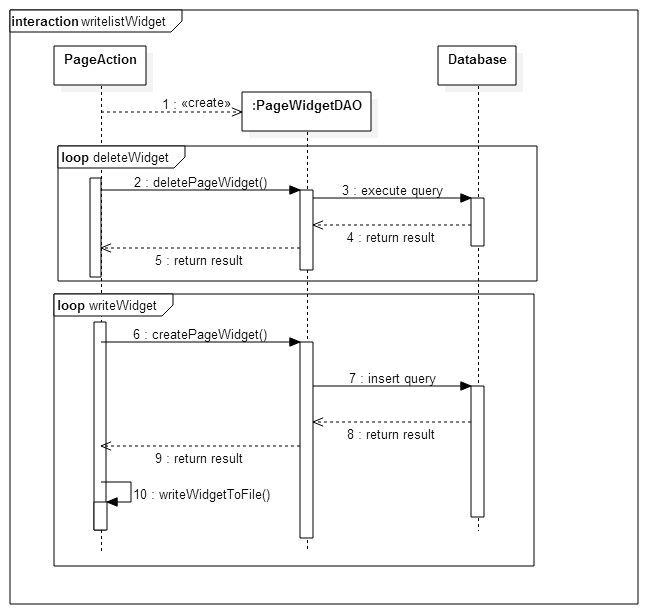


Figure 61: <User> Write Widget sequence diagram

##### <User> View Statistic

Summary: This diagram shows how users view statistic of website.

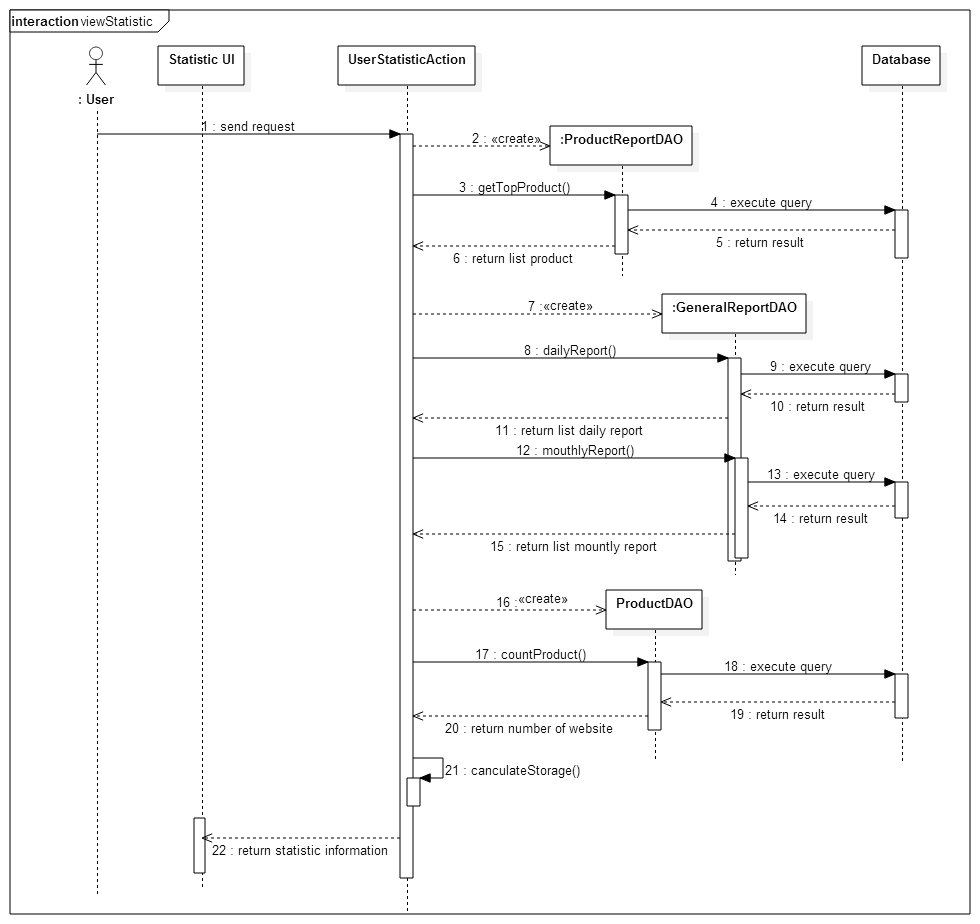


Figure 62: <User> View Statistic sequence diagram

## User Interface Design

### Guest Interface Design

#### Register

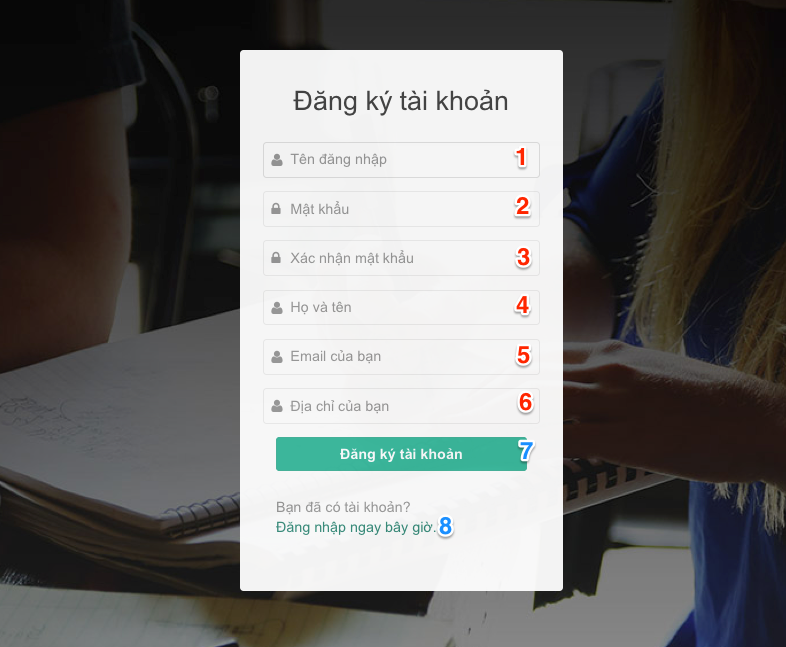


Figure 64: Register interface design

**Fields**:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Field name | Description | Read only | Mandatory | Control Type | Data Type |
| 1 | Username | Fill username | No | Yes | Textbox | String |
| 2 | Password | Fill password | No | Yes | Password | String |
| 3 | Confirm password | Fill confirm password | No | Yes | Password | String |
| 4 | Full name | Fill full name | No | Yes | Textbox | String |
| 5 | Email | Fill email | No | Yes | Textbox | String |
| 6 | Address | Fill address | No | Yes | Textbox | String |

Table 94: Register - Fields

**Buttons/Links**:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validate | Outcome |
| 7 | Register | Click to register a new account | Yes | Transfer to login page |
| 8 | Login | Click to login into the system | Yes | Transfer to login page |

Table 95: Register - Buttons/Links

### User Interface Design

#### User Manage Website

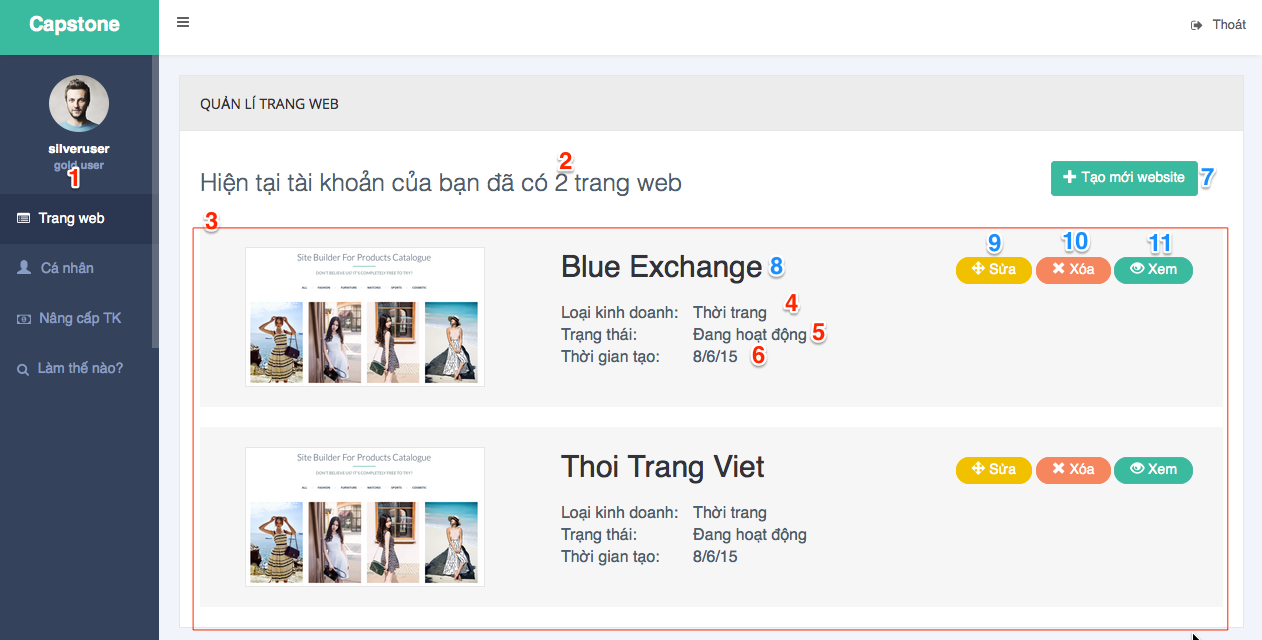


Figure 65: User Manage Website interface design

**Fields**:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Field name | Description | Read only | Mandatory | Control Type | Data Type |
| 1 | Menu | Navigation bar | Yes | Yes | Menu bar | N/A |
| 2 | Number Website | Number of available website | Yes | No | Label | int |
| 3 | lstWebsite | List user’s website | Yes | Yes | List | N/A |
| 4 | Business Type | Business type of website | Yes | Yes | Label | String |
| 5 | Status | Status of website | Yes | Yes | Label | String |
| 6 | Create Date | Create date of website | Yes | Yes | Label | String |

Table 96: User Manage Website - Fields

**Buttons/Links**:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validate | Outcome |
| 7 | btnCreateWeb | Create a new website | N/A | Transfer to choose template page |
| 8 | linkWebsiteDetail | Link to manage website | N/A | Transfer to manage website |
| 9 | btnEdit | Edit interface’s website | N/A | Transfer to custom page |
| 10 | btnDelete | Delete website | N/A | Show confirm pop up |
| 11 | btnView | View Website | N/A | Transfer to home page of website |

Table 97: User Manage Website - Buttons/Links

#### User Manage Product

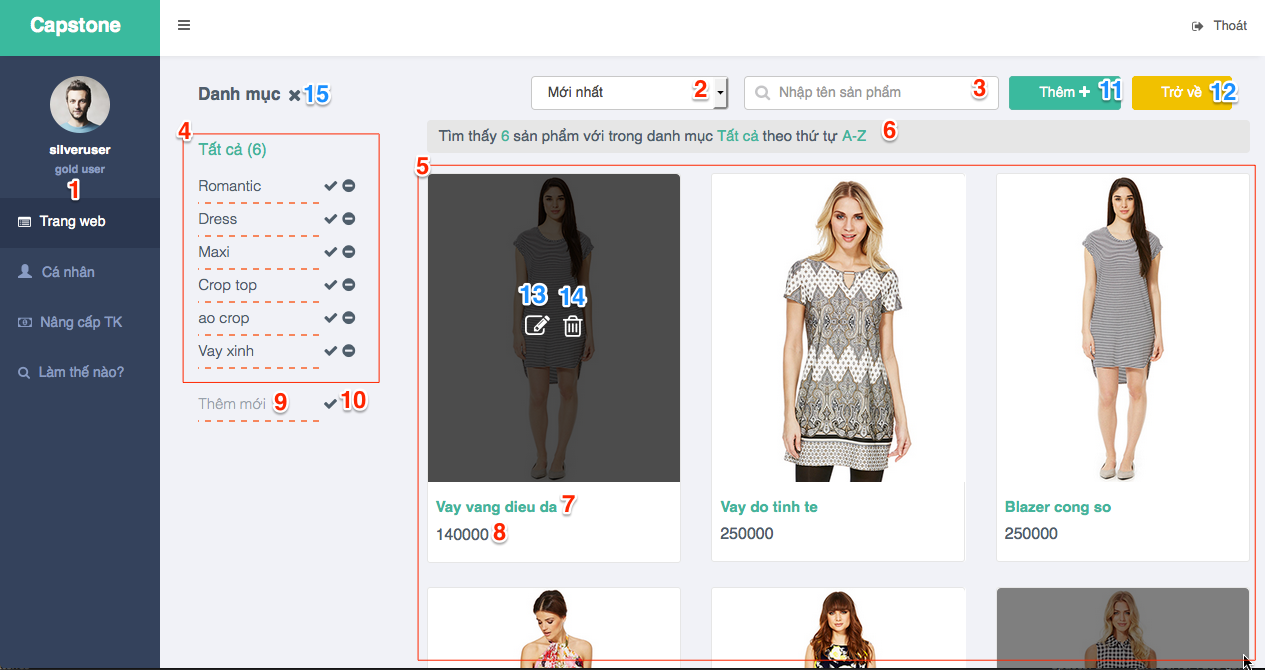


Figure 66: User Manage Product interface design

**Fields**:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Field name | Description | Read only | Mandatory | Control Type | Data Type |
| 1 | Menu | Navigation bar | Yes | Yes | Menu bar | N/A |
| 2 | cbSort | Sort product by condition | No | No | Select | String |
| 3 | txtSearch | Search product by name | No | No | Text box | String |
| 4 | Category List | Show category list | No | N/A | List | N/A |
| 5 | Product List | Show product list | No | N/A | List | N/A |
| 6 | Number result | Show number of result | No | N/A | Label | N/A |
| 7 | Product name | Show name of each product | No | N/A | Label | String |
| 8 | Product price | Show price of each product | No | N/A | Label | String |
| 9 | New category | Fill name of new category | No | N/A | Textbox | String |

Table 98: User Manage Product - Fields

**Buttons/Links**:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validate | Outcome |
| 10 | Add new Category | Click to create new category | No | Message completed |
| 11 | Add new product | Click to show popup add new product | No | Show popup to add new product |
| 12 | Back | Back to Website detail page | No | Transfer to website detail page |
| 13 | Edit product | Click to show pop up edit product | No | Show popup to edit product |
| 14 | Delete product | Click to delete product | No | Show popup confirm and delete product if choose yes |
| 15 | Edit category list | Click to add/update category list | No | Show update/delete button for each category and textbox to add new category |

Table 99: User Manage Product - Buttons/Links

#### User Create Product

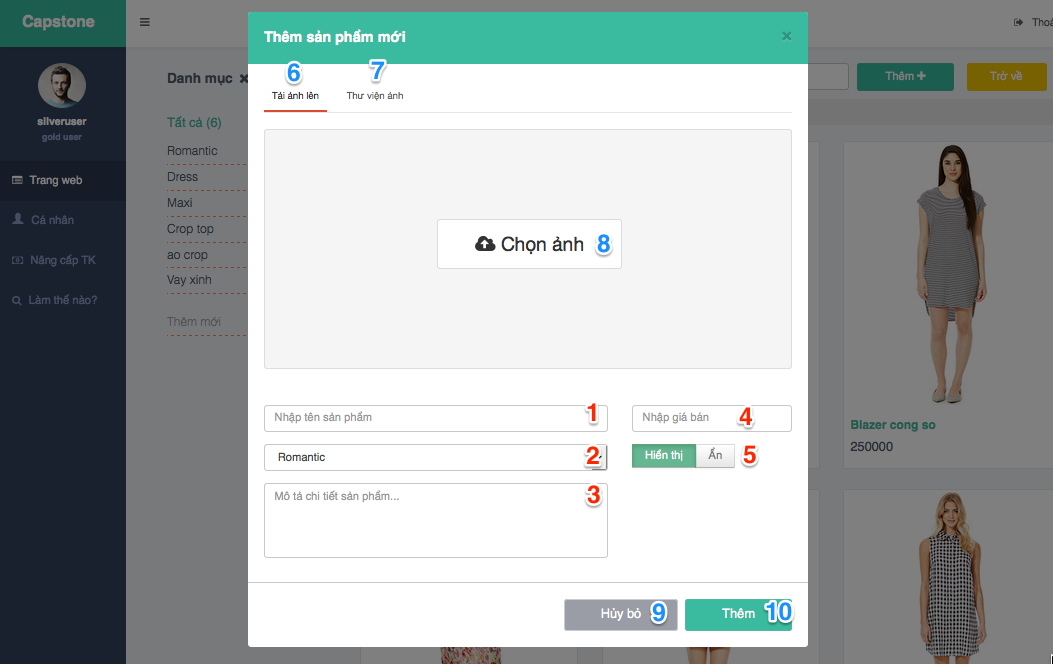


Figure 67: User Create Product interface design

**Fields**:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Field name | Description | Read only | Mandatory | Control Type | Data Type |
| 1 | Product name | Fill product name | No | Yes | Textbox | String |
| 2 | Product category | Choose product category | No | Yes | Select | String |
| 3 | Product description | Fill product description | No | Yes | Textarea | String |
| 4 | Product price | Fill product price | No | Yes | Textbox | Float |
| 5 | Product status | Choose to change status: visible/invisible | No | Yes | Radio button | Bit |

Table 100: User Create Product - Fields

**Buttons/Links**:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validate | Outcome |
| 6 | Upload tab | Move to upload tab to upload new images | No | Move to upload image tab |
| 7 | Media tab | Move to media tab to choose images from media | No | Move to media tab |
| 8 | Choose image | Click to choose image from computer | No | Show browse file window from computer |
| 9 | Cancel | Click to cancel | No | Close add new product popup |
| 10 | Add product | Click to add product into DB | No | Message completed |

Table 101: User Create Product - Buttons/Links

#### User Manage Media

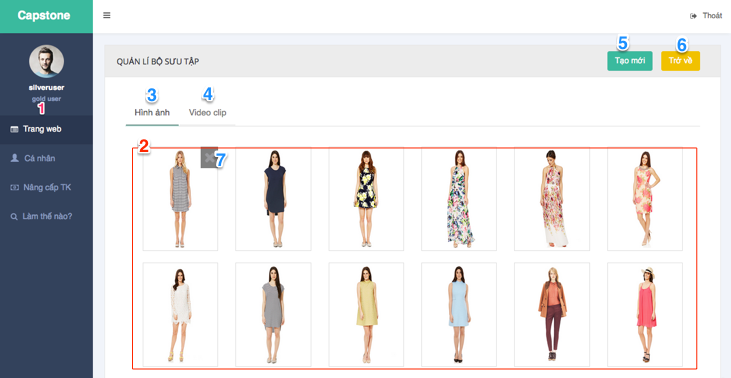


Figure 68: User Manage Media interface design

**Fields**:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Field name | Description | Read only | Mandatory | Control Type | Data Type |
| 1 | Menu | Navigation bar | Yes | Yes | Menu bar | N/A |
| 2 | Media List | Show list media (images and videos) | Yes | No | List | N/A |

Table 102: User Manage Media - Fields

**Buttons/Links**:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validate | Outcome |
| 3 | Images tab | Click to move to images tab | No | Show images list in media |
| 4 | Videos tab | Click to move to videos tab | No | Show videos list in media |
| 5 | Add new | Click to add new image / video | No | Show popup add new image / video |
| 6 | Back | Click to back to website detail page | No | Transfer to website detail page |
| 7 | Delete media | Click to delete an image or video | No | Message completed |

Table 103: User Manage Media - Buttons/Links

#### User Manage Article

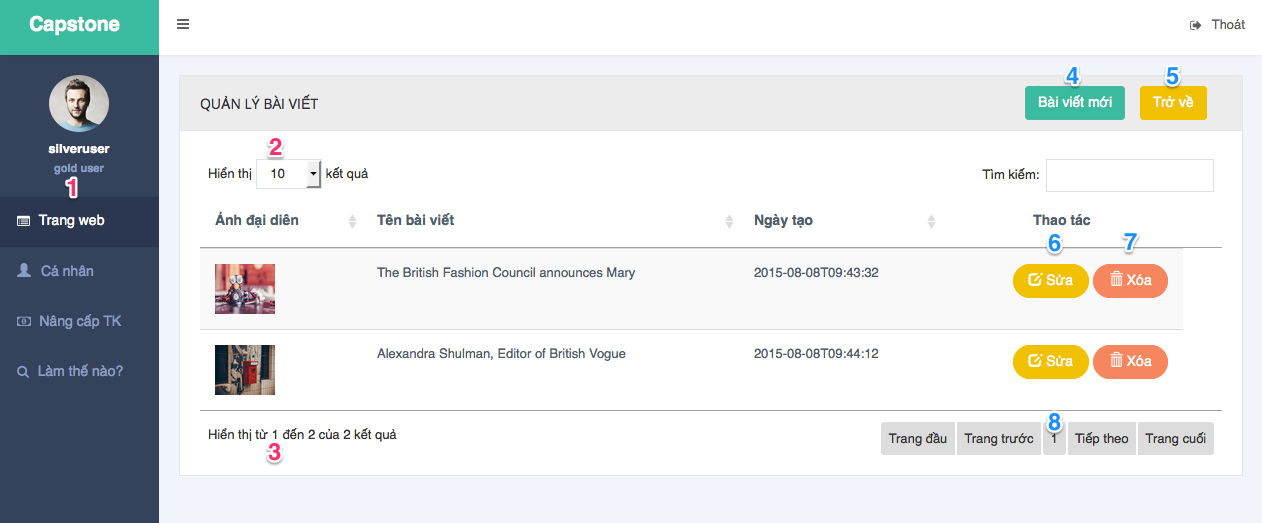


Figure 69: User Manage Article interface design

**Fields**:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Field name | Description | Read only | Mandatory | Control Type | Data Type |
| 1 | Menu | Navigation bar | Yes | Yes | Menu bar | N/A |
| 2 | Number result display | Change number of result display | No | Yes | Select | N/A |
| 3 | Number result | Show number of result | Yes | Yes | N/A | N/A |

Table 104: User Manage Article - Fields

**Buttons/Links**:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validate | Outcome |
| 4 | Add new | Click to show popup add new article | No | Show popup add new article |
| 5 | Back | Click to back to website detail page | No | Transfer to website detail page |
| 6 | Edit article | Click to show popup edit article | No | Show popup edit article |
| 7 | Delete article | Delete an article | No | Show popup confirm and message complete if choose yes |
| 8 | Pagination | Click to change page | No | Change page of list article |

Table 105: User Manage Article - Buttons/Links

#### User Create Article

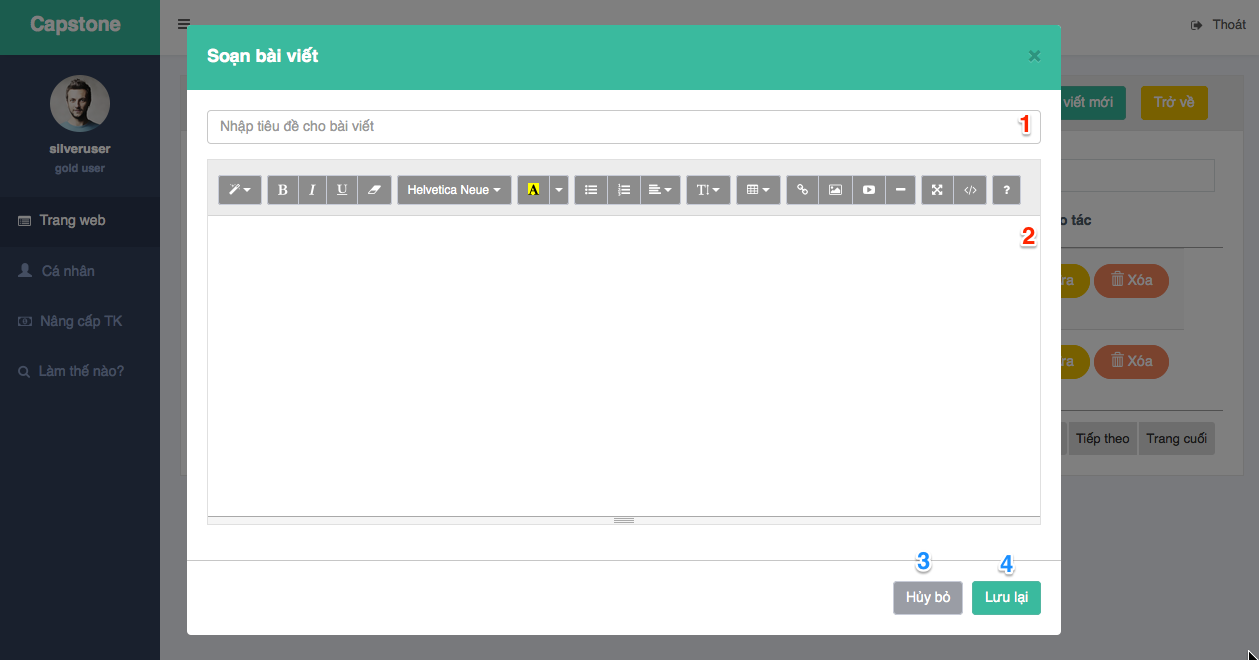


Figure 70: User Create Article interface design

**Fields**:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Field name | Description | Read only | Mandatory | Control Type | Data Type |
| 1 | Article title | Fill article title | No | Yes | Textbox | String |
| 2 | Article content | Fill article content | No | Yes | Textarea | String |

Table 106: User Create Article - Fields

**Buttons/Links**:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validate | Outcome |
| 3 | Cancel | Click to cancel | No | Close popup add new article |
| 4 | Save | Click to insert new article to DB | Yes | Message completed |

Table 107: User Create Article - Buttons/Links

#### User View Statistic

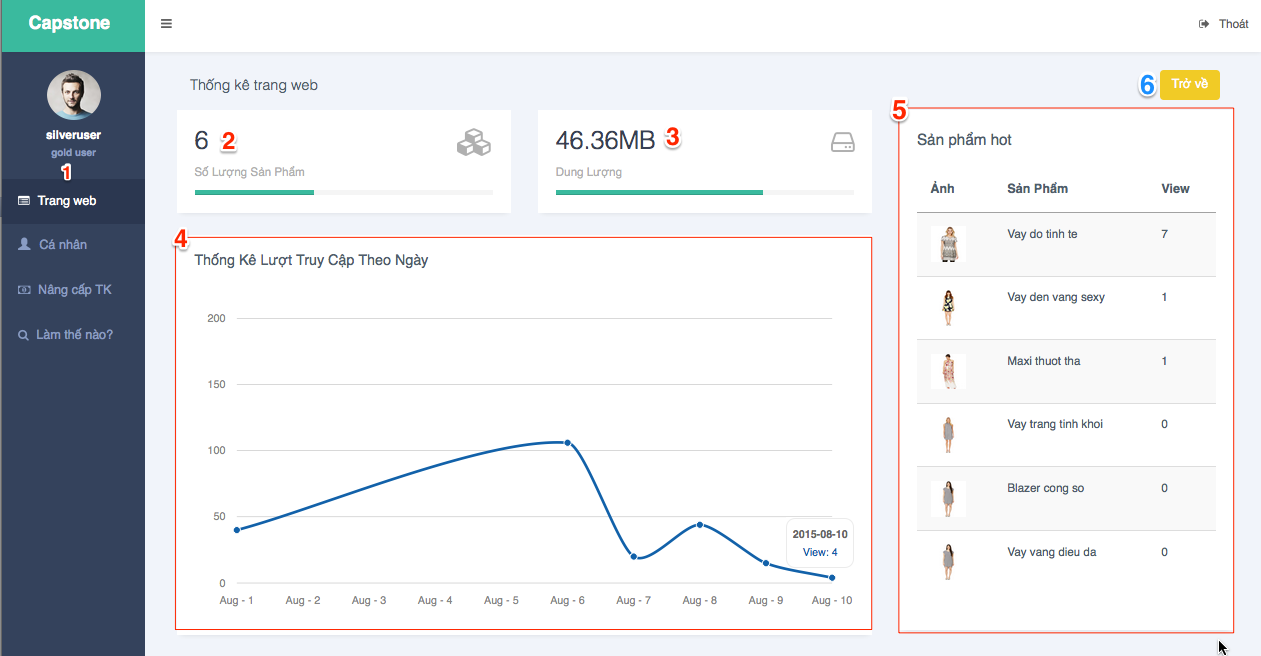




Figure 71: User View Statistic interface design

**Fields**:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Field name | Description | Read only | Mandatory | Control Type | Data Type |
| 1 | Menu | Navigation bar | Yes | Yes | Menu bar | N/A |
| 2 | Number of product | Show number of product on website | Yes | No | Label | N/A |
| 3 | Storage | Show total current storage on website | Yes | No | Label | N/A |
| 4 | Views line chart in month | Line chart compares number of views everyday | Yes | No | Line chart | N/A |
| 5 | Top product list | Show list most popular products | Yes | No | Table List | N/A |
| 7 | Views column chart in year | Column chart compares number of views every month | Yes | No | Column chart | N/A |

Table 108: User View Statistic - Fields

**Buttons/Links**:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validate | Outcome |
| 6 | Back | Click to back to website detail page | No | Transfer to website detail page |

Table 109: User View Statistic - Buttons/Links

### Admin Interface Design

#### Admin Manage Template

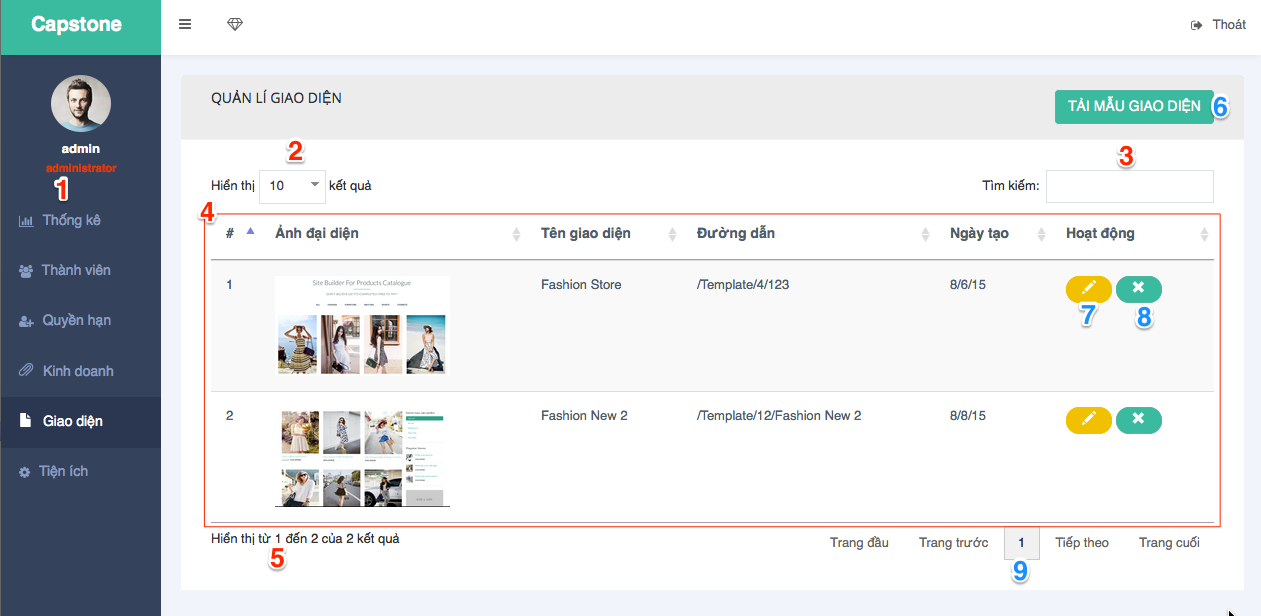


Figure 72: Admin Manage Template interface design

**Fields**:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Field name | Description | Read only | Mandatory | Control Type | Data Type |
| 1 | Menu | Navigation bar | Yes | Yes | Menu bar | N/A |
| 2 | Number result display | Change number of result display | No | Yes | Select | N/A |
| 3 | txtSearch | Search template | No | No | Textbox | N/A |
| 4 | Template List | Show template list | No | No | Table | N/A |
| 5 | Number result | Show number of result | Yes | Yes | N/A | N/A |

Table 110: Admin Manage Template - Fields

**Buttons/Links**:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validate | Outcome |
| 6 | Upload new template | Click to show popup upload new template | No | Show popup upload new template |
| 7 | Edit template | Click to edit template | No | Transfer to edit template page |
| 8 | Delete template | Click to delete template | No | Message completed |
| 9 | Pagination | Click to change page | No | Change page of templates list |

Table 111: Admin Manage Template - Buttons/Links

#### Admin Create Template

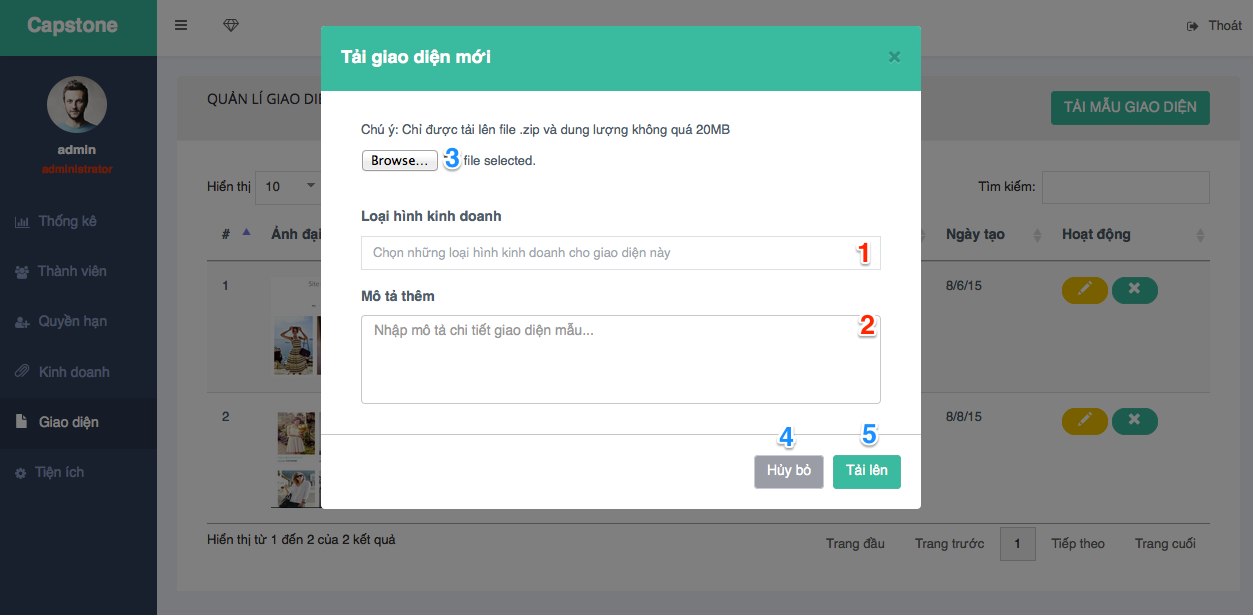


Figure 73: Admin Create Template interface design

**Fields**:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Field name | Description | Read only | Mandatory | Control Type | Data Type |
| 1 | Business type | Choose business type of template | No | Yes | Select2 | Array |
| 2 | Template description | Fill template description | No | No | Textarea | String |

Table 112: Admin Create Template - Fields

**Buttons/Links**:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validate | Outcome |
| 3 | Browse file | Click to brown template file zip from computer | No | Show browse file window from computer |
| 4 | Cancel | Click to cancel | No | Close popup upload new template |
| 5 | Upload template | Click to upload new template to system | Yes | Message completed |

Table 113: Admin Create Template - Buttons/Links

#### Admin Manage Widget

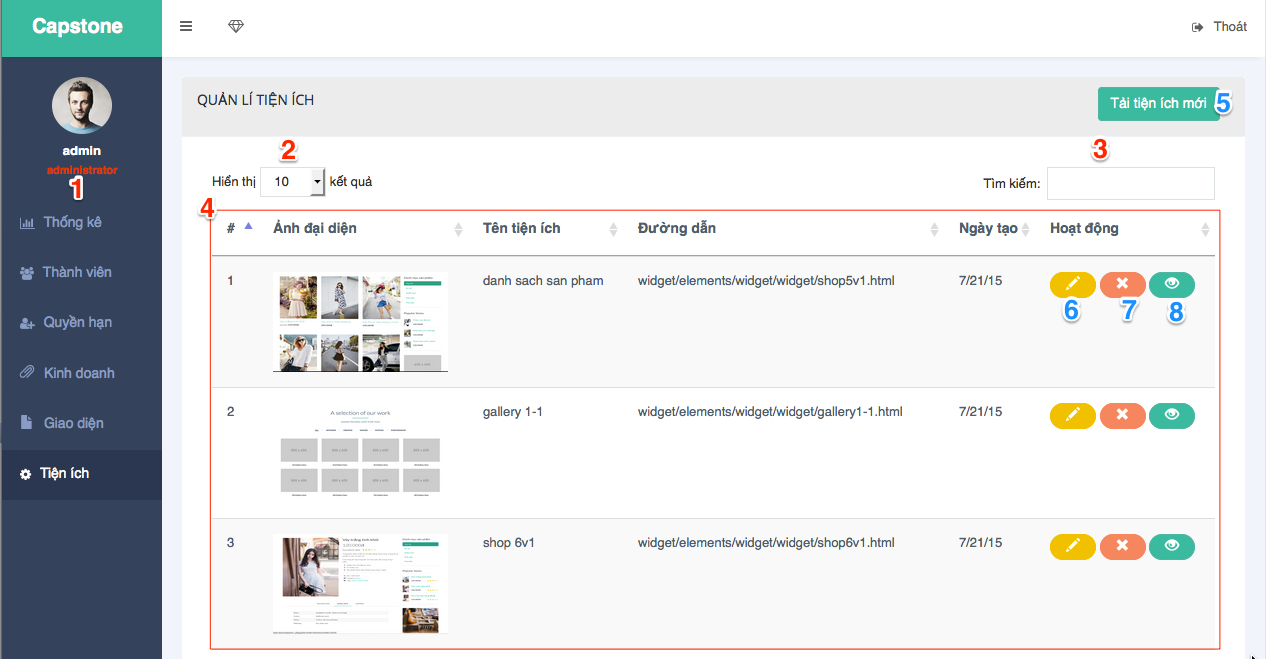


Figure 74: Admin Manage Widget interface design

**Fields**:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Field name | Description | Read only | Mandatory | Control Type | Data Type |
| 1 | Menu | Navigation bar | Yes | Yes | Menu bar | N/A |
| 2 | Number result display | Change number of result display | No | Yes | Select | N/A |
| 3 | txtSearch | Search widget | No | No | Textbox | N/A |
| 4 | Widget List | Show widgets list | No | No | Table | N/A |

Table 114: Admin Manage Widget - Fields

**Buttons/Links**:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validate | Outcome |
| 5 | Upload new widget | Click to show popup upload new widget | No | Show popup upload new widget |
| 6 | Edit widget | Click to edit widget | No | Transfer to edit widget page |
| 7 | Delete widget | Click to delete widget | No | Message completed |
| 8 | View widget | Click to view widget html | No | Show file html of widget on browser |

Table 115: Admin Manage Widget - Buttons/Links

#### Admin Create Widget

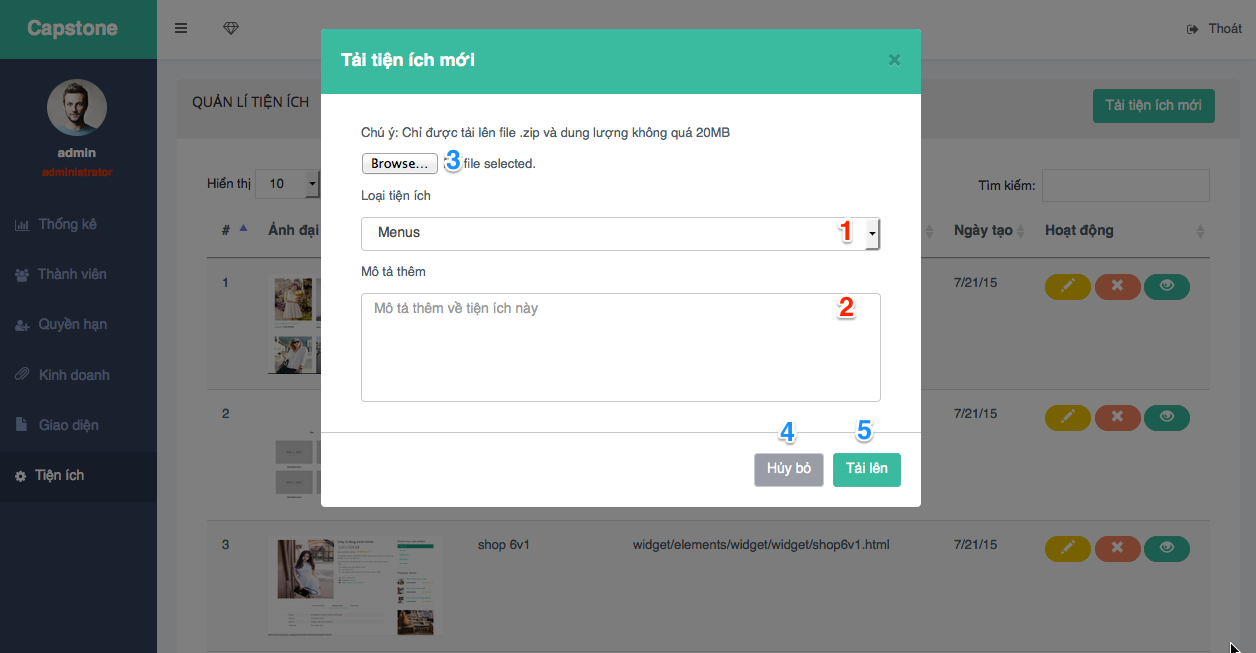


Figure 75: Admin Create Widget interface design

**Fields**:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Field name | Description | Read only | Mandatory | Control Type | Data Type |
| 1 | Widget category | Choose widget category | No | Yes | Select | String |
| 2 | Widget description | Fill widget description | No | No | Textarea | String |

Table 116: Admin Create Widget - Fields

**Buttons/Links**:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validate | Outcome |
| 3 | Browse file | Click to browse widget file zip from computer | Yes | Show browse file window from computer |
| 4 | Cancel | Click to cancel | No | Close popup upload new widget |
| 5 | Upload widget | Click to upload widget | Yes | Message completed |

Table 117: Admin Create Widget - Buttons/Links

#### Admin View Statistic

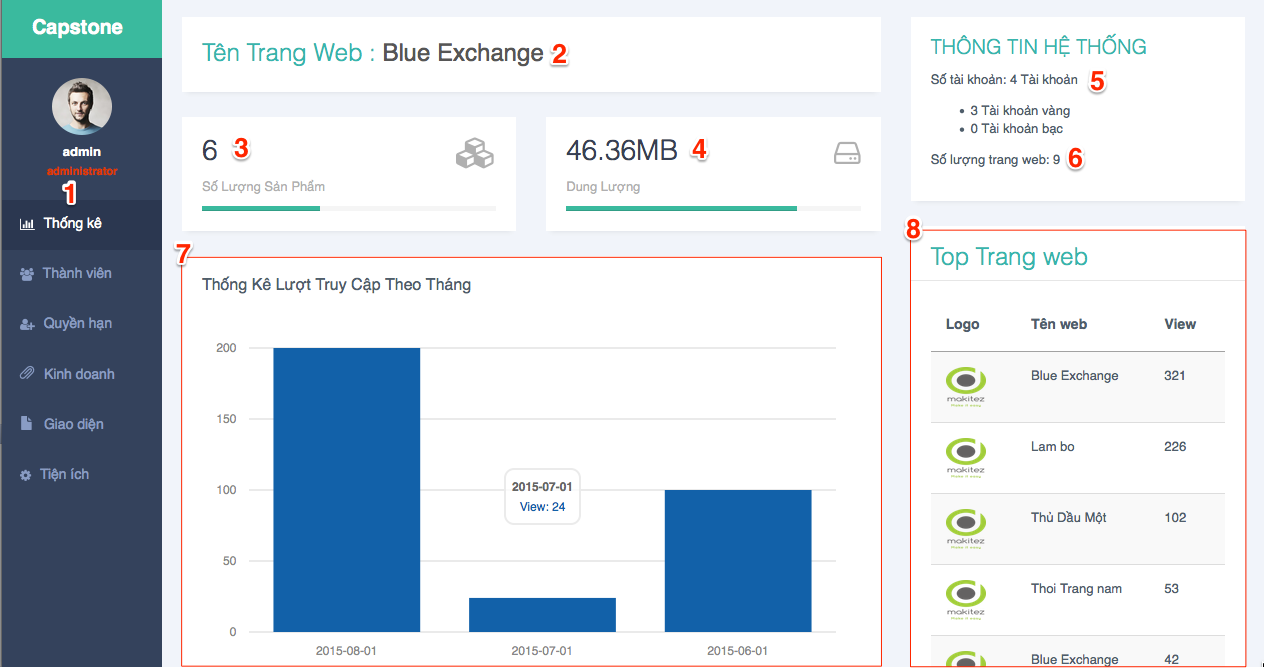


Figure 76: Admin View Statistic interface design

**Fields**:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Field name | Description | Read only | Mandatory | Control Type | Data Type |
| 1 | Menu | Navigation bar | Yes | Yes | Menu bar | N/A |
| 2 | Website name | Show website name | Yes | No | Label | String |
| 3 | Number of product | Show number of product on website | Yes | No | Label | N/A |
| 4 | Storage | Show total current storage on website | Yes | No | Label | N/A |
| 5 | Number of user | Show number of user in system | Yes | No | Label | N/A |
| 6 | Number of website | Show number of website in system | Yes | No | Label | N/A |
| 7 | Views column chart in year | Column chart compares number of views every month | Yes | No | Column chart | N/A |
| 8 | Top website | Show list website with most view | Yes | No | Table List | N/A |

Table 118: Admin View Statistic - Fields

**Buttons/Links**: N/A

#### Admin Manage Permission



Figure 77: Admin Manage Permission interface design

**Fields**:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Field name | Description | Read only | Mandatory | Control Type | Data Type |
| 1 | Menu | Navigation bar | Yes | Yes | Menu bar | N/A |
| 2 | Account type list | Show list of account type | Yes | Yes | List | N/A |
| 3 | Account type name | Show name of account type | Yes | Yes | Label | N/A |
| 4 | Account type price | Show price of account type | Yes | Yes | Label | N/A |
| 5 | Number of website | Show limit number of website | Yes | No | Label | int |
| 6 | Number of storage | Show limit number of storage | Yes | No | Label | String |
| 7 | View statistic | Show each account type can view statistic or not | Yes | No | Label | bit |

Table 120: Admin Manage Permission - Fields

**Buttons/Links**:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Function | Description | Validate | Outcome |
| 8 | Update account type | Click to edit account type info and permission | No | Show popup edit account type info |

Table 121: Admin Manage Permission - Buttons/Links

## Database Design

### Logical Diagram

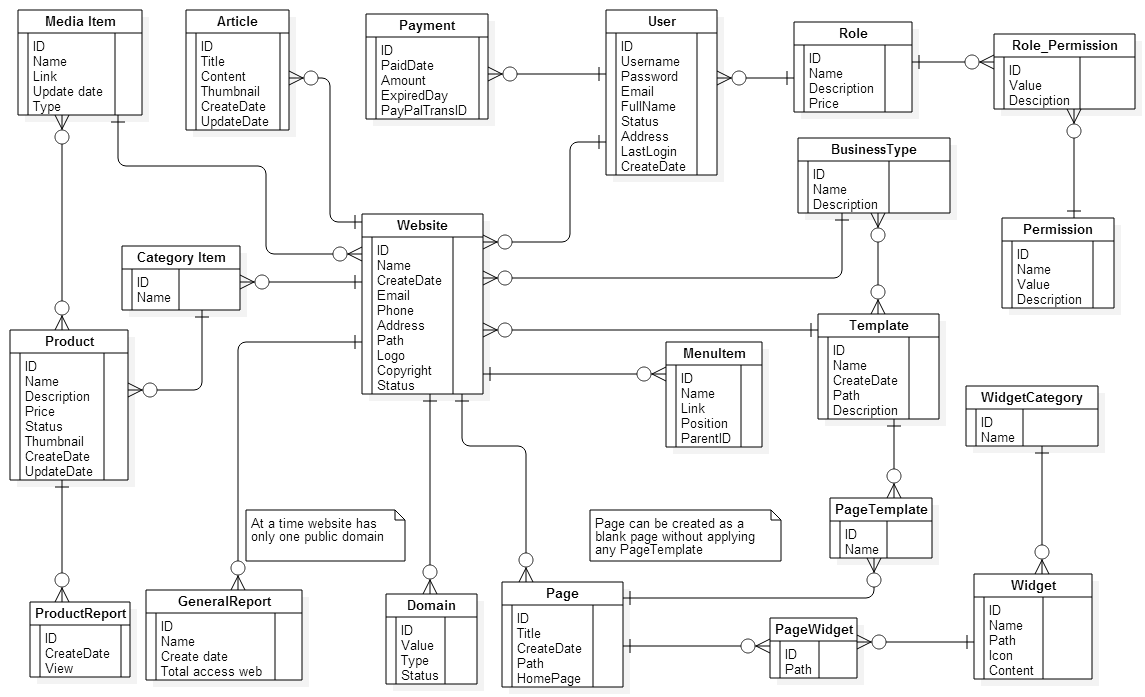


Figure 78: Logical Diagram

### Entity Dictionary

|  |  |
| --- | --- |
| Entity Data dictionary: describe content of all entities | |
| Entity Name | **Description** |
| User | Describe all user in the system |
| Businesstype | Describe all business type in the system |
| Category Item | Describe all category in the system |
| Domain | Describe all domain of website in the system |
| Media Item | Describe all media in the system |
| Page | Describe all page in the system |
| Permission | Describe all permission in the system |
| Product | Describe all prodcut in the system |
| Role | Describe all role in the system |
| Template | Describe all template in the system |
| Article | Describe all article in the system |
| Website | Describe all website in the system |
| Payment | Describe all payment in the system |
| Widget | Describe all widget in the system |
| General Report | Describe all general report in the system |
| PageTemplate | Describe all template of page in the system |
| ProductReport | Describe all report of product in the system |
| RolePermission | Describe all role permission in the system |
| WidgetCategory | Describe all category of widget in the system |
| PageWidget | Describe all widget page in the system |
| MenuItem | Desctibe menu item of website |

Table 122: Entity Dictionary

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| sss | Attributes | Description | Domain | Null |
| User | ID | Unique identifier of user, auto increment. | int | No |
| Username | Username of user (account) | nvarchar(50) | No |
| Password | Password of user (account) | nvarchar(50) | No |
| Email | Email of user (account) | nvarchar(255) | No |
| FullName | FullName of user | nvarchar(255) | No |
| Status | Status of user (account) | bit | No |
| Address | Address of user (account) | nvarchar(255) | No |
| LastLogin | Last time login of user (account) | datetime | Yes |
| CreateDate | Date create of user (account) | datetime | No |
| Businesstype | ID | Unique identifier of business type, auto increment. | int | No |
| Name | Name of business type | nvarchar(50) | No |
| Description | Description of business type | nvarchar(255) | No |
| Category Item | categoryId | Unique identifier of category, auto increment. | int | No |
| name | Name of category | nvarchar(50) | No |
| Domain | ID | Unique identifier of domain, auto increment. | int | No |
| Value | Name domain of website | nvarchar(255) | No |
| Type | Type domain of website ( buy or default) | bit | No |
| Status | Status domain of website | bit | No |
| Media Item | ID | Unique identifier of media, auto increment. | bit | No |
| Name | Name of media | nvarchar (255) | No |
| Link | Link of media | nvarchar(255) | No |
| CreateDate | Date create of media | datetime | No |
| Type | Type of media (image or video) | bit | No |
| Page | ID | Unique identifier of page, auto increment. | int | No |
| Title | Title of page | nvarchar (255) | No |
| CreateDate | Date create of page | datetime | No |
| Homepage | Home page of website | bit | No |
| Path | Path of page | nvarchar (255) | No |
| Permission | ID | Unique identifier of permission, auto increment. | int | No |
| Name | Name of permission | nvarchar(255) | No |
| Description | Description of permission | nvarchar(255) | Yes |
| Value | Value of permission | nvarchar(255) | No |
| Product | ID | Unique identifier of product, auto increment. | int | No |
| Name | Name of product | nvarchar (255) | No |
| Description | Description of product | nvarchar (255) | Yes |
| Price | Price of product | double | No |
| CreateDate | Date create of product | datetime | No |
| UpdateDate | Date update of product | datetime | No |
| Status | Status of product(active or inactive) | bit | No |
| Role | ID | Unique identifier of role, auto increment. | int | No |
| Name | Name of role | nvarchar(255) | No |
| Description | Description of role | nvarchar(255) | Yes |
| Price | Price of role | double | No |
| Template | ID | Unique identifier of template, auto increment | int | No |
| Path | Path of template | nvarchar(255) | No |
| Name | Name of template | nvarchar(255) | No |
| Description | Description of template | nvarchar(255) | Yes |
| CreateDate | Date create of template | datetime | No |
| Article | ID | Unique identifier of article, auto increment | int | No |
| Title | Title of article | nvarchar(50) | No |
| Content | Content of article | nvarchar(255) | No |
| CreateDate | Date create of article | datetime | No |
| UpdateDate | Date update of article | datetime | Yes |
| Thumbnail | Thumbnail of article | nvarchar(255) | No |
| Website | ID | Unique identifier of website, auto increment | int | No |
| Name | Name (title) of website | nvarchar(50) | No |
| CreateDate | Date create of website | Datetime | No |
| Email | Email of website | nvarchar(255) | No |
| Address | Address of website | nvarchar(255) | No |
| Status | Status of website (active or inactive) | Bit | No |
| Path | Path of website | nvarchar(255) | No |
| Logo | Logo of website | nvarchar(255) | Yes |
| Copyright | Copyright of website | nvarchar(255) | Yes |
| Payment | ID | Unique identifier of widget, auto increment | int | No |
| PaidDate | Paid date of transaction | datetime | No |
| Amount | Amount of type account | double | No |
| ExpiredDay | Expired Day of transaction | datetime | No |
| PaypalTransID | Transactions ID of paypal when payment | nvarchar(10) | No |
| Widget | ID | Unique identifier of widget, auto increment | int | No |
| Path | Path of widget | nvarchar(255) | No |
| Name | Name of widget | nvarchar(255) | No |
| Content | Description of widget | nvarchar(255) | Yes |
| Icon | Icon of widget | nvarchar(255) | No |
| GeneralReport | ID | Unique identifier of general report, auto increment | int | No |
| Name | Name of general report | nvarchar(255) | No |
| CreateDate | Date create of general report | datetime | No |
| Total Acess Web | Total access of general report | int | No |
| PageTemplate | ID | Unique identifier of media page template, auto increment | int | No |
| Name | Name of page template | nvarchar(255) | No |
| ProductReport | ID | Unique identifier of media page template, auto increment | int | No |
| View | Count view of product report | int | No |
| CreateDate | Date create of product reprot | datetime | No |
| Role\_Permission | ID | Unique identifier of media role permission, auto increment | int | No |
| Value | Value of role permission | int | No |
| PermissionId | Unique identifier of permission, foreign key | int | No |
| WidgetCategory | ID | Unique identifier of media widget category, auto increment | int | No |
| Name | Name of widget category | nvarchar(255) | No |
| PageWidget | ID | Unique identifier of media widget page, auto increment | int | No |
| Path | Path of page widget | nvarchar(255) | No |
| MenuItem | ID | Unique identifier of menu item, auto increment | int | No |
| Name | Name of menu item | nvarchar(255) | No |
| Link | Link that menu item link to | nvarchar(255) | No |
| Position | Position of menu item | int | No |
| ParentID | Unique identifier of menu item that is parent. | int | Yes |

Table 123: Detail Entity Dictionary

## Algorithms

### Widget’s structure in system

#### Definition

In SBPC system a widget is an element of a graphic user interface (GUI) that provides specific way for a user to interact with their website interface. A widget belongs to a widget category such as: product, blog, gallery, header, footer, video, contact.

#### Requirement

Each widget must be independence from other widgets. A widget can be removed from system as well as added to system without affect to websites that exits in system.

#### Solution

Each widget is structured as a folder includes:

* A file html to define the content of widget.
* May not have or have one or more Cascading Style Sheets files to define how widget is displayed.
* May not have or have one or more JavaScript files to program the behavior of widget.
* A thumbnail image to describe for widget.

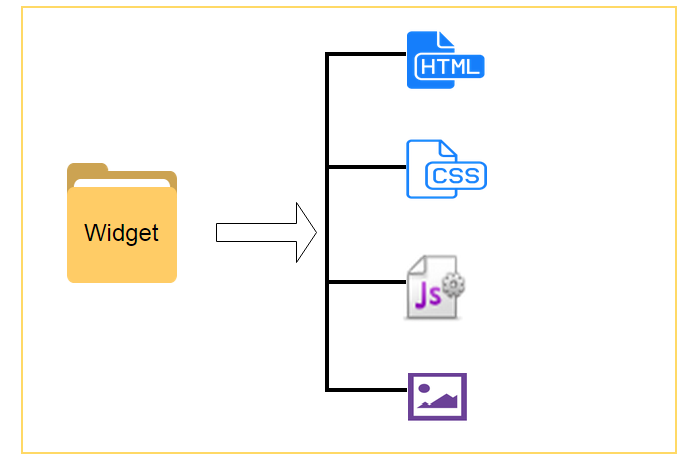


Figure 79: Widget’s Structure description

### Template’s structure in system

#### Definition

In SBPC system template is used to create website for user. Before users create their website, they must choose a template depend on their business. A business type can have many templates, so user can search on business types to find template that match their demand.

#### Requirement

A pre-designed template help users create pages easily and quickly. A template can be added to system at any time without affect to other template existed in system. A template also design base on widget’s structure.

#### Solution

Each template is structured as a folder includes:

* One or more page folders, each page folder includes:
* A file html that define how to widgets in this page are displayed.
* One or more file html that define content of widget (a blank page do not have any html widget file).
* A file html (customePage.html) to define all widgets and location of them in website.
* A thumbnail image to describe for template.

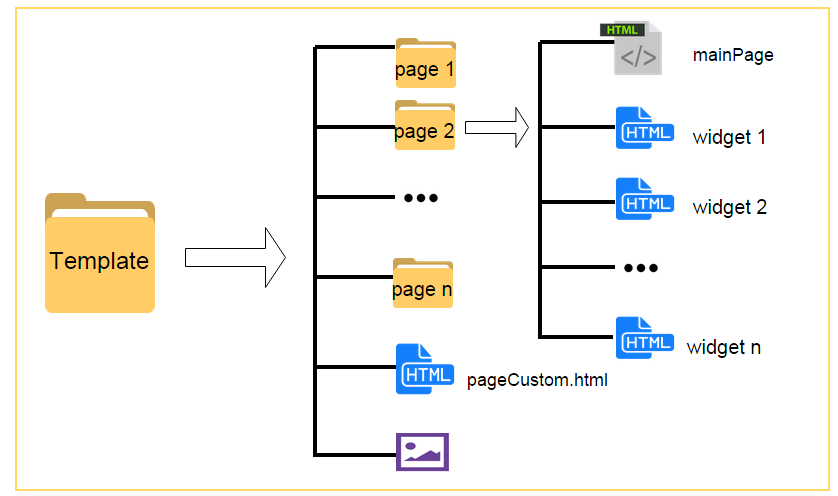


Figure 80: Template’ structure description

# System Implementation & Test

## Introduction

### Overview

This section describes the approach and methodologies used by the testing group to plan, organize and manage the testing of Site Builder for Product Catalogue. It describes the implementation details of test cases and technical details of how the product features should work. It includes test plans, test cases, test result and risks estimations.

### Test Approach

Test goal: To test the whole system based on core flow.

Test method: black-box test.

Test level: system and integration test.

## Database Relationship Diagram

### Physical Diagram

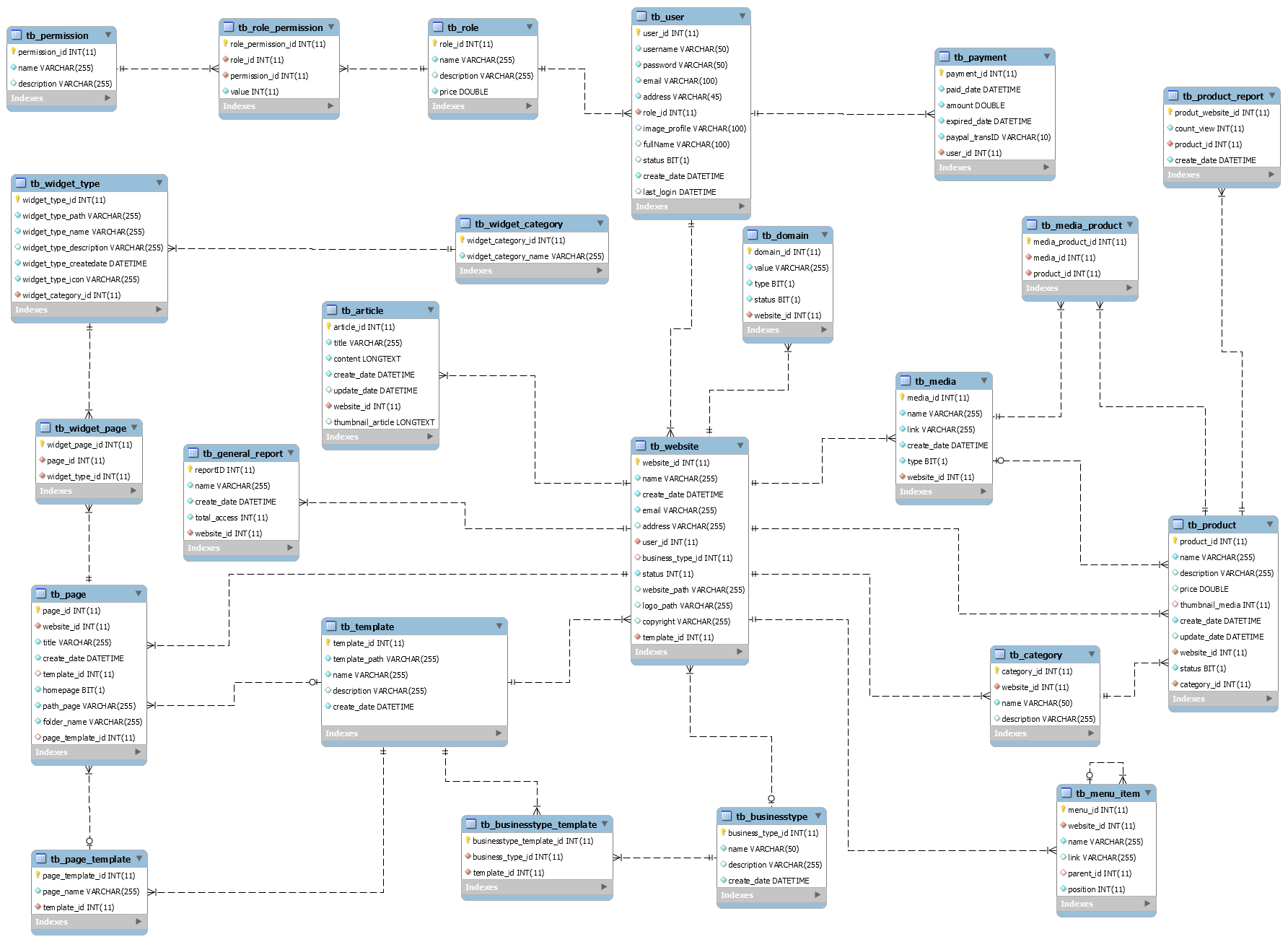


Figure 81: Physical Diagram

### Data Dictionary

|  |  |
| --- | --- |
| Data dictionary: describe content of all tables | |
| Entity Name | **Description** |
| tb\_user | Describe all user in the system |
| tb\_businesstype | Describe all business type in the system |
| tb\_Category | Describe all category in the system |
| tb\_domain | Describe all domain of website in the system |
| tb\_media | Describe all media in the system |
| tb\_page | Describe all page in the system |
| tb\_permission | Describe all permission in the system |
| tb\_product | Describe all prodcut in the system |
| tb\_role | Describe all role in the system |
| tb\_template | Describe all template in the system |
| tb\_article | Describe all article in the system |
| tb\_website | Describe all website in the system |
| tb\_payment | Describe all payment in the system |
| tb\_widget\_type | Describe all widget in the system |
| tb\_businesstype\_template | Describe all business of template in the system |
| tb\_general\_report | Describe all general report in the system |
| tb\_media\_product | Describe all media of product in the system |
| tb\_page\_template | Describe all template of page in the system |
| tb\_product\_report | Describe all report of product in the system |
| tb\_role\_permission | Describe all role permission in the system |
| tb\_widget\_category | Describe all category of widget in the system |
| tb\_widget\_page | Describe all widget page in the system |
| tb\_menu\_item | Describe all all website’s meu item of website |

Table 124: Data Dictionary

| Entity Name | Attributes | Description | Domain | Null |
| --- | --- | --- | --- | --- |
| tb\_user | user\_id | Unique identifier of user, auto increment. | int | No |
| username | Username of user (account) | nvarchar(50) | No |
| password | Password of user (account) | nvarchar(50) | No |
| email | Email of user (account) | nvarchar(100) | No |
| fullName | Full name of user | nvarchar(100) | Yes |
| status | Status of user (account) | bit | Yes |
| address | Address of user (account) | nvarchar(255) | No |
| last\_login | Last time login of user (account) | datetime | Yes |
| image\_profile | Image avatar of user (account) | nvarchar(100) | Yes |
| create\_date | Date create of user (account) | datetime | No |
| roleId | Unique identifier of role user, foreign key | int | No |
| tb\_businesstype | business\_type\_id | Unique identifier of business type, auto increment. | int | No |
| name | Name of business type | nvarchar(50) | No |
| description | Description of business type | nvarchar(255) | Yes |
| create\_date | Date create of business type | datetime | No |
| tb\_category | category\_id | Unique identifier of category, auto increment. | int | No |
| website\_id | Unique identifier of website, foreign key | int | No |
| name | Name of category | nvarchar(50) | No |
| description | Description of category | nvarchar(255) | Yes |
| tb\_domain | domain\_id | Unique identifier of domain, auto increment. | int | No |
| value | Name domain of website | nvarchar(255) | No |
| type | Type domain of website ( buy or default) | bit | No |
| status | Status domain of website | bit | No |
| website\_id | Unique identifier of website, foreign key | int | No |
| tb\_media | media\_id | Unique identifier of media, auto increment. | bit | No |
| name | Name of media | nvarchar (255) | No |
| link | Link of media | nvarchar(255) | No |
| create\_date | Date create of media | datetime | No |
| type | Type of media (image or video) | bit | No |
| website\_id | Unique identifier of website, foreign key | int | No |
| tb\_page | page\_id | Unique identifier of page, auto increment. | int | No |
| title | Title of page | nvarchar (255) | No |
| create\_date | Date create of page | datetime | No |
| homepage | Home page of website | bit | No |
| path\_page | Path of page | nvarchar (255) | No |
| folder\_name | Name website page of page | nvarchar(255) | No |
| website\_id | Unique identifier of website, foreign key | int | No |
| template\_id | Unique identifier of template, foreign key | int | Yes |
| page\_template\_id | Unique identifier of template’s page, foreign key | Int | Yes |
| Tb\_permission | permission\_id | Unique identifier of permission, auto increment. | int | No |
| name | Name of permission | nvarchar(255) | No |
| description | Description of permission | nvarchar(255) | Yes |
| tb\_product | product\_id | Unique identifier of product, auto increment. | int | No |
| name | Name of product | nvarchar (255) | No |
| description | Description of product | nvarchar (255) | Yes |
| price | Price of product | double | No |
| create\_date | Date create of product | datetime | No |
| update\_date | Date update of product | datetime | Yes |
| status | Status of product(active or inactive) | bit | No |
| website\_id | Unique identifier of website, foreign key | int | No |
| thumbnail\_media | Unique identifier of media, foreign key | int | No |
| category\_id | Unique identifier of category, foreign key | int | No |
| tb\_role | role\_id | Unique identifier of role, auto increment. | int | No |
| name | Name of role | nvarchar(255) | No |
| description | Description of role | nvarchar(255) | Yes |
| price | Price of role | double | No |
| tb\_template | template\_id | Unique identifier of template, auto increment | int | No |
| template\_path | Path of template | nvarchar(255) | No |
| name | Name of template | nvarchar(255) | No |
| description | Description of template | nvarchar(255) | Yes |
| create\_date | Date create of template | datetime | No |
| tb\_article | article\_id | Unique identifier of article, auto increment | int | No |
| title | Title of article | nvarchar(255) | No |
| content | Content of article | longtext | No |
| create\_date | Date create of article | datetime | No |
| update\_date | Date update of article | datetime | Yes |
| thumbnail\_article | Thumbnail of article | longtext | No |
| website\_id | Unique identifier of website, foreign key | int | No |
| tb\_website | website\_id | Unique identifier of website, auto increment | int | No |
| name | Name (title) of website | nvarchar(255) | No |
| create\_date | Date create of website | Datetime | No |
| email | Email of website | nvarchar(255) | No |
| address | Address of website | nvarchar(255) | Yes |
| status | Status of website (active or inactive) | Bit | No |
| website\_path | Path of website | nvarchar(255) | Yes |
| logo\_path | Logo of website | nvarchar(255) | Yes |
| copyright | Copyright of website | nvarchar(255) | Yes |
| user\_id | Unique identifier of user, foreign key | int | No |
| template\_id | Unique identifier of template, foreign key | int | No |
| business\_type\_id | Unique identifier of business type, foreign key | int | No |
| tb\_payment | payment\_id | Unique identifier of payment, auto increment | int | No |
| paid\_date | Paid date of transaction | datetime | No |
| amount | Amount of type account | double | No |
| expired\_date | Expired Day of transaction | datetime | No |
| paypal\_transID | Transactions ID of paypal when payment | nvarchar(10) | No |
| user\_id | Unique identifier of user, foreign key | int | No |
| tb\_widget\_type | widge\_type\_id | Unique identifier of widget, auto increment | int | No |
| widget\_type\_path | Path of widget | nvarchar(255) | No |
| widget\_type\_name | Name of widget | nvarchar(255) | No |
| widget\_type\_description | Description of widget | nvarchar(255) | Yes |
| widget\_type\_createdate | Date create of widget | datetime | No |
| widget\_type\_icon | Icon of widget | nvarchar(255) | No |
| widget\_type\_categoryId | Unique identifier of widget category, foreign key | int | No |
| tb\_businesstype\_template | businesstype\_template\_id | Unique identifier of business type template, auto increment | int | No |
| template\_id | Unique identifier of template, foreign key | int | No |
| businesstype\_type\_id | Unique identifier of business type, foreign key | int | No |
| tb\_general\_report | reportID | Unique identifier of general report, auto increment | int | No |
| name | Name of general report | nvarchar(255) | No |
| create\_date | Date create of general report | datetime | No |
| total\_access | Total access of general report | int | No |
| website\_id | Unique identifier of website, foreign key | int | No |
| tb\_media\_product | media\_product\_id | Unique identifier of media product, auto increment | int | No |
| product\_id | Unique identifier of product, foreign key | int | No |
| media\_id | Unique identifier of media, foreign key | int | No |
| tb\_page\_template | page\_template\_id | Unique identifier of media page template, auto increment | int | No |
| page\_name | Name of page template | nvarchar(255) | No |
| template\_id | Unique identifier of media, foreign key | int | No |
| tb\_product\_report | produt\_website\_id | Unique identifier of media page template, auto increment | int | No |
| count\_view | Count view of product report | int | No |
| createDate | Date create of product reprot | datetime | No |
| product\_id | Unique identifier of product, foreign key | int | No |
| tb\_role\_permission | role\_permission\_id | Unique identifier of media role permission, auto increment | int | No |
| value | Value of role permission | int | No |
| role\_id | Unique identifier of role, foreign key | int | No |
| permission\_id | Unique identifier of permission, foreign key | int | No |
| tb\_widget\_category | widget\_category\_id | Unique identifier of media widget category, auto increment | int | No |
| widget\_category\_name | Name of widget category | nvarchar(255) | No |
| tb\_widget\_page | widget\_page\_id | Unique identifier of media widget page, auto increment | int | No |
| page\_id | Unique identifier of page, foreign key | int | No |
| widget\_id | Unique identifier of widget, foreign key | int | No |
| tb\_menu\_item | menu\_id | Unique identifier of menu item, auto increment | Int | No |
| name | Menu display name | nvarchar(255) | No |
| link | Link of menu | nvarchar(255) | Yes |
| parent\_id | Unique identifier of menu item, foreign key | int | No |
| position | Position of menu item | int | No |

Table 125: Attribute Data Dictionary

## Performance Measures

### Definition

This section tests the general page load speed from all the page of the web application.

### Test environment

Server:

* Operation system: Windows Server 2008
* RAM: 1GB
* Processor: Intel® Core™ i3 (2.50GHz)
* Network: 1Mbps

### Test case

We use browser from client side and go to each page of the web application and record the page load time.

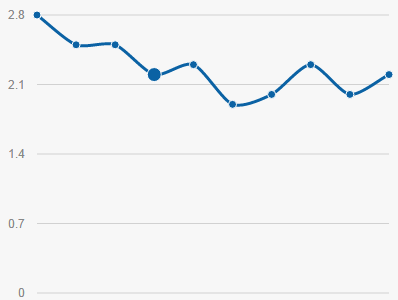
Test Page:

|  |  |
| --- | --- |
| Public | Home |
| Register |
| User | List website |
| Website Detail |
| Website Media |
| Website Article |
| Website Product |
| Website Statistic |
| Admin | Manage Website |
| Manage Account |
| Manage Account Type |
| Manage Template |
| Manage Widget |
| System Statistic |

Table 126: List page to test

### Test result

The test is run 10 times, each time we go through all the pages listed above and log down the average page load time.



Website have low response time, but some complicated processing as saving custom page require more time to execute.

## Test plan

The purpose of this section is to describe the overall test plan and strategy for testing the SPBC system .This section is to ensure SPBC system meets its entire technical, functional and business requirement. The following part will describe which features to be tested and which will not.

### Features to be tested

User: view list website, create website, view website detail, view list product, create categories, create product, view product detail, update product, delete product, upload image/video, delete image/video, update website page, save and publish website, upgrade account, view website statistic, login, logout, forgot password, register.

Admin: manage widget, manage template, manage business type, view account type detail, update account type detail, view account list, update account status, view system statistic.

### Features not to be tested

* View domain, change domain status, search product.

## System Testing Test Case

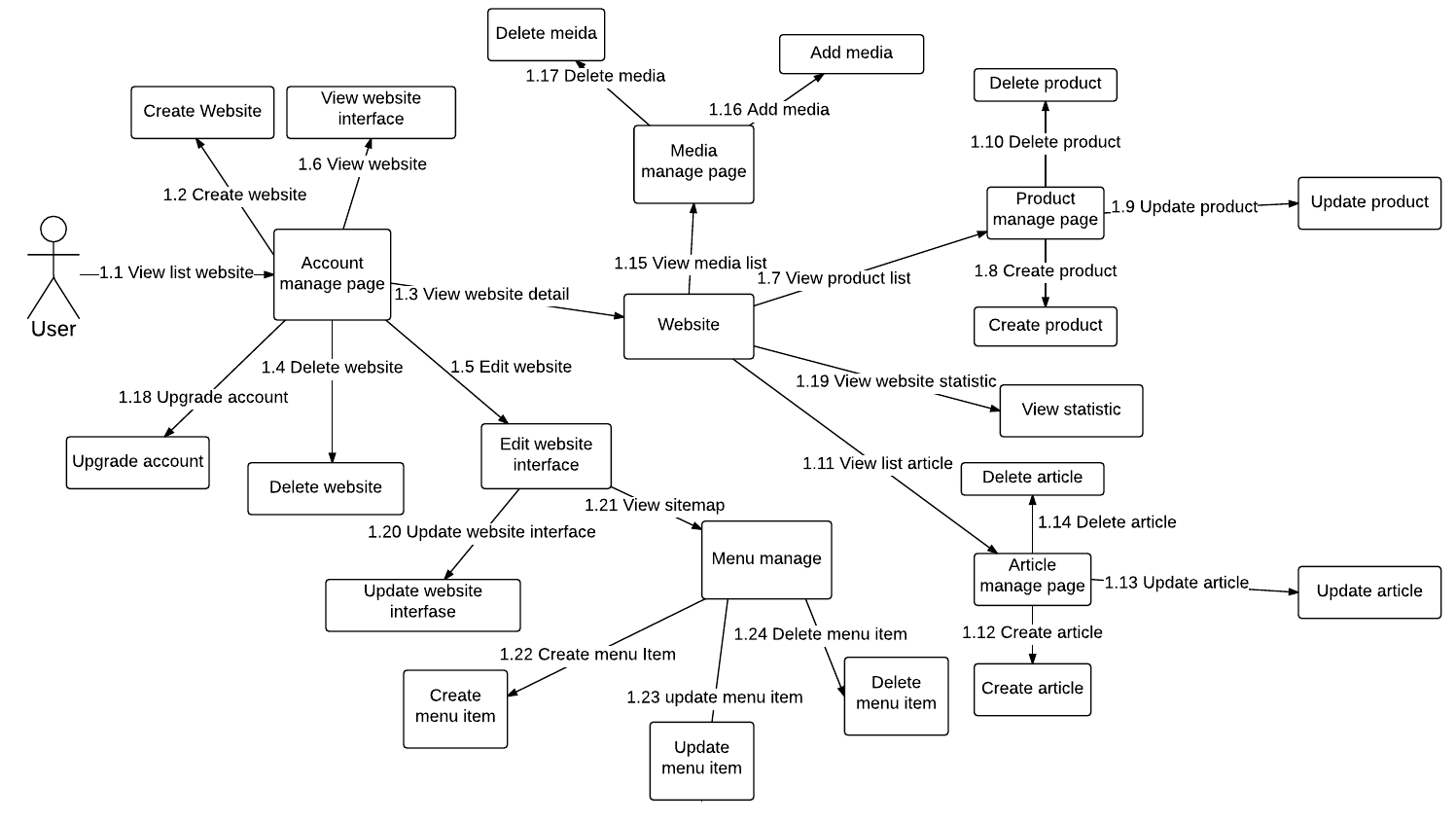


Figure 82: User Test case flow overview

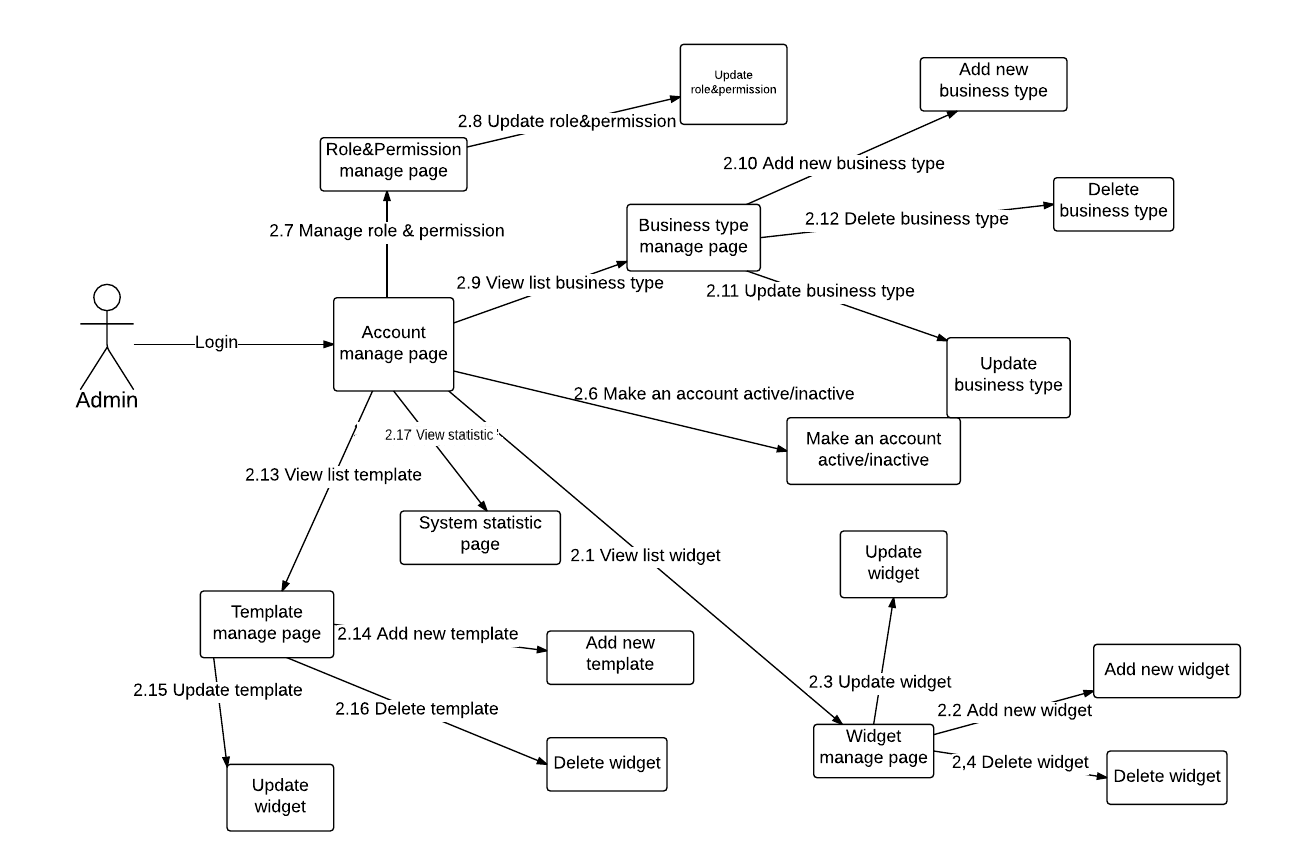


Figure 83: Admin Test case flow overview

### User’s core flow

#### View list website

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| LW01 | User view list website when don’t have website | Login new account or account don’t have website. | Show website page, include suggestion for create new website and button: “Tạo trang web”. | N/A | Pass | 10/08/2015 |  |
| LW02 | User view list website | Login user account which have websites. | Show website page with list website of that user. For each website have 3 buttons to edit, delete and view website; and a link to website detail page. | N/A | Pass | 10/08/2015 |  |

#### Create website

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| CW01 | User create new website | 1. On list website page, click button “Tạo trang web ".  2. On page choose business type and template.  3. On page input website name and choose website domain, click button “Tạo mới”. | - After step 1, go to choose template page.  - After step 2, go to choose website name and website domain.  - After step 3, new website are created and go to page to edit website interface. | LW01, LW02 | Pass | 10/08/2015 |  |
| CW02 | User create new website but number of website is reached the limit | 1. Login with account free user and account has 4 websites. Free user has permission to create 5 websites. On list website page, click button “Tạo trang web". | Show message: “Tài khoản của bạn đã đạt giới hạn website.” | LW02 | Pass | 10/08/2015 |  |
| CW03 | User create new website but lack of input | 1. On list website page, click button “Tạo trang web".  2. On page choose business type and template.  3. Not fill website name and click button.  4. Input website name and click button “Tạo mới”.  5. Choose a domain name and click button “Tạo mới”. | - Go to choose template page.  - Go to choose website name and website domain.  - Show message “Nhập tên trang web”.  - Show message “Chọn địa chỉ trang web”.  - Go to page to custom website interface. | LW01, LW02 | Pass | 10/08/2015 |  |
| CW04 | User create new website but domain name is existed | 1. On system existed domain ““/SBPC/abc/””.  2. On list website page, click button “Tạo trang web".  3. On page choose business type and template.  4. Input website name “abc” and choose domain name “/SBPC/abc/” | - After step 2, go to choose template page.  - After step 3, go to choose website name and website domain.  - Show message “Địa chỉ trang web đã tồn tại”. | LW02 | Pass | 10/08/2015 |  |

#### View website detail

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| WD01 | View website detail | On website list page, click on link to a website detail. | Show website detail page. That includes link to manage product page, manage media page, manage article page, manage website domain, view statistic for website and custom website page. | LW02 | Pass | 10/08/2015 |  |

#### Delete website

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| WD01 | Delete website | On website list page, click on a website’s button “Xóa”.  Click on button “Có”. | Show message “Bạn có chắc chắn muốn xóa trang này. Không thể khôi phục sau khi thay đổi ! Tiếp tục?”.  Delete website reload website list. | LW02 | Pass | 10/08/2015 |  |

#### Edit website

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| ED01 | Edit website | On website list page, click on a website’s button “Thay đổi”. | Go to page to update website interface | LW02 | Pass | 10/08/2015 |  |

#### View website

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| VW01 | View website | On website list page, click on a website’s button “Xem”. | Go to home page of sub website to review. | LW02 | Pass | 10/08/2015 |  |

#### View list product

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| VP01 | View list product | On website detail page, click on link to product list page. | Show manage product page, which include a bar to add, delete or change name of category, list product in this website and a button for create new product. For each product has a button group to view and update product detail, delete product. | WD01 | Pass | 10/08/2015 |  |

#### Create new product

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| CP01 | Create new product | On product manage page, click on “Thêm sản phẩm mới” button.  Input product information and click on button “Thêm”. | Show a popup to input product’s detail. Popup includes product’s name, category, price, description; 2 tabs to upload new image or choose existed image for product.  Popup is close and have one more product in product list. Show message “Thêm sản phẩm thành công”. | VP01 | Pass | 10/08/2015 |  |
| CP02 | Create new product but lack of input | On product manage page, click on “Thêm sản phẩm mới” button.  Click button “Thêm”.  Input product name and click button “Thêm”.  Input product price and click button “Thêm”. | Show a popup to input product’s detail. Popup includes product’s name, category, price, description; 2 tabs to upload new image or choose existed image for product.  Show message “Vui lòng nhập tên sản phẩm.”.  Show message “Vui lòng nhập giá sản phẩm.”.  Popup is close and have one more product in product list. Show message “Thêm sản phẩm thành công”. | VP01 | Pass | 11/08/2015 |  |
| CP03 | Create product when don’t have product category | On product manage page, click on “Thêm sản phẩm mới” button.  Input new category and click on link “Hoàn tất”.  Click on “Thêm sản phẩm mới” button.  Input product information and click on button “Thêm”. | Show message “Bạn hãy thêm danh mục trước khi thêm sản phẩm”.  Show a popup to input product’s detail. Popup includes product’s name, category, price, description; 2 tabs to upload new image or choose existed image for product.  Popup is close and have one more product in product list. Show message “Thêm sản phẩm thành công”. | VP01 | Pass | 11/08/2015 |  |

#### Update product

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| UD01 | Update Product | On product manage page, click on “Chỉnh sửa thông tin” button of a product.  Input product information and click on button “Thêm”. | Show a popup to input product’s detail. Product’s details are loaded to input fields.  Close popup and show message “Chỉnh sửa thông tin sản phẩm thành công” | VP01 | Pass | 11/08/2015 |  |

#### Delete product

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| DP01 | Delete Product | On product manage page, click on “Xóa sản phẩm” button of a product.  Click on button “Có”. | Show a popup to confirm deleting product.  Delete that product, close popup and show message “Đã xóa sản phẩm” | VP01 | Pass | 11/08/2015 |  |

#### View article list

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| LA01 | User view article list | On website detail page, click on link to article manager page | Show article manage page, that includes a table list of article belong to that website, a button to create new article. For each article has a button group to view detail, edit and delete that article. On header of table have an input for search in article list. | WD01 | Pass | 10/08/2015 |  |

#### Create article

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| CA01 | Create new article | On article manage page, click on “Bài viết mới” button.  Input article information and click on button “Thêm”. | Show a popup to input article’s detail. Popup includes article’s title, content, thumbnail.  Add new article and close popup. Show message “Thêm/Chỉnh sửa bài viết thành công”. | LA01 | Pass | 11/08/2015 |  |
| CA02 | Create new article but lack of input | On article manage page, click on “Bài viết mới” button.  Click button “Thêm”.  Input article title and click button “Thêm”.  Input product price and click button “Lưu lại”. | Show a popup to input article’s detail. Popup includes article’s title, content, thumbnail.  Show message “Vui lòng nhập tiêu đề bài viết.”.  Show message “Vui lòng nhập nội dung bài viết.”.  Add new article and close popup. Show message “Thêm/Chỉnh sửa bài viết thành công”. | LA01 | Pass | 11/08/2015 |  |

#### Update article

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| UA01 | Update Product | On article manage page, click on “Sửa” button of an article.  Input article’s details and click on button “Lưu lại”. | Show a popup to input article’s detail. Article’s details are loaded to input fields.  Change article’s detail. Show message “Thêm/Chỉnh sửa bài viết thành công”. | LA01 | Pass | 11/08/2015 |  |

#### Delete article

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| DA01 | Delete Article | On article manage page, click on “Xóa” button of a article.  Click on button “Xóa bài”. | Show a popup to confirm deleting Article.  Delete that article, close popup and show message “Xóa bài viết thành công”. | LA01 | Pass | 11/08/2015 |  |

#### View media list

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| LM01 | User view media list | On website detail page, click on link to media manager page.  On media manage page, click on tab navigation to go to video tab. | Show media manage page, which includes 2 tabs to manage image and video. On default, image list are shown.  Show list of video belong to that website | WD01 | Pass | 10/08/2015 |  |

#### Create media

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| AM01 | Add new image | On media manage page, click on “Tạo mới” button.  Click on file input.  Choose one or some image file, click on “Tải ảnh” button. | Show a popup to that content a file input to upload one or more images.  Open file choose dialog.  Open dialog for upload process bar. When upload complete, add media to list and close popup | LM01 | Pass | 11/08/2015 |  |
| AM02 | Add document file instead for image file | On media manage page, click on “Tạo mới” button.  Click on file input.  Choose one or some document file, click on “Tải ảnh” button. | Show a popup to that content a file input to upload one or more images.  Open file choose dialog.  Show message “Định dạng file không hợp lệ.”. | LM01 | Pass | 11/08/2015 |  |
| AM03 | Add new video | On media manage page, click on tab Video, click on “Tạo mới” button.  Click on file input.  Choose a video file, click on “Tải video” button. | Show a popup to that content a file input to upload video  Open file choose dialog.  Open dialog for upload process bar. When upload complete, add media to list and close popup | LM01 | Pass | 11/08/2015 |  |
| AM04 | Add new media when website reached storage limit | Website used 200MB storage and reached the storage limit.  On media manage page, click on tab Video, click on “Tạo mới” button. | Show message “Website của bạn đã đạt tới giới hạn dung lượng. Xin thử lại sau!” | LM01 | Pass | 11/08/2015 |  |
| AM05 | Media size bigger than the free storage of website | Website used 198MB storage. And account have storage limit is 200MBOn media manage page, click on tab Video, click on “Tạo mới” button.  Click on file input.  Choose a bigger than 5MB video file, click on “Tải video” button. | Show a popup to that content a file input to upload video  Open file choose dialog.  Show message “Dung lượng trang web không đủ, hãy chọn video khác.” | LM01 | Pass | 11/08/2015 |  |
| AM06 | Add new video that have size bigger than 100MB | On media manage page, click on tab Video, click on “Tạo mới” button.  Click on file input.  Choose a bigger than 100MB video file, click on “Tải video” button. | Show a popup to that content a file input to upload video  Open file choose dialog.  Show message “Dung lương video vượt quá mức cho phép của hệ thông.” | LM01 | Pass | 11/08/2015 |  |

#### Delete media

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| DM01 | Delete Media | On article manage page, click on “X” button on top right position of media.  Click on button “Có”. | Show a popup to confirm deleting media.  Delete that article, close popup and show message “Xóa media thành công”. | LM01 | Pass | 11/08/2015 |  |

#### Upgrade account

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| UA01 | Upgrade Account | Click on “Nâng cấp TK” on side bar  Choose account type and click “Nâng cấp” | Go to page to upgrade account.  Go to payment page of Paypal. After checkout. Auto return to User dashboard | N/A | Pass | 11/08/2015 |  |

#### View website statistic

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| VS01 | View website statistic | On website detail page, click on link to view website statistic page. | Show website statistic page. That includes number of product in website, storage of this website, top 10 product, viewing in a month and a year | WD01 | Pass | 11/08/2015 |  |

#### Save customization (update website interface)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| UI01 | Drag & drop widgets | On website manage page, click on “Thay đổi” button of a website, or click on link “Chỉnh sửa giao diện” on website detail page.  Click on radio button “Kéo thả”. On toolbar, choose a widget and drag to page. | Go to page to custom interface.  Generate an element on page. | EW01 | Pass | 11/08/2015 |  |
| UI02 | Edit page element | On page to custom interface, click on radio button “Sửa nội dung”. Click on an element on a page.  Edit text and click “Lưu”. | Show popup to edit text of that element.  Text of element change. | EW01 | Pass | 11/08/2015 |  |
| UI03 | Edit page element’s option | On page to custom interface, click on radio button “Chỉnh sửa nội dung”. Click on an element on a page.  Change element option and click “Lưu” | All option of element is show on toolbar.  Element option change. | EW01 | Pass | 11/08/2015 |  |
| UI04 | Save custom page | After custom page, click “Lưu”.  Click “Lưu thay đổi”. | Show popup to confirm saving web page.  Save all page change and reload page. | EW01 | Pass | 11/08/2015 |  |
| UI04 | Save custom page and publish website | After custom page, click “Lưu”.  Click “Lưu và công khai”. | Show popup to confirm saving web page.  Save all page change and reload page. | EW01 | Pass | 11/08/2015 |  |

#### View sitemap

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| VSM01 | User view sitemap | On website detail page, click on link to sitemap | Show menu item manage page that includes a list of menu items in website. Menu item are used to binding to menu widget on website. List menu will display like hierarchies tree. User can drag&drop a menu item to change position or set menu item as child of another. | WD01 | Pass | 11/08/2015 |  |

#### Create menu item

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| CMI | Create new menu item to existed page | On article manage page, click on “Thêm” button.  Click on button “Thêm”.  Fill name and click “Thêm”.  Choose an existed page and click “Thêm” | Show a popup to input menu item details. Popup includes item’s name, 2 checkbox to know item’s link is in same website or from another website and for item’s link is a text field if link from another web and a select list if link to a page in same web.  Show message “Tên menu phải từ 2 đến 35 ký tự”.  Show message “Đường dẫn không hợp lệ”.  Add new menu item, close popup and show message “Thêm thành công” | VSM01 | Pass | 11/08/2015 |  |
| CA02 | Create new article but lack of input | On article manage page, click on “Thêm” button.  Click on button “Thêm”.  Fill name and click “Thêm”.  Choose checkbox to page from other website. Input “http:\\www.google.com” | Show a popup to input menu item details. Popup includes item’s name, 2 checkbox to know item’s link is in same website or from another website and for item’s link is a text field if link from another web and a select list if link to a page in same web.  Show message “Tên menu phải từ 2 đến 35 ký tự”.  Show message “Đường dẫn không hợp lệ”.  Add new menu item, close popup and show message “Thêm thành công” | LA01 | Pass | 11/08/2015 |  |

#### Update menu item

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| UA01 | Update Product | On article manage page, click on “Sửa” button of an menu item.  Input new menu item details and click on button “Lưu lại”. | Show a popup to input article’s detail..  Change item details, close popup and show message “Thay đổi thành công” | LA01 | Pass | 11/08/2015 |  |

#### Delete menu

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| DMI01 | Delete menu item | On article manage page, click on “Xóa” button of menu item.  Click on button “Xóa”. | Show a popup to confirm deleting item.  Delete that item, close popup and show message “Xóa thành công”. | VMI01 | Pass | 11/08/2015 |  |

### Admin’ core flow

#### View list widget

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| LWI01 | View list widget | After login with account have admin role, click on “Tiện ích” on side bar. | Show widget manage page. Page includes a button for add new widget, a list of widget available in the system. For each widget, have a button group for delete, edit or review widget. | N/A | Pass | 11/08/2015 |  |

#### Create new widget

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| AWI01 | Add new widget | On widget manage page click on “Tải tiện ích mới” button.  Click “Tải lên”.  Choose a file .zip and click “Tải lên”. | Show popup to admin choose a zip file to upload a widget, choose category and write a little description for that widget.  Show message “File tiện ích có dạng .zip”.  Add new widget and reload list widget. | LWI01 | Pass | 11/08/2015 |  |

#### Update widget

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| UWI01 | Update widget | On widget manage page click on “Sửa” button.  Delete name and click “Chỉnh sửa”.  Fill a name, choose category “Menus”. Click “Chỉnh sửa”. | Go to page to edit and update widget’s details. Admin can change name, image, category or html of that widget  Show message “Tên tiện ích phải có ít nhất 4 ký tự và nhiều nhất là 32 ký tự”.  Turn back to widget manage page and show message “Tiện ích đã chỉnh sửa thành công”. | LWI01 | Pass | 11/08/2015 |  |

#### Delete widget

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| DWI01 | Delete widget | On widget manage page, click on “Xóa” button of a widget “Content-1”.  Click button “Xóa”. | Show popup to confirm deleting widget.  Reload list widget and show message “Tiện ích Content-1 đã xóa thành công”. | LWI01 | Pass | 12/08/2015 |  |

#### View list user

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| LU01 | View list user | After login with account have admin role. System will go to user manage page. | Show a page for manage user. Page includes a table of list user in system. List not content account with admin role. For each account have a group button for active or make an account inactive. | N/A | Pass | 12/08/2015 |  |

#### Change status user

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| UU01 | Make an account active/inactive | On user manage page, click on “Ngưng hoạt động” button of an on active account.  Click button “Có”. | Show popup to confirm changing status of user.  Change status of user, button “Ngưng hoạt động” turn to red and cannot be click, button “Hoạt động” turn to grey. Show message “Thay đổi trạng thái của thành viên thành công”. | LU01 | Pass | 12/08/2015 |  |

#### View role and permission

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| MR01 | View role & permission manage page | After login with account have admin role, click on link “Quyền hạn” to go to role & permission manage page. | Show page for manage role & permission. Page includes 3 tabs for 3 type of user: free, silver and gold. End of each tab have a button “Chỉnh sửa” to update that role’s details. | N/A | Pass | 12/08/2015 |  |

#### Update role and permission

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| UR01 | Update role & permission | On role & permission manage page, click on “Chỉnh sửa” button of an role.  Delete role name, click button “Cập nhật”.  Fill role name, fill role price -5, click button “Cập nhật”.  Fill role price 3, click button “Cập nhật”. | Show popup to admin edit role’s detail and permission.  That includes role name, role price, list permission and value.  Show message “Nhập tên quyền hạn”.  Show popover “Giá trị phải lớn hơn hoặc bằng 0”.  Change role’s details and permission. Show message “Thay đổi thành công”. | MR01 | Pass | 12/08/2015 |  |

#### View list business type

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| LB01 | View list business type | After login account which have admin role, click on link “Kinh doanh” to go to business type manage page. | Show page for manage business type. Page includes a button to add new business type, a table of list business type. For each row of business type have a button group to edit or delete business type. | N/A | Pass | 12/08/2015 |  |

#### Create business type

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| AB01 | Add new business type | On business type manage page, click on “Thêm loại kinh doanh”.  Click button “Lưu lại”.  Fill business type name, click button “Lưu lại”.  Fill business type description, click button “Lưu lại”. | Show popup to input business type’s details. It includes name, description.  Show message “Tên loại kinh doanh ít nhất 6 ký tự và nhiều nhất 32 ký tự”.  Show message “Xin nhập mô tả cho loại hình kinh doanh”.  Add new business type and show message “Thêm mới loại hình kinh doanh thành công”. | LB01 | Pass | 12/08/2015 |  |

#### Update business type

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| UB01 | Update business type | On business type manage page, click on button “Sửa” of a business type.  Chang detail of business type, click button “Lưu lại”. | Show popup that includes old details of business type. Admin can edit detail and update.  Update business type and show message “Cập nhật thành công”. | LB01 | Pass | 12/08/2015 |  |

#### Delete business type

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| UB01 | Delete business type | On business type manage page, click on button “Xóa” of a business type.  Click button “Có”. | Show popup to confirm business type deleting.  Remove business type and show message “Xóa loại hình kinh doanh thành công”. | LB01 | Pass | 12/08/2015 |  |

#### View list template

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| LTI01 | View list template | After login account which have admin role, click on “Giao diện” on side bar. | Show template manage page. Page includes a button for add new template, a list of template available in the system. For each template, have a button group for edit or template. | N/A | Pass | 11/08/2015 |  |

#### Create template

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| ATI01 | Add new template | On template manage page click on “Tải mẫu giao diện” button.  Click “Tải lên”.  Choose a file .zip and click “Tải lên”.  Choose one or some business type for template and click “Tải lên”. | Show popup to admin choose a zip file to upload a widget, choose business type and write a little description for that template.  Show message “File tiện ích có dạng .zip”.  Show message “Mẫu chưa nằm trong loại hình kinh doanh nào cả”.  Add new template and reload list template. | LT01 | Pass | 12/08/2015 |  |

#### Update template

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| UWI01 | Update template | On template manage page click on “Sửa” button.  Delete name and click “Chỉnh sửa”.  Fill a name, uncheck all business type. Click “Chỉnh sửa”. | Go to page to edit and update template’s details. Admin can change name, image, business type of template.  Show message “Tên mẫu phải có ít nhất 4 ký tự và nhiều nhất là 32 ký tự”.  Show message “Mẫu chưa thuộc về loại hình kinh doanh nào cả”. | LT01 | Pass | 12/08/2015 |  |

#### Delete template

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| DTI01 | Delete Template | On template manage page, click on “Xóa” button of a template”.  Click button “Xóa”. | Show popup to confirm deleting template.  Reload list template and show message “Hệ thống vừa xóa mẫu giao diện”. | LT01 | Pass | 12/08/2015 |  |

#### View statistic

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Test Case Description** | **Test Case Procedure** | **Expected Output** | **Inter-test Case Dependence** | **Result** | **Test Date** | **Note** |
| VSS01 | View system statistic | After login account which have admin role, click on link to view system statistic page. | Show system statistic page. That includes number of account and number of website in system, top 10 website, and statistic detail for each website. | N/A | Pass | 12/08/2015 |  |

# Software User’s Manual

## Installation Guide

### Setting up environment at server side

The following software must be installed into the server machine:

#### Hardware requirements

Server computer for deploying with the minimum configuration:

* CPU Intel® Core 2 Duo
* 2GB RAM.
* 30GB of hard disk.
* Internet

#### Software requirements

* Web Server: Apache Tomcat 8.0.23 with JDK 8.
* Operating system: Windows Server 2008, or above
* MySQL: used to create and manage database of web application.

### Deployment at server side

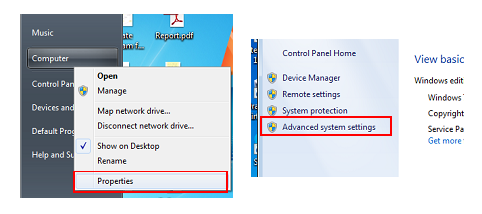
#### Prepare deployment package

Deployment package are write to disk. Package contains:

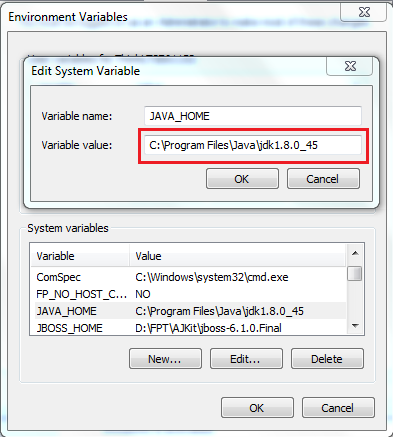
* File setup JDK 8.0.45, MySQL 5.5.6, Apache Tomcat 8.0.3.
* File script database sitebuilder.sql .
* Web service package SBPC.war.

#### Configure Server before deploy

Please follow bellow step: In **Start** menu, right click on **Computer**, choose **Properties**, and click on **Advanced system settings, in “Advanced”** tab select **Environment variables.**



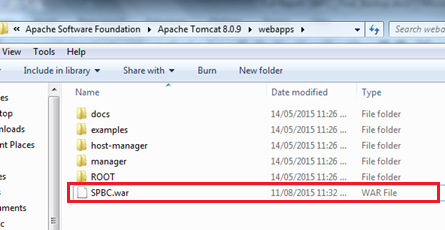
Add variable **JAVA\_HOME**: Directory path of Java Development Kit



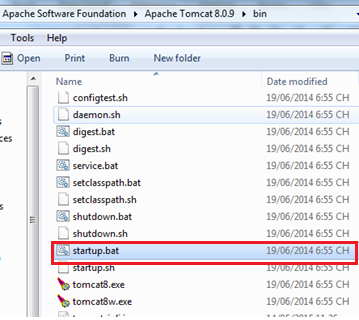
Run sitebuilder.sql to prepare database for application. Make sure MySQL run in port 3306.

#### Deploy web application on server

Please copy the deployment package SBPC.war to folder **webapps** in the directory of Tomcat server.



Click run startup.bat in the folder **bin.**



### Setting up the environment at client side

#### Setting up for computer

Client side should have one of these following browsers:

* Google Chrome
* Firefox
* Internet Explorer

## User Guide

This user guide describes how to use the application based on the process of site builder for product catalogue website.

### User

#### Manage Website

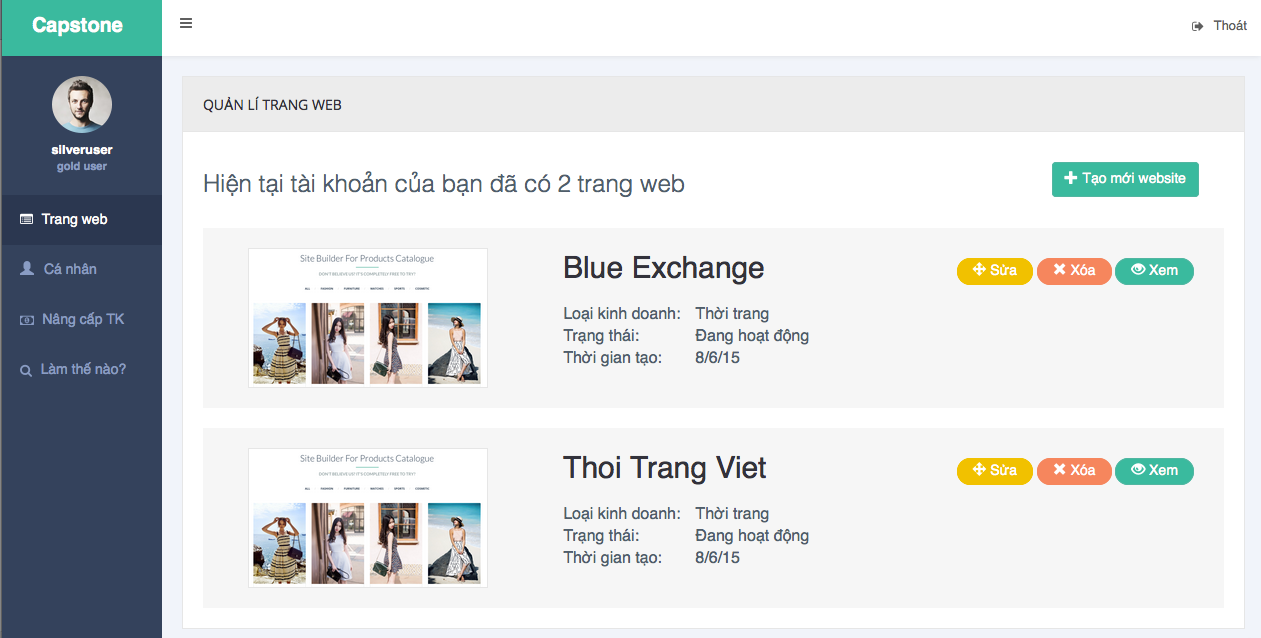


Figure 84: User – Manage Website Page

|  |  |
| --- | --- |
| **Step** | **Description** |
| 1 | Click “Trang Web” on menu |
| 2 | Show list website |

Table 127: Manage Website Step

#### Choose Template

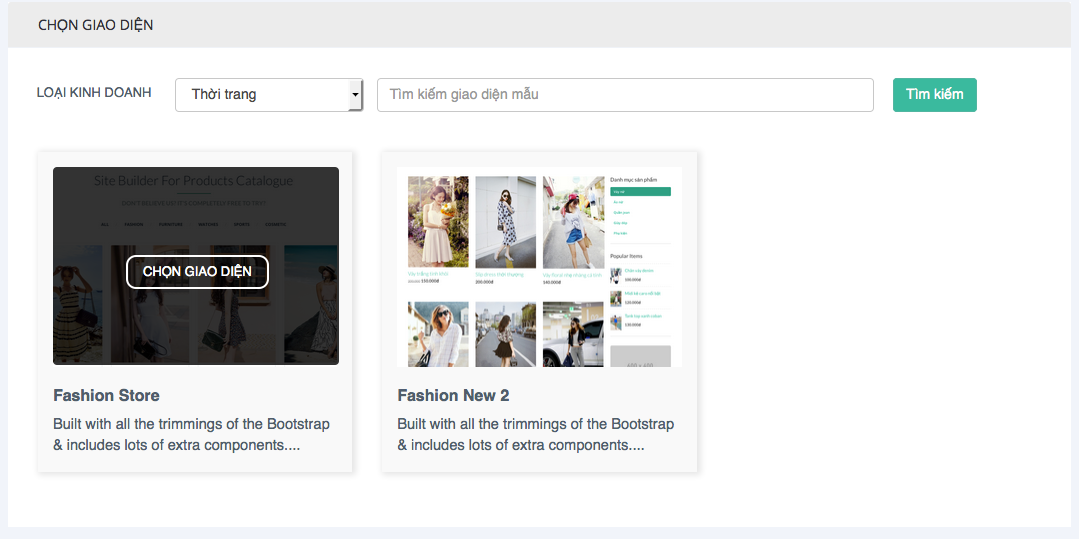


Figure 85: User- Choose Template

|  |  |
| --- | --- |
| **Step** | **Description** |
| 1 | Click “Tạo mới website” on manage website page |
| 2 | Show list template |
| 3 | Click “Chọn giao diện” button to choose template |

Table 128: Choose template Step

#### Create Website

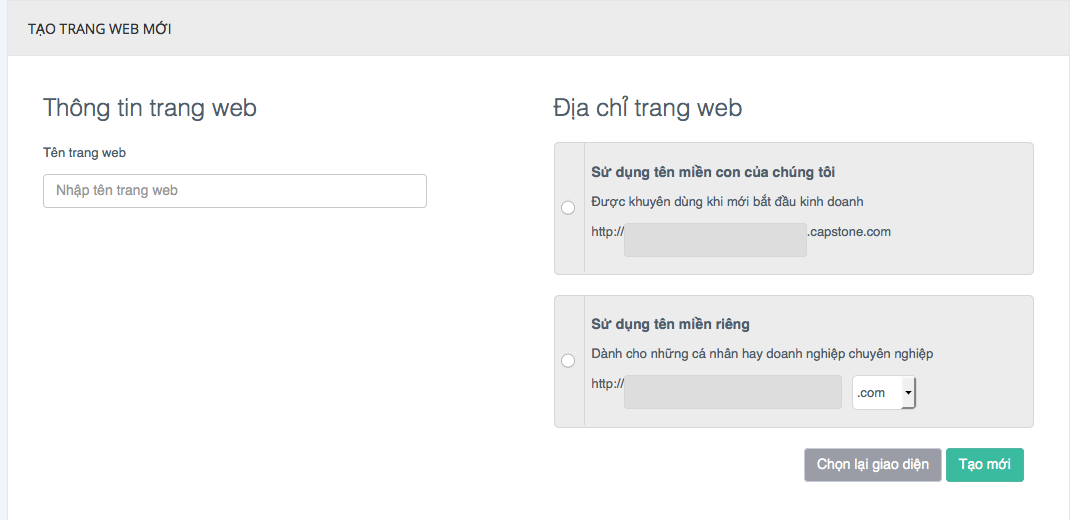


Figure 86: User – Create Website Page

|  |  |
| --- | --- |
| **Step** | **Description** |
| 1 | Click “Chọn giao diện” on choose template page |
| 2 | Fill all required input (website title, domain) |
| 3 | Click “Tạo mới” button to create new website |

Table 129: Create Website Step

#### Custom page

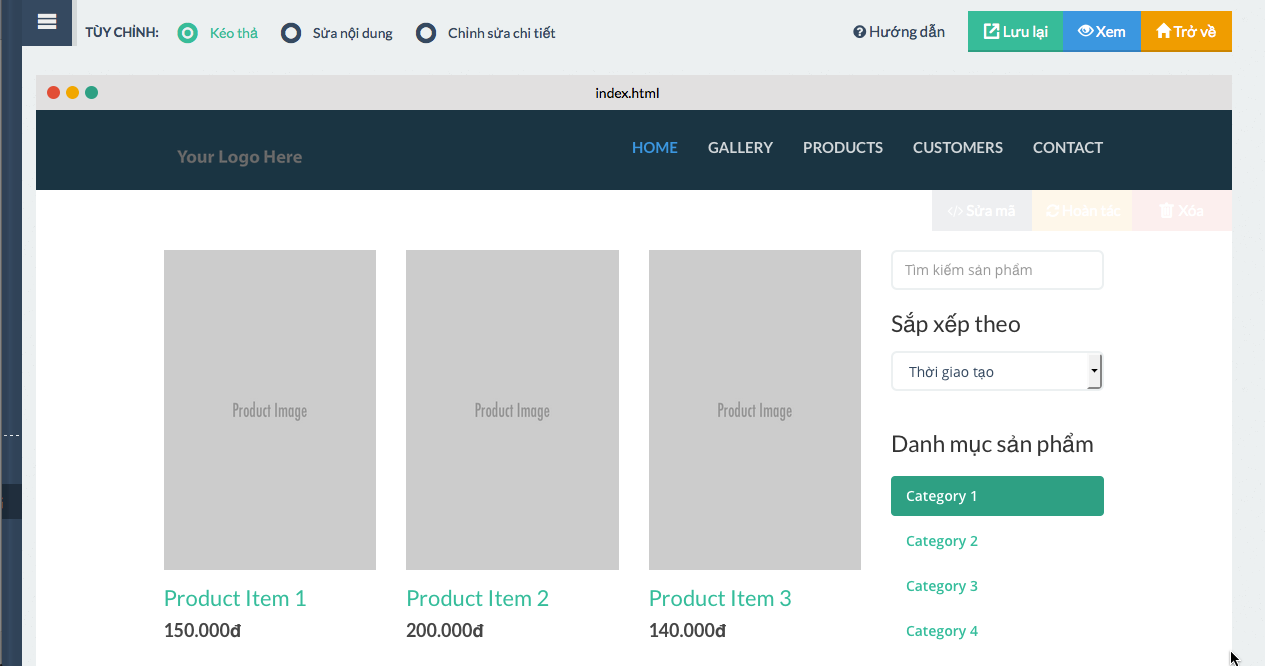


Figure 87: User – Custom Page

|  |  |
| --- | --- |
| **Step** | **Description** |
| 1 | Click “Tạo mới” on create website page |
| 2 | Show custom page with exist template |
| 3 | Drag and drop widgets on left menu to design website |
| 4 | Click “Lưu lại” to save design changes |

Table 130: Custom page Step

#### Website detail

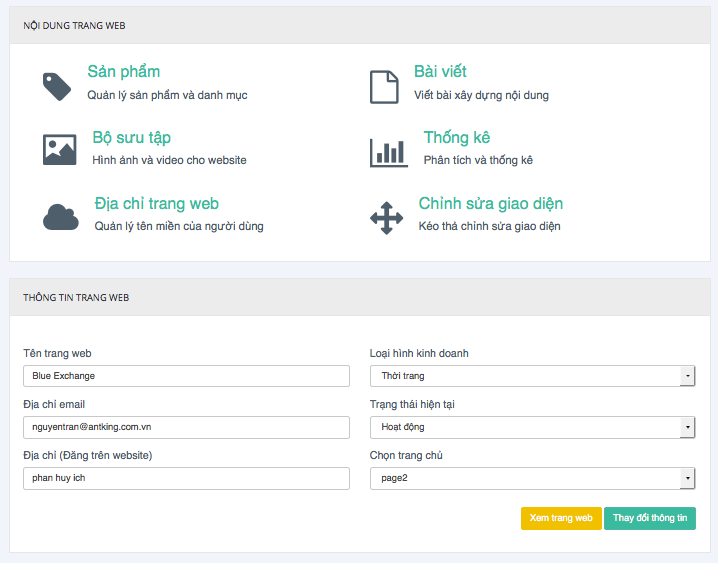


Figure 88: User – Website detail Page

|  |  |
| --- | --- |
| **Step** | **Description** |
| 1 | Click on website title on manage website page |
| 2 | Show website detail and links to build content pages |

Table 131: Website detail Step

#### Add New Product

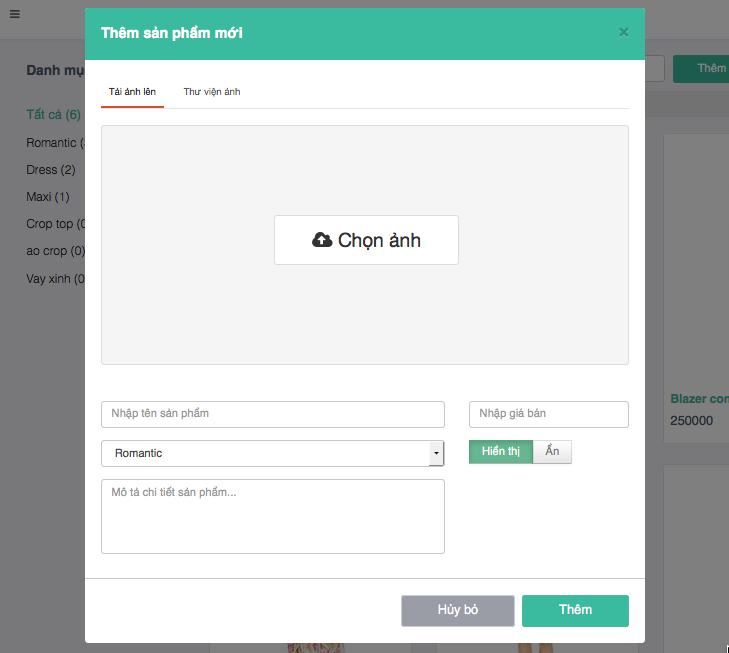


Figure 90: User – Add New Product Popup

|  |  |
| --- | --- |
| **Step** | **Description** |
| 1 | Click on “Thêm” on manage product page |
| 2 | Show “Thêm sản phẩm mới” popup |
| 3 | Choose image and fill all input required |
| 4 | Click “Thêm” button to save new product |

Table 133: Add Product Step

#### Manage Article

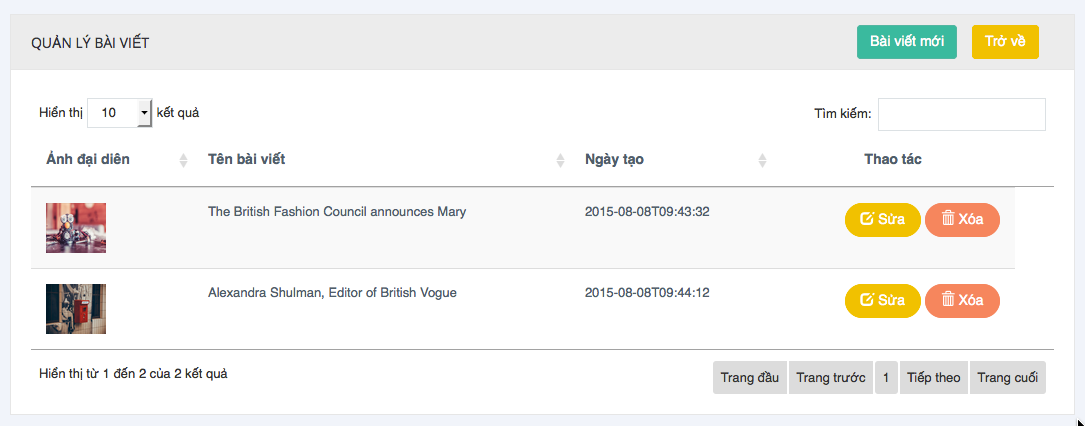


Figure 91: User – Manage Article Page

|  |  |
| --- | --- |
| **Step** | **Description** |
| 1 | Click on “Bài viết” on website detail page |
| 2 | Show article list |

Table 134: Manage Article Step

#### Create Article

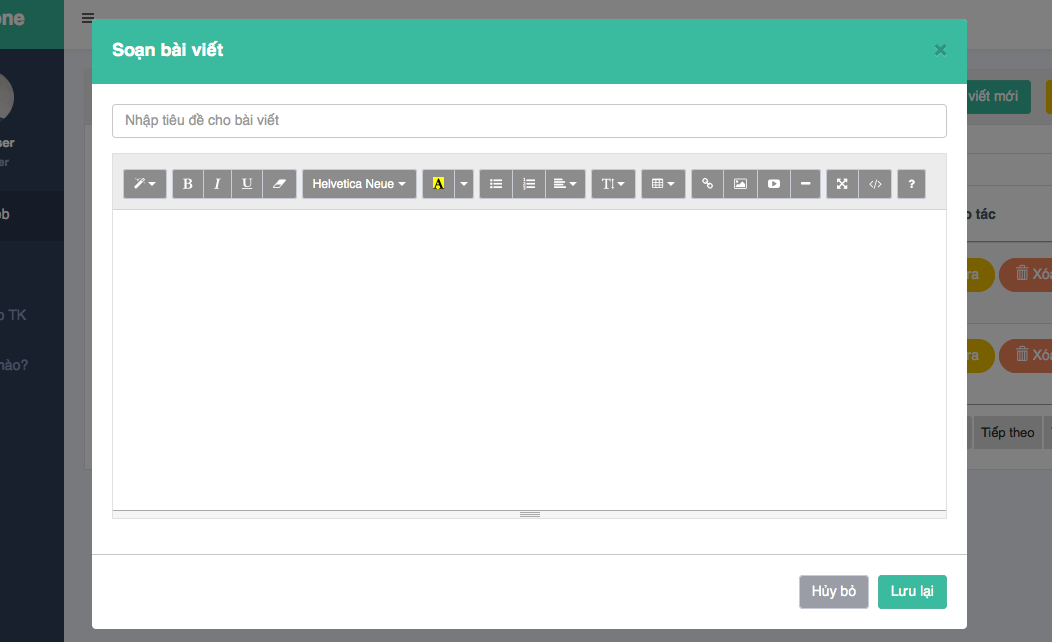


Figure 92: User - Create Article Popup

|  |  |
| --- | --- |
| **Step** | **Description** |
| 1 | Click on “Bài viết mới” on manage article page |
| 2 | Show “Soạn bài viết” popup |
| 3 | Enter article title and article content |
| 4 | Click “Lưu lại” to save new article |

Table 135: Create Article Step

#### View Statistic

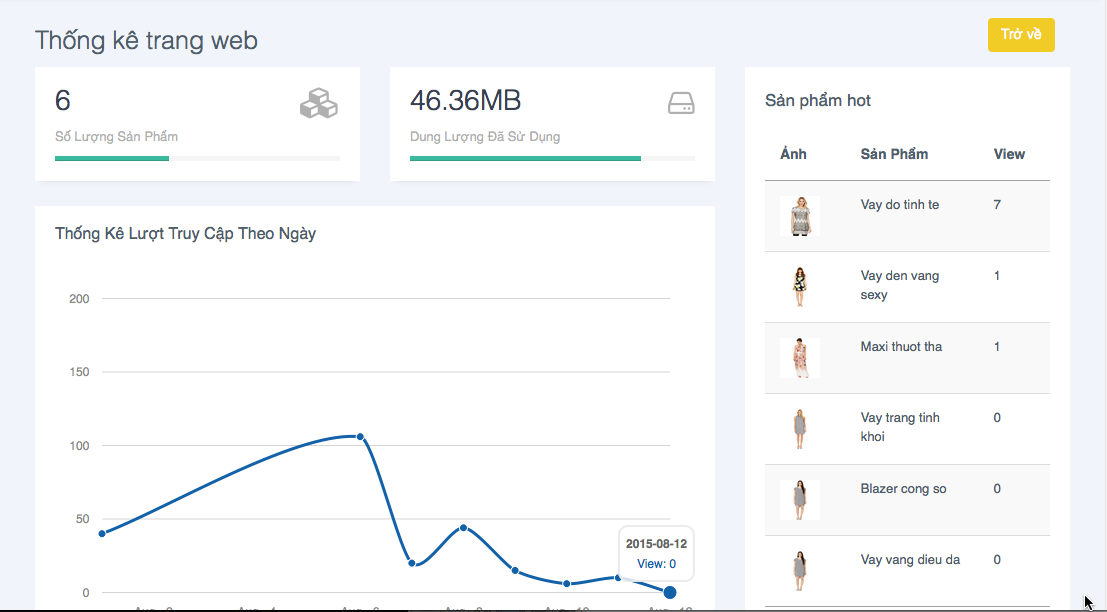


Figure 93: User – View Statistic Page

|  |  |
| --- | --- |
| **Step** | **Description** |
| 1 | Click on “Thống kê” on website detail page |
| 2 | Show statistic of website |

Table 136: View Statistic Step

### Admin

#### Manage Account Type and Permission

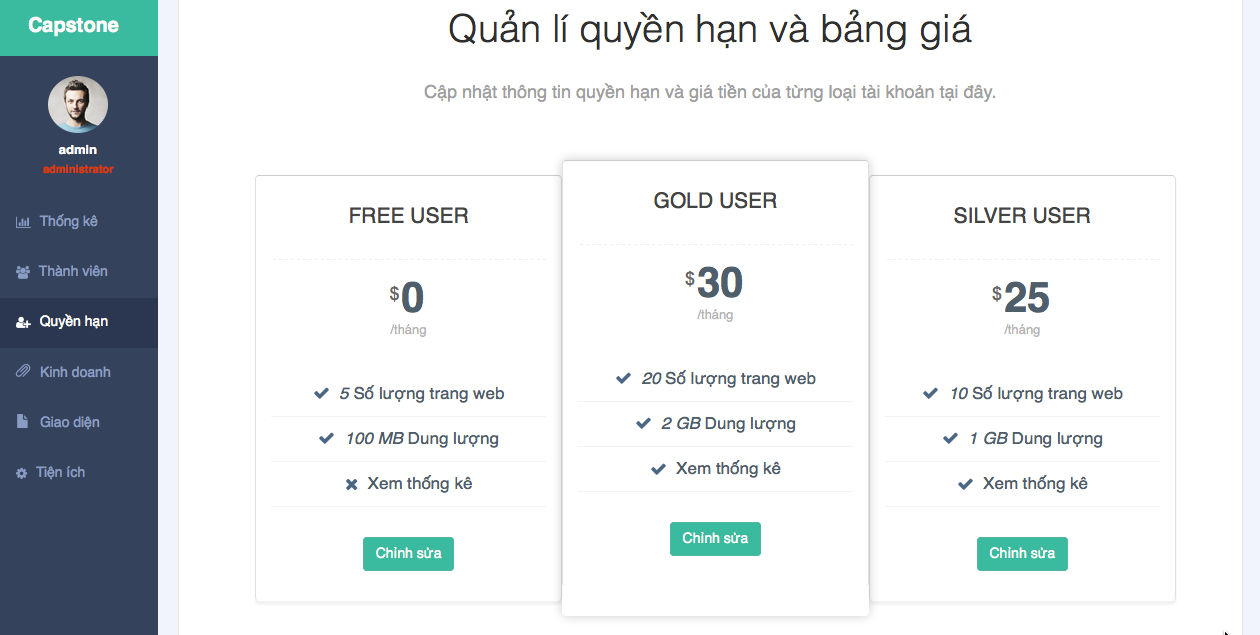


Figure 94: Admin – Manage Account Type and Permission Page

|  |  |
| --- | --- |
| **Step** | **Description** |
| 1 | Click on “Quyền hạn” on menu bar |
| 2 | Show 3 account types and permission for them |
| 3 | Click “Chỉnh sửa” button to show pop up edit account type information and permissions |
| 4 | Click “Lưu lại” to save |

Table 137: Manage Account Type and Permission Step

#### Manage Template

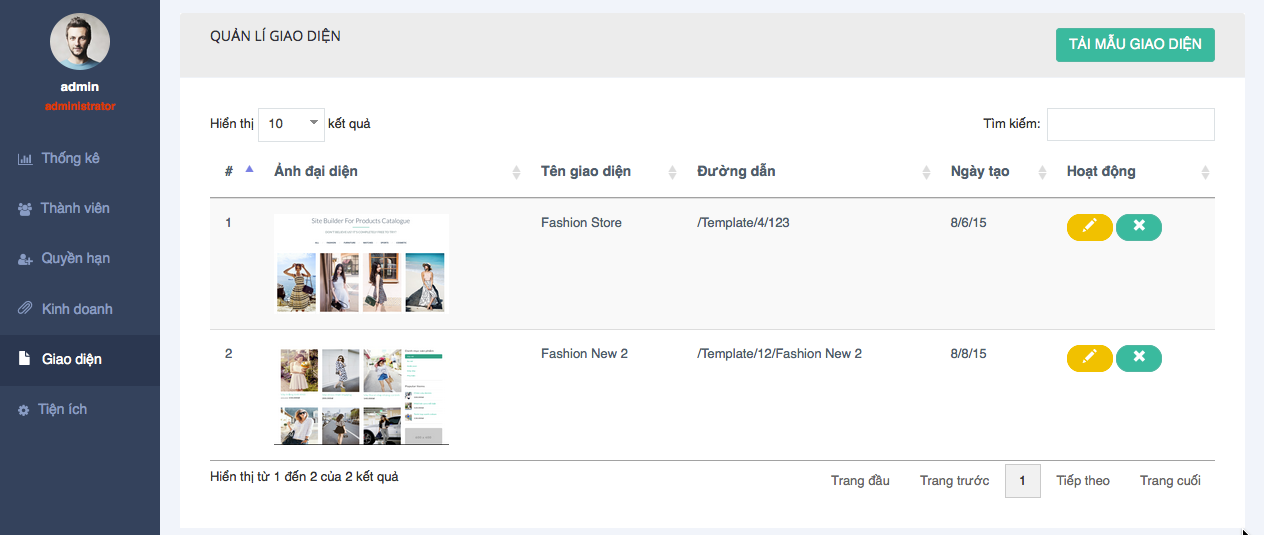


Figure 95: Admin – Manage Template Page

|  |  |
| --- | --- |
| **Step** | **Description** |
| 1 | Click on “Giao diện” on menu bar |
| 2 | Show template list |
| 3 | Click on edit button to edit template |
| 4 | Click on delete button to delete template |
| 5 | Click on “Tải mẫu giao diện” to upload new template |

Table 138: Manage Template Step

#### Manage Widget

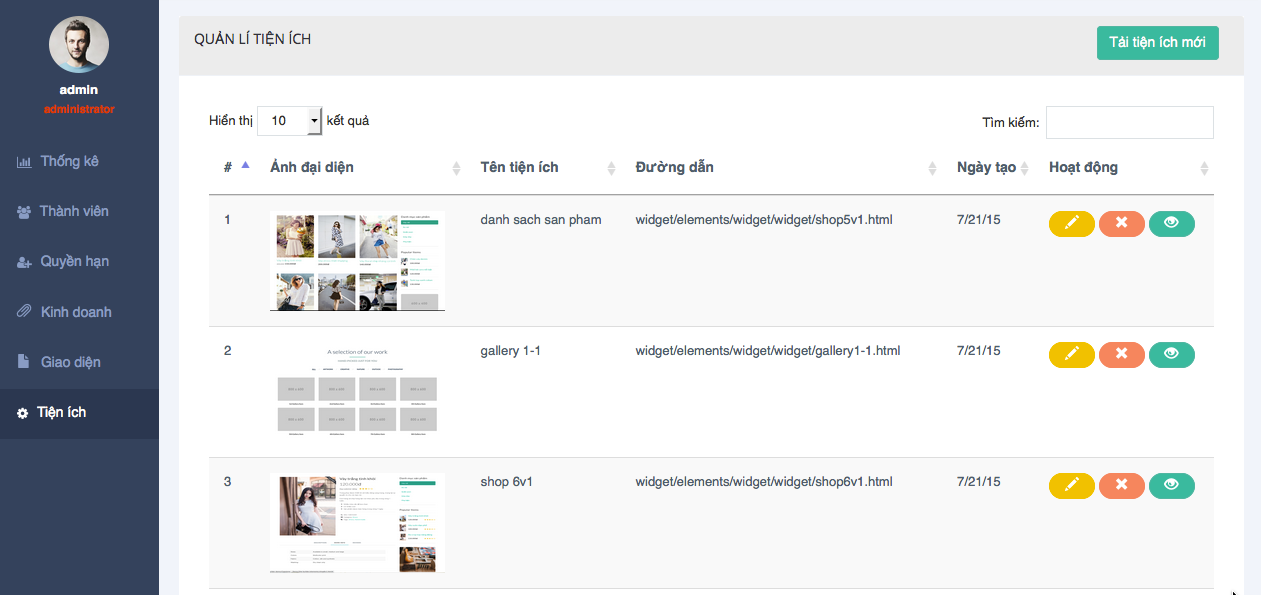


Figure 96: Admin – Manage Widget Page

|  |  |
| --- | --- |
| **Step** | **Description** |
| 1 | Click on “Tiện ích” on menu bar |
| 2 | Show widget list |
| 3 | Click on edit button to rdirect to edit widget page |
| 4 | Click on delete button to delete widget |
| 5 | Click on “Tải tiện ích mới” to upload new widget |

Table 139: Manage Widget Step

#### View statistic

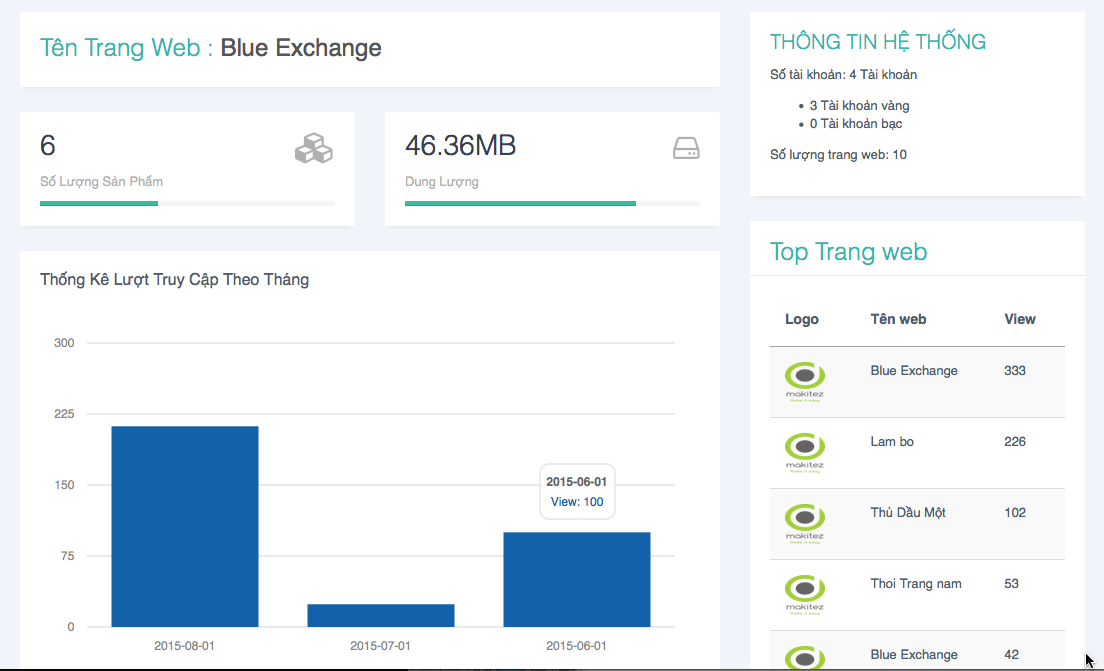


Figure 97: Admin – View Statistic Page

|  |  |
| --- | --- |
| **Step** | **Description** |
| 1 | Click on “Thống kê” on menu bar |
| 2 | Show statistic for system |
| 3 | Click on website logo in “Top trang web” panel to see statisctic for each website |

Table 140: View Statistic Step

# Appendix

1. SOFTWARE ENGINEERING Ninth Edition, by Ian Sommerville.
2. Learning UML 2.0by Kim Hamilton, Russell Miles, Publisher O'Reilly.
3. Scrum guide: <http://www.scrumguides.org/scrum-guide.html>