

Use-case Description

Van Lang Admissions

# Revision Table

|  |  |  |  |
| --- | --- | --- | --- |
| Author | Date | Reason for changes | Version |
|  |  |  |  |

Contents

[Revision Table 1](#_Toc473711027)

[I. INTRODUCTION 5](#_Toc473711028)

[1.1. Purpose 5](#_Toc473711029)

[1.2. Audience 5](#_Toc473711030)

[II. Use case Descriptions 5](#_Toc473711031)

[1. Manage accounts 6](#_Toc473711032)

[1.1. Log in 6](#_Toc473711033)

[1.2. Log out 8](#_Toc473711034)

[1.3. Create account 10](#_Toc473711035)

[1.4. Edit account 12](#_Toc473711036)

[1.5. Search account 14](#_Toc473711037)

[1.6. Forget password 15](#_Toc473711038)

[1.6. Forget password(cont.) 17](#_Toc473711039)

[1.7. View profile 19](#_Toc473711040)

[1.8. View list account 21](#_Toc473711041)

[1.9. Authorize 23](#_Toc473711042)

[1.10. Activate account 25](#_Toc473711043)

[1.11. Deactivate account 27](#_Toc473711044)

[2. Manage news 29](#_Toc473711045)

[2.1. View news 29](#_Toc473711046)

[2.2. Post news 31](#_Toc473711047)

[2.3. View draft 33](#_Toc473711048)

[2.4. Edit draft 35](#_Toc473711049)

[2.5. Delete draft 37](#_Toc473711050)

[2.6. Transfer 39](#_Toc473711051)

[2.7. Approve 41](#_Toc473711052)

[2.8. Deactivate 43](#_Toc473711053)

[2.9. Search News 45](#_Toc473711054)

[2.10. Sort News 47](#_Toc473711055)

[2.11. Push News 49](#_Toc473711056)

[2.12. Create news 51](#_Toc473711057)

[2.13. Share news 53](#_Toc473711058)

[III. Manage examination 55](#_Toc473711059)

[3.1. View individual point 55](#_Toc473711060)

[3.2. Register exam 57](#_Toc473711061)

[3.3. View profile 59](#_Toc473711062)

[3.4. Edit profile 61](#_Toc473711063)

[3.5. Import point 63](#_Toc473711064)

[3.6. Export profile 65](#_Toc473711065)

[3.7. Create exam 67](#_Toc473711066)

[IV. Manage categories 69](#_Toc473711067)

[4.1. View categories 69](#_Toc473711068)

[4.2. View category 71](#_Toc473711069)

[4.3. Add category 73](#_Toc473711070)

[4.4. Edit category 75](#_Toc473711071)

[4.5. Delete category 77](#_Toc473711072)

[5. Manage languages 79](#_Toc473711073)

[5.1. Change language 79](#_Toc473711074)

[6. Manage banner 81](#_Toc473711075)

[6.1. Create banner 81](#_Toc473711076)

[6.2. View banner 83](#_Toc473711077)

[6.3. Show hide banner 84](#_Toc473711078)

[6.4. Edit banner 86](#_Toc473711079)

[6.5. Delete banner 88](#_Toc473711080)

[6.6. Arrange images 90](#_Toc473711081)

[7. Manage questions 92](#_Toc473711082)

[7.1. Send questions 92](#_Toc473711083)

[7.2. Answer questions 94](#_Toc473711084)

[7.3. View questions 96](#_Toc473711085)

[7.4. Approve questions 98](#_Toc473711086)

[7.5. Search questions 100](#_Toc473711087)

[8. Manage pop-ups 102](#_Toc473711088)

[8.1. Create pop-ups 102](#_Toc473711089)

[8.2. View pop-ups 104](#_Toc473711090)

[8.3. Show-hide pop-ups 106](#_Toc473711091)

[8.4. Edit pop-ups 108](#_Toc473711092)

[8.5. Delete pop-ups 110](#_Toc473711093)

[8.6. Arrange pop-ups 112](#_Toc473711094)

[9. Manage homepage 114](#_Toc473711095)

[9.1. Add 114](#_Toc473711096)

[9.2. Edit homepage 116](#_Toc473711097)

[9.3. Delete homepage 118](#_Toc473711098)

[10. Use case Diagram 120](#_Toc473711099)

# INTRODUCTION

## Purpose

* This document is description about vision and scope of Van Lang Admissions project. It shows the vision what BSS – Team wants to toward and the scope of project for BSS – Team doesn’t exceed the limit of project.

## Audience

* The main audiences of this document are: Mentor, Base Steps Solution Team and customer.

# Use case Descriptions

## Manage accounts

### Log in

|  |  |  |  |
| --- | --- | --- | --- |
| UC-MA-01 Log in | | | |
| Creator: | Hien Nguyen | Person Update: |  |
| Creation date: | 4/12/2016 | Date Update: |  |
| Description: | Admin, Editor, Education staff, System admin log in system | | |
| Users: | Admin, Editor, Education staff, System admin | | |
| Trigger: | Admin, Editor, Education staff, System admin click button Login. | | |
| Pre-Conditions: | Admin, Editor, Education staff, System admin must have an account in system. | | |
| Post-Conditions: | Success: Admin, Editor, Education staff, System admin log-in system successfully. | | |
| Basic Flow: | 1. System open login form. 2. Admin, Editor, Education staff, System admin input account and password. 3. User click button “Login”. 4. System compare data with database. 5. System open “Home page”. 6. End of use-case | | |
| Alternate Flow: | AF1: Admin, Editor, Education staff, System admin cancel login: (after step 2)   1. Admin, Editor, Education staff, System admin click button “Cancel”. 2. System close login form. | | |
| Exception Flows: | EF1: Admin, Editor, Education staff, System admin input wrong account or password: (after step 4)   1. System notify “Incorrect account or password”. 2. System back to login form.   EF2: System can’t connect Database: (after step 3)   1. System notify “Can’t connect to database, please try again”. 2. System move to 404 page | | |
| Priority | High | | |
| Frequency of use | Usually | | |
| Business Rules | 1. Input fields have a limit under 100 characters. 2. Password must have censor. | | |
| Mock up |  | | |

### Log out

|  |  |  |  |
| --- | --- | --- | --- |
| UC-MA-02 Log out | | | |
| Creator: | Hien Nguyen | Person Update: |  |
| Creation date: | 4/12/2015 | Date Update: |  |
| Description: | Admin, Editor, Education staff, System admin log out system. | | |
| Users: | Admin, Editor, Education staff, System admin | | |
| Trigger: | Admin, Editor, Education staff, System admin click button “Log out” | | |
| Pre-Conditions: | Admin, Editor, Education staff, System admin logged in system. | | |
| Post-Conditions: | Success: Admin, Editor, Education staff, System admin log-out system successfully. | | |
| Basic Flow: | 1. System alert dialog “Ok | Cancel” 2. User click “Ok” 3. System show message “Log out successfully”. 4. End of use case. | | |
| Alternate Flow: | AF1: Admin, Editor, Education staff, System admin cancel log out: (after step 1)   1. User click button “Cancel”. 2. System close log out alert dialog | | |
| Exception Flows: | EF1: System can’t connect Database: (after step 2)   1. System notify “Can’t connect to database”. 2. System move to 404 page. | | |
| Priority | High | | |
| Frequency of use | Usually | | |
| Business Rules |  | | |
| Mock up |  | | |

### Create account

|  |  |  |  |
| --- | --- | --- | --- |
| UC-MA-03 Create Account | | | |
| Creator: | Hien Nguyen | Person Update: |  |
| Creation date: | 4/12/2016 | Date Update: |  |
| Description: | System admin creates a new account for somebody | | |
| Users: | System admin | | |
| Trigger: | System admin click button “Create Account” | | |
| Pre-Conditions: | System admin must have account’s admin | | |
| Post-Conditions: | Success: System admin creates a new account successfully. | | |
| Basic Flow: | 1. System show “create new account form”. 2. System admin inputs information of new account. 3. System admin clicks button “Create” 4. System check input data. 5. System add new account into database 6. System notify “Create a new account successfully” 7. End of use case. | | |
| Alternate Flow: | AF1: System admin cancel create new account: (after step 2)  3. System admin clicks button “Cancel”.  4.System close create new account form. | | |
| Exception Flows: | EF1: System admin input invalid information: (after step 4)   1. System notify “Invalid data. Please check again”. 2. System focus into invalid field.   EF2: System can’t connect Database: (after step 3)  4.System notify “Can’t connect to database”.  5.System move to 404 page | | |
| Priority | High | | |
| Frequency of use | Often | | |
| Business Rules | 1. Input fields have a limit under 100 characters. 2. Password must have censor. | | |
| Mock up |  | | |

### Edit account

|  |  |  |  |
| --- | --- | --- | --- |
| UC-MA-04 Edit Account | | | |
| Creator: | Hien Nguyen | Person Update: |  |
| Creation date: | 4/12/2016 | Date Update: |  |
| Description: | Admin, Editor, Education staff edit information of their account | | |
| Users: | Admin, Editor, Education staff | | |
| Trigger: | Admin, Editor, Education staff click button “Edit” | | |
| Pre-Conditions: | Admin, Editor, Education staff must have an account in system.  The system must be connected to the internet | | |
| Post-Conditions: | Success: Admin, Editor, Education staff edit their account successfully. | | |
| Basic Flow: | 1. System show “Edit account form”. 2. Admin, Editor, Education staff edit information. 3. Admin, Editor, Education staff click button “Save” 4. System check input data. 5. System update new information into database. 6. System notify “Edit successfully” 7. End of use case. | | |
| Alternate Flow: | AF1: User cancel edit account: (after step 2)   1. User click button “Cancel”. 2. System close “edit account form”. | | |
| Exception Flows: | EF1: Admin, Editor, Education staff input invalid information: (after step 4)   1. System notify “Invalid data. Please check again”. 2. System focus to invalid field   EF2: System can’t connect Database: (after step 5)  6.The system notifies “Can’t connect the database, please try again”.  7.System move to 404 page | | |
| Priority | High | | |
| Frequency of use | Sometimes | | |
| Business Rules | 1. Input fields have a limit under 100 characters. 2. Password must have censor. | | |
| Mock up |  | | |

### Search account

|  |  |  |  |
| --- | --- | --- | --- |
| UC-MA-05 Search Account | | | |
| Creator: | Hien Nguyen | Person Update: |  |
| Creation date: | 4/12/2016 | Date Update: |  |
| Description: | System admin search account | | |
| Users: | System admin | | |
| Trigger: |  | | |
| Pre-Conditions: | System admin must in “Manage account page” | | |
| Post-Conditions: | Success: System admin search account successfully. | | |
| Basic Flow: | 1. System admin inputs key search in search field. 2. System admin clicks button “Search” 3. System show accounts coincidence with key search. 4. End of use case. | | |
| Alternate Flow: |  | | |
| Exception Flows: | EF1: System can’t connect Database: (after step 2)   1. The system notifies “Can’t connect the database, please try again”. 2. System move to 404 page | | |
| Priority | High | | |
| Frequency of use | Often | | |
| Business Rules | Key search has a limit under 100 characters. | | |
| Mock up |  | | |

### Forget password

|  |  |  |  |
| --- | --- | --- | --- |
| UC-MA-06.1 Forget Password (Send link create new password) | | | |
| Creator: | Hien Nguyen | Person Update: |  |
| Creation date: | 4/12/2016 | Date Update: |  |
| Description: | Admin, Editor, Education staff forget password | | |
| Users: | Admin, Editor, Education staff. | | |
| Trigger: | Admin, Editor, Education staff. click button “Forget password” | | |
| Pre-Conditions: | Admin, Editor, Education staff must have an account in system.  The system must be connected to the internet | | |
| Post-Conditions: | Success: Admin, Editor, Education staff get link create new password successfully | | |
| Basic Flow: | 1. System show “Forget password form”. 2. User input their email. 3. User click button “Ok” 4. System check email. 5. System send link to their email. 6. System notify “Link to change password has been sent into your email. Please check your email” 7. User check their email to get password. 8. End of use case. | | |
| Alternate Flow: | AF1: Admin, Editor, Education staff cancel “forget password” (after step 2):   1. Admin, Editor, Education staff click button “Cancel”. 2. System close “Forget password form”. | | |
| Exception Flows: | EF1: Admin, Editor, Education staff input wrong email: (after step 4)   1. System notify “Invalid data. Please check again”. 2. System back to “Forget password form”.   EF2: System can’t connect database: (after step 4)  5.The system notifies “Can’t connect the database, please try again”.  6. System move to 404 page. | | |
| Priority | Medium | | |
| Frequency of use | Sometimes | | |
| Business Rules | 1. Input fields have a limit under 100 characters. | | |
| Mock up |  | | |

### Forget password(cont.)

|  |  |  |  |
| --- | --- | --- | --- |
| UC-MA-06.2 Forget Password (Create new password) | | | |
| Creator: | Hien Nguyen | Person Update: |  |
| Creation date: | 4/12/2016 | Date Update: |  |
| Description: | Admin, Editor, Education staff get new password | | |
| Users: | Admin, Editor, Education staff. | | |
| Trigger: | Admin, Editor, Education staff click link “Create new password” | | |
| Pre-Conditions: | Admin, Editor, Education staff must have an account in system.  Admin, Editor, Education staff must have link create new password in email | | |
| Post-Conditions: | Success: Admin, Editor, Education staff get new password successfully. | | |
| Basic Flow: | 1. System call mail server to change direction   1. Admin, Editor, Education staff login to their account 2. Admin, Editor, Education staff choose email of system 3. Admin, Editor, Education staff click link create new password 4. Mail server change direction to system 5. System show form “Create new password” 6. Admin, Editor, Education staff input new password 7. Admin, Editor, Education staff click button “Save” 8. System check input 9. System update information into database 10. System notify “Change password successfully” 11. End of use case | | |
| Alternate Flow: |  | | |
| Exception Flows: | EF1: Admin, Editor, Education staff input wrong password type: (after step 8)   1. System notify “Invalid password. Please check again”. 2. System focus to invalid field   EF2: System can’t connect database: (after step 10)  11.The system notifies “Can’t connect the database, please try again”.  12. System move to 404 page. | | |
| Priority | Medium | | |
| Frequency of use | Sometimes | | |
| Business Rules | 1. Input fields have a limit under 100 characters. | | |
| Mock up |  | | |

### View profile

|  |  |  |  |
| --- | --- | --- | --- |
| UC-MA-07 View Profile of Account | | | |
| Creator: | Hien Nguyen | Person Update: |  |
| Creation date: | 4/12/2016 | Date Update: |  |
| Description: | Admin, Editor, Education staff view the detail of their account | | |
| Users: | Admin, Editor, Education staff | | |
| Trigger: | Admin, Editor, Education staff click button “View profile” | | |
| Pre-Conditions: | Admin, Editor, Education staff must have an account in system.  Admin, Editor, Education staff must login into website  The system must be connected to the internet | | |
| Post-Conditions: | Success: Admin, Editor, Education staff view profile account successfully. | | |
| Basic Flow: | 1. System show profile account page. 2. End of use case | | |
| Alternate Flow: | N/A | | |
| Exception Flows: | EF1: System can’t connect database: (after step 1)   1. The system notifies “Can’t connect the database, please try again” | | |
| Priority | High | | |
| Frequency of use | Often | | |
| Business Rules |  | | |
| Mock up |  | | |

### View list account

|  |  |  |  |
| --- | --- | --- | --- |
| UC-MA-08 View List Account | | | |
| Creator: | Hien Nguyen | Person Update: |  |
| Creation date: | 4/12/2016 | Date Update: |  |
| Description: | System admin views list of account | | |
| Users: | System admin | | |
| Trigger: | System admin clicks button “Mange account” | | |
| Pre-Conditions: | System admin log in website.  The system must be connected to the internet  The system must have account in database. | | |
| Post-Conditions: | Success: System admin view accounts list successfully. | | |
| Basic Flow: | 1. System show list of account 2. End of use case | | |
| Alternate Flow: | N/A | | |
| Exception Flows: | EF1: System can’t connect database: (after step 1)   1. The system notifies “Can’t connect the database, please try again” 2. System move to 404 page | | |
| Priority | High | | |
| Frequency of use | Usually | | |
| Business Rules |  | | |
| Mock up |  | | |

### Authorize

|  |  |  |  |
| --- | --- | --- | --- |
| UC-MA-09 Authorize | | | |
| Creator: | Hien Nguyen | Person Update: |  |
| Creation date: | 4/12/2016 | Date Update: |  |
| Description: | System admin authorizes for account | | |
| Users: | System admin | | |
| Trigger: | System admin click button “Authorize” | | |
| Pre-Conditions: | System admin login to website  System admin must login to system  The system must have account into the internet | | |
| Post-Conditions: | Success: System admin authorizes for accounts successfully. | | |
| Basic Flow: | 1. System show Authorize form. 2. System admin choose account 3. System admin choose role for account 4. System admin clicks button “Save”. 5. System show alert dialog “Ok | Cancel” 6. System admin clicks “Ok” 7. System modify account in system. 8. The system notifies: “Authorize accounts successfully” 9. End of use case. | | |
| Alternate Flow: | AF1: System admin cancels “authorize”: (after step 3)   1. System admin clicks button “Cancel”. 2. System close “authorize form”. | | |
| Exception Flows: | EF1: System can’t connect Database: (after step 6)  7.The system notifies “Can’t connect the database, please try again”  8.System move to 404 page | | |
| Priority | High | | |
| Frequency of use | Sometimes | | |
| Business Rules |  | | |
| Mock up |  | | |

### Activate account

|  |  |  |  |
| --- | --- | --- | --- |
| UC-MA-10 Activate Account | | | |
| Creator: | Minh Doan | Person Update: |  |
| Creation date: | 16/1/2017 | Date Update: |  |
| Description: | System Admin activate the deactivated account | | |
| Users: | System Admin | | |
| Trigger: | System Admin choose activate account | | |
| Pre-Conditions: | User must have an account of System Admin and being in the “List of Account” GUI. | | |
| Post-Conditions: | Success: Account is activated. | | |
| Basic Flow: | 1. System Admin chooses account. 2. System Admin clicks “Edit Account”. 3. System shows information of account. 4. System Admin clicks “Activate Account”. 5. System notifies: “Activate Account?” 6. System Admin clicks “OK” button. 7. System shows “List of Account” GUI. 8. End of use-case | | |
| Alternate Flow: | AF1: System Admin clicks “cancel” in the notify GUI: (after step 5)   1. User click button “Cancel”. 2. System shows information of account GUI. | | |
| Exception Flows: |  | | |
| Priority | Medium | | |
| Frequency of use |  | | |
| Business Rules |  | | |
| Mock up |  | | |

### Deactivate account

|  |  |  |  |
| --- | --- | --- | --- |
| UC-MA-11 Deactivate Account | | | |
| Creator: | Minh Doan | Person Update: |  |
| Creation date: | 16/1/2017 | Date Update: |  |
| Description: | System Admin deactivate the existing account | | |
| Users: | System Admin | | |
| Trigger: | System Admin choose deactivate account | | |
| Pre-Conditions: | * User must have an account of System Admin and being in the “List of Account” GUI. * Account is activating. | | |
| Post-Conditions: | Success: Account is deactivated. | | |
| Basic Flow: | 1. System Admin chooses account. 2. System Admin clicks “Edit Account”. 3. System shows information of account. 4. System Admin clicks “Deactivate Account”. 5. System notifies: “Activate Account?” 6. System Admin clicks “OK” button. 7. System shows “List of Account” GUI. 8. End of use-case | | |
| Alternate Flow: | AF1: System Admin clicks “cancel” in the notify GUI: (after step 5)   1. User click button “Cancel”. 2. System shows information of account GUI. | | |
| Exception Flows: |  | | |
| Priority | Medium | | |
| Frequency of use |  | | |
| Business Rules |  | | |
| Mock up |  | | |

## Manage news

### View news

|  |  |  |  |
| --- | --- | --- | --- |
| Use case ID | UC-MN-01 | | |
| Name of use case | View News | | |
| Creator | Tai Nguyen | Person updated |  |
| Creation date | 4/12/2016 | Date updated |  |
| Primary actor | Admin, Editor, Education staff, reader | | |
| Brief description | Admin, Editor, Education staff, reader view news in the website | | |
| Trigger | Admin, Editor, Education staff, reader access website | | |
| Pre-conditions | The system must be connected to the internet. | | |
| Main success flow | 1. Admin, Editor, Education staff, reader choose news which need view 2. System get data from database 3. System show detail of that news 4. End of use case | | |
| Alternative flows |  | | |
| Exception flows | EF1: System can’t connect to database (after step 2)  3.The system notifies “Can’t connect to the database, please try again”  4.System move to 404 page | | |
| Post-conditions | Successful: user can view a news. | | |
| Priority | High | | |
| Relationships |  | | |
| Frequency of use |  | | |
| Business rules |  | | |
| Special requirement |  | | |

### Post news

|  |  |  |  |
| --- | --- | --- | --- |
| Use case ID | UC-MN-02 | | |
| Name of use case | Post News | | |
| Creator | Tai Nguyen | Person updated |  |
| Creation date | 4/12/2016 | Date updated |  |
| Primary actor | Admin | | |
| Brief description | Admin write news and post | | |
| Pre-conditions | Admin must access website  Admin must login  The system must be connected to the internet | | |
| Main success flow | 1. Admin clicks button “Post news” 2. System shows post news GUI 3. Admin choose news that approved 4. Admin clicks button “Post” 5. System save data into database 6. System notify “Post successfully” 7. End of use-case | | |
| Sub flows |  | | |
| Alternative flows | AF1: Admin click button “Cancel” (after step 3)   1. System save to draft 2. End of use case | | |
| Exception flows | EF1: Admin choose news that didn’t approve (after step 2)  3.The system notifies: “Please choose news that approved”  4. System back to current page.  EF2: System can’t connect database (after step 5)  6.The system notifies “Can’t connect the database, please check connection”  7.System move to 404 page | | |
| Trigger |  | | |
| Relationships |  | | |
| Post-conditions | Successful: Admin post a news successfully | | |
| Priority | High | | |
| Frequency of use | Usually | | |
| Business rules |  | | |
| Special requirement |  | | |

### View draft

|  |  |  |  |
| --- | --- | --- | --- |
| Use case ID | UC-MN-03 | | |
| Name of use case | View Draft | | |
| Creator | Tai Nguyen | Person updated |  |
| Creation date | 4/12/2016 | Date updated |  |
| Primary actor | Admin, Editor, Education staff | | |
| Brief description | Admin, Editor, Education staff view draft of the post | | |
| Pre-conditions | Admin, Editor, Education staff must access website  Admin, Editor, Education staff must login  The system must be connected to the internet | | |
| Main success flow | 1. Admin, Editor, Education staff choose draft which need view 2. System show detail of that draft 3. End of use case | | |
| Sub flows |  | | |
| Alternative flows | N/A | | |
| Exception flows | EF1: System can’t connect to database (after step 1)  2. The system notifies “Can’t connect to the database, please try again”  3. System move to page 404 | | |
| Trigger | Admin, Editor, Education staff must access website | | |
| Relationships |  | | |
| Post-conditions | Successful: Admin, Editor, Education staff can view a draft successfully. | | |
| Priority | High | | |
| Frequency of use | Sometimes | | |
| Business rules |  | | |
| Special requirement |  | | |

### Edit draft

|  |  |  |  |
| --- | --- | --- | --- |
| Use case ID | UC-MN-04 | | |
| Name of use case | Edit draft | | |
| Creator | Tai Nguyen | Person updated |  |
| Creation date | 4/12/2016 | Date updated |  |
| Primary actor | Admin, Editor, Education staff | | |
| Brief description | Actors edit draft of the post | | |
| Pre-conditions | Admin, Editor, Education staff must access website  Admin, Editor, Education staff must login  The system must be connected to the internet | | |
| Main success flow | 1. Admin, Editor, Education staff choose draft that to view 2. System shows edit draft GUI 3. Admin, Editor, Education staff edit information. 4. Admin, Editor, Education staff click button “Save” 5. System update information into database 6. System notify “Edit successfully” 7. End of use case. | | |
| Sub flows |  | | |
| Alternative flows | AF1: Admin, Editor, Education staff click button “Cancel” (after step 3)  4. System save current draft  5. System back to previous page | | |
| Exception flows | EF1: System can’t connect database (after step 5)  6. The system notifies “Can’t connect to the database, please try again!”  7. System move to 404 page | | |
| Trigger |  | | |
| Relationships |  | | |
| Post-conditions | Successful: Admin, Editor, Education staff edit a draft successfully | | |
| Priority | High | | |
| Frequency of use |  | | |
| Business rules |  | | |
| Special requirement |  | | |

### Delete draft

|  |  |  |  |
| --- | --- | --- | --- |
| Use case ID | UC-MN-05 | | |
| Name of use case | Delete Draft | | |
| Creator | Tai Nguyen | Person updated |  |
| Creation date | 4/12/2016 | Date updated |  |
| Primary actor | Admin, Editor, Education staff | | |
| Brief description | Actors delete draft of the post | | |
| Pre-conditions | Admin, Editor, Education staff must access website  Admin, Editor, Education staff must login  The system must be connected to the internet | | |
| Main success flow | 1. Admin, Editor, Education staff choose draft which need delete 2. Admin, Editor, Education staff click button “Delete” 3. System show alert dialog “Ok | Cancel” 4. Admin, Editor, Education staff click “Ok” 5. System delete information of draft 6. System update information into database 7. System notify “Delete success” 8. End of use case. | | |
| Sub flows |  | | |
| Alternative flows | AF1: Admin, Editor, Education staff click choose “Cancel” (after step 2)  3. System back to current page | | |
| Exception flows | EF1: System can’t connect to database (after step 5)  6. The system notifies:” Can’t connect to the database, please try again!”  7. System move to 404 page | | |
| Trigger |  | | |
| Relationships |  | | |
| Post-conditions | Successful: Admin, Editor, Education staff delete a draft successful | | |
| Priority | High | | |
| Frequency of use |  | | |
| Business rules |  | | |
| Special requirement |  | | |

### Transfer

|  |  |  |  |
| --- | --- | --- | --- |
| Use case ID | UC-MN-06 | | |
| Name of use case | Transfer Draft | | |
| Creator | Tai Nguyen | Person updated |  |
| Creation date | 4/12/2016 | Date updated |  |
| Primary actor | Admin, Editor, Education staff | | |
| Brief description | Admin, Editor, Education staff transfer the news to approve/edit | | |
| Pre-conditions | Admin, Editor, Education staff must access website  Admin, Editor, Education staff must login  The system must be connected to the internet | | |
| Main success flow | 1. Admin, Editor, Education staff choose a post which need transfer. 2. Admin, Editor, Education staff click button “Transfer” 3. System show a list member. 4. Admin, Editor, Education staff choose somebody which need transfer. 5. Admin, Editor, Education staff click button “Verify” 6. System transfer that post to people is chosen. 7. End of use case | | |
| Sub flows |  | | |
| Alternative flows |  | | |
| Exception flows | EF1: Admin, Editor, Education staff don’t choose a member (after step 3)  4. System show alert: “Please choose a member”.  5. System back to current page | | |
| Trigger |  | | |
| Relationships |  | | |
| Post-conditions | Successful: Admin, Editor, Education staff can transfer a draft. | | |
| Priority | High | | |
| Frequency of use |  | | |
| Business rules |  | | |
| Special requirement |  | | |

### Approve

|  |  |  |  |
| --- | --- | --- | --- |
| Use case ID | UC-MN-07 | | |
| Name of use case | Approve draft | | |
| Creator | Tai Nguyen | Person updated |  |
| Creation date | 4/12/2016 | Date updated |  |
| Primary actor | Admin | | |
| Brief description | Admin approve the post from who transfer | | |
| Pre-conditions | Admin must access website  Admin must login  Admin must receive message  System must be connected to the internet | | |
| Main success flow | 1. System send for admin a message: “You have a post need to approve”  2. Admin choose message.  3. System show approve GUI.  4. Admin check that post  5. Admin click button “Approve”  6. System change status of draft into database.  7.End of use case | | |
| Sub flows |  | | |
| Alternative flows |  | | |
| Exception flows | EF1: System can’t connect to database (after step 6)  7. The system notifies: “System can’t connect to the database, please try again!”.  8. System move to 404 page | | |
| Trigger |  | | |
| Relationships |  | | |
| Post-conditions | Successful: Admin can approve a draft. | | |
| Priority | High | | |
| Frequency of use |  | | |
| Business rules |  | | |
| Special requirement |  | | |

### Deactivate

|  |  |  |  |
| --- | --- | --- | --- |
| Use case ID | UC-MN-08 | | |
| Name of use case | Deactivate draft | | |
| Creator | Tai Nguyen | Person updated |  |
| Creation date | 4/12/2016 | Date updated |  |
| Primary actor | Admin | | |
| Brief description | Admin deactivate news that posted to website | | |
| Pre-conditions | Admin must access website  Admin must login  System must have news  The system must be connected to the internet | | |
| Main success flow | 1. Admin choose a post which need deactivate. 2. Admin click button “Deactivate” 3. System update data into database. 4. The system notifies:” Deactivate successfully” | | |
| Sub flows |  | | |
| Alternative flows | AF1: Admin click “Cancel” (after step 1)  2. System back to current page | | |
| Exception flows | EX1: Can’t connect to database (after step 2)  3. The system notifies:” Can’t connect to the database, please try again”  4. System move to 404 page | | |
| Trigger |  | | |
| Relationships |  | | |
| Post-conditions | Successful: Admin can deactivate a draft. | | |
| Priority | High | | |
| Frequency of use |  | | |
| Business rules |  | | |
| Special requirement |  | | |

### Search News

|  |  |  |  |
| --- | --- | --- | --- |
| Use case ID | UC-MN-09 | | |
| Name of use case | Search news | | |
| Creator | Tai Nguyen | Person updated |  |
| Creation date | 4/12/2016 | Date updated |  |
| Primary actor | Admin, Editor, Education staff, User | | |
| Brief description | Admin, Editor, Education staff, User search the news which they want to view | | |
| Pre-conditions | Admin, Editor, Education staff, User must access website  The system must be connected to the internet | | |
| Main success flow | 1. Admin, Editor, Education staff, User input information need to search 2. Admin, Editor, Education staff, User click button “Search” 3. System show related information 4. End of use case | | |
| Sub flows |  | | |
| Alternative flows |  | | |
| Exception flows | EF1: System can’t connect to database (after step 2)  3. The system notifies: “System can’t connect to the database, please try again”.  4. System move to 404 page | | |
| Trigger |  | | |
| Relationships |  | | |
| Post-conditions | Successful: Admin, Editor, Education staff, User can search information. | | |
| Priority | High | | |
| Frequency of use |  | | |
| Business rules |  | | |
| Special requirement |  | | |

### Sort News

|  |  |  |  |
| --- | --- | --- | --- |
| Use case ID | UC-MN-10 | | |
| Name of use case | Sort news | | |
| Creator | Tai Nguyen | Person updated |  |
| Creation date | 4/12/2016 | Date updated |  |
| Primary actor | Admin, Editor, Education staff, User | | |
| Brief description | Admin, Editor, Education staff, User sort the new follow the alphabet, posted date. | | |
| Pre-conditions | Admin, Editor, Education staff, User must access website  The system must be connected to the internet | | |
| Main success flow | 1. Admin, Editor, Education staff, User click button “Sort” 2. System show a method to soft 3. Admin, Editor, Education staff, User choose a method to soft 4. System show news follow that soft. 5. End of use case | | |
| Sub flows |  | | |
| Alternative flows |  | | |
| Exception flows | EF1: System can’t connect to database  3. The system notifies: “System can’t connect to the database, please try again”.  4. System move to 404 page | | |
| Trigger |  | | |
| Relationships |  | | |
| Post-conditions | Successful: Admin, Editor, Education staff, User can sort information. | | |
| Priority | High | | |
| Frequency of use |  | | |
| Business rules |  | | |
| Special requirement |  | | |

### Push News

|  |  |  |  |
| --- | --- | --- | --- |
| Use case ID | UC-MN-11 | | |
| Name of use case | Push news | | |
| Creator | Tai Nguyen | Person updated |  |
| Creation date | 4/12/2016 | Date updated |  |
| Primary actor | Admin | | |
| Brief description | Admin push the news to the top | | |
| Pre-conditions | Admin must access website  Admin must login  The system must be connected to the database | | |
| Main success flow | 1. Admin click button “Push” 2. System show list news to push 3. Admin choose pushing location. 4. System push that news. 5. System update database 6. End of use case | | |
| Sub flows |  | | |
| Alternative flows |  | | |
| Exception flows | EF1: System can’t connect to database (after step 2)  3. The system notifies: “System can’t connect to the database, please try again!”.  4. System move to 404 page | | |
| Trigger |  | | |
| Relationships |  | | |
| Post-conditions | Successful: Admin can push news successfully | | |
| Priority | High | | |
| Frequency of use |  | | |
| Business rules |  | | |
| Special requirement |  | | |

### Create news

|  |  |  |  |
| --- | --- | --- | --- |
| UC-MN-12 Create News | | | |
| Creator: | Minh Doan | Person Update: |  |
| Creation date: | 16/1/2017 | Date Update: |  |
| Description: | Content Admin, Editor, Education Staff create the news. | | |
| Users: | Content Admin, Education Staff, Editor | | |
| Trigger: | Content Admin, Editor, Education Staff click create news | | |
| Pre-Conditions: | * Content Admin, Editor, Education Staff must have account in system. * User is being in the “Manage News” GUI | | |
| Post-Conditions: | Success: News is created successfully. | | |
| Basic Flow: | 1. Content Admin, Editor, Education Staff clicks create news. 2. System shows “Create News” GUI. 3. Content Admin, Editor, Education Staff inputs information of news. 4. Content Admin, Editor, Education Staff clicks “OK”. 5. System checks the adequacy of information. 6. System notifies: “Create news successfully”. 7. Content Admin, Editor, Education Staff clicks “OK”. 8. End of use-case | | |
| Alternate Flow: | AF1: System Admin clicks “cancel” in “Create News” GUI: (after step 3)   1. User click button “Cancel”. 2. System shows “Manage News” GUI. | | |
| Exception Flows: | EF1: Content Admin, Editor, Education Staff didn’t input enough the needed information (after step 5).   1. System notifies: “Information is not enough”. 2. Content Admin, Editor, Education Staff clicks “OK”. 3. System shows “Create News” GUI. | | |
| Priority | High | | |
| Frequency of use | Usually | | |
| Business Rules |  | | |
| Mock up |  | | |

### Share news

|  |  |  |  |
| --- | --- | --- | --- |
| UC-MN-13 Share News | | | |
| Creator: | Minh Doan | Person Update: |  |
| Creation date: | 16/1/2017 | Date Update: |  |
| Description: | User share news to Facebook | | |
| Users: | User, Facebook SDK | | |
| Trigger: |  | | |
| Pre-Conditions: | User must have account Facebook  System must be connected to internet | | |
| Post-Conditions: | Success: News is shared successfully. | | |
| Basic Flow: | * + - 1. User click button “Share”       2. System call Facebook SDK       3. Facebook require User login       4. Facebook show “Share” UI       5. User click button “Share” on Facebook       6. News appear on user’s Facebook       7. End of use case | | |
| Alternate Flow: | AF1: User clicks “cancel” in “Share News” Facebook UI: (after step 3)   1. User click button “Cancel”. 2. System back to current page. | | |
| Exception Flows: |  | | |
| Priority | Normal | | |
| Frequency of use | Usually | | |
| Business Rules |  | | |
| Mock up |  | | |

## Manage examination

### View individual point

|  |  |  |
| --- | --- | --- |
| Use Case ID: | UC-ME-01 | |
| Name Use case: | View Individual Point | |
| Creator: | Minh Doan | Person updated: |
| Creation date: | 12/12/2016 | Date updated: |
| Primary actor: | Users | |
| Brief description: | Users view individual point by candidate code. | |
| Pre-conditions: | Network status is stable.  User must have candidate code  User is being in “View Result” GUI | |
| Post-conditions: | Successful: Website show points of user. | |
| Main Success flow: | 1. User clicks “View University Exam Result” link. 2. System views “View University Exam Result” GUI. 3. User inputs candidate code. 4. User clicks “View”. 5. System checks candidate code. 6. System shows scores. 7. End of Use case. | |
| Alternative flows: | AF1: Users cancel view university exam result (After step 2)   1. User clicks “Back” button or “Home Page”. 2. System shows “Home Page” GUI. | |
| Exception flows: | EF1: Candidate code does not exist (After step 5).   1. System notifies “candidate code does not exist!” 2. Return to step 3. | |
| Trigger: |  | |
| Relationships: | N/A | |
| Priority: | High | |
| Frequency of use: |  | |
| Business Rules: |  | |
| Special Requirement: | N/A | |

### Register exam

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-ME-02 | | |
| Name Use case: | Register Exam | | |
| Creator: | Minh Doan | Person updated: |  |
| Creation date: | 12/12/2016 | Date updated: |  |
| Primary actor: | Users | | |
| Brief description: | Users register skill examination. | | |
| Pre-conditions: | Network status is stable.  User is being in “Register Exam” GUI | | |
| Post-conditions: | Successful: Register successfully. | | |
| Main Success flow: | 1. User clicks “Register” link. 2. System views “Register” GUI. 3. User inputs information of user and upload image. 4. User clicks “OK”. 5. System checks information. 6. System saves profile. 7. System sends profile code to user’s email. 8. End of Use case. | | |
| Alternative flows: | AF1: User cancels register exam (After step 2)   1. User clicks “Cancel” button. 2. System shows “Home Page” GUI. | | |
| Exception flows: | EF1: User does not input enough information (After step 5).   1. System notifies “Input full needed fills, please!” 2. Return to step 3.   EF2: User inputs wrong data type (After step 5).   1. System focus on wrong fills. 2. Return to step 3. | | |
| Trigger: | * User clicks “Register” link. | | |
| Relationships: | N/A | | |
| Priority: | High | | |
| Frequency of use: |  | | |
| Business Rules: |  | | |
| Special Requirement: | N/A | | |

### View profile

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-ME-03 | | |
| Name Use case: | View profile | | |
| Creator: | Minh Doan | Person updated: |  |
| Creation date: | 12/12/2016 | Date updated: |  |
| Primary actor: | Users | | |
| Brief description: | User views candidate profile of this user. | | |
| Pre-conditions: | Network status is stable.  There is candidate profile.  There is profile code.  User is being in “Register Exam” GUI. | | |
| Post-conditions: | Successful: System views profile. | | |
| Main Success flow: | 1. User clicks “View Profile”. 2. System views “View Point” GUI. 3. User inputs profile code. 4. User clicks “OK”. 5. System checks profile code. 6. System shows profile. 7. End of Use case. | | |
| Alternative flows: | AF1: User cancels view profile (After step 2)   1. User clicks “Back” button. 2. System shows “Register Exam” GUI. | | |
| Exception flows: | EF1: Profile code does not exist (After step 4).   1. System notifies “Code does not exist!!” 2. Return to step 3. | | |
| Trigger: | User clicks “View Profile”. | | |
| Relationships: | N/A | | |
| Priority: | Medium | | |
| Frequency of use: |  | | |
| Business Rules: |  | | |
| Special Requirement: | N/A | | |

### Edit profile

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-ME-04 | | |
| Name Use case: | Edit profile | | |
| Creator: | Minh Doan | Person updated: |  |
| Creation date: | 12/12/2016 | Date updated: |  |
| Primary actor: | Users | | |
| Brief description: | User edits candidate profile. | | |
| Pre-conditions: | Network status is stable.  There is candidate profile.  User is being in “Profile” GUI. | | |
| Post-conditions: | Successful: Edit profile successfully. | | |
| Main Success flow: | 1. User clicks “Edit” button. 2. System views “Edit Profile” GUI. 3. User edits profile information. 4. User clicks “OK”. 5. System checks input information. 6. System saves profile. 7. End of Use case. | | |
| Alternative flows: | AF1: User cancels edit profile (After step 2)   1. User clicks “Cancel” button. 2. System shows “Profile” GUI. | | |
| Exception flows: | EF1: User does not input enough information (After step 4).   1. System notifies “Input full needed fills, please!” 2. Return to step 3.   EF2: User inputs wrong data type (After step 4).   1. System focus on wrong fills. 2. Return to step 3. | | |
| Trigger: | * User clicks “Import Point”. | | |
| Relationships: | N/A | | |
| Priority: | Medium | | |
| Frequency of use: |  | | |
| Business Rules: |  | | |
| Special Requirement: | N/A | | |

### Import point

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-ME-05 | | |
| Name Use case: | Import point | | |
| Creator: | Minh Doan | Person updated: |  |
| Creation date: | 12/12/2016 | Date updated: |  |
| Primary actor: | Education Staff | | |
| Brief description: | Education Staff imports point file to system. | | |
| Pre-conditions: | Network status is stable.  There is point file.  Education Staff logs in website. | | |
| Post-conditions: | Successful: Import successfully. | | |
| Main Success flow: | 1. Education Staff clicks “Manage Result”. 2. System shows “Manage Result” GUI. 3. Education Staff clicks “Import Point”. 4. System shows “Open File” GUI. 5. Education Staff choose point file. 6. Education Staff clicks “Open” button. 7. System saves list of points. 8. End of Use case. | | |
| Alternative flows: | AF1: Education Staff cancels import point (After step 4)   1. Education Staff clicks “Cancel” button. 2. System shows “Manage Result” GUI. | | |
| Exception flows: |  | | |
| Trigger: | Education Staff clicks “Import Point”. | | |
| Relationships: | N/A | | |
| Priority: | High | | |
| Frequency of use: |  | | |
| Business Rules: |  | | |
| Special Requirement: | N/A | | |

### Export profile

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-ME-06 | | |
| Name Use case: | Export profile | | |
| Creator: | Minh Doan | Person updated: |  |
| Creation date: | 12/12/2016 | Date updated: |  |
| Primary actor: | Education Staff | | |
| Brief description: | Education Staff exports profile of candidate. | | |
| Pre-conditions: | Network status is stable.  System must have profile of candidate  Education Staff logs in website. | | |
| Post-conditions: | Successful: Export profile successfully. | | |
| Main Success flow: | 1. Education Staff clicks “Manage Profile”. 2. System shows “Manage Profile” GUI. 3. Education Staff chooses profile. 4. Education Staff clicks “Export” button. 5. System shows message “Export Profile?” 6. Education Staff clicks “Yes” button. 7. System exports profile to MS-WORD. 8. End of Use case. | | |
| Alternative flows: | AF1: User cancels export profile (After step 6)   1. User clicks “Cancel” button. 2. System shows “Manage Profile” GUI. | | |
| Exception flows: |  | | |
| Trigger: | Education Staff clicks “Export”. | | |
| Relationships: | N/A | | |
| Priority: | Medium | | |
| Frequency of use: |  | | |
| Business Rules: |  | | |
| Special Requirement: | N/A | | |

### Create exam

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-ME-07 | | |
| Name Use case: | Create Exam | | |
| Creator: | Minh Doan | Person updated: |  |
| Creation date: | 12/12/2016 | Date updated: |  |
| Primary actor: | Education Staff | | |
| Brief description: | Education Staff exports profile of candidate. | | |
| Pre-conditions: | Network status is stable.  System must have profile of candidate  Education Staff logs in website. | | |
| Post-conditions: | Successful: Export profile successfully. | | |
| Main Success flow: | 1. Education Staff clicks “Manage Profile”. 2. System shows “Manage Profile” GUI. 3. Education Staff chooses profile. 4. Education Staff clicks “Export” button. 5. System shows message “Export Profile?” 6. Education Staff clicks “Yes” button. 7. System exports profile to MS-WORD. 8. End of Use case. | | |
| Alternative flows: | AF1: User cancels export profile (After step 6)   1. User clicks “Cancel” button. 2. System shows “Manage Profile” GUI. | | |
| Exception flows: |  | | |
| Trigger: | Education Staff clicks “Export”. | | |
| Relationships: | N/A | | |
| Priority: | Medium | | |
| Frequency of use: |  | | |
| Business Rules: |  | | |
| Special Requirement: | N/A | | |

## Manage categories

### View categories

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-MC-01 | | |
| Name Use case: | View Categories in Homepage | | |
| Creator: | Minh Doan | Person updated: | Minh Doan |
| Creation date: | 2/12/2016 | Date updated: | 12/12/2016 |
| Primary actor: | Admin System, Admin Content, Editor, Education Staff, Users | | |
| Brief description: | Admin System, Admin Content, Editor, Education Staff, Users view the categories on website | | |
| Main Success flow: | 1. Admin System, Admin Content, Editor, Education Staff, Users access to website with domain name: "". 2. System show "Home Page" GUI that includes the categories bar. 3. End of Use case. | | |
| Alternative flows: |  | | |
| Exception flows: |  | | |
| Trigger: | Admin System, Admin Content, Editor, Education Staff, Users access to website | | |
| Relationships: | N/A | | |
| Pre-conditions: | Network status is stable. | | |
| Post-conditions: | Successful: System show "Home Page" GUI. | | |
| Priority: | High | | |
| Frequency of use: |  | | |
| Business Rules: |  | | |
| Special Requirement: | N/A | | |

### View category

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-MC-02 | | |
| Name Use case: | View Categories In CMS | | |
| Creator: | Minh Doan | Person updated: | Minh Doan |
| Creation date: | 2/12/2016 | Date updated: | 12/12/2016 |
| Primary actor: | Admin Content | | |
| Brief description: | Admin Content view categories in manage page | | |
| Pre-conditions: | Network status is stable.  User must log in website by admin account and being in the CMS GUI. | | |
| Post-conditions: | Successful: System show “Manage Categories” interface with list of categories. | | |
| Main Success flow: | 1. Admin Content clicks "Manage Categories". 2. System show "Manage Categories" GUI with the list of categories. 3. End of Use case. | | |
| Alternative flows: |  | | |
| Exception flows: | EF1: System can’t connect to database (after step 1)  2.The system notifies:” Can’t connect to the database, please try again!”  3.System move to 404 page | | |
| Trigger: | Admin Content clicks “Manage Categories” | | |
| Relationships: | N/A | | |
| Priority: | High | | |
| Frequency of use: |  | | |
| Business Rules: |  | | |
| Special Requirement: | N/A | | |

### Add category

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-MC-03 | | |
| Name Use case: | Add Category | | |
| Creator: | Minh Doan | Person updated: | Minh Doan |
| Creation date: | 2/12/2016 | Date updated: | 12/12/2016 |
| Primary actor: | Admin Content | | |
| Brief description: | Admin Content adds new category to categories bar. | | |
| Pre-conditions: | Network status is stable.  Admin Content must log in and being in the CMS GUI. | | |
| Post-conditions: | Successful: Add new category successfully. | | |
| Main Success flow: | 1. Admin Content clicks "Manage Categories". 2. System show "Manage Categories" GUI. 3. Admin Content clicks "Add" button. 4. System show "Add Category" GUI. 5. Admin Content fills information of category. 6. Admin Content clicks “OK” 7. System checks information of category 8. System saves and views list of categories. 9. End of use case. | | |
| Alternative flows: |  | | |
| Exception flows: | EF1: Category have empty information (after step 4)  5.System notify “Please input into field!”  6.System focus into empty field.  EF2: Input have invalid type (after step 6)  7.System notify “Invalid input, please try again”  8. System focus to invalid input  EF3: System can’t connect to database (after step 7)  8.The system notifies:” Can’t connect to the database, please try again!”  9. System move to 404 page. | | |
| Trigger: | Admin Content clicks “Add” button. | | |
| Relationships: | N/A | | |
| Priority: | High | | |
| Frequency of use: |  | | |
| Business Rules: |  | | |
| Special Requirement: | N/A | | |

### Edit category

|  |  |  |
| --- | --- | --- |
| Use Case ID: | UC-MC-04 | |
| Name Use case: | Edit Category | |
| Creator: | Minh Doan | Person updated: |
| Creation date: | 12/12/2016 | Date updated: |
| Primary actor: | Admin Content | |
| Brief description: | Admin Content edits information of category. | |
| Pre-conditions: | Network status is stable.  Admin Content must log in  Admin Content being in the CMS GUI.  Category is existing. | |
| Post-conditions: | Successful: Edit category successfully. | |
| Main Success flow: | 1. Admin Content clicks "Manage Categories". 2. System show "Manage Categories" GUI. 3. Admin Content chooses category 4. Admin Content clicks "Edit" button. 5. System show "Edit Category" GUI. 6. Admin Content edits information of category. 7. Admin Content clicks “OK” 8. System checks information of category 9. System saves new information of category. 10. End of Use case. | |
| Alternative flows: | AF1: Admin Content cancels (after step 3)   1. System back to current page | |
| Exception flows: | EF1: Invalid information (after step 7)  8.The system notifies:” Invalid input, please try another”  9. The system focus into invalid field  EF2: Can’t connect to database (after step 8)  9.The system notifies:” Can’t connect to the database, please try again!”  10. System move to 404 page | |
| Trigger: | Admin Content clicks “Edit” button. | |
| Relationships: | N/A | |
| Priority: | High | |
| Frequency of use: |  | |
| Business Rules: |  | |
| Special Requirement: | N/A | |

### Delete category

|  |  |  |
| --- | --- | --- |
| Use Case ID: | UC-MC-05 | |
| Name Use case: | Delete Category | |
| Creator: | Minh Doan | Person updated: |
| Creation date: | 12/12/2016 | Date updated: |
| Primary actor: | Admin Content | |
| Brief description: | Admin Content deletes information of category. | |
| Pre-conditions: | Network status is stable.  Admin Content must log in  Admin Content being in the CMS GUI.  Category is existing. | |
| Post-conditions: | Successful: Delete category successfully. | |
| Main Success flow: | * 1. Admin Content clicks "Manage Categories".   2. System show "Manage Categories" GUI.   3. Admin Content chooses category   4. Admin Content clicks "Delete" button.   5. System show alert dialog “Ok | Cancel”   6. Admin Content clicks “OK”   7. System update data to database   8. System show list of categories.  1. End of use case. | |
| Alternative flows: | AF1: Admin Content cancels (after step 3)  4.System back to current page | |
| Exception flows: | EF1: Can’t connect to database (after step 6)  7.The system notifies:” Can’t connect to the database, please try again!”  8. System move to 404 page | |
| Trigger: | Admin Content clicks “Delete” button. | |
| Relationships: | N/A | |
| Priority: | High | |
| Frequency of use: |  | |
| Business Rules: |  | |
| Special Requirement: | N/A | |

## Manage languages

### 5.1. Change language

|  |  |  |
| --- | --- | --- |
| Use Case ID: | UC-CLG-01 | |
| Name Use case: | Change Languages | |
| Creator: | Hai Tran | Person updated: |
| Creation date: |  | Date updated: |
| Primary actor: | Admin, Editor, Education staff, User | |
| Brief description: | Actor change language of System | |
| Pre-conditions: | System must have connected with internet | |
| Main Success flow: | 1.Admin, Editor, Education staff, User click “Change Languages”  2.System handler change.  3.System change language.  4.End of use case. | |
| Sub flows: | None | |
| Alternative flows: | None | |
| Exception flows: | None | |
| Trigger: |  | |
| Relationships: | None | |
| Post-conditions: | Successful: Change language successfully. | |
| Priority: | Low | |
| Frequency of use: | None | |
| Business Rules: |  | |
| Special Requirement: | None | |

## Manage banner

### Create banner

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-MB-01 | | |
| Name Use case: | Create Banner | | |
| Creator: | Hai Tran | Person updated: |  |
| Creation date: |  | Date updated: |  |
| Primary actor: | Admin | | |
| Brief description: | Admin create new banner | | |
| Main Success flow: | * + - 1. Admin choose “Banner” tab       2. System display “Banner” page       3. Admin click button “Create”   1. System show form choose content(images)   2. Admin choose images   3. Admin click button “Save”   4. System save data into database   5. System notify “Success”   6. End of use case | | |
| Sub flows: | N/A | | |
| Alternative flows: | AF1: Admin click cancel (after step 3)  4. System show alert dialog “Do you want to save in draft”  5. Admin click button “Yes”  6. System save banner into draft  7. System back to “Banner” page | | |
| Exception flows: | EF1: Image size too large for display (begin in step 3):  4. System notify “Resize images for fit with System”  5. System back to form choose content  EF2: System can’t connect to database (after step 4)  5. System notify “Can’t connect to database, please try again”  6. System move to 404 page | | |
| Trigger: |  | | |
| Relationships: | None | | |
| Pre-conditions: | Admin must have logged in with user’s account | | |
| Post-conditions: | Successful: Create new banner successfully | | |
| Priority: | Medium | | |
| Frequency of use: | None | | |
| Business Rules: |  | | |
| Special Requirement: | None | | |

### View banner

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-MB-02 | | |
| Name Use case: | View Banner | | |
| Creator: | Hai Tran | Person updated: |  |
| Creation date: |  | Date updated: |  |
| Primary actor: | Admin, Editor, Education staff, User | | |
| Brief description: | Admin, Editor, Education staff, User view banner | | |
| Pre-conditions: | The system must be connected to the internet | | |
| Main Success flow: | * + 1. Admin, Editor, Education staff, User access website     2. System show content of website     3. Admin, Editor, Education staff, User view banner     4. End of use case | | |
| Sub flows: | None | | |
| Alternative flows: | None | | |
| Exception flows: | None | | |
| Trigger: |  | | |
| Relationships: | None | | |
| Post-conditions: | Successful: View banner successfully. | | |
| Priority: | Medium | | |
| Frequency of use: | None | | |
| Business Rules: |  | | |
| Special Requirement: | None | | |

### Show hide banner

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-MB-03 | | |
| Name Use case: | Show-Hide Banner | | |
| Creator: | Hai Tran | Person updated: |  |
| Creation date: |  | Date updated: |  |
| Primary actor: | Admin | | |
| Brief description: | Admin do show/hide banner | | |
| Pre-conditions: | Admin must login  System must have banner | | |
| Main Success flow: | * + - 1. Admin choose “Manage banner “tab       2. System display “Manage banner” page       3. Admin show/hide banner displayed in page       4. System save information into database       5. End of use case | | |
| Sub flows: | N/A | | |
| Alternative flows: |  | | |
| Exception flows: | EF1: System can’t connect database (after step 3):  4. System notify “Can’t connect to database, please try again!”  5. System move to 404 page | | |
| Trigger: |  | | |
| Relationships: | None | | |
| Post-conditions: | Successful: Show/hide banner successfully | | |
| Priority: | High | | |
| Frequency of use: | Always | | |
| Business Rules: |  | | |
| Special Requirement: | None | | |

### Edit banner

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-MB-04 | | |
| Name Use case: | Edit Banner | | |
| Creator: | Hai Tran | Person updated: |  |
| Creation date: |  | Date updated: |  |
| Primary actor: | Admin | | |
| Brief description: | Admin edit banner | | |
| Pre-conditions: | Admin must login  The system must be connected to the internet | | |
| Main Success flow: | 1. Admin choose “Manage banner” tab 2. System display “Manage banner” page 3. Admin choose banner 4. System show content banner 5. Admin edit banner 6. Admin choose “Save” 7. System save content to database 8. System notify “Success” 9. System show content edited 10. End of use case | | |
| Sub flows: | N/A | | |
| Alternative flows: | AF1: Admin click button “Cancel” (after step 5)  6. System show alert dialog” Do you want to save this into draft?”  7. Admin click button “Yes | No”  8. System back to “Manage banner” page | | |
| Exception flows: | EF1: Images size can’t be fitted with System (after step 5):  5. System notify “Can’t solve images size”  6. System back to edit form  EF2: System can’t connect to database (after step 6)  7. System notify “Can’t connect to database, please try again!”  8. System move to 404 page | | |
| Trigger: |  | | |
| Relationships: | None | | |
| Post-conditions: | Successful: Edit banner successfully | | |
| Priority: | High | | |
| Frequency of use: | Sometimes | | |
| Business Rules: |  | | |
| Special Requirement: | None | | |

### Delete banner

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-MB-05 | | |
| Name Use case: | Delete Banner | | |
| Creator: | Hai Tran | Person updated: |  |
| Creation date: |  | Date updated: |  |
| Primary actor: | Admin | | |
| Brief description: | Admin delete banner | | |
| Pre-conditions: | Admin must login  System must have banner  Admin inside “Manage banner” page | | |
| Main Success flow: | 1. Admin choose “Manage banner” 2. System show interface manage banner 3. Admin choose banner 4. System show content banner 5. Admin choose delete 6. System show alert “Ok | Cancel” 7. Admin choose “Ok” 8. System update database 9. System notify “Success” 10. System update interface 11. End of use case | | |
| Sub flows: | N/A | | |
| Alternative flows: | AF1: User click button “Cancel” (after step 5)  6. System back to “Manage banner” page | | |
| Exception flows: | EF1: System can’t connect to database (after step 7):  8. System notify “Can’t connect to database, please try again!”  9. System move to 404 page | | |
| Trigger: |  | | |
| Relationships: | None | | |
| Post-conditions: | Successful: Delete banner successfully | | |
| Priority: | High | | |
| Frequency of use: | Sometimes | | |
| Business Rules: |  | | |
| Special Requirement: | None | | |

### Arrange images

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-MB-06 | | |
| Name Use case: | Arrange Images | | |
| Creator: | Hai Tran | Person updated: |  |
| Creation date: |  | Date updated: |  |
| Primary actor: | Admin | | |
| Brief description: | Admin arrange images in banner | | |
| Pre-conditions: | Admin inside “Manage banner” page  Admin must login  System must have banner | | |
| Main Success flow: | 1. Admin choose “Manage banner” 2. System display “Manage banner” page 3. Admin chooses pop-ups to arrange images 4. Admin chooses images to arrange 5. Admin click button “Save” 6. System save information into database 7. System notify “Arrange successfully” 8. End of use case | | |
| Sub flows: | N/A | | |
| Alternative flows: | AF1: Admin cancel arrange pop-up (after step 3)  4. Admin click button “Cancel”  5. System back to “Manage banner” page | | |
| Exception flows: | EF1: System can’t connect to database (after step 4)  5. System notify “Can’t connect to database, please try again!”  6. System move to 404 page | | |
| Trigger: |  | | |
| Relationships: | None | | |
| Post-conditions: | Successful: Arrange images successfully | | |
| Priority: | High | | |
| Frequency of use: | N/A | | |
| Business Rules: |  | | |
| Special Requirement: | N/A | | |

## Manage questions

### Send questions

|  |  |  |  |
| --- | --- | --- | --- |
| **UC-MQA-01: Send questions** | | | |
| Creator: | Khoi Nguyen | Person Update: |  |
| Creation date: | 5/12/2016 | Date Update: |  |
| Brief Description: | User send the question | | |
| Primary Actors: | User | | |
| Trigger: | User click “Q&A” tab | | |
| Pre-Conditions: | User access “Q&A” page | | |
| Post-Conditions: | Success: User send question successfully | | |
| Basic Flow: | 1. System displays “Q&A” page include question form. 2. User fill information into question form. 3. User click button “Send” 4. System check input. 5. System save data into database 6. System notify “Send successfully” 7. End of use case | | |
| Alternate Flow: | AF1: User cancel send question (after step 2)   1. User click button “Cancel” 2. System empty question form | | |
| Exception Flows: | EF1: User fill invalid information (after step 3)  4.System notify: “Invalid input, please try again!”  5.System focus on wrong field.  EF3: System can’t connect to database (after step 5)  6.System notify “Can’t connect database, please try again”  7.System move to 404 page | | |
| Relationships: | None | | |
| Business Rules: | Name, Email: <= 100 characters  Question: <= 500 characters  Email: have @... | | |
| Priority: | Low | | |
| Frequency of use: | Medium | | |
| Special Requirements: | None | | |

### Answer questions

|  |  |  |  |
| --- | --- | --- | --- |
| **UC-MQA-02: Answer questions** | | | |
| Creator: | Khoi Nguyen | Person Update: |  |
| Creation date: | 5/12/2016 | Date Update: |  |
| Brief Description: | Admin, Editor, Education staff answer the question, Mail server send answer to User | | |
| Primary Actors: | Admin, Editor, Education staff, Mail server | | |
| Trigger: | Admin, Editor, Education staff click “Q&A” tab | | |
| Pre-Conditions: | Admin, Editor, Education staff inside “Q&A” page  Admin, Editor, Education staff must login  Mail server must active  User must have email | | |
| Post-Conditions: | Success: Answer has been sent to User successfully | | |
| Basic Flow: | 1. Admin, Editor, Education staff choose question to answer 2. System display answer form. 3. Admin, Editor, Education staff write answer 4. Admin, Editor, Education staff click button “Send” 5. System call Mail server send answer to User 6. System save answer into database 7. System notify “Send answer successfully” 8. End of use case. | | |
| Alternate Flow: | AF1: Admin cancel send answer (after step 3)   1. Admin, Editor, Education staff click button “Cancel” 2. System save answer into draft | | |
| Exception Flows: | EF1: System can’t connect to database (after step 5)   1. System notify “Can’t connect database, please try later” 2. System move to 404 page | | |
| Relationships: | Include View Question function | | |
| Business Rules: |  | | |
| Priority: | Medium | | |
| Frequency of use: | Medium | | |
| Special Requirements: |  | | |

### View questions

|  |  |  |  |
| --- | --- | --- | --- |
| **UC-MQA-03: View questions** | | | |
| Creator: | Khoi Nguyen | Person Update: |  |
| Creation date: | 5/12/2016 | Date Update: |  |
| Brief Description: | Admin, Editor, Education staff, User view questions have been answered | | |
| Primary Actors: | Admin, Editor, Education staff, User | | |
| Trigger: | Admin, Editor, Education staff, User choose “Q&A” tab | | |
| Pre-Conditions: | Admin, Editor, Education staff, User inside “Q&A”page  System must have questions | | |
| Post-Conditions: | Success: Admin, Editor, Education staff, User view questions have been answered successfully. | | |
| Basic Flow: | 1. System display “Q&A” page 2. Admin, Editor, Education staff, User choose question that they want to view 3. System show an answer of question 4. End of use case | | |
| Alternate Flow: | None | | |
| Exception Flows: | EF1: System can’t connect to database (after step 2)   1. System notify “Can’t connect to database, please try again” 2. System move to 404 page | | |
| Relationships: | None | | |
| Business Rules: | None | | |
| Priority: | Medium | | |
| Frequency of use: | Medium | | |
| Special Requirements: | None | | |

### Approve questions

|  |  |  |  |
| --- | --- | --- | --- |
| **UC-MQA-04: Approve questions** | | | |
| Creator: | Khoi Nguyen | Person Update: |  |
| Creation date: | 5/12/2016 | Date Update: |  |
| Brief Description: | Admin approve questions have been answered to show on website | | |
| Primary Actors: | Admin | | |
| Trigger: |  | | |
| Pre-Conditions: | Admin inside “Q&A” page  Questions must have been answered | | |
| Post-Conditions: | Success: Admin approve question successfully | | |
| Basic Flow: | 1. Admin choose “Q&A” tab  2. System display “Q&A” page  3. Admin choose question has not approved yet  4. System display detail of this question.  5. Admin read detail  6. Admin click button “Approve & Post”  7. System update information into database  8. System show updated status of question on “Q&A” page  9. System notify “Approve success”  10. End of use case | | |
| Alternate Flow: | AF1: Admin cancel approve question (after step 5)  6. System show alert dialog “Ok | Cancel”  7. Admin click button “Ok”  8. System back to “Q&A” page | | |
| Exception Flows: | EF1: System can’t connect with database (after step 6)  7. System notify “Can’t connect to database, please try again!”  8. System move to 404 page | | |
| Relationships: | Include View Question & Answer Question | | |
| Business Rules: | None | | |
| Priority: | High | | |
| Frequency of use: | Medium | | |
| Special Requirements: | None | | |

### Search questions

|  |  |  |  |
| --- | --- | --- | --- |
| **UC-MQA-05: Search questions** | | | |
| Creator: | Khoi Nguyen | Person Update: |  |
| Creation date: | 5/12/2016 | Date Update: |  |
| Brief Description: | Admin, Editor, Education staff, User search the question posted in website. | | |
| Primary Actors: | Admin, Editor, Education staff, User | | |
| Trigger: | Admin, Editor, Education staff, User choose “Q&A” tab | | |
| Pre-Conditions: | Admin, Editor, Education staff, User inside “Q&A” page | | |
| Post-Conditions: | Success:Admin, Editor, Education staff, User search question posted in list of frequently asked questions successfully | | |
| Basic Flow: | 1. System display “Q&A” page.  2. Admin, Editor, Education staff, User input key search into find box  3. Admin, Editor, Education staff, User click button “Search”  4. System select question match with key search  5. System display list of related question.  6. End of use case | | |
| Alternate Flow: | None | | |
| Exception Flows: | EF1: Actors did not enter information (after step 1)  2. System display notify “Please enter information”  3. System still in “Q&A” page  EF2: System don’t have question match with key search (after step 3)  4. System notify “Question not found”  5. System show blank page | | |
| Relationships: | None | | |
| Business Rules: | None | | |
| Priority: | Low | | |
| Frequency of use: | Medium | | |
| Special Requirements: | None | | |

## Manage pop-ups

### Create pop-ups

|  |  |  |  |
| --- | --- | --- | --- |
| **UC-PU-01: Create Pop-ups** | | | |
| Creator: | Khoi Nguyen | Person Update: |  |
| Creation date: | 5/12/2016 | Date Update: |  |
| Brief Description: | Admin creates new pop-up | | |
| Primary Actors: | Admin | | |
| Trigger: | Admin choose “Manage Pop-ups” tab | | |
| Pre-Conditions: | Admin inside Manage Pop-ups page  System must have internet connected | | |
| Post-Conditions: | Success: Admin creates pop-up successfully. | | |
| Basic Flow: | 1. System display “Manage pop-ups” page  2. Admin clicks button “Create Pop-up”  3. System display create pop-up GUI.  4. Admin adds image & set timer for pop-up.  5. Admin clicks button “Save”  6. System check input  6. System save data into database  7. System notify “Create successfully”  8. End of use case | | |
| Alternate Flow: | AF1: Admin cancels create pop-up (after step 4)  5. System show alert dialog “Ok | Cancel”  6. Admin clicks button “Cancle”  7. System save data into draft version  8. System back to “Manage Popups” page | | |
| Exception Flows: | EF1: Admin does not fill all information (after step 6)  7. System notify “Please fill all information”  8. System focus into blank field  EF2: System can’t connect with database (after step 5)  6. System notify “Can’t connect database, please try later”  7. System move to 404 page | | |
| Relationships: | None | | |
| Business Rules: | None | | |
| Priority: | High | | |
| Frequency of use: | Medium | | |
| Special Requirements: | None | | |

### View pop-ups

|  |  |  |  |
| --- | --- | --- | --- |
| **UC-PU-02: View Pop-ups** | | | |
| Creator: | Khoi Nguyen | Person Update: |  |
| Creation date: | 5/12/2016 | Date Update: |  |
| Brief Description: | Admin, Editor, Education staff, User view pop-up on website | | |
| Primary Actors: | Admin, Editor, Education staff, User | | |
| Trigger: |  | | |
| Pre-Conditions: | Admin, Editor, Education staff, User must access website  System must have internet connected | | |
| Post-Conditions: | Success: Admin, Editor, Education staff, User view pop-up successfully. | | |
| Basic Flow: | 1. System display Pop-ups on screen.  2. End of use case | | |
| Alternate Flow: | None | | |
| Exception Flows: | None | | |
| Relationships: | None | | |
| Business Rules: | None | | |
| Priority: | Medium | | |
| Frequency of use: | Medium | | |
| Special Requirements: | None | | |

### Show-hide pop-ups

|  |  |  |  |
| --- | --- | --- | --- |
| **UC-PU-03: Show-Hide Pop-ups** | | | |
| Creator: | Khoi Nguyen | Person Update: |  |
| Creation date: | 5/12/2016 | Date Update: |  |
| Brief Description: | Admin show/hide the pop-up | | |
| Primary Actors: | Admin | | |
| Trigger: | Admin chooses “Manage Pop-ups” tab | | |
| Pre-Conditions: | Admin inside “Manage Pop-ups” page  System must have internet connected  System must have pop-ups activating | | |
| Post-Conditions: | Success: Admin show/hide pop-up successfully. | | |
| Basic Flow: | 1. System display “Manage pop-ups” page  2. Admin show/hide pop-ups displayed in page  3. System save information into database  4. End of use case | | |
| Alternate Flow: | None | | |
| Exception Flows: | EF1: System can’t connect to database (after step 2)  3. System notify “Can’t connect to database, please try later”  4. System move to 404 page | | |
| Relationships: | Include View pop-ups function | | |
| Business Rules: | None | | |
| Priority: | Medium | | |
| Frequency of use: | Medium | | |
| Special Requirements: | None | | |

### Edit pop-ups

|  |  |  |  |
| --- | --- | --- | --- |
| **UC-PU-04: Edit Pop-ups** | | | |
| Creator: | Khoi Nguyen | Person Update: |  |
| Creation date: | 5/12/2016 | Date Update: |  |
| Brief Description: | Admin edits information of pop-up | | |
| Primary Actors: | Admin | | |
| Trigger: | Admin chooses “Manage Pop-ups” tab | | |
| Pre-Conditions: | Admin inside “Manage Pop-ups” page  System must have internet connected  Admin must login  System must have pop-ups | | |
| Post-Conditions: | Success: Admin edits pop-up successfully. | | |
| Basic Flow: | 1. System display “Manage pop-ups” page  2. Admin chooses pop-ups to edit  3. System display “Edit pop-up” form  4. Admin edits information of pop-up  5. Admin clicks button “Save”  6. System check input  7. System save information into database  8. System notify “Edit successfully”  9. End of use case | | |
| Alternate Flow: | AF1: Admin cancels create pop-up (after step 4)  5. System show alert dialog “Ok | Cancel”  6. Admin clicks button “Ok”  7. System back to “Manage Pop-ups” page | | |
| Exception Flows: | EF1: Admin does not fill full information (after step 5)  6. System notify “Please fill all information”  7. System focus into empty field  EF2: System can’t connect to database (after step 7)  8. System notify “Can’t connect to database, please try again!”  9. System move to 404 page | | |
| Relationships: | Include View pop-ups function | | |
| Business Rules: | None | | |
| Priority: | Medium | | |
| Frequency of use: | Medium | | |
| Special Requirements: | None | | |

### Delete pop-ups

|  |  |  |  |
| --- | --- | --- | --- |
| **UC-PU-05: Delete Pop-ups** | | | |
| Creator: | Khoi Nguyen | Person Update: |  |
| Creation date: | 5/12/2016 | Date Update: |  |
| Brief Description: | Admin delete pop-up | | |
| Primary Actors: | Admin | | |
| Trigger: | Admin choose “Manage Pop-ups” tab | | |
| Pre-Conditions: | Admin inside “Manage Pop-ups” page  Admin must login  System must have pop-ups | | |
| Post-Conditions: | Success: Admin delete pop-up successfully. | | |
| Basic Flow: | 1. System display “Manage pop-ups” page  2. Admin delete pop-ups  3. System show alert dialog “Ok | Cancel”  4. Admin click button “OK”.  5. System update information into database  6. System notify “Delete success”  7. End of use case | | |
| Alternate Flow: | AF1: Admin cancel delete pop-up ( after step 2 )  3. System back to “Manage Pop-ups” page | | |
| Exception Flows: | EF1: System can’t connect to database (after step 4)  5. System notify “Can’t connect to database, please try again!”  6. System move to 404 page. | | |
| Relationships: | Include View pop-ups function | | |
| Business Rules: | None | | |
| Priority: | Medium | | |
| Frequency of use: | Medium | | |
| Special Requirements: | None | | |

### Arrange pop-ups

|  |  |  |  |
| --- | --- | --- | --- |
| **UC-PU-06: Arrange Pop-ups** | | | |
| Creator: | Khoi Nguyen | Person Update: |  |
| Creation date: | 5/12/2016 | Date Update: |  |
| Brief Description: | Admin arrange images in pop-up | | |
| Primary Actors: | Admin | | |
| Trigger: | Admin click “Manage Pop-ups” tab | | |
| Pre-Conditions: | Admin inside “Manage Pop-ups” page  Admin must login  System must have pop-ups | | |
| Post-Conditions: | Success: Admin arrange images in pop-up successfully. | | |
| Basic Flow: | 1. System display “Manage pop-ups” page  2. Admin chooses pop-ups to arrange images  3. Admin click button “Arrange Imange”  4. System show arrange image interface  5. Admin arrange images in pop-ups  6. Admin click button “Save”  7. System save information into database  8. System notify “Arrange successfully”  9. End of use case | | |
| Alternate Flow: | AF1: Admin cancel arrange pop-up (after step 5)  6. Admin click button “Cancel”  7. System back to “Manage Pop-ups” page | | |
| Exception Flows: | EF1: System can’t connect to database (after step 7)  8. System notify “Can’t connect to database, please try again!”  9. System move to 404 page | | |
| Relationships: | Include View pop-ups function | | |
| Business Rules: | None | | |
| Priority: | Medium | | |
| Frequency of use: | Medium | | |
| Special Requirements: | None | | |

## Manage homepage

### Add

|  |  |  |  |
| --- | --- | --- | --- |
| **UC-MH-01: Add Homepage** | | | |
| Creator: | Khoi Nguyen | Person Update: |  |
| Creation date: | 5/12/2016 | Date Update: |  |
| Brief Description: | Content Admin add new position in homepage | | |
| Primary Actors: | Content Admin | | |
| Trigger: | Content Admin click “Manage Homepage” tab | | |
| Pre-Conditions: | Content Admin inside “Manage Homepage” page  Content Admin must login | | |
| Post-Conditions: | Success: Content Admin add position for homepage successfully. | | |
| Basic Flow: | 1. System display “Manage Homepage” page  2. Content Admin choose position  3. Content clicks button “Add”  4. System save data into database  5. End of use case | | |
| Alternate Flow: |  | | |
| Exception Flows: | EF1: System can’t connect to database (after step 3)  4. System notify “Can’t connect to database, please try again!”  5. System move to 404 page | | |
| Relationships: |  | | |
| Business Rules: | None | | |
| Priority: | Medium | | |
| Frequency of use: | Medium | | |
| Special Requirements: | None | | |

### Edit homepage

|  |  |  |  |
| --- | --- | --- | --- |
| **UC-MH-02: Edit Homepage** | | | |
| Creator: | Khoi Nguyen | Person Update: |  |
| Creation date: | 5/12/2016 | Date Update: |  |
| Brief Description: | Content Admin edit position in homepage | | |
| Primary Actors: | Content Admin | | |
| Trigger: | Content Admin click “Manage Homepage” tab | | |
| Pre-Conditions: | Content Admin inside “Manage Homepage” page  Content Admin must login | | |
| Post-Conditions: | Success: Content Admin edit position for homepage successfully. | | |
| Basic Flow: | 1. System display “Manage Homepage” page  2. Content Admin choose position  3. Content clicks button “Edit”  4.Content Admin edit position  5. Content Admin click “Save”  6. System update data into database  7. End of use case | | |
| Alternate Flow: | AF1: Content Admin clicks “Cancel” (after step 4)  5.System back to “Manage Homepage” page | | |
| Exception Flows: | EF1: System can’t connect to database (after step 5)  6. System notify “Can’t connect to database, please try again!”  7. System move to 404 page | | |
| Relationships: |  | | |
| Business Rules: | None | | |
| Priority: | Medium | | |
| Frequency of use: | Medium | | |
| Special Requirements: | None | | |

### Delete homepage

|  |  |  |  |
| --- | --- | --- | --- |
| **UC-MH-03: Delete Homepage** | | | |
| Creator: | Khoi Nguyen | Person Update: |  |
| Creation date: | 5/12/2016 | Date Update: |  |
| Brief Description: | Content Admin delete position in homepage | | |
| Primary Actors: | Content Admin | | |
| Trigger: | Content Admin click “Manage Homepage” tab | | |
| Pre-Conditions: | Content Admin inside “Manage Homepage” page  Content Admin must login | | |
| Post-Conditions: | Success: Content Admin delete position for homepage successfully. | | |
| Basic Flow: | 1. System display “Manage Homepage” page  2. Content Admin choose position  3. Content clicks button “Delete”  4. System show alert dialog “Ok | Cancel”  5. Content Admin click “Ok”  6. System update data into database  7. End of use case | | |
| Alternate Flow: | AF1: Content Admin clicks “Cancel” (after step 4)  5.System back to “Manage Homepage” page | | |
| Exception Flows: | EF1: System can’t connect to database (after step 5)  6. System notify “Can’t connect to database, please try again!”  7. System move to 404 page | | |
| Relationships: |  | | |
| Business Rules: | None | | |
| Priority: | Medium | | |
| Frequency of use: | Medium | | |
| Special Requirements: | None | | |

## Use case Diagram

C:\Users\LAPTOP24H\Desktop\New folder\usecase diagram\ChangeLanguages.png

Figure 1: Change Language

C:\Users\LAPTOP24H\Desktop\New folder\usecase diagram\ManageAccount.png

Figure 2: Manage accounts

C:\Users\LAPTOP24H\Desktop\New folder\usecase diagram\ManageBanner.png

Figure 3: Manage banner

C:\Users\LAPTOP24H\Desktop\New folder\usecase diagram\ManageCate.png

Figure 4: Manage category

C:\Users\LAPTOP24H\Desktop\New folder\usecase diagram\ManageExamination.png

Figure 5: Manage Examination

C:\Users\LAPTOP24H\Desktop\New folder\usecase diagram\ManageNews.png

Figure 6: Manage news

C:\Users\LAPTOP24H\Desktop\New folder\usecase diagram\ManagePopus.png

Figure 7: Manage Pop-ups

C:\Users\LAPTOP24H\Desktop\New folder\usecase diagram\ManageQ&A.png

Figure 8: Manage questions

C:\Users\haitr\AppData\Local\Microsoft\Windows\INetCacheContent.Word\ManageHomepage.png

Figure 9: Manage homepage

C:\Users\haitr\AppData\Local\Microsoft\Windows\INetCacheContent.Word\ManageResource.png

*Figure 10: Manage resource*