

Use-case Description

Van Lang Admissions

# Revision Table

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

Contents

[Revision Table 1](#_Toc470078799)

[1. INTRODUCTION 3](#_Toc470078800)

[1.1. Purpose 3](#_Toc470078801)

[1.2. Audience 3](#_Toc470078802)

[2. Usecase Descriptions 3](#_Toc470078803)

[Manage news 20](#_Toc470078804)

[1.1 Post News 22](#_Toc470078805)

[1.2 Edit News 24](#_Toc470078806)

[1.3 View Draft 25](#_Toc470078807)

[1.4 Edit draft 27](#_Toc470078808)

[1.5 Delete Draft 29](#_Toc470078809)

[1.6 Transfer 31](#_Toc470078810)

[1.7 Approve 33](#_Toc470078811)

[1.8 Deactivate draft 35](#_Toc470078812)

[1.9 Search News 37](#_Toc470078813)

[1.10 Sort News 39](#_Toc470078814)

[1.11 Push News 40](#_Toc470078815)

# 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.

# Usecase Descriptions

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

|  |  |  |  |
| --- | --- | --- | --- |
| UC-MA-02 Log out | | | |
| Creator: | Hien Nguyen | Person Update: |  |
| Creation date: | 4/12/2015 | Date Update: |  |
| Description: | User log out system. | | |
| Users: | Admin, Editor, Education staff | | |
| Trigger: | User click button log out | | |
| Pre-Conditions: | User logged in system. | | |
| Post-Conditions: | Success: User log-out system successfully.  Fail: User can’t log out system. | | |
| Basic Flow: | 1. System alert dialog “Ok | Cancel” 2. User click “Ok” 3. System show message “Log out successfully”. 4. End of usecase. | | |
| Alternate Flow: | AF1: User 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 Database”. 2. System back to current page. | | |
| Priority | High | | |
| Frequency of use | Usually | | |
| Business Rules |  | | |
| Mock up |  | | |

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

|  |  |  |  |
| --- | --- | --- | --- |
| UC-MA-04 Edit Account | | | |
| Creator: | Hien Nguyen | Person Update: |  |
| Creation date: | 4/12/2016 | Date Update: |  |
| Description: | User edit information of their account | | |
| Users: | Admin, Editor, Education staff | | |
| Trigger: | User click button “Edit” | | |
| Pre-Conditions: | User must have an account in system. | | |
| Post-Conditions: | Success:User edit their account successfully.  Fail:User can’t edit their account. | | |
| Basic Flow: | 1. System show “edit account form”. 2. User edit information. 3. User click button “Save” 4. System show alert dialog “Ok|Cancel”. 5. User click “Ok”. 6. System check input data. 7. System update new information into database. 8. System notify “Edit successfully” 9. End of usecase. | | |
| Alternate Flow: | AF1: User cancel edit account: (after step 2)   1. User click button “Cancel”. 2. System close “edit account form”. | | |
| Exception Flows: | EF1: User input invalid information: (after step 6)   1. System notify “Invalid data. Please check again”. 2. System back to “edit an account form”   EF2: System can’t connect Database: (after step 7)  8.System notify “Can’t connect database”.  9.System back to “edit account form”. | | |
| Priority | High | | |
| Frequency of use | Sometimes | | |
| Business Rules | 1. Input fields have a limit under 100 characters. 2. Password must have censor. | | |
| Mock up |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| UC-MA-05 Search Account | | | |
| Creator: | Hien Nguyen | Person Update: |  |
| Creation date: | 4/12/2016 | Date Update: |  |
| Description: | Admin search account | | |
| Users: | Admin | | |
| Trigger: | Admin must in “manage account page” | | |
| Pre-Conditions: | User must be admin. | | |
| Post-Conditions: | Success:Admin search account successfully.  Fail:Admin can’t search account. | | |
| Basic Flow: | 1. User input a key search in search field. 2. User click button “Search” 3. System show accounts coincidence with key search. 4. End of usecase. | | |
| Alternate Flow: |  | | |
| Exception Flows: | EF1: System can’t connect Database: (after step 2)   1. System notify “Can’t connect database”. | | |
| Priority | High | | |
| Frequency of use | Often | | |
| Business Rules | Key search has a limit under 100 characters. | | |
| Mock up |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| UC-MA-06 Forget Password | | | |
| Creator: | Hien Nguyen | Person Update: |  |
| Creation date: | 4/12/2016 | Date Update: |  |
| Description: | User forget password and create password. | | |
| Users: | Admin, Editor, Education staff. | | |
| Trigger: | User click button “Forget password” | | |
| Pre-Conditions: | User must have an account in system. | | |
| Post-Conditions: | Success:User get new password successfully.  Fail:User can’t get new password. | | |
| 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 new password to their email. 6. System notify “Please check your email” 7. User check their email to get password. 8. End of usecase. | | |
| Alternate Flow: | AF1: User cancel “forget password” (after step 2):   1. User click button “Cancel”. 2. System close “forget password form”. | | |
| Exception Flows: | EF1: User 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.System notify “Can’t connect database”.  6. System back to “forget password form”. | | |
| Priority | Medium | | |
| Frequency of use | Sometimes | | |
| Business Rules | 1. Input fields have a limit under 100 characters. | | |
| Mock up |  | | |

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

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

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

## Manage 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 | Actors view news in the website | | |
| Main success flow | 1. User choose news which need view 2. System show detail of that news 3. End of usecase | | |
| Sub flows |  | | |
| Alternative flows |  | | |
| Exception flows | EF 1: System can’t connect to database ( after step 1)  2.System notify “Can’t connect to database, please try a gain”  3.System move to 404 page | | |
| Trigger | User access website | | |
| Relationships |  | | |
| Pre-conditions | System must be connected internet. | | |
| Post-conditions | Successful: user can view a news. | | |
| Priority | High | | |
| 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 posts the news | | |
| Main success flow | 1. Admin clicks button “Post news” 2. System shows post news GUI 3. Admin input information. 4. Admin clicks button “Post” 5. System check input 6. System save data into database 7. System notify “Post successfully” 8. End of use-case | | |
| Sub flows |  | | |
| Alternative flows | AF1: Admin click button “Cancel” (after step 3)   1. System back into current page | | |
| Exception flows | EF1: News have empty field ( after step 5)  6. System notify: “Field must be full filled”.  7. Back to step 3  EX2: System can’t connect database ( after step 6)  7.System notify “Can’t connect database, please check connection”  8.System move to 404 page | | |
| Trigger |  | | |
| Relationships |  | | |
| Pre-conditions | Admin must accesses website  Admin must login which user’s Account | | |
| Post-conditions | Successful: Admin post a news successfully | | |
| Priority | High | | |
| Frequency of use | Usually | | |
| Business rules |  | | |
| Special requirement |  | | |

## Edit News

|  |  |  |  |
| --- | --- | --- | --- |
| Use case ID | UC-MN-03 | | |
| Name of use case | Edit News | | |
| Creator | Tai Nguyen | Person updated |  |
| Creation date | 4/12/2016 | Date updated |  |
| Primary actor | Admin, Editor, Education staff | | |
| Brief description | When the post is transferred and approved if the new have problem, actors will fix it. | | |
| Main success flow | 1. Admin, Editor, Education staff click button “Edit News” 2. System show edit news GUI 3. Admin, Editor, Education staff edit information. 4. Admin, Editor, Education staff click button “Save” 5. System check input 6. System update information into database 7. System notify “ Edit news successfully” | | |
| Sub flows |  | | |
| Alternative flows | AF1:Admin click button “ Cancel” (after step 3)  4.Admin click button “Cancel”  5.System save all information have edited  6.System back to “Manage news” page | | |
| Exception flows | EX1: News have empty field (after step 5)  6. System notify “ Field must be full filled”  7. System back focus into that field  EX2: 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 |  | | |
| Pre-conditions | Admin, Editor, Education staff must accesses website  Admin, Editor, Education staff must login which user’s Account | | |
| Post-conditions | Successful: Admin, Editor, Education staff can edit a news successfully | | |
| Priority | High | | |
| Frequency of use |  | | |
| Business rules |  | | |
| Special requirement |  | | |

## View Draft

|  |  |  |  |
| --- | --- | --- | --- |
| Use case ID | UC-MN-04 | | |
| 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 | Actors view draft of the post | | |
| Main success flow | 1. User choose draft which need view 2. System show detail of that draft 3. End of usecase | | |
| Sub flows |  | | |
| Alternative flows | N/A | | |
| Exception flows | EX1: System can’t connect to database (after step 1)  2. System notify “Can’t connect to database, please try again”  3. System move to page 404 | | |
| Trigger | Admin, Editor, Education staff must accesses website | | |
| Relationships |  | | |
| Pre-conditions | Admin, Editor, Education staff must accesses website  Admin, Editor, Education staff must login | | |
| 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-05 | | |
| 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 | | |
| 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 usecase. | | |
| 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 | EX1: System can’s connect database (after step 5)  6. System notify “Can’t connect database, please try again!”  7. System move to 404 page | | |
| Trigger |  | | |
| Relationships |  | | |
| Pre-conditions | Admin, Editor, Education staff must accesses website  Admin, Editor, Education staff must login | | |
| 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-06 | | |
| 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 | | |
| 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 usecase. | | |
| Sub flows |  | | |
| Alternative flows |  | | |
| Exception flows | EX1:System can’t connect to database(after step 5)  6. System notify”Can’t connect to database, pleasetry again!”  7. System move to 404 page | | |
| Trigger |  | | |
| Relationships |  | | |
| Pre-conditions | Admin, Editor, Education staff must accesses website  Admin, Editor, Education staff must login | | |
| 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-07 | | |
| 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 | Actors transfer the news to approve/edit | | |
| 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 usecase | | |
| 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 |  | | |
| Pre-conditions | Admin, Editor, Education staff must accesses website  Admin, Editor, Education staff must login | | |
| 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 | | |
| 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 usecase | | |
| Sub flows |  | | |
| Alternative flows |  | | |
| Exception flows | EF1: System can’t connect to database (after step 6)  7. System notify: “System can’t connect to database, please try again!”.  8. System move to 404 page | | |
| Trigger |  | | |
| Relationships |  | | |
| Pre-conditions | Admin must accesses website  Admin must login which user’s Account  Admin must receive message | | |
| Post-conditions | Successful: Admin can approve a draft. | | |
| Priority | High | | |
| Frequency of use |  | | |
| Business rules |  | | |
| Special requirement |  | | |

## Deactivate draft

|  |  |  |  |
| --- | --- | --- | --- |
| Use case ID | UC-MN-05 | | |
| Name of use case | Deactivate draft | | |
| Creator | Tai Nguyen | Person updated |  |
| Creation date | 4/12/2016 | Date updated |  |
| Primary actor | Admin | | |
| Brief description | Admin deactive | | |
| Main success flow | 1. Admin choose a post which need deactivate. 2. Admin Press deactivate button 3. System shows Successful notice | | |
| Sub flows |  | | |
| Alternative flows | Actors choose draft and close website | | |
| Exception flows | EX1: Can’t connect to database  3. System show a notice: “System can’t connect to database”.  4. back Main GUI | | |
| Trigger | Admin must accesses deactivate draft GUI | | |
| Relationships |  | | |
| Pre-conditions | Admin must accesses website  Admin must login which user’s Account | | |
| Post-conditions | Successful: Admin can deactivate a draft.  Fail: Admin couldn’t deactivate a draft. | | |
| Priority | High | | |
| Frequency of use |  | | |
| Business rules |  | | |
| Special requirement |  | | |

## Search News

|  |  |  |  |
| --- | --- | --- | --- |
| Use case ID | UC-MN-05 | | |
| 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 | | |
| 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 usecase | | |
| Sub flows |  | | |
| Alternative flows | AF1: Admin, Editor, Education staff, User input information but don’t click button “Search” (after step 1) | | |
| Exception flows | EF1: System can’t connect to database(after step 2)  3. System notify: “System can’t connect to database, please try again”.  4. System move to 404 page | | |
| Trigger |  | | |
| Relationships |  | | |
| Pre-conditions | Admin, Editor, Education staff, User must access website  System must be connected internet | | |
| 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-05 | | |
| 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 | Actors sort the new follow the alphabet, posted date. | | |
| 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 usecase | | |
| Sub flows |  | | |
| Alternative flows |  | | |
| Exception flows | EX1: System can’t connect to database  3. System notify: “System can’t connect to database, please try again”.  4. System move to 404 page | | |
| Trigger |  | | |
| Relationships |  | | |
| Pre-conditions | Admin, Editor, Education staff, User must accesses website  System must be connected internet | | |
| Post-conditions | Successful: Admin, Editor, Education staff, User can search information. | | |
| Priority | High | | |
| Frequency of use |  | | |
| Business rules |  | | |
| Special requirement |  | | |

## Push News

|  |  |  |  |
| --- | --- | --- | --- |
| Use case ID | UC-MN-05 | | |
| 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 | | |
| 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 usecase | | |
| Sub flows |  | | |
| Alternative flows |  | | |
| Exception flows | EX1: System can’t connect to database(after step 2)  3. System notify: “System can’t connect to database, please try again!”.  4. System move to 404 page | | |
| Trigger |  | | |
| Relationships |  | | |
| Pre-conditions | Admin must accesses website  Admin must login | | |
| Post-conditions | Successful: Admin can push news. | | |
| Priority | High | | |
| Frequency of use |  | | |
| Business rules |  | | |
| Special requirement |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 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: | * Admin System clicks “Delete” button. | | |
| Relationships: | N/A | | |
| Priority: | High | | |
| Frequency of use: |  | | |
| Business Rules: |  | | |
| Special Requirement: | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 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 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 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 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 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 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 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: User cancels import point (After step 4)   1. User clicks “Cancel” button. 2. System shows “Manage Result” GUI. | | |
| Exception flows: |  | | |
| Trigger: | * User clicks “Import Point”. | | |
| Relationships: | N/A | | |
| Priority: | High | | |
| Frequency of use: |  | | |
| Business Rules: |  | | |
| Special Requirement: | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 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. * There is candidate profile. * 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: | * User clicks “Export”. | | |
| Relationships: | N/A | | |
| Priority: | Medium | | |
| Frequency of use: |  | | |
| Business Rules: |  | | |
| Special Requirement: | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 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. Website views "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: Website shows "Home Page" GUI. | | |
| Priority: | High | | |
| Frequency of use: |  | | |
| Business Rules: |  | | |
| Special Requirement: | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 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 System | | |
| Brief description: | Admin System 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: Website shows “Manage Categories” interface with list of categories. | | |
| Main Success flow: | 1. Admin System clicks "Manage Categories". 2. Website views "Manage Categories" GUI with the list of categories. 3. End of Use case. | | |
| Alternative flows: |  | | |
| Exception flows: |  | | |
| Trigger: | * Admin System clicks “Manage Categories” | | |
| Relationships: | N/A | | |
| Priority: | High | | |
| Frequency of use: |  | | |
| Business Rules: |  | | |
| Special Requirement: | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 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 System | | |
| Brief description: | Admin System adds new category to categories bar. | | |
| Pre-conditions: | Network status is stable.  User must log in website by admin account and being in the CMS GUI. | | |
| Post-conditions: | * Successful: Add new category successfully. | | |
| Main Success flow: | 1. Admin System clicks "Manage Categories". 2. Website views "Manage Categories" GUI. 3. Admin System clicks "Add" button. 4. Website views "Add Category" GUI. 5. Admin System fills information of category. 6. Admin System clicks “OK” 7. Website checks information of category 8. Website saves and views list of categories. 9. End of Use case. | | |
| Alternative flows: |  | | |
| Exception flows: |  | | |
| Trigger: | * Admin System clicks “Add” button. | | |
| Relationships: | N/A | | |
| Priority: | High | | |
| Frequency of use: |  | | |
| Business Rules: |  | | |
| Special Requirement: | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 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 System | | |
| Brief description: | Admin System edits information of category. | | |
| Pre-conditions: | * Network status is stable. * User must log in website by admin account and being in the CMS GUI. * Category is existing. | | |
| Post-conditions: | * Successful: Edit category successfully. | | |
| Main Success flow: | 1. Admin System clicks "Manage Categories". 2. Website views "Manage Categories" GUI. 3. Admin System chooses category and clicks "Edit" button. 4. Website views "Edit Category" GUI. 5. Admin System edits information of category. 6. Admin System clicks “OK” 7. Website checks information of category 8. Website saves new information of category. 9. End of Use case. | | |
| Alternative flows: |  | | |
| Exception flows: |  | | |
| Trigger: | * Admin System clicks “Edit” button. | | |
| Relationships: | N/A | | |
| Priority: | High | | |
| Frequency of use: |  | | |
| Business Rules: |  | | |
| Special Requirement: | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 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 System | | |
| Brief description: | Admin System deletes information of category. | | |
| Pre-conditions: | * Network status is stable. * User must log in website by admin account and being in the CMS GUI. * Category is existing. | | |
| Post-conditions: | * Successful: Delete category successfully. | | |
| Main Success flow: | 1. Admin System clicks "Manage Categories". 2. Website views "Manage Categories" GUI. 3. Admin System chooses category and clicks "Delete" button. 4. Website show message: “Delete Category?” 5. Admin System clicks “OK” 6. Website view list of categories. 7. End of Use case. | | |
| Alternative flows: |  | | |
| Exception flows: |  | | |
| Trigger: | * Admin System clicks “Delete” button. | | |
| Relationships: | N/A | | |
| Priority: | High | | |
| Frequency of use: |  | | |
| Business Rules: |  | | |
| Special Requirement: | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 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 | | |
| Main Success flow: | * + - 1. Admin, Editor, Education staff, User click “Change Languages”  1. System handler change. 2. System change language. 3. End of usecase. | | |
| Sub flows: | None | | |
| Alternative flows: | None | | |
| Exception flows: | None | | |
| Trigger: |  | | |
| Relationships: | N/A | | |
| Pre-conditions: | System must have connected with internet | | |
| Post-conditions: | * Successful: Change language successfully. | | |
| Priority: | Low | | |
| Frequency of use: | N/A | | |
| Business Rules: |  | | |
| Special Requirement: | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 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 usecase | | |
| 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: | N/A | | |
| Pre-conditions: | Admin must have logged in with user’s account | | |
| Post-conditions: | * Successful: Create new banner successfully. | | |
| Priority: | Medium | | |
| Frequency of use: | N/A | | |
| Business Rules: |  | | |
| Special Requirement: | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 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 | | |
| 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 usecase | | |
| Sub flows: | N/A | | |
| Alternative flows: | N/A | | |
| Exception flows: | N/A | | |
| Trigger: |  | | |
| Relationships: | N/A | | |
| Pre-conditions: | System must have connected with internet | | |
| Post-conditions: | Successful: View banner successfully. | | |
| Priority: | Medium | | |
| Frequency of use: | N/A | | |
| Business Rules: |  | | |
| Special Requirement: | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 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 | | |
| 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 usecase | | |
| Sub flows: | N/A | | |
| Alternative flows: |  | | |
| Exception flows: | Exception flow 1: 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: | N/A | | |
| Pre-conditions: | Admin must login  System must have banner | | |
| Post-conditions: | Successful: Show/hide banner successfully | | |
| Priority: | High | | |
| Frequency of use: | N/A | | |
| Business Rules: |  | | |
| Special Requirement: | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 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 | | |
| 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 usecase | | |
| 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: | N/A | | |
| Pre-conditions: | Admin must login with user’s account.  System must be connected internet | | |
| Post-conditions: | Successful: Edit banner successfully | | |
| Priority: | High | | |
| Frequency of use: | N/A | | |
| Business Rules: |  | | |
| Special Requirement: | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 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 | | |
| 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 usecase | | |
| 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: | N/A | | |
| Pre-conditions: | Admin must login  System must have banner  Admin inside “Manage banner” page | | |
| Post-conditions: | Successful: Delete banner successfully | | |
| Priority: | High | | |
| Frequency of use: | N/A | | |
| Business Rules: |  | | |
| Special Requirement: | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 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 | | |
| 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 usecase | | |
| 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: | N/A | | |
| Pre-conditions: | Admin inside “Manage banner” page  Admin must login  System must have banner | | |
| Post-conditions: | Successful: Arrange images successfully | | |
| Priority: | High | | |
| Frequency of use: | N/A | | |
| Business Rules: |  | | |
| Special Requirement: | N/A | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **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 usecase | | |
| 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 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **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 usecase. | | |
| 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: |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **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 usecase | | |
| 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 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **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 usecase | | |
| 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 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **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 usecase | | |
| 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 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **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 usecase | | |
| 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 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 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **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 usecase | | |
| Alternate Flow: | None | | |
| Exception Flows: | None | | |
| Relationships: | None | | |
| Business Rules: | None | | |
| Priority: | Medium | | |
| Frequency of use: | Medium | | |
| Special Requirements: | None | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **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 usecase | | |
| 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 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **UC-PU-04 : Edit Pop-ups** | | | |
| Creator: | Khoi Nguyen | Person Update: |  |
| Creation date: | 5/12/2016 | Date Update: |  |
| Brief Description: | Admin edits informations 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 usecase | | |
| 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 al 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 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **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 usecase | | |
| 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 | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **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 chooses images to arrange  4. Admin click button “Save”  5. System save information into database  6. System notify “Arrange successfully”  7. End of usecase | | |
| Alternate Flow: | AF1 : Admin cancel arrange pop-up ( after step 3 )  4. Admin click button “Cancel”  5. 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 | | |

# Usecase Diagram

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

Figure : Change Language

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

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

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

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

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

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

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