**24H Website**

**Development’s Guide**

**Faculty: Pham Cong Thanh**

**Student:**

|  |  |  |
| --- | --- | --- |
| **No** | **Full Name** | **Roll No** |
| **1** | **Duong Anh Tuan** | **SE62386** |
| 2 | Vo Xuan Minh Tuan | SE62347 |
| 3 | Truong Le Gia Luat | SE62310 |

**Table of contents**

Table of contents 2

**Review 1** 3

* 1. [Introduction](#_Introduction) 4
  2. [Existing Scenario](#_Existing_Scenario) 4
  3. [Customer Requirement Specification](#_Existing_Scenario) 4
  4. [Functional Requirement Specification](#_Functional_Requirement_Specificatio) 4
  5. [System Requirements](#_System_Requirements_1) 7
  6. [Development Software](#_Development_Software) 7

[Technology](#_Techonology) 7

**Review 2** 9

* 1. [Use Case and Sequence Diagram](#_Use_Case_and) .10
  2. Entity Relationship Diagram (ERD) xx

**Review 3** xx

* 1. Database Relationship Diagram xx
  2. Table Definition xx
  3. Other Constraint [xx](#_Table_Definition_1)
  4. Site map xx
  5. GUI Design xx
  6. Report xx

**Task sheet** xx

Task sheet xx

**Monitoring Report** xx

Unit Test xx

REVIEW 1

# Introduction

24H Online Advertising Joint Stock Company owns the Vietnam's leading online websites 24h.com.vn. Over 12 years of establishment and development, 24H now has more than 200 people working together.

# Existing Scenario

24h.com.vn is always one of largest electronic information sites in the country.

Today, along with the development of competitors, website 24h.com.vn is becoming obsolete. This leads to a decrease in the number of visitors and revenue.

So, directorate of the company decided to build a new system with a more modern and appropriate design.

# Customer Requirement Specification

The new system must have an account system which contain Customer and Employee.

Customer can be Reader account or Guest account (Not login customer). Customer can search and view articles easily. Reader can comment on article while Guest can’t.

Employee can be Journalist, Editor or Administrator. Journalist can create and modify article. Editor manage article, comment and FAQs. Administrator manage account.

System must provide functions to users for search article by title or category.

Graphic interface design must be modern and attractive.

# Functional Requirement Specification

## Guest can register as Reader

### Description

The system must provide a page to allow guest create their accounts with the following information:

* Username
* Password
* Name
* Birthday
* Gender
* Address
* Phone
* Email
* People Identity card

Username must satisfy:

* Contain only lower case and number characters.
* If user input upper cases in username, system automatically change them to lower case characters.

Password must satisfy:

* Upper case and lower case characters
* Number characters
* Special character such as: $#!
* Password length between 6-20 chars

Genders are:

* Male
* Female
* Other

Username, password, gender and email are required.

User’s type default is Reader.

User’s status default is Available.

### Functional Requirements

1. User must provide Username, Password, Name, Birthday, Gender, Address, Phone, and Email to create an account.
2. User’s type default is Reader.
3. Username is unique.
4. Email is unique.

## Guest can log in as Reader/Journalist/Editor

### Description

The system must provide a page to allow guest login as Reader/Journalist/Editor. The system must distinguish Reader, Journalist and Editor accounts.

### Functional Requirements

REQ-01. User must provide Username and Password to login into the website.

REQ-02. System must distinguish user type to display pages which user can view.

REQ-03. User can only access to pages they have permission.

## Users can modify their information

### Description

The system must provide a page to allow users view and modify their information:

* Name.
* Birthday.
* Gender.
* Address.
* Phone.

If user is Journalist, user can edit Press Card Number.

### Functional Requirements

REQ-01. Users can change their Password, Name, Birthday, Gender, Address and Phone.

REQ-02. Users must provide old password to update new password.

## Users can change their password

### Description

The system must provide a page to allow users change their password.

The page contain:

* Old password box
* New password box
* Confirm new password box

Users must provide old password to update new password.

### Functional Requirements

REQ-01. Users can change their Password.

REQ-02. Users must provide old password to update new password.

## Administrator can manage users

### Description

System must provide a page to allow Administrators manage user, search users, view user’s information, change user type, block user.

List of information which can be viewed by administrator:

* Name.
* Birthday.
* Gender.
* Address.
* Phone.

### Functional Requirements

REQ-01. Administrators can search account by username.

REQ-02. Administrators can view user’s information.

REQ-03. Administrators can block user.

REQ-04. Administrators can change type of other accounts to Reader/Journalist/Editor.

REQ-05. Administrators can set new Press Card Number for Journalist.

## Guests/Users can view articles which available

### Description

System must allow Guests/User view articles which available.

### Functional Requirements

1. System must allow Guests/User view articles which available.

## Guests/Users can search articles by title

### Description

The system must allow Guests/Users search articles by keyword, return available articles which title contain provided keyword.

### Functional Requirements

1. Guests/Users can search for articles by title.

## Journalist can create new article

### Description

System must provide a page which text editor to allow Journalist create and modify new article.

The article must have:

* Title
* Content
* Category
* Journalist’s information
* Created time

Categories are:

* Daily news
* Sport
* World
* Fashion
* Security - Social
* Hi-tech fashion
* Finance - real estate
* Food
* Beauty
* Showbiz
* Entertainment
* Young - life
* Education
* Car
* Bike – motorbike
* Market – Consumer
* Travel
* Health

Journalist’s information is:

* Name
* Press Card Number.

Journalist’s information and created time are added automatically using journalist information.

The text editor must allow journalists to:

* Input and edit text.
* Make bold/italic text.
* Insert image.

### Functional Requirements

1. Journalist can create new article.
2. Title and content must be not blank.
3. Category must be one of provided categories.
4. Journalist can modify article created by himself if article’s status is unavailable.
5. Journalist information is added automatically in the end of the article by using creator information.

## Editor can manage articles

### Description

Editor can view any articles (include unavailable articles), change article’s status from unavailable to available and opposite.

### Functional Requirements

1. Editor can view any articles.
2. Editor can change article’s status from unavailable to available and opposite.

## Users can comment on articles

### Description

System must provide comment function to allow user comment on articles.

Comment contain:

* Content
* Time

Time is added automatically using system time.

### Functional Requirements

1. User can comment on articles.
2. Time is added automatically using system time.

## Editors can manage comments

### Description

System must allow Editors hide/unhide comments.

### Functional Requirements

1. Editor can hide/unhide comments.

## Supporting Categories

### Description

System must provide list of categories. The Administrators can manage this list. Guests/Users can search articles by categories.

List of default categories (4.7.1).

### Functional Requirements

1. Administrators can manage directories, such as add new, modify, delete.
2. Guests/Users can search articles by categories.

## FAQs

### Description

System must provide to Customers a page which displays frequently questions and answers. Administrators can add, modify or delete these.

### Functional Requirements

1. Customers can view FAQs page.
2. Administrators can add, modify or delete these.

## GUI Requirements

### Description

System should provide a modern and attractive user interface.

### Functional Requirements

1. A modern and attractive user interface.

# System Requirements

## Hardware

### Web Server

|  |  |
| --- | --- |
| Processor | Core i5 IV or higher. |
| Memory | 8GB RAM or greater. |
| Modem/ADSL | Connecting to internet 24/24 |

### Client

|  |  |
| --- | --- |
| Processor | Pentium III or higher. |
| Memory | 512MB RAM or greater. |
| Monitor | Super VGA (800x600) or higher resolution with 256 colors. |
| Pointing Device | Microsoft Mouse, Microsoft IntelliMouse®, or compatible pointing device. |
| Modem/ADSL | Connecting to internet |

## Software

### Web Server

|  |  |
| --- | --- |
| Operation System | Windows 7/10. |
| Browser | Internet Explorer 6.0 or higher. |
| Database | Microsoft SQL Server 2012 |
| Software | JDK8. |
|  | Apache Tomcat 5.5 |
|  | ZK Framework 3.0 |

### Client

|  |  |
| --- | --- |
| Operation System | Windows XP or later. |
| Browser | Internet Explorer 6.0 or higher. |

# Development Software

* Microsoft Windows 10 Education
* Netbean IDE 8.0.2
* Java JDK 8
* Apache Tomcat 5.5
* Microsoft SQL Server 2012
* Chrome 60.0.3112.113 (64 bit)

# Technology

* Strut2 Framework
* Java EE 7
* Java Web Service
* JSP, Servlet
* HTML
* EJB
* Strut2 Framework
* Bootstrap 3
* jQuery 3.2.1

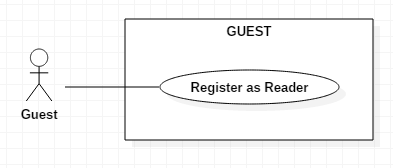
|  |  |  |
| --- | --- | --- |
|  | **Prepared by (Student)** | **Approved by (Faculty)** |
| **Signature** | Team leader  **Duong Anh Tuan** | **Pham Cong Thanh** |
| **Date** |  |  |

REVIEW 2

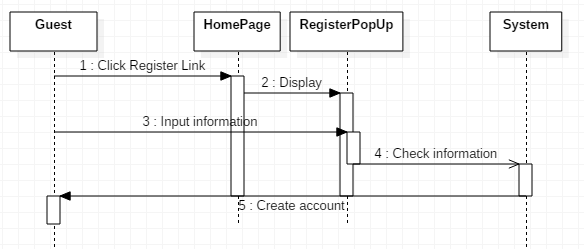
# Use Case and Sequence Diagram

***8.1. Guest***

**8.1.1. Register as Reader**

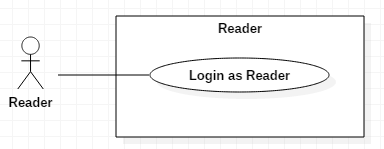


|  |  |  |
| --- | --- | --- |
| **Author** | Vo Xuan Minh Tuan | |
| **Use Case Name** | Register as Reader | |
| **Actors** | Guest | |
| **Description** | Guest register as a Reader user | |
| **Requirements** | Guest provides Guest’s information and accept user policy | |
| **Pre-conditions** |  | |
| **Post-conditions** | Success: Guest creates a Reader account in this website  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor clicks ‘Register’ link in Home page  3. Actor input information  5. Actor inputs register’s information, then check ‘Agree with user policy’ checkbox, finally click ‘Register’ button. [Alternative 1] | System Responses:  2. System show Register pop-up contain following Registration form with the following control :  - ‘Username’ text field  - ‘Password’ password field  - ‘Confirm Password’ text field  - ‘Name’ text field  - ‘Birthday’ date box  - ‘Gender’ checkbox group  - ‘Address’ text field  - ‘Phone’ text field  - ‘Email’ text field  - ‘Agree with user policy’  Checkbox  - ‘Policy’ text field  - ‘Register’ button  - Close pop-up button  4. System checks the information  [Exception 1]  6. System inserts the account into database, show message: ‘Your account has been created!’ and then redirects to Login page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click close register pop-up button | System Responses:  2. System closer register pop-up. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor lefts ‘Username’ text field blank.   1. Username has already exists.   3. Username doesn’t meet requirement (4.1.1)  4. Actor lefts ‘Password’ text field blank.  5. Actor lefts ‘Confirm Password’ text field blank.  6. Password doesn’t meet requirement (4.1.1).  7. Password and confirm password don’t match.  8. Actor lefts ‘Name’ text field blank.  9. Input age to ‘Birthday’ less than 16.  10. Input phone not in numeric or phone length not in range 8 – 14 character.   1. Actor lefts ‘Email’ text field blank or input an invalid email address.   12. Input people identity card not in numeric or its length not in range 8 – 14 character. | System Responses:  System displays message to Actor :  - ‘Username cannot be blank’.  - ‘This Username has already exist!’.  - ‘Username is invalid!’.  - ‘Password cannot be blank’.  - ‘Confirm Password can’t be blank’.  - ‘Password is invalid’  - ‘Confirm password not match password.’  - ‘Name cannot be blank’.  - ‘Age must over 16’.  - ‘The phone number should be numeric only and must be between 8 and 14 digits’.  - ‘Email is invalid’.  - ‘People identity card number is not invalid’. |

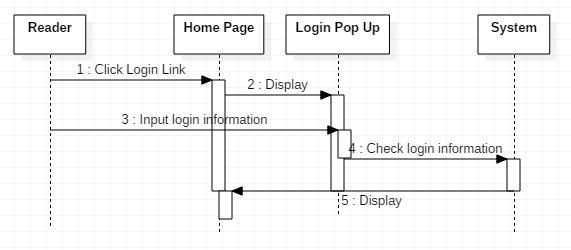


***8.2. Reader***

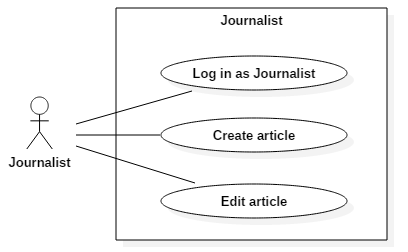
**8.2.1. Log in as reader**



|  |  |  |
| --- | --- | --- |
| **Author** | Vo Xuan Minh Tuan | |
| **Use Case Name** | Log in | |
| **Actors** | Reader | |
| **Description** | Guest login to website using reader account. | |
| **Requirements** | Actor provides username and password | |
| **Pre-conditions** |  | |
| **Post-conditions** | Success: User logged in to website as reader.  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor click ‘Log in’ link in the right-top of home page.  3. Actor inputs username and password, then click the Log in button. | System Responses:  2. System show register pop-up with Login form contain following control:  - ‘Username’ text field  - ‘Password’ password field  - ‘Log in’ button  - ‘Register’ link  - Close pop-up button  4. System checks the information  [Exception 1]  5. System redirects to Home page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click close pop-up button. | System Responses:  2. System close log in pop-up. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor input invalid username and password. | System Responses:  - System redirect to Home page with login pop-up with the following controls:  - ‘Username’ text field  - ‘Password’ text field  - ‘Login’ button.  - ‘Register’ link  - Close pop-up button  - Red-text notification: ‘Username or password is incorrect!” |

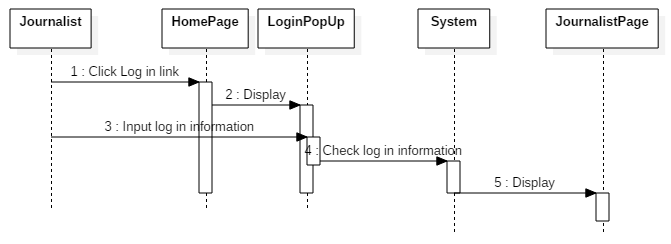


***8.3. Journalist***



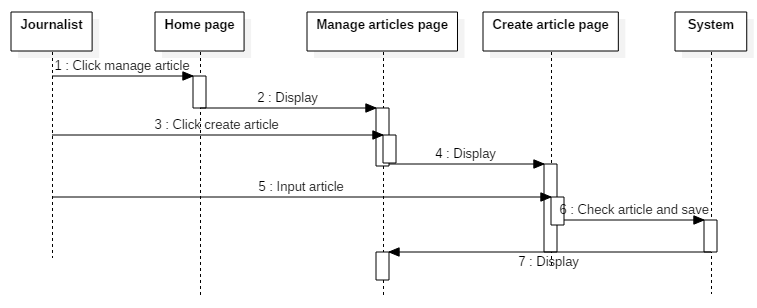
**8.3.1. Log in as Journalist**

|  |  |  |
| --- | --- | --- |
| **Author** | Vo Xuan Minh Tuan | |
| **Use Case Name** | Log in | |
| **Actors** | Journalist | |
| **Description** | Guest login to website using journalist account. | |
| **Requirements** | Actor provides username and password | |
| **Pre-conditions** |  | |
| **Post-conditions** | Success: User logged in to website as reader.  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor click ‘Log in’ link in the right-top of home page.  3. Actor inputs username and password, then click the Log in button. | System Responses:  2. System show register pop-up with Login form contain following control:  - ‘Username’ text field  - ‘Password’ password field  - ‘Log in’ button  - ‘Register’ link  - Close pop-up button  4. System checks the information  [Exception 1]  5. System redirects to ‘Manage articles: Journalist’ page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click close pop-up button. | System Responses:  2. System close log in pop-up. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor input invalid username and password. | System Responses:  - System redirect to Home page with login pop-up with the following controls:  - ‘Username’ text field  - ‘Password’ text field  - ‘Login’ button.  - ‘Register’ link  - Close pop-up button  - Red-text notification: ‘Username or password is incorrect!” |



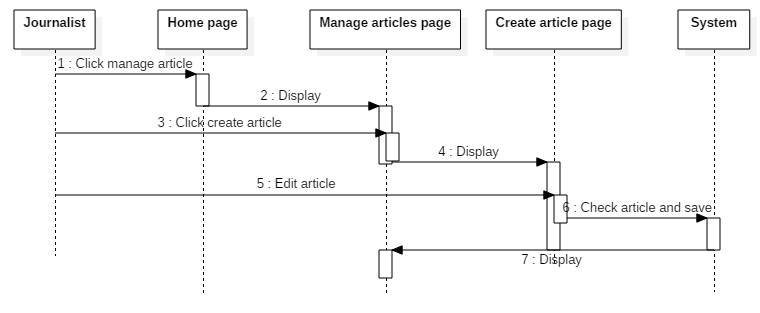
**8.3.2. Create new article**

|  |  |  |
| --- | --- | --- |
| **Author** | Vo Xuan Minh Tuan | |
| **Use Case Name** | Create new article | |
| **Actors** | Journalist | |
| **Description** | Journalist create new article. | |
| **Requirements** | Journalist must provide article title, content and select a type of categories. | |
| **Pre-conditions** | Actor must logged in as Journalist. | |
| **Post-conditions** | Success: New article is added to database.  Fail: Refill article | |
| **Basic flow** | Actor Actions:  1. Actor click ‘manage articles’ link at Home page.  3. Actor click ‘Create new article’ button.  5. Actor input article title, content. Then click ‘Save’ button.  [Alternative 1] | System Responses:  2. System redirect to ‘Manage articles’ page which contain:  - List of articles which are created by this user.  - ‘Edit’ button per unavailable article.  - 'Create new article’ button.   1. System redirect to ‘Create article’ page which contain:   - ‘Title’ text box.  - ‘Content’ text field with text-editor.  - ‘Category’ combo box.  - ‘Save’ button.  - ‘Cancel’ button.  6. System automatically add Journalist’s information using actor’s information and created time using system time. Save new article. Finally redirects to ‘Manage articles’ page.  [Exception 1] |
| **Alternative flow** | Actor Actions:  [Alternative 1]   1. Actor click cancel button.   3. Actor click ‘Yes’ button. | System Responses:   1. System show confirmation pop-up which contain:   - ‘Do you really want to cancel?’ notification.  - ‘Yes’ button.  - ‘No’ button.  4. System redirect to ‘Manage articles’ page. |
| **Exception** | Actor Actions:  [Exception 1]   1. Actor lefts ‘Title’ empty. 2. Actor lefts ‘Content’ empty. 3. Actor don’t select category. | System Responses:  System display message to actor:  - ‘Title can’t be blank’.  - ‘Content can’t be blank’.  - ‘Select one category’. |

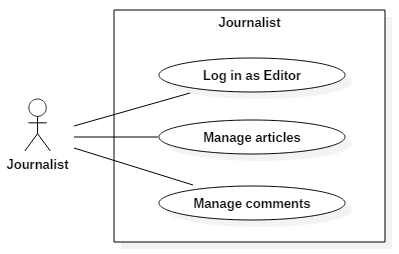


**8.3.3. Edit articles**

|  |  |  |
| --- | --- | --- |
| **Author** | Vo Xuan Minh Tuan | |
| **Use Case Name** | Edit article | |
| **Actors** | Journalist | |
| **Description** | Journalist edit a unavailable article. | |
| **Requirements** | Journalist must provide new article title, content and select a type of categories. | |
| **Pre-conditions** | Actor must logged in as Journalist.  Article’s status must be unavailable. | |
| **Post-conditions** | Success: New article is updated to database.  Fail: Refill article | |
| **Basic flow** | Actor Actions:  1. Actor click ‘manage articles’ link at Home page.  3. Actor click ‘edit’ button.  5. Actor input new article title, content. Then click ‘Save’ button.  [Alternative 1] | System Responses:  2. System redirect to ‘Manage articles’ page which contain:  - List of articles which are created by this user.  - ‘Edit’ button per unavailable article.  - 'Create new article’ button.   1. System redirect to ‘Edit article’ page which contain:   - ‘Title’ text box.  - ‘Content’ text field with text-editor.  - ‘Category’ combo box.  - ‘Save’ button.  - ‘Cancel’ button.  Tile, Content, Category have old value of article.  6. System automatically add Journalist’s information using actor’s information and created time using system time. Save new article. Finally redirects to ‘Manage articles’ page.  [Exception 1] |
| **Alternative flow** | Actor Actions:  [Alternative 1]   1. Actor click cancel button.   3. Actor click ‘Yes’ button. | System Responses:   1. System show confirmation pop-up which contain:   - ‘Do you really want to cancel?’ notification.  - ‘Yes’ button.  - ‘No’ button.  4. System redirect to ‘Manage articles’ page. |
| **Exception** | Actor Actions:  [Exception 1]   1. Actor lefts ‘Title’ empty. 2. Actor lefts ‘Content’ empty. 3. Actor don’t select category. | System Responses:  System display message to actor:  - ‘Title can’t be blank’.  - ‘Content can’t be blank’.  - ‘Select one category’. |

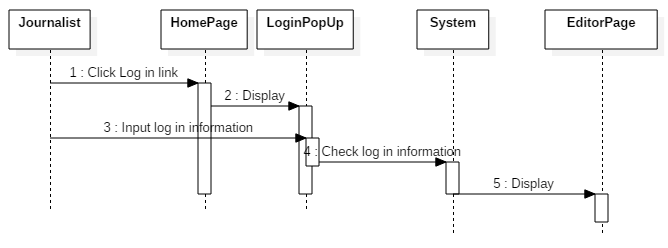


***8.4. Editor***

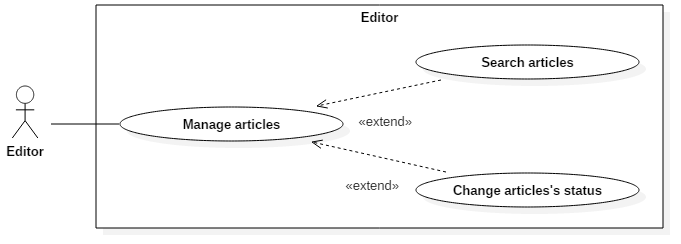


**8.4.1. Log in as Editor**

|  |  |  |
| --- | --- | --- |
| **Author** | Vo Xuan Minh Tuan | |
| **Use Case Name** | Log in | |
| **Actors** | Editor | |
| **Description** | Guest login to website using editor account. | |
| **Requirements** | Actor provides username and password | |
| **Pre-conditions** |  | |
| **Post-conditions** | Success: User logged in to website as editor.  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor click ‘Log in’ link in the right-top of home page.  3. Actor inputs username and password, then click the Log in button. | System Responses:  2. System show register pop-up with Log in form contain following control:  - ‘Username’ text field  - ‘Password’ password field  - ‘Log in’ button  - ‘Register’ link  - Close pop-up button  4. System checks the information  [Exception 1]  5. System redirects to ‘Manage articles: editor’ page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click close pop-up button. | System Responses:  2. System close log in pop-up. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor input invalid username and password. | System Responses:  - System redirect to Home page with login pop-up with the following controls:  - ‘Username’ text field  - ‘Password’ text field  - ‘Login’ button.  - ‘Register’ link  - Close pop-up button  - Red-text notification: ‘Username or password is incorrect!” |

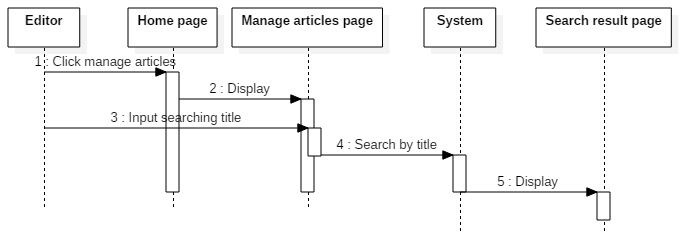


**8.4.2. Manage articles**



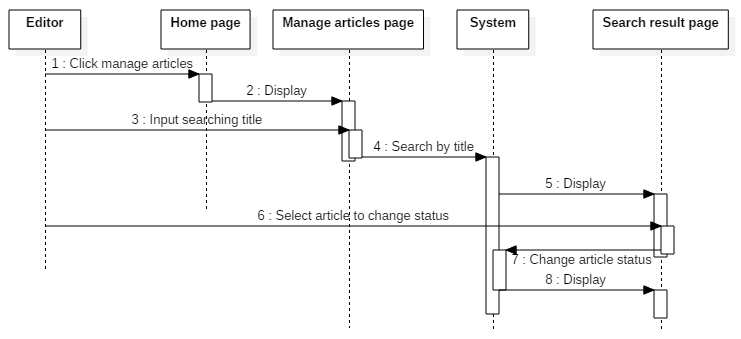
***8.4.2.1. Search articles***

|  |  |  |
| --- | --- | --- |
| **Author** | Vo Xuan Minh Tuan | |
| **Use Case Name** | Search articles | |
| **Actors** | Editor | |
| **Description** | Editor search articles by title. | |
| **Requirements** | Actor provides searching title. | |
| **Pre-conditions** | Actor logged in as Editor. | |
| **Post-conditions** | Success: System display list of articles which satisfy search condition. | |
| **Basic flow** | Actor Actions:  1. Actor click ‘Manage articles’ link at home page.  3. Actor inputs article title, then click the search button. | System Responses:  2. System show ‘Manage articles: editor’ which contain:  - ‘Search’ text box  - ‘Search’ button  - Top 50 ‘unavailable’ most recently created   1. System find list of articles which title contain searching title. If there are more than 100 articles. Show first 100 results.   Finally redirects to ‘Search result’ page contain:  - All articles title which satisfy search condition.  - ‘Set to available’ button each unavailable article.  - ‘Set to unavailable’ button each available article. |
| **Alternative flow** |  |  |
| **Exception** |  |  |

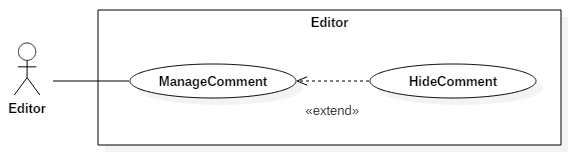


***8.4.2.2. Change article’s status***

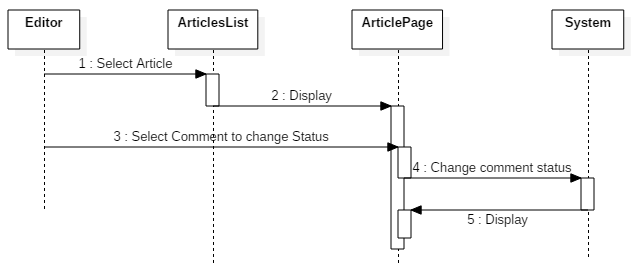
|  |  |  |
| --- | --- | --- |
| **Author** | Vo Xuan Minh Tuan | |
| **Use Case Name** | Search and change status of articles | |
| **Actors** | Editor | |
| **Description** | Editor search articles by title and change status of articles. | |
| **Requirements** | Actor provides searching title. | |
| **Pre-conditions** | Actor logged in as Editor. | |
| **Post-conditions** | Success: System change status of articles. | |
| **Basic flow** | Actor Actions:  1. Actor click ‘Manage articles’ link at home page.  3. Actor inputs article title, then click the search button.  5. Actor click change status button (Set to available/unavailable) | System Responses:  2. System show ‘Manage articles: editor’ which contain:  - ‘Search’ text box  - ‘Search’ button  - Top 50 ‘unavailable’ most recently created  4. System find list of articles which title contain searching title. If there are more than 100 articles. Show first 100 results.  Finally redirects to ‘Search result’ page contain:  - All articles title which satisfy search condition.  - ‘Set to available’ button each unavailable article.  - ‘Set to unavailable’ button each available article.  6. System change status of selected article, than update result page with changed article. |
| **Alternative flow** |  |  |
| **Exception** |  |  |



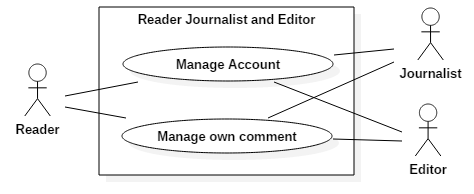
**8.4.3. Manage comments**



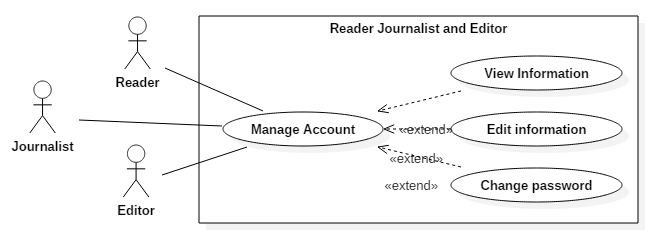
|  |  |  |
| --- | --- | --- |
| **Author** | Vo Xuan Minh Tuan | |
| **Use Case Name** | Editor change status of comments | |
| **Actors** | Editor | |
| **Description** | Editor change status of other’s comments | |
| **Requirements** |  | |
| **Pre-conditions** | Actor logged in as Editor. | |
| **Post-conditions** | Success: System change comment’s status. | |
| **Basic flow** | Actor Actions:  1. Actor access to an article from any articles list (Home page/Search Result page/Editor home page).  3. Actor click change status button (hide/unhide) | System Responses:  2. System show article page contain:  - Article’s title  - Article’s content  - Article’s category  - List of comment (include hiding comment)  + ‘Hide’ button on each comment which is not hidden.  + ‘Unhide’ button on each comment which is hidden.  - ‘Comment’ text area.  - ‘Send comment’ button.  4. System change status of selected comment, then update article page with status-changed comment. |
| **Alternative flow** |  |  |
| **Exception** |  |  |



***8.5. Reader/Journalist/Editor***

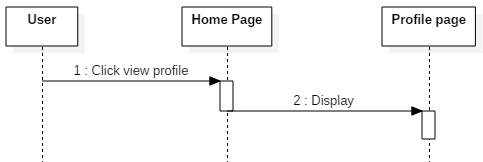


**8.5.1. Manage account**



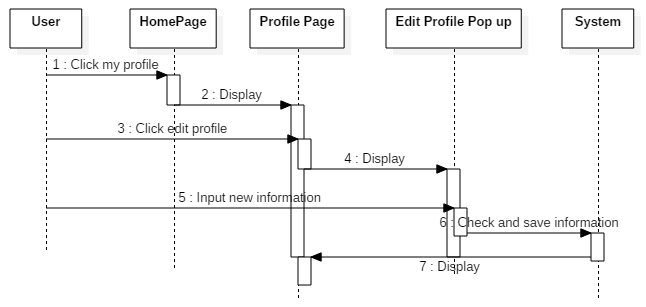
**8.5.1.1. View information**

|  |  |  |
| --- | --- | --- |
| **Author** | Vo Xuan Minh Tuan | |
| **Use Case Name** | Reader/Journalist/Editor view their own information | |
| **Actors** | Reader/Journalist/Editor | |
| **Description** | Reader/Journalist/Editor view their own information | |
| **Requirements** |  | |
| **Pre-conditions** | Actor logged in as Reader/Journalist/Editor. | |
| **Post-conditions** | Success: User’s profile is showed. | |
| **Basic flow** | Actor Actions:  1. Actor click ‘My profile’ link at Home Page. | System Responses:   1. System redirect to ‘My profile’ page which contain:   - Name  - Birthday  - Gender  - Address  - Phone  - Email  - People Identity card  - Edit profile button  - Change password button |
| **Alternative flow** |  |  |
| **Exception** |  |  |



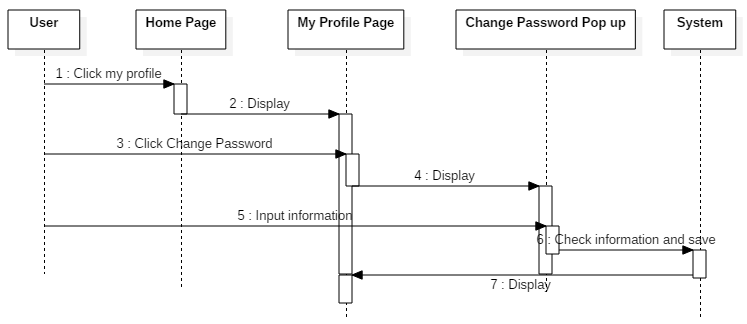
**8.5.1.2. Edit information**

|  |  |  |
| --- | --- | --- |
| **Author** | Vo Xuan Minh Tuan | |
| **Use Case Name** | Reader/Journalist/Editor edit their own information | |
| **Actors** | Reader/Journalist/Editor | |
| **Description** | Reader/Journalist/Editor view their own information and than edit | |
| **Requirements** |  | |
| **Pre-conditions** | Actor logged in as Reader/Journalist/Editor. | |
| **Post-conditions** | Success: System change account’s profile.  Fail: refill information | |
| **Basic flow** | Actor Actions:   1. Actor click ‘My profile’ link at Home Page. 2. Actor click edit profile button.   5. Actor input new information then click Save. [Alternative 1] | System Responses:  2. System redirect to ‘My profile’ page which contain:  - Name  - Birthday  - Gender  - Address  - Phone  - Email  - People Identity card  - Edit profile button  - Change password button   1. System show edit profile pop-up which contain:   - ‘Name’ text box  - ‘Birthday’ date box  - ‘Gender’ combo box  - ‘Address’ text box  - ‘Phone’ text box  - ‘Email’ text box  - ‘People Identity card’ text box  - ‘Save’ button  - ‘Cancel’ button  All fields are default filled by old information.  6. System checks new information. Then saves to database and redirects to My profile page with new information. [Exception 1] |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click cancel button. | System Responses:  2. System close edit profile pop-up. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor lefts ‘Name’ text field blank.  2. Input age to ‘Birthday’ less than 16.  3. Input phone not in numeric or phone length not in range 8 – 14 character.  4. Actor lefts ‘Email’ text field blank or input an invalid email address.  5. Input people identity card not in numeric or its length not in range 8 – 14 character. | System Responses:  System displays message to Actor :  - ‘Name cannot be blank’.  - ‘Age must over 16’.  - ‘The phone number should be numeric only and must be between 8 and 14 digits’.  - ‘Email is invalid’.  - ‘People identity card number is not invalid’. |

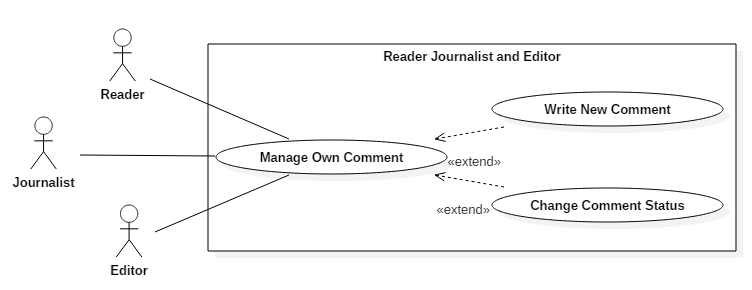


**8.5.1.3. Change password**

|  |  |  |
| --- | --- | --- |
| **Author** | Vo Xuan Minh Tuan | |
| **Use Case Name** | Reader/Journalist/Editor change their password | |
| **Actors** | Reader/Journalist/Editor | |
| **Description** | Reader/Journalist/Editor access to profile page and change their password | |
| **Requirements** |  | |
| **Pre-conditions** | Actor logged in as Reader/Journalist/Editor. | |
| **Post-conditions** | Success: System change account’s password.  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor click ‘My profile’ link at Home Page.  3. Actor click change password button.  5. Actor input information then click Save. [Alternative 1] | System Responses:  2. System redirect to ‘My profile’ page which contain:  - Name  - Birthday  - Gender  - Address  - Phone  - Email  - People Identity card  - Edit profile button  - Change password button  4. System show change password pop-up which contain:  - ‘Old password’ password field  - ‘New password’ password field  - ‘Confirm password’ field  - ‘Save’ button  - ‘Cancel’ button  6. System checks information. Then saves to database and close change password pop-up and show success notification pop-up. [Exception 1] |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click cancel button. | System Responses:  2. System close edit profile pop-up. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor lefts ‘Old password’ text field blank.  2. Actor lefts ‘New password’ text field blank.  3. Actor lefts ‘Confirm password’ text field blank.  4. Old password is incorrect  3. New password isn’t satisfy requirement (4.1.1)  4. New password doesn’t match Confirm password | System Responses:  System displays message to Actor :  - ‘This field can’t be blank’  - ‘This field can’t be blank’  - ‘This field can’t be blank’  - ‘Password is incorrect’  - ‘Password isn’t valid’  - ‘Confirm password isn’t match’ |

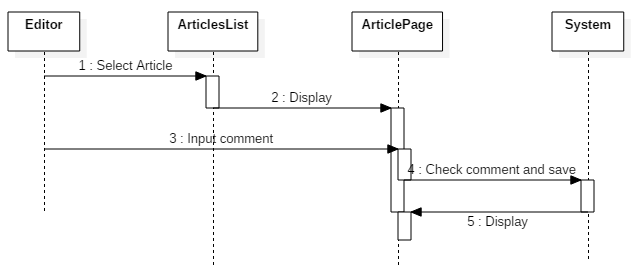


**8.5.2. Manage own comments on articles**



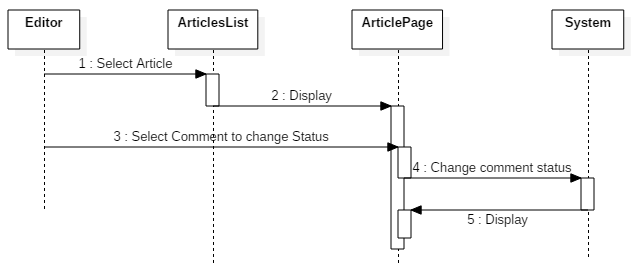
**8.5.2.1. Write new comment**

|  |  |  |
| --- | --- | --- |
| **Author** | Vo Xuan Minh Tuan | |
| **Use Case Name** | Users change status of their own comments | |
| **Actors** | Editor | |
| **Description** | User view their comment in article and change status | |
| **Requirements** |  | |
| **Pre-conditions** | Actor logged in as Reader/Journalist/Editor. | |
| **Post-conditions** | Success: System change comment’s status. | |
| **Basic flow** | Actor Actions:  1. Actor access to an article from any articles list (Home page/Search Result page/Editor home page).  3. Actor input comment and click send. | System Responses:  2. System show article page contain:  - Article’s title  - Article’s content  - Article’s category  - List of comment (include hiding comment)  + ‘Hide’ button each not hidden comment.  + ‘Unhide’ button each hidden comment.  ‘Hide/unhide’ button is only available for user’s own comment if user’s role is Reader or Journalist.  - ‘Comment’ text area.  - ‘Send comment’ button.  4. System validate comment, add commentator information using actor information and time using system time. Then save comment and re-display article. [Exception 1] |
| **Alternative flow** |  |  |
| **Exception** | Actor actions:  [Exception 1]  1. Comment’s content is empty | System response:  - Display message: ‘Comment can’t be blank!’ |

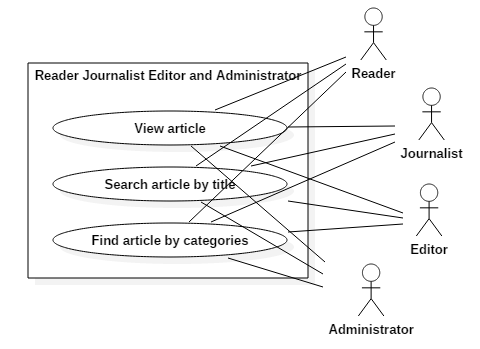


**8.5.2.2. Change comment status**

|  |  |  |
| --- | --- | --- |
| **Author** | Vo Xuan Minh Tuan | |
| **Use Case Name** | Users change status of their own comments | |
| **Actors** | Editor | |
| **Description** | User view their comment in article and change status | |
| **Requirements** |  | |
| **Pre-conditions** | Actor logged in as Reader/Journalist/Editor. | |
| **Post-conditions** | Success: System change comment’s status. | |
| **Basic flow** | Actor Actions:  1. Actor access to an article from any articles list (Home page/Search Result page/Editor home page).  3. Actor click change status button (hide/unhide) | System Responses:  2. System show article page contain:  - Article’s title  - Article’s content  - Article’s category  - List of comment (include hiding comment)  + ‘Hide’ button each not hiding comment.  + ‘Unhide’ button each hiding comment.  ‘Hide/unhide’ button is only available for user’s own comment if user’s role is Reader or Journalist.  - ‘Comment’ text area.  - ‘Send comment’ button.  4. System change status of selected comment, then update article page with status-changed comment. |
| **Alternative flow** |  |  |
| **Exception** |  |  |

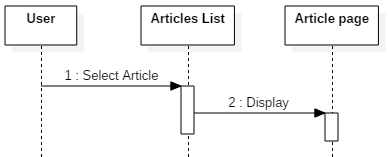


***8.6. Guest/Reader/Journalist/Editor***



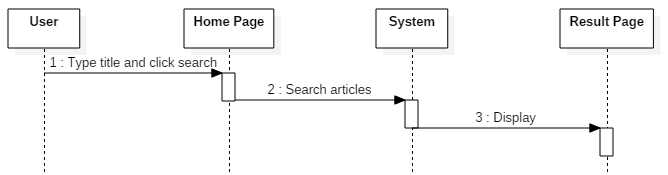
**8.6.1. View articles**

|  |  |  |
| --- | --- | --- |
| **Author** | Vo Xuan Minh Tuan | |
| **Use Case Name** | Users view an article | |
| **Actors** | Guest/Reader/Journalist/Editor | |
| **Description** | Users view an article | |
| **Requirements** |  | |
| **Pre-conditions** |  | |
| **Post-conditions** | Success: System display the article. | |
| **Basic flow** | Actor Actions:  1. Actor access to an article from any articles list (Home page/Search Result page/Editor home page). | System Responses:  2. System show article page contain:  - Article’s title  - Article’s content  - Article’s category  - List of comment (include hiding comment)  + ‘Hide’ button each not hiding comment.  + ‘Unhide’ button each hiding comment.  ‘Hide/unhide’ button is only available for user’s own comment if user’s role is Reader or Journalist. And don’t visible for guest.  - ‘Comment’ text area.  - ‘Send comment’ button.  Comment area and send comment button is only visible for logged in users. |
| **Alternative flow** |  |  |
| **Exception** |  |  |



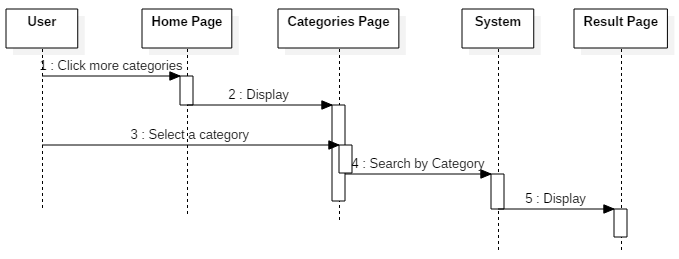
**8.6.2. Search articles by title**

|  |  |  |
| --- | --- | --- |
| **Author** | Vo Xuan Minh Tuan | |
| **Use Case Name** | Users search articles by title | |
| **Actors** | Guest/Reader/Journalist/Editor | |
| **Description** | Users input title then system search and display articles satisfy search condition | |
| **Requirements** | Users input non empty searching title | |
| **Pre-conditions** |  | |
| **Post-conditions** | Success: System display list of results. | |
| **Basic flow** | Actor Actions:  1. Actor input title in search box at Home Page. Than click Search button or press enter. | System Responses:   1. System find all articles which title contain searching title. Redirects to Search Result Page which contain:   - Result list: List of articles which title contain searching title.  - If result list have more than 50 items, show most recently 50 results.  - If no article found, show message: “No article found!”  - ‘Next’ button to show next 50 results.  - ‘Prev’ button to show last 50 results. |
| **Alternative flow** |  |  |
| **Exception** |  |  |

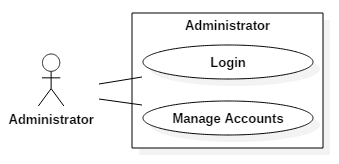


**8.6.3. Find articles by category**

|  |  |  |
| --- | --- | --- |
| **Author** | Vo Xuan Minh Tuan | |
| **Use Case Name** | Users search articles using categories | |
| **Actors** | Guest/Reader/Journalist/Editor | |
| **Description** | Users select a category than system display articles of that category | |
| **Requirements** | Users must select a category | |
| **Pre-conditions** |  | |
| **Post-conditions** | Success: System display list of articles which category is same with the selected category. | |
| **Basic flow** | Actor Actions:   1. Actor click ‘More categories’ in the Home Page. [Alternative 1]   3. Actor select a category. | System Responses:   1. System show Pop-up contains list of available categories. 2. System find all articles which category is the same with the selected category. Then redirects to result page contain:   - List of result.  - If result list have more than 50 items, show most recently 50 results.  - ‘Next’ button to show next 50 results.  - ‘Prev’ button to show last 50 results. |
| **Alternative flow** | Actor actions:  [Alternative 1]  1. Actor select a category which is available in the Home Page. | System response:  2. Move to 4th step of Basic flow. |
| **Exception** |  |  |

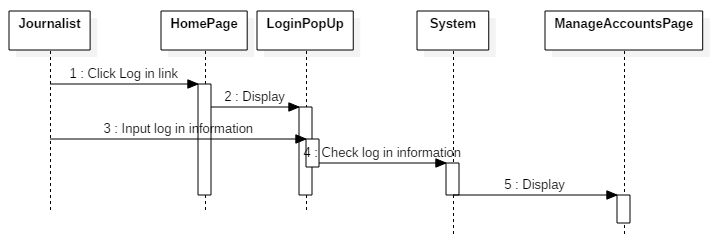


***8.7. Administrator***

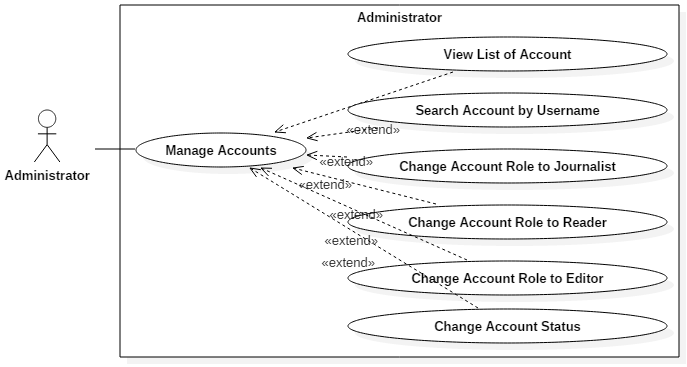


**8.7.1. Log in as administrator**

|  |  |  |
| --- | --- | --- |
| **Author** | Vo Xuan Minh Tuan | |
| **Use Case Name** | Log in | |
| **Actors** | Administrator | |
| **Description** | Guest login to website using Administrator account. | |
| **Requirements** | Actor provides username and password | |
| **Pre-conditions** |  | |
| **Post-conditions** | Success: User logged in to website as reader.  Fail: Refill information | |
| **Basic flow** | Actor Actions:  1. Actor click ‘Log in’ link in the right-top of home page.  3. Actor inputs username and password, then click the Log in button. | System Responses:  2. System show register pop-up with Log in form contain following control:  - ‘Username’ text field  - ‘Password’ password field  - ‘Log in’ button  - ‘Register’ link  - Close pop-up button  4. System checks the information  [Exception 1]  5. System redirects to ‘Admin’ page. |
| **Alternative flow** | Actor Actions:  [Alternative 1]  1. Actor click close pop-up button. | System Responses:  2. System close log in pop-up. |
| **Exception** | Actor Actions:  [Exception 1]  1. Actor input invalid username and password. | System Responses:  - System redirect to Home page with login pop-up with the following controls:  - ‘Username’ text field  - ‘Password’ text field  - ‘Login’ button.  - ‘Register’ link  - Close pop-up button  - Red-text notification: ‘Username or password is incorrect!” |

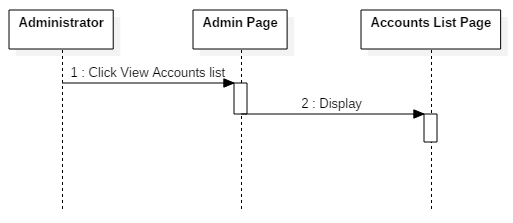


**8.7.2. Manage accounts**



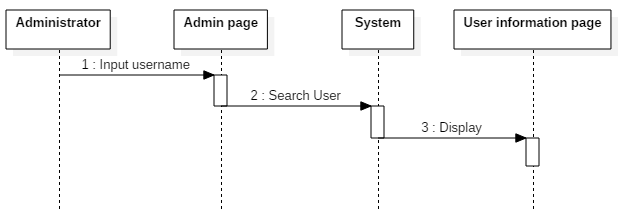
**8.7.2.1. View accounts list**

|  |  |  |
| --- | --- | --- |
| **Author** | Vo Xuan Minh Tuan | |
| **Use case name** | View List of accounts | |
| **Actors** | Administrator | |
| **Description** | View List of accounts | |
| **Requirements** |  | |
| **Pre-conditions** | Actor has already logged into the System as Administrator. | |
| **Post-conditions** | Success: View List of accounts | |
| **Basic flow** | Actor actions:  1. From ‘Admin’ page, actor clicks on to the ‘View accounts list’ button. | System response:  2. System redirects to ‘Accounts list’ page and display list of all of accounts in table form:   * No. * Username * Name * Role * Status |
| **Alternative flow** |  |  |
| **Exception** |  |  |



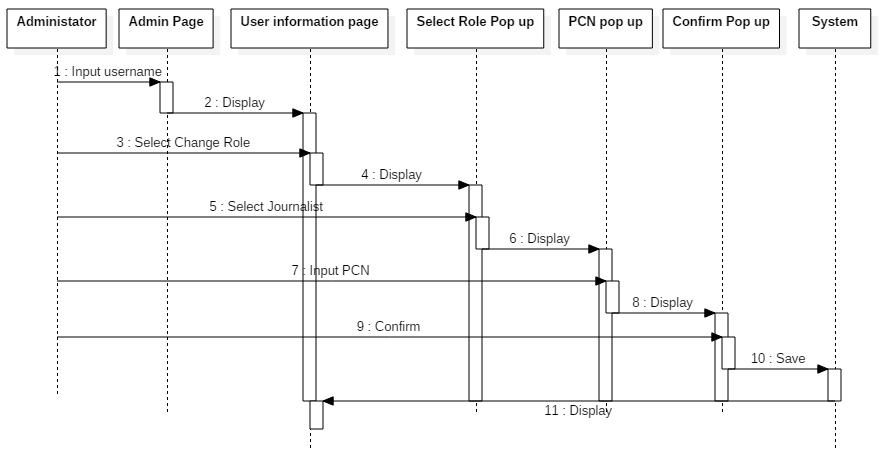
**8.7.2.2. Search account by username**

|  |  |  |
| --- | --- | --- |
| **Author** | Vo Xuan Minh Tuan | |
| **Use case name** | View List of accounts | |
| **Actors** | Administrator | |
| **Description** | View List of accounts | |
| **Requirements** |  | |
| **Pre-conditions** | Actor has already logged into the System as Administrator. | |
| **Post-conditions** | Success: View List of accounts | |
| **Basic flow** | Actor actions:  1. From ‘Admin’ page, actor types Username in user searching text box and click Search User button. | System response:  System find information of user which that username. Redirects to user information page which following items:  - Username  - Birthday  - Gender  - Address  - Phone  - Email  - People Identity card  - Role  - Press card number (if user’s role is Journalist)  - Status  - Change status button  - Change status button  - Back button  [Exception 1] |
| **Alternative flow** |  |  |
| **Exception** | Actor actions:  1. Actor input an unexist username. | Systems response:  - Show message: ‘Username is not exist!’. |



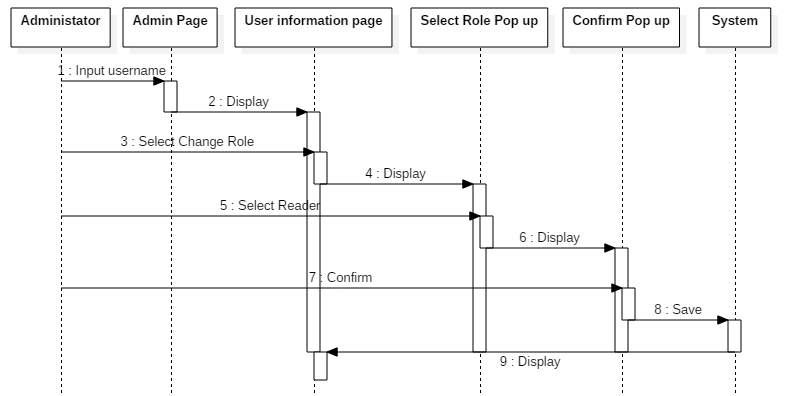
**8.7.2.3. Change account’s role to Journalist**

|  |  |  |
| --- | --- | --- |
| **Author** | Vo Xuan Minh Tuan | |
| **Use case name** | Change account’s role to Journalist | |
| **Actors** | Administrator | |
| **Description** | Search account by username and change role to Journalist. | |
| **Requirements** | Actor provide username. | |
| **Pre-conditions** | Actor has already logged into the System as Administrator. | |
| **Post-conditions** | Success: Account’s role is changed to Journalist. | |
| **Basic flow** | Actor actions:   1. From ‘Admin’ page, actor type Username in user searching text box and click Search User button. 2. Actor select Change role button.   [Alternative 1][Alternative 2]   1. Actor select ‘Journalist’ and click Save.   [Alternative 3][Alternative 4]   1. Actor input press card number and select Save. 2. Actor select yes.   [Alternative 5] | System response:  2. System find information of user which that username. Redirects to user information page which following items:  - Username  - Birthday  - Gender  - Address  - Phone  - Email  - People Identity card  - Role  - Press card number (if user’s role is Journalist)  - Status  - Change status button  - Change role button  - Back button  [Exception 1]   1. System show Change role pop-up contain:   - ‘Role list’ combo box.  - ‘Save’ button.  - ‘Cancel’ button.   1. System show Press card number (PCN) pop-up contains:   - ‘Press card number’ text box.  - ‘Save’ button.  - ‘Cancel’ button.   1. System show confirmation pop-up contain:   - ‘Are you sure to change role of ‘username’ to Journalist?’ message  - ‘Yes’ button  - ‘No’ button  10. System update role and redirect to User information page. |
| **Alternative flow** | Actor actions:  [Alternative 1]   1. Actor select Back button.   [Alternative 2]   1. Actor select change status button.   [Alternative 3]   1. Actor select Cancel.   [Alternative 4]   1. 1. Actor select other role.   [Alternative 4]  1. Actor select No. | System response:   1. System redirects to Admin page. 2. System show change status pop-up. 3. System close change role pop-up. 4. Show confirmation pop-up.   2. System close confirmation pop-up. |
| **Exception** | Actor actions:  1. Actor input an unexist username. | Systems response:  - Show message: ‘Username is not exist!’. |



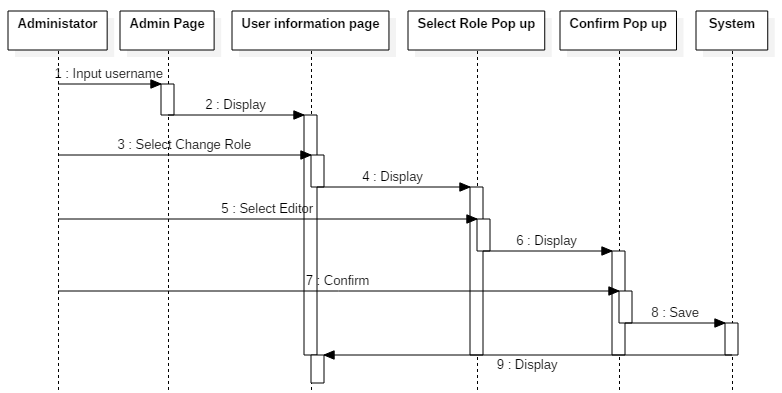
**8.7.2.4. Change account’s role to Reader**

|  |  |  |
| --- | --- | --- |
| **Author** | Vo Xuan Minh Tuan | |
| **Use case name** | Change account’s role to Reader | |
| **Actors** | Administrator | |
| **Description** | Search account by username and change role to Reader. | |
| **Requirements** | Actor provide username. | |
| **Pre-conditions** | Actor has already logged into the System as Administrator. | |
| **Post-conditions** | Success: Account’s role is changed to Reader. | |
| **Basic flow** | Actor actions:  1. From ‘Admin’ page, actor type Username in user searching text box and click Search User button.  3. Actor select Change role button.  [Alternative 1][Alternative 2]  5. Actor select ‘Reader’ and click Save.  [Alternative 3][Alternative 4]  7. Actor select yes.  [Alternative 5] | System response:  2. System find information of user which that username. Redirects to user information page which following items:  - Username  - Birthday  - Gender  - Address  - Phone  - Email  - People Identity card  - Role  - Press card number (if user’s role is Journalist)  - Status  - Change status button  - Change role button  - Back button  [Exception 1]  4. System show Change role pop-up contain:  - ‘Role list’ combo box.  - ‘Save’ button.  - ‘Cancel’ button.  6. System show confirmation pop-up contain:  - ‘Do you sure to change role of ‘username’ to Reader?’ message  - ‘Yes’ button  - ‘No’ button  8. System update role and redirect to User information page. |
| **Alternative flow** | Actor actions:  [Alternative 1]   1. Actor select Back button.   [Alternative 2]   1. Actor select change status button.   [Alternative 3]   1. Actor select Cancel.   [Alternative 4]   1. 1. Actor select other role.   [Alternative 4]  1. Actor select No. | System response:   1. System redirects to Admin page. 2. System show change status pop-up. 3. System close change role pop-up. 4. Show confirmation pop-up.   2. System close confirmation pop-up. |
| **Exception** | Actor actions:  1. Actor input an unexist username. | Systems response:  - Show message: ‘Username is not exist!’. |



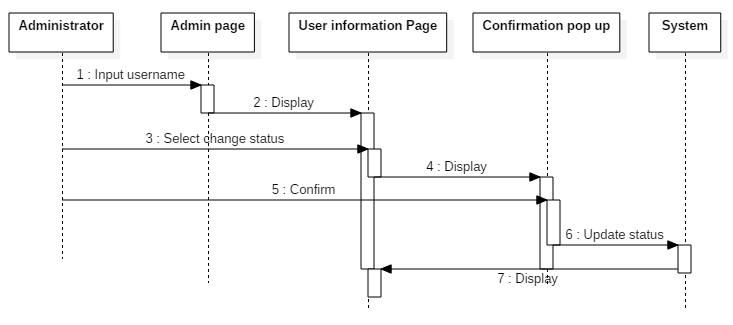
**8.7.2.5. Change account’s role to Editor**

|  |  |  |
| --- | --- | --- |
| **Author** | Vo Xuan Minh Tuan | |
| **Use case name** | Change account’s role to Editor | |
| **Actors** | Administrator | |
| **Description** | Search account by username and change role to Editor. | |
| **Requirements** | Actor provide username. | |
| **Pre-conditions** | Actor has already logged into the System as Administrator. | |
| **Post-conditions** | Success: Account’s role is changed to Reader. | |
| **Basic flow** | Actor actions:  1. From ‘Admin’ page, actor type Username in user searching text box and click Search User button.  3. Actor select Change role button.  [Alternative 1][Alternative 2]  5. Actor select ‘Editor’ and click Save.  [Alternative 3][Alternative 4]  7. Actor select yes.  [Alternative 5] | System response:  2. System find information of user which that username. Redirects to user information page which following items:  - Username  - Birthday  - Gender  - Address  - Phone  - Email  - People Identity card  - Role  - Press card number (if user’s role is Journalist)  - Status  - Change status button  - Change role button  - Back button  [Exception 1]  4. System show Change role pop-up contain:  - ‘Role list’ combo box.  - ‘Save’ button.  - ‘Cancel’ button.  6. System show confirmation pop-up contain:  - ‘Do you sure to change role of ‘username’ to Editor?’ message  - ‘Yes’ button  - ‘No’ button  8. System update role and redirect to User information page. |
| **Alternative flow** | Actor actions:  [Alternative 1]   1. Actor select Back button.   [Alternative 2]   1. Actor select change status button.   [Alternative 3]   1. Actor select Cancel.   [Alternative 4]   1. 1. Actor select other role.   [Alternative 4]  1. Actor select No. | System response:   1. System redirects to Admin page. 2. System show change status pop-up. 3. System close change role pop-up. 4. Show confirmation pop-up.   2. System close confirmation pop-up. |
| **Exception** | Actor actions:  1. Actor input an unexist username. | Systems response:  - Show message: ‘Username is not exist!’. |



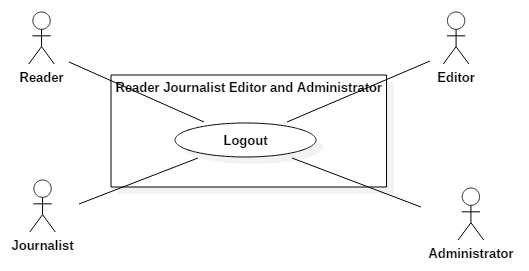
**8.7.2.6. Change account’s status**

|  |  |  |
| --- | --- | --- |
| **Author** | Vo Xuan Minh Tuan | |
| **Use case name** | Change account’s status | |
| **Actors** | Administrator | |
| **Description** | Search account by username and change account’s status | |
| **Requirements** | Administrator provide username. | |
| **Pre-conditions** | Actor has already logged into the System as Administrator. | |
| **Post-conditions** | Success: System change account’s status | |
| **Basic flow** | Actor actions:  1. From ‘Admin’ page, actor type Username in user searching text box and click Search User button.   1. Actor select Change status button.   [Alternative 1][Alternative 2]   1. Actor select Yes.   [Alternative 3] | System response:  2. System find information of user which that username. Redirects to user information page which following items:  - Username  - Birthday  - Gender  - Address  - Phone  - Email  - People Identity card  - Role  - Press card number (if user’s role is Journalist)  - Status  - Change status button  - Change role button  - Back button  [Exception 1]   1. System show confirmation pop-up:   - ‘Do you want to change status of “username” to “new status”?’  + ‘username’ is username of showing account.  + ‘new status’ is Unavailable if account’s status is Available and opposite.  - ‘Yes’ button  - ‘No’ button  5. System save new status and redirect to user information page. |
| **Alternative flow** | Actor actions:  [Alternative 1]   1. Actor select Back button.   [Alternative 2]   1. Actor select change role button.   [Alternative 3]  1. Actor select No. | System response:   1. System redirects to Admin page. 2. System show change role pop-up.   2. System close confirmation pop-up. |
| **Exception** | Actor actions:  1. Actor input an unexist username. | Systems response:  - Show message: ‘Username is not exist!’. |

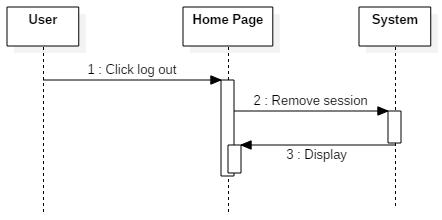


***8.9. Administrator/Reader/Journalist/Editor***

**8.8.1. Log out**



|  |  |  |
| --- | --- | --- |
| **Author** | Vo Xuan Minh Tuan | |
| **Use Case Name** | Log out | |
| **Actors** | Administrator/Reader/Journalist/Editor | |
| **Description** | Users log out their account | |
| **Requirements** |  | |
| **Pre-conditions** | Actor logged in. | |
| **Post-conditions** | Success: User logged out from website. | |
| **Basic flow** | Actor Actions:  1. Actor click ‘Log out’ link in the home page. | System Responses:  2. System redirect to log out page, remove session and redirects to Home Page. |
| **Alternative flow** |  |  |
| **Exception** |  |  |

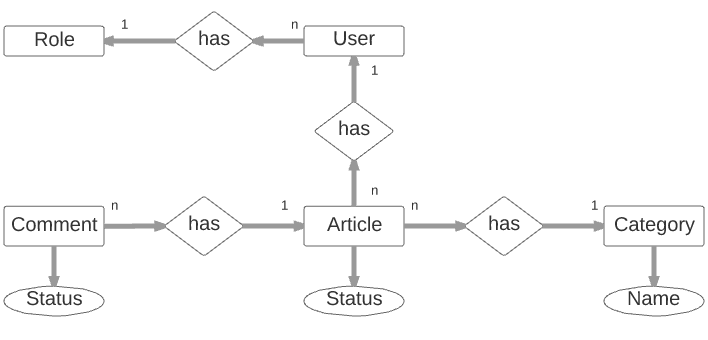


# Entity Relationship Diagram (ERD)

## Conceptual

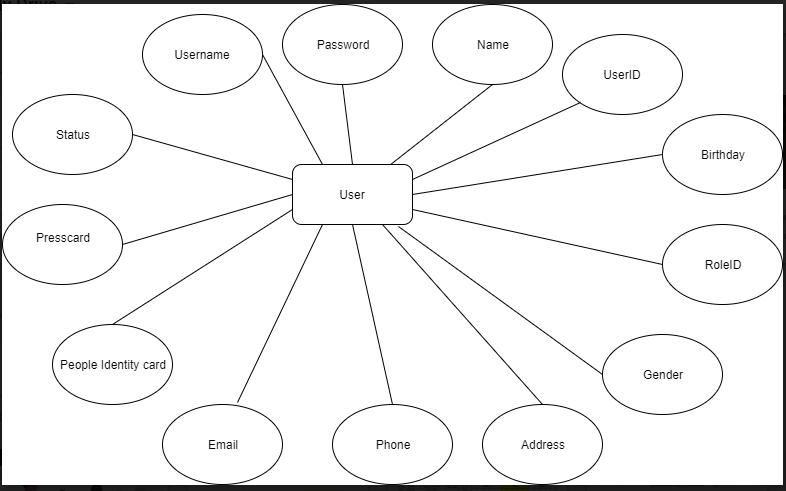
## 

## 9.2. Logical

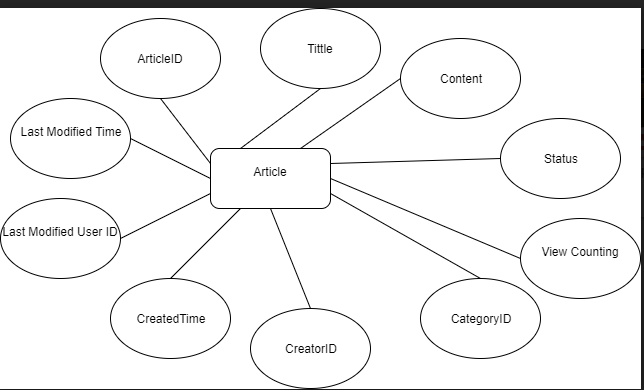


***9.3. Table***

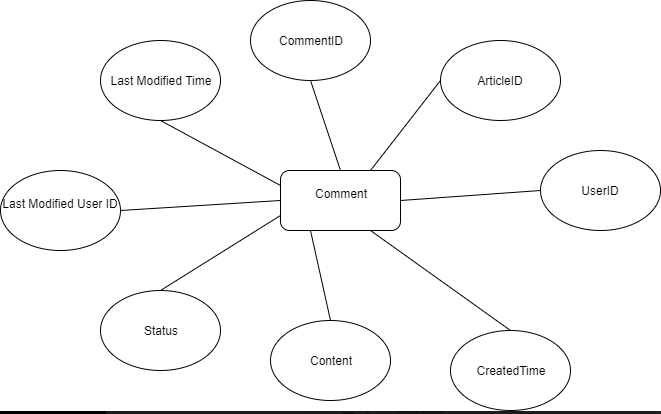
**9.3.1. User**

****

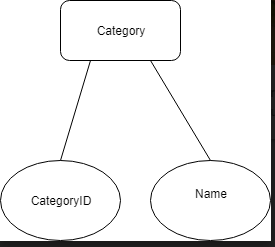
**9.3.2. Article**

****

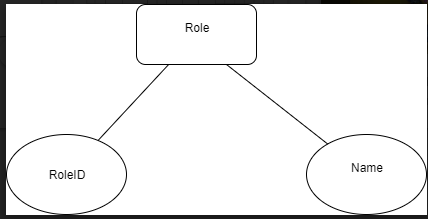
**9.3.3. Comment**

****

**9.3.4. Category**

****

**9.3.5. Role**

****

REVIEW 3

# 10. Database Relationship Diagram

# 22140183_1876889965670709_1219535866_o

# 11. Table Definition

# *11.1. User*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Key** | **Field Name** | **Data Type** | **Length** | **Allow Null** | **Default value** | **Table-Field-Ref** | **Description** |
| PK | **UserID** | int |  | No |  |  | User’s ID in database |
| Unique | Username | varchar | 45 | No |  |  | Username |
|  | Password | varchar | 200 | No |  |  | Password |
|  | Name | nvarchar | 45 | No |  |  | Name of the user |
|  | Birthday | date |  | No |  |  | Birthday of the user |
|  | Gender | int |  | No |  |  | Gender of the user:  - 0 for Female  - 1 for Male  - 2 for Others |
|  | Address | nvarchar | 200 | No |  |  | Address of the user |
|  | Phone | varchar | 20 | No |  |  | Phone of the user |
|  | Email | varchar | 45 | No |  |  | E-mail of the user |
|  | PeopleIdentityCard | varchar | 45 | No |  |  | People Identity Card Number of the user |
|  | PressCard | varchar | 45 | Yes |  |  | Press Card number of the user |
|  | RoleID | int |  | No | Role’s ID of “Reader” | Role.RoleID | Role’s ID of user’s role |
|  | Status | varchar | 45 | No |  |  | Status of the user’s account:  - Available  - Unavailable |

**Constraint**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| User | +  - Role ID cannot be RoleID of “Admin” (the system has only 1 administrator)  - Email must be in e-mail’s form ([something@something](mailto:something@something))  - Phone must be string of numeric and have length at least 9 | - | +  - Cannot update UserID, Username |

# *11.2. Article*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Key** | **Field Name** | **Data Type** | **Length** | **Allow Null** | **Default value** | **Table-Field-Ref** | **Description** |
| PK | **ArticleID** | int |  | No |  |  | Article’s ID in database |
|  | Tittle | nvarchar | 200 | No |  |  | Article’s Tittle |
|  | Content | nvarchar | MAX | No |  |  | Article’content URL |
|  | Status | varchar | 45 | No |  |  | Status of article:  - Available  - Unavailable |
|  | CategoryID | int |  | No |  | Category.CategoryID | Category’s ID of article’s category |
|  | CreatorID | int |  | No |  | User.UserID | User’s ID of article’s creator |
|  | CreatedTime | datetime | 200 | No |  |  | Time that article created |
|  | LastModifiedUserID | int |  | No |  | User.UserID | User’s ID of the user last modifying article |
|  | LastModifiedTime | datetime |  | No |  |  | Time of article’s last modified |
|  | ViewCounting | int |  | No | 0 |  | Number of view of article |
|  | Thumbnail | varchar | 50 | No |  |  | URL of article’s thumbnail |

**Constraint**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| Article | + | - | +  - Cannot update ArticleID, CreatorID, CreatedTime |

# *11.3. Comment*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Key** | **Field Name** | **Data Type** | **Length** | **Allow Null** | **Default value** | **Table-Field-Ref** | **Description** |
| PK | **CommentID** | int |  | No |  |  | Comment’s ID in database |
|  | ArticleID | int |  | No |  | Article.ArticleID | Article’s ID of the article contains comment |
|  | UserID | int |  | No |  | User.UserID | User’s ID of comment’s creator |
|  | CreatedTime | datetime |  | No |  |  | Time the comment created |
|  | Content | nvarchar | 200 | No |  |  | Category’s ID of article’s category |
|  | Status | varchar | 45 | No |  |  | Comment’s status :  - show  - hide |
|  | LastModifedUserID | int |  | No |  | User.UserID | User’s ID of the user last modifying comment |
|  | LastModifiedTime | datetime |  | No |  |  | Time of last modifying comment |

**Constraint**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| Comment | + | - | +  - Cannot update  CommentID, ArticleID, CreatorID, CreatedTime |

# *11.4. Role*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Key** | **Field Name** | **Data Type** | **Length** | **Allow Null** | **Default value** | **Table-Field-Ref** | **Description** |
| PK | **RoleID** | int |  | No |  |  | Role’s ID in database |
|  | Name | varchar | 45 | No |  |  | Role’s name |

**Constraint**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| Role | + | - | - |

# *11.5. Category*

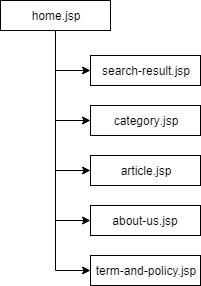
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Key** | **Field Name** | **Data Type** | **Length** | **Allow Null** | **Default value** | **Table-Field-Ref** | **Description** |
| PK | **CategoryID** | int |  | No |  |  | Category’s ID in database |
|  | Name | varchar | 45 | No |  |  | Category’s name |

**Constraint**

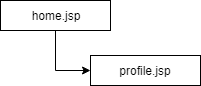
|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Add** | **Delete** | **Update** |
| Role | + | - | - |

# 12. Site map

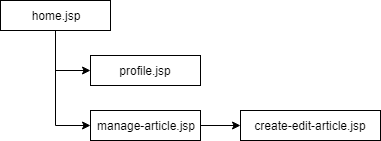
## Common Side

****

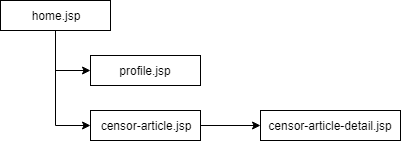
## Reader Side

****

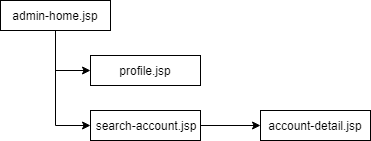
## Journalist Side

****

## Editor

****

## Admin

****

## 

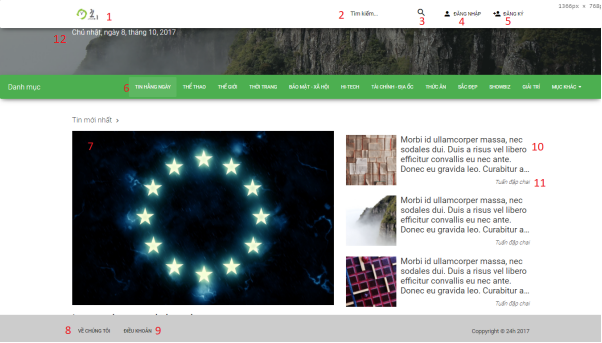
# GUI Design



## Common side

### Home page:

This is Home page of the website, the users. readers, journalist, editors will go to this page first.



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | Img24hLogo | Image |  | Click | Redirect to Home page |  |
| 2 | txtSearch | Text field | 0<length<50 |  | Search article by title |  |
| 3 | btnSearch | Button |  | Click | Redirect to Search result page |  |
| 4 | btnLogin | Button |  | Click | Show Login pop-up |  |
| 5 | btnRegister | Button |  | Click | Show Register pop-up |  |
| 6 | btnCategory | Button |  | Click | Redirect to Category page |  |
| 7 | imgPreviewArticle | Image |  | Click | Redirect to Article page |  |
| 8 | btnAboutUs | Button |  | Click | Redirect to About Us Page |  |
| 9 | btnTermAndPolicy | Button |  | Click | Redirect to Term and policy page |  |
| 10 | lbTitle | Label |  |  | Show title of the article |  |
| 11 | lbAuthor | Label |  |  | Show author of this article |  |
| 12 | lbDate | Label |  |  | Show current date |  |

# **Login pop-up**

# 

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | lbLogin | Label |  |  | Show label of this pop-up |  |
| 2 | btnCloseSmall | Button |  | Click | Close the pop-up without login |  |
| 3 | txtUsername | Text field | 0<length<=45 |  | Input Username |  |
| 4 | txtPassword | Password field | 0<length<=200 |  | Input Password |  |
| 5 | btnLogin | Button |  | Click | Login and redirect to Home page |  |
| 6 | btnClose | Button |  | Click | Close the pop-up without login |  |

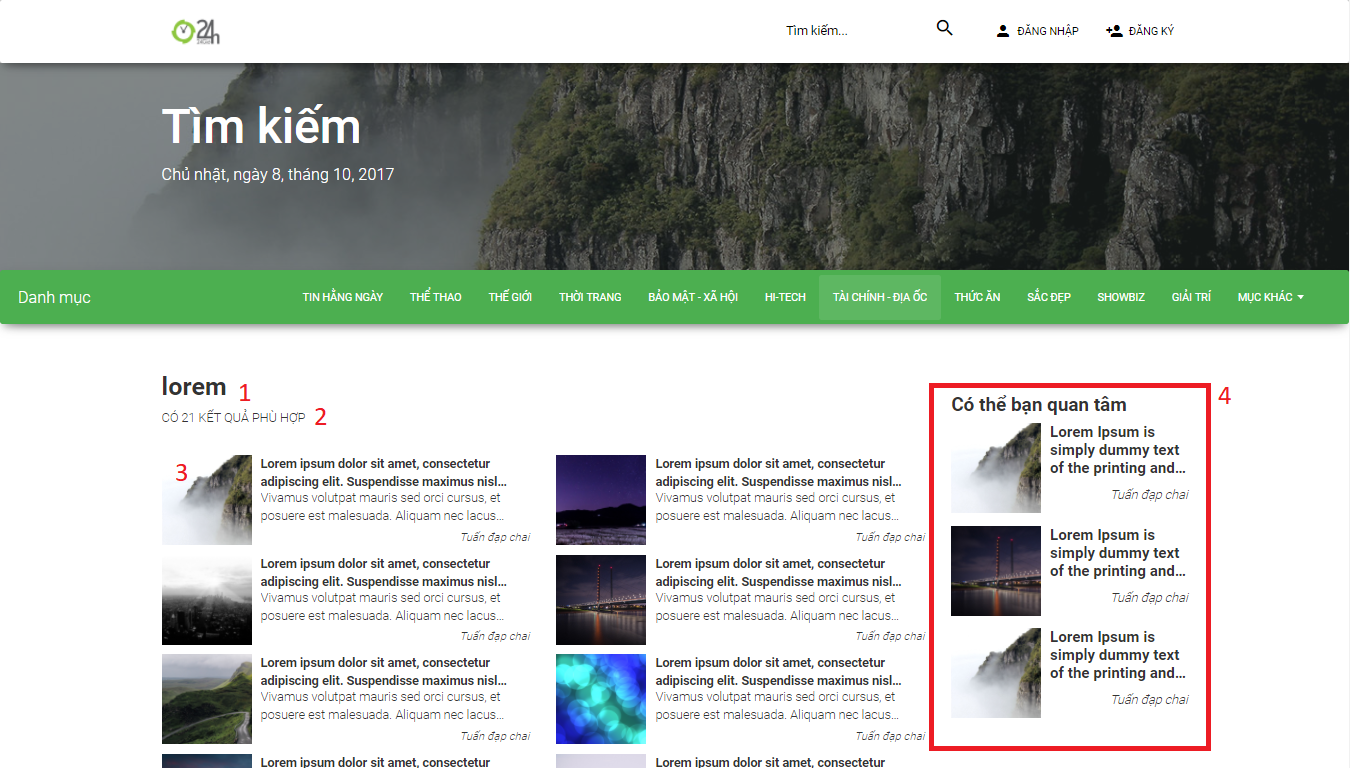
# **Register pop-up**

# 

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | lbRegister | Label |  |  | Show label of this pop-up |  |
| 2 | btnCloseSmall | Button |  | Click | Close the pop-up without login |  |
| 3 | txtUsername | Text field | 0<length<=45 |  | Username |  |
| 4 | txtName | Text field | 0<length<=45 |  | Name |  |
| 5 | txtPassword | Password field | 0<length<=200 |  | User’s Password |  |
| 6 | txtConfirmPassword | Password field | 0<length<=200 |  | Confirm password |  |
| 7 | dbBirthday | Date box | Not null  Date<=current year - 16 | Click | User’s Birthday |  |
| 8 | rbMale | Radio button |  | Click | Gender selection |  |
| 9 | rbFemale | Radio button |  | Click | Gender selection |  |
| 10 | txtAddress | Text field | 0<length<=200 |  | User’s Address |  |
| 11 | txtPhone | Text field | 8<length<=14  [0-9] |  | User’s Phone |  |
| 12 | txtEmail | Text field | 0<length<=30  /.+@.+\.[a-z]+/ |  | User’s Email |  |
| 13 | cbAgreeTerm | Check box | Not null | Click | Agree term and policy of page |  |
| 14 | lnkTermAndPolicy | Link |  | Click | Redirect to Term and policy page |  |
| 15 | btnRegister | Button |  | Click | Register and redirect to Home page |  |
| 16 | btnClose | Button |  | Click | Close pop-up |  |

### Search Result page:

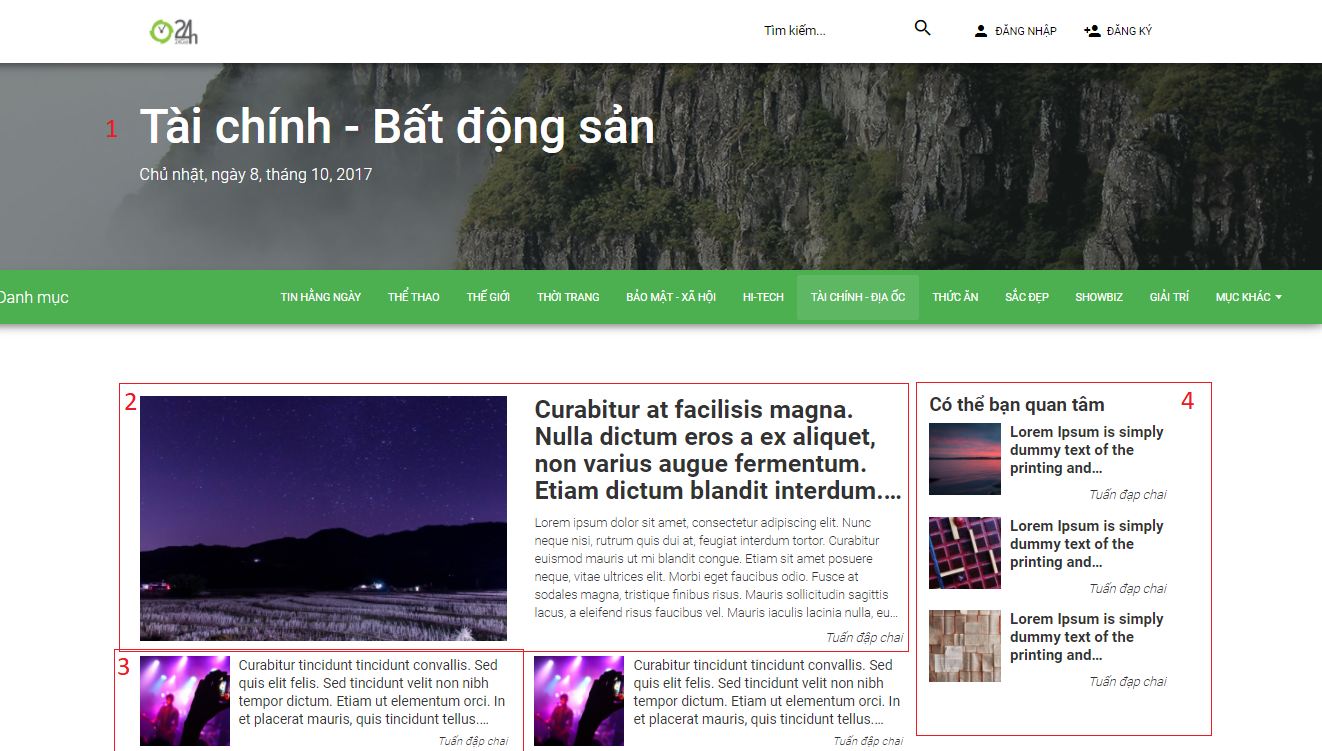
This is Search result page of the website. Show articles whose title likes search keyword.



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | lbSearchValue | Label |  |  | Last search value |  |
| 2 | lbResultCount | Label |  |  | Count result that match the search value |  |
| 3 | imgPreviewArticle | Image |  | Click | Redirect to Article Page |  |
| 4 | gridRandomArticle | Grid |  |  | Show 3 random articles in last 10 days |  |

### Category page:

This is Category page of the website. Show articles which belong to this category.

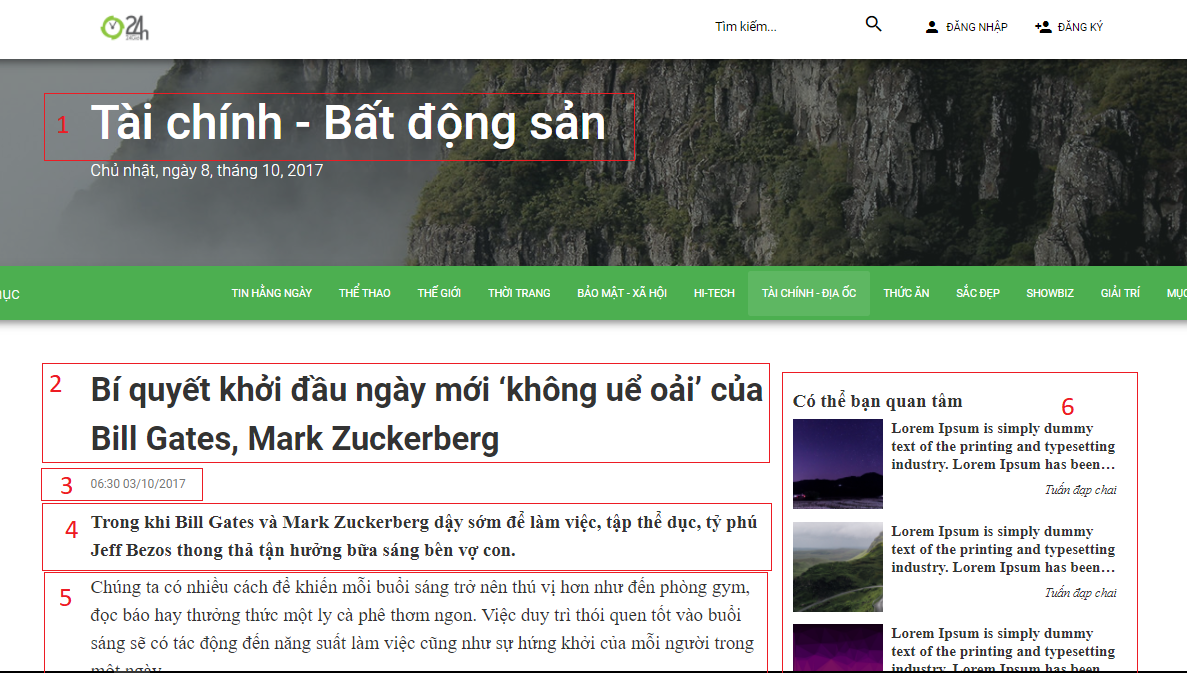


|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | lbCategoryName | Label |  |  | Name of this category |  |
| 2 | gridPreviewNewestArticle | Grid |  | Click | Redirect to Newest article |  |
| 3 | gridPreviewArticle | Grid |  | Click | Redirect to article |  |
| 4 | gridRandomArticle | Grid |  |  | Show 3 random articles in last 10 days |  |

### Article page:

### Article.

This is the article section. It shows article content.



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | lbCategoryName | Label |  |  | Name of category which contains this article |  |
| 2 | lbTitle | Label |  |  | Title of article |  |
| 3 | lbCreatedDate | Label |  |  | Last modify date of this article |  |
| 4 | lbMainContent | Label |  |  | Brief content of this article |  |
| 5 | lbContent | Label |  |  | Content of this article |  |
| 6 | gridRandomArticle | Grid |  |  | Show 3 random articles in last 10 days |  |

### Footer and Comment.

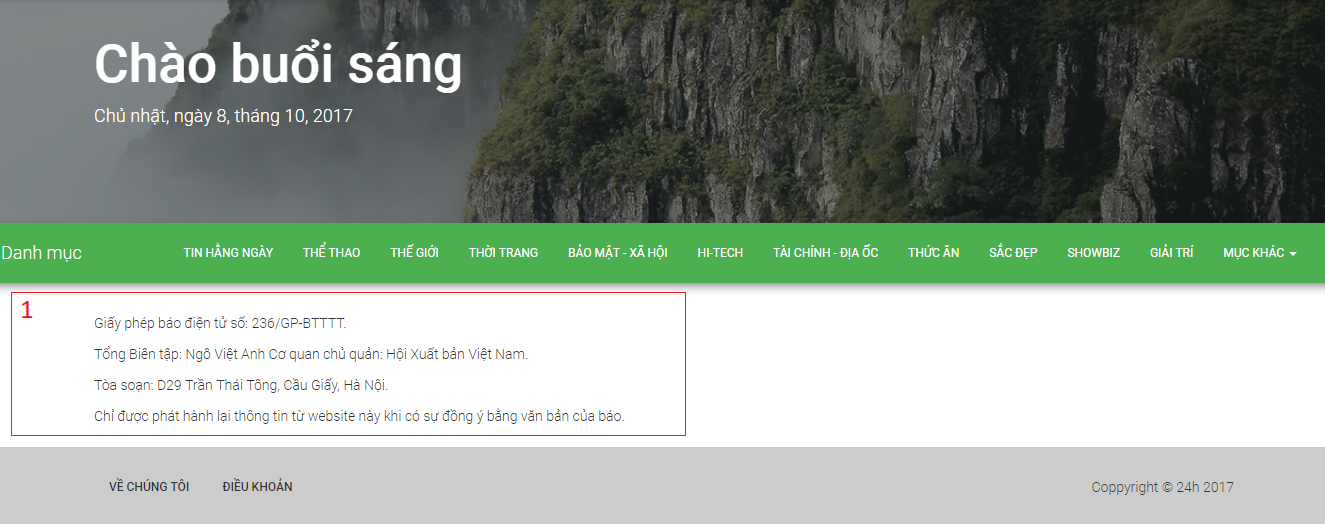
This is the footer and comment section in Article page. It shows author name, comments, and allow reader, editor, journalist, admin to comment.



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | lbAuthor | Label |  |  | Author of this article |  |
| 2 | imgAvatar | Image |  |  | User’s avatar. |  |
| 3 | lbFullname | Label |  |  | User’s fullname |  |
| 4 | lbCommentDate | Label |  |  | Comment date |  |
| 5 | lbCommentContent | Label |  |  | Content of this comment |  |
| 6 | txtComment | Text Area |  |  | Write new comment |  |
| 7 | btnPostComment | Button |  |  | Submit comment and reload this page. |  |

### Term and policy page:

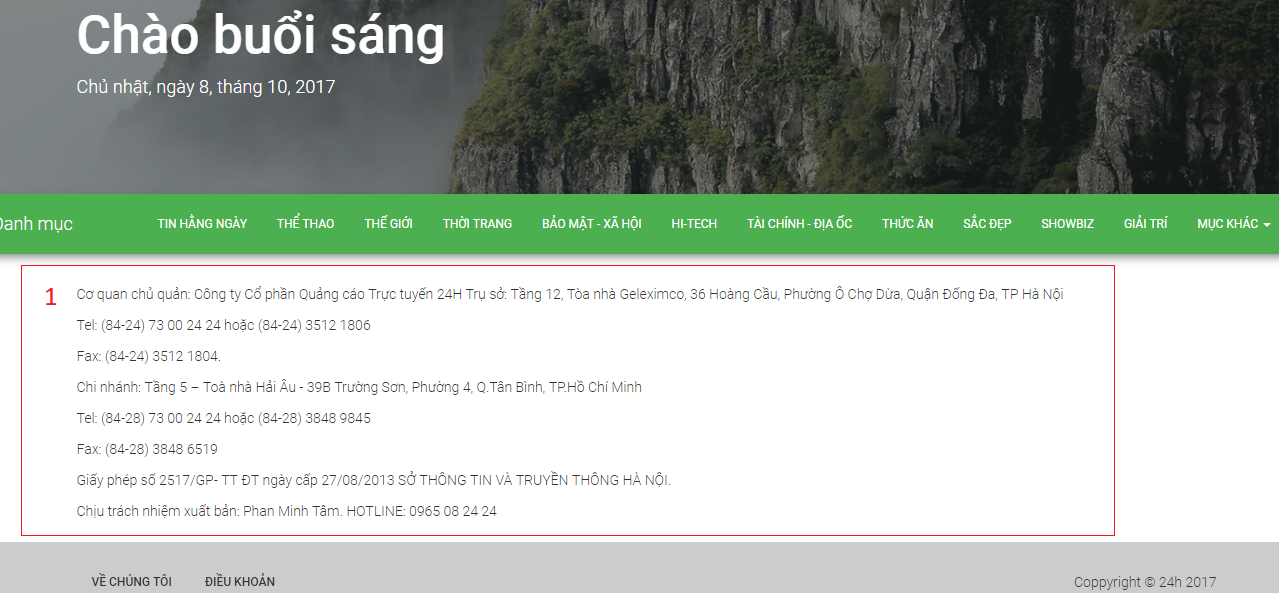
This is Term and policy page of the website. User have to agree with term in order to register.



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | lbContent | Label |  |  | Term and policy |  |

### About us page:

This is About us page of the website. This page containts information about 24h Company.

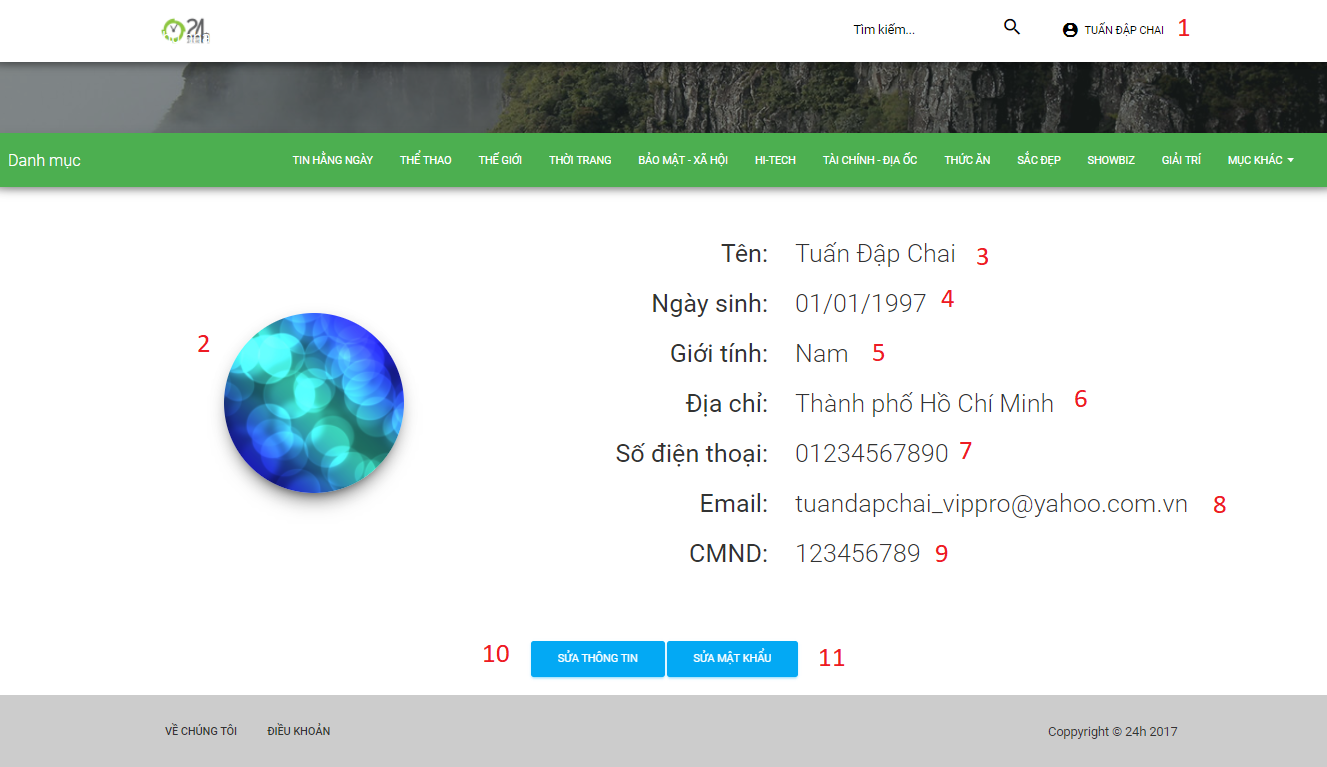


|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | lbContent | Label |  |  | Information about 24h |  |

## Reader side

### Profile page:

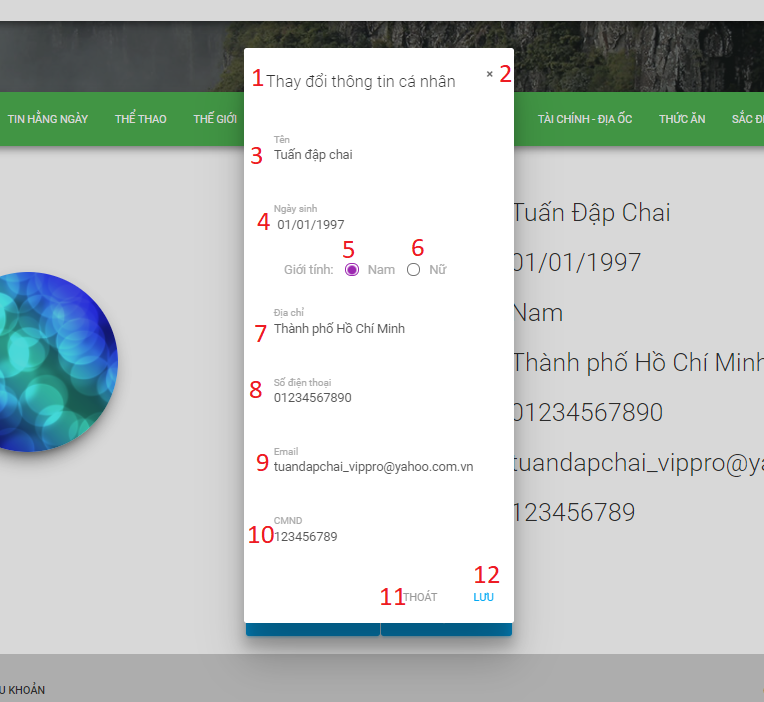
This is About us page of the website. This page containts information about 24h Company.



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | lbNavName | Label |  | Click | Reload this page |  |
| 2 | imgAvatar | Image |  |  | Avatar of this user |  |
| 3 | lbFullname | Label |  |  | Full name of user |  |
| 4 | lbBirthdate | Label |  |  | Birthdate of this user |  |
| 5 | lbGender | Label |  |  | Gender of this user |  |
| 6 | lbAddress | Label |  |  | Address of this user |  |
| 7 | lbPhone | Label |  |  | Phone number of this user |  |
| 8 | lbEmail | Label |  |  | Email of this user |  |
| 9 | lbIdenityNumber | Label |  |  | People Identity number |  |
| 10 | btnEditProfile | Button |  | Click | Show pop-up edit profile |  |
| 11 | btnEditPassword | Button |  | Click | Show pop-up edit password |  |

### Edit Profile Pop-up:

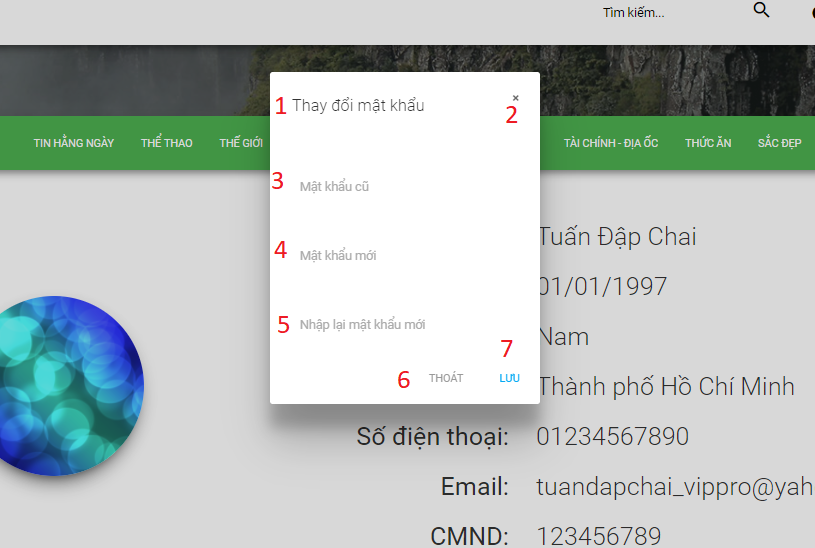
This is Edit profile pop-up allow reader, journalist, editor edit information.



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | lbRegister | Label |  |  | Show label of this pop-up |  |
| 2 | btnCloseSmall | Button |  | Click | Close the pop-up without login |  |
| 3 | txtName | Text field | 0<length<=45 |  | Name |  |
| 4 | dbBirthday | Date box | Not null  Date<=current year - 16 | Click | User’s Birthday |  |
| 5 | rbMale | Radio button |  | Click | Gender selection |  |
| 6 | rbFemale | Radio button |  | Click | Gender selection |  |
| 7 | txtAddress | Text field | 0<length<=200 |  | User’s Address |  |
| 8 | txtPhone | Text field | 8<length<=14  [0-9] |  | User’s Phone |  |
| 9 | txtEmail | Text field | 0<length<=30  /.+@.+\.[a-z]+/ |  | User’s Email |  |
| 10 | txtIdentityNumber | Text field | 9<=length<=12 |  | User’s People Identity number |  |
| 11 | btnRegister | Button |  | Click | Register and redirect to profile page |  |
| 12 | btnClose | Button |  | Click | Close pop-up |  |

### Edit Profile Pop-up:

This is Edit profile pop-up allow reader, journalist, editor edit information.

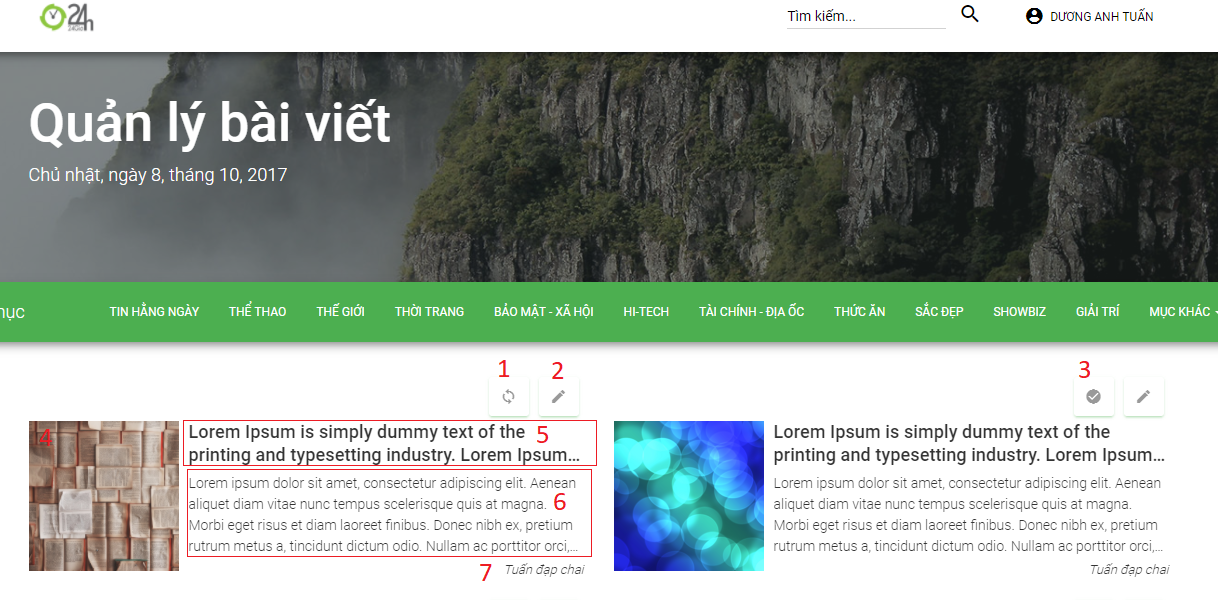


|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | lbRegister | Label |  |  | Show label of this pop-up |  |
| 2 | btnCloseSmall | Button |  | Click | Close the pop-up without login |  |
| 3 | txtOldPassword | Password field | 0<length<=200 |  | Old password |  |
| 4 | txtNewPassword | Password field | 0<length<=200 |  | New Password |  |
| 5 | txtConfirmNewPassword | Password field | 0<length<=200  txtNewPassword=txtConfirmPassword |  | Confirm Password |  |
| 6 | btnClose | Button |  | Click | Close Button |  |
| 7 | btnUpdate | Button |  | Click | Change password and redirect to profile page |  |

## Journalist side

### Manage Article page:

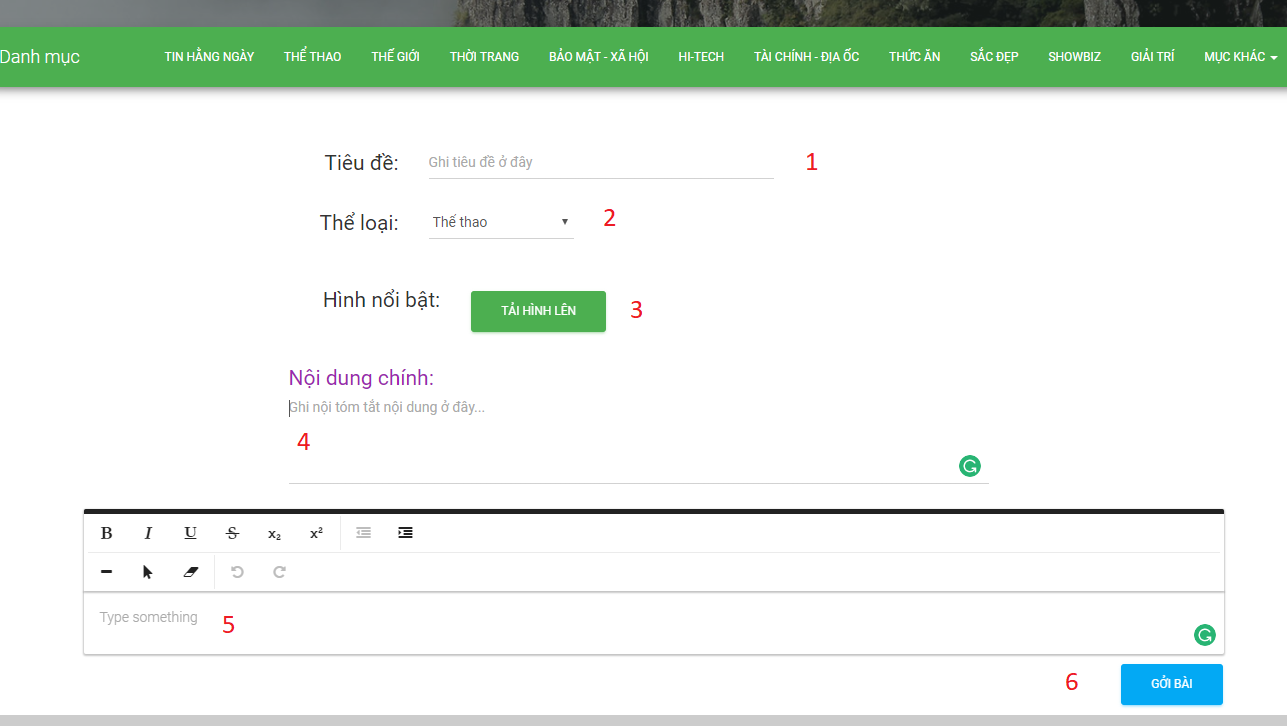
This is Manage article page, which shows articles created by this journalist.



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | btnWaitingStatus | Button |  |  | It Show that article is not available to public |  |
| 2 | btnEditArticle | Button |  | Click | Redirect to edit page |  |
| 3 | btnAvailableStatus | Button |  |  | It Show that article is available to public |  |
| 4 | imgArticle | Image |  |  | A image of this article |  |
| 5 | lbTitle | Label |  |  | Title of this article |  |
| 6 | lbMainContent | Label |  |  | Main content of article |  |
| 7 | lbAuthor | Label |  |  | Author of this article |  |

### Create or Edit Article page:

This is Create edit article page, allow journalist create new article or edit his article.

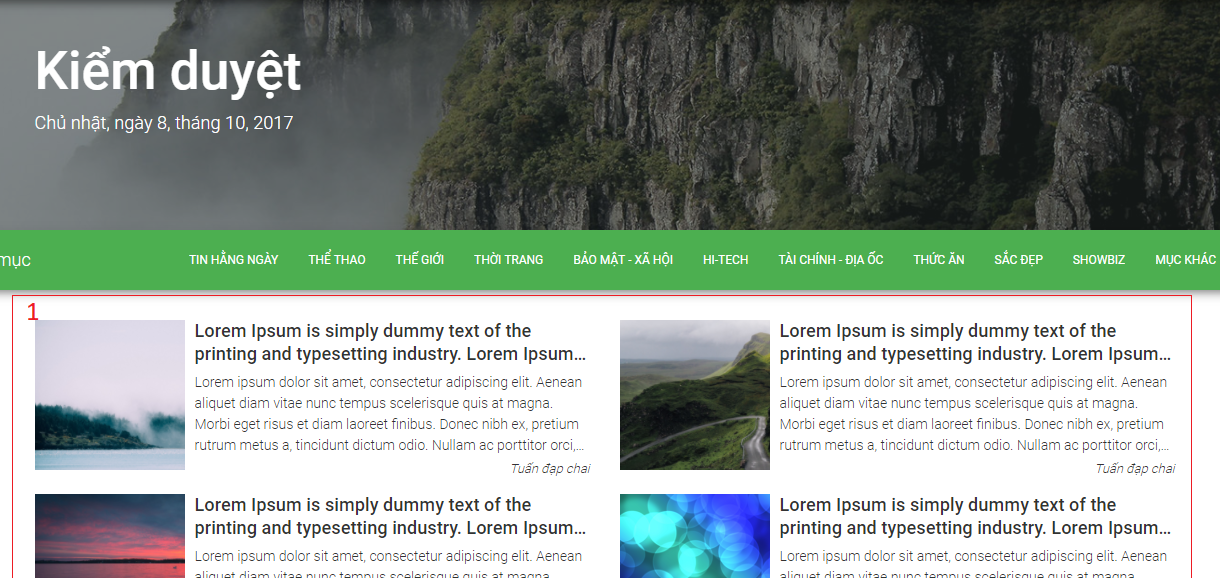


|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | txtTitle | Label |  |  | Title of new article |  |
| 2 | cbCategory | Combo Box |  | Click | Choose categories for this article |  |
| 3 | btnUploadImg | Button |  | Click | Upload thumbnail of this article |  |
| 4 | txtMainContent | Text area |  |  | Brief content of this article |  |
| 5 | txtContent | Froala text editor |  |  | Content of this article  Using third party text area |  |
| 6 | btnSubmit | Button |  | Click | Submit article and redirect to manage article page |  |

## Editor side

### Censor article page:

This is Censor article page, which shows unavailble articles.



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | gridPreviewArticle | Grid |  |  | Show 50 unavailable article increasing by time created |  |

### Censor article detail:

This is Censor article detail page, Almost the same at Article page (14.1.4). It provide button to change status of this article.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Name** | **Type** | **Validation** | **Event** | **Description** | **Status** |
| 1 | txtTitle | Label |  |  | Title of new article |  |

TASK SHEET

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Project Ref. No 1** | | **Project Title** | | **Activity Plan Prepared By** | | **Date of Preparation of Activity Plan** | | | | | |
| **Sr. No** | **Task** | |  | |  | | **Start Date** | **End Date** | **Actual Days** | **Member’s name** | **Status** |
| **CRS document** | | | | | | | | | | | |
| 1 | Introduction | | 24H Website | | V.X.M. Tuấn | | 09/17/2017 | 09/22/2017 | 5 | All of Team | FINISH |
| 2 | Existing Scenario | | 24H Website | | V.X.M. Tuấn | | 09/17/2017 | 09/22/2017 | 5 | All of Team | FINISH |
| 3 | Customer Requirement Specification | | 24H Website | | V.X.M. Tuấn | | 09/17/2017 | 09/22/2017 | 5 | All of Team | FINISH |
| 4 | Functional Requirement Specification | | 24H Website | | V.X.M. Tuấn | | 09/17/2017 | 09/22/2017 | 5 | All of Team | FINISH |
| 5 | System Requirements | | 24H Website | | V.X.M. Tuấn | | 09/17/2017 | 09/22/2017 | 5 | All of Team | FINISH |
| 6 | Development Software | | 24H Website | | V.X.M. Tuấn | | 09/17/2017 | 09/22/2017 | 5 | All of Team | FINISH |
| **Use Case and Sequence Diagrams** | | | | | | | | | | | |
| 7 | Guest register as Reader | | 24H Website | | V.X.M. Tuấn | | 09/22/2017 | 09/22/2017 | 1 | V.X.M. Tuấn | FINISH |
| 8 | Guest Login as Reader | | 24H Website | | V.X.M. Tuấn | | 09/22/2017 | 09/22/2017 | 1 | V.X.M. Tuấn | FINISH |
| **ERD** | | | | | | | | | | | |
| **GUI** | | | | | | | | | | | |
| **Report** | | | | | | | | | | | |
| **Permission Security** | | | | | | | | | | | |
| **Task sheet** | | | | | | | | | | | |
| 112 | Task sheet | | 24H Website | | V.X.M. Tuấn | | 09/17/2017 |  | 1 | V.X.M. Tuấn | FINISH |