References

[1] Kotler P. (1994), Marketing Management, Prentice-Hall Inc, 8, pp 62-69

[2] Ramakrishnan R, Gehrke J.(2003), Database Management Systems, McGraw-Hill Education Inc, 3(1), pp 59-85

[3] Summerville I. (2004), Software Engineering, Pearson Education Ltd, 7(1), pp 87-144

[4] Sampath Bank PLC (2005), Annual Report, pp 5

[5] Tiwana, A. (2001), The Essential Guide to Knowledge Management, Pearson Education Inc, 1, pp 78-84

[6] Tiwana, A. (2001), Knowledge Management Toolkit, Pearson Education Inc, 2(1), pp 89-120

[7] Vaswani V. (2007), PHP Programming Solutions, Tata McGraw-Hill Publishing Company Litmited, 1,pp 132 -182

[8] Whitten J.L.,Bentley L.D., Dittman K.C.(2003), System Analysis and Design Methods, Tata McGraw-Hill Publishing Company Litmited, 6(1), pp 430-444, pp 270 -288

[9] www.aiai.ed.ac.uk/links/cbr.html

[10] www.calstatela.edu/faculty/msabet/c09_Turban_0471705225.pdf

[11] www.free-college-says.com/Business/5250KnowledgeManagementsystems.html

[12] www.knowledgeTree.com

[13] www.php.net/

[14]www.syllogisticsoftware.com/papers/Web_Development_Technology_Comparison.html

[15] www.tlainc.com/articl120.htm

[16]aioarticles.com/an-easier-way-to-install-apache-php-mysql-and-phpmyadmin-a218694.html

[17] en.wikipedia.org/wiki/Adobe_Dreamweaver

[18] en.wikipedia.org/wiki/Apache_HTTP_Server

[19] en.wikipedia.org/wiki/MySQL

[20] en.wikipedia.org/wiki/PHP

[21] en.wikipedia.org/wiki/Smarty

[22] en.wikipedia.org/wiki/Test_plan

[23] en.wikipedia.org/wiki/Unified_Modeling_Language

Appendix A

Comparison of Web Technologies

Comparison of web technologies are done and table 3.1: Comparison of web technologies consists the comparison among commonly used web technologies.

Table 3.1: Comparison of web technologies

Area	LAMP	ASP.NET	J2EE
Licensing cost	Nil	Expensive	Nil
Support options and cost	Free support via community Paid support options available	Free support via community Paid support options available	Free support via community Paid support options available
Platforms	Multiple	Windows only	Multiple
Hardware Costs	Runs on very inexpensive servers	Requires slightly more expensive servers	Requires expensive servers
Staffing	Somewhat difficult to find qualified people	Very easy to find qualified people	Reasonably easy to find qualified people
External Hosting	Widely available and inexpensive	Widely available, but more expensive	Not widely available
Security	Very good	Historically very bad, but improved recently	Good
Performance	Very good	Often requires more expensive hardware to perform well	Often requires substantial configuration and expensive hardware
Scalability	Scales very well	Can be difficult to scale	Scales well when configured properly

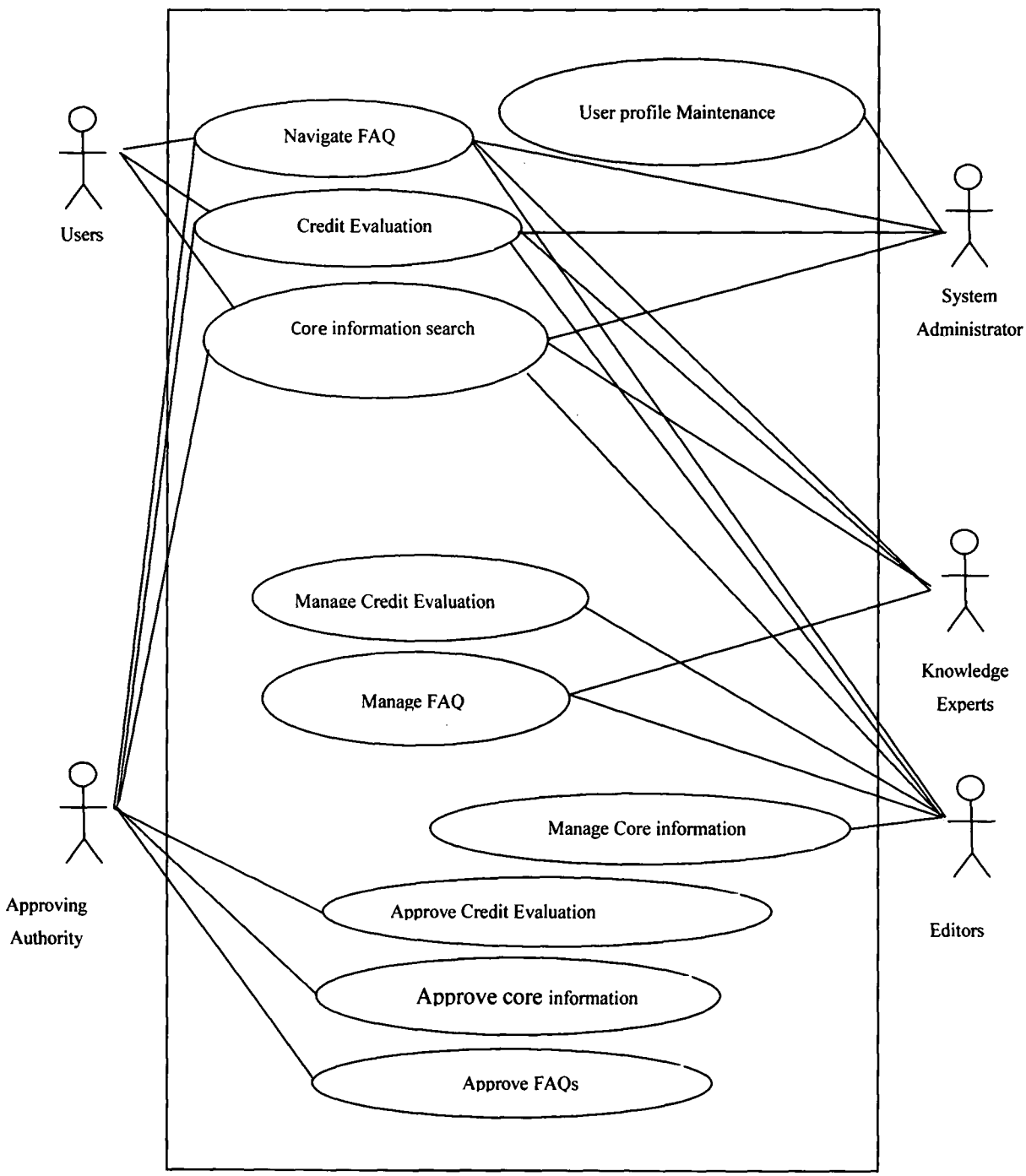
Area	LAMP	ASP.NET	J2EE
Administration	Difficult: Often requires reading documentation and editing text files	Easy: Often can be done through point and click interface	Moderate: Sometimes can be done visually
Configuration ease of use	Can be difficult to configure properly	Easy to configure	Moderately difficult to configure
Configuration flexibility	Extremely flexible	Not very flexible	Moderately flexible
Frameworks	Many available : often difficult to choose	One standardized framework	One standardized framework
Components	Widely available	Widely available	Widely available
Compatibility	Very good: New versions usually backwards-compatible	Moderate: New versions often break functionality	Bad: Many problems between old and new versions

Appendix B

Design Diagrams

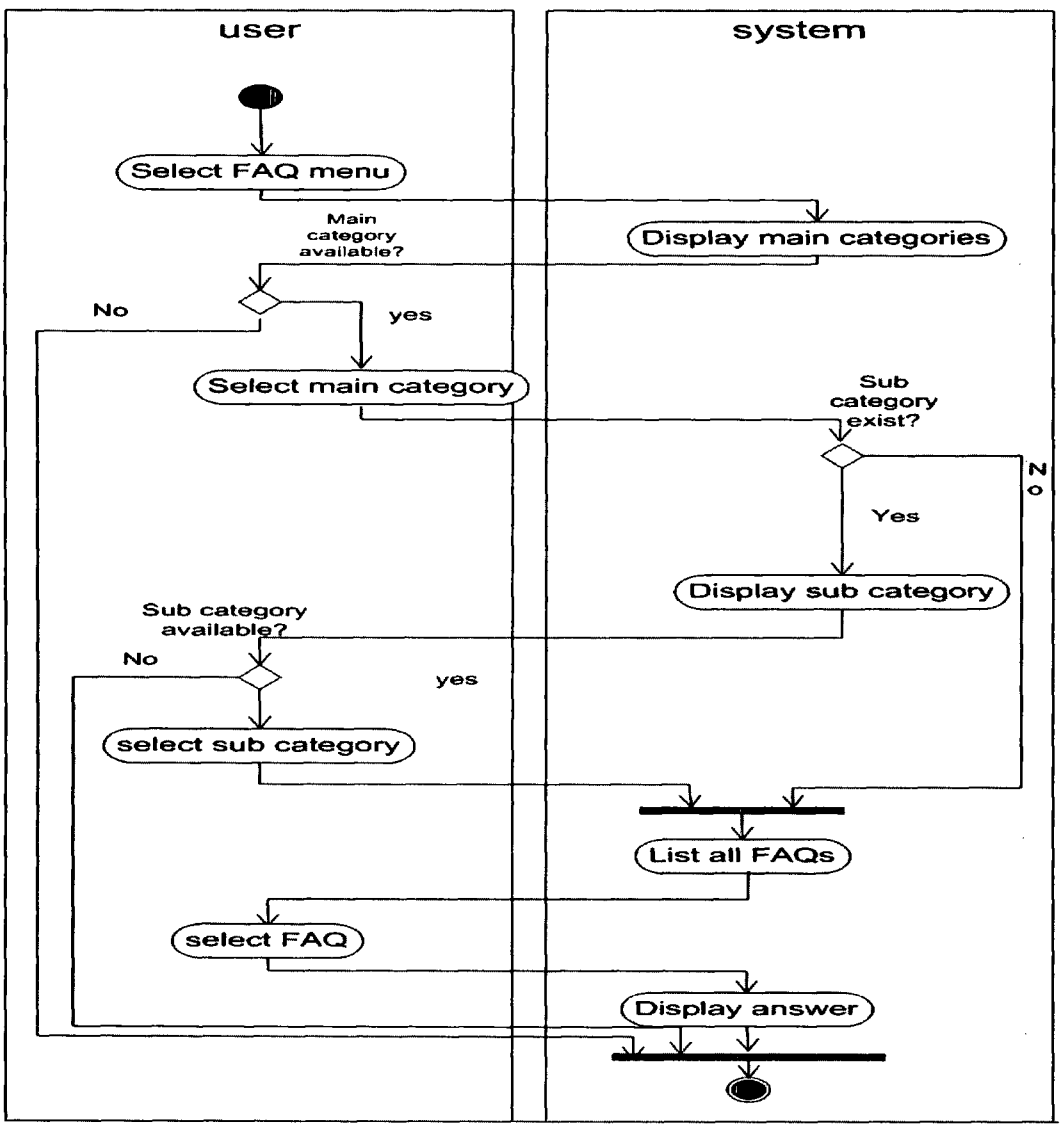
System is described using business use case diagram. Figure 5.4: Business use case diagram was depicted to identify user interaction with the system.

Figure 5.4: Business Use Case Diagram for (KMS)



Activity diagram was depicted to identify user FAQ navigation activity. Figure 5.5 Activity diagram for user FAQ navigation is given below,

Figure 5.5: Activity diagram for user FAQ navigation



Sequence diagram was depicted to identify behavior of the user FAQ navigation.

Figure 5.6 Sequence diagram for user FAQ navigation

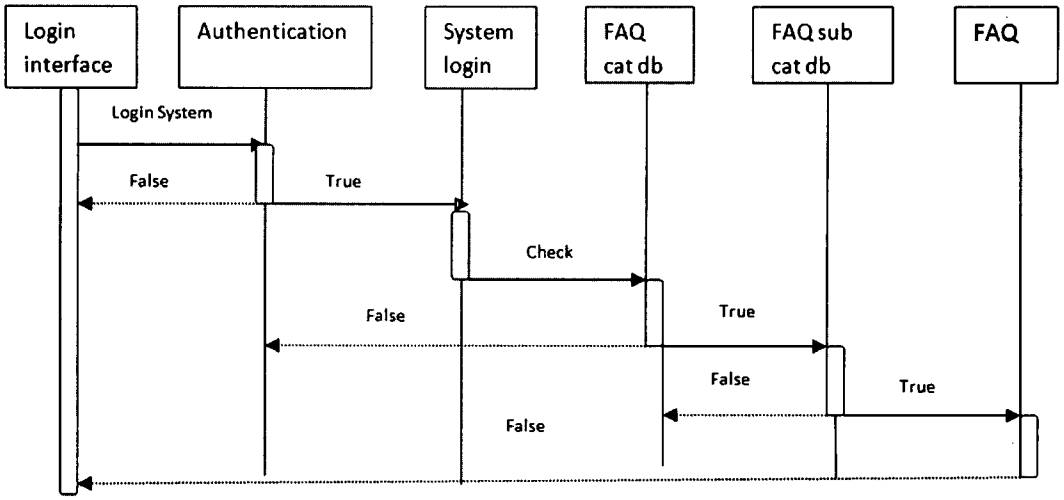
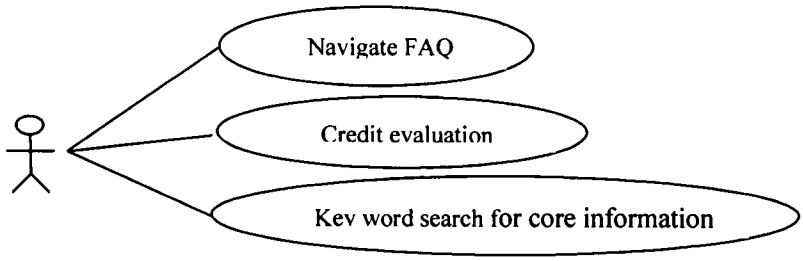
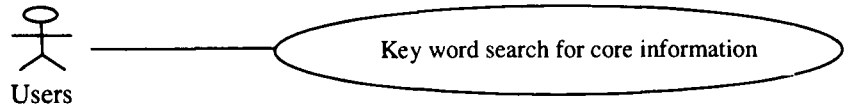


Figure 5.7: Detailed use case diagram for users.

User (team member) interaction with the system use cases given below



Key word search for core information



Brief Description

User selects to search information

Initial step by step description

Before this use case can be initiated, user has already accessed the main page

1 User enters key word, reference no or article category for search and perform search

2 System uploads the information and displays

3 User selects to search another word or selects to go to main page

Use Case	Key word search for core information (phrase base, ref number base search)	
Typical course of events	Actor action	System Response
	Step 1:. User chooses to search article by key word or phrase while selecting main category. Choices available are Category based search and key word or phrase base search **Step 3:** User chooses relevant sub category and perform search **Step 5:** User enters key word or phrase and perform search	**Step 2:** System loads and displays relevant sub categories in dropdown menu under the main categories **Step 4:** System performs search and prompt to type key word or phrase **Step 6:** System matches upload contents in database related to key word or phrase
Alternative courses	**In Step 1:** If user selects article to search by main category. Return to step 2, and step 3. **Step 2:** System performs search and displays article title in alphabetical order **Step 3:** User select article by clicking title **Step 4:** System uploads article and display content	

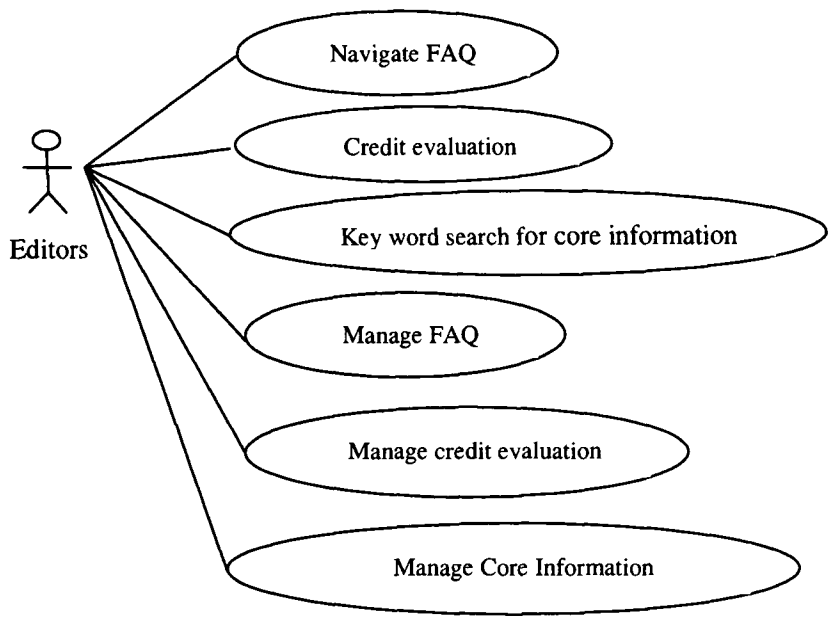
Table 5.3: Use case description for user core information search

Use Case Name	Navigate FAQ	
Typical course of events	Actor action	System Response
	Step 1: User selects to navigate FAQs	**Step 2:** System responds by displaying list of main FAQ categories in alphabetical order.
	Step 3: User selects main FAQ category. **Step 5:** User select the required question from the list	**Step 4:** System responds by displaying list of questions under selected Main FAQ category in alphabetical order. **Step 6:** System responds with the answer for the selected question
Alternative courses	**Step4:**If main FAQ categories have sub FAQ categories, system responds by displaying list of sub category FAQs in alphabetical order. **Step 5:** User selects required sub category from the FAQ sub category list. **Step 6:** System responds by displaying list of questions under selected sub category FAQ. Return to **Step 5:**	

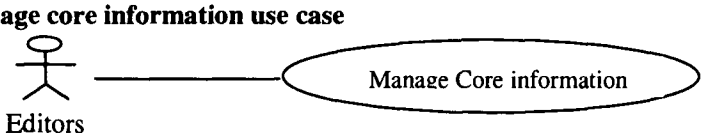
Table 5.4: Use case description for user FAQ navigation

Figure 5.8: Detailed use case diagram for Editors

Editors have following use cases



Manage core information use case



Brief Description

Editor selects manage core information option

Initial step by step description

Before this use case can be initiated, user has already accessed the main page

1 Editor selects to manage core information

2 System displays add/update/delete article options

3 Editor Selects add or update or deleted options

4 System loads form to add article or prompts to enter article reference number or name to update or delete

5 Editor add contents and submits or update contents or delete article

6 System prompts to go back to main menu

Table 5.5: Use case description for Editor manage core information

Use Case Name	Manage Core information	
Typical course of events	Actor action	System Response
	Step 1: Editor select add editing option. The selections available are add article, delete article and update article. Adding of articles are done based on category and memo reference numbers	**Step 2:** If editor select to add article system will display main Category list in alphabetical order.
		Step 4: System will display all sub categories relate to selected main category.
	Step 3: Editor selects main category from the list.	**Step 6:** System will loads the form to add content.
	Step 5: Editor select relevant sub category from the list	**Step 8:** System saves content in database and apply Article ID
	Step 7: Editor enters Article title, reference number and contents on the form and submits.	**Step 10:** System displays list of names of approving authority
	Step 9: Editor selects relevant approving authority	**Step 12:** System performs forward to relevant authority and updates file status to pending status.
	Step 11: Editor selects relevant approving authority and forward for approval	

Alternative courses	**In Step 1:** If Editor chooses update option to update an article. **Step 2:** System prompts to enter Article/memo reference number **Step 3 :** Editor enters reference number and submits **Step 4:** System displays contents of the article in edit mode **Step 5:** Editor selects area/part of the content to be updated and click edit button. **Step 6:** System pops up editor box/window with selected content **Step 7:** Editor updates the content and click finish button **Step 9:** System performs saving of previous content to the update table with unique ID(update ID) and updated date **Step 10:** System replaces previous content with updated contents by applying unique identifiers. Return to **Step 9** in typical cause of events. **In Step 1:** If Editor chooses delete option to delete an article. **Step 2:** System prompts to enter Article/memo reference number **Step 3 :** Editor enters reference number and submits **Step 4:** System displays contents of the article in delete mode **Step 5:** Editor delete article **Step 9:** System performs saving of deleted content and deletes unique ID. Return to **Step 9** in typical cause of events.

Manage FAQ use cases

Brief Description

Editor selects manage FAQ option

Initial step by step description

Before this use case can be initiated, user has already accessed the main page

1 Editor selects to manage FAQ

2 System displays add/update/delete FAQ options

3 Editor selects add or update or deleted options

4 System loads form to add FAQ or prompts to enter FAQ reference number or name to update or delete

5 Editor add contents and submits or update contents or delete article

6 System prompts to go back to main menu

Xref: Section

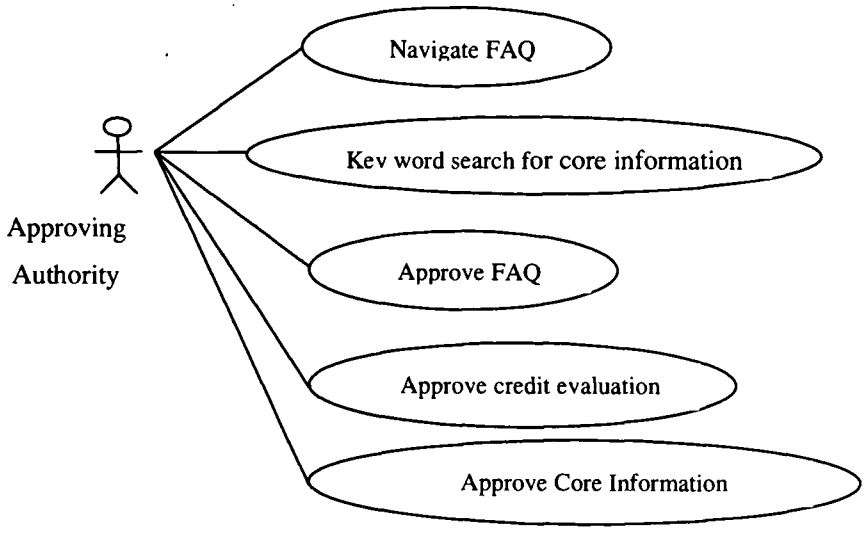
Table 5.6: Use case description for Editor manage FAQ

Use Case Name	Manage FAQ	
Typical course of events	Actor action	System Response
	Step 1: Knowledge experts select add Case study option. The selections available are add, delete and modify FAQ.	**Step 2:** If Knowledge experts or Editors select to add new FAQ system will display main FAQ Category list in alphabetical order.
	Step 3: Knowledge experts or Editors select main category from the list.	**Step 4:** System will display all sub categories relate to selected main category.
	Step 5: Knowledge expert or Editors select relevant sub	**Step 6:** System will loads the form to add content.

	category from the list **Step 7:** Knowledge experts or Editors enters Article title, reference number and contents on the form and submits. **Step 9:** Knowledge expert or Editor selects relevant approving authority **Step 11:** Knowledge Experts or Editors select relevant approving authority and forward for approval	**Step 8:** System saves content in database and apply Article ID **Step 10:** System displays list of names of approving authority **Step 12:** System performs forward to relevant authority and updates file status to pending status.

Figure 5.9: Detailed use case diagram for Approving Authority

Approving authorities have following use cases



Approve Core information use case

Brief Description

Approving Authority selects approve core information option

Initial step by step description

Before this use case can be initiated, user has already accessed the main page

1 Approving authority selects to approve core information

2 System displays approve/reject/modify core information options

3 Approving Authority selects approve or modify or reject options

4 System loads form to modify core information or prompts approve/delete Core information with reference number and name of the article

5 Approving Authority modify contents and submits or approve/reject core information

6 System prompts to go back to main menu

Approve FAQ use case

Brief Description

Approving Authority selects approve FAQ option

Initial step by step description

Before this use case can be initiated, user has already accessed the main page

1 Approving authority selects to approve FAQ

2 System displays approve/reject/modify FAQ options

3 Approving Authority selects approve or modify or reject options

4 System loads form to modify FAQ or prompts approve/delete FAQ

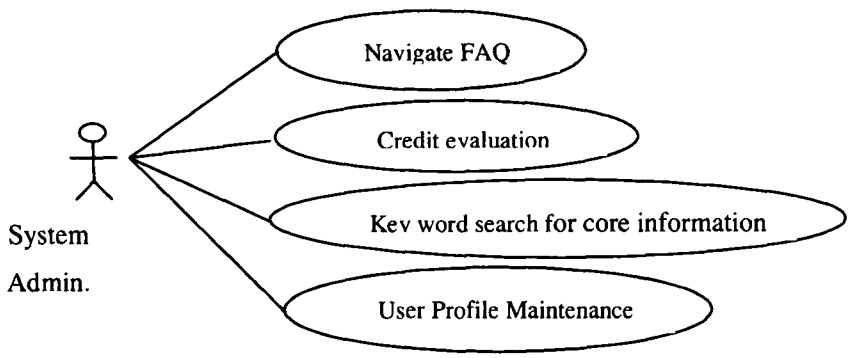
5 Approving Authority modify contents and submits or approve/reject FAQ

6 System prompts to go back to main menu

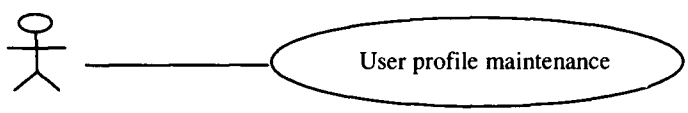
Xref: Section

Figure 5.10: Detailed use case diagram for System Administrators

System administrators have following use cases



User profile maintenance use case



System Administrator

Brief Description

System admin selects to maintain user profiles

Initial step by step description

Before this use case can be initiated, user has already accessed the main page

1System Administrator selects to add/delete/modify user levels

2 System displays add, delete/modify user levels options

3 System Administrator selects to add or delete or modify user levels

4 System loads add user forms or prompts to type employee number to be deleted

5 System Admin enter new user data and submits or enter employee number and search.

6 System adds new user or displays user details to be deleted

7 System Administrator modifies user levels or delete user

8 System updates user details and returns to main menu

Xref: Section

Table 5.7: Use case description user profile maintenance of System administrators

Use Case Name	User profile maintenance	
Typical course of events	Actor action	System Response
	Step 1:.System Administrator to create new user. Options available to System Admin are create new users, delete existing users and update user levels **Step 3:** System Admin. enters data in new user page and submits	**Step 2:** System uploads new user page to add user name, employee ID, user password, email address, user level **Step 4:** System saves submitted content with current date, time in the database by applying active status. System will encrypt the password with MD5 hash encryption **Step 5:** Users will be notified by the system by an email containing user name, password, login URL
Alternative courses	**In Step 1:** If System Admin selects to delete existing user **Step 2:** System uploads user information with account status **Step 3:** System Admin delete the employee ID and change account status to delete **Step 4:** System updates database with deleted account status **In Step 1:** If System Admin selects to update user levels **Step 2:**. System displays employee ID and user account level in edited mode **Step 3:** System Admin changes the account level and submits **Step 4:** System updates user account level in the database	

Appendix C

User Interface and Coding

Screen shot of the login page of the Knowledge management System is shown in Figure 6.1: Login page of the system

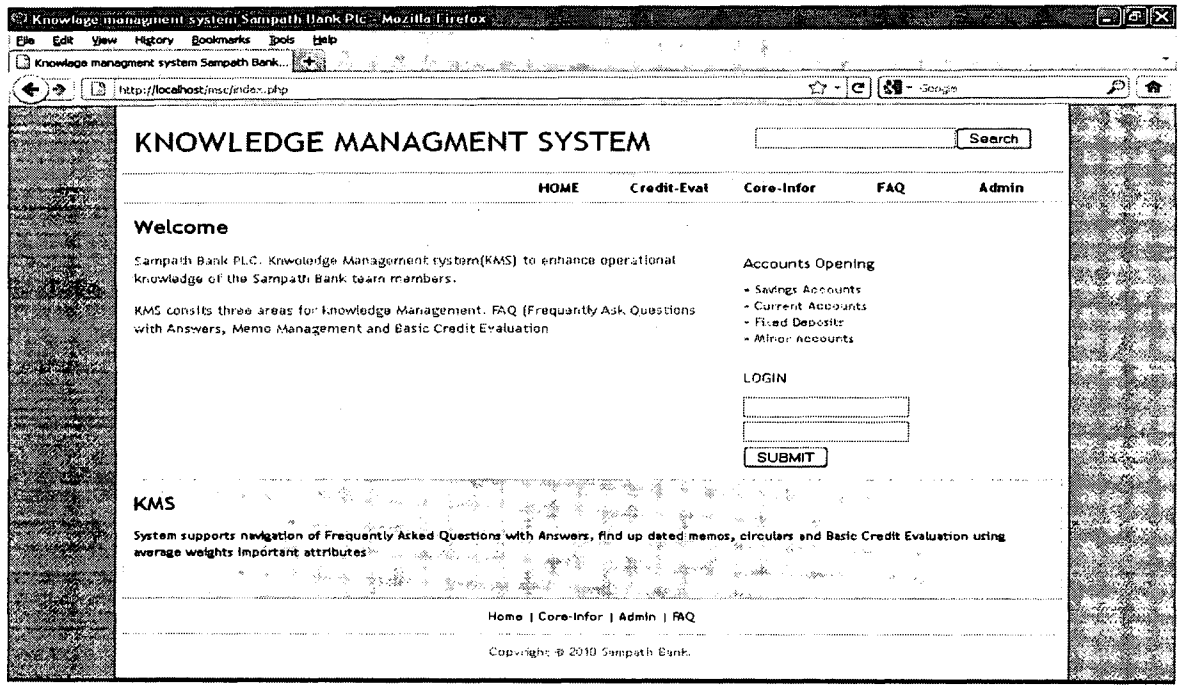


Figure 6.1: Login page of the system

Main interface supports all user access to FAQ's, Core information, user access requests and key word search options. FAQ's, Core information and key word base search could be accessed from this menu once the system is deployed in bank's intranet.

Code segments of the login page of the system was given below,

```
<?php
@session_start();
include('../inc_func.php');
$con  = mysql_connect('localhost','root','root');
$ds   = mysql_select_db('msc',$con);
if($_POST['login']=="yes"){
        $username=$_POST['name'];
        $password=$_POST['password'];
        $status="user"; $encPass=md5($password);
```

```
$sel=mysql_query("select * from faq_user where
user_name='$username' and password='$encPass'")or die ("Internal Server Error");
if(mysql_num_rows($sel)>0){
$row=mysql_fetch_array($sel);
$user_id=$row['id'];
echo $_SESSION['user_id']=$user_id;
print "<script>";
print "window.location='../index.php';";
print "</script>";
}
}
else{

$username_check=is_username_exists($username);
if($username_check){

$return_access_level=check_active($username,$password,false,"1");
if($return_access_level){
//unauthorzed user lavel
echo $msg='<img src="images/error.gif"
border="0" />'.'<span class="red_msg">Unauthorized access.</span>';
}
else{

$ret=check_active($username,$_POST['email'],'user',0);
if($ret==true){
$rrsel=mysql_query("select * from
tbl_user where t_username='$username' ");
if(mysql_num_rows($rrsel)>0){

$rrselrw=mysql_fetch_array($rrsel);
$id=$rrselrw['id'];
```

```
                    echo $msg='<img src="images/error.gif" border="0" />'.'<span class="red_msg">Account not yet activated. <a href="register1.php?id='.$id.'">Click here</a> to activate.</span>';
                    }
                    else{
                    echo $msg='<img src="images/error.gif" border="0" />'.'<span class="red_msg">The username or password you entered is incorrect.</span>';
                    }
                    }
                    else{
                    echo $msg='<img src="images/error.gif" border="0" />'.'<span class="red_msg">The username or password you entered is incorrect.</span>';
                    }
                    }
        }
                    else{
                    echo $msg='<img src="images/error.gif" border="0" />'.'<span class="red_msg">Username: '.$username.' is not yet taken. <a href="registration.php" class="link2"> Click here</a> to register as a new user.</span>';
                    }
        }
//}
?>
```

Screen shot of the add new FAQ menu is shown Figure 6.2 Create new FAQ

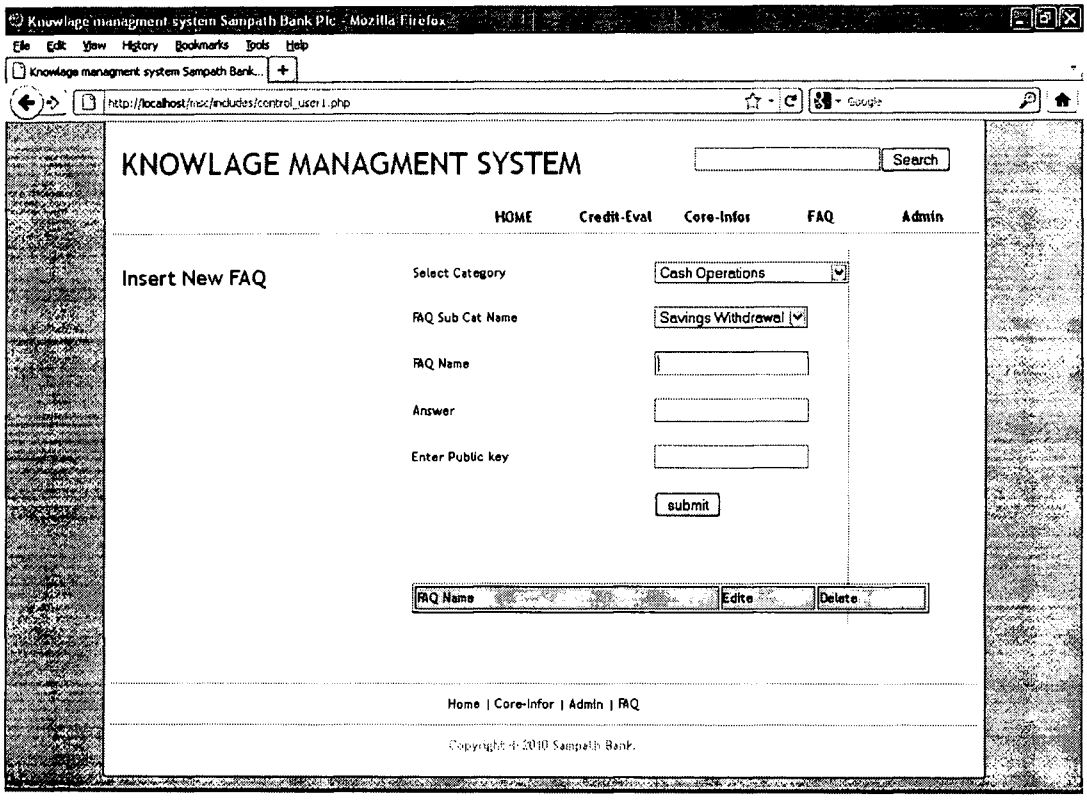


Figure 6.2 Create new FAQ

Editors can enter FAQ's in to the system via insert FAQ interface where FAQ main category, FAQ sub category, FAQ name and Answer for the FAQ to be entered. Further public key also need to be entered. Delete and edit options are also available on this interface.

Code segments of the create new FAQ option was given below

```
<?phP
@session_start();
$con  = mysql_connect('localhost','root','root');
$ds   = mysql_select_db('msc',$con);
if(isset($_POST['post_update']) and $_POST['faq']!='' and $_POST['answer']!=''){
$dic = mysql_query("insert into faq_digital_signature
(sub_cat_id,faq_encode_name,faq_encode_answer,adddate,editer) values
('".$_POST['subcat']."','".$_POST['faq']."','".$_POST['answer']."',now(),'".$_SESSIO
N['user_id']."')")or die(mysql_error());
$str = $_POST['faq'];
```

```
$str2 = $_POST['answer'];

$converter = new Digital_signature;
$encoded = $converter->encode($str );
$encoded2 = $converter->encode($str2);
$dic = mysql_query("insert into faq_digital_signature
(sub_cat_id,faq,answer,public_key,editer) values
('".$_POST['subcat']."','".$encoded."','".$encoded2."','".$_POST['public_key']."','".$_S
ESSION['user_id']."')")or die(mysql_error());
}
if(isset($_GET['del_id'])){
$del =mysql_query("delete from faq where faq_id=".$_GET['del_id']) or
die(mysql_error());
}
if(isset($_POST['faq_updt'])){
$faqs = mysql_query("update faq set
answer='".$_POST['answer_edi']."',faq='".$_POST['faq_edi']."' where
faq_id='".$_GET['edi_id']."'");
if($faqs){
                print "<script>";
                  print "window.location='control_user1.php';";
                   print "</script>";
        }
        }
?>
```

Screen shot of the approve FAQ menu is shown Figure 6.3: Approve FAQ

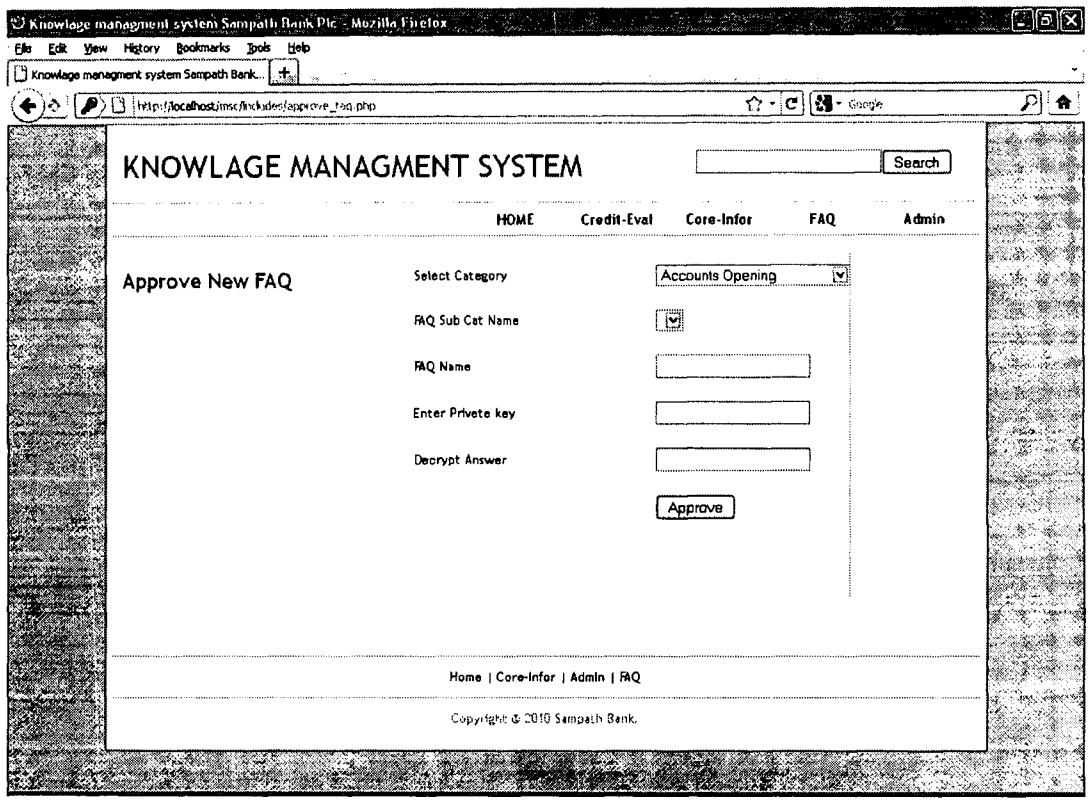


Figure 6.3: Approve FAQ

Approving Authority personnel has to enter own private key to see the answer. Then decrypted answer will be displayed and with submission of the approval button answer is published.

Code segments of the Approve FAQ was given below

```
<?phP
@session_start();
$con  = mysql_connect('localhost','root','root');
$ds   = mysql_select_db('msc',$con);
if(isset($_POST['post_update']) and $_POST['faq']!='' and $_POST['answer']!=''){
$dic = mysql_query("insert into faq (sub_cat_id,faq,answer,adddate,editer) values
('".$_POST['subcat']."','".$_POST['faq']."','".$_POST['answer']."',now(),'".$_SESSIO
N['user_id']."')")or die(mysql_error());
$str = $_POST['faq'];
$str2 = $_POST['answer'];
$converter = new Digital_signature;
```

```
$encoded = $converter->encode($str );
$encoded2 = $converter->encode($str2);
$dic = mysql_query("insert into faq_digital_signature (sub_cat_id,faq,answer,public_key,editer) values ('".$_POST['subcat']."','".$encoded."','".$encoded2."','".$_POST['public_key']."','".$_SESSION['user_id']."')")or die(mysql_error());
}
if(isset($_GET['del_id'])){
$del =mysql_query("delete from faq where faq_id=".$_GET['del_id']) or die(mysql_error());
}
if(isset($_POST['faq_updt'])){
$faqs = mysql_query("update faq set answer='".$_POST['answer_edi']."',faq='".$_POST['faq_edi']."' where faq_id='".$_GET['edi_id']."'");
if($faqs){
                print "<script>";
                  print "window.location='control_user1.php';";
                   print "</script>";
        }
        }
?>
```

Screen shot of the navigation of a FAQ menu is shown Figure 6.4: FAQ navigation

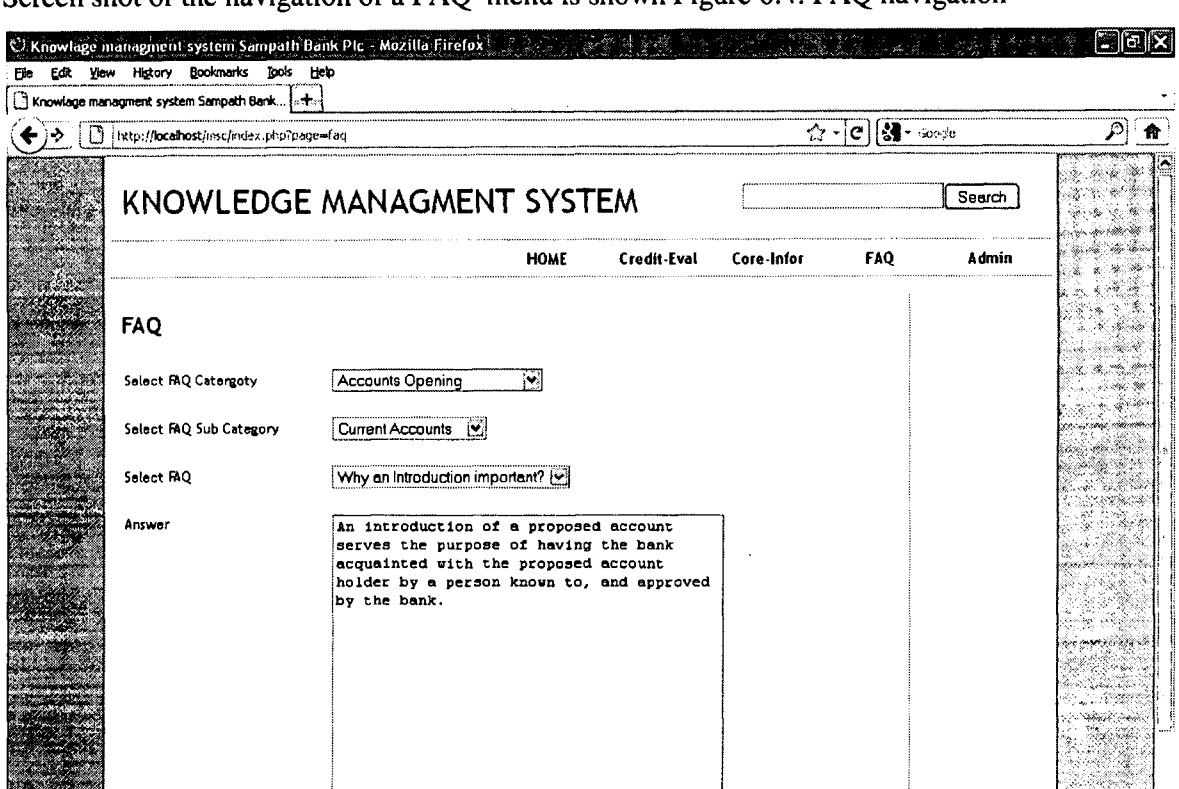


Figure 6.4: FAQ navigation

FAQ option provides FAQ navigation via FAQ search interface where answer for a FAQ could find out selecting appropriate FAQ category, FAQ sub category and FAQ. Code segments of user FAQ navigation was given below

```
<?php
classFaq {
        static private $_tablename    = 'faq';
        static private $_primarykey   = 'faq_id';
        private $faq_id                    = null;
        private $faq                 = null;
        private $answer                    =      null;
        private $adddate                   =      null;
        private $meta_key                  =      null;
        private $meta_title                =      null;
        private $meta_des                  =      null;
        private $active        =  0;
```

```
        function __construct($faq_id=null,$faq=null,

        $answer=null,$adddate=null,$meta_key=null,$meta_title=null,$meta_des=nul
l,$meta_des=0) {
                $this->faq_id    =    $faq_id;
                $this->faq       =    $faq;
                $this->answer    =  $answer;
                $this->adddate   =  $adddate;
                $this->meta_key  =   $meta_key;
                $this->meta_title =   $meta_title;
                $this->meta_des  =   $meta_des;
                $this->active    =  $active;
        }
        public function getId(){
                return $this->faq_id;
        }
                public function getfaq(){
                return $this->faq;
        }
                public function getanswer(){
                return $this->answer;
        }
        public function getadddate(){
                return $this->adddate;
        }
        public function getMetaKey(){
                return $this->meta_key;
        }
        public function getMetaTitle(){
                return $this->meta_title;
        }
public function getMetaDes(){
                return $this->meta_des;
        }
```

```
		public function getactive(){
		return $this->active;
	}
	static function getfaqans($con=""){
		$result =
DBAccess::querySelect(self::$_tablename,$con,"*",array(),"A",0,0);
		$arr = array();
		while($row = $result->fetch_object()){
		$arr[] = new Faq($row->faq_id, $row->faq, $row->answer, $row-
>adddate, $row->meta_key, $row->meta_title, $row->meta_des);
		}
		return $arr;
		}
	}
?>
<?php
$faqcats	=	Faqcat::getfaqcat();
$_smarty -> assign('faqcats',$faqcats);
$faqanswers	=	Faq::getfaqans();
$_smarty -> assign('faqanswers',$faqanswers);
$content	=	$_smarty->fetch('faq.html');

?>
```

Screen shot of the user key generation menu is shown Figure 6.5: User key Generation

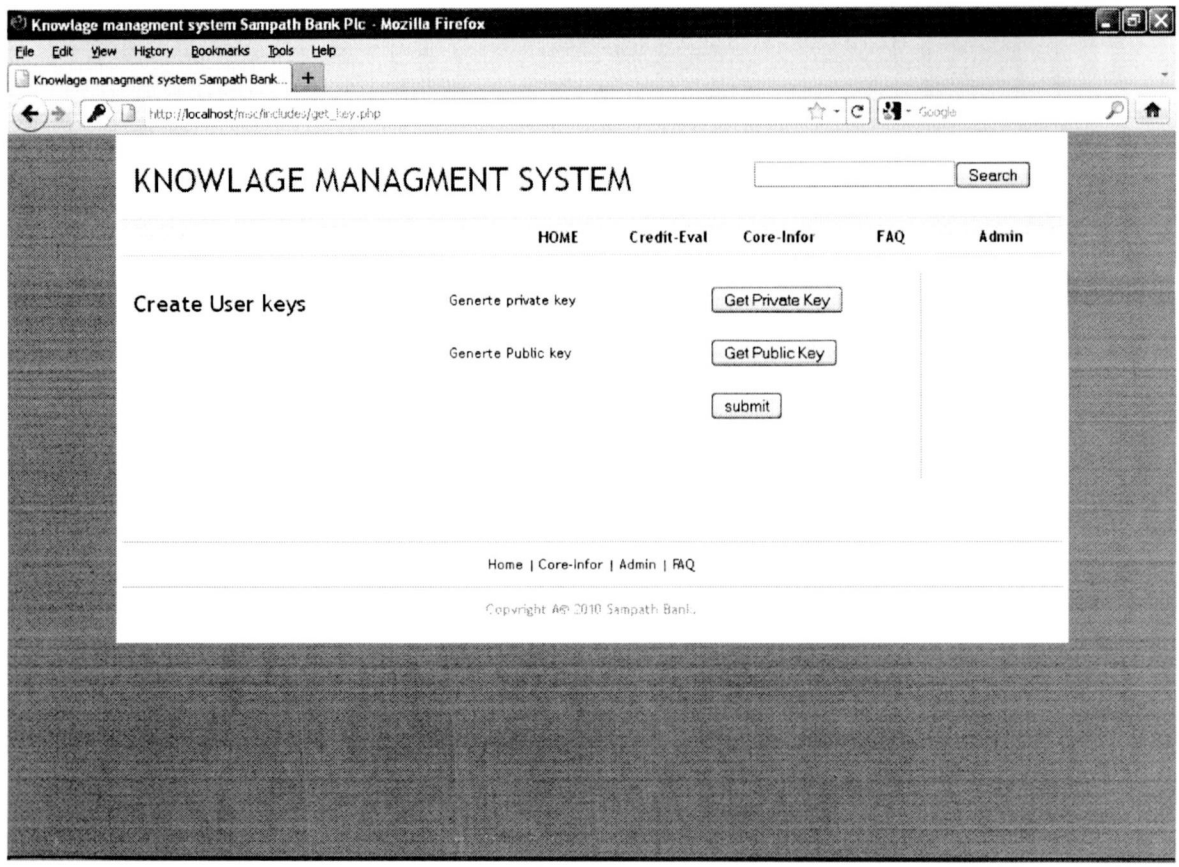


Figure 6.5: User key Generation

In order to incorporate digital signature concept a public key and a private key had to be generated for each editor and approving authority personnel. Two random eight digit keys were generated using A-Z characters and 1-9 numbers initially. Then these public and private keys are stored in the database.

Code segments of user key generation was given below

```
public function get_key
{
        $characters = array(
"A","B","C","D","E","F","G","H","J","K","L","M","N","P","Q","R","S","T","U","V",
"W","X","Y","Z","1","2","3","4","5","6","7","8","9");
$keys = array();
while(count($keys) < 8) {
```

```
    $x = mt_rand(0, count($characters)-1);
    if(!in_array($x, $keys)) {
      $keys[] = $x;
    }
}
foreach($keys as $key){
  $random_chars .= $characters[$key];
}
                    }
```

Screen shot of the user access rights is shown Figure 6.6: Validate user

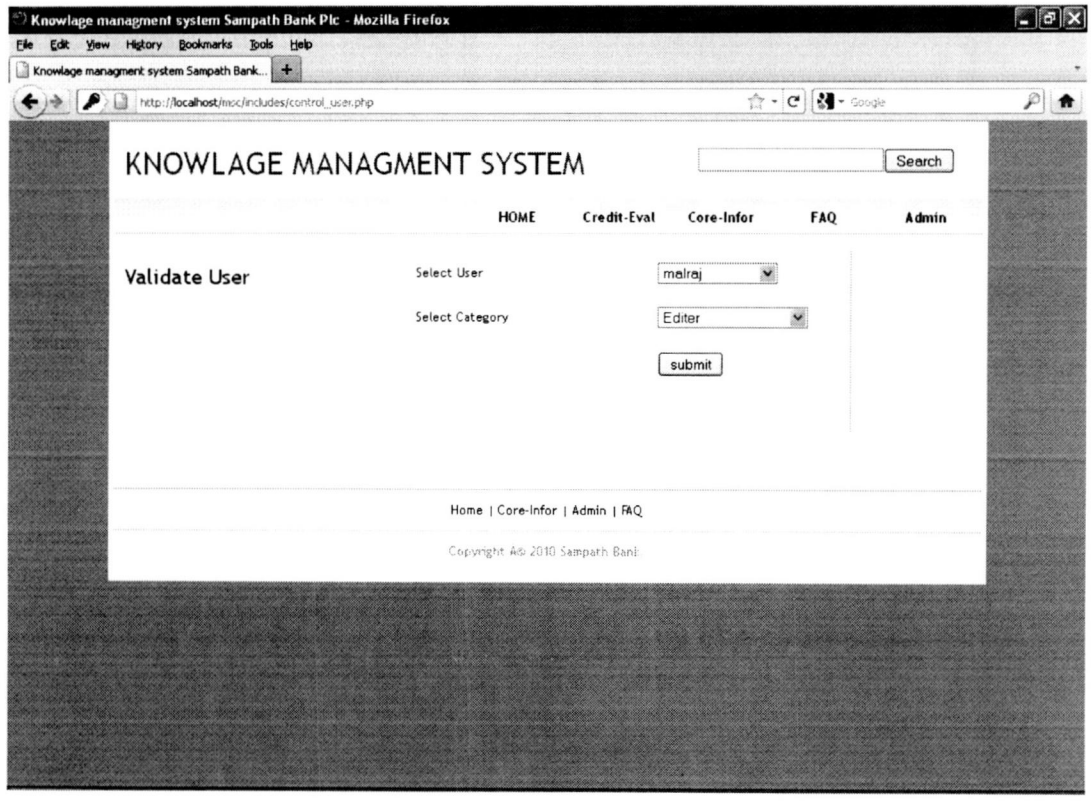


Figure 6.6: Validate user

Administrators could log into the system and provide user access rights via user validate interface.

Code segments of the user access right control was given below

```
<?phP
@session_start();
if(isset($_POST['post_update'])){
```

```
$con  = mysql_connect('localhost','root','root');
$ds   = mysql_select_db('msc',$con);
$dic  = mysql_query("update faq_user set type='".$_POST['user_cat']."' where
id='".$_POST['user']."'")or die(mysql_error());
        }
?>
```

Appendix D

Test Cases

Generated test cases and status of the system reply is given in Table 7.1: Test cases for Login and FAQ model

Table 7.1 : Test cases for Login and FAQ module

Test case name	Test case	Input	Expected results	Status
Login case	Test that the users log with correct user name and password	User name and password	Log to the system Display successful message	ok
	Test that the user cannot log with invalid user name and valid password	User name and password	Log in fail Display error message	fail
	Test that the user cannot log with valid user name and invalid password	User name and password	Log in fail Display error message	fail
	Test that the user cannot log with invalid user name and invalid password	User name and password	Log in fail Display error message	fail
Search FAQ	Test search FAQ option	Select FAQ cat option	Display cat	ok
		Display FAQ cat	Display Sub cat	ok
		Display sub cat	Display FAQ	ok
		Not display cat	Return to select FAQ cat option	ok
		Not display sub cat	Return to FAQ cat option	ok
Add FAQ	Test add FAQ option	Select FAQ cat option	Display cat	ok
		Display FAQ cat	Display Sub cat	ok
		Display sub cat	add FAQ	ok
		Submit added FAQ	Added to system	ok
		Not display cat	Return to select FAQ cat option	ok
		Not display sub cat	Return to FAQ cat option	ok

Test case name	Test case	Input	Expected results	Status
Add FAQ cat	Adding FAQ category	Prompt FAQ cat heading	Get heading inserted	ok
Add FAQ sub cat	Adding FAQ sub category	Prompt FAQ sub cat heading	Get sub cat heading inserted	ok
Delete FAQ cat	Test delete FAQ cat option	Select delete option	Delete selected cat FAQ	ok
		Select other option	Deletion failed	fail
Delete FAQ sub cat	Test delete FAQ sub cat option	Select delete option	Delete selected sub cat FAQ	ok
		Select other option	Deletion failed	fail
Delete FAQ	Test delete FAQ option	Select delete option	Delete selected FAQ	ok
		Select other option	Deletion failed	fail
Edit FAQ	Test edit FAQ option	Select edit option	display FAQ	ok
		Select other option	FAQ not Edit	fail
Edit FAQ cat	Test edit FAQ cat option	Select edit option	Delete selected cat FAQ	ok
		Select other option	Deletion failed	fail
Edit FAQ sub cat	Test edit FAQ sub cat option	Select edit option	Delete selected sub cat FAQ	ok
		Select other option	Deletion failed	fail

Generated test cases and status of the system reply is given in Table 7.1: Test cases for core information model

Table. 7.2: Test cases for Core information module

Test case name	Test case	Input	Expected results	Status
Key word Search	Key word / phrase search	Key word/phrase	Match key word / phrase Display results	ok
			Not mach	fail

Test case name	Test case	Input	Expected results	Status
Search from category	Test search core information cat option	Select article cat option	Display cat	ok
		Display article cat	Display article Sub cat	ok
		Display article sub cat	Prompt to enter key word	ok
		Submit key word	Display results matched to key word	ok
		Not display cat	Return to select article option	ok
		Not display sub cat	Return to article cat option	ok
Add Article	Test add article option	Select article cat option	Display article cat	ok
		Display article cat	Display article Sub cat	ok
		Display article sub cat	add article	ok
		Submit added article	Added to system	ok
		Not display article cat	Return to select article option	ok
		Not display article sub cat	Return to article cat option	ok
Add article cat	Adding article category	Prompt article cat heading	Get heading inserted	ok
Add article sub cat	Adding article sub category	Prompt article sub cat heading	Get sub cat heading inserted	ok
Delete article cat	Test delete article cat option	Select delete option	Delete selected article cat	ok
		Select other option	Deletion failed	fail
Delete article sub cat	Test delete article sub cat option	Select delete option	Delete selected article sub cat	ok
		Select other option	Deletion failed	fail
Delete article	Test delete article option	Select delete option	Delete selected article	ok
		Select other option	Deletion failed	fail
Edit article	Test edit article option	Select edit option	display article	ok
		Select other option	article not Edit	fail

Test case name	Test case	Input	Expected results	Status
Edit article cat	Test edit article cat option	Select edit option	Delete selected article cat	ok
		Select other option	Deletion failed	fail
Edit article sub cat	Test edit article sub cat option	Select edit option	Delete selected article sub cat	ok
		Select other option	Deletion failed	fail

Appendix E

Questioner used to assess whether developed system met set objectives

1) Branch level knowledge can be shared across the entire branch network using the system

i) Strongly Agree ii) Agree iii) Disagree iv)Strongly Disagree

2) Knowledge Management tools are important when operational decisions are taken

i) Strongly Agree ii) Agree iii) Disagree iv) Strongly Disagree

3) Answers for special cases/ issues are available for easy reference in the system

i) Strongly Agree ii) Agree iii) Disagree iv) Strongly Disagree

4) Updated operational procedures / rules are available easy reference in the system

i) Strongly Agree ii) Agree iii) Disagree iv) Strongly Disagree

5) Branch level knowledge sharing would help other branches

i) Strongly Disagree ii)Disagree iii) Agree iv) Strongly Agree

6) Web base knowledge Management tools would enhance knowledge of the team members

i) Strongly Disagree ii)Disagree iii) Agree iv) Strongly Agree

7)Availability of the Questions & Answers for special cases would enhance knowledge

i) Strongly Disagree ii)Disagree iii) Agree iv) Strongly Agree

8) Availability of the Updated operational procedures / rules on web would increase knowledge

i) Strongly Disagree ii)Disagree iii) Agree iv) Strongly Agree

Appendix F

Definitions, Acronyms and Abbreviations

This section provides definitions for common terms used throughout the document. They are provided here to help minimize ambiguity throughout the document.

SRS :	Software Requirements Specification - specifies customer and detailed requirements.
KM :	Knowledge Management –KM comprises a range of practices used by organisations to identify, create, represent, distribute and enable adoption of what it knows, and how it knows it
KMS :	Knowledge Management System refers to a (generally IT based) system for managing knowledge in organizations, supporting creation, capture, storage and dissemination of information. It can comprise a part (neither necessary or sufficient) of a knowledge management initiative
FAQ :	Frequently Asked Questions
Tacit Knowledge :	Tacit knowledge resides in the minds of individuals. It is the tacit knowledge that never is quantified into a manual or other accessible form, but resides in the minds of the people who have worked with and developed that information.
Explicit knowledge :	Explicit knowledge is knowledge contained in documents, computer programs, databases, etc., and can be articulated easily.