

**สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง**

**DEVELOPMENT OF WEB-BASED CRC**



E071837

MISS PIYAWAN KETYUNGYUENWONG

MISS CHUTTRAPORN SONGNAWARAD

MISS BOONTARIKA SILPIPAT

สารบัญ  
เลขทะเบียน 71837  
วันเดือนปี 27 ส.ค. 2554

b.....	x
i.....	

A SPECIAL PROJECT SUBMITTED IN PARTIAL FULFILLMENT  
OF THE REQUIREMENT FOR THE DEGREE OF BACHELOR OF SCIENCE  
IN DEPARTMENT OF MATHEMATICS AND COMPUTER SCIENCE  
FACULTY OF SCIENCE  
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG  
ACADEMIC YEAR 2009

**Title** Development of Web-based CRC

**Student** Miss Piyawan Ketyungyuenwong  
Miss Chuttraporn Songnawarad  
Miss Boontarika Silpipat

**Degree** Bachelor of Science

**Major Program** Computer Science International Program

**Academic Year** 2009

**Advisor** Asst. Prof. Dr. Sarun Intakosum

### ABSTRACT

CRC card is the one of techniques that helps to design object-oriented software. The problems of offline CRC card are project team must live in the same place and they have to type all details from CRC card paper to store in computer. Goal of this special project is to develop web application that provides online conferencing for defining CRC cards.

## **Acknowledgement**

We would like to express our gratitude to all those who support us to successfully finish this project. I want to thank Asst. Prof. Dr. Sarun Intakosum, our project advisor, who provided useful advices and comments all the time of research and writing of this thesis.

We are deeply indebted to Asst. Prof. Dr. Veera Boonjing and Asst. Prof. Dr. Jeeraporn Werapun for their good cares and helpful suggestions. We also want to thank every staff in the international office for information related to senior project.

My friends from Department of Computer Science International Program who supported us in our research work. I want to thank them for all help, goodwill, and friendship.

We are extremely grateful to say thank you at this opportunity.



**Miss Piyawan      Ketyungyuenwong**

**Miss Chuttraporn   Songnawarad**

**Miss Boontarika    Silpipat**

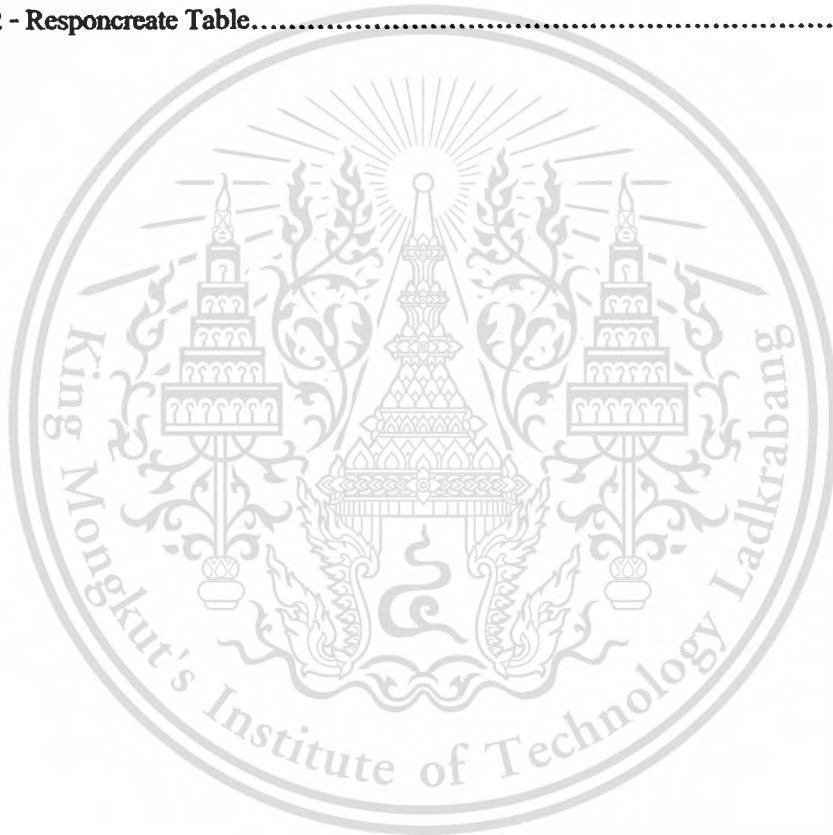
# Table of Contents

	<b>Page</b>
Abstract.....	I
Acknowledgement.....	II
Table of Contents.....	III
List of Tables.....	V
List of Figures.....	VI
<b>Chapter 1 Introduction.....</b>	<b>1</b>
1.1 Rational.....	1
1.2 Objective.....	1
1.3 Scope of study.....	2
1.4 Contribution.....	2
<b>Chapter 2 Related Literatures.....</b>	<b>3</b>
2.1 Introduction of CRC card.....	3
2.2 Survey of Current CRC-card and Conference Web-based Application.....	4
2.2.1 QuickCRC.....	4
2.2.2 Dimdim Web Conference.....	7
<b>Chapter 3 Development of Web-based CRC.....</b>	<b>14</b>
3.1 System Requirement.....	14
3.2 System Analysis.....	14
3.2.1 Use Case Diagram.....	14
3.2.2 Data Flow Diagram.....	15
3.3 System Design.....	18
3.3.1 The Entity-Relationship Diagram.....	18
3.3.2 Flowchart.....	22

	<b>Page</b>
<b>Chapter 4 Result</b> .....	26
4.1 System Operation.....	26
4.1.1 Create class names.....	28
4.1.2 Vote for choosing class name of each responsibility.....	31
4.1.3 Create and vote response name.....	32
4.1.4 Show CRC card.....	34
<b>Chapter 5 Conclusion</b> .....	36
5.1 Conclusion.....	36
5.2 Recommendation.....	36
<b>Appendix A User Manual</b> .....	37
A.1 Select Team Members (Chairman only).....	37
A.2 Upload Requirement (Uploader only).....	38
A.3 Conference Room.....	39
<b>Appendix B Database System</b> .....	47
<b>Appendix C Installation</b> .....	48
C.1 Installing Apache, PHP, MySQL and phpMyAdmin.....	48
C.2 Setup the database.....	48
C.3 Setup project folder.....	50

## List of Table

<b>Table</b>	<b>Page</b>
Table 3.1 - User Table.....	20
Table 3.2 - Requirement Table.....	21
Table 3.3 - Class Table.....	21
Table 3.4 - Responsibility Table.....	22
Table B.1 - Classcreate Table.....	47
Table B.2 - Responcreate Table.....	47



## List of Figures

<b>Figure</b>	<b>Page</b>
Figure 2.1: CRC Card.....	4
Figure 2.2: Create CRC Card.....	5
Figure 2.3: Front side of CRC card.....	5
Figure 2.4: Back side of CRC card.....	6
Figure 2.5: Inheritance Graph.....	6
Figure 2.6: CRC cards.....	7
Figure 2.7: Schedule & Start.....	8
Figure 2.8: Schedule Meeting.....	8
Figure 2.9: Conference room page.....	9
Figure 2.10: Video Conference.....	10
Figure 2.11: Share screen.....	11
Figure 2.12: Insert URL.....	11
Figure 2.13: Share web.....	12
Figure 2.14: Share whiteboard.....	12
Figure 2.15: Upload document page.....	13
Figure 2.16: Share document.....	13
Figure 3.1: Use Case Diagram.....	15
Figure 3.2: Data Flow Diagram.....	16
Figure 3.3: DFD fragment 1.....	16
Figure 3.4: DFD fragment 2.....	17
Figure 3.5: DFD fragment 3.....	17
Figure 3.6: DFD fragment 4.....	18
Figure 3.7: Entity-Relationship Diagram.....	19
Figure 3.8: Flowchart of managing user's permission.....	22
Figure 3.9: Flowchart of managing team members.....	23
Figure 3.10: Flowchart of upload requirement file.....	24

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, **VI** cite the document when use.

<b>Figure</b>	<b>Page</b>
Figure 3.11: Flowchart of brainstorm for defining CRC card.....	25
Figure 4.1: Admin permission page for managing user’s permission.....	26
Figure 4.2: Chairman permission page for managing team members.....	27
Figure 4.3: Uploader permission page for upload requirement file.....	28
Figure 4.4: Create class names in conference room.....	29
Figure 4.5: Choose for all class names of requirement.....	30
Figure 4.6: Create class names of uploader.....	31
Figure 4.7: Vote class name of each responsibility.....	32
Figure 4.8: Create response name page.....	33
Figure 4.9: Vote response name page.....	34
Figure 4.10: Show CRC card.....	35
Figure A.1: CRCConference’s chairman permission page.....	37
Figure A.2: CRCConference’s upload requirement page.....	38
Figure A.3: CRCConference’s conference room page.....	39
Figure A.4: CRCConference’s creating class names page.....	40
Figure A.5: CRCConference’s voting class names page.....	40
Figure A.6: CRCConference’s result of voting class names page.....	41
Figure A.7: CRCConference’s voting class name of responsibility page.....	42
Figure A.8: CRCConference’s creating response name page.....	43
Figure A.9: CRCConference’s creating reponse name page for chairman.....	44
Figure A.10: CRCConference’s voting response name page.....	45
Figure A.11: CRCConference’s result of voting response name page.....	46
Figure C.1: Create new database.....	49
Figure C.2: Import “crc.sql” file.....	49
Figure C.3: All tables in “crc” database.....	50
Figure C.4: C:\AppServ\www.....	51
Figure C.5: http://localhost/CRC/LoginFrame.php.....	51

## Chapter 1

### Introduction

#### 1.1 Rationale

CRC (Class Responsibility Collaborator) card is a brainstorming tool used for design classes. *Each card contains the responsibilities of a class and the collaborations of the class.* Procedures for designing CRC card are as follows: Chairman distributes a requirement to project team. After they read the requirement, they brainstorm for all class names that should have for this requirement. When they have class names, chairman distributes cards for each member equally. Then they brainstorm for responsibility and vote for the best class where that responsibility should be. Finally, they think about collaborators of each card.

For offline CRC card development, project team must travel to meet together at the same place. When they finish brainstorming, they have to type all the details from CRC card paper to store in computer.

Our project can solve all above problems by having online conferencing for defining online CRC cards and all data of online CRC cards are stored in database automatically.

#### 1.2 Objective

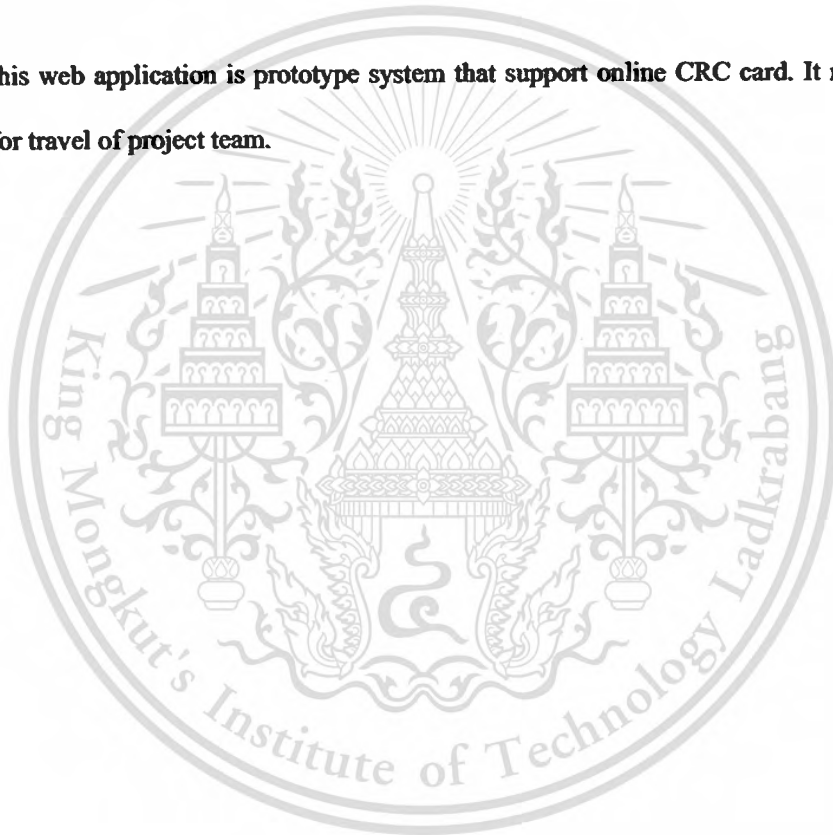
The objective of this project is to develop web application that support online CRC card for project team.

### 1.3 Scope of study

The system provides chatting and voting function to support collaboration among team developer to specify classes and responsibilities. The specified information is kept into the database for later use.

### 1.4 Contribution

This web application is prototype system that support online CRC card. It reduces expense and time for travel of project team.



## Chapter 2

### Related Literatures

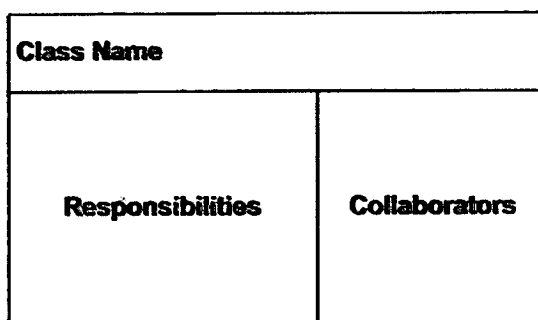
In this chapter, we introduce the meaning of CRC card and describe the features of QuickCRC and Dimdim application to be example that general online conference and CRC card application can do.

#### 2.1 Introduction of CRC card

Class Responsibility Collaboration (CRC) cards are a conference tool that use to design the object-oriented of software. Developers in a modeling, design process and bring it together for the development team to form a common understanding of an object oriented development project. A card is made for each class containing responsibilities and collaborators.

Procedure for designing CRC card as follows: Chairman distributes a requirement to project team. After they read the requirement, they brainstorm for all class names that should have for this requirement. When they have class names, chairman distributes cards for each member equally. Then they brainstorm for responsibility and vote for the best class where that responsibility should be. Finally, they think about collaborators of each card and typewrite all of the details to store in computer. They have to carefully typewrite because the details can easily change from original version on this step.

CRC card is consists of class name, responsibilities and collaborators which shown in *Figure 2.1*.



*Figure 2.1: CRC Card (Credit by <http://www.agilemodeling.com/artifacts/crcModel.htm>)*

### **Class Name**

Class name is name of the class and class name appears on the top of CRC card.

### **Responsibility**

Responsibility is a method that describes what the class can do.

### **Collaborators**

Collaborators are classes that have relation or share any resources with another class.

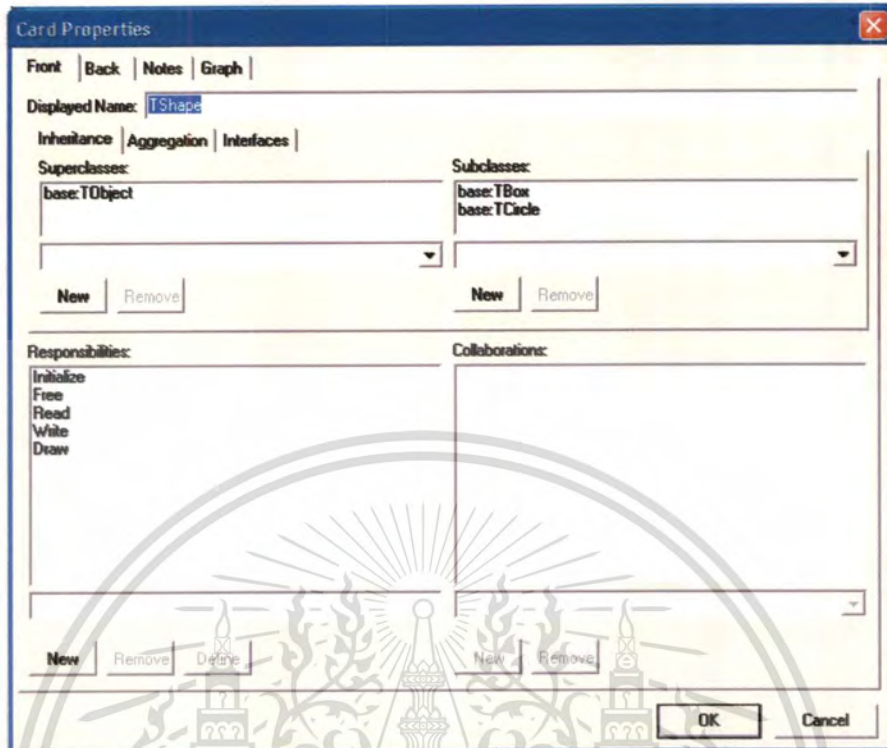
## **2.2 Survey of Current CRC-card and Conference Web-based Application**

### **2.2.1 QuickCRC**

QuickCRC is a commercial program for developing online CRC card. We can show some features that are come from QuickCRC user's guide because it's not free program.

#### **Features**

1. **Create CRC cards** – Users can create CRC card by typing information into all fields in Card Properties Dialog. Card Properties Dialog has 4 tap on the top: Front, Back, Notes and Graph for users to create card information as shown in *Figure 2.2*.



*Figure 2.2: Create CRC Card*

When users finish create CRC card, the program will show the cards. In the front side of CRC card includes Superclasses, Subclasses, Responsibilities and Collaborators as shown in *Figure 2.3*.

TWindow	
Superclasses: TEventHandler	
Subclasses:	
Responsibilities:	Collaborators:
Initialize	
Close	
DoMemCommand	
DoMouseCommand	TWindow
Draw	TWindow, TPalette,
SetupMems	
Erase	

*Figure 2.3: Front side of CRC card*

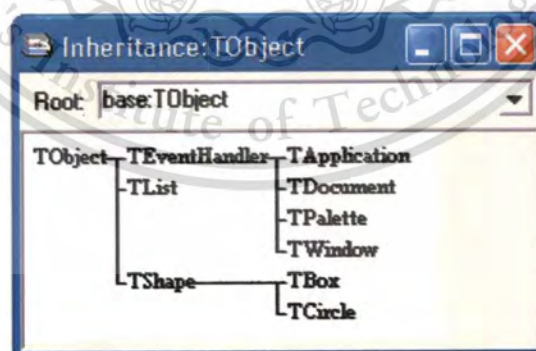
In the backside of CRC card includes its description and list of attributes as shown in

*Figure 2.4.*

<b>TWindow</b>	Flip
<b>Description:</b>	
this class implements a window through which	
<b>Attributes:</b>	
fDocument	
fPalette	

*Figure 2.4: Back side of CRC card*

2. On-screen editing – Users can use the Drag tool for adding, editing, moving or deleting information on cards.
3. Inheritance Graph – The QuickCRC program can generate inheritance graphs from information on CRC cards as shown in *Figure 2.5*. These diagrams show relationships of all cards that base on Root which is chosen by user.



*Figure 2.5: Inheritance Graph*

4. Complete the design of CRC card – program will show all cards as shown in *Figure 2.6*.

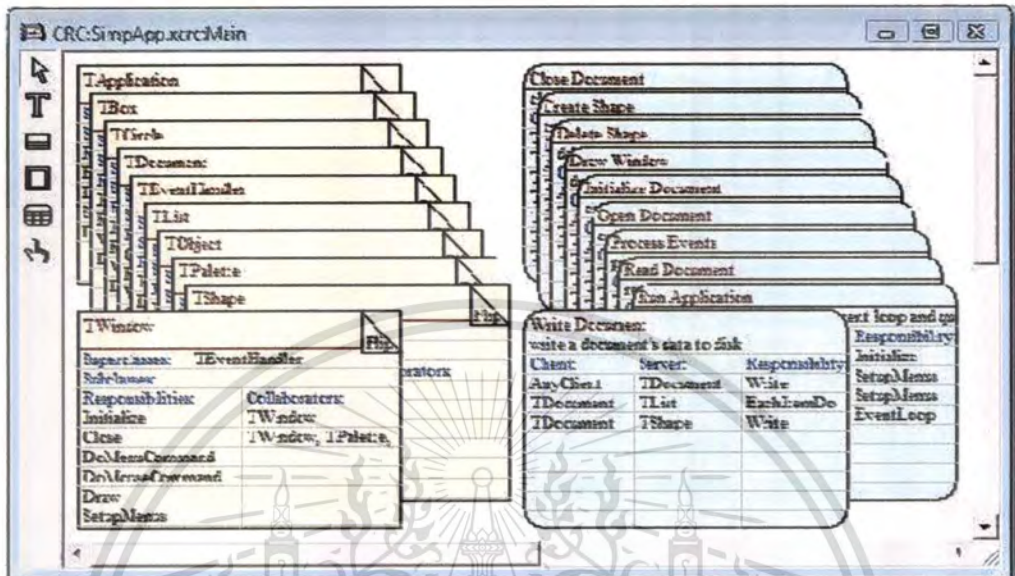


Figure 2.6: CRC cards

### 2.2.2 Dimdim Web Conference

Dimdim Web Conference is online conference that allows maximum twenty people can come to meeting room and allows chairman to share video, voice, file (.pdf and .ppt and .pptx) and whiteboard for the meeting members.

#### Features

1. First page of Dimdim after login, it will allow users to choose option between schedule and start as shown in *Figure 2.7*.

**Account Info**  
 You have a Doodle Free account.  
 • You can host web meetings with up to 250 people  
 • Upgrade to do more

**Learn More**  
 Your account is missing features.  
 Share 2 webcams. Host larger web meetings. Customize your room. Get faster service & more.  
 Try all our premium features for 30 days - free!  
 Office Employees save up to 24% on Annual Plans

**Compare Accounts**  
 Want more?  

	Your Plan	Pro	Webinar
Basic Meetings	•	•	•
Advanced Tools	•	•	•
Webinar Tools	•	•	•

 Try any Premium plan for 30 days Free!

**Instant Action Buttons**

**SCHEDULE**  
 Plan a meeting with 300+ people

**START**  
 Jump into a meeting now

Figure 2.7: Schedule & Start

Schedule meeting which shown in Figure 2.8 – this option allows chairman to specify the details of an appointment and sent the schedule to team members that was invited.

**Schedule Meeting**

General | Features | Dial-in | Widget

Room Name: default | Room URL: <http://my.doodle.com/laziqueen>

Meeting Name: laziqueen | Repeat Option: Once Only

Agenda: CRC-card | Start Date: April 21, 2010 | End Date: April 21, 2010

Invitees: boontarika\_sil@hotmail.com; piyawan@hotmail.com; 0 Registrations | Start Time: 1:15 AM | Timezone: GMT+07:00 Bangkok, Hano

Keys: Meeting Key | Meeting key makes meeting private

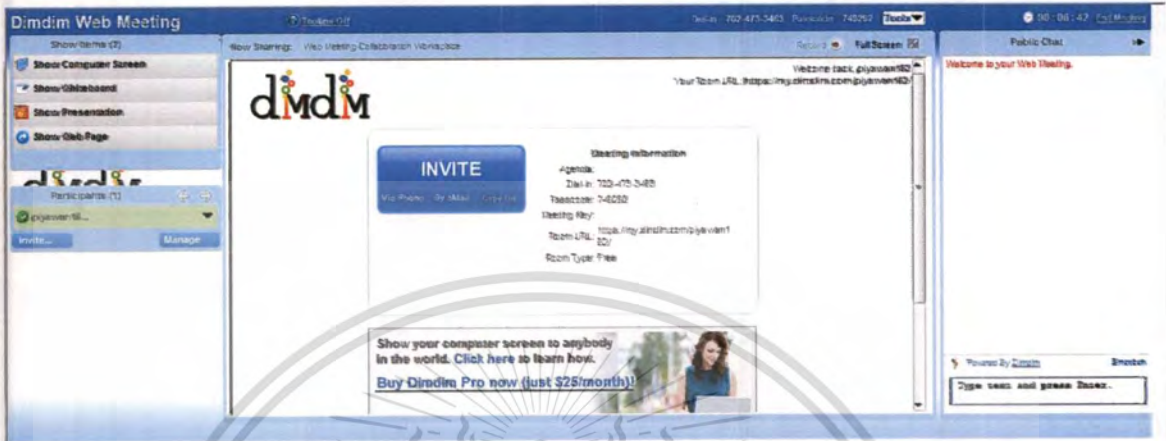
On Join:  Require email |  Pre-registered email only

Send email when meeting starts

Schedule

Figure 2.8: Schedule Meeting

Start – it will go directly to conference room and conference room has the window that looks like *Figure 2.9*.



*Figure 2.9: Conference room page*

2. Video conference on dimdim – Dimdim can show only chairman video and the video is on everywhere of the window that users want. Chairman can talk with others, freeze the video or mute the sound. This is shown in *Figure 2.10*.

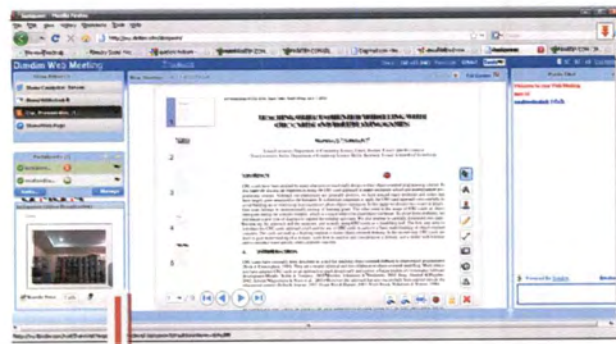
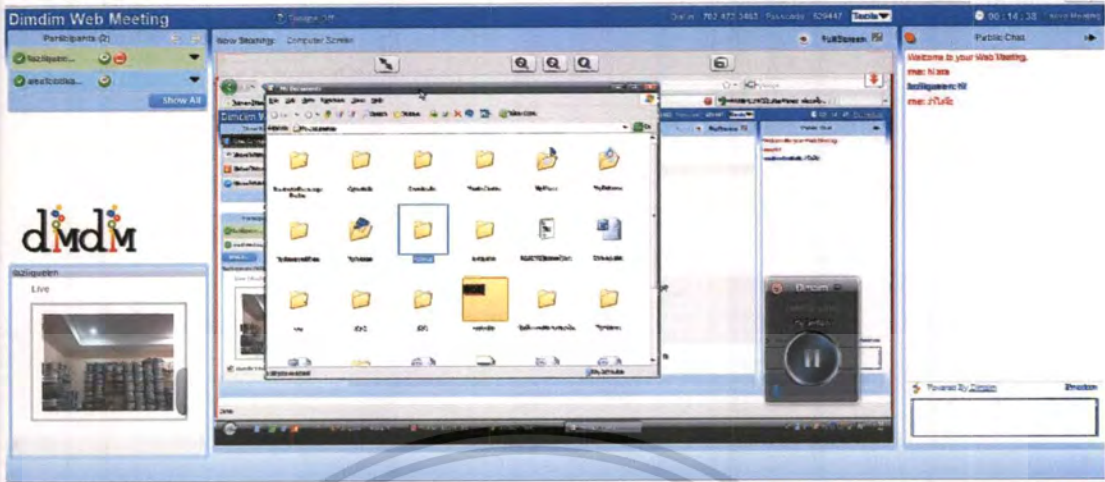


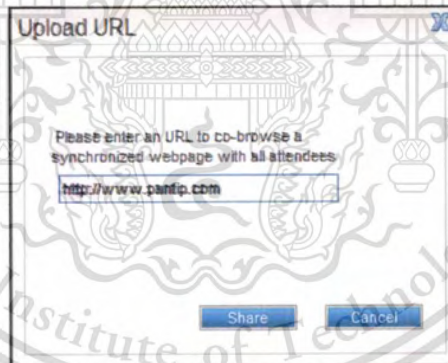
Figure 2.10: Video Conference

3. Show Computer Screen – Chairman can share his screen to team members in meeting room. In this feature, all members can see everything that chairman does on his screen as shown in *Figure 2.11*.



*Figure 2.11: Share screen*

4. Show Web Page – Chairman can share web page to members by insert the url of web page as shown in *Figure 2.12*.



*Figure 2.12: Insert URL*

Then all members in the meeting room will see the same web page as shown in *Figure*

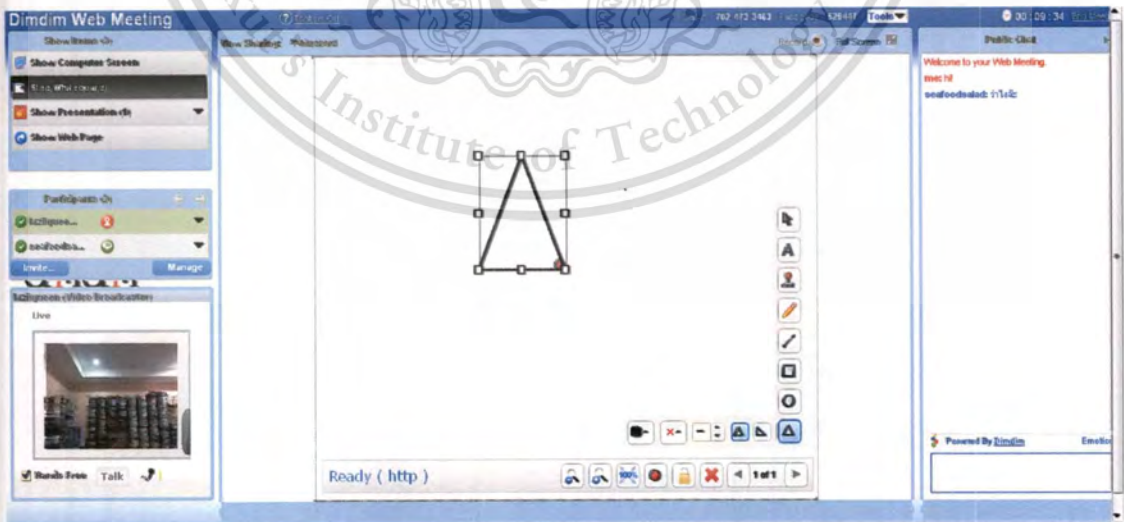
2.13.



*Figure 2.13: Share web*

5. Show Whiteboard – Chairman can use whiteboard option to write or draw things that help chairman to describe problems and requirement for members easily to understand.

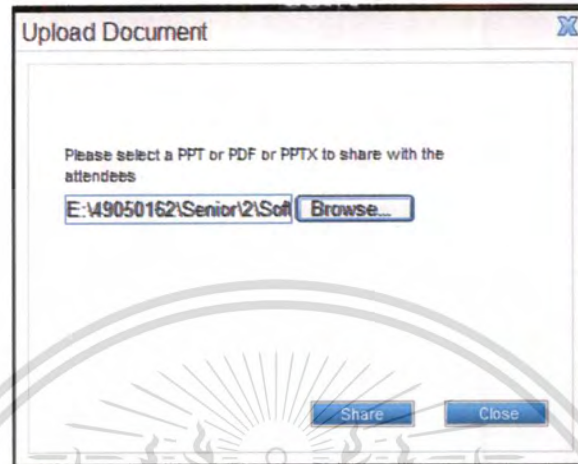
This is shown in *Figure 2.14*.



*Figure 2.14: Share whiteboard*

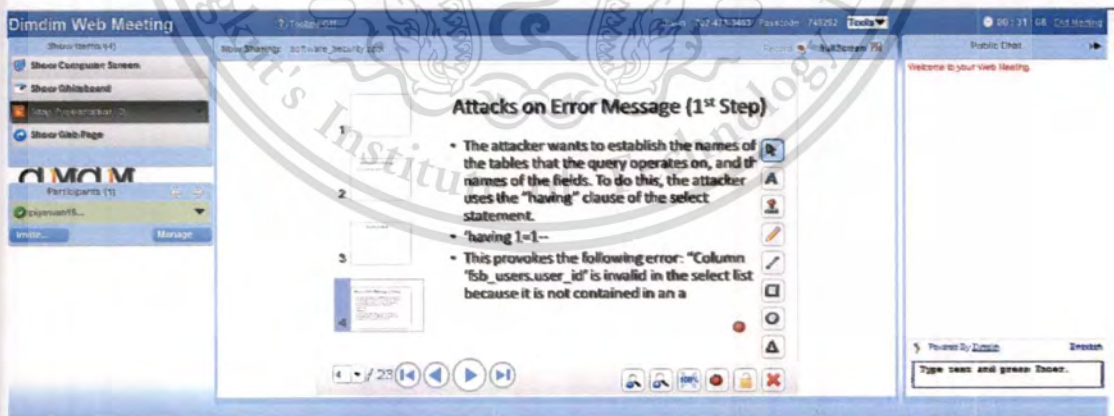
6. Show Presentation – Chairman can share file (.pdf or .ppt or .pptx) to other members.

First, chairman has to upload file as shown in *Figure 2.15*.



*Figure 2.15: Upload document page*

Then Dimdim will automatically share that file to all members as shown in *Figure 2.16*.



*Figure 2.16: Share document*

## Chapter 3

# Development of Web-based CRC

### 3.1 System Requirement

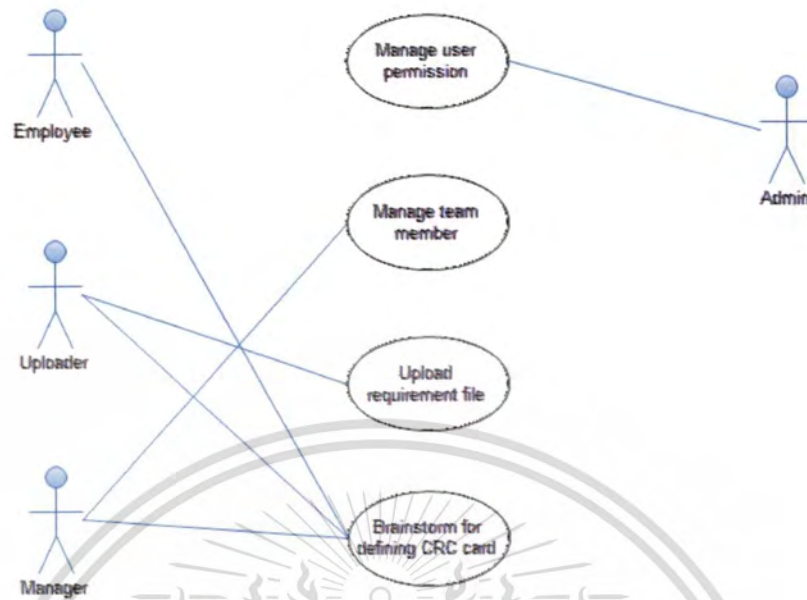
According to the survey from Chapter 2, prototype of our system should have the following basic functions:

1. System has process to assign class names and responsibilities for each requirement.
2. System allows users to use this website as intermediate for brainstorming by chatting and voting.

### 3.2 System Analysis

#### 3.2.1 Use Case Diagram

We analyze the system by creating use case diagram that shown in Figure 3.1. The use case diagram is shown that administrator can manage user's permission, manager can manage team members, uploader can upload requirement file, and all users except administrator can go to meeting to define CRC cards.



*Figure 3.1: Use Case Diagram*

### 3.2.2 Data Flow Diagram

We analyze data flow diagram to show all the main processes for an information system. First, all user must log in to our system and then manager will manage team members for brainstorming for each requirement. After that uploader will upload requirement file and finally all users in team members will go to meeting room for brainstorming to define CRC card. The details of data flow diagram will be shown from *Figure 3.2* to *Figure 3.6*.

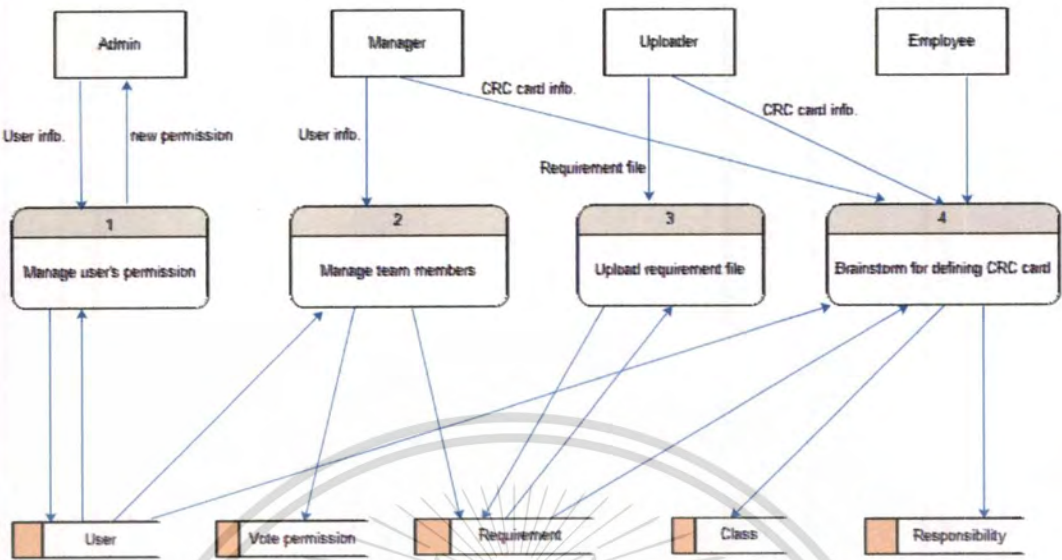


Figure 3.2: Data Flow Diagram

Process of administrator for managing user's permission is shown in Figure 3.3. Administrator chooses user from username and then assigns new permission for that user.

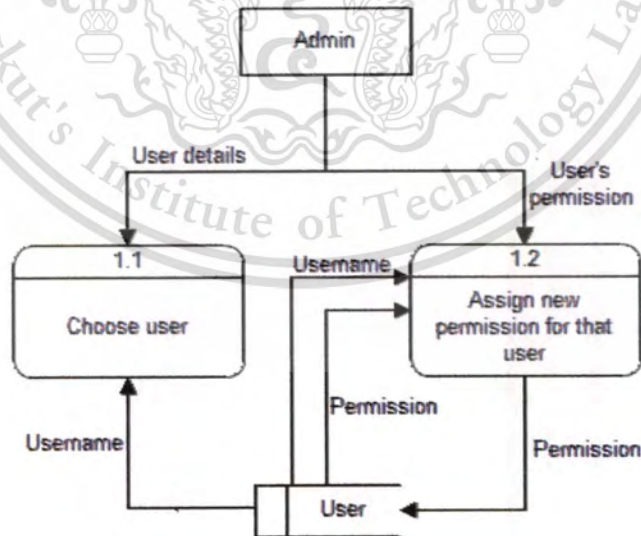


Figure 3.3: DFD fragment 1

Process of chairman for managing team members is shown in *Figure 3.4*. First, chairman must assign requirement name and then select uploader and team members for that requirement.

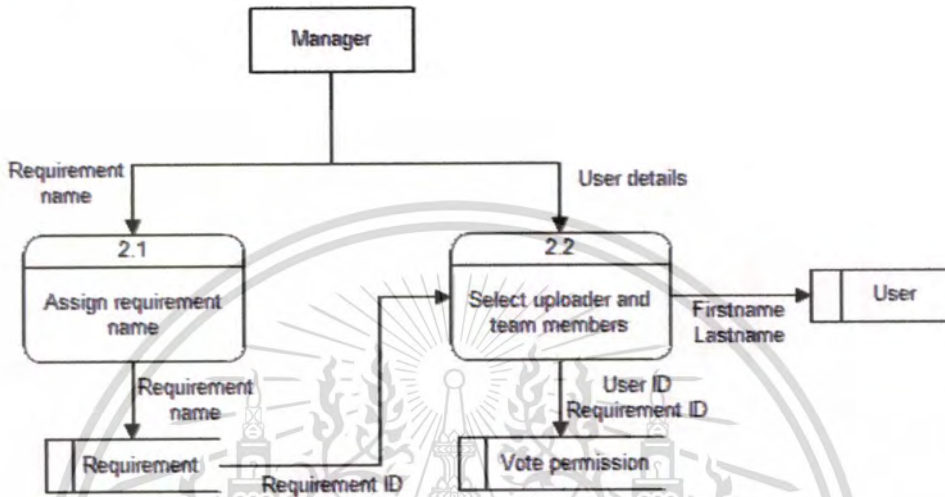


Figure 3.4: DFD fragment 2

Process of uploader to upload requirement file is shown in *Figure 3.5*. Uploader must choose requirement name and requirement file.

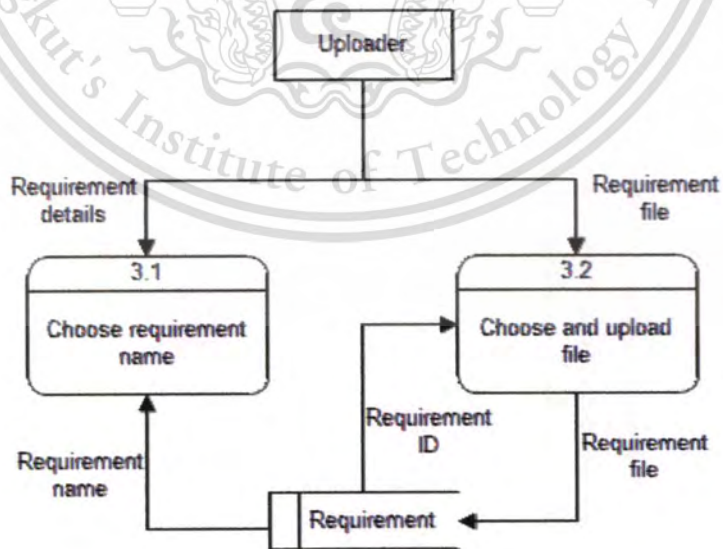
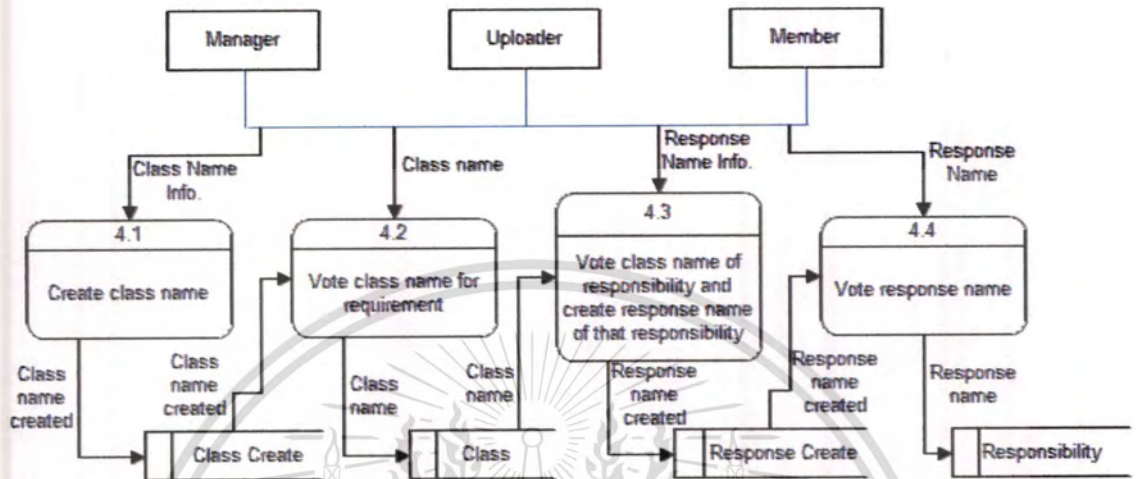


Figure 3.5: DFD fragment 3

Process of brainstorming for defining CRC card is shown in *Figure 3.6*. Users create and vote class names and responsibilities, respectively.

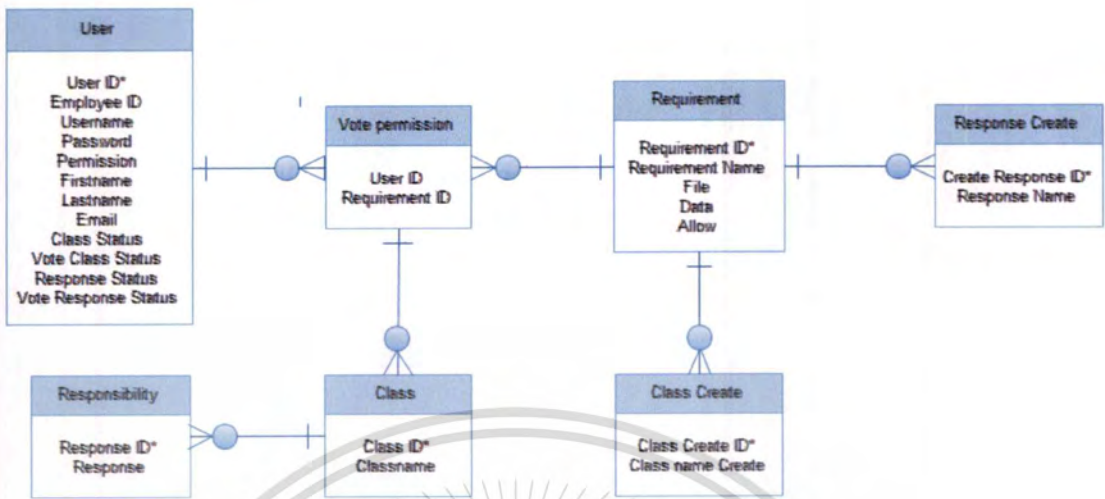


*Figure 3.6: DFD fragment 4*

### 3.3 System Design

#### 3.3.1 The Entity-Relationship Diagram

From data flow diagram, we can design the entity-relationship diagram which shown in *Figure 3.7*.



**Figure 3.7: Entity-Relationship Diagram**

The details of some tables will be shown from table 3.1 to table 3.4. The rest will be explained in Appendix B.



User table stores user information.

Table 3.1 - User Table

No.	Field Name	Data type	Description
1	id	integer	id
2	eid	varchar	employee id
3	username	varchar	username
4	password	varchar	password
5	permission	varchar	user's permission
6	fname	varchar	first name
7	lname	varchar	last name
8	email	varchar	e-mail
9	voteclasstatus	integer	status of creating class names for requirement
10	votesstatus	integer	status of voting class name of each responsibility
11	responstatus	integer	status of creating response name
12	votesresstatus	integer	status of voting response name of each responsibility

Requirement table stores all requirements in the system.

Table 3.2 - Requirement Table

No.	Field Name	Data type	Description
1	rid	integer	requirement id
2	rname	varchar	requirement name
3	file	longblob	requirement file
4	date	datetime	created date and time
5	allow	integer	allowable meeting room

Class table stores class names related with requirement.

Table 3.3 - Class Table

No.	Field Name	Data type	Description
1	cid	integer	class id
2	rid	integer	requirement id
3	classname	varchar	class name

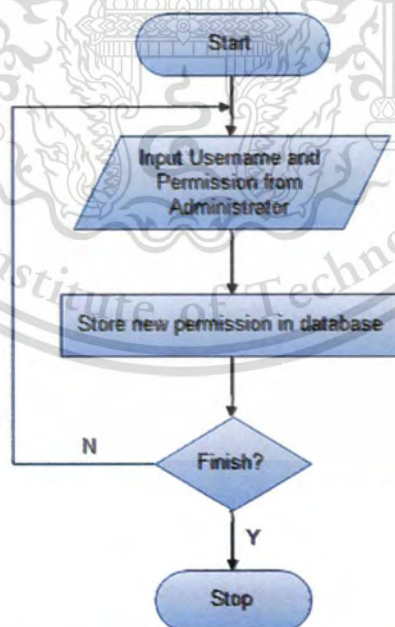
Responsibility table stores responsibilities related with class name and requirement.

Table 3.4 - Responsibility Table

No.	Field Name	Data type	Description
1	resid	integer	response id
2	cid	integer	class id
3	respond	text	responsibilities

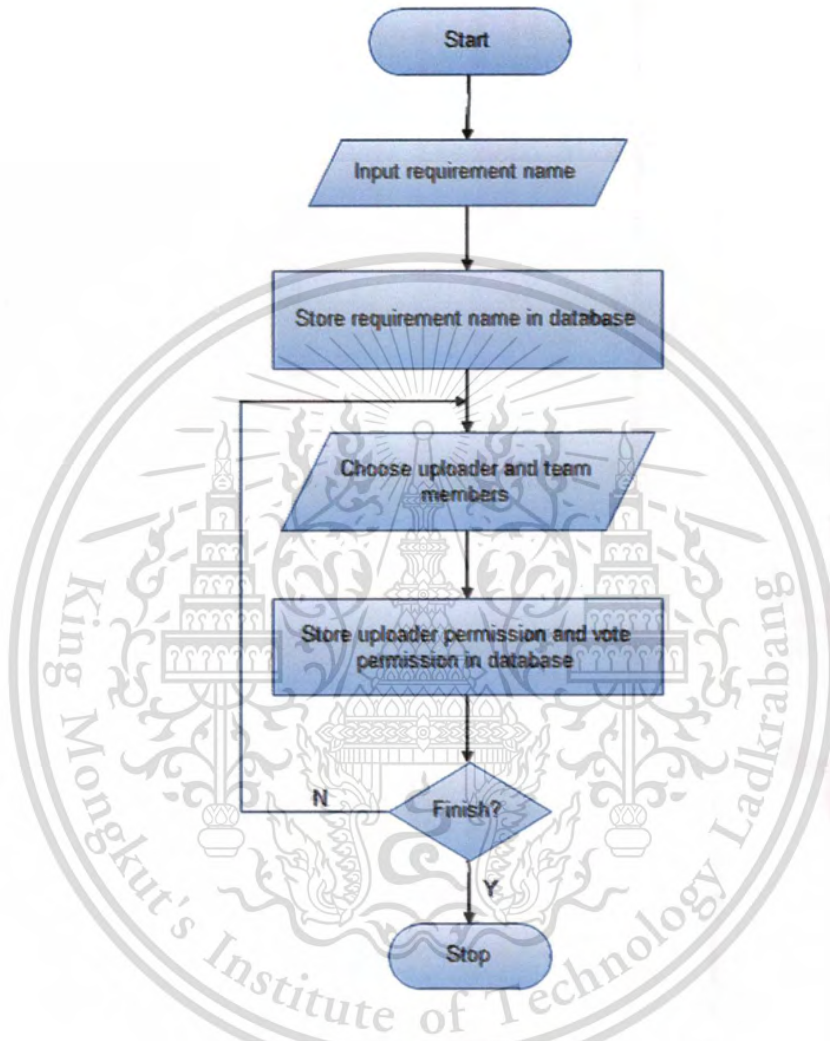
### 3.3.2 Flowchart

In this section, we design flowchart to show workflow of our system, which are shown from *Figure 3.8* to *Figure 3.11*. For *Figure 3.8*, this flowchart shows process of managing user's permission of Administrator.



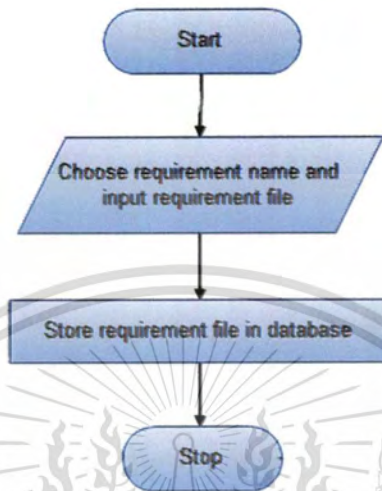
*Figure 3.8: Flowchart of managing user's permission*

For *Figure 3.9*, this flowchart shows process of managing team members of chairman.



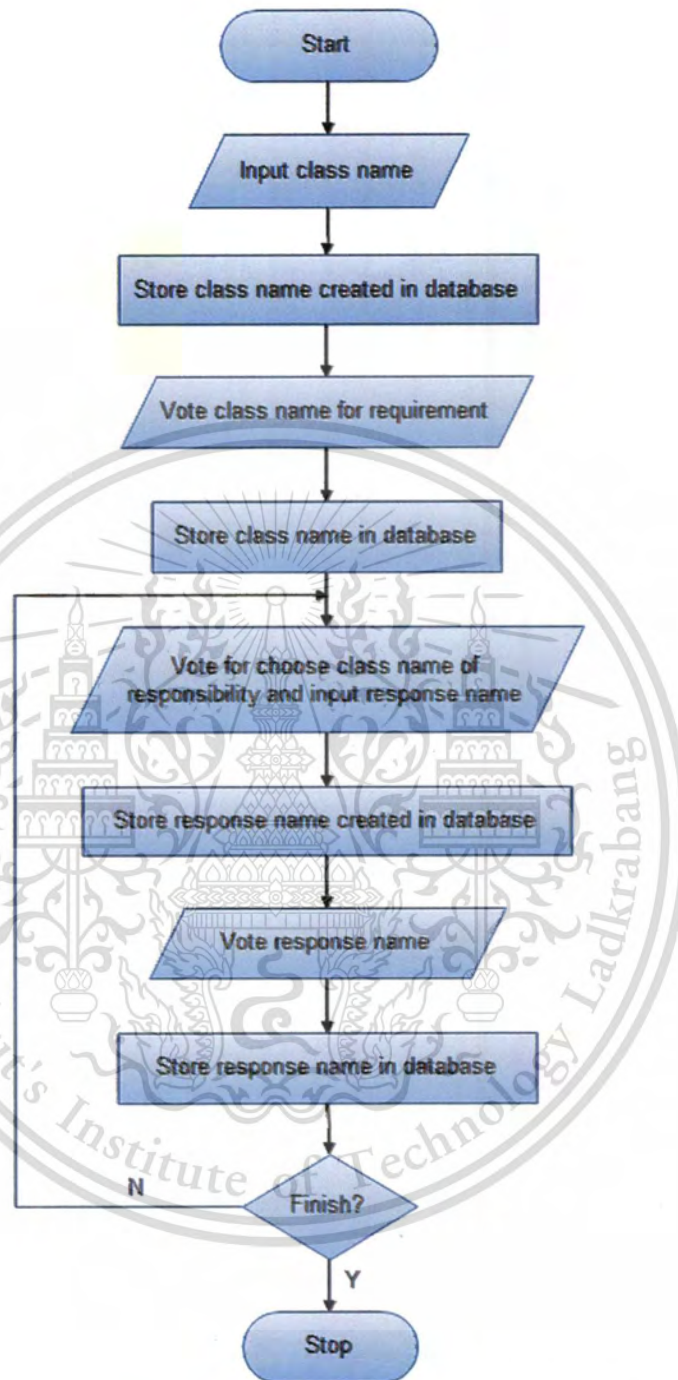
*Figure 3.9: Flowchart of managing team members*

For *Figure 3.10*, this flowchart shows process of upload requirement file of uploader.



*Figure 3.10: Flowchart of upload requirement file*

For *Figure 3.11*, this flowchart shows process of brainstorm of defining CRC card in meeting room.



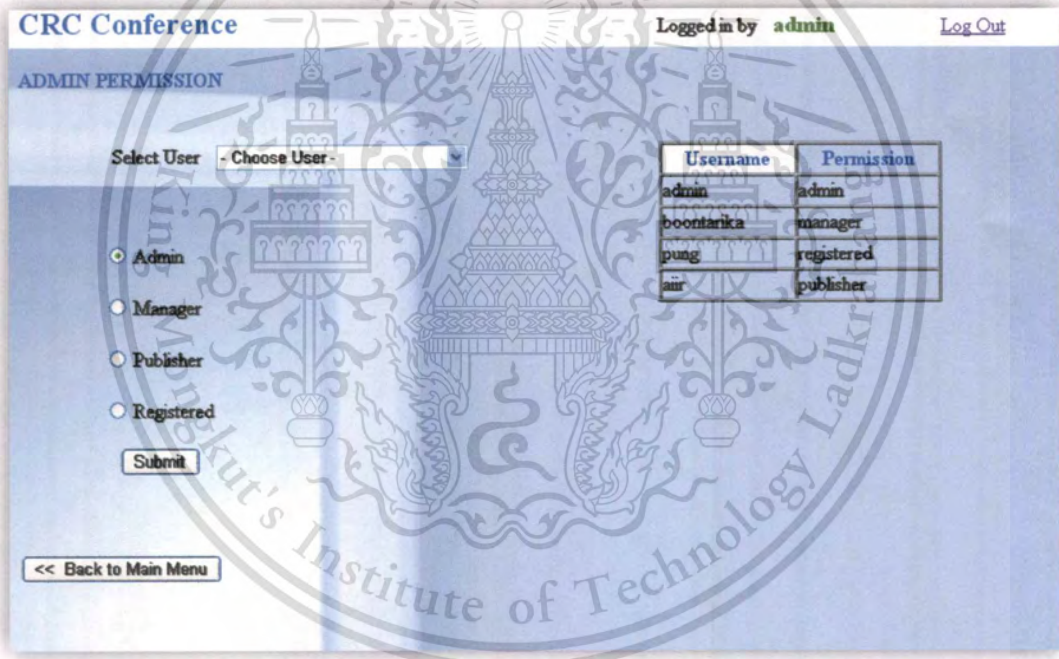
*Figure 3.11: Flowchart of brainstorm for defining CRC card*

## Chapter 4

### Result

#### 4.1 System Operation

In this section, we'll show the results of our prototype system. Our web application has 4 permissions: Administrator, Manager, Publisher and Registered. Administrator can manage user's permission as shown in *Figure 4.1*.



*Figure 4.1: Admin permission page for managing user's permission*

Manager permission is meant to chairman that can manage team members as shown in

Figure 4.2.

CRC Conference Logged in by **boontarika** [Log Out](#)

CHAIRMAN PERMISSION

Requirement Name

Uploader

Member

Member List	
<input type="checkbox"/>	Piyawan Ketyungyuenwong
<input type="checkbox"/>	Chuttraporn Songnawarad

\* กรุณาใส่ชื่อ Requirement เมื่อคุณ upload และเลือกสมาชิก !!!

Figure 4.2: Chairman permission page for managing team members

Publisher permission is meant to uploader that can upload requirement file as shown in

Figure 4.3.

**CRC Conference** Logged in by **aiir** [Log Out](#)

**FILE UPLOADER**

ส่วนของการ Upload File

Requirement Name : - Pls Select Meeting Room Requirement File :

ส่วนของการ Add Class

Class Name

Class Name List

Pls Select and Upload File Requirement Before Add Class !!!

Name	Delete

Figure 4.3: Uploader permission page for upload requirement file

Registered permission is meant to general user. Manager, Publisher and Registered can brainstorm for defining CRC card.

#### 4.1.1 Create class names

For creating class names, this web application has 2 ways:

- I. *Figure 4.4* illustrates the first way of creating class names. Users can brainstorm for creating and voting class names.

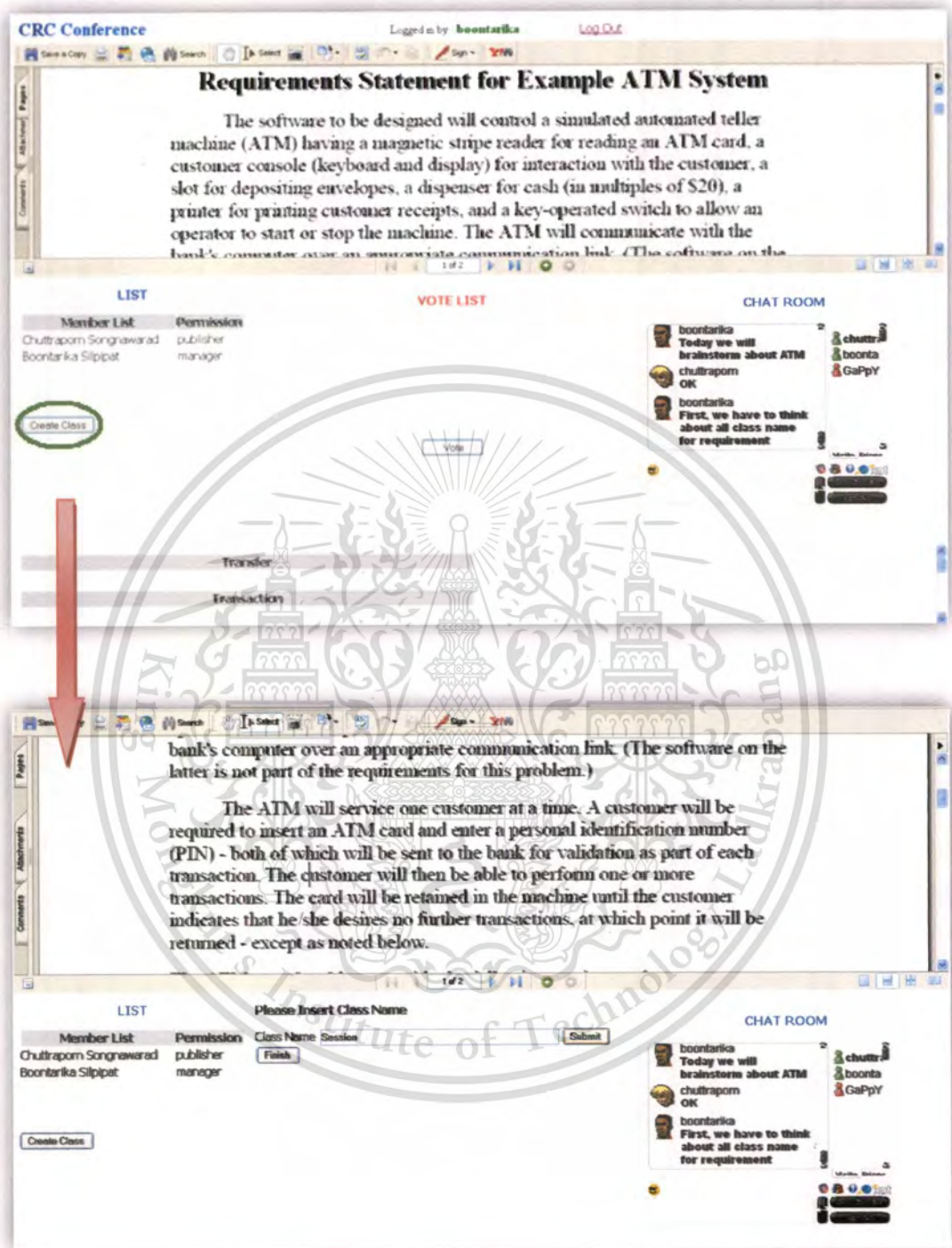
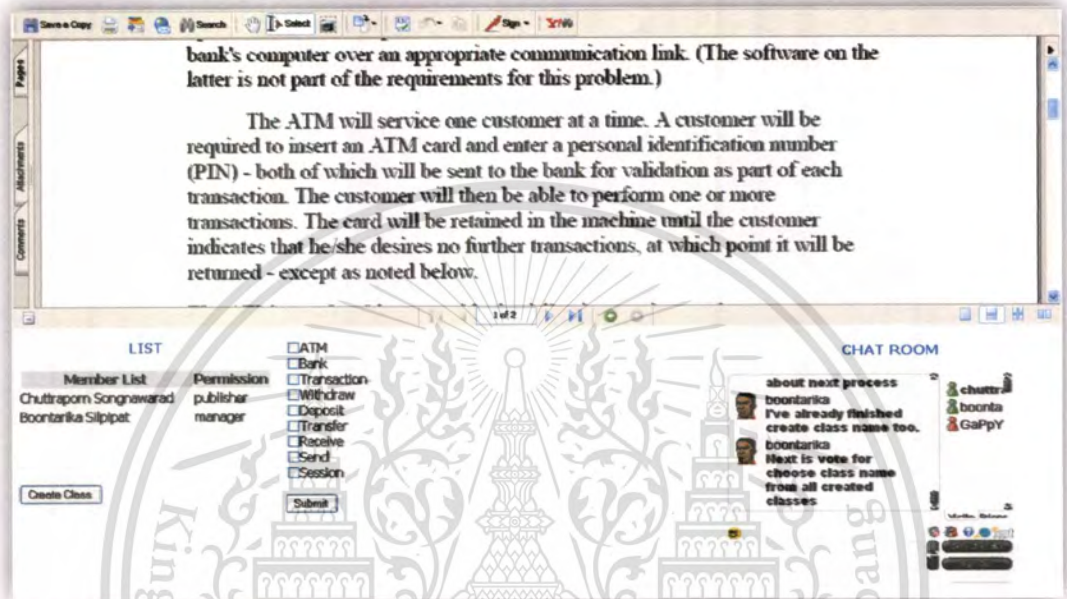


Figure 4.4: Create class names in conference room

When users finish creating class names, system will show choosing class names page automatically as shown in *Figure 4.5*. Each user can choose more than one class names.



*Figure 4.5: Choose for all class names of requirement*

- II. *Figure 4.6* illustrates the second way of creating class names. Uploader can create or delete class names.

The screenshot shows a web interface for a 'CRC Conference'. At the top, it says 'Logged in by pung' and 'Log Out'. Below that is the 'FILE UPLOADER' section. There is a red heading 'Upload File' and a 'Requirement Name' dropdown menu with the text '-Pls Select Meeting Room'. To the right is a 'Requirement File' input field with 'Browse...' and 'Upload' buttons. Below this is another red heading 'Add Class'. There is a 'Class Name' input field and an 'Add & Check Class' button. Underneath is a 'Class Name List' table with two columns: 'Name' and 'Delete'. The table contains four rows: 'balle', 'fanta', 'pepsi', and 'cola', each with a 'Delete' link. At the bottom left, there is a button labeled '<< Back to Main Menu'.

Name	Delete
balle	Delete
fanta	Delete
pepsi	Delete
cola	Delete

*Figure 4.6: Create class names of uploader*

#### 4.1.2 Vote for choosing class name of each responsibility

Vote for choosing class name of each responsibility page will contain list of all class names of that requirement. Users will be able to choose class name which is needed for each responsibility. This is shown in *Figure 4.7*.

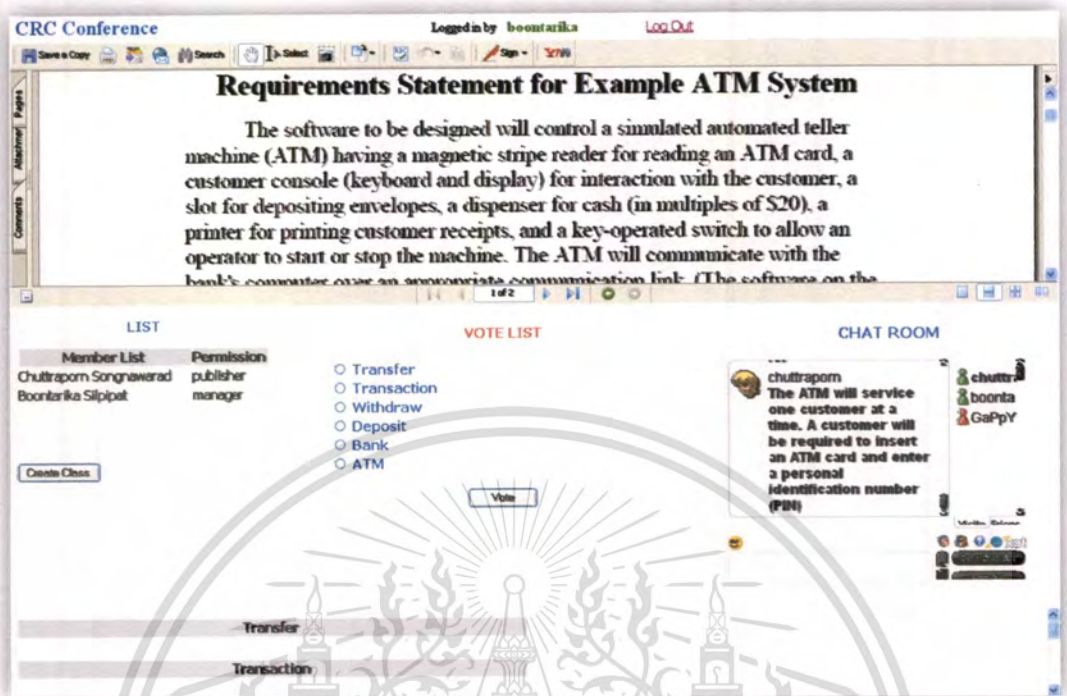
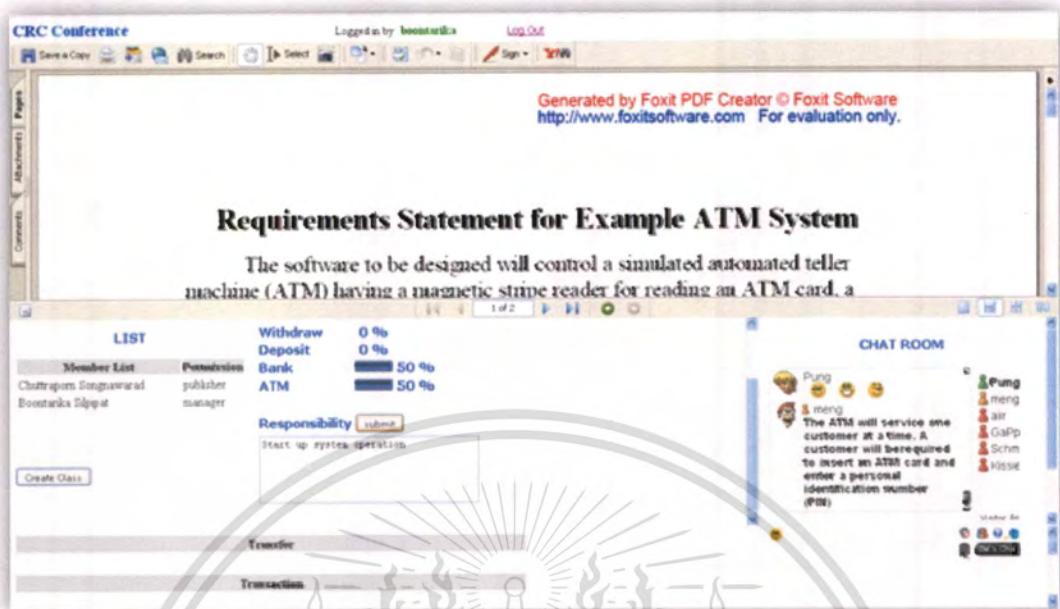


Figure 4.7: Vote class name of each responsibility

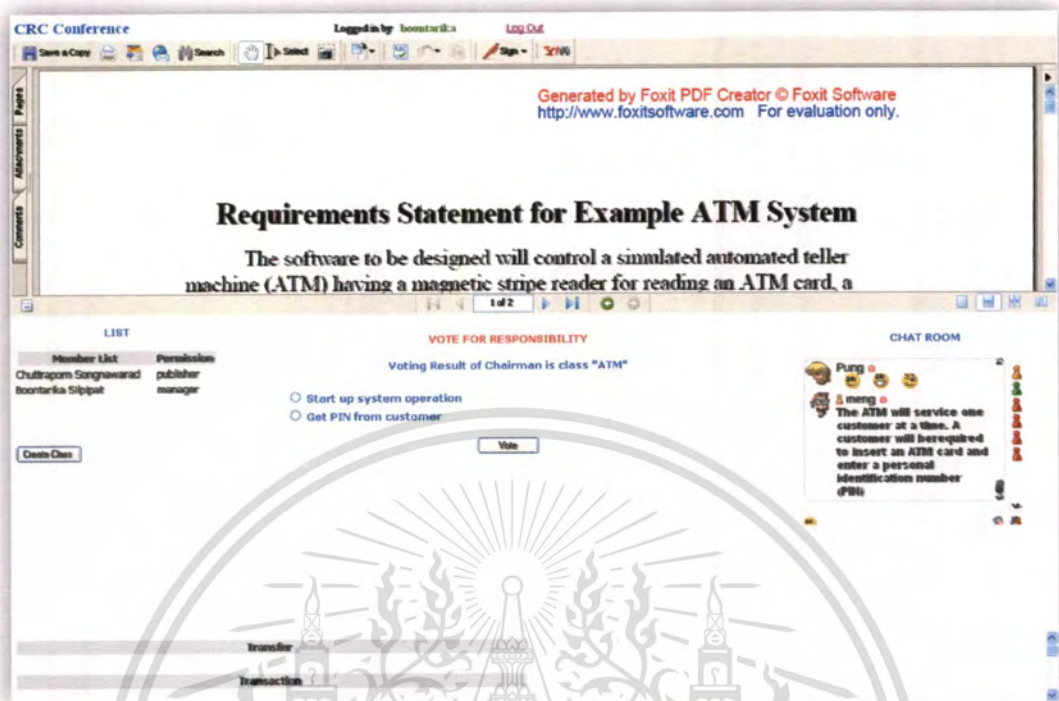
#### 4.1.3 Create and vote response name

Users will be able to add name of responsibility which is shown in Figure 4.8.



*Figure 4.8: Create response name page*

After all users finish adding response name, system will show voting response name page which is shown in *Figure 4.9*.



*Figure 4.9: Vote response name page*

#### 4.1.4 Show CRC card

Users can look CRC card and order to print it. This is shown in *Figure 4.10*.

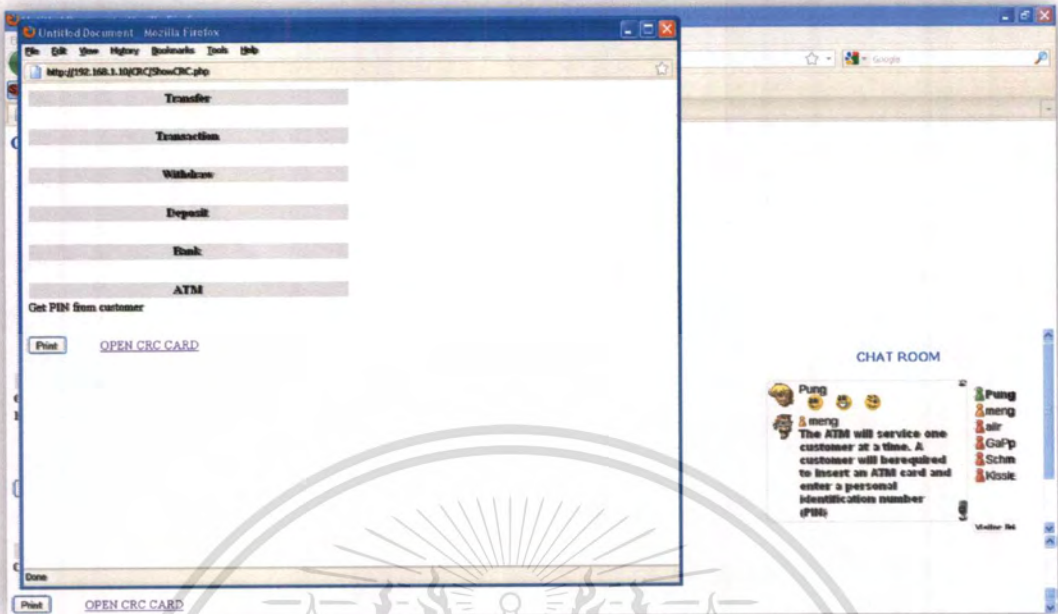
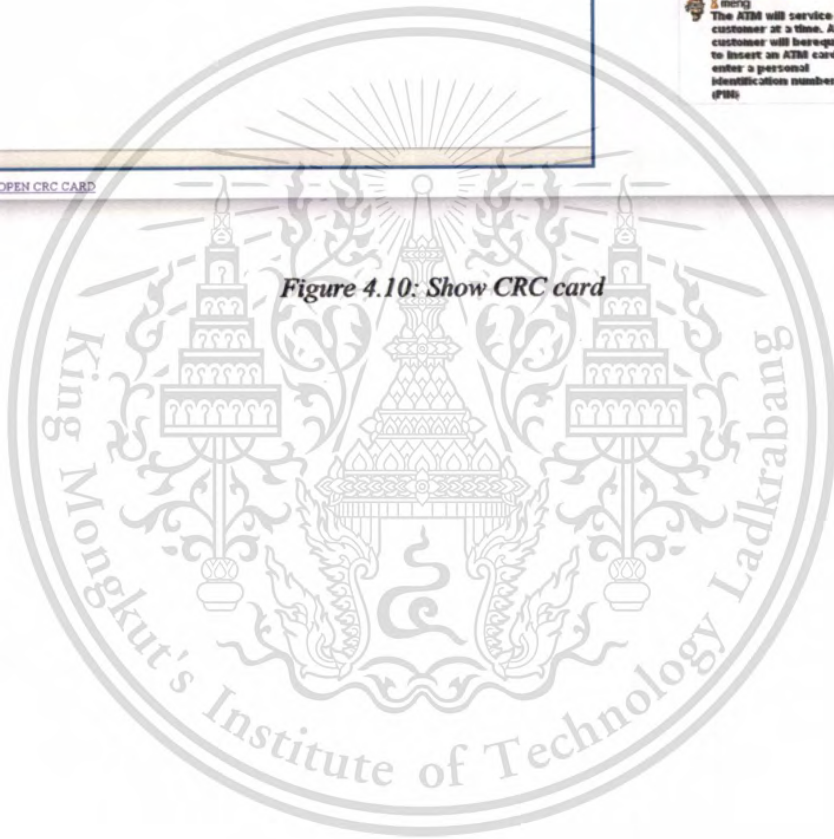


Figure 4.10: Show CRC card



## Chapter 5

### Conclusion

#### 5.1 Conclusion

For offline CRC card development, the project team must meet together by face to face and they have to type all the data from CRC card paper to database. So we have developed this web application to get rid of those problems. Our web application has chat room for chatting among members in meeting room, voting window is used for users to vote the commitment or class among members and they can present responsibility. Moreover they can see the history of CRC card and print it. All of above can increase user comfortable.

#### 5.2 Recommendation

Now our web application is prototype system, there are many areas where we can improve our system. For example:

1. For prototype system, user cannot make an appointment with other users. The system should be able to display calendar that allow users to add and show meeting day and free time of each user.
2. Our web application can keep information of CRC card in database but cannot take the information for creating class diagrams.

## Appendix A

### User Manual

#### A.1 Select Team Members (Chairman only)

- I. Add requirement name in Requirement Name text field
- II. Choose uploader for that requirement name
- III. Select team members by mark in check box field in Member table and chairman must select uploader name in this field again
- IV. Click Submit

**CRC Conference** Logged in by **pung** [Log Out](#)

**CHAIRMAN PERMISSION**

Requirement Name

Uploader **- Choose User -**

**Member**

Member List	
<input type="checkbox"/>	Boontarika Silpipat
<input type="checkbox"/>	Chuttraporn Songnawarad

*Figure A.1: CRCConference's chairman permission page*

## A.2 Upload Requirement (Uploader only)

- I. In the path of Upload File, choose requirement name and click Browse... for choose requirement file and then click Upload
- II. In the path of Add Class, uploader can add add class names in Class Name text field and Click Add & Check Class. Uploader can also delete class name by click Delete after that class name in Class Name List table

**CRC Conference** Logged in by **pung** [Log Out](#)

**FILE UPLOADER**

ส่วนงาน Upload File

Requirement Name :  Requirement File :

ส่วนงาน Add Class

Class Name

**Class Name List**

Name	Delete
baile	<input type="button" value="Delete"/>
fanta	<input type="button" value="Delete"/>
pepsi	<input type="button" value="Delete"/>
cola	<input type="button" value="Delete"/>

Figure A.2: CRCConference's upload requirement page

### A.3 Conference Room

On the top of this page, the system will be shown requirement that was uploaded by uploader. In the bottom of the page, the system will be shown the list of class names but it will not have anything in the first. On the left side of the page, the system will be shown list of all users in the conference room and the button that uses for creating and voting class names. On the right of the page, we will have chat room for user to chat or consult with others.

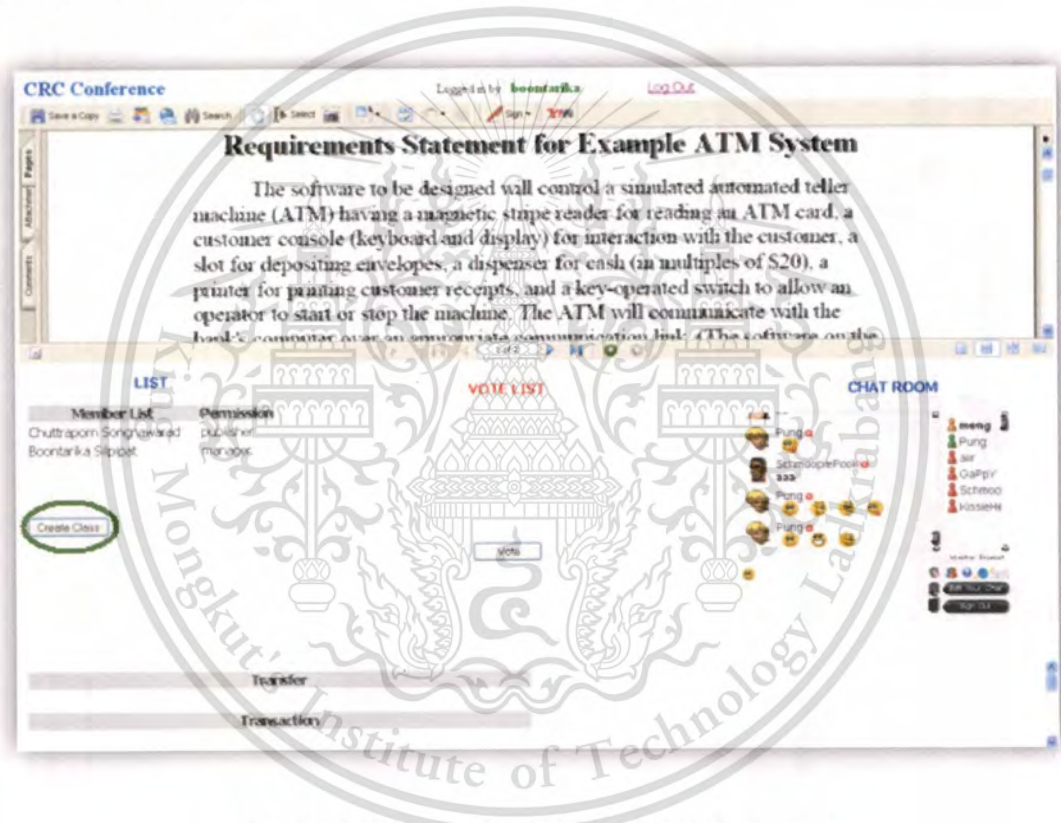


Figure A.3: CRCConference's conference room page

First, users must create class names for that requirement by click on "Create Class" button. Then the system will be redirect to the page for creating class names as shown in Figure A.4. For this page, users can create class names in Class name text field as much as users want and then click "Submit" button. Users must click "Finish" when they finish create class names.

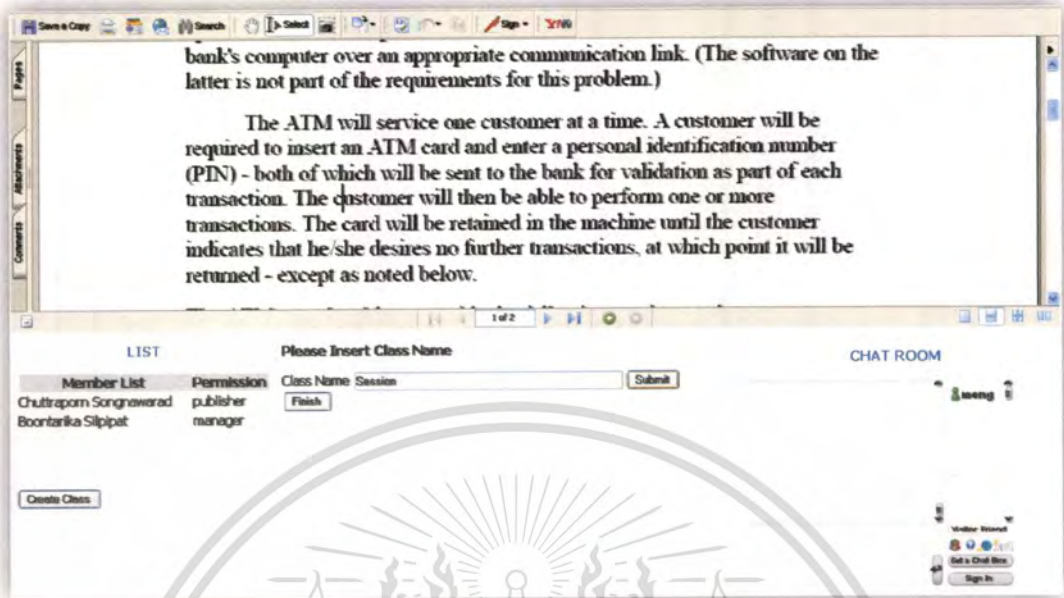


Figure A.4: CRCConference's creating class names page

When users finish creating class names, the system will show the page for voting class names as shown in Figure A.5.

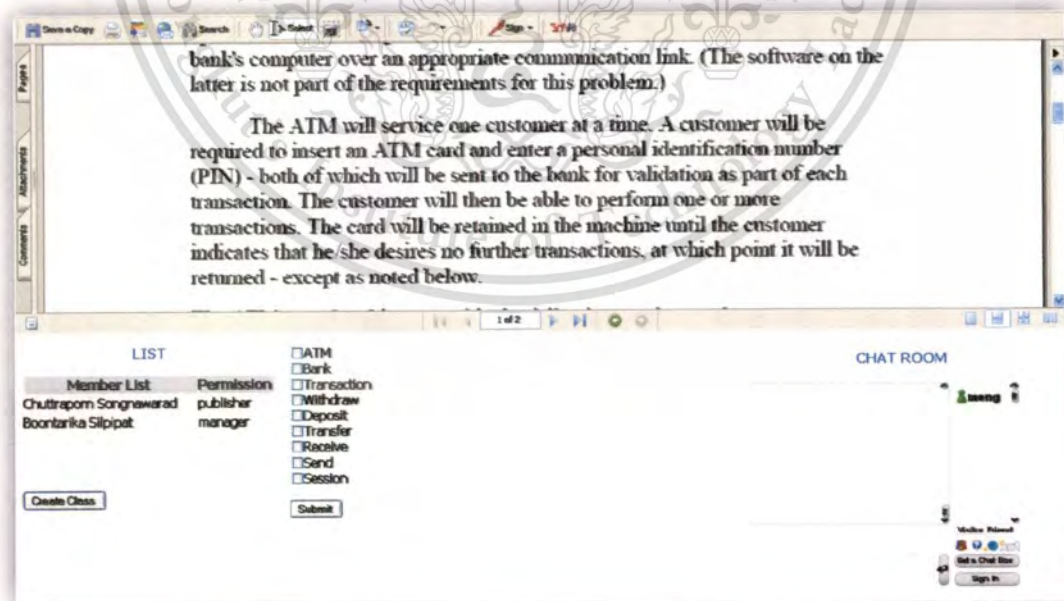
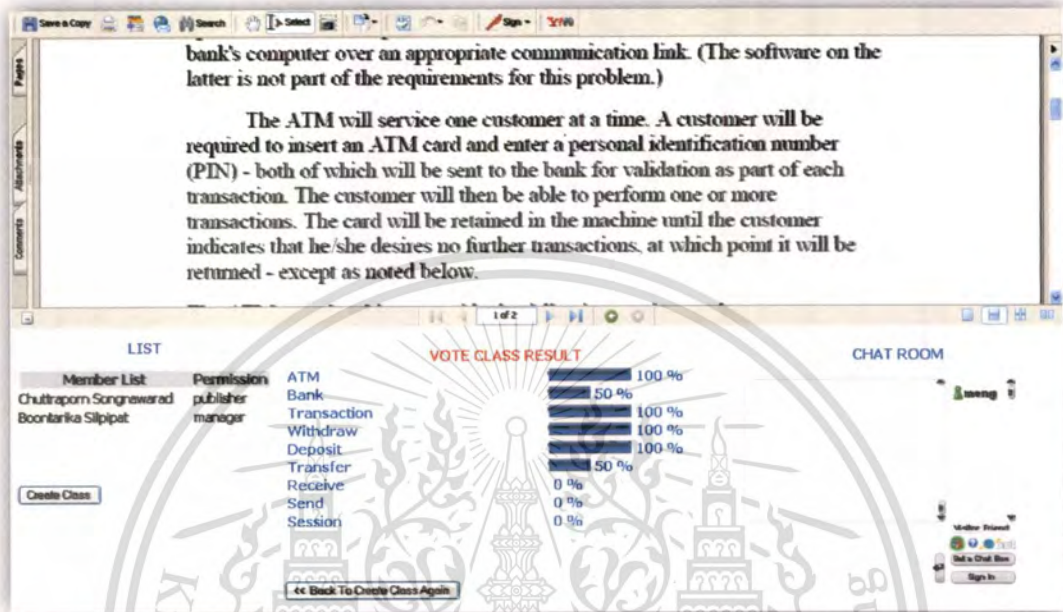


Figure A.5: CRCConference's voting class names page

Each user can choose more than one class names. For the result as shown in *Figure A.6*, the system will store class names that are chosen more than or equal 50% into database.



*Figure A.6: CRCConference's result of voting class names page*

After users finish creating and voting class names, now the system will show the list of all class names as shown in *Figure A.7*. Then each user will choose class name which is needed for each responsibility.

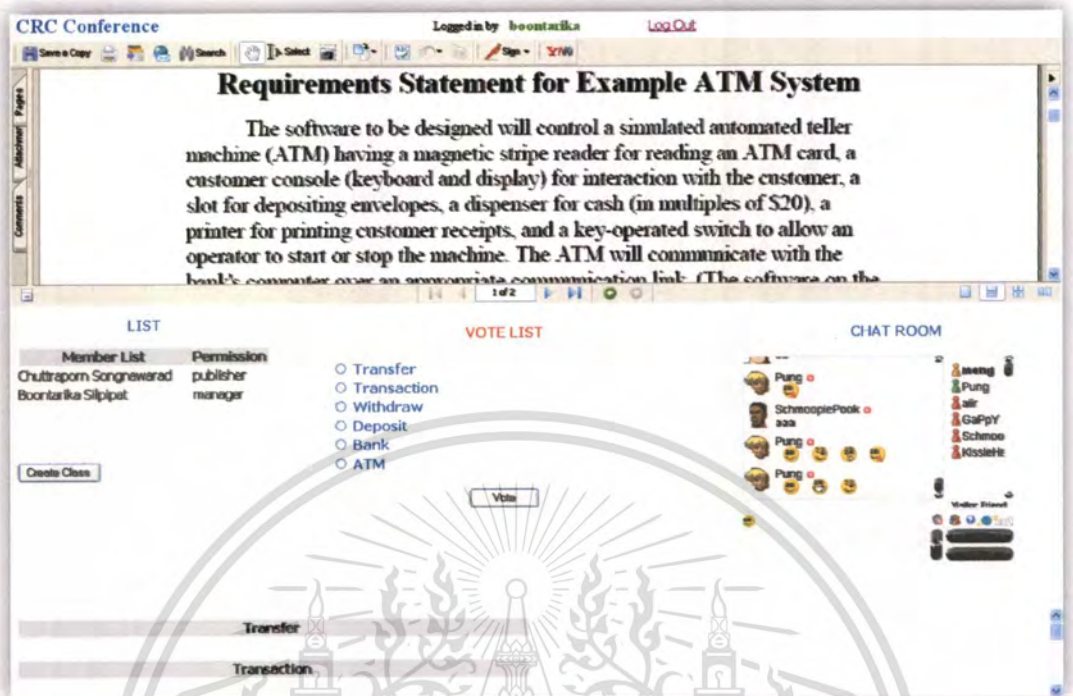


Figure A.7: CRCConference's voting class name of responsibility page

When all users operated suggestion, in the center of next page will be shown suggestion result. In the bottom of suggestion result, we will have box for each user to add name of responsibility which they want for requirement as shown in Figure A.8.

**Requirements Statement for Example ATM System**

The software to be designed will control a simulated automated teller machine (ATM) having a magnetic stripe reader for reading an ATM card. a

LIST	Withdraw	0 %
Member List	Deposit	0 %
Chutraporn Songwarat	Bank	50 %
Boontarika Sāpāt	ATM	50 %

**Responsibility**  **System**

Start up system operation

Transfer

Transaction

**CHAT ROOM**

Pung  
meng  
air  
GAP  
Schm  
Kisse

The ATM will service one customer at a time. A customer will be required to insert an ATM card and enter a personal identification number (PIN)

*Figure A.8: CRCConference's creating response name page*

In case that have equally result, chairman must decide that responsibility should be in which class name as shown in *Figure A.9*.

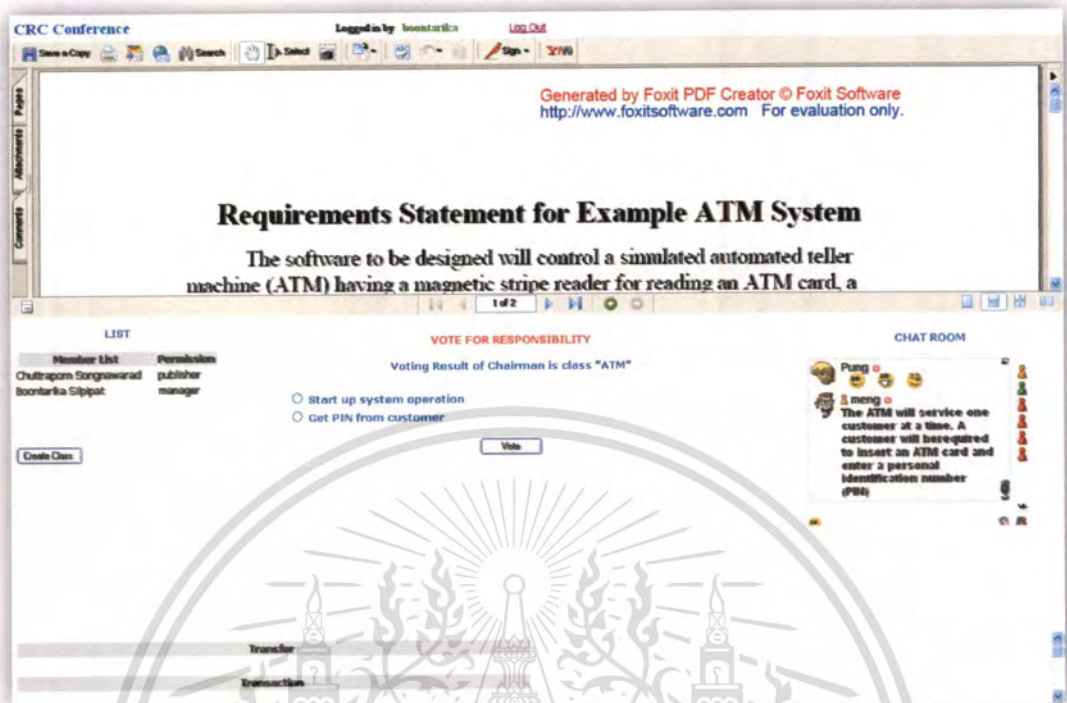
The screenshot shows a web browser window titled "CRC Conference". The page content includes:

- Header: "CRC Conference", "Logged in by boosurika", "Log Out", and a watermark "Generated by Foxit PDF Creator © Foxit Software http://www.foxitsoftware.com For evaluation only."
- Section: "Requirements Statement for Example ATM System".
- Text: "The software to be designed will control a simulated automated teller machine (ATM) having a magnetic stripe reader for reading an ATM card a..."
- Table:
 

LIST	Transfer	0 %
Transaction	0 %	
Withdraw	0 %	
Deposit	0 %	
Bank	50 %	
ATM	50 %	
- Form: "Responsibility" section with a "submit" button and a text input field containing "Get PIN from customer".
- Section: "CHAIRMAN DECIDING" with radio buttons for "Bank" and "ATM", and a "Confirm" button.
- Section: "Transaction" with a "Transfer" button.
- Chat Room: "CHAT ROOM" with a message: "The ATM will service one customer at a time. A customer will be required to insert an ATM card and enter a personal identification number (PIN)".

*Figure A.9: CRCConference's creating response name page for chairman*

The next page will be shown the deciding of chairman and responsibility that was added by each user from the last page for all users in the room can vote as shown in *Figure A.10*.



*Figure A.10: CRCConference's voting response name page*

When all users complete voting for responsibility, the result will show on the center of the next page as shown in *Figure A.11*. If any responsibility has the highest score, the system will store that responsibility into the database and suggest the class for next responsibility.

CRC Conference Logged in by boontarika [Log Out](#)

The ATM will service one customer at a time. A customer will be required to insert an ATM card and enter a personal identification number (PIN) - both of which will be sent to the bank for validation as part of each transaction. The customer will then be able to perform one or more transactions. The card will be retained in the machine until the customer indicates that he/she desires no further transactions, at which point it will be returned - except as noted below.

The ATM will be able to provide the following services to the customer:

---

**LIST**

Member List	Permission
Chuttraporn Songnaward	publisher
Boontarika Silpipat	manager

[Return to VoteClass](#)

[Create Class](#)

**VOTE RESPONSIBILITY RESULT**

Start up system operation	0 %
Get PIN from customer	100 %

[Return to VoteClass](#)

**CHAT ROOM**

Pung  
  
 meng  
 Pung  
 air  
 GaPy  
 Schmao  
 Kasihe

The ATM will service one customer at a time. A customer will be required to insert an ATM card and enter a personal identification number (PIN).

Transfer  
Transaction

Figure A.11: CRCConference's result of voting response name page

## Appendix B

### Database System

In Chapter 3, we've mentioned about database in this system and we explained some of tables in detailed already. In this section, we'll continue describe detail of the rest table in our database. However, please note that, the table that has been explained already will not be included in this section.

Classcreate table stores class name created from user's input

Table B.1 - Classcreate Table

No.	Field Name	Data type	Description
1	Ccid	integer	class created id
2	rid	integer	requirement id
3	classn	varchar	class name created

Responcreate table stores response name created from user's input

Table B.2 - Responcreate Table

No.	Field Name	Data type	Description
1	crid	integer	response name created id
2	rid	integer	requirement id
3	responname	text	response name created

## Appendix C

### Installation

In order to run our web application, there is some software that needs to be installed first. These software include Apache, PHP, MySQL and phpMyAdmin. We'll show you how to install these software on a computer.

#### C.1 Installing Apache, PHP, MySQL and phpMyAdmin

To install Apache, PHP, MySQL and phpMyAdmin, we'll use AppServ, to manage the installation for us. During the process, you'll be prompted to enter your MySQL and phpMyAdmin root password. You can set the password to anything you want, but please make it secure and note it down, as you'll need to use it again during the installation process.

#### C.2 Setup the database

First, you need to create database in phpMyAdmin (<http://localhost/phpMyAdmin/index.php>) by using name "crc" as shown in *Figure C.1*.

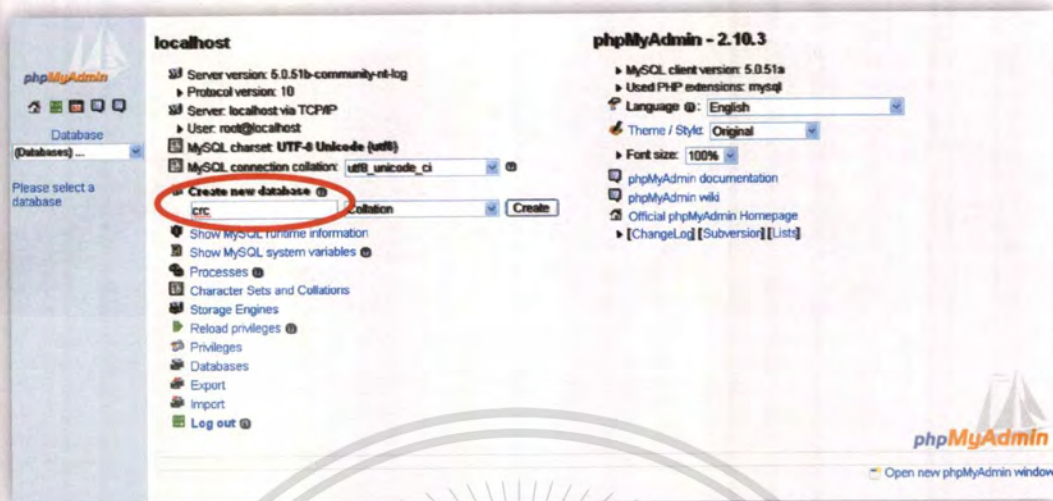


Figure C.1: Create new database

We have script file of “crc” database that is “crc.sql” for you to import this file by click “Import” on top menu. In the Import page, you click “Choose File” and browse our script file as shown in Figure C.2.

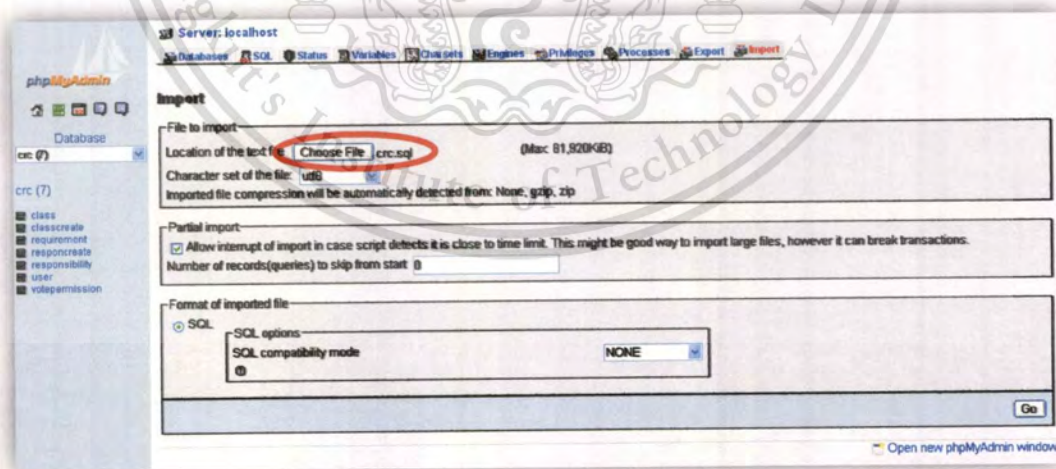
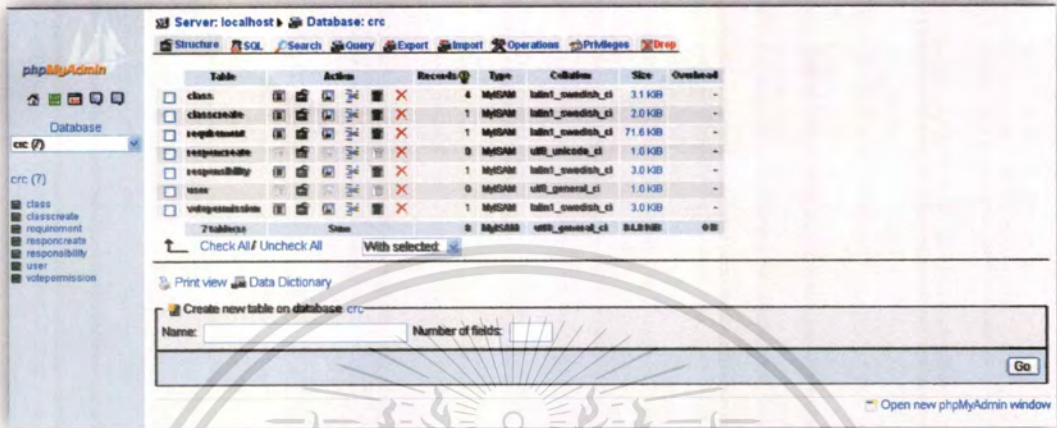


Figure C.2: Import “crc.sql” file

Finally, you click Go and all tables will be created to “crc” database automatically as shown in *Figure C.3*.



*Figure C.3: All tables in “crc” database*

### C.3 Setup project folder

We have “CRC.rar” file for you and you have to unzip project file to C:\AppServ\www as shown in *Figure C.4*.

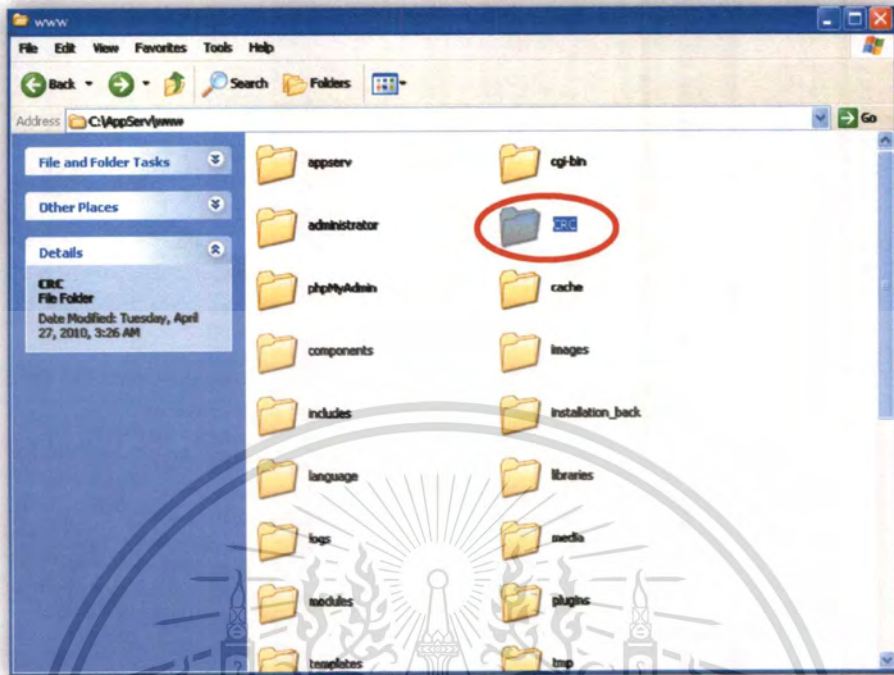


Figure C.4: C:\AppServ\www

Now you can run our web application on localhost by using <http://localhost/CRC/LoginFrame.php> as shown in Figure C.5.

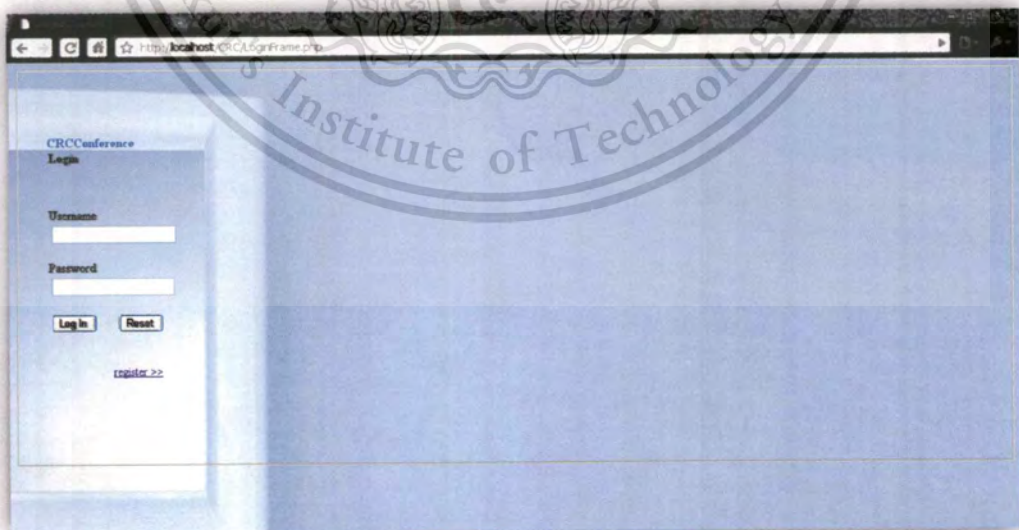


Figure C.5: <http://localhost/CRC/LoginFrame.php>