VAASAN AMMATTIKORKEAKOULU UNIVERSITY OF

APPLIED SCIENCES

CAO Kai

ASPIRE MODEL ONLINE DEVELOPMENT SOLUTION

Information Technology
2009

VAASAN AMMATTIKORKEAKOULU UNIVERSITY OF APPLIED SCIENCES
Degree Program of Information Technology

ABSTRACT

Author	Cao Kai
Topic	Aspire Model Online Development Solution
Year	2009
Language	English
Pages	64+2 Appendix
Name of supervisor	Timo Kankaanpää

The purpose of this thesis is to find out the solution for web development of Aspire EU project. It can let user to create, modify or delete forms, and create, modify or delete fields for the form, so some users can take full control of the software solution after authentication. The solution also provides user control, reporting, and login functions. Different types of users will have different rights of using this software.
Technically it is a software solution based on J2EE technology and Tiles2 Struts2, Spring and Hibernate Frameworks.

Keywords Aspire EU, J2EE, Struts2, Tiles2, Spring, Hibernate

CONTENTS

ABSTRACT
1. INTRODUCTION 4
   1.1. Summary 4
   1.2. Users and how often to use the solution 4
   1.3. Existing System 4
   1.4. Terminology 4
2. GENERAL DESCRIPTION 5
3. DATA AND DATABASE 7
   3.1. Aspire Model Solution ER database design 7
   3.2. Detail of tables 7
4. FUNCTIONS 14
   4.1. List of all functions 14
   4.2. List of functions' detail 15
5. DESIGN AND STRUCTURE 56
   5.1. Static Version 56
   5.2. Dynamic Version 56
6. OTHER PROPERTIES 61
   6.1. Technical 61
   6.2. Deploy solution 61
   6.3. Scheduling 61
   6.4. Platforms 61
   6.5. Portability 61
   6.6. Expandability 62
   6.7. Support & Maintenance 62
   6.8. Configuration Management 62
7. CONCLUSION 63
REFERENCES 64
APPENDIX 65

INTRODUCTION

Aspire-Model Solution is to support EU Aspire Project; this software application is made for Vaasa Energy Institute to make an online development solution for ASPIRE SEC Model.

1.1. Summary

Aspire-Model solution is to support EU Aspire Project. It is based on 4 phases, SEC Planning and Creation Phase, Capacity Building Phase, SEAP Phase, Evaluation Phase, and it can enable partners to review the initial impacts within each Sustainable Energy Community of the ASPIRE approach and to evaluate both the initial and future longer-term impacts.

1.2. Users and how often to use the solution

Users are from Aspire Project groups, local action plan executor and other people who want to see the project and give suggestion. The software solution can be used for monthly or weekly.

1.3. Existing System

There is an excel document for the Aspire SEC Model, all the functions or user interfaces are designed based on this excel, paper work, and experts' experiences.

(ASPIRE SEC Model 20081202.excel & ASPIRE Model.ppt)

1.4. Terminology

SEC is for Sustainable Energy Community

SEAP is for Sustainable Energy Action Plan

2. GENERAL DESCRIPTION

There is a need for a software solution of creating web application for Aspire EU. As a web application the software will be running in a server like tomcat or JBoss and can be used by Firefox at least.

Figure 2.1 shows the workflow of this software.

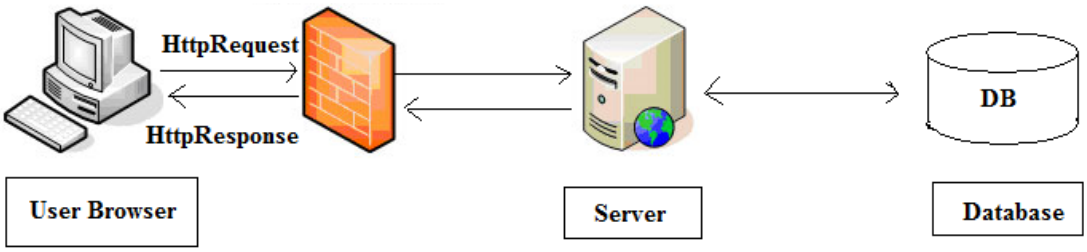


Figure 2.1

The development core should include development model solution to supply functions to control Aspire SEC Development Model: SEC Planning and Definition Phase, Capacity Building Phase, SEAP Phase, Evaluation Phase.

There is also a requirement for reporting, which is based on specified template and input information from development model solution, automatically generating reports for the working group or publish to the public.

And also there is a login solution. There are 4 different types of characters in this system. No.1 is administrator who has the right to manage SEC and create super member account. No.2 is super member who is the man taking control of SEC, and they are capable of confirming working group member to join the system, to use the development solution. No.3 is the member, they can be a working group member link to one SEC, or a normal member link to an empty SEC. Working group member, who can monitor the process of the development solution, but does not have rights to modify or adds new values to any form. The registered normal member, they can see the road map of development solution, and they are linked to an empty SEC.

Finally the guest, they can register to be a normal member.

Figure 2.2 shows the use case for the Aspire Model Solution

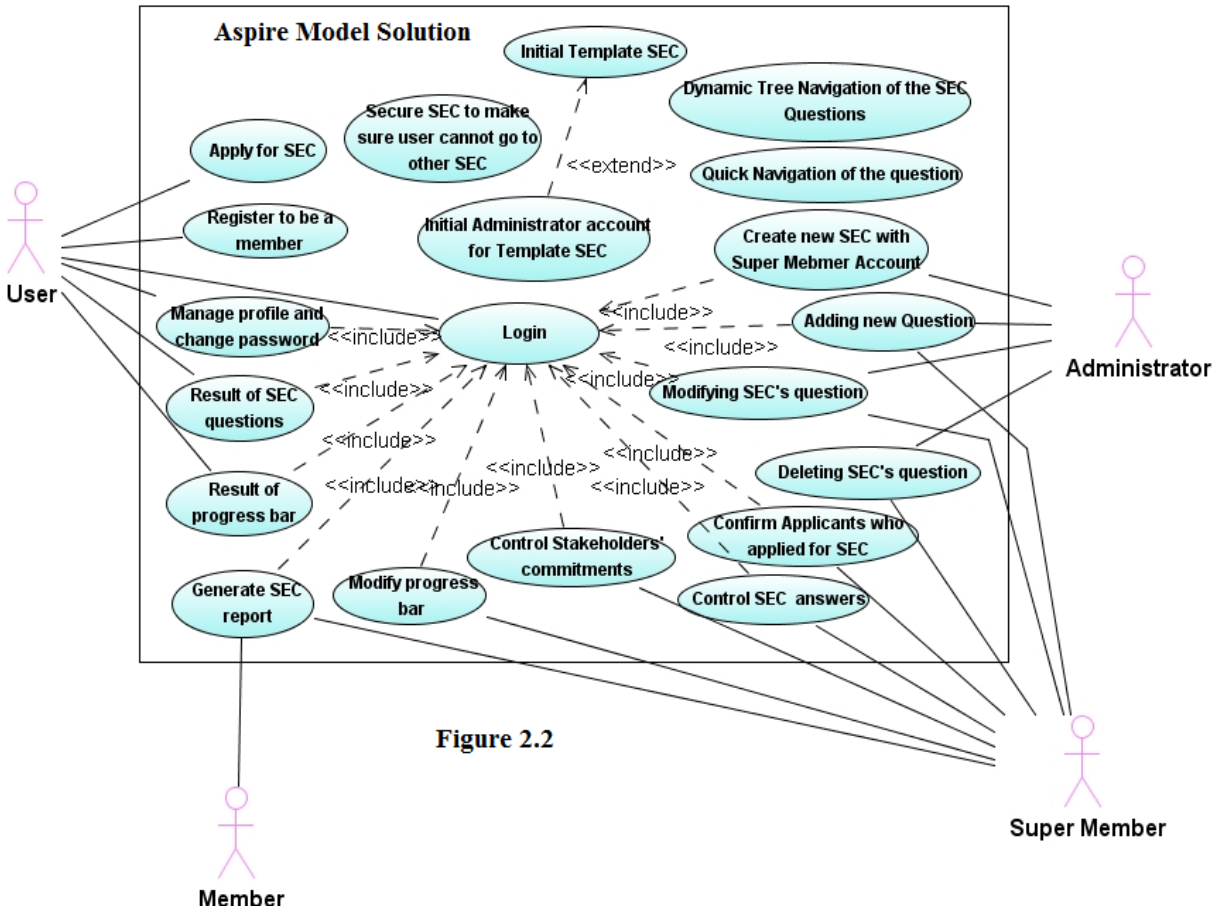
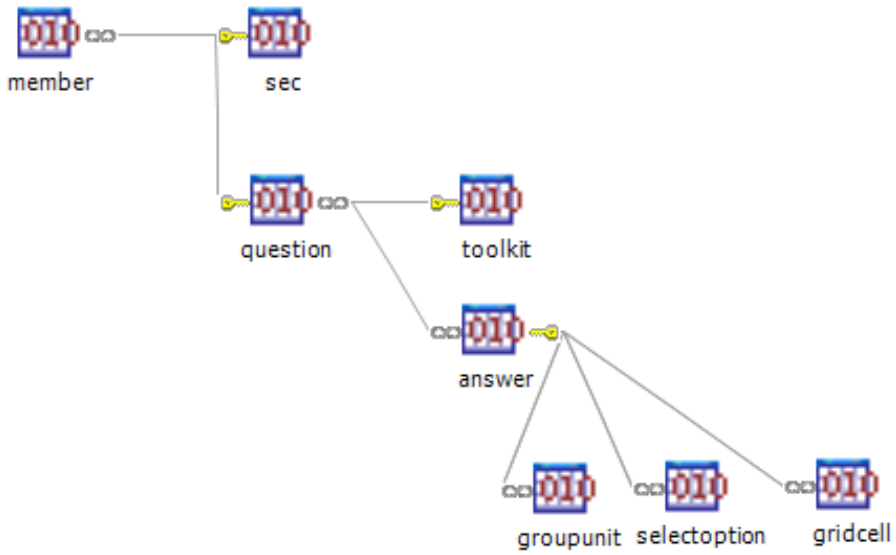


Figure 2.2

3. DATA AND DATABASE

3.1. Aspire Model Solution ER database design



It is designed to support the Aspire web project acts as online development kit.

3.2. Detail of tables

Question table and Answer table are created to support the excel version of the Aspire SEC Model, and for each question there are several different types of answers, by defining the answer's name, type and sequence, and link to the question, when user wants to see the question's detail information, the web user interface will create based on these information. To show the answer name and value, for different type there is different GUI component, and they are order by the sequence.

Member Table

Name	Datatype	Not Null	Comment
id	int(11)	true	Primary key for Member table
username	varchar(50)	true	Username for the registered member, super member or admin

password	varchar(50)	true	Password for the registered member, super member or admin
firstName	varchar(50)		
middleName	varchar(50)		
lastName	varchar(50)		
career	varchar(255)		
age	int(11)		
gender	varchar(10)		
email	varchar(100)		
phone	varchar(30)		
country	varchar(50)		
region	varchar(50)		
city	varchar(50)		
street	varchar(100)		
doorNumber	int(11)		
postcode	varchar(20)		
role	varchar(255)		Store the role of the registered user can be “member”, “superMember” or “admin”
fk_sec_member	int(11)		Foreign key to store id of the SEC, and show which SEC the member belongs to
fk_sec_appliedMember	int(11)		Foreign key to store id of the SEC, and show which SEC the member applied for before the Super Member of such SEC confirmed his/her applicant

SEC Table

Name	**Datatype**	**Not Null**	**Comment**
id	int(11)	true	Primary key for the SEC Table
Foundation_Date	varchar(255)	true	
country	varchar(50)		
region	varchar(50)		
city	varchar(50)		
street	varchar(100)		
doorNumber	int(11)		
postcode	varchar(20)		

Question Table

Name	Datatype	Not Null	Comment
id	int(11)	true	Primary key for the Question Table
question	varchar(255)		Question topic of the ASPIRE SEC Development Model
phase	varchar(255)		Phase information of Development Model e.g. “SEC Planning and Creation Phase”, “Capacity Building Phase”, “SEAP Phase”, “Evaluation Phase”
Key_Success_Factors	varchar(255)		
Strategic_Objectives	varchar(255)		
Operational_Activities	varchar(255)		
type	varchar(255)		Type of Question, will be used to show different type of web interfaces, can be “menu”(Mostly the parent Question, interface will be used to show a list of child questions also see Appendix 1), “group”(Shows the group answer also see Appendix 1), “complex”(Question which have modifiable answers like “textfield”, “textarea”… also see Appendix 1)
,k_toolkit_id	int(11)		Foreign Key to store id of the Toolkit, for each Question has one-to-one relationship with Toolkit
fk_sec_question	int(11)		Foreign Key to store id of the SEC, for each SEC can have many Questions, they have one-to-many relationship
fk_parentQuestion_childQuestion	int(11)		Foreign Key to store id of the parent Question, for each question there is a parent question except

			the root question, so this link makes the questions like a tree model. For each Parent Question can have 0 to many Child Questions, so it is a one-to-many relationship

Toolkit

Name	Datatype	Not Null	Comment
id	int(11)	true	Primary key for Toolkit table
criteria	varchar(255)		Criteria for progress bar
unit	varchar(255)		Unit for progress bar
grade	int(11)		Grade for progress
owner	varchar(255)		Owner of progress bar
targetDate	varchar(255)		Target Date of progress bar

Answer Table

Name	Datatype	Not Null	Comment
id	int(11)	true	Primary key for Answer table
answer_type	varchar(255)		Type of answer, can be: “boldLabel”: shows text with bold font style and big font size “smallFont”: shows text with small font size “textfield”: Text field for input type text “textarea”: Text area for input “datetimepicker”: “Dojo” Date time picker component for Date selecting “select”: Form select, can have a list of options “number”: One text field for number one select for unit “file”: File upload for form “gird”: Grid complex type, which looks like a grid table and for each cell has one specified answer also

			see Appendix 1 “group”: Group complex type, which will use one page to show list of group units and part of their information, super member can have functions to add new group unit, see detail page and change the current group unit or delete current group unit. also see Appendix 1
answer_sequence	int(11)		The sequence of the answer for the parent question
boldLabel_value	varchar(255)		
smallFont_value	varchar(255)		
textfield_value	varchar(255)		
textarea_value	longtext		
number_value	double		
number_unit	varchar(255)		
datetimepicker_value	varchar(255)		
select_value	varchar(255)		
file_path	varchar(255)		
grid_maxX	int(11)		If answer type is ”grid”, this column will store the Maximum X value of the grid table start from 0
grid_maxY	int(11)		If answer type is ”grid”, this column will store the Maximum Y value of the grid table start from 0
group_main	varchar(255)		If answer type is ”group”, this column will store the Boolean information “true” or “false”, if true the answer will showed on the first page for group question which has a list of exist group units.
fk_groupAnswer_id	int(11)		Foreign key for GroupUnit table, if answer type is ”group”, so the detail answers for the group answer will store the id of the GroupUnit means they belongs to the same groupUnit and one group answer can have 0 to many groupUnits. The relationship between GroupUnit and the group detail answer is one-to-many

fk_question_answer	int(11)		Foreign key for Question table, each Question can have 0 to many answers, the relationship between Question and Answer is one-to-many

SelectOption Table

Name	**Datatype**	**Not Null**	**Comment**
id	int(11)	true	Primary key for SelectOption table
selectOption_key	varchar(255)		Key of the Select Option (Showed on the webpage)
selectOption_value	varchar(255)		Value of the Select Option(Hidden from the webpage but will be stored in the Answer table select_value if been selected)
fk_numberAnswer_selectOption	int(11)		Foreign key for Answer table when the answer type is ”number”, means this selectOption belongs certain number answer’s unit option list. Relationship between Answer and SelectOption is one-to-many
fk_selectAnswer_selectOption			Foreign key for Answer table when the answer type is ”select”, means this selectOption belongs certain select answer’s option list. Relationship between Answer and SelectOption is one-to-many

GridCell Table

Name	Datatype	Not Null	Comment
id	int(11)	true	Primary key of GridCell table
axis_x	int(11)	true	X coordinate of the grid table
axis_y	int(11)	true	Y coordinate of the grid table
fk_gridAnswer_id	int(11)		Foreign key for Answer table, it stores the id of the certain cell's answer which can be "smallFont","textfield","textarea","select","number","datetimepicker","file", relationship between GridCell and Answer is one-to-one
fk_gridAnswer_gridCell	int(11)		Foreign key for Answer table when the answer type is "grid", and it stores the id for the grid answer, which can have 0 to many grid cells, the relationship between Answer and GridCell is one-to-many

GroupUnit Table

Name	Datatype	Not Null	Comment
id	int(11)	true	Primary key for GroupUnit table
fk_groupAnswer_groupUnit	int(11)		Foreign key for Answer table when the answer type is "group", it stores the id of the group answer, the group answer can have 0 to many groupUnits, relationship between Answer and GroupUnit is one-to-many

4. FUNCTIONS

4.1. List of all functions

Reference	Description	Priories
F1	Initial template SEC	2
F2	Initial Administrator account	2
F3	Administrator and super member can control the SEC by adding new question	1
F4	Administrator and super member can control SEC by modifying current question	1
F5	Administrator and super member control SEC by deleting current question	1
F6	Administrator can create new SEC with super member account	2
F7	Super member can modify the answers for the SEC questions, user can see the result of SEC questions after login	1
F8	Super member can modify the progress bar for the SEC questions, and user can see the result of progress bar after login	1
F9	Super member and member can generate his/her SEC report	2
F10	Member can apply for one SEC	2
F11	Super member confirm applicant to be a member	2
F12	User can register to be a member	2
F13	Dynamic Tree Navigation of the SEC Questions	2
F14	Quick Navigation of the question	2
F15	User login and logout	2
F16	User can manage profile and change password	2
F17	Secure SEC to make sure user cannot go to other SEC	3

4.2. List of functions' detail

F1 Initial template SEC

Figure 4.2.1 shows the sequence diagram for initial template SEC

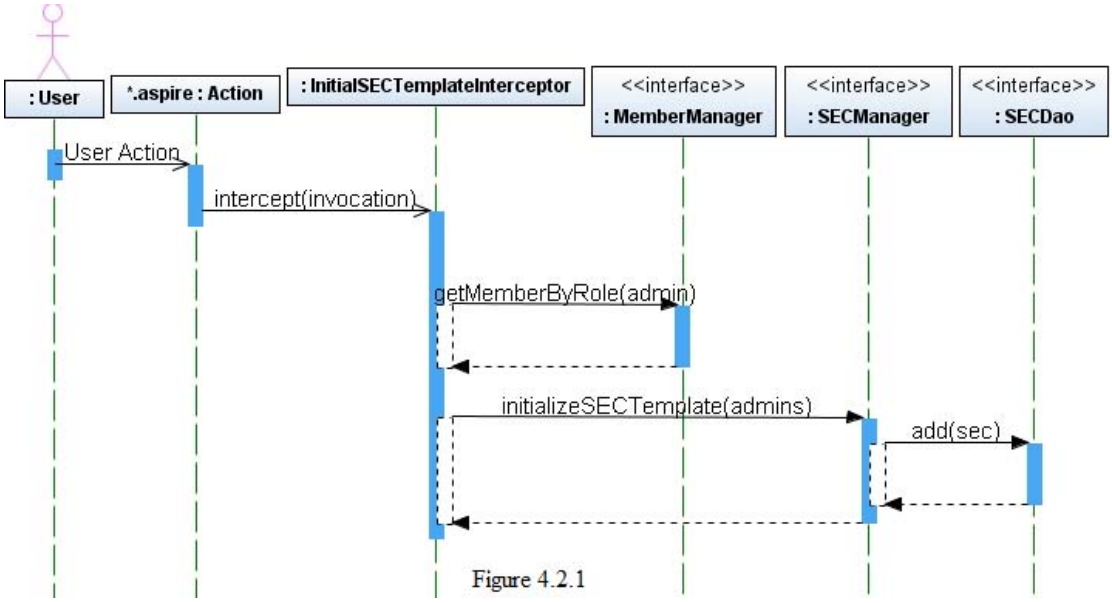


Figure 4.2.1

Preconditions	1. System is started at its first time running 2. Database has aspire schema 3. The aspire database is empty
Input	1. User request link like *.aspire by click on button, link or redirect from browser
Description	This function makes sure there will be a template SEC for the administrator, and the SEC's questions structure is based on the excel sheet for ASPIRE SEC Development Model. When user request the link like *.aspire Struts2 framework will know user is requesting resource from its action classes, so at the beginning Struts2 will call interceptors intercept function, there is one interceptor for checking template SEC existence, it will check whether there is SEC with name "Template", if there is no such SEC all the administrators account will be retrieved from Member database table, and then the process to create template SEC starts, and inserting hardcoded questions of template SEC into Question database table. After insert all the questions, the administrators' accounts retrieved earlier will link to the template SEC. Template SEC will have a root question with topic "Aspire-Model" and 4 different phases "SEC Planning and Creation Phase", "Capacity Building Phase", "SEAP Phase" and "Evaluation Phase", they are from the excel sheet for ASPIRE SEC Development Model.

Exception	1. Database not available 2. Hibernate context initialize fail 3. Spring context initialize fail 4. Struts2 context initialize fail
Result	Template SEC created in the database and administrator can access to it.

F2 Initial Administrator account

Figure 4.2.2 shows the sequence diagram for initial administrator account when there is an empty database

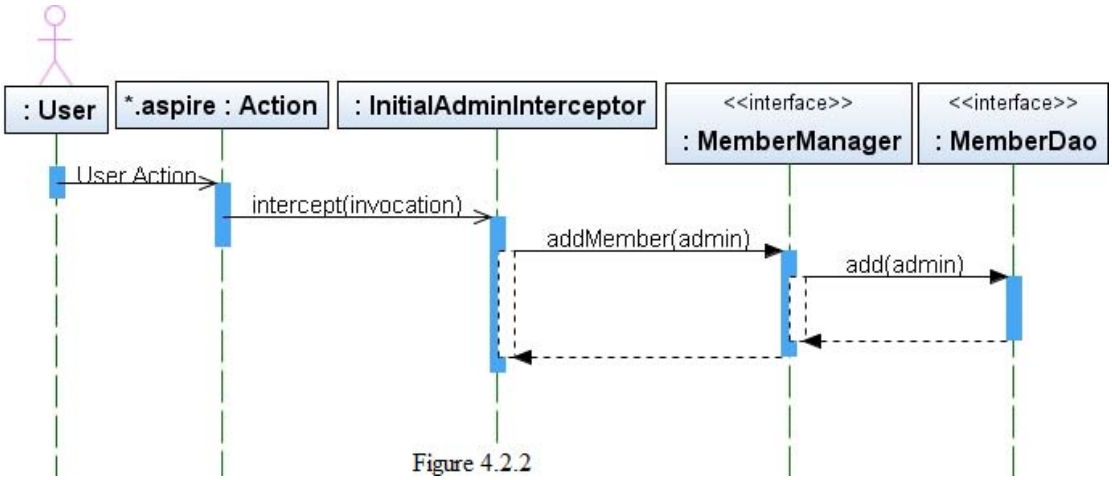


Figure 4.2.2

Preconditions	1. System is started at its first time running 2. Database has aspire schema 3. The aspire database has no administrator account
Input	1. User request link *.aspire by clicking button link, or redirect from browser
Description	This function makes sure there will be an administrator account created when the database is empty. When user request link like *.aspire Struts2 framework will know user is requesting resource from its action classes, so at the beginning Struts2 will run interceptors which there is a interceptor to create administrator account when the database is empty. It will check whether there is any administrator member account with username “admin” and role “admin” if there is no such member account, new member with username “admin”; password “123”; role “admin” will be created and then insert the administrator account into database.
Exception	1. Database not available 2. Hibernate context initialize fail 3. Spring context initialize fail

	4. Struts2 context initialize fail
Result	Administrator account with username ”admin”, password “123”, role “admin” will be created in the database

F3 Administrator and super member can control SEC by adding new question

Figure 4.2.3 shows the sequence diagram for controlling the SEC by adding new question

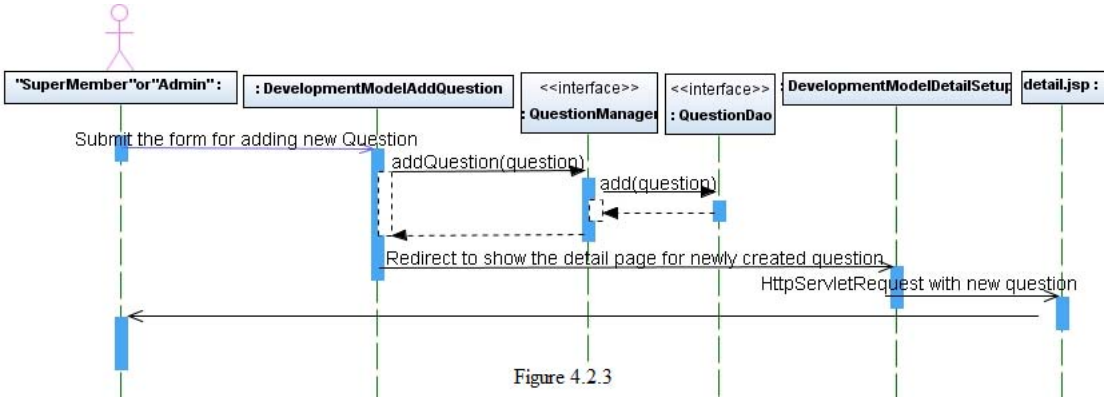


Figure 4.2.3

Preconditions	1. System is not running at its first time, database has template SEC and administrator account, or the super member’s SEC and account 2. User login as a administrator or super member
Input	1. Administrator click on the link ***add new sub question*** or redirect to link ***dmAddQuestionSetup.aspire*** 2. In the dmAddQuestion.jsp form to input Question name, Question type 3. In the dmAddQuestion.jsp form to input Answer name, Answer type, maximum 9 answers
Description	From the Question detail page only the super member or administrator can see the link ***add new sub question***, when he/she click on this link, Struts2 framework will notice and call the execute() function to prepare new page “dmAddQuestion.jsp” for adding new sub question. In the “dmAddQuestion.jsp” page there is a form with text fields asking for question name and type, and 9 rows for inputting answer name and type. Question type can be “menu” “group” “complex”, Answer type can be “boldLabel” “smallFont” “textfield” “textarea” “datetimepicker” “number” “file”. After fill in the form and submit to the server, Struts2 will execute function to take care of creating the new Question based on the

	submitted form information, and creating answers based on the submitted form information, if any answer name is empty that answer will not be created. New question and answers will be inserted into database. The new question will have link to its parent question which is the one where ***add new sub question*** link be clicked. After insert the new question successfully the request will be redirect to show the detail page for the newly created question, which will dynamically generate the detail view for the newly created question and show to the super member or administrator. The left side navigation tree and top quick navigation will be created based on the newly created question. Later all the member, super member, administrator who has link with the SEC can see the new question.
Exception	1. Database not available 2. Hibernate context initialize fail 3. Spring context initialize fail 4. Struts2 context initialize fail 5. No parent question id in the HttpServletRequest when the action dmAddQuestionSetup be executed
Result	New Question with its new answers will be created in the database, and it connects to its parent Question to create the navigation.

Figure 4.2.4 shows link where to add new sub question

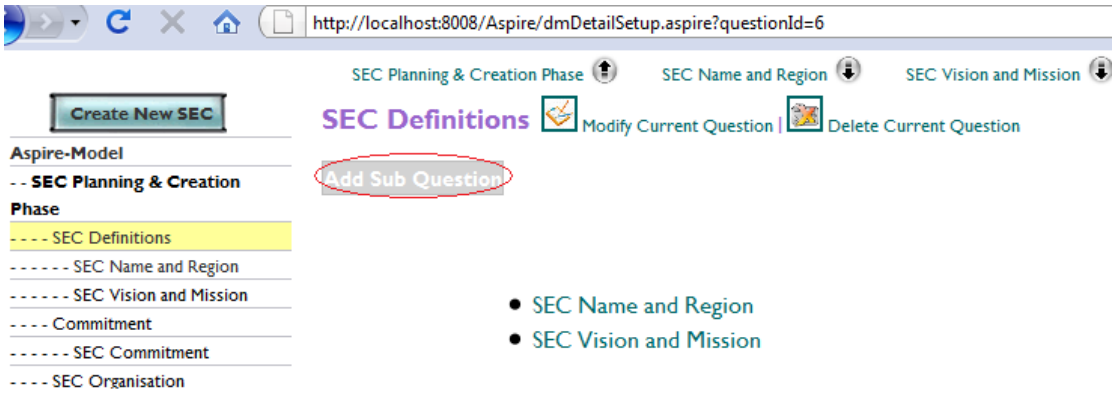


Figure 4.2.4

Figure 4.2.5 shows the form for adding new question after click on ***Add Sub Question***:

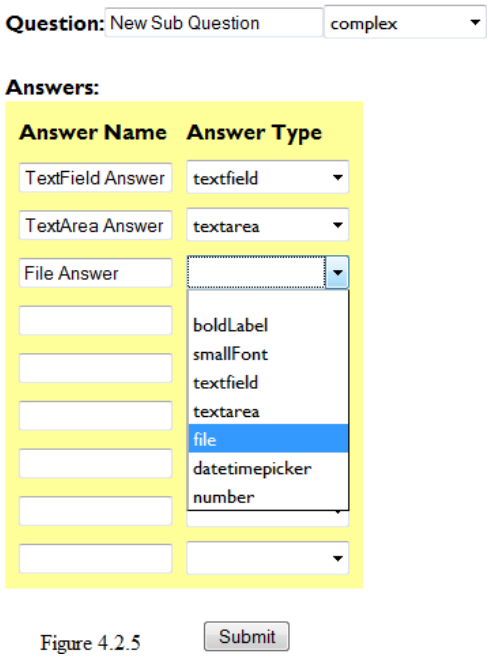


Figure 4.2.5

Figure 4.2.6 shows after submit the form for the new question the detail page for the new question will show. The left side navigation and top quick navigation changed also:

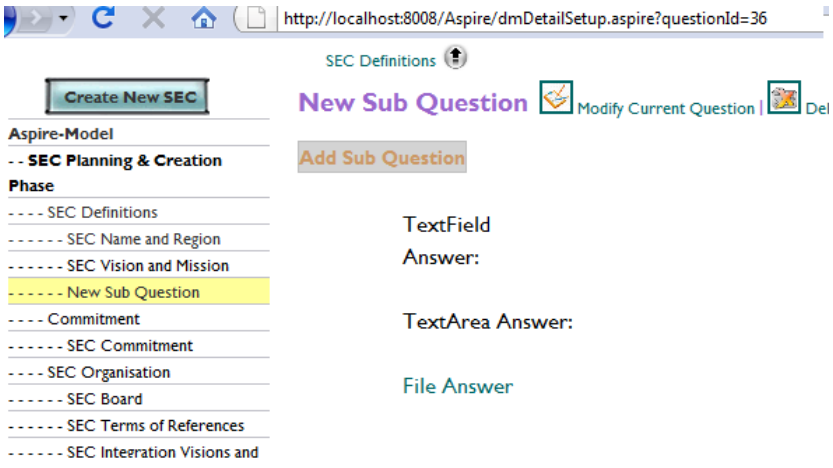


Figure 4.2.6

F4 Administrator and super member can control the SEC by modifying question

Figure 4.2.7 shows the sequence diagram for controlling SEC by modifying question

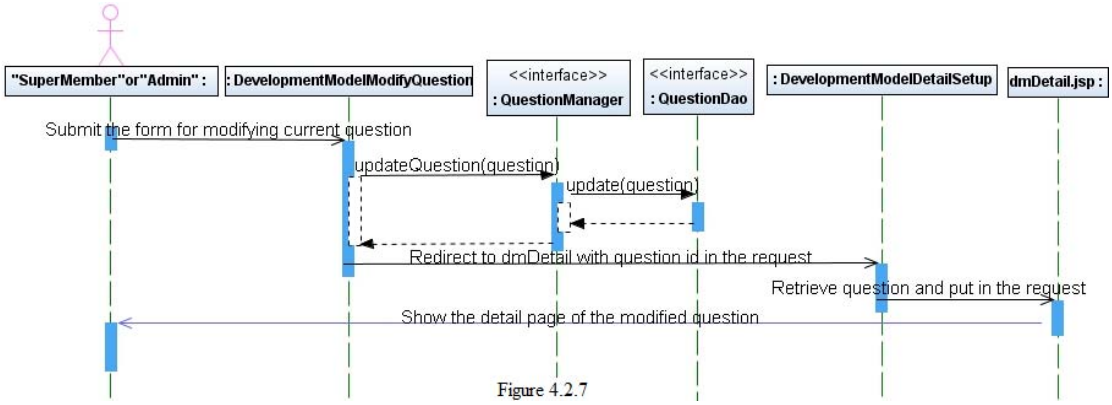


Figure 4.2.7

Preconditions	1. System is not running at its first time, database has template SEC and administrator account or the super member's SEC and account 2. User login as a administrator or super member
Input	1. Administrator click on the link ***modify current question*** or redirect to link ***dmModifyQuestionSetup.aspire*** 2. In the dmModfiyQuestion.jsp form to modify Question name, Question type 3. In the dmModifyQuestion.jsp form to Modify exist Answer name, Answer type 4. In the dmModifyQuestion.jsp form is possible to add maximum 3 more Answers with answer name and answer type
Description	From the question's detail page only the super member or administrator can see the link ***modify current question***, when he/she click on this link, Struts2 framework will notice and preparing the page "dmModifyQuestion.jsp" to show form for modifying the question. In the "dmModifyQuestion.jsp" page there is a form with several text fields, with default value of from the question detail page, it asks for question name and type, all the answers of the question and 3 more rows for inputting answer name and type. Question type can be "menu" "group" "complex", answer type can be "boldLabel" "smallFont" "textfield" "textarea" "datetimepicker" "number" and "file". After filling in the form and submitting to the server, Struts2 will execute to take care of updating question based on the submitted form information, and updating answers based on the submitted form information, also possible to create maximum 3 more answers, if any answer name in those 3 new answers is empty that answer will not be created. "QuestionManager" and "QuestionDao" will be used to update the

	question, “AnswerManager” and “AnswerDao” will update or insert answers into database. After successfully updating the question, user will be redirect to the “detail.jsp” which will dynamically generate the detail view for the modified question. The left side navigation tree and top quick navigation will be updated based on the updated question. Later all the member, super member, administrator who has link with the SEC can see the modified question.
Exception	1. Database not available 2. Hibernate context initialize fail 3. Spring context initialize fail 4. Struts2 context initialize fail 5. No question id in the HttpServletRequest when the action “dmModifyQuestionSetup” be executed
Result	Question with its new answers will be updated in the database, and it connects to its parent Question to create the navigation.

Figure 4.2.8 shows where to modify current question:

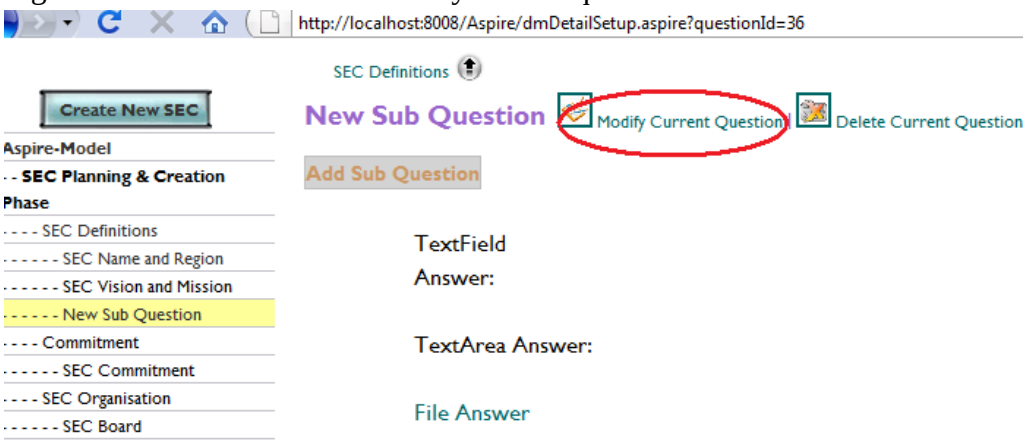


Figure 4.2.8

Figure 4.2.9 shows the “dmModifyQuestion.jsp” page with question information by clicking on ***Modify Current Question***:

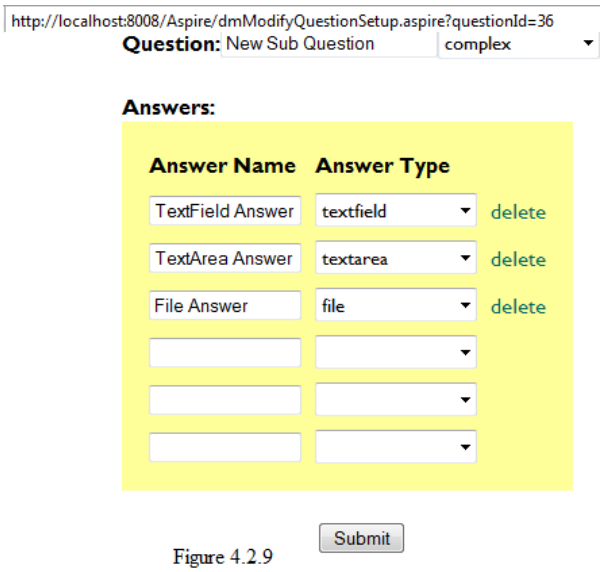


Figure 4.2.9

Figure 4.2.10 shows the changed form of question information before submit:

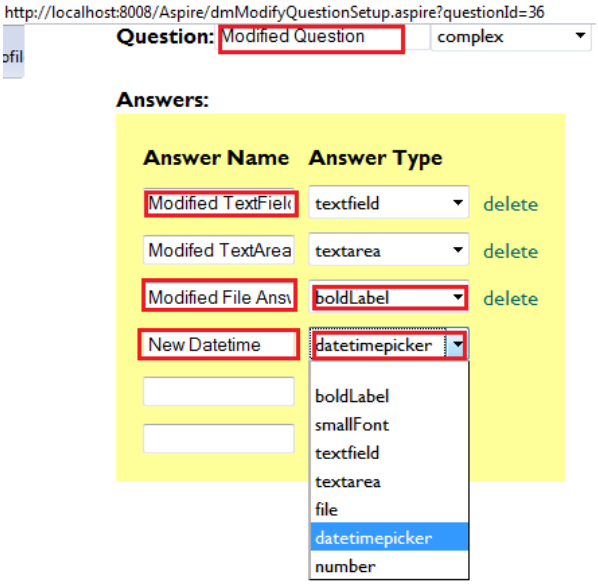


Figure 4.2.10

Figure 4.2.11 shows the detail of the question after submit the form:

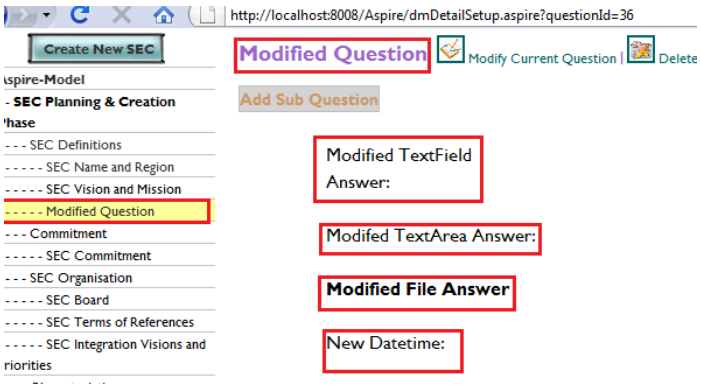


Figure 4.2.11

F5 Administrator and super member control SEC by deleting current question

Figure 4.2.12 shows the sequence diagram for controlling SEC by deleting question

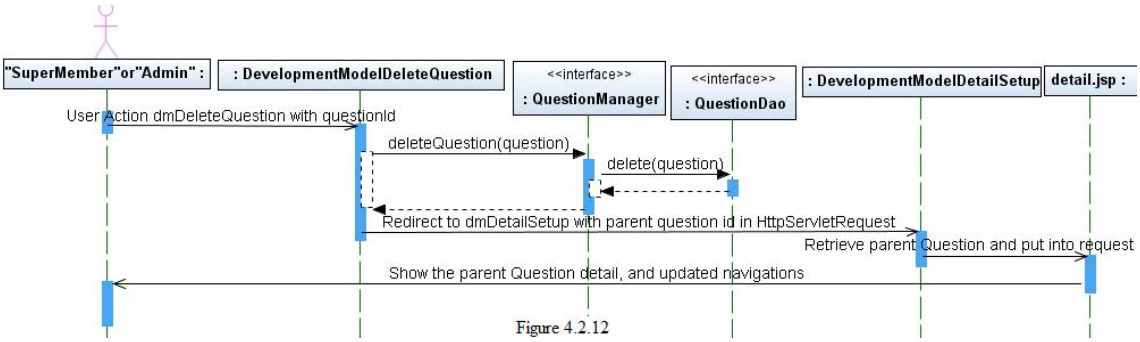


Figure 4.2.12

Preconditions	1. System is not running at its first time, database has template SEC and administrator account or the super member's SEC and account 2. User login as a administrator or super member
Input	1. Administrator click on the link ***Delete Current Question*** or redirect to link "***dmDeleteQuestion.aspire"***
Description	From the question's detail page only the super member or administrator can see the link ***Delete Current Question***, when he/she click on this link, Struts2 framework will notice and function to find and delete the question from the database with all the its answers, toolkit, all the links under this question will also be deleted from the database. After delete the question successfully user will be redirected to the page "detail.jsp" which will dynamically generate the detail view for the parent question and show to the super member or administrator. The left side navigation tree and top quick navigation will be updated also.
Exception	1. Database not available 2. Hibernate context initialize fail 3. Spring context initialize fail 4. Struts2 context initialize fail 5. No question id in the HttpServletRequest when the action "dmDeleteQuestion" be executed
Result	Question with the id which send to the action DevelopmentModelDeleteQuestion will be deleted from the database and all the answers, toolkit will be deleted from the database.

Figure 4.2.13 shows where to delete current question:

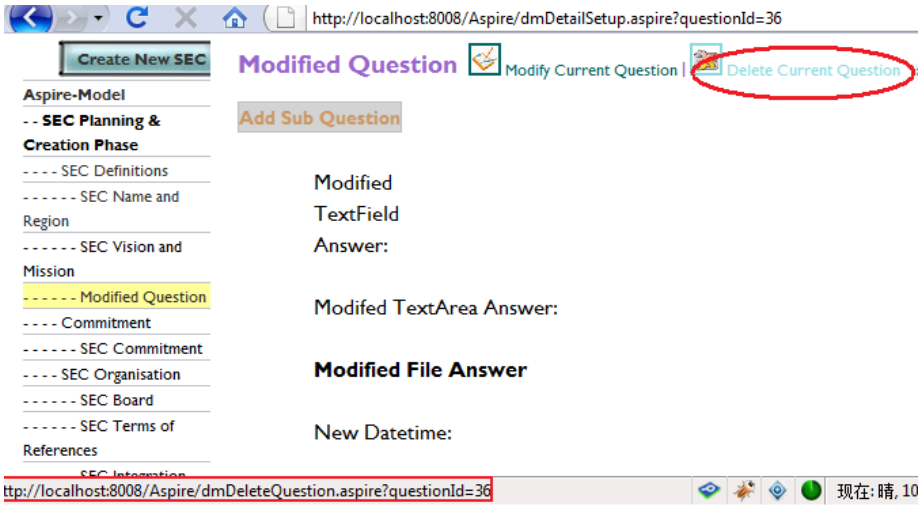


Figure 4.2.13

Figure 4.2.14 shows the parent question's detail after delete the question:

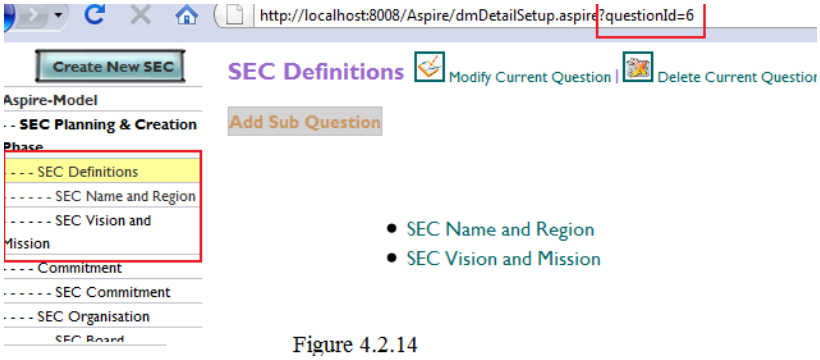


Figure 4.2.14

F6 Administrator can create new SEC with super member account

Figure 4.2.15 shows the sequence diagram for creating new SEC by administrator and create a super member account for the new SEC

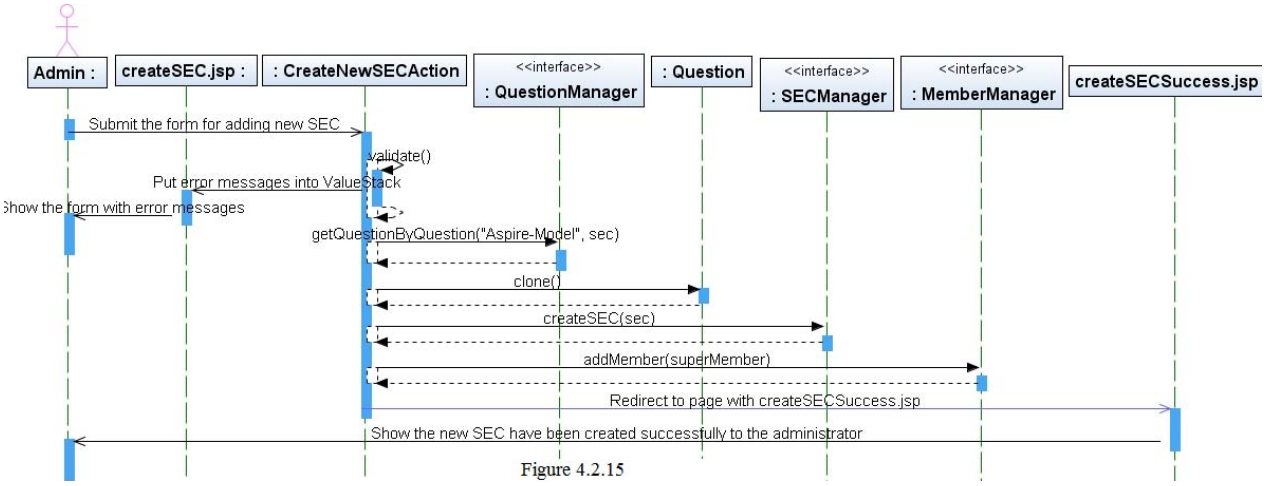


Figure 4.2.15

Preconditions	1. System is not running at its first time, database has template SEC and administrator account 2. User login as a administrator
Input	1. Administrator click on the ***Creaate New SEC*** button to open the createSECSetupAction which prepares createSEC.jsp page 2. Administrator must input SEC Name, Foundation Date 3. Administrator must input Super Member Account's Username and Password 4. Administrator can input SEC Address Country, Region, City, Street, Postcode
Description	This function is used for the Administrator only. There is require creating SEC, there will be lots of SECs in the future, but only the administrator has the right to create new SEC. The new SEC is cloned from the template SEC. To make it happen class like "Question" "Toolkit" "Answer" "SelectOption" "GridCell" "GroupUnit" have implemented "Cloneable" interface, by providing the clone() function which "Cloneable" interface is needed, all these classes will take care of the its shadow clone and deep clone, shadow clone is for fields like "int" "float"… and deep clone is for object which only stores their memory address in the linked objects, they are cloned by creating new object and do shadow clone recursively until all the objects have been cloned. By clicking on the button ***Create New SEC*** the page "createSEC.jsp" will be prepared which has a form for SEC Name, Foundation Date, SEC Super Member account, SEC Address. After filling in and confirming the form and send to the server if it's a unique SEC Name and have a Foundation Date, Super Member account has unique Username, password inputted. New SEC based on the SEC information and template SEC will be created. Template SEC will be retrieved from the database by getting member from the HttpSession which is the administrator account and by calling function to get the root question of Template SEC will be retrieved, because it's the root question when it tries to clone itself all the toolkit answers and its children questions will be cloned, so all the questions, answers, toolkits, select options, grid cells and group units will be cloned and linked with the new SEC. Also super member account with its username password and role of "superMember" will be created, and linked with new SEC. "SECManager" with "SECDao" will create new SEC and insert

	into database, “MemberManager” with “MemberDao” will create new super member account and link to the new SEC. Finally user will be redirected to the “createSECSuccess.jsp” page and show success information to the administrator
Exception	1. Database not available 2. Hibernate context initialize fail 3. Spring context initialize fail 4. Struts2 context initialize fail 5. SEC Name cannot be empty if not struts2 will redirect back to the create SEC page and show error message “SEC Name Must be input” 6. SEC Name has to be unique if not after submit the form struts2 will redirect back to the create SEC page and show error message “SEC Name exist, please choose a new one!” 7. Foundation Date must be chosen if its empty after submit struts2 will redirect back to the create SEC page and show error message “Foundation Date must be chosen” 8. Super Member name cannot be empty if its empty after submit struts2 will redirect back to the create SEC page and show error message “Super Member’s username must be input” 9. Super Member name must be unique if the same username exist in the database after submit struts2 will redirect back to the create SEC page and show error message “Super Member’s username exist, please choose a new one!!” 10. Super Member’s password must be input if its empty after submit struts2 will redirect back to the create New SEC page and show error message”Super Member’s password must be input!”
Result	The new SEC with the

Figure 4.2.16 shows the button for ***Create New SEC***:

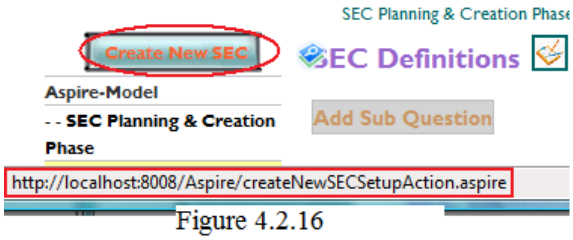


Figure 4.2.16

Figure 4.2.17 shows if all the SEC information and Super Member Account information have been correctly inputted:

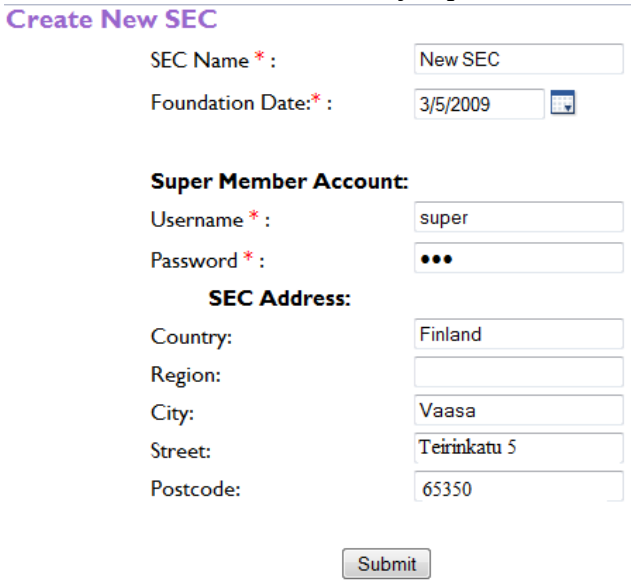


Figure 4.2.17

Figure 4.2.18 shows after submit new SEC will be created and redirect " createSECSuccess.jsp" page to show success information to the administrator:

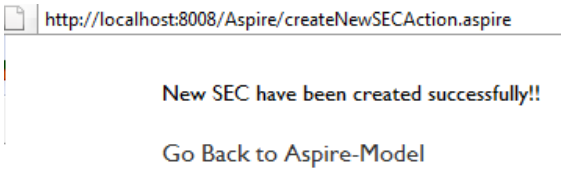


Figure 4.2.18

F7 Super member can modify the answers for the SEC questions, user can see the result of SEC questions after login

Figure 4.2.19 shows sequence diagram of modifying answers when question type is "complex"

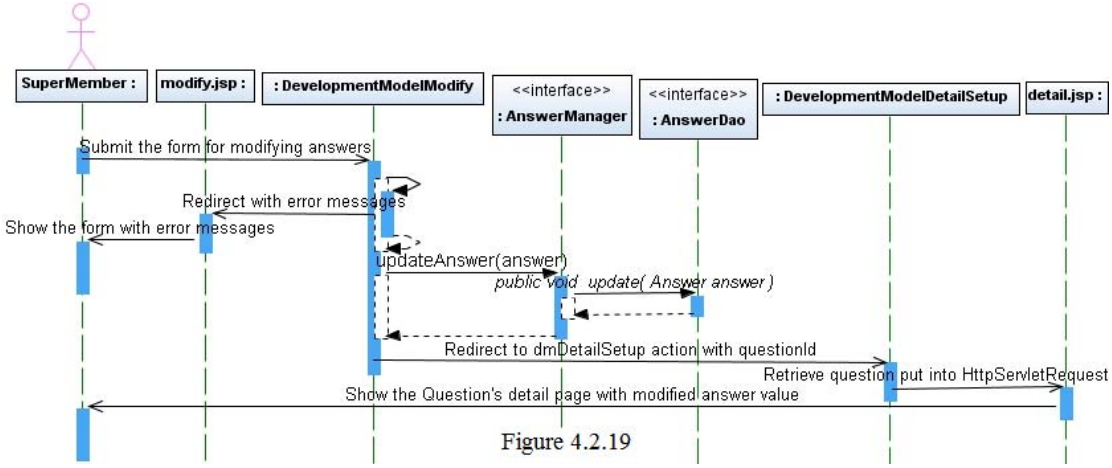


Figure 4.2.19

Figure 4.2.20 shows the sequence diagram of adding new group unit when the question type is “group”

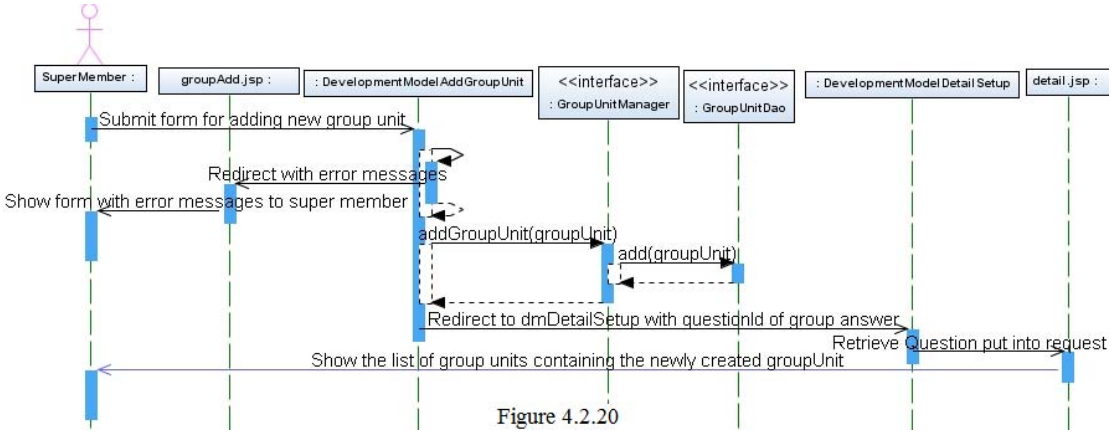


Figure 4.2.20

Figure 4.2.21 shows the sequence diagram of modifying group unit when the question type is “group”

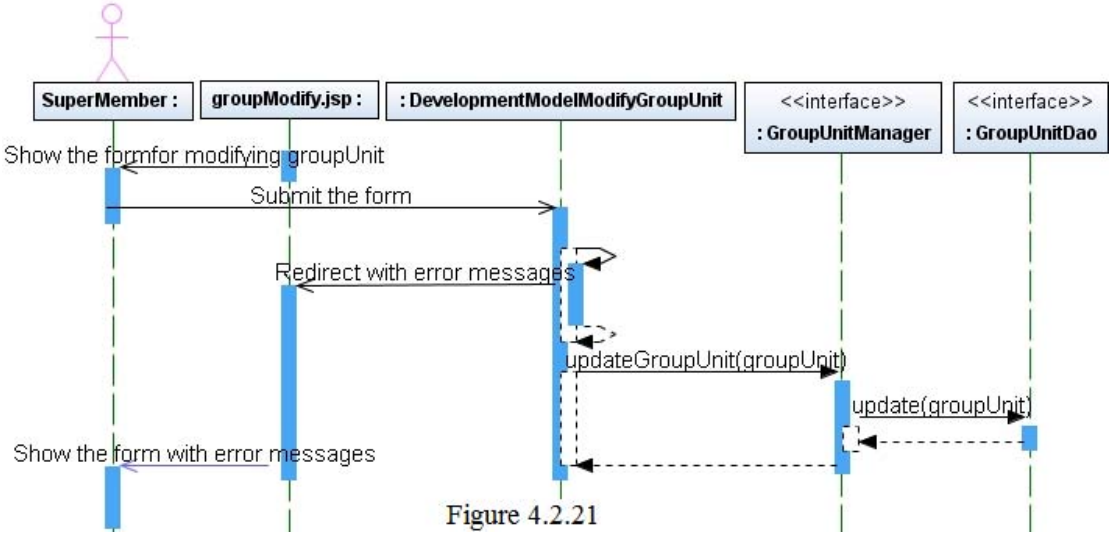


Figure 4.2.21

Figure 4.2.22 shows the sequence diagram of deleting group unit when the question type is “group”

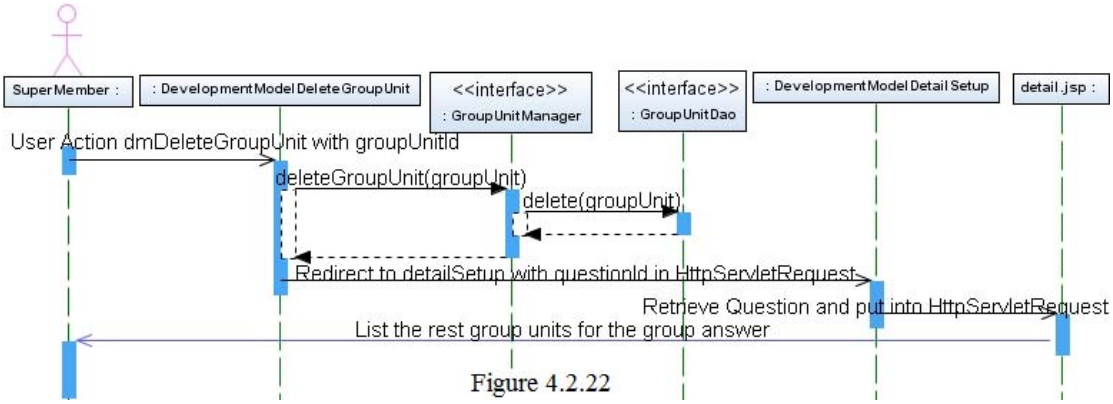


Figure 4.2.22

Figure 4.2.23 shows the sequence diagram when to upload a file

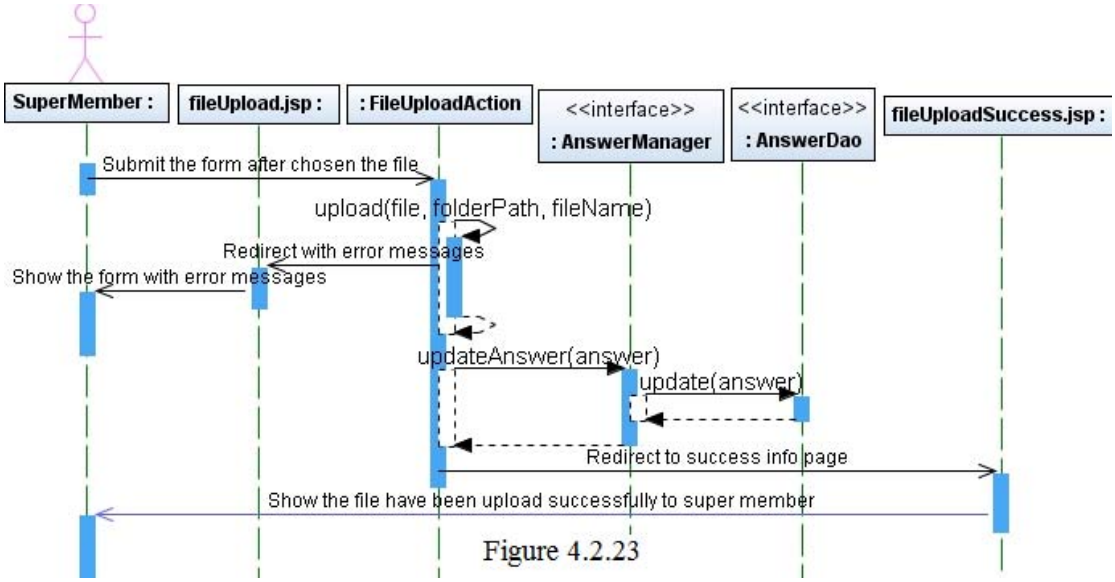


Figure 4.2.23

Preconditions	1. System is not running at its first time, database has super member's SEC and super member account 2. User login as a super member 3. To see the result of SEC questions user have to login as a super member or member, member need link to the SEC
Input	1. Click on the modify for on the question detail page or redirect from browser to the link dmModifySetup 2. In the modify page, modify the value for "textfield", "textarea", "select", "number", "datetimepicker", or upload file for file input.
Description	Super member is the person who takes responsible to fill in the information for the SEC questions. This function is the core part of this project where a lot of dynamical control codes have been implemented in order to show the detail of question's answers, the grid cells for grid answer or the group units of group answer. All the answer type are stored in the database, so by changing the answer type in database the web user interface of the answer will be changed dynamically. All the answers for the same question have sequence to make sure they can be showed sequentially in the form. And for the grid answer it's a 2 dimensional grid table, two loops are implemented to generate the whole grid table with maxX*maxY grid cells, which maxX and maxY are stored in the database, and the grid cell is identified by coordinates axisX and axisY which stored in the database also. For the Question with type menu it's not possible to modify

	anything, because it's a question with no answer, it only shows the list of the children questions so it works like a menu. For the Question with type "complex", things goes like the normal form, on the question's detail page there will be link for modifying the answers' values by clicking on the Modify link the "dmModify.jsp" will be prepared dynamically based on conditional check for the answer's type and loops for the grid answer. In that page the super member can change the value of the "textfield", "textarea", "select", "number", "datetimepicker" or click on the "file" link to open a new window to upload file to the sever. After submitting the form all the submitted parameters will be looped in order to do the dynamical conditional check of the answer's type and answer's name, based on the answer's type and answer's name the new value for that answer will be updated by using "AnswerManager" and "AnswerDao". For the Question with type "group", at the beginning question's detail page will show a list of the group units, which is like a table with answers' names on the first header row and following are the stored group units, only part of the group unit's information showed on this page, and for each data row, there is link for modifying the group unit and the link to delete the group unit. All the group unit should have same list of answers, and to make this happen, a sample group unit with template list of answers will be created before creating any group unit, this template list of answers have no value for any of its answers, answer in the list only store the answer type, answer name, answer sequence. To add new group unit by clicking on the ***Add*** link on top right of the group table, will show "groupAdd.jsp" page, in this page the list of template answers for the group unit will be showed and after super member filling in and submitting, the template answers will be copied and the inputted values will be set to specified answer by using conditional check of answer's name, type. After the answer finished copy and value setting, the new answer will be inserted into database. To modify the group unit's by clicking on the modify link on the right side of the data row, page "groupDetail.jsp" which shows the group unit's answers with their initial values will display, super member can modify the value of the "textfield", "textarea", "select", "number", "datetimepicker" or click on the "file" link to open a new window to upload file to the sever. After submitting the form all the inputted parameters will be looped in order to do the dynamical conditional check of the submitted answer's type

	and answer's name, based on the answer's type and answer's name the new value for that answer will be updated. To delete the group unit just click on the delete link on the right side of the data row, the specified group unit will be retrieved from database and deleted, then user will be redirected to the detail page of the question where show the list of the other group units.
Exception	1. Database not available 2. Hibernate context initialize fail 3. Spring context initialize fail 4. Struts2 context initialize fail 5. No question id in the HttpServletRequest when the action dmModifySetup be executed 6. Answer with type "number" must have numeric input otherwise the page will be redirect back. 7. Answer with type "datetimepicker" must have the date format as "dd/MM/yyyy" otherwise "NaN/NaN/NaN" will be stored into database 8. FileUploadAction detect the file try to upload is null 9. FileUploadAction detect the file try to upload is larger than 2M
Result	Question with type "complex" its answers will be updated after submit correctly. Question with type "group", in this question can have more than one Answer with type "group", and for each group answer its group unit's answers will be updated, or the group unit with its answers will be deleted.

Figure 4.2.24 shows the detail page for the question with type "complex" and where the ***Modify*** link is:

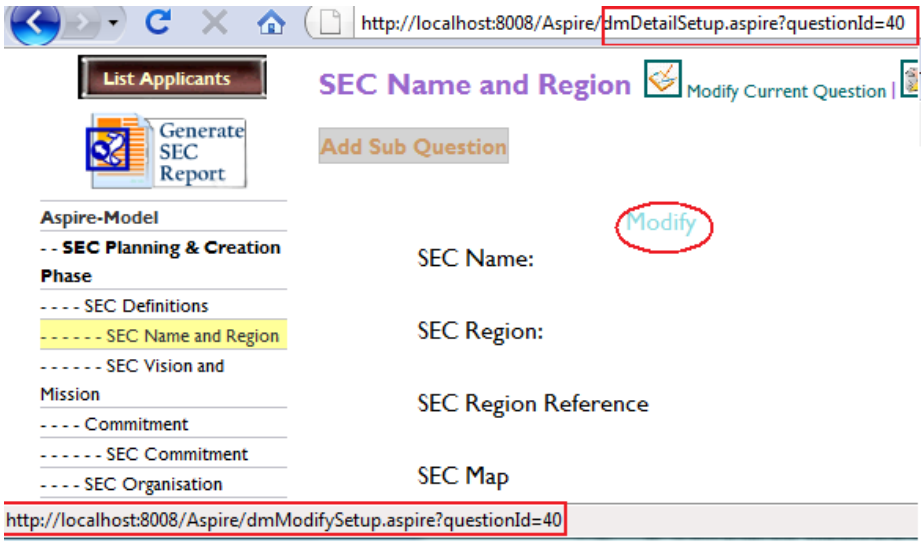


Figure 4.2.24

Figure 4.2.25 shows the “modify.jsp” page:

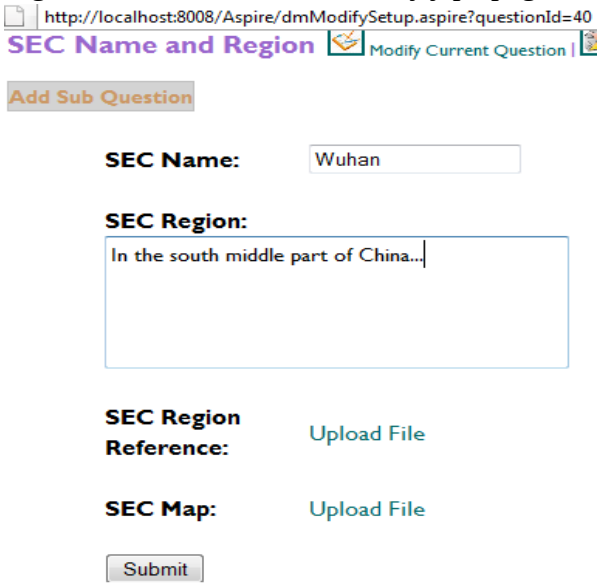


Figure 4.2.25

Figure 4.2.26 shows the after clicking on the ***Upload File*** link will show page “fileUpload.jsp” and open it in a new window:

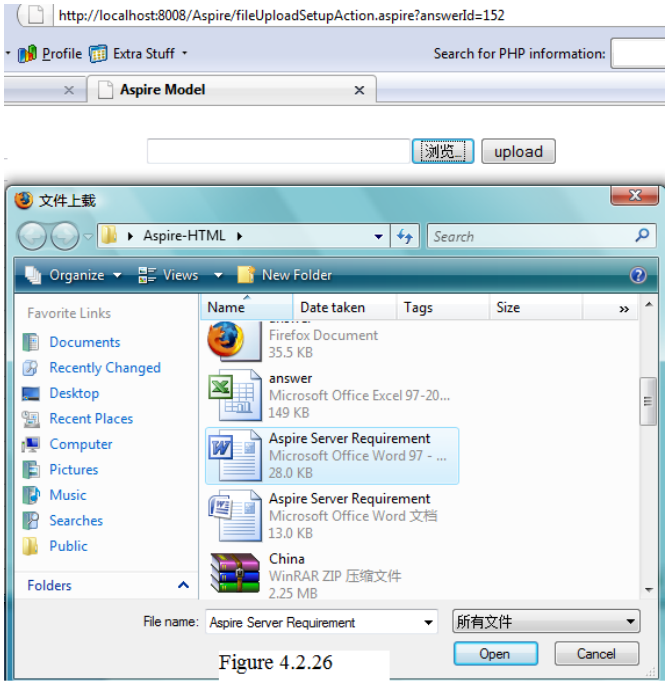


Figure 4.2.26

Figure 4.2.27 shows if the file has been uploaded successfully:

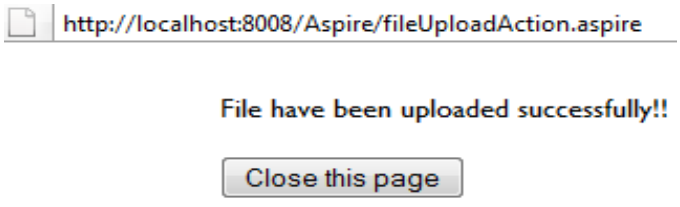


Figure 4.2.27

Figure 4.2.28 shows the question's detail page after updating the answers' values:

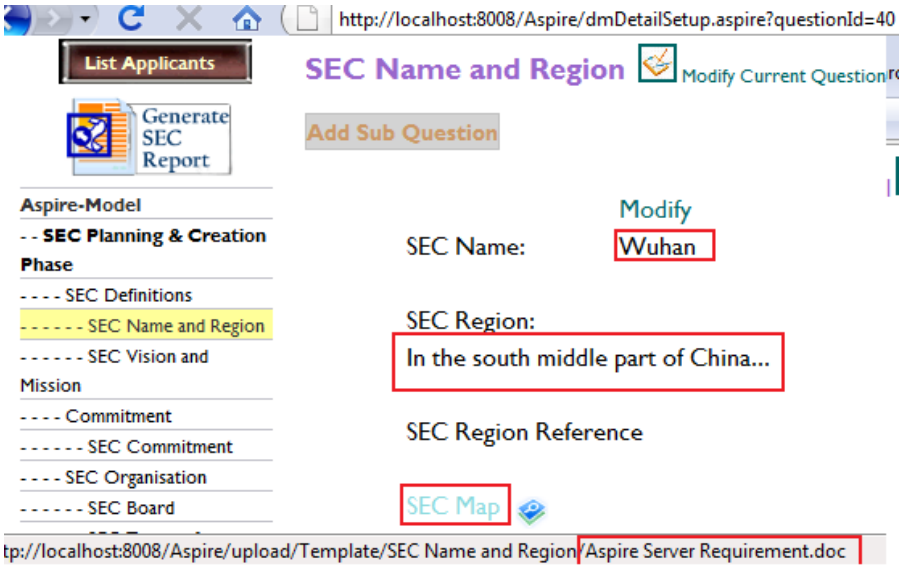


Figure 4.2.28

Figure 4.2.29 shows the detail page of the question with type "group" and ***Add*** link:

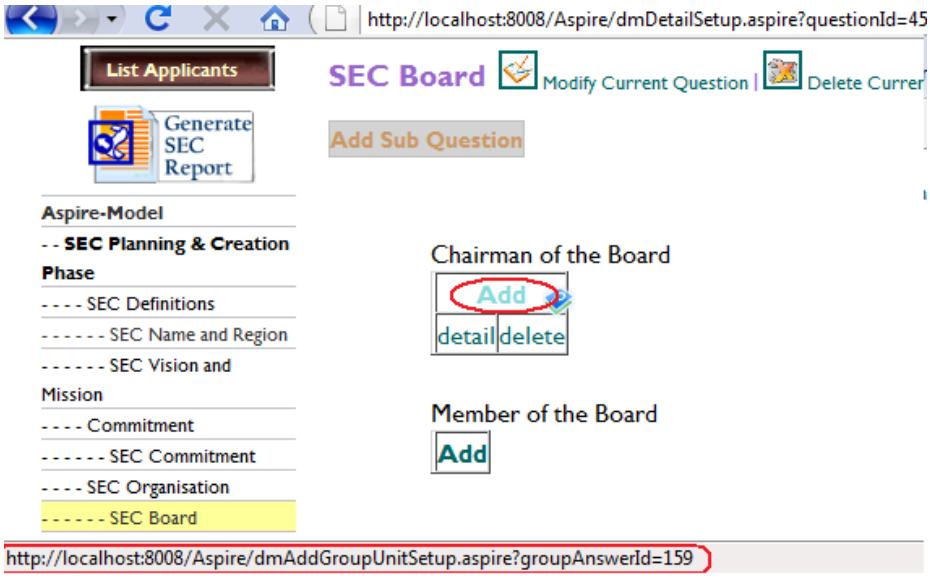


Figure 4.2.29

Figure 4.2.30 shows page “groupAdd.jsp”:

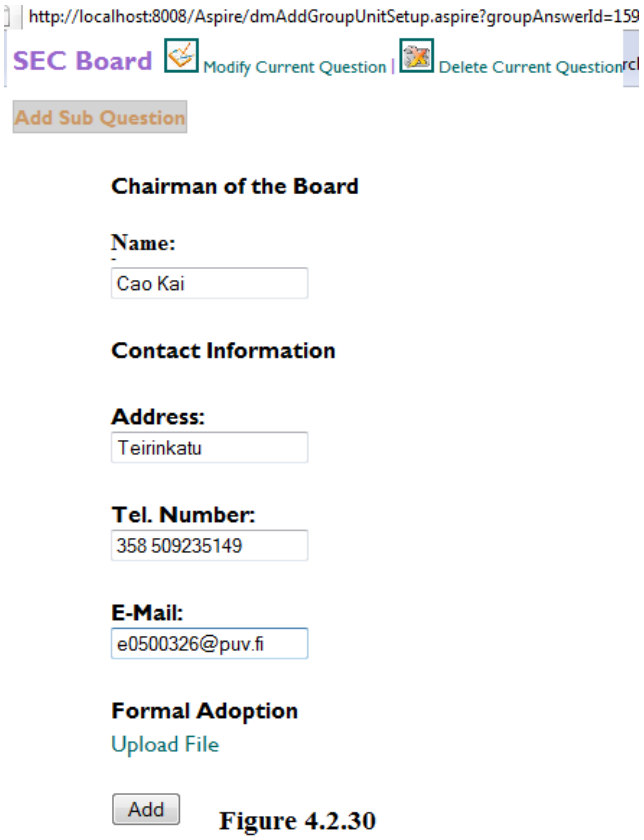


Figure 4.2.30

Figure 4.2.31 shows the detail of question after creating new group unit, the refreshed group units list:

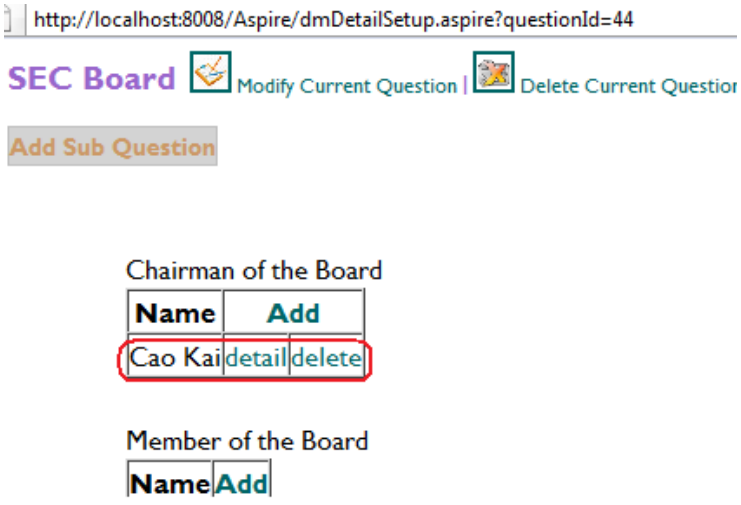


Figure 4.2.31

Figure 4.2.32 shows after click on the ***detail*** link on the right side of the data row the "groupDetail.jsp" page with specified group unit information:

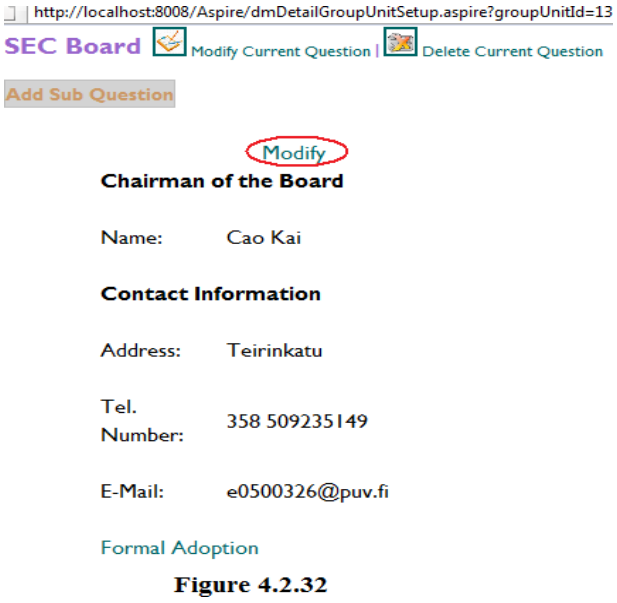


Figure 4.2.32

Figure 4.2.33 shows "groupModify.jsp" page by clicking on the ***Modify*** link:

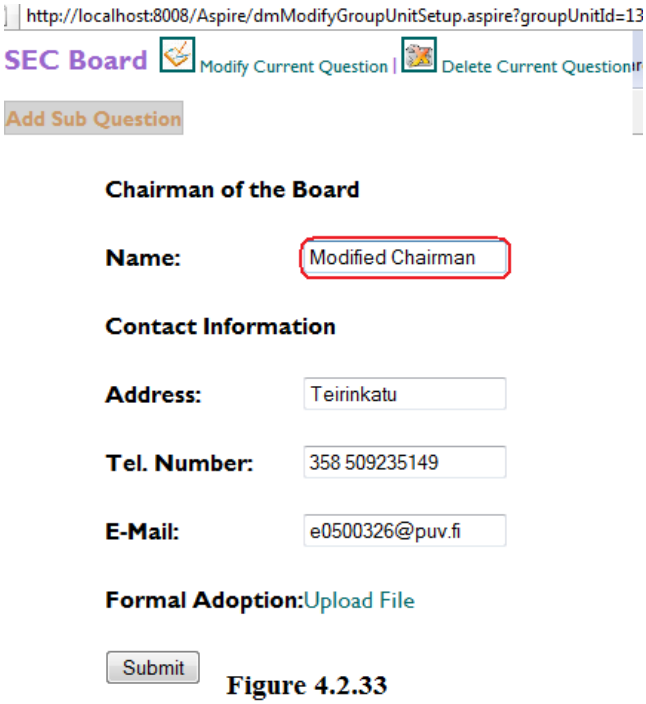


Figure 4.2.33

Figure 4.2.34 shows detail page for the detail of updated group unit after submit the form for modifying group unit information:

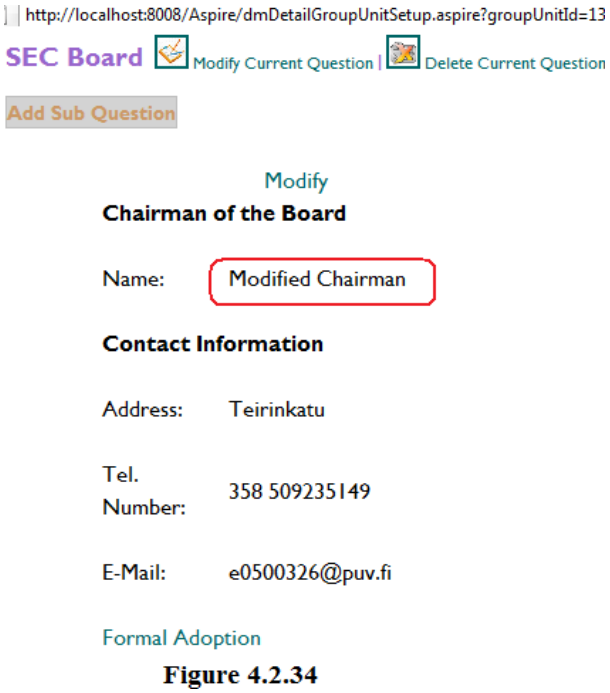


Figure 4.2.34

Figure 4.2.35 shows question's detail page the list of updated group units will show:

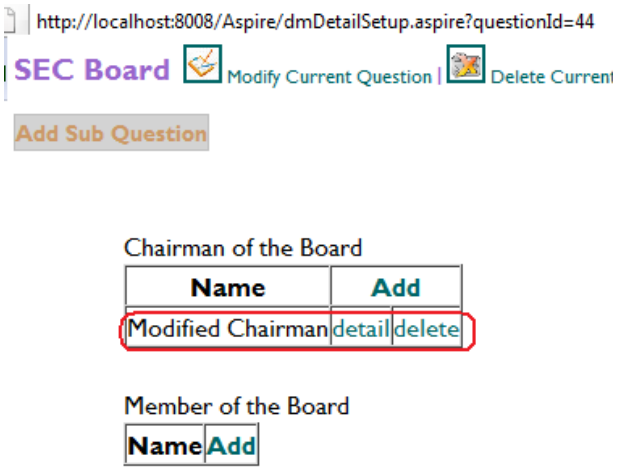


Figure 4.2.35

Figure 4.2.36 shows where to click on the ***delete*** link:

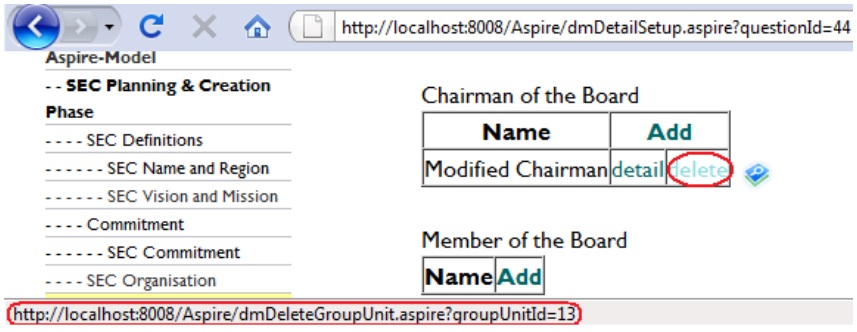


Figure 4.2.36

Figure 4.2.37 shows updated list of group units after clicking on the ***delete*** link:

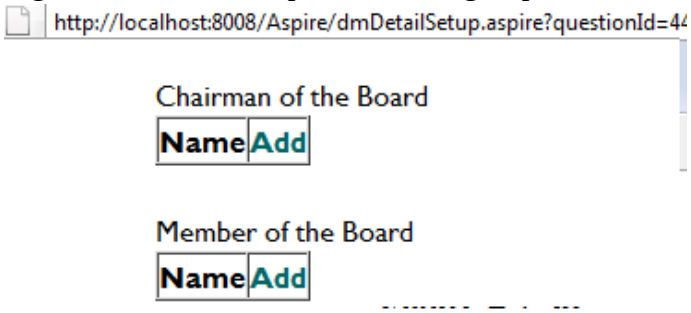


Figure 4.2.37

F8 Super member can modify the progress bar for the SEC questions, and user can see the result of progress bar after login

Figure 4.2.38 shows the sequence diagram for modifying the progress bar:

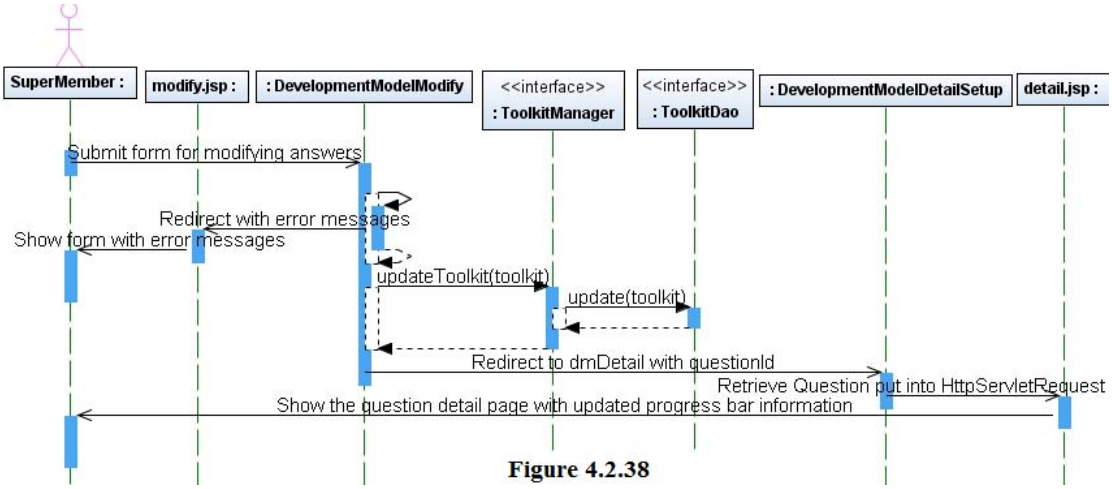


Figure 4.2.38

Preconditions	1. System is not running at its first time, database has super member's SEC and super member account 2. User login as a super member 3. To see the result of progress bar user has to login as a super member or member, member need link to the SEC
Input	1. Click on the modify for on the question detail page or redirect from browser to the link "dmModifySetup.aspire" 2. In the modify page, modify the value for "criteria", "unit", "owner", "target date" of the progress bar. 3. In the modify page, click on the grade button to grade the progress.
Description	Super Member has the right to change the progress of the SEC question, which is related with "Toolkit" table in the database. Progress bar on the page has two selects for "criteria" and "unit",

	and one panel for progress which contains 5 buttons from grade 1 to 5, and after clicking on the button the progress bar will generate a line of progress from 0% to 100%, it's controlled by the JavaScript. After super member submit the modified question, the progress bar value will also updated to the "Toolkit" table. Later if super member or member go back to the question's detail page, the progress bar will show the "criteria" "unit" "grade" "owner" "target date" and progress to the user.
Exception	1. Database not available 2. Hibernate context initialize fail 3. Spring context initialize fail 4. Struts2 context initialize fail 5. No question id in the HttpServletRequest when the action dmModifySetup be executed 6. Browser doesn't support DOM object for JavaScript
Result	Table "Toolkit" will be updated, and progress bar with updated information will showed to the member and super member.

Figure 4.2.39 shows the progress bar in the page "modify.jsp":

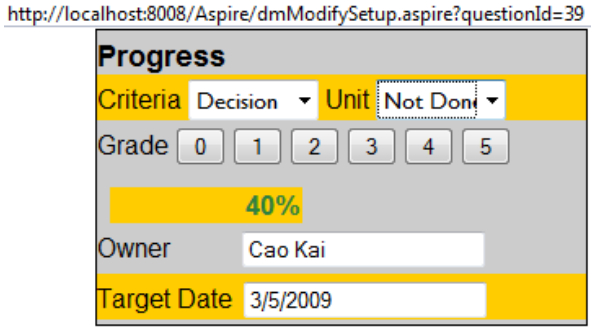


Figure 4.2.39

Figure 4.2.40 shows the updated progress bar in the question's detail page:

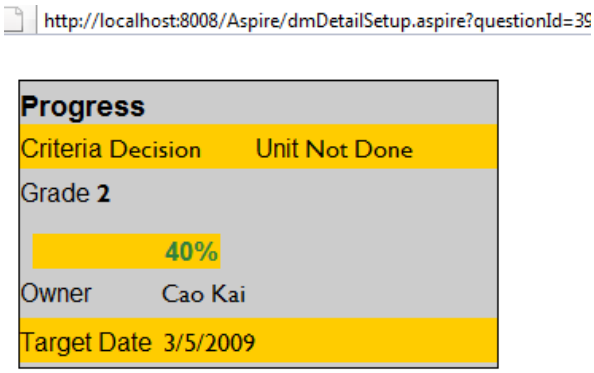


Figure 4.2.40

F9 Super member and member can generate his/her SEC report

Figure 4.2.41 shows the sequence diagram for generating SEC report:

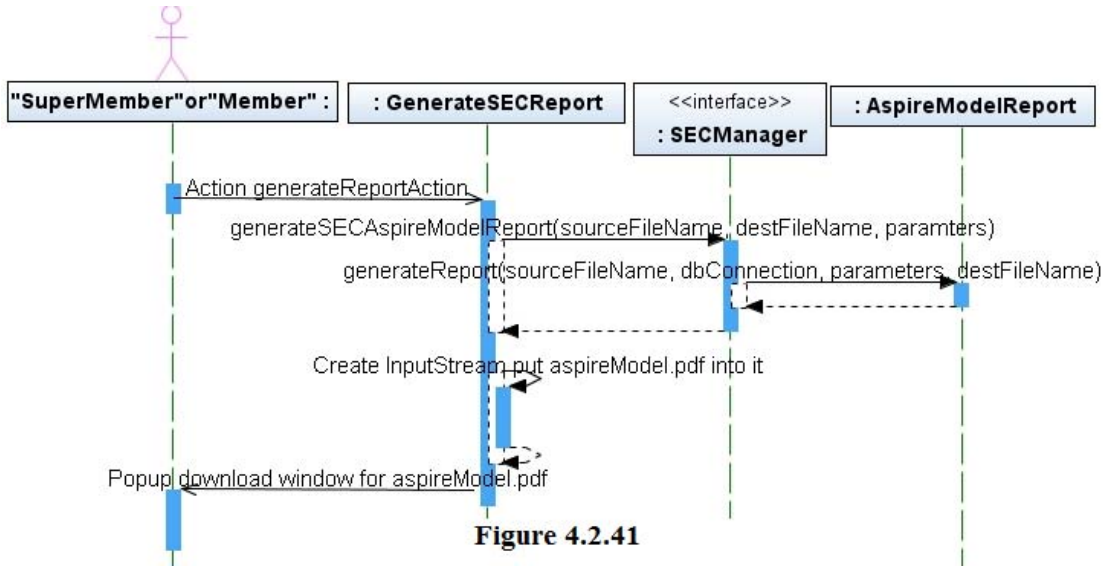


Figure 4.2.41

Preconditions	1. System is not running at its first time, database has super member's SEC and super member account 2. Member has been accepted to join the SEC 3. User login as a super member or member
Input	1. Click on the top left link ***Generate SEC Report*** 2. Super member or member redirect to "generateReportAction.aspire" from browser
Description	There is a need for generating SEC report based on its questions, answers and toolkits. The report is a PDF file, and created by using "JasperReport", the report template is created by "iReport" and the compiled template files have been stored in the "resources/jasperReport/template" folder. When the super member or member click on the link generate SEC report, Struts2 will notice to generate aspireReport.pdf into folder "resources/jasperReport/report". After the aspireReport.pdf is created successfully, a stream of this PDF file will transfer to user's browser, so they can see the download window of the aspireReport.pdf.
Exception	1. Database not available 2. Hibernate context initialize fail 3. Spring context initialize fail 4. Struts2 context initialize fail
Result	aspireReport.pdf will be put into an InputStream and work as a

	stream to let the super member or member to download

Figure 4.2.42 shows the ***Generate SEC Report*** link on the top left side of the page:

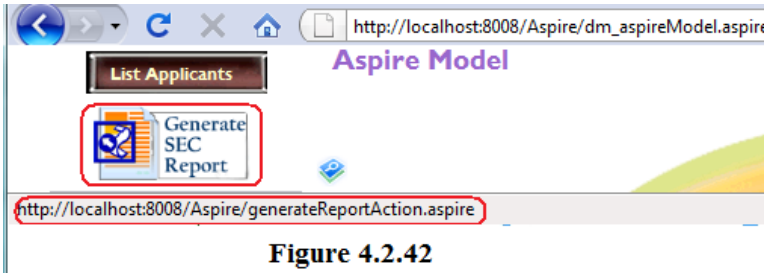


Figure 4.2.42

Figure 4.2.43 shows after click on the ***Generate SEC Report*** link a download window will popup to ask super member or member to download the file:

Figure 4.2.43

F10 Member can apply for one SEC

Figure 4.2.44 shows the sequence diagram for applying for SEC:

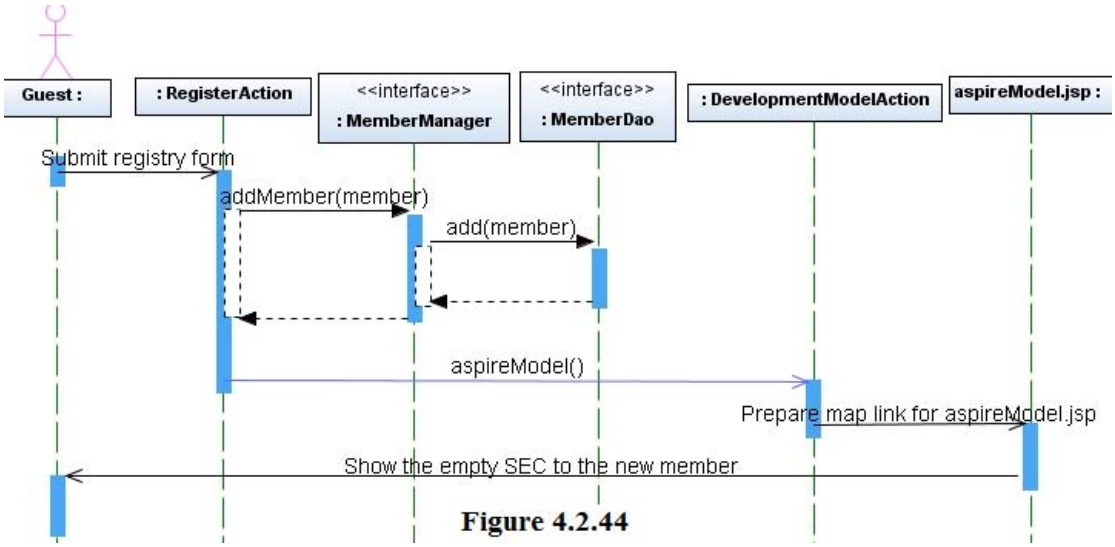


Figure 4.2.44

Preconditions	1. System is not running at its first time, database has super member's SEC and super member account
Input	1. User click on the ***register*** link on the left side in the login panel 2. User redirect to "registerSetupAction.aspire" from browser
Description	From a guest to be a member, this function provides the key for the member be a part of the SEC. In the register page there is a select field for the user to choose which SEC he/she want to join except the "Template" SEC. The list of SECs is dynamically created based on the SECs in the database without "Template" SEC. After user choose the SEC and submits the register form, the user account will be put in the list of SEC applicants who applied for joining this SEC; super member can check the list and confirm the user as a member. If super member confirm the user to be a member of SEC, user can see the questions and answers of such SEC after login.
Exception	1. Database not available 2. Hibernate context initialize fail 3. Spring context initialize fail 4. Struts2 context initialize fail
Result	New member will be put into the list of applicants for SEC he/she applied for and waiting for the super member of that SEC to confirm his/her account to join SEC.

Figure 4.2.45 shows where the register link is:

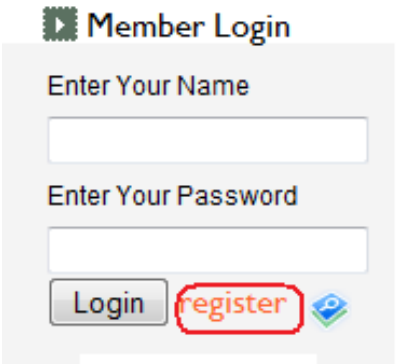


Figure 4.2.45

Figure 4.2.46 shows the register page when the user applied for SEC:

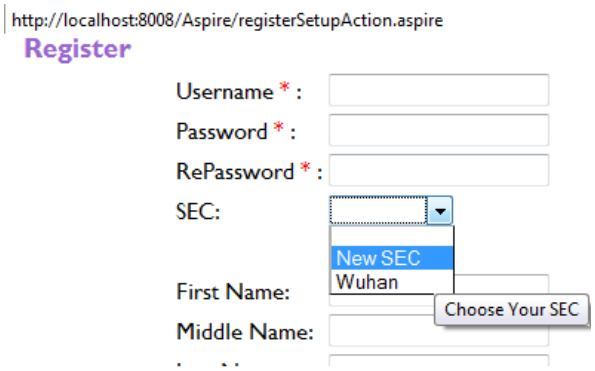


Figure 4.2.46

F11 Super member confirm applicant to be a member

Figure 4.2.47 shows the sequence diagram for confirming applicant to be a member:

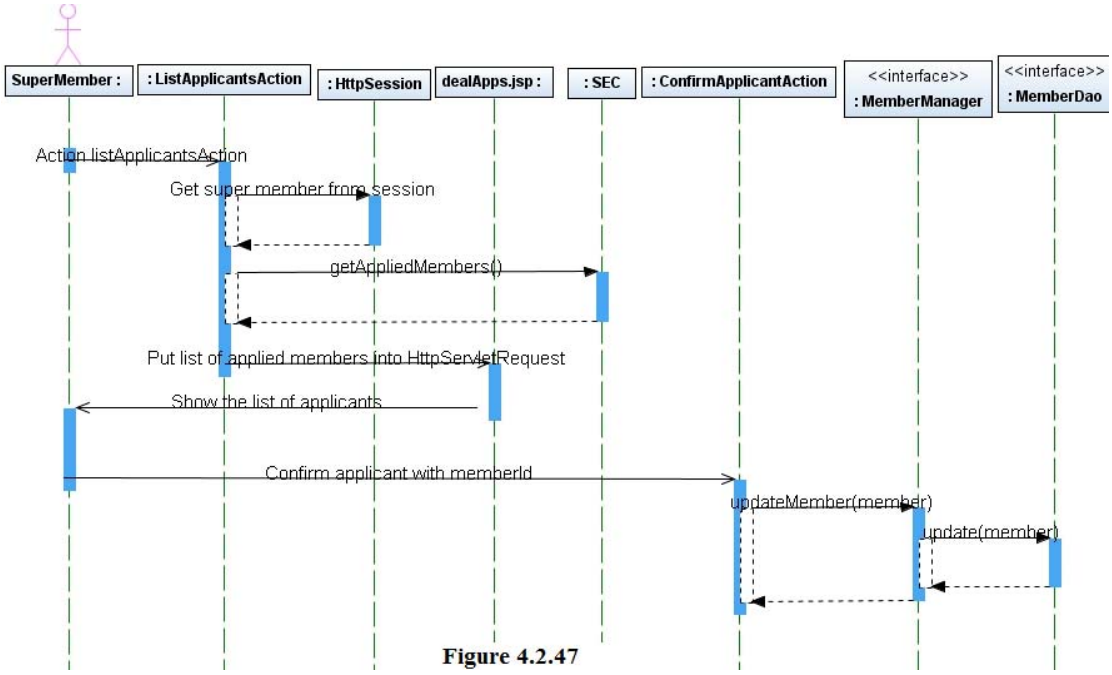


Figure 4.2.47

Preconditions	1. System is not running at its first time, database has super member's SEC and super member account 2. New member applied to be a part of SEC 3. User login as a super member
Input	1. Click on the button ***List Applicants*** on the top left side of the page 2. Confirm member applicant by clicking on the ***confirm*** button 3. Confirm member applicant by checking the detail information of the member who applied for the SEC and click on the ***confirm*** button

Description	SEC super member is the person who has the right to confirm the applicants of members who applied for the SEC. To implement this function, the super member can click on the ***List Applicants*** button on the top left side of the page after login. A list of applicants will show with applicants' username and real name, super member can confirm the member row by row, by clicking on the ***confirm*** button. Super member can also see the detail information of the applicant's account by clicking on the username or real name of the applicant will show the page "detailApplicant.jsp" with the applicant's information. After look through the detail information of the applicant, super member can confirm it by clicking the ***confirm*** button on the bottom of the detail content. After the applicant is confirmed the list of the rest applicants will be updated and showed to the super member. And the confirmed applicant owner can login and work as a member of the SEC.
Exception	1. Database not available 2. Hibernate context initialize fail 3. Spring context initialize fail 4. Struts2 context initialize fail
Result	Applicant will be accepted and the applicantion owner can login as a member of the SEC

Figure 4.2.48 shows the "dealApps.jsp" page after clicking on the ***List Applicants*** button on the top left:

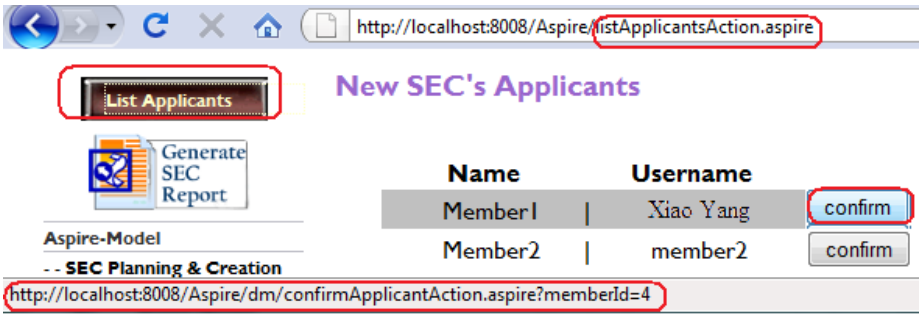


Figure 4.2.48

Figure 4.2.49 shows page “detailApplicant.jsp” after clicking on the name or username:

Figure 4.2.49

Figure 4.2.50 shows after the updated list of applicants after confirm the applicant:

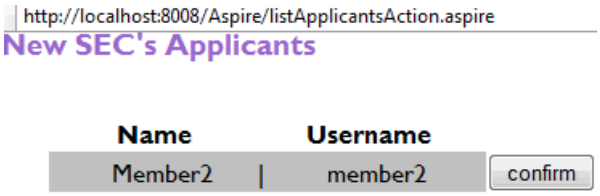


Figure 4.2.50

F12 User can register to be a member

Figure 4.2.51 shows the sequence diagram for registering to be a member:

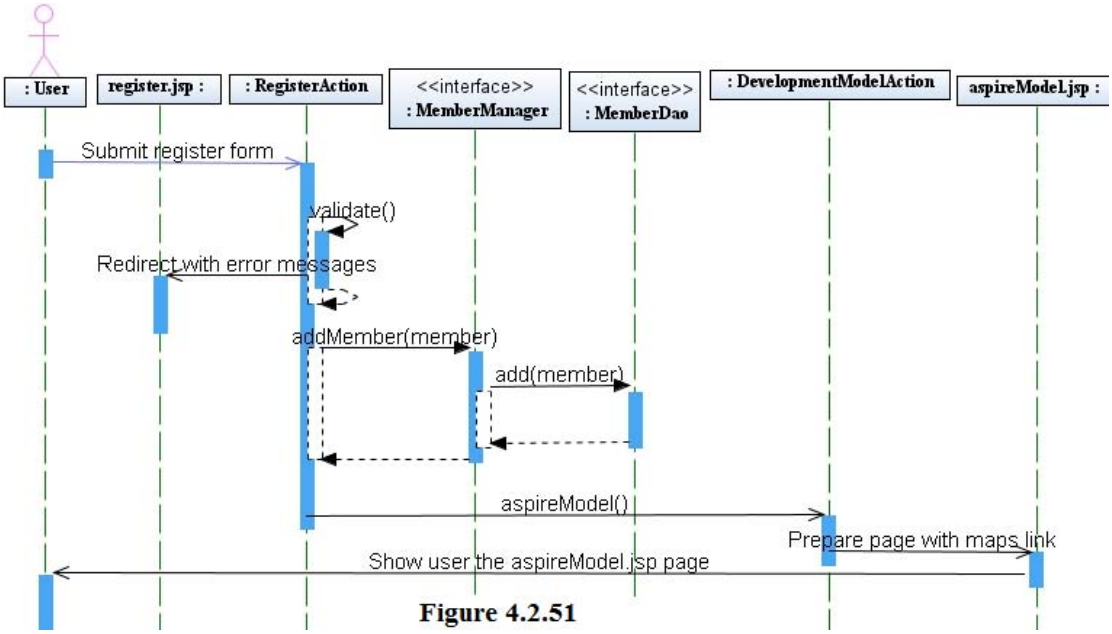


Figure 4.2.51

Preconditions	
Input	1. User click the ***register*** link on the login div 2. User action registerSetup.aspire from browser 3. Register must input username, password, repassword 4. Register will ask user to choose one SEC to join. 5. Register personal information first name, middle name, last name, age, gender, career, email, phone 6. Register address information country, region, city, street, postcode
Description	User can register and apply for a membership of the Aspire Model solution by clicking the ***register*** link on the login panel or action "registerSetup.aspire" from browser, the register.jsp page will be prepared and showed to user ask he/she to input account, personal, address information and user can choose one SEC to join(mentioned in F10).
Exception	1. Database not available 2. Hibernate context initialize fail 3. Spring context initialize fail 4. Struts2 context initialize fail 5. Username, password, repassword must not be empty 6. Username must be unique otherwise "Username exist!" warning message will show 7. Password and repassword must be the same otherwise

	“Password and repassword are not the same!” will show
Result	New member will be inserted into ”member” table in the database

F13 Dynamic Tree Navigation of the SEC Questions

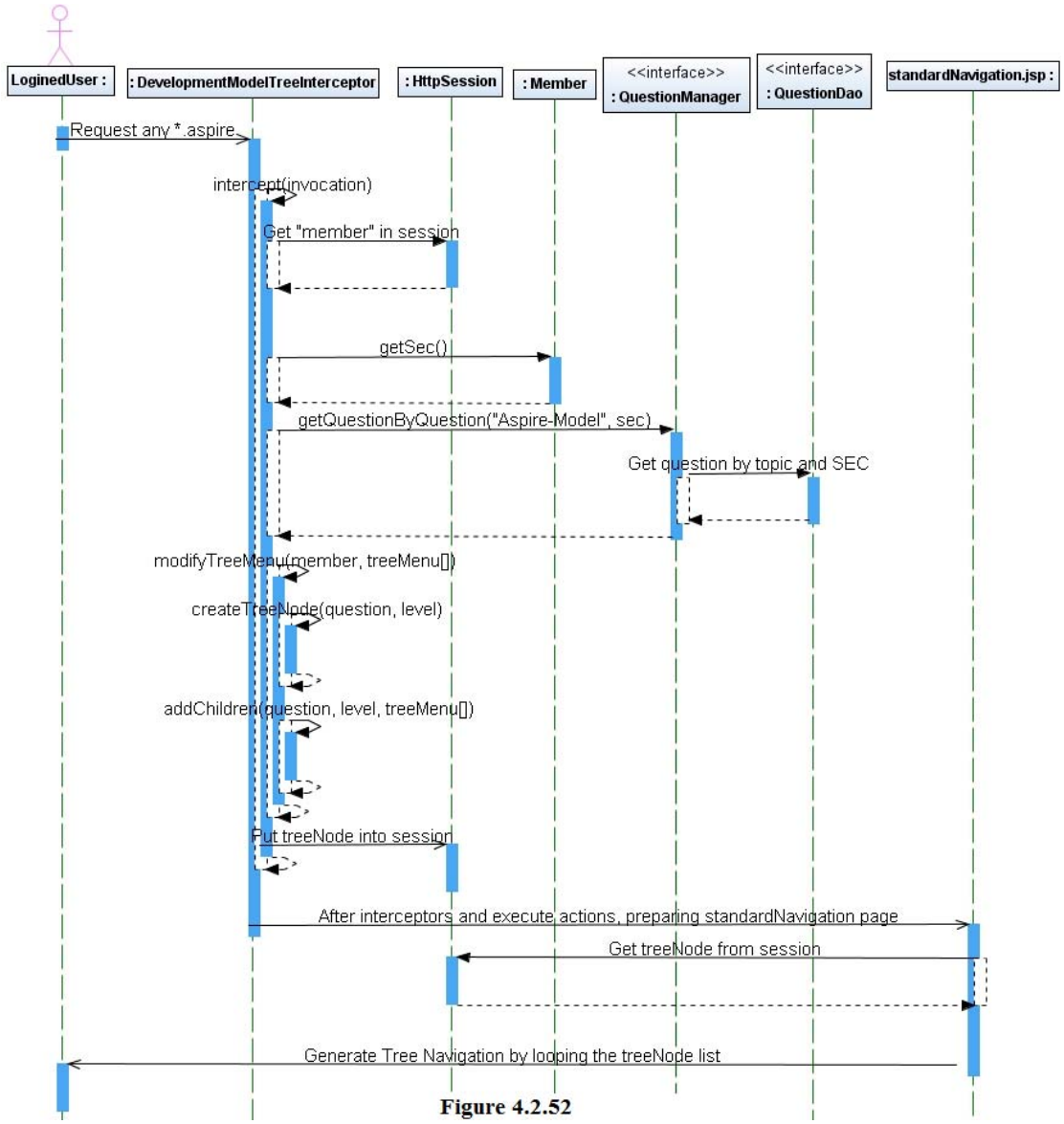


Figure 4.2.52

Preconditions	1. System is not running at its first time, database has super member’s SEC and super member account 2. User login as a super member, member or administrator. Member need link to SEC
Input	1. User request any *.aspire link
Description	By requesting *.aspire resources, Struts2 will go to struts.xml to check the resource, and before run any action class the interceptors will run firstly. Here a interceptor defined for create tree menu navigation for the SEC’s questions.

	The procedure to create the tree navigation is at the beginning get the "member" object from HttpSession which has been put into it when login and then get SEC of that "member" object. Next the root question with name "Aspire-Model" in the SEC will be retrieved and put into the first place of the list of tree nodes. Then all its children questions will be recursively retrieved and decorated with tabulates and <a> link, after that put them into the list of tree nodes. After all the questions are decorated and put into list of tree nodes, the list will be put into the session. Finally, "standardNavigation.jsp" has loop to show all the data in the tree nodes on the left side of the page.
Exception	1. Database not available 2. Hibernate context initialize fail 3. Spring context initialize fail 4. Struts2 context initialize fail
Result	All the questions for login user's SEC will be decorated with tabulates and <a> link put into HttpSession, and showed on the left side of the page.

Figure 4.2.53 shows a part of the tree menu:

Figure 4.2.53

F14 Quick Navigation of the question

Figure 4.2.54 shows the sequence diagram for quick navigation for the detail page of question with type "complex":

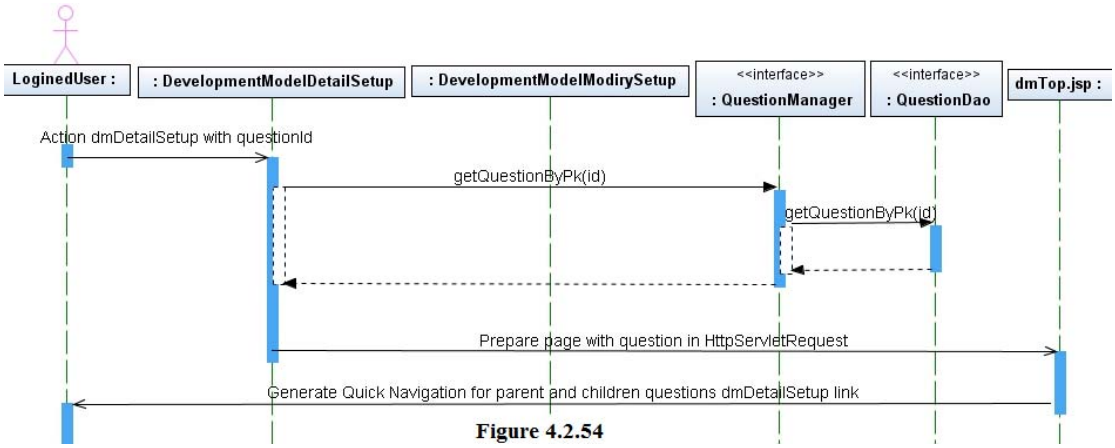


Figure 4.2.54

Figure 4.2.55 shows the sequence diagram for quick navigation of adding group unit:

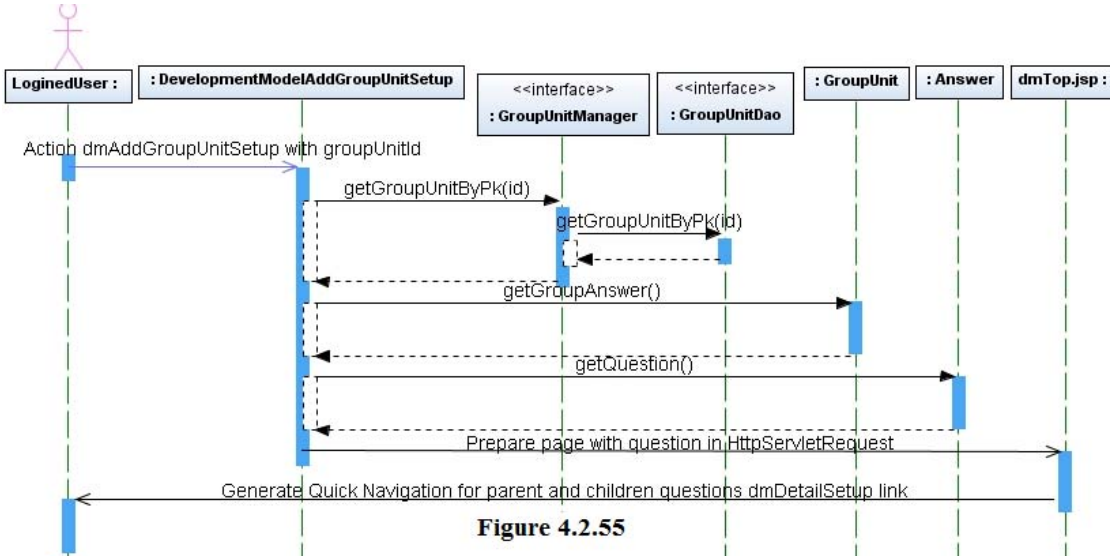


Figure 4.2.55

Preconditions	1. System is not running at its first time, database has super member's SEC and super member account 2. User login as a super member, member or administrator. Member need link to SEC 3. User execute action "dmDetailSetup.aspire", "dmModifySetup.aspire" with question id or "dmAddGroupUnitSetup.aspire" "dmDetailGroupUnitSetup.aspire" "dmModifyGroupUnitSetup.aspire" with group unit id, from navigation or browser
Input	
Description	SEC's questions have a tree structure, from the root question "Aspire-Model" the tree expands to its children and children's children... until no further child node exists. When the question's detail page is requested there will show the

	quick navigation for the question's parent node and children nodes on the top of the content on the page. When group unit's detail page is requested the group answer for such group unit will be retrieved, and then the question for that group answer will be got and based on the question the quick navigation will show the parent node and children nodes on the top of the content on the page. "dmTop.jsp" is the place to dynamically generate the quick navigation.
Exception	1. Database not available 2. Hibernate context initialize fail 3. Spring context initialize fail 4. Struts2 context initialize fail
Result	dmTop.jsp will generate the quick navigation for the related question's parent question and children questions dynamically, then show to the user.

Figure 4.2.56 shows one example of the quick navigation compared with tree navigation:

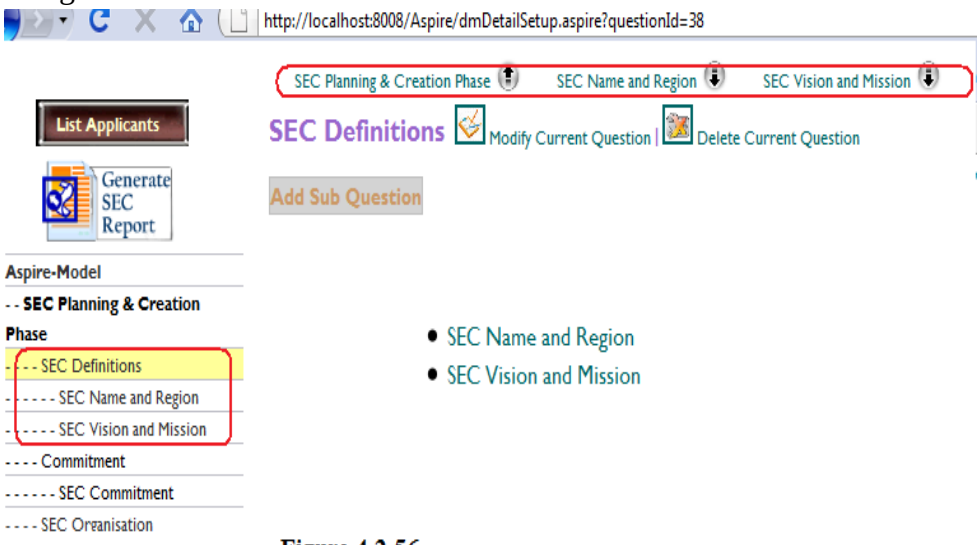


Figure 4.2.56

F15 User login and logout

Figure 4.2.57 shows the sequence diagram for user login:

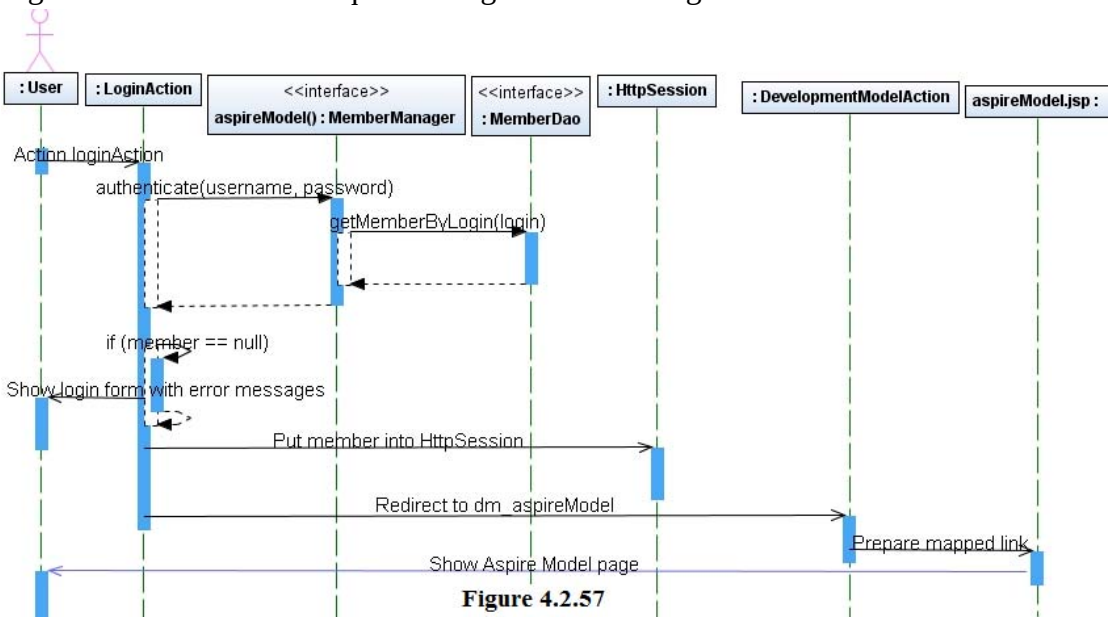


Figure 4.2.57

Figure 4.2.58 shows the sequence diagram for user logout:

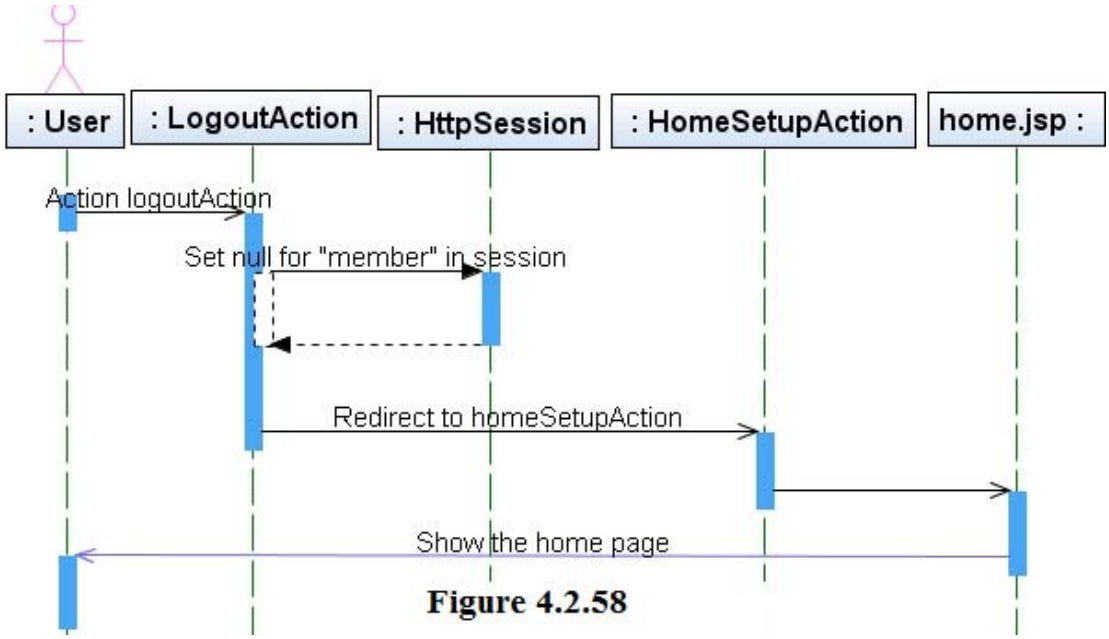


Figure 4.2.58

Preconditions	1. Among the member, super member and administrator have been created. 2. Logout need user login first
Input	1. Username of the account 2. Password of the account
Description	User can login to the Aspire Model solution after register as a member.

	To implement such function, the "loginAction.aspire" to authenticate the username and password, if member's account with such login information exists in the "Member" table, the member will be retrieved and put into HttpSession for later use. After authenticate user will redirected to aspire model road map.
Exception	1. Database not available 2. Hibernate context initialize fail 3. Spring context initialize fail 4. Struts2 context initialize fail 5. Username and password must input otherwise warning messages will show to ask user to input. 6. Username or password is invalid, when login failed
Result	Login successfully and redirect to the action dm_aspireModel.aspire to show the Aspire Model.

F16 User can manage profile and change password

Figure 4.2.59 shows the sequence diagram for managing profile:

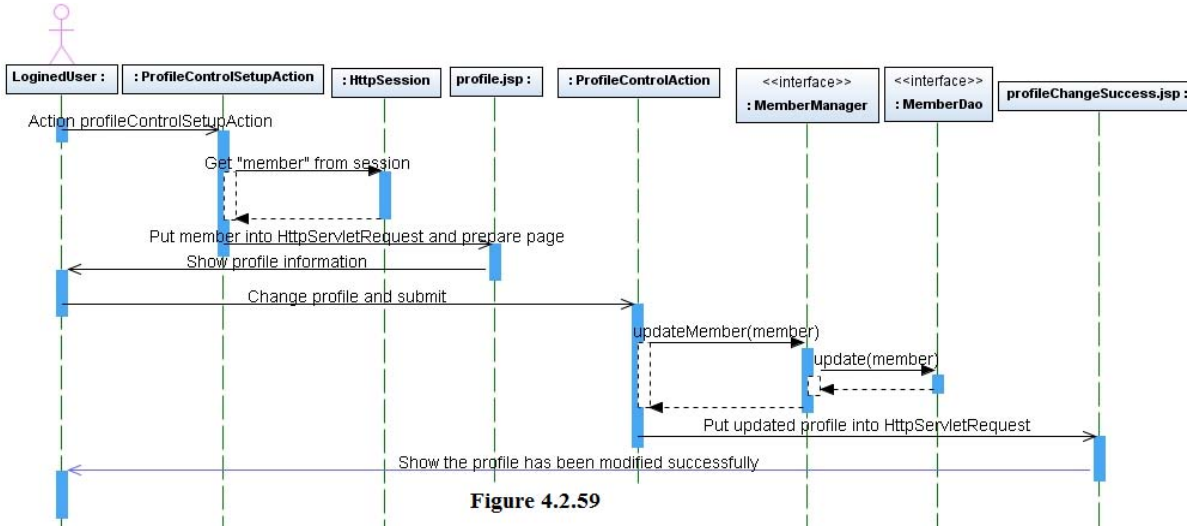


Figure 4.2.59

Figure 4.2.60 shows the sequence diagram for changing password:

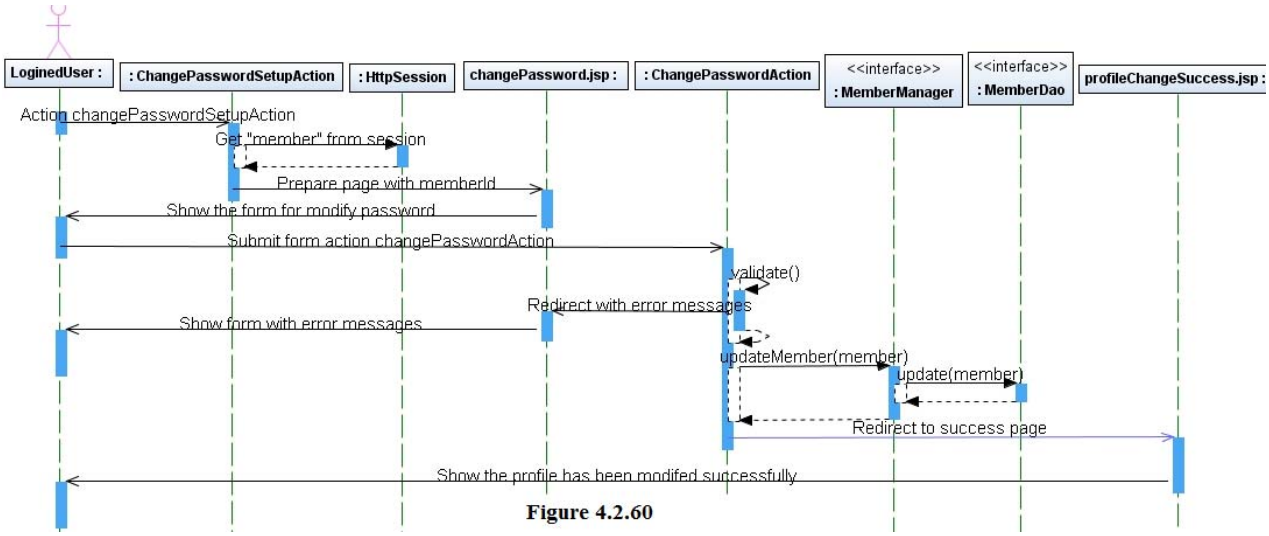


Figure 4.2.60

Preconditions	1. System is not running at its first time, database has super member's SEC and super member account 2. User login as a super member, member or administrator. Member need link to SEC
Input	1. For profile control, need personal information first name, middle name, last name, age, gender, career, phone, email. 2. For profile control, need address information country, region, city, street, postcode 3. For password change must input pre-password, new password, and re-password
Description	After user login to the Aspire Model solution, user can control his/her own profile and change the password of the account. User can click on the button ***Profile Control*** on the top right of the page. By clicking on the button "profile.jsp" will be prepared with current user's information. User can change the personal information and address information in "profile.jsp" and submit form to update the "Member" table. After the profile is changed successfully the success information of profileChangeSuccess.jsp will show. To change the password, click link ***change password*** on the "profile.jsp" page "changePassword.jsp" will be showed, user can change password by inputting previous password, new password and rewrite new password, before update the new password. Previous password must be the same password before change, and new password must be the same as the rewrite new password. After all the validate checks successfully, the new password will be updated. So later the user can login with new password. Finally the "profileChangeSuccess.jsp" will show.
Exception	1. Database not available 2. Hibernate context initialize fail 3. Spring context initialize fail 4. Struts2 context initialize fail 5. When changing password, pre-password, new password, re-password have to be input, otherwise error messages will show and ask user to input. 6. Pre-password must be exact the same of the password before change, otherwise error message will show 7. New password and rewrite new password must be the same, otherwise error message will show
Result	User profile, password changed successfully.

Figure 4.2.61 shows the link for profile control on the top right of the page:

Figure 4.2.61

Figure 4.2.62 shows after click on the profile control link the “profile.jsp” will show:

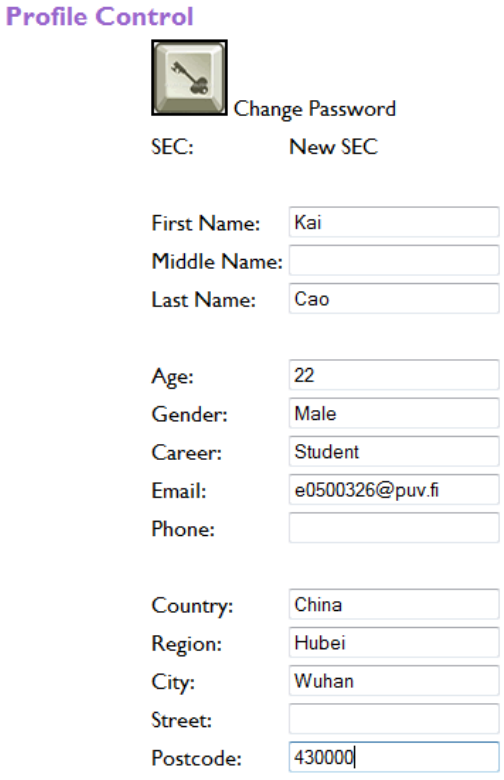


Figure 4.2.62

Figure 4.2.63 shows after submit the form for “profile.jsp”, “profileChangeSuccess.jsp” shows:

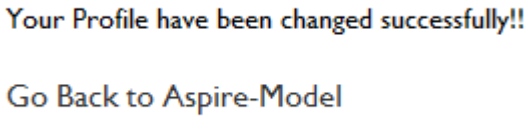


Figure 4.2.63

F17 Secure SEC to make sure user cannot go to other SEC

Figure 4.2.64 shows the sequence diagram for securing SEC:

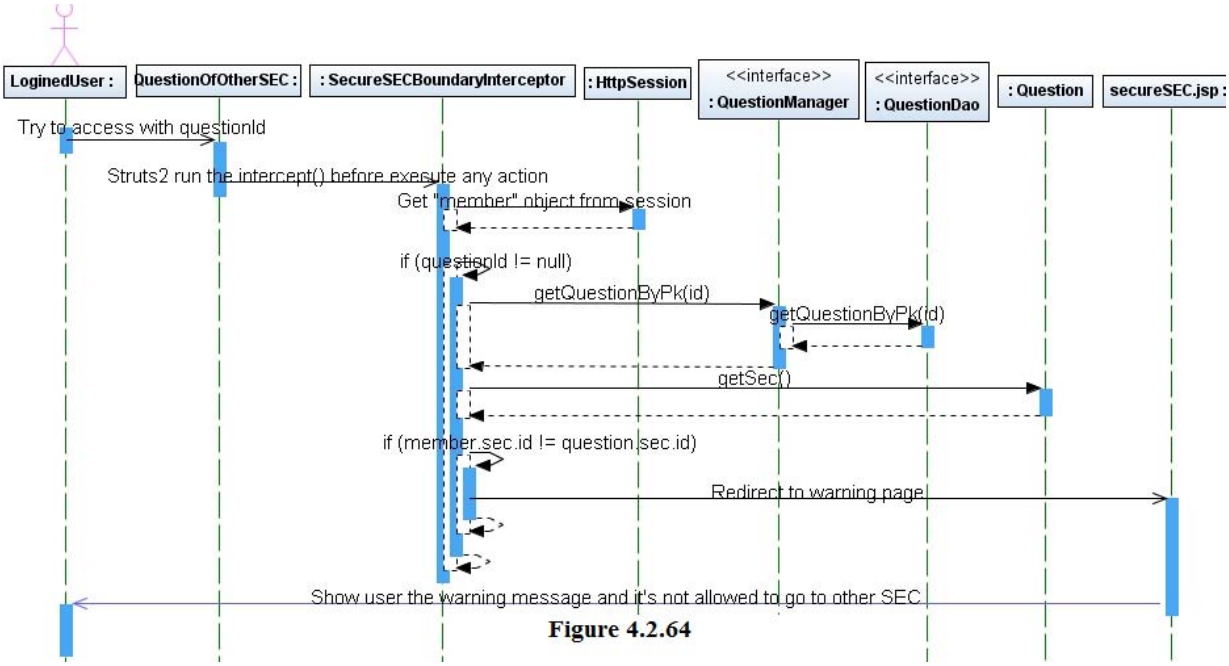
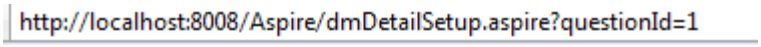


Figure 4.2.64

Preconditions	1. System is not running at its first time, database has super member's SEC and super member account 2. User login as a super member, member or administrator. Member need link to SEC
Input	
Description	After user login to the Aspire Model solution, user will belong to one SEC, if no SEC assigned to the user an empty one will show. But the solution use parameters like questionId groupUnitId… to redirect page from browser, if the user tried to break the system by typing the question's id for questionId of other SEC for example, user is not allowed to do that and need show warning message. So "SecureSECBoundaryInterceptor" be created, it works as other interceptors runs before the Struts2 actions execute, and run intercept function automatically. So this interceptor has been set for every request of *.aspire as a default interceptor for any Struts2 Action. In the intercept function, if question id exists in the request parameters, question with that id will be retrieved and the "member" object in HttpSession will be got. Both question and member has linked to SEC, so by comparing question's SEC and member's SEC, whether they have the same SEC id, we know who is cheating! If anything likes this happened intercept will return "secureSEC" result, and redirect to secureSEC.jsp to show warning message.

Exception	1. Database not available 2. Hibernate context initialize fail 3. Spring context initialize fail 4. Struts2 context initialize fail
Result	secureSEC.jsp will be showed and warn user do not try to go to others' SEC by changing the link value.

Figure 4.2.65 shows one super member want to go to Template SEC's question with "questionId=1" which is only can see by administrator, page redirect to secureSEC.jsp and show the warning message:



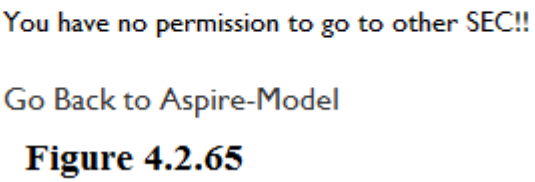


Figure 4.2.65

5. DESIGN AND STRUCTURE

Aspire Model has two major versions, and implemented during 3 months.

5.1. Static Version

For the first main version is a static version, the most important difference between two versions, is the data model structure, for this one the data model was created based on the form, and each form has table to support its input fields. A lot of different types of database tables were hardcoded to be the sub tables of the “Answer” database table, so in the Answer table there was a column to store the discriminator of sub table’s type. So the database tables will explosion if there are too many forms in the project.

5.2. Dynamic Version

For the second version the data model has changed to be more dynamic and more reusable (Showed in next chapter), so the core part of the Aspire model became dynamic and based on answer type and answer value, so each form can link several answers work as its fields. Using this advanced version the administrator can create Template and super member can change the template by adding, modifying or deleting form’s fields. Later the form itself became dynamic and super member or administrator can create new form or modifying, deleting form as they want, so the structure of the Aspire Model solution can be fully controlled.

Figure 5.1.1 shows the 3-tier structure for Aspire-Model Solution

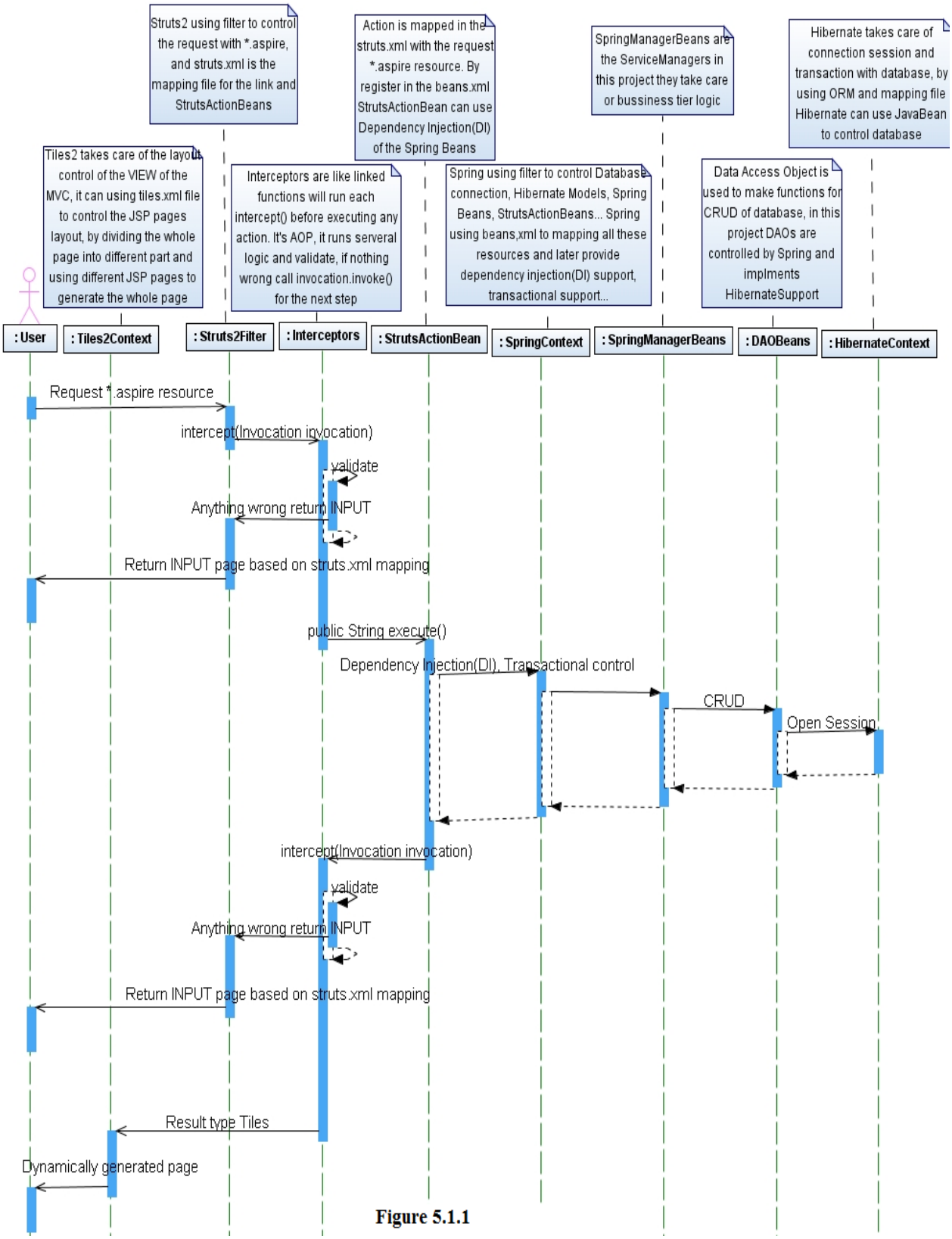


Figure 5.1.1

Figure 5.1.2 shows the package diagram for the Aspire Model Solution

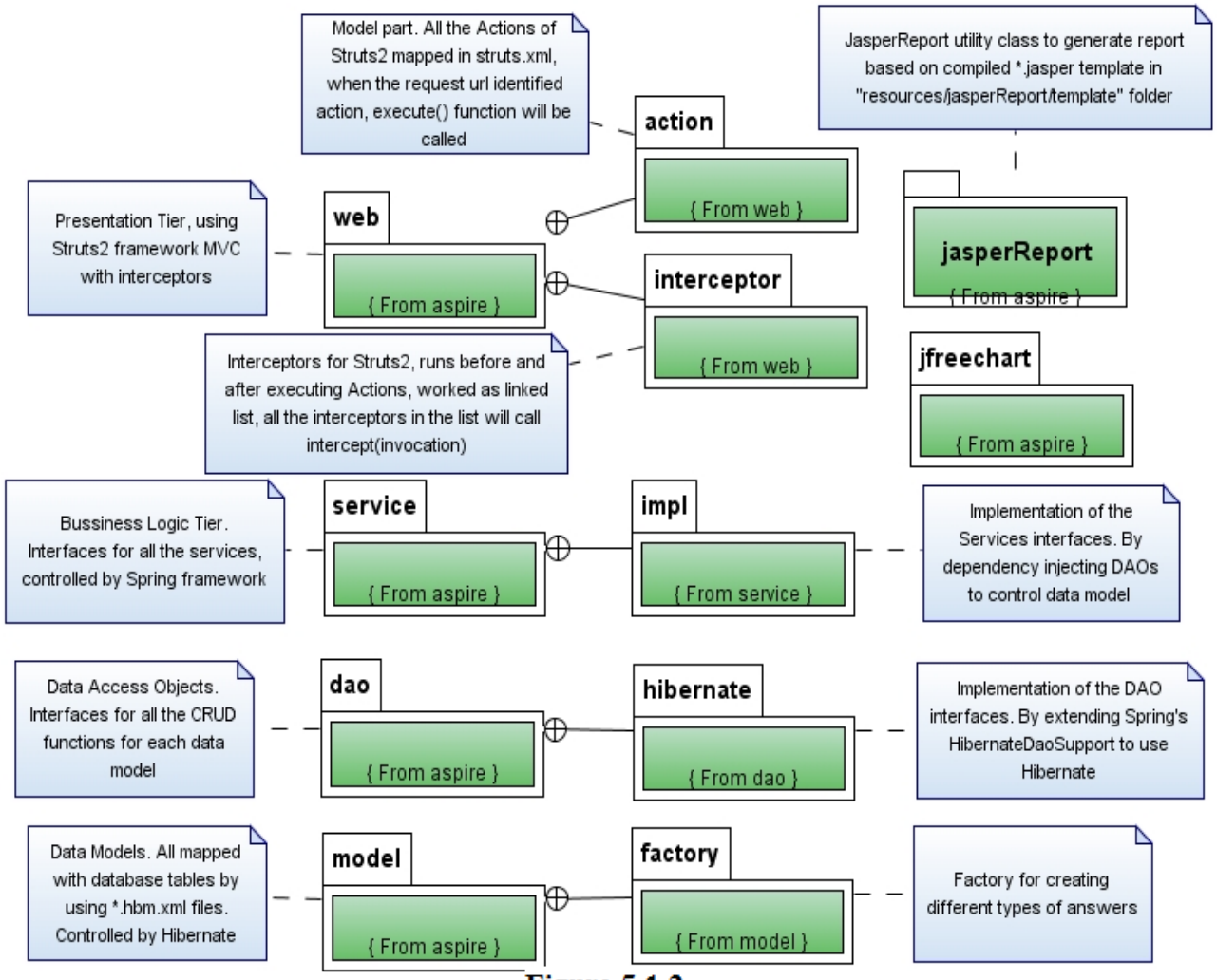


Figure 5.1.2

From technical point, this project is implemented based on Tiles2 Struts2 Spring and Hibernate frameworks, and using JasperReport iReport to generate report. And the structure of the whole work flow is based on Struts2's MVC (Model View Controller).

Struts2 is registered in the web.xml as a filter whenever user request url like *.aspire, Struts2 will be notified, and try to find the action class which is StrutsActionBean from the struts2.xml mapping file. There are a stack of interceptors works as linked AOP (Aspect Oriented Programming) functions, Struts2 will call the interceptor's intercept function one by one and do things like initial template SEC, check the priority of the login user, validate parameters… all the newly defined interceptors

have been put in the “src/com.vei.aspire.web.interceptor” folder and there are several default interceptors for Struts2 framework also. After the interceptors finish their jobs the StrutsActionBean class will call execute function or mapped function in struts.xml, it works as “Command Mode”, so all the logic for one action can put into the execute() or mapped function to take the responsible of the Model in MVC.

Spring is the factory for the whole project, and it takes control of the business logic tier, it’s a factory because Spring registered itself in web.xml as a filter also and filter all the request, and using beans.xml to map database connection inforamtion, Hibernate configuration, Hibernate mappings, Struts2 interceptors, Struts2 StrutsActionBeans, ServiceManagerBeans, DaoBeans, and provide them Dependency Injection, AOP, Transactional control and all kinds of services.

Most popular service is Dependency Injection DI for short. DI can be used for any resource registered in beans.xml, and by using annotation @Resource, for example:

@Resource MemberManager memberManager

“memberManager” has to be the same name of the bean registered in the beans.xml and in the class using DI, the set function for the resource field must be implemented to let the Spring to use Java Reflection call the set function to set the value for the “memberManager” field.

ServiceManagers are interfaces that provide all the needed business logic for almost each database table in the database, and it will use DI to call DaoManagers to do CRUD (Create Retrieve Update Delete) functions on the database tables, or use other resources like AspireModelReport. All the ServiceManagers interfaces are in the folder “src/com.vei.aspire.service”, and ServiceManagerBeans implement those interfaces are in the folder “src/com.vei.aspire.service.impl”.

DAOs (Data Access Objects) are created to take control of database tables for each

table. And the DAO in this project extends HibernateDaoSupport which is a support class of Spring, to decrease duplicate work on creating session, close session, database connection control and exception handling. DAO interfaces are in the folder "src/com.vei.aspire.dao", DAO implementations for Hibernate are in the folder "src/com.vei.aspire.hibernate"

Hibernate is the ORM framework to using JavaBeans and *.hbm.xml files to map the database table, and using the database connection information which is registered in Spring's beans.xml.

Tiles2 is the framework to create layout of JSP View, and it uses "/WEB-INF/tiles.xml" to control the layout.

Figure 5.1.3 shows the deployment diagram for Aspire Model Solution

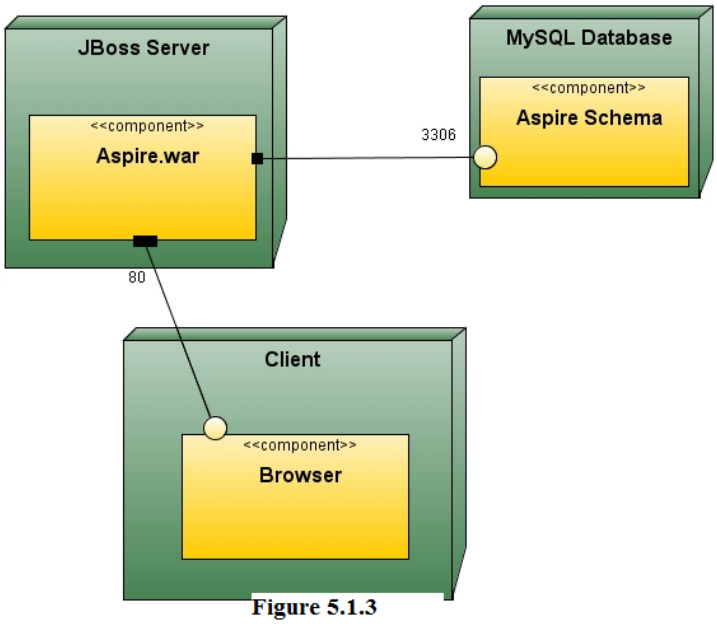


Figure 5.1.3

6. OTHER PROPERTIES

6.1. Technical

Aspire System is building with JSP Tiles2 Struts2 Spring2.5 Hibernate3.2, so these technologies are needed, and also they need a lot of XML Configuration files, so the combination of these frameworks and avoid XML hell is very important.

6.2. Deploy solution

To deploy the Aspire Model Solution we need at least JRE5, Tomcat 5.5 all kinds of operating systems.

6.3. Scheduling

The deadline is 30th April 2009, and we start at the end of January, for 2 week meetings, we have about 12 weeks to do the project. And by creating 2 versions of projects, the first version cost about 6 weeks. And second version using 2 weeks to implement core part of dynamic form, the rest 4 weeks implement SEC Report, administrator control, and super member control, progress bar and changing the whole GUI interface.

By the way, all the DAO and Spring Manager Beans have been tested using JUnit 4, and there are about 50 java classes for test now.

6.4. Platforms

For client need browser Firefox 2,3

For Server need Tomcat5.5,6 or JBoss 4.2.2 and Operating system can be anyone.

6.5. Portability

Portability is based on the server, and Aspire Model System will be deployed as a .war

file including all needed compiled class files, libraries, configuration files, and user interface resources. So if the new server can support Java5 J2EE1.4 and it's a JSP Servlet Container it can be used.

6.6. Expandability

The system is building using J2EE1.4 Tiles2 Struts2 Spirng2.5 Hibernate3.2 MySQL5.0, so if the system needs to expand, programmer need to have skills mentioned before.

6.7. Support & Maintenance

After deploying the Aspire Model System into a server, the Struts2 Spring Hibernate frameworks will give information about the system running, like, Hibernate can give SQL it execute the database, Spring will show the Spring beans status, and Struts2 will show the interceptor information. For maintenance all the Exceptions will be throws and finally showed on the browser page, so based on the error message the bug can be found and fixed.

6.8. Configuration Management

Using omega.cc.puv.fi, this is a CVS system of VAMK, and makes synchronizing using team control of MyEclipse or Eclipse.

7. CONCLUSION

Aspire solution has been implemented by 3 tiers design, and uses many interfaces to make the design more abstract and reusable.

Everything starts from the data model design, and by tasting the hard work in static design and dynamic design with better integration, the solution gained more flexibility. But it can not support enough validation for the data, so it is hard to be loose coupled. In some case the user interface is implemented highly coupled with database data, and consumes much condition and validation checking for input and output.

To support the dynamic design of data model, the data from database go throw DAO and Service Bean to the Struts2 Action, finally reach the JSP View page, in that view page there are a lot nested conditional checking and recursive looping implemented by Struts2 tags, the more flexibility we get the harder to generate the code, and it consumes more system circles.

Flexibility and loose couple for a system cannot be easily implemented at the same time, before going deep into the coding, the developer needs to balance which one is more important, and analyze from the data model. Take time with governance discussions about the requirement and relative merits, so that you can make sure you will do the right thing.

REFERENCES

Don Brown, Chad Davis, and Scott Stanlick. *Sturts2 in action:* Manning Publications (May 1, 2008)

Craig Walls and Ryan Breidenbach. *Spring in action (Programmer to Programmer)*: Manning Publications; 2 edition (August 16, 2007)

Christian Bauer, Gavin King. *Java Persistence with Hibernate*: Manning Publications; Revised edition (November 24, 2006)

Eric Pugh, Joseph D. Gradecki. *Professional Hibernate (Programmer to Programmer)*: Wrox; 1st edition (October 8, 2004)

Alan Shalloway, James R. Trott. *Design Patterns Explained A New Perspective on Object Oriented Design:* Addision-Wesley Professional; 1st edition (July 9, 2001)

Deepak Alur, Dan Malks, John Crupi. *Core J2EE Patterns: Best Practices and Design Strategies (Sun Core Series):* Prentice Hall PTR; 2nd Edition (May 10, 2003)

APPENDIX

Question Type

complex question with complex type can have answer with type “boldLabel”, “smallFont”, “textfield”, “textarea”, “datetimepicker”, “select”, “number”, “file” and “grid”. Super member can modify the values for all these answers.

menu question with menu type have no answer, when the web user interface request detail page for such question, the list of its children questions will be showed.

group question with group type have at least one answer with type group.

Answer type

boldLabel shows text with bold font style and big font size

smallFont shows text with small font size

textfield Text field for input type text

textarea Text area for input

datetimepicker “Dojo” Date time picker component for Date selecting

select Form select field, can have a list of select options

number One text field for number one select field for number unit

file File upload for form

gird Grid complex type, which looks like a grid table and for each cell

	there is one answer
group	Group complex type, which will use one page to show list of group units and part of their information, super member can have functions to add new group unit